
Blocs LIN

Réseau local instruments

Manuel de référence

© 1992-1997, 1999-2001 Eurotherm Automation TCS Systèmes. Tous droits réservés.

La récupération du présent document sur un serveur ou la transmission, même partielle, sous quelque forme que ce soit, sont soumises à l'autorisation préalable du détenteur des droits réservés. EUROTHERM AUTOMATION TCS SYSTEMES poursuit une politique de développement et d'amélioration continus de ses produits. Les spécifications du présent document peuvent donc être modifiées sans préavis. Les informations du présent document sont données en toute bonne foi, mais uniquement à titre d'information. La responsabilité de EUROTHERM AUTOMATION TCS SYSTEMES ne sera pas engagée en cas de pertes résultant d'erreurs dans le document.



Avril 2002

RECAPITULATIF DES VERSIONS DU PRESENT MANUEL

Chapitre	Version
Page de garde	8(F)
Table des matières	8(F)
Chapitre 1	8(F)
Chapitre 2	8(F)
Chapitre 3	8(F)
Chapitre 4	5(F)
Chapitre 5	7(F)
Chapitre 6	4(F)
Chapitre 7	5(F)
Chapitre 8	5(F)
Chapitre 9	5(F)
Chapitre 10	7(F)
Chapitre 11	3/B(F)
Chapitre 12	8(F)
Chapitre 13	3(F)
Chapitre 14	8(F)
Chapitre 15	8(F)
Chapitre 16	8(F)
Chapitre 17	8(F)
Chapitre 18	8(F)
Chapitre 19	7(F)
Chapitre 20	8(F)
Annexe A	8(F)
Annexe B	8(F)
Index	8(F)
Dépliant	3/B (F)

Notas

- 1 Les chapitres sont mis à jour séparément et les versions peuvent donc différer.
- 2 La page de titre et le manuel dans son intégralité prennent toujours le numéro de version du chapitre de la dernière réédition.
- 3 La version de certaines pages dans les chapitres du présent manuel peut être antérieure à celle des autres. C'est le cas, si ces pages ont été rééditées séparément et mises à niveau dans le manuel existant pour le mettre à jour — *politique suivie par Eurotherm Systèmes SA pour économiser le papier et protéger l'environnement*. Mais, le numéro de version de l'ensemble du chapitre — voir la liste de la table ci-dessus — est toujours le numéro de version de la dernière réédition de ou des pages de ce chapitre.

TCS Intelligent Instruments sont des produits de Eurotherm Systèmes SA.

TCS est une marque déposée de Eurotherm Systèmes SA.

Toutes les marques déposées et non-déposées sont les propriétés de leurs détenteurs respectifs.

Table des matières

MANUEL DE REFERENCE BLOCS LIN

Chapitre 1 INTRODUCTION

Description des blocs de fonction	1-1
Exécution des blocs de fonction dans les instruments	1-1
Symboles d'état dans les tables des paramètres des blocs	1-1

Chapitre 2 BLOCS DE FONCTION E/S

ANIN : Bloc d'entrée analogique	2-1
TCOUPLE : Bloc d'entrée thermocouple	2-1
RTD : Bloc d'entrée sonde à résistance	2-1
ANOP : Bloc de sortie analogique	2-9
DGIN_8 : Bloc d'entrée logique à 8 voies	2-13
DGOUT_8 : Bloc de sortie logique à 8 voies	2-17
FULL_TC8 : Bloc d'entrée thermocouple	2-21
FREQIN : Bloc d'entrée fréquence	2-26
AN_IP : Bloc d'entrée analogique	2-32
AN_OUT : Bloc de sortie analogique	2-42
DG_IN : Bloc d'entrée logique à 8 voies	2-46
DG_OUT : Bloc de sortie logique à 8 voies	2-50
DGPULS_4 : Bloc d'impulsions logiques à 4 voies	2-54
AI_CALIB : Bloc d'étalonnage des entrées logiques	2-62
AO_CALIB : Bloc d'étalonnage des sorties logiques	2-68
AN8_OUT : Bloc de sortie analogique à 8 voies	2-72
RTD_6 : Bloc d'entrée sonde à résistance	2-75
ANIN_6 : Bloc d'entrée analogique	2-79
DGIN_1 : Bloc d'entrée logique à voie unique	2-83

Chapitre 3 BLOCS DE FONCTION S6000

Description générale	3-1
Blocs série S6000 maîtres	3-1
S6360 : Bloc modèle de régulateur	3-4
S6366 : Bloc modèle de régulateur	3-8

6432 AI : Bloc modèle d'entrée analogique	3-13
6432 AO : Bloc modèle de sortie analogique	3-18
6432 DI : Bloc modèle d'entrée logique	3-22
6432 DO : Bloc modèle de sortie logique	3-26
GEN_COMM : Bloc de communication générique S6000	3-28
Blocs série SL6000 esclaves	3-32
SL6432AI : Bloc modèle d'entrée analogique	3-33
SL6432AO : Bloc modèle de sortie analogique	3-36
SL6432DI : Bloc modèle d'entrée logique	3-39
SL6432DO : Bloc modèle de sortie logique	3-42
SL0832 : Bloc modèle de pseudo-instrument 0832	3-45
SL6437 : Bloc modèle de pseudo-instrument 6437	3-47
SL6366 : Bloc modèle de régulateur 6366	3-49
SL6360 : Bloc modèle de régulateur 6360	3-55
SL6366EX : Bloc d'extension SL6366	3-62
CONN32AI: Bloc de liaison d'entrée analogique	3-65
CONN32AO: Bloc de liaison de sortie analogique	3-69
SL6366R : Bloc modèle de régulateur 6366	3-72
SL6360R : Bloc modèle de régulateur 6360	3-77

Chapitre 4 BLOCS DE FONCTION CONDITN

INVERT : Bloc d'inversion analogique	4-1
CHAR : Bloc de caractérisation	4-3
UCHAR : Bloc de caractérisation T100	4-5
FILTER : Bloc filtre	4-7
LEADLAG : Bloc avance -retard	4-9
AN_ALARM : Bloc d'alarme analogique	4-11
DIGALARM : Bloc d'alarme logique	4-14
LEAD_LAG : Bloc (filtre) avance -retard	4-16
RANGE : Bloc échelle	4-19
FLOWCOMP : Bloc de débit corrigé	4-21
GASCONC : Bloc de données de concentration du gaz naturel	4-24
AGA8DATA : Bloc de calcul AGA8	4-29

Chapitre 5 BLOCS DE FONCTION DE REGULATION

PID : Bloc de régulation	5-1
ANMS : Bloc station manuelle analogique	5-11
DGMS : Bloc station manuelle logique	5-13
SIM : Bloc de simulation	5-15
AN_CONN : Bloc de liaisons analogiques	5-18
DG_CONN : Bloc de liaisons logiques	5-20
TP_CONN : Bloc de liaisons des données tièdes	5-23
SETPOINT : Bloc consigne	5-25
3_TERM : Bloc PID incrémentiel	5-30
MAN_STAT : Bloc station manuelle	5-38
MODE : Bloc mode	5-43
PID_CONN : Bloc de liaison PID	5-48
PID_LINK: Bloc de chaînage PID	5-51

Chapitre 6 BLOCS DE FONCTION TEMPS

SEQ : Bloc séquence	6-1
SEQUE : Bloc d'extension de séquence	6-8
TOTAL : Bloc de totalisation	6-9
DTIME : Bloc temps mort	6-12
TIMER : Bloc horloge	6-15
TIMEDATE : Bloc événement heure/date	6-17
TPO : Bloc de sortie proportionnelle au temps	6-22
DELAY : Bloc retard	6-25
RATE_ALM : Bloc alarme de vitesse	6-28
RATE_LMT : Bloc limite de vitesse	6-32
TOT_CON: Bloc de liaisons de totalisation	6-35
TOTAL2: Bloc de totalisation	6-37

Chapitre 7 BLOCS DE FONCTION SELECTION

SELECT : Bloc sélecteur	7-1
SWITCH : Bloc commutation	7-3
ALC : Bloc de regroupement d'alarmes	7-5
PAGE : Bloc page	7-7
2OF3VOTE : Bloc meilleure moyenne	7-9
TAG : Bloc repère	7-12

Chapitre 8 BLOCS DE FONCTION LOGIQUE

PULSE : Bloc impulsions	8-1
AND4 : Bloc ET	8-4
OR4 : Bloc OU	8-4
XOR4 : Bloc OU exclusif	8-4
NOT : Bloc non	8-6
LATCH : Bloc verrouillage	8-7
COUNT : Bloc de comptage	8-9
COMPARE : Bloc comparaison	8-12

Chapitre 9 BLOCS DE FONCTION MATHS

ADD2 : Bloc addition	9-1
SUB2 : Bloc soustraction	9-1
MUL2 : Bloc multiplication	9-1
DIV2 : Bloc division	9-1
EXPR : Bloc expression	9-3
ACTION : Bloc gestion d'action	9-7
DIGACT: Bloc d'action logique	9-10

Chapitre 10 BLOCS DE FONCTION CONFIG

T100 : Bloc de configuration du T100	10-1
T1000 : Bloc de configuration du T1000	10-6
T231 : Bloc de configuration du T231	10-8
T600 : Bloc de configuration du T600	10-13
T221 : Bloc de configuration du T221	10-21
T102: Bloc de configuration du T102	10-23
T302: Bloc de configuration du T302	10-28
T932: Bloc de configuration du T932	10-33
T940: Bloc de configuration du T940	10-37
T800: Bloc de configuration du T800	10-43
T2900: Bloc de configuration du T2900	10-43

Chapitre 11 BLOCS DE FONCTION HIST

HIST : Bloc historique	11-1
------------------------------	------

Chapitre 12 BLOCS DE FONCTION DIAG

LRA : Bloc algorithme de redondance LIN	12-1
LIN_DIAG : Bloc de diagnostic LIN	12-10
UCTUNE : Bloc performances du T1000	12-12
MDTUNE : Bloc performances du T100	12-13
DB_DIAG : Bloc diagnostic de la base de données	12-15
LIN_DEXT : Bloc d'extension diagnostic haut niveau LIN	12-17
EDB_DIAG : Bloc diagnostic de la base de données externe	12-19
DTU_DIAG : Bloc de diagnostic DTU	12-21
T231TUNE : Bloc performances du T231	12-22
S6_DIAG : Bloc diagnostic S6000	12-24
ALINDIAG : Bloc diagnostic MAC/LCC ALIN	12-27
EDB_TBL : Bloc table de la base de données externe	12-28
T600TUNE : Bloc performances du T600	12-29
ROUTETABL : Bloc table d'acheminement	12-31
RTB_DIAG : Bloc diagnostic table d'acheminement	12-32
ISB_DIAG : Bloc performances du bus série interne	12-33
ISB_DEXT : Bloc d'extension de diagnostic ISB	12-34
T221TUNE : Bloc de mise au point T221	12-35
NET_DIAG : Bloc diagnostic réseau	12-36
FWD_LOG : Bloc consignation d'acheminement	12-37
CON_TBL : Bloc table des liaisons	12-38
CON_DIAG : Bloc diagnostic des liaisons	12-39
CON_ENT : Bloc entrée liaisons	12-40
POL_TBL : Bloc table d'interrogation	12-42
POL_DIAG : Bloc diagnostic des interrogations des installations	12-43
POL_ENT : Bloc entrée des interrogations des installations	12-45
NODE_MAP : Bloc protocole noeud LIN	12-46
FWD_DIAG : Bloc statistiques d'acheminement	12-47
BCS_DIAG : Bloc diagnostic de diffusion du routage	12-48
XEC_DIAG : Bloc diagnostic des tâches	12-49
SUM_DIAG : Bloc de diagnostic récapitulatif	12-50
TOD_DIAG : Bloc de diagnostic de l'heure du jour	12-54
RED_CTRL : Bloc de contrôle de redondance	12-56
SDX_RSRC : Bloc de ressources d'échange de données série	12-59
FT_TUNE : Bloc performances du T102	12-62

FTQ_DIAG: Bloc de diagnostic des files d'attente PRMT	12-64
ICM_DIAG: Bloc statistiques module communication interprocesseur .	12-66
MDBDIAG: Bloc de diagnostic Modbus	12-68
SFC_DIAG: Bloc de diagnostic des sequences	12-71
ISE_DIAG: Bloc de diagnostic du moteur de stratégie industrielle	12-72
PNL_DIAG: Bloc de diagnostic de la face avant	12-74
ALH_DIAG: Bloc de diagnostic de l'historique des alarmes	12-75
DDR_DIAG: Bloc de diagnostic fonction d'enregistrement des données	12-76
AMC_DIAG: Bloc de diagnostic communications maîtres applications	12-77
PS_TUNE: Bloc de performances T940	12-79
PMC_DIAG: Bloc de diagnostic de la ligne Profibus	12-81
AGA8DIAG: Bloc de diagnostic AGA8	12-84
ISE_TUNE: Bloc mise au point moteur de stratégie industrielle .	12-88

Chapitre 13 BLOCS DE FONCTION TAN

RS485_NODE : Bloc de configuration du noeud TAN	13-1
TX_AN : Bloc de transmission analogique	13-4
TX_DG : Bloc de transmission logique	13-6
RX_AN : Bloc de réception analogique	13-7
RX_DG : Bloc de réception logique	13-9
TAN_DIAG : Bloc de diagnostic TAN	13-10

Chapitre 14 BLOCS DE FONCTION LOT

SFC_CON : Bloc de commande SFC	14-1
SFC_MON : Bloc de supervision SFC	14-5
SFC_DISP : Bloc d'affichage SFC	14-7
RECORD : Bloc enregistrement	14-10
DISCREP : Bloc discordance	14-15
RCP_SET: Bloc ensemble de recettes	14-20
RCP_LINE: Bloc ligne de recette	14-22
BAT_CTRL: Bloc de contrôle des lots	14-24

Chapitre 15 BLOCS DE FONCTION DCM

Sous-catégories des blocs de fonction DCM	15-1
Utilisation des blocs de fonction E/S DCM	15-2
Fichiers de définition de profil	15-3
DX2_LOOP: Bloc boucle série 2000	15-6
DX2_TUNE: Bloc de mise au point des boucles série 2000	15-8
D25_LOOP: Bloc boucle 2500	15-10
D25eLOOP: Bloc boucle 2500	15-10
D25_TUNE: Bloc de mise au point des boucles 2500	15-13
D25eTUNE: Bloc de mise au point des boucles 2500	15-13
D25_RAMP: Bloc rampe pour programmeur de consignes	15-15
D25eRAMP: Bloc rampe pour programmeur de consignes	15-15
D25_MOD: Bloc module anonyme	15-17
D25_AI2: Bloc module entrée analogique à deux voies	15-19
D25_AI3: Bloc module entrée analogique à trois voies	15-19
D25_AI4: Bloc module entrée analogique à quatre voies	15-19
D25_AO2: Bloc module sortie analogique à deux voies	15-21
D25_AO4: Bloc module sortie analogique à quatre voies	15-21
D25_DI4: Bloc module entrée logique à quatre voies	15-23
D25_DI6: Bloc module entrée logique à six voies	15-23
D25_DI8: Bloc module entrée logique à huit voies	15-23
D25_DO4: Bloc module sortie logique à quatre voies	15-25
D25_AICH: Bloc entrée analogique	15-27
D25_AOCH: Bloc sortie analogique	15-30
D25_DICH: Bloc entrée logique	15-32
D25_DOCH: Bloc sortie logique	15-34
D25_R_CV: Bloc valeurs calculées réelles à liaisons filaires utilisateur	15-36
D25_B_CV: Bloc valeurs calculées booléennes à liaisons filaires utilisateur	15-37
D25_R_UV: Bloc valeurs réelles utilisateur	15-38
D25_UALM: Bloc alarme utilisateur 2500	15-39
DCM_I8: Bloc paramètre entier 16 bits avec signe à 8 voies	15-40
DCM_S8: Bloc paramètre entier 8 bits avec signe à 8 voies	15-40
DCM_UI8: Bloc paramètre entier 16 bits sans signe à 8 voies	15-40
DCM_US8: Bloc paramètre entier 8 bits sans signe à 8 voies	15-40
DCM_R8: Bloc paramètre nombre réel à 8 voies	15-40
DCM_T8: Bloc paramètre durée à 8 voies	15-40

DCM_W8: Bloc paramètre mot/sous-champ à 8 voies	15-40
DCM_Y8: Bloc paramètre octet hex 'AB' à 8 voies	15-40
DCM_B8: Bloc paramètre booléen à 8 voies	15-40
DCM_D8: Bloc paramètre double entier à 8 voies	15-40
D2500: Bloc instrument 2500	15-42
D2000: Bloc instrument série 2000	15-44
D25_AI: Bloc entrée analogique à voie unique	15-46
D25_AO: Bloc sortie analogique à voie unique	15-50
D25_DI: Bloc entrée logique à voie unique	15-53
D25_DO: Bloc sortie logique à voie unique	15-56
D25_AI_T: Bloc d'alarmes temporisées d'une entrée analogique à voie unique	15-59
D25_DI_T: Bloc d'alarmes temporisées d'une entrée logique à voie unique	15-65

Chapitre 16 BLOCS DE FONCTION ORGANISATION

AREA: Bloc Area	16-3
GROUP: Bloc Group	16-6
LOGDEV: Bloc unité de consignation	16-9
LGROUP: Bloc groupe de consignation	16-12
LOGGRPEX: Bloc d'extension de groupe de consignation	16-14
LPTDEV: Bloc imprimante	16-15
PGROUP: Bloc groupe imprimante	16-18

Chapitre 17 BLOCS DE FONCTION ENREGISTREUR

DR_ANCHP: Bloc point voie analogique d'enregistrement de données	17-2
DR_DGCHP: Bloc point voie logique d'enregistrement de données	17-5
DR_ALARM: Bloc point alarme	17-7
DR_REPRT: Bloc point Report	17-8

Chapitre 18 BLOCS DE FONCTION PROGRAMMEUR

SPP_CTRL: Bloc controle programmeur consignes	18-2
SPP_DIG: Bloc logique programmeur de points de consigne	18-4
SPP_RAMP: Bloc rampe locale programmeur consignes	18-6
SPP_EXT: Bloc extension programmeur consignes	18-7

Chapitre 19 BLOCS DE FONCTION CONVERSION

ENUMENUM: Bloc convertisseur énumératif à énumératif	19-1
UINTENUM: Bloc convertisseur entier à énumératif	19-2
ENUMUINT: Bloc convertisseur énumératif à entier	19-3

Chapitre 20 BLOCS DE FONCTION OPERATEUR

PNL_CMD: Bloc commande panneau	20-2
PNL_MSG: Bloc message panneau	20-7
PNL_DLG: Bloc boîte de dialogue panneau	20-12
PNL_ACC: Bloc accès panneau	20-17
READER: Bloc lecteur	20-19

Annexe A

MODES DE FONCTIONNEMENT DE LA BOUCLE DE REGULATION

Mode de régulation actif	A-1
Priorités du mode de régulation	A-1
Modes de reprise (Fallback)	A-1
Sélection des modes de régulation	A-2
Mode Déporté (Remote)	A-2
Algorithme de régulation PID à 3 termes	A-2
Equation de régulation analogique à 3 termes	A-2
Algorithme de régulation numérique	A-3
Equivalence entre les équations analogiques et numériques ...	A-4
Equilibrage de l'intégrale & désaturation de l'intégrale	A-5
Equilibrage de l'intégrale	A-5
Désaturation de l'intégrale	A-7
Mise en oeuvre mathématique	A-10

Annexe B LA COUCHE D'APPLICATION LIN

- 1 Configuration d'un bloc en mémoire cacheB-2
- 2 Mise en mémoire cache d'un bloc en conduiteB-3
- 3 Mise en mémoire cache d'un second blocB-4
- 4 Mise en mémoire cache d'un autre bloc dans la direction opposée B-5
- 5 Mise en mémoire cache de blocs entre trois noeudsB-6

INDEX

IND-1

Chapitre 1 INTRODUCTION

Description des blocs de fonction

Le présent manuel décrit tous les blocs de fonction disponibles pour tourner sur des instruments qui peuvent être connectés à un réseau local instruments (LIN) d'Eurotherm Systèmes. Le manuel décrit le but et le fonctionnement de chaque bloc, définit les paramètres des menus de spécification, indique les entrées et sorties, et fournit des indications détaillées sur ce qui est nécessaire pour configurer le bloc dans un schéma de boucles.

Les blocs sont regroupés en dix-sept catégories, un chapitre complet du manuel est consacré à chaque catégorie. Les onglets noirs au bord des pages, ainsi que le répertoire à onglets au début et à la fin du manuel permettent de repérer rapidement la catégorie en question. La *table des matières* ou *l'index* vous permet également de localiser rapidement un bloc particulier. La table 1-2 (voir page suivante) donne la liste de toutes les catégories et décrit brièvement chacun des blocs de fonction.









Exécution des blocs de fonction dans les instruments

Il faut noter que tous les blocs décrits dans le présent manuel ne peuvent pas être exécutés sur tous les instruments d'un réseau LIN. La table 1-2 indique le niveau de prise en charge par gamme d'instruments pour chaque bloc.

Consultez le *manuel produit* qui accompagne l'instrument en question et qui donne les détails sur tous les paramètres d'un bloc qui ne sont pas pris en compte, ou seulement partiellement, ou qui ont des fonctions particulières, lorsqu'ils sont utilisés sur cet instrument.

Symboles d'état dans les tables des paramètres des blocs

La table 1-1 donne la signification des symboles qui sont utilisés dans les tables des paramètres des blocs dans le présent manuel pour indiquer "l'état" du paramètre. Les symboles sont également imprimés sur la page à volet à la fin du manuel, qui comprend également un tableau de conversion binaire/hexadécimal/décimal.

Symbole	Signification
	Connectable comme sortie au schéma de boucles à partir du bloc
	Connectable comme entrée au bloc à partir du schéma de boucles
	Le paramètre est en lecture uniquement s'il est connecté
	Connectable en entrée et en sortie
	Lecture uniquement. Indications d'exceptions ou de cas particuliers le cas échéant
	Ecriture uniquement. (c'est à dire, revient automatiquement à FAUX après être passé à VRAI)
	Sur la liste d'inspection de la base de données partielle par défaut (touche 'INS' des instruments de la série T600)
	Champ(s) d'alarme

* Accès partiel peut être personnalisé par le logiciel LINtools T500. Voir manuel produit T500.

Table 1-1 Symboles d'état des paramètres des blocs

Catégorie	Bloc	Fonction
E/S (I/O)	AI_CALIB	Guide l'opérateur dans l'étalonnage des entrées analogiques
	AN8_OUT	Voies de sortie analogique, 8 voies
	ANIN	Voies d'entrée analogiques
	ANIN_6	Voies d'entrée analogiques, 6 voies
	ANOP	Voies de sortie analogiques
	AN_IP	Voies d'entrée analogiques
	AN_OUT	Voies de sortie analogiques
	AO_CALIB	Guide l'opérateur dans l'étalonnage des sorties analogiques
	DGIN_1	Voie d'entrée logique, unique
	DGIN_8	Voies d'entrée logiques, 8-voies
	DGOUT_8	Voies de sortie logiques, 8-voies
	DGPULS_4	Sorties impulsions (unique, train d'impulsions, impulsion double, sortie proportionnelle temps)
	DG_IN	Voies d'entrée logiques
	DG_OUT	Voies de sortie logiques
	FREQIN	Entrée fréquence
	FULL_TC8	Entrée thermocouple
	RTD	Entrées analogiques de lecture de température
	RTD_6	Entrées analogiques de lecture de température, 6 voies
	TCOUPLE	Entrées analogiques de lecture de température
S6000	6432 AI	Modèle d'entrée analogique
	6432 AO	Modèle de sortie analogique
	6432 DI	Modèle d'entrée logique
	6432 DO	Modèle de sortie logique
	CONN32AI	Liaison d'entrée analogique
	CONN32AO	Liaison de sortie analogique
	GEN_COMM	Modèle générique
	S6360	Modèle régulateur
	S6366	Modèle régulateur
	SL0832	Modèle régulateur
	SL6360	Modèle régulateur
	SL6360R	Modèle régulateur
	SL6366	Modèle régulateur
	SL6366EX	Modèle régulateur (ext.)
	SL6366R	Modèle régulateur
	SL6432 AI	Modèle d'entrée analogique
	SL6432 AO	Modèle de sortie analogique
	SL6432 DI	Modèle d'entrée logique
	SL6432 DO	Modèle de sortie logique
	SL6437	Modèle régulateur
CONDITN	AGA8DATA	Calculs rapport n° 8 Association américaine du gaz
	AN_ALARM	Bloc alarme avec alarmes absolues/écart/vitesse
	CHAR	Caractériser analogique 16 points
	DIGALARM	Bloc alarme logique
	FILTER	Filtre du premier ordre
	FLOWCOMP	Calcul du débit, correction de pression, température et densité
	GASCONC	Calcul du débit, corrigé en pression, température et densité

Table 1-2 suite...

Block	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
AI_CALIB						✓					
AN8_OUT	5/1 ✓	✓	3/1 ✓		5/1 X					X	
ANIN	✓		3/1 ✓		X		X			X	
ANIN_6		4/1 ✓	3/1 ✓							X	
ANOP	✓	✓	3/1 ✓		X		X			X	
AN_IP						✓					
AN_OUT						✓					
AO_CALIB						✓					
DGIN_1		✓								X	
DGIN_8	✓	✓	3/1 ✓		X		X			X	
DGOUT_8	✓	✓	3/1 ✓		X		X			X	
DGPULS_4						✓					
DG_IN						✓					
DG_OUT						✓					
FREQIN	2/1 ✓	✓	3/1 ✓		X		X			X	
FULL_TC8	2/1 ✓	✓	3/1 ✓		X		X			X	
RTD	✓	✓	3/1 ✓		X		X			X	
RTD_6		4/1 ✓	3/1 ✓							X	
TCOUPLE	✓	✓	3/1 ✓		X		X			X	
6432 AI			✓		✓		✓		X		
6432 AO			✓		✓		✓		X		
6432 DI			✓		✓		✓		X		
6432 DO			✓		✓		✓		X		
CONN32AI			✓								
CONN32AO			✓								
GEN_COMM			✓		✓		3/2 ✓				
S6360			✓		✓		✓	X	X		
S6366			✓		✓		✓	X	X		
SL0832			3/1 ✓			2/1 ✓					
SL6360						✓					
SL6360R			3/1 ✓								
SL6366						✓					
SL6366EX			3/1 ✓			3/3 ✓					
SL6366R			3/1 ✓								
SL6432 AI			3/1 ✓			✓					
SL6432 AO			3/1 ✓			✓					
SL6432 DI			3/1 ✓			✓					
SL6432 DO			3/1 ✓			✓					
SL6437			3/1 ✓			2/1 ✓					
AGA8DATA		4/1 A	3/1 A			4/1 A				✓	✓
AN_ALARM	✓	✓	✓		✓	✓	✓			✓	✓
CHAR	✓	✓	✓		✓	✓	✓			✓	✓
DIGALARM	✓	✓	✓		✓	✓	✓			✓	✓
FILTER	✓	✓	✓		✓	2/1 ✓	✓			✓	✓
FLOWCOMP	5/1 ✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
GASCONC		4/1 A	3/1 A			4/1 A				✓	✓

Table 1-2 suite...

...Table 1-2 suite

Catégorie	Bloc	Fonction
...CONDITN	INVERT	Inverse le signal autour des limites HR, LR
	LEADLAG	Bloc avance-retard
	LEAD_LAG	Bloc avance-retard
	RANGE	Modification d'échelle d'une entrée analogique
	UCHAR	Caractériser 16 points pour les blocs d'entrée analogique
REGULATION (CONTROL)	3_TERM	Forme incrémentielle du bloc PID
	ANMS	Station manuelle analogique
	AN_CONN	Bloc liaisons analogiques
	DGMS	Station manuelle logique
	DG_CONN	Bloc liaisons logiques
	MAN_STAT	Station manuelle, avec liaisons aux affichages de la face avant
	MODE	Sélection du mode de régulation, avec masquage des boutons-poussoirs
	PID	Fonction de régulation PID
	PID_CONN	'Face avant' pour combinaison SETPOINT/3_TERM/MAN_STAT/MODE
	PID_LINK	'Face avant' pour combinaison SETPOINT/3_TERM/MAN_STAT/MODE
	SETPOINT	Génération d'une consigne avec décalages, seuils et alarmes
TEMPS (TIMING)	SIM	Bloc simulation, 2 x premier ordre ou capacité, et bruit
	TP_CONN	Spécifie 9 champs maxi comme données tièdes EEPROM à la mise hors tension
	DELAY	Retard pour les applications temps mort
	DTIME	Bloc retard pour les applications temps mort
	RATE_ALM	Alarme vitesse haute & basse appliquée à PV, OP étant maintenu à la dernière valeur de non-alarme
	RATE_LMT	Limiteur de vitesse et générateur de rampe
	SEQ	Bloc rampe/palier/temps multi-segments, 15 sorties logiques
	SEQE	Bloc extension SEQ
	TIMEDATE	Bloc événement horloge et calendrier
	TIMER	Horloge
	TOTAL	Totalisateur (intégrateur) pour variable analogique
	TOTAL2	Totalisateur (intégrateur) pour variable analogique, avec mode étalonnage
	TOT_CON	Réserve pour les valeurs de paramètres à 32 bits (ex. totalisations)
	TPO	Sortie proportionnelle au temps
SELECTION (SELECTOR)	2OF3VOTE	Sélection de la meilleure entrée sur 3, et moyenne des entrées dans les tolérances
	ALC	Bloc collecte alarme produisant une sortie logique commune
	PAGE	Sélectionne jusqu'à 8 pages d'affichage par la fonction page en attente
	SELECT	Choisit la valeur la plus haute, la plus basse ou médiane parmi 2, 3 ou 4 entrées
	SWITCH	Commutateur unipolaire à deux directions pour les signaux analogiques
LOGIQUE (LOGIC)	TAG	Spécifie un nom de repère de tâche utilisateur (boucle) sélectionné dans une liste de 8 repères
	AND4	Fonction booléenne ET à 4 entrées
	COMPARE	Indique plus grand que/plus petit que/égal à 2 entrées
	COUNT	Compteur/décompteur d'impulsions avec cible comptage DEBUT/FIN
	LATCH	Fonction bascule de type D
	NOT	Fonction non-booléenne
	OR4	Fonction booléenne OU à 4 entrées
	PULSE	Sortie impulsion, fonction (monostable)
	XOR4	Fonction booléenne OU exclusif à 4 entrées

Table 1-2 suite...

...Table 1-2 suite

Bloc	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
INVERT	✓	✓	✓		✓		✓			✓	✓
LEADLAG	✓	✓	✓		✓		✓			✓	✓
LEAD_LAG	5/1 ✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
RANGE	5/1 ✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
UCHAR	✓	✓	✓		✓	✓	✓			✓	✓
3_TERM		✓	✓			✓				✓	✓
ANMS	✓	✓	✓		✓	✓	✓	X	X	✓	✓
AN_CONN	✓	✓	✓		✓	✓	✓	X	X	✓	✓
DGMS	✓	✓	✓		✓	✓	✓	X	X	✓	✓
DG_CONN	✓	✓	✓		✓	✓	✓	X	X	✓	✓
MAN_STAT		✓	✓			✓				✓	✓
MODE		✓	✓			✓				✓	✓
PID	✓	✓	✓		✓	2/1 ✓	✓	X	X	✓	✓
PID_CONN						✓					✓
PID_LINK		✓†	✓							✓	
SETPOINT		✓	✓			✓				✓	✓
SIM	✓	✓	✓		✓	2/1 ✓	✓			✓	✓
TP_CONN						2/1 ✓				✓	✓
DELAY	✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
DTIME	✓	✓	✓		✓		✓			✓	
RATE_ALM	✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
RATE_LMT	✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
SEQ	✓	✓	✓		✓	✓	✓			✓	✓
SEQE	✓	✓	✓		✓	✓	✓			✓	✓
TIMEDATE	✓	✓	✓		✓	2/1 ✓	✓			✓	✓
TIMER	3/2 ✓	✓	✓		✓		3/2 ✓			✓	✓
TOTAL	✓	✓	✓		✓	✓	✓			✓	✓
TOTAL2		4/1 ✓	3/1 ✓			4/1 ✓				✓	✓
TOT_CON		2/1 ✓				4/1 ✓				✓	
TPO	3/2 ✓	✓	✓		✓	4/1 ✓	3/2 ✓			✓	✓
2OF3VOTE	✓	✓	✓		5/1 ✓	✓	5/1 ✓			✓	✓
ALC	✓	✓	✓		✓	✓	✓			✓	✓
PAGE							✓				
SELECT	✓	✓	✓		✓	✓	✓			✓	✓
SWITCH	✓	✓	✓		✓	✓	✓			✓	✓
TAG						✓					
AND4	✓	✓	✓		✓	✓	✓			✓	✓
COMPARE	✓	✓	✓		✓	✓	✓			✓	✓
COUNT	✓	✓	✓		✓	✓	✓			✓	✓
LATCH	✓	✓	✓		✓	✓	✓			✓	✓
NOT	✓	✓	✓		✓	✓	✓			✓	✓
OR4	✓	✓	✓		✓	✓	✓			✓	✓
PULSE	✓	✓	✓		✓	✓	✓			✓	✓
XOR4	✓	✓	✓		✓	✓	✓			✓	✓

Table 1-2 suite...

...Table 1-2 suite

Catégorie	Bloc	Fonction
MATHS	ACTION	Gestion d'action, en utilisant des variables enregistrées et temps écoulé
	ADD2	Additionne 2 signaux
	DIGACT	Gestion d'action, et utilisation de variables logiques enregistrées et temps écoulé
	DIV2	Divise 2 signaux
	EXPR	Expression math. format libre avec un maximum de 4 entrées
	MUL2	Multiplie 2 signaux
	SUB2	Soustrait 2 signaux
CONFIG	T100	Bloc système
	T1000	Bloc système
	T102	Bloc système
	T221	Bloc système
	T231	Bloc système
	T302	Bloc système
	T600	Bloc système
	T932	Bloc système
	T800/T2900	Bloc système
	T940	Bloc système
HIST	HIST	Collecte et sauvegarde des données pour les tendances historiques à long terme
DIAG	AGA8DIAG	Diagnostic de calcul AGA8
	ALH_DIAG	Diagnostic de l'historique des alarmes T800/T2900
	ALINDIAG	Diagnostic du réseau des instruments locaux Arcnet (ALIN)
	AMC_DIAG	Diagnostic des communications maître des applications
	BCS_DIAG	Diagnostic de diffusion de routage
	CON_DIAG	Diagnostic liaisons
	CON_ENT	Journal de consignation des transmissions
	CON_TBL	Table liaisons
	DB_DIAG	Informations sur les ressources de la base de données
	DDR_DIAG	Diagnostic de l'unité d'enregistrement des données T800/T2900
	DTU_DIAG	Diagnostic de l'unité de transfert des données (DTU)
	EDB_DIAG	Informations sur la base de données déportée externe
	EDB_TBL	Diagnostic base de données externe
	FTQ_DIAG	Statistiques et état des files d'attente PRMT
	FT_TUNE	Statistiques des performances
	FWD_DIAG	Statistiques d'acheminement
	FWD_LOG	Journal de consignation
	ICM_DIAG	Diagnostic ICM
	ISB_DEXT	Statistiques performances haut niveau ISB (bus série interne)
	ISB_DIAG	Diagnostic ISB (bus série interne)
	ISE_DIAG	Diagnostic (T800/T2900) du moteur de stratégie industrielle
	LIN_DEXT	Statistiques des performances haut niveau LIN
	LIN_DIAG	Diagnostic du réseau local instrument (LIN)
	LRA	Contrôle les modes de fonctionnement de l'algorithme de redondance LIN
	MDBDIAG	Table de diagnostic Modbus
	MDTUNE	Statistiques des performances
	NET_DIAG	Diagnostic réseau
	NODE_MAP	Protocole des noeuds LIN

Table 1-2 suite...

...Table 1-2 suite

Bloc	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
ACTION	5/1 ✓	✓	✓		5/1 ✓	2/1 ✓	5/1 ✓			✓	✓
ADD2	✓	✓	✓		✓	✓	✓			✓	✓
DIGACT	✓	✓	✓		✓	✓	5/1 ✓			✓	✓
DIV2	✓	✓	✓		✓	✓	✓			✓	✓
EXPR	✓	✓	✓		✓	✓	✓			✓	✓
MUL2	✓	✓	✓		✓	✓	✓			✓	✓
SUB2	✓	✓	✓		✓	✓	✓			✓	✓
T100	✓	✓	✓		✓		✓	P		✓	
T1000							✓				
T102		✓								✓	
T221				✓							
T231			✓		✓		5/1 ✓				
T302		✓								✓	
T600						✓					
T932			✓								
T800/T2900											✓
T940										✓	
HIST							✓				
AGA8DIAG		4/1 A	3/1 A			4/1 A				✓	2/4 ✓
ALH_DIAG											✓
ALINDIAG		✓	✓	✓	✓	2/1 ✓				✓	
AMC_DIAG										✓	✓
BCS_DIAG				✓							
CON_DIAG				✓							
CON_ENT				✓							
CON_TBL				✓							
DB_DIAG	✓	✓	✓	✓	✓	✓	3/1 ✓	P	✓	✓	✓
DDR_DIAG											✓
DTU_DIAG							3/1 ✓				
EDB_DIAG	✓	✓	✓	✓	✓	2/1 ✓	3/1 ✓	P	✓	✓	
EDB_TBL	✓	✓	✓	✓	5/1 ✓	2/1 ✓	5/1 ✓	5/1 P		✓	
FTQ_DIAG		✓								✓	
FT_TUNE		✓								✓	
FWD_DIAG				✓							
FWD_LOG				✓							
ICM_DIAG		✓								✓	
ISB_DEXT						2/1 ✓					
ISB_DIAG						2/1 ✓					
ISE_DIAG											✓
ISE_TUNE											2/4 ✓
LIN_DEXT	✓	✓	✓	✓	✓	2/1 ✓	3/1 ✓	P	✓	✓	
LIN_DIAG	✓	X	✓	✓	✓		3/1 ✓	P	✓		
LRA	3/1 ✓		✓	✓	✓		3/1 ✓	P	✓		
MDTUNE	✓	X	✓		✓		X	P		✓	
MDBDIAG		2/1 ✓				4/1 ✓				✓	
NET_DIAG				✓						✓	
NODE_MAP	✓		✓	✓	5/1 ✓		5/1 ✓	5/1 P			

Table 1-2 suite...

...Table 1-2 suite

Catégorie	Bloc	Fonction
DIAG	PMC_DIAG	Diagnostic des lignes Profibus
	PNL_DIAG	Diagnostic des faces avant T800/T2900
	POL_DIAG	Diagnostic des interrogations des installations
	POL_ENT	Entrée des interrogations des installations
	POL_TBL	Table d'interrogation
	PS_TUNE	Performances T940
	RED_CTRL	Bloc de contrôle de la redondance
	ROUTETBL	Table de routage
	RTB_DIAG	Diagnostic table de routage
	S6_DIAG	Configuration/diagnostic du gestionnaire communications asynchrones binaires S6000
	SDX_RSRC	Etat interface entrées/sorties
	SFC_DIAG	Diagnostic et statistiques séquences (ressources utilisées et disponibles)
	SUM_DIAG	Diagnostic récapitulatif
	T221TUNE	Mise au point
	T231TUNE	Statistiques des performances
	T600TUNE	Statistiques des performances
	TOD_DIAG	Diagnostic et contrôle heure du jour
	UCTUNE	Statistiques des performances
	XEC_DIAG	Diagnostic des tâches
TAN	RS485_NODE	Configuration du réseau asynchrone TCS (TAN)
	RX_AN	Bloc de réception analogique
	RX_DG	Bloc de réception analogique et logique
	TAN_DIAG	Diagnostic TAN
	TX_AN	Bloc d'émission analogique
	TX_DG	Bloc d'émission logique
BATCH	DISCREP	Correspondance signaux logiques transmis/reçus pour diagnostic défauts installations
	RCP_LINE	Contrôle le téléchargement de recettes d'un fichier .UYR des blocs RCP_SET
	RCP_SET	Contrôle le jeux de recettes (le fichier à utiliser) pour certaines chaines de recettes
	RECORD	Sauvegarde/extraction de valeurs analogiques/logiques pour utilisation conduite
	SFC_CON	Contrôle, sélection et exécution séquence (SFC)
	SFC_DISP	Affichage/surveillance/contrôle d'une séquence exécutée à distance (SFC)
	SFC_MON	Contrôle conduite séquence (SFC)
DCM ...	D2X_LOOP	Accès boucles de régulation PID dans les 2200/2400/2500
	D2X_TUNE	Mise au point boucles PID dans les 2200/2400/2500
	D25_LOOP	Accès boucles de régulation PID dans le 2500C
	D25eLOOP	Accès boucles de régulation PID dans le 2500E
	D25_TUNE	Mise au point boucles PID dans le 2500C
	D25eTUNE	Mise au point boucles PID dans le 2500E
	D25_RAMP	Consigne déportée de la rampe dans le 2500C
	D25eRAMP	Consigne déportée de la rampe dans le 2500E
	D25_MOD	Accès unique au module physique E/S dans le 2500
	D25_AI2	Accès au module d'entrée analogique 2 voies dans le 2500
	D25_AI3	Accès au module d'entrée analogique 3 voies dans le 2500
	D25_AI4	Accès au module d'entrée analogique 4 voies dans le 2500
	D25_AO2	Accès au module de sortie analogique 2 voies dans le 2500
	D25_AO4	Accès au module de sortie analogique 4 voies dans le 2500

Table 1-2 suite...

...Table I-2 suite

Block	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
PMC_DIAG										1/4 ✓	2/7 ✓
PNL_DIAG											✓
POL_DIAG				✓							
POL_ENT				✓							
POL_TBL				✓							
PS_TUNE										1/3 ✓	
RED_CTRL		✓								✓	
ROUTETBL	✓	✓	✓	✓	5/1 ✓	2/1 ✓	5/1 ✓	5/1 P		✓	
RSRCDIAG		3/1 ✓	3/1 ✓	2/1 ✓		4/1 ✓					
RTB_DIAG	✓	✓	✓	✓	5/1 ✓	2/1 ✓	5/1 ✓	5/1 P		✓	
S6_DIAG			✓		✓		5/1 ✓				
SDX_RSRC		✓								✓	
SFC_DIAG		2/1 ✓	✓			4/1 ✓				✓	
SUM_DIAG	✓		✓	✓	5/1 ✓		5/1 ✓	5/1 P		✓	
T221TUNE				✓							
T231TUNE			✓		✓		5/1 ✓				
T600TUNE						✓					
TOD_DIAG		2/1 ✓	✓	✓		✓	5/1 ✓			✓	
UCTUNE							3/1 ✓				
XEC_DIAG	✓	✓	✓	✓	5/1 ✓	2/1 ✓	5/1 ✓	5/1 P			✓
RS485_NODE							2/1 ✓	} Non géré après v3			
RX_AN							2/1 ✓				
RX_DG							2/1 ✓				
TAN_DIAG							2/1 ✓				
TX_AN							2/1 ✓				
TX_DG							2/1 ✓				
BAT_CTRL										✓	3.0 ✓
DISCREP	3/1 ✓	✓	✓		✓	2/1 ✓	3/1 ✓			✓	✓
RCP_LINE											2/7 ✓
RCP_SET											2/7 ✓
RECORD	3/1 ✓	✓	✓		✓	2/1 ✓	3/1 ✓		X	✓	✓
SFC_CON	3/1 S	S	S		S	2/1 S	3/1 S		X	✓	✓
SFC_DISP	3/1 S	S	S		S	2/1 S	3/1 S			✓	✓
SFC_MON	3/1 S	S	S		S	2/1 S	3/1 S		X	✓	✓
D2X_LOOP										✓	✓
D2X_TUNE										✓	✓
D25_LOOP										✓	✓
D25eLOOP										1/4 ✓	2.6 ✓
D25_TUNE										✓	✓
D25eTUNE										1/4 ✓	2.6 ✓
D25_RAMP										✓	✓
D25eRAMP										1/4 ✓	2.6 ✓
D25_MOD										✓	✓
D25_AI2										✓	✓
D25_AI3										✓	✓
D25_AI4										✓	2/7 ✓
D25_AO2										✓	✓
D25_AO4										✓	✓

Table I-2 suite...

...Table 1-2 suite

Catégorie	Bloc	Fonction
...DCM	D25_DI4	Accès au module d'entrée logique à 4 voies dans le 2500
	D25_DI6	Accès au module d'entrée logique à 6 voies dans le 2500
	D25_DI8	Accès au module d'entrée logique à 8 voies dans le 2500
	D25_DO4	Accès au module de sortie logique à 4 voies dans le 2500
	D25_AI4CH	Accès à la voie d'entrée analogique unique dans le 2500
	D25_AO4CH	Accès à la voie de sortie analogique unique dans le 2500
	D25_DICH	Accès à la voie d'entrée logique unique dans le 2500
	D25_DOCH	Accès à la voie de sortie logique unique dans le 2500
	D25_R_CV	Accès à un maximum de 8 valeurs réelles calculées de liaisons filaires dans le 2500
	D25_B_CV	Accès à un maximum de 8 valeurs booléennes calculées de liaisons filaires dans le 2500
	D25_R_UV	Accès à 8 valeurs utilisateur réelles dans le 2500
	D25_UALM	Accès aux alarmes utilisateur analogiques ou logiques dans le 2500
	DCM_I8	Accès à un maximum de 8 paramètres entiers (16 bits) dans le 2500
	DCM_UI8	Accès à un maximum de 8 paramètres entiers sans signe (16 bits) dans le 2500
	DCM_US8	Accès à un maximum de 8 paramètres entiers sans signe (8 bits) dans le 2500
	DCM_R8	Accès à un maximum de 8 paramètres à nombre réel (32 -bits) dans le 2500
	DCM_W8	Accès à 8 paramètres à mot hexadécimal 'ABCD' (16 bits) maximum dans le 2500
	DCM_Y8	Accès à un maximum de 8 paramètres à octet hexadécimal 'AB' (8 bits) dans le 2500
	DCM_B8	Accès à un maximum de 8 paramètres booléens dans le 2500
	DCM_D8	Accès à 8 paramètres entiers à double précision (32 bits) maximum dans le 2500
	DCM_S8	Accès à 8 paramètres entiers avec signe (8 bits) dans le 2500
	DCM_T8	Accès à un maximum de 8 paramètres de durée (32 bits) dans le 2500
	D2500	Fournit une vue globale d'un instrument 2500 déporté
	D2000	Fournit une vue globale d'un instrument 2200 ou 2400 déporté
	D25_AI	Accès au module d'entrée analogique à voie unique dans le 2500
	D25_AI_T	Accès au module d'entrée analogique à voie unique dans le 2500 - alarmes temporisées
	D25_AO	Accès au module de sortie analogique à voie unique dans le 2500
	D25_DI	Accès au module d'entrée logique à voie unique dans le 2500
	D25_DI_T	Accès au module d'entrée logique à voie unique dans le 2500 - alarmes temporisées
	D25_DO	Accès au module de sortie logique à voie unique dans le 2500
ORGANISATIONAREA		Associe des blocs GROUP à une zone
	GROUP	Associe les voies d'affichage et d'enregistrement des T800/T2900 à un groupe
	LGROUP	Collecte des données de blocs point pour les archiver
	LOGDEV	Définit et contrôle l'accès à un support d'archivage
	LOGGRPEX	Elargit le nombre de points consignés par le bloc LGROUP block
	LPDEV	Définit et contrôle l'accès à une imprimante
ENREGISTREUR	PGROUP	Collecte les données de blocs points pour les imprimer
	DR_ALARM	Filtrage des alarmes au cours de l'enregistrement pour la consignation dans des fichiers et/ou sur des imprimantes
	DR_ANCHP	Bloc point à voie analogique d'enregistrement des données
	DR_DGCHP	Bloc point à voie logique d'enregistrement des données
	DR_REPRT	Génération de rapports texte définis par l'utilisateur envoyés sur une imprimante
PROGRAMMEUR	SPP_CTRL	Supervision, planification et contrôle du programmeur de consignes (T800/T2900)
	SPP_DIG	Etablit des liaisons filaires de consignes logiques à partir du programmeur de consignes (T800/T2900)
	SPP_EXT	Contrôle supplémentaire de l'exécution du programme de consignes (T800/T2900)
	SPP_RAMP	Permet la mise en rampe locale de consignes dans le programmeur de consignes (T800/T2900)

Table 1-2 suite...

...Table 1-2 suite

Bloc	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
D25_DI4										✓	✓
D25_DI6										✓	2/7 ✓
D25_DI8										✓	✓
D25_DO4										✓	✓
D25_AICH										✓	✓
D25_AOCH										✓	✓
D25_DICH										✓	✓
D25_DOCH										✓	✓
D25_R_CV										✓	✓
D25_B_CV										✓	✓
D25_R_UV										✓	✓
D25_UALM										✓	✓
DCM_I8										✓	✓
DCM_UI8										✓	✓
DCM_US8										1/4 ✓	2.3 ✓
DCM_R8										✓	✓
DCM_W8										✓	✓
DCM_Y8										1/4 ✓	2.3 ✓
DCM_B8										✓	✓
DCM_D8										✓	✓
DCM_S8										1/4 ✓	2.3 ✓
DCM_T8										1/4 ✓	2.3 ✓
D2500										✓	✓
D2000										✓	✓
D25_AI										✓	✓
D25_AI_T										2.1 ✓	3.1 ✓
D25_AO										✓	✓
D25_DI										✓	✓
D25_DI_T										2.1 ✓	3.1 ✓
D25_DO										✓	✓
AREA											✓
GROUP											✓
LGROUP											✓
LOGDEV											✓
LOGGRPEX											2/7 ✓
LPTDEV											2/7 ✓
PGROUP											2/7 ✓
DR_ALARM											2/7 ✓
DR_ANCHP											✓
DR_DGCHP											✓
DR_REPRT											2/7 ✓
SPP_CTRL											✓
SPP_DIG											✓
SPP_EXT											2/7 ✓
SPP_RAMP											2/7 ✓

Table 1-2 suite...

...Table 1-2 suite

Catégorie	Bloc	Fonction
CONVERSION	ENUMENUM	Convertit une valeur énumérée d'une énumération à l'autre
	UINTENUM	Convertit un entier en une énumération
	ENUMUINT	Convertit une énumération en un entier
OPERATEUR	PNL_ACC	Bloc accès panneau - valide & contrôle les accès des utilisateurs (Visual Supervisor)
	PNL_CMD	Bloc contrôle panneau - prise de contrôle du panneau (Visual Supervisor)
	PNL_DLG	Bloc boîte de dialogue panneau - création de boîtes de dialogue (Visual Supervisor)
	PNL_MSG READER	Bloc message panneau Création de messages (Visual Supervisor) Bloc lecteur - utilisation de lecteurs de codes à barres, etc. (Visual Supervisor)

...Table 1-2 suite

Bloc	T100 ^[1]	T920/1	T932	T221	T231	T640	T1000	T2001 ^[2]	T4000 ^[2]	T940	T800/2900
ENUMENUM											2/7 ✓
UINTENUM											2/7 ✓
ENUMUINT											2/7 ✓
PNL_ACC											3.0 ✓
PNL_CMD											3.0 ✓
PNL_DLG											3.0 ✓
PNL_MSG											3.0 ✓
READER											3.0 ✓

✓ = pleinement pris en charge; S = pleinement pris en charge dans la version Séquence uniquement; P = prise en charge locale sur carte PCLIN

A = prise en charge dans l'option "Advanced" uniquement.

Numéro de révision (par ex. 2/1) indique la version du logiciel où le bloc est apparu pour la première fois.

X = prise en charge partielle par le matériel — peut être utilisé en mode simulation ou cache uniquement;

^[1]Et T241. ^[2]Peut avoir des vues de modèles particulières différentes de celles des instruments LIN courants - voir les manuels produits.

†Prise en charge uniquement sur les T920 v3/1 et ultérieures & T921 v2/1 et ultérieures.

Table 1-2 Catégories des blocs de fonction & prise en charge instruments

[Page laissée intentionnellement blanche]

Chapitre 2 BLOCS DE FONCTION E/S

ANIN : BLOC D'ENTREE ANALOGIQUE

TCOUPLE : BLOC D'ENTREE THERMOCOUPLE

RTD : BLOC D'ENTREE SONDE A RESISTANCE

Fonction du bloc

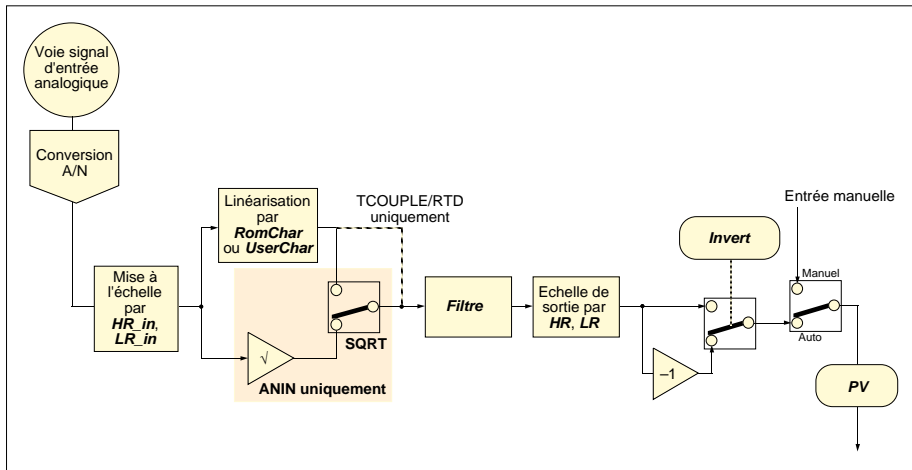


Figure 2-1 Schéma du bloc ANIN/TCOUPLE/RTD

Se reporter au schéma de la figure 2-1 qui s'applique aux trois blocs. Les blocs entrée analogique, thermocouple ou RTD convertissent les entrées de tension ou de courant (sélectionnées par logiciel) ou dans le cas de RTD une mesure de résistance en une valeur à point flottant en unités physiques. Ces blocs conditionnent, étalonnent et filtrent les signaux, et disposent d'une fonction d'alarme.

Les blocs peuvent être utilisés en mode manuel où la variable procédé (PV) peut être forcée à prendre une valeur spécifique pour effectuer des essais. Le fonctionnement normal de PV reprend, lorsque le bloc est en mode non-manuel.

NOTA. Les unités de température - °C, °F, K et R - peuvent être sélectionnées par l'intermédiaire du bloc T100. Voir les détails au chapitre 10, *Blocs de fonction CONFIG*.

Paramètres du bloc

Les paramètres sont les mêmes pour les trois blocs, sauf indication contraire, et sont répertoriés dans la table ci-dessous. Les symboles utilisés dans la Table 2-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications des blocs*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
PV	Variable procédé (sortie bloc)	Eng	
HR, LR	Echelle haute & basse de PV	Eng	
HAA, LAA	Niveaux d'alarme absolue haute & basse PV	Eng	
Filter	Constante de temps du filtre du premier ordre	Sec.	
Invert	Inversion du signal d'entrée	V/F	
Sqrt	Sélection fonction racine carrée (ANIN uniquement)	V/F	
Cutoff	Seuil bas racine carrée (ANIN uniquement)	Eng	
Default	Valeur PV si Sqrt < Cutoff (ANIN uniquement)	Eng	
LeadRes	Valeur résistance cordon (2 fils uniquement)	Eng	
RomChar	Caractérisation sélectionnée (voir table 2-2.)	Menu	
UserChar	Caractérisation 16 points définie par l'utilisateur	Alphanumérique	
Alarmes			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts Module Txxx/alim. transmetteur, etc.	V/F	
HiLevel	Alarme haute absolue (HAA)	V/F	
LoLevel	Alarme basse absolue (LAA)	V/F	
OutOfRange	PV en dehors de l'échelle HR-LR (>5%)	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Site du T100	Entier	
Site Type	Nombre de voies d'entrée (ANIN uniquement)	Menu	
Channel	Numéro de voie (1-8) (ANIN uniquement)	Entier	
InType	Type d'entrée (pas TCOUPLE)	Menu	
HR_in	Point d'étalonnage HR		
LR_in	Point d'étalonnage LR		
Break	Protection rupture haute ou basse (pas RTD)	Menu	
CJ_temp	Température soudure froide (TCOUPLE uniquement)		°C
Status	Etat communicat./matériel	(A)BCD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC du T100)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version d'étalonnage des données	V/F — 1	C
BadCal	Données d'étalonnage corrompue	V/F — 2	
PSUShort	Surcharge alim. transmetteur (ANIN uniquement)	V/F — 4	
BadRef	Défaut tension de référence	V/F — 8	
BlockErr	Données RAM corrompues (drapeau transitoire)	V/F — 1	B
Break	Entrée circuit ouvert sur RTD (RTD uniquement)	V/F — 2	
		— 4	
		— 8	

Table 2-1 Paramètres des blocs ANIN/TCOUPLE/RTD

Menu de spécifications des blocs

Cette section s'applique aux trois blocs, sauf indication contraire.

Dbase. Permet de spécifier si le bloc est un bloc local ou un bloc image - c'est à dire l'emplacement physique de la base de données du bloc. Un bloc *local* fonctionne dans l'instrument où se trouve sa base de données ou là où elle se trouvera en définitive (par ex. un T100 ou T1000). Son champ *Dbase* est mis sur <local>, valeur par défaut. Un bloc *image* est une "image" d'un bloc déporté, c'est à dire un bloc fonctionnant dans un autre instrument sur le LIN, qui permet le dialogue avec le bloc déporté. Dans un bloc image, *Dbase* permet de spécifier le nom de la base de données déportée contenant le bloc "réel", qui doit être le même que le nom de repère du bloc en-tête du T100 ou T1000 - le paramètre *Bloc*.

Afin de localiser la base de données et son logiciel de conduite dans un autre instrument mis en réseau, entrez *Dbase* pour accéder à un menu secondaire de trois champs - *Dbase*, *Node* et *Rate ms* (*Dbase*, *Noeud* et *vitesse en ms*). Sélectionner et entrer le nom de *Dbase* (par défaut = <local>) pour afficher un menu avec la liste des bases de données <new>, <local> (nouvelle, locale), ainsi que les noms de toutes les bases de données disponibles. Sélectionner une base de données, ou <new> qui permet d'entrer un nom de base de données, suivi par son *noeud*, à l'affichage de l'invite. Il faut également entrer une valeur en millisecondes pour *Rate*. *Rate* est la période de mise à jour minimale (c'est à dire la vitesse maximale), à laquelle un bloc image individuel est transmis sur le réseau local instruments (LIN). La valeur par défaut est de 10 ms minimum, c'est à dire 100 Hz maximum. La valeur de *Rate* peut se situer entre 10 ms et 64 s. Il faut noter que les valeurs de vitesse ne représentent que les périodes de mise à jour minimales, et que des réseaux très chargés peuvent ne pas atteindre les vitesses de mise à jour les plus rapides.

NOTA. Afin que le système fonctionne correctement, il faut que la base de données dont le nom a été sélectionné réside au noeud spécifié, et doit contenir un bloc réel correspondant du même type que le bloc image local.

Block (Bloc). Contient le nom de repère du bloc défini par l'utilisateur (8 caractères maximum), qui l'identifie et le distingue des autres blocs du même *Type*. Le nom par défaut est "NoName" (pas de nom). Le nom de repère est affiché comme un libellé sur l'icône du bloc, sous le nom du type. Un schéma de boucles ne peut être sauvegardé que si tous les blocs ont un nom de repère.

Type. Le champ de lecture uniquement *Type* indique le mnémonique du nom de type du bloc de fonction, par ex. ANIN, PID, SIM, etc., qui est également affiché comme un libellé sur l'icône du bloc, au-dessus de son nom de repère.

Mode. (AUTO*/INT CJ†/EXT CJ†/MANUEL). Mode de fonctionnement. Pour tous les blocs, Le mode manuel permet de forcer le paramètre *PV* à prendre une valeur spécifiée au cours de la conduite. Lorsque le mode INT CJ est sélectionné (TCOUPLE uniquement) la compensation interne de la soudure froide calculée à partir du thermocouple intégré dans le bloc de liaison T110 est appliquée. Lorsque le mode EXT CJ est sélectionné (TCOUPLE uniquement), la compensation externe calculée à partir de

l'entrée du bloc CJ_temp est appliquée.

*Autre que bloc TCOUPLE † Bloc TCOUPLE uniquement

NOTA. Dans le bloc TCOUPLE, une compensation de soudure froide est *toujours* appliquée, sauf si "None" (aucune) est sélectionné pour *RomChar* et que *UserChar* ne soit pas spécifié. Il faut suivre cette procédure pour mesurer des entrées en millivolts de faible niveau.

PV. Sortie bloc (Variable procédé). En lecture uniquement en mode automatique.

HR, LR. Echelle haute & basse de *PV* en unités physiques. *HR* et *LR* permettent de définir deux points sur une échelle linéaire en unités physiques qui sont projetés respectivement sur les points *HR_in* et *LR_in* sur une échelle linéaire (millivolts). Ces deux échelles sont utilisées par le bloc pour calculer la sortie *PV* du signal mesuré à l'entrée.

L'équation de conversion équivalente est la suivante :

$$PV = LR + \left[\frac{\text{Input} - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

HAA, LAA. Limites d'alarmes absolues hautes & basses pour *PV*, avec une hystérésis intégrée de 0,5 % à l'intérieur des limites.

Filter (Filtre). Constante de temps du filtre du premier ordre (0 à 250 sec.)

Invert (Inversion). Permet d'inverser le sens du signal d'entrée.

Sqrt. (*ANIN uniquement*). Permet de sélectionner la fonction racine carrée.

Cutoff (Coupure). (*ANIN uniquement*). Permet de spécifier une valeur de seuil bas pour *PV*, auquel cas *PV* passe à Default (par défaut).

Default (Par défaut). (*ANIN uniquement*). Valeur adoptée par *PV* lorsque la valeur calculée est inférieure à *Cutoff* (coupure).

LeadRes. (*RTD uniquement*). Valeur de résistance du cordon (Ω). Utilisé conjointement avec les mesures 2 fils uniquement.

RomChar. (None† /J/K/T/S/R/E/B/N/W/W5/W3/MoRe/PRT†*/CU10†*). Permet de sélectionner le type de caractérisation dans la bibliothèque ROM. Voir les détails table 2-2.

† Ces trois uniquement dans le bloc RTD. * Pas dans le bloc TCOUPLE.

NOTA. Afin d'éviter des conflits, la spécification d'une valeur dans *RomChar* autre que "None" (aucune) supprime automatiquement toute entrée dans le champ *UserChar*. De même, l'écriture dans nom de repère dans *UserChar* supprime toute entrée *RomChar*.

Type	Plage (°C)	Type	Plage (°C)
J	-200 à 1200	B	0 à 1820
K	-270 à 1370	N	0 à 1300
T	-2270 à 400	W	1000 à 2300
S	-50 à 1760	W₃	0 à 2490
R	-50 à 1760	W₅	0 à 2300
E	-270 à 1000	MoRe	0 à 1990

Table 2-2 Table de caractérisation

UserChar. Permet de définir une table de linéarisation/caractérisation personnalisée à 16 points (x,y). Entrer le nom de repère du bloc UCHAR dans le schéma de boucles contenant la table requise (Voir nota ci-dessus).

Alarms (Alarmes). Ce champ permet d’afficher le nom de l’alarme active avec la priorité la plus haute dans le bloc (la priorité est définie ci-dessous). Il permet également d’accéder à une fenêtre avec trois colonnes - Nom, Etat et Priorité. La colonne *Nom* (Name) donne la liste des noms de toutes les alarmes du bloc, par ex. Logiciel, Matériel, etc. La colonne *Etat* (Status) affiche pour chaque alarme le message “In Alarm” (en alarme) si la condition d’alarme est présente, ou un blanc dans le cas contraire. La colonne *Priorité* (Priority) montre le numéro de priorité défini par l’utilisateur pour chaque alarme. Le numéro de priorité affiché clignote si l’alarme a une priorité de 6 ou plus et n’est pas acquittée, même si la condition d’alarme a disparu. “Priorité” est le seul champ de lecture/écriture dans la fenêtre alarmes.

Numéro de priorité d’alarme. Il y a quatre types de priorité d’alarme. *Priorité 0*, la priorité la plus basse, coupe l’alarme. *Les priorités de 1 à 5* permettent de spécifier des alarmes qui s’auto-acquittent. Dans ce cas, le message “In Alarm” apparaît lorsque la condition d’alarme se produit et disparaît, lorsque l’alarme a disparu, il n’est pas nécessaire de l’acquitter. *Les priorités de 6 à 10* permettent de spécifier des alarmes à “acquitter”, dont le numéro de priorité clignote lorsque la condition d’alarme se produit (et qui font également clignoter les touches de fonction de conduite du T1000). Pour acquitter l’alarme et arrêter le clignotement, mettre l’alarme en surbrillance et appuyer sur la touche de fonction ACK du T1000. Lorsque la condition d’alarme disparaît, le message “In Alarm” disparaît - mais le clignotement continue et l’alarme reste active, si elle n’a pas encore été acquittée. *Les priorités 11 à 15* (priorités les plus importantes) sont les mêmes que les priorités de 6 à 10, mais elles mettent à 1 le bit AlmRelay du bloc T1000 et activent le relais d’alarme matériel du T1000. Ceux-ci ne sont remis à zéro que lorsque l’alarme n’est plus active (après un court délai).

■ **Software (Logiciel).** Tous les blocs ont une alarme logiciel (et une alarme combinée), au minimum. Une alarme logiciel (priorité par défaut 1) est déclenchée après une erreur de total de contrôle dans un bloc, c’est à dire après corruption de sa base de données. En ce qui concerne les blocs images, S6000 et TAN, un défaut de communication du bloc principal active également l’alarme.

■ **Hardware (Matériel).** Une alarme matériel se déclenche dans les conditions suivantes :

- Coupure de courant
- Type de module incorrect
- Défaut alimentation transmetteur (T120 uniquement)
- Interruption entrée RTD (T111 uniquement)
- Défaut matériel circuit (voir champ Etat).

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 = Off.

- **HiLevel, LoLevel (Niveau haut, Niveau bas).** Ces limites d'alarmes sont définies par les paramètres *HAA* et *LAA*. Il y a une bande d'hystérésis de 0,5 % sur chaque limite d'alarme, ce qui permet une transition nette entre la condition d'alarme et de non-alarme.
 - **Outrange (Hors échelle).** Cette alarme est déclenchée dès que *PV* atteint une valeur de 5 % au-dessus ou en-dessous de *HR* ou *LR*. Il y a une hystérésis de 0,5 % pour chaque valeur de l'échelle.
 - **Combined (Combinée).** Tous les blocs ont une alarme combinée (et logiciel), au minimum. Une alarme combinée est une copie de l'alarme active avec la priorité la plus haute dans le bloc. Elle adopte le même message d'état et le numéro de priorité, y compris le clignotement que l'alarme copiée. Le nom de cette alarme à la priorité la plus haute est affiché dans le champ alarmes du menu de spécifications.
- Lorsque plusieurs alarmes sont actives, l'alarme combinée est sélectionnée d'après la liste suivante d'attributs :
1. Numéro de priorité (attribut de priorité la plus haute)
 2. Alarme non-acquittée et In (en alarme)
 3. Non-acquittée
 4. En alarme
 5. La plus haute dans la liste des noms du champ des alarmes (attribut de priorité la plus basse).

Il est à noter que, à égalité, "Non-acquittée" est traité comme plus important que "En alarme".

NOTA. Le drapeau d'alarme dans le coin supérieur gauche de l'écran de conduite affiche l'alarme ayant la priorité la plus haute dans tout le schéma de boucles (si elle existe). La priorité est définie comme dans la liste ci-dessus, sauf pour l'attribut de la priorité la plus basse qui se lit alors comme :

5. La plus haute dans le menu déroulant de conduite, accès par la touche de fonction ENGR.

(Sur le T100, le menu ENGR donne la liste des blocs dans l'ordre dans lequel ils ont été placés sur la page de configuration des schémas de boucles en utilisant la touche de fonction MAKE (FAIRE), le premier bloc étant placé en haut de la liste.

SiteNo (N° de site). Permet de définir l'emplacement à l'intérieur d'un T100 (1 à 16). Ce champ est actif dans une base de données T100, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs *SiteNo* des blocs E/S *image* correspondants copient automatiquement les paramètres "réels".

SiteType (Type de site). (1 CHAN/2 CHAN/8 CHAN). (*ANIN uniquement*). Le nombre de voies d'entrée. La valeur doit être compatible avec le paramètre de la voie.

Channel (Voie). (*ANIN uniquement*). Permet de définir le numéro de voie (1 à 8). Ce champ est actif dans la base de données T100, uniquement lorsqu'il est utilisé conjointement avec un module d'entrée multi-voies. La valeur doit être compatible avec le paramètre du *SiteType*.

InType (Type d'entrée). (*ANIN*: Volts/mA. *TCOUPLE*: non utilisé. *RTD*: 4 WIRE/3 WIRE/2 WIRE (4 FILS/3 FILS/2 FILS)). Permet de définir le type d'entrée.

HR_in, LR_in. Echelle haute & basse du signal mesuré de l'entrée. *HR_in* et *LR_in* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR* et *LR* sur une échelle linéaire *PV* en unités physiques. Voir les détails à la section *HR, LR* ci-dessus.

Break (Rupture). (haut/bas). (*Pas RTD*). Permet de définir la protection de rupture de mesure haute ou basse de l'échelle.

NOTA. Les entrées mA doivent avoir une protection de rupture basse de l'échelle.

C_J temp (Température soudure froide). (*TCOUPLE uniquement*).

Température soudure froide. Lorsque le mode INT CJ est sélectionné, *CJ_temp* indique la température du bloc connecteur du module T100, mesurée par sa sonde intégrée. Lorsque le mode EXT CJ est sélectionné, vous pouvez écrire une température de soudure froide connue dans *CJ_temp*, ou entrer une valeur par l'intermédiaire d'une liaison du schéma de boucles. Il est à noter que, sauf si, à la fois, *RomChar* et *UserChar* ne sont pas spécifiés, la mesure de l'entrée est toujours soumise à une compensation de soudure froide - calculée à partir de *CJ_temp* - et convertie en unités de température. Voir le schéma de la figure 2-2.

NOTA. Toutes les températures sont indiquées et doivent être entrées suivant les unités spécifiées par le paramètre *IP_Type* du bloc T100 correspondant, c'est à dire: °C, °F, K ou R.

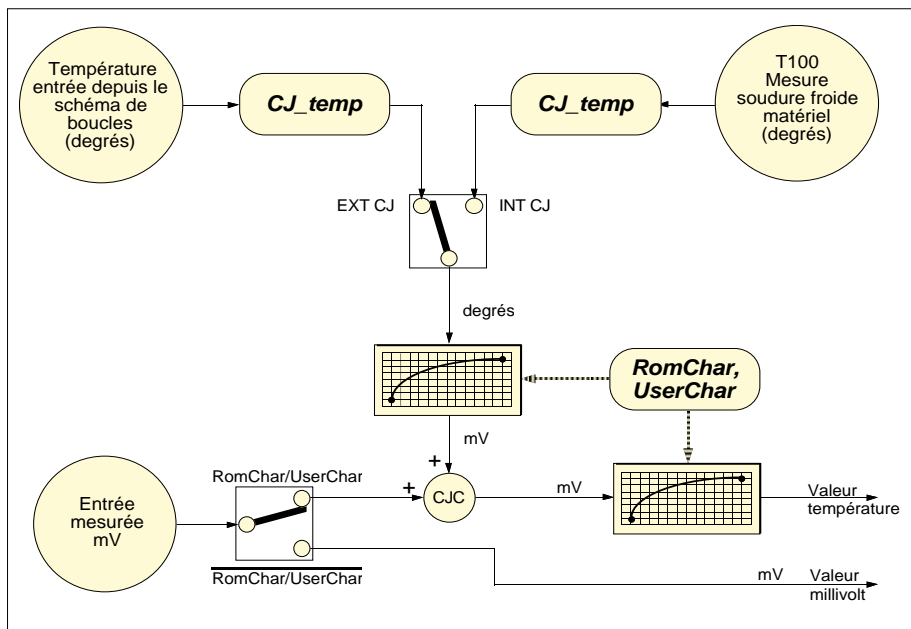


Figure 2-2 Compensation Soudure Froide

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-1.

NOTA. (ANIN uniquement). *BlockErr* est mis à 1, si la RAM est insuffisante pour exécuter le bloc. Le bloc n'est pas mis à jour dans ce cas, et d'autres blocs doivent être supprimés pour libérer la RAM.

ANOP : BLOC DE SORTIE ANALOGIQUE

Fonction du bloc

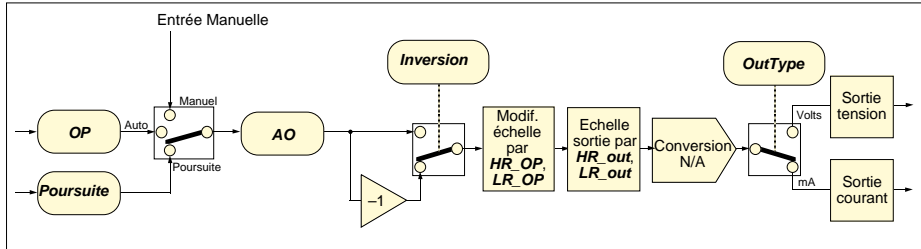


Figure 2-3 Schéma du bloc ANOP

Se reporter à la figure 2-3. Le bloc de sortie analogique convertit les variables analogiques mises à l'échelle (nombres à point flottant) d'un schéma de boucles en sorties de tension ou de courant (sélectionnées par logiciel) dans le T100. Le bloc dispose du contrôle Auto/Manuel/Poursuite, d'inversion et des alarmes.

Configurations de sortie double. Les modules de sortie analogique dans différents T100 peuvent être interconnectés, en connectant les sorties de tension en parallèle et les sorties de courant en série pour augmenter le niveau de l'intégrité des systèmes. Un circuit de détection du retour de la sortie dans chaque module permet de détecter lorsque sa sortie est surchargée par le module opposé, et l'unité de sortie principale est automatiquement sélectionnée d'après le critère du "plus fort gagne". Les figures 2-4 et 2-5 montrent des exemples de configurations redondantes.

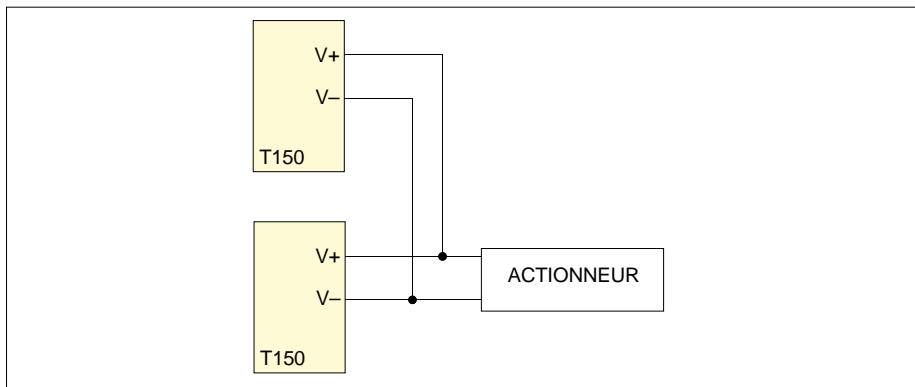


Figure 2-4 Configuration redondante T150 - Sortie tension

Paramètres du bloc

Les symboles utilisés dans la table 2-3 sont expliqués au chapitre 1. Des informations supplémentaires sont données dans la section suivante *Menu de spécifications du bloc*.

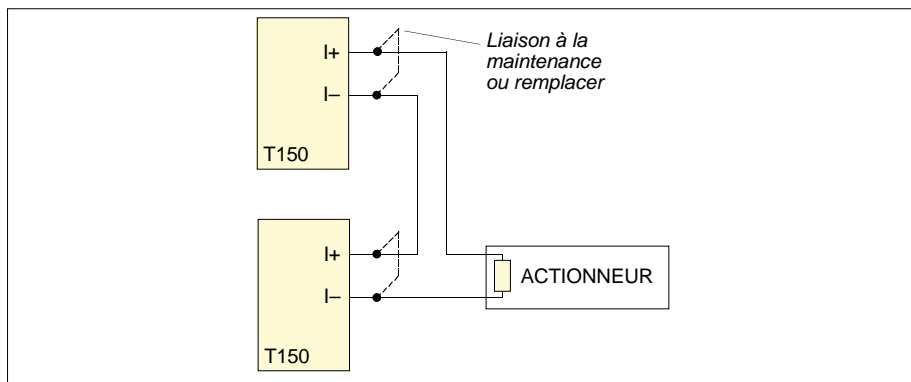


Figure 2-5 Configuration redondante T150 - Sortie courant

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Fallback	Mode de fonctionnement supprimé	Menu	
OP	Demande de sortie	Eng	
HR_OP, LR_OP	Echelle haute et basse AO	Eng	
AO	Sortie analogique	Eng	
Track	Valeur de poursuite	Eng	
SelTrack	Sélection mode poursuite	V/F	
Invert	Sélection inversion	V/F	
NotAuto	Etat mode PAS Auto	V/F	
Alarmes			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut Module Txxx	V/F	
CctFault	Défaut circuit (Sortie circuit ouvert)	V/F	
OvrDrive	Condition surcharge sortie	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Site du T100	Entier	
OutType	Type de sortie (Volts/mA)	Menu	
Channel	Numéro de voie		
HR_out	Point d'étalonnage HR	Eng	
LR_out	Point d'étalonnage LR	Eng	
Status	Etat communicat./matériel	(AB)CD Hex	
Reset	Drap. transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur vers. d'étalonn. des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
		V/F — 4	
		V/F — 8	

Table 2-3 Paramètres du bloc ANOP

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUEL/POURSUITE). Mode de fonctionnement en cours.

Fallback (Reprise). (AUTO/MANUEL). Le mode de fonctionnement de retour (supprimé).

OP. Demande de sortie. Entrée dans le bloc à partir du schéma de boucles, et la source *AO* en mode Auto.

HR_OP, LR_OP. Echelle haute et basse de *AO* en unités physiques. *HR_OP* et *LR_OP* permettent de définir deux points sur une échelle linéaire en unités physiques qui sont projetés respectivement sur les points *HR_out* et *LR_out* sur une échelle linéaire (volts ou milliampères). Ces deux échelles sont utilisées par le bloc pour calculer le signal de sortie de *AO*. L'équation de conversion équivalente est la suivante :

$$\text{Sortie} = LR_out + \left[\frac{AO - LR_OP}{HR_OP - LR_OP} \times (HR_out - LR_out) \right]$$

AO. Sortie analogique. Sortie du bloc avant l'inversion (si spécifiée) et modification de l'échelle.

Track (Poursuite). Source de *AO* en mode poursuite.

SelTrack (Sélection poursuite). Permet de sélectionner le mode poursuite.

Invert (Inversion). Permet de sélectionner l'inversion. Inverse le sens du signal de sortie pour le fonctionnement à sécurité intégrée et pour les vannes à action inverse.

NotAuto (Pas auto). VRAI lorsque le bloc ne fonctionne pas en mode Auto.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes :

- Coupure de courant
- Type de module incorrect
- Défaut matériel circuit (voir champ Etat)

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

■ **CctFault (Défaut circuit).** Signalé lorsque le module détecte une condition de circuit ouvert sur la sortie.

■ **OvrDrive (Condition de surcharge en sortie).** Cette alarme indique qu'un signal de sortie est surchargé par un autre module de sortie. Utilisée dans les schémas de boucles redondants.

- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de définir l'emplacement d'un T100 (1 à 16). Ce champ est actif dans une base de donnée T100, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site de T100 valable a été entré, il en peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro* ; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *image* correspondant copient automatiquement les paramètres "réels".

OutType (Type de sortie). (Volts/mA). Permet de définir un type de sortie de *tension* ou de *courant*.

HR_out, LR_out (Echelle haute & basse du signal de sortie). *HR_out* et *LR_out* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR_OP* et *LR_OP* sur une échelle linéaire *AO* en unités physiques. Voir les détails à la section *HR_OP, LR_OP* ci-dessus. Noter que *HR_out* ne peut dépasser 10 V ou 20 mA lorsque *OutType* est en volts ou mA.

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-3.

DGIN_8 : BLOC D'ENTREE LOGIQUE A 8 VOIES

Fonction du bloc

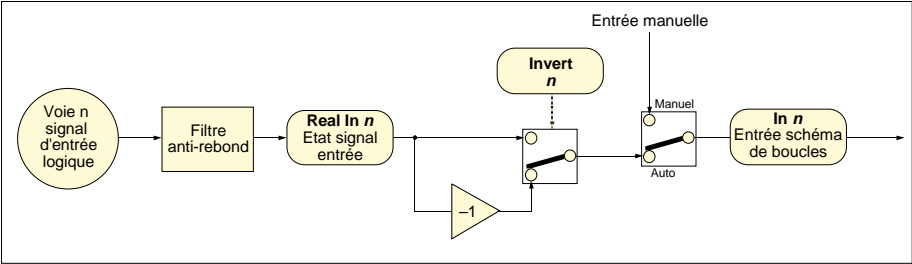


Figure 2-6 Schéma du bloc DGIN_8

Se reporter à la figure 2-6, qui schématise l’une des huit voies logiques. Le bloc DGIN_8 permet d’entrer 8 signaux logiques dans le schéma de boucles, et dispose du contrôle Auto/Manuel, de l’inversion et de l’anti-rebond. Les entrées peuvent être configurées soit comme détectant toutes des contacts ou des tensions. Les contacts utilisent la sortie de l’alimentation transmetteur, qui dispose d’une alarme de surcharge; les entrées de détection de tension ont des seuils de commutation variables.

Les champs alarme ne sont pas directement disponibles sur les voies d’entrée logiques. Le bloc DIG-ALARM devrait être utilisé pour les alarmes sur chaque entrée, si nécessaire.

Paramètres du bloc

Les symboles utilisés dans la table 2-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Real In	Etats des liaisons du signal d’entrée (réels)	CD Hex	
Bit0	Etat signal d’entrée voie 1	V/F — 1	D
Bit1	Etat signal d’entrée voie 2	V/F — 2	
Bit2	Etat signal d’entrée voie 3	V/F — 4	
Bit3	Etat signal d’entrée voie 4	V/F — 8	
Bit4	Etat signal d’entrée voie 5	V/F — 1	C
Bit5	Etat signal d’entrée voie 6	V/F — 2	
Bit6	Etat signal d’entrée voie 7	V/F — 4	
Bit7	Etat signal d’entrée voie 8	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
Invert	Etats inversés des signaux entrées (réels)	CD Hex	
Bit0	Signal d'entrée inversé voie 1	V/F — 1	D
Bit1	Signal d'entrée inversé voie 2	V/F — 2	
Bit2	Signal d'entrée inversé voie 3	V/F — 4	
Bit3	Signal d'entrée inversé voie 4	V/F — 8	
Bit4	Signal d'entrée inversé voie 5	V/F — 1	C
Bit5	Signal d'entrée inversé voie 6	V/F — 2	
Bit6	Signal d'entrée inversé voie 7	V/F — 4	
Bit7	Signal d'entrée inversé voie 8	V/F — 8	
In	Etats entrée du schéma de boucles	CD Hex	
Bit0	Etat entrée schéma voie 1	V/F — 1	D
Bit1	Etat entrée schéma voie 2	V/F — 2	
Bit2	Etat entrée schéma voie 3	V/F — 4	
Bit3	Etat entrée schéma voie 4	V/F — 8	
Bit4	Etat entrée schéma voie 5	V/F — 1	C
Bit5	Etat entrée schéma voie 6	V/F — 2	
Bit6	Etat entrée schéma voie 7	V/F — 4	
Bit7	Etat entrée schéma voie 8	V/F — 8	
Alarms			
Software	Données corrompues/Défaut de communication		
Hardware	Défauts Module Txxx/Alim transmetteur		
Combined	Alarme combinée du bloc		
SiteNo	Site du T100	Entier	
InType	Type d'entrée	Menu	
Thresh	Seuil de commutation	Numérique	
DeBounce	Temps anti-rebond (sec.)	Numérique	
Status	Etat communicat./matériel	(AB)CD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version étalonnage données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
PSUShort	Surcharge alim. transmetteur	V/F — 4	

Table 2-4 Paramètres du bloc DGIN_8

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUEL). En mode Automatique, le champ état *In* suit les stimuli d'entrée des liaisons du champ. En mode Manuel, le champ d'état *In* peut être manipulé indépendamment des stimuli du champ.

RealIn (Entrée réelle). Champ binaire indiquant l'état réel de chacune des huit liaisons d'entrée.

Invert (Inversion). Champ binaire permettant l'inversion de l'une des huit entrées dans le schéma de boucles.

In (Entrée). Champ binaire indiquant les états des huit entrées dans les schémas de boucles. Les entrées peuvent être sous la forme vraie ou inversée suivant les états des champs Invert.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel peut être déclenchée dans les conditions suivantes :

- Coupure de courant
- Surcharge alimentation
- Type de module incorrect
- Défaut matériel circuit (voir champ Etat).

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de définir l'emplacement d'un T100 (1 à 16). Ce champ est actif dans une base de donnée T100, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro* ; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *image* correspondant copient automatiquement les paramètres "réels".

InType (Type d'entrée). (Volts/Contacts). Permet de définir le type d'entrée.

Thresh (Seuil de commutation). Permet de définir le seuil pour l'option entrée tension.

DeBounce (Anti-rebond). Temps anti-rebond en secondes. Il s'agit de la durée minimum qu'une modification du signal d'entrée logique doit prendre avant que le paramètre *RealIn* n puisse passer au nouvel état. L'action de filtrage de *DeBounce* (Anti-rebond) est représentée à la figure 2-7. (Le temps anti-rebond minimum dépend du temps d'échantillonnage).

Status (Etat). Champ binaire indiquant des situations d'erreur générales communicat./matériel. Voir les détails, table 2-4.

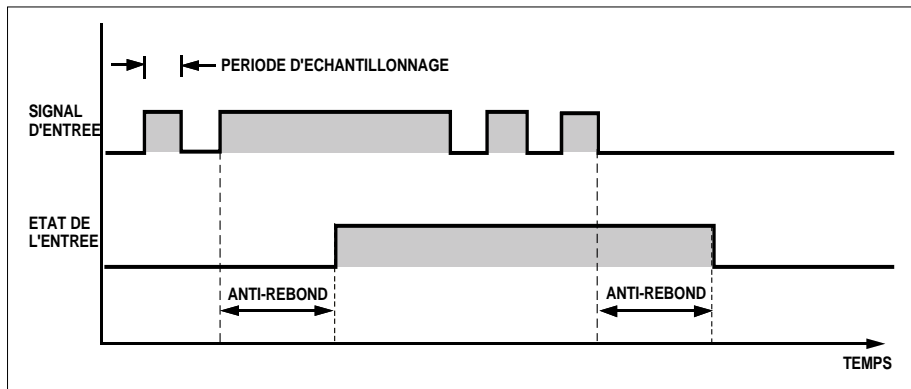


Figure 2-7 Effets de l'anti-rebond sur le signal d'entrée logique

DGOUT_8 : BLOC DE SORTIE LOGIQUE A 8 VOIES

Fonction du bloc

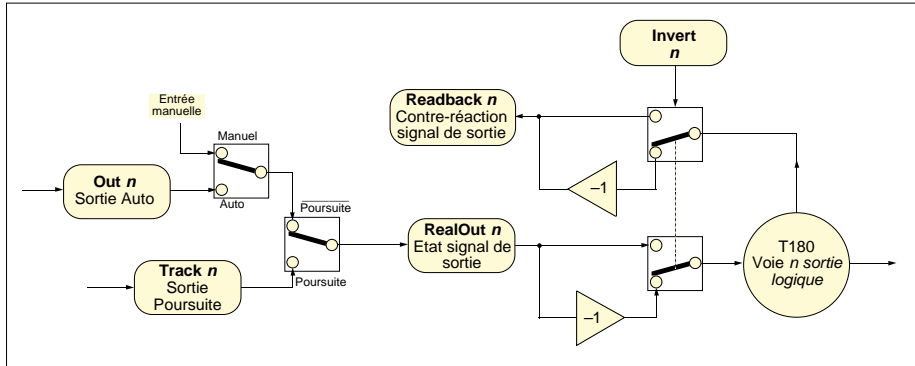


Figure 2-8 Schéma bloc DGOUT_8

Se reporter à la figure 2-8, qui schématise l'une des huit voies logiques. Le bloc de sortie logique permet de transférer 8 signaux logiques vers des voies de sorties logiques sur le T100 (Module T180). Le bloc dispose du contrôle Auto/Manuel/Poursuite, d'inversion et de vérification de discordance du retour de la sortie.

Paramètres du bloc

Les symboles utilisés dans la table 2-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Fallback	Mode de fonctionnement supprimé	Menu	
Out	Etats sortie du schéma de boucles	CD Hex	☐➡
Bit0	Etat sortie voie 1	V/F — 1	D
Bit1	Etat sortie voie 2	V/F — 2	
Bit2	Etat sortie voie 3	V/F — 4	
Bit3	Etat sortie voie 4	V/F — 8	
Bit4	Etat sortie voie 5	V/F — 1	C
Bit5	Etat sortie voie 6	V/F — 2	
Bit6	Etat sortie voie 7	V/F — 4	
Bit7	Etat sortie voie 8	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
Track	Etats sortie poursuite	CD Hex	
Bit0	Etat sortie poursuite voie 1	V/F —1	D
Bit1	Etat sortie poursuite voie 2	V/F —2	
Bit2	Etat sortie poursuite voie 3	V/F —4	
Bit3	Etat sortie poursuite voie 4	V/F —8	
Bit4	Etat sortie poursuite voie 5	V/F —1	C
Bit5	Etat sortie poursuite voie 6	V/F —2	
Bit6	Etat sortie poursuite voie 7	V/F —4	
Bit7	Etat sortie poursuite voie 8	V/F —8	
RealOut	Etats sortie (avant inversion)	CD Hex	
Bit0	Etat sortie voie 1	V/F —1	D
Bit1	Etat sortie voie 2	V/F —2	
Bit2	Etat sortie voie 3	V/F —4	
Bit3	Etat sortie voie 4	V/F —8	
Bit4	Etat sortie voie 5	V/F —1	C
Bit5	Etat sortie voie 6	V/F —2	
Bit6	Etat sortie voie 7	V/F —4	
Bit7	Etat sortie voie 8	V/F —8	
Invert	Etats inversés signal de sortie (réels)	CD Hex	
Bit0	Signal de sortie inversé voie 1	V/F —1	D
Bit1	Signal de sortie inversé voie 2	V/F —2	
Bit2	Signal de sortie inversé voie 3	V/F —4	
Bit3	Signal de sortie inversé voie 4	V/F —8	
Bit4	Signal de sortie inversé voie 5	V/F —1	C
Bit5	Signal de sortie inversé voie 6	V/F —2	
Bit6	Signal de sortie inversé voie 7	V/F —4	
Bit7	Signal de sortie inversé voie 8	V/F —8	
ReadBack	Contre-réaction des états sortie du T180	CD Hex	
Bit0	Contre-réaction voie 1	V/F —1	D
Bit1	Contre-réaction voie 2	V/F —2	
Bit2	Contre-réaction voie 3	V/F —4	
Bit3	Contre-réaction voie 4	V/F —8	
Bit4	Contre-réaction voie 5	V/F —1	C
Bit5	Contre-réaction voie 6	V/F —2	
Bit6	Contre-réaction voie 7	V/F —4	
Bit7	Contre-réaction voie 8	V/F —8	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts alim. Module Txxx	V/F	
CctFault	Détection différence RealOut/ReadBack	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Site du T100	Entier	

suite...

...suite




Paramètre	Fonction	Unités	Etat
Pullup	Tension de sortie état logique haut	Menu	
SelTrack	Sélection mode poursuite	V/F	
NotAuto	Etat mode PAS Auto	V/F	
Status	Etat communicat./matériel	(AB)CD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version d'étalonnage des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	

Table 2-5 Paramètres du bloc DGOUT_8

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUEL/POURSUITE). Le mode de fonctionnement en cours.

Fallback (Reprise). (AUTO/MANUEL). Le mode de fonctionnement de retour (supprimé).

Out (Sortie). En mode automatique, ce paramètre de champ binaire indique l'état des voies de sortie défini par le schéma de boucles. En mode manuel, ce paramètre est en lecture/écriture. Ce champ peut indiquer la forme vraie ou inversée des voies de sorties réelles suivant le statut des champs *inversion*.

Track (Poursuite). Champ binaire contrôlant l'état des sorties en mode poursuite.

RealOut (Sortie réelle). Champ binaire indiquant l'état (avant toute inversion) des liaisons de sortie du champ.

Invert (Inversion). Champ binaire spécifiant les inversions du champ.

Readback (Retour d'état). Champ binaire indiquant le retour des états réels de sortie à partir du module T180 (avant toute inversion). Permet de vérifier une différence par comparaison avec *RealOut* qui devrait être identique.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes :

- Coupure de courant
- Type de module incorrect

- Défaut matériel circuit (voir champ Etat).

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

- **CctFault (Défaut circuit).** Cette alarme se déclenche lorsqu'une différence est détectée entre l'état de RealOut et de ReadBack, indiquant qu'il y a un défaut sur une des voies de sortie.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de définir l'emplacement à l'intérieur d'un T100 (1 à 16). Ce champ est actif dans une base de donnée T100, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *image* correspondant copient automatiquement les paramètres "réels".

Pullup (Tirage). (Externe/5/15/24). Permet de définir la tension de sortie utilisée pour l'état logique haut.

SelTrack (Choix mode poursuite). Permet de sélectionner le mode poursuite.

NotAuto (Non auto). Indique que le bloc ne fonctionne pas en mode automatique.

Status (Etat). Champ binaire indiquant une condition d'erreur générale de communicat./matériel. Voir les détails dans la table 2-5.

FULL_TC8 : BLOC D'ENTREE THERMOCOUPLE

Fonction du bloc

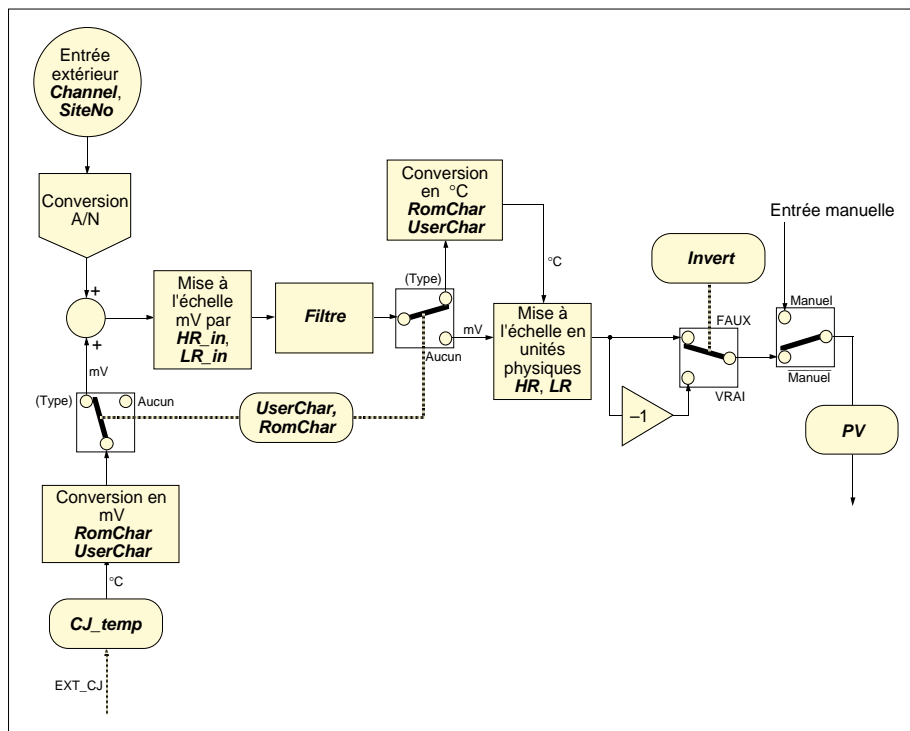


Figure 2-9 Schéma du bloc FULL_TC8

La figure 2-9 montre les unités °C uniquement. Les températures sont affichées, et doivent être entrées dans les unités spécifiées par le paramètre *IP_type* du bloc T100 correspondant, c'est à dire °C, °F, K, or R. Le bloc FULL_TC8 traite les signaux entrés à partir de l'extérieur par l'une des voies d'un boîtier T d'entrée thermocouple à huit voies T112. Le bloc peut mesurer directement les millivolts ou la température en utilisant les tables de caractérisation résidentes (*RomChar*) ou personnalisées (*UserChar*) et la compensation de soudure froide. La température de la soudure froide (*CJ_temp*) peut provenir soit de la sonde intégrée dans chaque bloc de liaison T112 (mode INT_CJ) ou à partir d'un bloc isothermique externe (mode EXT_CJ). Pour toutes les mesures, le bloc fournit une valeur *PV* mise à l'échelle en unités physiques.

Le mode manuel, les alarmes absolues hautes/basses et hors échelle, le filtrage premier ordre, et l'inversion des signaux sont également disponibles dans le bloc FULL_TC8.

Paramètres du bloc

Les symboles utilisés dans la table 2-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode fonctionnement/compensation CJ en cours	Menu	
PV	Variable procédé (sortie bloc)	Eng	
CJ_temp	Température soudure froide	°C	
HR, LR	Echelle haute & basse PV	Eng	
HAA, LAA	Niveaux d'alarmes absolues hautes et basses PV	Eng	
Filter	Constante de temps du filtre du premier ordre	Secs.	
Invert	Inversion signal d'entrée	V/F	
RomChar	Caractérisation sélectionnée (voir table 2-2)	Menu	
UserChar	Caractérisation 16 points définie par l'utilisateur	Alphanumérique	
Alarms			
Software	Données corrompues/Défaut de communication		
Hardware	Défauts alim. Module Txxx		
HiLevel	Alarme absolue haute (HAA)		
LoLevel	Alarme absolue basse (LAA)		
Range	PV en dehors de l'échelle HR-LR (>5%)		
Combined	Alarme combinée du bloc		
SiteNo	Site du T100	Entier	
Channel	Numéro de voie T112	Entier	
HR_in, LR_in	Echelle haute & basse entrée millivolts		
Break	Protection échelle incrément ou décrément	Menu	
Status	Etat communicat./matériel	(A)BCD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version d'étalonnage des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompue	V/F — 2	
	(non utilisé)	V/F — 4	
BadRef	Défaut tension de référence	V/F — 8	
BlockErr	Données RAM corrompues (drapeau transitoire)	V/F — 1	B
		2	
		4	
		8	

Table 2-6 Paramètres du bloc FULL_TC8

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (INT_CJ/EXT_CJ/MANUEL). Mode de fonctionnement. Sélectionner INT_CJ (Auto), lorsque la compensation interne de température de la soudure froide est appliquée, basée sur la température mesurée de façon interne par le matériel d'entrée du T100/T122, et indiquée par le paramètre *CJ_temp*. Sélectionner EXT_CJ (Auto), lorsqu'une température externe de soudure froide est écrite dans *CJ_temp* par l'opérateur ou par une liaison de schéma. La compensation n'est pas appliquée pour les mesures en millivolts. Le mode MANUEL permet d'ajuster *PV* au cours de la conduite.

PV. Sortie bloc (variable procédé). Lecture uniquement en mode Auto.

CJ_temp. Température de la soudure froide (°C, °F, K ou R). En mode INT_CJ, ce paramètre est en lecture uniquement et indique la température (compensée) de la sonde de soudure froide dans le bloc de liaison du module T112. En mode EXT_CJ, *CJ_temp* peut être écrit et lié à un bloc de liaison isothermique externe, qui maintient généralement la température. Lorsqu'une table de caractérisation ROM ou utilisateur a été sélectionnée (par *UserChar* ou *RomChar*), *PV* est compensé au niveau de la soudure froide par *CJ_temp*.

NOTA. Voir figure 2-2 (page 2-8 dans la section du bloc TCOUPLE) pour plus de détails sur l'implémentation de la compensation soudure froide.

HR, LR. (Echelle haute & basse de *PV* en unités physiques). *HR* et *LR* définissent deux points sur une échelle linéaire en unités physiques qui sont projetés sur les points *HR_in* et *LR_in* sur une échelle linéaire (millivolts). Ces deux échelles sont utilisées par le bloc pour dériver la sortie *PV* à partir du signal mesuré à l'entrée. L'équation de conversion équivalente est la suivante :

$$PV = LR + \left[\frac{\text{Input} - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

HAA, LAA. Limites d'alarmes absolues hautes & basses pour *PV*, avec une hystérésis intégrée de 0,5 % à l'intérieur des limites.

Filter (Filtre). Constante de temps du filtre du premier ordre (0 à 250 sec.)

Invert (Inversion). Inverse le sens du signal de sortie. Par ex., *Invert* étant VRAI, *PV* est au sommet de son échelle (*HR*), lorsque l'entrée mesurée est au bas de son échelle (*LR_in*).

RomChar (Caractérisation). (None/J/K/T/S/R/E/N/W/W5/W3/MoRe). Permet de sélectionner le type de caractérisation dans la bibliothèque ROM. Voir les détails table 2-2.

NOTA. Afin d'éviter tout conflit, la spécification d'une valeur dans *RomChar* autre que 'None' (Aucune) supprime automatiquement toute entrée dans le champ *UserChar*. De même, l'écriture d'un nom de repère dans *UserChar* supprime toute entrée *RomChar*.

UserChar (Caractérisation utilisateur). Permet de définir une table de linéarisation/caractérisation personnalisée de 16 points (x,y). Entrer le nom de repère du bloc UCHAR dans le schéma contenant la table en question.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes :

- Coupure de courant
- Type de module incorrect
- Interruption entrée
- Défaut matériel circuit (voir champ Etat).

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

■ **HiLevel, Lolevel (Alarme haute, Alarme basse).** Ces limites d'alarmes sont définies par les paramètres *HAA* et *LAA*. Il y a une bande d'hystérésis de 0,5 % sur chaque limite d'alarme, ce qui permet une transition nette entre la condition d'alarme et de non-alarme.

■ **Range (Echelle).** Cette alarme est déclenchée dès que *PV* dépasse une valeur de 5 % au-dessus ou en-dessous de *HR* ou *LR*.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° du site). Permet de définir l'emplacement à l'intérieur d'un T100 (1 à 16). Ce champ est actif dans une base de donnée T100, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *image* correspondants automatiquement copient les paramètres "réels".

Channel (Voie). Permet de définir le numéro de voie (1 à 8) dans le module T112 associé.

HR_in, LR_in. Echelle haute & basse du signal mesuré en millivolts. *HR_in* et *LR_in* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR* et *LR* sur une échelle linéaire *PV* en unités physiques. Voir les détails à la section *HR, LR* ci-dessus.

Break (Rupture). (haut/bas). Permet de définir la protection de rupture haute ou basse de l'échelle.

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-6.

NOTA. *BlockErr* est mis à 1 si la RAM est insuffisante pour exécuter le bloc. Le bloc n'est donc pas mis à jour, et d'autres blocs doivent être supprimés pour libérer de la RAM.

FREQIN : BLOC D'ENTREE FREQUENCE

Fonction du bloc

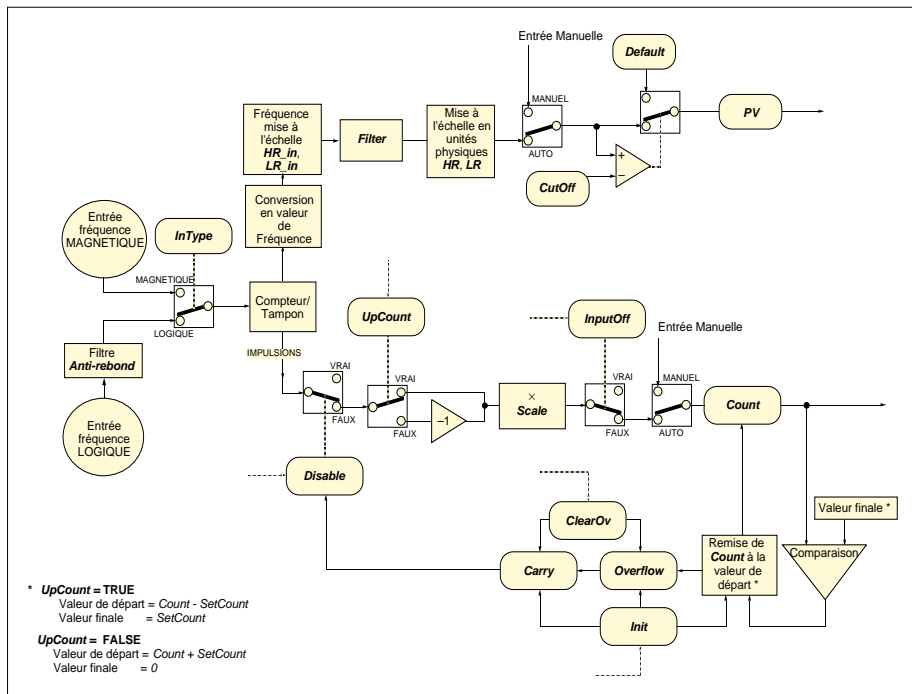


Figure 2-10 Schéma du bloc FREQIN

Voir le schéma de la figure 2-10. Le bloc d'entrée fréquence traite les signaux d'entrée à partir de l'extérieur par l'intermédiaire du module E/S d'entrée de fréquence T130. Les signaux magnétiques ou logiques peuvent être entrés dans le T130 (définis dans *InType*). Les impulsions d'entrée sont sauvegardées dans une mémoire tampon (buffer) et ensuite traitées et sorties de deux manières distinctes et indépendantes: une valeur de fréquence analogique (PV) et également comme un comptage (Count) d'impulsions analogiques mises à l'échelle. Voir le processus dans la partie supérieure et inférieure du schéma.

Valeur de fréquence. Les impulsions sauvegardées dans une mémoire tampon sont converties en unités de fréquence, traitées par filtre du premier ordre, et ensuite converties en unités physiques par les paramètres de mise à l'échelle *HR_in/LR_in* et *HR/LR*. Cette valeur analogique forme le paramètre *PV* et la sortie *PV*. Une valeur par défaut lui est automatiquement substituée, si la valeur calculée est en-dessous de la valeur seuil du paramètre *CutOff*. En ce qui concerne les fréquences inférieures à la fréquence de mise à jour du bloc, l'extrapolation est utilisée pour recalculer *PV* à chaque mise à jour, tandis que le bloc "attend" l'entrée suivante.

Comptage des impulsions mises à l'échelle. La partie comptage du bloc peut être configurée (ou reconfigurée à n'importe quel moment) comme un compteur ou un décompteur en utilisant le paramètre *Upcount*. Si *UpCount* est VRAI, à chaque mise à jour du bloc, le nombre d'impulsions (cycles) dans le tampon est multipliée par le paramètre d'unités physiques *Scale*, et le résultat ajouté au registre de comptage. Lorsque le comptage (*Count*) dépasse la "valeur finale" spécifiée par le paramètre *SetCount*, il revient à une "valeur de départ" égale à *Count - Setcount*, un drapeau de dépassement est verrouillé et un drapeau report est défini temporairement (généralement jusqu'à la mise à jour suivante du bloc) pour permettre le report en cascade des compteurs. Si *Upcount* est FAUX, les impulsions de mises à l'échelle sont soustraites de comptage (*Count*) jusqu'à ce qu'il tombe en-dessous de sa "valeur finale", cette fois égal à zéro. Le comptage revient alors à sa "valeur de départ", *Count + SetCount*, et les drapeaux *Overflow* (Dépassement) et *Carry* (Report) sont définis comme ci-dessus.

Des entrées d'initialisation (*Init*) et de remise à zéro du drapeau de dépassement (*ClearOv*) sont prévues, ainsi que deux formes de fonction d'invalidation : *Disable* qui arrête le compteur après qu'il a atteint une "valeur finale", et *InputOff* qui agit immédiatement.

NOTA. Limites de la mesure de fréquence. Les fréquences en dessous de 0,01Hz ne peuvent pas être mesurées par la partie fréquence du bloc et sont enregistrées comme nulles, bien que le compteur d'impulsions enregistre toutes les impulsions d'entrée, quelle que basse que soit la fréquence. La fréquence d'entrée maximale pour les deux fonctions du bloc est de 30 kHz.

Paramètres du bloc

Les symboles utilisés dans la table 2-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs "en-tête" à la page 2-3.

MODE. (AUTO/MANUEL). Mode de fonctionnement en cours.

PV. Sortie variable procédé. Fréquence mesurée (ou valeur par *défaut*) exprimée en unités physiques.

HR, LR. (Echelle haute & basse de *PV* en unités physiques). *HR* et *LR* définissent deux points sur une échelle linéaire en unités physiques qui sont projetés sur les points *HR_in* et *LR_in* sur une échelle linéaire de fréquence. Ces deux échelles sont utilisées par le bloc pour calculer le paramètre de sortie *PV* (en unités physiques) à partir de la fréquence mesurée de l'entrée (Hz). L'équation de conversion équivalente est :

$$PV = LR + \left[\frac{\text{Frequency} - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
PV	Sortie variable procédé	Eng	
HR, LR	Echelle haute & basse (unités phys.)	Eng	
Filter	Constante de temps filtre du premier ordre (sec.)	Eng	
Cutoff	Seuil bas PV	Eng	
Default	Valeur par défaut PV (PV < CutOff)	Eng	
Count	Comptage impulsions mises à l'échelle	Eng	
Carry	Drapeau report	V/F	
Overflow	Drapeau dépassement	V/F	
SetCount	Valeur de comptage début/fin	Eng	
Scale	Facteur d'échelle des impulsions	Eng	
Control	Contrôle	Champ binaire	
Init	Initialisation compteur	V/F	
Disable	Arrêt compteur valeur finale	V/F	
ClearOv	Effacement drapeaux Dépasst. & report	V/F	
UpCount	Sélection comptage/décomptage	V/F	
InputOff	Arrêt immédiat comptage	V/F	
		0 0 0 0 0 0 0	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts Module Txxx/alim. transmetteur, etc.	V/F	
OpenCct	Capteur en circuit ouvert	V/F	
CutOff	Fréquence mesurée inférieure à valeur de coupure	V/F	
OutOfRange	Condition de surcharge de la sortie	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Emplacement du T130 dans le T100	Entier	
HR_in, LR_in	Echelle haute et basse Entrée fréquence (Hz)	Eng	
Intype	Type d'entrée	Menu	
Thresh	Seuil déclenchement tension (Entrée logique uniquement)		Menu
Burden	Valeur résistance de charge (Entrée logique uniquement)		Menu
Debounce	Sélection filtre anti-rebond (Entrée logique uniquement)		Menu
PSU	Tension alimentation du capteur	Menu	
Status	Etat communicat./matériel	(A)BCD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module T130 manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
IPOpenCC	Capteur logique en circuit ouvert	V/F — 1	C
NoPulse	PV a une valeur extrapolée	V/F — 2	
PSUShort	Surcharge alim. transmetteur	V/F — 4	
	(non utilisé)	8	
BlockErr	Données RAM corrompues (drapeau transitoire)	V/F — 1	B
		2	
		4	
		8	

Table 2-7 Paramètres du bloc FREQUIN

Filter (Filtre). Constante de temps (secondes) d'un filtre du premier ordre appliquée à la valeur de fréquence mesurée pour éliminer le bruit du signal.

Cutoff (Coupure). Valeur de seuil bas pour *PV*. Si *PV* est en-dessous de *CutOff*, *PV* automatiquement adopte la valeur spécifiée par le paramètre par défaut.

Default (Défaut). Valeur par défaut de *PV*, automatiquement adoptée si *PV* descend en-dessous de la valeur spécifiée par le paramètre *CutOff*.

Count (Total). Sortie (analogique) de compteur mise à l'échelle. Renvoie au nombre d'impulsions (une par cycle) entrée dans le bloc multiplié par le paramètre échelle. Le comptage peut s'incrémenter ou décrémenter avec les impulsions, suivant l'état de *UpCount*, et se remet automatiquement à une "valeur de départ" après avoir atteint la "valeur finale", spécifiée dans le paramètre *SetCount*.

Carry (Retenue). Passe à VRAI (TRUE) lorsque le total de comptage a atteint sa "valeur finale" (définie par *UpCount* et *SetCount*) et a alors été remis à la "valeur de départ". Lorsque le comptage s'est éloigné de sa "valeur de départ", *Report* (Carry) automatiquement repasse à FAUX (FALSE). L'initialisation du bloc par *Init* remet également *Carry* à sa valeur de départ.

Overflow (Dépassement). Est verrouillé sur VRAI, lorsque le compteur a atteint sa "valeur finale" et a été remis à sa "valeur de départ". Le drapeau est effacé, lorsque *Init* est VRAI ou lorsque *ClearOv* est VRAI.

SetCount (Valeur finale). Permet de définir les "valeurs de départ" et les "valeurs de fin" du paramètre *Count*. Lorsque le bloc est configuré comme un compteur (*UpCount* VRAI), la "valeur finale" est égale à *SetCount* et la "valeur de départ" est égale à *Count - SetCount*. Il est à noter que cette "valeur de départ" (*Count - SetCount*) n'est pas nécessairement égale à zéro, dans la mesure où le total dans *Count* - augmenté du dernier lot d'impulsions collectées dans le tampon - est généralement supérieur à *SetCount*, lorsque la remise à la valeur de départ est déclenchée. Lorsque le bloc est configuré comme un décompteur (*UpCount* FAUX), la "valeur finale" est égale à zéro et la valeur de départ est égale à *Count + SetCount*.

Scale (Echelle). Permet de spécifier la valeur ajoutée à (ou soustraite de) comptage (*Count*) pour chaque impulsion détectée. Par ex. Au cas où *UpCount* est VRAI, si 10 impulsions sont détectées entre les mises à jour et échelle (*Scale*) = 2,5, alors 25,0 est ajouté à *Count*.

Control (Contrôle). Champ binaire de cinq paramètres de contrôle qui peut être relié aux entrées logiques du schéma de boucles. Il est à noter que ces bits de contrôle ne s'appliquent qu'aux fonctions de comptage du bloc FREQUIN, et n'ont aucun effet sur la fonction de fréquence mesurée.

■ **Init (Initialisation).** Une valeur vraie initialise le compteur, c'est à dire redéfinit *Carry*, *Overflow* et *Count*. *Init* proprement dit repasse à FAUX.

■ **Disable (Invalidation).** Lorsque Invalidation (*Disable*) est VRAI, le compteur s'arrête - mais uniquement après que le total de comptage a atteint sa "valeur finale" et

est revenu à sa “valeur de départ”, et que les drapeaux *Carry* et *Overflow* ont été mis à 1. Dans ce cas, les impulsions d’entrée ne sont pas ajoutées à *Count*, et les drapeaux sont figés, jusqu’à ce que *Disable* passe à FAUX. Il faut noter que la remise à zéro de *Disable*, ne remet pas à zéro le drapeau *Overflow*.

- **ClearOv (Effacement de la retenue).** Une entrée vraie remet à zéro les drapeaux *Overflow* et *Carry* ; ensuite, *ClearOv* repasse automatiquement à FAUX.
- **UpCount (Mode de comptage).** VRAI permet de sélectionner une fonction de comptage : les impulsions mises à l’échelle *s’ajoutent* au total de comptage. FAUX permet de sélectionner une fonction de *décomptage* : les impulsions mises à l’échelle sont *soustraites* du total de comptage. Il faut noter que la valeur de comptage n’est pas ré-initialisée au cours de la commutation de *UpCount*, de sorte que la fonction comptage peut être modifiée à mi-comptage, avant que la “valeur finale” ne soit atteinte.
- **InputOff (Blocage de l’entrée).** VRAI empêche les impulsions de modifier immédiatement le total de comptage, à l’inverse de *Disable*. Les autres fonctions de régulations ne sont pas affectées (par ex. *Init*).

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes :
 - Coupure de courant
 - Type de module incorrect
 - Défaut alim. transmetteur
 - Défaut matériel circuit (voir champ *Etat*).

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

- **OpenCct.** Capteur en circuit ouvert.
- **CutOff (Coupure).** Fréquence mesurée en-dessous de la valeur du paramètre *CutOff*. Cette alarme a une hystérésis de 0,5 %.
- **OutRange (Hors échelle).** Cette alarme est déclenchée, dès que *PV* dépasse une valeur de 5 % au-dessus ou en-dessous de *HR* ou *LR*. *Outrange* est appliquée à la valeur de fréquence avant l’action du filtre.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° du site). Permet de définir l’emplacement à l’intérieur d’un T100 (1 à 16). Ce champ est actif dans une base de donnée T100, et lie effectivement le bloc à un site matériel. Une fois qu’un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du

site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *image* correspondants copient automatiquement les paramètres “réels”.

HR_in, LR_in. Echelle haute & basse de la fréquence mesurée en Hz. *HR_in* et *LR_in* permettent de définir deux points sur une échelle linéaire de fréquence qui sont projetés sur les points *HR* et *LR* sur une échelle linéaire en unités physiques. Voir les détails dans la section *HR, LR* ci-dessus.

InType (Type d'entrée). (Magnétique/logique). Permet de spécifier le type d'entrée connectée au module E/S T130 associé au bloc FREQUIN.

Thresh (Seuil). (10 Volts/3,5 Volts/1,6 Volts/0,4 Volts). Permet de définir le seuil de déclenchement de la tension pour un capteur logique (c'est à dire définit l'état HAUT comme étant $\approx 2 \times$ le seuil).

Burden (Charge). (Aucune/5k Ohms/1k Ohms/50 Ohms). Permet de sélectionner la valeur de résistance de charge pour un capteur logique.

Debounce (Anti-rebond). (OFF/ON). Sélection filtre anti-rebond pour des entrées de contact sans tension (*InType* = logique). *Debounce* étant sur ON, les impulsions de moins de 25 ms sont filtrées. Le maximum d'impulsions passant par le filtre est de 20 par seconde.

PSU (Alimentation). (OFF/8 Volts/12 Volts/24 Volts). Paramètre de sélection de tension pour l'alimentation programmable, ce qui permet de l'utiliser avec des pré-amplificateurs de courant, des détecteurs de proximité et des contacts sans tension.

Status (Etat). Champ binaire hexadécimal de huit paramètres d'état communicat./matériel qui peuvent être connectés aux sorties logiques du schéma de boucles. Voir les détails de la table 2-7.

NOTA. *BlockErr* est mis à 1 si la RAM est insuffisante pour exécuter le bloc. Le bloc n'est donc pas mis à jour, et d'autres blocs doivent être supprimés pour libérer de la RAM.

AN_IP : BLOC D'ENTREE ANALOGIQUE

Fonction du bloc

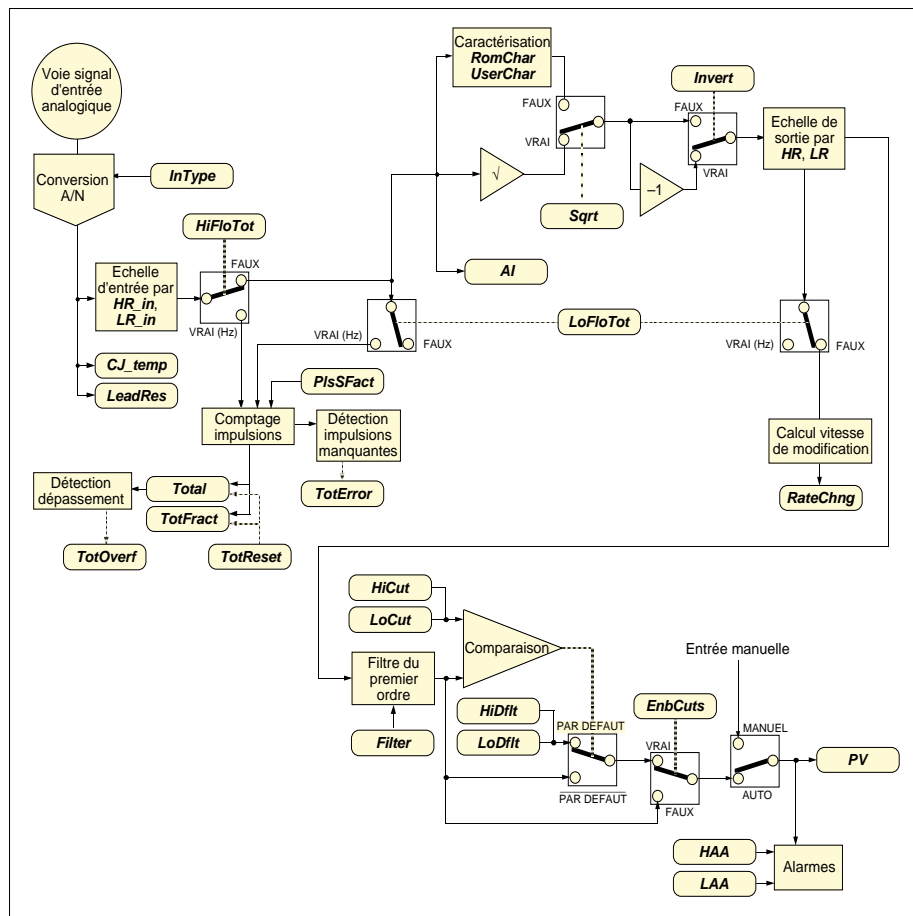


Figure 2-11 Schéma du bloc AN_IP

Se reporter au schéma de la figure 2-11. Le bloc AN_IP convertit les entrées de mesure de tension, de fréquence ou de résistance des instruments de la série T600 en valeurs à point flottant en unités physiques. Le bloc dispose en outre de fonctions de conditionnement des signaux, d'étalonnage, de filtrage et d'alarme.

Le bloc peut être utilisé en mode manuel, au cours duquel la variable procédé PV peut être forcée à une valeur spécifiée pour faire des essais. Le fonctionnement normal de PV reprend, lorsque le bloc est en mode auto.

Totalisation des impulsions

Lorsque la fonction fréquence est sélectionnée (*InType* = Hz) et les options *LoFloTot* ou *HiFloTot* sont VRAIES, le bloc compte, met à l'échelle et totalise les impulsions d'entrée. La définition de *LoFloTot* permet de totaliser avec précision des débits d'impulsion de 1kHz maximum et fournit également des sorties de mesure de fréquence non-traitées et traitées (*AI* et *PV* respectivement). Lorsque *HiFloTot* est mis à 1, les sorties de fréquence ne sont pas fournies, mais le bloc peut traiter des débits d'impulsion de 30 kHz maximum. Lorsque les bits des deux options sont mis sur FAUX, le comptage d'impulsions est invalidé, ce qui permet d'effectuer des mesures de fréquence et de vitesse de changement de fréquence (voir figure 2-11).

Détection/protection circuit ouvert

Certains modules d'E/S prennent en compte la détection de rupture sur leurs entrées analogiques, tandis que d'autres ne prennent en compte que la protection de rupture. Voir les détails dans la documentation spécifique du module d'E/S. Le bloc AN_IP peut fonctionner soit en mode de détection de rupture ou en mode de protection de rupture, mais pas dans les deux modes à la fois. La protection de rupture est sélectionnée, lorsque *OptionsBrkDetect* est VRAI; FAUX permet de sélectionner la détection de rupture. Si vous tentez d'utiliser un mode de rupture, qui n'est pas pris en compte par une voie E/S particulière, le bit *Status.BadBrk* est activé.

Détection de rupture. Les modules prenant en compte la détection de rupture peuvent détecter une rupture au niveau de l'entrée avant toute mauvaise lecture. Dès qu'une rupture est détectée, le bit d'état *BrkDctd* est mis à 1. Le bit d'alarme *OCctdel* est également activé, sous réserve de la temporisation spécifiée dans le champ *Delay*. La temporisation peut être soit appliquée au début de la rupture (c'est à dire que le passage à VRAI du bit d'alarme est retardé) ou à la fin de la rupture (c'est à dire que le passage du bit d'alarme à FAUX est retardé), ou les deux à la fois, en utilisant respectivement les bits *Options.OCdelSt* et *OCdelEnd*.

Le bit *Options.InitFilt*, si VRAI, permet d'initialiser le filtre du premier ordre du bloc, si *OCctdel* ou *BrkDctd* est VRAI, ce qui signifie que de mauvaises lectures déjà filtrées ne corrompent pas les bonnes lectures ultérieures. Il faut noter que l'alarme *OCctdel* n'a pas besoin d'être validée pour qu'*InitFilt* agisse.

L'effet sur *PV* lorsqu'une rupture est détectée dépend de la définition de deux bits *Options*. *HoldDect*, si VRAI, permet de maintenir la dernière valeur correcte de *PV*. Si FAUX, *PV* passe soit à *HR* (si *BreakUp* est VRAI) ou à *LR* (si *BreakUp* FAUX). Ces mouvements sont inversés, lorsque le bit *Options.Invert* est VRAI.

Protection de rupture. Dans les modules E/S ne prenant en compte que la protection de rupture, lorsqu'une rupture se produit, la valeur de *PV* dérive soit vers le haut (*BreakUp* VRAI) ou vers le bas (*BreakUp* FAUX). Le sens est inversé si le bit *Options.Invert* est VRAI.

Exécution des blocs d'entrée

Les blocs d'entrée sont normalement exécutés au début d'une tâche utilisateur, avant d'autres blocs dans la tâche. C'est pour cette raison que les liaisons à un bloc d'entrée à partir d'autres blocs dans la même tâche utilisateur - par ex. contre-réaction à partir d'un bloc de sortie - transmettent les valeurs provenant de l'exécution *précédente* de la tâche utilisateur. Les liaisons d'entrée à partir d'autres tâches utilisateur, sur ce noeud ou d'autres noeuds, sont toujours exécutées avant l'exécution de la tâche utilisateur en cours, et sont garanties provenir d'une exécution complète de la tâche utilisateur source.

Paramètres du bloc

Les symboles utilisés dans la table 2-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement	Menu	
PV	Entrée traitée	Eng	
RateChng	Vitesse de changement de la valeur PV non-filtrée	Eng/Sec	
HR, LR	Echelle haute & basse de PV	Eng	
HAA, LAA	Seuils d'alarme absolue haute & basse	Eng	
Filter	Constante de temps du filtre du premier ordre	Secs	
HiCut	Seuil de coupure haut	Eng	
HiDflt	Valeur PV par défaut, si PV > HiCut	Eng	
LoCut	Seuil de coupure bas	Eng	
LoDflt	Valeur PV par défaut, si PV < LoCut	Eng	
RomChar	Caractérisation fixe	Menu	
UserChar	Caractérisation utilisateur	Nom du bloc	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut matériel E/S	V/F	
HiLevel	En alarme haute	V/F	
LoLevel	En alarme basse	V/F	
OutOfRange	PV hors échelle LR-HR (>5%)	V/F	
OCctdel	Temporisation OpenCct	V/F	
UCharErr	Anomalie mémoire dans caract. utilisateur	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	1-8	
Channel	Spécifie le numéro de voie dans le module E/S	1-n	
InType	Spécifie le type d'entrée	Menu	
HR_in, LR_in	Equivalents du signal HR, LR		
AI	Entrée non-traitée		
CJ_temp	Température soudure froide		
LeadRes	Résistance du cordon	Ohms	
Delay	Temporisation avant déclenchemt. alarme OCctdel	Secs	

suite...

...suite








Paramètre	Fonction	Unités	Etat
Options		ABCD Hex	 
Invert	Inverseur signal d'entrée	V/F — 1	D
Sqrt	Signal d'entrée racine carrée	V/F — 2	
InitFilt	Initialisation filtre si OCctdel OU BrkDtctd VRAI	V/F — 4	
OCdelSt	Application temporisation C/O au début	V/F — 8	
OCdelEnd	Application temporisation C/O à la fin	V/F — 1	C
BrkDetct	Détection rupture	V/F — 2	
BreakUp	PV à HR (VRAI), ou LR (FAUX) sur rupture	V/F — 4	
TAbsolut	Base de température (K ou R)	V/F — 8	
DsblFilt	Invalidation filtre niveau bas dans module E/S	V/F — 1	B
HoldDect	Sur rupture : VRAI, PV maintenu ; FAUX, PV à l'échelle	V/F — 2	
EnbCuts	Validation défauts de coupure haute & basse	V/F — 4	
LoFloTot	Validation totalisation ($\leq 1000\text{Hz}$)	V/F — 8	
HiFloTot	Validation totalisation ($> 1000\text{Hz}$)	V/F — 1	A
TotRset	Mise à 0.0 de Total.TotFract	V/F — 2	
TotOverf	Dépassement du total	V/F — 4	
		8	
Status		ABCD Hex	 
Reset	Remise à 0 module E/S	V/F — 1	D
Missing	Aucun module E/S sur site	V/F — 2	
CommsErr	Défaut communication module E/S	V/F — 4	
BadType	Voie erronée sur module E/S	V/F — 8	
TotError	Impulsions d'entrée manquées (invalidation total)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
PSUShort	Surcharge alim. transmetteur	V/F — 4	
BadRef	Défaut tension de référence	V/F — 8	
BlockErr	Données bloc corrompues (RAM insuffisante)	V/F — 1	B
BrkDtctd	VRAI si détection circuit ouvert ou hors échelle	V/F — 2	
BadBrk	Config. "rupture" non prise en compte par mat. E/S	V/F — 4	
BadRange	Erreur sélection échelle	V/F — 8	
BadSetup	Config. incorrecte pour module E/S sélectionné	V/F — 1	A
HwFault	Défaut dans module E/S	V/F — 2	
TmStmpOv	Dépassement horodatage (invalidation RateChng)	V/F — 4	
BrkWarn	Rupture partielle RTD	V/F — 8	
Total	Partie intégrale comptage impuls. mises à l'échelle	Entier	
TotFract	Partie fractionnelle comptage impuls. mises à l'échelle		
PlsSFact	Facteur échelle impulsions (diviseur)		

Table 2-8 Paramètres du bloc AN_IP

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

MODE. (AUTO/MANUAL/CALIB). Mode de fonctionnement en cours. MANUAL (MANUEL) permet de forcer *PV* à prendre une valeur spécifiée au cours de la conduite. Le mode CALIB (Etalonnage) ne peut pas être sélectionné directement, mais est “forcé” lorsqu’un bloc AI_CALIB est rattaché au bloc AN_IP. Le mode CALIB est similaire au mode AUTO, sauf que tous les champs qui peuvent affecter le calcul de *AI* sont en lecture uniquement (*InType*, *SiteNo*, *Channel*, *HR_in*, *LR_in*). Voir la description du bloc AI_CALIB à la page 2-62 du présent chapitre.

PV. Variable procédé (entrée traitée), c’est à dire sortie du bloc. Lecture uniquement en mode auto.

RateChng. Vitesse de changement de la valeur *PV* non-filtrée (dPV/dt) calculée entre l’échantillon utilisé par l’exécution précédente de la tâche et l’échantillon utilisé par l’exécution en cours de la tâche. dPV/dt est particulièrement utile dans les applications de contrôle de perte de poids. Il faut noter que si *TmStmpOv* - dans le paramètre *Status* - est VRAI, dPV/dt ne peut pas être calculé et *Ratechng* maintient sa dernière valeur correcte.

NOTA. Mettre *Options.DsblFilt* à VRAI, si vous utilisez *Ratechng*. Le filtre de niveau bas est ainsi invalidé dans le convertisseur A/N (non représenté sur la figure 2-11).

HR, LR. Echelle haute et basse de *PV* en unités physiques. *HR* et *LR* permettent de définir deux points sur une échelle linéaire en unités physiques qui sont normalement projetés respectivement sur les points *HR_in* et *LR_in* sur une échelle linéaire (millivolts). Ces deux échelles sont utilisées par le bloc pour calculer la sortie *PV* à partir du signal mesuré à l’entrée. L’équation de conversion équivalente est la suivante:

$$PV = LR + \left[\frac{\text{Input} - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

NOTA. Le bit *Options.Invert* affecte le calcul de *PV*. Voir les *Options* ci-dessous.

HAA, LAA. Seuils d’alarmes absolues hautes & basses appliqués à *PV* avec une hystérésis intégrée de 0,5 % à la sortie de l’état d’alarme. Si un seuil est dépassé, le bit d’alarme approprié (*HiLevel* ou *LoLevel*) est mis à 1.

Filter (Filtre). Permet de spécifier la constante de temps (0 à 250 secondes) d’un filtre du premier ordre simple appliqué à la valeur dans *PV*. Si *Filter* est égal à zéro, le filtre est invalidé.

HiCut. Permet de spécifier une valeur de coupure haute pour *PV*, dans ce cas, *PV* passe à *HIDflt* (à condition que *Options.EnbCuts* soit VRAI).

HiDflt. Valeur par défaut adoptée par *PV* lorsque la valeur calculée est supérieure à *HiCut*. Voir *HiCut*.

LoCut. Permet de spécifier une valeur de coupure basse pour *PV*, dans ce cas, *PV* passe à *LoDflt* (à condition que *Options.EnbCuts* soit VRAI).

LoDflt. Valeur par défaut adoptée par *PV* lorsque la valeur calculée est inférieure à *LoCut*. Voir *LoCut*.

RomChar. (Aucun/J/K/T/S/R/E/B/N/W/W5/W3/MoRe/PRT/CU10). La valeur dans *AI* peut être traitée par l'un des trois procédés de caractérisation qui s'excluent mutuellement, sélectionnés par l'intermédiaire de *RomChar*, *UserChar* et *Sqrt*. *RomChar* sélectionne un type de caractérisation dans la bibliothèque ROM. Voir les détails dans la table 2-2. Le bloc AN_IP fait automatiquement correspondre les tables ROM à la base de température sélectionnée du bloc.

NOTA. Afin d'éviter tout conflit, la spécification d'une valeur dans *RomChar* autre que "None" (Aucune) supprime automatiquement toute entrée dans le champ *UserChar*. De même, l'écriture d'un nom de repère dans *UserChar* supprime toute entrée *RomChar*.

UserChar. Permet de spécifier le nom d'un bloc UCHAR définissant un table de linéarisation/caractérisation personnalisée de 16 points (*x,y*) à appliquer à *AI*. (Voir nota ci-dessus). Le bloc AN_IP n'applique pas de correction de base de température à la table UCHAR, celle-ci doit donc être définie en unités qui correspondent à la base de température prédominante.

NOTA. Une anomalie mémoire détectée dans le bloc UCHAR déclenche le bit d'alarme *UCharErr* du bloc AN_IP.

Alarms (Alarmes). Voir la description générale du champ alarme à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Hardware (Matériel).** Une alarme matériel est déclenchée en cas de défaut du matériel E/S.
- **HiLevel, LoLevel (Niveau haut, Niveau bas).** Ces seuils d'alarme sont définis par les paramètres *HAA* et *LAA*. Une bande d'hystérésis de 0,5 % appliquée en fin d'alarme permet des transitions nettes entre le passage en alarme et la fin de l'alarme.
- **OutOfRange (Hors échelle).** Cette alarme est déclenchée dès que *PV* atteint une valeur supérieure ou inférieure de 5 % à *HR* ou *LR*. Une bande d'hystérésis de 0,5 % est appliquée à chaque valeur de l'échelle.
- **OCctdel.** "Temporisation circuit ouvert". Lorsque *OCdelSt* (Options) est VRAI, cette alarme se déclenche après une temporisation définie par le paramètre *Delay* si un circuit ouvert est détecté dans l'entrée du bloc. Lorsque *OCdelSt* est FAUX, l'alarme se déclenche sans temporisation. De même, lorsque *OCdelEnd* (Options) est VRAI,

OCdel la remet à zéro quelques secondes après la restauration de l'entrée circuit ouvert. Lorsque *OCdelEnd* est FAUX, l'alarme est remise à zéro immédiatement. (Voir également *Delay*).

- **UCharErr.** Cette alarme est déclenchée si une anomalie mémoire est détectée dans le fichier de caractérisation utilisateur défini dans le paramètre *UserChar*.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

SiteNo (N° de site). Permet de spécifier l'adresse (1 à 8) sur le bus série interne des instruments de la série T600, où le module E/S correspondant à ce bloc AN_IP est situé. Noter qu'un *SiteNo* incorrect met à 1 le bit d'état *Missing* (manquant).

Channel (Voie). Permet de définir le numéro de voie (1 à *n*) dans le module E/S défini dans le paramètre *SiteNo*, auquel ce bloc AN_IP est rattaché. Le nombre de voies disponibles (*n*) dépend du type de module E/S. Noter qu'un *Channel* incorrect met à 1 le bit d'état *BadType*.

InType (Type d'entrée). (Volts/mV_Int/mV_Ext/4 WIRE/3 WIRE/2 WIRE/Hz). Permet de définir le type d'entrée. *Volts* permet de configurer le bloc comme une entrée de tension de haut niveau avec une échelle maximale de -10 à + 10 V. En ce qui concerne les entrées en millivolt, *mV_Int* et *mV_Ext* permettent de sélectionner respectivement la compensation de soudure froide interne et externe. Pour les mesures directes en millivolts, il faut spécifier CJC externe et mettre le paramètre *CJ_temp* à zéro. L'échelle mV maximale est de -100 à + 100 mV. Les options 4 WIRE (4 FILS), 3 WIRE (3 FILS) et 2 WIRE (2 FILS) permettent de prendre en compte respectivement des mesures de résistance à 4, 3 et 2 fils (Ohms) dans une échelle maximale de 0 à 1000 Ω . *Hz* permet de sélectionner le mode d'entrée de fréquence dans une plage maximale de 0 à 30 kHz.

Les échelles d'entrée indiquées ci-dessus ne représentent que les maximum des blocs - consulter la documentation des modules E/S pour obtenir les seuils spécifiques.

NOTA. Lorsque certaines options ne sont pas disponibles sur certaines voies, le bit d'état *BadSetup* est mis à 1.

HR_in, LR_in. Echelle haute & basse du signal mesuré de l'entrée dans les unités adaptées à la sélection en cours d'*InType* (V, mV, W, Hz). *HR_in* et *LR_in* permettent de définir deux points sur une échelle linéaire, qui sont projetés respectivement sur les points *HR* et *LR* sur une échelle linéaire *PV* en unités physiques. Lorsque l'entrée réelle est à la valeur *HR_in*, *PV* est égal à *HR*; lorsque l'entrée réelle est à la valeur *LR_in*, *PV* est égal à *LR*. Voir les détails dans le paragraphe *HR, LR* ci-dessus.

NOTA. Chaque fois que *Volts* est sélectionné pour *InType* ou qu'aucune caractérisation n'est appliquée, *HR_in* et *LR_in* peuvent être définis indépendamment de *HR* et *LR*. Dans tous les autres cas, le bloc calcule automatiquement *HR_in* et *LR_in* à partir de *HR* et *LR* par l'intermédiaire de la

caractérisation. Le bloc compare les valeurs entrées de *HR_in* et *LR_in* à la liste des échelles matériel disponibles prises en compte par le module E/S et sélectionne l'échelle qui convient le mieux pour optimiser la résolution de l'entrée. Les valeurs exactes de *HR_in* et *LR_in* sont mises à l'échelle par le logiciel.

AI. La valeur d'entrée non-traitée, à partir de laquelle *PV* est calculé.

NOTA. Le bloc calcule normalement *AI* à partir de la moyenne des quatre derniers échantillons d'entrée. Ce calcul de moyenne, ainsi que le filtrage interne fourni par certains modules E/S améliore la résolution, mais réduit le temps de réponse. Si le temps de réponse est plus important, le filtre matériel du module E/S et le calcul de moyenne de l'entrée du bloc AN_IP peuvent être invalidés, en mettant à VRAI *Options.DsblFilt*.

CJ_temp (Température soudure froide). Température de la soudure froide, uniquement applicable dans les modes d'entrée "mV". Lorsque *mV_Int* est sélectionné pour *InType*, *CJ_temp* est en lecture uniquement et indique la température du bloc connecteur du module E/S, mesurée par sa sonde intégrée (si présente). Lorsque *mV_Ext* est sélectionné, vous pouvez écrire une température de soudure froide connue dans *CJ_temp* ou entrer une valeur par l'intermédiaire d'une liaison d'un schéma de boucles. Si *CJ_temp* est mis à zéro, la compensation de soudure froide est invalidée, et le bloc peut fonctionner comme une simple entrée mV.

NOTA. Toutes les températures sont affichées, et doivent être entrées dans les unités spécifiées par le paramètre *TAbsolut* et le paramètre *IP_type* du bloc T600 correspondant, c'est à dire °C, °F, K ou R.

NOTA. Lorsque certaines options ne sont pas disponibles sur certaines voies, le bit d'état *BadSetup* est mis à 1.

LeadRes. Permet de spécifier la valeur de résistance du cordon (Ω) pour les mesures à 2 fils uniquement. En mode à 2 fils, *LeadRes* est en lecture-écriture et une valeur mesurée auparavant doit être inscrite dans ce champ. En ce qui concerne les RTD utilisés en modes 3 ou 4 fils, *LeadRes* est en lecture uniquement et est rétro-calculé par le bloc. Donc, s'il devait se produire une "rupture partielle" en fonctionnement à 2 fils, l'entrée continue de fonctionner (et *Status.BrkWarn* est activé).

Delay (Temporisation). Lorsque *OCdelSt* est VRAI, *Delay* spécifie la temporisation (sec), avant que l'alarme *OCdel* ne soit déclenchée après détection d'un circuit ouvert dans le signal d'entrée, et également - lorsque *OCdelEnd* est VRAI - la temporisation avant que l'alarme ne soit remise à zéro après restauration de l'entrée. Lorsque FAUX est sélectionné pour *OCdelSt* et/ou *OCdelEnd*, ces temporisations sont annulées séparément. La figure 2-12 montre schématiquement la mise en oeuvre de *Delay*. La fonction *Delay* est utile, lorsque des interruptions à court terme du signal d'entrée doivent être ignorées "bruit" et également lorsque la restauration d'une entrée à circuit ouvert risque d'être "bruyante".

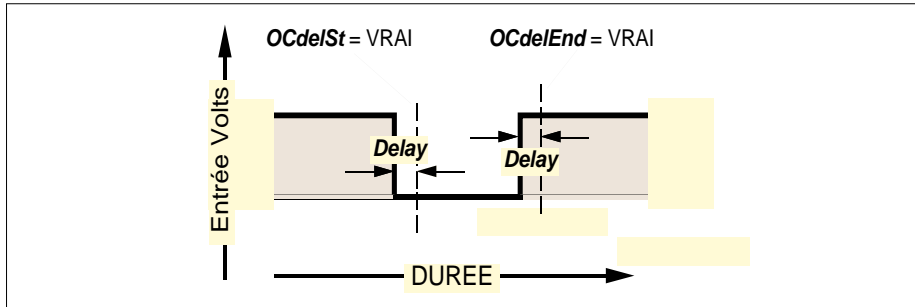


Figure 2-12 Mise en oeuvre du paramètre Delay (schéma de principe)

Options. Champ binaire permettant de sélectionner une variété d'options de traitement des signaux. Les informations suivantes sont données en plus de récapitulatif de la table 2-8.

- **Invert (Inversion).** Si VRAI, *HR* est projeté sur *LR_in* et *LR* sur *HR_in*, c'est à dire que le signal d'entrée est inversé. Voir ci-dessus, le paragraphe *HR*, *LR*.
- **Sqrt.** Si VRAI, la caractérisation de racine carrée est appliquée à *AI*, c'est à dire

$$\text{Résultat} = \sqrt{\frac{AI - LR_in}{HR_in - LR_in}} \times (HR_in - LR_in) + LR_in$$

- **TAbsolut.** Utilisé conjointement avec le champ *IP_Type.Imperial* du bloc T600 associé pour sélectionner les unités de température prédominantes du bloc AN_IP. Le bloc T600 permet de sélectionner globalement le système impérial (°F/R) ou SI (°C/K), mais chaque bloc AN_IP peut définir soit des unités de température absolues (K/R) ou relatives (°C/°F) dans le cadre de ce système. Les unités de température s'appliquent à *CJ_temp* et à toute caractérisation *RomChar* (c'est à dire aux champs fixes). Aucune correction de base de température n'est appliquée aux caractérisations définies par l'utilisateur dans les blocs UCHAR.
- **DsbIFilt.** VRAI invalide le filtre passe-bas dans le convertisseur A/N du module E/S - il ne s'agit pas du filtre du premier ordre du bloc AN_IP. Les caractéristiques des modules E/S varient, et pas toutes les entrées analogiques prennent en compte les filtres passe-bas. Voir les détails dans la documentation spécifique au module.
- **LoFlotot.** Permet de sélectionner la totalisation de débit "basse", c'est à dire que les débits d'entrée d'impulsions ne dépassent pas 1 kHz. Applicable uniquement lorsque *InType* = Hz. Si VRAI, le bloc compte les impulsions d'entrée, met à l'échelle le comptage par *PlsFact*, et sauvegarde le total cumulé dans *Total* plus *TotFract*. *AI* et *PV* indiquent les valeurs de fréquence correctes, mais *RateChng* est invalidé.

■ **HiFloTot.** Permet de sélectionner la totalisation de débit “haute”, c’est à dire que les débits d’entrée d’impulsion vont de 1 kHz à 30 kHz environ. Applicable uniquement lorsque *InType* = Hz. Si VRAI, le bloc compte et totalise les impulsions d’entrée comme pour *LoFloTot*, mais *AI*, *PV*, *RateChng*, et le reste des fonctions du bloc sont invalidées. Voir le schéma de la figure 2-11. Noter que si VRAI, *HiFloTot* annule *LoFloTot*.

■ **TotOverf.** Passe à VRAI, si Total dépasse 99 999 999.

Status (Etat). Champ binaire indiquant des défauts de communication et de matériel. Les informations suivantes sont données en plus du récapitulatif de la table 2-8.

■ **TotError.** Activé lorsque le bloc détecte que des impulsions d’entrée manquent, et donc que *Total* n’est pas incrémenté correctement. Noter que le drapeau *TotError* est non-verrouillant et doit donc être repris par le schéma de boucles, si nécessaire.

■ **BlockErr.** Est mis à 1 si la RAM est insuffisante pour exécuter le bloc. Le bloc n’est donc pas mis à jour, et d’autres blocs doivent être supprimés pour libérer de la RAM.

Total. Partie intégrale du comptage total d’impulsions mis à l’échelle.

TotFract. Partie fractionnelle comptage total d’impulsions mis à l’échelle.

PlsFact. Facteur de mise à l’échelle des impulsions. Nombre par lequel la fréquence des impulsions d’entrée (Hz) est divisée pour produire le comptage d’impulsions mis à l’échelle, *Total* plus *TotFract*. Par ex., pour 1000 impulsions entrées par seconde, *PlsFact* = 100.000, *Total* est incrémenté de 10 unités par seconde.

AN_OUT : BLOC DE SORTIE ANALOGIQUE

Fonction du bloc

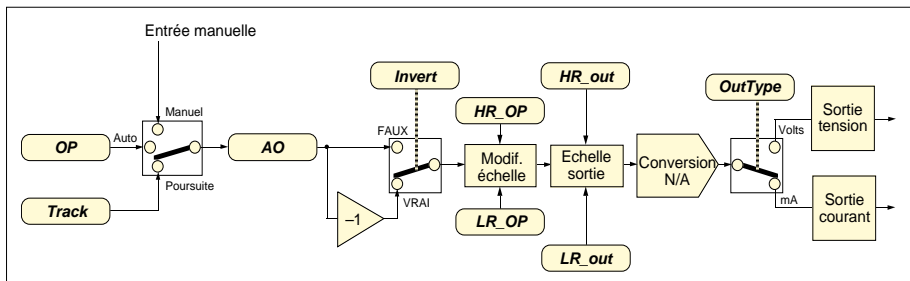


Figure 2-13 Schéma du bloc AN_OUT

Se reporter au schéma de la figure 2-13. Le bloc AN_OUT convertit les variables analogiques mises à l'échelle (nombres à point flottant) d'un schéma de boucles exécuté dans un régulateur de la série T600 en sorties de tension ou de courant (suivant les possibilités du matériel E/S). Le bloc dispose en outre des modes Auto/Manuel/Poursuite, d'inversion et d'alarmes.

Les sorties des différents instruments peuvent être interconnectées dans des configurations redondantes pour améliorer le niveau d'intégrité des systèmes. Dans ces configurations, une sortie "maître" contrôle normalement les installations; si l'instrument maître tombe en panne, un "esclave" prend automatiquement le relais. Voir les détails dans la documentation spécifique au matériel.

Paramètres du bloc

Les symboles utilisés dans la table 2-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs "en-tête" à la page 2-3.

MODE. (AUTO/MANUAL/TRACK/CALIB). Mode de fonctionnement en cours.

AUTO est le mode de fonctionnement normal du bloc. MANUAL (MANUEL) permet de forcer AO à prendre une valeur spécifiée au cours de la conduite. En mode TRACK (POURSUITE) (*Seltrack* VRAI), AO poursuit la valeur dans le paramètre *Track*. Le mode CALIB (Étalonnage) ne peut pas être sélectionné directement, mais est "forcé" lorsqu'un bloc AO_CALIB est rattaché au bloc AN_OUT. Le mode CALIB est similaire au mode AUTO, sauf que tous les champs qui peuvent affecter le calcul de AO sont en lecture uniquement (*Mode*, *SelTrack*, *Invert*, *OutType*, *SiteNo*, *Channel*, *HR_out*, *LR_out*). Voir la description du bloc AO_CALIB à la page 2-68 du présent chapitre.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement	Menu	
FailBack	Mode de fonctionnement de reprise	Menu	
OP	Demande de sortie	Eng	
HR_OP, LR_OP	Echelle haute & basse PV	Eng	
AO	Sortie analogique	Eng	
Track	Valeur de sortie de poursuite	Eng	
SelTrack	VRAI sélectionne le mode poursuite (AO=Poursuite)	V/F	
Invert	Sélectionne l'inversion du signal de sortie	V/F	
NotAuto	VRAI si le mode est Manuel ou Poursuite	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut module E/S/Défauts alim. transmetteur	V/F	
CctFault	Sortie en circuit ouvert/court-circuit	V/F	
OvrDrive	Surcharge sortie	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	1-8	
OutType	Spécifie le type de sortie	Menu	
HR_out, LR_out	Equivalents des signaux HR, LR		
Status		ABCD Hex	
Reset	Remise à zéro du module E/S	V/F — 1	
Missing	Aucun module E/S sur site	V/F — 2	
CommsErr	Défaut communication du module E/S	V/F — 4	
BadType	Voie erronée sur module E/S	V/F — 8	
WrongCal	Mauvaise version données d'étalonnage (EEPROM)	V/F — 1	
BadCal		2	
	Données d'étalonnage corrompues	4	
		V/F — 8	
		1	
		2	
		4	
FaultCct	Circuit ouvert ou court-circuit	V/F — 8	
BadRange	Sélection échelle erronée	V/F — 1	
BadSetup	Configuration incorrecte pour module E/S connecté	V/F — 2	
HwFault	Défaut dans module E/S	V/F — 4	
OverDrv	Sortie surchargée	V/F — 8	
Killed	Sortie arrêtée	V/F —	
Options		(C)D Hex — 1	
PwrFILO	VRAI = sortie basse à la mise sous tension	V/F — 2	
CPUFILO	VRAI = sortie basse sur défaut UC	V/F — 4	
		V/F — 8	
Channel	Spécifie le numéro de voie dans le module E/S	1-n	

Fallback (Reprise). (AUTO/MANUAL/TRACK/CALIB). Le mode de fonctionnement suivant (supprimé) que le bloc adopte si le mode en cours est invalidé ou désélectionné.

OP. Demande de sortie. Entrée dans le bloc à partir du schéma de boucles, et source de AO en mode automatique.

HR_OP, LR_OP. Echelle haute et basse de AO (et OP) en unités physiques. HR_OP et LR_OP permettent de définir deux points sur une échelle linéaire en unités physiques qui sont normalement projetés respectivement sur les points HR_out et LR_out sur une échelle linéaire (volts ou milliampères). Ces deux échelles sont utilisées par le bloc pour calculer le signal de sortie AO. L'équation de conversion équivalente est la suivante:

$$\text{Sortie} = LR_out + \left[\frac{AO - LR_OP}{HR_OP - LR_OP} \times (HR_out - LR_out) \right]$$

NOTA. Le bit *Invert* affecte le calcul de la sortie. Voir *Invert* ci-dessous.

AO. Sortie analogique. Sortie réelle (par opposition à la sortie demandée OP ou la valeur Track) du bloc avant toute inversion ou modification d'échelle. AO est un paramètre spécial, dans la mesure où il est garanti qu'il sera configuré correctement avant le début d'exécution de tous les autres blocs (y compris les options défaut alimentation basse, le cas échéant). Cette fonction permet de relier AO au champ *InitDmnd* du bloc MAN_STAT.

Track (Poursuite). Source de AO en mode poursuite.

SelTrack (Sélection poursuite). VRAI sélectionne le mode poursuite.

Invert (Inversion). Si VRAI, HR_OP est projeté sur LR_out et LR_OP sur HR_out, ce qui inverse le sens du signal de sortie, par ex. pour le fonctionnement à sécurité intégrée ou pour les vannes à action inverse. Voir le paragraphe HR_OP, LR_OP ci-dessus.

NotAuto (Pas auto). VRAI, lorsque le bloc n'est pas en mode auto.

Alarms (Alarmes). Voir la description générale du champ alarme à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes:

- Coupure de courant
- Type de module incorrect
- Défaut matériel circuit (voir champ *Status* - état)

■ **CctFault (Défaut circuit).** Défaut circuit. Signalé lorsque le module détecte un circuit ouvert (sur les sorties courant) ou un court-circuit (sur les sorties tension).

■ **OvrDrive (Surcharge en sortie).** Situation de surcharge en sortie. Signalé lorsque le signal de sortie est surchargé par un autre module de sortie. Utilisé dans les schémas de boucles redondants. Noter que cette fonction n'est pas prise en compte par tous les matériels E/S; voir la documentation spécifique au matériel.

- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

SiteNo (N° de site). L'adresse du bus série interne (ISB) de l'instrument de la série T600 du module E/S auquel le bloc AN_OUT est relié. Les adresses ISB applicables dépendent du matériel E/S T600 utilisé. Par ex., dans un instrument T640, les sites E/S 1 et 2 sont respectivement affectés aux *SiteNos* 1 et 2, et tous les autres modules E/S externes prennent le SiteNo à partir de 3 et suivants. Noter qu'un *SiteNo* incorrect met à 1 le bit d'état *Missing* (manquant).

OutType (Type de sortie). (Volts/mA). Permet de sélectionner le type de sortie de tension ou de courant. *Volts* permet de configurer le bloc comme une sortie de tension de haut niveau avec une échelle maximale de -10 à +10 V. *mA* permet de configurer le bloc comme une sortie de courant avec une échelle maximale de -20 à +20 mA.

Les échelles de sorties données ci-dessus ne représentent que les maximum des blocs - voir les seuils spécifiques dans la documentation des modules E/S.

NOTA. Lorsque certaines options ne sont pas disponibles sur certaines voies, le bit d'état *BadSetup* est mis à 1.

HR_out, LR_out. Echelle haute & basse du signal de sortie dans les unités adaptées à la sélection en cours d'*OutType* (V ou mA). *HR_out* et *LR_out* permettent de définir deux points sur une échelle linéaire, qui sont projetés respectivement sur les points *HR_OP* et *LR_OP* sur une échelle linéaire *AO* en unités physiques. Lorsque *AO* est égal à *HR_OP*, la sortie réelle est à la valeur *HR_out*; lorsque la sortie réelle est à la valeur *LR_OP*, la sortie réelle est égale à *LR_OUT*. Voir les détails dans le paragraphe *HR_OP, LR_OP* ci-dessus. Noter que *HR_out* ne peut dépasser 10 V ou 20 mA respectivement, lorsque *OutType* = Volts ou mA. *HR_out* et *LR_out* peuvent toujours être définis indépendamment de *HR_OP* et *LR_OP*.

Le bloc compare les valeurs entrées de *HR_out* et *LR_out* à la liste des échelles matériel disponibles prises en compte par le module E/S et sélectionne l'échelle qui convient le mieux pour optimiser la résolution de l'entrée. Les valeurs exactes de *HR_out* et *LR_out* sont mises à l'échelle par le logiciel.

Status (Etat). Champ binaire indiquant des défauts de communication et de matériel. Les informations suivantes sont données en plus du récapitulatif de la table 2-9.

Options. Champ binaire dont les bits spécifient - si VRAI - ce qui se produit au niveau de la sortie analogique du bloc en cas de mise sous tension (après une coupure de courant) ou de défaut UC. Si FAUX, les bits n'ont aucun effet sur la sortie, c'est à dire qu'elle maintient son niveau existant en cas de démarrage à chaud ou "tiède" ou adopte sa valeur configurée en cas de démarrage à froid. Voir la table 2-9.

Channel (Voie). Permet de spécifier un numéro de voie dans le module E/S auquel le bloc AN_OUT est relié. Les numéros de voie applicables dépendent du type de matériel E/S utilisé - voir les détails dans la documentation des modules E/S.

Un numéro de *Channel* incorrect met à 1 le bit d'état *BadType*.

DG_IN : BLOC D'ENTREE LOGIQUE A 8 VOIES

Fonction du bloc

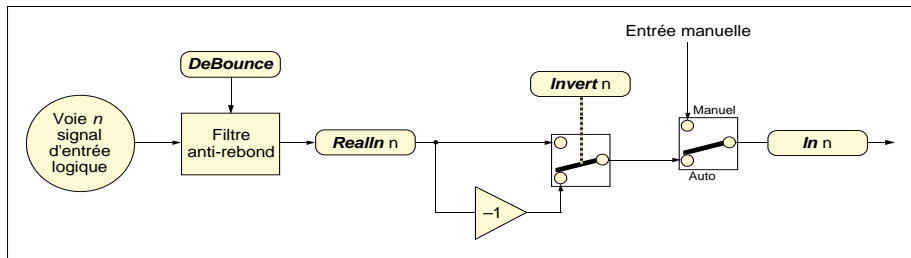


Figure 2-14 Schéma du bloc DGIN_8

Se reporter à la figure 2-14, qui schématise l'une des huit voies logiques. Le bloc DG_IN permet d'entrer 8 signaux logiques dans le schéma de boucles, et dispose du contrôle Auto/Manuel, de l'inversion et de l'anti-rebond. Les entrées peuvent être configurées soit comme détectant toutes des contacts ou des tensions. Les contacts utilisent la sortie de l'alimentation transmetteur (qui dispose d'une alarme de surcharge), à condition qu'elle n'alimente pas de transmetteurs analogiques. La borne de tension de tirage externe (IX) peut également être utilisée pour le tirage matériel de huit entrées logiques maximum. Les entrées de détection de tension ont des seuils de commutation variables.

Les champs alarme ne sont pas directement disponibles sur les voies d'entrée analogiques. Le bloc DIG_ALARM devrait être utilisé pour les alarmes sur chaque entrée, si nécessaire.

Paramètres du bloc

Les symboles utilisés dans la table 2-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Real In	Etats des liaisons du signal d'entrée (réels)	CD hex	
Bit0	Etat signal d'entrée voie 1	V/F — 1	D
Bit1	Etat signal d'entrée voie 2	V/F — 2	
Bit2	Etat signal d'entrée voie 3	V/F — 4	
Bit3	Etat signal d'entrée voie 4	V/F — 8	
Bit4	Etat signal d'entrée voie 5	V/F — 1	C
Bit5	Etat signal d'entrée voie 6	V/F — 2	
Bit6	Etat signal d'entrée voie 7	V/F — 4	
Bit7	Etat signal d'entrée voie 8	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
Invert	Etats inversés des signaux entrées (réels)	CD Hex	
Bit0	Signal d'entrée inversé voie 1	V/F — 1	D
Bit1	Signal d'entrée inversé voie 2	V/F — 2	
Bit2	Signal d'entrée inversé voie 3	V/F — 4	
Bit3	Signal d'entrée inversé voie 4	V/F — 8	
Bit4	Signal d'entrée inversé voie 5	V/F — 1	C
Bit5	Signal d'entrée inversé voie 6	V/F — 2	
Bit6	Signal d'entrée inversé voie 7	V/F — 4	
Bit7	Signal d'entrée inversé voie 8	V/F — 8	
In	Etats entrée du schéma de boucles	CD Hex	
Bit0	Etat entrée schéma voie 1	V/F — 1	D
Bit1	Etat entrée schéma voie 2	V/F — 2	
Bit2	Etat entrée schéma voie 3	V/F — 4	
Bit3	Etat entrée schéma voie 4	V/F — 8	
Bit4	Etat entrée schéma voie 5	V/F — 1	C
Bit5	Etat entrée schéma voie 6	V/F — 2	
Bit6	Etat entrée schéma voie 7	V/F — 4	
Bit7	Etat entrée schéma voie 8	V/F — 8	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut module E/S/Défauts alim. transmetteur	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISBT600 du module E/S	Entier	
InType	Spécifie le type d'entrée	Menu	
Thresh	Seuil de commutation	Numérique	
DeBounce	Temps anti-rebond (secs)	Numérique	
Status	Etat communication/matériel	ABCD hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Valeur N° de site incorrecte	V/F — 2	
CommsErr	Défaut communication (UC T600 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
			1
			2
PSUShort	Surcharge alim. transmetteur	V/F — 4	C
BadThr	Valeur seuil non gérée par module E/S	V/F — 8	
			1
			2
			4
			B
BadDeb	Durée anti-rebond non gérée par module E/S	V/F — 8	
BadSetup	Option configurée non indisponible sur matériel E/S	V/F — 1	A
HwFault	Défaut matériel dans module E/S	V/F — 2	
			4
			8
BitCount	Nombre de bits pris en charge par matériel E/S	Entier	

Table 2-10 Paramètres du bloc DG_IN

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUAL - AUTO/MANUEL). En mode Automatique, le champ état *In* suit les stimuli d’entrée des liaisons du champ. En mode Manuel, *In* peut être manipulé indépendamment des stimuli du champ.

RealIn (Entrée réelle). Champ binaire indiquant l’état réel de chacune des huit liaisons d’entrée.

Invert (Inversion). Champ binaire permettant l’inversion de l’une des huit entrées dans le schéma de boucles.

In (Entrée). Champ binaire indiquant les états des huit entrées dans les schémas de boucles. Les entrées peuvent être sous la forme vraie ou inversée suivant les états des champs Invert. Si le bloc est en mode manuel, *In* est indépendant de *RealIn*.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel peut être déclenchée dans les conditions suivantes :

- Coupure de courant
- Surcharge alimentation
- Type de module incorrect
- Défaut matériel circuit (voir champ *Etat*).

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de spécifier l’adresse (1 à 8) sur le bus série interne des instruments de la série T600 où le module E/S correspondant à ce bloc DG_IN est situé. Noter qu’un *SiteNo* incorrect met à 1 le bit d’état *Missing* (*manquant*).

NOTA. Dans le cas de la carte E/S haut niveau d’un instrument T640, chacun de ses deux sites (1 et 2) ne peut traiter que quatre bits (bits 0 à 3). Par conséquent, deux blocs DG_IN - les paramètres *SiteNo* étant respectivement mis à 1 et 2 - doivent traiter les huit bits disponibles dans l’instrument.

InType (Type d’entrée). (Volts/Contacts). Permet de définir le type d’entrée.

Thresh (Seuil de commutation). Permet de définir le seuil pour l’option entrée tension.

DeBounce (Anti-rebond). Temps anti-rebond en secondes. Il s’agit de la durée minimum qu’une modification du signal d’entrée logique doit prendre avant que le paramètre *RealIn* n’ait pu passer au nouvel état. L’action de filtrage de *DeBounce* (Anti-rebond) est représentée à la figure 2-15. (Le temps anti-rebond minimum dépend du temps d’échantillonnage ; durée maxi. 5 secondes).

Status (Etat). Champ binaire indiquant des situations d'erreur générales communicat./matériel. Voir les détails, table 2-10.

BitCount (Comptage des bits). Indique le nombre de bits (entrées logiques) pris en compte par le module matériel associé.

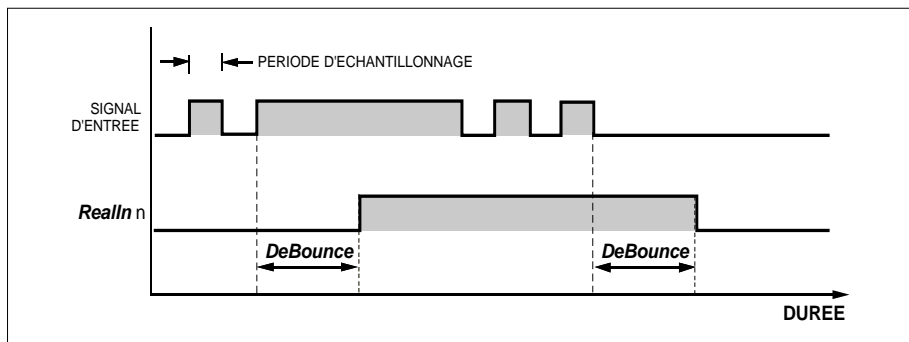


Figure 2-15 Effets de l'anti-rebond sur le signal d'entrée logique

DG_OUT : BLOC DE SORTIE LOGIQUE A 8 VOIES

Fonction du bloc

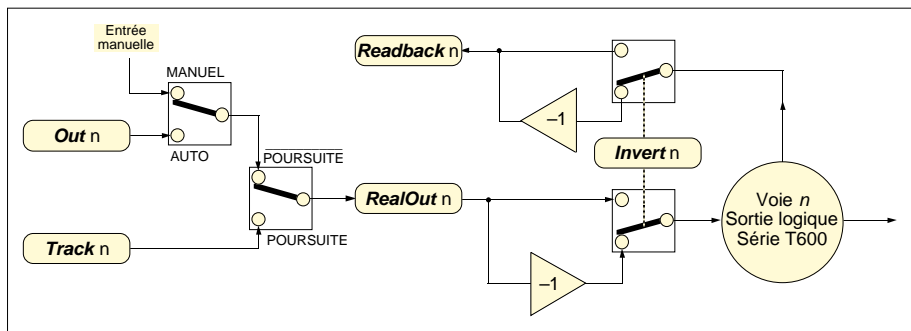





Figure 2-16 Schéma bloc DG_OUT

Se reporter à la figure 2-16, qui schématise l'une des huit voies logiques. Le bloc DG_OUT permet de transférer des signaux vers des voies de sorties logiques sur un module de sortie logique de la série T600. Le bloc dispose du contrôle Auto/Manuel/ Poursuite, d'inversion et de vérification de discordance du retour de la sortie.











Paramètres du bloc

Les symboles utilisés dans la table 2-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
FallBack	Mode de fonctionnement supprimé	Menu	
Out	Etats sortie du schéma de boucles	CD hex	 
Bit0	Etat sortie voie 1	V/F — 1	D
Bit1	Etat sortie voie 2	V/F — 2	
Bit2	Etat sortie voie 3	V/F — 4	
Bit3	Etat sortie voie 4	V/F — 8	
Bit4	Etat sortie voie 5	V/F — 1	C
Bit5	Etat sortie voie 6	V/F — 2	
Bit6	Etat sortie voie 7	V/F — 4	
Bit7	Etat sortie voie 8	V/F — 8	
Track	Etats sortie poursuite	CD hex	
Bit0	Etat sortie poursuite voie 1	V/F — 1	D
Bit1	Etat sortie poursuite voie 2	V/F — 2	
Bit2	Etat sortie poursuite voie 3	V/F — 4	
Bit3	Etat sortie poursuite voie 4	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
Bit4	Etat sortie poursuite voie 5	V/F — 1	C
Bit5	Etat sortie poursuite voie 6	V/F — 2	
Bit6	Etat sortie poursuite voie 7	V/F — 4	
Bit7	Etat sortie poursuite voie 8	V/F — 8	
RealOut	Etats sortie (avant inversion)	CD hex	
Bit0	Etat sortie voie 1	V/F — 1	D
Bit1	Etat sortie voie 2	V/F — 2	
Bit2	Etat sortie voie 3	V/F — 4	
Bit3	Etat sortie voie 4	V/F — 8	
Bit4	Etat sortie voie 5	V/F — 1	C
Bit5	Etat sortie voie 6	V/F — 2	
Bit6	Etat sortie voie 7	V/F — 4	
Bit7	Etat sortie voie 8	V/F — 8	
Invert	Etats inversés signal de sortie (réels)	CD hex	
Bit0	Signal de sortie inversé voie 1	V/F — 1	D
Bit1	Signal de sortie inversé voie 2	V/F — 2	
Bit2	Signal de sortie inversé voie 3	V/F — 4	
Bit3	Signal de sortie inversé voie 4	V/F — 8	
Bit4	Signal de sortie inversé voie 5	V/F — 1	C
Bit5	Signal de sortie inversé voie 6	V/F — 2	
Bit6	Signal de sortie inversé voie 7	V/F — 4	
Bit7	Signal de sortie inversé voie 8	V/F — 8	
Readback	Contre-réaction des états sortie du module E/S	CD hex	 
Bit0	Contre-réaction voie 1	V/F — 1	D
Bit1	Contre-réaction voie 2	V/F — 2	
Bit2	Contre-réaction voie 3	V/F — 4	
Bit3	Contre-réaction voie 4	V/F — 8	
Bit4	Contre-réaction voie 5	V/F — 1	C
Bit5	Contre-réaction voie 6	V/F — 2	
Bit6	Contre-réaction voie 7	V/F — 4	
Bit7	Contre-réaction voie 8	V/F — 8	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut module E/S/Défauts alim. transmetteur	V/F	
CctFault	Détection différence RealOut/Readback	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	Entier	
Pullup	Tension de sortie état logique haut	Menu	
SelTrack	Sélection mode poursuite	V/F	
NotAuto	Etat mode PAS Auto	V/F	 

suite...

...suite






Paramètre	Fonction	Unités	Etat
Status	Etat communic./matériel	ABCD hex	 
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Valeur N° de site incorrecte	V/F — 2	
CommsErr	Défaut communication (T600 à module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
		1 2 4 8	C
BadPulup	Valeur de tirage non prise en charge par module E/SV/F		
		1 2 4 8	
		1 2 4 8	
BadSetup	Option configurée indisponible sur matériel E/S	V/F — 1	A
HwFault	Défaut matériel dans module E/S	V/F — 2	
		4 8	
		1 2 4 8	
Options	Options bloc	(C)D hex	 
PwrFILO	VRAI = sortie basse à la mise sous tension	V/F — 1	D
CPUFILO	VRAI = sortie basse sur défaut UC	V/F — 2	
		4 8	
		1 2 4 8	
BitCount	Nombre de bits pris en charge par le matériel E/S	Entier	

Table 2-11 Paramètres du bloc DG_OUT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUAL/TRACK - AUTO/MANUEL/POURSUITE). Le mode de fonctionnement en cours. TRACK (POURSUITE) doit être sélectionné par l’intermédiaire du paramètre *SelTrack*

Fallback (Reprise). (AUTO/MANUAL - AUTO/MANUEL). Le mode de fonctionnement (supprimé) que le bloc adopte si le mode de fonctionnement en cours est invalidé ou désélectionné.

Out (Sortie). En mode automatique, ce paramètre de champ binaire indique l’état des voies de sortie défini par le schéma de boucles. En mode manuel, ce paramètre est en lecture/écriture. Ce champ peut indiquer la forme vraie ou inversée des voies de sorties réelles suivant le statut des champs *inversion*.

Track (Poursuite). Champ binaire contrôlant l’état des sorties en mode poursuite.

RealOut (Sortie réelle). Champ binaire indiquant l'état (avant toute inversion) des liaisons de sortie du champ.

Invert (Inversion). Champ binaire spécifiant les inversions du champ.

Readback (Retour d'état). Champ binaire indiquant le retour des états réels de sortie à partir du module de sortie logique de la série T600 (avant toute inversion). Permet de vérifier une différence par comparaison avec *RealOut* qui devrait être identique.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée dans les conditions suivantes :

- Coupure de courant
- Type de module incorrect
- Défaut matériel circuit (voir champ *Etat*).

■ **CctFault (Défaut circuit).** Cette alarme se déclenche lorsqu'une différence est détectée entre l'état de *RealOut* et de *ReadBack*, indiquant qu'il y a un défaut sur une des voies de sortie.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de spécifier l'adresse (1 à 8) sur le bus série interne des instruments de la série T600 où le module E/S correspondant à ce bloc DG_OUT est situé. Noter qu'un *SiteNo* incorrect met à 1 le bit d'état *Missing* (*manquant*).

NOTA. Dans le cas de la carte E/S haut niveau d'un instrument T640, chacun de ses deux sites (1 et 2) ne peut traiter que quatre bits (bits 0 à 3). Par conséquent, deux blocs DG_OUT - les paramètres *SiteNo* étant respectivement mis à 1 et 2 - doivent traiter les huit bits disponibles dans l'instrument.

Pullup (Tirage). (Externe/5/15/24). Permet de définir la tension de sortie utilisée pour l'état logique haut.

SelTrack (Choix mode poursuite). Permet de sélectionner le mode poursuite.

NotAuto (Non auto). Indique que le bloc ne fonctionne pas en mode automatique.

Status (Etat). Champ binaire indiquant une condition d'erreur générale de communicat./matériel. Voir les détails dans la table 2-11.

Options. Champ binaire pour sélectionner les options de fonctionnement du bloc. Voir table 2-11.

BitCount. Indique le nombre de bits (sorties logiques) pris en charge par le module matériel associé.

DGPULS_4 : BLOC D'IMPULSIONS LOGIQUES A 4 VOIES

Fonction du bloc

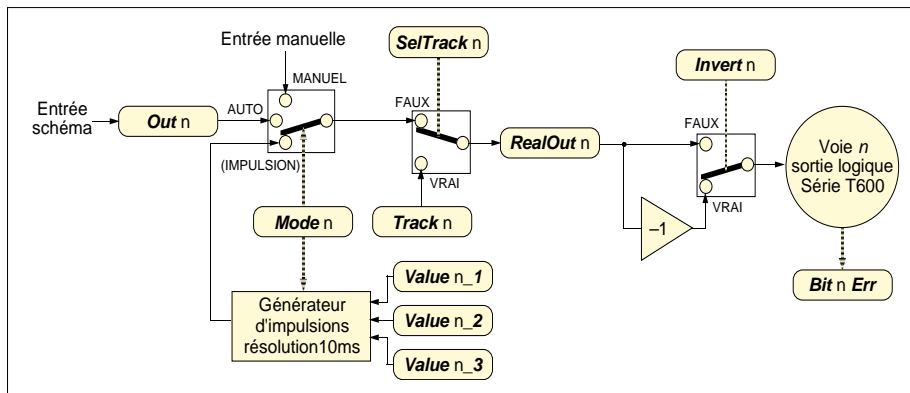


Figure 2-17 Paramètres du bloc DGPULS_4

Le schéma de la figure 2-17 montre l'une des quatre voies du bloc DGPULS_4. Le mode de chaque voie peut être sélectionné indépendamment. En mode automatique, une voie transmet les sorties logiques du schéma de boucles aux installations par l'intermédiaire du module de sortie logique associé de la série T600. Le bloc dispose également des modes manuel, poursuite et à quatre impulsions, ainsi que des modes inversion et vérification de discordance. Noter que pour un instrument T640 avec des cartes E/S haut niveau sur les deux sites, seul le site 1 peut prendre en compte le bloc DGPULS_4.

Modes impulsions

NOTA. Le matériel de sortie logique doit assurer une *fonction de synchronisation* sur chaque voie utilisée par une voie DGPULS_4 en mode impulsions. Voir les spécifications du matériel dans le manuel produit de l'instrument.

En mode impulsions, les voies du bloc peuvent sortir une impulsion unique (ONE_SHOT - UNIQUE), une onde rectangulaire proportionnelle au temps (TPO), un train d'impulsion (PLS_TRN) ou - en interconnectant une paire de voies - une impulsion double (DUAL_PLS). Les paramètres *Valuen_1* à *Valuen_3* permettent de spécifier les synchronisations et les rapports actif/inactif des impulsions.

Dans tous les modes impulsions, l'exécution de la tâche du bloc calcule des valeurs entières qui représentent des unités de 10 ou 20 ms (suivant le type de module E/S). Ces valeurs entières sont téléchargées dans le module E/S, afin qu'il puisse effectuer la synchronisation haute vitesse nécessaire.

Dans le schéma simplifié de la figure 2-17, le paramètre *RealOutn* est représenté comme pilotant le matériel de sortie, mais en réalité ce champ est mis à jour en relisant son état à partir du module E/S. Comme les deux états logiques demandés et sa valeur de relecture sont exécutés nettement plus rapidement dans le matériel que dans le bloc, ils sont comparés au niveau du module E/S, et une valeur saine (drapeau *BitnErr*) est renvoyée au bloc. Les drapeaux *BitnErr* sont verrouillants, afin que des discordances passagères ne soient pas perdues. Le module E/S active un drapeau sain, chaque fois que la valeur a été transmise avec succès au processeur principal. L'inversion est également mise en oeuvre dans le module E/S - non pas comme suggéré sur le schéma - pour des raisons similaires.

Mode ONE_SHOT (UNIQUE)

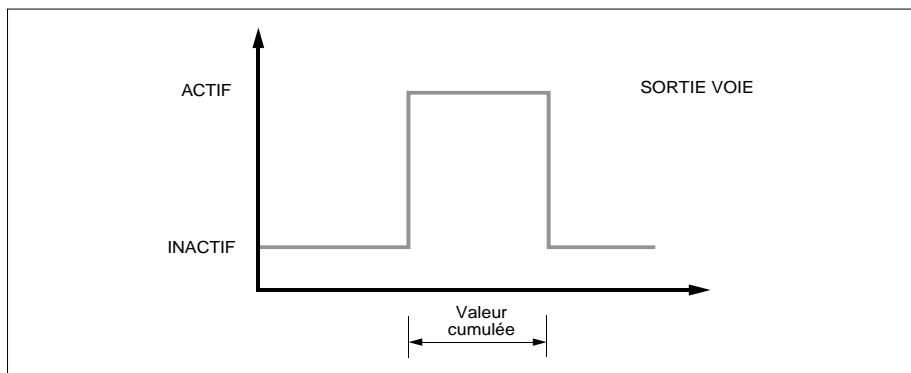


Figure 2-18 Sortie du mode ONE_SHOT

Les vannes entraînées par un moteur arrêt/marche représentent une application typique pour la sortie à impulsion unique. Dans ce mode, le bloc génère une impulsion unique (voir figure 2-18). A chaque exécution du bloc la valeur dans *Valuen_1* est ajoutée à un totalisateur qui représente la durée d'impulsion restante en secondes. Chaque fois que le totalisateur n'est pas à zéro, *RealOutn* = VRAI, c'est à dire que la sortie est à l'état "actif", et le totalisateur décompte en temps réel. Chaque fois que le totalisateur est à zéro, *RealOutn* = FAUX, c'est à dire que la sortie est à l'état "inactif", et le totalisateur est "au repos". Le totalisateur ONE_SHOT est sans signe.

Chaque fois que le bit *Accn_Rst (Options)* passe à VRAI, le totalisateur de la voie n est remis à zéro.

Mode TPO

Le mode TPO (sortie proportionnelle au temps) génère une onde rectangulaire continue. *Valuen_1* spécifie la période en secondes (voir figure 2-18), *Valuen_2* spécifie le pourcentage du cycle à l'état actif, et *Valuen_3* spécifie à la fois la durée minimale de l'état actif et la durée minimale de l'état inactif (en secondes). Si la durée d'impulsion demandée - soit actif ou inactif - est inférieure au minimum spécifié, l'impulsion n'est pas

générée, c'est à dire que la sortie est maintenue à un état inactif ou actif constants. Si la durée de l'état actif (*Valuen_2*) est modifiée, lorsque la sortie est dans cet état, la modification prend effet immédiatement, sans attendre la fin de la période en cours.

Une compensation de résolution est appliquée, de sorte que si la durée de l'état actif demandée ne peut être représentée dans les limites de la résolution de la durée d'impulsion du module E/S, la durée réelle de l'état actif varie dans les limites de celle demandée. Par ex., si l'état actif spécifié est de 25 ms, mais que le module E/S ne possède qu'une résolution à 10 ms près, la durée de l'état actif réelle variera entre 20 ms et 30 ms, ce qui représente une moyenne de 25 ms. La compensation de résolution est précise à 12 bits.

(Le bit *Options.Accn_Rst* n'est pas applicable en mode TPO).

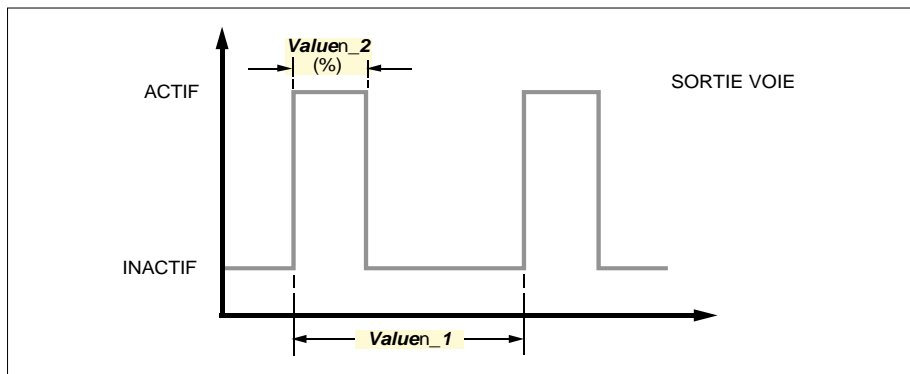


Figure 2-19 Sortie du mode TPO

Mode PLS_TRN

Le mode “train d’impulsions” génère un train d’impulsions utile dans les applications de moteur pas à pas et clapet restricteur, etc. *Valuen_2* (voir figure 2-20) permet de spécifier la durée d’impulsion. A chaque exécution du bloc, la valeur de *Valuen_1* est ajoutée à un totalisateur, qui contient le nombre d’impulsions qui restent à sortir. Seuls des cycles entiers peuvent être sortis, de sorte que si la valeur du totalisateur n’est pas intégrale, le nombre réel d’impulsions sorties est la valeur entière inférieure suivante. Le totalisateur est décrémenté pour chaque impulsion sortie.

Un nouveau train d’impulsions ne peut être lancé avant la fin du précédent, ce qui est défini comme une durée d’une demie période après le dernier front descendant.

Les impulsions ont un rapport actif/inactif de un (dans les limites de la résolution 10/20ms). Toute modification de *Valuen_2* (période) prend effet immédiatement.

Chaque fois que le bit *Accn_Rst* (*Options*) passe à VRAI, le totalisateur de la voie n est remis à zéro. Le totalisateur PLS_TRN est sans signe.

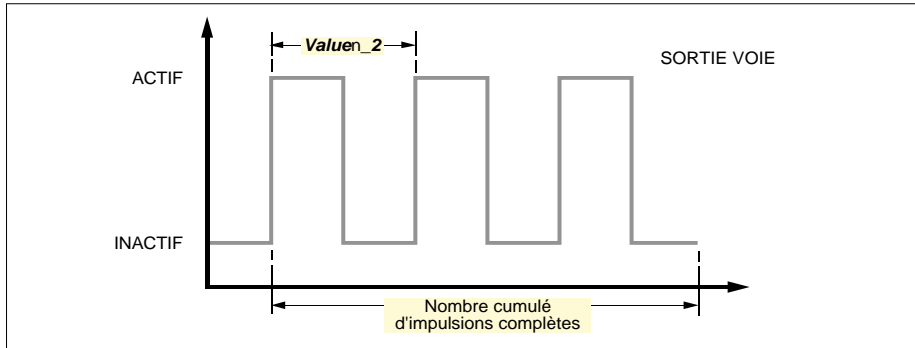


Figure 2-20 Sortie du mode PLS_TRN

Mode DUAL_PLS

Ce mode combine deux voies en une paire - les voies 1 et 2 forment une paire, les voies 3 et 4 une autre. Une paire de voie sort une paire d'impulsions liée, une sur chaque voie (voir figure 2-21).

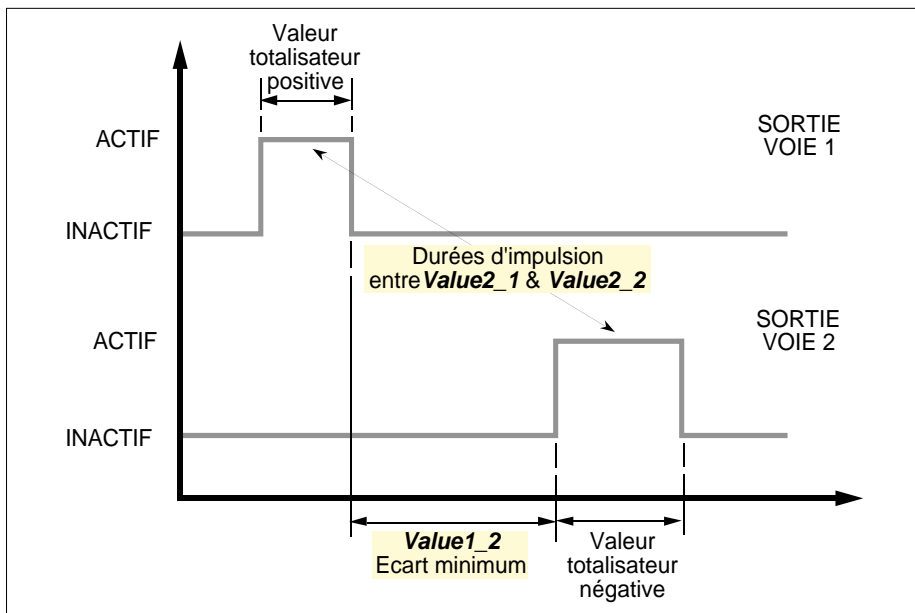


Figure 2-21 Sortie du mode DUAL_PLS

Considérons la paire Ch1-Ch2, à chaque exécution du bloc, la valeur de *Value1_1* - qui peut être positive ou négative - est ajoutée au totalisateur qui représente la durée

d'impulsion requise en secondes. Lorsque le totalisateur est à zéro, les deux voies sortent un état inactif. Lorsque le totalisateur est positif, (et la valeur supérieure à *Value2_1*) un état actif est sorti sur la voie 2, et l'accumulateur décompte en temps réel. Lorsque l'accumulateur est négatif (et d'une valeur absolue supérieure à *Value2_1*) un état actif est sorti sur la voie 2 et l'accumulateur compte en temps réel. *Value2_1* représente donc une durée d'impulsion minimale admissible. *Value2_1* définit une durée d'impulsion maximale qui ne peut être dépassée. *Value1_2* définit l'écart minimal qu'il doit y avoir entre un état actif sur la voie 1 et un état actif sur la voie 2.

Chaque fois que le bit *Acc1_Rst (Options)* est défini comme VRAI, la valeur du totalisateur repasse à une valeur positive, et lorsque *Acc2_Rst* est défini comme VRAI, la valeur de l'accumulateur repasse à une valeur négative. Noter que ces bits de réinitialisation ne sont pas réinitialisés automatiquement.

Paramètres du bloc

Les symboles utilisés dans la table 2-12 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Out			
Chan1	Etats sélectionnés voie 1	V/F	
Chan2	Etats sélectionnés voie 2	V/F	
Chan3	Etats sélectionnés voie 3	V/F	
Chan4	Etats sélectionnés voie 4	V/F	
Track			
Chan1	Etats poursuite voie 1	V/F	
Chan2	Etats poursuite voie 2	V/F	
Chan3	Etats poursuite voie 3	V/F	
Chan4	Etats poursuite voie 4	V/F	
Invert			
Chan1	Etats inversion voie 1	V/F	
Chan2	Etats inversion voie 2	V/F	
Chan3	Etats inversion voie 3	V/F	
Chan4	Etats inversion voie 4	V/F	
RealOut			
Chan1	Etats sortie voie 1	V/F	
Chan2	Etats sortie voie 2	V/F	
Chan3	Etats sortie voie 3	V/F	
Chan4	Etats sortie voie 4	V/F	

suite...

...suite

Paramètre	Fonction	Unités	Etat
SelTrack			
Chan1	Sélection mode poursuite voie 1	V/F	
Chan2	Sélection mode poursuite voie 2	V/F	
Chan3	Sélection mode poursuite voie 3	V/F	
Chan4	Sélection mode poursuite voie 4	V/F	
NotAuto			
Chan1	Si voie 1 en manuel ou poursuite	V/F	
Chan2	Si voie 2 en manuel ou poursuite	V/F	
Chan3	Si voie 3 en manuel ou poursuite	V/F	
Chan4	Si voie 4 en manuel ou poursuite	V/F	
Mode1-Mode4	Modes de fonctionnement voies 1 à 4	Menu	
Fallbak1-Fallbak4	Modes de reprise voies 1 à 4	Menu	
Value _n _1 à 3	Valeurs 1 à 3 mode impulsions, voie n	Val.flottante	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut module E/S/Défauts alim. transmetteur	V/F	
Bit1Err	Erreur relecture voie 1	V/F	
Bit2Err	Erreur relecture voie 2	V/F	
Bit3Err	Erreur relecture voie 3	V/F	
Bit4Err	Erreur relecture voie 4	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	Entier	
BitCount	Nombre de bits pris en compte par le matériel E/S	Entier	
Options		CD Hex	
PwrFILO	Etat bas sur défaut alimentation	V/F	
CPUFILO	Etat bas sur défaut UC	V/F	
Acc1_Rst	Remise à zéro bit 1 totalisateurs	V/F	
Acc2_Rst	Remise à zéro bit 2 totalisateurs	V/F	
Acc3_Rst	Remise à zéro bit 3 totalisateurs	V/F	
Acc4_Rst	Remise à zéro bit 4 totalisateurs	V/F	
Status		ABCD Hex	
Reset	Remise à zéro du module E/S	V/F	
Missing	Aucun module E/S sur le site	V/F	
CommsErr	Défaut communication du module E/S	V/F	
BadType	Voie erronée sur module E/S	V/F	
BadPulup	Valeur de tirage non gérée par module E/S	V/F	
BadPulse	Mode impulsions non pris en compte	V/F	

suite...

...suite

Paramètre	Fonction	Unités	Etat
			1 2 4 8
			B
BadSetup	Configuration incorrecte de la voie	V/F	1
HwFault	Défaut matériel dans le module E/S	V/F	2
			4 8
			A
Pullup	Spécifie la tension de sortie état logique 1	Menu	

Table 2-12 Paramètres du bloc DGPULS_4

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Out. Ce paramètre de champ binaire peut être relié directement au schéma de boucles. En mode automatique, *Out* contrôle les voies de sortie (sous réserve de l’état des champs *Invert*). En mode manuel et impulsions, *Out* est déconnecté des sorties. Voir figure 2-17.

Track (Poursuite). Champ binaire contrôlant les états des quatre voies de sortie en mode poursuite.

Invert (Inversion). Champ binaire spécifiant les inversions des quatre voies de sortie du champ.

RealOut (Sortie réelle). Champ binaire indiquant l’état (avant toute inversion) des quatre voies de sortie du champ.

SelTrack (Sélection poursuite). Un bit VRAI sélectionne le mode poursuite pour cette voie.

NotAuto (Pas auto). Un bit VRAI indique que la voie fonctionne en mode non-automatique.

Mode1 à Mode4. (AUTO/MANUAL/TRACK/ONE_SHOT/TPO/PLS_TRN/DUAL_PLS). Permet de sélectionner le mode de fonctionnement pour les voies 1 à 4. Noter que TRACK doit être sélectionné par l’intermédiaire du paramètre *SelTrack*. En mode AUTO, les sorties du champ sont contrôlées par *Out*; en mode MANUEL, par une entrée manuelle dans *RealOut*; et en mode TRACK, par le paramètre *Track* - sous réserve des inversions.

Voir la description détaillée des quatre modes d’impulsions dans le paragraphe *Modes impulsions* ci-dessus.

Fallbak1 à Fallbak4 (Reprise1 à 4). (AUTO/MANUAL/TRACK/ONE_SHOT/TPO/PLS_TRN/DUAL_PLS). Le mode de fonctionnement adopté par le bloc, lorsque le mode TRACK est désélectionné.

Valuen_m, ($n=1$ à 4, $m=1$ à 3). Un ensemble de trois valeurs (m) pour chacune des quatre voies (n). Ces valeurs permettent de spécifier les synchronisations (en secondes) et pourcentages liés à la génération des impulsions de sortie dans l'un des quatre modes d'impulsions, sélectionné par le paramètre *Mode* (n) de la voie correspondante. Voir les détails au paragraphe *Modes d'impulsions* ci-dessus.

Alarms (Alarmes). Voir la description générale du champ alarme à la page 2-5.

■ **Software (Logiciel)**. Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel)**. Une alarme matériel est déclenchée dans les conditions suivantes:

- Coupure de courant
- Type de module incorrect
- Défaut matériel circuit (voir champ *Status* - état)

■ **Bit1Err à Bit4Err**. Ces alarmes sont déclenchées pour les voies 1 à 4 respectivement, lorsqu'un écart est détecté entre la sortie souhaitée de la voie *RealOutn* et une sortie de relecture (retour d'état), indiquant un défaut sur une voie de sortie. Toute inversion configurée pour la voie est admissible pour la vérification de discordance.

■ **Combined (Combinée)**. VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

SiteNo (N° de site). Permet de spécifier l'adresse (1 à 8) sur le bus série interne des instruments de la série T600, où le module E/S correspondant à ce bloc DGPULS_4 est situé. Il faut noter qu'un *SiteNo* incorrect met à 1 le bit d'état *Missing* (manquant).

BitCount (Comptage bits). Signale le nombre de bits (sorties logiques) pris en compte par le module matériel associé.

Options. Champ binaire permettant de sélectionner les options de fonctionnement du bloc. Voir table 2-12.

Status (Etat). Champ binaire indiquant des défauts de communication et de matériel. Voir les détails dans la table 2-12.

Pullup (Tirage). (Externe/5/15/24). Permet de définir la tension de sortie utilisée pour l'état logique haut.

AI_CALIB: BLOC D'ÉTALONNAGE DES ENTREES ANALOGIQUES

Fonction du bloc

Le bloc AI_CALIB permet d'étalonner les voies des entrées analogiques dans les modules E/S de la série T600. Une base de données est créée dans l'instrument de la série T600 qui contient le bloc AN_IP associé au module d'E/S à étalonner, plus un bloc AI_CALIB. Le bloc d'étalonnage s'attache au bloc AN_IP désigné, le force à prendre le mode CALIB, puis le traite pour étalonner la voie E/S spécifiée. Grâce à son ou ses champs *Action*, le bloc d'étalonnage affiche l'une des séquences d'instructions à exécuter par l'opérateur, qui définit alors le champ *Status*. *CONTINUE*, lorsqu'il est prêt à passer à l'instruction suivante.

Dans un instrument déporté, une copie image du bloc d'étalonnage est créée, afin que la procédure d'étalonnage puisse être mise en oeuvre dans le bloc par l'intermédiaire d'une interface plein écran.

Les détails d'étalonnage des différents types de modules d'entrées analogiques sont donnés dans le paragraphe *Procédures d'étalonnage* ci-dessous.

Paramètres du bloc

Les symboles utilisés dans la table 2-13 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.








Paramètre	Fonction	Unités	Etat
Action	Donner des instructions d'étalonnage à l'opérateur	Menu	
(Champ invisible)		Menu	
Value	Valeurs entrées par opérateur demandées par "Action"		
Status	Etat de la procédure d'étalonnage (& Abandon I/P)	Menu	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Failed	Abandon procédure d'étalonnage (priorité par défaut=6)	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	Entier	
Channel	Spécifie le n° de voie dans le module E/S	Entier	
InType	Spécifie type d'entrée analogique en cours d'étalonnage	Menu	
Range	Signale l'échelle matériel en cours d'étalonnage	Menu	
AN_IP	Nom du bloc AN_IP de rattachement	Alphanumérique	
AI_Alarm	Indique si config. voie correcte/AN_IP en alarme	Menu	

Table 2-13 Paramètres du bloc AI_CALIB

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs "en-tête" à la page 2-3.

Action (partie 1 - Visible). (Done/Apply/Enter/Write - Terminé/Appliquer/Entrer/Ecrire). Permet d'afficher l'un des quatre messages courts de texte, qui, en combinaison avec un message affiché par la seconde partie (invisible) du champ *Action* qui se trouve en dessous, permet de donner un certain nombre d'instructions à l'opérateur.

“**Done**” (terminé) est affiché lorsque le bloc est au repos, c'est à dire avant le début de l'étalonnage ou lorsque l'étalonnage est terminé. Voir paragraphe suivant, *Action (partie 2 - invisible)*.

Action (partie 2 - Invisible). ([blank]/High rng/Low rng/E2 data?/Test 1/Test 2/R/V1/V3/Vjc/Vbase/switches - [blank]/Echelle haute/Echelle basse/Données E2?/Test 1/Test 2/R/V1/V3/Tension Cjc/Tension basse/commutateurs) Ce champ invisible fournit la seconde partie d'une instruction opérateur, par ex. **Apply Low rng** (Appliquer échelle basse). L'opérateur exécute l'instruction donnée, et sélectionne ensuite *CONTINUE* dans le paramètre *Status* pour indiquer au bloc que l'instruction a été dûment exécutée. Voir les détails au paragraphe *Procédures d'étalonnage*.

Les instructions *Action* sont très brèves, et l'opérateur est supposé se référer au présent manuel pour une description complète des actions à entreprendre.

Value (Valeur). Lorsqu'il est demandé à l'opérateur d'entrer une valeur, la valeur doit être entrée dans ce champ.

Status (Etat). (WAITING/CONTINUE/PROCESSING/ABORT) Montre à l'opérateur l'état de la procédure d'étalonnage, et lui permet également de l'abandonner.

- **WAITING (EN ATTENTE).** Affiché chaque fois que le bloc AI_CALIB attend une action de l'opérateur.
- **CONTINUE (CONTINUER).** Doit être entré par l'opérateur, lorsque l'action en attente est achevée.
- **PROCESSING (TRAITEMENT EN COURS).** Affiché lorsque le bloc AI_CALIB est sur le point d'exécuter une action.
- **ABORT (ABANDON).** La procédure d'étalonnage est abandonnée lorsque ce champ est sélectionné.

Alarms (Alarmes). Voir la description générale du champ alarme à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Failed (Echec).** Indique un abandon de la procédure d'étalonnage. Cette alarme de priorité 6 est automatiquement définie par le bloc et signifie que, si l'alarme est déclenchée, celle-ci doit être acquittée avant de poursuivre.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

SiteNo (N° de site). Permet de spécifier l'adresse (1 à 8) sur le bus série interne des instruments de la série T600, dont l'entrée analogique doit être étalonnée. Le bloc AI_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_IP, dès qu'il lui est rattaché, mais *SiteNo* peut être modifié ultérieurement.

Channel (Voie). Permet de définir le numéro (1 à 8) de voie d'entrée analogique à étalonner. Le bloc AI_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_IP, dès qu'il lui est rattaché, mais *Channel* peut être modifié ultérieurement.

InType (Type d'entrée). (HiLevel/Therm/RTD/Freq) Permet de spécifier le type d'entrée analogique à étalonner. Le bloc AI_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_IP, dès qu'il lui est rattaché, mais *InType* peut être modifié ultérieurement.

NOTA. L'ensemble de la procédure d'étalonnage pour un *InType* particulier doit être effectuée jusqu'au bout pour toutes les échelles prises en compte, ou abandonnée. Il n'est pas possible de n'étalonner qu'une partie des échelles.

Range (Echelle). (0-1.25V/0-2.5V/0-5V/0-10V/0-12.5mV/0-25mV/0-50mV/0-100mV/0-50 Ohm/0-100 Ohm/0-500 Ohm/0-1 kOhm/0-30 kHz) Champ de lecture uniquement signalant laquelle des échelles matériel de la voie spécifiée est en cours d'étalonnage. Voir le nota du paragraphe *InType* ci-dessus.

AN_IP. Spécifie le nom du bloc AN_IP auquel le bloc AI_CALIB doit être rattaché. Une fois qu'un nom de bloc valable a été sélectionné, le bloc AI_CALIB s'y rattache et le force à passer en mode CALIB.

AI_ALARM. (Invalid/Bad Loop/In Alarm/OK) Indique l'état du bloc d'étalonnage.

- **Invalid (Incorrect)** signifie que la configuration pour la voie est incorrecte ou qu'aucun bloc AN_IP n'a été trouvé.
- **Bad Loop (Erreur boucle)** signifie que les blocs AN_IP et AI_CALIB ne sont pas exécutés dans la même tâche utilisateur ("boucle"), ce qui doit être le cas pour pouvoir effectuer l'étalonnage.
- **In Alarm (En alarme)** signifie que le bloc AN_IP rattaché est en alarme.
- **OK** est affiché dans tous les autres cas.

Procédures d'étalonnage

Attention

Seul le personnel qualifié utilisant des équipements homologués est autorisé à ré-étalonner le matériel.

Pour lancer le processus d'étalonnage, entrer le nom du bloc E/S "source" dans le champ *AN_IP* du bloc AI_CALIB. Si le bloc source peut être trouvé, le bloc d'étalonnage s'y rattache et copie ensuite les autres champs, le cas échéant - *SiteNo*, *Channel*, *InType*, *Range*. Le champ *Action* affiche **Done**, et le champ *Status* indique **WAITING**. Commencer le processus d'étalonnage est mettant *Status* sur **CONTINUE**.

Les paragraphes ci-après présentent sous forme de tables chacune des commandes données par le bloc et les actions à entreprendre par l'opérateur. *InType* doit être défini en fonction du matériel avant le début de la procédure d'étalonnage, dans la mesure où chaque type est étalonné au cours d'une séquence spécifique.

Séquence d'étalonnage des entrées analogiques de haut niveau (*InType* = *HiLevel*)

Phase	Message Action	Action demandée à l'opérateur, avant d'entrer CONTINUE
1	Apply Low rng	Appliquer à l'entrée la valeur équivalente au minimum de l'échelle donné dans le champ Range. (Toujours 0 V pour ce <i>InType</i>).
2	Enter Low rng	Entrer la tension EXACTE appliquée (en Volts) dans le champ Value.
3	Apply High rng	Appliquer la valeur équivalente à environ 100 % du maximum de l'échelle donné dans le champ Range.
4	Enter High rng	Entrer la tension EXACTE appliquée (en Volts) dans le champ Value.
5	<i>Les phases 1 à 4 doivent être répétées pour chaque échelle gérée par le matériel connecté.</i>	
6	Write E2 data?	CONTINUE uniquement si les données d'étalonnage doivent être mises à jour dans le module E/S. Une interruption avant ce point arrête l'étalonnage, sans modifier la configuration de l'unité.

Séquence d'étalonnage des entrées analogiques thermocouple/mV (*InType* = *Therm*)

Phase	Message Action	Action demandée à l'opérateur, avant d'entrer CONTINUE
1	Apply Low rng	Appliquer à l'entrée la valeur équivalente au minimum de l'échelle donné dans le champ Range - toujours 0 mV pour ce <i>InType</i>).
2	Enter Low rng	Entrer la tension EXACTE appliquée (en mV) dans le champ Value
3	Apply High rng	Appliquer la valeur équivalente à environ 100 % du maximum de l'échelle donné dans le champ Range. OU: si ce type de matériel est limité à 77 % de l'échelle nominale (comme par ex. pour la carte E/S directe), appliquer 70 % du maximum de l'échelle (soit 8 mV, 17 mV, 35 mV ou 70 mV).
4	Enter High rng	Entrer la tension EXACTE appliquée (en mV) dans le champ Value
5	<i>Les phases 3 à 4 doivent être répétées pour chaque échelle gérée par le matériel connecté.</i>	
6	Apply switches	Connecter l'entrée suivant la figure 2-22, les deux commutateurs étant ouverts.
7	Enter Vcjc	Voir la figure. Mesurer Vcjc et entrer la valeur en Volts dans le champ Value.
8	Apply switches	Fermer le commutateur SW1 et laisser SW2 ouvert.
9	Enter Vbase	Mesurer Vcjc (qui est alors "Vbase" du transistor CJC) et entrer la valeur (en Volts) dans le champ Value.
10	Apply switches	Ouvrir le commutateur SW1 et fermer le commutateur SW2.
11	Enter Vcjc	Mesurer Vcjc et entrer la valeur (en Volts) dans le champ Value.
12	Write E2 data?	CONTINUE uniquement si les données d'étalonnage doivent être mises à jour dans le module E/S. Une interruption avant ce point arrête l'étalonnage, sans modifier la configuration de l'unité.

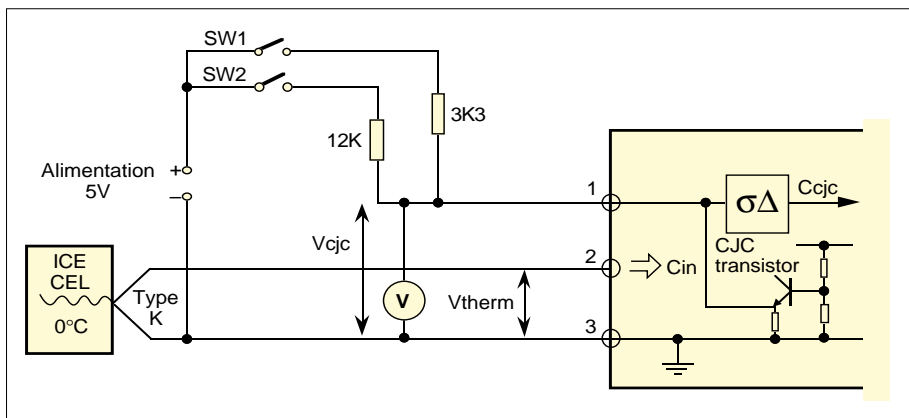


Figure 2-22 Montage d'étalonnage Thermocouple/mV

Séquence d'étalonnage des entrées analogiques résistance (InType = RTD)

Phase	Message Action	Action demandée à l'opérateur, avant d'entrer CONTINUE
1	Apply Test 1	Connecter l'entrée suivant la figure 2-23.
2	Enter R	Mesurer R entre C1 et C2 et entrer la valeur en Ohms.
3	Enter V1	Entrer la valeur mesurée de V1, en mV
4	Enter V3	Entrer la valeur mesurée de V3, en mV
5	Apply Test 2	Connecter l'entrée suivant la figure 2-24. Utiliser une valeur de résistance R_{ni} égale au maximum de l'échelle (= champ Range).
6	Enter R	Mesurer R entre C1 et C2 et entrer la valeur en Ohms.
7	Enter V1	Entrer la valeur mesurée de V1, en mV
8	Enter V3	Entrer la valeur mesurée de V3, en mV
9	Write E2 data?	CONTINUE uniquement si les données d'étalonnage doivent être mises à jour dans le module E/S. Une interruption avant ce point arrête l'étalonnage, sans modifier la configuration de l'unité. NOTA: pour les RTD, il y a une écriture séparée dans l'E ² PROM pour chaque échelle.
10	<i>Les phases 1 à 9 doivent être répétées pour chaque échelle gérée par le matériel connecté.</i>	

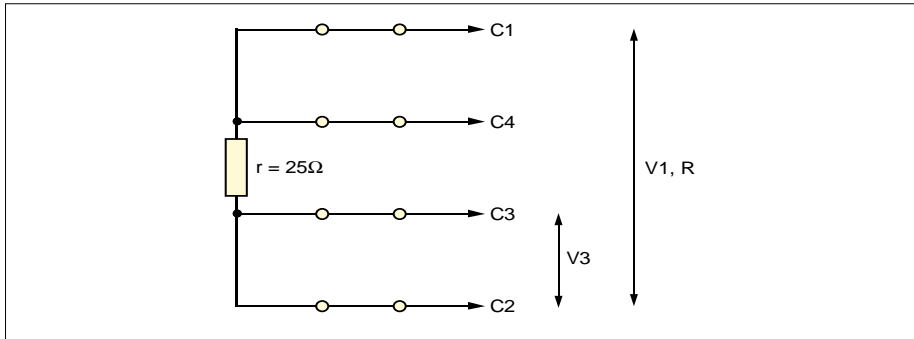


Figure 2-23 Montage d'étalonnage RTD - Essai 1

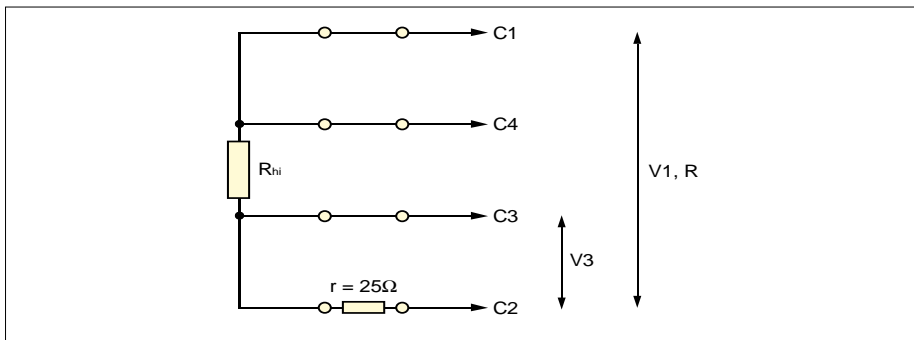


Figure 2-24 Montage d'étalonnage RTD - Essai 1

Séquence d'étalonnage des entrées analogiques fréquence (InType = Freq)

Phase	Message Action	Action demandée à l'opérateur, avant d'entrer CONTINUE
1	Apply High rng	Appliquer une fréquence égale au maximum de l'échelle (toujours 30 kHz).
2	Enter High rng	Entrer la valeur exacte de la fréquence appliquée en Hz.
3	Write E2 data?	CONTINUE uniquement si les données d'étalonnage doivent être mises à jour dans le module E/S. Une interruption avant ce point arrête l'étalonnage, sans modifier la configuration de l'unité.

AO_CALIB : BLOC D'ETALONNAGE DES SORTIES ANALOGIQUES

Fonction du bloc

Le bloc AO_CALIB permet d'étalonner les voies des sorties analogiques dans les modules E/S de la série T600. Une base de données est créée dans l'instrument de la série T600 qui contient le bloc AN_OUT associé au module d'E/S à étalonner, plus un bloc AO_CALIB. Le bloc d'étalonnage s'attache au bloc AN_OUT désigné, le force à prendre le mode CALIB, puis le traite pour étalonner la voie E/S spécifiée. Grâce à son ou ses champs *Action*, le bloc d'étalonnage affiche l'une des séquences d'instructions à exécuter par l'opérateur, qui définit alors le champ *Status*. *CONTINUE*, lorsqu'il est prêt à passer à l'instruction suivante.

Dans un instrument déporté, une copie image du bloc d'étalonnage est créée, afin que la procédure d'étalonnage puisse être mise en oeuvre dans le bloc par l'intermédiaire d'une interface plein écran.

Les détails d'étalonnage des différents types de modules de sorties analogiques sont donnés dans le paragraphe *Procédures d'étalonnage* ci-dessous.

Paramètres du bloc

Les symboles utilisés dans la table 2-14 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.










Paramètre	Fonction	Unités	Etat
Action (<i>Champ invisible</i>)	Donner des instructions d'étalonnage à l'opérateur	Menu	
Value		Menu	
Status	Valeurs entrées par l'opérateur demandées		
Alarms	Etat de la procédure d'étalonnage (& Abandon I/P)	Menu	
Software	Données corrompues/Défaut de communication	V/F	  
Failed	Abandon procédure étalonnage (priorité par défaut=6)	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Liaison à l'adresse ISB T600 du module E/S	Entier	
Channel	Spécifie le n° de voie dans le module E/S	Entier	
OutType	Spécifie type de sortie analogique en cours d'étalonnage		Menu 
Range	Signale l'échelle matériel en cours d'étalonnage	Menu	
AN_OUT	Nom du bloc AN_OUT de rattachement	Alphanumérique	
AO_Alarm	Indique si config. voie correcte/AN_OUT en alarme	Menu	

Table 2-14 Paramètres du bloc AO_CALIB

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Action (partie 1 - Visible). (Done/Supply/Enter/Write - Terminé/Fournir/Entrer/Ecrire). Permet d’afficher l’un des quatre messages courts de texte, qui, en combinaison avec un message affiché par la seconde partie (invisible) du champ *Action* qui se trouve en dessous, permet de donner un certain nombre d’instructions à l’opérateur.

“**Done**” (terminé) est affiché lorsque le bloc est au repos, c’est à dire avant le début de l’étalonnage ou lorsque l’étalonnage est terminé. Voir paragraphe suivant, *Action (partie 2 - invisible)*.

Action (partie 2 - Invisible). ([blank]/High rng/Low rng/E2 data? - [blanc]/Echelle haute/Echelle basse/données E2?) Ce champ invisible fournit la seconde partie d’une instruction opérateur, par ex. **Supplying Low rng** (Fournir échelle basse). L’opérateur exécute l’instruction donnée, et sélectionne ensuite *CONTINUE* dans le paramètre *Status* pour indiquer au bloc que l’instruction a été dûment exécutée. Voir les détails au paragraphe *Procédures d’étalonnage*.

Les instructions *Action* sont très brèves, et l’opérateur est supposé se référer au présent manuel pour une description complète des actions à entreprendre.

Value (Valeur). Lorsqu’il est demandé à l’opérateur d’entrer une valeur, la valeur doit être entrée dans ce champ.

Status (Etat). (WAITING/CONTINUE/PROCESSING/ABORT) Montre à l’opérateur l’état de la procédure d’étalonnage, et lui permet également de l’abandonner.

- **WAITING (EN ATTENTE).** Affiché chaque fois que le bloc AO_CALIB attend une action de l’opérateur.
- **CONTINUE (CONTINUER).** Doit être entré par l’opérateur, lorsque l’action en attente est achevée.
- **PROCESSING (TRAITEMENT EN COURS).** Affiché lorsque le bloc AO_CALIB est sur le point d’exécuter une action.
- **ABORT (ABANDON).** La procédure d’étalonnage est abandonnée lorsque ce champ est sélectionné.

Alarms (Alarmes). Voir la description générale du champ alarme à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Failed (Echec).** Indique un abandon de la procédure d’étalonnage. Cette alarme de priorité 6 est automatiquement définie par le bloc et signifie que, si l’alarme est déclenchée, celle-ci doit être acquittée avant de poursuivre.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active à la priorité la plus haute du bloc.

SiteNo (N° de site). Permet de spécifier l’adresse (1 à 8) sur le bus série interne des instruments de la série T600, dont la sortie analogique doit être étalonnée. Le bloc

AO_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_OUT, dès qu'il lui est rattaché, mais *SiteNo* peut être modifié ultérieurement.

Channel (Voie). Permet de définir le numéro (1 à 8) de voie de sortie analogique à étalonner. Le bloc AO_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_OUT, dès qu'il lui est rattaché, mais *Channel* peut être modifié ultérieurement.

OutType (Type de sortie). (Volts/mA) Permet de spécifier le type de sortie analogique à étalonner. Le bloc AO_CALIB copie automatiquement ce champ du champ correspondant dans le bloc AN_OUT, dès qu'il lui est rattaché, mais *OutType* peut être modifié ultérieurement.

NOTA. L'ensemble de la procédure d'étalonnage pour un *OutType* particulier doit être effectuée jusqu'au bout pour toutes les échelles prises en compte, ou abandonnée. Il n'est pas possible de n'étalonner qu'une partie des échelles.

Range (Echelle). (0-1.25V/0-2.5V/0-5V/0-10V/0-2.5mA/0-5mA/0-10mA/0-20mA) Champ de lecture uniquement signalant laquelle des échelles matériel de la voie spécifiée est en cours d'étalonnage. Voir le nota du paragraphe *OutType* ci-dessus.

AN_OUT. Spécifie le nom du bloc AN_OUT auquel le bloc AO_CALIB doit être rattaché. Une fois qu'un nom de bloc valable a été sélectionné, le bloc AO_CALIB s'y rattache et le force à passer en mode CALIB.

AO_ALARM. (Invalid/Bad Loop/In Alarm/OK) Indique l'état du bloc d'étalonnage.

- **Invalid (Incorrect)** signifie que la configuration pour la voie est incorrecte ou qu'aucun bloc AN_OUT n'a été trouvé.
- **Bad Loop (Erreur boucle)** signifie que les blocs AN_OUT et AO_CALIB ne sont pas exécutés dans la même tâche utilisateur ("boucle"), ce qui doit être le cas pour pouvoir effectuer l'étalonnage.
- **In Alarm (En alarme)** signifie que le bloc AN_OUT rattaché est en alarme.
- **OK** est affiché dans tous les autres cas.

Procédures d'étalonnage

Attention

Seul le personnel qualifié utilisant des équipements homologués est autorisé à ré-étalonner le matériel.

Pour lancer le processus d'étalonnage, entrer le nom du bloc E/S "source" dans le champ *AN_OUT* du bloc AO_CALIB. Si le bloc source peut être trouvé, le bloc d'étalonnage s'y rattache et copie ensuite les autres champs, le cas échéant - *SiteNo*, *Channel*, *OutType*,

Range. Le champ *Action* affiche **Done**, et le champ *Status* indique **WAITING**. Commencer le processus d'étalonnage est mettant *Status* sur **CONTINUE**.

Le paragraphe ci-après présente sous forme de table chacune des commandes données par le bloc et les actions à entreprendre par l'opérateur. *OutType* doit être défini en fonction du matériel avant le début de la procédure d'étalonnage, dans la mesure où chaque type est étalonné au cours d'une séquence spécifique.

Séquence d'étalonnage des sorties analogiques de haut niveau

Phase	Message Action	Action demandée à l'opérateur, avant d'entrer CONTINUE
1	Supplying Low rng	Mesurer la tension/courant de sortie générés
2	Enter Low rng	Entrer la tension ou le courant EXACTS mesurés (en Volts ou mA) dans le champ Value.
3	Supplying High rng	Mesurer la tension/courant de sortie générés
4	Enter High rng	Entrer la tension ou le courant EXACTS mesurés (en Volts ou mA) dans le champ Value.
5	<i>Les phases 1 à 4 doivent être répétées pour chaque échelle prise en compte par le matériel connecté.</i>	
6	Write E2 data?	CONTINUE uniquement si les données d'étalonnage doivent être mises à jour dans le module E/S. Une interruption avant ce point arrête l'étalonnage, sans modifier la configuration de l'unité.

AN8_OUT : BLOC DE SORTIE ANALOGIQUE A 8 VOIES

Fonction du bloc

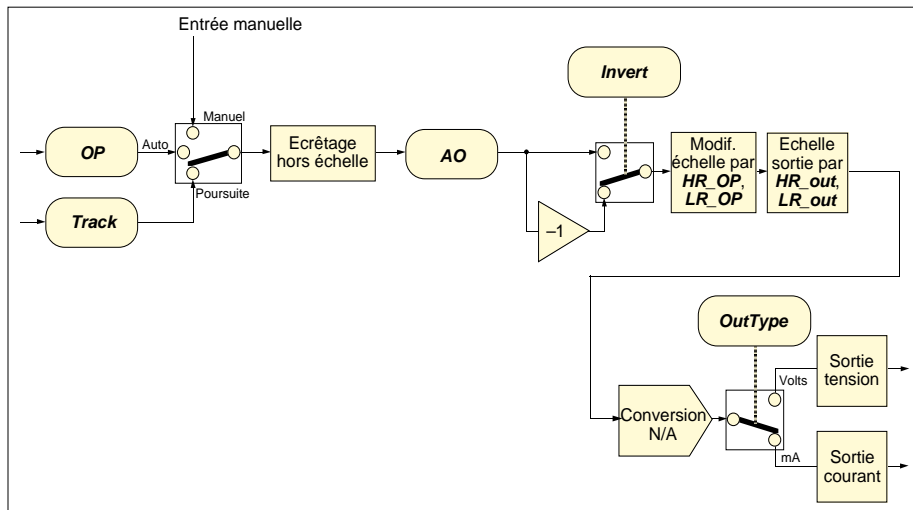


Figure 2-25 Schéma du bloc AN8_OUT

Voir figure 2-25. Le bloc de sortie analogique à 8 voies permet de convertir une variable analogique mise à l'échelle (nombre à point flottant) d'un schéma de boucles en l'une des huit sorties de courant (sélectionnable par logiciel) sur le T100/102. Le bloc dispose des contrôles Auto/Manuel/Poursuite, Inversion, Alarmes et d'un écrêtage hors échelle.

Paramètres du bloc

Les symboles utilisés dans la table 2-15 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUEL/POURSUIITE). Mode de fonctionnement en cours.

Fallback (Reprise). (AUTO/MANUEL). Le mode de fonctionnement de retour (supprimé).

OP. Demande de sortie. Entrée dans le bloc à partir du schéma de boucles, et la source AO en mode Auto.

HR_OP, LR_OP. Echelle haute et basse de AO en unités physiques. HR_OP et LR_OP permettent de définir deux points sur une échelle linéaire en unités physiques qui sont















Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Fallback	Mode de fonctionnement supprimé	Menu	
OP	Demande de sortie	Eng	
HR_OP, LR_OP	Echelle haute et basse AO	Eng	
AO	Sortie analogique	Eng	
Track	Valeur de poursuite	Eng	
SelTrack	Sélection mode poursuite	V/F	
Invert	Sélection inversion	V/F	
NotAuto	Etat mode PAS Auto	V/F	
Alarmes			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut Module Txxx	V/F	
CctFault	Défaut circuit (Sortie circuit ouvert)	V/F	
OvrDrive	Condition surcharge sortie	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Site du T100	Entier	
OutType	Type de sortie (Volts/mA)	Menu	
Channel	Numéro de voie		
HR_out	Point d'étalonnage HR	Eng	
LR_out	Point d'étalonnage LR	Eng	
Status	Etat communicat./matériel	(AB)CD Hex	
Reset	Drap. transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC T100 au module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur vers. d'étalonn. des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
		V/F — 4	
		V/F — 8	
Options	(non-utilisées)		
Channel	Numéro de voie (1 à 8) du module T151	Entier	

Table 2-15 Paramètres du bloc AN8_OUT

projetés respectivement sur les points *HR_out* et *LR_out* sur une échelle linéaire (milliampères). Ces deux échelles sont utilisées par le bloc pour calculer le signal de sortie de AO. Si l'inversion n'est pas appliquée (*Invert* FALSE) L'équation de conversion équivalente est la suivante :

$$\text{Sortie} = LR_out + \left[\frac{AO - LR_OP}{HR_OP - LR_OP} \times (HR_out - LR_out) \right]$$

Lorsque *Invert* est TRUE (vrai), la conversion utilisée est la suivante:

$$\text{Sortie} = HR_out - \left[\frac{AO - LR_OP}{HR_OP - LR_OP} \times (HR_out - LR_out) \right]$$

AO. Sortie analogique. Sortie du bloc avant l'inversion (si spécifiée) et modification de l'échelle, mais après écrêtage hors échelle.

Track (Poursuite). Source de AO en mode poursuite.

SelTrack (Sélection poursuite). Permet de sélectionner le mode poursuite.

Invert (Inversion). Permet de sélectionner l'inversion. Inverse le sens du signal de sortie pour le fonctionnement à sécurité intégrée et pour les vannes à action inverse.

NotAuto (Pas auto). VRAI lorsque le bloc ne fonctionne pas en mode Auto.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matériel est déclenchée, lorsque l'un des bits du paramètre *Status* est mis à 1, par exemple, en cas de coupure de courant, de type de module incorrect, de défaut d'un circuit physique, etc.

Une alarme matériel provoque également le clignotement continu de la LED concernée sur la face avant du T100, lorsque le commutateur SB1/SW5 est sur Off.

■ **UndDrive (Condition de sous-charge en sortie).** Sortie écrêtée à 0 mA.

■ **OvrDrive (Condition de surcharge en sortie).** Sortie écrêtée à 22 mA.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de définir l'emplacement du module entrées/sorties correspondant d'un T100/T102 (1 à 15). Ce champ est actif dans une base de donnée T100/T102, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro* ; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100/T102 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S en *mémoire cache* correspondants copient automatiquement les paramètres "réels".

NOTA. Dans la mesure où le module de sortie analogique à 8 voies (T151) occupe un emplacement double hauteur, le paramètre *SiteNo* ne peut être qu'impair.

OutType (Type de sortie). (Volts/mA). Permet de définir le type de sortie en mA. Notez que l'option *Volts* ne peut être sélectionnée pour ce bloc.

HR_out, LR_out (Echelle haute & basse du signal de sortie). *HR_out* et *LR_out* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR_OP* et *LR_OP* sur une échelle linéaire AO en unités physiques. Voir les détails à la section *HR_OP, LR_OP* ci-dessus. Noter que *HR_out* ne peut dépasser 22 mA.

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-15.

Channel (Voie). Définit le numéro de voie (1 à 8) dans le module entrées/sorties associé du T151.

RTD_6 : BLOC D'ENTREE SONDE A RESISTANCE

Fonction du bloc

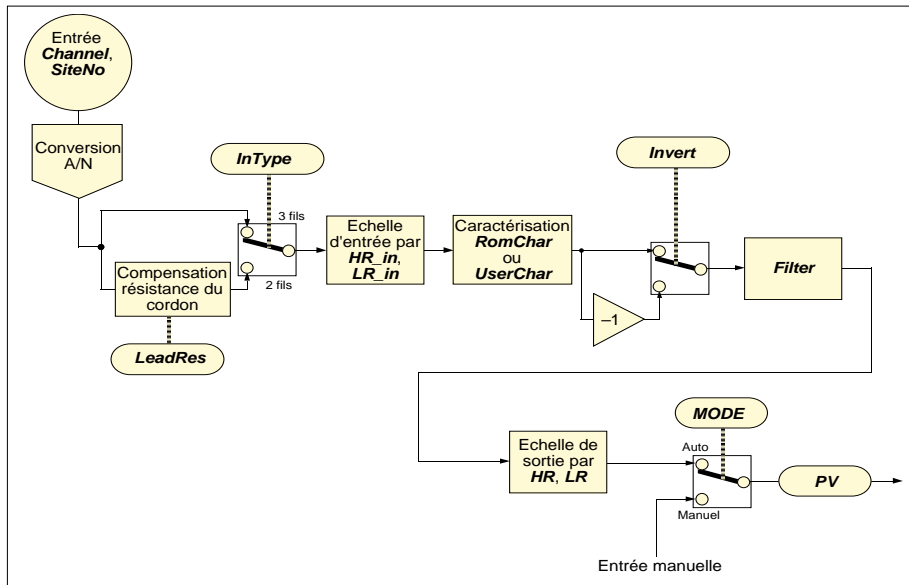


Figure 2-26 Schéma du bloc RTD_6

Voir figure 2-26. Le bloc RTD_6 traite une entrée de mesure de résistance par l'intermédiaire de l'une des voies du module d'entrée de sonde à résistance à six voies d'un T113. Le bloc applique une compensation de résistance du câble définie par l'utilisateur, calcule la température à partir de tables de caractérisation résidentes (*RomChar*) ou personnalisées (*UserChar*), et sort une valeur de température *PV* en unités physiques mise à l'échelle.

Le mode manuel, les alarmes absolues hautes et basses et hors échelle, le filtrage du premier ordre, ainsi que l'inversion des signaux de sorties sont également disponibles dans le bloc RTD_6.

NOTA. Les unités de température - °C, °F, K et R - peuvent être sélectionnées par l'intermédiaire du paramètre *IP_type* du bloc T100 correspondant. Voir les détails au chapitre 10, *Blocs de fonction CONFIG*.

Paramètres du bloc

Les symboles utilisés dans la table 2-16 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
PV	Variable procédé (sortie bloc)	DegC	
HR, LR	Echelle haute & basse de PV	DegC	
HAA, LAA	Niveaux d'alarme absolue haute & basse PV	DegC	
Filter	Durée du filtre du premier ordre (0 à 250 sec)	Sec.	
Invert	Inversion du signal d'entrée	V/F	
LeadRes	Valeur résistance câble (2 fils uniquement)	Ohms	
RomChar	Caractérisation sélectionnée	Menu	
UserChar	Caractérisation 16 points définie par l'utilisateur	Alphanumérique	
Alarmes			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts Module Txxx/défaut circuit/alim., etc.	V/F	
HiLevel	Alarme haute absolue (HAA)	V/F	
LoLevel	Alarme basse absolue (LAA)	V/F	
OutOfRange	PV en dehors de l'échelle HR-LR (>5%)	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Emplacement fond de panier du module (1,3,5, ...15)	Entier	
Channel	Numéro de voie sur le module (1 à 6)	Entier	
InType	Type d'entrée (2 ou 3 fils)	Menu	
HR_in	Point d'étalonnage HR		
LR_in	Point d'étalonnage LR		
Status	Etat communicat./matériel	(A)BCD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC du T100)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version d'étalonnage des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompue	V/F — 2	
		V/F — 4	
BadRef	Défaut tension de référence	V/F — 8	
BlockErr	Données RAM corrompues /mémoire insuffisante	V/F — 1	B
Break	Entrée circuit ouvert	V/F — 2	
Range	Valeur d'entrée hors échelle	V/F — 4	
		V/F — 8	
BadSync	Erreur de communication module et unité de base	V/F — 1	A
BadChks		V/F — 2	
FifoNful		V/F — 4	
		V/F — 8	

Table 2-16 Paramètres du bloc RTD_6

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (Auto/Manuel). Mode de fonctionnement. En mode Auto, *PV* est définie par l'entrée du champ mise à l'échelle, caractérisée et filtrée. En mode Manuel, vous pouvez saisir une *PV* en conduite pour annuler le conditionnement du signal.

PV. Sortie bloc (Variable procédé). En lecture uniquement en mode automatique.

HR, LR. Echelle haute & basse de *PV* en unités physiques. *HR* et *LR* permettent de définir deux points sur une échelle linéaire en unités physiques qui sont projetés respectivement sur les points *HR_in* et *LR_in* sur une échelle linéaire (millivolts). Ces deux échelles sont utilisées par le bloc pour calculer la sortie *PV* du signal mesuré à l'entrée.

L'équation de conversion équivalente est la suivante :

$$PV = LR + \left[\frac{\text{Input} - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

HAA, LAA. Limites d'alarmes absolues hautes & basses pour *PV*, avec une hystérésis intégrée de 0,5 % à l'intérieur des limites.

Filter (Filtre). Constante de temps du filtre du premier ordre (0 à 250 sec.) appliquée au signal avant la mise à l'échelle de la sortie.

Invert (Inversion). Si TRUE (vrai) *HR* correspond à *LR_in* et *LR* à *HR_in*, c'est à dire que le signal de sortie *PV* est inversé (mais pas en mode manuel). Voir ci-dessus, le paragraphe *HR, LR*.

LeadRes. Valeur de résistance du câble (Ω). Utilisé conjointement avec les mesures 2 fils uniquement pour compenser la résistance du câble.

RomChar. (None/PRT/CU10). Permet de sélectionner le type de caractérisation dans la bibliothèque ROM. La table 2-18 du bloc ANIN_6 donne les échelles PRT et CU10.

UserChar. Permet de définir une table de linéarisation/caractérisation personnalisée à 16 points (*x,y*). Entrer le nom de repère du bloc UCHAR dans le schéma de boucles contenant la table requise.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Alarme logiciel déclenchée après une erreur de total de contrôle dans un bloc, (corruption de sa base de données) ou un défaut de communication du bloc déporté (en mémoire cache).
- **Hardware (Matériel).** Une alarme matériel est déclenchée, lorsque l'un des bits du paramètre *Status* est mis à 1, par exemple, en cas de coupure de courant, de type de module incorrect, de défaut d'un circuit physique, etc.
- **HiLevel, LoLevel (Niveau haut, Niveau bas).** Ces limites d'alarmes sont définies par les paramètres *HAA* et *LAA*. Il y a une bande d'hystérésis de 0,5 % sur chaque limite d'alarme, ce qui permet une transition nette entre la condition d'alarme et de non-alarme.
- **Outrange (Hors échelle).** Cette alarme est déclenchée dès que *PV* atteint une valeur de 5 % au-dessus ou en-dessous de *HR* ou *LR*. Il y a une hystérésis de 0,5 % pour chaque valeur de l'échelle.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Relie le bloc à la paire de numéros de site du châssis T102/T103 occupé par le module double hauteur T113. Saisir un nombre impair de 1 à 15. Une fois qu'un numéro de site de T100 valable a été entré, il ne peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs *SiteNo* des blocs entrées/sorties *en mémoire cache* correspondants copient automatiquement les paramètres "réels".

Channel (Voie). Permet de définir le numéro de voie (1 à 6).

InType (Type d'entrée). (3 WIRE/2 WIRE (3 FILS/2 FILS)). Permet de définir le type d'entrée. La compensation de la résistance du câble n'est appliquée que pour les mesures à 2 fils.

HR_in, LR_in. Echelle haute & basse du signal mesuré de l'entrée. *HR_in* et *LR_in* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR* et *LR* sur une échelle linéaire *PV* en unités physiques. Voir les détails à la section *HR, LR* ci-dessus.

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-16.

ANIN_6 : BLOC D'ENTREE ANALOGIQUE

Fonction du bloc

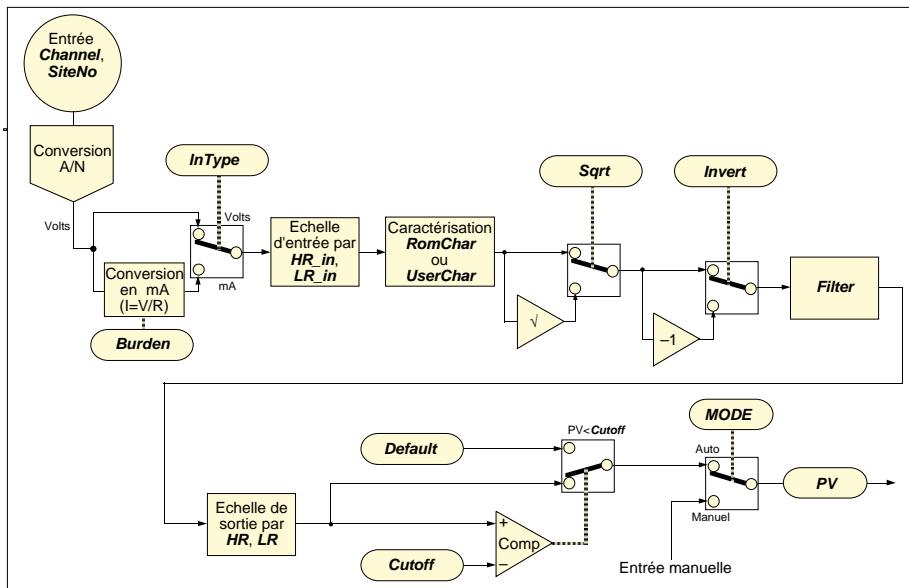


Figure 2-27 Schéma du bloc ANIN_6

Voir le schéma de la figure 2-27. Le bloc ANIN_6 convertit une entrée de tension - par l'intermédiaire de l'une des voies d'un module d'entrée analogique haut niveau à 6 voies T124 - en une valeur à point flottant en unités physiques. L'utilisation d'une résistance de charge externe plus une compensation logicielle permet de traiter des entrées de courant.

Le bloc peut calculer des températures à partir d'entrées de thermocouples conditionnées, en utilisant des tables de caractérisation (*RomChar* - voir table 2-18) résidentes ou définies par l'utilisateur (*UserChar*).

Le mode manuel, la détection de coupure, les alarmes absolues hautes et basses et hors échelle, le calcul de racines carrées, le filtrage du premier ordre, ainsi que l'inversion des signaux de sorties sont également disponibles dans le bloc ANIN_6.

NOTA. Les unités de température - °C, °F, K et R - peuvent être sélectionnées par l'intermédiaire du paramètre *IP_type* du bloc T100 correspondant. Voir les détails au chapitre 10, *Blocs de fonction CONFIG*.

Paramètres du bloc

Les symboles utilisés dans la table 2-17 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement du bloc	Menu	
PV	Variable procédé (sortie bloc)	Eng	
HR, LR	Echelle haute & basse de PV	Eng	
HAA, LAA	Niveaux d'alarme absolue haute & basse PV	Eng	
Filter	Constante du filtre du premier ordre (0 à 250 sec)	Sec.	
Invert	Inversion du signal d'entrée	V/F	
Sqrt	Sélection fonction racine carrée	V/F	
Cutoff	Seuil bas de PV	Eng	
Default	Valeur PV si Sqrt < Cutoff	Eng	
RomChar	Caractérisation sélectionnée	Menu	
UserChar	Caractérisation 16 points définie par l'utilisateur	Alphanumérique	
Alarmes			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts Module Txxx/défaut circuit/alim., etc.	V/F	
HiLevel	Alarme haute absolue (HAA)	V/F	
LoLevel	Alarme basse absolue (LAA)	V/F	
OutOfRange	PV hors échelle HR-LR ou échelle A/N ou <BrkThres	V/F	
Combined	Alarme combinée du bloc	V/F	
SiteNo	Emplacement fond de panier du module (1,3,5, ...15)	Entier	
Channel	Numéro de voie sur le module (1 à 6)	Entier	
InType	Type d'entrée (2 ou 3 fils)	Menu	
HR_in	Point d'étalonnage HR (unités InType)		
LR_in	Point d'étalonnage LR (unités InType)		
BrkThresh	Seuil électrique détection de rupture (unités InType)		
Burden	Valeur résistance de charge externe (mA uniquement)	Ohms	
Status	Etat communicat./matériel	(A)BCD Hex	
Reset	Drapeau transitoire défini à la mise sous tension	V/F — 1	D
Missing	Module Txxx - manquant	V/F — 2	
CommsErr	Défaut communication (UC /module fond panier)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version d'étalonnage des données (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
		V/F — 4	
BadRef	Défaut tension de référence	V/F — 8	
BlockErr	Données RAM corrompues /mémoire insuffisante	V/F — 1	B
Break	PV < seuil bas (BrkThres)	V/F — 2	
Range	PV hors HR-LR ou entrée hors échelle A/N	V/F — 4	
		V/F — 8	
BadSync	Erreur de communication module et unité de base	V/F — 1	A
BadChks		V/F — 2	
FifoNful		V/F — 4	
		V/F — 8	

Table 2-17 Paramètres du bloc ANIN_6

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Mode. (Auto/Manuel). Mode de fonctionnement. En mode Auto, *PV* est définie par l’entrée du champ mise à l’échelle, caractérisée et filtrée. En mode Manuel, vous pouvez saisir une *PV* en conduite pour annuler le conditionnement du signal.

PV. Sortie bloc (Variable procédé). En lecture uniquement en mode automatique.

HR, LR. Echelle haute & basse de *PV* en unités physiques. *HR* et *LR* permettent de définir deux points sur une échelle linéaire en unités physiques qui sont projetés respectivement sur les points *HR_in* et *LR_in* sur une échelle linéaire (millivolts). Ces deux échelles sont utilisées par le bloc pour calculer la sortie *PV* du signal mesuré à l’entrée.

L’équation de conversion équivalente est la suivante :

$$PV = LR + \left[\frac{Input - LR_in}{HR_in - LR_in} \times (HR - LR) \right]$$

HAA, LAA. Limites d’alarmes absolues hautes & basses pour *PV*, avec une hystérésis intégrée de 0,5 % à l’intérieur des limites.

Filter (Filtre). Constante de temps du filtre du premier ordre (0 à 250 sec.).

Invert (Inversion). Si TRUE (vrai) *HR* correspond à *LR_in* et *LR* à *HR_in*, c’est à dire que le signal de sortie *PV* est inversé (mais pas en mode manuel). Voir ci-dessus, le paragraphe *HR, LR*.

Sqrt. Permet de sélectionner la fonction racine carrée.

Cutoff (Coupure). Permet de spécifier une valeur de seuil bas pour *PV*, auquel cas *PV* passe à Default (par défaut).

Default (Par défaut). Valeur adoptée par *PV* lorsque la valeur calculée est inférieure à *Cutoff* (coupure).

RomChar. (None/J/K/T/S/R/E/B/N/W/W5/W3/MoRe/PRT/CU10). Permet de sélectionner le type de caractérisation dans la bibliothèque ROM. Voir les détails table 2-18.

UserChar. Permet de définir une table de linéarisation/caractérisation personnalisée à 16 points (x,y). Entrer le nom de repère du bloc UCHAR dans le schéma de boucles contenant la table requise.

Type	Plage (°C)	Type	Plage (°C)
J	-200 à 1200	N	0 à 1300
K	-270 à 1370	W	1000 à 2300
T	-2270 à 400	W ₃	0 à 2490
S	-50 à 1760	W ₅	0 à 2300
R	-50 à 1760	MoRe	0 à 1990
E	-270 à 1000	PTR	-220 à 1050
B	0 à 1820	CU10	-20 à 255

Table 2-18 Types et échelles de caractérisation RomChar

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Alarme logiciel déclenchée après une erreur de total de contrôle dans un bloc, (corruption de sa base de données) ou un défaut de communication du bloc déporté (en mémoire cache).
- **Hardware (Matériel).** Une alarme matériel est déclenchée, lorsque l'un des bits du paramètre *Status* est mis à 1, par exemple, en cas de coupure de courant, de type de module incorrect, de défaut d'un circuit physique, etc.
- **HiLevel, LoLevel (Niveau haut, Niveau bas).** Déclenchée si les limites d'alarme absolue haute (*HAA*) et basse (*LAA*) sont dépassées. Il y a une bande d'hystérésis de 0,5 % sur chaque limite d'alarme, ce qui permet une transition nette entre la condition d'alarme et de non-alarme.
- **Outrange (Hors échelle).** Cette alarme est déclenchée dès que *PV* atteint une valeur au-dessus ou en-dessous de *HR* ou *LR* ou si la valeur de *PV* tombe en dessous de celle du paramètre *BrkThres*. Aucune hystérésis n'est appliquée. *Outrange* passe également à 1 si le signal d'entrée dépasse l'échelle du convertisseur analogique/numérique (± 11 V nom.).
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Relie le bloc à la paire de numéros de site du châssis T102/T103 occupé par le module double hauteur T124. Saisir un nombre impair de 1 à 15. Une fois un numéro de site valable entré, il ne peut être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T100 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs *SiteNo* des blocs E/S *en mémoire cache* correspondants copient automatiquement les paramètres "réels".

Channel (Voie). Permet de définir le numéro de voie (1 à 6).

InType (Type d'entrée). (Volts/mA) Définit le type d'entrée tension ou courant.

HR_in, LR_in. Echelle haute & basse du signal mesuré de l'entrée. *HR_in* et *LR_in* définissent deux points sur une échelle linéaire qui sont projetés sur les points *HR* et *LR* sur une échelle linéaire *PV* en unités physiques. Voir *HR*, *LR* ci-dessus.

BrkThres (Seuil de rupture). Définit un seuil électrique bas pour la détection de rupture. Les unités sont définies par *InType*. Si le signal d'entrée est inférieur à ce seuil, le bit d'alarme *OutRange* et le bit *Break* du mot *Status* sont mis à 1. Si, par exemple, *InType* = mA et que les signaux varient de 4 à 20 mA, saisissez une valeur de 2 mA pour *BrkThres* pour signaler la rupture d'un fil.

Burden (Charge). Valeur de la résistance de charge externe - utilisée uniquement si le type d'entrée (*InType*) doit mesurer des mA. Le bloc utilise la valeur de *Burden* pour convertir l'entrée mV en un signal mA. Notez que l'erreur introduite par la résistance parasite interne du module est automatiquement corrigée dans les calculs.

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails, table 2-17.

DGIN_1: BLOC D'ENTRÉE ANALOGIQUE À VOIE UNIQUE

Fonction du bloc

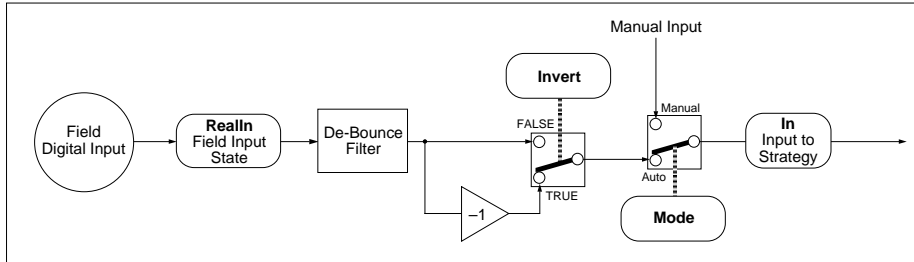


Figure 2-28 Schéma du bloc DGIN_1

Voir figure 2-28. Le bloc DGIN_1 permet d'affecter un signal logique unique à un schéma de boucles, tout en disposant des fonctions Auto/Manuel, d'inversion et d'antirebond.

L'entrée peut être configurée soit comme contact ou détection de tension. Les contacts utilisent la sortie de l'alimentation du transmetteur, qui comprend une alarme de surcharge. L'entrée de détection de tension comprend un seuil de commutation variable.

Paramètres du bloc

Les symboles utilisés dans la table 2-19 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode fonctionnement actif	Menu	
Realln	Etat de l'entrée (réel). VRAI = 'haut'/'1'	V/F	
Invert	VRAI inverse l'entrée au schéma de boucles	V/F	
InactStr	Chaine 8 caractères copiée dans <i>InStr</i> si <i>In</i> FAUX	Alphanumérique	
ActStr	Chaine 8 caractères copiée dans <i>InStr</i> si <i>In</i> VRAI	Alphanumérique	
AlmWhen	<i>In</i> state entraînant une alarme d'entrée	Menu	
InStr	Etat entrée sans rebond comme chaîne caractères	Alphanumérique	
In	Etat entrée sans rebond sous forme booléenne	V/F	
NotIn	Inverse de <i>In</i>	V/F	
Alarms			
Software	Anomalie mémoire des données RAM du bloc	V/F	
Hardware	Défauts module-T ou alim transmetteur	V/F	
Input	<i>In</i> state = <i>AlmWhen</i>	T/F	
Combined	Combinaison de tous les bits d'alarme	T/F	
SiteNo	Emplacement module entrée dans T103 (1-16)	Entier	
Channel	Sélection voie d'entrée (0 - 8; '0' = non affectée)	Entier	
InType	Type d'entrée du module	Menu	
Thresh	Tension seuil pour le module (1 - 10 volts)	Entier	

Table 2-19 suite ...

...Table 2-19 suite



Paramètre	Fonction	Unités	Etat
DeBounce Status	Période antirebnd de la voie (0 - 25 sec) Etat communications/matériel	Numérique (AB)CD hex	 
Reset	Drapeau temporaire mis à 1 à la mise sous tension	V/F — 1	D
Missing	Module T manquant	V/F — 2	
CommsErr	Défaut communication (UC T103 CPU vers module)	V/F — 4	
BadType	Type de module erroné sur le site	V/F — 8	
WrongCal	Erreur version données d'étalonnage (EEPROM)	V/F — 1	C
BadCal	Données d'étalonnage corrompues	V/F — 2	
PSUShort	Surcharge alim. transmetteur	V/F — 4	
		8	

Table 2-19 Paramètres du bloc DGIN_1

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 2-19.

Dbase, Block, Type. Voir les détails de ces champs "en-tête" à la page 2.3.

Mode (AUTO/MANUEL). En mode automatique, le champ d'état *In* suit les stimuli d'entrée des connexions des installations. En mode manuel, le champ d'état *In* peut être manipulé indépendamment des stimuli des installations.

InactStr, ActStr. Ces paramètres permettent de définir une paire de 8 chaînes de caractères (maximum) correspondant aux états FAUX et VRAI du paramètre *In*. Le paramètre *InStr* adopte la valeur correspondante de la chaîne de caractères, en fonction de l'état actif de *In*.

AlmWhen. (InactStr/ActStr/Neither) Définit l'état *In* — sous son format de chaîne de caractères — qui va déclencher un alarme d'*Entrée*. 'Neither' (aucun des deux) signifie que l'alarme d'*Entrée* n'est jamais déclenchée.

In. Etat de l'entrée sans rebond sous forme booléenne, qui peut être reliée à un schéma de boucles. *In* peut être vrai ou une forme inversée de *RealIn* suivant l'état du champ *Invert*. En mode manuel, *In* peut faire l'objet d'une écriture et d'une liaison filaire. En mode Auto, *In* ne peut faire l'objet que d'une liaison filaire sortante.

Alarms (Alarmes). Voir la description générale du champ Alarmes à la page 2-5.

■ **Software (Logiciel).** Erreur de total de contrôle dans les données RAM du bloc.

■ **Hardware (Matériel).** Une alarme matérielle est générée, lorsque l'un des bits du paramètre *Status* est mis à 1, par exemple, en cas de coupure de courant, de type de module incorrect, de surcharge de l'alimentation électrique, de défaut d'un circuit physique, etc.

■ **Input (Entrée).** Déclenchée, lorsque l'état *In* équivaut à l'état spécifié dans *AlmWhen*. Si 'Neither' a été sélectionné, cette alarme n'est jamais déclenchée.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et le numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SiteNo (N° de site). Permet de définir l'emplacement dans un T103 (1 à 16). Ce champ est actif dans une base de donnée T920/921, et lie effectivement le bloc à un site matériel. Une fois qu'un numéro de site T103 valable a été entré, il en peut pas être modifié en entrant un autre numéro de site. Pour déconnecter le bloc du site, entrer un *zéro*; le site du bloc peut alors être modifié, si nécessaire. Il est à noter que les champs du site dans le bloc T102 correspondant sont automatiquement mis à jour avec les noms de repères du bloc E/S pertinent, à mesure que les champs de n° de site sont configurés. De même, les champs de numéros de site des blocs E/S *en mémoire cache* correspondants copient automatiquement les paramètres "réels".

InType (Type d'entrée). (Volts/Contact). Définit le type d'entrée.

Thresh (Seuil). Tension seuil. Définit le seuil pour l'option entrée de tension (1 - 10 V).

DeBounce (Antirebond). Temps antirebond en secondes. Il s'agit de la durée minimale de persistance d'un changement dans le signal d'entrée logique avant que le paramètre *In* ne puisse passer au nouvel état. L'action de filtrage de *DeBounce* (Antirebond) est représentée à la figure 2-29. (Le temps antirebond minimum dépend du temps d'échantillonnage, mais se situe dans la plage de 0 à 25 secondes.)

Status (Etat). Champ binaire indiquant les erreurs de communication/matériel. Voir les détails dans la table 2-19.

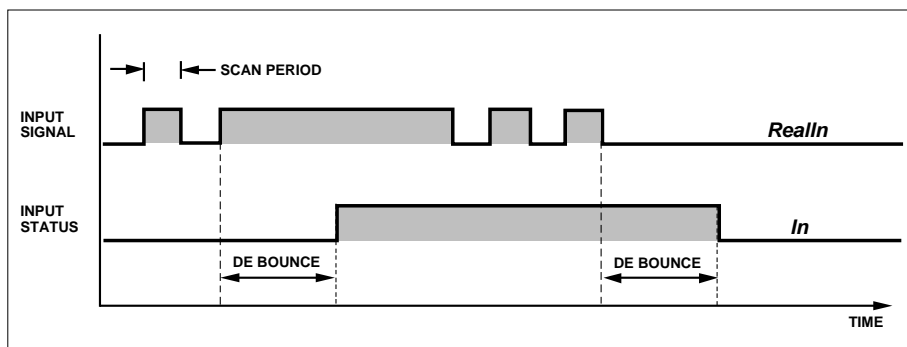


Figure 2-29 Effets de l'antirebond sur un signal d'entrée logique

[Page laissée intentionnellement blanche]

Chapitre 3 BLOCS DE FONCTION S6000

DESCRIPTION GENERALE

Cette catégorie de blocs de fonction est constituée de deux groupes: les blocs "maîtres" de la série S6000 et les blocs "esclaves" de la série SL6000. Les blocs maîtres S6000 sont décrits dans la première partie du présent chapitre; les blocs esclaves SL6000 sont décrits dans la seconde partie du chapitre à partir de la page 3-32.

BLOCS SERIE S6000 MAITRES

Interface de communication en conduite

Les blocs maîtres de la série S6000 sont un ensemble de modèles de paramètres qui fournissent une interface de communication en conduite entre les instruments à base de données LIN et les instruments du système 6000 sur le bus de données RS422 par l'intermédiaire du protocole binaire Eurotherm Systèmes. Ils permettent le contrôle par la supervision à partir de l'instrument maître, et l'utilisation des ressources du configurateur de schémas de boucles et des synoptiques. L'instrument maître peut également être utilisé avec le multiplexeur de communication 6255 d'Eurotherm Systèmes pour partager l'interface de communication RS422 avec un autre système informatique.

NOTA: Voir la liste des instruments qui gèrent les blocs maîtres S6000, table 1-2, chapitre 1.

Scrutation par exception & multi-paramètres

Chaque bloc S6000 est lié à la base de données de l'instrument correspondant à partir du champ numéro instrument (*Instr_No*) et le numéro de voie (*Chan_No*) dans les unités multi-voies. L'instrument maître scrute en permanence la base de données au cours de la conduite en utilisant à la fois le mode de scrutation par exception et une séquence de scrutation multi-paramètres (MMP) ou les deux. La séquence de scrutation précise dépend de la valeur de *MMP ratio* (rapport MMP), défini dans le paramètre *Dbase* du bloc S6000 dans le champ *Rate ms*. Au démarrage ou lorsqu'un instrument hors ligne ou hors scrutation est remis en ligne ou en scrutation, le maître effectue toujours une scrutation par exception. Dans d'autres cas, lorsque *MMP ratio* = 0, le maître n'utilise que la scrutation par exception. Si *MMP ratio* = 1, n'utilise que la scrutation multi-paramètres (qui doit être évitée pour des raisons de performances). Pour toutes les autres valeurs de *MMP ratio* (dans la plage admissible de 2 à 99), le maître tente d'effectuer une scrutation multi-paramètres après chaque scrutation par exception de *<MMP ratio>*.

Notez que chaque bloc dispose de plusieurs paramètres qui ne sont pas couverts par la scrutation par exception. Si vous sélectionnez zéro comme *MMP ratio*, ces paramètres ne seront mis à jour qu'au moment du démarrage ou de la reprise hors ligne ou s'ils ont été modifiés par l'opérateur sur un instrument de supervision.

Interaction en conduite

Les valeurs des paramètres de l'instrument peuvent être modifiées à partir de l'affichage Technicien au cours de la conduite, ou d'autres paramètres peuvent être liés aux affichages graphiques et modifiés à partir de ces objets, si la saisie est autorisée. Toutefois, il n'y a pas d'options pour télécharger une base de données complète pour configurer les instruments.

Modèles contrefaits

Les modèles des paramètres RS422 ont été conçus pour être projetés sur des types d'instruments spécifiques. Le bloc interroge le paramètre *II* (Identité Instrument), et déclenche l'alarme matériel, lorsqu'il n'y a pas concordance. Cette alarme peut également être annulée, de sorte que les modèles puissent être utilisés comme des "contrefaçons" pour d'autres types d'instruments. Les modèles contrefaits fournissent normalement une interface de conduite adéquate, aussi longtemps que les paramètres des instruments ressemblent d'assez près à la disposition du modèle principal. Toutefois, certains des paramètres de l'affichage technicien renvoient des valeurs incorrectes ou non pertinentes, et les blocs S6000 ne disposent pas des fonctions complètes de traitement des alarmes.

Surcharge de données

Les communications S6000 ont une largeur de bande limitée pour les écritures de données, et selon les cas, une base de données peut soit générer constamment des requêtes trop rapidement ou trop de requêtes à la fois pour la file d'attente interne d'écritures. Si c'est le cas, l'alarme *S6OvrLd* du bloc en-tête T1000 est alors activée. Toute écriture demandée lorsque la file d'attente est saturée est rejetée. Cette alarme ne devrait jamais se produire dans une base de données configurée correctement.

N'oubliez pas que l'alarme indique que certains éléments des données ont été perdus. Si cette alarme se déclenche, vous devez simplifier la base de données ou réduire les demandes d'écritures pour la faire disparaître. Si l'alarme se produit pendant un court instant dans le cas d'un événement spécifique, vous devez essayer d'étaler les écritures effectuées à ce moment-là.

[Page laissée intentionnellement blanche]

S6360 : BLOC MODELE DE REGULATEUR

Fonction du bloc

Le bloc S6360 permet de faire l'interface avec les instruments 6360 et 6350. Il est à noter qu'il y a une discordance mnémonique avec le 6350 - les paramètres *HD* et *LD* dans le modèle représentent *HA* et *LA* dans l'instrument. Ces paramètres sont corrects pour le 6360.

Numéros des paramètres 6360. La table 3-1 donne les numéros des paramètres pour les communications S6360. Les rectangles grisés indiquent que les paramètres sont soumis à la scrutation par exception.

0 II	1 DP	2 1H	3 1L	4 HD	5 LD	6 MN	7 SP
8 PV	9 OP	10 HA	11 LA	12 HS	13 LS	14 HO	15 LO
16 HR	17 LR	18 SL	19 EL	20 XP	21 TI	22 TD	23 2H
24 2L	25 3H	26 3L	27 MP	28 RS	29 RB	30 IC	31 SW
32 DS	33 IF	34 TS	35 ER	36 MD	37 —	38 —	39 —

Table 3-1 Numéros des paramètres pour le S6360

Paramètres du bloc

Les symboles utilisés dans la table 3-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
II	Identité de l'instrument	ABCD hex	
DP	Position du point décimal	ABCD hex	
1H, 1L	Echelle haute & basse PV	Eng	
HD, LD	Alarmes d'écart hautes & basses	Eng	
MN	Mode de fonctionnement en cours	Menu	
FALLBACK	Mode de fonctionnement supprimé	Menu	
SP	Point de consigne résultant	Eng	
PV	Variable procédé	Eng	
OP	Sortie actionneur	%	

suite...

...suite

Paramètre	Fonction	Unités	Etat
HA, LA	Limites hautes & basses alarmes absolues	Eng	
HS, LS	Limites hautes & basses point de consigne	Eng	
HO, LO	Limites hautes & basses sortie	%	
HR, LR	Limites hautes & basses consigne rapport	Numérique	
SL	Point de consigne local	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné / Circuit ouvert PV etc.	V/F	
HiLevel	Alarme haute absolue et d'écart (HA, HD)	V/F	
LoLevel	Alarme basse absolue et d'écart (LA, LD)	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
EL	Limite d'erreur	%	
XP	Bande proportionnelle	%	
TI	Temps de l'intégrale	secs/mins	
TD	Temps de la dérivée	secs/mins	
2H, 2L	Echelles hautes & basses voie 2	Eng	
3H, 3L	Echelles hautes & basses voie 3	Eng	
MP	Puissance mesurée	Eng	
RS	Consigne rapport	Numérique	
RB	Décalage rapport	Eng	
IC	Caractérisation voie d'entrée/masquage bouton	ABCD hex	
SW	Position des commutateurs de l'appareil	ABCD hex	
DS	Etat des E/S logiques	ABCD hex	
IF	Constante de temps du filtre d'entrée	secs	
TS	Temps de répétition de l'algorithme à 3 termes	secs	
ER	Terme d'erreur (PV – SP)	Eng	
MD	Etats de fonctionnement	ABCD hex	
Touche M		V/F — 1	D
Touche A		V/F — 2	
Touche R		V/F — 4	
Touche S		V/F — 8	
MP sel		V/F — 1	C
LED test		V/F — 2	
Raise		V/F — 4	
Lower		V/F — 8	
Sumcheck		V/F — 1	B
Open cct		V/F — 2	
RemEna		V/F — 4	
R/R mode		V/F — 8	
Auto		V/F — 1	A
Manual		V/F — 2	
Track		V/F — 4	
Hold		V/F — 8	

Table 3-2 Paramètres du bloc S6360

Menu de spécifications du bloc

Seuls les paramètres qui ne sont pas dans le 6360 ou dont la signification est différente figurent sur la liste. Pour avoir des informations sur les paramètres normaux de l'instrument, se reporter au manuel technique ou de référence en question.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

1H, 1L. Echelle haute & basse de *PV*. Permet de définir en unités physiques l'échelle *PV* de la variable procédé (Voie 1). *1H* est équivalent à une entrée de 10V sur la broche 13 (ou 5V sur la broche 10 lorsque le commutateur S3/1 est sur on), et *1L* est équivalent à une entrée de 0V sur la broche 13 (ou 1V sur la broche 10 lorsque S3/1 est sur on).

NOTA. Dans le 6360, et tous les instruments du système 6000, les valeurs *PV* en dehors des paramètres d'échelle haute et basse ne sont pas mesurables, et sont enregistrées comme étant égales au paramètre de l'échelle la plus proche. Par ex., si *1H* est mis à 300°C pour une entrée de 10V, alors une entrée de 11V sera encore lue comme étant à 300°C.

HD, LD. Valeurs des alarmes d'écart hautes et basses. (Il faut noter que ces paramètres reviendront aux valeurs de *HA* et *LA* dans les 6350).

MN. (HOLD/TRACK/MANUAL/AUTO/RATIO/CASCADE/F. MAN/F. AUTO - MAINTENIR/POURSUITE/MANUEL/AUTO/RAPPORT/CASCADE/MANUEL FORCE/AUTO FORCE). Mode de fonctionnement en cours. Les modes manuel, automatique, cascade et ratio peuvent être sélectionnés dans ce champ.

FALLBACK (REPRISE). (HOLD/TRACK/MANUAL/AUTO/RATIO/CASCADE/F. MAN/F. AUTO - MAINTENIR/POURSUITE/MANUEL/AUTO/RAPPORT/CASCADE/MANUEL FORCE/AUTO FORCE). Mode de fonctionnement supprimé.

HA, LA. Valeurs des alarmes hautes et basses absolues. Ne sont pas utilisées sur les 6350 (voir *HD* et *LD*).

HR, LR. Limites hautes & basses de consigne rapport (ratio). Uniquement pertinentes, lorsque le 6360 fonctionne en mode Ratio (c'est à dire si le commutateur S2/1 est sur ON). *HR* et *LR* limitent absolument l'échelle pour laquelle le paramètre *RS* de consigne rapport ajusté peut être modifié. Si *HR* est défini comme étant égal à *LR*, *RS* est verrouillé à sa valeur.

SL. Point de consigne local. Permet de définir en unités physiques le point de consigne interne avant l'application de l'ajustement. *SL* peut être calculé à partir de différentes sources suivant le mode de fonctionnement du 6360 et l'état des commutateurs S2/1 et S2/6, voir les détails dans le *manuel technique du régulateur 6360*.

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Ce type d'alarme est déclenché dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs image uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422

■ **Hardware (Matériel).** Ce type d'alarme est déclenché dans les conditions suivantes :

- Paramètre *II* incorrect renvoyé (c'est à dire pas 350X ou 360X)
- Indication erreur de total de contrôle des données (sumcheck) de l'instrument (Paramètre *DS* Bit 2)
- Détection entrée *PV* circuit ouvert (Paramètre *MD* Bit 9)
- Détection tension batterie basse (Paramètre *DS* Bit 3)

■ **HiLevel (Niveau haut).** Ces limites d'alarme sont définies par les paramètres *HA* et *HD*. Il y a une bande d'hystérésis de 0,5 % pour chaque limite, ce qui permet une transition nette entre une situation d'alarme et une situation sans alarme. *HiLevel* est lié à *DS* Bit 0.

■ **LoLevel (Niveau bas).** Ces limites d'alarme sont définies par les paramètres *LA* et *LD*. Il y a une bande d'hystérésis de 0,5 % pour chaque limite, ce qui permet une transition nette entre une situation d'alarme et une situation sans alarme. *LoLevel* est lié à *DS* Bit 1.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). $INO = (16 \times GID) + UID$ dans la plage de 0 à 127.

S6366 : BLOC MODELE DE REGULATEUR

Fonction du bloc

Le bloc S6366 permet de faire l'interface entre les instruments 6366 et 6356. Il peut également être utilisé avec les instruments 6370/72/80/82, à condition que leur paramètres *ID* soient définis pour émuler la carte des paramètres 6356/66 sur la ligne de communication RS422.

Les bases de données importantes dans ces instruments excluent un accès complet à partir du modèle S6366. Toutefois, les paramètres importants peuvent être liés à un ensemble de constantes à usage général défini dans l'instrument et utilisé pour les communications. Quatre de ces constantes sont disponibles dans les 6356/6366 : les paramètres *1K* à *4K*. Ils ne sont pas disponibles dans les 6370/72 et 6380/82 où les pseudo-voies peuvent être utilisées à la place.




















Numéros des paramètres S6366. La table 3-11 donne les numéros des paramètres pour les communications S6366. Les rectangles grisés indiquent les paramètres soumis à la scrutation par exception.

0	1	2	3	4	5	6	7
II	DP	PH	PH	HD	LD	MN	SP
8	9	10	11	12	13	14	15
PV	OP	HA	LA	HS	LS	HO	LO
16	17	18	19	20	21	22	23
HR	LR	SL		XP	TI	TD	
24	25	26	27	28	29	30	31
				RS	RB		SW
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
GP_ST							
48	49	50	51	52	53	54	55
				1V			
56	57	58	59	60	61	62	63
		2 V					
64	65	66	67	68	69	70	71
3V							
72	73	74	75	76	77	78	79
		DI			DO	SP_SR	
80	81	82	83	84	85	86	87
					FF		
88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103
104	105	106	107	108	109	110	111
					DK	1K	2K
112	113	114	115	116	117	118	119
3K	4K						

Table 3-3 Numéros des paramètres pour S6366

Paramètres du bloc

Les symboles utilisés dans la table 3-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
II	Identité de l'instrument	ABCD hex	
DP	Position du point décimal	ABCD hex	
PH, PL	Echelle haute & basse PV & SP	Eng	
HD, LD	Limites d'alarme d'écart haute & basse	Eng	
MN	Mode de fonctionnement en cours	Menu	
FALLBACK	Mode de fonctionnement supprimé	Menu	
SP	Point de consigne résultant	Eng	 
PV	Variable procédé	Eng	
OP	Sortie actionneur	%	
HA, LA	Limites hautes & basses alarmes absolues	Eng	
HS, LS	Limites hautes & basses point de consigne	Eng	
HO, LO	Limites hautes & basses sortie	%	
HR, LR	Limites hautes & basses consigne rapport	Numérique	
SL	Point de consigne local	Eng	
SP_SR	Point de consigne déporté	Eng	
ALARM			  
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné	V/F	
HiLevel	Alarmes absolues d'écart hautes (HA, HD)	V/F	
LoLevel	Alarmes absolues d'écart basses (LA, LD)	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
XP	Bande proportionnelle	%	
TI	Temps de l'intégrale	secs./mins.	
TD	Temps de la dérivée	secs./mins.	
RS	Consigne rapport	Numérique	
RB	Décalage rapport	Eng	
SW	Positions commutateurs instrument	ABCD hex	
TS	Temps de répétition de l'algorithme à 3 termes	secs	
1V, 2V, 3V	Valeurs entrée analogique voies 1, 2, 3	Eng	
DI	Etat entrée logique	ABCD hex	
Bit0	Entrée logique Bit 0	V/F — 1	D
Bit1	Entrée logique Bit 1	V/F — 2	
Bit2	Entrée logique Bit 2	V/F — 4	
Bit3	Entrée logique Bit 3	V/F — 8	
Bit4	Entrée logique Bit 4	V/F — 1	C
Bit5	Entrée logique Bit 5	V/F — 2	
Bit6	Entrée logique Bit 6	V/F — 4	
Bit7	Entrée logique Bit 7	V/F — 8	

suite ...

... suite

Paramètre	Fonction	Unités	Etat
Bit8	Entrée logique Bit 8	V/F — 1	B
Bit9	Entrée logique Bit 9	V/F — 2	
BitA	Entrée logique Bit A	V/F — 4	
BitB	Entrée logique Bit B	V/F — 8	
BitC	Entrée logique Bit C	V/F — 1	A
BitD	Entrée logique Bit D	V/F — 2	
BitE	Entrée logique Bit E	V/F — 4	
BitF	Entrée logique Bit F	V/F — 8	
DO	Etat sortie logique	ABCD hex	
Bit0	Sortie logique Bit 0	V/F — 1	D
Bit1	Sortie logique Bit 1	V/F — 2	
Bit2	Sortie logique Bit 2	V/F — 4	
Bit3	Sortie logique Bit 3	V/F — 8	
Bit4	Sortie logique Bit 4	V/F — 1	C
Bit5	Sortie logique Bit 5	V/F — 2	
Bit6	Sortie logique Bit 6	V/F — 4	
Bit7	Sortie logique Bit 7	V/F — 8	
Bit8	Sortie logique Bit 8	V/F — 1	B
Bit9	Sortie logique Bit 9	V/F — 2	
BitA	Sortie logique Bit A	V/F — 4	
BitB	Sortie logique Bit B	V/F — 8	
BitC	Sortie logique Bit C	V/F — 1	A
BitD	Sortie logique Bit D	V/F — 2	
BitE	Sortie logique Bit E	V/F — 4	
BitF	Sortie logique Bit F	V/F — 8	
FF	Tendance	%	
DK	Positions point décimal (1K à 4K)	ABCD hex	
1K à 4K	Constantes analogiques à usage général	Numérique	
GP_ST	Diagnostic défauts instrument	ABCD hex	

Table 3-4 Paramètres du bloc S6366

Menu de spécifications du bloc

Seuls les paramètres qui ne sont pas dans le 6366 ou dont la signification est différente figurent sur la liste. Pour avoir des informations sur les paramètres normaux de l'instrument, se reporter au manuel technique ou de référence en question.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

PH, PL. Echelle haute & basse *SP, PV*. Permet de définir en unités physiques l'échelle du point de consigne résultant *SP*, ainsi que de la variable procédé *PV*.

NOTA. Dans le 6366, et tous les instruments du système 6000, les valeurs *PV* en dehors des paramètres d'échelle haute et basse ne sont pas mesurables, et sont enregistrées comme étant égales au paramètre de l'échelle la plus proche. Par ex., si *PH* est mis à 300°C pour une entrée de 10V, alors une entrée de 11V sera encore lue comme étant à 300°C.

MN. (HOLD/TRACK/MANUAL/AUTO/RATIO/CASCADE/F. MAN/F. AUTO - MAINTENIR/POURSUITE/MANUEL/AUTO/RAPPORT/CASCADE/MANUEL FORCE/AUTO FORCE). Mode de fonctionnement en cours. Les modes manuel, automatique, cascade et ratio peuvent être sélectionnés dans ce champ.

FALLBACK (REPRISE). (HOLD/TRACK/MANUAL/AUTO/RATIO/CASCADE/F. MAN/F. AUTO - MAINTENIR/POURSUITE/MANUEL/AUTO/RAPPORT/CASCADE/MANUEL FORCE/AUTO FORCE). Mode de fonctionnement supprimé calculé à partir de *DCI.ST*.

HR, LR. Limites hautes & basses de consigne rapport (ratio). *HR* et *LR* limitent absolument l'échelle pour laquelle le paramètre *RS* de consigne rapport ajusté et décalé peut être modifié. Si *HR* est défini comme étant égal à *LR*, *RS* est verrouillé à sa valeur.

SP_SR. Point de consigne déporté. Equivalent au paramètre 6366, *SP_SR*. *SP_SR* devraient être utilisés de préférence à *SL* pour relier les régulateurs dans les schémas de boucles en cascade.

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Ce type d'alarme est déclenché dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs image uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422
- Erreurs de total de contrôle de l'instrument indiquées dans le paramètre *GP.ST* bits 5, 6, 7, 8 ou 10.

■ **Hardware (Matériel).** Ce type d'alarme est déclenché dans les conditions suivantes :

- Paramètre *II* incorrect renvoyé (c'est à dire pas 356X ou 366X)
- Détection tension batterie basse (Paramètre *GP.ST* Bit 11)

■ **HiLevel (Niveau haut).** Ces limites d'alarme sont définies par les paramètres *HA* et *HD*. Il y a une bande d'hystérésis de 0,5 % pour chaque limite, ce qui permet une transition nette entre une situation d'alarme et une situation sans alarme. *HiLevel* est lié à *DP* Bits 7,5.

- **LoLevel (Niveau bas).** Ces limites d'alarme sont définies par les paramètres *LA* et *LD*. Il y a une bande d'hystérésis de 0,5 % pour chaque limite, ce qui permet une transition nette entre une situation d'alarme et une situation sans alarme. *LoLevel* est lié à *DP* Bits 6, 4.
 - **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.
- Instr_No (Numéro de l'instrument).** Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). $INO = (16 \times GID) + UID$ dans la plage de 0 à 127.
- GP_ST.** Diagnostic défauts instrument. Equivalent au paramètre *GP.ST* du 6366.

6432 AI : BLOC MODELE D'ENTREE ANALOGIQUE

Fonction du bloc

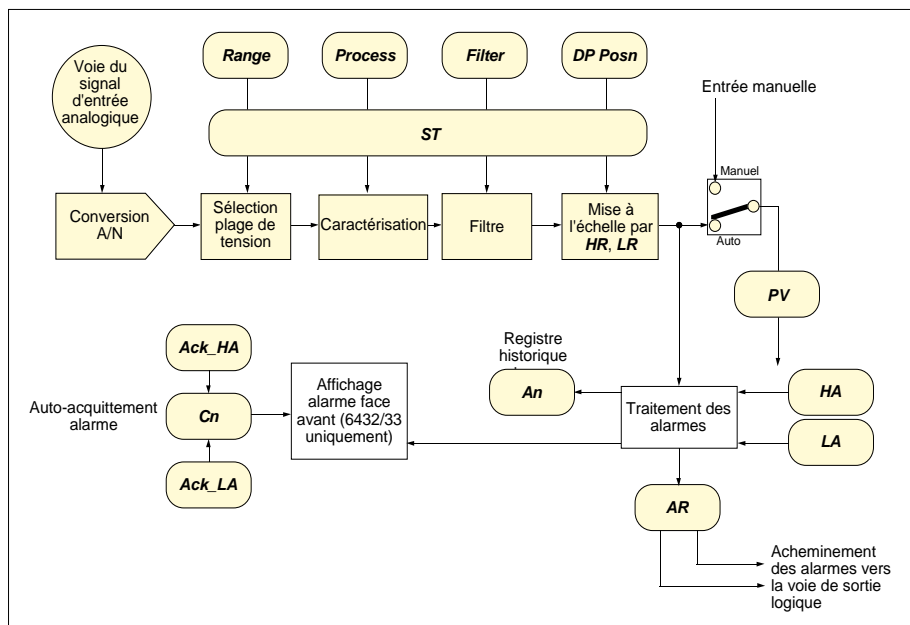


Figure 3-1 Schéma du bloc 6432 AI

Voir le schéma de la figure 3-1. Le bloc 6432 AI permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte d'entrée analogique de module de traitement et d'acquisition 6432. Ce bloc peut donc être utilisé pour communiquer avec les 6433, 6445, 6372, 6382 et T100/RS422. Il faut noter qu'un bloc séparé 6432 AI est nécessaire pour chaque voie analogique.




Numéros des paramètres 6432 AI. La table 3-5 donne les numéros des paramètres pour les communications 6432 AI. Les rectangles grisés indiquent les paramètres soumis à la scrutation par exception.

Paramètres du bloc

Les symboles utilisés dans la table 3-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

	0	1	2	3	4	5	6	7
	II	S1	A1	S2	A2	S3	A3	S4
	8	9	10	11	12	13	14	15
	A4	SW	MD	C1	C2	C3	C4	
Chn 1	16	17	18	19	20	21	22	23
	ST	HR	LR	HA	LA	PV	AR	
Chn 2	24	25	26	27	28	29	30	31
	ST	HR	LR	HA	LA	PV	AR	
Chn 3	32	33	34	35	36	37	38	39
	ST	HR	LR	HA	LA	PV	AR	
Chn 4	40	41	42	43	44	45	46	47
	ST	HR	LR	HA	LA	PV	AR	
Chn 5	48	49	50	51	52	53	54	55
	ST	HR	LR	HA	LA	PV	AR	
Chn 6	56	57	58	59	60	61	62	63
	ST	HR	LR	HA	LA	PV	AR	
Chn 7	64	65	66	67	68	69	70	71
	ST	HR	LR	HA	LA	PV	AR	
Chn 8	72	73	74	75	76	77	78	79
	ST	HR	LR	HA	LA	PV	AR	
	80	81	82	83	84	85	86	87
	TAG 1₁	TAG 1₂	TAG 1₃	TAG 1₄	TAG 2₁	TAG 2₂	TAG 2₃	TAG 2₄
	88	89	90	91	92	93	94	95
	TAG 3₁	TAG 3₂	TAG 3₃	TAG 3₄	TAG 4₁	TAG 4₂	TAG 4₃	TAG 4₄
	96	97	98	99	100	101	102	103
	TAG 5₁	TAG 5₂	TAG 5₃	TAG 5₄	TAG 6₁	TAG 6₂	TAG 6₃	TAG 6₄
	104	105	106	107	108	109	110	111
	TAG 7₁	TAG 7₂	TAG 7₃	TAG 7₄	TAG 8₁	TAG 8₂	TAG 8₃	TAG 8₄

Table 3-5 Numéros des paramètres pour 6432 AI

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
TAG	Caractères repère voie (1T, 2T)	Alphanumérique	
PV	Variable procédé	Eng	
HR, LR	Echelle haute & basse PV	Eng	
HA, LA	Limites alarmes absolues hautes & basses	Eng	
AR	Acheminement des alarmes	ABCD hex	
Range	Plage de tension du convertisseur A-N	Menu	
DP Posn	Position point décimal	Menu	
Filter	Constante de temps filtre d'entrée	Menu	
Process	Type de caractérisation	Menu	

suite...

...suite






Paramètre	Fonction	Unités	Etat
ALARM			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné/Batterie faible, etc.	V/F	
Range	Entrée voie hors échelle	V/F	
HiLevel	Alarme absolue haute (HA)	V/F	
LoLevel	Alarme absolue basse (LA)	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
Chan_No	Numéro de voie	(0-8)	
Sn	Paramètre emplacement	ABCD hex	
An	Registre historique alarmes	ABCD hex	
Cn	Auto-acquittement alarmes	ABCD Hex	
Ack_HA	Auto-acquittement HA (face avant 6432/33)	V/F	
Ack_LA	Auto-acquittement LA (face avant 6432/33)	V/F	
Id	Identité instrument	ABCD hex	
MD	Etat fonctionnement instrument (défauts)	ABCD hex	
SW	Etat positions commutateurs instrument	ABCD hex	
ST	Etat voie instrument	ABCD hex	

Table 3-6 Paramètres du bloc 6432 AI

Menu de spécifications du bloc

Seuls les paramètres qui ne sont pas dans le 6432 ou dont la signification est différente figurent sur la liste. Pour avoir des informations sur les paramètres normaux de l'instrument, se reporter au *Manuel technique du module d'acquisition 6432*.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

MODE. (AUTO/MANUEL). Mode de fonctionnement en cours.

TAG (REPERE). Permet d'afficher les informations entrées dans les paramètres *IT* et *2T* de l'instrument. Il faut noter que les paramètres Repère (TAG) ne sont pas pleinement pris en charge dans les pseudo-voies de certains instruments, et donc les informations renvoyées dans ce champ devraient être ignorées.

Range (Echelle). (0 à 10V/1 à 5V). Permet de spécifier la plage de tension du convertisseur A/N pour l'entrée de la voie. La plage est liée au paramètre *ST* Bit 0, et est redondante dans les pseudo-voies.

DP Posn. Position du point décimal. Définit la position du point décimal pour tous les paramètres mesurés en unités physiques. *DP Posn* est lié au caractère A dans le paramètre *ST*.

Filter (Filtre). (0/0,2/0,4/0,6/0,8/1,0/2,0/4,0/6,0/8,0/10,0/15, 0/20,0/25,0/30,0/60,0). Constante de temps du filtre d'entrée. Le filtre est lié au caractère C dans le paramètre *ST* et est redondant dans les pseudo-voies.

Process (Caractérisation). Permet de sélectionner la fonction de caractérisation. Process est lié au caractère B du paramètre *ST* et est redondant dans les pseudo-voies.

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Ce type d'alarme est déclenché dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs image uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422
- Erreurs de total de contrôle de l'instrument indiquées dans le paramètre *MD* bit 8 ou le paramètre *ST* bit 3.

■ **Hardware (Matériel).** Ce type d'alarme est déclenché dans les conditions suivantes :

- Paramètre *II* incorrect renvoyé
- Type de bloc incorrect
- Détection tension batterie faible (*MD* Bit 11)
- Indication de défaut matériel (*Sn* Bit 12)

■ **Range (Echelle).** Ce champ d'alarme est lié au paramètre *ST* Bit 2 qui indique une situation hors échelle à l'entrée de la voie. Cette fonction n'est pas prise en charge par certains types de configurations d'entrée, et n'est pas utilisée avec les pseudo-voies.

■ **HiLevel, LoLevel (Niveau haut, Niveau bas).** Ces limites d'alarme sont définies par les paramètres *HA* et *LA*. Il y a une bande d'hystérésis de 0,5 % pour chaque limite, ce qui permet une transition nette entre une situation d'alarme et une situation sans alarme.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). INO = (16 x GID) + UID dans la plage de 0 à 127.

Chan_No (N° de voie). Il y a huit voies analogiques groupées pour chaque numéro d'instrument. Ce champ permet de sélectionner la voie à lier au bloc.

Sn. Paramètre emplacement, équivalent aux paramètres *SI* à *S4* du 6432. Renvoie les informations sur le type d'emplacement, le nombre de voies validées et l'état du matériel.

An. Registres historiques des alarmes, équivalent aux paramètres *A1* à *A4* du 6432.

Cn. Auto-acquittement des alarmes (voir *Ack_HA* et *Ack_LA*).

Ack_HA. Permet de valider l'acquittement automatique de l'alarme absolue haute sur la

face avant des instruments 6432 et 6433. *Ack_HA* est lié au bit pertinent dans le paramètre *Cn*.

Ack_LA. Permet de valider l'acquittement automatique de l'alarme absolue basse sur la face avant des instruments 6432 et 6433. *Ack_LA* est lié au bit pertinent dans le paramètre *Cn*.

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

ST. Paramètre d'état de la voie. *ST* peut être utilisé pour configurer/contrôler différents aspects de la voie d'entrée analogique (voir les données techniques spécifiques 6432 en ce qui concerne les significations des bits de *ST*). Toutefois, pour configurer la plage de tension, la position du point décimal, la constante de temps du filtre, et la caractérisation, il est plus facile d'utiliser les paramètres *individuels* correspondants qui sont liés à *ST* (c'est à dire *Range*, *DP Posn*, *Filter* et *Process*).

6432 AO : BLOC MODELE DE SORTIE ANALOGIQUE

Fonction du bloc

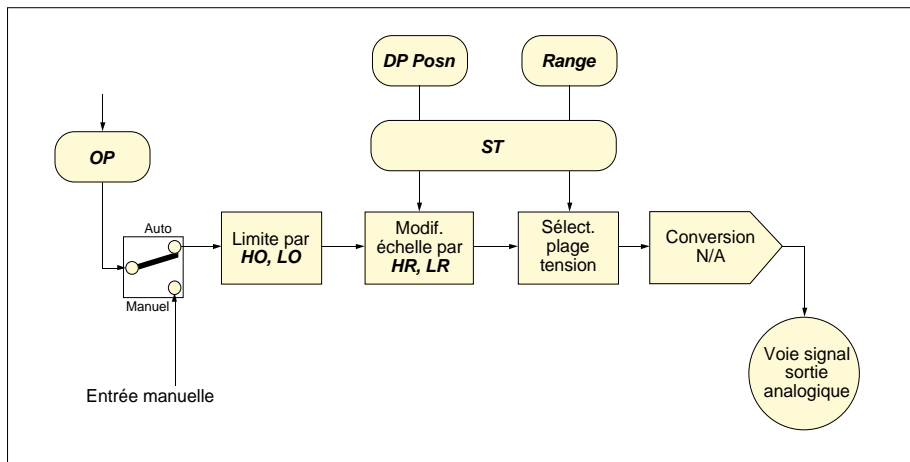


Figure 3-2 Schéma du bloc 6432 AO

Voir le schéma de la figure 3-2. Le bloc 6432 AO permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte de sortie analogique de module de traitement et d'acquisition 6432. Ce bloc peut donc être utilisé pour communiquer avec les 6433, 6445, 6372, 6382 et T100/RS422. Il faut noter qu'un bloc séparé 6432 AO est nécessaire pour chaque voie analogique.

Numéros des paramètres 6432 AO. La table 3-7 donne les numéros des paramètres pour les communications 6432 AO. Les rectangles grisés indiquent les paramètres soumis à la scrutation par exception.

Paramètres du bloc

Les symboles utilisés dans la table 3-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Seuls les paramètres qui ne sont pas dans le 6432 ou dont la signification est différente figurent sur la liste. Pour avoir des informations sur les paramètres normaux de l'instrument, se reporter au *Manuel technique du module d'acquisition 6432*.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

	0	1	2	3	4	5	6	7
	II	S1	A1	S2	A2	S3	A3	S4
	8	9	10	11	12	13	14	15
	A4	SW	MD	C1	C2	C3	C4	
Chn 1	16	17	18	19	20	21	22	23
	ST	HR	LR	OP	HO	LO		
Chn 2	24	25	26	27	28	29	30	31
	ST	HR	LR	OP	HO	LO		
Chn 3	32	33	34	35	36	37	38	39
	ST	HR	LR	OP	HO	LO		
Chn 4	40	41	42	43	44	45	46	47
	ST	HR	LR	OP	HO	LO		
Chn 5	48	49	50	51	52	53	54	55
	ST	HR	LR	OP	HO	LO		
Chn 6	56	57	58	59	60	61	62	63
	ST	HR	LR	OP	HO	LO		
Chn 7	64	65	66	67	68	69	70	71
	ST	HR	LR	OP	HO	LO		
Chn 8	72	73	74	75	76	77	78	79
	ST	HR	LR	OP	HO	LO		
	80	81	82	83	84	85	86	87
	TAG 1₁	TAG 1₂	TAG 1₃	TAG 1₄	TAG 2₁	TAG 2₂	TAG 2₃	TAG 2₄
	88	89	90	91	92	93	94	95
	TAG 3₁	TAG 3₂	TAG 3₃	TAG 3₄	TAG 4₁	TAG 4₂	TAG 4₃	TAG 4₄
	96	97	98	99	100	101	102	103
	TAG 5₁	TAG 5₂	TAG 5₃	TAG 5₄	TAG 6₁	TAG 6₂	TAG 6₃	TAG 6₄
	104	105	106	107	108	109	110	111
	TAG 7₁	TAG 7₂	TAG 7₃	TAG 7₄	TAG 8₁	TAG 8₂	TAG 8₃	TAG 8₄

Table 3-7 Numéros de paramètres pour 6432 AO

MODE. (AUTO/MANUEL). Mode de fonctionnement en cours.

TAG (REPERE). Permet d'afficher les informations entrées dans les paramètres *IT* et *2T* de l'instrument. Il faut noter que les paramètres Repère (TAG) ne sont pas pleinement pris en charge dans les pseudo-voies de certains instruments, et donc les informations renvoyées dans ce champ devraient être ignorées.

Range (Echelle). (0 à 10V/1 à 5V). Permet de spécifier la plage de tension du convertisseur N/A pour l'entrée de la voie. La plage est liée au paramètre *ST* Bit 0, et est redondante dans les pseudo-voies.

DP Posn. Position du point décimal. Définit la position du point décimal pour tous les paramètres mesurés en unités physiques. *DP Posn* est lié au caractère A dans le paramètre *ST*.

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
TAG	Caractères repère voie (1T, 2T)	Alphanumérique	
OP	Valeur de sortie	Eng	
HR, LR	Echelle haute & basse OP	Eng	
HO, LO	Limites hautes & basses OP	Eng	
Range	Plage de tension du convertisseur N-A	Menu	
DP Posn	Position point décimal	Menu	
ALARM			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné/Batterie faible, etc.	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
Chan_No	Numéro de voie	(0-8)	
Sn	Paramètre emplacement	ABCD hex	
An	Registre historique alarmes	ABCD hex	
Cn	Auto-acquittement alarmes	ABCD hex	
Ack_HA	Auto-acquittement HA (<i>non utilisé</i>)	V/F	
Ack_LA	Auto-acquittement LA (<i>non utilisé</i>)	V/F	
II	Identité instrument	ABCD hex	
MD	Etat fonctionnement instrument (défauts)	ABCD hex	
SW	Etat positions commutateurs instrument	ABCD hex	
ST	Etat voie instrument	ABCD hex	

Table 3-8 Paramètres du bloc 6432 AO

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Ce type d'alarme est déclenché dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs image uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422
- Erreurs de total de contrôle de l'instrument indiquées dans le paramètre *MD* bit 8 ou le paramètre *ST* bit 3.

■ **Hardware (Matériel).** Ce type d'alarme est déclenché dans les conditions suivantes :

- Paramètre *II* incorrect renvoyé
- Type de bloc incorrect
- Détection tension batterie faible (*MD* Bit 11)
- Indication de défaut matériel (*Sn* Bit 12)

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne TS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). INO = (16 x GID) + UID dans la plage de 0 à 127.

Chan_No (N° de voie). Il y a huit voies analogiques groupées pour chaque numéro d'instrument. Ce champ permet de sélectionner la voie à lier au bloc.

Sn. Paramètre emplacement, équivalent aux paramètres *S1* à *S4* du 6432. Renvoie les informations sur le type d'emplacement, le nombre de voies validées et l'état du matériel.

An. Registres historiques des alarmes. (*Non utilisés avec les voies de sortie analogique*).

Cn. Auto-acquittement des alarmes (voir *Ack_HA* et *Ack_LA*).

Ack_HA, Ack_LA. Ces paramètres sont disponibles pour la compatibilité avec les blocs d'entrée analogique, et ne sont pas utilisés sur les voies de sortie analogique.

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

ST. Paramètre d'état de la voie. *ST* peut être utilisé pour configurer/contrôler différents aspects de la voie de sortie analogique (voir les données techniques spécifiques 6432 en ce qui concerne les significations des bits de *ST*). Toutefois, pour configurer la plage de tension, la position du point décimal, la constante de temps du filtre, et la caractérisation, il est plus facile d'utiliser les paramètres *individuels* correspondant qui sont liés à *ST* (c'est à dire *Range* et *DP Posn*).

6432 DI : BLOC MODELE D'ENTREE LOGIQUE

Fonction du bloc

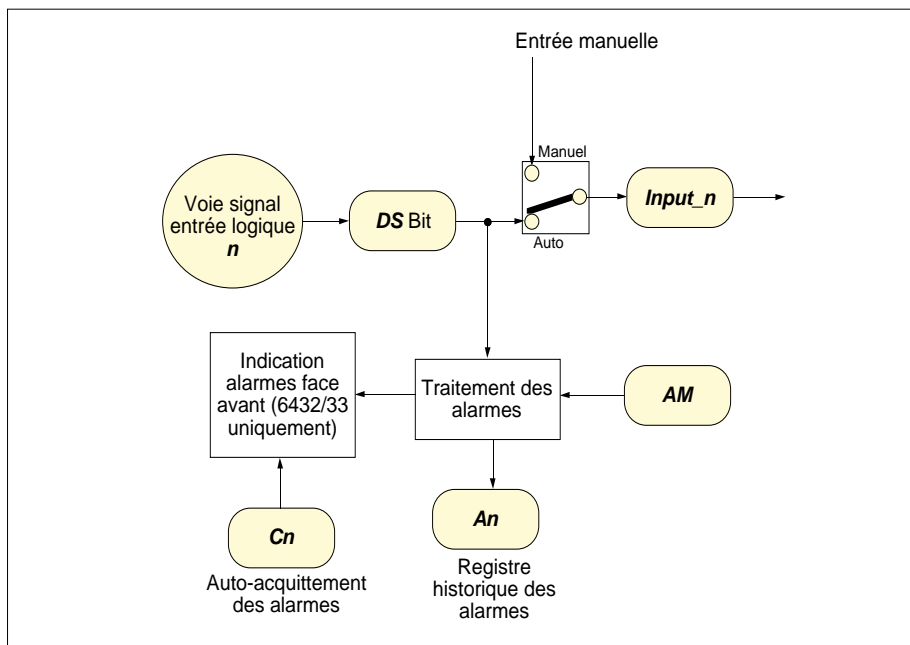


Figure 3-3 Schéma du bloc 6432 DI

Le schéma de la figure 3-3 montre seulement l'une des voies du bloc 6432 DI à 8 voies. Ce bloc permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte d'entrée logique du module de traitement et d'acquisition 6432. Ce bloc peut donc être utilisé pour communiquer avec les 6433, 6445, 6372, 6382 et T100/RS422.

Numéros des paramètres 6432 DI. La table 3-9 donne les numéros des paramètres pour les communications 6432 DI et (6432 DO). Les rectangles grisés indiquent les paramètres soumis à la scrutation par exception.

Paramètres du bloc

Les symboles utilisés dans la table 3-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

0	1	2	3	4	5	6	7
II	S1	A1	S2	A2	S3	A3	S4
8	9	10	11	12	13	14	15
A4	SW	MD	C1	C2	C3	C4	
16	17	18	19	20	21	22	23
ST	AM	DS					
24	25	26	27	28	29	30	31
80	81	82	83	84	85	86	87
TAG 1₁	TAG 1₂	TAG 1₃	TAG 1₄	TAG 2₁	TAG 2₂	TAG 2₃	TAG 2₄
88	89	90	91	92	93	94	95
TAG 3₁	TAG 3₂	TAG 3₃	TAG 3₄	TAG 4₁	TAG 4₂	TAG 4₃	TAG 4₄
96	97	98	99	100	101	102	103
TAG 5₁	TAG 5₂	TAG 5₃	TAG 5₄	TAG 6₁	TAG 6₂	TAG 6₃	TAG 6₄
104	105	106	107	108	109	110	111
TAG 7₁	TAG 7₂	TAG 7₃	TAG 7₄	TAG 8₁	TAG 8₂	TAG 8₃	TAG 8₄

Table 3-9 Numéros des paramètres pour 6432 DI & 6432 DO

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
Input_1 à Input_8	Etats des entrées logiques	V/F	
DS	Etats des entrées logiques	ABCD hex	
AM	Validation contrôle des alarmes	ABCD hex	
ALARM			
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné/Batterie faible, etc..	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
TAG1 à TAG8	Paramètres de repère des voies (1T, 2T)	Alphanumérique	
Sn	Paramètre emplacement	ABCD hex	
An	Registre historique alarmes	ABCD hex	
Cn	Auto-acquittement alarmes	ABCD hex	
II	Identité instrument	ABCD hex	
MD	Etat fonctionnement instrument (défauts)	ABCD hex	
SW	Etat position des commutateurs instrument	ABCD hex	
ST	Etat voie instrument	ABCD hex	

Table 3-10 Paramètres du bloc 6432 DI

Menu de spécifications du bloc

Seuls les paramètres qui ne sont pas dans le 6432 ou dont la signification est différente figurent ur la liste. Pour avoir des informations sur les paramètres normaux de l'instrument, se reporter au *Manuel technique du module d'acquisition 6432*.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

MODE. (AUTO/MANUEL). Mode de fonctionnement en cours.

Input_1 à Input_8. Indique l'état de l'entrée respective. Ces paramètres sont calculés à partir des caractères A et B du paramètre *DS*.

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Ce type d'alarme est déclenché dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs image uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422
- Erreurs de contrôle des données de l'instrument indiquées dans le paramètre *MD* bit 8 ou le paramètre *ST* bit 3.

■ **Hardware (Matériel).** Ce type d'alarme est déclenché dans les conditions suivantes :

- Paramètre *II* incorrect renvoyé
- Type de bloc incorrect
- Détection tension batterie faible (*MD* Bit 11)
- Indication de défaut matériel (*Sn* Bit 12)

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). $INO = (16 \times GID) + UID$ dans la plage de 0 à 127.

TAG1 à TAG8. Permet d'afficher le contenu les paramètres *IT* et *2T* de l'instrument. Il faut noter que les paramètres Repère (TAG) ne sont pas pleinement pris en charge dans les pseudo-voies de certains instruments, et donc les informations renvoyées dans ce champ devraient être ignorées.

Sn. Paramètre emplacement, équivalent aux paramètres *S1* à *S4* du 6432. Renvoie les informations sur le type d'emplacement, le nombre de voies validées et l'état du matériel.

An. Registres historiques des alarmes, équivalent aux paramètres *A1* à *A4* du 6432.

Cn. Auto-acquittement des alarmes. Permet de valider l'acquittement automatique des alarmes de transition hautes et basses sur la face avant des instruments 6432 et 6433.

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

6432 DO : BLOC MODELE DE SORTIE LOGIQUE

Fonction du bloc

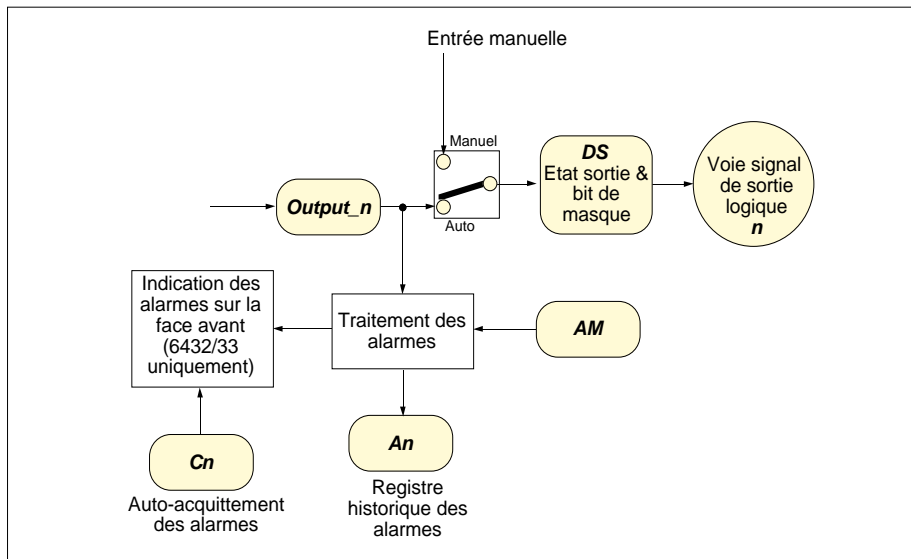


Figure 3-4 Schéma du bloc 6432 DO

Le schéma de la figure 3-4 montre seulement l'une des voies du bloc 6432 DO à 8 voies. Ce bloc permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte de sortie logique du module de traitement et d'acquisition 6432. Ce bloc peut donc être utilisé pour communiquer avec les 6433, 6445, 6372, 6382 et T100/RS422.

Numéros des paramètres 6432 DO. Les numéros des paramètres pour les communications 6432 DO sont les mêmes que pour 6432 DI. Voir la table 3-9 à la page 3-21.

Paramètres du bloc

Les symboles utilisés dans la table 3-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.









Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement en cours	Menu	
Output_1 à _8	Etats des sorties logiques	V/F	
DS	Etats des sorties logiques & bits de masque	ABCD hex	
AM	Validation contrôle des alarmes	ABCD hex	
ALARM			  
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Module erroné/Batterie faible, etc..	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro de l'instrument RS422 (INO)	(0-127)	
TAG1 à TAG8	Paramètres de repère des voies (1T, 2T)	Alphanumérique	
Sn	Paramètre emplacement	ABCD hex	
An	Registre historique alarmes	ABCD hex	
Cn	Auto-acquittement alarmes	ABCD hex	
II	Identité instrument	ABCD hex	
MD	Etat fonctionnement instrument (défauts)	ABCD hex	
SW	Etat position des commutateurs instrument	ABCD hex	
ST	Etat voie instrument	ABCD hex	

Table 3-11 Paramètres du bloc 6432 DO

Menu de spécifications du bloc

La plupart des paramètres 6432 DO correspondent à ceux du bloc 6432 DI décrits dans la section précédente - voir les détails dans cette section. En ce qui concerne les significations des paramètres normaux du 6432, voir le *Manuel technique du module d'acquisition 6432*.

GEN_COMM : BLOC DE COMMUNICATION GÉNÉRIQUE S6000

Fonction du bloc

Le bloc GEN_COMM est un modèle de communication générique pour les communications RS422 du système 6000 et les communications binaires BI-SYNCH d'Eurotherm Systèmes SA.

Chaque bloc GEN_COMM permet de sélectionner un maximum de 18 paramètres - définis par leurs numéros de paramètres - à contrôler et/ou à mettre à jour dans un instrument du système 6000 donné communiquant sur le bus RS422. Les blocs peuvent être reliés à des synoptiques standard de face avant Eurotherm Systèmes, ce qui simplifie le dialogue avec les régulateurs déportés.

Les paramètres sont composés d'une chaîne de 8 caractères (repère), douze valeurs analogiques à usage général, quatre valeurs logiques à champ binaire de 16 bits et un entier de type totalisation à 8 chiffres. À l'exception du repère et du total à 8 chiffres, toutes les valeurs spécifiées des paramètres rapportées par le bloc sont disponibles comme liaisons d'entrée et de sortie dans le schéma de boucles.










Les numéros de paramètre pour un instrument particulier peuvent être saisis individuellement dans le modèle à usage général - obligatoire pour les instruments autres que ceux d'Eurotherm Systèmes. Mais, pour les instruments standard d'Eurotherm Systèmes, il suffit de sélectionner l'un des dix-neuf types indiqués dans le champ *Type*, ce qui permet de configurer automatiquement l'ensemble correct de PNO, avec leurs codes mnémoniques correspondants, et les positionne également (dans le cas de régulateurs) dans les champs de modèle corrects pour pouvoir les relier à un synoptique de face avant de régulation.

La saisie d'une valeur de champ dans le bloc GEN_COMM permet de transmettre la valeur à la base de données correspondante de l'instrument. Si elle est acceptée, la valeur est renvoyée au bloc comme étant la nouvelle valeur active. Le bloc affiche toujours les valeurs de paramètre actives de l'instrument (certaines peuvent n'être qu'en lecture uniquement).

Un maximum de huit blocs GEN_COMM peuvent être configurés sur l'une des adresses d'instrument (*Instr_No*). Mais, les blocs GEN_COMM ne peuvent être mélangés à d'autres blocs de la catégorie S6000 sur une adresse d'instrument unique.

Paramètres du bloc

Les symboles utilisés dans la table 3-12 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Type	Instrument du type système 6000	Menu	
II	Identité instrument	ABCD hex	
PNO_0 à PNO_17	Numéros de paramètre (PNO) (-1 = invalide)	(-1 à 127)	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0 à 127)	
VALUE_0	Caractères de repère (8 octets) pour PNO_0	Alphanumérique	
VALUE_n (n=1-12)	Valeurs analogiques pour PNO_n	Eng	
VALUE_n (n=13-16)	Valeurs champ binaire pour PNO_n	ABCD hex	
VALUE_17	Valeur de totalisation pour PNO_17 (8 chiffres)	Entier	

*En conduite

Table 3-12 Paramètres du bloc GEN_COMM

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Type. (General/6350/6351/6352/6353/6355/6356/6358/6360/6363/6366/6370/6372/6380/6382/6434/6435/6436/6437). Permet de sélectionner le type de modèle d'instrument préconfiguré nécessaire pour la communication avec l'instrument déporté. *General* (général) met tous les champs *PNO_n* à -1 (invalidation) et remet à zéro la colonne des unités, qui est ainsi prête à être configurée pour un instrument non-standard. La sélection de l'un des types de régulateur Eurotherm Systèmes permet de préconfigurer un groupe de PNO importants avec les codes mnémoniques appropriés dans un ordre particulier sur le modèle adapté à la liaison à un synoptique de face avant ('CONTROL').

Cet ordre ne devrait pas être modifié pour les régulateurs standard, dans la mesure où la face avant reprend et affiche toujours les paramètres nécessaires à partir des mêmes champs *VALUE_n* du bloc; par ex. *PV* est toujours lu à partir de *VALUE_1*, *SP* à partir de *VALUE_2*, *OP* à partir de *VALUE_3*, et ainsi de suite.

Si l'ordre est modifié, l'affichage sur la face avant de conduite sera incohérent, mais cette fonction est utile pour les régulateurs autres que ceux d'Eurotherm Systèmes. Les PNO permettent de forcer les valeurs pertinentes dans les champs *VALUE_n* appropriés pour que l'affichage de la face avant soit cohérent. Vous pouvez également écrire ou relier vos propres valeurs dans les champs *VALUE_n*, si le régulateur non-Eurotherm Systèmes ne les produit pas automatiquement. Par exemple, si les paramètres *HR/LR* ne sont pas fournis, vous devez les écrire respectivement dans *VALUE_4* et *VALUE_5*, pour que le synoptique de la face avant puisse mettre correctement à l'échelle ses sorties.

NOTA. Vous pouvez relier des types d'instruments non-régulateurs (par ex. 6436) à des faces avant de régulateurs, mais l'affichage en conduite ne sera pas cohérent.

Il vaut mieux les relier point à point à d'autres objets graphiques, par ex.

READOUT, TREND ou BAR (AFFICHAGE, TENDANCE ou BARGRAPHE).

II. Identité de l'instrument. Renvoie le type de base de données de l'instrument qui communique au niveau de ce numéro d'instrument (*Instr_No*). Noter que certains instruments peuvent émuler les structures de base de données d'autres types d'instrument sur le RS422 en protocole binaire. *II* peut donc ne pas correspondre au type d'instrument réel qui communique à une adresse particulière.

PNO_0 à PNO_17. Numéros de paramètre des paramètres qui doivent être contrôlés/mis à jour dans la base de données de l'instrument défini par *Instr_No*. Les valeurs contrôlées apparaissent respectivement dans *VALUE_0* à *VALUE_17*. Les valeurs utilisées des PNO vont de 0 à 127; une valeur de -1 indique que le PNO ne doit pas être contrôlé. Les champs PNO ne peuvent être configurés que lorsqu'ils sont hors ligne; en conduite, ils passent en lecture uniquement. Si l'option *General* (général) est sélectionnée dans le champ *Type*, toutes les valeurs *PNO_n* sont mises à -1 par défaut (invalidation). La sélection de l'un des types d'instrument standard d'Eurotherm Systèmes permet de préconfigurer automatiquement les valeurs *PNO_n*, ainsi que leurs champs d'unités ('units').

NOTA. Il est utile d'avoir un code mnémonique dans le champ d'unités de chaque PNO pour identifier le paramètre contrôlé. Si vous utilisez des codes mnémoniques non-standard ou modifiez des codes préconfigurés, veillez à ce que les champs d'unités des blocs image et des blocs locaux correspondants coïncident - ces valeurs se sont pas transmises automatiquement sur le LIN.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No (N° de l'instrument). Définit le *numéro de l'instrument* (INO) - c'est à dire l'adresse RS422 - de l'instrument que le bloc GEN_COMM doit contrôler/mettre à jour au cours de la conduite. $INO = 16 (16 \times GID) + UID$ dans la plage de 0 à 127. Des commutateurs ou un paramètre de l'instrument définissent ses identificateurs de groupe et d'unité, GID et UID.

VALUE_0 (VALEUR_0). Champ de type chaîne de caractères pour repères à 8 octets (8 caractères). Le *PNO_0* en question représente l'adresse de départ de quatre PNO consécutifs dans lesquels le repère est sauvegardé. *VALUE_0* affiche les caractères alphanumériques ayant des valeurs ASCII à ces adresses. Noter que *VALUE_0* n'est pas disponible comme liaison d'entrée/sortie dans le schéma de boucles.

VALEUR_1 à VALEUR_12. Champs de valeurs réelles contrôlant/mettant à jour les valeurs analogiques des paramètres au niveau du PNO spécifié respectivement par *PNO_1* à *PNO_12*.

NOTA. Normalement, ces champs affichent des valeurs physiques de 0 à 9999 correspondant à un PNO unique. Le paramètre *PI* constitue une exception dans les “pseudo-instruments” 0832 (résident dans les versions spéciales des instruments 6437, 6433 et 6445). Dans ce cas, deux PNO consécutifs sont lus automatiquement - pour la paire *PI/P2* - et indiqués comme un nombre à point flottant à 32 bits dans la plage de $\pm 3.4E\pm 38$.

VALEUR_13 à VALEUR_16. Champs binaires à 16 bits contrôlant/mettant à jour les valeurs logiques des paramètres au niveau du PNO spécifié respectivement par *PNO_13* à *PNO_16*. Des liaisons peuvent être effectuées entre le schéma de boucles et l'un des bits dans ces champs binaires. (Si *VALUE_13* à *VALUE_16* ne sont pas reliés correctement à des paramètres analogiques, ils affichent toujours 0000 hex).

VALUE_17 (VALEUR_17). Champ de type totalisation contrôlant/mettant à jour la valeur au niveau de la paire PNO spécifiée par *PNO_17*, plus le PNO consécutif suivant. *VALUE_17* est affiché comme un total entier à 8 chiffres (32 bits). Noter que *VALUE_17* ne permet pas de liaison d'entrée/sortie au schéma de boucles, mais peut être relié à un synoptique de conduite ou à un bloc HIST.

BLOCS SERIE SL6000 ESCLAVES

Ces types de bloc fournissent une interface entre la liaison de communication binaire et une base de données T600. Ils se décomposent en trois groupes:

- **Instruments mono-boucle.** Ce groupe comprend deux blocs - les types de bloc SL6366 et SL6437. Chaque bloc représente un instrument discret S6000 sur un INO unique (numéro d'instrument des communications binaires).
- **Instruments mono-voie, multi-boucles.** Ce groupe comprend les types de bloc SL6432DI et SL6432DO. Bien qu'il ne soit possible de n'associer qu'un seul bloc de base de données à chaque INO, le bloc peut être considéré comme faisant partie d'un 6432 qui peut être rattaché à d'autres INO, et par conséquent, être associé à d'autres blocs de base de données.
- **Instruments multi-voies, multi-boucles.** Ce groupe comprend les types de bloc SL6432AI et SL6432AO. Chaque INO peut être associé à un maximum de huit blocs du même type, représentant les voies (8 maxi.). Le SL0832 figure également dans ce groupe, mais est particulier, dans la mesure où l'ensemble des huit voies se trouve dans un seul bloc.

NOTA. *Affectation de l'adresse de communication du T640 sur le RS422 - blocs SL6432AI, SL6432AO, SL6432DI et SL6432DO.* Les adresses pour les 6432 sont affectées en groupes de quatre, l'adresse de départ étant un multiple de quatre. Le T640 émule le 6432 sur les communications RS422 avec les blocs ci-dessus. Les mêmes restrictions d'affectation d'adresse sont applicables, lorsqu'un T640 est utilisé avec un système Maxi-Vis. Donc, si un T640 dispose d'un bloc d'émulation 6432 configuré pour fournir des informations à un Maxi-Vis, l'ensemble du groupe de quatre adresses devrait être affecté (mais pas nécessairement configuré) à ce T640. Noter que la restriction ne s'applique pas lorsque le système de supervision est un T2001 ou un T1000.

SL6432AI : BLOC MODELE D'ENTREE ANALOGIQUE

Fonction du bloc

Le bloc SL6432AI fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments multi-voies, multi-boucles du système 6000 par l'intermédiaire des communications binaires. Un maximum de huit blocs SL6432AI, représentant chacun une voie d'entrée unique peuvent être associés à un numéro d'instrument unique (INO).

Numéros des paramètres SL6432AI. La table 3-13 donne les numéros des paramètres pour les communications SL6432AI. "0" dans une case indique une valeur fictive, lue comme zéro et écrite comme NAK (accusé de réception négatif). Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception.

	0	1	2	3	4	5	6	7
	II	Sn	0	Sn	0	Sn	0	Sn
	8	9	10	11	12	13	14	15
	0	SW	0	0	0	0	0	ID
Chn 1	16	17	18	19	20	21	22	23
	ST	HR	LR	HA	LA	PV	0	
Chn 2	24	25	26	27	28	29	30	31
	ST	HR	LR	HA	LA	PV	0	
Chn 3	32	33	34	35	36	37	38	39
	ST	HR	LR	HA	LA	PV	0	
Chn 4	40	41	42	43	44	45	46	47
	ST	HR	LR	HA	LA	PV	0	
Chn 5	48	49	50	51	52	53	54	55
	ST	HR	LR	HA	LA	PV	0	
Chn 6	56	57	58	59	60	61	62	63
	ST	HR	LR	HA	LA	PV	0	
Chn 7	64	65	66	67	68	69	70	71
	ST	HR	LR	HA	LA	PV	0	
Chn 8	72	73	74	75	76	77	78	79
	ST	HR	LR	HA	LA	PV	0	
	80	81	82	83	84	85	86	87
	88	89	90	91	92	93	94	95
	96	97	98	99	100	101	102	103
	104	105	106	107	108	109	110	111

Table 3-13 Numéros des paramètres pour SL6432AI

Paramètres du bloc

Les symboles utilisés dans la table 3-14 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications de bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé	Eng	
HR, LR	Echelle PV haute & basse	Eng	
HA, LA	Paramètres fictifs (utilisés comme sauvegardes données)	Eng	
Range	Position du point décimal	Menu	
DP_Posn	Position du point décimal	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
Chan_No	Numéro de voie dans l'INO spécifié	(0-8)	
II	Identité de l'instrument (432A)	ABCD hex	
Sn	Paramètre emplacement	ABCD hex	
SW	Etat positions commutateurs de l'instrument T600	ABCD hex	
ID	Modèle & version de l'instrument T600	ABCD hex	
ST	Indique la valeur DP_Posn (chiffre A)	A(BCD) hex	

Table 3-14 Paramètres du bloc SL6432AI

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV (Variable Procédé). Un nombre réel dont la valeur doit être calculée par l'intermédiaire d'une liaison dans ce champ. *PV* n'est pas mis à jour si la valeur qui est écrite est en dehors de l'échelle définie par *HR* et *LR*. Le champ est indiqué comme **WRITE_PROT** (PROTEGE EN ECRITURE), ce qui fait que toute tentative d'écrire dans ce champ à partir des communications binaires fait l'objet d'un accusé de réception négatif (NAK).

HR, LR. Nombres réels qui définissent l'échelle admissible de *PV*. *HR* doit être supérieur à *LR*.

HA, LA. Nombres réels qui sont rattachés aux PNO affectés aux paramètres *HA* et *LA*. Ils n'ont pas de fonctions autre que celle de servir de sauvegardes de données en lecture/écriture.

Range & DP_Posn (Echelle & Position décimale). Ces champs définissent la position du point décimal (intégré dans *ST* ou transmis par l'intermédiaire des communications binaires) sous forme de chaîne de texte ou sous forme numérique. Noter que, en interne, les deux champs accèdent à la même zone de données du bloc.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No & Chan_No (N° d'instrument & N° de voie). Ces champs définissent l'INO (*Instr_No*) et la voie dans le cadre de cet INO (*Chan_No*) occupé par ce bloc de base de données. Noter que si deux ou plusieurs blocs sont affectés au même INO, ils doivent tous être du même type, mais avoir des valeurs uniques de *Chan_No*.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >432A.

Sn. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >0n00, où n est égal à la voie au chiffre le plus élevé au niveau de cet INO. Le numéro de paramètre (PNO) occupé par ce paramètre dépend de INO:

$$PNO = [(INO.AND.3) * 2] + 1$$

Les paramètres *Sn* des autres INO sont des paramètres fictifs, c'est à dire qu'ils sont lus comme zéro et produisent un NAK en cas de tentative d'écriture.

SW. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale qui représente les positions des commutateurs du T600.

ID. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >640n (indiquant T640, version n).

ST. Il s'agit d'une valeur hexadécimale >d000, où d est identique à *DP_Posn*. Noter que d peut être mis à jour par les communications binaires.

SL6432AO : BLOC MODELE DE SORTIE ANALOGIQUE

Fonction du bloc

Le bloc SL6432AO fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments multi-voies, multi-boucles du système 6000 par l'intermédiaire des communications binaires. Un maximum de huit blocs SL6432AO, représentant chacun une voie de sortie unique peuvent être associés à un numéro d'instrument unique (INO).

Numéros des paramètres SL6432AO. La table 3-15 donne les numéros des paramètres pour les communications SL6432AO. "0" dans une case indique une valeur fictive, lue comme zéro et écrite comme NAK (accusé de réception négatif). Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception.

	0	1	2	3	4	5	6	7
	II	Sn	0	Sn	0	Sn	0	Sn
	8	9	10	11	12	13	14	15
	0	SW	0	0	0	0	0	ID
Chn 1	16	17	18	19	20	21	22	23
	ST	HR	LR	OP	HO	LO		
Chn 2	24	25	26	27	28	29	30	31
	ST	HR	LR	OP	HO	LO		
Chn 3	32	33	34	35	36	37	38	39
	ST	HR	LR	OP	HO	LO		
Chn 4	40	41	42	43	44	45	46	47
	ST	HR	LR	OP	HO	LO		
Chn 5	48	49	50	51	52	53	54	55
	ST	HR	LR	OP	HO	LO		
Chn 6	56	57	58	59	60	61	62	63
	ST	HR	LR	OP	HO	LO		
Chn 7	64	65	66	67	68	69	70	71
	ST	HR	LR	OP	HO	LO		
Chn 8	72	73	74	75	76	77	78	79
	ST	HR	LR	OP	HO	LO		
	80	81	82	83	84	85	86	87
	88	89	90	91	92	93	94	95
	96	97	98	99	100	101	102	103
	104	105	106	107	108	109	110	111

Table 3-15 Numéros des paramètres pour SL6432AO

Paramètres du bloc

Les symboles utilisés dans la table 3-16 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications de bloc*.

Paramètre	Fonction	Unités	Etat
OP	Sortie du bloc	Eng	
Track	OP suit Track si SelTrack = VRAI	Eng	
HR, LR	Echelle OP haute & basse	Eng	
HO, LO	Seuils OP hauts & bas	Eng	
SelTrack	Sélection du mode poursuite	Eng	
Range	Position du point décimal	Menu	
DP_Posn	Position du point décimal	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
Chan_No	Numéro de voie dans l'INO spécifié	(0-8)	
II	Identité de l'instrument (432A)	ABCD hex	
Sn	Paramètre emplacement	ABCD hex	
SW	Etat des positions commutateurs de l'instrument T600	ABCD hex	
ID	Modèle & version de l'instrument T600	ABCD hex	
ST	Indique la valeur DP_Posn (chiffre A)	A(BCD) hex	

Table 3-16 Paramètres du bloc SL6432AO

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

OP (Sortie). Un nombre réel dont la valeur doit être écrite par l'intermédiaire d'une liaison dans ce champ. *OP* n'est pas mis à jour si la valeur qui est écrite est en dehors de l'échelle définie par *HR* et *LR*. Si la valeur écrite se situe dans le cadre de l'échelle, mais est en dehors des limites définies par *HO* et *LO*, la valeur de *OP* sera limitée soit à *HO* ou à *LO*.

Track (Poursuite). *OP* suit la valeur dans *Track* si *SelTrack* est VRAI (TRUE).

HR, LR. Nombres réels qui définissent l'échelle admissible de *OP*. *HR* doit être supérieur à *LR*.

HO, LO. Nombres réels qui définissent les limites de *OP*. *HO* doit être supérieur à *LO*.

SelTrack. *SelTrack* VRAI sélectionne le mode poursuite (voir *Track*).

Range & DP_Posn (Echelle & Position décimale). Ces champs définissent la position du point décimal (intégré dans *ST* ou transmis par l'intermédiaire des communications binaires) sous forme de chaîne de texte ou sous forme numérique. Noter que, en interne, les deux champs accèdent à la même zone de données du bloc.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No & Chan_No (N° d'instrument & N° de voie). Ces champs définissent l'INO (*Instr_No*) et la voie dans le cadre de cet INO (*Chan_No*) occupé par ce bloc de base de données. Noter que si deux ou plusieurs blocs sont affectés au même INO, ils doivent tous être du même type, mais avoir des valeurs uniques de *Chan_No*.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >432A.

Sn. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >0n08, où *n* est égal à la voie au chiffre le plus élevé au niveau de cet INO, et "08" dans les deux chiffres de poids faible indique qu'il s'agit d'une voie A0. Le numéro de paramètre (PNO) occupé par ce paramètre dépend de INO:

$$PNO = [(INO.AND.3) * 2] + 1$$

Les paramètres Sn des autres INO sont des paramètres fictifs, c'est à dire qu'ils sont lus comme zéro et produisent un NAK en cas de tentative d'écriture.

SW. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale qui représente les positions des commutateurs du T600.

ID. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >640n (indiquant T640, version *n*).

ST. Il s'agit d'une valeur hexadécimale >d000, où *d* est identique à *DP_Posn*. Noter que *d* peut être mis à jour par les communications binaires.

SL6432DI : BLOC MODELE D'ENTREE LOGIQUE

Fonction du bloc

Le bloc SL6432DI fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments mono-voie, multi-boucles du système 6000 par l'intermédiaire des communications binaires. Bien qu'il ne soit possible de n'associer qu'un seul bloc SL6432DI à un numéro d'instrument unique (INO), le bloc peut être considéré comme faisant partie d'un instrument 6432 qui peut être rattaché à d'autres INO, et par conséquent, être associé à d'autres blocs de base de données.

Numéros des paramètres SL6432DI. La table 3-17 donne les numéros des paramètres pour les communications SL6432DI (et SL6432DO). "0" dans une case indique une valeur fictive, lue comme zéro et écrite comme NAK (accusé de réception négatif). Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception.

0	1	2	3	4	5	6	7
II	Sn	0	Sn	0	Sn	0	Sn
8	9	10	11	12	13	14	15
0	SW	0	0	0	0	0	ID
16	17	18	19	20	21	22	23
ST	AM	DS					
24	25	26	27	28	29	30	31

Table 3-17 Numéros des paramètres des blocs SL6432DI & SL6432DO

Paramètres du bloc

Les symboles utilisés dans la table 3-18 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications de bloc*.

Paramètre	Fonction	Unités	Etat
DS	Etat de l'entrée logique	AB(CD) hex	☐→
Bit8	Etat voie 8	V/F — 1	B
Bit9	Etat voie 7	V/F — 2	
Bit10	Etat voie 6	V/F — 4	
Bit11	Etat voie 5	V/F — 8	
Bit12	Etat voie 4	V/F — 1	A
Bit13	Etat voie 3	V/F — 2	
Bit14	Etat voie 2	V/F — 4	
Bit15	Etat voie 1	V/F — 8	

suite...

...suite










Paramètre	Fonction	Unités	Etat
AM	Champ binaire fictif (sauvegarde données lect./écrit.)	ABCD hex	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
II	Identité de l'instrument (432A)	ABCD hex	
Sn	Paramètre emplacement	ABCD hex	
SW	Etat des positions commutateurs de l'instrument T600	ABCD hex	
ID	Modèle & version de l'instrument T600	ABCD hex	
ST	Paramètre d'état fictif (>0000)	ABCD hex	

Table 3-18 Paramètres du bloc S6432DI

Menu de spécifications du bloc.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

DS. Un nombre hexadécimal de 16 bits dont la valeur doit être calculée par l'intermédiaire de liaisons dans les champs binaires. Le champ est indiqué comme PROTEGE EN ECRITURE (WRITE_PROT), ce qui produit un accusé de réception négatif (NAK) en cas de tentative d'écriture à partir des communications binaires. Seuls les huit bits de poids fort (chiffres A et B) sont utilisés. Les *bit8* à *bit15* sont respectivement libellés “Chan8” à “Chan1” pour donner une correspondance directe par rapport à la convention de désignation des bits dans un instrument 6432 réel.

AM. Un nombre hexadécimal à 16 bits rattaché au PNO affecté au paramètre AM. Il n'a pas de fonctions autre que celle de servir de sauvegarde de données en lecture/écriture.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Notez qu'il ne peut y avoir qu'un bloc SL6432DI pour chaque INO.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >432A.

Sn. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >0n10, où *n* est égal à la voie au chiffre le plus élevé au niveau de cet INO, et “10” dans les deux chiffres de poids faible indique qu'il s'agit d'une voie DI. *n* prend toujours la valeur “8”, dans la mesure où ce bloc fournit

toutes les huit voies pour l'INO. Le numéro de paramètre (PNO) occupé par ce paramètre dépend de INO:

$$\text{PNO} = [(\text{INO}.\text{AND}.3) * 2] + 1$$

SW. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale qui représente les positions des commutateurs du T600.

ID. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale $>640n$ (indiquant T640, version n).

ST. Prend toujours la valeur hexadécimale >0000 .

SL6432DO : BLOC MODELE DE SORTIE LOGIQUE






Fonction du bloc

Le bloc SL6432DO fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments mono-voie, multi-boucles du système 6000 par l'intermédiaire des communications binaires. Bien qu'il ne soit possible de n'associer qu'un seul bloc SL6432DO à un numéro d'instrument unique (INO), le bloc peut être considéré comme faisant partie d'un instrument 6432 qui peut être rattaché à d'autres INO, et par conséquent, être associé à d'autres blocs de base de données.

Numéros des paramètres SL6432DO. La table 3-17 à la page 3-37 donne les numéros des paramètres pour les communications SL6432DO (qui sont les mêmes que pour SL6432DI). "0" dans une case indique une valeur fictive, lue comme zéro et écrite comme NAK (accusé de réception négatif). Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception.

Paramètres du bloc

Les symboles utilisés dans la table 3-19 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
DS	Etat de la sortie logique & des masques des bits	ABCD hex	
Bit0	Etat masque8 (masque voie8)	V/F — 1	D 
Bit1	Etat masque7 (masque voie7)	V/F — 2	
Bit2	Etat masque6 (masque voie6)	V/F — 4	
Bit3	Etat masque5 (masque voie5)	V/F — 8	
Bit4	Etat masque4 (masque voie4)	V/F — 1	C 
Bit5	Etat masque3 (masque voie3)	V/F — 2	
Bit6	Etat masque2 (masque voie2)	V/F — 4	
Bit7	Etat masque1 (masque voie1)	V/F — 8	
Bit8	Etat sortie voie8	V/F — 1	B 
Bit9	Etat sortie voie7	V/F — 2	
Bit10	Etat sortie voie6	V/F — 4	
Bit11	Etat sortie voie5	V/F — 8	
Bit12	Etat sortie voie4	V/F — 1	A 
Bit13	Etat sortie voie3	V/F — 2	
Bit14	Etat sortie voie2	V/F — 4	
Bit15	Etat sortie voie1	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
Track	Etats poursuite sortie logique	CD hex	
Bit0	Etat poursuite voie8 (Seltrack.Bit0 VRAI)	V/F — 1	D
Bit1	Etat poursuite voie7 (Seltrack.Bit1 VRAI)	V/F — 2	
Bit2	Etat poursuite voie6 (Seltrack.Bit2 VRAI)	V/F — 4	
Bit3	Etat poursuite voie5 (Seltrack.Bit3 VRAI)	V/F — 8	
Bit4	Etat poursuite voie4 (Seltrack.Bit4 VRAI)	V/F — 1	C
Bit5	Etat poursuite voie3 (Seltrack.Bit5 VRAI)	V/F — 2	
Bit6	Etat poursuite voie2 (Seltrack.Bit6 VRAI)	V/F — 4	
Bit7	Etat poursuite voie1 (Seltrack.Bit7 VRAI)	V/F — 8	
AM	Champ binaire fictif (sauvegarde données lect/écrit)	ABCD hex	
SelTrack	Sélection mode de poursuite de la sortie logique	CD hex	
Bit0	Sélection mode poursuite voie8	V/F — 1	D
Bit1	Sélection mode poursuite voie7	V/F — 2	
Bit2	Sélection mode poursuite voie6	V/F — 4	
Bit3	Sélection mode poursuite voie5	V/F — 8	
Bit4	Sélection mode poursuite voie4	V/F — 1	C
Bit5	Sélection mode poursuite voie3	V/F — 2	
Bit6	Sélection mode poursuite voie2	V/F — 4	
Bit7	Sélection mode poursuite voie1	V/F — 8	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
II	Identité de l'instrument (432A)	ABCD hex	
Sn	Paramètre emplacement	ABCD hex	
SW	Etat positions des commutateurs de l'instrument T600	ABCD hex	
ID	Modèle & version de l'instrument T600	ABCD hex	
ST	Paramètre d'état fictif (>0000)	ABCD hex	

Table 3-19 Paramètres du bloc SL6432DO

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

DS. Un nombre hexadécimal de 16 bits dont la valeur doit être écrite par l'intermédiaire des communications binaires. Les huit bits de poids fort (chiffres A et B) représentent les bits de sortie logique destinés à être reliés comme sorties au schéma de boucles. Les huit bits de poids faible (chiffres C et D) sont des masques pour les huit bits supérieurs correspondants. Si un bit de masque est VRAI, le bit de sortie logique correspondant ne peut changer d'état au moment de l'écriture. Noter que les bits de masque sont en écriture uniquement, c'est à dire qu'ils se remettent immédiatement à zéro après utilisation et sont donc toujours relus comme FAUX.

Track (Poursuite). Les bits de sortie A et B dans DS (*Bit8 à Bit5*) suivent les bits correspondants dans *Track* (*Bit0 à Bit7*), lorsque les bits correspondants dans *SelTrack* (*Bit0 à Bit7*) sont VRAIS.

AM. Un nombre hexadécimal à 16 bits occupant le PNO affecté au paramètre *AM*. Il n'a pas de fonction autre que celle de servir de sauvegarde de données en lecture/écriture.

SelTrack. Les huit bits *SelTrack* permettent de sélectionner le mode poursuite pour les bits de sortie *DS* correspondants - *Bit8 à Bit15*. Voir *Track* ci-dessus.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Noter qu'il ne peut y avoir qu'un bloc SL6432DO pour chaque INO.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >432A.

Sn. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >0n18, où *n* est égal à la voie au chiffre le plus élevé au niveau de cet INO, et "18" dans les deux chiffres de poids faible indique qu'il s'agit d'une voie DO. *n* prend toujours la valeur "8", dans la mesure où ce bloc fournit toutes les huit voies pour l'INO. Le numéro de paramètre (PNO) occupé par ce paramètre dépend de INO:

$$PNO = [(INO.AND.3) * 2] + 1$$

SW. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale qui représente les positions des commutateurs du T600.

ID. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >640n (indiquant T640, version *n*).

ST. Prend toujours la valeur hexadécimale >0000.

SL0832 : BLOC MODELE DE PSEUDO-INSTRUMENT 0832

Fonction du bloc

Le bloc SL0832 fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments multi-voies, multi-boucles du système 6000 par l'intermédiaire des communications binaires. Les huit voies sont regroupées dans un seul bloc.

Le bloc est considéré par les communications binaires comme étant identique aux pseudo-instruments de type 0832 fournis dans les S6437-113.

Numéros des paramètres SL0832. La table 3-20 donne les numéros des paramètres pour les communications 0832. "0" dans une case indique une valeur fictive lue comme zéro et écrite comme NAK (accusé de réception négatif).

	0	1	2	3	4	5	6	7
	II							
	8	9	10	11	12	13	14	15
							ID	
Chn 1	16	17	18	19	20	21	22	23
	0	0	P1_1	P2_1	0			
Chn 2	24	25	26	27	28	29	30	31
	0	0	P1_2	P2_2	0			
Chn 3	32	33	34	35	36	37	38	39
	0	0	P1_3	P2_3	0			
Chn 4	40	41	42	43	44	45	46	47
	0	0	P1_4	P2_4	0			
Chn 5	48	49	50	51	52	53	54	55
	0	0	P1_5	P2_5	0			
Chn 6	56	57	58	59	60	61	62	63
	0	0	P1_6	P2_6	0			
Chn 7	64	65	66	67	68	69	70	71
	0	0	P1_7	P2_7	0			
Chn 8	72	73	74	75	76	77	78	79
	0	0	P1_8	P2_8	0			
	80	81	82	83	84	85	86	87

Table 3-20 Numéros des paramètres pour le bloc SL0832

Paramètres du bloc

Les symboles utilisés dans la table 3-21 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.









Paramètre	Fonction	Unités	Etat
PV_1 à PV_8	Variables procédé à 32 bits 1 à 8 respectivement	Eng	
P1_1 à P1_8	16 MSB de PV_1 à PV_8 respectivement	ABCD hex	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
II	Identité de l'instrument (8320)	ABCD hex	
ID	Modèle & version de l'instrument T600	ABCD hex	
P2_1 à P2_8	16 LSB de PV_1 à PV_8 respectivement	ABCD hex	

Table 3-21 Paramètres du bloc SL0832

Menu de spécifications du bloc.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV_1 à PV_8. Huit nombres IEEE à point flottant de 32 bits. Chaque champ *PV_n* occupe une voie d'espace de communications binaires et afin de transférer avec succès l'ensemble des 32 bits par l'intermédiaire des communications binaires, chaque champ occupe une paire de paramètres de communications binaires - *PI_n* et *P2_n*.

P1_1 à P1_8, P2_1 à P2_8. *PI_n* contient les 16 bits de poids fort du nombre *PV_n* à 32 bits, et *P2_n* les 16 bits de poids faible. La version réelle du nombre à point flottant à 32 bits (*PV_n*), ainsi que la paire des versions entières à 16 bits du nombre sont disponibles dans les champs du bloc.

Afin de maintenir la cohérence des données au moment de l'écriture sur le réseau de communications binaires, *PI_n* doit être écrit en premier, suivi de *P2_n*. *PI_n* n'est pas mis à jour immédiatement, mais continue d'indiquer sa valeur originelle jusqu'à l'écriture de *P2_n*. *PV_n* n'est pas mis à jour jusqu'à l'écriture de *P2_n*.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Noter qu'il ne peut y avoir qu'un bloc SL0832 pour chaque INO.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >8320. (Il s'agit de l'II retourné par les pseudo-instruments 0832 dans les 6437-113).

ID. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient la valeur hexadécimale >640*n* (indiquant T640, version *n*).

SL6437 : BLOC MODELE DE PSEUDO-INSTRUMENT 6437

Fonction du bloc

Le bloc SL6437 fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec des instruments mono-boucle, multi-voies du système 6000 par l'intermédiaire des communications binaires. Chaque bloc SL6437 représente un instrument S6000 discret sur un INO unique (n° d'instrument des communications binaires).

Ce bloc est considéré par les communications binaires comme étant identique au pseudo-instrument de type 6437 fourni dans le micro-superviseur 6445.

Numéros des paramètres SL6437. La table 3-22 donne les numéros des paramètres pour les communications SL6437.

0	1	2	3	4	5	6	7
II							
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
				ST	HR	LR	HA
24	25	26	27	28	29	30	31
LA	FL	F1	F2				
32	33	34	35	36	37	38	39

Table 3-22 Numéros des paramètres pour le bloc SL6437.

Paramètres du bloc

Les symboles utilisés dans la table 3-23 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

FL, HR, LR, HA, LA, ST. Ces paramètres sont de simples sauvegardes de données.

Total. Un nombre entier de 32 bits. Afin de transférer avec succès l'ensemble des 32 bits par l'intermédiaire des communications binaires, *Total* est également exprimé comme un nombre décimal à huit chiffres enregistré dans deux paramètres de communications binaires - *F1* et *F2*.

F1, F2. *F1* contient les 4 chiffres décimaux de poids fort de l'entier *Total*, exprimé sous la forme d'un nombre décimal à huit chiffres, et *F2* les 4 chiffres décimaux de poids faible. Dans la base de données LIN, *Total* peut dépasser huit chiffres décimaux. Dans ce cas, *F1* et *F2* sont affichés sous la forme "9999".

Afin de maintenir la cohérence des données au moment de l'écriture sur le réseau de communications binaires, *F1* doit être écrit en premier, suivi de *F2*. *F1* n'est pas mis à jour immédiatement, mais continue d'indiquer sa valeur originelle jusqu'à l'écriture de *F2*. *Total* n'est pas mis à jour jusqu'à l'écriture de *F2*.

Paramètre	Fonction	Unités	Etat
FL	Paramètre fictif (sauvegarde de données lect/écrit)	Eng	
HR	Paramètre fictif (sauvegarde de données lect/écrit)	Eng	
LR	Paramètre fictif (sauvegarde de données lect/écrit)	Eng	
HA	Paramètre fictif (sauvegarde de données lect/écrit)	Eng	
LA	Paramètre fictif (sauvegarde de données lect/écrit)	Eng	
Total	Valeur "total" à 32 bits	Entier	
F1	16 MSB de total	ABCD hex	
F2	16 LSB de total	ABCD hex	
Range	Position du point décimal	Menu	
DP_Posn	Position du point décimal	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
II	Identité de l'instrument (4370)	ABCD hex	
ST	Paramètre d'état fictif (>0000)	ABCD hex	
Bit0	IP3oor	T/F — 1	D
Bit1	IP2oor	V/F — 2	
Bit2	IP1oor	V/F — 4	
Bit3	Sumcheck (défaut paramètre voie)	V/F — 8	
Bit4	(Inutilisé)	V/F — 1	C
Bit5	TotalDft (totalisation en défaut)	V/F — 2	
Bit6	(Inutilisé)	V/F — 4	
Bit7	TotalRol (drapeau dépassement total)	V/F — 8	
Bit8	TotalHld (bit maintien totalisation)	V/F — 1	B
Bit9	PulseFlt (défaut voie d'entrée impulsion)	V/F — 2	
Bit10	HHtstrobe (valid. écrit. micro-console total débit)	V/F — 4	
Bit11	ForthErr (erreur équation débit)	V/F — 8	
Bit12	(Inutilisé)	V/F — 1	A
Bit13	(Inutilisé)	V/F — 2	
Bit14	(Inutilisé)	V/F — 4	
Bit15	LowFreq (mode basse fréquence)	V/F — 8	

Table 3-23 Paramètres du bloc SL6437

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI en cas d'alarme active dans le bloc. Adopte le même message d'état et n° de priorité que l'alarme à la priorité la plus haute du bloc.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Noter qu'il ne peut y avoir qu'un bloc SL6437 pour chaque INO.

II. Il s'agit d'un champ WRITE_PROT (PROTEGE EN ECRITURE) en lecture uniquement qui contient toujours la valeur hexadécimale >4370. (II s'agit de l'II retourné par les pseudo-instruments 6437 dans micro-superviseur 6445).

ST. Le champ peut être utilisé pour le stockage de données.

SL6366 : BLOC MODELE DE REGULATEUR 6366

Fonction du bloc

Le bloc SL6366 fournit un modèle de paramètres pour les instruments T600, ce qui permet de communiquer avec les instruments 6366 et 6356 du système 6000 par l'intermédiaire des communications binaires. Un bloc d'extension SL6366EX peut également être utilisé en liaison avec un bloc SL6366. Le bloc d'extension se rattache automatiquement au bloc SL6366 de base par un numéro INO commun et permet de gérer des paramètres supplémentaires (sauvegarde de données et positions du point décimal).

Numéros des paramètres SL6366. La table 3-24 donne les numéros des paramètres pour les communications SL6366. "0" dans une case indique une valeur fictive, lue comme zéro et écrite comme NAK (accusé de réception négatif). Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception. "DP = n" indique un paramètre à point décimal fixe avec n positions décimales

0	1	2	3	4	5	6	7
II	DP	PH	PL	HD	LD	DC_ST	SP
8	9	10	11	12	13	14	15
PV	OP	HA	LA	HS	LS	HO	LO
16	17	18	19	20	21	22	23
HR	LR	SL		XP	TI	TD	
24	25	26	27	28	29	30	31
				RS	RB		SW
32	33	34	35	36	37	38	39
		TS	ER				
40	41	42	43	44	45	46	47
GP_ST	0	0	0	0			
48	49	50	51	52	53	54	55
0	0	0	0	0		0	0
56	57	58	59	60	61	62	63
0	0	0		0	0	0	0
64	65	66	67	68	69	70	71
0		0	0	0	0	0	0
72	73	74	75	76	77	78	79
0	0	0	0	0	0	SR	SB
80	81	82	83	84	85	86	87
RL	RB_ST	RB_RT		3T_ST	FF	3T_FB	3T_OP
88	89	90	91	92	93	94	95
MS_ST	HV	LV	MS_AO	OT		0	0
96	97	98	99	100	101	102	103
0	0	ES	SM		AB_ST	AB_HV	AB_LV
104	105	106	107	108	109	110	111
AB_HL	AB_LL	AB_PV	AB_SP	AB_AH	CB_ST	CB_1K	CB_2K
112	113	114	115	116	117	118	119
CB_3K	CB_4K	CB_US	FB_ST	FB_XK	FB_1T	FB_2T	FB_FF
120	121	122	123	124	125	126	127
FB_FI	FB_OP	DB_ST	DB_DT	TB_ST	TB_FS	TB_FT	





Table 3-24 Numéros des paramètres pour le bloc SL6366 & SL6366R

"DP = A" indique un paramètre dp "primaire" calculé à partir de PH/PL; "DP = B" indique un dp calculé à partir de RB_ST, chiffre A; "DP = C" indique un dp calculé à partir de CB_ST, chiffres A, B, C et D. Les paramètres en lettres creuses (HR, par ex.) ne sont disponibles qu'en cas de rattachement d'un bloc SL6366EX, sinon les valeurs sont fictives.

Paramètres du bloc

La table 3-25 donne la liste des paramètres SL6366 et de leurs sources. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*. Tous les paramètres du bloc SL6366 - à l'exception de *Loop*, *Alarms*, *Instr_No* et *Update* - sont calculés ou copiés à partir d'autres blocs dans la boucle (tâche utilisateur) spécifiés dans le champ *Loop*. Le bloc SL6366 s'associe automatiquement à ces blocs, sans que des liaisons de schéma de boucles ne soient nécessaires. Notez que si le bloc SL6366 ne peut trouver les sources de ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale des données.

La plupart des champs calculés sont des copies directes de champs équivalents soit dans un bloc PID ou PID_CONN. Les tentatives d'écriture dans ces champs sont redirigées vers le bloc source authentique de ces valeurs. Donc, si vous utilisez le bloc 6366 pour faire l'interface avec les blocs SETPOINT, 3_TERM, MAN_STAT et MODE, un bloc PID_CONN est également indispensable.

Paramètre	Source	Unités	Etat
II	Identité instrument (>366A)	ABCD hex	
DP		ABCD hex	
Bit0	(FAUX)	V/F — 1	D
Bit1	(FAUX)	V/F — 2	
Bit2	(FAUX)	V/F — 4	
Bit3	(FAUX)	V/F — 8	
Bit4	PID*.Alarms.LowDev.InAlarm	V/F — 1	C   
Bit5	PID*.Alarms.HighDev.InAlarm	V/F — 2	
Bit6	PID*.Alarms.LowAbs.InAlarm	V/F — 4	
Bit7	PID*.Alarms.HighAbs.InAlarm	V/F — 8	
Bit8	(FAUX)	V/F — 1	B
Bit9	(FAUX)	V/F — 2	
BitA	PID*.Options.EnbTrack	V/F — 4	
BitB	NOT(PID*.Options.IntBalSL)	V/F — 8	
BitC	} Calculé à partir de PH & PL	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
PH	PID*.HR_SP	Eng	
PL	PID*.LR_SP	Eng	
HD	PID*.HDA	Eng	
LD	PID*.LDA	Eng	
DC_ST		ABCD hex	










suite...

...suite

Paramètre	Source	Unités	Etat
Bit0	<div> <div> <div>Bit0</div> <div>Bit1</div> <div>Bit2</div> <div>Bit3</div> </div> <div> <div>PID*.Mode.value</div> <div>(FAUX)</div> </div> </div>	V/F — 1	<div> <div>D</div> <div></div> <div></div> <div></div> </div>
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
Bit4	PID*.ModeAct.NotRem	V/F — 1	<div> <div></div> <div>C</div> <div></div> <div></div> </div>
Bit5	PID*.ModeSel.EnaRem	V/F — 2	
Bit6	PID*.ModeAct.AutoAct	V/F — 4	
Bit7	PID*.Alarms.HighAbs.InAlarm	V/F — 8	
Bit8	(FAUX)	V/F — 1	<div> <div></div> <div></div> <div>B</div> <div></div> </div>
Bit9	(FAUX)	V/F — 2	
BitA	PID*.SelMode.EnaRem (Read Only)	V/F — 4	
BitB	PID*.ModeSel.RemSel	V/F — 8	
BitC	PID*.ModeSel.AutoSel	V/F — 1	<div> <div></div> <div></div> <div>A</div> <div></div> </div>
BitD	PID*.ModeSel.ManSel	V/F — 2	
BitE	PID*.ModeSel.TrackSel	V/F — 4	
BitF	PID*.ModeSel.HoldSel	V/F — 8	
SP	PID*.SP	Eng	
PV	PID*.PV	Eng	
OP	PID*.OP	Eng	
HA	PID*.HAA	Eng	
LA	PID*.LAA	Eng	
HS	PID*.HL_SP	Eng	
LS	PID*.LL_SP	Eng	
HO	PID*.HL_OP	Eng	
LO	PID*.LL_OP	Eng	
SL	PID*.SL	Eng	
SR	PID*.RemoteSP	Eng	
Loop	Boucle contenant les blocs source authentiques (1-4)	Entier	
Alarms			<div> <div></div> <div></div> <div></div> </div>
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	Numéro instrument RS422 (INO)	(0-127)	
XP	PID*.XP	Eng	
TI	PID*.TI	Eng	
TD	PID*.TD	Eng	
SW	Etat des positions commutateurs de l'instrument T600	ABCD hex	
TS	3_TERM.TS	Eng	
SB	PID*.TrimSP	Eng	
RL	SETPOINT.RateLim	Eng	
3T_ST		ABCD hex	
Bit0	(FAUX)	V/F — 1	<div> <div>D</div> <div></div> <div></div> <div></div> </div>
Bit1	(FAUX)	V/F — 2	
Bit2	(FAUX)	V/F — 4	
Bit3	(FAUX)	V/F — 8	







suite...

...suite

Paramètre	Source	Unités	Etat
Bit4	3_TERM.Status.LoLim	V/F — 1	
Bit5	3_TERM.Status.HighLim	V/F — 2	
Bit6	PID*.Options.IntBal	V/F — 4	
Bit7	PID*.Options.InvPID	V/F — 8	
Bit8	PID*.TimeBase.value (Secs = 0; Mins = 1)	V/F — 1	
Bit9	(FALSE)	V/F — 2	
BitA	(FALSE)	V/F — 4	
BitB	(FALSE)	V/F — 8	
BitC	(FALSE)	V/F — 1	
BitD	(FALSE)	V/F — 2	
BitE	(FALSE)	V/F — 4	
BitF	(FALSE)	V/F — 8	
FF	PID*.FF_PID	Eng	
HV	MAN_STAT.UpRate	Eng	
LV	MAN_STAT.DnRate	Eng	
OT	PID*.Track	Eng	
ES		ABCD hex	
Bit0	PID*.SelMode.SelMan	V/F — 1	
Bit1	PID*.SelMode.SelAuto	V/F — 2	
Bit2	PID*.SelMode.SelRem	V/F — 4	
Bit3	PID*.SelMode.SelFMan	V/F — 8	
Bit4	(FALSE)	V/F — 1	
Bit5	PID*.SelMode.EnaRem	V/F — 2	
Bit6	PID*.SelMode.SelTrack	V/F — 4	
Bit7	NOT(PID*.SelMode.SelHold)	V/F — 8	
Bit8	Masque d'écriture pour le bit0	V/F — 1	
Bit9	Masque d'écriture pour le bit1	V/F — 2	
BitA	Masque d'écriture pour le bit2	V/F — 4	
BitB	Masque d'écriture pour le bit3	V/F — 8	
BitC	Masque d'écriture pour le bit4	V/F — 1	
BitD	Masque d'écriture pour le bit5	V/F — 2	
BitE	Masque d'écriture pour le bit6	V/F — 4	
BitF	Masque d'écriture pour le bit7	V/F — 8	
SM		ABCD hex	
Bit0	MODE.PBmasks.Manual	V/F — 1	
Bit1	MODE.PBmasks.Auto	V/F — 2	
Bit2	MODE.PBmasks.Remote	V/F — 4	
Bit3	SETPOINT.Options.DisSL_RL	V/F — 8	
Bit4	MAN_STAT.Options.Dis_M_RL	V/F — 1	
Bit5	(FAUX)	V/F — 2	
Bit6	(FAUX)	V/F — 4	
Bit7	(FAUX)	V/F — 8	

suite...

...suite

Paramètre	Source	Unités	Etat
Bit8	(FAUX)	V/F — 1	B 
Bit9	(FAUX)	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A 
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
GP_ST		ABCD hex	
Bit0	T600.Alarms.UTskEr3.InAlarm [UTskEr1]†	V/F — 1	D 
Bit1	T600.Alarms.UTskEr2.InAlarm [UTskEr2]†	V/F — 2	
Bit2	T600.Alarms.UTskEr3.InAlarm [UTskEr3]†	V/F — 4	
Bit3	T600.Alarms.UTskEr4.InAlarm [UTskEr4]†	V/F — 8	
Bit4	(FAUX)	V/F — 1	C 
Bit5	(FAUX)	V/F — 2	
Bit6	(FAUX)	V/F — 4	
Bit7	(FAUX)	V/F — 8	
Bit8	(FAUX)	V/F — 1	B 
Bit9	T600.Status.PwrFail	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	T600.Status.TmpPFail	V/F — 1	A 
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
Update	Vitesse de mise à jour du bloc (sec)	Eng	

†Nom de bit adressable individuellement *Peut aussi être PID_CONN

Table 3-25 Paramètres du bloc SL6366

Certains champs SL6366 sont des copies de champs d'autres blocs: 3_TERM, SETPOINT, MAN_STAT, MODE et le bloc T600.

Menu de spécifications du bloc.

Les informations suivantes sont complémentaires de celles de la table 3-25.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

DC_ST. *Bit0 à Bit2* (c'est à dire chiffre hexadécimal D, dans la mesure où *Bit3* est toujours FAUX) peuvent être écrits par l'intermédiaire des communications binaires, afin de changer de mode. Les modes et numéros de mode (valeurs du chiffre D) sont les suivants: Maintien = 0, Poursuite = 1, Manuel = 2, Auto local = 3, Rapport = 4, Déporté = 5, Manuel forcé = 6, Reprise auto = 7.

Notez que si c'est simplement le bloc PID qui est utilisé, plutôt que le bloc PID_CONN, *Options.MSelFunc* doit être VRAI pour empêcher que le paramètre ModeSel ait plus d'un de ses bits de sélection de mode actif à un moment donné.

OP. *OP* peut uniquement faire l'objet d'une écriture lorsque le bloc PID ou PID_CONN est en mode manuel.

Loop (Boucle). Permet de spécifier quelle boucle (tâche utilisateur) doit faire l'objet d'une recherche de blocs source authentiques (PID/PID_CONN, SETPOINT, 3_TERM, MAN_STAT, MODE). Les valeurs qui peuvent être utilisées sont 1 à 4. Si *Loop* est mis à 0, ou si les blocs source en question ne peuvent être trouvés, le bloc traitera les champs pertinents comme de simples sauvegardes de données locales.

La valeur de *Loop* ne s'applique pas au calcul du paramètre *GP_ST* qui est toujours fourni par le bloc T600 (racine).

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Notez qu'il ne peut y avoir qu'un bloc SL6366 pour chaque INO.

Un bloc d'extension SL6366EX peut être rattaché à un bloc SL6366, en donnant au bloc d'extension la même valeur *Instr_No*.

NOTA. Affectez la même valeur de numéro d'instrument unique au bloc SL6366 et au bloc d'extension SL6366EX à rattacher. Seul un bloc SL6366EX peut être rattaché à un bloc SL6366.

ES. *Bit8* à *BitF* servent de masques d'écriture pour le *Bit0* à *Bit7* correspondant, c'est à dire que si un masque de bit est VRAI, le bit *ES* correspondant reste inchangé en cas d'écriture.

GP_ST. *Bit9* est le seul bit de lecture/écriture du paramètre *GP_ST*, mais peut également faire l'objet d'une écriture uniquement pour passer de VRAI à FAUX.

Update (Mise à jour). Permet de définir la vitesse de mise à jour du bloc (en secondes).

Notez que la vitesse de mise à jour du bloc ne peut être supérieure à la vitesse de répétition de la boucle à laquelle le bloc SL6366 est affecté. L'utilisation d'*Update* est destinée à ralentir la vitesse de mise à jour du bloc, afin d'éviter que les drapeaux de scrutation par exception ne soient mis à 1 trop fréquemment, ce qui peut surcharger les communications binaires.

SL6360 : BLOC MODELE DE REGULATEUR 6360

Fonction du bloc

Le bloc SL6360 fournit un modèle de paramètres pour les instruments T600, ce qui permet d'émuler un régulateur 6360 du système 6000 sur les communications binaires.

Numéros des paramètres SL6360. La table 3-26 donne les numéros des paramètres pour les communications SL6360. Un rectangle vierge indique l'absence de paramètres. Les rectangles grisés indiquent des paramètres soumis à la scrutation par exception. "DP = *n*" indique un paramètre à point décimal fixe avec *n* positions décimales. "DP = A, B, C ou D" indique une position de point décimal déterminée par le chiffre du paramètre DP indiqué.

0	1	2	3	4	5	6	7
II	DP	1H DP=A	1L DP=A	HD DP=A	LD DP=A	MN	SP DP=A
8	9	10	11	12	13	14	15
PV DP=A	OP DP=2	HA DP=A	LA DP=A	HS DP=A	LS DP=A	HO DP=2	LO DP=2
16	17	18	19	20	21	22	23
HR DP=D	LR DP=D	SL DP=A	EL DP=2	XP DP=1	TI DP=2	TD DP=2	2H DP=B
24	25	26	27	28	29	30	31
2L DP=B	3H DP=C	3L DP=C	MP DP=C	RS DP=D	RB DP=A	IC	SW
32	33	34	35	36	37	38	39
DS	IF DP=2	TS DP=2	ER DP=A	MD			
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71
72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87
88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103
104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119
120	121	122	123	124	125	126	127

Table 3-26 Numéros des paramètres pour le bloc SL6360 & SL6360R






Paramètres du bloc

Tous les paramètres du bloc SL6360 - à l'exception de *Loop*, *Alarms*, *Instr_No* et *Update* - sont calculés ou copiés à partir d'autres blocs dans la boucle (tâche utilisateur) spécifié dans le champ *Loop*. Le bloc SL6360 s'associe automatiquement à ces blocs, sans que des liaisons de schéma de boucles ne soient nécessaires. Notez que si le bloc SL6360 ne peut trouver les sources de ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale des données.

La plupart des champs calculés sont des copies directes de champs équivalents soit dans un bloc PID ou PID_CONN. Les tentatives d'écriture dans ces champs sont redirigées vers le bloc source authentique de ces valeurs. Donc, si vous utilisez le bloc 6360 pour faire l'interface avec les blocs SETPOINT, 3_TERM, MAN_STAT et MODE, un bloc PID_CONN est également indispensable.


























Certains champs SL6360 sont des copies de champs d'autres blocs: 3_TERM, SETPOINT, MAN_STAT, MODE et le bloc T600.

La table 3-27 donne la liste des paramètres SL6360 et de leurs sources. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Source	Unités	Etat
II	Identité de l'instrument (> 360A)	ABCD hex	
DP	Points décimaux (sélectionnables par l'utilisateur)	ABCD hex	
Bit0	} 'D' = 2 - 'A' + 'B' (0-4)	V/F — 1	   
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
Bit4	} Sélection du point décimal 'C' (= 'D' si MD.MPDisSel est VRAI)	V/F — 1	C
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8	} Sélection du point décimal 'B' (0-4)	V/F — 1	B
Bit9		V/F — 2	
BitA		V/F — 4	
BitB		V/F — 8	
BitC	} Sélection du point décimal 'A' (0-4)	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
1H	PID*.HR_SP	Eng	
1L	PID*.LR_SP	Eng	
HD	PID*.HDA	Eng	
LD	PID*.LDA	Eng	

suite...



...suite

Paramètre	Source	Unités	Etat
MN		ABCD hex	
Bit0	} PID*.Mode.value (FAUX)	V/F — 1	D 
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
Bit4	} (FAUX)	V/F — 1	C    
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8	(FAUX)	V/F — 1	B    
Bit9	T600.Status.TmpPFail [PwrFail]†	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A    
BitD	(FAUX)	V/F — 2	
BitE	Bit de changement d'état [ChgState]†	V/F — 4	
BitF	(FAUX)	V/F — 8	
SP	PID*.SP	Eng	
PV	PID*.PV	Eng	
OP	PID*.OP	Eng	
HA	PID*.HAA	Eng	
LA	PID*.LAA	Eng	
HS	PID*.HL_SP	Eng	
LS	PID*.LL_SP	Eng	
HO	PID*.HL_OP	Eng	
LO	PID*.LL_OP	Eng	
HR	(Stockage local des données)	Eng	
LR	(Stockage local des données)	Eng	
SL	PID*.SL	Eng	
Loop	Boucles contenant les vrais blocs sources (1-4)	Entier	
Alarms			  <)
Software	Anomalie mémoire données RAM/défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	n° d'instrument RS422 (INO)	(0-127)	
EL	(Stockage local des données)	Eng	
XP	PID*.XP	Eng	
TI	PID*.TI	Eng	
TD	PID*.TD	Eng	
2H	(Stockage local des données)	Eng	
2L	(Stockage local des données)	Eng	
3H	(Stockage local des données)	Eng	
3L	(Stockage local des données)	Eng	
MP	MAN_STAT.MeasPos	Eng	
RS	(Stockage local des données)	Eng	
RB	(Stockage local des données)	Eng	

†Nom de bit adressable individuellement

suite...







...suite

Paramètre	Source	Unités	Etat
IC		ABCD hex	
Bit0	MODE.PB_Masks.Manual [MaskMan]†	V/F — 1	D
Bit1	MODE.PB_Masks.Auto [MaskAuto]†	V/F — 2	
Bit2	MODE.PB_Masks.Remote [MaskRem]†	V/F — 4	
Bit3	VRAI = désactivation BP R, A, & M [MaskAll]†	V/F — 8	
Bit4	} (Stockage local des données)	V/F — 1	C
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8	} (Stockage local des données)	V/F — 1	B
Bit9		V/F — 2	
BitA		V/F — 4	
BitB		V/F — 8	
BitC	} (Stockage local des données)	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
SW		ABCD hex	 
S18_GID0	} (Stockage local des données)	V/F — 1	D
S17_GID1		V/F — 2	
S16_GID2		V/F — 4	
S15_BIN		V/F — 8	
S14_BAUD	} (Stockage local des données)	V/F — 1	C
S13_BAUD		V/F — 2	
S12_BAUD		V/F — 4	
IntBalnc		V/F — 8	
TimeBase	} (Stockage local des données)	V/F — 1	B
Chan3OP		V/F — 2	
SPrkPV		V/F — 4	
PFailRet		V/F — 8	
Inv3Term	} (Stockage local des données)	V/F — 1	A
InvRatio		V/F — 2	
OPDisInv		V/F — 4	
RatioEnb		V/F — 8	

†Nom de bit adressable individuellement

suite...

...suite

Paramètre	Source	Unités	Etat
DS		ABCD hex	
NotHiAlm	NOT (PID*.Alarms.HighAbs OR PID*.Alarms.HighDev)	V/F — 1	D
NotLoAlm	NOT (PID*.Alarms.LowAbs OR PID*.Alarms.LowDev)	V/F — 2	
NotHWAIm	(calculé à partir de la liaison d'entrée)	V/F — 4	
NotBatLo	NOT (Batterie faible) (toujours VRAI)	V/F — 8	
NotRmRat	NOT (PID*.ModeAct.RemAct)	V/F — 1	C
NotHdMan	NOT (PID*.ModeSel.HoldSel OR PID*.ModeSel.ManSel)	V/F — 2	
UserBit1	(bit logique utilisateur — stockage des données)	V/F — 4	
UserBit2	(bit logique utilisateur — stockage des données)	V/F — 8	
UID1	('sélection UID' — toujours FAUX)	V/F — 1	B
UID2		V/F — 2	
UID4		V/F — 4	
UID8		V/F — 8	
CompEnb	Ordinateur activé (toujours VRAI)	V/F — 1	A
RmRatEnb	Rapport activé (toujours VRAI)	V/F — 2	
TrackEnb	PID*.SelMode.SelTrack	V/F — 4	
NotHdEnb	NOT (PID*.SelMode.SelHold)	V/F — 8	
IF	(Stockage local des données)	Secs	
TS	3_TERM.TS	Eng	 
ER	= PV — SP	Eng	
MD		ABCD hex	
ManualPB	MODE.PBTests.Manual	V/F — 1	D
AutoPB	MODE.PBTests.Auto	V/F — 2	
RemotePB	MODE.PBTests.Remote	V/F — 4	
SetpntPB	MODE.PBTests.SetPoint	V/F — 8	
MPDisSel	MAN_STAT.Options.MPosDisp	V/F — 1	C 
LEDtest	T600.Options.LEDtest	V/F — 2	
RaisePB	MODE.PBTests.Up	V/F — 4	
LowerPB	MODE.PBTests.Down	V/F — 8	
Sumcheck	T600.Alarms.ComS/W	V/F — 1	B
PVOCct	(calculé à partir de la liaison d'entrée)	V/F — 2	
RmRatEnb	Ratio enable (toujours VRAI)	V/F — 4	
RemRatio	PID*.ModeAct.RemAct	V/F — 8	
Auto	PID*.ModeAct.AutoAct	V/F — 1	A
Manual	PID*.ModeAct.ManAct OR PID*.ModeAct.FManAct	V/F — 2	
Track	PID*.SelMode.SelTrack	V/F — 4	
Hold	PID*.SelMode.SelHold	V/F — 8	
Update	Intervalle de mise à jour du bloc (secs)	Eng	

*peut également être un bloc PID_CONN

Table 3-27 Paramètres du bloc SL6360

Menu de spécifications du bloc.

Les informations suivantes sont complémentaires de celles de la table 3-27.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

DP. Ce paramètre permet de définir trois positions de point décimale différentes pour les communications binaires (l'autre est calculée par le bloc). Chaque valeur de point décimal est limitée à l'échelle de 0 à 4.

- Le point décimal "A" s'applique aux paramètres suivants lorsqu'ils sont transmis par l'intermédiaire des communications binaires: *1H, 1L, HS, LS, HA, LA, HD, LD, SL; RB, SP, PV* et *ER*. "A" est automatiquement copiés en fonction de la valeur de *Dis_DP* du bloc *SETPOINT* de cette boucle.
- Le point décimal "B" s'applique aux paramètres *2H & 2L* lorsqu'ils sont transmis par l'intermédiaire des communications binaires.
- Le point décimal "C" s'applique aux paramètres *3H, 3L & MP* lorsqu'ils sont transmis par l'intermédiaire des communications binaires. Si *MD.MPDisSel* est à l'état haut, le point décimal peut être défini librement. Sinon, il est forcé à la même valeur que le point décimal "D".
- Le point décimal "D" s'applique aux paramètres *HR, LR & RS* lorsqu'ils sont transmis par l'intermédiaire des communications décimales, et est calculé d'après la formule $D = 2 - A + B$, limité à l'échelle de 0 à 4.

MN. *Bit0 à Bit2* (c'est à dire le chiffre hexadécimal D, puisque *Bit3* est toujours FALSE - Faux) peuvent être écrits par l'intermédiaire des communications binaires pour changer de mode. Les modes et leur codes (valeurs du chiffre D) sont: Hold (Maintien) = 0, Track (Poursuite) = 1, Manual (Manuel) = 2, Local Auto = 3, Ratio (Rapport) = 4, Remote (Déporté) = 5, Forced Manual (Manuel forcé) = 6 et Auto Fallback (Reprise auto) = 7.

NOTA. Si le bloc *PID* simple est utilisé, plutôt qu'un bloc *PID_CONN*, *PID.Options.MSelFunc* doit être TRUE (vrai) pour empêcher que plus d'un bit de sélection de mode du paramètre *ModeSel* ne soit actif à la fois.

Le *Bit9* est mis à 1 au moment de activation du bloc - il n'est pas nécessaire qu'il soit mis à jour continuellement. Le *BitE* est mis à 1 chaque fois qu'un paramètre non scrutés par exception change sans que ce soit à la suite d'une écriture des communications binaires.

Loop (Boucle). Permet de spécifier quelle boucle (tâche utilisateur) doit faire l'objet d'une recherche de vrais blocs source (*PID/PID_CONN, SETPOINT, 3_TERM, MAN_STAT, MODE*). Les valeurs admissibles sont 1 à 4. Si *Loop* est mis à 0, ou si les blocs source en question ne peuvent être trouvés, le bloc traitera les champs pertinents comme de simples sauvegardes de données locales.

Alarms (Alarmes). Voir la description générale du champ *Alarms* à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc à la priorité la plus haute.

Instr_No (N° d'instrument). Ce champ définit l'INO (numéro d'instrument) que ce bloc de base de données occupe. Noter qu'il ne peut y avoir qu'un bloc SL6360 pour chaque INO.

IC. La fonction de conditionnement de l'entrée de ce paramètre n'est pas gérée par le bloc SL6360. Les Bits 4 à 15 peuvent être écrits, mais ne sont utilisés que comme de simples sauvegardes de données.

SW. Chacun des bits individuels de ce paramètre peut être associé au bit logique d'un schéma de boucles qui représente la fonction du bit équivalent dans le 6360 réel. La fonctionnalité 6360 *SW* des communications peut ainsi être émulée. *SW* ne peut être écrit par l'intermédiaire des communications binaires.

DS. Notez que *DS* ne permet pas de gérer les bits logiques utilisateur standard du 6360 ou les bits de sélection *UID*. Les bits logiques utilisateur ne peuvent être écrits, mais sont utilisés comme de simples sauvegardes de données. Le Bit 2 peut être mis à jour par une liaison de schéma de boucles.

IF. Filtre d'entrée. Pour vous aider à créer des schémas de boucles qui émulent un instrument 6360 réel, vous pouvez relier ce paramètre au champ *Filter* d'un bloc *AN_IP* pour les opérations de lecture et d'écriture.

Update (Mise à jour). Permet de définir la vitesse de mise à jour du bloc (en secondes).

Notez que la vitesse de mise à jour du bloc ne peut être supérieure à la vitesse de répétition de la boucle à laquelle le bloc SL6360 est affecté. L'utilisation d'*Update* est destinée à ralentir la vitesse de mise à jour du bloc, afin d'éviter que les drapeaux de scrutation par exception ne soient mis à 1 trop fréquemment, ce qui peut surcharger les communications binaires.

SL6366EX : BLOC D'EXTENSION SL6366

Fonction du bloc

Un bloc d'extension SL6366EX peut être utilisé en liaison avec un bloc SL6366 pour pouvoir disposer davantage de paramètres pour l'émulation d'un instrument 6366 - pour la sauvegarde de données et les spécifications de la position du point décimal. Le bloc d'extension est rattaché à un bloc de base SL6366 unique, en lui affectant le même numéro d'instrument (*Instr_No*).

NOTA. La même valeur unique de numéro d'instrument doit être affectée aux blocs SL6366 et SL6366EX à rattacher. Un seul bloc SL6366EX peut être rattaché à un bloc SL6366 donné.

Numéros des paramètres étendus. La table 3-24, dans la section sur le bloc SL6366, donne les numéros de paramètre pour les communications des blocs SL6366/SL6366EX. Dans la table, les paramètres imprimés en lettres creuses (par ex. HR) représentent les paramètres supplémentaires du bloc d'extension. Sans bloc d'extension rattaché, ces paramètres n'ont que des valeurs fictives.

Paramètres du bloc

La table 3-28 donne la liste des paramètres supplémentaires du bloc SL6366EX. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
HR	Limite haute consigne rapport	Eng	
LR	Limite basse consigne rapport	Eng	
RS	Consigne rapport	Eng	
RB	Décalage rapport	Eng	
ER*	Erreur	Eng	
RB_ST	Informations d'état du bloc rapport	ABCD hex	
RB_RT	Décalage rapport du bloc rapport	Eng	
3T_FB	Contre-réaction de la régulation PID	Eng	
3T_OP*	Sortie PID	Eng	
MS_ST	Informations d'état de la station manuelle	ABCD hex	
MS_AO	Sortie analogique de la station manuelle	Eng	
AB_ST	Informations d'état du bloc Alarmes	ABCD hex	
AB_HV	Limite haute de la valeur d'alarme	Eng	
AB_LV	Limite basse de la valeur d'alarme	Eng	
AB_HL	Limite haute du bloc Alarmes	Eng	
AB_LL	Limite basse du bloc Alarmes	Eng	
AB_PV	Variable procédé du bloc Alarmes	Eng	
AB_SP	Consigne du bloc Alarmes	Eng	
AB_AH	Hystérésis du bloc Alarmes	Eng	

*Ne peut faire l'objet d'une écriture par les communications binaires
suite...

...suite






















Paramètre	Fonction	Unités	Etat
Alarms			  
Software	Anomalie dans les données RAM du bloc/défaut réseau		V/F
BadINO	INO donné non-affecté à un bloc SL6366	V/F	
MultINO	> 1 SL6366EX bloc affecté au même INO	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	N° d'instrument RS422 (INO)	(0-127)	
CB_ST	Point décimal pour les constantes	ABCD hex	
CB_1K	Constante 1	Eng	
CB_2K	Constante 2	Eng	
CB_3K	Constante 3	Eng	
CB_4K	Constante 4	Eng	
CB_US	Mot d'état utilisateur	ABCD hex	
Bit0	Logique 0	V/F — 1	D
Bit1	Logique 1	V/F — 2	
Bit2	Logique 2	V/F — 4	
Bit3	Logique 3	V/F — 8	
Bit4	Logique 4	V/F — 1	C
Bit5	Logique 5	V/F — 2	
Bit6	Logique 6	V/F — 4	
Bit7	Logique 7	V/F — 8	
Bit8	Logique 8	V/F — 1	B
Bit9	Logique 9	V/F — 2	
Bit10	Logique 10	V/F — 4	
Bit11	Logique 11	V/F — 8	
Bit12	Logique 12	V/F — 1	A
Bit13	Logique 13	V/F — 2	
Bit14	Logique 14	V/F — 4	
Bit15	Logique 15	V/F — 8	
FB_ST	Informations d'état du bloc Filtre	ABCD hex	
FB_XK	Gain du bloc Filtre	Eng	
FB_1T	Constante d'avance du bloc Filtre	Eng	
FB_2T	Constante de retard du bloc Filtre	Eng	
FB_FF	Décalage du bloc Filtre	Eng	
FB_FI	Entrée du bloc Filtre	Eng	
FB_OP	Sortie du bloc Filtre	Eng	
DB_ST	Informations d'état du bloc Retard	ABCD hex	
DB_DT	Retard maximum (sec)	Eng	
TB_ST	Informations d'état du bloc Totalisation	ABCD hex	
TB_FS	Mise à l'échelle du débit du bloc Totalisation	Eng	
TB_FT	Total du débit du bloc Totalisation	Eng	

Table 3-28 Paramètres du bloc SL6366EX

Menu de spécifications du bloc.

Les informations suivantes sont complémentaires de celles de la table 3-28.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

ER. Ce paramètre ne peut faire l'objet d'écritures par l'intermédiaire des communications binaires comme dans l'instrument 6366 réel.

3T_OPT. Ce paramètre ne peut faire l'objet d'écritures par l'intermédiaire des communications binaires comme dans l'instrument 6366 réel.

Instr_No. (N° de l'instrument). Ce champ définit l'INO (numéro de l'instrument) occupé par le bloc SL6366EX. Sa valeur doit être la même que celle de *Instr_No* du bloc SL6366 auquel il est rattaché. Il ne peut y avoir qu'un bloc SL6366 plus un bloc SL6366EX pour chaque INO.

NOTA. Affectez la même valeur de numéro d'instrument unique au bloc SL6366 et au bloc SL6366EX qui lui est rattaché. Un seul bloc SL6366EX peut être rattaché à un bloc SL6366 donné.

Si, par erreur, vous utilisez un numéro d'INO non-affecté à un bloc SL6366, le bit d'alarme *BadINO* est mis à 1 en conduite.

Si, par erreur, vous configurez plus d'un bloc SL6366EX avec le même INO que le bloc SL6366 dont vous voulez utiliser les paramètres étendus, au cours de la conduite un seul bloc d'extension est rattaché (le premier créé). Les bits d'alarme *MultiINO* des blocs d'extension restants et non-rattachés sont alors mis à 1.

CONN32AI: BLOC DE LIAISON D'ENTREE ANALOGIQUE

Fonction du bloc

Le bloc CONN32AI permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte d'entrée analogique d'un processeur de signaux 6432. Il peut donc communiquer avec les produits 6432, 6433, 6445, 6372, 6382 et T100/RS422.

Un seul bloc CONN32AI permet de communiquer avec toutes les huit voies analogiques d'une carte d'entrée analogique 6432, ce qui concentre véritablement les communications S6000 par une mise en correspondance plus efficace des paramètres et champs au cours des lectures/écritures de la base de données. Le trafic LIN nécessaire pour retourner les données d'entrée à un superviseur est ainsi minimisé, ce qui améliore efficacement les communications LIN.

Certains paramètres 6432 sont directement disponibles sous la forme de champs - PV, HA, LA, AH et MD. D'autres - ST, HR et LR - sont "cachés" et utilisés pour former des champs de bloc CONN32AI calculés.

NOTA. Un seul bloc CONN32AI permet de communiquer avec le même nombre de voies que huit blocs 6432 AI, bien qu'il accède à moins de paramètres instrument. Seul un bloc CONN32AI peut communiquer avec une carte d'entrée donnée, et les blocs 6432 ne peuvent pas communiquer avec la carte en même temps.

Paramètres du bloc

Les symboles utilisés dans la table 3-29 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc.

Les informations suivantes sont complémentaires de celles de la table 3-29. Voir les détails sur les paramètres des instruments 6432 dans le *Manuel Technique Processeur de signaux 6432* (réf. HA 075 416 U003).

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

MODE. (AUTO/MANUEL). Le mode de fonctionnement en cours. Notez que bien que les paramètres *PV_n* puissent faire l'objet d'une écriture en mode Manuel, les valeurs sont rapidement écrasées par la scrutation suivante de l'instrument cible S6000. Ce mode n'est donc utile que lorsque l'instrument n'est pas en ligne.

Alarms (Alarmes). Voir la description du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Cette alarme est déclenchée dans les conditions suivantes:

- Corruption de la base de données dans un bloc
- Défaut de communication LIN (blocs cache uniquement)
- Dépassement du temps imparti sur la ligne de communication RS422
- Erreurs de total de contrôle de l'instrument indiquées dans le paramètre *MD* bit 8.

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement actif	Menu	
PV_1 à PV_8	Entrées du procédé (voies 1 à 8)	Eng	
HA_1 à HA_8	Limites de l'alarme haute absolue (voies 1 à 8)	Eng	
LA_1 à LA_8	Limites de l'alarme basse absolue (voies 1 à 8)	Eng	
Alarms			
Software	Anomalie données RAM di bloc/défaut réseau	V/F	
Hardware	Défaut instrument/type erroné/batterie faible, etc.	V/F	
CmnChSC	Alarme total de contrôle voie commune	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	N° d'instrument RS422 (INO)	(0 à 127)	
OutRange	Etat de l'entrée hors échelle	CD hex	
Chan_8	PV_8	V/F — 1	D
Chan_7	PV_7	V/F — 2	
Chan_6	PV_6	V/F — 4	
Chan_5	PV_5	V/F — 8	
} entrée hors échelle			
Chan_4	PV_4	V/F — 1	C
Chan_3	PV_3	V/F — 2	
Chan_2	PV_2	V/F — 4	
Chan_1	PV_1	V/F — 8	
} entrée hors échelle			
AC	Etat des alarmes actives	ABCD hex	
Chan_8	PV_8	V/F — 1	D
Chan_7	PV_7	V/F — 2	
Chan_6	PV_6	V/F — 4	
Chan_5	PV_5	V/F — 8	
} entrée en alarme basse absolue			
Chan_4	PV_4	V/F — 1	C
Chan_3	PV_3	V/F — 2	
Chan_2	PV_2	V/F — 4	
Chan_1	PV_1	V/F — 8	
} entrée en alarme basse absolue			
Chan_8	PV_8	V/F — 1	B
Chan_7	PV_7	V/F — 2	
Chan_6	PV_6	V/F — 4	
Chan_5	PV_5	V/F — 8	
} entrée en alarme haute absolue			
Chan_4	PV_4	V/F — 1	A
Chan_3	PV_3	V/F — 2	
Chan_2	PV_2	V/F — 4	
Chan_1	PV_1	V/F — 8	
} entrée en alarme haute absolue			

Table 3-29 suite...

■ **Hardware (Matériel).** Cette alarme est déclenchée dans les conditions suivantes:

- Paramètre *II* retourné incorrect
- Type de bloc incorrect
- Détection tension batterie faible (*MD* Bit 11)
- Indication de défaut matériel (*Sn* Bit 12)




Paramètre	Fonction	Unités	Etat
AH	Etat des alarmes historiques	ABCD hex	
Chan_8	PV_8	V/F — 1	D
Chan_7	PV_7	V/F — 2	
Chan_6	PV_6	V/F — 4	
Chan_5	PV_5	V/F — 8	
Chan_4	PV_4	V/F — 1	C
Chan_3	PV_3	V/F — 2	
Chan_2	PV_2	V/F — 4	
Chan_1	PV_1	V/F — 8	
Chan_8	PV_8	V/F — 1	B
Chan_7	PV_7	V/F — 2	
Chan_6	PV_6	V/F — 4	
Chan_5	PV_5	V/F — 8	
Chan_4	PV_4	V/F — 1	A
Chan_3	PV_3	V/F — 2	
Chan_2	PV_2	V/F — 4	
Chan_1	PV_1	V/F — 8	
II	Identité de l'instrument	ABCD hex	
Sn	Etat physique alvéole (type carte, scrutation voie)	ABCD hex	
MD	Etat de fonctionnement de l'instrument	ABCD hex	
PARbuton	Appui sur le bouton paramètre (PAR)	V/F — 1	D
ALMbuton	Appui sur le bouton alarme (ALM)	V/F — 2	
LOWbuton	Appui sur le bouton décrémentation (▽)	V/F — 4	
CHNbuton	Appui sur le bouton voie (CHN)	V/F — 8	
INSbuton	Appui sur le bouton inspection (INS)	V/F — 1	C
RAISEbtn	Appui sur le bouton incrémentation (Δ)	V/F — 2	
TestBit	Bit de test	V/F — 4	
HHTconn	Micro-console connectée	V/F — 8	
InsParSC	Défaut total de contrôle du paramètre Instrument	V/F — 1	B
(non utilisé)		V/F — 2	
ComonOOR	Alarme hors échelle entrée commune 1 à 5 V	V/F — 4	
LowBatry	Tension batterie basse	V/F — 8	
CommonHW	Défaut matériel commun	V/F — 1	A
CmnChSC	Défaut total contrôle paramètre voie commune	V/F — 2	
ParamCng	Modif. paramètre micro-console ou BP	V/F — 4	
StateCng	Alarme changement de l'état collecté	V/F — 8	

Table 3-29 Paramètres du bloc CONN32AI

- **CmnChSC.** Alarme de total de contrôle de la voie commune. TRUE (Vrai) si le bit 3 (défaut de total de contrôle des voies) de l'un des champs ST 6432 "cachés" passe à l'état TRUE.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc à la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). $INO = (16 \times GID) + UID$ dans la plage de 0 à 127.

OutRange (Hors échelle). Champ à 8 bits calculé à partir du bit 2 (entrée hors échelle) du paramètre "caché" ST de chaque voie.

AC. Champ à 16 bits généré localement dans le bloc, qui indique les alarmes actives des entrées. Les bits 0 à 7 signalent les alarmes absolues basses, tandis que les bits 8 à 15 indiquent les alarmes absolues hautes. La voie n est en alarme absolue basse, lorsque $PV_n < LA_n$, et en alarme absolue haute lorsque $PV_n > LA_n$. L'hystérésis de 0,5 % de l'échelle (*HR-LR*) es appliquée à la disparition de l'état d'alarme.

AH. Champ à 16 bits qui indique les alarmes historiques (non-acquittées) des entrées. Les bits 0 à 7 signalent les alarmes absolues basses, tandis que les bits 8 à 15 indiquent les alarmes absolues hautes. Le paramètre AH est copié depuis le paramètre A_n ("alarmes historiques") correspondant de l'instrument 6432.

II. Identité de l'instrument. Retourne le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

Sn. Paramètre emplacement, équivalent aux paramètres *S1* à *S4* du 6432. Renvoie les informations sur le type d'emplacement, le nombre de voies activées et l'état du matériel.

CONN32AO: BLOC DE LIAISON DE SORTIE ANALOGIQUE

Fonction du bloc

Le bloc CONN32AO permet de faire l'interface avec tous les instruments qui disposent d'une interface de communication compatible avec une carte de sortie analogique d'un processeur de signaux 6432. Il peut donc communiquer avec les produits 6432, 6433, 6445, 6372, 6382 et T100/RS422.

Un seul bloc CONN32AO permet de communiquer avec toutes les huit voies analogiques d'une carte de sortie analogique 6432, ce qui concentre véritablement les communications S6000 par une mise en correspondance plus efficace des paramètres et champs au cours des lectures/écritures de la base de données. Le trafic LIN nécessaire pour retourner les données d'entrée à un superviseur est ainsi minimisé, ce qui améliore efficacement les communications LIN.

Certains paramètres 6432 sont directement disponibles sous la forme de champs - OP, HO et LO. Les paramètres ST sont sauvegardés dans la zone "cachée" du bloc et permettent de calculer l'alarme *CmnChSC*.

NOTA. Un seul bloc CONN32AO permet de communiquer avec le même nombre de voies que huit blocs 6432 AO, bien qu'il accède à moins de paramètres instrument. Seul un bloc CONN32AO peut communiquer avec une carte de sortie donnée, et les blocs 6432 AO ne peuvent pas communiquer avec la carte en même temps.

Paramètres du bloc

Les symboles utilisés dans la table 3-30 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc.

Les informations suivantes sont complémentaires de celles de la table 3-30. Voir les détails sur les paramètres des instruments 6432 dans le *Manuel Technique Processeur de signaux 6432* (réf. HA 075 416 U003).

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Alarms (Alarmes). Voir la description du champ Alarms à la page 2-5.

- **Software (Logiciel).** Cette alarme est déclenchée dans les conditions suivantes:
 - Corruption de la base de données dans un bloc
 - Défaut de communication LIN (blocs cache uniquement)
 - Dépassement du temps imparti sur la ligne de communication RS422
 - Erreurs de total de contrôle de l'instrument indiquées dans le paramètre *MD* bit 8.
- **Hardware (Matériel).** Cette alarme est déclenchée dans les conditions suivantes:
 - Paramètre *II* retourné incorrect
 - Type de bloc incorrect
 - Détection tension batterie faible (*MD* Bit 11)
 - Indication de défaut matériel (*Sn* Bit 12)

Paramètre	Fonction	Unités	Etat
MODE	Mode de fonctionnement actif (non mis en oeuvre)	Menu	
OP_1 à OP_8	Sorties du procédé (voies 1 à 8)	Eng	
HO_1 à HO_8	Limites hautes de la sortie (voies 1 à 8)	Eng	
LO_1 à LO_8	Limites basses de la sortie (voies 1 à 8)	Eng	
Alarms			
Software	Anomalie données RAM di bloc/défaut réseau	V/F	
Hardware	Défaut instrument/type erroné/batterie faible, etc.	V/F	
CmnChSC	Alarme total de contrôle voie commune	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	N° d'instrument RS422 (INO)	(0 à 127)	
II	Identité de l'instrument	ABCD hex	
Sn	Etat physique alvéole (type carte, scrutation voie)	ABCD hex	
MD	Etat de fonctionnement de l'instrument	ABCD hex	
PARbuton	Appui sur le bouton paramètre (PAR)	V/F — 1	D
ALMbuton	Appui sur le bouton alarme (ALM)	V/F — 2	
LOWbuton	Appui sur le bouton décrémentation (∇)	V/F — 4	
CHNbuton	Appui sur le bouton voie (CHN)	V/F — 8	
INSbuton	Appui sur le bouton inspection (INS)	V/F — 1	C
RAISEbtn	Appui sur le bouton incrémentation (Δ)	V/F — 2	
TestBit	Bit de test	V/F — 4	
HHTconn	Micro-console connectée	V/F — 8	
InsParSC	Défaut total de contrôle du paramètre Instrument	V/F — 1	B
(non utilisé)		V/F — 2	
ComonOOR	Alarme hors échelle entrée commune 1 à 5 V	V/F — 4	
LowBetry	Tension batterie basse	V/F — 8	
CommonHW	Défaut matériel commun	V/F — 1	A
CmnChSC	Défaut total contrôle paramètre voie commune	V/F — 2	
ParamCng	Modif. paramètre micro-console ou BP	V/F — 4	
StateCng	Alarme changement de l'état collecté	V/F — 8	

Table 3-30 Paramètres du bloc CONN32AO

- **CmnChSC.** Alarme de total de contrôle de la voie commune. TRUE (Vrai) si le bit 3 (défaut de total de contrôle des voies) de l'un des champs ST 6432 "cachés" passe à l'état TRUE.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc à la priorité la plus haute.

Instr_No (N° d'instrument). Ce paramètre lie le bloc à un instrument en communication sur la ligne RS422 au cours de la conduite, en spécifiant son adresse comme un *Numéro d'Instrument* (INO). Des commutateurs ou un paramètre dans l'instrument permettent de définir ses Identificateurs de Groupe et d'Unité (GID et UID). $INO = (16 \times GID) + UID$ dans la plage de 0 à 127.

II. Identité de l'instrument. Retourne le type de base de données de l'instrument en communication à son numéro d'instrument. Il faut noter que certains instruments peuvent émuler les structures de bases de données d'autres types d'instruments sur le protocole binaire RS422. Donc, *II* peut ne pas correspondre au type d'instrument réel en communication à cette adresse particulière.

Sn. Paramètre emplacement, équivalent aux paramètres *S1* à *S4* du 6432. Renvoie les informations sur le type d'emplacement, le nombre de voies activées et l'état du matériel.

SL6366R : BLOC MODELE DE REGULATEUR 6366

Fonction du bloc

Le bloc SL6366R fournit un modèle de paramètres pour les instruments T932, ce qui permet d'émuler des instruments 6366 ou 6356 du système 6000 sur les communications binaires. Un bloc d'extension SL6366EX peut également être utilisé en liaison avec chaque bloc SL6366R. Le bloc se rattache automatiquement au bloc de base SL6366R par un numéro INO commun, ce qui permet de disposer de paramètres supplémentaires (sauvegarde des données et positions du point décimal).

Numéros des paramètres SL6366R. La table 3-24, dans la section *SL6366* à la page 3-47, donne les numéros des paramètres pour les communications SL6366R. La signification des symboles utilisés dans la table est donnée dans la section *SL6366R*.

Paramètres du bloc

La table 3-31 donne la liste des paramètres SL6366R et de leurs sources. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.






Paramètre	Source	Unités	Etat
II	Identité de l'instrument (> 366A)	ABCD hex	
DP		ABCD hex	
Bit0	(FAUX)	V/F — 1	D
Bit1	(FAUX)	V/F — 2	
Bit2	(FAUX)	V/F — 4	
Bit3	(FAUX)	V/F — 8	
Bit4	PID*.Alarms.LowDev.InAlarm [LowDev]†	V/F — 1	C    
Bit5	PID*.Alarms.HighDev.InAlarm [HighDev]†	V/F — 2	
Bit6	PID*.Alarms.LowAbs.InAlarm [LowAbs]†	V/F — 4	
Bit7	PID*.Alarms.HighAbs.InAlarm [HighAbs]†	V/F — 8	
Bit8	(FAUX)	V/F — 1	B
Bit9	(FAUX)	V/F — 2	
BitA	PID*.Options.EnbTrack [SL_Track]†	V/F — 4	
BitB	NOT(PID*.Options.IntBalSL) [NtInBISL]†	V/F — 8	
BitC	} Calculé à partir de PH & PL	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
PH	PID*.HR_SP	Eng	
PL	PID*.LR_SP	Eng	
HD	PID*.HDA	Eng	
LD	PID*.LDA	Eng	

Table 3-31 suite...

... Table 3-31 suite

Paramètre	Source	Unités	Etat
DC_ST		ABCD hex	
Bit0	<div> <div> <div>Bit0</div> <div>Bit1</div> <div>Bit2</div> <div>Bit3</div> </div> <div> <div>PID*.Mode.value</div> <div>(FAUX)</div> </div> </div>	V/F — 1	<div>D</div> <div></div> <div></div> <div></div>

Table 3-31 suite...

...Table 3-31 suite












Paramètre	Source	Unités	Etat
Bit4	3_TERM.Status.LoLim [LoLim]†	V/F — 1	C 
Bit5	3_TERM.Status.HighLim [HighLim]†	V/F — 2	
Bit6	PID*.Options.IntBal [IntBal]†	V/F — 4	
Bit7	PID*.Options.InvPID [InvPID]†	V/F — 8	
Bit8	PID*.TimeBase.value (Sec=0; Min=1) [Minutes]†	V/F — 1	B 
Bit9	(FAUX)	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A 
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
FF	PID*.FF_PID	Eng	
HV	MAN_STAT.UpRate	Eng	
LV	MAN_STAT.DnRate	Eng	
OT	PID*.Track	Eng	
ES		ABCD hex	
Bit0	PID*.SelMode.SelMan [SelMan]†	V/F — 1	D 
Bit1	PID*.SelMode.SelAuto [SelAuto]†	V/F — 2	
Bit2	PID*.SelMode.SelRem [SelRem]†	V/F — 4	
Bit3	PID*.SelMode.SelFMan [SelfMan]†	V/F — 8	
Bit4	[ESbit4]†	V/F — 1	C
Bit5	PID*.SelMode.EnaRem [EnaRem]†	V/F — 2	
Bit6	PID*.SelMode.SelTrack [SelTrack]†	V/F — 4	
Bit7	NOT(PID*.SelMode.SelHold) [NtSelHld]†	V/F — 8	
Bit8	Masque d'écriture pour le Bit0	V/F — 1	B
Bit9	Masque d'écriture pour le Bit1	V/F — 2	
BitA	Masque d'écriture pour le Bit2	V/F — 4	
BitB	Masque d'écriture pour le Bit3	V/F — 8	
BitC	Masque d'écriture pour le Bit4	V/F — 1	A
BitD	Masque d'écriture pour le Bit5	V/F — 2	
BitE	Masque d'écriture pour le Bit6	V/F — 4	
BitF	Masque d'écriture pour le Bit7	V/F — 8	
SM		ABCD hex	
Bit0	MODE.PBmasks.Manual [MaskMan]†	V/F — 1	D
Bit1	MODE.PBmasks.Auto [MaskAuto]†	V/F — 2	
Bit2	MODE.PBmasks.Remote [MaskRem]†	V/F — 4	
Bit3	SETPOINT.Options.DisSL_RL [DisSL_RL]†	V/F — 8	
Bit4	MAN_STAT.Options.Dis_M_RL [Dis_M_RL]†	V/F — 1	C 
Bit5	(FAUX)	V/F — 2	
Bit6	(FAUX)	V/F — 4	
Bit7	(FAUX)	V/F — 8	

Table 3-31 suite...

...Table 3-31 suite

Paramètre	Source	Unités	Etat
Bit8	(FAUX)	V/F — 1	B 
Bit9	(FAUX)	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A 
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
GP_ST		ABCD hex	
Bit0	(FAUX)	V/F — 1	D 
Bit1	(FAUX)	V/F — 2	
Bit2	(FAUX)	V/F — 4	
Bit3	(FAUX)	V/F — 8	
Bit4	(FAUX)	V/F — 1	C 
Bit5	(FAUX)	V/F — 2	
Bit6	(FAUX)	V/F — 4	
Bit7	(FAUX)	V/F — 8	
Bit8	(FAUX)	V/F — 1	B 
Bit9	(FAUX)	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A 
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
Update	Intervalle de mise à jour du bloc (sec)	Eng	

†Nom de bit adressable individuellement *Peut également être un bloc PID_LINK

Table 3-31 Paramètres du bloc SL6366R

Tous les paramètres du bloc SL6366R - sauf *Block*, *Alarms*, *Instr_No* et *Update* - sont calculés ou copiés dans d'autres blocs définis dans le champ *Block*.

Le bloc SL6366R s'associe automatiquement à ces blocs, sans que des liaisons de schéma de boucles ne soient nécessaires. Notez que si le bloc SL6366R ne peut trouver les sources de ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale des données.

La plupart des champs calculés sont des copies directes de champs équivalents soit dans un bloc PID ou PID_LINK. Les tentatives d'écriture dans ces champs sont redirigées vers le vrai bloc source de ces valeurs. Donc, si vous utilisez le bloc SL6366R pour faire l'interface avec les blocs SETPOINT, 3_TERM, MAN_STAT et MODE, un bloc PID_LINK est également indispensable.

Certains champs SL6366R sont des copies de champs d'autres blocs: 3_TERM, SETPOINT, MAN_STAT, MODE.

Menu de spécifications du bloc.

La plupart des champs du bloc SL6366R sont identiques à ceux du bloc SL6366 - voir les détails dans la section *Menu de spécifications du bloc SL6366* à la page 51. Les informations ci-dessous n'indiquent que les différences.

DC_ST. *Bit0* à *Bit2* (c'est à dire chiffre hexadécimal D, dans la mesure où *Bit3* est toujours FAUX) peuvent être écrits par l'intermédiaire des communications binaires, afin de changer de mode. Les modes et numéros de mode (valeurs du chiffre D) sont les suivants: Maintien = 0, Poursuite = 1, Manuel = 2, Auto local = 3, Rapport = 4, Déporté = 5, Manuel forcé = 6, Reprise auto = 7.

Notez que si c'est simplement le bloc PID qui est utilisé, plutôt que le bloc PID_LINK, *Options.MSelFunc* doit être VRAI pour empêcher que le paramètre *ModeSel* ait plus d'un de ses bits de sélection de mode actif à un moment donné.

OP. OP ne peut faire l'objet d'écritures que si le bloc PID ou PID_LINK est en mode manuel.

Block (Bloc). Définit par nom de repère quel bloc PID ou PID_LINK contient les données du vrai bloc source. Si les blocs sources en question ne peuvent être trouvés, le bloc source traitera les champs en question comme de simples emplacements de sauvegarde de données.

Instr_No. Ce champ définit l'INO (numéro de l'instrument) occupé par ce bloc de base de données. Notez qu'il ne peut y avoir qu'un bloc SL6366R pour chaque INO.

Un bloc d'extension SL6366EX peut être rattaché à un bloc SL6366R, en donnant au bloc d'extension la même valeur *Instr_No*.

NOTA. Affectez la même valeur de numéro d'instrument unique au bloc SL6366R et au bloc d'extension SL6366EX à rattacher. Seul un bloc SL6366EX peut être rattaché à un bloc SL6366R.

SW. SW permet de stocker localement les données, sauf pour le Bit 11, qui sélectionne la fonction "double boucle sur communications". Ce bit doit être TRUE (vrai), (c'est à dire SW = 0400 hexadécimal) pour permettre à un MaxiVis de communiquer avec la boucle 2 (adresse impaire).

SL6360R : BLOC MODELE DE REGULATEUR 6360

Fonction du bloc

Le bloc SL6360R fournit un modèle de paramètres pour les instruments T932, ce qui permet d'émuler un régulateur 6360 du système 6000 sur les communications binaires.

Numéros des paramètres SL6360R. La table 3-26, dans la section *SL6360* à la page 3-53, donne les numéros des paramètres pour les communications SL6360R. La signification des symboles utilisés dans la table est donnée dans la section *SL6360R*.

Paramètres du bloc

Tous les paramètres du bloc SL6360R - sauf *Block*, *Alarms*, *Instr_No* et *Update* - sont calculés ou copiés dans d'autres blocs définis dans le champ *Block*. Le bloc SL6360R s'associe automatiquement à ces blocs, sans que des liaisons de schéma de boucles ne soient nécessaires. Notez que si le bloc SL6360R ne peut trouver les sources de ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale des données.

Tous les paramètres peuvent faire l'objet d'une écriture soit par le schéma de boucles ou les communications binaires, sauf indication contraire dans la section *Menu de spécifications du bloc* ci-dessous.

La plupart des champs calculés sont des copies directes de champs équivalents soit dans un bloc PID ou PID_LINK. Les tentatives d'écriture dans ces champs sont redirigées vers le bloc source authentique de ces valeurs. Donc, si vous utilisez le bloc SL6360R pour faire l'interface avec les blocs SETPOINT, 3_TERM, MAN_STAT et MODE, un bloc PID_LINK est également indispensable.

Certains champs SL6360R sont des copies de champs d'autres blocs: 3_TERM, SETPOINT, MAN_STAT, MODE.

La table 3-32 donne la liste des paramètres SL6360R et de leurs sources. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.










Paramètre	Source	Unités	Etat
II	Identité de l'instrument (> 360A)	ABCD hex	
DP	Points décimaux (sélectionnables par l'utilisateur)	ABCD hex	
Bit0	} 'D' = 2 - 'A' + 'B' (0-4)	V/F — 1	   
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
Bit4	} Sélection point décimal 'C' (= 'D' si MD.MPDisSel est VRAI)	V/F — 1	   
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	

Table 3-32 suite...

...Table 3-32 suite






Paramètre	Source	Unités	Etat
Bit8	} Sélection du point décimal 'B' (0-4)	V/F — 1	B
Bit9		V/F — 2	
BitA		V/F — 4	
BitB		V/F — 8	
BitC	} Sélection du point décimal 'A' (0-4)	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
1H	PID*.HR_SP	Eng	
1L	PID*.LR_SP	Eng	
HD	PID*.HDA	Eng	
LD	PID*.LDA	Eng	
MN		ABCD hex	
Bit0	} PID*.Mode.value	V/F — 1	D
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
	(FAUX)		
Bit4	} (FAUX)	V/F — 1	C
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8	(FAUX)	V/F — 1	B
Bit9	(FAUX)	V/F — 2	
BitA	(FAUX)	V/F — 4	
BitB	(FAUX)	V/F — 8	
BitC	(FAUX)	V/F — 1	A
BitD	(FAUX)	V/F — 2	
BitE	(FAUX)	V/F — 4	
BitF	(FAUX)	V/F — 8	
SP	PID*.SP	Eng	
PV	PID*.PV	Eng	
OP	PID*.OP	Eng	
HA	PID*.HAA	Eng	
LA	PID*.LAA	Eng	
HS	PID*.HL_SP	Eng	
LS	PID*.LL_SP	Eng	
HO	PID*.HL_OP	Eng	
LO	PID*.LL_OP	Eng	
HR	(stockage local des données)	Eng	
LR	(stockage local des données)	Eng	
SL	PID*.SL	Eng	
Block	Nom de repère source du bloc PID ou PID_LINK	Alphanumérique	
Alarms			  
Software	Anomalie données RAM du bloc/défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
Instr_No	N° d'instrument RS422 (INO)	(0-127)	





Table 3-32 suite...

...Table 3-32 suite

Paramètre	Source	Unités	Etat
EL	(Stockage local des données)	Eng	
XP	PID*.XP	Eng	
TI	PID*.TI	Eng	
TD	PID*.TD	Eng	
2H	(Stockage local des données)	Eng	
2L	(Stockage local des données)	Eng	
3H	(Stockage local des données)	Eng	
3L	(Stockage local des données)	Eng	
MP	MAN_STAT.MeasPos	Eng	
RS	(Stockage local des données)	Eng	
RB	(Stockage local des données)	Eng	
IC		ABCD hex	
Bit0	MODE.PB_Masks.Manual [MaskMan]†	V/F — 1	D
Bit1	MODE.PB_Masks.Auto [MaskAuto]†	V/F — 2	
Bit2	MODE.PB_Masks.Remote [MaskRem]†	V/F — 4	
Bit3	TRUE = désactive BP R, A, & M [MaskAll]†	V/F — 8	
Bit4	} (Stockage local des données)	V/F — 1	C
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8	} (Stockage local des données)	V/F — 1	B
Bit9		V/F — 2	
BitA		V/F — 4	
BitB		V/F — 8	
BitC	} (Stockage local des données)	V/F — 1	A
BitD		V/F — 2	
BitE		V/F — 4	
BitF		V/F — 8	
SW		ABCD hex	
S18_GID0	} (Stockage local des données)	V/F — 1	D
S17_GID1		V/F — 2	
S16_GID2		V/F — 4	
S15_BIN		V/F — 8	
S14_BAUD	} (Stockage local des données)	V/F — 1	C
S13_BAUD		V/F — 2	
S12_BAUD		V/F — 4	
IntBalnc		V/F — 8	
TimeBase	} (Stockage local des données)	V/F — 1	B
Chan3OP		V/F — 2	
SPrkPV		V/F — 4	
PFailRet		V/F — 8	
Inv3Term	} (Stockage local des données)	V/F — 1	A
InvRatio		V/F — 2	
OPDisInv		V/F — 4	
RatioEnb		V/F — 8	

Table 3-32 suite...

...Table 3-32 suite

Paramètre	Source	Unités	Etat
DS		ABCD hex	
NotHiAlm	NOT (PID*.Alarms.HighAbs OR PID*.Alarms.HighDev)	V/F — 1	D
NotLoAlm	NOT (PID*.Alarms.LowAbs OR PID*.Alarms.LowDev)	V/F — 2	
NotHWAIm	(calculé à partir de la liaison d'entrée)	V/F — 4	
NotBatLo	NOT (Batterie faible) (toujours VRAI)	V/F — 8	
NotRmRat	NOT (PID*.ModeAct.RemAct)	V/F — 1	C
NotHdMan	NOT (PID*.ModeSel.HoldSel OR PID*.ModeSel.ManSel)	V/F — 2	
UserBit1	(Bit logique utilisateur — stockage données)	V/F — 4	
UserBit2	(Bit logique utilisateur — stockage données)	V/F — 8	
UID1	} ('sélection UID ' — toujours FAUX)	V/F — 1	B
UID2		V/F — 2	
UID4		V/F — 4	
UID8		V/F — 8	
CompEnb	Activation PC (toujours VRAI)	V/F — 1	A
RmRatEnb	Activation rapport (toujours VRAI)	V/F — 2	
TrackEnb	PID*.SelMode.SelTrack	V/F — 4	
NotHdEnb	NOT (PID*.SelMode.SelHold)	V/F — 8	
IF	(Stockage local des données)	Secs	
TS	3_TERM.TS	Eng	
ER	= PV – SP	Eng	
MD		ABCD hex	
ManualPB	MODE.PBTests.Manual	V/F — 1	D
AutoPB	MODE.PBTests.Auto	V/F — 2	
RemotePB	MODE.PBTests.Remote	V/F — 4	
SetpntPB	MODE.PBTests.SetPoint	V/F — 8	
MPDisSel	MAN_STAT.Options.MPosDisp	V/F — 1	C
LEDtest	(FAUX)	V/F — 2	
RaisePB	MODE.PBTests.Up	V/F — 4	
LowerPB	MODE.PBTests.Down	V/F — 8	
Sumcheck	(FAUX)	V/F — 1	B
PVOCct	(calculé à partir de la liaison d'entrée)	V/F — 2	
RmRatEnb	Activation rapport (toujours VRAI)	V/F — 4	
RemRatio	PID*.ModeAct.RemAct	V/F — 8	
Auto	PID*.ModeAct.AutoAct	V/F — 1	A
Manual	PID*.ModeAct.ManAct OR PID*.ModeAct.FManAct	V/F — 2	
Track	PID*.SelMode.SelTrack	V/F — 4	
Hold	PID*.SelMode.SelHold	V/F — 8	
Update	Intervalle de mise à jour du bloc (sec)	Eng	

*Peut également être un bloc PID_LINK

Table 3-32 Paramètres du bloc SL6360R

Menu de spécifications du bloc.

La plupart des champs du bloc SL6360R sont identiques à ceux du bloc SL6360 - voir les détails dans la section *Menu de spécifications du bloc SL6360* à la page 3-58. Les informations ci-dessous n'indiquent que les différences.

MM. *Bit0* à *Bit2* (c'est à dire chiffre hexadécimal D, dans la mesure où *Bit3* est toujours FAUX) peuvent être écrits par l'intermédiaire des communications binaires, afin de changer de mode. Les modes et numéros de mode (valeurs du chiffre D) sont les suivants: Maintien = 0, Poursuite = 1, Manuel = 2, Auto local = 3, Rapport = 4, Déporté = 5, Manuel forcé = 6, Reprise auto = 7.

NOTA. Si c'est simplement le bloc PID qui est utilisé, plutôt que le bloc PID_LINK, *PID.Options.MSelFunc* doit être VRAI pour empêcher que le paramètre *ModeSel* ait plus d'un de ses bits de sélection de mode actif à un moment donné.

Le *Bit9* est mis à 1 au moment de activation du bloc - il n'est pas nécessaire qu'il soit mis à jour continuellement. Le *BitE* est mis à 1 chaque fois qu'un paramètre non scruté par exception change sans que ce soit à la suite d'une écriture des communications binaires. Tous les autres bits sont à zéro.

Block (Bloc). Définit par nom de repère quel bloc PID ou PID_LINK contient les données du vrai bloc source. Si les blocs sources en question ne peuvent être trouvés, le bloc source traitera les champs en question comme de simples emplacements de sauvegarde de données.

Instr_No. Ce champ définit l'INO (numéro de l'instrument) occupé par ce bloc de base de données. Notez qu'il ne peut y avoir qu'un bloc SL6360R pour chaque INO.

IC. La fonction de conditionnement de l'entrée de ce paramètre n'est pas gérée par le bloc SL6360R. Les Bits 4 à 15 peuvent faire l'objet d'une écriture, mais ne sont utilisés que comme de simples sauvegardes de données.

Chapitre 4 BLOCS DE FONCTION CONDITN

INVERT : BLOC D'INVERSION ANALOGIQUE

Fonction du bloc

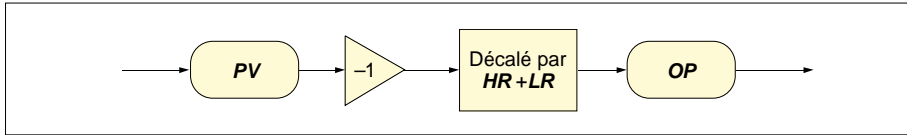


Figure 4-1 Schéma du bloc INVERT

Voir le schéma de la figure 4-1. Le bloc INVERT inverse la direction du signal d'entrée analogique, et le décale par la quantité $(HR + LR)$.

Paramètres du bloc

Les symboles utilisés dans la table 4-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée bloc)	Eng	
OP	Valeur de sortie inversée	Eng	
HR, LR	OP décalée (= HR + LR) Echelle haute & basse des objets graphiques (PV & OP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 4-1 Paramètres du bloc INVERT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d'entrée.

OP. Valeur de sortie inversée et décalée.

HR, LR. Ces paramètres décalent la sortie inversée : $OP = HR + LR - PV$. Ils mettent également à l'échelle les objets graphiques (Bargraphes ou Tendances) liés à PV ou OP ; HR et LR définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

CHAR : BLOC DE CARACTERISATION

Fonction du bloc

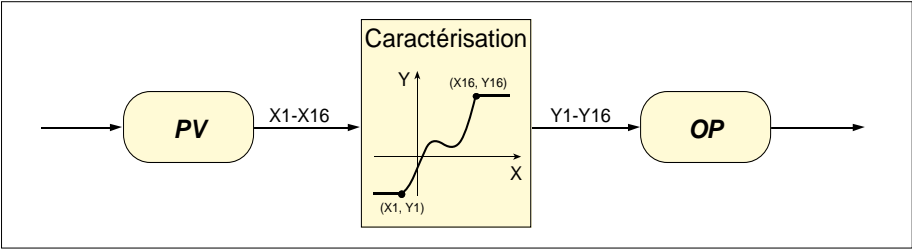


Figure 4-2 Schéma du bloc CHAR

Voir le schéma de la figure 4-2. Le bloc CHAR (caractériseur) dispose d’une fonction de caractériser analogue à 4 quadrants avec 16 points d’arrêt, entrée comme les coordonnées X (entrée) et Y (sortie). Les coordonnées X doivent croître de manière invariante, alors que les coordonnées Y peuvent prendre des valeurs croissantes, décroissantes ou récurrentes. Toute coordonnée X non utilisée est automatiquement remplie jusqu’au seizième point en copiant la valeur X la plus haute de la table.

La courbe caractéristique est interpolée linéairement entre les points d’arrêt. Les entrées inférieures à X1 ou supérieures à X16 sont sorties en Y1 ou Y16 (comme le montre le schéma).

Paramètres du bloc

Les symboles utilisés dans la table 4-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée bloc)	Eng	
OP	Sortie du bloc	Eng2	
X1 à X16	Valeurs d'entrée	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
HR_OP, LR_OP	Echelle haute & basse des objets graphiques (OP)	Eng2	
Y1 à Y16	Valeurs de sortie	Eng2	

Table 4-2 Paramètres du bloc CHAR

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d'entrée pour la caractérisation. Si $PV < X1$ ou $PV > X16$, la sortie *OP* est égale à *Y1* ou *Y16*. L'interpolation linéaire est utilisée entre les points d'arrêt. Les objets graphiques liés (bargraphe ou tendance (Bar ou Trend)) sont mis à l'échelle par *X1* et *X16*, qui définissent respectivement les affichages 0 % et 100 %.

OP. Valeur de sortie après caractérisation. Les objets graphiques liés sont mis à l'échelle par *HR_OP* (affichage 100 %) et *LR_OP* (affichage 0 %).

X1 à X16. Coordonnées d'entrée X qui doivent croître de manière invariante.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

HR_OP, LR_OP. Paramètres de mise à l'échelle haute & basse *OP* pour les objets graphiques liés.

Y1 à Y16. Coordonnées de sortie Y sans restrictions de valeur.

NOTA. Dans un bloc en mémoire cache, Y9 à Y16 ne sont pas mis à jour et ne peuvent être utilisés pour modifier les valeurs dans le bloc déporté. Il vaut mieux éviter de mettre le bloc CHAR en mémoire cache.

UCHAR : BLOC DE CARACTERISATION T100

Fonction du bloc

Le bloc UCHAR fournit la base de données pour la fonction de caractérisation à 16 points d'arrêt définie par l'utilisateur dont l'accès se fait par les blocs ANIN, TCOUPLE et RTD (adressés par l'intermédiaire de leur champs *UserChar*). Les coordonnées X (entrée) et Y (sortie) sont sauvegardées dans le menu de spécifications du bloc sous la forme de nombres normalisés à 16 bits de 0 à 65535. Ils renvoient aux valeurs équivalentes en unités physiques par l'intermédiaire d'une échelle et d'un décalage pour chaque axe.




NOTA 1. Dans une version ultérieure du bloc, les paires X et Y sont sauvegardées en valeurs physiques, et le facteur de mise à l'échelle et le décalage n'ont aucune fonction.

Plutôt que d'entrer ces nombres à 16 bits dans le bloc UCHAR, il est plus facile d'utiliser l'option LIN TABLE du configurateur T100, ce qui permet d'entrer des valeurs en unités physiques, qui sont automatiquement décalées, mises à l'échelle et sauvegardées dans un bloc appelé UCHAR. Se reporter aux détails du *Manuel de référence T1000/T100*.

NOTA 2. Un bloc UCHAR utilisé pour linéariser des mesures de thermocouple avec compensation de la soudure froide devrait comprendre un point au niveau de la température de soudure froide (par ex. 0 °C, 0 mV).

Paramètres du bloc

Les symboles utilisés dans la table 4-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
X0 à X15	Valeurs d'entrée (normalisées 16 bits*)	Numérique*	
Scale1*	Facteur de mise à l'échelle échelle entrée capteur	Eng	
Bias1*	Décalage échelle entrée capteur	Eng	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Y0 à Y15	Valeurs de sortie (normalisées 16 bits*)	Numérique*	
Scale2*	Facteur de mise à l'échelle échelle sortie caractériser	Eng	
Bias2*	Décalage échelle de sortie caractériser	Eng	

*Voir Nota 1 dans la section *Fonction du bloc*.

Table 4-3 Paramètres du bloc UCHAR

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

X0 à X15. Coordonnées d’entrée X exprimées sous le format normalisé de 16 bits de 0 à 65535. Aucune valeur ne doit être inférieure à la précédente. La valeur équivalente en unités physiques pour la coordonnée X_n est donnée par

$$X_n (Eng.) = (X_n \times Scale1) + Bias1$$

NOTA. Les valeurs de température pour $X0$ à $X15$ (et $Y0$ à $Y15$) doivent être saisies dans les unités spécifiées par le paramètre *IP_type* du bloc T100 correspondant.

Scale 1, Bias1 (Echelle1, Décalage1). Ces paramètres* permettent de convertir les valeurs X de 16 bits en unités physiques, voir la définition ci-dessus.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

Y0 à Y15. Ces coordonnées de sortie Y correspondent à l’ensemble $X0$ à $X15$ décrit ci-dessus*. La valeur équivalente en unités physiques pour la coordonnée Y_n est donnée par

$$Y_n (Eng.) = (Y_n \times Scale2) + Bias2$$

Scale2, Bias2 (Echelle2, Décalage2). Ces paramètres* renvoient les valeurs Y à 16 bits à leurs valeurs équivalentes en unités physiques, voir ci-dessus.

*Voir Nota 1 ci-dessus dans le paragraphe *Fonction du bloc*.

FILTER : BLOC FILTRE

Fonction du bloc

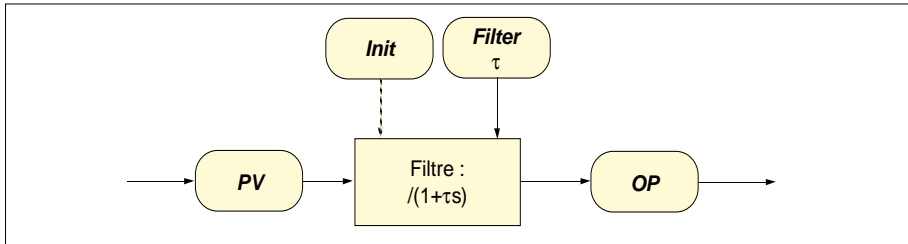


Figure 4-3 Schéma du bloc FILTRE

Voir le schéma de la figure 4-3. Le bloc FILTRE met en oeuvre une fonction de filtre du premier ordre qui peut être exprimée comme suit :

$$OP = PV / (1 + \tau s)$$

où τ est la constante de temps du filtre et s l'opérateur de Laplace ($d()/dt$). Dans la pratique, l'algorithme recalcule OP pour chaque temps d'échantillonnage du bloc comme suit :

$$OP_n = OP_{n-1} + \frac{T_s}{\tau} (PV_n - OP_{n-1})$$

- où :
- OP_n = OP au moment de l'itération en cours de l'algorithme
 - OP_{n-1} = OP au moment de l'itération précédente de l'algorithme
 - TS = Temps d'échantillonnage du bloc (secondes)
 - τ = Constante de temps du filtre (secondes). (Voir figure 4-4)
 - PV_n = PV au moment de l'itération en cours de l'algorithme

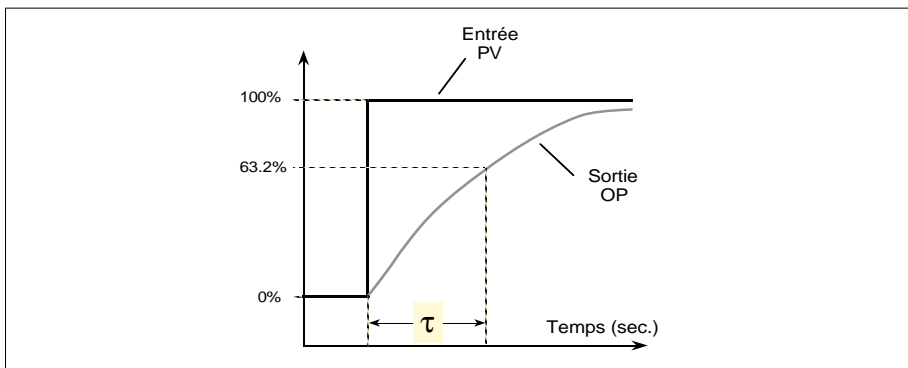


Figure 4-4 Définition de la constante de temps du bloc FILTRE

Paramètres du bloc

Les symboles utilisés dans la table 4-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée bloc)	Eng	
OP	Sortie filtrée	Eng	
HR, LR	Echelle haute/basse objets graphiques (PV & OP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Filter	Constante de temps du filtre	Secs	
Init	Initialisation	V/F	

Table 4-4 Paramètres du bloc FILTER

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d’entrée.

OP. Valeur de sortie filtrée.

HR, LR. Echelle haute & basse pour les objets graphiques (bargraphe, tendance) liée à *PV* ou *OP*. *HR* et *LR* définissent les affichages de 100 % et de 0 %.

Filter (Filtre). Constante de temps du filtre (= t secondes). Voir la définition de la figure 4-4.

Init. Initialisation. Permet d’initialiser le filtre en définissant $OP = OP_{n-1} = PV$, lorsque *Init* = VRAI. *Init* se repasse à FAUX après l’opération (écriture uniquement).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

LEADLAG : BLOC AVANCE - RETARD

Fonction du bloc

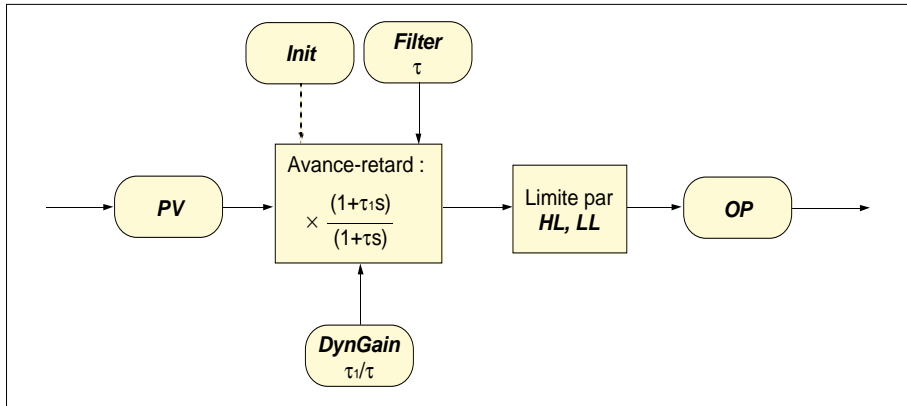


Figure 4-5 Schéma du bloc LEADLAG

Voir le schéma de la figure 4-5. Le bloc Avance-Retard permet à la sortie d’avancer ou de retarder la mesure dans le domaine de la fréquence. La fonction du bloc peut être exprimée comme suit :

$$OP = PV \frac{(1+\tau_1 s)}{(1+\tau s)}$$

où τ_1 est la constante de temps d’avance, τ la constante de temps du filtre (retard), et s l’opérateur de Laplace $d()/dt$.

Le gain dynamique (paramètre *DynGain*) est défini par τ_1/τ . Il faut noter que, si le gain dynamique est mis à 0, la fonction avance-retard est convertie en un filtre du premier ordre, et s’il est mis à 1, *OP* poursuit *PV*.

Paramètres du bloc

Les symboles utilisés dans la table 4-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d’entrée.

OP. Valeur de sortie soumise à l’avance-retard.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée bloc)	Eng	
OP	Valeur de sortie	Eng	
HL, LL	Limite sortie haute & basse Echelle haute & basse des objets graphiques (PV & OP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Filter	Constante de temps du filtre	Secs	
DynGain	Gain dynamique	Numérique	
Init	Initialisation	V/F	

Table 4-5 Paramètres du bloc LEADLAG

HL, LL. Limites hautes & basses de sortie. Ces paramètres servent également d'échelle haute & basse pour les objets graphiques (bargraphe, tendance) liés à *PV* ou *OP*. *HL* et *LL* définissent les affichages de 100 % et de 0 %.

Filter (Filtre). Constante de temps du filtre (retard) ($= \tau$ secondes).

DynGain. Gain dynamique défini par τ_1 / τ .

Init. Initialisation. Permet d'initialiser le filtre en définissant $OP = OP_{n-1} = PV$, lorsque *Init* = VRAI. *Init* repasse à FAUX après l'opération (écriture uniquement).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

AN_ALARM : BLOC D'ALARME ANALOGIQUE

Fonction du bloc

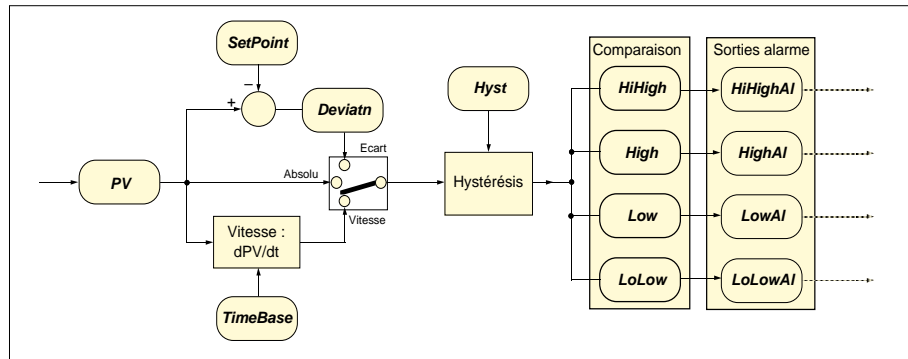


Figure 4-6 Schéma du bloc AN_ALARM

Voir le schéma de la figure 4-6. Le bloc d'alarme analogique offre des fonctions d'alarmes supplémentaires à celles intégrées dans certains des blocs de fonctions. Le bloc peut être défini pour indiquer soit les alarmes absolues, d'écart ou de vitesse, avec une bande d'hystérésis réglable et deux niveaux séparés d'alarme.

Paramètres du bloc

Les symboles utilisés dans la table 4-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Type. (ABSOLUTE/DEVIATION/RATE - ABSOLU/ECART/VITESSE). Permet de sélectionner le type de fonction d'alarme.

PV. Valeur d'entrée.

HR, LR. Echelle haute & basse pour les objets graphiques (bargraphe, tendance) liés à PV. HR et LR définissent les affichages 100 % et 0 %.

Timebase. (SECONDES/MINUTES). Base de temps des alarmes vitesse.

Setpoint. Point de consigne pour la fonction alarme d'écart.

Hyst (Hystérésis). Valeur de durée de la bande de l'hystérésis appliquée à tous les niveaux d'alarme. Hyst est exprimée en unités physiques/secondes (ou/minute) lorsqu'appliquée aux alarmes de vitesse.

Paramètre	Fonction	Unités	Etat
Type	Sélection type alarme	Menu	
PV	Variable procédé (entrée bloc)	Eng	
HR, LR	Echelle haute & basse des objets graphiques PV	Eng	
TimeBase	Sélection base de temps des alarmes vitesse	Menu	
SetPoint	Point de consigne pour les alarmes d'écart	Eng	
Hyst	Largeur de la bande hystérésis	Eng	
HiHigh	Niveau alarme très haute	Eng	
High	Niveau alarme haute	Eng	
Low	Niveau alarme basse	Eng	
LoLow	Niveau alarme très basse	Eng	
Deviatn	Ecart (PV-SP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
HiHighAl	Niveau alarme très haute	V/F	
HighAl	Alarme haute	V/F	
LowAl	Alarme basse	V/F	
LoLowAl	Niveau alarme très basse	V/F	
Combined	Alarme combinée du bloc	V/F	
Disable	Invalidation alarmes	V/F	

Table 4-6 Paramètres du bloc AN_ALARM

HiHigh, High, & Low, LoLow. Niveau des alarmes très haute, haute, basse et très basse. L'action de ces quatre paramètres polyvalents dépend du type de fonction d'alarme sélectionnée (en utilisant le paramètre *Type*) :

- **Alarmes absolues.** Les alarmes sont déclenchées si l'entrée du bloc *PV* dépasse les niveaux définis :

HiHighAl = VRAI lorsque $PV > HiHigh$

HighAl = VRAI lorsque $PV > High$

LowAl = VRAI lorsque $PV < Low$

LoLow = VRAI lorsque $PV < LoLow$

Une alarme n'est pas remise à zéro immédiatement, si *PV* revient au niveau d'alarme - *PV* doit être à l'intérieur d'une marge égale au paramètre *Hyst* avant la remise à zéro de l'alarme. L'hystérésis permet une transition nette entre une situation d'alarme et sans alarme.

- **Alarmes d'écart.** Les alarmes hautes sont déclenchées, lorsque l'écart positif dépasse les niveaux définis. Les alarmes basses sont déclenchées, lorsque l'écart négatif dépasse les niveaux définis :

HiHighAl = VRAI lorsque $(PV-SetPoint) > HiHigh$

HighAl = VRAI lorsque $(PV-SetPoint) > High$

LowAl = VRAI lorsque $(SetPoint-PV) > Low$

LoLowAl = VRAI lorsque $(SetPoint-PV) > LoLow$

L'hystérésis est appliquée aux valeurs d'écart comme dans le cas de *PV* pour les alarmes absolues.

- **Alarmes de vitesse.** Les alarmes hautes sont déclenchées, lorsque la vitesse d'augmentation de *PV* dépasse les niveaux définis. Les alarmes basses sont déclenchées, lorsque la vitesse de diminution de *PV* dépasse les niveaux :

HiHighAl = VRAI lorsque $dPV/dt > HiHigh$

HighAl = VRAI lorsque $dPV/dt > High$

LowAl = VRAI lorsque $-dPV/dt > Low$

LoLowAl = VRAI lorsque $-dPV/dt > LoLow$

La résolution de cette fonction dépend du temps d'échantillonnage de *PV*.

L'hystérésis est appliquée aux changements de dPV/dt d'une manière correspondant aux autres alarmes, mais pour les alarmes de vitesse, le paramètre *Hyst* est exprimé en unités physiques *par seconde ou minute* (définies par la base de temps (*Timebase*)).

Deviatn. Ecart (erreur) définie par *PV-SetPoint*.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **HiHighAl.** Alarme très haute.
- **HighAl.** Alarme haute.
- **LowAl.** Alarme basse.
- **LoLowAl.** Alarme très basse.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Disable (Invalidation). Cette entrée logique invalide l'indication d'alarme (les alarmes logiciel ne sont pas affectées).

DIGALARM : BLOC D'ALARME LOGIQUE

Fonction du bloc

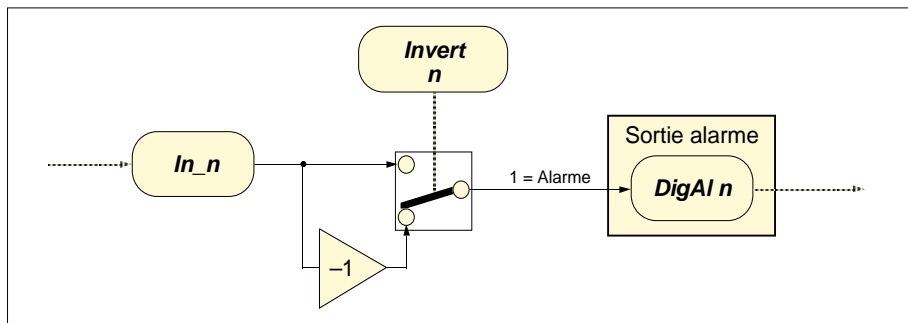


Figure 4-7 Schéma du bloc DIGALARM

Voir le schéma de la figure 4-7, qui montre uniquement l'une (la *nième*) des huit voies du bloc. Le bloc d'alarme logique permet aux signaux logiques d'un schéma de boucles d'activer les alarmes. Les états d'alarme des signaux d'entrée de déclenchement peuvent être individuellement mis sur VRAI ou FAUX (en utilisant le paramètre *Invert*), et peuvent être collectivement validés ou invalidés.

Paramètres du bloc

Les symboles utilisés dans la table 4-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Invert	Etats d'inversion des entrées alarmes	CD hex	☐➡
Bit0	Inversion signal d'entrée voie 1	V/F — 1	D
Bit1	Inversion signal d'entrée voie 2	V/F — 2	
Bit2	Inversion signal d'entrée voie 3	V/F — 4	
Bit3	Inversion signal d'entrée voie 4	V/F — 8	
Bit4	Inversion signal d'entrée voie 5	V/F — 1	C
Bit5	Inversion signal d'entrée voie 6	V/F — 2	
Bit6	Inversion signal d'entrée voie 7	V/F — 4	
Bit7	Inversion signal d'entrée voie 8	V/F — 8	
In_1 à In_8	Entrées logiques du schéma de boucles	V/F	☐➡
Alarms			☐➡ 📖 ⏪ ⏩
Software	Données corrompues/Défaut communication	V/F	
DigAl1 à DigAl8	Etats des sorties des alarmes logiques	V/F	
Combined	Alarme combinée du bloc	V/F	
Disable	Invalidation DigAl1 à DigAl8	V/F	☐➡

Table 4-7 Paramètres du bloc DIGALARM

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Invert (Inversion). Champ binaire définissant les états d’entrée nécessaires pour déclencher les alarmes. Lorsqu’un bit *Invert* est FAUX, (l’état par défaut), l’alarme correspondante est déclenchée lorsque l’entrée est VRAIE (haute). Lorsque le bit *Invert* est VRAI, une entrée FAUSSE (basse) déclenche l’alarme.

In_1 à In_8 (Entrée 1 à Entrée 8). Montre l’état réel des huit voies d’entrée (qui peuvent être liées à un schéma de boucles).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **DigAl1 à DigAl8.** Etat d’alarme de chaque entrée. Vous pouvez affecter un nom de 8 caractères (maxi.) à chacune de ces alarmes, en le saisissant dans le champ "unités" correspondant *In_1* - *In_8* au moment de la configuration. Ce nom est affiché au cours de la conduite dans la barre de messages si l’alarme est signalée, ainsi que sur la page du sommaire d’alarmes et sur les sorties papier. Le nom par défaut de l’alarme (champ de commentaire vierge) est "DigAln".

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

Disable (Invalidation). Permet d’invalider l’indication et les sorties du bloc pour les alarmes *Dig11* à *Dig18*, et également *Dig11* à *Dig18* dans la sortie alarme combinée. Disable (invalidation) n’affecte pas l’alarme logiciel ou la sortie d’alarme combinée logiciel.

LEAD_LAG : BLOC (FILTRE) AVANCE - RETARD

Fonction du bloc

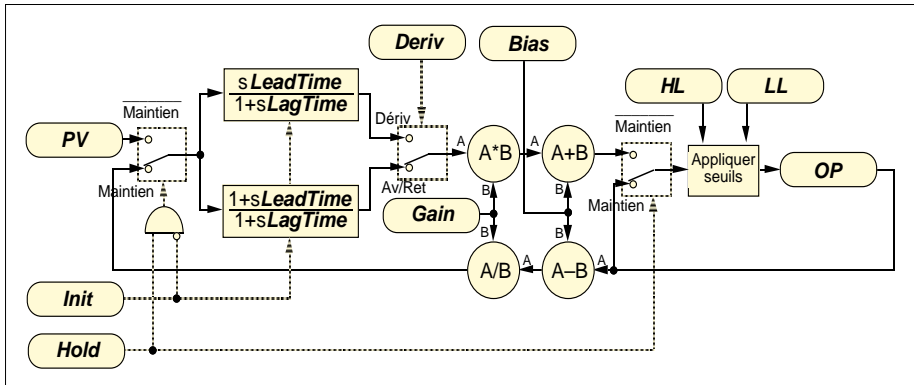


Figure 4-8 Schéma du bloc LEAD_LAG

Voir le schéma de la figure 4-8. Le bloc LEAD_LAG fournit un filtre avance-retard qui peut être utilisé pour les calculs prédictifs ou pour configurer des modèles d'installations.

Opération Lead-Lag. Dans l'opération lead-lag, (*Options/Deriv FAUX*), comme filtre avance-retard conventionnel, la sortie du bloc *OP* et l'entrée *PV* ont la relation suivante :

$$L(OP) = \text{Gain} \left(\frac{1 + s\text{LeadTime}}{1 + s\text{LagTime}} \right) L(PV) + L(\text{Bias})$$

- où
- Gain = gain global du filtre
 - Leadtime, Lagtime = constantes de temps avance, retard (sec. ou min.)
 - Bias = décalage ajouté avant la sortie
 - s = opérateur de transformation de Laplace $d()/dt$
 - L (variable) = Transformation de Laplace de la variable

Dérivée filtrée. *Options/Deriv* étant VRAI, le gain DC est nul et *OP* a alors une caractéristique de dérivée filtrée :

$$L(OP) = \text{Gain} \left(\frac{s\text{LeadTime}}{1 + s\text{LagTime}} \right) L(PV) + L(\text{Bias})$$

Paramètres du bloc

Les symboles utilisés dans la table 4-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Entrée filtre	Eng1	
OP	Sortie filtre	Eng2	
HL	Seuil haut du filtre	Eng2	
LL	Seuil bas du filtre	Eng2	
Gain	Gain du filtre	Flottant	
Bias	Tendance	Eng2	
TimeBase	Sélection base de temps	Menu	
LeadTime	Constante de temps avance	Sec/Min	
LagTime	Constante de temps retard	Sec/Min	
Options			
Init	Initialisation du filtre	V/F	
Hold	Maintien valeur OP	V/F	
Deriv	VRAI=Dérivée, FAUX=avance-retard	V/F	
		0 0 0 0 0 0 0 0	
Alarms			
Software	Données corrompues	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 4-8 Paramètres du bloc LEAD_LAG

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- PV.** Entrée du filtre.
- OP.** Sortie du filtre.
- HL, LL.** Seuils de sortie hauts & bas. Ces paramètres servent également d’échelle haute & basse pour les objets graphiques (Bargraphes, tendances) liée à *PV* ou *OP*. *HL* et *LL* définissent respectivement les affichages à 100 % et 0 %.
- Gain.** Gain du filtre défini dans les équations données dans le paragraphe *Fonction du bloc* ci-dessus.
- Bias.** Tendance. Permet de définir le décalage de la sortie ajouté à l’équation du filtre avant que le résultat ne soit transmis à la sortie du filtre *OP*.
- TimeBase (Base de temps).** (Secondes/Minutes). Sélection de la base de temps. Permet de sélectionner les unités pour les deux constantes de temps du filtre, *LeadTime* et *LagTime*.

LeadTime, LagTime (Temps d'avance, Temps de retard). Les constantes de temps du filtre avance et retard sont définies dans les équations données ci-dessus.

Options.

- **Init.** Initialisation du filtre en définissant $OP = OP_{n-1} = PV$, lorsqu'*Init* est VRAI.
- **Hold (Maintien).** Maintien de la valeur d'*OP*. Noter que *Hold* est calculé en arrière et initialise le filtre pour qu'il produise la sortie souhaitée, de sorte que l'élimination de *Hold* se fasse sans à-coups. Si *Init* et *Hold* sont actifs tous les deux, *Init* a priorité sur *Hold* dans l'initialisation du filtre.
- **Deriv.** VRAI permet de sélectionner une action de dérivée filtrée, avec un gain DC nul; FAUX permet de sélectionner une action de filtrage avance-retard conventionnelle.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

RANGE : BLOC ECHELLE

Fonction du bloc

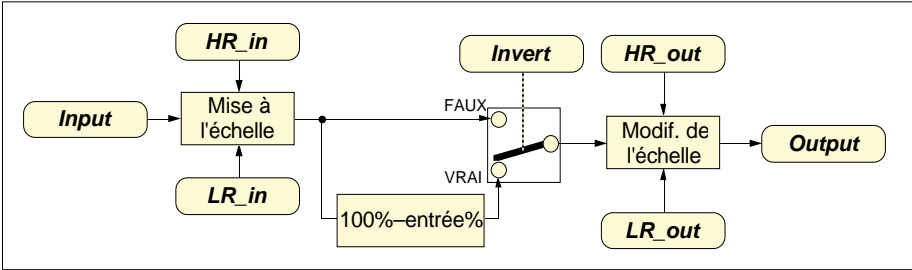


Figure 4-9 Schéma du bloc RANGE

Voir le schéma de la figure 4-9. Le bloc RANGE modifie l’échelle d’une entrée analogique.

Paramètres du bloc

Les symboles utilisés dans la table 4-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Input	Entrée dont l’échelle doit être modifiée	EngA	
HR_in	Echelle haute de l’entrée	EngA	
LR_in	Echelle basse de l’entrée	EngA	
Output	Sortie à échelle modifiée	EngB	
HR_out	Echelle haute de la sortie	EngB	
LR_out	Echelle basse de la sortie	EngB	
Options			
Invert	Inversion du sens de la sortie	V/F	
Alarms			
Software	Données corrompues	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 4-9 Paramètres du bloc RANGE

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- Input.** Entrée du bloc.

HR_in, LR_in. Echelle haute et basse de l'entrée. Permet de définir en unités physiques l'étendue de la valeur de l'entrée analogique du bloc. Noter que *LR_in* peut être supérieur à *HR_in*, si nécessaire.

Output. Sortie du bloc dont l'échelle a été modifiée.

HR_out, LR_out. Echelle haute et basse de la sortie. Permet de définir en unités physiques l'étendue de la valeur de la sortie analogique du bloc. Noter que *LR_out* peut être supérieur à *HR_out*, si nécessaire.

Options

- **Invert (Inversion).** Permet d'inverser le sens de la sortie du bloc si VRAI. Par ex., si *LR_out* = 0.000 unités et *HR_out* = 100.000 unités, une valeur *Output* (sortie) de 30.000 unités passe à 70.000 unités lorsque *Invert* est VRAI.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

FLOWCOMP : BLOC DE DEBIT CORRIGE

Fonction du bloc

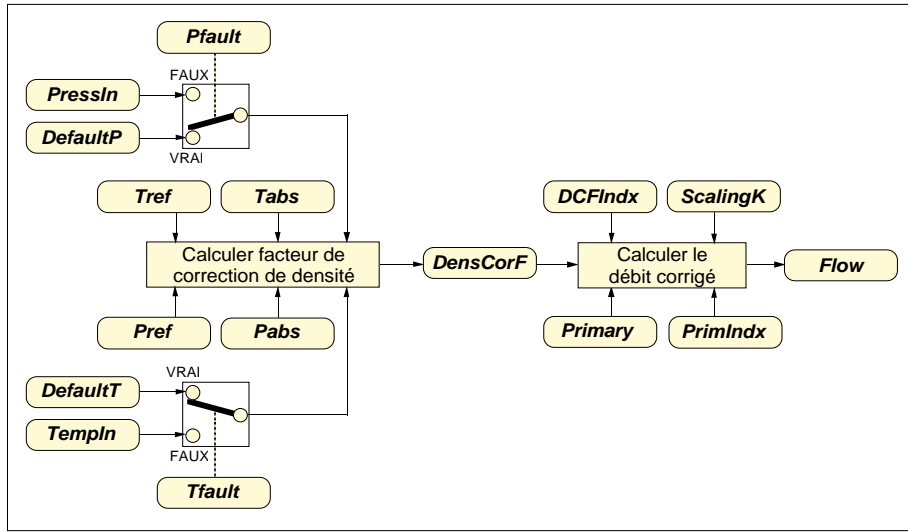


Figure 4-10 Schéma du bloc FLOWCOMP

Voir le schéma de la figure 4-10. Le bloc FLOWCOMP convertit une mesure de débit de gaz brute en un signal de sortie de débit corrigé en pression et en température.

Le bloc calcule un facteur de correction de densité à partir des mesures de pression et de température et l'utilise pour compenser la mesure de débit brute. Les valeurs de pression et de température définies par défaut par l'utilisateur sont automatiquement remplacées si les entrées logiques du bloc indiquent que ces mesures sont erronées.

Le facteur de correction de densité est calculé comme suit:

$$DensCorF = \frac{(Tref + Tabs) \times (\{PressIn \text{ OR } DefaultP\} + Pabs)}{(\{TempIn \text{ OR } DefaultT\} + Tabs) \times (Pref + Pabs)}$$

et à partir de là, le débit intégralement compensé est calculé comme suit:

$$Flow = ScalingK \times Primary^{PrimIdx} \times DensCorF^{DCFIdx}.$$

(Les codes mnémoniques sont définis dans la table 4-10 et dans le paragraphe *Menu de spécifications du bloc*).

NOTA. $DensCorF$ prend la valeur par défaut de 1, si le dénominateur de l'expression $DensCorF$ est 0.

Paramètre	Fonction	Unités	Etat
Primary	Mesure débit brut ou mesure de pression différentielle	EngA	
PressIn	Mesure de la pression	EngB	
Templn	Mesure de la température	EngC	
DefaultP	Pression par défaut	EngB	
DefaultT	Température par défaut	EngC	
Pref	Pression de référence	EngB	
Tref	Température de référence	EngC	
Pabs	Pression absolue		
Tabs	Température absolue		
ScalingK	Facteur de mise à l'échelle	EngD	
DensCorF	Facteur de correction de la densité		
Flow	Débit corrigé	EngE	
Alarms			
Software	Données corrompues	V/F	
Combined	Alarme combinée du bloc	V/F	
Faults			
Pfault	Défaut entrée PressIn	V/F	
Tfault	Défaut entrée Templn	V/F	
PrimIdx	Indice primaire	Menu	
DCFIdx	Indice du facteur de correction de densité	Menu	

Table 4-10 Paramètres du bloc FLOWCOMP

Paramètres du bloc

Les symboles utilisés dans la table 4-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Primary (Primaire). Mesure du débit brut ou pression différentielle (par ex. plaque à orifice).

PressIn. Mesure de pression du gaz, utilisée dans l’expression du facteur de correction de densité, si *Pfault* est FAUX. Il s’agit d’une surpression et les unités doivent être les mêmes que pour *Pabs*.

Templn. Mesure de température du gaz, utilisée dans l’expression du facteur de correction de densité, si *Tfault* est FAUX. Les unités doivent être les mêmes que pour *Tabs*.

DefaultP. Valeur de pression par défaut, utilisée dans l’expression du facteur de correction de densité au lieu de *PressIn*, si *Pfault* est VRAI. Les unités doivent être les mêmes que pour *Pabs*.

DefaultT. Valeur de température par défaut, utilisée dans l'expression du facteur de correction de densité au lieu de *TempIn*, si *Tfault* est VRAI. Les unités doivent être les mêmes que pour *TabS*.

Pref. Pression de référence dans les mêmes unités que *Pabs*. *Pref* est une surpression et est utilisé dans le calcul du facteur de correction de densité pour la mise à l'échelle des valeurs de pression mesurées.

Tref. Température de référence dans les mêmes unités que *TabS*. *Tref* est utilisé dans le calcul du facteur de correction de densité pour la mise à l'échelle des valeurs de température mesurées.

Pabs. Pression atmosphérique absolue par rapport à un vide. *Pabs* permet de convertir des valeurs de surpression en valeurs absolues nécessaires pour le calcul du facteur de correction de densité.

TabS. La température sur l'échelle absolue qui est équivalente à zéro degrés sur l'échelle relative correspondante. Par ex. 273.000 (K) équivaut à 0 (°C) environ. *TabS* permet de convertir les valeurs de température relatives en valeurs absolues nécessaires pour le calcul du facteur de correction de densité.

ScalingK. Un facteur de mise à l'échelle utilisé pour le calcul du débit corrigé.

DensCorF. Facteur de correction de densité, calculé d'après la description dans le paragraphe *Fonction du bloc* ci-dessus. Permet de compenser le débit de gaz mesuré en cas de variations de température et de pression par rapport à leurs valeurs de référence.

Flow (Débit). Le débit mis à l'échelle et corrigé, calculé à partir du débit brut mesuré après compensation des variations de température et de pression.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Faults (Défauts).

■ **PFault.** Une entrée VRAI dans ce paramètre permet de remplacer *PressIn* par *DefaultP* pour le calcul du facteur de correction de densité.

■ **Tfault.** Une entrée VRAI dans ce paramètre permet de remplacer *TempIn* par *DefaultT* pour le calcul du facteur de correction de densité.

PrimIdx. (1/2, 1, 3/2, 5/2). La puissance à laquelle la valeur de mesure de débit brute *Primary* est élevée dans le calcul du débit corrigé.

DCFIdx. (0, 1/2, 1 -1/2, -1). La puissance à laquelle le facteur de correction de densité *DensCorF* est élevée pour le calcul du débit corrigé.

GASCONC : BLOC DE DONNEES DE CONCENTRATION DU GAZ NATUREL

Fonction du bloc

Le bloc GASCONC contient les valeurs de concentration de gaz utilisées dans les calculs de compressibilité et supercompressibilité des mélanges de gaz naturel - en particulier les calculs AGA8 (American Gas Association Report #8 - Rapport n° 8 de l'association américaine du gaz). Les calculs proprement dits sont effectués dans un bloc associé - le bloc AGA8DATA par ex., décrit séparément.

Le bloc GASCONC enregistre les concentrations de 21 gaz maximum, exprimées en unités de pourcentage Mole. Des champs permettent d'indiquer le pourcentage Mole de gaz total (*Total*) et de définir une tolérance pour ce total (*TotTol*). Une alarme est déclenchée (*BadTotal*), si le pourcentage total des données entrées dépasse cette tolérance par rapport à 100 %. Le bloc vérifie également les données par rapport aux limites des échelles normales et étendues de l'AGA8 (Table 1 du rapport). Des alarmes sont déclenchées (*NotNorm* et *OutRange*, respectivement), si une entrée dépasse ces limites.

NOTA. Les limites de l'échelle "normale" définissent des limites de concentration pour chacun des gaz pour lesquels les calculs AGA8 produisent les résultats les plus précis. Les limites de l'échelle "étendue" définissent des limites de concentration, au-delà desquelles la précision des calculs n'est plus indiquée dans le rapport, et l'utilisation de ces valeurs n'est pas recommandée.

Le bloc maintient deux niveaux de données - les données visibles saisies dans les champs de % Mole et une copie (cachée) validée et normalisée de ces données qui sont utilisées par le ou les blocs AGA8DATA pour effectuer les calculs. Normalement, ces deux ensembles de données sont absolument identiques; mais peuvent différer si de nouvelles valeurs ont été saisies. Le paramètre - *GasData* - sert d'interface de supervision/contrôle entre les deux ensembles de données, et alerte l'opérateur s'ils diffèrent et le laisse soit utiliser les nouvelles données ou rappeler les données déjà utilisées pour écraser les champs visibles du bloc. Le nouveau pourcentage Mole total et les limites des échelles sont vérifiés à chaque demande de mise à jour des données en utilisation par l'intermédiaire de *GasData*. Si le pourcentage Mole total est hors tolérance, l'alarme est déclenchée et *GasData* signale que les données sont erronées (*Invalid*). Dans ce cas, les nouvelles données ne peuvent être utilisées pour mettre à jour les données déjà utilisées. Les calculs se poursuivent alors sur la base de données déjà utilisées, jusqu'à ce que l'opérateur saisisse des données valables ou rappelle les données en utilisation dans les champs du bloc.

Le champ *GasData* peut être configuré pour charger un ensemble de données par défaut et effectuer automatiquement et immédiatement les calculs, en cas de démarrage à froid.

Notez qu'un seul bloc GASCONC peut servir de source de données à plusieurs blocs AGA8DATA.

Paramètres du bloc

Les symboles utilisés dans la table 4-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Methane	Concentration de méthane	% Mole	
Nitrogen	Concentration d'azote	% Mole	
CrbDiOx	Concentration de dioxyde de carbone	% Mole	
Ethane	Concentration d'éthane	% Mole	
Propane	Concentration de propane	% Mole	
Water	Concentration d'eau	% Mole	
HSulphid	Concentration de sulfure d'hydrogène	% Mole	
Hydrogen	Concentration d'hydrogène	% Mole	
CrbMonOx	Concentration de monoxyde de carbone	% Mole	
Oxygen	Concentration d'oxygène	% Mole	
iButane	Concentration de butane i	% Mole	
nButane	Concentration de butane n	% Mole	
iPentane	Concentration de pentane i	% Mole	
nPentane	Concentration de pentane n	% Mole	
Alarms			
Software	Anomalie données RAM du bloc	V/F	
BadTotal	Total ≠ 100 % de plus de TotTol %	V/F	
NotNorm	Composition hors échelle normale AGA8	V/F	
OutOfRange	Composition hors échelle étendue AGA8	V/F	
Combined	Alarme combinée du bloc	V/F	
nHexane	Concentration d'hexane n	% Mole	
nHeptane	Concentration heptane n	% Mole	
nOctane	Concentration d'octane n	% Mole	
nNonane	Concentration de nonane n	% Mole	
nDecane	Concentration de décane n	% Mole	
Helium	Concentration d'hélium	% Mole	
Argon	Concentration d'argon	% Mole	
Total	Total des pourcentages Mole	% Mole	
TotTol	Tolérance autorisée dans le % total	% Mole	
GasData	Etat des données saisies/action requise (Fig 4-11)	Menu	

Table 4-11 Paramètres du bloc GASCONC

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Methane à Argon. Ces 21 champs définissent les concentrations de chaque gaz dans le mélange, en termes de pourcentage Mole sur une échelle de 0 à 100% Notez que les valeurs dans ces champs peuvent être différentes des valeurs (cachées) utilisées par le bloc AGA8DATA qui effectue les calculs. (Voir la section *Fonction du bloc* ci-dessus). Si elles sont identiques - ce qui représente l'état de fonctionnement normal du bloc - le paramètre GasData affiche la légende "No Action" (Aucune action nécessaire)

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **BadTotal (Total erroné).** Alarme déclenchée en cas de mise à jour (par GasData) et que le total de toutes les valeurs de pourcentage Mole des 21 gaz (Total) dépasse la tolérance spécifiée (TotTol) par rapport à 100 %.

L'alarme ne disparaît qu'en cas de mise à jour où l'ensemble des données saisies a une valeur totale de pourcentage Mole qui se situe dans les limites de la tolérance. Pendant que l'alarme est active, l'ensemble des nouvelles données est rejeté, et les calculs se poursuivent sur la base des anciennes valeurs.

- **NotNorm (Hors limites).** Déclenchée en cas de mise à jour où au moins une des valeurs de pourcentage Mole du gaz est en dehors des limites de l'échelle normale de l'AGA8. L'alarme ne disparaît qu'en cas de mise à jour où toutes les valeurs des compositions se situent dans les limites de l'échelle normale. Notez que malgré le déclenchement de l'alarme *NotNorm*, les données sont néanmoins acceptées et utilisées dans les calculs. L'opérateur doit décider si les données utilisées doivent être modifiées.
- **OutOfRange (Hors échelle).** Déclenchée en cas de mise à jour où au moins une des valeurs de concentration des gaz est en dehors des limites de l'échelle étendue de l'AGA8. L'alarme ne disparaît qu'en cas de mise à jour où toutes les valeurs des compositions se situent dans les limites de l'échelle étendue. Notez que malgré le déclenchement de l'alarme *OutOfRange*, les données sont néanmoins acceptées et utilisées dans les calculs. L'opérateur doit décider si les données utilisées doivent être modifiées.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.
- **Total.** Somme des 21 pourcentages Mole saisis dans les champs *Methane* à *Argon*. L'alarme *BadTotal* est déclenchée si la somme dépasse la tolérance spécifiée dans le champ *TotTol* par rapport à 100.0 %.

NOTA. Malgré la tolérance autorisée dans les données saisies, le bloc normalise automatiquement les données utilisées, afin que les chiffres de pourcentage Mole effectivement utilisés pour les calculs fassent exactement 100.0 %.

- **TotTol.** Définit une tolérance autorisée pour la valeur de *Total*. (Voir NOTA ci-dessus).
- **GasData.** (Waiting/No Action/Changed/Invalid/Update/Recall). Ce champ sert d'interface de supervision/contrôle opérateur entre les données visibles saisies dans les champs *Methane* à *Argon*, et la copie validée (cachée) de ces données utilisées par le ou les blocs AGA8DATA. (Voir la section *Fonctions du bloc*, ci-dessus). La figure 4-11 montre les rapports entre les différents éléments du menu *GasData*. Ces éléments sont les suivants:
- **Waiting (En attente).** Indique qu'il n'y a aucune donnée dans la zone de données cachée utilisée, et donc que les calculs ne peuvent être effectués par les blocs AGA8DATA. Le bloc attend que l'opérateur charge les données des champs visibles dans la zone d'utilisation avec l'élément de menu "Update" (voir ci-dessous). Cette situation peut se produire après un démarrage à froid, mais peut être évitée en enregistrant la base de données LIN avec des données de composition par défaut, et en utilisant l'option *Update* du paramètre *GasData*. Ensuite, immédiatement après un démarrage à froid, les données par défaut sont chargées automatiquement et les calculs

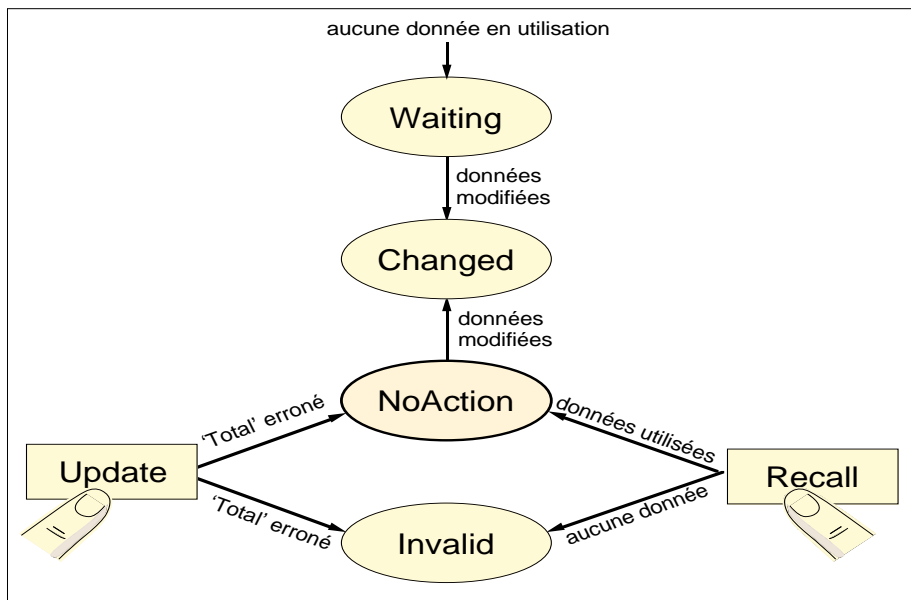


Figure 4-11 Etat des paramètres GasData

sont effectués. Notez que toute modification dans les champs de concentration de gaz fait passer l'état de *GasData* de "Waiting" à "Changed", mais les calculs ne peuvent être effectués tant que des données valables n'ont pas été copiées dans la zone d'utilisation.

- **NoAction (Aucune intervention).** Etat de fonctionnement normal du bloc (grisé sur la figure 4-11). Indique que les calculs sont effectués et que les données de composition visibles saisies et les données cachées sont les mêmes. Aucune intervention opérateur n'est nécessaire.
- **Changed (Modifiées).** Indique que les données visibles saisies dans les champs de pourcentage Mole sont différentes de celles dans les champs d'utilisation cachés, sur lesquels les calculs sont basés. "Changed" reste affiché tant que l'opérateur n'est pas intervenu.
Si les données sont en utilisation, elles sont recopiées dans les champs visibles (en utilisant la commande Recall), l'état repasse de "Change" à "NoAction". Les données modifiées peuvent également être chargées dans la zone d'utilisation grâce à la commande "Update" - voir ci-dessous. L'état adopté par GasData dépendra alors de la validité ou non des données modifiées (voir ci-dessous).
- **Invalid (Erronées).** Indique que les données que l'opérateur tente de charger dans la zone d'utilisation (avec la commande "Update") a une valeur de *Total* qui dépasse la tolérance spécifiée *TotTol* par rapport à 100.0 %. Autrement dit, les valeurs de pourcentage Mole n'atteignent pas 100 % et ne peuvent donc être utilisées. Les calculs se poursuivent donc, en utilisant les données existantes (cachées) et l'alarme

BadTotal est déclenchée. L'état reste "Invalid" tant que l'opérateur n'a pas saisi un ensemble de données valables et demandé une autre mise à jour ("Update"). Vous pouvez également utiliser la commande "Recall" pour restaurer les données valables en utilisation dans les champs visibles des blocs.

- **Update (Mise à jour).** "Update" peut être sélectionné à tout moment, quelque soit l'état affiché dans le champ *GasData*. Cette commande permet à l'opérateur d'écraser les données actives (cachées) par les nouvelles données dans les champs visibles du bloc.

La commande "Update" lance un sous-programme qui vérifie chaque valeur dans les champs de concentration des gaz par rapport aux limites des échelles normale et étendue de l'AGA8. Des alarmes sont déclenchées si une valeur est hors échelle (*NotNorm* et *OutOfRange*), ce qui n'empêche pas les nouvelles données d'être considérées comme valables. Le sous-programme vérifie également la nouvelle valeur *Total* par rapport à 100 % (plus la tolérance spécifiée) pour déterminer si le total des compositions est valable. En cas d'échec du test, l'alarme *BadTotal* est déclenchée et *GasData* adopte l'état "Invalid", et les nouvelles données sont rejetées.

Sinon, les données valables (même si hors échelle) sont chargées dans la zone d'utilisation et *GasData* repasse à l'état "NoAction", qui est l'état de fonctionnement normal. L'opérateur doit décider des mesures à prendre en cas d'alarme(s) *NotNorm* et/ou *OutOfRange*.

NOTA. Afin de s'assurer que les calculs commencent automatiquement après un démarrage à froid du système, enregistrez la base de données LIN avec les données de composition par défaut et sélectionnez l'option "Update" pour le paramètre *GasData*.

- **Recall (Rappel).** "Recall" peut être sélectionné à tout moment, quelque soit l'état affiché dans le champ *GasData*. Cette commande permet de copier les données actives (cachées) dans les champs de concentration visibles du bloc. *GasData* affiche alors "NoAction" qui est le mode de fonctionnement normal du bloc. Notez que si aucune donnée n'est active dans la zone cachée pour une raison quelconque - après un démarrage à froid, par ex. - et que vous sélectionnez "Recall", le paramètre *GasData* adopte l'état "Invalid".

AGA8DATA : BLOC DE CALCUL AGA8

Fonction du bloc

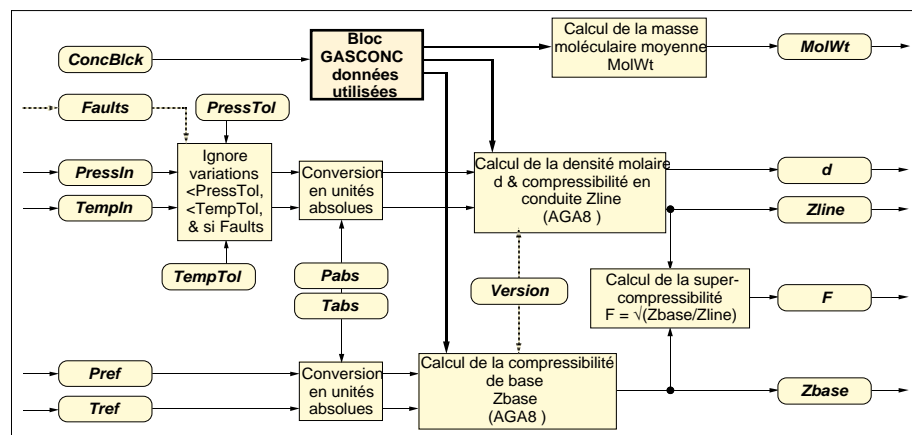


Figure 4-12 Schéma du bloc AGA8

Voir le schéma de la figure 4-12. Le bloc AGA8DATA utilise les équations AGA8 (American Gas Association Report #8 - Rapport n° 8 de l'association américaine du gaz) pour calculer la compressibilité (*Zline*) et la supercompressibilité (*F*) d'un mélange de gaz naturel à partir de la pression (*PressIn*) et de la température (*TempIn*) mesurées de la conduite. Un bloc GASCONC associé, défini dans le champ *ConcBlk*, fournit les données de composition du mélange. Le bloc calcule également la compressibilité de base du mélange (*Zbase*), la densité molaire (*d*) et la masse moléculaire moyenne (*MolWt*).

Des entrées logiques (par l'intermédiaire du champ *Faults*) permettent de connecter des sorties de défaut entrées/sorties pour signaler des mesures de pression et/ou de température erronées. Lorsque ces entrées sont définies, les lectures en question sont ignorées et les valeurs précédentes (correctes) sont utilisées pour les calculs.

Des tolérances de pression et de température (*PressTol*, *TempTol*) peuvent être définies pour réduire l'utilisation de la mémoire de l'instrument. Des changements inférieurs à ces valeurs sont traités comme du "bruit", ce qui évite la ré-évaluation inutile des éléments liés à la pression/température dans les calculs AGA8.

Des alarmes sont générées si l'accès au bloc est impossible, si les calculs sont arrêtés, si le calcul de la densité échoue ou si les compressibilités calculées sont en dehors d'une échelle acceptable.

Précision. La précision de calcul des blocs AGA8DATA/GASCONC a été testée, en entrant la pleine échelle des compositions, pressions et températures de gaz naturel figurant dans les tableaux A.3-1 et A.5-2 en Annexe A du rapport AGA8. Les résultats obtenus concordent avec ceux des tableaux du rapport avec une précision supérieure à 0,001 %, sauf pour le cas du "CO₂ haut" où la concordance est supérieure à 0,0014%.

Intervalles de mise à jour. En raison de la complexité des algorithmes utilisés, le bloc AGA8DATA ne met pas à jour les valeurs calculées des sorties à chaque itération de la base de données. Plus précisément, si les changements de pression seuls déclenchent des ré-évaluations des éléments concernés du calculs, les sorties sont mises à jour une fois pour 20 itérations de la base de données environ. Pour les changements de température, l'intervalle de mise à jour est de 30 itérations de la base de données, et pour les changements dans la composition des gaz (par l'intermédiaire du bloc GASCONC), l'intervalle est de 300 itérations de la base de données. En cas de combinaisons de changements, c'est l'intervalle de mise à jour le plus lent qui est utilisé. Si, par exemple, il se produit à la fois un changement de pression et de température, les sorties sont mises à jour une fois toutes les 30 itérations — autrement dit, toutes les 3 secondes si l'intervalle de scrutation de la base de données est de 100 ms.

NOTA. Les chiffres de compressibilité sortis par le bloc AGA8DATA peuvent être utilisés pour normaliser les débits mesurés du gaz naturel — en compensant les écarts par rapport aux lois des gaz parfaits — pour obtenir des mesures de débit précises.

Paramètres du bloc

Les symboles utilisés dans la table 4-12 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
ConcBck	Nom repère données sources bloc GASCONC	Alphanumérique	
PressIn	Pression d'entrée (conduite) (unités relatives, ex. BarG)	EngA	
TempIn	Température d'entrée (conduite) (unités relatives, ex. °C)	EngB	
Pref	Pression de référence	EngA	
Tref	Température de référence	EngB	
Pabs	Pression atmosphérique par rapport à un vide	EngA	
Tabs	Ecart de température absolu (ex. 273,15°C)	EngB	
PressTol	Tolérance pression conduite ("bruit" de la mesure)	EngA	
TempTol	Tolérance température conduite ("bruit" de la mesure)	EngB	
Alarms			
Software	Anomalie données RAM du bloc	V/F	
ConcBck	Bloc GASCONC introuvable/inaccessible	V/F	
NoCalcs	Calculs arrêtés (erreur temp & press, ou 'Waiting')	V/F	
Converge	Echec itération calcul densité molaire (d)	V/F	
ZorFOOR	Zline, Zbase, ou F hors échelle 0,5-1,5	V/F	
Combined	Alarme combinée du bloc	V/F	
Zbase	Compressibilité du mélange en conditions de référence		
Zline	Compressibilité du mélange dans la conduite		
F	Supercompressibilité ($F = \sqrt{Z_{base}/Z_{line}}$)		
d	Densité molaire du mélange (mol/dm³)		
MolWt	Masse moléculaire moyenne du mélange		
Faults			
Pfault	Défaut entrée PressIn	V/F	
Tfault	Défaut entrée TempIn	V/F	
Version	Sélection de la version AGA8 de l'algorithme (ex. 1992)	Menu	

Table 4-12 Paramètres du bloc AGA8DATA

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

ConcBlick. Permet de définir au moment de la configuration le nom de repère (champ *Block*) du bloc GASCONC qui doit être la source des données de composition du gaz pour ce calcul. *ConcBlick* ne peut être modifié en conduite. Notez que plusieurs blocs AGA8DATA peuvent être associés à un bloc GASCONC donné. Si le bloc spécifié n'existe pas, est en mémoire cache ou inaccessible, l'alarme *ConcBlick* est déclenchée.

NOTA. Les blocs AGA8DATA et GASCONC doivent être des blocs locaux dans la même base de données. Autrement dit, le système de supervision doit télécharger les mêmes données à toutes les unités qui en ont besoin.

PressIn. Mesure de la pression de la conduite du gaz débité, utilisée dans les calculs AGA8, en unités relatives (ex. BarG). Un changement de la valeur de *PressIn* (supérieur à *PressTol*) déclenche une ré-évaluation du calcul des éléments liés à la pression, qui utilisent la nouvelle valeur, à condition que l'entrée de pression soit valable (*Pfault* FAUX). Sinon, la lecture erronée est ignorée et la dernière bonne valeur de pression est utilisée à la place.

Templn. Mesure de la température de la conduite du gaz débité, utilisée dans les calculs AGA8, en unités relatives (ex. °C ou °F). Un changement de la valeur de *Templn* (supérieur à *TempTol*) déclenche une ré-évaluation du calcul des éléments liés à la température, qui utilisent la nouvelle valeur, à condition que l'entrée de température soit valable (*Tfault* FAUX). Sinon, la lecture erronée est ignorée et la dernière bonne valeur de température est utilisée à la place.

NOTA. Les unités de température utilisées doivent correspondre au système de température défini dans le bloc en-tête de la base de données par l'intermédiaire du paramètre *IP_type* parameter (c'est à dire, Imperial = °F, SI = °C). Mais, notez que la sélection des unités absolues ou relatives en utilisant *IP_type* n'a aucune importance pour le bloc AGA8DATA.

Pref. Pression de référence dans les mêmes unités que *PressIn* (ex. BarG). *Pref* définit les conditions de référence (de base) pour la mesure de la pression par rapport à laquelle la compressibilité de base est calculée.

Tref. Température de référence dans les mêmes unités que *Templn* (ex. °C). *Tref* définit les conditions de référence (de base) pour la mesure de la température, par rapport à laquelle la compressibilité de base est calculée.

Pabs. Pression atmosphérique par rapport à un vide dans les mêmes unités que *PressIn* (ex. 1,01325 BarG). *Pabs* permet de convertir les valeurs de pression du manomètre en valeurs absolues utilisées dans les calculs AGA8. La valeur par défaut du champ est celle spécifiée pour les unités SI dans le rapport AGA8.

Tabs. La température sur une échelle absolue qui est équivalente à zéro degré sur l'échelle relative correspondante. Par ex. 273,15 (K) équivaut à 0 (°C) environ. *Tabs* permet de convertir les valeurs de température relatives en valeurs absolues utilisées dans les calculs AGA8. La valeur par défaut du champ est celle spécifiée pour les unités SI dans le rapport AGA8.

PressTol. Définit la variation de l'entrée de pression du bloc (*PressIn*) avant que les éléments de calcul AGA8 liés à la pression ne soient ré-évalués. De petites variations de pression (en particulier celle en dehors de la précision du transmetteur) peuvent ainsi être ignorées, ce qui réduit l'utilisation de la mémoire de l'instrument. La pression utilisée par les calculs reste à sa dernière valeur, jusqu'à ce que *PressIn* dépasse la valeur utilisée par rapport à la tolérance *PressTol*. A ce stade — si l'entrée *Pfault* est valable — *PressIn* est la nouvelle valeur utilisée.

NOTA 1. Si vous mettez *PressTol* à zéro, toute variation de pression, quelque soit son amplitude, déclenchera une ré-évaluation. Une valeur négative de *PressTol* forcera des calculs continus, ce qui ralentit l'intervalle de mise à jour du bloc.

NOTA 2. Après un démarrage à froid, un téléchargement de la base de données ou une mise à jour des données dans le bloc GASCONC associé, des calculs complets sont effectués.

TempTol. Définit la variation de l'entrée de température du bloc (*TempIn*) avant que les éléments de calcul AGA8 liés à la température ne soient ré-évalués. De petites variations de température peuvent ainsi être ignorées, ce qui réduit l'utilisation de la mémoire de l'instrument. La température utilisée par les calculs reste à sa dernière valeur, jusqu'à ce que *TempIn* dépasse la valeur utilisée par rapport à la tolérance *TempTol*. A ce stade — si l'entrée *Tfault* est valable — *TempIn* est la nouvelle valeur utilisée. (Zéro ou une valeur négative pour *TempTol* aura le même que pour *PressTol*. Voir les NOTAS ci-dessus).

Alarms. Voir la description générale du champ Alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **ConcBlck.** Déclenchée si le bloc spécifié dans le champ *ConcBlck* est introuvable, en mémoire cache ou inaccessible.
- **NoCalcs.** Alarme déclenchée si les calculs AGA8 ne sont pas effectués à cause de problèmes aux niveaux des entrées de température et de pression (*Pfault* et *Tfault* VRAI) ou en raison du chargement de données de composition erronées dans le bloc après un démarrage à froid (le champ *GasData* du bloc GASCONC associé indique 'Waiting').
- **Converge.** Déclenchée si le processus d'itération pour calculer *d* (densité molaire) échoue. Cette alarme disparaît automatiquement, lorsque l'itération aboutit.
- **ZorFOOR.** Compressibilités calculées hors échelle. Déclenchée si *Zline*, *Zbase* ou *F* sont en dehors de l'échelle 0,5 à 1,5. Notez que les valeurs ne sont pas limitées, mais seulement signalées en alarme.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Zbase. Compressibilité de base du gaz. La compressibilité du gaz dans des conditions de référence de température (*Tref*) et de pression (*Pref*). Ce champ est mis à jour par l'instrument à chaque variation des données de concentration ou de *Tref* ou *Pref*.

Zline. Compressibilité du gaz dans la conduite. La compressibilité du gaz dans des conditions de température (*TempIn*) et de pression (*PressIn*) dans la conduite. Ce champ est mis à jour par l'instrument à la fin de chaque cycle de calculs.

F. Supercompressibilité du gaz. $F=\sqrt{[Zbase/Zline]}$. Ce champ est mis à jour à la fin de chaque cycle de calculs.

d. Densité molaire du mélange de gaz. Ce champ est mis à jour à la fin de chaque cycle de calculs. Les unités sont mol/dm³. Notez que la densité du gaz est en unités de kg/m³ (équivalent à g/dm³) est obtenue par $d \times MolWt$.

MolWt. Masse moléculaire moyenne du mélange de gaz — valeur moyenne pondérée. Ce champ est mis à jour par l'instrument à chaque fois que les données dans le bloc GASCONC associé changent.

Faults. Entrée connectée à partir de signaux logiques représentant l'état des entrées/sorties de pression et de température.

■ **Pfault.** L'entrée VRAIE du paramètre indique un défaut, dans ce cas, la dernière bonne lecture de pression est utilisée comme valeur pour les calculs AGA8.

■ **Tfault.** L'entrée VRAIE du paramètre indique un défaut, dans ce cas, la dernière bonne lecture de température est utilisée comme valeur pour les calculs AGA8.

NOTA. Si à la fois *Pfault* et *Tfault* sont VRAIS, *Zline*, *Zbase*, *F* et *d* maintiennent leurs dernières valeurs et l'alarme *NoCalcs* est déclenchée.

Version. (1992) Permet de sélectionner la version du rapport AGA8 à utiliser comme source de l'algorithme de mise à jour du bloc. (*La version 1992 du rapport, réimprimé en juillet 1994 est la seule actuellement mise en oeuvre*). Notez que ce bloc met en oeuvre la *méthode de caractérisation détaillée* définie dans le rapport.

[Page laissée intentionnellement blanche]

Chapitre 5 BLOCS DE FONCTION DE REGULATION

PID : BLOC DE REGULATION

Fonction du bloc

Voir la figure 5-1. Le bloc PID effectue deux fonctions principales. Il génère un point de consigne interne résultant SP (partie supérieure du schéma) et utilise ce point de consigne avec une entrée de la variable procédé PV pour produire une sortie de régulation PID OP (partie inférieure du schéma).

Les principales fonctions du bloc PID sont les suivantes :

Régulation non-interactive à 3 termes

P, P+I, P+D, PID et régulation tout ou rien avec hystérésis

7 modes de fonctionnement

Verrouillage pour un transfert sans à-coups en régulation de type cascade

Tendance variable (remise à zéro manuelle)

Option de gestion du gain

Ajustage du point de consigne

Désaturation du terme de l'intégrale

Equilibrage du terme de l'intégrale

Terme de la dérivée filtré (réduit la sensibilité au bruit)

Alarmes absolues et d'écart

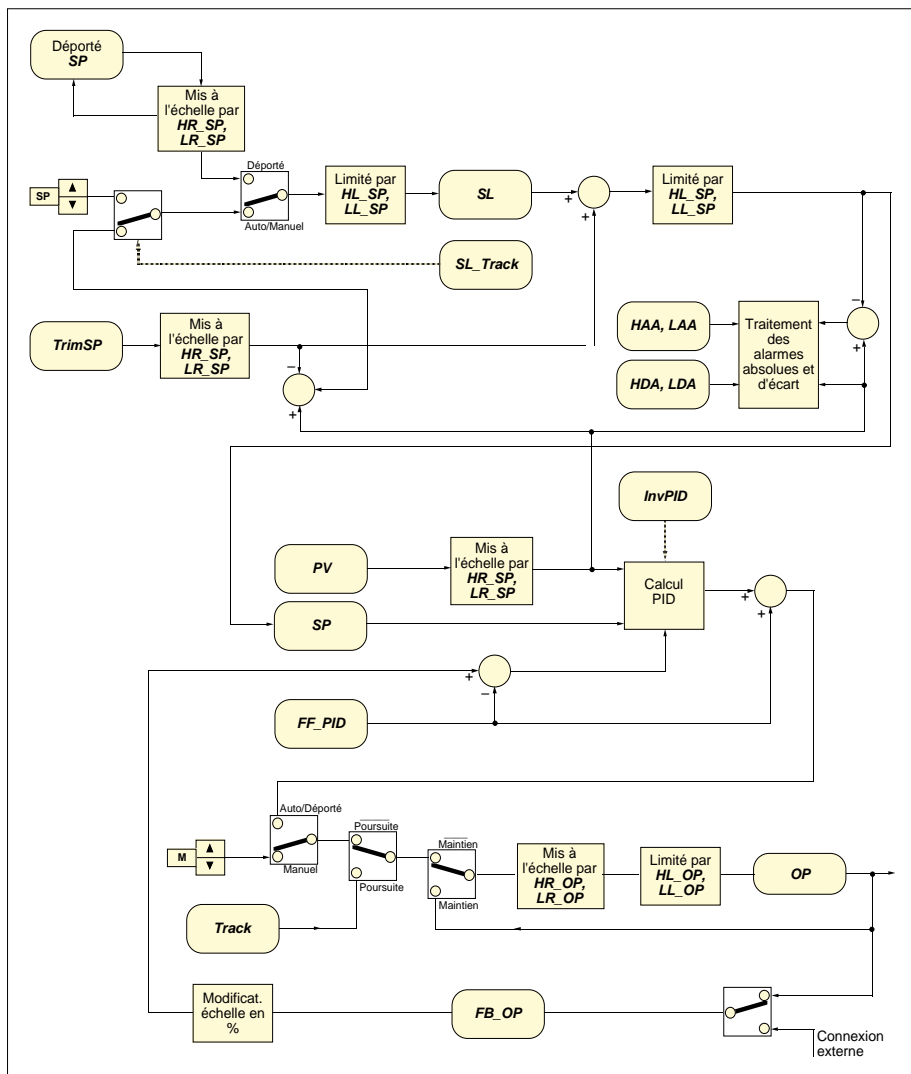


Figure 5-1 Schéma du bloc PID

Régulation tout ou rien

Une régulation tout ou rien simple est mise en oeuvre en utilisant le bloc PID et en mettant la bande proportionnelle XP à zéro, ce qui modifie la fonction de TI qui définit alors la bande morte (hystérésis). Il est à noter que la fonction de bande morte fonctionne en dessous du point de consigne résultant (voir figure 5-2).

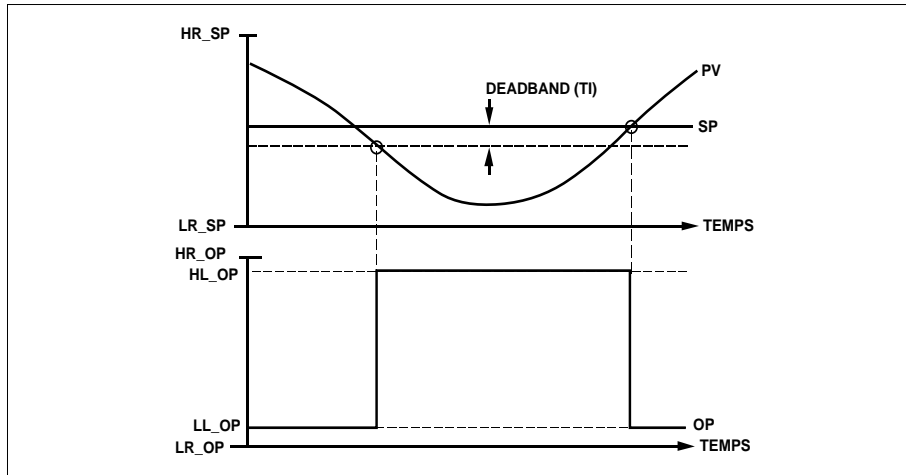


Figure 5-2 Régulation Tout ou Rien du bloc PID

Définition du point de consigne

Le point de consigne résultant SP est en définitive la valeur utilisée pour contrôler la variable procédé dans une boucle. SP est déduit d'un paramètre appelé le point de consigne local (SL) qui est le point de consigne que l'opérateur peut modifier (sauf en mode d'exploitation déporté).

En mode automatique, le point de consigne local peut toujours être ajusté, à moins que des mesures ne soient prises pour l'empêcher.

En mode déporté, le point de consigne local est contrôlé par *RemoteSP* (SP déporté), ce qui permet un transfert sans à-coups du mode déporté à tout autre mode d'exploitation. Toutefois, la procédure de transfert en mode déporté ne se fait pas nécessairement sans à-coups, et nécessite souvent une protection, ce qui implique l'utilisation des modes poursuite dans les blocs précédents, cette pratique est devenue standard dans les systèmes à cascade.

Dans les modes manuel, poursuite et maintien, il y a deux alternatives pour le contrôle du point de consigne local définies par SL_Track (point de consigne local_poursuite) dans le champ d'options.

Si $SL_Track = VRAI$, le point de consigne suit automatiquement la variable procédé, ce qui simplifie le fonctionnement de la plupart des boucles de régulation, et évite la possibilité de perturbations du point de consigne par suite d'un fonctionnement négligent. La deuxième alternative est de permettre à l'opérateur de garder le contrôle du point de consigne local dans ces modes d'exploitation, ce qui est réalisé en entrant $SL_Track = FALSE$.

L'option *TrimSp* permet d'ajouter un décalage au point de consigne local. Cette fonction est souvent utilisée pour optimiser les installations et contrôler la production.

Il est à noter que les limites du point de consigne définies par les paramètres HL_SP et LL_SP agissent indépendamment sur le point de consigne local (SL) et sur le point de consigne résultant (SP) dans tous les modes d'exploitation.

Caractéristiques des modes d'exploitation

Les blocs de régulation dans le T1000 et T100 peuvent fonctionner dans l'un des différents *modes de régulation*, chacun ayant son propre mode de contrôle de la boucle. Les caractéristiques de fonctionnement de ces modes sont décrites ci-dessous. Se reporter aux informations sur la sélection des modes, l'interaction et les priorités, l'algorithme de régulation à 3 termes PID du T1000/T100, et les techniques d'équilibrage et de désaturation de l'intégrale au chapitre 15, *Modes de fonctionnement de la boucle de régulation*.

Mode Maintien (Hold)

La valeur de sortie du régulateur (OP) est bloquée en mode maintien. Le point de consigne local suit PV ou reste constant suivant le statut de SL_Track .

Mode Poursuite (Track)

La valeur de sortie du régulateur (OP) est contrôlée par $Track$. Le point de consigne local suit PV ou reste constant suivant le statut de SL_Track .

Mode Manuel (Manual)

La sortie du régulateur (OP) a le statut lecture/écriture. Le point de consigne local (SL) suit PV ou reste constant suivant le statut de SL_Track .

Mode Automatique (AUTO)

OP est contrôlé comme une fonction de PV et SP . SL a le statut de lecture/écriture.

Mode déporté (Remote)

OP est contrôlé comme une fonction de PV et SP . SL est contrôlé par $RemoteSp$ qui a le statut de lecture/écriture. Le mode déporté est un mode de fonctionnement conditionnel. *EnaRem* (validation mode déporté) doit être $VRAI$ pour permettre le fonctionnement en mode déporté. Si *EnaRem* est $FAUX$, la boucle de régulation passera par défaut en mode automatique forcé, si le mode déporté est sélectionné.

Mode automatique forcé (F_AUTO)

Ce mode est sélectionné par défaut, lorsque le mode déporté est sélectionné sans être validé. Le mode automatique forcé a les mêmes caractéristiques de fonctionnement que le mode automatique, si ce n'est que la boucle de régulation passe en mode déporté dès que *EnaRem* est VRAI.

Mode manuel forcé (F_MAN)

Les caractéristiques de fonctionnement de ce mode sont similaires au mode manuel. Ce mode est normalement utilisé pour indiquer une condition de défaut qui exclut le fonctionnement en mode automatique ou déporté, c'est à dire perte de l'entrée *PV*, données corrompues, etc.

Le mode manuel forcé ne peut être sélectionné qu'à partir du champ *ModeSel*, et son application est donc définie par l'utilisateur.

NOTA. Les limites du point de consigne et les limites de sortie sont actives dans tous les modes de fonctionnement.

Paramètres du bloc

Les symboles utilisés dans la table 5-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block ,Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Mode. (HOLD/TRACK/MANUAL/AUTO/REMOTE/F_MAN/F_AUTO - MAINTENIR/POURSUITE/MANUEL/AUTO/DEPORTE/MANUEL FORCE/AUTO FORCE). Mode de fonctionnement en cours. Peut être utilisé pour sélectionner les modes manuel, automatique et déporté suivant le statut du paramètre *SelMode*. Pour les modes poursuite et maintien, *Mode* sélectionne le mode de reprise (Fallback).

Fallback (Reprise). (MANUAL/AUTO/REMOTE/F_AUTO - MANUEL/AUTO/DEPORTE/AUTO FORCE). Indique le mode de fonctionnement adopté si aucun mode n'a été sélectionné par le paramètre *SelMode*. (Voir chapitre 15).

PV. Variable procédé.

SP. Point de consigne résultant.

OP. Sortie du régulateur. Ce champ a un statut de lecture écriture dans les modes MANUEL et F_MAN, et un statut de lecture uniquement dans tous les autres modes.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
FallBack	Mode de fonctionnement supprimé	Menu	
PV	Variable procédé	Eng	
SP	Point de consigne résultant	Eng	
OP	Sortie régulateur	%	
SL	Point de consigne local	Eng	
TrimSP	Ajustage point de consigne local	Eng	
RemoteSP	Point de consigne déporté	Eng	
Track	Entrée poursuite	%	
HR_SP	Point de consigne/Echelle haute variable procédé	Eng	
LR_SP	Point de consigne/Echelle basse variable procédé	Eng	
HL_SP	Limite haute point de consigne	Eng	
LL_SP	Limite basse point de consigne	Eng	
HR_OP	Echelle haute sortie	%	
LR_OP	Echelle basse sortie	%	
HL_OP	Limite haute sortie	%	
LL_OP	Limite basse sortie	%	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
HighAbs	Alarme haute absolue	V/F	
LowAbs	Alarme basse absolue	V/F	
HighDev	Alarme d'écart haute	V/F	
LowDev	Alarme d'écart basse	V/F	
Combined	Alarme combinée du bloc	V/F	
HAA	Limite alarme absolue haute	Eng	
LAA	Limite alarme absolue basse	Eng	
HDA	Limite alarme d'écart haute	Eng	
DA	Limite alarme d'écart basse	Eng	
TimeBase	Unités de temps TI, TD (sec/min)	Menu	
XP	Bande proportionnelle	%	
TI	Constante de temps de l'intégrale/Band morte %	%	
TD	Constante de temps de la dérivée	%	
Options	Options du régulateur	Champ binaire	
InvPID	Action à 3 termes	V/F	
SL_Track	SL poursuit PV	V/F	
IntBaISL	Equilibrage de l'intégrale après modif. SL	V/F	
IntBaIXP	Equilibrage de l'intégrale après modif. XP	V/F	
IntBaI	Forçage équilibrage intégrale	V/F	
MSelFunc	Seul un bit de mode de ModeSel peut être actif	V/F	

suite...

...suite

Paramètre	Fonction	Unités	Etat
LSelMode	Sélection mode	Champ binaire	
SelHold	Sélection mode maintien	V/F	
SelTrack	Sélection mode poursuite	V/F	
SelRem	Sélection mode déporté	V/F	
EnaRem	Validation mode déporté	V/F	
SelAuto	Sélection mode automatique	V/F	
SelMan	Sélection mode manuel	V/F	
	(Non utilisé)		
SelfMan	Sélection mode manuel forcé	V/F	
ModeSel	Modes sélectionnés	Champ binaire	
EnaRem	Validation mode déporté (esclave)	V/F	
HoldSel	Mode maintien sélectionné	V/F	
TrackSel	Mode poursuite sélectionné	V/F	
RemSel	Mode déporté sélectionné	V/F	
AutoSel	Mode automatique sélectionné	V/F	
ManSel	Mode manuel sélectionné	V/F	
FAutoSel	Mode automatique forcé sélectionné	V/F	
FmanSel	Mode manuel forcé sélectionné	V/F	
ModeAct	Mode actif	Champ binaire	
NotRem	Non déporté actif	V/F	
HoldAct	Mode maintien actif	V/F	
TrackAct	Mode poursuite actif	V/F	
RemAct	Mode déporté actif	V/F	
AutoAct	Mode automatique actif	V/F	
ManAct	Mode manuel actif	V/F	
FAutoAct	Mode automatique forcé actif	V/F	
FManAct	Mode manuel forcé actif	V/F	
FF_PID	Tendance	%	
FB_OP	Contre-réaction sortie	%	

Table 5-1 Paramètres du bloc PID

SL. Point de consigne local. Cette valeur représente le point de consigne opérateur pour la boucle de régulation. Un ajustage de point de consigne est ajouté à *SL* pour produire *SP* qui est utilisé comme la valeur souhaitée pour la régulation. *SL* a un statut de lecture/écriture en mode automatique, et un statut de lecture uniquement en mode déporté où *SL* suit *RemoteSP*. Le statut de lecture/écriture de *SL* dans les modes manuel, poursuite et maintien est conditionnel, et configuré dans *SL_Track*. Les limites du point de consigne sont utilisées dans tous les modes de fonctionnement et agissent indépendamment sur les valeurs *SL* et *SP*.

TrimSP. Ajustage du point de consigne.

RemoteSP. Point de consigne déporté. Contrôle le point de consigne en mode déporté uniquement.

Track. Contrôle *OP* en mode poursuite.

HR_SP, LR_SP. Echelles hautes & basses du point de consigne et de la variable procédé en unités physiques.

HL_SP, LL_SP. Limites hautes & basses. Les limites du point de consigne sont actives dans tous les modes de fonctionnement et agissent indépendamment sur *SL* et *SP*.

HR_OP, LR_OP. Echelles hautes & basses de la sortie en unités physiques.

HL_OP, LL_OP. Echelles hautes & basses de la sortie en unités physiques. Les limites de sortie sont actives dans tous les modes de fonctionnement.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **HighAbs, LowAbs.** Alarmes absolues hautes et basses définies par les paramètres *HAA* et *LAA*, avec une bande d'hystérésis de 0,5 % sur chaque seuil.
- **HighDev, LowDev.** Alarmes d'écart hautes et basses définies par les paramètres *HDA* et *LDA*, avec une bande d'hystérésis de 0,5 % sur chaque seuil. Une alarme d'écart haute est déclenchée si $PV-SP > HDA$ et une alarme d'écart basse si $SP-PV > LDA$.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

HAA, LAA. Limites d'alarmes absolues hautes & basses en unités physiques.

HDA, LDA. Limites d'alarmes d'écart hautes & basses en unités physiques.

Timebase (sec./min.). Permet de spécifier l'unité de temps pour *TI* et *TD*.

XP. Bande proportionnelle (%). L'option de régulation tout ou rien avec hystérésis est sélectionnée, lorsque $XP = 0$. La fonction de *TI* est modifiée et permet de spécifier la bande morte, lorsque la régulation tout ou rien est sélectionnée.

TI. Fonction double : constante de temps de l'intégrale, secondes/minutes ($XP = 0$) ou bande morte % ($XP = 0$). Si $XP = 0$, *TI* spécifie la bande morte comme le pourcentage de l'échelle du point de consigne. La fonction bande morte fonctionne en-dessous du point de consigne résultant.

TD. Constante de temps de la dérivée (secondes/minutes).

Options

- **InvPID (Inversion PID).** Sélection de l'action de régulation à 3 termes. Le terme d'erreur dans les régulateurs Eurotherm Systèmes est défini par $ER = PV - SP$. Lorsque *InvPID* = FAUX, une augmentation de *ER* produit une diminution correspondante de la sortie à 3 termes en compensation. Lorsque *InvPID* = VRAI, l'augmentation de *ER* produit une augmentation correspondante de la sortie à 3 termes en compensation.

- **SL_Track (Sélection poursuite).** Fait que le point de consigne local suit la variable procédé, si les modes maintien, poursuite ou manuel sont actifs. Il faut noter que le point de consigne local est toujours soumis à l'action des limites du point de consigne.
- **IntBalSL.** Affecte le fonctionnement du régulateur en mode automatique. Lorsque *IntBalSL* = VRAI, toute modification du point de consigne local déclenche automatiquement une procédure d'équilibrage du terme de l'intégrale sur la sortie à 3 termes. (Voir les détails de l'équilibrage du terme de l'intégrale au chapitre 15, *Modes de fonctionnement de la boucle de régulation*). Lorsque *IntBalSL* = FAUX, une modification de *SL* produit une réponse 3 termes normale.
- **IntBalXP.** Permet le déclenchement automatique de l'équilibrage du terme de l'intégrale en cas de modification de la bande proportionnelle (*XP*). Normalement sur VRAI, *IntBalXP* devrait être invalidé dans le cas d'applications de gestion de gain, dans la mesure où il y aurait des interférences par rapport à la réaction correcte de la régulation.
- **IntBal.** Un front montant dans ce bit force l'exécution d'un équilibrage du terme de l'intégrale, quelles que soient les entrées des autres *Options*.
- **MSelfFunc.** Si VRAI, ce bit modifie l'action du paramètre *ModeSel*, de sorte qu'un seul des bits de mode à exclusion mutuelle soit actif à la fois. VRAI par défaut à la création du bloc.

SelMode (Sélection du mode). Ce champ binaire permet de sélectionner les modes du régulateur par l'intermédiaire des sorties logiques du schéma. Les informations suivantes sont données en plus des informations de la table 5-1.

- **SelRem (Sélection mode déporté).** Permet de sélectionner le mode déporté. Le mode déporté est un mode de fonctionnement conditionnel qui doit être validé. *EnaRem* doit être VRAI pour permettre le fonctionnement en mode déporté, sinon le mode automatique forcé est sélectionné par défaut.
- **EnaRem (Validation mode déporté).** Permet de valider le mode déporté. VRAI permet le fonctionnement en mode déporté. FAUX provoque le transfert de la régulation en mode automatique forcé, lorsque le mode déporté est sélectionné.

ModeSel (Modes sélectionnés). Ce champ binaire indique que les modes ont été *sélectionnés*. Le mode *actif* en cours est le mode sélectionné avec la priorité la plus haute. Les informations suivantes sont données en plus des informations de la table 5-1.

- **EnaRem (Validation mode déporté).** Liaison de sortie à partir du régulateur maître vers l'esclave pour les applications de régulation en cascade. Cette liaison est FAUSSE, lorsque le mode maintien ou manuel est sélectionné (et pas nécessairement actif) dans le régulateur maître. Cette sortie devrait être liée à l'entrée *SelMode.EnaRem* sur un régulateur esclave.

ModeAct (Mode actif). Ce champ binaire montre le mode de régulation actif en cours. Les informations suivantes sont données en plus des informations de la table 5-1.

- **NotRem (Mode déporté non actif).** La liaison de sortie du régulateur esclave au maître dans les applications de régulation à cascade. Cette liaison est VRAIE, lorsque le régulateur esclave ne fonctionne pas en mode déporté. Cette sortie devrait être liée en mode *SelMode.SelTrack* au régulateur maître.

FF_PID. Tendance. Permet d'ajouter un décalage à la sortie à 3 termes, ce qui est quelquefois appelé "remise à zéro manuelle".

FB_OP. Valeur de contre-réaction. Utilisée dans la désaturation de l'intégrale pour indiquer que la sortie régulateur a été limitée par l'action des limites de sortie. (Voir les détails sur la désaturation de l'intégrale au chapitre 15).

ANMS : BLOC STATION MANUELLE ANALOGIQUE

Fonction du bloc

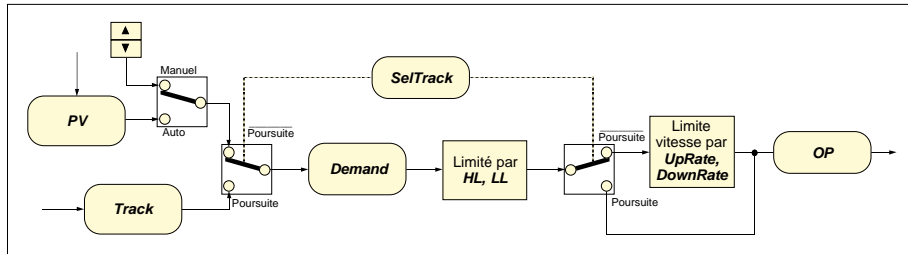


Figure 5-3 Schéma du bloc ANMS

Voir le schéma de la figure 5-3. Le bloc de station manuelle analogique permet à l'opérateur de dialoguer avec une variable des installations au cours de la conduite. La station dispose de limites de sortie, de limites de vitesse asymétriques et du contrôle Auto/Manuel/Poursuite avec reprise en automatique.

En mode poursuite (la priorité la plus haute), *OP* est déduit du paramètre *Poursuite*. En mode automatique, *OP* suit *PV*, et en mode manuel, *OP* peut être ajusté en utilisant le paramètre *Demand*. Il faut noter qu'en mode poursuite, *OP* est soumis aux limites de sortie, mais pas aux limites de vitesse. En mode Auto et Manuel, les limites de sorties et les limites de vitesse (si activées) sont appliquées.

Paramètres du bloc

Les symboles utilisés dans la table 5-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Mode. (AUTO/MANUAL/TRACK - AUTO/MANUELPOURSUITE). Mode de fonctionnement en cours. Peut être utilisé pour sélectionner les modes manuel et automatique. (*SelTrack* étant VRAI, mode sélectionne le mode de reprise).

Fallback (Reprise). (AUTO/MANUAL - AUTO/MANUEL). Indique le mode de fonctionnement suivant (supprimé).

PV. Variable procédé (valeur de l'entrée).

Track (Poursuite). Contrôle *OP* en mode poursuite.

Demand (Demande). Contrôle *OP* en mode manuel.

OP. Valeur de sortie de la station.

















Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Fallback	Mode de fonctionnement supprimé	Menu	
PV	Variable procédé	Eng	
Track	Contrôle OP en mode poursuite	Eng	
Demand	Contrôle OP en mode manuel	Eng	
OP	Sortie analogique	Eng	 
HR, LR	Echelle haute & basse des objets graphiques PV, OP	Eng	 
HL, LL	Limites hautes & basses de OP	Eng	 
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
UpRate	Limite la vitesse d'augmentation de OP	Eng/Base de temps	
DownRate	Limite la vitesse de diminution de OP	Eng/Base de temps	
TimeBase	Sélection unités de temps (limites de vitesse)	Menu	
SelRate	Active les limites de vitesse	V/F	
SelTrack	Sélection mode poursuite	V/F	
NotAuto	Pas mode Auto	V/F	 

Table 5-2 Paramètres du bloc ANMS

HR, LR. Echelle haute & basse pour les objets graphiques (bargraphe, tendance) liés à *PV* et *OP*. *HR* et *LR* définissent les affichages 100 % et 0 %.

HL, LL. Limites hautes & basses de la sortie .

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

UpRate. Spécifie la vitesse d'augmentation maximale de *OP* (c'est à dire la "limite de vitesse") en unités physiques/temps. Les unités de temps sont spécifiées par le paramètre *TimeBase*.

DownRate. Spécifie la vitesse de diminution maximale de *OP* (c'est à dire la "limite de vitesse") en unités physiques/temps

TimeBase (Base de temps). (Sec./Min./Heures/Jours). Permet de sélectionner les unités de temps pour les fonctions de limites de vitesse *UpRate* et *DownRate*.

SelRate. Active les fonctions de limites de vitesse *UpRate* et *DownRate*.

SelTrack. Permet de sélectionner le mode poursuite. *SelTrack* étant VRAI, le paramètre *Mode* définit le mode de reprise (supprimé).

NotAuto. Indique que la station ne fonctionne pas en mode Auto.

DGMS : BLOC STATION MANUELLE LOGIQUE

Fonction du bloc

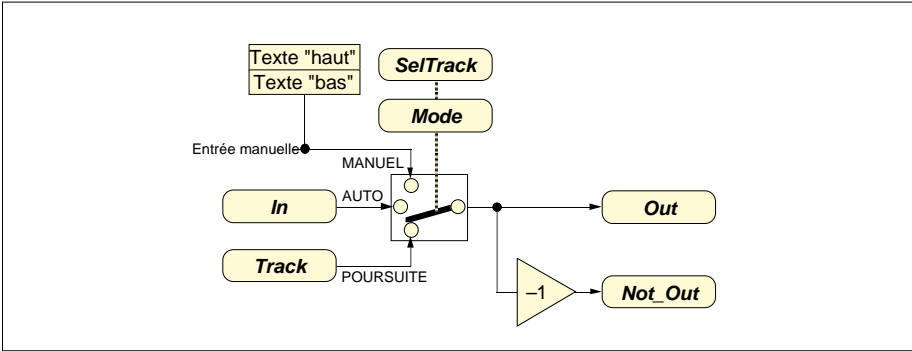


Figure 5-4 Schéma du bloc DGMS

Voir le schéma de la figure 5-4. Le bloc de station manuelle logique permet à l’opérateur de commuter un signal logique d’un schéma de boucles au cours de la conduite. En mode Auto, le paramètre *Out* suit l’état du paramètre *In*. En mode manuel, *Out* peut être défini indépendamment (en utilisant le menu de spécifications ou un objet graphique DGMAN_ST). En mode poursuite, *Out* suit le paramètre *Track* (poursuite).

Paramètres du bloc

Les symboles utilisés dans la table 5-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
In	Entrée	V/F	
Out	Sortie	V/F	
Not_Out	Complément de Out (= \overline{Out})	V/F	
Track	Valeur de poursuite	V/F	
SelTrack	VRAI= mode poursuite sélectionné	V/F	
NotAuto	VRAI= mode n'est pas automatique	V/F	
Status	Informations d'état du mode	(ABC)D hex	
FallMan	Track: F = Auto, T = Man; sinon reprise indiquée	V/F 1 V/F 2 V/F 4 V/F 8	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 5-3 Paramètres du bloc DGMS

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUAL/TRACK - AUTO/MANUEL/POURSUITE). Mode de fonctionnement en cours. Peut être utilisé pour sélectionner les modes manuel et automatique. TRACK* doit être sélectionné par *SelTrack*.*

In. Etat de l'entrée. Contrôle la sortie en mode auto.

Out. Etat de la sortie. Contrôle la sortie en mode manuel.

Not_Out. Forme complémentaire de la sortie (*Out*).

Track (Poursuite). Contrôle *Out* en mode poursuite.

SelTrack. VRAI permet de sélectionner le mode poursuite. Si *SelTrack* est VRAI, le paramètre *Mode* définit le mode de reprise (supprimé).

NotAuto. Indique que le poste ne fonctionne pas en mode automatique.

Status. Champ d'état à 16 bits, seul le bit 0 est utilisé.

■ **FallMan.** Lorsque le bloc n'est pas en mode Track (Poursuite), *FallMan* est FAUX, en mode Auto, VRAI si en mode manuel. Lorsque le bloc est en mode poursuite, *FallMan* indique le mode de reprise du bloc lorsque la poursuite est supprimée — FAUX signifie Auto, VRAI manuel.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

*NOTA. La fonction poursuite n'est pas disponible dans les premières versions du bloc DGMS.

SIM : BLOC DE SIMULATION

Fonction du bloc

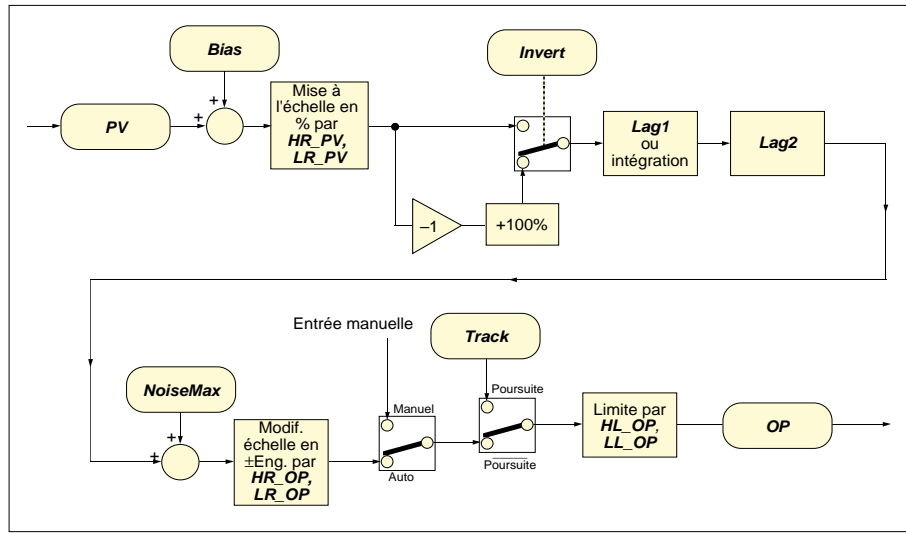


Figure 5-5 Schéma du bloc SIM

Voir le schéma de la figure 5-5. Le bloc de simulation permet de simuler les caractéristiques des installations et peut être utilisé pour vérifier le fonctionnement d'un schéma de boucles hors ligne.

Le bloc SIM dispose de deux fonctions de retard du premier ordre avec un bruit pseudo-aléatoire. Une option supplémentaire permet de substituer une fonction intégrateur à la première fonction de retard pour la simulation de la capacité. Le bloc comprend également le décalage **PV**, l'inversion, l'initialisation et le contrôle Auto/Manuel/Poursuite avec reprise en automatique.

Le mode poursuite, qui a la priorité la plus haute, est sélectionné avec le paramètre **SelTrack**. En mode poursuite, **OP** est contrôlé par la *poursuite*. En mode automatique, **OP** est calculé à partir de **PV** d'après les caractéristiques du bloc. En mode manuel, **OP** peut être ajusté indépendamment. Il faut noter que les limites de sortie sont actives dans *tous* les modes de fonctionnement.

Paramètres du bloc

Les symboles utilisés dans la table 5-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.


















Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
Fallback	Mode de fonctionnement supprimé	Menu	
PV	Variable procédé	Eng1	
Bias	Décalage PV	Eng1	
Track	Contrôle OP en mode poursuite	Eng1	
HR_PV	Echelle haute entrée (& Echelle objets graphiques PV)	Eng1	
LR_PV	Echelle basse entrée (& Echelle objets graphiques PV)	Eng1	
OP	Sortie du bloc	Eng2	
HR_OP	Echelle haute sortie (& Echelle objets graphiques OP)	Eng2	
LR_OP	Echelle basse sortie (& Echelle objets graphiques OP)	Eng2	
HL_OP	Limite haute sortie	Eng2	
LL_OP	Limite basse sortie	Eng2	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
NoiseMax	Amplitude maxi. bruit pseudo-aléatoire	Eng2	
Lag1	Constante de temps 1er filtre (ou intégrateur)	Temps	
Lag2	Constante de temps 2ème filtre	Temps	
TimeBase	Sélection unités de temps Lag1 & Lag2	Menu	
Intgr	Sélection fonction Lag1 (Intégrateur/Filtre)	V/F	
Invert	Sélection inversion sens PV	V/F	
Init	Initialisation filtres du bloc	V/F	 
SelTrack	Sélection mode poursuite	V/F	

Table 5-4 Paramètres du bloc SIM

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUAL/TRACK - AUTO/MANUEL/POURSUITE). Mode de fonctionnement en cours.

Fallback (Reprise). (AUTO/MANUAL - AUTO/MANUEL). Indique le mode de fonctionnement suivant (supprimé).

PV. Valeur de l’entrée.

Bias (Décalage). Décalage de l’entrée

Track (Valeur de poursuite). Contrôle *OP* en mode poursuite.

HR_PV, LR_PV. Echelle haute et basse de l'entrée. Egalement, échelle haute & basse pour les objets graphiques liés à *PV* (bargraphes, tendance). *HR_PV* et *LR_PV* définissent les affichages 100 % et 0 %.

OP. Valeur de sortie après les fonctions de simulation. Cette valeur a un statut de lecture/écriture en mode Manuel.

HR_OP, LR_OP. Echelle haute et basse de la sortie. Egalement, échelle haute & basse pour les objets graphiques liés à *OP* (bargraphes, tendance)

HL_OP, LL_OP. Limites hautes et basses de la sortie.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

NoiseMax (Bruit maximum). Définit la valeur maximale de l'amplitude de la fonction de bruit pseudo-aléatoire.

Lag1. Constante de temps du 1er filtre ou constante de l'intégrateur suivant la valeur du paramètre *Intgr*. Définit soit la constante de temps du filtre passe-bas du premier ordre ou la constante de l'intégrateur (exprimée en unités physiques par unité de temps).

Lag2. Constante de temps du 2ème filtre. Définit la constante de temps du filtre passe-bas du premier ordre.

TimeBase (Base de temps). (Sec./Min./Heures/Jours). Unités de temps pour *Lag1* et *Lag2*.

Intgr (Intégration). Permet de sélectionner la fonction pour le paramètre *Lag1* :

TRUE (VRAI) = Intégrateur

FALSE (FAUX) = filtre passe-bas du premier ordre

Invert (Inversion). Inverse le sens de *PV* (après décalage et mise à l'échelle en pourcentage) en soustrayant *PV%* de 100%.

Init (Initialisation). Initialise les filtres du bloc en rendant momentanément (pour deux itérations d'algorithme) les sorties égales aux entrées. Ne fonctionne qu'en mode Auto.

SelTrack (Sélection du mode poursuite). Permet de sélectionner le mode poursuite, et fait que le mode en cours est le mode de reprise.

AN_CONN : BLOC DE LIAISONS ANALOGIQUES

Fonction du bloc

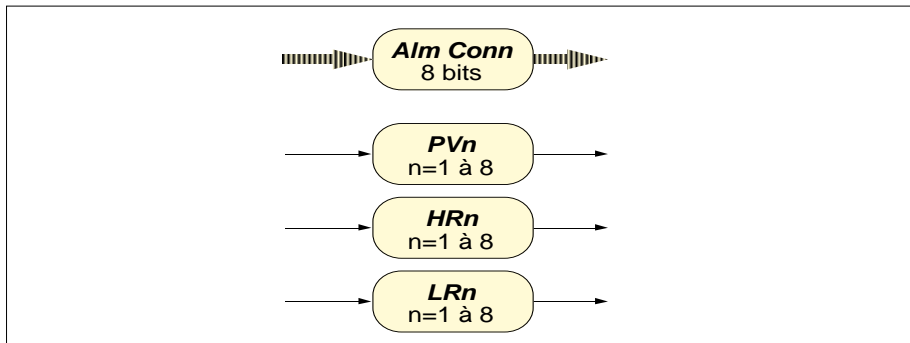


Figure 5-6 Schéma du bloc AN_CONN

Voir la figure 5-6. Le bloc de liaison analogique collecte et retransmet un maximum de 24 signaux analogiques, ainsi qu'un ensemble de 8 signaux logiques. Les signaux analogiques sont groupés en huit ensembles de signaux *PV*, *HR* et *LR* pour correspondre au trio de sorties analogiques d'un bloc ANIN unique. La principale utilisation du bloc AN_CONN est de regrouper en un bloc unique les signaux de huit blocs ANIN maximum fonctionnant dans une unité E/S d'un T100, avant de les transmettre sur le LIN (à un bloc AN_CONN image). Les liaisons *Alm Conn* permettent la transmission d'un signal logique sélectionné (généralement une sortie alarme) à partir de chaque bloc ANIN, bien que les liaisons puissent en fait être utilisées dans n'importe quel but.

Les avantages de transmettre des signaux analogiques sur le LIN par l'intermédiaire de blocs AN_CONN sont les suivants : moins de blocs à mettre en image, moins de trafic réseau, et un échantillonnage plus rapide du bloc, dans la mesure où le T100 passe moins de temps sur le réseau. Le bloc peut également être utilisé pour tamponner et synchroniser les données transmises sur le LIN.

Paramètres du bloc

Les symboles utilisés dans la table 5-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Alm Conn	Liaisons alarme (usage général)	CD Hex	
Alarm 1	Liaison alarme 1	V/F 1	D
Alarm 2	Liaison alarme 2	V/F 2	
Alarm 3	Liaison alarme 3	V/F 4	
Alarm 4	Liaison alarme 4	V/F 8	
Alarm 5	Liaison alarme 5	V/F 1	C
Alarm 6	Liaison alarme 6	V/F 2	
Alarm 7	Liaison alarme 7	V/F 4	
Alarm 8	Liaison alarme 8	V/F 8	
PV1 à PV8	Entrée analogique 1 à entrée 8	Eng1-Eng8	
HR1 à HR8	Echelle haute PV1 à échelle haute PV8	Eng1-Eng8	
LR1 à LR8	Echelle basse PV1 à échelle basse PV8	Eng1-Eng8	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 5-5 Paramètres du bloc AN_CONN

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Alm Conn. Champ binaire contenant huit paramètres logiques, *Alarme 1* à *Alarme 8*, qui peut être utilisé pour transmettre un signal logique. En général, chaque liaison devrait être utilisée pour transmettre un signal logique sélectionné à partir du champ *Alarmes* d'un bloc d'entrée analogique exécuté dans un T100.

NOTA. Les paramètres logiques dans ce champ binaire *ne repassent pas* à l'état d'alarme en cas d'alarme logiciel de communication du bloc.

PV1 à PV8. Les entrées analogiques retransmises comme des sorties mises à l'échelle par les paramètres *HR1*, *LR1* à *HR8*, *LR8* correspondants. Normalement, mais pas nécessairement, *PV1* à *PV8* devraient provenir des blocs ANIN correspondants.

HR1, LR1 à HR8, LR8. Entrées analogiques retransmises comme des sorties, qui mettent à l'échelle les paramètres *PV1* à *PV8* correspondants. Normalement, mais pas nécessairement, ces paramètres de mise à l'échelle devraient être liés à *HR* et *LR* du bloc ANIN qui fournit *PV* ; mais ils peuvent également être utilisés pour modifier les échelles des *PV* pour d'autres valeurs, si nécessaire.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

DG_CONN : BLOC DE LIAISONS LOGIQUES

Fonction du bloc

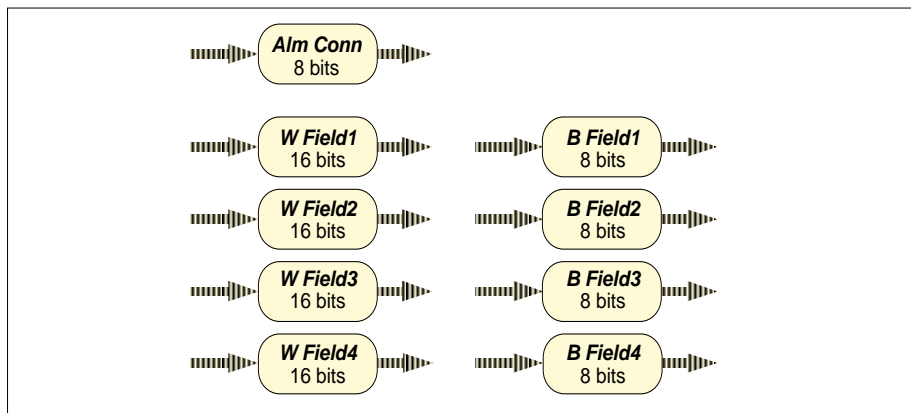


Figure 5-7 Schéma du bloc DG_CONN

Voir la figure 5-7. Le bloc de liaisons logiques collecte et retransmet un maximum de 104 signaux logiques. Les signaux logiques sont groupés pour des raisons pratiques en quatre mots de 16 bits (*W field1* à *W field4*), quatre octets (*B Field1* à *B Field4*), plus un champ *Alm Conn* à 8 bits. Cette capacité devrait permettre à un seul bloc DG_CONN de collecter les signaux d'entrée du schéma à partir de treize blocs DGIN_8 exécutés dans une unité E/S T100, avant de les transmettre sur le LIN (à un bloc DG_CONN image correspondant). Alternativement, moins de blocs DGIN étant regroupés, des signaux supplémentaires associés (par ex. les sorties statut) pourraient être collectés et transmis. En fait, les liaisons peuvent être utilisées pour les signaux logiques de n'importe quel bloc.

NOTA. Les liaisons ne devraient pas être utilisées pour les alarmes, dans la mesure où elles *ne repassent pas* à l'état d'alarme (c'est à dire à la sécurité intégrée) en cas d'alarme logiciel de communication d'un bloc.

Les avantages de transmettre des signaux analogiques sur le LIN par l'intermédiaire de blocs DG_CONN sont les suivants : moins de blocs à mettre en image, moins de trafic réseau, et un échantillonnage plus rapide du bloc, dans la mesure où le T100 passe moins de temps sur le réseau. Le bloc peut également être utilisé pour tamponner et synchroniser les données transmises sur le LIN.

Paramètres du bloc

Les symboles utilisés dans la table 5-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.







































Paramètre	Fonction	Unités	Etat
Alm Conn	Liaisons alarme (usage général)	CD Hex	
Alarm 1	Liaison alarme 1	V/F 	D
Alarm 2	Liaison alarme 2	V/F 	
Alarm 3	Liaison alarme 3	V/F 	
Alarm 4	Liaison alarme 4	V/F 	
Alarm 5	Liaison alarme 5	V/F 	C
Alarm 6	Liaison alarme 6	V/F 	
Alarm 7	Liaison alarme 7	V/F 	
Alarm 8	Liaison alarme 8	V/F 	
W Field n ($n=1$ à 4)	Mots entrée/sortie logiques	ABCD Hex	
Bit0	Bit 0 entrée/sortie logiques	V/F 	D
Bit1	Bit 1 entrée/sortie logiques	V/F 	
Bit2	Bit 2 entrée/sortie logiques	V/F 	
Bit3	Bit 3 entrée/sortie logiques	V/F 	
Bit4	Bit 4 entrée/sortie logiques	V/F 	C
Bit5	Bit 5 entrée/sortie logiques	V/F 	
Bit6	Bit 6 entrée/sortie logiques	V/F 	
Bit7	Bit 7 entrée/sortie logiques	V/F 	
Bit8	Bit 8 entrée/sortie logiques	V/F 	B
Bit9	Bit 9 entrée/sortie logiques	V/F 	
BitA	Bit A entrée/sortie logiques	V/F 	
BitB	Bit B entrée/sortie logiques	V/F 	
BitC	Bit C entrée/sortie logiques	V/F 	A
BitD	Bit D entrée/sortie logiques	V/F 	
BitE	Bit E entrée/sortie logiques	V/F 	
BitF	Bit F entrée/sortie logiques	V/F 	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
B Field n ($n=1$ à 4)	Octets entrée/sortie logiques	CD Hex	
Bit0	Bit 0 entrée/sortie logiques	V/F 	D
Bit1	Bit 1 entrée/sortie logiques	V/F 	
Bit2	Bit 2 entrée/sortie logiques	V/F 	
Bit3	Bit 3 entrée/sortie logiques	V/F 	
Bit4	Bit 4 entrée/sortie logiques	V/F 	C
Bit5	Bit 5 entrée/sortie logiques	V/F 	
Bit6	Bit 6 entrée/sortie logiques	V/F 	
Bit7	Bit 7 entrée/sortie logiques	V/F 	

Table 5-6 Paramètres du bloc DG_CONN

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Alm Conn. Champ binaire contenant huit paramètres logiques, désignés par *Alarm 1* à *Alarm 8*, qui peut être utilisé pour collecter et retransmettre les signaux d’alarme des blocs d’entrée logique ou tout autre signal logique.

NOTA. Les paramètres logiques dans ce champ binaire *ne repassent pas* à l’état d’alarme en cas d’alarme logiciel de communication du bloc.

W Field1 à W Field4. Champs Mot 1 à 4. Quatre champs de 16 bits à usage général (*Bit0* à *BitF*) qui peuvent être utilisés pour collecter et retransmettre les signaux logiques.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

B Field1 à B Field4. Champs octet 1 à 4. Quatre champs de 8 bits à usage général (*Bit0* à *Bit7*) qui peuvent être utilisés pour collecter et retransmettre les signaux logiques.

TP_CONN : BLOC DE LIAISONS DES DONNEES TIEDES

Fonction du bloc

Le bloc TP_CONN permet de définir un maximum de neuf champs pour les sauvegardes dans l'E'PROM des "données tièdes" à la mise hors tension. S'il devait y avoir un "démarrage tiède" ultérieurement, ces champs reprennent les valeurs qu'ils avaient au moment de la mise hors tension. (Voir les détails dans les manuels produits en question de la série T600 sur le démarrage et les données tièdes).

Les neuf champs sont définis par nom de bloc (dans les paramètres *Blockn*) et nom de champ (dans les paramètres *Fieldn*). Ils peuvent se trouver dans n'importe quel bloc dans la base de données, mais ne peuvent être des champs *Alarms*, des champs de chaînes de texte ou le champ *EXPR* du bloc EXPR. La spécification d'un champ interdit ou inexistant fait passer le bit *BadField* à VRAI.

Il ne peut y avoir qu'un bloc TP_CONN par base de données. Son nom de bloc est écrit dans le paramètre *AnConBlk* du bloc T600. Noter que si un bloc TP_CONN est désigné dans le bloc T600, l'utilisation d'un bloc AN_CONN ou DG_CONN pour la définition des données tièdes est exclue.

Paramètres du bloc

Les symboles utilisés dans la table 5-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
BadField	Invalid field flags	(A)BCD hex	
Field1	VRAI = Champ1 inexistant ou interdit	V/F	<div>1</div> <div>2</div> <div>4</div> <div>8</div> <div>D</div>
Field2	VRAI = Champ2 inexistant ou interdit	V/F	
Field3	VRAI = Champ3 inexistant ou interdit	V/F	
Field4	VRAI = Champ4 inexistant ou interdit	V/F	
Field5	VRAI = Champ5 inexistant ou interdit	V/F	<div>1</div> <div>2</div> <div>4</div> <div>8</div> <div>C</div>
Field6	VRAI = Champ6 inexistant ou interdit	V/F	
Field7	VRAI = Champ7 inexistant ou interdit	V/F	
Field8	VRAI = Champ8 inexistant ou interdit	V/F	
Field9	VRAI = Champ9 inexistant ou interdit	V/F	<div>1</div> <div>2</div> <div>4</div> <div>8</div> <div>B</div>
Block1 à Block9	Définit noms de bloc contenant champs Field1 à Field9		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Field1 à Field9	Définit les champs contenus dans Block1 à Block9	Alphanumérique	

Table 5-7 Paramètres du bloc TP_CONN

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

BadField (Champ incorrect). Champ binaire en lecture uniquement dont les neuf bits - *Field1* à *Field9* - indiquent si les champs correspondants définis par le paramètre *Fieldn* sont des champs valables. Un bit VRAI indique qu'un champ interdit ou inexistant a été défini.

Block1 à Block9. Permet de définir un maximum de neuf noms de bloc de base de données (paramètres Block) contenant des champs - spécifiés dans les paramètres *Field1* à *Field9* - qui doivent être sauvegardés comme données tièdes à la mise hors tension.

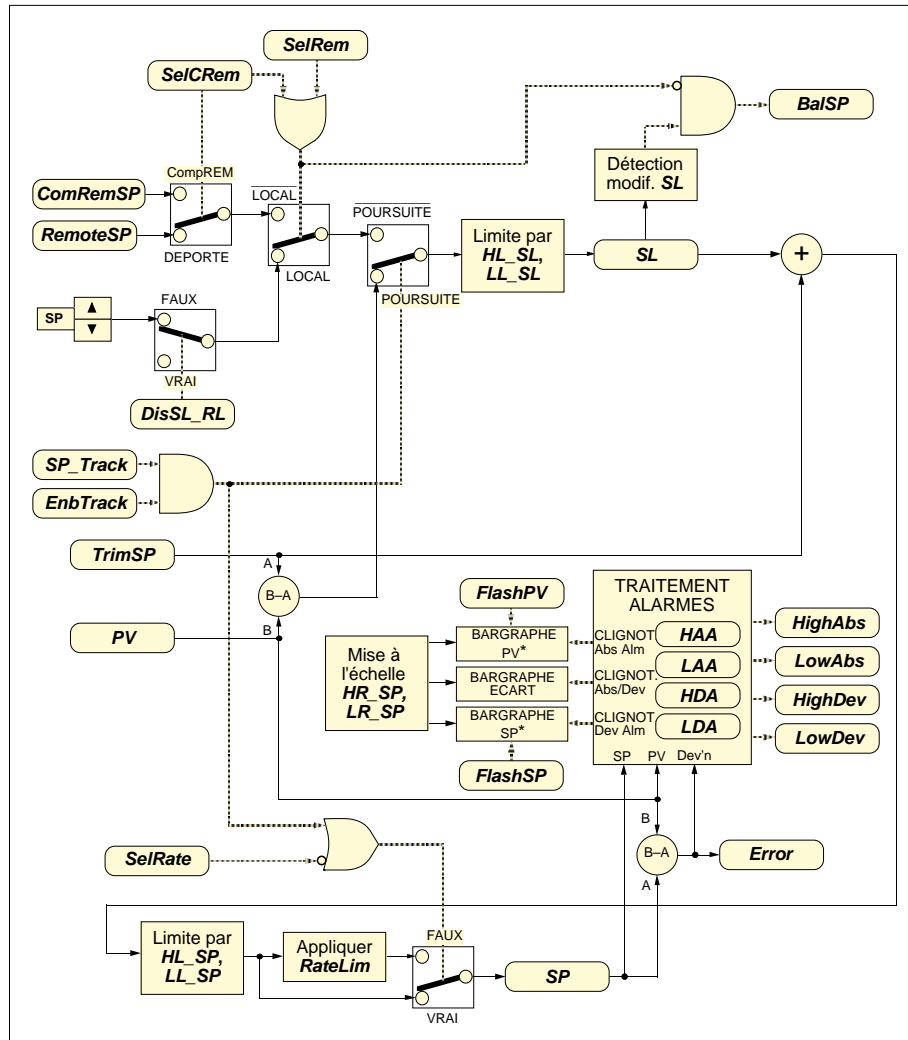
Field1 à Field9. Permet de définir un maximum de neuf noms de champ contenus dans les blocs spécifiés dans les paramètres *Block1* à *Block9* qui doivent être sauvegardés comme données tièdes à la mise hors tension.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

SETPOINT : BLOC CONSIGNE

Fonction du bloc



*Les sorties bargraphes PV & SP sont liées aux affichages de la face avant du T600 uniquement si la boucle affectée du bloc est celle sélectionnée pour l'affichage principal.

Figure 5-8 Schéma du bloc SETPOINT

Se reporter au schéma de la figure 5-8. Le bloc SETPOINT est destiné à être utilisé comme partie de la combinaison de blocs de régulation SETPOINT/3_TERM/ MAN_STAT/MODE, ce qui permet de disposer de plus de souplesse qu'avec un simple bloc PID. Le bloc génère un point de consigne *SP* en lecture uniquement provenant de différentes sources, et le soumet à des seuils hauts/bas, des seuils d'ajustage, de vitesse, et fournit des alarmes absolues et d'écart.

Paramètres du bloc

Les symboles utilisés dans la table 5-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
FallBack	Mode de fonctionnement de reprise (supprimé)	Menu	
PV	Variable procédé	Eng	
SP	Point de consigne résultant	Eng	
SL	Point de consigne local	Eng	
TrimSP	Décalage ajouté à SL	Eng	
RemoteSP	Point de consigne déporté	Eng	
ComRemSP	Point de consigne déporté ordinateur	Eng	
Error	PV – SP	Eng	
HR_SP	Echelle haute (pour affichages)	Eng	
LR_SP	Echelle basse (pour affichages)	Eng	
HL_SP	Seuil haut de SP	Eng	
LL_SP	Seuil bas de SP	Eng	
HL_SL	Seuil haut de SL	Eng	
LL_SL	Seuil bas de SL	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
HighAbs	En alarme haute absolue	V/F	
LowAbs	En alarme basse absolue	V/F	
HighDev	En alarme d'écart haute	V/F	
LowDev	En alarme d'écart basse	V/F	
Combined	Alarme combinée du bloc	V/F	
HAA	Seuil haut d'alarme absolue	Eng	
LAA	Seuil bas d'alarme absolue	Eng	
HDA	Seuil haut d'alarme d'écart	Eng	
LDA	Seuil bas d'alarme d'écart	Eng	
Hyst	Hystérésis des alarmes	Eng	
Dis_DP	Position du point décimal (face avant T600)	0 - 4	
DevnBar	Personnalise le bargraphe d'écart face avant	Menu	
RateLim	Seuil de vitesse pour SP	Eng	
TimeBase	Unités de temps du seuil de vitesse	Menu	

suite...

...suite





Paramètre	Fonction	Unités	Etat
Options		(A)BCD hex 	
SP_Track	Sélection PV SP_Track	V/F — 1	D
EnbTrack	Validation bit SP_Track	V/F — 2	
SelRem	Sélection déporté	V/F — 4	
SelCRem	Sélection déporté ordinateur	V/F — 8	
DisSL_RL	Invalidation incrémentation/décrémentation SL	V/F — 1	C
SelRate	Validation limitation de vitesse	V/F — 2	
FlashPV	Clignotement du bargraphe PV	V/F — 4	
FlashSP	Clignotement du bargraphe SP	V/F — 8	
Show_SP	Force l’affichage SP en unités d’affichage face avant	V/F — 1	B
		2	
		4	
		8	
Status		(ABC)D Hex 	
BalSP	Demande d’équilibrage sur modif. SP (une itération)	V/F — 1	D
		2	
		4	
		8	

Table 5-8 Paramètres du bloc SETPOINT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (LOCAL/REMOTE/CompREM/TRACK - LOCAL/DEPORTE/CompREM/POURSUITE). *Mode* est un paramètre de lecture uniquement qui montre le mode de fonctionnement actif du bloc qui résulte de différents bits d’options. Dans l’ordre de priorité, les modes possibles sont: TRACK (priorité la plus haute), LOCAL, CompREM et REMOTE (priorité la plus basse):

- **TRACK (POURSUITE)** est sélectionné lorsque *Options.SP_Track* et *Options.EnbTrack* sont VRAI. Noter que dans les schémas de boucles standard (préconfigurés) du T600, *EnbTrack* n’est pas relié, ce qui permet à l’utilisateur de valider/invalider la fonction poursuite par l’intermédiaire de la face avant, en sélectionnant respectivement *EnbTrack* VRAI ou FAUX.
- **LOCAL** ce mode est sélectionné lorsque *Options.SelRem* et *Options.SelCRem* sont FAUX. En mode local, la valeur de *SL* peut être réglée en utilisant les boutons-poussoirs incrémentation/décrémentation de la face avant. Le réglage de *SL* peut être invalidé en sélectionnant VRAI pour le bit *Options.DisSL_RL*.
- **CompREM (DEPORTE ORDINATEUR)** ce mode est sélectionné lorsque *Options.SelCRem* est VRAI.
- **REMOTE (DEPORTE)** ce mode est sélectionné lorsque *Options.SelRem* est VRAI et que *Options.SelCRem* est FAUX.

FallBack (Reprise). (LOCAL/REMOTE/CompREM - LOCAL/DEPORTE/CompREM). Indique le mode de fonctionnement adopté si le mode actif est désélectionné.

PV. Variable procédé.

SP. Point de consigne résultant.

SL. Point de consigne local. *SL* est soumis à l'ajustage (trim), aux seuils hauts et bas et aux seuils de vitesse de changement avant de devenir *SP*, le point de consigne résultant. Les seuils du point de consigne sont utilisés dans tous les modes de fonctionnement et agissent indépendamment à la fois sur les valeurs *SL* et *SP*.

TrimSP. Réglage du point de consigne.

RemoteSP. Point de consigne déporté. Permet de contrôler le point de consigne en mode déporté uniquement.

ComRemSP. Point de consigne déporté de l'ordinateur. Permet de contrôler le point de consigne en mode CompREM uniquement.

Error (Erreur). Ce champ montre la différence entre *PV* et *SP*, c'est à dire $Error = PV - SP$.

HR_SP, LR_SP. Echelles haute et basse pour les affichages de la face avant du T600 du point de consigne, de l'écart et de la variable procédé (en unités physiques).

HL_SP, LL_SP. Seuils haut et bas du point de consigne résultant *SP*.

HL_SL, LL_SL. Seuils haut et bas du point de consigne local *SL*. Noter qu'en appliquant une limitation indépendante à *SL* et *SP*, l'échelle de *SP* n'est pas réduite en augmentant la valeur de *TrimSP* - ce qui serait le cas avec seulement un ensemble unique de seuils.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **HighAbs, LowAbs.** Alarmes *PV* absolues hautes et basses définies par les paramètres *HAA* et *LAA* avec une bande d'hystérésis définie par l'utilisateur (*Hyst*) pour chaque seuil. Lorsqu'en alarme absolue, le bargraphe *PV* de la face avant du T600 clignote. Noter qu'il clignote aussi si *Options.FlashPV* est VRAI.

■ **HighDev, LowDev.** Alarmes d'écart haute et basse (*Error*) définies par les paramètres *HDA* et *LDA* avec une bande d'hystérésis définie par l'utilisateur (*Hyst*) pour chaque seuil. Une alarme d'écart haute se produit lorsque $Error > HDA$ et une alarme d'écart basse lorsque $-Error > LDA$. Lorsqu'en alarme d'écart, le bargraphe *SP* de la face avant du T600 clignote. Noter qu'il clignote aussi si *Options.FlashSP* est VRAI.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

HAA, LAA. Seuils d'alarme absolue haute et basse de *PV* en unités physiques. Ces consignes d'alarme peuvent être affichées sur le bargraphe *PV* de la face avant du T600

comme des LED en éclairage inverse, en appuyant simultanément sur les boutons-poussoirs incrémentation et décrémentation.

HDA, LDA. Seuil d'alarme d'écart haute et basse (*Error*) en unités physiques. Ils permettent d'afficher les alarmes lorsque l'écart entre *PV* et *SP* est supérieur à la valeur spécifiée. *HDA* et *LDA* peuvent être affichées sur le bargraphe SP de la face avant du T600 comme des LED en éclairage inverse, en appuyant simultanément sur les boutons-poussoirs incrémentation et décrémentation.

Hyst. Permet de spécifier une valeur d'hystérésis en unités physiques pour les alarmes absolues et d'écart. Une fois affichée, l'alarme n'est pas supprimée tant que la valeur qui a provoqué l'alarme n'est pas revenue au seuil et dans les limites de la valeur spécifiée par le paramètre hystérésis.

Dis_DP. Permet de spécifier la position du point décimal sur l'affichage à 5 chiffres de la face avant du T600. La plage admissible de *Dis_DP* est de 0 à 4, ce qui définit le nombre de chiffres affichés après le point décimal.

DevnBar. (*Abs_PV*, 1/2/3,1/5/10,10/20/30). Permet de spécifier les graduations et l'utilisation du bargraphe d'écart récapitulatif de la face avant du T600 lié à ce bloc.

Abs_PV le configure comme un bargraphe "PV absolu" où chaque segment de LED représente 16,67 % de l'échelle de PV. **1/2/3**, **1/5/10** et **10/20/30** en font un bargraphe d'écart avec des graduations de 1-2-3, 1-5-10 et 10-20-30 % d'écart par segment.

RateLim. Seuil de vitesse de changement pour *SP* en unités physiques par seconde ou par minute (spécifié par le paramètre *TimeBase*). La limitation de vitesse est uniquement appliquée si *Options.SelRate* est VRAI, et si le bloc n'est pas en mode poursuite.

TimeBase (Base de temps). (sec/min). Permet de spécifier les unités de temps pour le paramètre de limitation de vitesse *RateLim*.

Options. Champ binaire définissant le mode de fonctionnement et différentes autres options d'exploitation du bloc. Les indications suivantes sont données en plus de celles de la table 5-8 et de la figure 5-8.

- **Show_SP.** La valeur de *SP* est uniquement affichée sur le bargraphe SP-W de la face avant du T600 si la tâche utilisateur (boucle) à laquelle le bloc est affecté est sélectionnée comme l'affichage principal de la boucle. Lorsque *Options.Show_SP* est VRAI, *SP* est également affiché numériquement sur l'affichage à 5 chiffres.

Status (Etat).

- **BalSP.** Ce drapeau passe à VRAI (uniquement pour une itération de bloc) si la valeur de *SL* a changé au cours de cette itération, et si le bloc n'est pas en mode déporté, c'est ce qui se produit si *SL* a été écrit directement ou si les boutons-poussoirs incrémentation/décrémentation de la face avant ont été utilisés. *BalSP* est également VRAI si le changement dans *SL* est du au passage à l'un des modes déportés ou à un changement entre modes déportés.

BalSP peut être relié à une entrée *SPbal* du bloc 3_TERM pour empêcher des "à-coups" dans la sortie du régulateur. Voir les détails dans la section sur le bloc 3_TERM.

3_TERM : BLOC PID INCREMENTIEL

Fonction du bloc

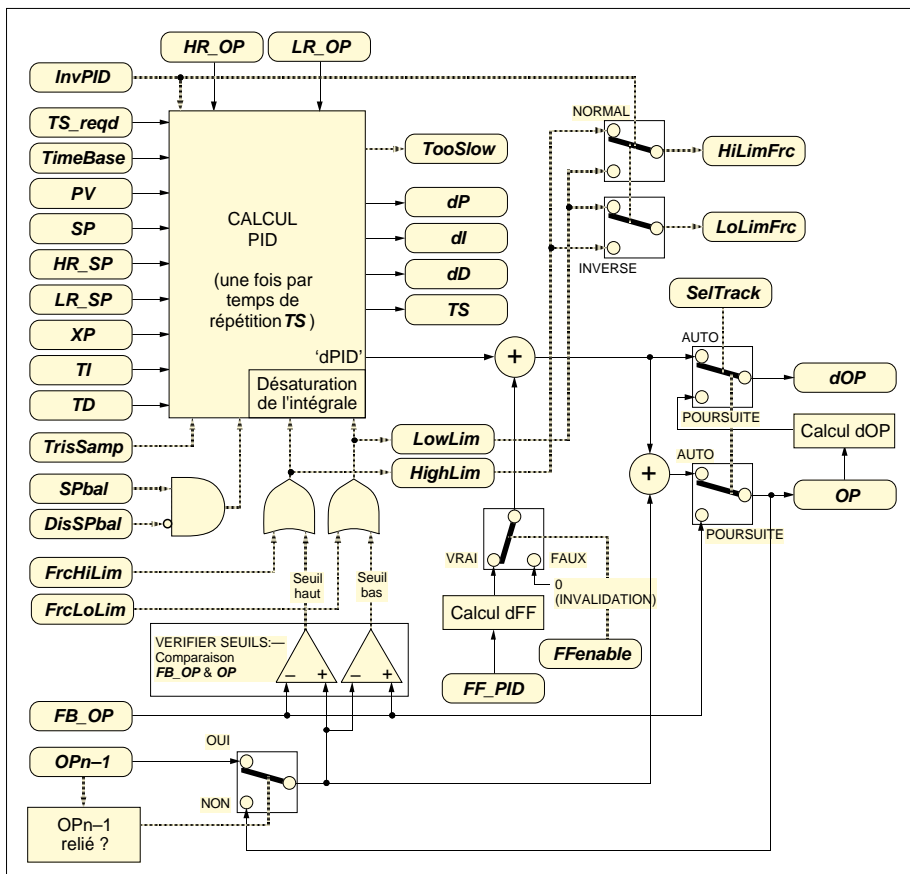


Figure 5-9 Schéma du bloc 3_TERM

Voir le schéma de la figure 5-9. Le bloc 3_TERM est une forme incrémentielle du bloc de régulation à actions proportionnelle intégrale dérivée, destiné à être utilisé comme partie de la combinaison de blocs de régulation SETPOINT/3_TERM/MAN_STAT/MODE, ce qui donne plus de souplesse qu'un simple bloc PID. Le bloc génère une sortie incrémentielle dOP qui permet de piloter des actionneurs de type incrémentiel, ainsi qu'une sortie normale OP .

Le bloc 3_TERM calcule la sortie incrémentielle dOP et la sortie normale OP à la n ème itération comme suit:

$$dOP_n = -(100/XP_n)[\Delta ER_n + (TS_n/TL_n)ER_n + (TD_n/TS_n)\Delta^2PV_n] + \Delta FF_PID_n,$$

$$\text{et } OP_n = OP_{n-1} + dOP_n$$

où dOP , OP , PV , SP , XP , TS , TL , TD , FB_OP , et FF_PID sont les paramètres définis dans le paragraphe *Menu de spécifications du bloc* ci-dessous, et

$$\Delta FF_PID_n = FF_PID_n - FF_PID_{n-1}$$

$$ER_n = PV_n - SP_n$$

$$\Delta ER_n = ER_n - ER_{n-1}$$

$$\Delta^2PV_n = \Delta PV_n - \Delta PV_{n-1}$$

$$\Delta PV_n = PV_n - PV_{n-1}$$

$$\Delta SP_n = SP_n - SP_{n-1}.$$

Cette forme de l'algorithme PID présente certaines fonctions avantageuses, notamment:

- Les constantes de mise au point peuvent être modifiées dynamiquement sans provoquer "d'à-coups" dans la sortie.
- La régulation de type 6351 (avec sorties d'incrément/décroissement) est facile à émuler
- Le sens de la régulation PID peut être modifié sans à-coups.

Les fonctions suivantes correspondent à celles de la mise en oeuvre PID TCS standard (par ex. comme celle utilisée dans le bloc PID):

- L'à-coups proportionnel sur modifications SP peut être éliminé en mettant ΔSP_n , dans l'expression $\Delta ER_n = \Delta PV_n - \Delta SP_n$, à zéro pour cette itération.
- La désaturation de l'intégrale est effectuée en remplaçant ER_n dans le terme de l'intégrale par: $XP_n/100 \times (FB_OP_n - OP_{n-1})$

Paramètres du bloc

Les symboles utilisés dans la table 5-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement	Menu	
PV	Variable procédé	EngA	◀◻▶
SP	Point de consigne de régulation	EngA	◀◻▶
OP	Sortie PID	EngB	◻▶ 📖
HR_SP	Echelle haute pour PV et SP	EngA	◀◻▶
LR_SP	Echelle basse pour PV et SP	EngA	◀◻▶
HR_OP	Echelle haute pour OP, FF_PID, FB_OP	EngB	◀◻▶
LR_OP	Echelle basse pour OP, FF_PID, FB_OP	EngB	◀◻▶

suite...

... suite

Paramètre	Fonction	Unités	Etat
OPn-1	Valeur précédente d'OP	EngB	
dOP	Modification d'OP pour cette itération	EngB	
dP	Modification due au terme P	EngB	
dI	Modification due au terme I	EngB	
dD	Modification due au terme D	EngB	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
TooSlow	Si TS > (TI/10)	V/F	
Combined	Alarme combinée du bloc	V/F	
TimeBase	Sélection des unités pour TI, TD, TS	Menu	
XP	Bande proportionnelle	%	
TI	Temps de l'intégrale	Secs/Mins	
TD	Temps de la dérivée	Secs/Mins	
TS_reqd	Valeur demandée de TS	Secs/Mins	
TS	Temps d'échantillonnage de l'algorithme	Secs/Mins	
Options		(A)BCD Hex	
TrigSamp	Déclenchement exécution PID	V/F — 1	D
InvPID	Inversion du sens de PID	V/F — 2	
SPbal	Mise à zéro dSP (équilibrage)	V/F — 4	
DisSPbal	Invalidation SPbal	V/F — 8	
FFenable	Validation tendance	V/F — 1	C
SelTrack	Invalidation PID (sélection mode poursuite)	V/F — 2	
SelOnOff	Sélection mode de régulation tout ou rien	V/F — 4	
HiDband	Bande morte haute pour mode tout ou rien	V/F — 8	
LoDband	Bande morte basse pour mode tout ou rien	V/F — 1	B
AbsNoI	Utilisation algorithme PD absolu si TI = 0	V/F — 2	
FrcHiLim	Forçage PID comme si en seuil haut	V/F — 4	
FrcLoLim	Forçage PID comme si en seuil bas	V/F — 8	
Status		(AB)CD Hex	
LowLim	OP en dessous du seuil bas	V/F — 1	D
HighLim	OP au dessus du seuil haut	V/F — 2	
ThisTime	Mise à jour PID au cours de cette itération	V/F — 4	
HiLimFrc	Forçage régulateur maître comme si en seuil haut	V/F — 8	
LoLimFrc	Forçage régulateur maître comme si en seuil bas	V/F — 1	C
		2	
		4	
		8	
FF_PID	Tendance	EngB	
FB_OP	Contre-réaction (après limitation)	EngB	
DeadBand	Bande morte pour mode de régulation tout ou rien	EngA	

Table 5-9 Paramètres du bloc 3_TERM

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (AUTO/TRACK - AUTO/POURSUITE). Mode de fonctionnement actif.

PV. Variable procédé.

SP. Point de consigne déporté.

OP. Sortie PID. En mode automatique, OP est calculé en ajoutant dOP (le changement dans la sortie PID) à la valeur OP de l’itération précédente, OP_{n-1} . OP_{n-1} peut être fourni de manière interne à partir d’ OP ou de manière externe à partir de l’entrée $OPn-1$ si la liaison existe (détection automatique par le bloc). Voir figure 5-9. En mode poursuite, OP poursuit simplement l’entrée FB_OP .

HR_SP, LR_SP. Echelles haute et basse de SP et PV en unités physiques.

HR_OP, LR_OP. Echelles haute et basse d’ OP , FF_PID et FB_OP en unités physiques.

OPn-1. Cette entrée peut être reliée à une sortie de “position mesurée” externe comme source de OP_{n-1} , à partir de laquelle est produite une valeur OP_n valable et active (par ex., pour des besoins d’affichage). En outre, si $OPn-1$ n’est pas relié, la valeur OP interne de l’itération du bloc (OP_{n-1}) précédente est utilisée automatiquement.

dOP. Modification de la valeur d’ OP au cours de cette itération du bloc, c’est à dire $OP_n - OP_{n-1}$.

dP. Modification de la valeur du terme proportionnelle (P) au cours de cette itération du bloc.

dI. Modification de la valeur du terme de l’intégrale (I) au cours de cette itération du bloc.

dD. Modification de la valeur du terme de la dérivée (D) au cours de cette itération du bloc.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **TooSlow (Trop lent).** Si la valeur du temps d’échantillonnage de l’algorithme PID TS dépasse $TI/10$, la période de répétition de l’algorithme est considérée comme dangereusement basse et l’alarme *TooSlow* est déclenchée, ce qui permet d’éviter des situations où, à la suite d’une mise au point dynamique, le bloc 3_TERM exige une période de répétition de boucle plus rapide que ce qui est demandé. Noter que *TooSlow* est invalidé lorsque TI est mis à zéro.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active à la priorité la plus haute du bloc.

TimeBase (Base de temps). (sec/min). Permet de spécifier les unités de temps pour TI , TD et TS .

XP. Bande proportionnelle (%)

TI. Constante de temps de l'intégrale en secondes ou minutes suivant la valeur de *TimeBase*. Si $TI=0$, le terme de l'intégrale est invalidé, c'est à dire configure le mode PD incrémentiel.

TD. Constante de temps de la dérivée en secondes ou minutes suivant *TimeBase*.

TS_reqd. Permet de spécifier une valeur demandée de *TS*, le temps d'échantillonnage de l'algorithme en secondes ou minutes suivant la valeur de *TimeBase*, ce qui n'est pas nécessairement la valeur adoptée par *TS* (voir paragraphe suivant).

TS. Temps d'échantillonnage de l'algorithme PID en secondes ou minutes suivant la valeur de *TimeBase*. *TS* prend la plus grande des deux valeurs suivantes:

- La plus grande valeur de *TI* et *TD* divisée par 512
- La valeur spécifiée dans *TS_reqd*

Si la valeur ainsi calculée de *TS* n'est pas un multiple du cycle de répétition de la boucle, elle est arrondie au multiple suivant le plus proche du cycle de répétition de la boucle. Noter que *TS* donne toujours l'intervalle réel entre les exécutions PID.

Si *TS* dépasse la période de répétition du bloc, *PV* et *SP* sont échantillonnés à des intervalles de *TS* pour être utilisés dans l'algorithme PID. (et *SPbal* est verrouillé entre les exécutions PID). En outre, si *TS* est supérieur à la période de répétition du bloc, un front montant de *TrigSamp* déclenche l'exécution de l'algorithme PID et remet à zéro l'horloge *TS*.

Options. Un champ binaire permettant aux entrées de contrôler le fonctionnement du bloc.

- **TrigSamp.** Permet de déclencher l'exécution de l'algorithme PID.
- **InvPID.** Permet de sélectionner le sens de l'action de régulation à 3 termes. Si *InvPID* est FAUX, une augmentation du terme d'erreur ($PV - SP$) produit une diminution correspondante dans la sortie à 3 termes pour compenser. Si *InvPID* est VRAI, l'augmentation de l'erreur produit une augmentation correspondante dans la sortie à 3 termes pour compenser.
- **SPBal.** Permet d'éviter un "à-coups proportionnel" dans la sortie de régulation qui peut se produire au cours des changements de *SP*. Dans ce cas, le terme proportionnel est calculé à partir des changements de *PV* plutôt qu'à partir des changements d'erreur. Plus précisément, une entrée VRAIE de *SPbal* met ΔSP_n (c'est à dire, $SP_n - SP_{n-1}$) à zéro pendant une itération de l'algorithme PID. Noter que le bit *Options.DisSPbal* doit être FAUX pour que *SPbal* fonctionne.
SPbal est destiné à être relié à partir de la sortie *Status.BalSP* du bloc SETPOINT, de sorte que des écritures manuelles dans *SL* - localement ou par l'intermédiaire des communications du superviseur - ne provoquent pas d'à-coups de sortie du régulateur.
- **DisSPbal.** VRAI invalide l'action du bit *SPbal*.
- **FFenable.** Valide l'application de la tendance, *FF_PID* sur la sortie d'OP. Noter qu'en mode PD absolu (*Options.AbsNol* VRAI), la tendance ne peut être invalidée et *FFenable* est ignoré.

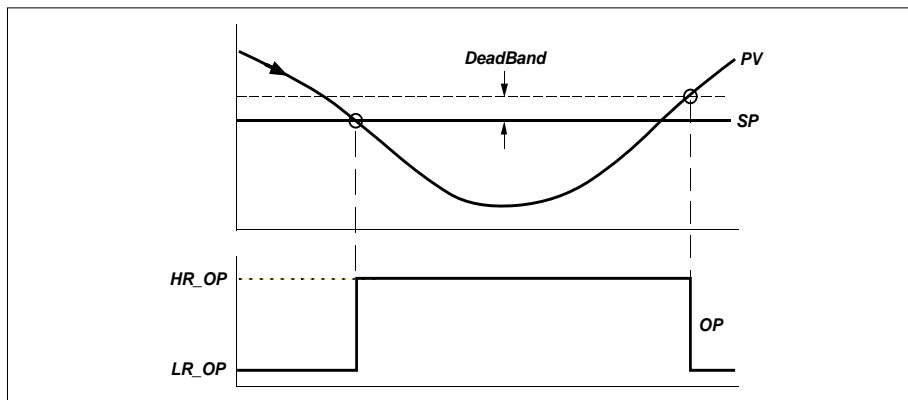


Figure 5-10 Régulation tout ou rien du bloc à 3 termes - HiDband VRAI, LoDband FAUX

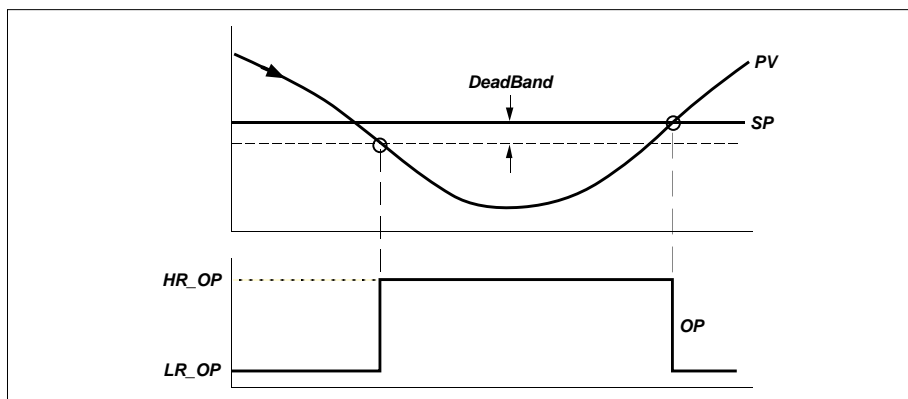


Figure 5-11 Régulation tout ou rien du bloc à 3 termes - HiDband FAUX, LoDband VRAI

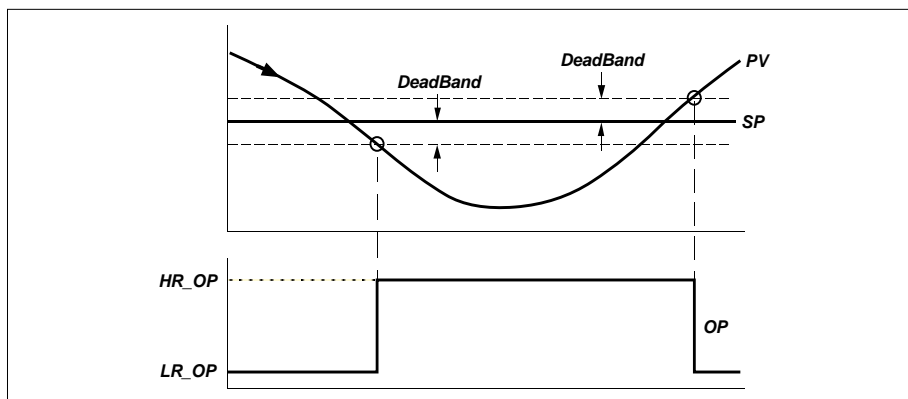


Figure 5-12 Régulation tout ou rien du bloc à 3 termes - HiDband VRAI, LoDband VRAI

- **SelTrack.** Permet de sélectionner le mode poursuite, ce qui invalide l'action PID et force *OP* à poursuivre *FB_OP* (sortie de contre-réaction après limitation) et *dOP* à être calculé à partir de là.

Si *SelTrack* est VRAI, tous les totalisateurs intermédiaires sont mis à zéro et les sauvegardes internes des valeurs de l'itération précédente de *PV* et *FF_PID* sont égales à leurs valeurs actives (par ex. $PV_{n-1} = PV_n$) pour assurer un transfert absolument sans à-coups lors du retour à la régulation PID.

- **SelOnOff.** Permet de sélectionner le mode de régulation tout ou rien, avec une bande morte (hystérésis) spécifiée par *DeadBand*, fonctionnant au-dessus ou en-dessous du point de consigne spécifié par les bits *HiDband* et *LoDband*.
- **HiDband, LoDband.** Permet de sélectionner une bande morte "haute" (Figure 5-10) ou une bande morte "basse" (Figure 5-11) en mode de régulation tout ou rien d'une plage spécifiée par *DeadBand*. Lorsque les deux bits sont VRAI, deux bandes mortes opèrent autour de *SP* (Figure 5-12). Lorsque les deux bits sont FAUX, la bande morte est complètement invalidée.
- **AbsNol.** Si *TI* = 0 pour invalider le terme de l'intégrale, *AbsNol* VRAI permet de sélectionner un algorithme PD absolu (non-incrémentiel) dans lequel *dP*, *dI*, *dD* et *dOP* sont invalidés (zéro).
- **FrcHiLim, FrcLoLim.** VRAI force l'algorithme PID à agir comme si la sortie avait respectivement une limite haute ou basse, et la désaturation de l'intégrale est alors appliquée.

Status (Etat). Champ binaire indiquant l'état de l'algorithme PID.

- **LowLim.** VRAI indique que *OP* est en-dessous d'une limite basse, c'est à dire en-dessous de la sortie de contre-réaction (limitée) ou est forcée d'agir comme si c'était le cas (par *FrcLoLim*).
- **HighLim.** VRAI indique que *OP* est au-dessus d'une limite haute, c'est à dire au-dessus de la sortie de contre-réaction (limitée) ou est forcée d'agir comme si c'était le cas (par *FrcHiLim*).
- **ThisTime.** VRAI indique que l'algorithme PID a été mis à jour au cours de cette itération du bloc 3_TERM.
- **HiLimFrc.** VRAI indique que le paramètre *HighLim* est VRAI (ou avec la sortie à action inverse que *LowLim* est VRAI). Cette sortie est destinée à être utilisée pour forcer un bloc régulateur maître à agir comme s'il était limité, par ex. dans une paire en cascade.
- **LoLimFrc.** VRAI indique que le paramètre *LowLim* est VRAI (ou avec la sortie à action inverse que *HighLim* est VRAI). Cette sortie est destinée à être utilisée pour forcer un bloc régulateur maître à agir comme s'il était limité, par ex. dans une paire en cascade.

FF_PID. Tendence. Permet d'ajouter un décalage à la sortie à 3 termes, c'est ce qui est parfois appelé "remise à zéro manuelle". Noter que *FF_PID* est mis à l'échelle $\pm(HR_OP - LR_OP)$.

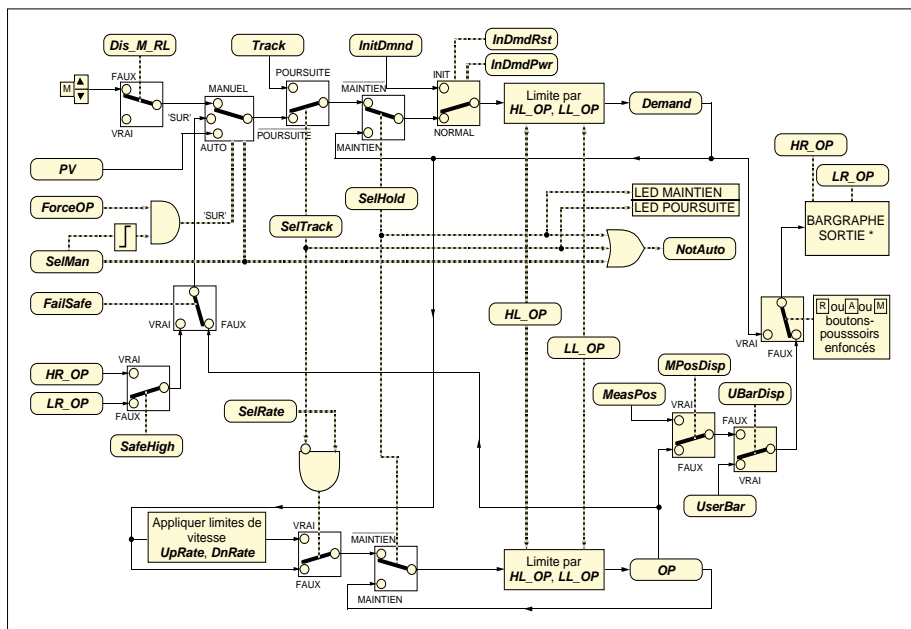
FB_OP. Une valeur de contre-réaction d'*OP* de la dernière itération du bloc PID, après avoir été soumis à une limitation externe. *FB_OP* permet de maintenir la valeur de sortie en mode poursuite et également pour détecter la limitation de sortie - en la comparant à *OP* - de sorte que la désaturation de l'intégrale puisse être appliquée.

DeadBand (Bande morte). Permet de spécifier la plage de la ou des bandes mortes à appliquer lorsque le bloc est en mode de régulation tout ou rien.

Options.HiDband et *.LoDband* détermine comment la ou les bandes mortes opèrent autour de *SP* - voir figures 5-10 à 5-12. Noter que si *HiDband* et *LoDband* sont tous les deux FAUX, l'action de la bande morte est invalidée.

MAN_STAT : BLOC STATION MANUELLE

Fonction du bloc



* Lié à l'affichage de la face avant réelle uniquement si la boucle associée est celle sélectionnée pour l'affichage principal de la boucle.

Figure 5-13 Schéma du bloc MAN_STAT

Voir le schéma de la figure 5-13. Le bloc de station manuelle MAN_STAT est destiné à être utilisé comme partie de la combinaison des blocs de régulation SETPOINT/3_TERM/MAN_STAT/MODE, ce qui donne plus de souplesse qu'un simple bloc PID.

En mode automatique, la sortie suit l'entrée **PV** - en général une sortie du bloc 3_TERM - soumise à la limitation de vitesse haute/basse. En mode manuel, l'opérateur peut utiliser les boutons-poussoirs incrémentation/décrémentation de la face avant du T600 pour faire varier **OP** dans le cadre des limites. En mode poursuite, (priorité la plus haute), **OP** est calculé à partir du paramètre **Track** et reste limité par les seuils haut/bas (mais pas en vitesse). Le mode maintien (hold) permet de "geler" à la fois le paramètre **OP** et **Demand**.

Les paramètres **MeasPos** et **MPosDisp** du bloc permettent de sélectionner entre l'affichage d'une "position mesurée" ou **OP** sur le bargraphe de sortie du T600, et avec **UBarDisp** vous pouvez contrôler directement chaque segment du bargraphe à partir du schéma de boucles (par l'intermédiaire du champ binaire **UserBar**). Le bloc contrôle également les LED **H** et **T** sous le bargraphe d'écart de boucle associé. Noter que le bargraphe de sortie est contrôlé uniquement si la tâche utilisateur du bloc est celle sélectionnée pour l'affichage principal de la boucle.

Paramètres du bloc

Les symboles utilisés dans la table 5-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement	Menu	
FallBack	Mode de fonctionnement de reprise (supprimé)	Menu	
PV	Entrée station manuelle	Eng	
Track	Valeur de poursuite de sortie	Eng	
Demand	Sortie (avant limitation de vitesse)	Eng	
MeasPos	Position mesurée (pour affichage)	Eng	
OP	Sortie station manuelle	Eng	
HR_OP	Echelle haute (bargraphe sortie & valeur sécurité intégrée)	Eng	
LR_OP	Echelle basse (bargraphe sortie & valeur sécurité intégrée)	Eng	
HL_OP	Seuil haut de la sortie	Eng	
LL_OP	Seuil bas de la sortie	Eng	
InitDmnd	Valeur pour Demand au début de la tâche	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
UpRate	Seuil de vitesse montant	Eng	
DnRate	Seuil de vitesse descendant	Eng	
TimeBase	Spécifie les unités de temps de limitation de vitesse	Menu	
Dis_DP	Position du point décimal	0 - 4	
Options		ABCD hex	
SelTrack	Sélection mode poursuite	V/F	1
SelHold	Sélection mode maintien	V/F	2
SelMan	VRAI sélectionne manuel ; FAUX sélectionne automatique	V/F	4
SelRate	Validation limitation de vitesse	V/F	8
Dis_M_RL	Invalidation modifications par bouton-poussoir "M"	V/F	1
ForceOP	Validation action à sécurité intégrée	V/F	2
FailSafe	Sélection valeur sécurité intégrée OP (VRAI=HR ou LR)	V/F	4
SafeHigh	Condition "sûre" est à l'état haut	V/F	8
FlashOP	Clignotement du bargraphe OP	V/F	1
SelOnOff	Sélection mode tout or rien (demande soit HL ou LL)	V/F	2
InDmdRst	Valeur Demand reprise tâche (pas mise sous tension)	V/F	4
InDmdPwr	Valeur Demand mise sous tension	V/F	8
UBarDisp	Liaison champ UserBar au bargraphe sortie face avant	V/F	1
MPosDisp	Liaison champ MeasPos au bargraphe sortie face avant	V/F	2
InDmdOP	OP copie Demand relance tâche & mise sous tension	V/F	4

suite...

...suite

Paramètre	Fonction	Unités	Etat
Status		(ABC)D Hex	
NotAuto	Pas en mode auto (càd, en maintien, poursuite ou manuel)	V/F	<div> <div>1</div> <div>2</div> <div>4</div> <div>8</div> </div> D
UserBar	Contrôle direct schéma des segments du bargraphe sortie	(A)BCD hex	
Bit0	Segment 0 bargraphe allumé (plus à gauche)	V/F	1
Bit1	Segment 1 bargraphe allumé	V/F	2
Bit2	Segment 2 bargraphe allumé	V/F	4
Bit3	Segment 3 bargraphe allumé	V/F	8
Bit4	Segment 4 bargraphe allumé	V/F	1
Bit5	Segment 5 bargraphe allumé	V/F	2
Bit6	Segment 6 bargraphe allumé	V/F	4
Bit7	Segment 7 bargraphe allumé	V/F	8
Bit8	Segment 8 bargraphe allumé	V/F	1
Bit9	Segment 9 bargraphe allumé (plus à droite)	V/F	2
			4
			8
			B

Table 5-10 Paramètres du bloc MAN_STAT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (MANUAL/AUTO/TRACK/HOLD/INIT - MANUEL/AUTO/POURSUITE/MAINTIEN/INIT). Mode de fonctionnement actif (lecture uniquement). Les paramètres *Options* correspondants (*SelMan*, *SelTrack* et *SelHold*) permettent de sélectionner le mode. Le mode INIT est indiqué brièvement à la mise sous tension ou à la mise en route de la tâche si *InitDmnd* a été configuré pour fournir une valeur initiale *Demand*.

Fallback (Reprise). (MANUAL/AUTO/TRACK - MANUEL/AUTO/POURSUITE). Indique le mode de fonctionnement suivant (supprimé).

PV. Entrée de la station manuelle (variable procédé).

Track (Poursuite). Contrôle *Demand* et *OP* en mode poursuite.

Demand (Demande). Sortie de la station manuelle avant toute limitation de vitesse affichée sur le bargraphe de sortie de la face avant du T600 lorsque les boutons-poussoirs M, A ou R sont actionnés.

MeasPos. Valeur de “position mesurée” générée en externe affichée sur le bargraphe de sortie de la face avant du T600, soumise à l’état de *MPosDisp* et *UBarDisp*.

OP. Valeur de sortie de la station affichée sur le bargraphe de sortie de la face avant du T600, soumise à l'état de *MPosDisp* et *UBarDisp*.

HR_OP, LR_OP. Echelle haute et basse pour l'affichage du bargraphe de sortie du T600. *HR_OP* et *LR_OP* définissent respectivement les affichages de 100 % et 0 %. Ces paramètres, ainsi que *SafeHigh*, définissent également la valeur à "sécurité intégrée" adoptée par *Demand* lorsque le bloc est mis en mode manuel par une entrée à front montant dans *Options.SelMan*.

HL_OP, LL_OP. Seuil de sortie haut et bas agissant à la fois sur *OP* et *Demand*.

InitDmnd. Valeur adoptée par *Demand* (soumise au seuils haut/bas) à la mise sous tension ou à la reprise d'une tâche, si les paramètres *InDmndPwr* ou *InDmndRst* sont VRAIS. Cette initialisation du champ *Demand* peut être utilisée pour assurer une mise en route sans à-coups.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

UpRate. Permet de spécifier la vitesse maximale d'augmentation d'*OP* (c'est à dire limite de vitesse) en unités physiques/temps. Les unités de temps sont spécifiées par la paramètre *TimeBase*. La limitation est uniquement appliquée lorsque *SelRate* est VRAI et *SelTrack* est FAUX (c'est à dire pas en mode poursuite).

DnRate. Permet de spécifier la vitesse maximale de diminution d'*OP* (c'est à dire limite de vitesse) en unités physiques/temps. Les unités de temps sont spécifiées par la paramètre *TimeBase*. La limitation est uniquement appliquée lorsque *SelRate* est VRAI et *SelTrack* est FAUX (c'est à dire pas en mode poursuite).

TimeBase (Base de temps). (sec/min). Permet de sélectionner les unités de temps pour les fonctions de limitation de vitesse *UpRate* et *DnRate*.

Dis_DP. Permet de spécifier la position du point décimal sur l'affichage à 5 chiffres de la face avant du T600. La plage admissible est de 0 à 4, ce qui définit le nombre de chiffres affichés après le point décimal.

Options. Les informations suivantes sont données en plus de celles de la table 5-10.

■ **ForceOP, FailSafe, SafeHigh.** Si *ForceOp* est VRAI, une transition *SelMan* FAUX à VRAI fait passer le bloc en mode "sûr" pendant une itération uniquement avant la reprise du mode manuel normal. Dans ce mode transitoire, *Demand* adopte une valeur qui dépend des états de *FailSafe* et *SafeHigh* suivant les indications de la table ci-après.

	FailSafe = FALSE	FailSafe = TRUE
SafeHigh = FALSE	Demand = OP	Demand = LR_OP*
SafeHigh = TRUE	Demand = OP	Demand = HR_OP*

* Limité par *HL_OP, LL_OP*

Cette fonction permet de mettre en oeuvre des schémas de boucles “de type manuel/bas sur circuit ouvert de la mesure PV” ou des schémas similaires.

Status (Etat).

■ **NotAuto.** Indique que la station manuelle ne fonctionne pas en mode automatique.

UserBar. Les dix entrées logiques *UserBar.Bit0* (segment le plus à gauche) à *UserBar.Bit9* (segment le plus à droite) permettent au schéma de boucles de contrôler directement et individuellement les segments du bargraphe de sortie de la face avant du T600. *UserBar* est relié à la face avant réelle uniquement si le bit *Options.UBarDisp* est VRAI et si la tâche utilisateur associée du bloc est celle sélectionnée pour l’affichage principal de la boucle.

MODE : BLOC MODE

Fonction du bloc

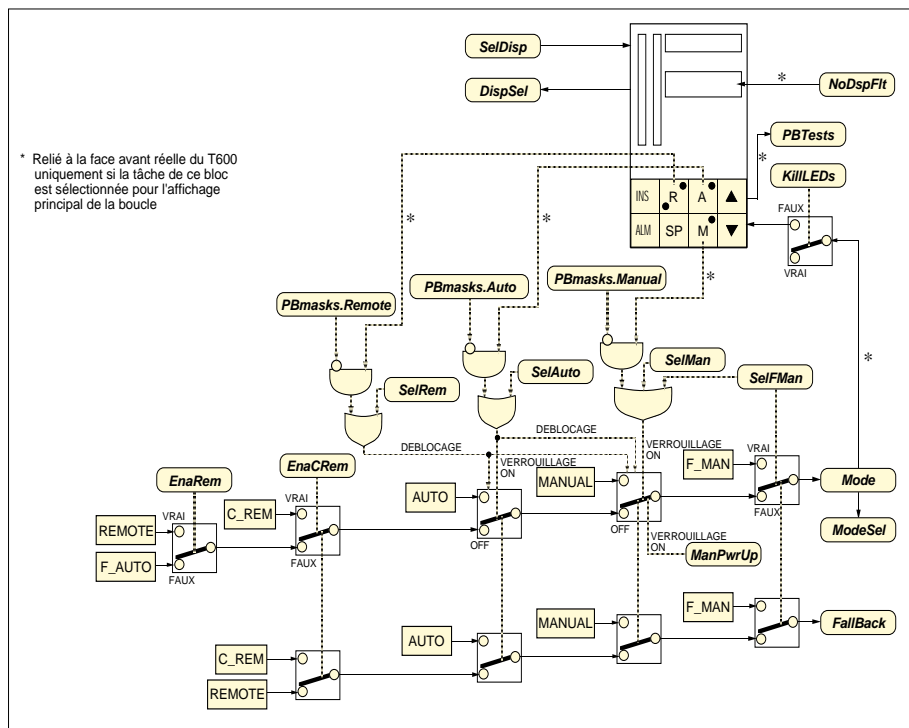


Figure 5-14 Schéma du bloc MODE

Voir le schéma de la figure 5-14. Le bloc MODE est destiné à être utilisé comme partie de la combinaison des blocs de régulation SETPOINT/3_TERM/MAN_STAT/MODE, ce qui donne plus de souplesse qu'un simple bloc PID.

Le bloc est basé sur un algorithme de régulation à mode verrouillant et à classement par priorité schématisé sur la figure 5-14 par une série de commutateurs, dont certains sont verrouillants. La position de chaque commutateur le long du cheminement jusqu'au paramètre *Mode* détermine la priorité du mode de fonctionnement sélectionné par ce commutateur. Par ex. le mode F_MAN (manuel forcé), si sélectionné par *SelFMan* qui est VRAI, annule tous les autres modes dans la mesure où son commutateur est le plus proche du paramètre *Mode*. A l'inverse, les modes F_AUTO (auto forcé) et REMOTE (déporté) ont la priorité la plus basse, dans la mesure où leur commutateur est le plus éloigné de *Mode* et peut donc être annulé par tous les autres.

Les commutateurs MANUAL et AUTO sont verrouillants, c'est à dire qu'ils restent à 1 même si la situation VRAI qui les a mis à 1 passe à FAUX. Dans ce cas, un commutateur MANUAL verrouillé peut être débloqué en sélectionnant soit AUTO ou "remote" et MANUAL et AUTO sont débloqués en sélectionnant "remote".

Le mode de fonctionnement en vigueur est indiqué par *Mode* et est disponible comme sortie par l'intermédiaire des bits *ModeSel*. Le mode de reprise, maintenu dans *FallBack*, n'est pas disponible comme sortie et est uniquement utilisé pour des raisons de contrôle. La source de *FallBack* est schématisée dans la figure 5-14 comme une seconde série de commutateurs liés aux commutateurs de sélection de mode correspondants.

Lorsque la tâche utilisateur associée a été affectée à l'affichage principal de la boucle, le bloc MODE contrôle les LED des boutons-poussoirs **R**, **A** et **M** de la face avant du T600, et les caractères du mode sous le bargraphe d'écart de la boucle. Les LED des boutons-poussoirs peuvent être invalidés par l'intermédiaire de l'entrée *KilledLEDs* (dans le champ binaire *Options*). Les masques des boutons-poussoirs **R**, **A** et **M** permettent d'empêcher des changements de mode par l'opérateur par l'intermédiaire des entrées *PBmasks*. Tous les huit boutons-poussoirs de la face avant sont verrouillés sur la base d'une exécution par tâche, de sorte que des appuis éphémères sur les boutons ne sont pas ignorés par la tâche. Ces appuis verrouillés sont également disponibles comme sorties par l'intermédiaire du champ binaire *PBTests* pour être utilisé par d'autres blocs dans la tâche.

Paramètres du bloc















Les symboles utilisés dans la table 5-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Mode. (MANUAL/F_MAN/AUTO/C_REM/REMOTE/F_AUTO - MANUEL/MANUEL FORCE/AUTO/C_REM/DEPORTE/AUTO FORCE). Le mode de fonctionnement sélectionné du bloc. Manual et Auto peuvent être sélectionnés directement par l'intermédiaire du champ *Mode*, mais les autres sont en lecture uniquement. Dans l'ordre de priorité, les modes possibles sont les suivants: F_MAN (priorité la plus haute), MANUAL, AUTO, C_REM, REMOTE et F_AUTO (priorité la plus basse):

- **F_MAN (MANUEL FORCE)** est sélectionné uniquement si *SelMode.SelFMan* est VRAI et annule tous les autres modes.
- **MANUAL (MANUEL).** Ce mode peut être sélectionné en écrivant directement dans *Mode*. Il peut également être sélectionné soit en rendant *SelMode.SelMan* VRAI ou en appuyant sur le bouton-poussoir **M** de la face avant (à condition que *PBMask.Manual* soit FAUX). Le mode manuel est "verrouillé" lorsqu'il est sélectionné et le reste jusqu'à ce qu'il soit débloqué (en sélectionnant AUTO ou un mode déporté). Noter que le mode manuel est également verrouillé lorsque *SelMode.SelFMan* est VRAI, mais est annulé par F_MAN jusqu'à ce que *SelFMan* repasse à FAUX, à ce stade, MANUAL reprend la main.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement	Menu	
FallBack	Mode de reprise	Menu	
SelMode		(AB)CD hex	 
EnaRem	Validation mode déporté	V/F — 1	D
EnaCRem	Validation mode déporté ordinateur	V/F — 2	
SelRem	Sélection mode déporté	V/F — 4	
SelAuto	Sélection mode auto	V/F — 8	
SelMan	Sélection mode manuel	V/F — 1	C
SelFMan	Sélection mode manuel forcé	V/F — 2	
		4	
		8	
Options		(ABC)D hex	
SelDisp	Forçage de cette boucle à l'affichage	V/F — 1	D
KillLEDs	Invalidation LED M, A, & R	V/F — 2	
ManPwrUp	Forçage en manuel à la mise sous tension	V/F — 4	
		8	
PBmasks			
Remote	Invalidation bouton-poussoir sélection de déporté	V/F —	  
Auto	Invalidation bouton-poussoir sélection d'auto	V/F —	
Manual	Invalidation bouton-poussoir sélection de manuel	V/F —	
		0 0 0 0 0 0 0 0	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défaut module E/S/Alim. transmetteur fausse	V/F	
Combined	Alarme combinée du bloc	V/F	
ModeSel		(AB)CD Hex	 
RemSel	Mode déporté sélectionné	V/F — 1	D
AutoSel	Mode auto sélectionné	V/F — 2	
MorFMsel	Mode manuel ou FM sélectionné	V/F — 4	
FAutoSel F.	Mode auto (retour à R) sélectionné	V/F — 8	
CRemSel	Mode déporté ordinateur sélectionné	V/F — 1	C
		2	
		4	
		8	
Status		(ABC)D Hex	 
DispSel	Cette boucle à l'affichage	V/F — 1	D
		2	
		4	
		8	

suite...

...suite


Paramètre	Fonction	Unités	Etat
PBtests			
Down	Bouton-poussoir décrémentation enfoncé	V/F	
Up	Bouton-poussoir incrémentation enfoncé	V/F	
Remote	Bouton-poussoir R enfoncé	V/F	
Auto	Bouton-poussoir A enfoncé	V/F	
Manual	Bouton-poussoir M enfoncé	V/F	
SetPoint	Bouton-poussoir SP enfoncé	V/F	
Alarm	Bouton-poussoir ALM enfoncé	V/F	
Inspect	Bouton-poussoir INS enfoncé	V/F	

Table 5-11 Paramètres du bloc MODE

- **AUTO.** Ce mode peut être sélectionné en écrivant directement dans *Mode*. Il peut également être sélectionné soit en rendant *SelMode.SelAuto* VRAI ou en appuyant sur le bouton-poussoir **A** de la face avant (à condition que *PBMask.Auto* soit FAUX). Le mode automatique “est verrouillé” lorsqu’il est sélectionné et le reste jusqu’à ce qu’il soit débloqué (en sélectionnant un mode déporté).
- **C_REM (DEPORTE ORDINATEUR).** Ce mode est uniquement actif lorsque *SelMode.EnaCRem* est VRAI, et qu’aucun mode de priorité supérieure n’est actif. Il faut rendre *SelMode.SelRem* VRAI ou appuyer sur le bouton-poussoir **R** de la face avant (à condition que *PBMask.Remote* soit FAUX) pour débloquer les modes AUTO et MANUAL et permettre à C_REM de fonctionner.
- **REMOTE (DEPORTE).** Ce mode est uniquement actif si *SelMode.EnaRem* est VRAI, *.EnaCRem* FAUX et qu’aucun mode de priorité supérieure n’est actif.
- **F_AUTO (AUTO FORCE).** Ce mode est uniquement actif si les bits *SelMode* sont FAUX et qu’AUTO ou MANUAL ne sont pas verrouillés.

FailBack (Reprise). (MANUAL/F_MAN/AUTO/C_REM/REMOTE - MANUEL/F_MAN/AUTO/C_REM/DEPORTE). Le mode de reprise du bloc est uniquement utilisé pour des raisons de contrôle. La figure 5-14 montre schématiquement comment le mode de reprise est calculé suivant la valeur de *Mode*.

SelMode. Champ binaire permettant de valider/sélectionner la valeur *Mode* du bloc. Voir les explications dans le paragraphe *Mode* ci-dessus sur le fonctionnement de ces bits, ainsi que la figure 5-14 et la table 5-11. Le mode peut également être modifié (en AUTO ou MANUAL) en écrivant directement dans le champ *Mode*, à condition que tous les bits de sélection de mode *SelMode* (*SelRem*, *SelAuto*, *SelMan* ou *SelFMan*) soient FAUX.

En raison des caractéristiques de verrouillage/débloquage des “commutateurs de sélection” MANUAL et AUTO, il suffit que les bits *SelRem*, *SelAuto*, *SelMan* et *SelFMan* soient VRAI transitoirement pour provoquer des changements de mode.

Noter qu’un bouton-poussoir de mode de la face avant d’un T600 ne peut provoquer de changement de mode que s’il n’est pas masqué et si tous les bits de sélection de mode *SelMode* (*SelRem*, *SelAuto*, *SelMan* ou *SelFMan*) sont FAUX. Des écritures de

communication dans ces quatre champs sont verrouillants (supprimés au moment de la mise à jour du bloc), de sorte que les écritures de changement de mode par un système de supervision sont traités même lorsque ces champs sont reliés à d'autres blocs dans la base de données.

Options. Champ binaire permettant de sélectionner différentes options d'affichage de la face avant du T600 (et le mode adopté à la mise sous tension).

- **SelDisp.** VRAI force la tâche utilisateur à laquelle le bloc est affecté à occuper l'affichage principal de la boucle sur la face avant du T600. Noter que si *SelDisp* est VRAI pour plus d'une tâche utilisateur, c'est la tâche à la priorité la plus haute qui est affichée. (La tâche utilisateur 1 a la priorité la plus haute et la tâche utilisateur 4 la plus basse).
- **KillLEDs.** VRAI invalide les LED des boutons-poussoirs **R**, **A** et **M** de la face avant du T600 (uniquement si cette tâche utilisateur occupe l'affichage principal de la boucle).
- **ManPwrUp.** VRAI force le bloc à adopter le mode MANUAL à la mise sous tension du T600.
- **NoDspFlt.** VRAI invalide le filtre d'une constante de temps de 1 seconde du filtre du premier ordre fonctionnant sur l'affichage à 5 chiffres de la face avant du T600. Cette fonction est utile lorsqu'un point de consigne est affiché, qui à l'inverse d'une variable procédé, ne nécessite pas, en général, l'action de filtrage.

PBmasks. Champ binaire pour masquer, c'est à dire inhiber l'action des boutons-poussoirs de changement de mode **R**(emote), **A**(uto) et **M**(anual) de la face avant du T600. Un bit *Remote*, *Auto* ou *Manual* VRAI masque le bouton-poussoir correspondant, lorsque la tâche utilisateur contenant le bloc MODE est affectée à l'affichage principal de la boucle. Si la boucle n'est pas affichée, les boutons-poussoirs n'ont aucun effet.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **F_MANUAL.** VRAI si le bloc est en mode manuel forcé (F_MAN).
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

ModeSel. Champ binaire indiquant le mode actif du bloc, c'est à dire la valeur du paramètre *Mode*. Voir table 5-11. Seul un bit *ModeSel* peut être actif à la fois. Noter que les modes manuel et manuel forcé ne sont pas indiqués séparément.

Status (Etat). Champ binaire indiquant (*DispSel* VRAI) si la tâche utilisateur contenant ce bloc MODE occupe l'affichage principal de la boucle sur la face avant du T600.

PBTests. Champ binaire indiquant (par des bits VRAI) lequel des huit boutons-poussoirs de la face avant du T600 sont enfoncés. Voir table 5-11. *PBTests* ne fonctionne que lorsque la tâche utilisateur du bloc MODE occupe l'affichage principal de la boucle sur le T600; sinon, tous les bits restent FAUX.

PID_CONN : BLOC DE LIAISONS PID

Fonction du bloc

Le bloc PID_CONN présente le même modèle (menu de spécifications du bloc) que le bloc PID, mais est utilisé comme un “bloc face avant” pour la combinaison SETPOINT/3_TERM/MAN_STAT/MODE. Le bloc PID_CONN s’associe automatiquement aux blocs appropriés dans la même boucle; il peut alors être utilisé comme un bloc cache dans un T1000 pour servir d’affichage de face avant.

La plupart des champs PID_CONN sont des copies directes des champs équivalents dans les blocs associés SETPOINT/3_TERM/MAN_STAT ou MODE. Les écritures ou les tentatives d’écriture dans ces champs sont redirigées dans les champs correspondants des blocs sources réels. Certains champs PID_CONN sont des dérivées logiques plus complexes des champs sources. Noter que si le bloc PID_CONN ne peut trouver les sources pour ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale des données. La table 5-12 donne la liste des sources des champs PID_CONN.

NOTA. Liaisons. Lorsque vous utilisez un bloc PID_CONN pour concentrer des blocs SETPOINT/3_TERM/MAN_STAT/MODE, il faut établir des liaisons cohérentes. Reliez les liaisons du schéma de boucles soit au bloc PID_CONN ou aux vrais blocs sources, *mais pas aux deux*. Le même principe s’applique également aux liaisons des copies en mémoire cache de blocs sources déportés, en particulier lorsqu’il y a des champs binaires. Les liaisons aux bits individuels de champs binaires dans différents emplacements peuvent entraîner la défaillance de certaines liaisons.

Traitement des alarmes

Bien que les valeurs de HAA, LAA, HDA et LDA soient des copies des valeurs équivalentes dans le bloc SETPOINT, les alarmes sont traitées localement dans le bloc PID_CONN, ce qui permet de sélectionner différentes priorités d’alarme et de traiter les alarmes dans un seul bloc.

Champ PID_CONN	Source
Mode	IF MAN_STAT.Mode = Hold or Track, THEN Mode = MAN_STAT.Mode ELSE IF MODE.Mode = C_REM, THEN Mode = REMOTE ELSE Mode = MODE.Mode
FallBack	IF MAN_STAT.Mode = NOT(Hold ou Track) THEN IF MODE.FallBack = C_REM, THEN FallBack = REMOTE ELSE FallBack = MODE.FallBack ELSE IF MODE.Mode = C_REM, THEN FallBack = REMOTE ELSE FallBack = MODE.Mode
PV	SETPOINT.PV
SP	SETPOINT.SP
OP	MAN_STAT.Demand
SL	SETPOINT.SL

suite...

...Table 5-12 suite

Champ	PID_CONN	Source
TrimSP		SETPOINT.TrimSP
RemoteSP		SETPOINT.RemoteSP/ComRemSP (suivant Mode)
Track		MAN_STAT.Track
HR_SP		SETPOINT.HR_SP
LR_SP		SETPOINT.LR_SP
HL_SP		SETPOINT.HL_SP
LL_SP		SETPOINT.LL_SP
HR_OP		MAN_STAT.HR_OP
LR_OP		MAN_STAT.LR_OP
HL_OP		MAN_STAT.HL_OP
LL_OP		MAN_STAT.LL_OP
HAA		SETPOINT.HAA
LAA		SETPOINT.LAA
HDA		SETPOINT.HDA
LDA		SETPOINT.LDA
TimeBase		3_TERM.TimeBase
XP		3_TERM.XP
TI		3_TERM.TI
TD		3_TERM.TD
Options	.InvPID	3_TERM.Options.InvPID
	.SL_Track	SETPOINT.Options.EnbTrack
	.IntBalSL	NOT(3_TERM.Options.DisSPbal)
	.IntBal	3_TERM.Options.SPbal
SelMode	.SelHold	MAN_STAT.Options.SelHold
	.SelTrack	MAN_STAT.Options.SelTrack
	.SelRem	MODE.SelMode.SelRem
	.EnaRem	MODE.SelMode. (EnaRem OU EnaCRem)
	.SelAuto	MODE.SelMode.SelAuto
	.SelMan	MODE.SelMode.SelMan
	.SelFMan	MODE.SelMode.SelFMan
ModeSel	.EnaRem	NOT(PID_CONN.ModeSel.HoldSel OR PID_CONN.ModeSel.TrackSel OR PID_CONN.ModeSel.ManSel OR (PID_CONN.FallBack = Manual))
'Sel' bits	.HoldSel	PID_CONN.SelMode.SelHold
	.TrackSel	PID_CONN.SelMode.SelTrack
	.RemSel	MODE.ModeSel.(RemSel OU CRemSel)
	.AutoSel	MODE.ModeSel.AutoSel
	.ManSel	MODE.ModeSel.MorFMsel AND (MODE.Mode = Manual)
	.FAutoSel	MODE.ModeSel.FAutoSel
	.FManSel	MODE.ModeSel.MorFMsel AND (MODE.Mode = Forced Manual)

suite...

...Table 5-12 suite

Champ PID_CONN	Source
ModeAct .NotRem	NOT(PID_CONN.Mode = REMOTE)
.HoldAct	IF ModeSel.HoldSel = VRAI THEN ModeAct.HoldAct = VRAI, et le reste de ces sept bits 'Act' sont FAUX, ELSE IF ModeSel.TrackSel = VRAI THEN ModeAct.TrackAct = VRAI, et le reste de ces sept bits 'Act' sont FAUX, ELSE ModeAct.('Act' bits) = ModeSel.('Sel' bits)
.TrackAct	
.RemAct	
.AutoAct	
.ManAct	
.FAutoAct	
.FManAct	
FF_PID	3_TERM.FF_PID
FB_OP	3_TERM.FB_OP

Table 5-12 Sources des champs du bloc PID_CONN

PID_LINK: BLOC DE CHAINAGE PID

Fonction du bloc

Le bloc PID_LINK présente presque le même modèle (menu de spécifications du bloc) que le blocs PID_CONN et PID, mais est utilisé comme un “bloc face avant” pour la combinaison des blocs SETPOINT/3_TERM/MAN_STAT/MODE. Le bloc PID_LINK est associé à ces quatre blocs de régulation en saisissant leur nom de repère dans quatre champs spéciaux - *SPblock*, *T3block*, *MSblock* et *MDblock*. Le bloc peut alors être mis en mémoire cache dans un ordinateur de supervision pour servir d’affichage de face avant.

La plupart des champs PID_LINK sont des copies directes des champs équivalents dans les blocs associés SETPOINT/3_TERM/MAN_STAT ou MODE. Les écritures ou les tentatives d’écriture dans ces champs sont redirigées dans les champs correspondants des blocs sources réels. Certains champs PID_LINK sont des dérivées logiques plus complexes des champs sources. Noter que si le bloc PID_LINK ne peut trouver les sources pour ses champs, ces champs ne sont pas écrasés et sont disponibles pour la sauvegarde locale de données. La table 5-13 donne la liste des sources des champs PID_CONN.

NOTA. Liaisons. Lorsque vous utilisez un bloc PID_LINK pour concentrer des blocs SETPOINT/3_TERM/MAN_STAT/MODE, il faut établir des liaisons cohérentes. Reliez les liaisons du schéma de boucles soit au bloc PID_LINK ou aux vrais blocs sources, *mais pas aux deux*. Le même principe s’applique également aux liaisons des copies en mémoire cache de blocs sources déportés, en particulier s’il y a des champs binaires. Les liaisons aux bits individuels de champs binaires dans différents emplacements peuvent entraîner la défaillance de certaines liaisons.

Traitement des alarmes

Bien que les valeurs de *HAA*, *LAA*, *HDA* et *LDA* soient des copies des valeurs équivalentes dans le bloc SETPOINT, les alarmes sont traitées localement dans le bloc PID_LINK, ce qui permet de sélectionner différentes priorités d’alarme et de traiter les alarmes dans un seul bloc.

Champ PID_LINK	Source
Mode	IF MAN_STAT.Mode = Hold or Track, THEN Mode = MAN_STAT.Mode ELSE IF MODE.Mode = C_REM, THEN Mode = REMOTE ELSE Mode = MODE.Mode
FallBack	IF MAN_STAT.Mode = NOT(Hold ou Track) THEN IF MODE.FallBack = C_REM, THEN FallBack = REMOTE ELSE FallBack = MODE.FallBack ELSE IF MODE.Mode = C_REM, THEN FallBack = REMOTE ELSE FallBack = MODE.Mode
PV	SETPOINT.PV
SP	SETPOINT.SP
OP	MAN_STAT.Demand

Table 5-13 suite...

...Table 5-13 suite

Champ	PID_LINK	Source	
SL		SETPOINT.SL	
TrimSP		SETPOINT.TrimSP	
RemoteSP		SETPOINT.RemoteSP/ComRemSP (suivant Mode)	
Track		MAN_STAT.Track	
HR_SP		SETPOINT.HR_SP	
LR_SP		SETPOINT.LR_SP	
HL_SP		SETPOINT.HL_SP	
LL_SP		SETPOINT.LL_SP	
HR_OP		MAN_STAT.HR_OP	
LR_OP		MAN_STAT.LR_OP	
HL_OP		MAN_STAT.HL_OP	
LL_OP		MAN_STAT.LL_OP	
HAA		SETPOINT.HAA	
LAA		SETPOINT.LAA	
HDA		SETPOINT.HDA	
LDA		SETPOINT.LDA	
TimeBase		3_TERM.TimeBase	
XP		3_TERM.XP	
TI		3_TERM.TI	
TD		3_TERM.TD	
Options	.InvPID	3_TERM.Options.InvPID	
	.SL_Track	SETPOINT.Options.EnbTrack	
	.IntBalSL	NOT(3_TERM.Options.DisSPbal)	
	.IntBal	3_TERM.Options.SPbal	
SelMode	.SelHold	MAN_STAT.Options.SelHold	
	.SelTrack	MAN_STAT.Options.SelTrack	
	.SelRem	MODE.SelMode.SelRem	
	.EnaRem	MODE.SelMode. (EnaRem OU EnaCRem)	
	.SelAuto	MODE.SelMode.SelAuto	
	.SelMan	MODE.SelMode.SelMan	
	.SelFMan	MODE.SelMode.SelFMan	
ModeSel	.EnaRem	NOT(PID_LINK.ModeSel.HoldSel OR PID_LINK.ModeSel.TrackSelOR PID_LINK.ModeSel.ManSel OR (PID_LINK.FallBack = Manuel))	
'Sel' bits	{	.HoldSel	PID_CONN.SelMode.SelHold
		.TrackSel	PID_LINK.SelMode.SelTrack
		.RemSel	MODE.ModeSel.(RemSel OU CRemSel)
		.AutoSel	MODE.ModeSel.AutoSel
		.ManSel	MODE.ModeSel.MorFMsel AND (MODE.Mode = Manuel)
		.FAutoSel	MODE.ModeSel.FAutoSel
		.FManSel	MODE.ModeSel.MorFMsel AND (MODE.Mode = Manuel forcé)

Table 5-13 suite...

...Table 5-13 suite

Champ	PID_LINK	Source
ModeAct	.NotRem	NOT(PID_LINK.Mode = REMOTE)
	.HoldAct	IF ModeSel.HoldSel = VRAI THEN ModeAct.HoldAct = VRAI, et le reste de ces sept bits 'Act' sont FAUX, ELSE IF ModeSel.TrackSel = VRAI THEN ModeAct.TrackAct = VRAI, et le reste de ces sept bits 'Act' sont FAUX, ELSE ModeAct.('Act' bits) = ModeSel.('Sel' bits)
	.TrackAct	
	.RemAct	
	.AutoAct	
	.ManAct	
	.FAutoAct	
	.FManAct	
FF_PID		3_TERM.FF_PID
FB_OP		3_TERM.FB_OP
SPblock		Nom de repère du bloc SETPOINT [Alphanumérique]
T3block		Nom de repère du bloc 3_TERM [Alphanumérique]
MSblock		Nom de repère du bloc MAN_STAT [Alphanumérique]
MDblock		Nom de repère du bloc MODE [Alphanumérique]

Table 5-13 Sources des champs du bloc PID_LINK

[Page laissée intentionnellement blanche]

Chapitre 6 BLOCS DE FONCTION TEMPS

SEQ : BLOC SEQUENCE

Fonction du bloc

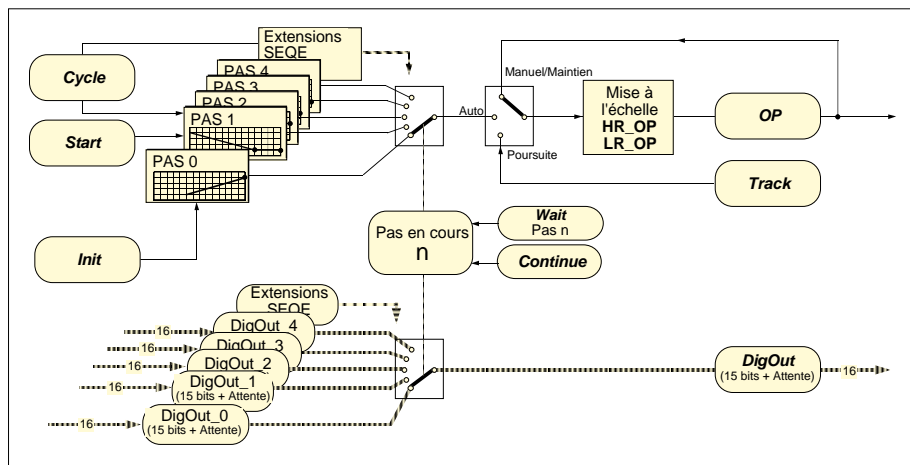


Figure 6-1 Schéma du bloc SEQ

Voir la figure 6-1. Un bloc séquence (SEQ) permet de profiler les points de consigne analogiques et/ou de mettre en séquence dans le temps 15 voies de sortie logique comme une série de quatre *pas*, plus un pas d'initialisation. Davantage de pas peuvent être ajoutés à la séquence en multiples de huit en utilisant les blocs d'extension de séquence (SEQE). Les blocs SEQE sont contrôlés à partir du bloc principal SEQ, et contiennent la base de données du paramètre pour chaque pas. Les séquences peuvent être cyclées, interreliées et mises en cascade en série.

Chaque pas définit une pente de point de consigne, une valeur finale (cible) et un palier, ainsi que l'état de 15 sorties logiques pour ce pas. Les sorties logiques peuvent également être écrites (par l'intermédiaire des champs *DigOut_n*) à partir de l'écran de conduite ou par l'intermédiaire des liaisons du schéma de boucles.

Le bloc séquence dispose d'un contrôle Auto/Manuel/Poursuite avec reprise en automatique. Les verrouillages séquence contrôlent l'exécution de la séquence, ce qui permet une attente et continuation conditionnelle à chaque pas. L'ensemble de la séquence peut être recyclé automatiquement ou conditionnellement à la détection d'une entrée.

Modes de fonctionnement. La séquence est uniquement exécutée, lorsque le bloc fonctionne en mode automatique. Dans les modes manuel, poursuite ou maintien, les fonctions de rampe ou de temps sont interrompues, ce qui permet le calcul de la valeur de sortie *OP* à partir d'autres sources. De cette manière, *OP* peut s'éloigner de la valeur définie par la séquence.

Toutefois, ces actions ne font que simplement différer la durée en temps réel de la séquence, dans la mesure où les vitesses de la rampe sont définies comme une pente et une valeur cible pour chaque pas, et l'horloge de palier est figée. L'exécution normale de la séquence reprend au moment "différé", dès que le mode de fonctionnement est remis en automatique.

Priority 1 (Priorité 1).

Mode maintien (HOLD). Le mode maintien est sélectionné à partir de l'entrée de *SelHold*. En mode maintien, *OP* est verrouillé et ne peut être modifié.

Priority 2 (Priorité 2).








Mode poursuite (TRACK). Le mode poursuite est sélectionné à partir de l'entrée de *SelTrack*. En mode poursuite, *OP* est contrôlé par la poursuite.

Priority 3 (Priorité 3).

Mode Auto/Manuel (Auto/Manual). En mode automatique, *OP* est contrôlé par la fonction de séquence. En mode manuel, *OP* peut être ajusté indépendamment.













Paramètres du bloc

Les symboles utilisés dans la table 6-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement en cours	Menu	
FallBack	Mode de fonctionnement supprimé	Menu	
MaxSteps	Numéro du dernier pas en séquence	Entier	
CurrStep	Pas actif en cours	Entier	
OP	Sortie bloc	Eng	
HR_OP	Echelle haute sortie	Eng	
LR_OP	Echelle basse sortie	Eng	
Slope	Vitesse rampe du pas en cours		
Target	Valeur cible du pas en cours	Eng	
Slope_0	Vitesse rampe pas 0		
Slope_1 à Slope_4	Vitesses rampe pas 1 à 4		
StartVal	Valeurs cible pas 0	Eng	
EndVal_1 à EndVal_4	Valeurs cible pas 1 à 4	Eng	

... suite

suite...

Paramètre	Fonction	Unités	Etat
Alarms			  
Software	Données corrompues/Défaut de communication		V/F
Combined	Alarme combinée du bloc	V/F	
Extend	Nom du bloc d'extension	Alphanumérique	
TimeBase	Unités de temps pour paramètres pente, palier	Menu	
Track	Définit la sortie (OP) en mode poursuite	Eng	
StatIn	Etat séquence d'entrée	Champ binaire	
Init	Initialis. de la séquence (Force retour au pas 0)	V/F	
Start	Lancement séquence (pas 0 à pas 1)	V/F	
Continue	Continuation après le drapeau attente	V/F	
Cycle	Cycle continu de la séquence	V/F	
Track_0	Sélectionne le mode poursuite au pas 0	V/F	
SelHold	Sélection du mode maintien	V/F	
SelTrack	Sélection mode poursuite	V/F	
		0 0 0 0 0 0 0 0	
StatOut	Etat séquence de la sortie	Champ binaire	
Ramping	Mise en rampe du pas	V/F	
StepDone	Fin du pas (attente) ou de la séquence	V/F	
CycDone	Séquence terminée (pas le cyclage)	V/F	
NotTrack	Mode poursuite pas actif	V/F	
		0 0 0 0 0 0 0 0	
TimeLeft	Durée restante du palier du pas courant	Eng	
DigOut	Sorties logiques & champ attente	ABCD hex	
Bit0	Bit 0 sortie logique	V/F — 1	D
Bit1	Bit 1 sortie logique	V/F — 2	
Bit2	Bit 2 sortie logique	V/F — 4	
Bit3	Bit 3 sortie logique	V/F — 8	
Bit4	Bit 4 sortie logique	V/F — 1	C
Bit5	Bit 5 sortie logique	V/F — 2	
Bit6	Bit 6 sortie logique	V/F — 4	
Bit7	Bit 7 sortie logique	V/F — 8	
Bit8	Bit 8 sortie logique	V/F — 1	B
Bit9	Bit 9 sortie logique	V/F — 2	
BitA	Bit A sortie logique	V/F — 4	
BitB	Bit B sortie logique	V/F — 8	
BitC	Bit C sortie logique	V/F — 1	A
BitD	Bit D sortie logique	V/F — 2	
BitE	Bit E sortie logique	V/F — 4	
Wait	Etat champ attente	V/F — 8	
DigOut_n (n = 0 à 4)	Pas n sorties logiques et champ attente	ABCD hex	
Bit0	Bit 0 sortie logique	V/F — 1	D
Bit1	Bit 1 sortie logique	V/F — 2	
Bit2	Bit 2 sortie logique	V/F — 4	
Bit3	Bit 3 sortie logique	V/F — 8	

suite...

... suite


Paramètre	Fonction	Unités	Etat
Bit4	Bit 4 sortie logique	V/F — 1	C
Bit5	Bit 5 sortie logique	V/F — 2	
Bit6	Bit 6 sortie logique	V/F — 4	
Bit7	Bit 7 sortie logique	V/F — 8	
Bit8	Bit 8 sortie logique	V/F — 1	B
Bit9	Bit 9 sortie logique	V/F — 2	
BitA	Bit A sortie logique	V/F — 4	
BitB	Bit B sortie logique	V/F — 8	
BitC	Bit C sortie logique	V/F — 1	A
BitD	Bit D sortie logique	V/F — 2	
BitE	Bit E sortie logique	V/F — 4	
Wait	Etat champ attente	V/F — 8	
Dwell_n (n = 1 à 4)	Temps de palier pour le pas n	Eng	

Table 6-1 Paramètres du bloc SEQ

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (AUTO/MANUAL/TRACK/HOLD - AUTO/MANUEL/POURSUITE/MAINTIEN). Mode de fonctionnement en cours.

Fallback (Reprise). (AUTO/MANUAL - AUTO/MANUEL). Indique le mode de fonctionnement de retour (supprimé).

MaxSteps. Nombre total de pas dans la séquence (à l'exclusion du pas d'initialisation 0). Les séquences avec plus de quatre pas nécessitent les blocs d'extension SEQE, qui ajoutent des pas par multiples de huit. MaxSteps peuvent avoir n'importe quel nombre - pas nécessairement un multiple de 4 ou de 8.

CurrStep. Pas en cours. Indique le pas actif en cours dans la séquence. Ce nombre ne correspond pas au nombre de pas utilisés dans le bloc d'extension de séquence.

OP. Valeur de la sortie. La sortie analogique du bloc de séquence. Souvent appelé “point de consigne”, dans la mesure où un bloc de séquence est fréquemment utilisé comme un point de consigne déporté pour un régulateur. Le paramètre *OP* contrôle la sortie en mode manuel uniquement. En mode Auto et poursuite, *OP* est contrôlé par les entrées séquence et poursuite. En mode maintien, *OP* est verrouillé.

HR_OP, LR_OP. Echelles hautes & basses pour les objets graphiques *OP*. Ces paramètres permettent également de définir la valeur de la pente d'un pas dans le cas particulier où *Slope_n* est mis à zéro. (Voir *Slope_n*).

Slope. Pente du pas en cours (valeur absolue).

Target (Cible). Valeur finale du pas en cours.

Slope_0. Vitesse de rampe du pas 0. Ce paramètre est utilisé à la ré-initialisation d'une séquence, et définit la vitesse de rampe (valeur absolue) par rapport à la cible définie par *StartVal*. Si *Slope_0* est égal à 0.0, *OP* saute immédiatement (c'est à dire au cours d'une période d'échantillonnage) à la valeur *StartVal*.

StartVal. Valeur cible du pas 0 utilisée lorsqu'une séquence est ré-initialisée.

Slope_n. Vitesse de rampe. Définit la vitesse de rampe par rapport à la cible spécifiée par *EndVal* où *n* représente le numéro du pas du bloc. Les vitesses de rampe sont définies comme des gradients qui peuvent avoir des pentes positives ou négatives suivant la position de la valeur cible. Il est donc inutile d'assigner un signe (-/+) à cette valeur. Les unités de temps sont définies dans le champ *TimeBase* (Base de temps).

Si *Slope_n* est égal à 0.0, *OP* se rapproche de *EndVal* à la vitesse donnée par (*HR_OP* - *LR_OP*) en unités physiques par unité de *TimeBase*. Par exemple, si les valeurs par défaut du bloc sont *HR_OP* = 100.0, *LR_OP* = 0.0, *TimeBase* = sec et *Slope_1* = 0.0, la pente du pas 1 est égale à 100 unités physiques par seconde.

NOTA. Il faut éviter de définir une valeur trop petite pour *Slope_n*, pour que les erreurs d'arrondi arithmétique à chaque mise à jour du bloc ne laissent *OP* inchangé, c'est à dire avec un gradient effectif de zéro. *Exemple*: si la pente est de 0.1/hr et le temps de mise à jour du bloc de 0.1 sec, *OP* est incrémenté (ou décrémenté) de $0.1 \times 0.1 / 3600 = 2.8 \times 10^{-6}$ à chaque échantillonnage. Si *OP* = 1000, le résultat de l'addition est d'environ 1000.0000028, ce qui nécessite au moins 10 chiffres significatifs pour l'exprimer. En pratique, avec les 7 à 8 chiffres disponibles, *OP* resterait inchangé pour une valeur de 1000.

Si de très petites pentes sont inévitables, il ne faut pas utiliser de bloc SEQ. Utilisez une horloge pour incrémenter *OP* moins souvent que la vitesse d'échantillonnage par des quantités proportionnellement plus larges.

EndVal_n. Valeur cible. Cette valeur représente la valeur cible pour le pas, où *n* est le numéro de pas du bloc. Cette valeur peut être modifiée, même lorsque le pas est actif.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Extend (Extension). Permet d'entrer le nom du bloc d'extension SEQE. Une entrée dans ce champ automatiquement lie le contrôle et la base de données du bloc d'extension au bloc de séquence principal. Les autres blocs d'extension sont liés en utilisant les champs *Extend* dans le bloc précédent.

Timebase (Base de temps). (Sec/Min./Heures/jours). Permet de spécifier les unités de temps pour la pente et l'arrêt.

Track (Valeur de poursuite). Contrôle *OP* en mode poursuite.

Statin. Ce champ binaire permet de contrôler la séquence par l'intermédiaire des entrées logiques à partir du schéma.

- **Init.** Initialisation de la séquence. Initialise la séquence en forçant un retour au pas 0 quel que soit le mode de fonctionnement. L'initialisation a lieu, chaque fois que *Init* est VRAI (TRUE), et à la fin, le champ repasse automatiquement à FALSE (FAUX). En mode Auto, l'initialisation provoque la mise en rampe de *OP* d'une valeur définie par *StartVal* à une vitesse déterminée par *Slope_0*. La séquence reste au pas 0 jusqu'à ce que *Start* passe à TRUE (VRAI).
- **Start (Départ).** Permet de commencer l'exécution de la séquence en incrémentation depuis le pas 0 au pas 1 après initialisation. Le champ *Start* (Départ) revient automatiquement à FALSE. Il faut noter que les programmes de séquence qui bouclent un cycle en continu, ne nécessitent qu'un démarrage après un "démarrage à froid". Il est également possible de démarrer les séquences automatiquement après un "démarrage à froid" en modifiant les consignes par défaut dans les champs suivants au cours de la configuration :

<i>CurrStep</i>	=	1 (ou tout autre pas)
<i>Start</i>	=	TRUE (VRAI)
<i>Init</i>	=	FALSE (FAUX)

- **Continue (Continue).** Utilisé conjointement avec le drapeau *Wait* (Attente), qui peut être défini individuellement pour chaque pas dans les champs *DigOut*. Si le drapeau *Wait* (Attente) dans un pas particulier est TRUE (VRAI), la séquence attendra à la fin du pas respectif, jusqu'à ce que soit *Continue* passe à TRUE (VRAI) ou que le drapeau *Wait* (Attente) passe à FALSE (FAUX).

Le drapeau *StepDone*(*pas effectué*) est également mis à TRUE (VRAI) durant la période où la fonction *Wait* (Attente) est active.

- **Cycle.** Permet de cycler en continu un programme de séquence. Donc, lorsque *Cycle* est TRUE (VRAI), une séquence revient automatiquement au pas 1 à la fin du dernier pas (défini dans *MaxStep*). Il est à noter que la ré-initialisation d'une séquence cyclique en utilisant l'entrée *Init* interrompt le cycle en forçant le programme de séquence à passer au pas 0.

Un programme de séquence reste au dernier pas à la fin, lorsque le *cycle* est mis sur FALSE (FAUX). La sortie *CycDone* est TRUE (VRAIE) dans cette situation pour indiquer que le programme de séquence est terminé. Si l'entrée *Cycle* est définie TRUE (VRAIE), la séquence recommence au pas 1.

- **Track_0.** Lorsque cette entrée est TRUE (VRAIE), le mode poursuite est sélectionné automatiquement, lorsque le bloc de séquence passe au pas 0, quel que soit le statut de *SelTrack*. Donc, *OP* suit *Track*, à moins que le mode maintien ne soit sélectionné. Le mode poursuite est également automatiquement désélectionné, lorsque le programme de séquence (*Start* = TRUE) est lancé, à moins que *SelTrack* = TRUE (VRAI).

Le mode de fonctionnement n'est pas affecté au pas 0, si *Track_0* est FALSE (FAUX).

■ **SelHold.** Permet de sélectionner le mode maintien.

■ **SelTrack.** Permet de sélectionner le mode poursuite.

StatOut. Ce champ binaire indique le statut en cours de la séquence.

■ **Ramping (Mise en Rampe).** Indique la phase de fonctionnement du pas. Une phase de mise en rampe est indiquée par VRAI et une phase de temps d'arrêt est indiquée par FAUX. Il est à noter que l'exploitation en modes Manuel, Poursuite ou Maintien n'affecte pas l'état de ce champ.

■ **StepDone (Pas effectué).** Utilisé conjointement avec l'option Wait/Continue (Attente/Continuation) à la fin de chaque pas. *StepDone* est VRAI, si la séquence est maintenue par Wait à la fin d'un pas, et revient à FAUX, dès que la séquence continue ou est ré-initialisée. *StepDone* est également VRAI à la fin du dernier pas dans une séquence où *Cycle* est FAUX.

■ **CycDone (Cycle effectué).** Indique la fin de la séquence et passe à VRAI à la fin du dernier pas, lorsque *Cycle* est FAUX. Le champ repasse à FAUX, dès que la séquence est ré-initialisée ou lorsque *Cycle* est VRAI.

■ **NotTrack.** FAUX, lorsque le mode poursuite est actif.

Timeleft. Horloge de décomptage indiquant le temps d'arrêt restant dans le pas en cours. Sa valeur est figée, lorsque les modes Maintien, Poursuite ou Manuel sont actifs.

DigOut. Ce champ binaire indique l'état de la sortie logique en cours, c'est à dire l'état de chacun des 15 bits de sortie logique du bloc et du champ *Wait* pour le pas en cours. *DigOut* est en lecture uniquement en mode HOLD et TRACK (MAINTIEN ET POURSUITE).

DigOut_n. ($n = 0$ à 4) Ces cinq champs binaires définissent les états de sortie logique et les champs *Wait* pour chacun des pas 0 à 4 du bloc.

■ **Bit 0 - E.** Définissent l'état de sortie de chaque bit dans le champ en question.

■ **Wait (Attente).** Provoque l'attente de la séquence à la fin d'un pas pour une situation donnée. Lorsque le champ *Wait* est VRAI, l'exécution de la séquence s'arrête jusqu'à ce que soit *Continue* = VRAI ou *Wait* est FAUX. La séquence reprend au pas suivant, dès que l'une de ces conditions est remplie.

Dwell_n. ($n = 1$ à 4) Temps du palier pour le pas n . Spécifie la durée pendant laquelle la sortie *OP* est maintenue à la valeur cible pour chaque pas. Les unités de durée sont définies dans le champ *TimeBase*. Le temps de palier peut être ajusté dynamiquement, mais est ignoré, si la valeur est modifiée lorsque la synchronisation est en cours.

SEQE : BLOC D'EXTENSION DE SEQUENCE

Fonction du bloc

Les blocs d'extension de séquence (SEQE) permettent d'ajouter un nombre quelconque de pas supplémentaires à une séquence - jusqu'à huit par bloc SEQE. Chaque bloc SEQE est lié au précédent (ou bloc SEQ principal) en entrant son nom de repère dans le champ Extend de ce bloc.

Les liaisons E/S du bloc SEQE sont limitées à la sortie logique et aux champs d'alarme ; les autres liaisons sont effectuées dans le bloc SEQ proprement dit. Les paramètres du bloc SEQE ont exactement les fonctions et formats correspondants à ceux décrits pour le bloc SEQ. Voir les détails dans la section précédente.

Paramètres du bloc

Les symboles utilisés dans la table 6-2 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
Extend	Nom du bloc d'extension	Alphanumérique	
Slope_1 à Slope_8	Vitesses rampe Pas 1 à 8		▶◻▶
EndVal_1 à EndVal_8	Valeurs cible Pas 1 à 8	Eng	▶◻▶
Alarms			◻▶ ◻▶ ◻▶ ◻▶
Software	Données corrompues/Défaut de communication		V/F
Combined	Alarme combinée du bloc	V/F	
Dwell_1 à Dwell_8	Temps de palier Pas 1 à 8	Eng	▶◻▶
DigOut_n (n = 1 à 8)	Pas n Sorties logiques & champ attente	ABCD hex	▶◻▶
Bit0	Bit 0 Sortie logique	V/F — 1	D
Bit1	Bit 1 Sortie logique	V/F — 2	
Bit2	Bit 2 Sortie logique	V/F — 4	
Bit3	Bit 3 Sortie logique	V/F — 8	
Bit4	Bit 4 Sortie logique	V/F — 1	C
Bit5	Bit 5 Sortie logique	V/F — 2	
Bit6	Bit 6 Sortie logique	V/F — 4	
Bit7	Bit 7 Sortie logique	V/F — 8	
Bit8	Bit 8 Sortie logique	V/F — 1	B
Bit9	Bit 9 Sortie logique	V/F — 2	
BitA	Bit A Sortie logique	V/F — 4	
BitB	Bit B Sortie logique	V/F — 8	
BitC	Bit C Sortie logique	V/F — 1	A
BitD	Bit D Sortie logique	V/F — 2	
BitE	Bit E Sortie logique	V/F — 4	
Wait	Champ attente	V/F — 8	

Table 6-2 Paramètres du bloc SEQE

TOTAL : BLOC DE TOTALISATION

Fonction du bloc

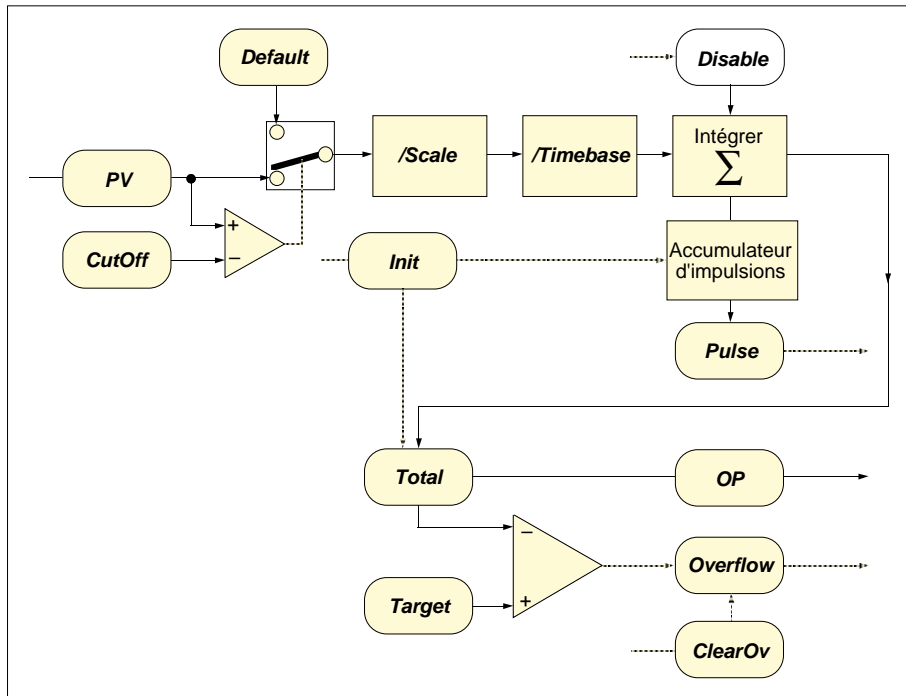


Figure 6-2 Schéma du bloc TOTAL

Voir la figure 6-2. Le bloc de totalisation intègre une entrée analogique *PV*, sauvegarde le résultat dans le paramètre *Total* sous la forme d'un nombre entier à 8 chiffres et également sous le format à point flottant dans le paramètre *OP*. *Scale* (Echelle) permet de spécifier la taille des unités qui sont intégrées, et *Timebase* (Base de temps) définit l'unité de temps pour l'intégration et *PV*.

Le bloc comprend une coupure sur le seuil bas de *PV* avec une totalisation par défaut, un sortie d'impulsion pour les compteurs et une conversion tension/fréquence, un drapeau de dépassement cible et des entrées d'initialisation et d'invalidation.

Paramètres du bloc

Les symboles utilisés dans la table 6-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée bloc)	Eng	
CutOff	Spécification valeur seuil bas	Eng	
Default	Valeur PV lorsque l'entrée < CutOff	Eng	
Scale	Valeur de mise à l'échelle de l'entrée		
Total	Valeur totalisateur	Entier	
OP	Sortie totalisateur	Eng	
HR_OP, LR_OP	Echelles haute & basse des objets graphiques (PV, OP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Target	Définit la valeur maximale pour le Total	Entier	
Overflow	Avertissement de dépassement totalisateur	V/F	
ClearOv	Efface le drapeau dépassement	V/F	
Disable	Invalide la totalisation	V/F	
Pulse	Sortie impulsion intégrateur	V/F	
Init	Initialise la totalisation	V/F	
Timebase	Spécifie les unités de temps de l'intégration	Menu	

Table 6-3 Paramètres du bloc TOTAL

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d'entrée à totaliser. (Les valeurs négatives de *PV* sont traitées comme des zéros).

Cutoff (Coupure). Si $PV < CutOff$, la valeur du paramètre par défaut est totalisée au lieu de l'entrée PV.

Default (Défaut). Valeur totalisée lorsque l'entrée *PV* tombe en-dessous de la valeur *CutOff*. (Les valeurs *par défaut négatives* décrémentent *Total* et *OP*, et l'accumulateur d'impulsions).

Scale (Echelle). Met *PV* à l'échelle en rendant la variable à totaliser égale à *PV/Echelle*.

Total (Total). Valeur du totalisateur au format entier avec une pleine résolution de 8 caractères. Il peut être lié aux affichages graphiques (par l'intermédiaire de l'objet graphique READOUT).

OP. Sortie de la valeur du totalisateur au format à 7 chiffres à point flottant.

HR_OP, LR_OP. Echelle haute & basse pour les objets graphiques liés à *PV* ou *OP* (bargraphe, tendance). *HR_OP* & *LR_OP* définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Target (Cible). Définit la valeur maximale (intégrale) pour le *Total*. Lorsque *Total* est près de dépasser la valeur dans la cible, la sortie dépassement passe à VRAI et l'intégration continue à partir d'une valeur de zéro. Etant un entier, *Target* ne peut être lié à un bargraphe. Noter que la modification de *Target* en cours de fonctionnement initialise le bloc TOTAL et remet à zéro *Total*.

Overflow (Dépassement). Drapeau de dépassement, qui passe à VRAI, lorsque la valeur du *Total* dépasse la valeur spécifiée dans la cible. *Overflow* peut être remis à zéro grâce aux paramètres *ClearOv* et *Init*.

ClearOv (Effacement dépassement). Remet à zéro le drapeau de dépassement et repasse automatiquement à FAUX.

Disable (Invalidation). Invalide la totalisation en arrêtant la fonction d'intégration. La sortie des impulsions qui restent dans l'accumulateur d'impulsions n'est pas affectée.

Pulse (Impulsion). Ce champ de sortie est piloté par les impulsions sauvegardées dans l'accumulateur d'impulsions, qui est décrémenté à la sortie de chaque impulsion, jusqu'à ce qu'il soit vide. Pour chaque impulsion sauvegardée, *Pulse* passe à VRAI et se remet automatiquement à l'état initial après deux mises à jour du bloc (~0,2 s). L'accumulateur lui-même est incrémenté (ou décrémenté) chaque fois que le total est incrémenté (ou décrémenté). (Les valeurs *par défaut négatives* peuvent décrémenter le total). Il faut noter que les impulsions sont fournies uniquement, lorsque l'accumulateur contient un nombre positif.

Lorsque les vitesses d'intégration de *PV* sont élevées, les impulsions relativement lentes ne peuvent suivre, mais aucune n'est omise - elles s'accumulent et finissent par être sorties. De cette manière, le total d'impulsions peut refaire son retard au cours des périodes plus calmes. Bien que l'accumulateur s'assure qu'aucune impulsion n'est perdue à l'intérieur du bloc lui-même, elles peuvent être perdues au cours de la transmission sur le LIN, si les temps d'échantillonnage des systèmes associés sont différents. Il faut alors ralentir la vitesse d'impulsion en augmentant la valeur du paramètre *Echelle*.

L'accumulateur de sortie des impulsions est ré-initialisé par *Init*, et chaque fois que le paramètre base de temps est enregistré.

Init. Initialise les valeurs *Total* et *OP* à zéro, remet à zéro l'accumulateur d'impulsions et fait repasser *Overflow* à FAUX.

Timebase. (Sec./Min./Heures/Jours). Permet de spécifier l'unité de temps de *PV* et de la fonction d'intégration.

DTIME : BLOC TEMPS MORT

Fonction du bloc

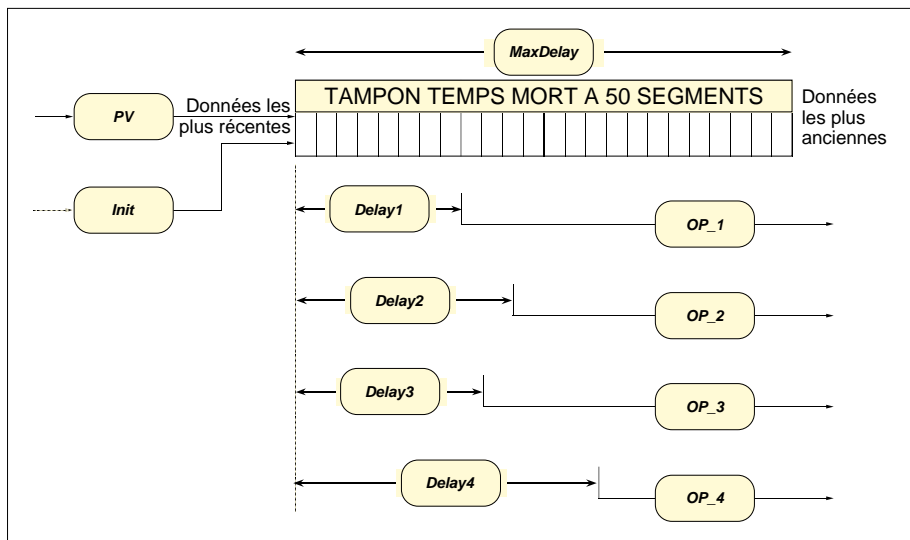


Figure 6-3 Schéma du bloc DTIME

Voir la figure 6-3. Le bloc temps mort introduit des temporisations dans le signal de mesure **PV**. Le tampon du bloc contient 50 zones de sauvegarde et 4 pointeurs pour l'extraction des sorties. A chaque mise à jour d'un échantillon, une nouvelle mesure est insérée et la mesure la plus ancienne est effacée. La vitesse d'échantillonnage est calculée à partir de $MaxDelay/50$ (temporisation maximale/50), où *MaxDelay* est la temporisation maximale en secondes.

Chaque pointeur peut extraire une sortie pour n'importe quel temporisation jusqu'à *MaxDelay*. Lorsque le point requis se situe entre deux échantillons de tampon, la sortie est estimée par interpolation linéaire entre les échantillons. La figure 6-4 montre un exemple de sorties temporisées à partir d'une impulsion d'entrée rectangulaire **PV**. Le "retard" des signaux de sortie, par suite de l'action de moyenne entre les points du tampon, dépend de la valeur de *MaxDelay*.

Le bloc est initialisé par l'intermédiaire du paramètre *Init*, qui définit tous les éléments comme étant égaux à la valeur active de **PV**.

Paramètres du bloc

Les symboles utilisés dans la table 6-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

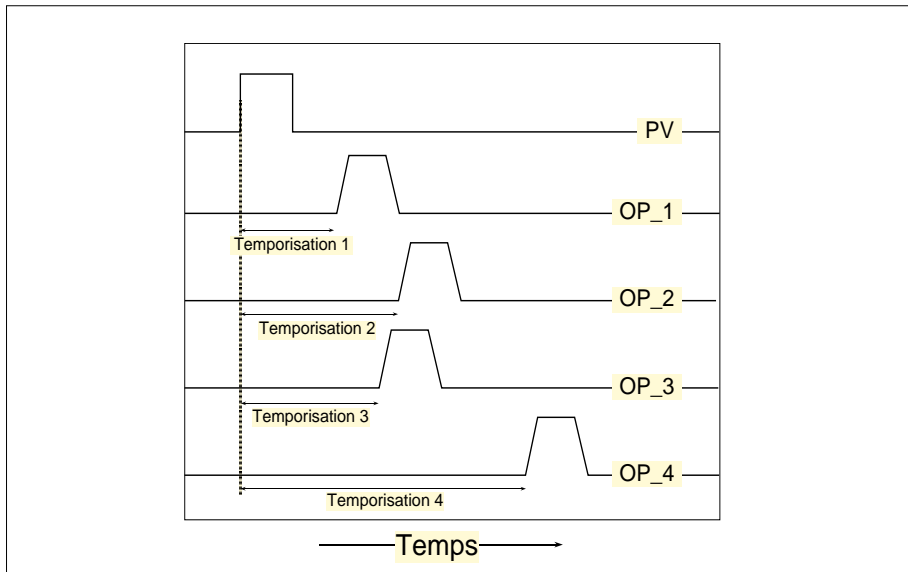


Figure 6-4 Sortie du bloc DTIME (Exemple)

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (entrée du bloc)	Eng	
OP_1	Valeur de la sortie à la temporisation 1	Eng	
OP_2	Valeur de la sortie à la temporisation 2	Eng	
OP_3	Valeur de la sortie à la temporisation 3	Eng	
OP_4	Valeur de la sortie à la temporisation 4	Eng	
HR, LR	Echelle haute & basse des objets graphiques (PV, O/Ps)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Delay1	Durée temporisation 1	Eng	
Delay2	Durée temporisation 2	Eng	
Delay3	Durée temporisation 3	Eng	
Delay4	Durée temporisation 4	Eng	
MaxDelay	Durée maximale temporisation	Eng	
Init	Initialisation tampon temporisation	V/F	

Table 6-4 Paramètres du bloc DTIME

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur de l’entrée.

OP_1 à OP_4. Valeur de la sortie à la temporisation 1 à 4. (Les sorties sont modifiées par les effets de l'interpolation linéaire, dans la mesure déterminée par la valeur de *MaxDelay*.)

HR, LR. Echelle haute & basse pour les objets graphiques liés à *PV* ou *OP_1* à *OP_4* (bargraphes, tendance). *HR* et *LR* définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Delay1 à Delay4 (Temporisation 1 à 4). Permet de spécifier la durée (secondes) de delay1 à delay4. Il faut noter que les valeurs de temporisation entrées ne peuvent dépasser la valeur de *MaxDelay*. De même, si une valeur de *MaxDelay* entrée est inférieure aux temporisations existantes, elles sont automatiquement réduites à la nouvelle valeur de *MaxDelay*.

MaxDelay (Temporisation maximale). Permet de spécifier la durée maximale de la temporisation (c'est à dire la longueur du tampon de temporisation en secondes). Les valeurs *MaxDelay* entrées sont automatiquement arrondies au multiple immédiatement supérieur de 5 secondes.

Init. Initialise le tampon en définissant tous les éléments comme étant égaux à la valeur active de *PV*.

TIMER : BLOC HORLOGE

Fonction du bloc

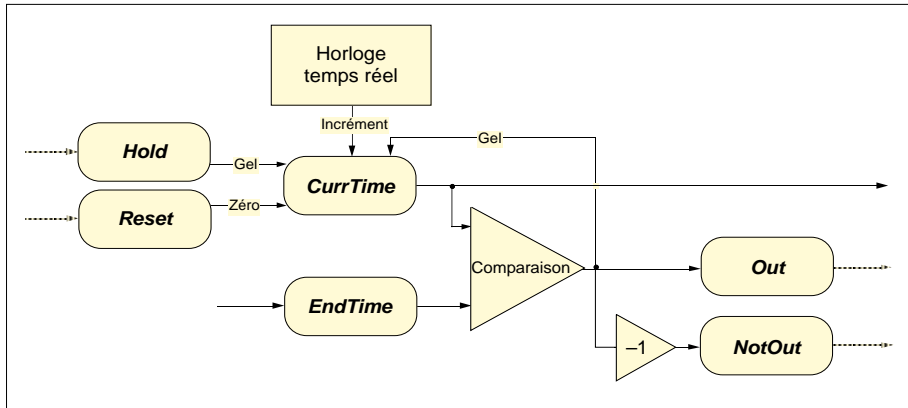


Figure 6-5 Schéma du bloc TIMER

Voir le schéma de la figure 6-5. Le bloc TIMER est une horloge incrémentielle qui peut être remise à zéro, et qui donne en permanence l'heure et définit des drapeaux logiques lorsqu'une valeur d'heure de fin prédéfinie est atteinte et que le décompte du temps s'arrête. Une entrée logique *Hold* permet de "geler" l'horloge. Le paramètre *Timebase* permet de sélectionner les secondes, minutes ou heures.

Précision horaire. La précision de la sortie heure dépend du rythme de mise à jour du bloc, c'est à dire du temps d'échantillonnage. Au mieux, il se répète toutes les 0,1 seconde, mais pour un schéma de taille moyenne (de 75 à 100 blocs environ), les temps d'échantillonnage sont plus proches de 0,2 à 0,3 seconde.







Paramètres du bloc

Les symboles utilisés dans la table 6-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2.13.

EndTime (Heure de fin). La valeur de temps à laquelle l'horloge s'arrête. Les unités de temps sont spécifiées par le paramètre *Timebase*. Si *EndTime* est défini pour une valeur qui dépasse *CurrTime - Reset* et *Hold* étant FAUX - *CurrTime* s'incrémente jusqu'à ce que la valeur *EndTime* soit atteinte. A ce stade, *CurrTime* est gelé et les drapeaux *Out/NotOut* sont définis.

Paramètre	Fonction	Unités	Etat
EndTime	Valeur heure de fin (en unités base de temps)	Eng	     

TIMEDATE : BLOC EVENEMENT HEURE/DATE

Fonction du bloc

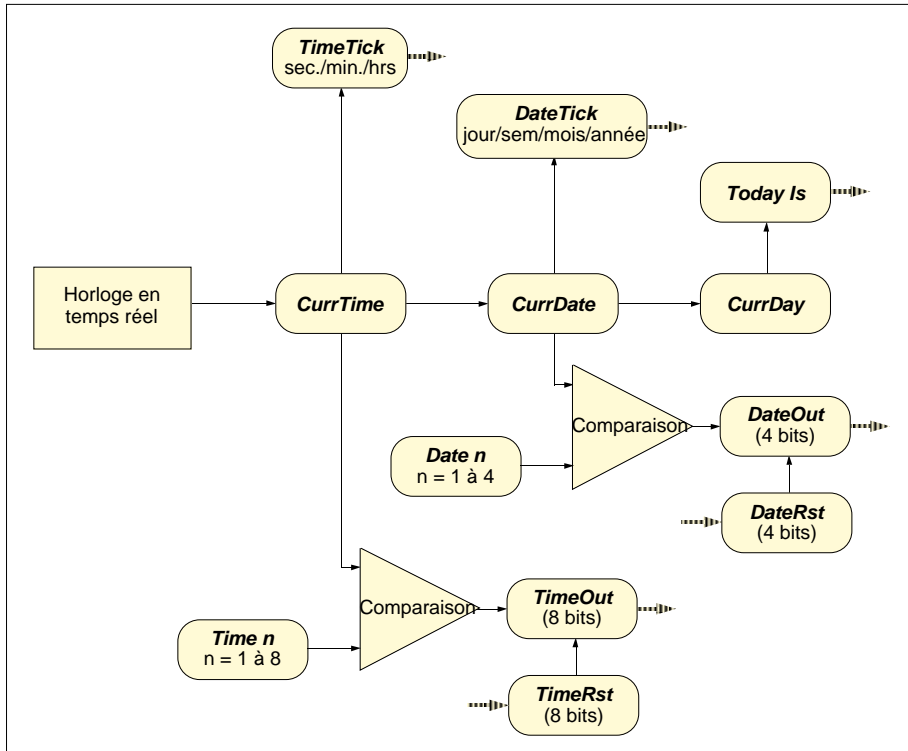


Figure 6-6 Schéma du bloc TIMEDATE

Voir la figure 6-6. Le bloc événement Time/Date agit comme une horloge/calendrier calculant l'heure, la date et le jour à partir de l'horloge système du T1000. Il émet des impulsions logiques - "tic-tac" à chaque incrément de l'heure/date, et indique le jour de la semaine. Les week-ends sont indiqués différemment. Le bloc agit également comme un "réveil" programmable, définissant des bits de sortie (*TimeOut* et *DateOut* - Sortie heure et date), lorsque des heures et dates précises arrivent. Le bloc permet de programmer jusqu'à huit heures différentes d'événement et quatre dates d'événement.

L'horloge système peut être remise à zéro et contrôlée par l'intermédiaire du bloc TIMEDATE.

Paramètres du bloc

Les symboles utilisés dans la table 6-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
CurrTime	Heure (horloge système)	hh:mm:ss	
TimeTick	Impulsions heure (sec./min./hrs)	Champ binaire	
SecsTick	Sortie impulsions secondes	V/F	
MinsTick	Sortie impulsions minutes	V/F	
HrsTick	Sortie impulsions heures	V/F	
Time1 à Time8	Heures événement	nn:nn:nn	
TimeOut	Bits de sortie événement heure	Champ binaire	
Time1out	Bit de sortie événement 1 heure	V/F	
Time2out	Bit de sortie événement 2 heure	V/F	
Time3out	Bit de sortie événement 3 heure	V/F	
Time4out	Bit de sortie événement 4 heure	V/F	
Time5out	Bit de sortie événement 5 heure	V/F	
Time6out	Bit de sortie événement 6 heure	V/F	
Time7out	Bit de sortie événement 7 heure	V/F	
Time8out	Bit de sortie événement 8 heure	V/F	
TimeRst	RAZ des bits de sortie événement heure	Champ binaire	
Time1rst	RAZ du bit de sortie événement 1 heure	V/F	
Time2rst	RAZ du bit de sortie événement 2 heure	V/F	
Time3rst	RAZ du bit de sortie événement 3 heure	V/F	
Time4rst	RAZ du bit de sortie événement 4 heure	V/F	
Time5rst	RAZ du bit de sortie événement 5 heure	V/F	
Time6rst	RAZ du bit de sortie événement 6 heure	V/F	
Time7rst	RAZ du bit de sortie événement 7 heure	V/F	
Time8rst	RAZ du bit de sortie événement 8 heure	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
CurrDate	Date (JJ/MM/AA)	dd:mm:yy	
CurrDay	Jour	Menu	
DateTick	Impulsions date (jour, semaine, mois, année)	Champ binaire	
DayTick	Sortie impulsion quotidien	V/F	
WeekTick	Sortie impulsion hebdomadaire	V/F	
MnthTick	Sortie impulsion mensuelle	V/F	
YearTick	Sortie impulsion annuelle	V/F	

suite...

suite...

Paramètre	Fonction	Unités	Etat
DayTick	Jour	Champ binaire	
Monday	Jour de la semaine	V/F	
Tuesday		V/F	
Wednesday		V/F	
Thursday		V/F	
Friday		V/F	
Saturday		V/F	
Sunday		V/F	
Weekend	Samedi ou dimanche	V/F	
Date1 à Date4	Dates événement	dd:mm:yy	
DateOut	Bit de sortie événement date	Champ binaire	
Date1out	Bit de sortie événement 1 date	V/F	
Date2out	Bit de sortie événement 2 date	V/F	
Date3out	Bit de sortie événement 3 date	V/F	
Date4out	Bit de sortie événement 4 date	V/F	
DateRst	RAZ bits de sortie événement date	Champ binaire	
Date1rst	RAZ bit de sortie événement 1 date	V/F	
Date2rst	RAZ bit de sortie événement 2 date	V/F	
Date3rst	RAZ bit de sortie événement 3 date	V/F	
Date4rst	RAZ bit de sortie événement 4 date	V/F	

Table 6-6 Paramètres du bloc TIMEDATE

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

CurrTime (Heure). L’heure indiquée par l’horloge du système en hr:min:sec.
L’horloge système peut être ré-initialisée par ce paramètre (ou par le bloc du T1000).

TimeTick (Top lié au temps). Champ binaire fournissant des impulsions à chaque incrément du temps (“tic-tac”)

- **SecsTick (Top des secondes).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur des secondes est incrémentée.
- **MinsTick (Top des minutes).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur des minutes est incrémentée.
- **HrsTick (Top des heures).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur des heures est incrémentée.

Time1 à Time8 (Temps réveil). Huit heures événements “réveil” programmables exprimés en hr:min:sec. Lorsqu’une de ces heures est atteinte, le paramètre *TimeOut* passe à VRAI.

TimeOut (Sortie temps réveil). Champ binaire de huit bits, *Time1out* à *Time8out*. Un bit est mis à 1, lorsque l'heure de l'horloge système atteint l'heure spécifiée dans le paramètre *Time1* à *Time8* correspondant. Les bits *TimeOut* peuvent être remis à zéro individuellement en utilisant le paramètre *TimeRst*.

TimeRst (RAZ temps réveil). Champ binaire de huit bits, *Time1Rst* à *Time8Rst*. En passant de FAUX à VRAI, chaque bit remet à zéro le bit *TimeOut* correspondant (*Time1out* à *Time8out*).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

CurrDate (Date). Date indiquée par l'horloge système dans le format JJ:MM:AA. Le calendrier système peut être ré-initialisé par ce paramètre (ou le bloc T1000).

NOTA. La date ne peut être sélectionnée que par l'intermédiaire de *CurrDate* dans un T100.

CurrDay (Jour). (MONDAY/TUESDAY/WEDNESDAY/THURSDAY/FRIDAY/SATURDAY/SUNDAY - LUNDI/MARDI/MERCREDI/JEUDI/VENDREDI/SAMEDI/DIMANCHE). Jour calculé à partir du calendrier système.

DateTick (Top des jours). Champ binaire fournissant des impulsions à chaque incrément de date ("tic-tac")

- **DayTick (Top jour).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur du jour est incrémentée.
- **WeekTick (Top semaine).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur de la semaine est incrémentée.
- **MnthTick (Top mois).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur du mois est incrémentée.
- **YearTick (Top année).** Le bit passe à VRAI pour une période de mise à jour du bloc, et repasse à FAUX, lorsque la valeur de l'année est incrémentée.

DayTick (Jours et fin de semaine). Champ binaire dont les huit bits indiquent le jour. Les sept premiers bits indiquent (VRAI) Lundi à Dimanche, et le huitième "Weekend" est VRAI le Samedi et le Dimanche.

Date1 à Date4 (Date réveil). Quatre dates événement "réveil" indépendantes programmables exprimées en JJ:MM:AA. Lorsque l'une de ces dates est atteinte, le bit du paramètre *DateOut* correspondant passe à VRAI.

DateOut (Sortie date réveil). Champ binaire de quatre bits, *Date1out* à *Date4out*. Un bit est défini, lorsque la date du calendrier système atteint la date spécifiée dans le paramètre *Date1* à *Date4* correspondant. Les bits *DateOut* peuvent être remis à zéro individuellement en utilisant le paramètre *DateRst*.

DateRst (RAZ date réveil). Champ binaire de quatre bits, *Date1Rst* à *Date4Rst*. En passant de FAUX à VRAI, chaque bit remet à zéro le bit DateOut correspondant (*Date1out* à *Date4out*).

TPO : BLOC DE SORTIE PROPORTIONNELLE AU TEMPS

Fonction du bloc

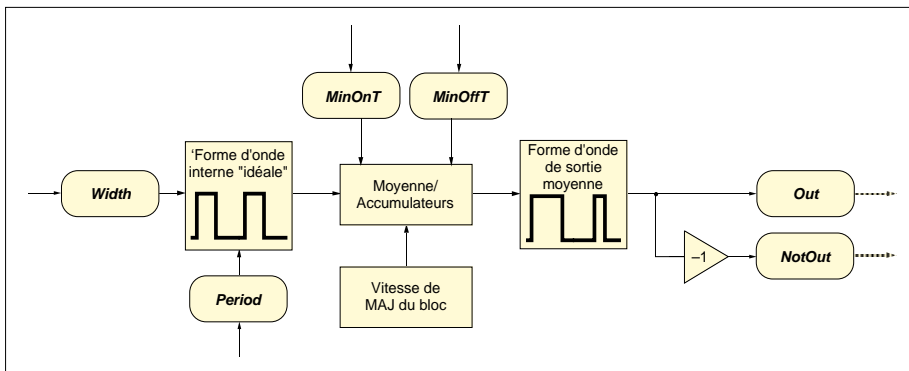


Figure 6-7 Schéma du bloc TPO

Le bloc de sortie proportionnelle au temps convertit une entrée analogique (*Width* - Durée) en une sortie logique (*Out* - Sortie) d'onde carrée, avec un rapport variable actif/inactif proportionnel à l'entrée. Une application typique pour le bloc TPO consiste à utiliser un signal analogique continu pour contrôler l'entrée d'énergie dans un procédé où l'énergie peut être 'on' ou 'off' (marche ou arrêt), par exemple, dans les fours électriques.

Voir le schéma de la figure 6-7. Le bloc génère une forme d'onde interne "idéale" ayant une période définie par l'entrée *Period* (Période), et un rapport actif/inactif défini par l'entrée *Width* (Durée). *Width* est plus particulièrement le pourcentage de la période pendant laquelle la sortie est haute (VRAIE). Voir figure 6-8.

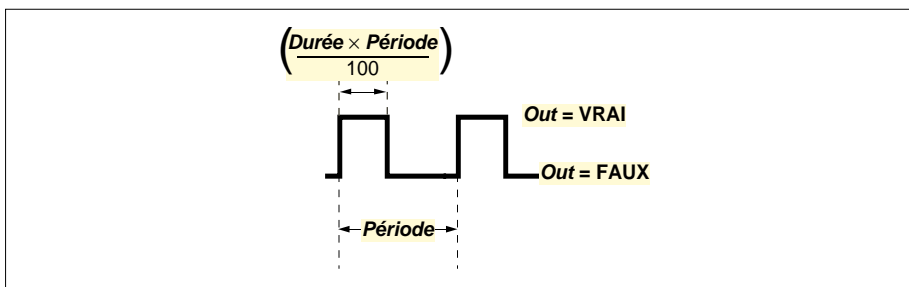


Figure 6-8 Paramètres de forme d'onde interne du bloc TPO

Temps 'On' et 'Off' minimum. La forme d'onde idéale générée par le bloc peut être soumise à une modification automatique avant d'être transmise par l'intermédiaire de la liaison *Out*. C'est ce qui se produit si les entrées *MinOnT* ou *MinOffT* spécifient des temps 'on' ou 'off' minimum qui sont en conflit avec les temps idéaux calculés. Les temps

on/off minimum peuvent être utilisés lorsqu’une installation pourrait être endommagée ou déstabilisée par de très courtes périodes de commutation. L’algorithme du bloc résout ces conflits en maintenant la sortie ‘on’ ou ‘off’ plus longtemps que la valeur idéale, en sauvegardant les erreurs de temps qui en résultent dans des accumulateurs pour qu’elles puissent être appliquées dans une période ultérieure. (Voir les détails dans la section ci-dessous, *Menu de spécifications du bloc*).

Précision de la sortie. La forme d’onde idéale est également automatiquement ajustée, chaque fois que le rapport actif/inactif calculé ne peut en pratique être sorti avec précision, dans la mesure où la vitesse de mise à jour du bloc est importante par rapport à la période du bloc TPO. Dans ces cas, l’algorithme fait la moyenne de la sortie pendant des cycles successifs. Par exemple, pour une période de 1 seconde, une durée de 25 % et une vitesse de mise à jour du bloc de 0,1 seconde, le bloc sort des cycles alternés de 0,2 et 0,3 seconde (pour obtenir une moyenne de 0,25 seconde).

Paramètres du bloc

Les symboles utilisés dans la table 6-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.
















Paramètre	Fonction	Unités	Etat
Width	‘On’ comme pourcentage de période “idéale” (0-100)	%	 
Period	Période de forme d’onde interne “idéale” (sec.)	Eng	 
MinOnT	Temps ‘ON’ minimum sortie (sec.)	Eng	 
MinOffT	Temps ‘OFF’ minimum sortie (sec.)	Eng	 
Alarms			  
Software	Données corrompues/Défaut communications	V/F	
Combined	Alarme combinée du bloc	V/F	
Out	Forme d’onde carrée sortie	V/F	 
NotOut	Inverse de Out (= NOT-Out)	V/F	 

Table 6-7 Paramètres du bloc TPO

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Width (Durée). Pourcentage de la période pendant laquelle la forme d’onde interne est “on”. Voir figure 6-8. Plus précisément:

Temps “On” = *Durée* x *Période* ÷ 100 (secondes)

Il faut noter que la sortie effective du bloc *Out* peut être ‘on’ pour des durées autres que celle-ci, en raison des effets de moyenne automatique. Les entrées de durée devraient s’échelonner entre 0 et 100 ; les entrées en dehors de cette échelle sont tronquées.

Period (Période). Période de la forme d'onde interne en secondes. Voir figure 6-8. Il faut noter que la forme d'onde de la sortie du bloc *Out* peut avoir une période apparente autre que celle-ci, en raison des effets de moyenne automatique.

MinOnT & MinOffT. Permet de définir les temps 'on' et 'off' minimum pour le signal *Out*, qui peuvent être nécessaires pour éviter d'endommager les actionneurs, etc. Si le temps 'on' de la forme d'onde interne est inférieur à *MinOnT*, la sortie reste *off* pendant cette période, mais le temps 'on' inutilisé s'ajoute dans un accumulateur d'erreurs. Au cours de la période suivante, ou d'une période ultérieure, lorsque l'erreur accumulée finit par dépasser *MinOffT*, la sortie passe à l'état haut pour la durée d'erreur totale et l'accumulateur est remis à zéro. *MinOffT* fonctionne de manière similaire, la sortie restant *on* jusqu'à ce que la valeur de l'accumulateur d'erreur 'off time' dépasse *MinOffT*. L'effet net de ces ajustements est d'augmenter la période apparente de la forme d'onde de sortie (par un multiple entier) pour prendre en compte *MinOnT* et *MinOffT*, tout en maintenant le rapport actif/inactif à un niveau de *moyenne* correct.

NOTA. Pour éviter la saturation d'erreurs, les accumulateurs ne peuvent contenir des erreurs supérieures à la période de la forme d'onde interne. *Les erreurs qui s'accumulent au-delà de ce niveau sont tronquées, ce qui produit une forme d'onde de sortie dont le rapport moyen actif/inactif est incorrect.* C'est ce qui peut se produire si *MinOnT* ou *MinOffT* dépassent la moitié de la période, mais dépend également de la valeur de la durée. La troncature se produit en particulier si

$$50 \% < \text{Durée} < (\text{MinOnT}/\text{Période}) \%$$

$$\text{et/ou si} \quad 50 \% < (100 - \text{Durée}) < (\text{MinOffT}/\text{Période}) \%$$

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Out (Sortie). La forme d'onde logique effectivement sortie par le bloc. Il s'agit d'une version traitée de la forme d'onde interne "idéale" (voir figure 6-8) qui a le même rapport moyen actif/inactif, mais peut être une période apparente différente, suivant les valeurs de *MinOnT*, *MinOffT*, et la fréquence de mise à jour du bloc. Voir le nota de la section *MinOnT & MinOffT* ci-dessus.

NotOut. Inverse du paramètre *Out*, c'est à dire NOT-*Out*.

DELAY : BLOC RETARD

Fonction du bloc

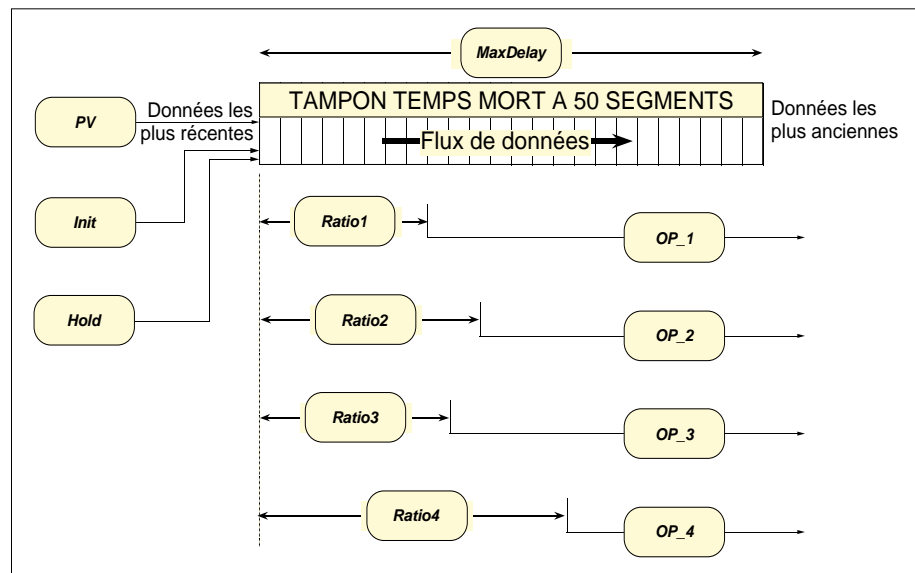


Figure 6-9 Schéma du bloc DELAY

Voir le schéma de la figure 6-9. Le bloc DELAY introduit des retards dans les signaux de mesure *PV*. Le tampon du bloc contient 50 emplacements de sauvegarde et 4 pointeurs pour l'extraction des sorties. A chaque mise à jour des échantillons, toutes les mesures existantes se déplacent d'un emplacement dans le tampon, une nouvelle mesure est insérée, et la mesure la plus ancienne est supprimée. La vitesse d'échantillonnage du tampon est calculée suivant $MaxDelay/50$, où *MaxDelay* représente le retard maximum en secondes.

Chaque pointeur peut extraire une sortie pour n'importe quel retard jusqu'à 100 % de *MaxDelay*. Lorsque le point se situe entre deux échantillons du tampon, la sortie est estimée par interpolation linéaire entre les échantillons.

Le bloc est initialisé par l'intermédiaire du paramètre *Init*, qui rend tous les paramètres égaux à la valeur active de *PV*. Lorsque le paramètre *Hold* est VRAI, l'action du bloc est "gelée", c'est à dire que le flux de données dans le tampon s'arrête, et les sorties restent donc inchangées.

Paramètres du bloc

Les symboles utilisés dans la table 6-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Variable procédé (Entrée du bloc)	Eng	
OP_1	Valeur de sortie au Retard 1	Eng	
OP_2	Valeur de sortie au Retard 2	Eng	
OP_3	Valeur de sortie au Retard 3	Eng	
OP_4	Valeur de sortie au Retard 4	Eng	
HR, LR	Echelle graphique haute & basse (PV, O/Ps)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Ratio1	Durée Retard 1 (% de MaxDelay)	%	
Ratio2	Durée Retard 2 (% de MaxDelay)	%	
Ratio3	Durée Retard 3 (% de MaxDelay)	%	
Ratio4	Durée Retard 4 (% de MaxDelay)	%	
MaxDelay	Durée maximale du retard	Secs	
Init	Initialisation du tampon de retard	V/F	
Hold	Gel du tampon de retard	V/F	

Table 6-8 Paramètres du bloc DELAY

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Valeur d’entrée

OP_1 à OP_4. Valeurs de sortie aux retards de pour-cent *Ratio1* à *Ratio4* *MaxDelay*.

HR, LR. Echelle haute & basse pour les objets graphiques reliés à *PV* ou *OP_1* à *OP_4* (bargraphe, tendance). *HR* et *LR* définissent respectivement les affichages 100 et 0 %.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active à la priorité la plus haute du bloc.

Ratio1 à Ratio4 (Rapport1 à 4). Permet de spécifier les durées (secondes) de retard de *OP_1* à *OP_4*, comme des pourcentages de *MaxDelay*. Noter que les valeurs ne peuvent être supérieures à 100 %.

MaxDelay. Permet de spécifier la durée de retard maximale (c’est à dire la durée du tampon) en secondes. Noter que *MaxDelay* peut être modifié pendant l’exécution du bloc

sans perte de données. Il s'agit d'une simple modification de la vitesse du flux de données et de leur apparition en *OP_1* à *OP_4*.

Init. Une entrée vraie initialise le tampon en rendant tous les éléments égaux à la valeur active de *PV*. *Init* est en écriture uniquement, c'est à dire qu'*Init* repasse à FAUX après utilisation.

Hold (Maintien). Lorsque *Hold* est VRAI, le flux de données dans le tampon est arrêté et *OP_1* à *OP_4* restent donc constants. L'initialisation par l'intermédiaire d'*Init* n'est pas affectée par l'état de *Hold*.

RATE_ALM : BLOC ALARME DE VITESSE

Fonction du bloc

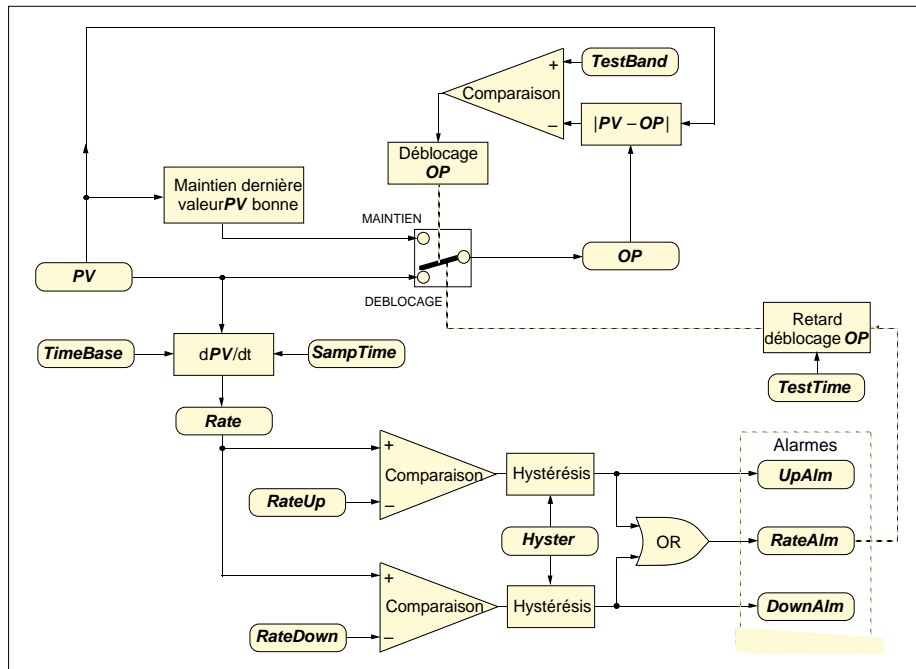


Figure 6-10 Schéma du bloc RATE_ALM

Voir le schéma de la figure 6-10. Le bloc RATE_ALM contrôle la vitesse de modification d'un signal d'entrée analogique *PV* et copie le signal sur la sortie *OP*. Si *PV* change à une vitesse supérieure à celle définie par l'utilisateur, *OP* est maintenu à la dernière "bonne" valeur de *PV* (c'est à dire pas en alarme) et l'alarme appropriée est déclenchée.

Le déblocage d'*OP* après la disparition d'une alarme de vitesse n'est pas immédiat, mais se produit après une durée spécifiée (*TestTime*) ou lorsque *PV* revient à une valeur spécifiée (*TestBand*) d'*OP*, suivant ce qui se produit en premier. Le bit d'état *OPrelsd* signale le déblocage d'*OP* en passant à VRAI pendant une seule itération de la tâche. Les figures 6-11 et 6-12 illustrent le comportement d'*OP*, *OPrelsd* et le bit d'alarme de vitesse *RateAlm* à mesure que *dPV/dt* sort et rentre dans les limites.

Paramètres du bloc

Les symboles utilisés dans la table 6-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

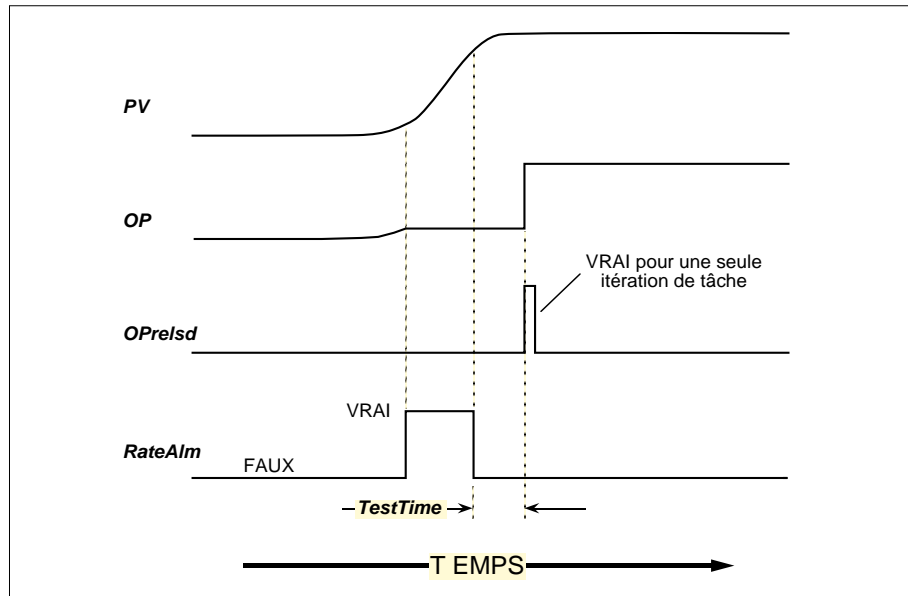


Figure 6-11 Actions des paramètres RATE_ALM - Déblocage OP après TestTime

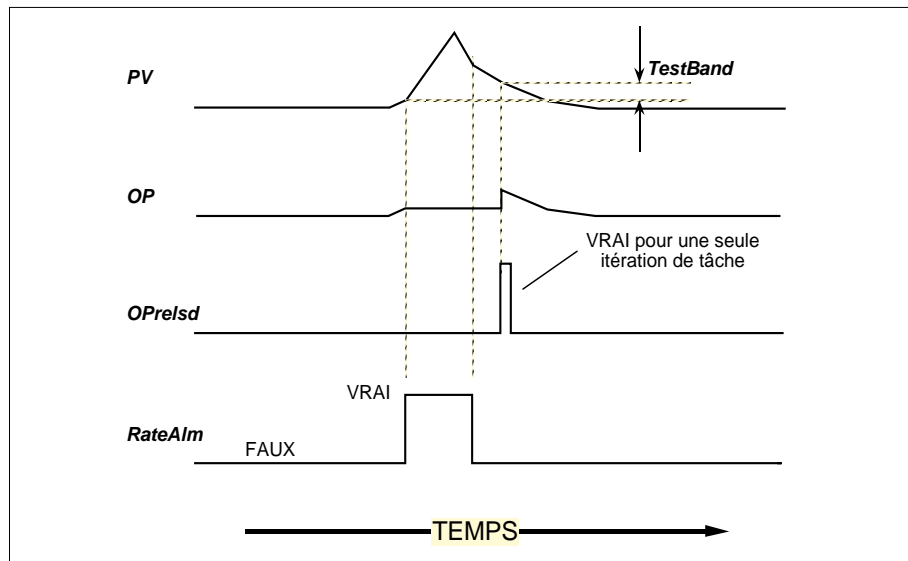


Figure 6-12 Actions des paramètres RATE_ALM - Déblocage OP contrôlé par TestBand

Paramètre	Fonction	Unités	Etat
PV	Valeur à soumettre à l'alarme de vitesse	Eng	
OP	Valeur après l'alarme de vitesse	Eng	
TestBand	Valeur de déblocage d'erreur PV	Eng	
HR	Echelle haute	Eng	
LR	Echelle basse	Eng	
Alarms			
Software	Données corrompues	V/F	
RateAlm	UpAlm ou DownAlm	V/F	
UpAlm	Alarme de vitesse croissante	V/F	
DownAlm	Alarme de vitesse décroissante	V/F	
Combined	Alarme combinée du bloc	V/F	
RateUp	Valeur d'alarme de vitesse croissante	Eng/Temps	
RateDown	Valeur d'alarme de vitesse décroissante	Eng/Temps	
Hyster	Hystérésis de l'alarme de vitesse	Eng/Temps	
Rate	Vitesse effective	Eng/Temps	
SampTime	Temps d'échantillonnage de PV	Temps	
TestTime	Temps de déblocage d'OP	Temps	
TimeBase	Unités de temps du calcul de vitesse	Menu	
Status			
OPrelsd	Drapeau OP déblocué (une fois seulement)	V/F	

Table 6-9 Paramètres du bloc RATE_ALM

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV. Signal d'entrée. Valeur à soumettre à l'alarme de vitesse.

OP. Signal de sortie. Valeur après application de l'alarme de vitesse. Poursuit normalement *PV*, à moins d'être maintenu à la dernière “bonne” valeur de *PV* (c'est à dire pas en alarme) durant et après une alarme de vitesse.

TestBand. La différence absolue entre *PV* et *OP* lorsqu'*OP* sort de l'état de maintien, si cela se produit avant la fin de *TestTime*, après que *RateAlm* est repassé à FAUX. Voir figure 6-12.

HR, LR. Echelle haute & basse pour les objets graphiques reliés à *PV* ou *OP_1* à *OP_4* (bargraphe, tendance). *HR* et *LR* définissent respectivement les affichages 100 et 0 %.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **RateAlm.** *UpAlm* OU *DownAlm*

■ **UpAlm.** Alarme de vitesse croissante.

■ **DownAlm.** Alarme de vitesse décroissante.

- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

RateUp. Limite d'alarme de vitesse croissante. Si *PV* croît à une vitesse supérieure à *RateUp*, une alarme *UpAlm* est déclenchée. L'alarme disparaît uniquement, lorsque la vitesse de changement de *PV* descend en-dessous de *RateUp* dans les limites d'*Hyster* au minimum (valeur d'hystérésis de l'alarme).

RateDown. Limite d'alarme de vitesse décroissante. Si *PV* décroît à une vitesse supérieure à *RateDown*, une alarme *DownAlm* est déclenchée. L'alarme disparaît uniquement, lorsque la vitesse de changement de *PV* descend en-dessous de *RateDown* dans les limites d'*Hyster* au minimum (valeur d'hystérésis de l'alarme).

Hyster. Permet de spécifier la valeur d'hystérésis de l'alarme de vitesse, n'agit qu'au moment de la disparition d'une alarme de vitesse. Voir les paragraphes *RateUp* et *RateDown* ci-dessus.

Rate. Vitesse de changement effective calculée de *PV*, c'est à dire dPV/dt , qui peut être positive ou négative. Les unités de temps de *Rate* sont définies par le paramètre *TimeBase*.

SampTime. Temps d'échantillonnage de *PV* dans les unités spécifiées par *TimeBase*. L'intervalle entre les échantillons *PV* pris pour le calcul de dPV/dt . Sélectionner une valeur *SampTime* suffisamment petite pour suivre toutes les variations importantes de *PV*, mais pas trop petite, afin que les calculs ne produisent pas de vitesses zéro (en raison des erreurs d'arrondi).

TestTime. Intervalle de temps normal entre la disparition d'une alarme de vitesse et la sortie d'*OP* de l'état de maintien. Les unités sont spécifiées par *TimeBase*. Le déblocage d'*OP* peut se produire plus tôt suivant le comportement de *PV* et de la valeur de *TestBand* - voir le paragraphe *TestBand* ci-dessus.

TimeBase (Base de temps). (sec./min./heures). Permet de spécifier les unités de temps utilisées pour le bloc.

Status (Etat). Champ binaire indiquant l'état du bloc.

- **OPrelsd.** Ce bit passe à VRAI pour une itération de la tâche, puis repasse à FAUX, lorsqu'*OP* sort de l'état de maintien après l'apparition d'une alarme de vitesse. Voir les figures 6-11 et 6-12 qui illustrent l'action du drapeau *OPrelsd*.

RATE_LMT : BLOC LIMITE DE VITESSE

Fonction du bloc

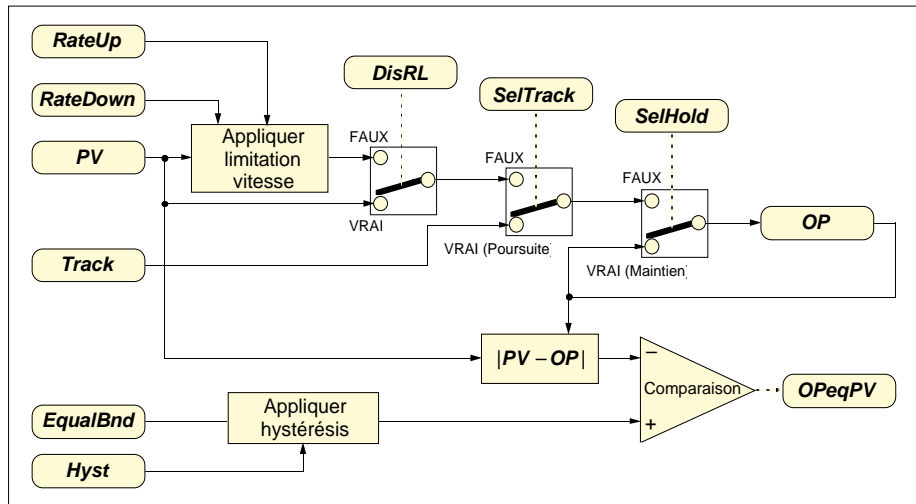


Figure 6-13 Schéma du bloc RATE_LMT

Voir le schéma de la figure 6-13. Le bloc RATE_LMT limite la vitesse de changement d'une sortie **OP**, lorsque celle-ci essaie de suivre une entrée **PV** de "valeur cible" dynamique. Des vitesses de croissance et de décroissance **OP** peuvent être définies séparément.

Un bit d'état **OPeqPV** se met à 1 lorsqu'**OP** est égal à **PV** dans les limites d'une tolérance spécifiée par **EqualBnd**, avec une tolérance d'hystérésis supplémentaire **Hyst** appliquée à mesure que l'état d'égalité disparaît. La figure 6-14 montre la manière dont **EqualBnd** et **Hyst** sont appliqués. Noter que la courbe doit être suivie dans le sens des flèches et n'est pas valable dans le sens inverse.

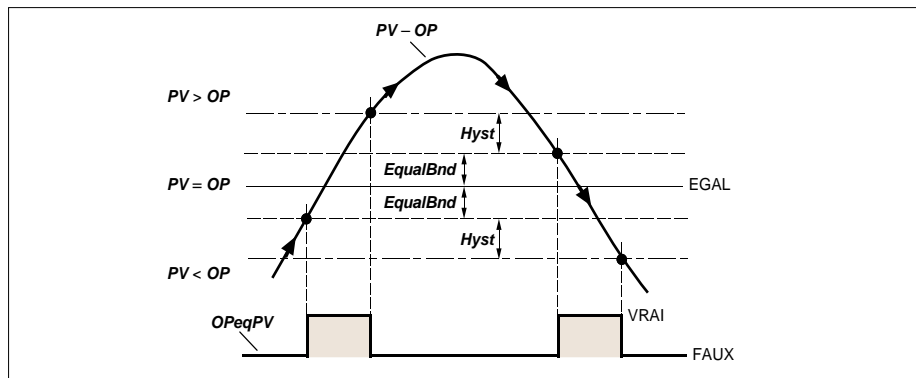


Figure 6-14 Bloc RATE_LMT - Action EqualBnd & Hyst

Paramètres du bloc

Les symboles utilisés dans la table 6-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement du bloc	Menu	
FallBack	Mode de reprise du bloc	Menu	
PV	Valeur cible	Eng	
Track	Valeur de sortie poursuite	Eng	
OP	Sortie de rampe	Eng	
HR	Echelle haute	Eng	
LR	Echelle basse	Eng	
EqualBnd	Erreur admissible PV – OP	Eng	
Hyst	Hystérésis pour OP = PV	Eng	
Alarms			
Software	Données corrompues	V/F	
Combined	Alarme combinée du bloc	V/F	
RateUp	Limite de vitesse croissante	Eng/Temps	
RateDown	Limite de vitesse décroissante	Eng/Temps	
Timebase	Unités de temps utilisées par le bloc	Menu	
Options			
SelTrack	Poursuite sélectionnée	V/F	
SelHold	Maintien sélectionné	V/F	
DisRL	Invalidation limitation de vitesse (VRAI)	V/F	
Status			
OPeqPV	Drapeau fin rampe	V/F	

Table 6-10 Paramètres du bloc RATE_LMT

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- Mode, FallBack (Mode, Reprise)** (Auto/Track/Hold - Auto/Poursuite/Maintien). Modes de fonctionnement et de reprise du bloc par l’intermédiaire des paramètres *Options* correspondants. Le schéma de la figure 6-13 montre les priorités de ces modes et leur action sur *PV* et *OP*. Noter que plus d’un mode peut être sélectionné à la fois, mais seul le mode à la priorité la plus haute est exploité. Le mode sélectionné à la priorité immédiatement inférieure (ou Auto par défaut) constitue le mode de reprise.
- **Auto.** Il s’agit du mode de fonctionnement normal du bloc. Le bloc est en mode Auto lorsque ni Track ou Hold n’ont été sélectionnés. En mode Auto, *OP* se rapproche de *PV* dans les limites de vitesse spécifiées, à moins qu’*Options./DisRL* ne soit VRAI - lorsqu’*OP* poursuit *PV*. Le mode Auto a la priorité la plus faible.
 - **Track (Poursuite).** En mode Track, *OP* poursuit la valeur de paramètre *Track*.

■ **Hold (Maintien).** En mode Hold, *OP* maintient la valeur de sortie au moment de la sélection de Hold. Le mode Hold a la priorité la plus haute.

PV. Valeur cible pour *OP*.

Track (Poursuite). Valeur qu'*OP* poursuit lorsque le bloc fonctionne en mode Track.

OP. Sortie du bloc. En mode automatique normal de limitation de vitesse, *OP* se rapproche de *PV* dans les limites des vitesses spécifiées. Voir le paragraphe *Mode*, *FallBack* pour les autres modes.

HR, LR. Echelle haute & basse pour les objets graphiques reliés à *PV* ou *OP* (bargraphe, tendance). *HR* et *LR* définissent respectivement les affichages 100 et 0 %.

EqualBnd. Bande égal. Permet de spécifier une bande de tolérance "symétrique" pour laquelle *PV* et *OP* sont définis comme étant égaux. Voir figure 6-14.

Hyst. Bande d'hystérésis. Permet de spécifier une bande de tolérance "asymétrique" ajoutée à la bande égal. Voir figure 6-14.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

RateUp. Limite de vitesse appliquée à *OP* à mesure que la sortie se rapproche d'une valeur *PV* supérieure.

RateDown. Limite de vitesse appliquée à *OP* à mesure que la sortie se rapproche d'une valeur *PV* inférieure.

TimeBase (Base de temps). (sec./min./heures). Permet de spécifier les unités de temps pour le bloc.

Options. Champ binaire permettant de sélectionner les modes de fonctionnement du bloc. Le mode de fonctionnement actif est celui qui a la priorité la plus haute.

■ **SelTrack.** VRAI sélectionne le mode Track (Poursuite).

■ **SelHold.** VRAI sélectionne le mode Hold (Maintien).

■ **DisRL.** VRAI invalide la limitation de vitesse en mode Auto, et *OP* poursuit *PV*.

Status (Etat). Champ binaire indiquant l'état du bloc.

■ **OPeqPV.** Ce bit est VRAI chaque fois que *OP* est égal à *PV* dans les limites de la tolérance spécifiée par *EqualBnd* et soumis à l'hystérésis définie par *Hyst*. Sinon, *OPeqPV* est FAUX.

TOT_CON: BLOC DE LIAISONS DE TOTALISATION

Fonction du bloc

Le bloc TOT_CON est utilisé comme réserve pour les valeurs des paramètres de 32 bits, en particulier lorsque les valeurs sont acquises sur une liaison Modbus, en utilisant la fonction des nombres à 32 bits à 2 registres. Ces grands nombres sont souvent générés par des calculs de totalisation. Le complément de paramètres est le même que pour le bloc TOTAL, mais avec un champ supplémentaire — *UserAlrm*. Le bloc TOT_CON par contre n'a pratiquement pas de sous-programme de mise à jour, et la plupart de ses champs représentent des sauvegardes de données en lecture/écriture à usage général. Les paramètres "fictifs" du bloc peuvent être utilisés pour maintenir des totaux et des valeurs associées pour communiquer avec d'autres noeuds LIN/ALIN.

En raison de leur structure similaire, les blocs TOT_CON et TOTAL sont recouvrables en mémoire cache, autrement dit, l'un ou l'autre bloc peut être utilisé comme image en mémoire cache de l'autre sur le réseau LIN ou ALIN, ce qui peut être utile dans des instruments qui gèrent les blocs TOTAL, mais pas les blocs TOT_CON.

Paramètres du bloc

Les symboles utilisés dans la table 6-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV	Point flottant à 32 bits (usage général)	Eng	
CutOff	Point flottant à 32 bits (réserve)	Eng	
Default	Point flottant à 32 bits (réserve)	Eng	
Scale	Point flottant à 32 bits (réserve)	Eng	
Total	32 bits sans signe (usage général)	Entier	
OP	Point flottant à 32 bits (=Total)	Eng	
HR_OP, LR_OP	Point flottant à 32 bits (échelle pour OP)	Eng	
Alarms			
Software	Données corrompues/défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Target	32 bits sans signe (usage général)	Entier	
Overflow	Booléen (usage général)	V/F	
ClearOv	Booléen (usage général)	V/F	
Disable	Booléen (usage général)	V/F	
Pulse	Booléen (usage général)	V/F	
Init	Booléen (usage général)	V/F	
Timebase	Enumération (usage général)	Menu	
UserAlrm	Booléen (VRAI déclenche l'alarme logiciel)	V/F	

Table 6-11 Paramètres du bloc TOT_CON

Menu de spécifications du bloc

Les indications suivantes sont complémentaires de celles de la table 6-11.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

OP. Version point flottant à 32 bits de *Total*, écrite par le sous-programme de mise à jour. Mise à l'échelle par *HR_OP* et *LR_OP*.

UserAlrm. VRAI déclenche l'alarme logiciel du bloc.

NOTA. Le champ *UserAlrm* peut être relié au champ *TableOfI* du bloc MDBDIAG, afin qu'un système SCADA puisse détecter un défaut de communication Modbus.

TOTAL2: BLOC DE TOTALISATION

Fonction du bloc

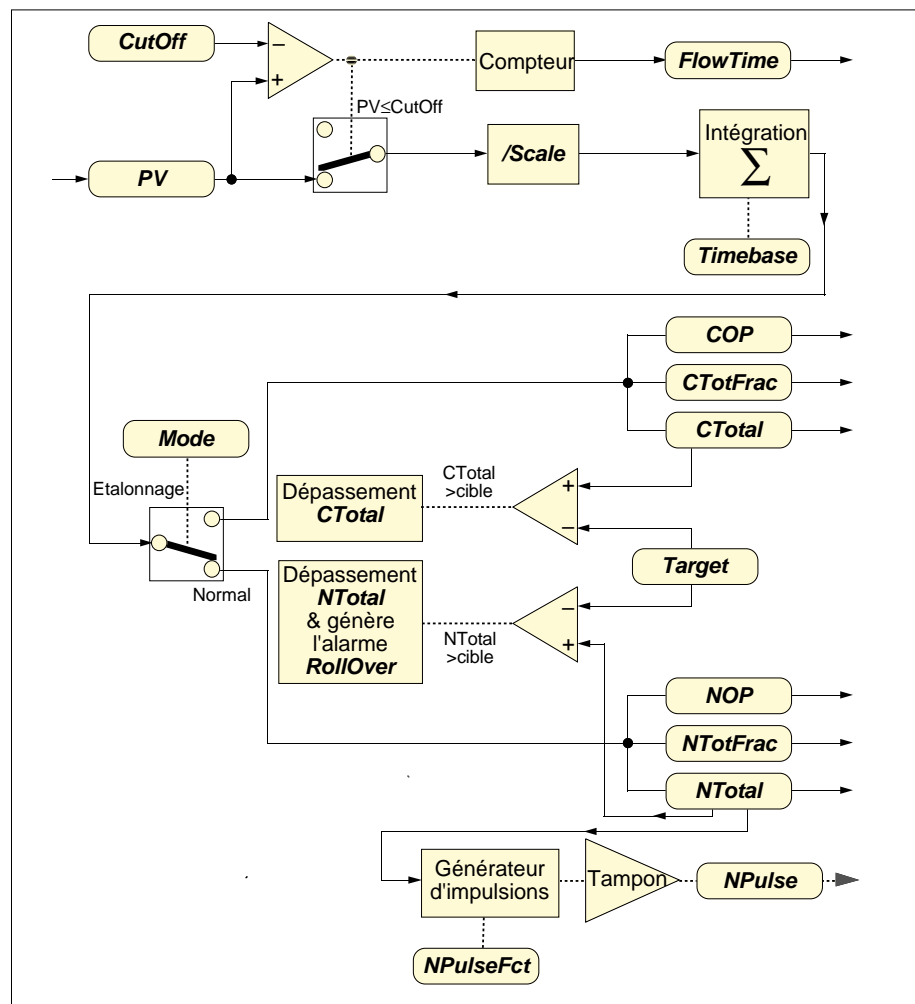


Figure 6-15 Schéma du bloc TOTAL2

Voir figure 6-15. En mode "Normal", le bloc TOTAL2 intègre une entrée analogique *PV*, qui enregistre une partie du résultat dans le paramètre *NTotal*, la partie fractionnelle dans *NTotFrac*, et le total en format à point flottant dans *NOP*. Ces totaux "normaux" ne peuvent être remis à zéro. *Scale* définit la taille des unités intégrées et *Timebase* définit *PV*, ainsi que les unités d'intégration de temps.

Le bloc comprend une coupure *PV* à seuil bas avec une totalisation à zéro défaut, une sortie de compteur de débit *FlowTime*, une sortie d'impulsion avec buffeur pour les compteurs, une valeur de dépassement (*Target*) et un drapeau de dépassement (*RollOver*).

En mode "Calib" (étalonnage), *PV* intégrée et mise à l'échelle est ajouté à *CTotal*, *CTotFrac* et *COP* au lieu des paramètres "normaux" correspondants. Le passage en mode étalonnage remet à zéro ces totaux, mais pas les totaux normaux.

Le bloc dispose d'alarmes du mode étalonnage, de faible débit et de dépassement, en plus des alarmes logicielle et combinée habituelles.

Paramètres du bloc

Les symboles utilisés dans la table 6-12 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement du bloc	Menu	
PV	Variable procédé (entrée du bloc)	Eng1	
CutOff	Spécification du seuil bas de la valeur PV	Eng1	
Scale	Valeur de mise à l'échelle de l'entrée		
Timebase	Définit les unités de temps de l'intégration	Menu	
HR_PV, LR_PV	Echelles (graphiques) haute & basse de PV	Eng1	
Alarms			
Software	Données corrompues/défaut de communication	V/F	
Calib	Utilisation du mode étalonnage	V/F	
NoFlow	PV ≤ CutOff	V/F	
RollOver	NTotal a dépassé Target & remise à zéro	V/F	
(Réserve)		V/F	
Combined	Alarme combinée du bloc	V/F	
Target	Valeur à laquelle NTotal & CTotal repassent à 0	Entier	
NTotal	Total mode normal, part entière	Entier	
NTotFrac	Total mode normal, part fractionnelle	Eng2	
NOP	Total mode normal, format à point flottant	Eng2	
HR_NOP, LR_NOP	Echelles (graphiques) haute & basse de NOP	Eng2	
NPulsFct	Nbre de totalisations NTotal égal à 1 impulsion	Entier	
NPulse	Sortie impulsion intégrateur (tamponnée)	V/F	
CTotal	Total mode étalonnage, part entière	Entier	
CTotFrac	Total mode étalonnage, part fractionnelle	Eng2	
COP	Total mode étalonnage, format à point flottant	Eng2	
HR_COP, LR_COP	Echelles (graphiques) haute & basse de COP	Eng2	
FlowTime	Délai dernier dépassement de CutOff par PV	Secs	
ClearRol	Supprime l'alarme de dépassement	V/F	

Table 6-12 Paramètres du bloc TOTAL2

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Mode. (Normal/Calib) Mode de fonctionnement du bloc. En mode "normal", l'entrée de variable procédé mise à l'échelle et validée est ajoutée à l'ensemble des totaux normaux du bloc (*NTotal*, *NTotFrac*, and *NOP*). En mode "Calib" (étalonnage), l'entrée mise à l'échelle est ajoutée aux totaux d'étalonnage (*CTotal*, *CTotFrac*, and *COP*). Notez que le mode ne peut être modifié si *PV* dépasse la valeur *CutOff* spécifiée. Au passage en mode étalonnage, les totaux d'étalonnage sont mis à zéro, mais pas à la sortie du mode. (Notez que les totaux normaux ne sont jamais remis à zéro).

PV. Signal d'entrée analogique à totaliser (ex. débit en kg/hr). Si *PV* atteint ou descend en dessous de la valeur spécifiée par *CutOff*, la totalisation s'arrête.

CutOff (Coupure). *PV* est validé par comparaison au champ *CutOff*. Si *PV* est inférieur ou égal à *CutOff*, la totalisation s'arrête.

NOTA. Si une valeur totalisée par défaut est nécessaire au moment de la coupure, elle peut être obtenue par l'intermédiaire du paramètre *PV* en recouvrant l'entrée/sortie réelle.

Scale (Echelle). Met à l'échelle *PV* en rendant la variable à totaliser égale à *PV/Scale*. Si, par exemple, *Scale* = 1000,0 et *PV* est en kg/h, alors les totaux sont en tonnes/h (*Timebase* étant en "heures"). Notez que si vous modifiez *Scale* pendant le comptage, les totaux ne sont pas remis à zéro.

Timebase (Base temps). (Sec/Min/Heures/Jours) Définit les unités de temps de *PV* et de la fonction d'intégration. La même base temps est appliquée aux totaux normaux et d'étalonnage. *Timebase* peut être modifié dans tous les modes et à tout moment.

HR_PV, LR_PV. Echelle haute & basse pour les objets graphiques associés à *PV* (Barre, Tendence). *HR_PV* et *LR_PV* définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software.** Anomalie mémoire des données RAM du bloc.
- **Calib.** Utilisation du mode étalonnage. Lorsqu'il est appliqué, l'alarme *NoFlow* est supprimée.
- **NoFlow.** Déclenchée si *PV* est inférieur ou égal à *CutOff*. Aucune hystérésis n'est appliquée.
- **RollOver.** Cette alarme est déclenchée, chaque fois que *NTotal* dépasse *Target* et repasse à zéro. Notez que lorsque *RollOver* est appliqué, l'opérateur peut acquitter l'alarme comme d'habitude, mais elle reste déclenchée tant que *CleaRol* n'a pas fait l'objet d'une écriture (par un technicien qualifié, par exemple).
- **Combined.** Alarme combinée du bloc.

Target (Cible). Cet entier définit la valeur à laquelle *NTotal* et *CTotal* repassent à zéro, autrement dit, sont remis à zéro. La valeur par défaut de *Target* est 999 999 999. Etant un entier, *Target* ne peut être associé à un bargraphe.

Notez que si vous modifiez *Target* en cours d'exploitation, les totaux ne sont pas remis à zéro, sauf si la nouvelle valeur de *Target* est inférieure au total en cours.

NTotal. La partie intégrale du total "normal", qui ne s'incrémente que lorsque *Mode* est "Normal". *NTotal* repasse à zéro lorsqu'il atteint la valeur définie dans *Target*. Notez que *NTotal* est un entier long à signe avec une valeur théorique maximale de 2147483647. Dans la mesure où il a un signe, *NTotal* dispose d'une bonne connectivité par l'intermédiaire du réseau LIN à d'autres champs similaires, par ex. *I0* à *I7* dans les blocs ACTION. *NTotal* ne peut faire l'objet d'écritures, une fois que le bloc est configuré ou pendant l'exécution de la base de données.

NOTA. Si la base de données est enregistrée, *NTotal* est maintenu comme valeur de champ de la base de données LIN. Donc, en cas de démarrage à froid, le total indiqué s'incrémente à partir de la valeur enregistrée, sauf si la base de données n'a jamais été enregistrée. Dans ce cas, *NTotal* commence à zéro.

NTotFrac. Nombre à point flottant à simple précision, qui représente la part fractionnelle du total normal (*NTotal*). Ne s'incrémente que lorsque *Mode* est "Normal".

NOP. Forme à point flottant de *NTotal*. *NOP* peut être utilisé pour piloter des affichages de tendance ou des bases de données de la série T3000 et des scripts de commandes qui n'acceptent pas les entiers. Notez que pour les valeurs *NTotal* très importantes, *NOP* a une résolution inférieure à *NTotal*.

HR_NOP, LR_NOP. Echelle haute & basse pour les objets graphiques associés à *NOP* (Barre, Tendance). *HR_NOP* et *LR_NOP* définissent les affichages 100 % et 0 %.

NPulsFct. Entier long sans signe qui définit le nombre d'incréments de *NTotal* pour sortir une "impulsion" par l'intermédiaire du champ *NPulse*. Si, par exemple, la valeur de *NPulsFct* est de 1000 et que la totalisation est en kg, la sortie de l'impulsion représentera le total en tonnes. (Voir *NPulse* ci-dessous).

NPulse. Cet opérateur booléen passe à l'état VRAI pendant un cycle de la base de données, lorsque *NTotal* a été incrémenté des décomptes *NPulsFct*. (Voir *NPulsFct* ci-dessus). Si plus d'une impulsion est nécessaire par cycle de la base de données, les impulsions supplémentaires sont conservées en mémoire tampon et émises pendant les périodes "plus calmes" pour éviter de les perdre. Notez que si vous modifiez *Timebase* ou *Scale*, le nombre d'impulsions qui se trouvent en mémoire tampon n'en est pas affecté. Les impulsions en attente continuent à être sorties de la mémoire tampon, et toute nouvelle impulsion (avec une mise à l'échelle éventuellement différente) est ajoutée à la mémoire tampon.

NOTA. La mémoire tampon évite la perte d'impulsions dans le bloc lui-même, mais certaines peuvent se perdre lorsqu'elles sont transmises sur le réseau LIN, si les temps de scrutation des systèmes concernés sont différents. Si la réduction de l'intervalle des impulsions - en incrémentant *NPulsFct* — peut résoudre le problème, il n'est cependant pas conseillé de contrôler les impulsions sur le réseau LIN.

CTotal. La partie intégrale du total d'étalonnage, qui ne s'incrémente que lorsque *Mode* est "Calib". *CTotal* repasse à zéro, lorsque la valeur définie dans *Target* est atteinte. *CTotal* ne peut faire l'objet d'écritures lorsque le bloc est configuré ou pendant l'exécution de la base de données. Mais, *CTotal*, *CTotFrac* et *COP* sont mis à zéro à chaque fois que le mode sélectionné est "Calib".

CTotFrac. Nombre à point flottant à simple précision, qui représente la partie fractionnelle du total d'étalonnage (*CTotal*). Ne s'incrémente que lorsque *Mode* est "Calib". *CTotFrac* est mis à zéro, lorsque le mode d'étalonnage est sélectionné.

COP. Forme à point flottant de *CTotal*. Notez que pour les valeurs *CTotal* très importantes, la résolution de *COP* est inférieure à celle de *CTotal*.

HR_COP, LR_COP. Echelle haute & basse pour les objets graphiques associés à *COP* (Barre, Tendance). *HR_COP* et *LR_COP* définissent les affichages 100 % et 0 %.

FlowTime. S'incrémente pour afficher la durée en secondes depuis le dernier dépassement par le débit mesuré de la valeur *CutOff*, quelque soit le mode sélectionné. Lorsque le débit mesuré est inférieur ou égal à *CutOff*, l'horloge ne s'incrémente plus. A mesure que le débit augmente et dépasse la valeur de *CutOff*, *FlowTime* est réinitialisé et s'incrémente à nouveau.

ClearRol. VRAI déclenche l'alarme *RollOver*, et ensuite *ClearRol* repasse automatiquement à FAUX.

[Page laissée intentionnellement blanche]

Chapitre 7 BLOCS DE FONCTION SELECTION

SELECT : BLOC SELECTEUR

Fonction du bloc

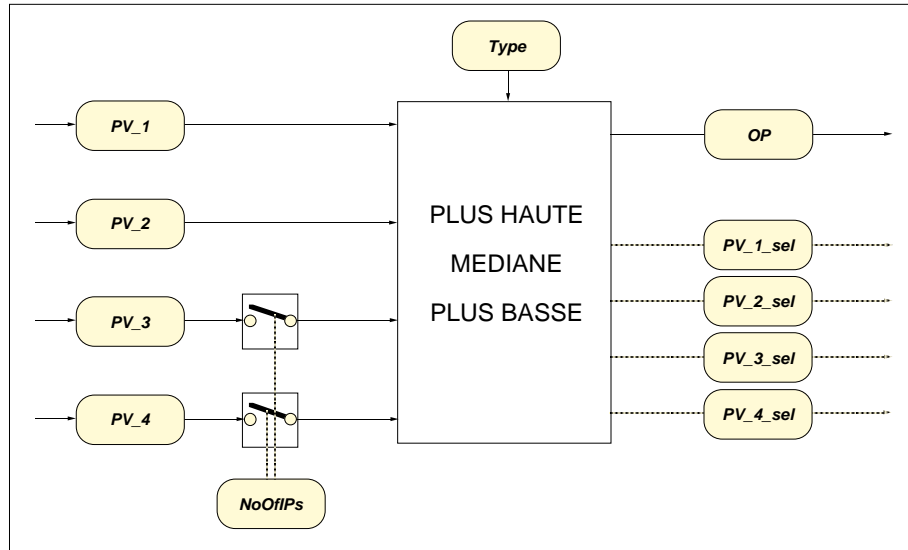


Figure 7-1 Schéma du bloc SELECT

Voir la figure 7-1. Le bloc SELECT dispose de quatre liaisons d'entrée et peut être configuré pour sélectionner la valeur la plus haute ou la plus basse de deux, trois ou quatre entrées ou la valeur médiane de trois entrées. L'entrée sélectionnée est indiquée par une sortie logique et sa valeur est retransmise comme une sortie analogique *OP*.

Paramètres du bloc

Les symboles utilisés dans la table 7-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Type (Type). (HIGHEST/MIDDLE/LOWEST - PLUS HAUTE/MEDIANE/PLUS BASSE). Spécifie la fonction du bloc.











Paramètre	Fonction	Unités	Etat
Type	Spécifie la fonction du bloc	Menu	
NoOfIPs	Spécifie le nombre d'entrées (2 - 4)	Entier	
PV_1 à PV_4	Entrées 1 à 4	Eng	
Alarms			  
Software	Données corrompues/Défaut de communication		V/F
Combined	Alarme combinée du bloc	V/F	
OP	Valeur d'entrée sélectionnée	Eng	 
PV_1_sel à PV_4_sel	PV_1 à PV_4 sélectionné	V/F	 
HR_OP, LR_OP	Echelle haute & basse objets graphiques (PV_n, OP)Eng		 

Table 7-1 Paramètres du bloc SELECT

NoOfIPs (Nombre d'entrées). Permet de spécifier le nombre d'entrées pour les fonctions plus hautes et plus basses. Lorsque la fonction médiane est sélectionnée, trois entrées sont automatiquement prises par défaut.

PV_1 à PV_4. Valeurs d'entrée analogique de 1 à 4.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

OP. Valeur analogique active de l'entrée sélectionnée.

PV_1_sel à PV_4_sel (PV1 choisi à PV4 choisi). Indique laquelle des entrées correspondantes *PV_1* à *PV_4* est sélectionnée. VRAI = sélectionnée.

HR_OP, LR_OP. Echelle haute & basse pour les objets graphiques liés à une entrée *PV_n*, ou à la sortie *OP* (bargraphe, tendance). *HR_OP* et *LR_OP* définissent les affichages 100 % et 0 %.

SWITCH : BLOC COMMUTATION

Fonction du bloc

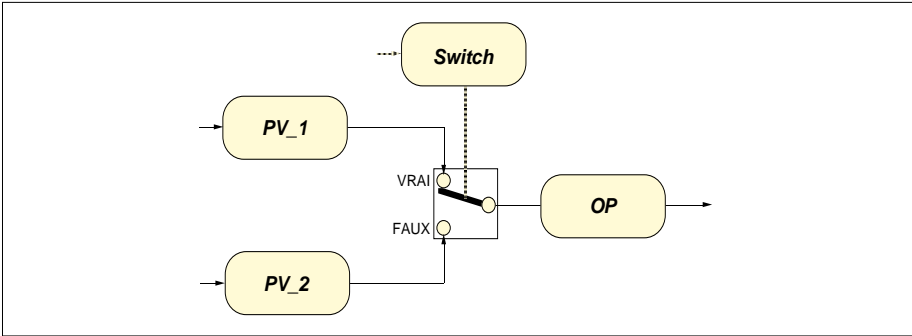


Figure 7-2 Schéma du bloc SWITCH

Voir la figure 7-2. Le bloc SWITCH fonctionne comme un commutateur unipolaire à deux directions. Lorsque le paramètre *Switch* est VRAI, la sortie *OP* transmet *PV_1*. Lorsque *Switch* est FAUX, *OP* transmet *PV_2*.

Paramètres du bloc

Les symboles utilisés dans la table 7-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Switch	Sélection entrée	V/F	
PV_1	Entrée 1	Eng	
PV_2	Entrée 2	Eng	
HR, LR	Echelle haute & basse objets graphiques (PV_n, OP)	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
OP	Valeur d'entrée sélectionnée retransmise	Eng	

Table 7-2 Paramètres du bloc SWITCH

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- Switch (Sélection).** (VRAI/FAUX). Sélection de l’entrée. VRAI commute la sortie *OP* sur l’entrée *PV_1*, et FAUX sur *PV_2*.

PV_1, PV_2. Valeurs des entrées analogiques 1 et 2.

HR, LR. Echelle haute & basse pour les objets graphiques liés à une entrée *PV_n*, ou à la sortie *OP* (bargraphe, tendance). *HR* et *LR* définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

OP. Valeur analogique active de l'entrée sélectionnée. Noter que *OP* peut faire l'objet d'une écriture (par ex. par l'intermédiaire du réseau LIN), ce qui permet de créer un verrouillage en reliant *OP* à *PV_1* ou *PV_2*.

ALC : BLOC DE REGROUPEMENT D’ALARMES

Fonction du bloc

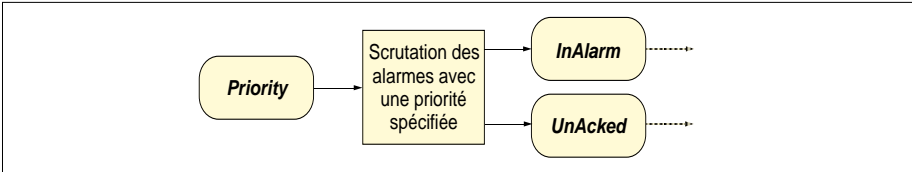


Figure 7-3 Schéma du bloc ALC

Voir la figure 7-3. Le bloc de regroupement d’alarmes simplifie le contrôle des alarmes en regroupant les alarmes ayant la même priorité spécifiée en une seule sortie logique d’alarme combinée.

Paramètres du bloc

Les symboles utilisés dans la table 7-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Priority	Priorité des alarmes regroupées (0-15)	Entier	
InAlarm	Sortie des alarmes regroupées active	V/F	
UnAcked	Sortie des alarmes regroupées non-acquittées	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 7-3 Paramètres du bloc ALC

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- Priority (Priorité).** Spécifie le niveau de priorité des alarmes à regrouper. Une priorité zéro invalide le bloc.
- InAlarm (En alarme).** VRAI indique une alarme active avec le niveau de priorité spécifié.
- UnAcked (Non-acquitté).** VRAI indique une alarme non-acquittée avec le niveau de priorité spécifié. Pour les alarmes avec une priorité de 6 ou plus, le drapeau *UnAcked* est verrouillé à VRAI, jusqu’à l’acquiescement de l’alarme, même si elle n’est plus active. Les alarmes de priorité 5 ou moins n’ont pas besoin d’être acquittées, et donc, pour ces alarmes le drapeau *UnAcked* reste FAUX.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

PAGE : BLOC PAGE

Fonction du bloc

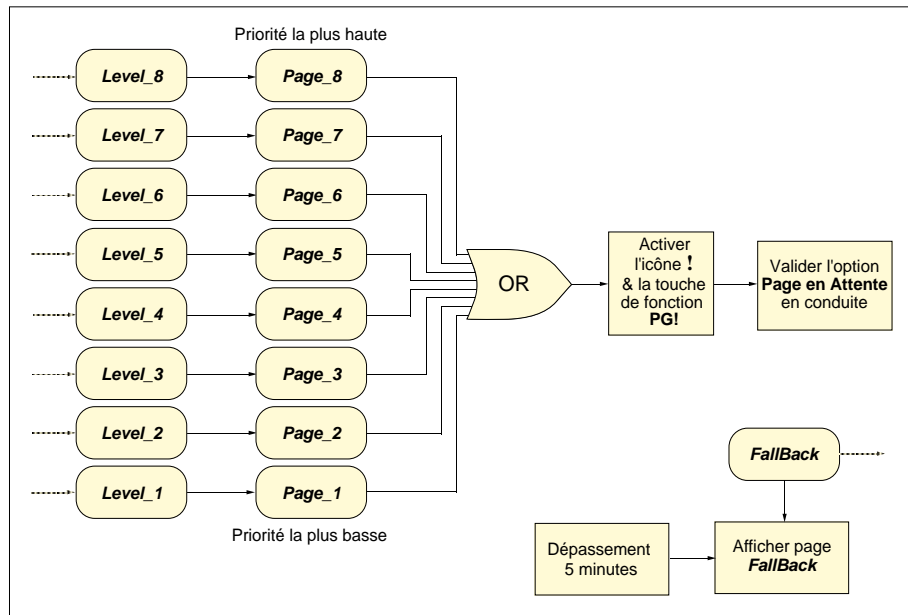


Figure 7-4 Schéma du bloc PAGE

Voir la figure 7-4. L'option *Page-Pending* (page en attente) de la conduite du T1000 est configurée par l'intermédiaire du bloc PAGE. Les huit entrées logiques mises en priorité peuvent activer jusqu'à huit pages ou fenêtres d'objets graphiques, qui sont mises en attente à l'attention de l'opérateur. Lorsqu'au moins une des pages est active, le symbole clignotant ! apparaît dans le coin supérieur gauche de l'écran de conduite. Les pages activées peuvent alors être affichées dans l'ordre de priorité en appuyant sur la touche **PG!**, et l'opérateur peut alors prendre les mesures nécessaires. L'option page en attente est décrite intégralement dans le *Manuel de référence Tacticien T1000/T100* (référence HA 080 194 U003).

Le bloc PAGE permet également de définir une page graphique comme étant l'affichage de reprise ou de "dépassement du temps imparti" (timeout), qui apparaît automatiquement après cinq minutes sans intervention de l'opérateur.

NOTA. Un seul bloc PAGE peut être configuré par schéma de boucles.

Paramètres du bloc

Les symboles utilisés dans la table 7-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.






Paramètre	Fonction	Unités	Etat
FallBack	Numéro de la page de reprise	Entier	
Level_8 à Level_1	Entrées d'activation Page_8 à Page_1	V/F	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Page_8 à Page_1	Numéros de page activés par Level_8 à Level_1	Entier	

Table 7-4 Paramètres du bloc PAGE

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

FallBack (Reprise). Spécifie le numéro de la page (ou fenêtre) graphique qui est affichée automatiquement, lorsqu’il n’y a pas de dialogue opérateur avec les claviers du T1000 pendant 5 minutes. Un numéro de page zéro ou inexistant invalide la fonction reprise.

Level_8 à Level_1 (Niveau_8 à Niveau_1). Ces entrées activent les pages graphiques de conduite spécifiées dans les paramètres *Page_8 à Page_1*. (VRAI = activée). Les pages activées sont mises en attente dans l’option page en attente avec un ordre décroissant de priorité : *Level_8* a la priorité la plus élevée et *Level_1* la plus basse.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

Page_8 à Page_1. Permet de spécifier les numéros des pages graphiques de conduite activées par les entrées logiques *Level_8 à Level_1*. La *Page_8* a la priorité la plus élevée dans l’option de page en attente, et *Page_1* la plus basse.

2OF3VOTE : BLOC MEILLEURE MOYENNE

Fonction du bloc

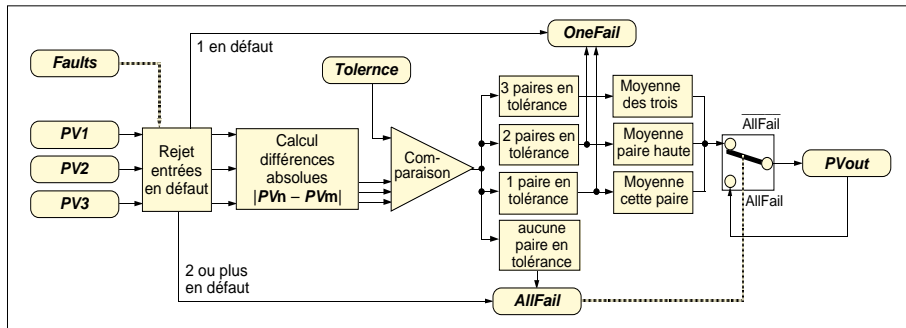


Figure 7-5 Schéma du bloc 2OF3VOTE

Voir le schéma de la figure 7-5. Le bloc 2OF3VOTE permet de sortir une valeur de “meilleure” moyenne (moyenne arithmétique) $PVout$ calculée à partir de trois entrées analogiques $PV1$, $PV2$ et $PV3$.

Si l’une des entrées PVn est en défaut - suivant les déclarations par les entrées dans le champ binaire *Faults* - celle-ci est rejetée et n’est pas utilisée dans la moyenne. Si deux ou plusieurs entrées PVn sont en défaut, les trois sont rejetées et $PVout$ n’est pas recalculé, mais maintient sa dernière bonne valeur.

La variation des entrées PVn qui ne sont pas en défaut est également vérifiée, en comparant leurs différences avec une valeur de tolérance spécifiée par l’utilisateur (*Tolerance*). La moyenne est produite uniquement à partir des valeurs PVn qui ne diffèrent entre elles que par des valeurs inférieures ou égales à *Tolerance*. Si aucune paire d’entrées n’est dans les limites de la tolérance, elles sont toutes rejetées et $PVout$ maintient à nouveau sa dernière bonne valeur.

Les bits d’alarme appropriés sont mis à 1 lorsque les entrées sont en défaut ou hors tolérance.

Paramètres du bloc

Les symboles utilisés dans la table 7-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV1, PV2, PV3. Entrées analogiques à partir desquelles $PVout$ est calculé.

Paramètre	Fonction	Unités	Etat
PV1	Entrée 1	Eng	
PV2	Entrée 2	Eng	
PV3	Entrée 3	Eng	
HR, LR	Echelle graphique haute & basse PVn, PVout, Tolernce	Eng	
Faults			
PV1fault	Défaut sur PV1	V/F	
PV2fault	Défaut sur PV2	V/F	
PV3fault	Défaut sur PV3	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
AllFail	Toutes les entrées rejetées	V/F	
OneFail	Une entrée rejetée	V/F	
Combined	Alarme combinée du bloc	V/F	
Tolernce	Tolérance de dérive	Eng	
PVout	PV résultant	Eng	

Table 7-5 Paramètres du bloc 2OF3VOTE

HR, LR. Echelle haute et basse pour les paramètres des unités physiques (voir table 7-5). Voir également échelle haute/basse pour les objets graphiques liés à ces paramètres (bargraphe, tendance). *HR* et *LR* définissent respectivement les affichages 100 % et 0 %.

Faults (Défauts). Champ binaire de 3 bits définissant la présence de défauts sur chacune des sorties *PV1*, *PV2*, *PV3*. Un bit *PVnfault* VRAI indique un défaut sur l'entrée *PVn*. Une entrée en défaut ne peut être utilisée pour le calcul de *PVout*.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **AllFail (Toutes en défaut).** Cette alarme est signalée si deux ou plusieurs entrées *PVn* sont définies comme étant en défaut (par le paramètre *Fault*) ou si aucune paire de valeurs *PVn* ne présente de différence d'une valeur inférieure ou égale à *Tolernce*. Une alarme *AllFail* VRAIE empêche le recalcul de *PVout*, en le maintenant à sa dernière bonne valeur.

■ **OneFail (Une en défaut).** Cette alarme est signalée si seulement l'une des entrées *PVn* est définie comme étant en défaut (par le paramètre *Faults*) ou si seulement une ou deux paires de valeurs *PVn* présentent une différence d'une valeur inférieure ou égale à *Tolernce*.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Tolernce (Tolérance). Permet de spécifier la valeur maximale de différence que peuvent présenter des paires d'entrées *PVn* avant qu'elles ne soient considérées comme "hors tolérance". Le nombre de paires d'entrées non en défaut qui sont dans les limites de la tolérance détermine comment - ou si - *PVout* est recalculé. Voir figure 7-5 et le paragraphe *PVout* ci-après.

PVout. Il s'agit de la "meilleure" moyenne calculée à partir des trois entrées *PV1* à *PV3*.

Sous réserve du rejet d'une ou d'entrées en défaut, le bloc calcule les différences absolues entre les trois paires d'entrées possibles comme $|PV1-PV2|$, $|PV1-PV3|$ et $|PV2-PV3|$. Ces valeurs sont comparées à *Tolernce* pour déterminer combien de paires sont dans les limites de la tolérance et *PVout* est alors calculé en conséquence. La table 7-6 décrit le fonctionnement, donne des exemples simples et indique comment les bits d'alarme sont mis à 1.

Paires en tolérance	Alarmes	PVout moyenne de :	Exemple (Tolernce = 2.0)			
			PV1	PV2	PV3	PVout
3	(Aucune)	PV1, PV2, PV3	<u>7.0</u> *	<u>6.0</u>	<u>5.0</u>	6.0
2	OneFail	Paire de valeur supérieure uniquement	4.0	<u>6.0</u>	<u>8.0</u>	7.0
1	OneFail	Cette paire uniquement	<u>4.0</u>	<u>6.0</u>	9.0	5.0
0	AllFail	(pas de recalcul)	3.0	6.0	9.0	(Maintien)

* Seules les valeurs soulignées permettent de produire une moyenne

Table 7-6 Calcul de PVout

TAG : BLOC REPERE

Fonction du bloc

Le bloc TAG fournit une chaîne de caractères qui est normalement affichée dans l’affichage de repère de la face avant du T640, lorsqu’une tâche utilisateur contenant le bloc occupe l’affichage principal de la boucle. Il devrait y avoir un bloc TAG par tâche utilisateur. Si un bloc TAG n’est pas présent dans la tâche utilisateur active sur la face avant, l’affichage de repère est généré suivant les règles données dans le *Manuel d'utilisation du T640* (réf. HA 082 468 U005) dans le chapitre *Affichages Opérateur & Commandes*.

Paramètres du bloc

Les symboles utilisés dans la table 7-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.


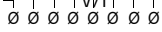





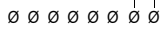
Paramètre	Fonction	Unités	Etat
SelTag			
Tag1	VRAI sélectionne le champ Tag1 (priorité la plus basse)		V/F
Tag2	VRAI sélectionne le champ Tag2	V/F	
Tag3	VRAI sélectionne le champ Tag3	V/F	
Tag4	VRAI sélectionne le champ Tag4	V/F	
Tag5	VRAI sélectionne le champ Tag5	V/F	
Tag6	VRAI sélectionne le champ Tag6	V/F	
Tag7	VRAI sélectionne le champ Tag7	V/F	
Tag8	VRAI sélectionne le champ Tag8 (priorité la plus haute)		
Tag1 à Tag8	Chaînes de texte à 8 caractères ou noms de champ	Alphanumérique	
TAG	Affichage repère réel si boucle sur face avant	Alphanumérique	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Block1 à Block8	Références du bloc, comme sources de Tag1-Tag8	Nom	
Options			
NoMsg	Suppression des messages courants de la face avant	V/F	
NoInval	Suppression du message "INVALID" de la face avant	V/F	

Table 7-7 Paramètres du bloc TAG

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

SelTag. Un champ binaire de 8 bits permettant de sélectionner lequel des paramètres *Tag1* à *Tag8* sont des sources possibles pour l’affichage de repère de la face avant du

T600. Si aucun bit n'est VRAI, le nom de bloc TAG est le repère par défaut pour cette tâche utilisateur. Si plus d'un bit est VRAI, le bit de poids fort sélectionne le champ *Tagn* correspondant comme la source du repère pour cette tâche utilisateur.

Tag1 à Tag8 (Repère1 à Repère8). Il s'agit de chaînes à 8 caractères fournissant 8 repères possibles pour l'affichage de repère.

TAG (REPERE). Affiche toujours le repère actif pour la boucle, c'est à dire ce qui est en fait affiché sur l'affichage de repère de la face avant du T600, lorsque cette tâche utilisateur occupe l'affichage principal de la boucle.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Block1 à Block8 (Bloc1 à Bloc8). Ces champs renvoient à d'autres blocs - ou même à ce bloc TAG - par le nom (paramètre *Block*). Si un champ *Blockn* contient une référence de bloc valable, le champ *Tagn* correspondant est pris comme un nom de champ et le contenu du bloc/champ référencé est affiché comme le repère pour cette tâche utilisateur. Si le nom de champ est incorrect, "**BadField**" (Champ erroné) est affiché dans le repère.

Options.

- **NoMsg.** VRAI invalide tous les messages d'affichage standard de repère, ce qui permet de retenir le repère constamment sauf au cours du mode inspection de la base de données.
- **NoInval.** VRAI supprime le message "**INVALID**" (INCORRECT), utile lorsque le schéma de boucles utilise des combinaisons de touches normalement incorrectes pour générer des fonctions spéciales.

[Page laissée intentionnellement blanche]

Chapitre 8 BLOCS DE FONCTION LOGIQUE

PULSE : BLOC IMPULSIONS

Fonction du bloc

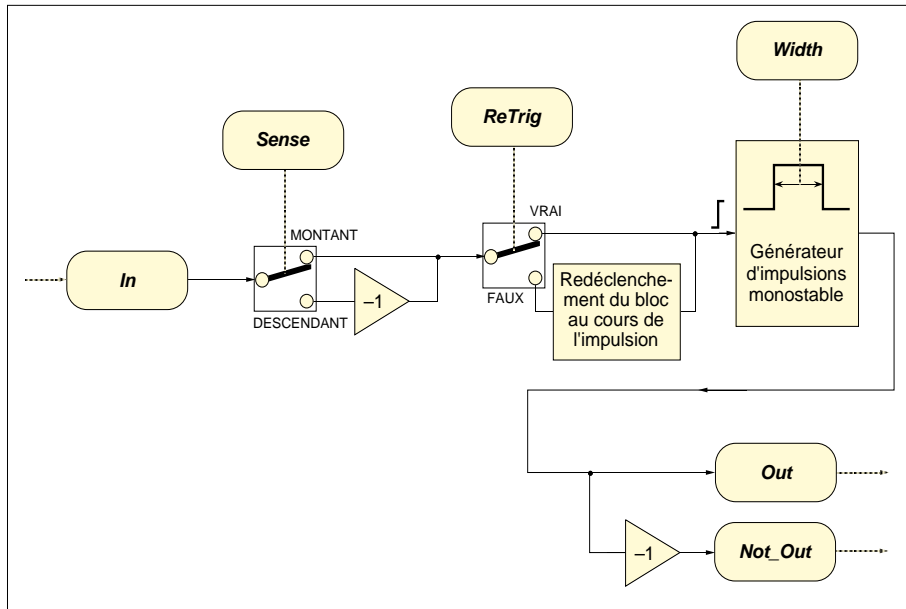


Figure 8-1 Schéma du bloc PULSE

Voir la figure 8-1. Le bloc PULSE agit comme un générateur d'impulsions monostable qui produit une impulsion rectangulaire, lorsqu'il est déclenché par l'intermédiaire de l'entrée. Les fronts montants ou descendants peuvent être spécifiés comme déclencheur (par le paramètre *Sense*), et la sortie est disponible sous forme vraie et complémentaire. Le paramètre *ReTrig* peut être mis à 1 pour empêcher les fronts d'entrée de redéclencher l'impulsion monostable au cours de la période "on".

La figure 8-2 montre des exemples et la manière dont ces paramètres affectent la sortie des impulsions.

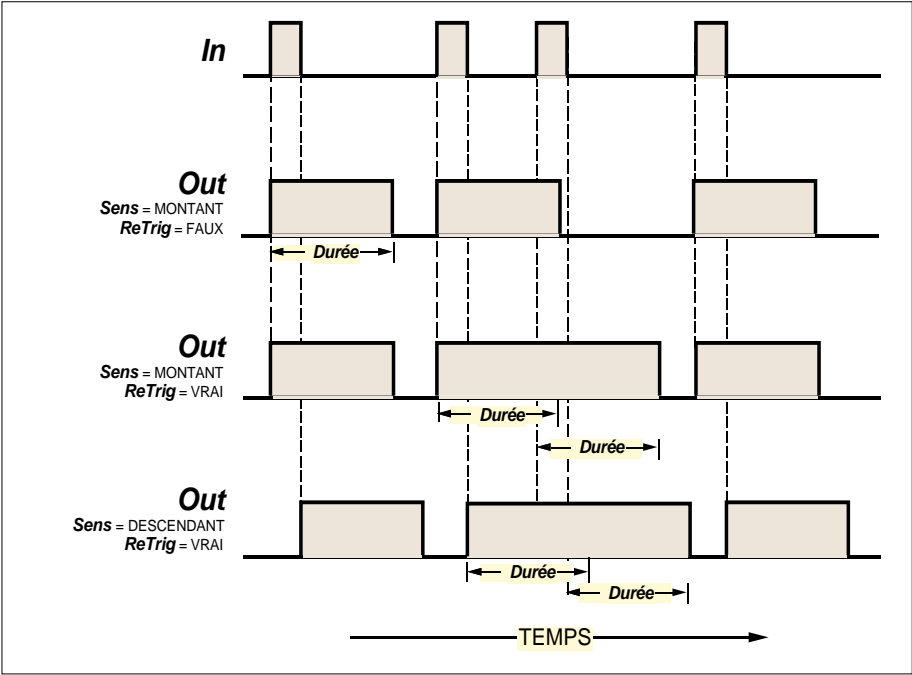


Figure 8-2 Sortie du bloc PULSE (Exemples)

Paramètres du bloc

Les symboles utilisés dans la table 8-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
In	Entrée de déclenchement	V/F	
Width	Durée de l'impulsion monostable	Secs	
Sense	Sélection front de déclenchement de l'entrée	Menu	
ReTrig	Validation redéclenchement de l'horloge impulsions		V/F
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Out	Sortie	V/F	
Not_Out	Sortie complémentaire	V/F	

Table 8-1 Paramètres du bloc PULSE

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

In (Entrée). Le front d’entrée de déclenchement - soit descendant ou montant - suivant les spécifications du paramètre *Sense* (sens). (Si le bloc est exécuté sans une liaison d’entrée, *In* revient automatiquement à son état de repos, lorsque l’impulsion déclenchée repasse à FAUX).

Width (Durée). Durée de l’impulsion monostable en secondes. La durée peut être modifiée, lorsque l’impulsion monostable est “on”.

Sense (Sens). (FALLING/RISING - DESCENDANT/MONTANT). Permet de spécifier le front nécessaire pour déclencher l’impulsion de sortie. Descendant = VRAI à transition d’entrée FAUX, et Montant = FAUX à transition d’entrée VRAI.

ReTrig (Redéclenchement). Validation redéclenchement. *ReTrig* étant VRAI, le bloc émule une impulsion monostable conventionnelle, dont l’horloge peut être redéclenchée à tout moment. *ReTrig* étant FAUX, les entrées de redéclenchement sont ignorées, tandis que l’impulsion de sortie est “on” (c’est à dire *Out* = TRUE). La figure 8-2 montre ces effets.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

Out (Sortie). Sortie de la fonction monostable, VRAIE lorsque l’impulsion est “on”.

Not_Out (Non sortie). Egal au complément logique de *Out*. Forme complémentaire de la sortie du bloc, FAUX lorsque l’impulsion est “on”.

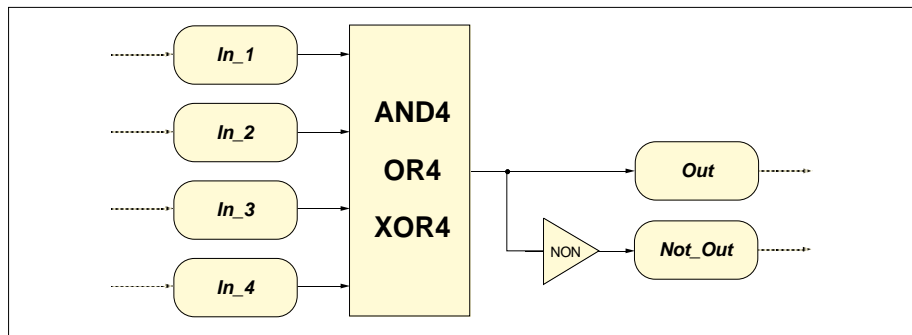
AND4 : BLOC ET**OR4 : BLOC OU****XOR 4 : BLOC OU EXCLUSIF****Fonction du bloc**

Figure 8-3 Schémas des blocs AND4, OR4, XOR 4

Voir le schéma de la figure 8-3, qui s'applique aux trois blocs. Chaque bloc dispose de quatre entrées logiques et effectue la fonction booléenne correspondante pour produire la sortie, qui est disponible à la fois sous forme vraie (*Out*) et complémentaire (*Not_Out*).

Dans le bloc AND4, *Out* est VRAI uniquement si toutes les quatre entrées sont VRAIES.

Dans le bloc OR4, *Out* est VRAI uniquement si au moins une des entrées est VRAIE.

Dans le bloc XOR4, *Out* est VRAI uniquement si une, et uniquement une entrée impaire (1 ou 3) est VRAIE. Dans tous les autres cas, le paramètre *Out* est FAUX.

Paramètres du bloc

Les symboles utilisés dans la table 8-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
In_1	Entrée 1	V/F	
In_2	Entrée 2	V/F	
In_3	Entrée 3	V/F	
In_4	Entrée 4	V/F	
Out	Sortie	V/F	
Not_Out	Sortie complémentaire	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 8-2 Paramètres du bloc AND4/OR4/XOR4

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

In_1 à In_4. Entrées 1 à 4.

Out. Sortie de l’opération booléenne.

Not_Out. Egal au complément logique de *Out*. Forme complémentaire de la sortie du bloc.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

NOT : BLOC NON

Fonction du bloc

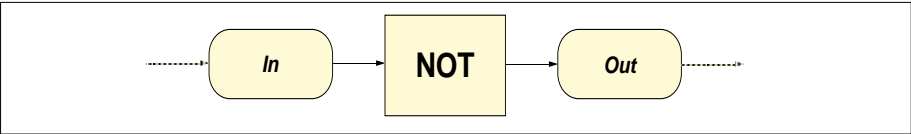


Figure 8-4 Schéma du bloc NOT

Voir la figure 8-4. Le bloc NOT inverse une sortie logique, c’est à dire si *In* est VRAI, *Out* est FAUX ; si *In* est FAUX, *Out* est VRAI.

Paramètres du bloc

Les symboles utilisés dans la table 8-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
In	Entrée	V/F	
Out	Sortie	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 8-3 Paramètres du bloc NOT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

In. Entrées du bloc.

Out. Sortie de l’opération NOT.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

LATCH : BLOC VERROUILLAGE

Fonction du bloc

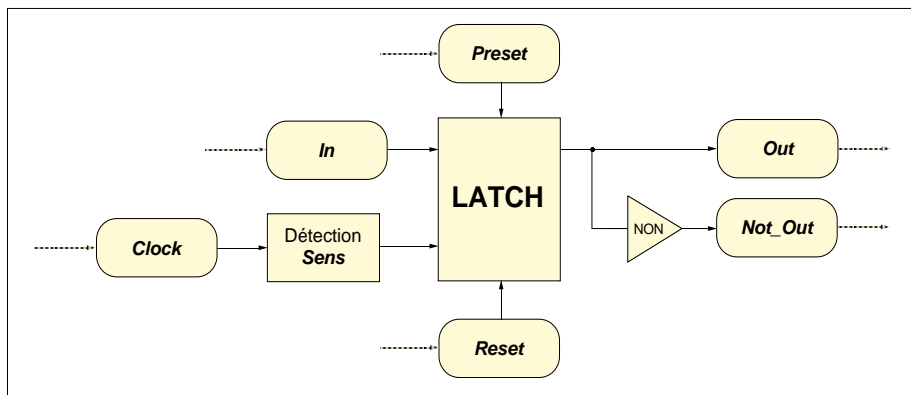


Figure 8-5 Schéma du bloc LATCH

Voir la figure 8-5. Le bloc LATCH dispose d'une fonction à bascule de type D à usage général, avec une entrée d'horloge qui permet de faire fonctionner le verrouillage sur les fronts montants ou descendants (spécifiés par le paramètre *Sense*). L'entrée de l'horloge met *Out* égal à *In*, à moins que *Preset* ou *Reset* (présélection ou RAZ) ne soient actifs. *Preset* force *Out* à passer à VRAI et annule *In*. *Reset* force *Out* à FAUX, et annule à la fois *Preset* et *In*. La table de vérité 8-4 récapitule l'action du bloc Latch :

In	Clock	Preset	Reset	Out	Not_Out
X	X	VRAI	FAUX	VRAI	FAUX
X	X	X	VRAI	FAUX	VRAI
FAUX	↑/↓	FAUX	FAUX	FAUX	VRAI
VRAI	↑/↓	FAUX	FAUX	VRAI	FAUX

LEGENDE : X = état indifférent ↑/↓ = montant ou descendant défini par *Sense*

Table 8-4 Table de vérité du bloc LATCH

Paramètres du bloc

Les symboles utilisés dans la table 8-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
In	Entrée donnée	V/F	
Clock	Entrée horloge	V/F	
Preset	Entrée présélectionnée	V/F	
Reset	Entrée remise à zéro	V/F	
Sense	Sélection du front de déclenchement entrée horloge		Menu
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Out	Sortie	V/F	
Not_Out	Sortie complémentaire	V/F	

Table 8-5 Paramètres du bloc LATCH

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

In. Entrée de données.

Clock (Horloge). Une transition d’entrée d’horloge met le champ *Out* dans le même état que le champ *In*, à moins que *Preset* et *Reset* ne soient VRAIS. L’entrée de l’horloge répond soit aux fronts montant ou descendant - ou aux deux - suivant la spécification du paramètre *Sense*.

Preset (Présélection). Lorsque *Preset* est VRAI, *Out* est forcé à passer à VRAI, et *Clock* et *In* sont annulés. *Preset* est annulé par *Reset*.

Reset (RAZ). Lorsque *Reset* est vrai, *Out* est forcé à passer à FAUX, et toutes les autres entrées sont annulées.

Sense (Sens). (FALLING/RISING - DESCENDANT/MONTANT). Permet de spécifier le front de déclenchement auquel l’horloge doit répondre. DESCENDANT = VRAI à transition FAUX ; MONTANT = FAUX à transition VRAI ; SOIT = aux deux transitions.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

Out. Sortie de la fonction Latch.

Not_Out. Egal au complément logique de *Out*. Forme complémentaire de la sortie du bloc.

COUNT : BLOC DE COMPTAGE

Fonction du bloc

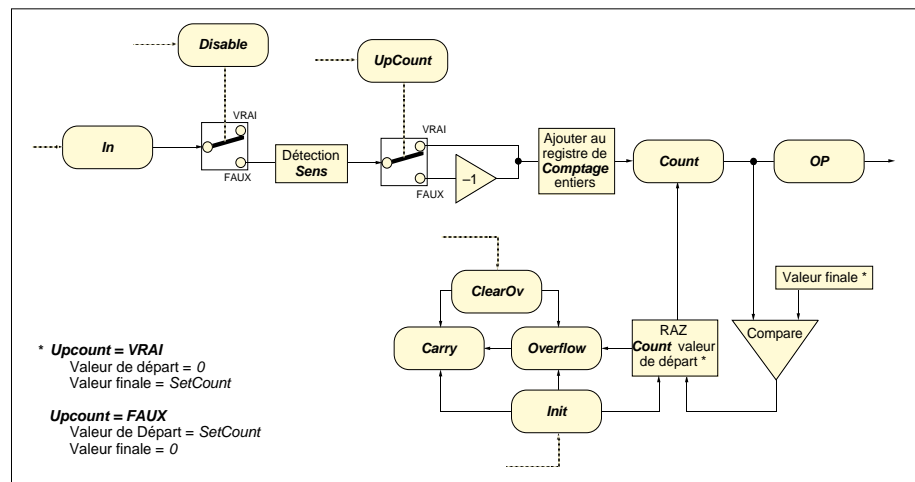


Figure 8-6 Schéma du bloc COUNT

Voir la figure 8-6. Le bloc compte les impulsions d'entrée, soit les fronts montants ou descendants (spécifiés par le paramètre *Sense*). Des totaux de huit chiffres sont maintenus dans le paramètre *Count* comme des entiers, qui peuvent être affichés en utilisant l'objet graphique READOUT (AFFICHAGE) du T1000. Le total de comptage est également disponible comme une liaison de sortie analogique *OP*, avec une précision de 7 chiffres à point flottant. *OP* est mis à l'échelle pour la liaison à l'objet graphique bargraphe.

Le compteur peut être configuré (ou reconfiguré à n'importe quel moment) soit comme un compteur ou un décompteur par l'intermédiaire du paramètre *UpCount* (comptage). Lorsqu'il est configuré comme un compteur, les impulsions d'entrée incrémentent le total dans *Count*, jusqu'à ce qu'il atteigne une "valeur finale" spécifiée par le paramètre *SetCount* (définition du comptage). À l'impulsion suivante, *Count* revient à la "valeur de départ" de zéro, un drapeau de dépassement (*Overflow*) est verrouillé et un drapeau de retenue (*Carry*) est affiché (en général pour un comptage de plus) pour permettre la mise en cascade des compteurs. Lorsqu'il est configuré comme un décompteur, les impulsions d'entrée décrémentent le total de *Count*, jusqu'à ce qu'il atteigne la "valeur finale", qui est cette fois égale à zéro. À l'impulsion suivante, *Count* revient à la "valeur de départ" *SetCount*, et les drapeaux dépassement et retenue sont affichés comme précédemment.

Des entrées d'initialisation, d'invalidation et de remise à zéro des drapeaux de dépassement sont également disponibles.

Paramètres du bloc

Les symboles utilisés dans la table 8-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.






















Paramètre	Fonction	Unités	Etat
In	Entrée des impulsions	V/F	
SetCount	Valeur comptage début/Fin	Entier	
Sense	Sélection front de déclenchement compteur	Menu	
ClearOv	Effacement drapeau dépassement (& retenue)	V/F	 
Disable	Arrêt compteur à la valeur finale	V/F	
UpCount	Sélection comptage/décomptage	V/F	
Init	Initialisation compteur	V/F	 
Alarms			  
Software	Données corrompues/Défaut de communication		
Combined	Alarme combinée du bloc		
OP	Valeur de sortie du compteur (point flottant)	Eng	 
Count	Total comptage (entier)	Entier	 
Carry	Drapeau retenue	V/F	 
Overflow	Drapeau dépassement	V/F	 
HR_OP, LR_OP	Echelle haute & basse objets graphiques de OP	Eng	 

Table 8-6 Paramètres du bloc COUNT

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

In. Entrée logique du compteur.

SetCount (Définition comptage). Permet de définir la “valeur finale” ou la “valeur de départ” du paramètre *Count*, lorsque le bloc est configuré comme un compteur ou un décompteur.

Sense (Sens). (FALLING/RISING - DESCENDANT/MONTANT). Permet de spécifier le front de déclenchement pour lequel l’entrée (*In*) du compteur doit réagir.

DESCENDANT = VRAI à transition FAUX ; MONTANT = FAUX à transition VRAI.

ClearOv (Effacement dépassement). Une entrée VRAIE remet à zéro les drapeaux dépassement et retenue ; ensuite, *ClearOv* repasse automatiquement à FAUX.

Disable (Invalidation). Disable étant VRAI, le compteur s’arrête - mais uniquement après que le total de *Count* a atteint sa “valeur finale” et a été remis à sa “valeur de départ” (soit par *SetCount* ou zéro), et que les drapeaux retenue et dépassement ont été définis. Alors, les impulsions d’entrée ne sont pas ajoutées à *Count*, et les drapeaux sont “figés”, jusqu’à ce que *Disable* repasse à FAUX.

UpCount (Comptage). VRAI permet de sélectionner une fonction de compteur : les impulsions incrémentent le total de comptage qui va de zéro à *SetCount*. FAUX permet de sélectionner un *décompteur* : les impulsions décrémentent le total qui se déroule dans le sens inverse. Il faut noter que la valeur de *Count* n'est pas ré-initialisée lorsque *UpCount* est modifié, de sorte que la fonction compteur peut être modifiée à mi-comptage, avant que la valeur finale ne soit atteinte.

Init (Initialisation). Une valeur VRAIE initialise le compteur, c'est à dire remet à zéro *Carry*, *Overflow* et *Count*, et alors, *Init* repasse à FAUX.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

OP. Sortie du compteur (format à 7 chiffres à point flottant). Permet la liaison à des blocs dans un schéma de boucles et aux bargraphes dans les affichages graphiques. La précision de *OP* est limitée à 7 chiffres comparés aux 8 chiffres du paramètre *Count* entier.

Count (Total). Sortie du compteur (format entier). Peut être lié aux affichages graphiques par l'intermédiaire de l'objet graphique READOUT (AFFICHAGE). La valeur affichée a une résolution entiers de 8 chiffres.

Carry (Retenue). Passe à VRAI, lorsque le total dans *Count* a atteint la "valeur finale" (définie par *UpCount* et *SetCount*), et a ensuite été remise à sa "valeur de départ". Lorsque *Count* s'éloigne de sa valeur de départ (après une impulsion d'entrée de plus), *Carry* automatiquement repasse à FAUX. L'initialisation du bloc par *Init* remet également *Carry* à zéro.

Overflow (Dépassement). Se verrouille sur VRAI, lorsque le compteur a atteint sa "valeur finale" et a été remis à sa "valeur de départ". Le drapeau est remis à zéro, lorsque *Init* est VRAI ou lorsque *ClearOv* est VRAI.

HR_OP, LR_OP. Echelle haute & basse pour les objets graphiques liés à *OP* (bargraphe, tendance). *HR_OP* et *LR_OP* définissent les affichages 100 % et 0 %.

COMPARE : BLOC COMPARAISON

Fonction du bloc

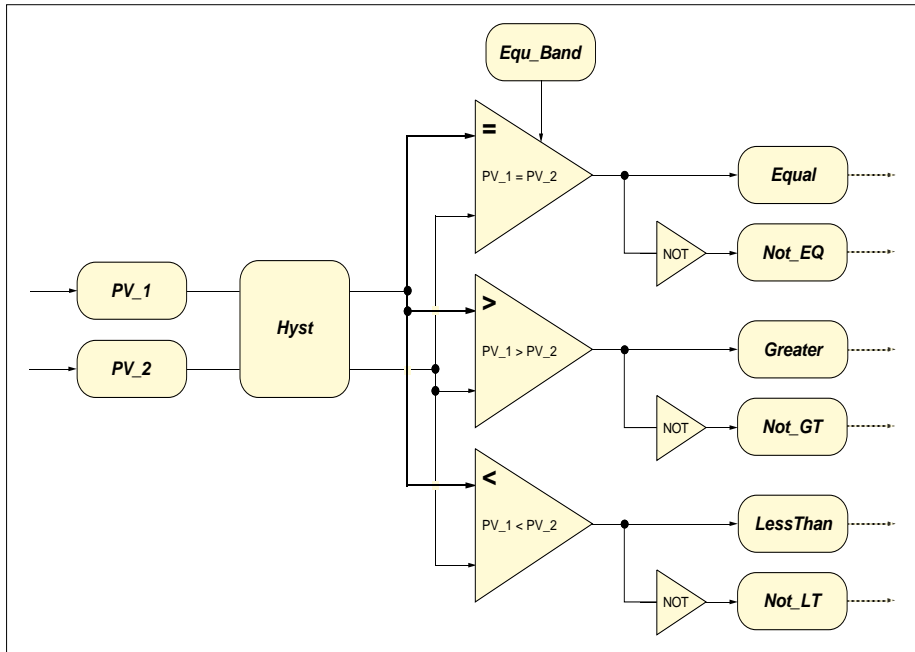


Figure 8-7 Schéma du bloc COMPARE

Voir la figure 8-7. Le bloc COMPARE compare deux entrées analogiques qui réalisent trois fonctions logiques : Egal à, Plus Grand que et Plus Petit que (Equal To, Greater Than et Less Than). Les trois paires de sorties indiquent les résultats logiques des comparaisons sous forme vraie et complémentaire.

Hystérésis & Bandes égal. Une bande d'hystérésis (*Hyst*) définie par l'utilisateur agit sur chaque limite, et une "bande égal" (*Equ_Band*) agit en plus dans la fonction Egal à. Ces paramètres font fonction de "bandes de tolérance" utilisées dans les comparaisons de deux entrées. Les figures 8-8 à 8-10 montrent la procédure d'application de *Hyst* et *Equ_Band*.

NOTA. Dans les figures, les courbes doivent être suivies de gauche à droite le long de l'axe horizontal du temps. Elles ne sont pas valables dans le sens inverse.

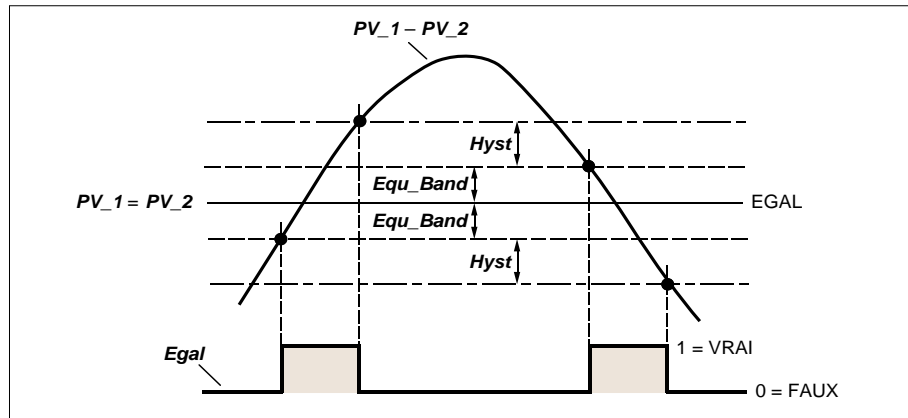


Figure 8-8 Bloc COMPARE - Action "Egal"

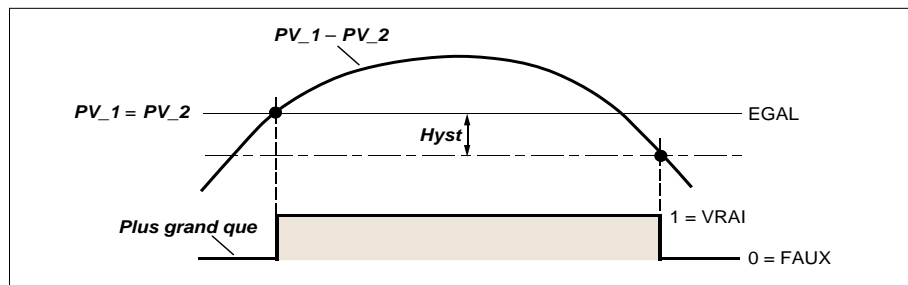


Figure 8-9 Bloc COMPARE - Action "Plus grand que"

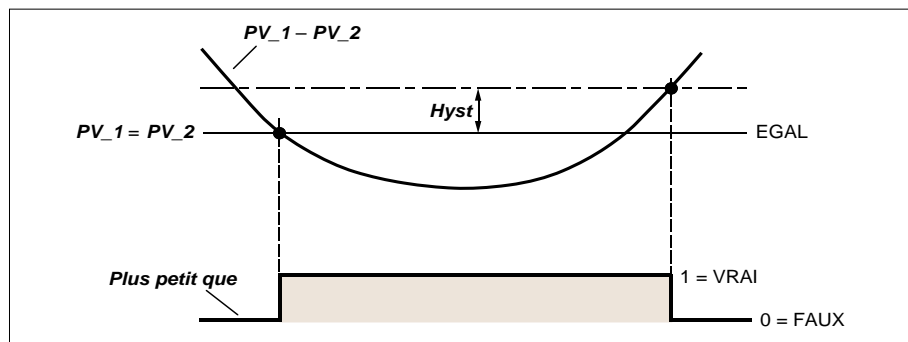


Figure 8-10 Bloc COMPARE - Action "Plus petit que"

Paramètres du bloc

Les symboles utilisés dans la table 8-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV_1	Entrée 1	Eng	
PV_2	Entrée 2	Eng	
Hyst	Bande d'hystérésis	Eng	
Equ_Band	Bande égal	Eng	
HR, LR	Echelle haute & basse objets graphiques de PV_1,PV_2	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Equal	$PV_1 = PV_2$	V/F	
Not_EQ	$PV_1 \neq PV_2$	V/F	
Greater	$PV_1 > PV_2$	V/F	
Not_GT	$PV_1 \leq PV_2$	V/F	
LessThan	$PV_1 < PV_2$	V/F	
Not_LT	$PV_1 \geq PV_2$	V/F	

Table 8-7 Paramètres du bloc COMPARE

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV_1, PV_2. Entrées 1 et 2 respectivement.

Hyst. Bande d'hystérésis. Les figures 8-8 à 8-10 montrent l'application de *Hyst*.

Equ_Band. Bande égale. Permet de spécifier une bande de “tolérance” symétrique où *PV_1* et *PV_2* sont définis comme étant égaux. Voir figure 8-8.

HR, LR. Echelle haute & basse pour les objets graphiques liés à *PV_1* et *PV_2* (bargraphe, tendance). *HR_OP* et *LR_OP* définissent les affichages 100 % et 0 %.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Equal (Egal). VRAI indique que *PV_1* et *PV_2* se trouvent dans la bande Equal, soumis à l'hystérésis.

Not_EQ. NON “Egal” logique. Forme de sortie complémentaire de Equal.

Greater (Plus grand que). VRAI indique que $PV_1 > PV_2$ soumis à l'hystérésis.

Not_GT. NON “plus grand que” logique. Forme de sortie complémentaire de “plus grand que”.

LessThan (Plus petit que). VRAI indique que $PV_1 < PV_2$ soumis à l'hystérésis.

Not_LT. NON “plus petit que” logique. Forme de sortie complémentaire de “plus petit que”.

[Page laissée intentionnellement blanche]

Chapitre 9 BLOCS DE FONCTION MATHS

ADD2 : BLOC ADDITION

SUB2 : BLOC SOUSTRACTION

MUL2 : BLOC MULTIPLICATION

DIV2 : BLOC DIVISION

Fonction du bloc

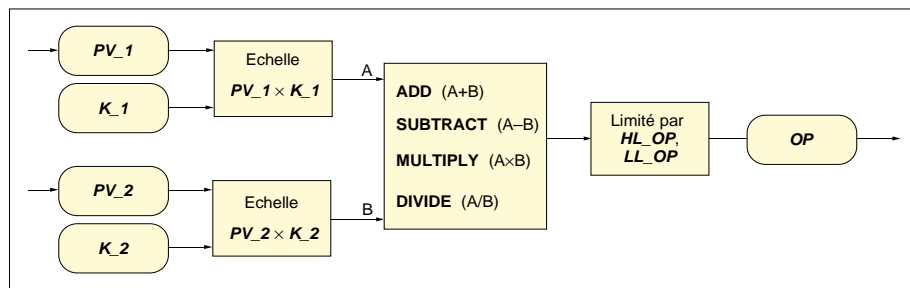


Figure 9-1 Schémas du bloc ADD2, SUB2, MUL2, DIV2

Voir le schéma de la figure 9-1, qui s'applique aux quatre blocs. Les blocs Maths permettent d'effectuer des opérations arithmétiques de base sur deux entrées analogiques mises à l'échelle. La sortie *OP* qui en résulte est soumise aux limites hautes et basses.

Paramètres du bloc

Les paramètres sont les mêmes pour les quatre blocs Maths et sont indiqués dans la table 9-1. Les symboles utilisés dans la table sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PV_1	Entrée 1	Eng1	▶◻▶
K_1	Constante de mise à l'échelle pour PV_1		
PV_2	Entrée 2	Eng2	▶◻▶
K_2	Constante de mise à l'échelle pour PV_2		
OP	Sortie	Eng3	◻▶
HL_OP, LL_OP	Limites hautes & basses sortie	Eng3	◻▶ ◻▶ ?
Alarms			◻▶ ◻▶ ◻▶
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 9-1 Paramètres du bloc Maths

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

PV_1, PV_2. Valeurs de l’entrée 1 et 2. Peuvent être positives ou négatives.

K_1, K_2. Constantes de mise à l’échelle pour les entrées 1 et 2. Peuvent être positives ou négatives.

OP. Valeur de sortie des calculs.

NOTA. Dans le bloc DIV2, 0/0 est calculé comme 0.

HL_OP, LL_OP. Limites hautes & basses de sortie appliquées à *OP*.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

EXPR : BLOC EXPRESSION

Fonction du bloc

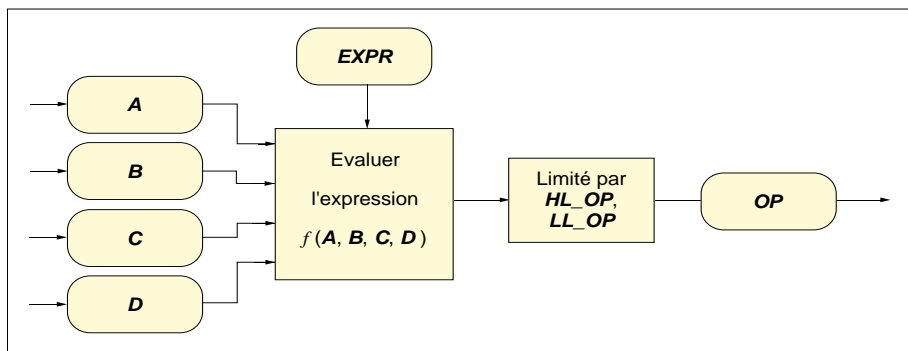


Figure 9-2 Schéma du bloc EXPR

Voir la figure 9-2. Le bloc Expression permet d'entrer (par exemple, par l'intermédiaire du clavier alphanumérique à l'écran du T1000) une expression ayant un maximum de 78 caractères pour manipuler un maximum de quatre variables d'entrée A , B , C et D . Le résultat est soumis aux limites hautes et basses avant d'être fourni en sortie comme OP . La table 9-2 donne la liste des opérateurs mathématiques qui peuvent être utilisés dans l'expression.

NOTA. Le T100 ne peut éditer que des expressions de 24 caractères de long, mais peut évaluer correctement des expressions plus longues téléchargées à partir d'un T1000. (La fenêtre du clavier écran du T1000 défile pour recevoir des entrées longues.)

Ordre d'évaluation

L'ordre de priorité mathématique est de gauche à droite, et ensuite comme ci-après:

- () parenthèses — *priorité la plus haute*
- – (négation), NOT (inversion logique), fonction standard (par ex. TRUNC, FLOAT...)
- ** (exponentiation)
- * (multiplier), / (diviser), MOD (modulo)
- + (ajouter), – (soustraire)
- =, <, >, <=, >=, <> (comparateurs)
- AND (ET logique)
- XOR (OU exclusif)
- OR (OU logique) — *priorité la plus basse*

Par exemple, $A * B + C / D$ est traité comme $(A * B) + (C / D)$, mais $A + B * C - D$ comme $A + (B * C) - D$.

Opérateur	But	Format
()	Parenthèses (modif. précedence)	$(A + B) * (C + D)$
-	Négation	-A
NOT	Inversion logique	NOT(A)
TRUNC	Conversion en entiers	TRUNC(A)
FLOAT	Conversion point flottant	FLOAT(A)
ROUND	Arrondir (vers haut ou bas)	ROUND(A)
ABS	Absolu	ABS(A)
DEG	Degrés (à partir de radians)	DEG(A)
RAD	Radians (à partir de degrés)	RAD(A)
SIN	Sinus (Radians)	SIN(A)
COS	Cosinus (Radians)	COS(A)
TAN	Tangente (Radians)	TAN(A)
ASIN	Arcsinus (Radians)	ASIN(A)
ACOS	Arccosinus (Radians)	ACOS(A)
ATAN2	Arctangente (Radians)	ATAN2(A,B) [où tangente = A/B]
SQRT	Racine carrée	SQRT(A)
LN	Logarithme népérien (base e)	LN(A)
LOG	Logarithme (base 10)	LOG(A)
EXP	Exponentiation (e^A)	EXP(A)
MIN	Valeur minimale	MIN(A,B,C,...)
MAX	Valeur maximale	MAX(A,B,C,...)
AVG	Moyenne (moyenne arithmétique)	AVG(A,B,C,...)
RANDOM	Valeur aléatoire	RANDOM(A) [A=modulus maxi.]
SWITCH	Sélection A (entier C < 1) sinon B	SWITCH(A,B,C)
**	Exponentiation (Puissance A^B)	A**B
*	Multiplication (x)	A*B
/	Division (÷)	A/B
MOD	Modulo (reste de A/B)	A MOD B
+	Addition	A + B
-	Soustraction	A - B
=	Egal	A = B
<	Plus petit que	A < B
>	Plus grand que	A > B
<=	Plus petit que ou égal à	A <= B
>=	Plus grand que ou égal à	A >= B
<>	Pas égal	A <> B
AND	ET logique	A AND B
OR	OU Logique	A OR B
XOR	XOR logique (OU exclusif)	A XOR B

Table 9-2 Opérateurs du bloc EXPR

Formats des nombres

Toutes les entrées et sorties entre le bloc expression et les autres blocs du schéma de boucles doivent être au format point flottant. Si nécessaire, les nombres peuvent être convertis en nombres entiers en utilisant TRUNC, après évaluation de l’expression et ensuite être reconvertis en point flottant en utilisant FLOAT, après évaluation de l’expression.

Fonctions trigonométriques

Le radian est l’unité par défaut, mais les fonctions trigonométriques peuvent être évaluées en degrés en utilisant des expressions telles que :

SIN(RAD(A)) pour sinus (degrés) et DEG(ASIN(A)) pour arcsinus (degrés).

Limites de calculs

Des parenthèses peuvent être imbriquées jusqu’à 10 niveaux, et 18 variables peuvent être incluses dans une expression (répétitions de A, B, C et D). Les puissances $x^{**}y$ sont calculées en utilisant des logarithmes, et donc, les valeurs négatives de x ne peuvent être calculées.

Erreurs de calcul

Si le calcul ne peut être effectué, (par ex. division par zéro), alors l’alarme logiciel est mise à 1 et OP prend la valeur du dernier résultat calculé.

Paramètres du bloc

Les symboles utilisés dans la table 9-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.











Paramètre	Fonction	Unités	Etat
EXPR	Expression	Alphanumérique	
A	Entrée A	EngA	
B	Entrée B	EngB	
C	Entrée C	EngC	
D	Entrée D	EngD	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
OP	Sortie		
HL_OP, LL_OP	Limites hautes & basses sortie		 

Table 9-3 Paramètres du bloc EXPR

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

EXPR. L'expression mathématique alphanumérique évaluée par le bloc. Les expressions peuvent contenir un maximum de 78 caractères - y compris *A*, *B*, *C* et *D*, ainsi que les opérateurs et chiffres. *EXPR* peut être édité dans des blocs locaux au cours de la conduite, mais ne peut être visualisé dans un bloc image. Il faut noter que le champ *EXPR* tel qu'il est affiché dans le menu de spécifications du bloc (avant la saisie) ne montre que les dix premiers caractères de l'expression.

A, B, C, D. Entrées *A*, *B*, *C*, *D*. Les arguments de l'expression définie dans le paramètre *EXPR*.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

OP. Sortie de l'expression, après la limitation par *HL_OP* et *LL_OP*.

HL_OP, LL_OP. Limites hautes & basses de sortie appliquées au résultat de l'expression avant qu'elle ne devienne *OP*.

ACTION : BLOC GESTION D'ACTION

Fonction du bloc

Le bloc ACTION permet d'exécuter une "action" de type séquence, créée séparément avec le logiciel LINtools dans le schéma de boucles. Voir les informations détaillées sur la configuration des actions de texte structuré dans le *Manuel produit LINtools T500* (réf. n° 082 377 U999).

Le bloc permet de disposer de variables à point flottant, d'entiers et de valeurs logiques qui peuvent être utilisées par l'action, et des champs définissant les qualificatifs de l'action et les durées associées des qualificatifs (le cas échéant). Une entrée logique fournit le moyen de valider l'action, et le temps écoulé depuis la validation de l'action est disponible comme sortie.

Paramètres du bloc

Les symboles utilisés dans la table 9-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

ActName. Permet de spécifier le nom d'une action contenue dans un fichier appelé *FileName.STO*, où *FileName* et son chemin sont les mêmes que le fichier associé de la base de données (.DBF). *FileName* (8 caractères maximum) est défini dans le paramètre *FileName* du bloc.

Les actions sont sauvegardées dans un format compilé et traduit dans le fichier .STO.

FileName (Nom de fichier). Permet de spécifier le nom racine (avec l'extension .STO) du fichier contenant l'action spécifiée par le paramètre *ActName*. *FileName* et son chemin sont les mêmes que pour le fichier de base de données associé (.DBF).

A0 à A7. Huit variables à point flottant qui peuvent être intégrées dans le texte structuré qui définit l'action. Ces variables peuvent être entrées ou sorties par l'intermédiaire du schéma de boucles.

D. Champ binaire contenant 16 variables logiques, *Bit0* à *Bit15* qui peuvent être intégrées dans le texte structuré qui définit l'action. Ces variables peuvent être entrées ou sorties par l'intermédiaire du schéma de boucles.

T. Temps écoulé depuis que le paramètre de validation de l'action (*EN*) est passé à VRAI dans les unités spécifiées par le paramètre *TimeBase*. L'horloge qui contrôle *T* s'arrête lorsque *EN* passe à FAUX, et repart de zéro lorsque *EN* repasse à VRAI. Le bloc utilise *T* pour calculer la durée de début et d'arrêt des actions qualifiées par *L*, *D* et *E* en comparant sa valeur active à *QualTime*.

Paramètre	Fonction	Unités	Etat
ActName	Nom de l'action à 8 caractères (maxi)	Alphanumérique	
FileName	Nom du fichier à 8 caractères (maxi)	Alphanumérique	
A0 à A7	Variables à point flottant	Eng	
D	Variable logique	ABCD Hex	
Bit0	Valeur logique 0	V/F —1	D
Bit1	Valeur logique 1	V/F —2	
Bit2	Valeur logique 2	V/F —4	
Bit3	Valeur logique 3	V/F —8	
Bit4	Valeur logique 4	V/F —1	C
Bit5	Valeur logique 5	V/F —2	
Bit6	Valeur logique 6	V/F —4	
Bit7	Valeur logique 7	V/F —8	
Bit8	Valeur logique 8	V/F —1	B
Bit9	Valeur logique 9	V/F —2	
Bit10	Valeur logique 10	V/F —4	
Bit11	Valeur logique 11	V/F —8	
Bit12	Valeur logique 12	V/F —1	A
Bit13	Valeur logique 13	V/F —2	
Bit14	Valeur logique 14	V/F —4	
Bit15	Valeur logique 15	V/F —8	
T	Temps écoulé depuis paramètre EN VRAI		
EN	Entrée validation action	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
NoAction	Impossible trouver action désignée	V/F	
BadActn	Erreur évaluation en mode conduite	V/F	
Combined	Alarme combinée du bloc	V/F	
I0 à I7	Variables entières à 32 bits	Entier	
Qual	Qualificateur d'action	Menu	
QualTime	Durée du qualificateur		
TimeBase	Unités de temps pour T et QualTime	Menu	

Table 9-4 Paramètres du bloc ACTION

EN. Entrée de validation de l'action. *EN VRAI* remet à zéro et déclenche l'horloge *T*, *FAUX* arrête l'horloge *T*, qui maintient sa valeur jusqu'à ce qu'elle soit remise en route. L'état d'*EN* contrôle l'exécution de l'action, voir figure 9-3.

Noter qu'une transition FAUX à VRAI sur EN est équivalente à l'activation d'un pas dans une séquence, et une transition VRAI à FAUX est équivalente à la désactivation d'un pas.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **NoAction.** Activé si le bloc ne peut trouver l'action désignée dans ActName.
- **BadActn.** Activé si une erreur d'évaluation se produit au cours de l'exécution de l'action.

■ **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

I0 à I7. Ensemble de huit variables entières à 32 bits qui peuvent être intégrées dans le texte structuré qui définit l'action. *I0* à *I7* peuvent constituer des entrées ou des sorties par l'intermédiaire du schéma de boucles.

Qual. (P(Initial)/N(Normal)/L(Limited)/D(Delayed)/E(Event)/F(Final) - P(Initial)/N(Normal)/L(Limité)/D(Retardé)/E(Evénement)/F(Final)) Qualificateur de l'action. *Qual* permet de spécifier à quel moment l'action doit être exécutée au cours de la période pendant laquelle l'entrée de validation *EN* est VRAIE, voir figure 9-3.

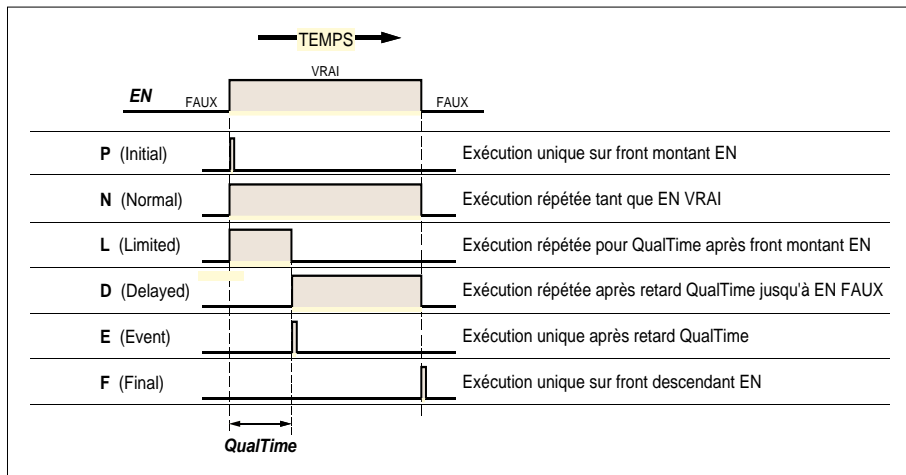


Figure 9-3 Fonctionnement des qualificateurs d'action - Paramètre Qual

QualTime. Valeur de temps associée aux qualificateurs d'action *L*, *D* et *E*, les unités étant spécifiées par *TimeBase*. *QualTime* permet de spécifier une durée d'exécution ou un retard pour l'action et agit au cours de la période où le paramètre *EN* est VRAI, voir figure 9-3.

TimeBase (Base de temps). (sec./min./heures). Permet de spécifier les unités de temps pour les paramètres *T* et *QualTime*.

DIGACT: BLOC D'ACTION LOGIQUE

Fonction du bloc

Le bloc DIGACT permet d'exécuter dans le schéma de boucles une "action" définie par l'utilisateur, créée séparément dans le configurateur d'actions LINtools. Voir les détails sur la configuration des actions à texte structuré dans le *Manuel Produit LINtools T500* (réf. HA 082 377 U999).

NOTA. Le modèle du bloc DIGACT se trouve dans un sous-répertoire appelé DEVICES — accessible par *SelDir* dans le menu de la catégorie MATHS de LINtools.

Le bloc dispose d'un ensemble de variables logiques, individuelles et groupées (en octets et mots de 16 bits), qui peuvent être utilisées par l'action et les champs spécifiant le qualificateur de l'action et la durée associée au qualificateur (le cas échéant). Une entrée logique permet d'activer l'action, et le temps écoulé depuis l'activation de l'action est disponible en sortie.

Notez que le bloc DIGACT est très semblable au bloc ACTION décrit dans les pages précédentes, et plusieurs de ses paramètres sont identiques.

Paramètres du bloc

Les symboles utilisés dans la table 9-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

ActName. Définit le nom d'une action contenu dans un fichier appelé *FileName.STO*; *FileName* est défini ci-dessous.

Les actions sont enregistrées sous un format compilé et résolu dans le fichier .STO.

FileName. Définit le nom racine (avec l'extension .STO) du fichier qui contient l'action spécifiée par le paramètre *ActName*. *FileName* peut être tout nom de fichier valable. Notez que le fichier .STO doit se trouver sur le lecteur E: de l'instrument.

T. Temps écoulé depuis que le paramètre d'activation de l'action (*EN*) est passé à l'état VRAI, en unités spécifiées par le paramètre *TimeBase*. L'horloge contrôlant *T* s'arrête lorsque *EN* passe à l'état FAUX, et reprend à zéro, lorsque *EN* repasse à l'état VRAI. Le bloc utilise *T* pour mesurer dans le temps le début et la fin des actions qualifiées *L*, *D* et *E* en comparant sa valeur active à *QualTime*.

EN. Entrée d'activation de l'action. Si *EN* passe à l'état VRAI, l'horloge *T* est mise à zéro et commence à décompter le temps, si *EN* passe à l'état FAUX, l'horloge *T* est arrêtée et sa valeur maintenue jusqu'à ce qu'elle soit relancée. L'état de *EN* contrôle l'exécution de l'action, voir figure 9-3 à la page précédente.

Notez qu'une transition de FAUX à VRAI sur EN est parallèle à l'activation d'une étape dans une séquence, et une transition VRAI à FAUX équivaut à la désactivation de l'étape.

Paramètre	Fonction	Unités	Etat
ActName	Nom de l'action 8 caractères maxi.	Alphanumérique	
FileName	Nom du fichier 8 caractères maxi.	Alphanumérique	
T	Temps écoulé depuis que le paramètre EN VRAI		
EN	Entrée d'activation de l'action	V/F	
Alarms			
Software	Anomalie données RAM du bloc/défaut réseau	V/F	
NoAction	Action désignée introuvable	V/F	
BadActn	Erreur d'évaluation en conduite	V/F	
Combined	Alarme combinée du bloc	V/F	
Qual	Qualificateur de l'action	Menu	
QualTime	Durée du qualificateur		
TimeBase	Unités de temps pour T et QualTime	Menu	
Bool_A à Bool_X	Variables logiques	V/F	
Word0 à Word3	Variables logiques à 16 bits	ABCD hex	
Bit0	Logique 0	V/F — 1	D
Bit1	Logique 1	V/F — 2	
Bit2	Logique 2	V/F — 4	
Bit3	Logique 3	V/F — 8	
Bit4	Logique 4	V/F — 1	C
Bit5	Logique 5	V/F — 2	
Bit6	Logique 6	V/F — 4	
Bit7	Logique 7	V/F — 8	
Bit8	Logique 8	V/F — 1	B
Bit9	Logique 9	V/F — 2	
Bit10	Logique 10	V/F — 4	
Bit11	Logique 11	V/F — 8	
Bit12	Logique 12	V/F — 1	A
Bit13	Logique 13	V/F — 2	
Bit14	Logique 14	V/F — 4	
Bit15	Logique 15	V/F — 8	
Byte0 à Byte3	Variables logiques à 8 bits	CD hex	
Bit0	Logique 0	V/F — 1	D
Bit1	Logique 1	V/F — 2	
Bit2	Logique 2	V/F — 4	
Bit3	Logique 3	V/F — 8	
Bit4	Logique 4	V/F — 1	C
Bit5	Logique 5	V/F — 2	
Bit6	Logique 6	V/F — 4	
Bit7	Logique 7	V/F — 8	

Table 9-5 Paramètres du bloc DIGACT

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **Software.** Anomalie dans les données RAM du bloc.
- **NoAction.** Activé si le bloc ne peut trouver l'action désignée dans *ActName*.
- **BadActn.** Activé s'il se produit une erreur d'évaluation au cours de l'exécution de l'action.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Qual. (P(Initial)/N(Normal)/L(Limited)/D(Delayed)/E(Event)/F(Final - P(Initial)/N(Normal)/L(Limité)/D(Retardé)/E(Evénement)/F(Final)) Qualificateur de l'action. *Qual* permet de spécifier à quel moment l'action doit être exécutée au cours de la période pendant laquelle l'entrée d'activation *EN* est VRAIE, voir figure 9-3.

QualTime. Valeur de temps associée aux qualificateurs d'action *L*, *D* et *E*, les unités étant spécifiées par *TimeBase*. *QualTime* permet de spécifier une durée d'exécution ou un retard pour l'action et agit au cours de la période où le paramètre *EN* est VRAI, voir figure 9-3.

TimeBase (Base de temps). (sec./min./heures). Permet de spécifier les unités de temps pour les paramètres *T* et *QualTime*.

Bool_A à Bool_X. 24 variables logiques qui peuvent être intégrées dans le texte structuré qui définit l'action. Ces variables peuvent être des entrées ou sorties par l'intermédiaire du schéma de boucles.

Word0 à Word3. Quatre champs binaires contenant 16 variables logiques, les *Bit0* à *Bit15*, qui peuvent être intégrés dans le texte structuré définissant l'action. Ces variables peuvent être des entrées ou sorties par l'intermédiaire du schéma de boucles.

Byte0 à Byte3. Quatre champs binaires contenant 8 variables logiques, les *Bit0* à *Bit7*, qui peuvent être intégrés dans le texte structuré définissant l'action. Ces variables peuvent être des entrées ou sorties par l'intermédiaire du schéma de boucles.

Chapitre 10 BLOCS DE FONCTION CONFIG

T100 : BLOC DE CONFIGURATION DU T100

Fonction du bloc

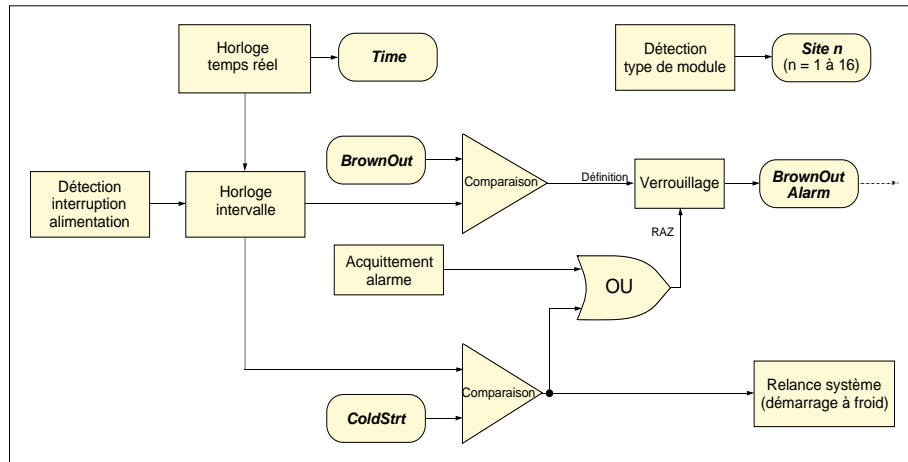


Figure 10-1 Schéma du bloc T100

Voir le schéma de la figure 10-1. Ce bloc est un type de bloc “en-tête” qui doit figurer dans un schéma de boucles pour être exécuté dans un T100. Il permet d’accéder à l’horloge en temps réel (qui est configurée différemment pour chaque unité T100) de l’unité de base, de spécifier un temps de démarrage à froid (relance système) et contient une alarme de coupure d’alimentation qui se verrouille, lorsque le courant est coupé pendant un temps donné. Le bloc indique également le type de module E/S monté dans chacun des seize sites de l’unité de base T100.

Paramètres du bloc

Les symboles utilisés dans la table 10-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Time (Heure). Horloge en temps réel indépendante dans le T100. Ce champ peut être utilisé pour modifier l’heure dans le T100, pendant que le système fonctionne en ligne.

Paramètre	Fonction	Unités	Etat
Time	Horloge temps réel	hh:mm:ss	
IP_type	Fréquence secteur, unités de température	(ABC)D Hex	
60Hz	Fréquence secteur (VRAI= 60Hz, FAUX= 50Hz)	V/F — 1	D
Imperial	Système température (VRAI= Impérial, FAUX= SI)	V/F — 2	
Absolute	Unités température (VRAI= Absolues, FAUX= Relatives)	V/F — 4	
		8	
Site1 à Site16	Nom de repère modules E/S pour les sites T100 1-16	Alphanumérique	
BrownOut	Durée coupure pour déclencher alarme d'alimentation	mins	
ColdStrt	Durée coupure mini. pour déclencher démarrage à froid	mins	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
BrownOut	Panne de courant plus longue que BrownOut	V/F	
Battery	Tension batterie faible	V/F	
PSU	Alimentation de secours en service	V/F	
Combined	Alarme combinée du bloc	V/F	
Status	Etat communications/matériel	ABCD Hex	
Main PSU	Alimentation principale en fonctionnement	V/F — 1	D
CommsAlm	Défaut communication bloc image	V/F — 2	
Batt bad	Défaut batterie	V/F — 4	
Pwr Fail	Exécution démarrage à chaud	V/F — 8	
S13 Cold	Bloc commutateurs 1, Etat Sw 3 (Valid. dém. à froid)	V/F — 1	C
S14 Warm	Bloc commutateurs 1, Etat Sw 4 (Valid. dém. à chaud)	V/F — 2	
S15 LEDs	Bloc commutateurs 1, Etat Sw 5 (fonction LED)	V/F — 4	
S16 CONF	Bloc commutateurs 1, Etat Sw 6 (mode config.)	V/F — 8	
S21	Bloc commutateurs 2, Etat commutateurs 1 à 8	V/F — 1	B
S22		V/F — 2	
S23		V/F — 4	
S24		V/F — 8	
S25	(Adresse LIN = Chiffres AB)	V/F — 1	A
S26		V/F — 2	
S27		V/F — 4	
S28		V/F — 8	

Table 10-1 Paramètres du bloc T100

IP_type. Champ binaire permettant de spécifier la fréquence du secteur et le système/unités de température. Les spécifications s'appliquent à un instrument de la série T100 dans son intégralité et affectent les blocs TCOUPLE, FULL_TC8, RTD et UCHAR, les tables de linéarisation interne et la compensation de soudure froide. Les unités ne peuvent être mélangées dans un instrument unique, mais différentes unités sur le LIN peuvent être configurées individuellement.

- **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz. Il s'agit de définir la fréquence pour les circuits de suppression du bruit dans les modules E/S du T100.

- **Imperial (Impérial).** VRAI permet de sélectionner le système de température impérial - Fahrenheit (°F)/Rankine (R) - FAUX le système SI - Celsius (°C)/Kelvin (K).
- **Absolute.** VRAI permet de sélectionner les unités de température absolues - Kelvin/Rankine - FAUX les unités relatives - Celsius/Fahrenheit .

Site1 à Site16. Chaque champ montre le nom de repère (pas le type) du bloc correspondant au module d'E/S monté à ce numéro de site. Un blanc indique pas de module ou un module non configuré sur le site.

BrownOut (Coupure d'alimentation). Permet de spécifier la durée en minutes d'une coupure de courant nécessaire pour déclencher l'alarme de coupure d'alimentation.

ColdStrt (Démarrage à froid). Permet de spécifier la durée minimale (en minutes) qui provoquera un démarrage à froid du T100. Un "démarrage à chaud" se produit après des coupures de courant plus courtes que *ColdStrt* ; c'est à dire que le schéma de boucles actif est relancé avec la base de données des paramètres existants et les modes de fonctionnement actifs. Toutefois, après un démarrage à froid, une ré-initialisation complète de la base de données des paramètres est effectuée, et le schéma indiqué dans le fichier E:MICRO_D.DBF est chargé et exécuté (si possible).

Lorsqu'un fichier de base de données *dbase.DBF* est téléchargé depuis la page déportée du T1000 ou chargé en utilisant le terminal local du T100, un fichier appelé *dbase.RUN* est créé, et tous les autres fichiers dont l'extension est .RUN sont effacés. Le fichier correspondant de la base de données, si trouvé, est chargé et exécuté. S'il n'y a pas de fichier .RUN ou si pour une raison ou autre il y en a plus d'un, le T100 tente de charger E:MICRO_D.DBF. En cas d'échec, le T100 ne lance pas sa conduite.

Software Version Number (Numéro de version du logiciel). Le numéro de version du logiciel du T100 est affiché à droite du champ alarmes, lorsque le T100 est connecté par l'intermédiaire du LIN.

A la suite du numéro de version du logiciel, il y a entre parenthèses la lettre de l'option de l'instrument: (C) pour l'option régulation uniquement (pas de logiciel séquence installé) et (S) pour l'option séquence (logiciel séquence installé). Les blocs SFC_CON, SFC_MON et SFC_DISP peuvent être exécutés (c'est à dire mis à jour) dans un instrument avec l'option S, bien qu'ils puissent être placés et configurés dans l'option (C) pour être utilisés comme blocs *image* à partir d'autres instruments (S).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **BrownOut (Coupure d'alimentation).** Indique une panne de courant qui dépasse la durée spécifiée dans le champ *BrownOut*. Cette alarme est affichée comme une alarme "événement" non-acquittée, mais ne dispose pas d'un état d'acquiescement.
- **Battery (Batterie).** Indique que la tension de la batterie est faible (priorité d'alarme par défaut = 10)..
- **PSU (Alimentation).** Indique que l'alimentation de secours fonctionne.

- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Status (Etat). Champ binaire indiquant l'état des communications et du matériel du T100.

- **Main PSU (Alimentation principale).** Indique le fonctionnement de l'unité d'alimentation principale.
- **CommsAlm (Alm communicat.).** Ce bit est défini si un bloc image du T100 est en alarme logiciel en raison d'un défaut de communication (pas une erreur de total de contrôle). Le signal *CommsAlm* peut donc être utilisé par une unité T1000 de supervision pour contrôler le bon fonctionnement des communications entre T100, même si les blocs affectés eux-mêmes ne sont pas visibles à partir du T1000. Il suffit de mettre en image les blocs de communication du *T100* dans le T1000 pour rendre les bits *CommsAlm* accessibles.
- **Batt bad (Panne batterie).** Indique une panne de batterie.
- **Pwr Fail (Panne de courant).** Indique que le T100 a effectué un "démarrage à chaud".
- **S13 Cold.** Etat du bloc de commutateurs 1 du T100, commutateur 3 - Validation démarrage à froid. VRAI = On, FAUX = Off.
- **S14 Warm.** Etat du bloc de commutateurs 1 du T100, commutateur 4 - Validation démarrage à chaud. VRAI = On, FAUX = Off.
- **S15 LEDs.** Etat du bloc de commutateurs 1 du T100, commutateur 5 - Mode LED. VRAI = On, FAUX = Off.

Le commutateur LED étant sur On, les 16 LED rouges de la face avant de l'unité de base du T100 indiquent le statut des sites des 16 modules correspondants:

LED	Etat du module correspondant
Off	OK, c'est à dire que le module est bon et bien configuré ou site non utilisé
On	Module configuré manquant ou défectueux
Clignotant	Erreur de type de module pour la configuration T100 ou mal étalonné

Le commutateur de LED étant sur Off, le bloc de LED indique quelle est la tâche logicielle active du T100 :

<div><div>1</div><div>3</div><div>5</div><div>7</div><div>9</div><div>11</div><div>13</div><div>15</div></div>	1 Fond	9 Serveur réseau
<div><div>2</div><div>4</div><div>6</div><div>8</div><div>10</div><div>12</div><div>14</div><div>16</div></div>	2 Principal/config	10 Réseau niveau inférieur
	3 Chargement réseau	11 Réseau RX inférieur
	4 Contrôle des données de la base de données	12 Serveur fichiers
	5 Serveur 3 (blocs image)	13 <i>Non utilisé</i>
	6 Serveur 2 (blocs locaux)	14 <i>Non utilisé</i>
	7 Serveur 1 (matériel)	15 Chargement séquence
	8 <i>Non utilisé</i>	16 <i>Non utilisé</i>

- **S16 CONF.** Etat du bloc de commutateurs 1, commutateur 6 - Mode Configurateur.
VRAI = On, FAUX = Off.

Le commutateur CONF étant sur On, 80 % du temps UC du T100 est consacré à la mise à jour des blocs et le reste est disponible pour la tâche du configurateur si elle est exécutée. Sinon, ces 20 % sont gaspillés. Donc, si CONF est sur On et bien que le schéma de boucles ne réponde peut-être pas à pleine vitesse, le configurateur peut être utilisé aussi souvent que nécessaire sans affecter les performances.

Si le commutateur CONF est sur Off, et que le configurateur n'est pas utilisé, l'UC passe environ 95 % du temps à mettre à jour les blocs. Mais, chaque fois que le configurateur est utilisé, le temps tombe à 50 %, 50 % étant alloués au configurateur. Donc, si CONF est sur Off, et bien que le contrôle soit normalement à l'efficacité maximale, il peut être sérieusement ralenti chaque fois que le configurateur est utilisé.

- **S21 à 28.** Etat du bloc de commutateurs 2, commutateurs 1 à 8 - Adresse LIN.
Pour chaque commutateur, VRAI = On, FAUX = Off.

T1000 : BLOC DE CONFIGURATION DU T1000

Fonction du bloc

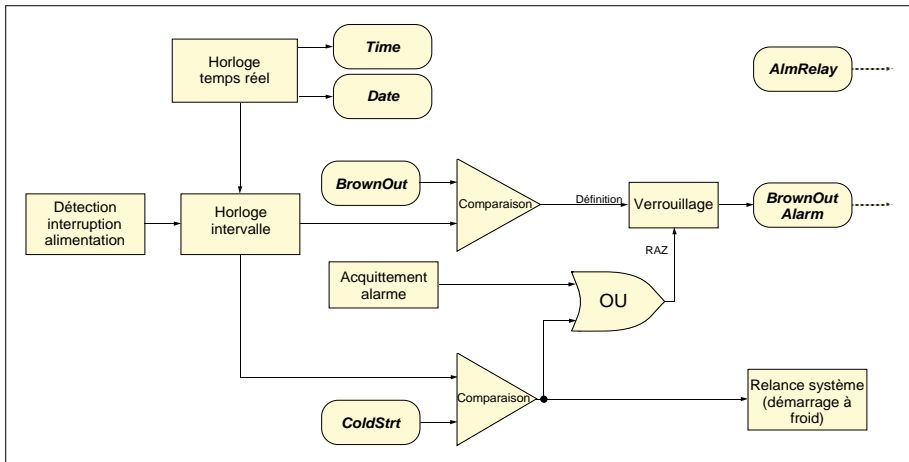


Figure 10-2 Schéma du bloc T1000

Voir le schéma de la figure 10-2. Ce bloc est un type de bloc “en-tête” qui doit figurer dans un schéma de boucles pour être exécuté dans un T1000. Il permet d’accéder à l’horloge en temps réel et au calendrier de l’unité (qui sont configurés différemment pour chaque T1000 dans un système), de spécifier un temps de démarrage à froid (relance système) et contient une alarme de coupure d’alimentation qui se verrouille, lorsque le courant est coupé pendant une durée donnée. Le bloc indique également le statut du relais d’alarme du panneau arrière du T1000, qui est déclenché lorsqu’une alarme non-acquittée de priorité 11 à 15 est présente.

Paramètres du bloc

Les symboles utilisés dans la table 10-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Time, Date (Heure, Date). Horloge en temps réel et calendrier indépendants dans le T1000. Ces champs peuvent être utilisés pour modifier l’heure et la date dans le T1000, pendant que le système fonctionne en ligne, *mais cette opération ne doit pas être effectuée si l’option Historique est en service*.






Paramètre	Fonction	Unités	Etat
Time	Horloge temps réel	hh:mm:ss	
Date	Réinitialisation calendrier	dd:mm:yy	
BrownOut	Durée de coupure pour déclencher l'alarme d'alim.	mins	
ColdStrt	Durée mini. coupure pour déclencher démarrage à froid	mins	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
BrownOut	Panne de courant plus longue que BrownOut	V/F	
Battery	Défaut de la batterie interne	V/F	
Combined	Alarme combinée du bloc	V/F	
AlmRelay	Indication de l'état du relais d'alarme	V/F	 

Table 10-2 Paramètres du bloc T1000

BrownOut (Coupure d'alimentation). Permet de spécifier la durée en minutes d'une coupure de courant nécessaire pour déclencher l'alarme de coupure d'alimentation (*BrownOut*).

ColdStrt (Démarrage à froid). Permet de spécifier la durée minimale (en minutes) qui provoquera un démarrage à froid du T1000. Un "démarrage à chaud" se produit après des coupures de courant plus courtes que *ColdStrt* ; c'est à dire que le schéma de boucles actif est relancé avec la base de données des paramètres existants et les modes de fonctionnement actifs. Toutefois, après un démarrage à froid, une ré-initialisation complète de la base de données des paramètres est effectuée, et le schéma de boucles indiqué dans le fichier CONFIG.SYS est exécuté. S'il n'y a pas de fichier Startup (démarrage), l'exploitation recommence à partir du menu d'ouverture.

Software Version Number (Numéro de version du logiciel). Le numéro de version du logiciel du T1000 est affiché à droite du champ alarmes.

A la suite du numéro de version du logiciel, il y a entre parenthèses la lettre de l'option de l'instrument: (C) pour l'option régulation uniquement (pas de logiciel séquence installé) et (S) pour l'option séquence (logiciel séquence installé). Les blocs SFC_CON, SFC_MON et SFC_DISP peuvent être exécutés (c'est à dire mis à jour) dans un instrument avec l'option S, bien qu'ils puissent être placés et configurés dans l'option (C) pour être utilisés comme blocs *image* à partir d'autres instruments (S).

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **BrownOut (Coupure d'alimentation).** Indique une panne de courant qui dépasse la durée spécifiée dans le champ *BrownOut*. Cette alarme est affichée comme une alarme "événement" non-acquittée, mais ne dispose pas d'un état d'acquiescement.
- **Battery (Batterie).** Indique un défaut de la batterie interne. (Priorité d'alarme par défaut = 10)
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

AlmRelay. Indique l'état du relais alarme qui est déclenché (*AlmRelay* = VRAI), lorsqu'une alarme non-acquittée de niveau de priorité 11 à 15 est présente.

T231 : BLOC DE CONFIGURATION DU T231

Fonction du bloc

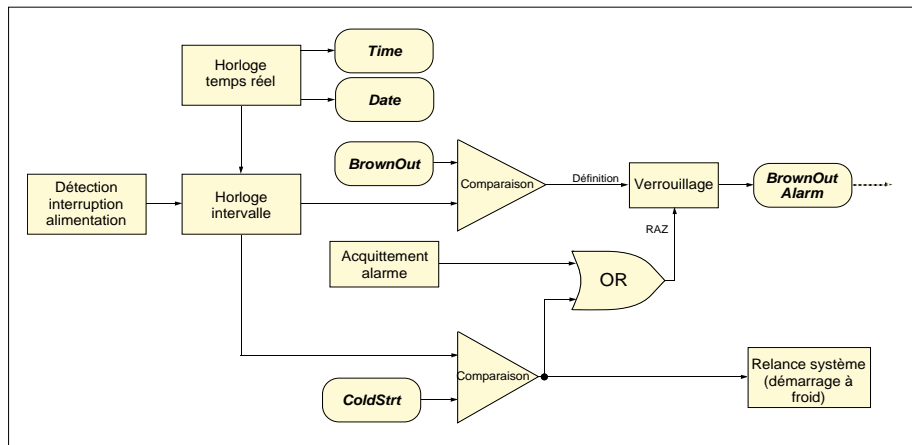


Figure 10-3 Schéma du bloc T231

Voir le schéma de la figure 10-3. Ce bloc est un bloc de type “en-tête” qui doit figurer dans un schéma de boucles pour être exécuté dans un T231. Il permet d’accéder à l’horloge en temps réel de l’unité de base (qui est configurée différemment pour chaque T231), de spécifier des unités de température pour un T231 unique, de définir une durée de démarrage à froid (relance système) et contient une alarme de coupure d’alimentation qui se verrouille, lorsque le courant est coupé pendant une durée donnée.

NOTA. Un bloc T100 peut être exécuté comme bloc en-tête dans un instrument T231 (au lieu d’un bloc T231) pour permettre la compatibilité avec les schémas de boucles existants.

Paramètres du bloc

Les symboles utilisés dans la table 10-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Time, Date (Heure, Date). Horloge et calendrier en temps réel indépendants. Ces champs permettent de modifier l’heure et la date dans le T231, lorsque le système fonctionne en ligne.

Paramètre	Fonction	Unités	Etat
Time, Date	Horloge/calendrier en temps réel	hh:mm:ss/dd:mm:yy	
IP_type	Fréquence secteur, unités de température	(ABC)D Hex	
60Hz	Fréquence secteur (VRAI = 60 Hz, FAUX = 50 Hz)	V/F — 1	D
Imperial	Système Température (VRAI = Impérial, FAUX = SI)	V/F — 2	
Absolute	Unités Température (VRAI = Absolues, FAUX = Relatives)	V/F — 4	
		V/F — 8	
Site1 à Site16	<i>(Non utilisé dans cette version logiciel/matériel)</i>	Alphanumérique	
BrownOut	Durée coupure pour déclencher alarme d'alimentation	mins	
ColdStrt	Durée coupure mini. pour déclencher démarrage à froid	mins	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
BrownOut	Panne de courant plus longue que BrownOut	V/F	
S6000	Alarme commune logiciel communication bloc S6000	V/F	
Battery	Tension batterie faible	V/F	
PSU	Alimentation de secours en marche	V/F	
Combined	Alarme combinée du bloc	V/F	
Status	Etat communications/matériel	ABCD Hex	
Main PSU	Alimentation secteur en marche	V/F — 1	D
CommsAlm	Défaut communication bloc image	V/F — 2	
Batt bad	Défaut batterie	V/F — 4	
Pwr Fail	Démarrage à chaud effectué	V/F — 8	
S13 Cold	Bloc commutateurs 1, Etat SW 3 (Valid. dém. à froid)	V/F — 1	C
S14 Warm	Bloc commutateurs 1, Etat SW 4 (Valid. dém. à chaud)	V/F — 2	
S15 LEDs	Bloc commutateurs 1, Etat SW 5 (Fonction LED)	V/F — 4	
S16 CONF	Bloc commutateurs 1, Etat SW 6 (Mode config.)	V/F — 8	
S21	Bloc commutateurs 2, Etat commutateurs 1 à 18 (Adresse LIN = Chiffres AB)	V/F — 1	B
S22		V/F — 2	
S23		V/F — 4	
S24		V/F — 8	
S25		V/F — 1	A
S26		V/F — 2	
S27		V/F — 4	
S28		V/F — 8	
LEDdrive	Entrée contrôle LED face avant T231 (1/0)	ABCD Hex	
LEDflash	Entrée clignot. LED face avant T231 (1/0)	ABCD Hex	

Table 10-3 Paramètres du bloc T231

IP_type. Champ binaire permettant de spécifier la fréquence du secteur et le système/unités de température. Les spécifications s'appliquent au T231 dans son intégralité et affectent les blocs TCOUPLE, FULL_TC8, RTD et UCHAR, les tables de linéarisation interne et la compensation de soudure froide. Les unités ne peuvent être mélangées dans un T231 unique, mais différents T231 sur le LIN peuvent être configurés individuellement.

- **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz. Il s'agit de définir la fréquence pour les circuits de suppression du bruit dans les modules E/S du T231.
- **Imperial (Impérial).** VRAI permet de sélectionner le système de température impérial - Fahrenheit (°F)/Rankine (R) - FAUX le système SI - Celsius (°C)/Kelvin (K).
- **Absolute (Absolu).** VRAI permet de sélectionner les unités de température absolues - Kelvin/Rankine - FAUX les unités relatives - Celsius/Fahrenheit.

Site1 à Site16. *(Non utilisé dans cette version du logiciel/matériel).*

BrownOut (Coupure d'alimentation). Permet de spécifier la durée (minutes) requise d'une interruption de courant pour déclencher l'alarme *BrownOut*.

ColdStrt (Démarrage à froid). Permet de spécifier la durée minimale (minutes) d'une interruption de courant qui provoquera un démarrage à froid du T231. Un "démarrage à chaud" se produit après des coupures de courant plus courtes que *ColdStrt*; c'est à dire que le schéma de boucles actif est relancé avec la base de données des paramètres et les modes de fonctionnement existants. Toutefois, après un démarrage à froid, une ré-initialisation complète de la base de données des paramètres est effectuée et le schéma de boucles spécifié dans le fichier avec l'extension .RUN est chargé et exécuté (si possible).

Lorsqu'un fichier de base de données *dbase.DBF* est téléchargé depuis la page déportée du T1000 ou chargé en utilisant le terminal local du T231, un fichier appelé *dbase.RUN* est créé, et tous les autres fichiers dont l'extension est .RUN sont effacés. Le fichier correspondant de la base de données, si trouvé, est chargé et exécuté. S'il n'y a pas de fichier .RUN ou si pour une raison ou autre il y en a plus d'un, le T231 tente de charger E:MICRO_D.DBF. En cas d'échec, le T231 ne lance pas sa conduite.

Software version number (N° de version du logiciel). Le numéro de version du logiciel du T231 est affiché à droite du champ *Alarms* lorsqu'un T231 est connecté par l'intermédiaire du LIN.

À la suite du numéro de version du logiciel, il y a entre parenthèses la lettre de l'option de l'instrument: (C) pour l'option régulation uniquement (pas de logiciel séquence installé) et (S) pour l'option séquence (logiciel séquence installé). Les blocs SFC_CON, SFC_MON et SFC_DISP peuvent être exécutés (c'est à dire mis à jour) dans un instrument avec l'option S, bien qu'ils puissent être placés et configurés dans l'option (C) pour être utilisés comme blocs *image* à partir d'autres instruments (S).

Alarms (Alarmes). Voir la description générale du champ *Alarms* à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **BrownOut (Coupure d'alimentation).** Indique une coupure de courant qui dépasse la durée définie dans le champ *BrownOut*. Cette alarme est affichée comme une alarme "d'événement" non acquittée, mais n'a pas d'état acquitté.
- **S6000.** Indique une alarme logiciel (communication) dans un bloc S6000 relié au T231.

- **Battery (Batterie).** Indique que la tension de la batterie est faible (priorité d'alarme par défaut = 10).
- **PSU (Alimentation).** Indique que l'alimentation de secours fonctionne.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Status (Etat). Champ binaire indiquant l'état des communications et du matériel du T231.

- **Main PSU (Alimentation principale).** (Secours/Principale). Indique que le T231 est alimenté par l'alimentation de secours ou l'alimentation principale.
- **CommsAlm.** Ce bit est mis à 1 si un bloc image dans le T231 est en alarme logiciel en raison d'un défaut de communication (pas d'une anomalie mémoire). Le signal *CommsAlm* peut donc être utilisé par une unité T1000 de supervision pour contrôler le bon fonctionnement des communications inter T231, même lorsque les blocs affectés eux-mêmes ne sont pas visibles depuis le T1000. Il suffit de mettre en image les blocs de communication du T1000 pour rendre les bits *CommsAlm* accessibles.
- **Batt Bad.** (Good/Bad - Bonne/Défectueuse). Indique une panne de batterie.
- **Pwr Fail (Panne de courant).** Indique que le T231 a effectué un démarrage à chaud.
- **S13 Cold.** (Disable/Enable - Validation/Invalidation). Etat du bloc de commutateurs 1 T231, commutateur 3 - Validation démarrage à froid. Validation = On, Invalidation = Off.
- **S14 Warm.** (Disable/Enable - Validation/Invalidation). Etat du bloc de commutateurs 1 T231, commutateur 4 - Validation démarrage à chaud. Validation = On, Invalidation = Off.
- **S15 LEDs.** (Tâches/Sites). Etat du bloc de commutateurs 1 T231, commutateur 5 - Mode LED. Sites = On, Tâches = Off.

Lorsque le commutateur LED est sur On ("Sites" et en présence d'un bloc en-tête T100, les 16 LED rouges de la face avant du T231 indiquent l'état des 16 sites correspondants du module, voir la description dans le chapitre consacré au bloc T100 dans le présent manuel. *(Cette fonction n'est pas disponible dans cette version du logiciel/matériel. Les LED restent donc éteintes dans ce cas).*

Lorsque le commutateur LED est sur On ("Sites" et en présence d'un bloc en-tête T100, les 16 LED rouges de la face avant du T231 sont contrôlées par les paramètres de bloc *LEDdrive* et *LEDflash*.

Lorsque le commutateur LED est sur Off ("Tâches"), le bloc de LED indique, comme suit, quelle tâche logicielle du T231 est active:

1	3	5	7	9	11	13	15	1 Tâche de fond	9 Serveur réseau
								2 Principal/Config	10 Réseau inférieur
								3 Chargement base de données	11 Réseau RX inférieur
								4 Total de contrôle base de données	12 Serveur de fichiers
								5 Serveur 3 (blocs image)	13 <i>Non utilisé</i>
								6 Serveur 2 (blocs locaux)	14 <i>Non utilisé</i>
								7 Serveur 1 (matériel)	15 Chargement séquence
								8 <i>Non utilisé</i>	16 <i>Non utilisé</i>

- **S16 CONF.** (CONFIG/PLANT - CONFIG/INSTALLATIONS). Etat du bloc de commutateurs 1, Commutateur 6 - Fonction port de communication. CONFIG = Off, INSTALLATIONS = On.

Lorsque le commutateur CONF est sur Off ("CONFIG"), le port RS232 à 25 points du T231 peut être connecté à un terminal "non intelligent" pour faciliter la configuration et la mise en service.

Lorsque le commutateur CONF est sur On ("PLANT"), le port RS232 peut être connecté par l'intermédiaire d'une unité D240 à une ligne RS422 pour permettre une communication Bi-Synch avec les instruments S6000.

- **S21 à S28.** Etat du bloc de commutateurs 2, commutateurs 1 à 8 - Adresse LIN.
Pour chaque commutateur, VRAI = On, FAUX = Off.

LEDdrive. Champ binaire à 16 bits - *LED_1* à *LED_16* - contrôlant (par l'intermédiaire des entrées logiques du schéma de boucles) les 16 LED de la face avant du T231. Ne fonctionne que lorsque le commutateur 5 du bloc de commutateurs 1 et sur "On" et lorsqu'un bloc en-tête T231 est exécuté. Un bit haut ("on") allume la LED correspondante (Voir *LEDflash* ci-après).

LEDflash. Champ binaire à 16 bits - *LED_1* à *LED_16* - contrôlant (par l'intermédiaire des entrées logiques du schéma de boucles) les 16 LED de la face avant du T231. Ne fonctionne que lorsque le commutateur 5 du bloc de commutateurs 1 et sur "On" et lorsqu'un bloc en-tête T231 est exécuté. Un bit haut ("on") fait clignoter en continu la LED correspondante, à condition qu'elle soit allumée par son bit *LEDdrive* (Voir *LEDdrive* ci-dessus).

NOTA. Lorsque le commutateur 5 du bloc de commutateurs 1 est sur "Off", les LED montrent quelles tâches du T231 sont actives. Lorsque le commutateur est sur "On" et qu'un bloc en-tête est exécuté dans le T231, *les 16 LED sont éteintes pour les versions actuelles des logiciels et matériels.*

T600 : BLOC DE CONFIGURATION DU T600

Fonction du bloc

Ce bloc est un bloc “en-tête” qui doit figurer dans un schéma de boucles à exécuter dans un instrument de la série 600. Il agit comme un bloc à usage général pour les informations d’état dans l’ensemble de l’instrument. Il permet en particulier l’accès local par l’intermédiaire du LIN aux données horloge/calendrier en temps réel du réseau - nécessaires pour le traitement des alarmes. Le bloc permet également de définir une fréquence de suppression du bruit du secteur, un système de température pour un instrument unique, une durée de démarrage à froid (relance système), et contient une alarme d’interruption d’alimentation qui se verrouille lorsque l’alimentation est interrompue pour une durée donnée. Le type de module affecté à chacune des huit adresses de modules E/S sur le bus série de communication interne est également indiqué par le bloc T600. Voir les détails complets dans le paragraphe *Menu de spécifications du bloc* ci-dessous.

Paramètres du bloc

Les symboles utilisés dans la table 10-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Time	Heure du jour	hh:mm:ss	
Date	Date du jour	dd/mm/yy	
IP_type	Fréquence secteur, système de température	(ABC)D Hex	
60Hz	Fréquence secteur (VRAI = 60 Hz, FAUX = 50 Hz)	V/F — 1	D
Imperial	Système température (VRAI = Impérial, FAUX = SI)	V/F — 2	
		V/F — 4	
		V/F — 8	
Site1 à Site8	Identificateur du type de module	Alphanumérique	
BrownOut	Durée coupure déclencher alarme d'alimentation	mins	
ColdStrt	Durée coupure mini. déclencher démarr. à froid	mins	
Node	Adresse réseau ALIN	(01-FE hex)	
Model	Identificateur modèle T600		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
BrownOut	Panne courant plus longue que BrownOut	V/F	
ComS/W	Erreur logiciel commune bloc	V/F	
UTskEr1 à 4	Erreur tâche utilisateur 1 à 4 (arrêt)	V/F	
MainPSU	Panne alimentation principale	V/F	
StbyPSU	Panne alimentation secours	V/F	
RTCinit	Horloge temps réel non-initialisée	V/F	
Combined	Alarme combinée du bloc	V/F	
UsrAlm	Relais alarme utilisateur en alarme (contacts ouverts)	V/F	
AreaNo	Zone clé de sécurité	(0 - 63)	
TimeOut	Dépassement temps imparti mode inspection BD	secs.	

suite...

... suite

Paramètre	Fonction	Unités	Etat
Options	Options config. face avant & sauvegarde BD	ABCD Hex	
FPdis1	Invalidation accès face avant de la boucle 1	V/F — 1	D
FPdis2	Invalidation accès face avant de la boucle 2	V/F — 2	
FPdis3	Invalidation accès face avant de la boucle 3	V/F — 4	
FPdis4	Invalidation accès face avant de la boucle 4	V/F — 8	
NoKeyPrt	Inspection partielle sans clé	V/F — 1	C
NoKeyFul	Inspection complète sans clé	V/F — 2	
LEDtest	Test LED face avant	V/F — 4	
CommsDis	Invalidation écritures champ	V/F — 8	
FullSave	Sauvegarde complète base de données dans fichier	V/F — 1	B
PartSave	Sauvegarde partielle base de données dans fichier	V/F — 2	
BinSpd1	Sélection vitesse communication binaire	V/F — 4	
BinSpd2	Sélection vitesse communication binaire	V/F — 8	
Protectd	Fichier base de données codé	V/F — 1	A
		V/F — 2	
E2Form1 } E2Form2 }	Commande reformatage E2ROM	V/F — 4	
		V/F — 8	
Status	Etat communications/matériel	ABCD Hex	
PwrFail	Indicateur panne de courant	V/F — 1	D
TmpPFail	Indicateur panne de courant temporaire	V/F — 2	
CommsAlm	Alarme communication bloc commune	V/F — 4	
Alarm	Relais alarme déclenché	V/F — 8	
PrtKey	Clé sécurité partielle active	V/F — 1	C
FulKey	Clé sécurité complète active	V/F — 2	
BoardID1 } BoardID2 }	Indication fréquence horloge UC T600	V/F — 4	
		V/F — 8	
S11	Bloc commutateurs DIP, Etat SW 1 (Modbus/TCS binaire)	V/F — 1	B
S12 Xisb	Bloc commutateurs DIP, Etat SW 2 (ON=ISB externe)	V/F — 2	
S13 Cold	Bloc commutateurs DIP, Etat SW 3 (ON=démarr. à froid)	V/F — 4	
S14 Warm	Bloc commutateurs DIP, Etat SW 4 (ON=démarr. à chaud)	V/F — 8	
S15 UWdg	Bloc commutateurs DIP, Etat SW 5 (Ch/grd valid sur TU)	V/F — 1	A
S16 Def0	Bloc commutateurs DIP, Etat SW 6 (Bit 0, préconfig BD)	V/F — 2	
S17 Def1	Bloc commutateurs DIP, Etat SW 7 (Bit 1, préconfig BD)	V/F — 4	
S18 Def2	Bloc commutateurs DIP, Etat SW 8 (Bit 2, préconfig BD)	V/F — 8	
MinRpt1-4	Répétition minimale, tâches utilisateur 1 à 4	secs.	
TaskHalt	Arrêt/marche tâche utilisateur (VRAI arrêt, FAUX reprise)	Champ binaire	
UsrTask1	Contrôle tâche utilisateur 1	V/F —	
UsrTask2	Contrôle tâche utilisateur 2	V/F —	
UsrTask3	Contrôle tâche utilisateur 3	V/F —	
UsrTask4	Contrôle tâche utilisateur 4	V/F —	
AnConBlk	"Données opérateur" nom bloc AN_CONN	Alphanumérique	
DgConBlk	"Données opérateur" nom bloc DG_CONN	Alphanumérique	
Log_File	Numéro fichier journal fermé dans E ² ROM	(1-99)	

Table 10-4 Paramètres du bloc T600

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Time, Date (Heure, Date). Permet l'accès local aux informations d'heure et de date nécessaires pour le traitement des alarmes. L'heure et la date sont maintenues entre les unités sur une liaison de communication d'égal à égal (dans le cas du régulateur T640 qui n'a pas d'horloge en temps réel physique). Si l'horloge en temps réel n'est pas initialisée, le champ d'alarme *RTCinit* est mis à 1.

IP_type. Champ binaire permettant de spécifier la fréquence du secteur et le système/unités de température. Les spécifications s'appliquent à un instrument de la série T600 dans son intégralité et affectent les blocs TCOUPLE, FULL_TC8, RTD et UCHAR, les tables de linéarisation interne et la compensation de soudure froide. Les unités ne peuvent être mélangées dans un instrument unique, mais différentes unités sur le LIN peuvent être configurées individuellement.

■ **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz. Il s'agit de définir la fréquence pour les circuits de suppression du bruit dans les modules E/S du T231.

■ **Imperial (Impérial).** VRAI permet de sélectionner le système de température impérial - Fahrenheit (°F)/Rankine (R) - FAUX le système SI - Celsius (°C)/Kelvin (K).

Site1 à Site8. Ces champs permettent d'identifier le type de module du bus série interne affecté à chacune des adresses (1 à 8) ISB du module E/S sur le bus série de communication interne.

BrownOut (Coupure d'alimentation). Permet de spécifier la durée (minutes) d'une interruption d'alimentation nécessaire pour déclencher l'alarme *BrownOut*. En pratique, il s'agit de la durée maximale d'interruption de l'alimentation électrique qui peut être tolérée sans réaction défavorable des installations.

ColdStrt (Démarrage à froid). Permet de spécifier la durée minimale (minutes) d'une interruption d'alimentation qui provoquera un démarrage à froid d'un instrument de la série T600. Après un démarrage à froid, une ré-initialisation complète de la base de données des paramètres et du schéma de boucles est effectuée. Un démarrage à chaud se produit après des interruptions de courant plus courtes que *Coldstrt*, c'est à dire que le schéma de boucles actif est relancé avec la base de données de paramètres et les modes de fonctionnement actifs.

NOTA. Le démarrage de l'instrument après une coupure de courant est déterminé par la position des commutateurs 3 et 4 (validation Démarrage à froid et à chaud) du bloc de commutateurs 1. Les deux commutateurs doivent être mis sur “ON” pour assurer les fonctions complètes de démarrage à chaud et à froid.

Node (Noeud). Adresse du réseau ALIN. Ce champ est en lecture uniquement - 01 à FE hexadécimal.

Model (Modèle). Partie numérique de l'identificateur du modèle de T600. Par ex. "640" = instrument T640.

Software version number (N° de version du logiciel). Le numéro de version du logiciel de la série T600 est affiché à droite du champ *Alarms*.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **BrownOut (Coupure d'alimentation).** Indique une coupure de courant qui dépasse la durée définie dans le champ *BrownOut*. Cette alarme est affichée comme une alarme "d'événement" non acquittée, mais n'a pas d'état acquitté.
- **ComS/W.** Indique une erreur logiciel de bloc commune.
- **UTskEr1 à UTskEr4.** Indique que la tâche utilisateur correspondante (1 à 4) présente une erreur de total de contrôle.
- **MainPSU.** Indique une panne de l'alimentation principale.
- **StbyPSU.** Indique une panne de l'alimentation de secours.
- **RTCinit.** Indique que l'horloge en temps réel n'a pas été initialisée.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

UsrAlm. Ce champ peut être relié pour permettre au schéma de boucles de contrôler le relais du chien de garde comme une alarme utilisateur. Une entrée VRAIE ouvre les contacts normalement fermés du relais. Mais, si le commutateur 5 du bloc de commutateurs 1 est sur "ON" (Validation chien de garde sur tâche utilisateur), *UsrAlm* est forcé à VRAI (c'est à dire contacts ouverts) par une erreur de total de contrôle de la tâche utilisateur.

AreaNo. Zone de clé de sécurité (0 à 31). Permet de spécifier la zone des installations dans laquelle cet instrument est situé. Si *AreaNo* est dans la plage de 1 à 31, une clé de sécurité avec le même numéro de zone est nécessaire pour accéder aux modes INSPECT (INSPECTION) de la base de données de la face avant. Si *AreaNo* est 0, n'importe quelle clé de sécurité de numéro de zone est acceptée. (*Dans la version actuelle du logiciel, seules les zones 0 à 8 sont disponibles*).

TimeOut (Dépassement du temps imparti). Utilisé dans les modes INSPECT (INSPECTION) de la base de données. Si aucun bouton-poussoir de la face avant n'est enfoncé pour les secondes de *TimeOut*, la face avant revient à son mode DISPLAY (AFFICHAGE) normal. La mise à 0 de *TimeOut* invalide cette fonction.

Options. Champ binaire indiquant l'état des communications et du matériel des instruments de la série T600.

- **FPdis1 à FPdis4.** Lorsque ces bits sont VRAIS, l'accès opérateur de la face avant aux boucles 1 à 4 est interdit.

- **NoKeyPrt.** Lorsque ce bit est VRAI, l'accès aux modes "INSPECT" (INSPECTION) "partiels" de la base de données sur la face avant est autorisé sans qu'il soit nécessaire d'utiliser la clé de sécurité appropriée.
- **NoKeyFul.** Lorsque ce bit est VRAI, l'accès aux modes "INSPECT" (INSPECTION) "complets" de la base de données sur la face avant est autorisé sans qu'il soit nécessaire la clé de sécurité appropriée. Noter que *NoKeyFul* annule *NoKeyPrt* si les deux sont VRAIS, et ces deux bits ne peuvent être modifiés par l'intermédiaire du mode INSPECT de la face avant, si seul l'accès partiel est utilisé.
- **LEDtest (Test LED).** Une entrée VRAIE de ce bit permet d'effectuer un test des LED de la face avant.
- **CommsDis.** Une entrée VRAIE invalide les écritures dans les blocs qui sont mis en image de manière externe, mais n'invalide pas les écritures dans les blocs image des instruments *locaux* de la série T600. *CommsDis* = VRAI permet d'empêcher les sélections de communications binaires.
- **FullSave, PartSave (Sauvegarde complète, partielle).** Permet de sauvegarder la base de données active (dans la RAM) avec le même nom de fichier à partir duquel elle a été chargée. Si *FullSave* est VRAI, la totalité de l'image RAM est sauvegardée. *PartSave* permet de sauvegarder l'image de la RAM à l'exception des "données (opérateur) tièdes" qui conservent les valeurs dans le fichier existant. *FullSave* annule *PartSave*, si les deux sont VRAIS.

NOTA. Les données tièdes sont sauvegardées une fois par itération de tâche, et à la mise hors tension, et comprennent les valeurs *SL*, *OP* et *MODE* pour chaque boucle PID, plus un bloc AN_CONN et un bloc DG_CONN (spécifié par *AnConBlk* et *DgConBlk*). Lors d'une tentative de démarrage à chaud, si l'image RAM est erronée, un démarrage à froid est effectué et les données tièdes sont écrasées. Noter qu'un seul bloc TP_CONN, appelé *AnConBlk*, peut définir des données tièdes pour servir d'alternative au bloc AN_CONN et au bloc DG_CONN.

- **BinSpd1, BinSpd2.** Permet de définir la vitesse de la ligne de la liaison de communication binaire RS422, voir table 10-5.

BinSpd1	BinSpd2	Débit en baud sélectionné
FAUX	FAUX	9600
FAUX	VRAI	4800
VRAI	FAUX	1200
VRAI	VRAI	300

Table 10-5 Sélection du débit en baud de la liaison de communication binaire - BinSpd1 & BinSpd2

- **Protectd.** Permet de protéger la base de données en la sauvegardant sous un format codé, à condition que le module de mémoire du T640 contienne un code de codage spécial. *Protectd* ne peut être mis à VRAI, à moins qu'un code valable ne soit présent,

et une fois VRAI, il n'est pas possible de repasser à FAUX. Si *Protectd* est VRAI, le fichier .DBF de la base de données en utilisant la sauvegarde partielle ou complète est sauvegardé sous forme codée sur la base de la valeur du code particulier présent dans le module. Les fichiers .DBF codés peuvent être uniquement décodés par un instrument T640 qui dispose du code de codage correct dans son module de mémoire.

(Il s'agit d'une fonction spéciale qui n'est pas prise en compte par les instruments T640 standard. Demandez les détails à Eurotherm Systèmes).

- **E2Form1, E2Form2.** Ces deux bits permettent de reformater complètement le système d'archivage de l'E²ROM, ce qui peut être nécessaire s'il a été corrompu, par ex. à cause de la perte de l'alimentation électrique au cours d'une opération d'archivage. Afin d'éviter un fonctionnement accidentel de cette fonction, les bits doivent être mis à 1 et remis à zéro suivant une séquence particulière avant le reformatage. Si l'une des phases est mal exécutée, tous les bits doivent être remis à FAUX et il faut reprendre la séquence à partir de la phase 1. Les quatre phases de la séquence sont les suivantes:

- 1 Mettre *E2Form1* à VRAI
- 2 Mettre *E2Form1* à FAUX
- 3 Mettre *E2Form2* à VRAI
- 4 Mettre *E2Form1* à VRAI

A mesure que chaque phase est exécutée, celle-ci est affichée sur la face avant par un message d'alarme clignotant correspondant: **E2ROM 1 à E2ROM 4**. Ce n'est que lorsque le quatrième message est affiché que le reformatage est exécuté. Après un reformatage effectué avec succès, les quatre bits repassent automatiquement à FAUX et les fichiers actifs .DBF et .RUN sont régénérés à partir de la RAM.

Status (Etat). Champ binaire en lecture uniquement indiquant l'état des communications et du matériel de l'instrument de la série T600.

- **PwrFail (Panne de courant).** Ce bit est verrouillé à VRAI chaque fois que l'instrument est mis sous tension et agit donc comme un drapeau de panne de courant.
- **TmpPFail.** Indique qu'il y a eu une panne de courant temporaire. Le bit se met à 1 au cours de la première itération après la mise sous tension de la tâche utilisateur contenant le bloc T600, et se remet automatiquement à zéro au cours de la seconde itération. Au cours d'une panne de courant "temporaire", certaines données sont perdues/corrompues et le fichier de la base de données doit donc être rechargé dans la RAM, mais les "données tièdes" sont également écrasées. (Voir les détails sur les "données tièdes" sous *FullSave*, *PartSave* dans le paragraphe *Options* ci-dessus).
- **CommsAlm.** Ce bit est mis à 1 si un bloc image dans le T600 est en alarme logiciel en raison d'un défaut de communication (pas d'une anomalie mémoire). Le signal *CommsAlm* peut donc être utilisé par une unité T1000 de supervision pour contrôler le bon fonctionnement des communications inter T600, même lorsque les blocs affectés eux-mêmes ne sont pas visibles depuis le T1000. Il suffit de mettre en image les blocs de communication du T1000 pour rendre les bits *CommsAlm* accessibles.

- **Alarm (Alarme).** Indique que le relais d’alarme fonctionne, c’est à dire que ses contacts sont *ouverts*.
- **PrtKey, FulKey.** VRAI indique qu’une clé de sécurité valable est utilisée, la face avant étant en modes d’inspection “partielle” et “complète”.
- **BoardID1 à BoardID2.** Ces deux bits indiquent la fréquence de l’horloge UC sur la carte mère de l’instrument T600. Dans la version actuelle du matériel, la seule version disponible est 12,5 MHz, ce qui est indiqué par le fait que les deux bits sont FAUX.
- **S11 à S18.** Indique l’état des commutateurs 1 à 8 du bloc de commutateurs 1 de l’instrument de la série T600. VRAI = On, FAUX = Off.

MinRpt1 à MinRpt4. Ces champs permettent de spécifier le cycle de répétition minimum souhaité pour les tâches 1 à 4 avec une marge de 10 ms. Lorsque le serveur est initialisé, il n’essaie pas d’exécuter la tâche *plus* fréquemment que l’intervalle spécifié, mais il peut l’exécuter *moins* fréquemment, si le temps UC disponible est insuffisant. La mise à zéro de ces champs fait que le système sélectionne les valeurs minimales possibles, c’est à dire les cycles de répétition les plus rapides possibles.

TaskHalt. Ce champ binaire permet au schéma de boucles d’arrêter ou de relancer les tâches utilisateur.

- **UsrTask1 à UsrTask4.** Lorsque l’entrée est VRAIE, la tâche utilisateur correspondante (1 à 4 respectivement) est arrêtée. La tâche est relancée lorsque l’entrée repasse à FAUX. Noter qu’il n’est pas possible d’interrompre la tâche utilisateur contenant le bloc T600.

AnConBlk. Ce champ peut être utilisé de deux manières. Il permet de spécifier le nom de repère (champ *Block*) d’un bloc AN_CONN unique qui doit contenir les “données tièdes”, et ainsi, toutes les données de configuration peuvent être sauvegardées au moment de la mise hors tension. Il permet également de spécifier le nom de repère d’un bloc TP_CONN qui remplace à la fois un bloc AN_CONN et un bloc DG_CONN. Dans ce cas, le nom du bloc TP_CONN dans *AnConBlk* supprime automatiquement le champ *DgConBlk*.

DgConBlk. Permet de spécifier le nom de repère (champ *Block*) d’un bloc DG_CONN unique qui doit contenir les “données tièdes” logiques et ainsi, toutes les données de configuration peuvent être sauvegardées au moment de la mise hors tension. Noter que si un bloc TP_CONN a été désigné dans *AnConBlk*, qui remplace à la fois un bloc AN_CONN et un bloc DG_CONN, le champ *DgConBlk* est effacé automatiquement et ne permet pas l’écriture.

Log_File (Fichier_Consignation). Chaque fois que des paramètres sont modifiés par l’intermédiaire des modes INSPECT de la base de données sur la face avant, toutes ces modifications sont enregistrées dans un fichier de consignation dans E²ROM de l’instrument de la série T600. Le nom de fichier du journal de consignation a la même racine que le fichier .DBF à partir duquel le schéma a été chargé, mais avec l’extension .Lnn, où *nn* peut prendre les valeurs de 01 à 99. “*nn*” est incrémenté à mesure que chaque fichier de consignation est rempli et fermé.

Le champ *Log_File* montre le numéro (*nn*) du dernier fichier de consignation qui a été fermé et qui est prêt à être chargé dans un superviseur, si nécessaire.

Noter que seuls deux fichiers de consignation sont conservés dans l'E²ROM: le fichier fermé spécifié dans *Log_File*, et le fichier ouvert qui reçoit les enregistrements. Lorsque le fichier de consignation actif est rempli - ce qui prend au moins une minute - le fichier de consignation précédent (fermé) est disponible pour être chargé. Mais, dès que le fichier actif est rempli et fermé, la valeur dans *Log_File* est incrémentée et le fichier précédent est écrasé. Une valeur zéro dans *Log_file* signifie qu'il n'y a pas de fichiers de consignation valables dans l'E²ROM.

T221 : BLOC DE CONFIGURATION DU T221

Fonction du bloc

Ce bloc est un bloc de type “en-tête” qui doit figurer dans un schéma de boucles à exécuter dans un T221. Ses fonctions permettent d’accéder à l’horloge et au calendrier en temps réel de l’unité de base (configurés séparément pour chaque unité T221), de tester les LED de la face avant et le chien de garde, d’indiquer les adresses de réseau, les états LIN/ALIN et les états du relais, ainsi que les positions des commutateurs.

Paramètres du bloc

Les symboles utilisés dans la table 10-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Time, Date	Horloge/calendrier temps réel	hh:mm:ss/dd:mm:yy	
Options	Tests d’exploitation	(ABC)D hex	
LEDTest	Essais VRAI des LED de la face avant	V/F	1
WDTest	Essais VRAI du chien de garde	V/F	2
			4
			8
			D
LINAddr	Adresse LIN: AB=segment; CD=MAC	ABCD hex	
ALINAddr	Adresse ALIN: AB=segment; CD=MAC	ABCD hex	
LoListen	Adresse MAC LIN du protocole ancien le plus bas	CD hex	
HiListen	Adresse MAC LIN du protocole ancien le plus haut	CD hex	
Alarms			
Software	Alarme de mise à jour du bloc	V/F	
CommsAlm	Blocs image non mis à jour	V/F	
MainPSU	Défaut alimentation principale	V/F	
StbyPSU	Défaut alimentation de secours	V/F	
RTCinit	Horloge en temps réel non-initialisée	V/F	
Combined	Alarme combinée du bloc	V/F	
Status		(AB)CD hex	
PwrFail	Mis à 1 à mise sous tension (RAZ par l'utilisateur)	V/F	1
TmpPFail	Mis à 1 à mise sous tension (RAZ auto. après 1 échantillonn.)	V/F	2
			4
			8
			D
CommsAlm	Au moins 1 bloc image n’est pas mis à jour	V/F	
			1
LINRly	Etat relais LIN (VRAI = fermé, càd ok)	V/F	2
ALINRly	Etat relais ALIN (VRAI = fermé, càd ok)	V/F	4
LINFail	Etat LIN (VRAI = défaut)	V/F	8
ALINFail	Etat ALIN (VRAI = défaut)	V/F	C

suite...

...suite

Paramètre	Fonction	Unités	Etat
SW_2&1	Position commutateurs carte mère (VRAI = "on")	ABCD hex	
S11 Diag	Position SW1/1 (valid. test auto à la mise sous tension)	V/F — 1	D
S12 ACfg	Position SW1/2 (validation auto config)	V/F — 2	
S13 LEDs	Position SW1/3 (validation LED mise au point)	V/F — 4	
S14 WDog	Position SW1/4 (validation chien de garde)	V/F — 8	
S15	Position SW1/5	V/F — 1	C
S16	Position SW1/6	V/F — 2	
S17	Position SW1/7	V/F — 4	
S18	Position SW1/8	V/F — 8	
S21	Position SW2/1	V/F — 1	B
S22	Position SW2/2	V/F — 2	
S23	Position SW2/3	V/F — 4	
S24	Position SW2/4	V/F — 8	
S25	Position SW2/5	V/F — 1	A
S26	Position SW2/6	V/F — 2	
S27	Position SW2/7	V/F — 4	
S28	Position SW2/8	V/F — 8	
SW_4&3	Positions des commutat. de la carte passerelle (VRAI="on")	ABCD hex	
S31	Position SW3/1	V/F — 1	D
S32	Position SW3/2	V/F — 2	
S33	Position SW3/3	V/F — 4	
S34	Position SW3/4	V/F — 8	
S35	Position SW3/5	V/F — 1	C
S36	Position SW3/6	V/F — 2	
S37	Position SW3/7	V/F — 4	
S38	Position SW3/8	V/F — 8	
S41	Position SW4/1	V/F — 1	B
S42	Position SW4/2	V/F — 2	
S43	Position SW4/3	V/F — 4	
S44	Position SW4/4	V/F — 8	
S45	Position SW4/5	V/F — 1	A
S46	Position SW4/6	V/F — 2	
S47	Position SW4/7	V/F — 4	
S48	Position SW4/8	V/F — 8	

Table 10-6 Paramètres du bloc T221

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Time, Date (Heure, Date). Horloge et calendrier en temps réel dans le T221. Si nécessaire, ces champs peuvent être définis par l'utilisateur après la mise sous tension.

LoListen, HiListen. Adresses MAC LIN de l'ancien protocole inférieur et supérieur disponible pour l'acheminement.

T102: BLOC DE CONFIGURATION DU T102

Fonction du bloc

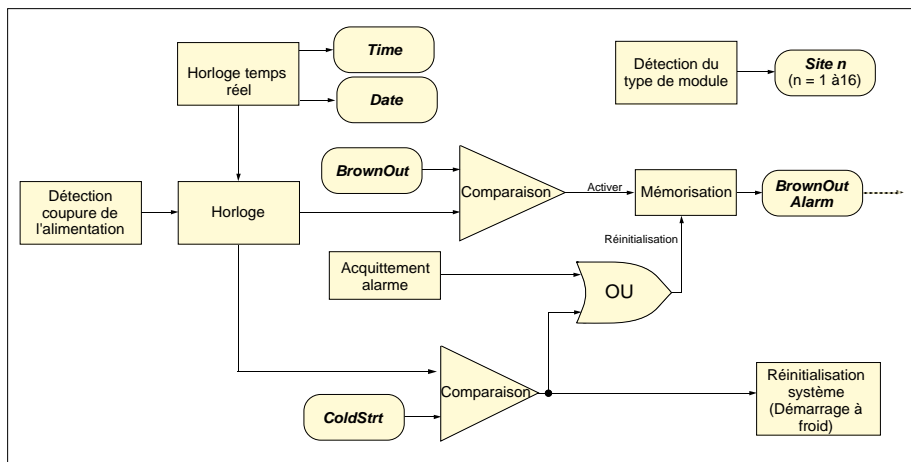


Figure 10-4 Schéma du bloc T102

Voir le schéma de la figure 10-4. Ce bloc est un type de bloc "en-tête" qui doit figurer dans un schéma de boucles à exécuter dans un T102. Il permet d'accéder à l'horloge temps réel de l'unité de base (configurée indépendamment pour chaque unité T102), spécifie les unités de température pour un T102 unique et une heure de démarrage à froid (réinitialisation du système, et comprend une alarme de coupure d'alimentation qui est mémorisée lorsque l'alimentation est coupée pendant un temps donné. Le bloc indique également le type de module entrées/sorties qui équipe chacun des 16 sites de l'unité de base T102, ainsi qu'un certain nombre de paramètre de communication/équipement du T102.

Paramètres du bloc

Les symboles utilisés dans la table 10-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Time, Date. Horloge et calendrier temps réel indépendants. Ces champs permettent de modifier l'heure et la date d'un T102 pendant que le système tourne en ligne.

IP_type. Champ binaire indiquant la fréquence du secteur et les unités de température. Les spécifications s'appliquent à l'ensemble d'un T102 et affectent les blocs TCOUPLE, FULL_TC8, RTD et UCHAR, les tables de linéarisation internes et la compensation de soudure froide. Les unités ne peuvent être mélangées sur un seul T102, mais différents T102 sur le réseau ALIN peuvent être configurés différemment.

Paramètre	Fonction	Unités	Etat
Time	Horloge temps réel	hh:mm:ss	
Date	Réinitialisation calendrier	jj/mm/aa	
IP_type	Fréquence secteur, unités de température	(ABC)D hex	
60Hz	Fréquence secteur (VRAI = 60Hz, FAUX = 50Hz)	V/F — 1	D
Imperial	Système temp. (VRAI = Impérial, FAUX = SI)	V/F — 2	
Absolute	Unités de temp. (VRAI = Absolu, FAUX = Relatif)	V/F — 4	
		V/F — 8	
Site1 to Site16	Repères du module E/S aux sites 1 à 16 T102	Alphanumérique	
ColdStrt	Durée coupure min. pour un démarrage à froid	min	
BrownOut	Durée coupure pour alarme BrownOut	min	
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
BrownOut	Coupure alim. plus longue que BrownOut	V/F	
ComS/W	Erreur logiciel dans blocs	V/F	
NodeSw	Commutat. adr. noeud modif. depuis mise sous tension	V/F	
RTCBat	Défaut batterie secours horloge temps réel	V/F	
MemBat	Défaut/absence batterie secours mémoire	V/F	
Combined	Alarme combinée du bloc	V/F	
Features	(Champ binaire non-utilisé)	ABCD hex	
Node	Adresse noeud ALIN, lue commutateurs d'adresse	(01-FE hex)	
Status	Etat communicat./équipement	ABCD hex	*
ColdStrt	Etat sw fonction 1 du T920 (dém. à froid active)	V/F — 1	D
WarmStrt	Etat sw fonction 2 du T920 (dém. à chaud active)	V/F — 2	
		V/F — 4	
		V/F — 8	
TermCfg	Etat sw fonction 4 du T920 (mode exploit./config.)	V/F — 8	
MemBat	VRAI = Défaut/absence batterie secours mémoire	V/F — 1	C
RTCBat	VRAI = Défaut batterie horloge temps réel	V/F — 2	
		V/F — 4	
		V/F — 8	
IOfinger	VRAI = Modules E/S disponibles (càd T102)	V/F — 8	
		V/F — 1	B
		V/F — 2	
		V/F — 4	
		V/F — 8	
		V/F — 1	A
CommsAlm	Défaut communication blocs en mémoire cache	V/F — 2	
PwrFail	BD lancée par mise sous tension (réinit. utilisateur)	V/F — 4	
TmpPFail	Comme PwrFail, mais réinit. auto à la 2è itération BD	V/F — 8	

Table 10-7 suite...

- **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz. Définit la fréquence pour les circuits de rejet du bruit dans les modules entrées/sorties du T102.
- **Imperial.** VRAI sélectionne le système de température Impérial — Fahrenheit (°F)/Rankine (R). FAUX le système SI — Celsius (°C)/Kelvin (K).
- **Absolute.** VRAI sélectionne les unités de température absolue — Kelvin/Rankine. FAUX les unités relatives — Celsius/Fahrenheit.

...Table 10-7 suite

Paramètre	Fonction	Unités	Etat
Options	Etat communication/équipement	(AB)CD hex	
CommsDis	VRAI désactive écritures entrantes du champ ALIN	V/F — 1	D
Protectd†	VRAI code/décode BD à l'enregistrement/chargement	V/F — 2	
CONFspd	VRAI=mode Configurateur, FAUX=mode exploitation	V/F — 4	
FullSave	VRAI enregistre BD active dans un fichier .DBF	V/F — 8	
PartSave	VRAI enregistre BD active dans un fichier .SBF	V/F — 1	C
		V/F — 2	
		V/F — 4	
		V/F — 8	
CldStPri	Ce T920 est l'UC principale de démarrage à froid	V/F	

Table 10-7 Paramètres du bloc T102

†Non géré à l'heure actuelle. *Sauf *PwrFail*

Site1 à Site16. Chaque champ affiche le nom de repère (pas le type) du bloc correspondant au module entrées/sorties équipant ce numéro de site. Un champ vierge indique qu'il n'y a aucun module ou qu'aucun module n'est configuré sur ce site.

ColdStrt. Définit la durée minimale (minutes) d'une coupure de l'alimentation qui entraînera un démarrage à froid du T102. Un "démarrage à chaud" se produit après des coupures d'alimentation inférieures à *ColdStrt*; autrement dit, le schéma de boucles actif est relancé avec la base de données et les modes d'exploitation existants. Après un démarrage à froid, par contre, la base de données est complètement réinitialisée, et le schéma de boucles défini dans un fichier ayant l'extension .RUN est chargé est exécuté (si possible).

Lorsqu'un fichier *basededonnées*.DBF est téléchargé depuis un noeud déporté ou chargé en utilisant le terminal local du T102, un fichier *basededonnées*.RUN est créé et tous les autres fichiers dont l'extension est .RUN sont supprimés. Au cours du démarrage à froid, le T102 recherche dans son système de fichiers un fichier ayant l'extension .RUN. S'il le trouve, le fichier de la base de données correspondante est chargé et exécuté. En l'absence d'un fichier .RUN ou si pour une raison quelconque il y en a plusieurs, le T102 tente de charger la base de données par défaut E:T920.DBF. En cas d'échec, le T102 ne lance pas sa conduite.

BrownOut. Définit la durée (minutes) d'une interruption de l'alimentation qui déclenche l'alarme *BrownOut*.

Numéro de version du logiciel. Le numéro de version de logiciel de l'instrument est automatiquement chargé dans le champs "units" du paramètres *Alarms* au démarrage. Vérifiez la version du logiciel dans ce champ en utilisant le port de configuration de l'instrument. Notez que les "units" ne sont pas transmises sur le réseau LIN et ne peuvent donc pas être vérifiées en utilisant la fonction VIEW de LINtools.

A la suite du numéro de version du logiciel, il y a entre parenthèses la lettre d'option de l'instrument. (C) pour l'option régulation uniquement (logiciel séquence non-installé) et (S) pour l'option séquence (logiciel séquence installé). Les blocs SFC_CON, SFC_MON et SFC_DISP ne peuvent être exécutés (c'est à dire mis à jour) que dans un instrument disposant de l'option (S), bien qu'ils puissent être définis et configurés dans l'option (C) pour être utilisés comme blocs *en mémoire cache* depuis d'autres instruments (S).

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **Software.** Anomalie dans les données RAM du bloc.
- **BrownOut.** Indique une défaillance de l'alimentation qui dépasse la durée définie dans le champ *BrownOut*. Cette alarme apparaît comme une alarme d'événement non-acquittée, mais n'a pas d'état acquitté.
- **ComS/W.** Passe à 1 si une alarme logiciel se produit dans un bloc de la base de données active.
- **NodeSw.** Passe à 1 si l'adresse du noeud lue sur les commutateurs d'adresse ALIN du T102 a changé depuis la mise sous tension.
- **RTCBat.** Passe à 1 en cas de défaut de la batterie de secours de l'horloge temps réel. *RTCBat* passe à 1 si le bit *Status.RTCBat* correspondant est VRAI.
- **MemBat.** Passe à 1 en cas de défaut ou en l'absence de batterie de secours mémoire. *MemBat* passe à 1 si le bit *Status.MemBat* correspondant est VRAI.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Features (Fonctions). (*Champ binaire non-utilisé à l'heure actuelle*).

Node (Noeud). Adresse du noeud ALIN de l'instrument lue par les commutateurs d'adresse du T102 en conduite.

Status (Etat). Champ binaire indiquant l'état des communications et des équipements du T102.

- **ColdStrt.** Etat du commutateur de fonction 1 du T920 — Démarrage à froid activé. VRAI = ON.
- **WarmStrt.** Etat du commutateur de fonction 2 du T920 — Démarrage à chaud activé. VRAI = ON.
- **TermCfg.** Etat du commutateur de fonction 4 du T920 — mode exploitation/configurateur. VRAI = ON.

Lorsque le commutateur 4 est sur ON, 80 % du temps UC du T920 est consacré à la mise à jour des blocs, le reste étant disponible pour la tâche du configurateur, s'il est utilisé. Sinon, ces 20 % sont gaspillés. Autrement dit, même si le schéma de boucles ne répond pas aussi vite que possible, le configurateur peut être utilisé aussi souvent que possible sans que les performances en soient affectées.

Lorsque le commutateur 4 est sur OFF, et que le configurateur n'est pas utilisé, l'UC passe 95 % de son temps à mettre à jour les blocs. Mais, lorsque le configurateur est utilisé, ce pourcentage tombe à 50 %, les autres 50 % étant affectés au configurateur. Donc, même si la régulation est normalement à son efficacité maximale, le ralentissement peut être très net à chaque fois que le configurateur est utilisé.

- **MemBat.** VRAI en cas de défaut de la batterie de secours de la mémoire.
- **RTCBat.** VRAI en cas de défaut de la batterie de l'horloge temps réel.

- **IOfinger.** VRAI si les modules entrées/sorties ("fingers ou module") sont disponibles. Ce bit peut être utilisé pour faire la distinction entre un T102 et un T302 (pas d'entrées/sorties).
- **CommsAlm.** Ce bit est mis à 1 si un bloc en mémoire cache dans l'unité T102 est en alarme logiciel, en raison d'un défaut de communication (et non pas, une erreur de total de contrôle). Le signal *CommsAlm* peut donc être utilisé par un instrument de supervision pour vérifier le bon fonctionnement des communications entre T102, même lorsque les blocs affectés proprement dits ne sont pas visibles depuis le superviseur. Il suffit de mettre en mémoire cache les blocs T102 qui communiquent pour que leurs bits *CommsAlm* soient accessibles.

Ce bit peut être utilisé en liaison avec l'alarme *ComS/W* pour déterminer s'il s'agit d'un défaut de communication OU d'une erreur de total de contrôle.
- **PwrFail.** Passe à 1 si la base de données a été lancée par une mise sous tension (plutôt que par une demande utilisateur, par ex. en utilisant le configurateur de terminal ou une commande ALIN déportée). Notez que *PwrFail* peut être réinitialisé par l'utilisateur.
- **TmpPFail.** Ce bit est mis à 1 de la même manière que *PwrFail*, mais est automatiquement remis à zéro au cours de la seconde itération de la base de données.

Options. Champ binaire définissant certaines options d'état des communications et des équipements du T102.

- **CommsDis.** Une entrée VRAIE désactive les écritures entrantes du champ ALIN.
- **Protectd.** VRAI entraîne le codage à l'enregistrement et le décodage au chargement des bases de données (*fonction indisponible à l'heure actuelle*).
- **CONFspd.** Utilisé dans les UC v2/1 et ultérieures pour sélectionner la *vitesse du configurateur*. (Les UC plus anciennes utilisaient un commutateur physique). VRAI sélectionne le mode Configurateur, tandis que FAUX le mode exploitation. Le mode Configurateur entraîne une régulation fonctionnant à une efficacité réduite (~85 %), mais n'est pas affectée par l'exécution du configurateur résident. Le mode exploitation permet une régulation à efficacité maximale lorsque le configurateur ne tourne pas, mais celle-ci est réduite à ~53 %, lorsque le configurateur est utilisé.
- **FullSave.** VRAI enregistre la base de données active dans le fichier .DBF actif.
- **PartSave.** VRAI enregistre la base de données active dans un fichier appelé *nomdufichier.SBF*, où *nomdufichier* est le nom racine du fichier de base de données actif. Notez que l'exécution de la base de données est suspendue pendant le processus d'enregistrement.

CldStPri. VRAI indique que cette UC T920 est l'UC principale de démarrage à froid, autrement dit, c'est celle qui tentera de démarrer comme régulateur principal après une coupure d'alimentation.

T302: BLOC DE CONFIGURATION DU T302

Fonction du bloc

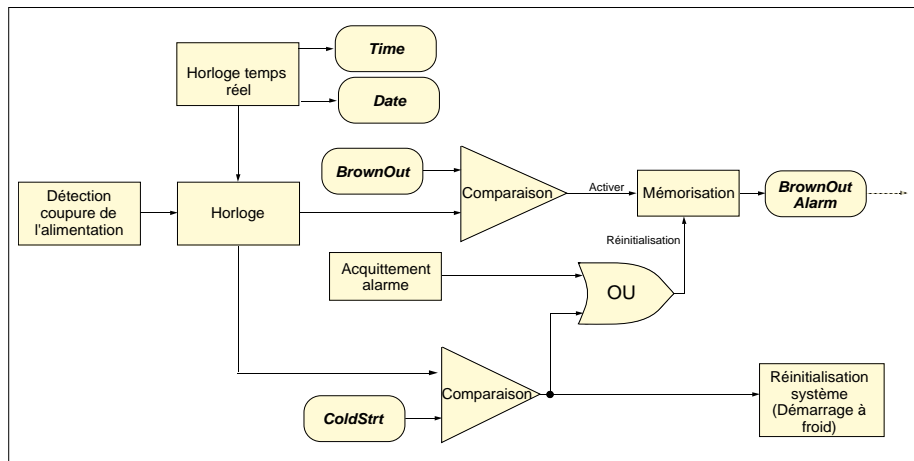


Figure 10-5 Schéma du bloc T302

Voir le schéma de la figure 10-5. Ce bloc est un type de bloc "en-tête" qui doit figurer dans un schéma de boucles à exécuter dans un T302. Il permet d'accéder à l'horloge temps réel de l'unité de base (configurée indépendamment pour chaque unité T302), spécifie les unités de température pour un T302 unique et une heure de démarrage à froid (réinitialisation du système, et comprend une alarme de coupure d'alimentation qui est mémorisée lorsque l'alimentation est coupée pendant un temps donné. Le bloc donne également un certain nombre de paramètres de communication/équipement du T302.

Notez que le bloc T302 est presque identique au bloc T102, sauf qu'il ne dispose pas des champs *Site1* à *Site16*, puisque le T302 ne permet pas de gérer les modules entrées/sorties.

Paramètres du bloc

Les symboles utilisés dans la table 10-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Time, Date. Horloge et calendrier temps réel indépendants. Ces champs permettent de modifier l'heure et la date d'un T302 pendant que le système tourne en ligne.

Paramètre	Fonction	Unités	Etat
Time	Horloge temps réel	hh:mm:ss	
Date	Réinitialisation calendrier	jj/mm/aa	
IP_type	Fréquence secteur, unités de température	(ABC)D hex	
60Hz	Fréquence secteur (VRAI = 60Hz, FAUX = 50Hz)	V/F — 1	D
Imperial	Système temp. (VRAI = Impérial, FAUX = SI)	V/F — 2	
Absolute	Unités de temp. (VRAI = Absolu, FAUX = Relatif)	V/F — 4	
		V/F — 8	
ColdStrt	Durée coupure min. pour un démarrage à froid	min	
BrownOut	Durée coupure pour alarme BrownOut	min	
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
BrownOut	Coupure alim. plus longue que BrownOut	V/F	
ComS/W	Erreur logiciel dans blocs	V/F	
NodeSw	Commutat. adr. noeud modif. depuis mise sous tension	V/F	
RTCBat	Défaut batterie secours horloge temps réel	V/F	
MemBat	Défaut/absence batterie secours mémoire	V/F	
Combined	Alarme combinée du bloc	V/F	
Features	(Champ binaire non-utilisé)	ABCD hex	
Node	Adresse noeud ALIN, lue commutateurs d'adresse	(01-FE hex)	
Status	Etat communicat./équipement	ABCD hex	
ColdStrt	Etat sw fonction 1 du T920 (dém. à froid activé)	V/F — 1	D
WarmStrt	Etat sw fonction 2 du T920 (dém. à chaud activé)	V/F — 2	
		V/F — 4	
TermCfg	Etat sw fonction 4 du T920 (mode exploit./config.)	V/F — 8	
MemBat	VRAI = Défaut/absence batterie secours mémoire	V/F — 1	C
RTCBat	VRAI = Défaut batterie horloge temps réel	V/F — 2	
		V/F — 4	
IOfinger	VRAI = Modules E/S disponibles (càd T302)	V/F — 8	
		V/F — 1	B
		V/F — 2	
		V/F — 4	
		V/F — 8	
CommsAlm	Défaut communication blocs en mémoire cache	V/F — 1	A
PwrFail	BD lancée par mise sous tension (réinit. utilisateur)	V/F — 2	
		V/F — 4	
TmpPFail	Comme PwrFail, mais réinit. auto à la 2è itération BD	V/F — 8	

Table 10-8 suite...

IP_type. Champ binaire indiquant la fréquence du secteur et les unités de température. Les spécifications s'appliquent à l'ensemble d'un T302 et affectent les blocs TCOUPLE, FULL_TC8, RTD et UCHAR, les tables de linéarisation internes et la compensation de soudure froide. Les unités ne peuvent être mélangées sur un seul T302, mais différents T302 sur le réseau ALIN peuvent être configurés différemment.

■ **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz.

...Table 10-8 suite

Paramètre	Fonction	Unités	Etat
Options	Etat communication/équipement	(AB)CD hex	
CommsDis	VRAI désactive écritures entrantes du champ ALIN	V/F — 1	D
Protectd†	VRAI code/décode BD à l'enregistrement/chargement	V/F — 2	
CONFspd	VRAI=mode Configurateur, FAUX=mode exploitation	V/F — 4	
FullSave	VRAI enregistre BD active dans un fichier .DBF	V/F — 8	
PartSave	VRAI enregistre BD active dans un fichier .SBF	V/F — 1	C
		V/F — 2	
		V/F — 4	
		V/F — 8	
CldStPri	Ce T920 est l'UC principale de démarrage à froid	V/F	

Table 10-8 Paramètres du bloc T302

†Non géré à l'heure actuelle. *Sauf PwrFail

- **Imperial.** VRAI sélectionne le système de température Impérial — Fahrenheit (°F)/Rankine (R). FAUX le système SI — Celsius (°C)/Kelvin (K).
- **Absolute.** VRAI sélectionne les unités de température absolues — Kelvin/Rankine. FAUX les unités relatives — Celsius/Fahrenheit.

ColdStrt. Définit la durée minimale (minutes) d'une coupure de l'alimentation qui entraînera un démarrage à froid du T302. Un "démarrage à chaud" se produit après des coupures d'alimentation inférieures à *ColdStrt*; autrement dit, le schéma de boucles actif est relancé avec la base de données et les modes d'exploitation existants. Après un démarrage à froid, par contre, la base de données est complètement réinitialisée, et le schéma de boucles défini dans un fichier ayant l'extension .RUN est chargé est exécuté (si possible).

Lorsqu'un fichier *basededonnées.DBF* est téléchargé depuis un noeud déporté ou chargé en utilisant le terminal local du T302, un fichier *basededonnées.RUN* est créé et tous les autres fichiers dont l'extension est .RUN sont supprimés. Au cours du démarrage à froid, le T102 recherche dans son système de fichiers un fichier ayant l'extension .RUN. S'il le trouve, le fichier de la base de données correspondante est chargé et exécuté. En l'absence d'un fichier .RUN ou si pour une raison quelconque il y en a plusieurs, le T302 tente de charger la base de données par défaut E:T920.DBF. En cas d'échec, le T302 ne lance pas sa conduite.

BrownOut. Définit la durée (minutes) d'une interruption de l'alimentation qui déclenche l'alarme *BrownOut*.

Numéro de version du logiciel. Le numéro de version de logiciel de l'instrument est automatiquement chargé dans le champs "units" du paramètres *Alarms* au démarrage. Vérifiez la version du logiciel dans ce champ en utilisant le port de configuration de l'instrument. Notez que les "units" ne sont pas transmises sur le réseau LIN et ne peuvent donc pas être vérifiées en utilisant la fonction VIEW de LINTools.

A la suite du numéro de version du logiciel, il y a entre parenthèses la lettre d'option de l'instrument. (C) pour l'option régulation uniquement (logiciel séquence non-installé) et (S) pour l'option séquence (logiciel séquence installé). Les blocs SFC_CON, SFC_MON et SFC_DISP ne peuvent être exécutés (c'est à dire mis à jour) que dans un instrument

disposant de l'option (S), bien qu'ils puissent être définis et configurés dans l'option (C) pour être utilisés comme blocs *en mémoire cache* depuis d'autres instruments (S).

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **Software.** Anomalie dans les données RAM du bloc.
- **BrownOut.** Indique une défaillance de l'alimentation qui dépasse la durée définie dans le champ *BrownOut*. Cette alarme apparaît comme une alarme d'événement non-acquittée, mais n'a pas d'état acquitté.
- **ComS/W.** Passe à 1 si une alarme logiciel se produit dans un bloc de la base de données active.
- **NodeSw.** Passe à 1 si l'adresse du noeud lue sur les commutateurs d'adresse ALIN du T302 a changé depuis la mise sous tension.
- **RTCBat.** Passe à 1 en cas de défaut de la batterie de secours de l'horloge temps réel. *RTCBat* passe à 1 si le bit *Status.RTCBat* correspondant est VRAI.
- **MemBat.** Passe à 1 en cas de défaut ou en l'absence de batterie de secours mémoire. *MemBat* passe à 1 si le bit *Status.MemBat* correspondant est VRAI.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Features (Fonctions). (*Champ binaire non-utilisé à l'heure actuelle*).

Node (Noeud). Adresse du noeud ALIN de l'instrument lue par les commutateurs d'adresse du T302 en conduite.

Status (Etat). Champ binaire indiquant l'état des communications et équipements du T302.

- **ColdStrt.** Etat du commutateur de fonction 1 du T920 — Démarrage à froid activé. VRAI = ON.
- **WarmStrt.** Etat du commutateur de fonction 2 du T920 — Démarrage à chaud activé. VRAI = ON.
- **TermCfg.** Etat du commutateur de fonction 4 du T920 — mode exploitation/configurateur. VRAI = ON.

Lorsque le commutateur 4 est sur ON, 80 % du temps UC du T920 est consacré à la mise à jour des blocs, le reste étant disponible pour la tâche du configurateur, s'il est utilisé. Sinon, ces 20 % sont gaspillés. Autrement dit, même si le schéma de boucles ne répond pas aussi vite que possible, le configurateur peut être utilisé aussi souvent que possible sans que les performances en soient affectées.

Lorsque le commutateur 4 est sur OFF, et que le configurateur n'est pas utilisé, l'UC passe 95 % de son temps à mettre à jour les blocs. Mais, lorsque le configurateur est utilisé, ce pourcentage tombe à 50 %, les autres 50 % étant affectés au configurateur. Donc, même si la régulation est normalement à son efficacité maximale, le ralentissement peut être très net à chaque fois que le configurateur est utilisé.

- **MemBat.** VRAI en cas de défaut de la batterie de secours de la mémoire.
- **RTCBat.** VRAI en cas de défaut de la batterie de l'horloge temps réel.
- **IOfinger.** VRAI si les modules entrées/sorties ("doigts") sont disponibles. Ce bit peut être utilisé pour faire la distinction entre un T102 et un T302 (pas d'entrées/sorties).
- **CommsAlm.** Ce bit est mis à 1 si un bloc en mémoire cache dans l'unité T302 est en alarme logiciel, en raison d'un défaut de communication (et non pas, une erreur de total de contrôle). Le signal *CommsAlm* peut donc être utilisé par un instrument de supervision pour vérifier le bon fonctionnement des communications entre T302, même lorsque les blocs affectés proprement dits ne sont pas visibles depuis le superviseur. Il suffit de mettre en mémoire cache les blocs T302 qui communiquent pour que leurs bits *CommsAlm* soient accessibles.
Ce bit peut être utilisé en liaison avec l'alarme *ComS/W* pour déterminer s'il s'agit d'un défaut de communication OU d'une erreur de total de contrôle.
- **PwrFail.** Passe à 1 si la base de données a été lancée par une mise sous tension (plutôt que par une demande utilisateur, par ex. en utilisant le configurateur de terminal ou une commande ALIN déportée). Notez que *PwrFail* peut être réinitialisé par l'utilisateur.
- **TmpPFail.** Ce bit est mis à 1 de la même manière que *PwrFail*, mais est automatiquement remis à zéro au cours de la seconde itération de la base de données.

Options. Champ binaire définissant certaines options d'état des communications et des équipements du T302.

- **CommsDis.** Une entrée VRAIE désactive les écritures entrantes du champ ALIN.
- **Protectd.** VRAI entraîne le codage à l'enregistrement et le décodage au chargement des bases de données (*fonction indisponible à l'heure actuelle*).
- **CONFspd.** Utilisé dans les UC v2/1 et ultérieures pour sélectionner la *vitesse du configurateur*. (Les UC plus anciennes utilisaient un commutateur physique). VRAI sélectionne le mode Configurateur, tandis que FAUX le mode exploitation. Le mode Configurateur entraîne une régulation fonctionnant à une efficacité réduite (~85 %), mais n'est pas affectée par l'exécution du configurateur résident. Le mode exploitation permet une régulation à efficacité maximale lorsque le configurateur ne tourne pas, mais celle-ci est réduite à ~53 %, lorsque le configurateur est utilisé.
- **FullSave.** VRAI enregistre la base de données active dans le fichier .DBF actif.
- **PartSave.** VRAI enregistre la base de données active dans un fichier appelé *nomdufichier.SBF*, où *nomdufichier* est le nom racine du fichier de base de données actif. Notez que l'exécution de la base de données est suspendue pendant le processus d'enregistrement.

CldStPri. VRAI indique que cette UC T920 est l'UC principale de démarrage à froid, autrement dit, c'est celle qui tentera de démarrer comme régulateur principal après une coupure d'alimentation.

T932: BLOC DE CONFIGURATION DU T932

Fonction du bloc

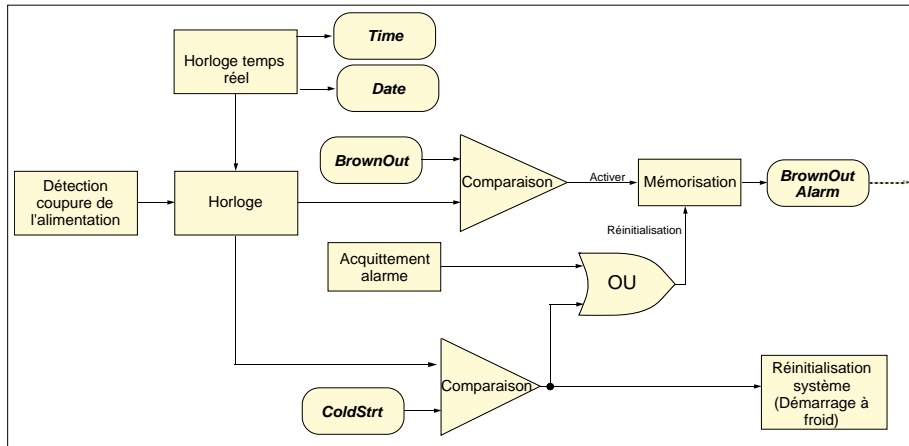


Figure 10-6 Schéma du bloc T932

Voir le schéma de la figure 10-6. Ce bloc est un type de bloc "en-tête" qui doit figurer dans un schéma de boucles à exécuter dans un T932. Il permet d'accéder à l'horloge temps réel de l'unité de base (configurée indépendamment pour chaque unité T932), spécifie les unités de température pour un T932 unique et une heure de démarrage à froid (réinitialisation du système, et comprend une alarme de coupure d'alimentation qui est mémorisée lorsque l'alimentation est coupée pendant un temps donnée.

NOTA. Un bloc T100 ou T231 peut être exécuté comme bloc en-tête dans un instrument T932 (au lieu d'un bloc T932) pour assurer la compatibilité avec les schémas de boucles existants.

Paramètres du bloc

Les symboles utilisés dans la table 10-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Time, Date. Horloge et calendrier temps réel indépendants. Ces champs permettent de modifier l'heure et la date d'un T932 pendant que le système tourne en ligne.

IP_type. Champ binaire indiquant la fréquence du secteur et les unités de température. Les spécifications s'appliquent à l'ensemble d'un T932 et affectent les blocs TCOUPLE,

Paramètre	Fonction	Unités	Etat
Time,Date	Horloge/calendrier temps réel	hh:mm:ss/jj/mm/aa	
IP_type	Fréquence secteur, unités de température	(ABC)D hex	
60Hz	Fréquence secteur (VRAI = 60Hz, FAUX = 50Hz)	V/F — 1	D
Imperial	Système temp. (VRAI = Impérial, FAUX = SI)	V/F — 2	
Absolute	Unités de temp. (VRAI = Absolu, FAUX = Relatif)	V/F — 4	
			8
Site1 to Site16	Repères du module E/S aux sites 1 à 16 T932 (v 3/I)	Alphanumérique	
BrownOut	Durée coupure pour alarme BrownOut	min	
ColdStrt	Durée coupure min. pour un démarrage à froid	min	
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
BrownOut	Coupure alim. plus longue que BrownOut	V/F	
S6000	Alarme logiciel bloc S6000	V/F	
MemBat	Défaut/absence batterie secours mémoire	V/F	
RTCBat	Défaut batterie secours horloge temps réel	V/F	
Combined	Alarme combinée du bloc	V/F	
Options	Options de communication	ABCD hex	
CommsDis	Désactivation des écritures dans les champs	V/F — 1	D
BinSpd1	Sélection vitesse communications binaires	V/F — 2	
BinSpd2	Sélection vitesse communications binaires	V/F — 4	
FullSave	VRAI enregistre BD active dans fichier .DBF	V/F — 8	
PartSave	VRAI enregistre BD dans fichier .SBF	V/F — 1	C
			2
			4
			8
Status	Etat communications/équipement	ABCD Hex	
RTCBat	VRAI = Défaut batterie horloge temps réel	V/F — 1	D
CommsAlm	Défaut communication bloc en mémoire cache	V/F — 2	
MemBat	VRAI = Défaut/absence batterie secours mémoire	V/F — 4	
Pwr Fail	Démarrage à chaud effectué	V/F — 8	
S21 Cold	Bloc 2, état Sw 1 (démarrage à froid activé)	V/F — 1	C
S22 Warm	Bloc 2, état Sw 2 (démarrage à chaud activé)	V/F — 2	
S23 Poll	Bloc 2, état Sw 3 (scrutation installations activée)	V/F — 4	
S24	Bloc 2, état Sw 4 (état maître/esclave)	V/F — 8	
LeftCPU	Position UC (VRAI=gauche, FAUX=droite)	V/F — 1	B
S11	Bloc 1, état commutateurs 1 à 7 (Adresse LIN = chiffres AB)	V/F — 2	
S12		V/F — 4	
S13		V/F — 8	
S14		V/F — 1	A
S15		V/F — 2	
S16		V/F — 4	
S17		V/F — 8	

Table 10-9 Paramètres du bloc T932

FULL_TC8, RTD et UCHAR, les tables de linéarisation internes et la compensation de soudure froide. Les unités ne peuvent être mélangées sur un seul T932, mais différents T932 sur le réseau ALIN peuvent être configurés différemment.

- **60Hz.** Fonctionnement secteur VRAI pour 60 Hz ou FAUX pour 50 Hz. Définit la fréquence pour les circuits de rejet du bruit dans les modules entrées/sorties du T932.
- **Imperial.** VRAI sélectionne le système de température Impérial — Fahrenheit (°F)/Rankine (R). FAUX le système SI — Celsius (°C)/Kelvin (K).
- **Absolute.** VRAI sélectionne les unités de température absolue — Kelvin/Rankine. FAUX les unités relatives — Celsius/Fahrenheit.

Site1 à Site16. Chaque champ affiche le nom de repère (pas le type) du bloc correspondant au module entrées/sorties équipant ce numéro de site. Un champ vierge indique qu'il n'y a aucun module ou qu'aucun module n'est configuré sur ce site. (*Entrées/sorties non gérées avant la version 3/1 du logiciel*).

BrownOut. Définit la durée (minutes) d'une interruption de l'alimentation qui déclenche l'alarme *BrownOut*.

ColdStrt. Définit la durée minimale (minutes) d'une coupure de l'alimentation qui entraînera un démarrage à froid du T932. Un "démarrage à chaud" se produit après des coupures d'alimentation inférieures à *ColdStrt*; autrement dit, le schéma de boucles actif est relancé avec la base de données et les modes d'exploitation existants. Après un démarrage à froid, par contre, la base de données est complètement réinitialisée, et le schéma de boucles défini dans un fichier ayant l'extension .RUN est chargé et exécuté (si possible).

Lorsqu'un fichier *basededonnées*.DBF est téléchargé par l'intermédiaire de Linfiler ou chargé en utilisant le terminal local du T932, un fichier *basededonnées*.RUN est créé et tous les autres fichiers dont l'extension est .RUN sont supprimés. Au cours du démarrage à froid, le T932 recherche dans son système de fichiers un fichier ayant l'extension .RUN. S'il le trouve, le fichier de la base de données correspondante est chargé et exécuté. En l'absence d'un fichier .RUN ou si pour une raison quelconque il y en a plusieurs, le T102 tente de charger la base de données par défaut E:T920.DBF. En cas d'échec, le T932 ne lance pas sa conduite.

Numéro de version du logiciel. Le numéro de version de logiciel de l'instrument est automatiquement chargé dans le champs "units" du paramètres *Alarms* au démarrage. Vérifiez la version du logiciel dans ce champ en utilisant le port de configuration de l'instrument. Notez que les "units" ne sont pas transmises sur le réseau LIN et ne peuvent donc pas être vérifiées en utilisant la fonction VIEW de LINtools.

A la suite du numéro de version du logiciel, il y a entre parenthèses la lettre d'option de l'instrument. (C) pour l'option régulation uniquement (logiciel séquence non-installé) et (S) pour l'option séquence (logiciel séquence installé) et (A) pour l'options fonctions évoluées. Les blocs SFC_CON, SFC_MON et SFC_DISP ne peuvent être exécutés (c'est à dire mis à jour) que dans un instrument disposant de l'option (S), bien qu'ils puissent être définis et configurés dans l'option (C) pour être utilisés comme blocs *en mémoire cache* depuis d'autres instruments (S). Les blocs AGA8DATA et GASCONC ne peuvent être utilisés qu'avec l'option (A) de l'instrument.

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **Software.** Anomalie dans les données RAM du bloc.
- **BrownOut.** Indique une défaillance de l'alimentation qui dépasse la durée définie dans le champ *BrownOut*. Cette alarme apparaît comme une alarme d'événement non-acquittée, mais n'a pas d'état acquitté.
- **S6000.** Indique une alarme logiciel (communication) dans un bloc S6000 connecté au T932.
- **MemBat.** Passe à 1 en cas de défaut ou en l'absence de batterie de secours mémoire. *MemBat* passe à 1 si le bit *Status.MemBat* correspondant est VRAI.
- **RTCBat.** Passe à 1 en cas de défaut de la batterie de secours de l'horloge temps réel. *RTCBat* passe à 1 si le bit *Status.RTCBat* correspondant est VRAI.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active à la priorité la plus haute du bloc.

Status (Etat). Champ binaire indiquant l'état des communications et des équipements du T932.

- **RTCBat.** (Good/Bad - bon/mauvais). "Bad" (VRAI) en cas de défaut de la batterie de l'horloge temps réel.
- **CommsAlm.** Ce bit est mis à 1 si un bloc en mémoire cache dans l'unité T932 est en alarme logiciel, en raison d'un défaut de communication (et non pas, une erreur de total de contrôle). Le signal *CommsAlm* peut donc être utilisé par un instrument de supervision T3000/T3500 pour vérifier le bon fonctionnement des communications entre T932, même lorsque les blocs affectés proprement dits ne sont pas visibles depuis le T3000/T3500. Il suffit de mettre en mémoire cache les blocs T932 qui communiquent dans le T3000/T3500 pour que leurs bits *CommsAlm* soient accessibles.
- **MemBat.** (Good/Bad - bon/mauvais). "Bad" (VRAI) en cas de défaut de batterie de secours de la mémoire.
- **Pwr Fail.** VRAI indique que le T932 a démarré à froid ou à chaud après une coupure de l'alimentation.
- **S21 Cold.** (Disable/Enable - désactiver/activer) Etat du bloc de commutateurs 2, commutateur 1 - Démarrage à froid activé. Enable = VRAI (On), Disable = FAUX (Off).
- **S22 Warm.** Etat du bloc de commutateurs 2, commutateur 2 - Démarrage à chaud activé. Enable = VRAI (On), Disable = FAUX (Off).
- **S23 Poll.** Etat du bloc de commutateurs 2, commutateur 3 - Fonction de scrutation des installations activée. Enable = VRAI (On), Disable = FAUX (Off).
- **S24.** Etat du bloc de commutateurs 2, commutateur 4 - Etat maître/esclave S6000. Enable = VRAI (On), Disable = FAUX (Off).
- **LeftCPU.** VRAI = UC dans l'alvéole gauche. FAUX = UC dans l'alvéole droite.
- **S11 à S17.** Etat du bloc de commutateurs 1, commutateurs 1 à 7 — Adresse LIN. Pour chaque commutateur, VRAI = On, FAUX = Off.

T940: BLOC DE CONFIGURATION T940

Fonction du bloc

Ce bloc est un bloc du type "en-tête" qui doit être intégré dans un schéma de boucles pour être exécuté dans un T940. Il permet d'accéder à l'horloge temps réel de l'unité de base (qui est réglée indépendamment pour chaque unité T940), de spécifier les unités de température pour un seul T940, une temporisation de démarrage à froid (réinitialisation système) et contient une alarme de baisse de tension qui est mémorisée lorsque l'alimentation est coupée pendant une durée spécifiée. Le bloc permet également de définir différents paramètres de communication/matériel, et de sélectionner différentes options utilisateur.

Paramètres du bloc

Les symboles utilisés dans la table 10-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Time, Date. Horloge temps réel et calendrier indépendants. Ces champs permettent de modifier l'heure et la date d'un T940, lorsque le système tourne "en ligne".

IP_type. Champ binaire permettant de définir la fréquence du secteur et le système/unités de température. Les spécifications s'appliquent à l'ensemble d'un T940 et affectent les tables de linéarisation internes, ainsi que la compensation de soudure froide. Les unités ne peuvent être mélangées dans un T940 individuel, mais différents T940 sur le réseau LIN/ALIN peuvent être paramétrés différemment.

- **60Hz.** Sélectionner TRUE (VRAI) pour le secteur à 60Hz ou FALSE (FAUX) pour un secteur à 50Hz.
- **Imperial.** TRUE sélectionne le système impérial — Fahrenheit (°F)/Rankine (R). FALSE sélectionne le système métrique — Celsius (°C)/Kelvin (K).
- **Absolute (Absolu).** TRUE sélectionne les unités de température absolues — Kelvin/Rankine. FALSE sélectionne les unités relatives — Celsius/Fahrenheit.

ColdStrt. Définit la durée minimale (minutes) d'une interruption de courant qui se traduira par un démarrage à froid du T940. Un "démarrage à chaud" se produit après des interruptions de courant d'une durée inférieure à *ColdStrt*; autrement dit, le schéma de boucles actif est rechargé avec la base de données de paramètres et les modes de fonctionnement existants. Après un démarrage à froid, par contre, la base de données de paramètres est entièrement réinitialisée et le schéma de boucles défini dans le fichier à l'extension .RUN est chargé et exécuté (si possible).

Lorsqu'un fichier de base de données *dbase.DBF* est téléchargé d'un noeud déporté ou chargé en utilisant le terminal local du T940, un fichier appelé *dbase.RUN* est créé, et tous les autres fichiers avec une extension .RUN sont supprimés. Au cours d'un démarrage à froid, le T940 recherche un fichier dont l'extension est .RUN dans son système d'archivage. En cas d'échec, le T940 ne lance pas sa conduite.








Paramètre	Fonction	Unités	Etat
Time	Horloge temps réel	hh:mm:ss	
Date	Date horloge temps réel	dd/mm/yy	
IP_type	Fréquence secteur, unités de température	(ABC)D hex	
60Hz	Fréquence secteur (TRUE = 60Hz, FALSE = 50Hz)	V/F — 1	D
Imperial	Système temp. (TRUE = Imperial, FALSE = SI)	V/F — 2	
Absolute	Unités temp. (TRUE=Absolues, FALSE=Relatives)	V/F — 4	
		V/F — 8	
ColdStrt	Coupure mini. pour déclencher démarrage à froid	Min	
BrownOut	Durée coupure pour déclencher alarme BrownOut	Min	
Relays	Entrées d'état des relais	(ABC)D hex	
Relais1	TRUE = relais désactivé (FALSE = par défaut)	V/F — 1	D
Relais2	TRUE = relais désactivé (FALSE = par défaut)	V/F — 2	
WdogRly	TRUE = relais désactivé (FALSE = par défaut)	V/F — 4	
		V/F — 8	
Alarms   			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
BrownOut	Coupure plus longue que BrownOut	V/F	
ComS/W	Erreur logiciel bloc commun	V/F	
PSU	Perte courant des alims A ou B	V/F	
OverTemp	Température interne excessive détectée	V/F	
ExtBat	Batterie externe absente ou défectueuse	V/F	
IntBat	Batterie interne absente ou défectueuse	V/F	
MainFan	Défaut ventilateur enceinte	V/F	
CPUFan	Défaut ventilateur UC	V/F	
Chngovr	Un basculement survenu dans un système redondant	V/F	
I/OComms	Une/plusieurs erreurs lignes communication E/S	V/F	
WdogLoom	Faisceau commande chien de garde débranché	V/F	
CPFFail	Défaut exécution fichier paramètres Coldstart	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Features	(Champ binaire non utilisé à l'heure actuelle)	ABCD hex	
Node	Adresse noeud LIN/ALIN, lue à partir de la source	(01-FE hex)	
Status	Etat communication/matériel	ABCD hex	
ColdStrt	TRUE = Bouton Restart face avant sur 'cold' OU 'hot/cold'	V/F — 1	D
HotStrt	TRUE = fBouton Restart face avant sur 'hot' OU 'hot/cold'	V/F — 2	
		V/F — 4	
		V/F — 8	
ExtBat	Batterie externe absente ou défectueuse	V/F — 1	C
IntBat	Batterie interne absente ou défectueuse	V/F — 2	
		V/F — 4	
		V/F — 8	

Table 10-10 suite...

...Table 10-10 suite

Paramètre	Fonction	Unités	Etat
Red	Machine redondante	V/F — 1	B
MainFan	Défaut ventilateur principal	V/F — 2	
CPUFan	Défaut ventilateur UC	V/F — 4	
WdogLoom	Faisceau commande chien garde débranché	V/F — 8	
Chngovr	Basculement survenu dans un système redondant	V/F — 1	A
CommsAlm	Défaut communication bloc cache	V/F — 2	
PwrFail	BdD lancée à mise sous tension (réinit. utilisateur)	V/F — 4	
TmpPFail	Comme PwrFail, auto-réinit. 2ème itération BdD	V/F — 8	
Options	Etat communications/matériel	(A)BCD hex	
CommsDis	TRUE désactive écritures entrantes champ ALIN	V/F — 1	D
Protectd	TRUE code/décode BdD si enregistrée/chargée	V/F — 2	
CONFspd	TRUE = Configurateur, FALSE = Conduite	V/F — 4	
FullSave	TRUE enregistre BdD dans fichier .DBF	V/F — 8	
PartSave	TRUE enregistre BdD dans fichier .SBF	V/F — 1	C
StallMB1	TRUE - ligne Modbus 1 arrête BdD jusqu'à interrogation	V/F — 2	
StallMB2	TRUE - ligne Modbus 2 arrête BdD jusqu'à interrogation	V/F — 4	
StallMB3	TRUE - ligne Modbus 3 arrête BdD jusqu'à interrogation	V/F — 8	
StallMB4	TRUE - ligne Modbus 4 arrête BdD jusqu'à interrogation	V/F — 1	B
StallPB1	TRUE - ligne Profibus 1 arrête BdD jusqu'à interrogation	V/F — 2	
StallPB2	TRUE - ligne Modbus 1 arrête BdD jusqu'à interrogation	V/F — 4	
		V/F — 8	
CldStPri	T940 est UC primaire démarrage à froid	V/F	
TaskRpt1 - 4	Intervalle répétition tâche requis (<i>non disponible</i>)	Sec	
TaskHalt	Paramètres arrêt tâche utilisateur (<i>non disponible</i>)	(ABC)D hex	
Task_1	TRUE arrêt tâche utilisateur 1	V/F — 1	D
Task_2	TRUE arrêt tâche utilisateur 2	V/F — 2	
Task_3	TRUE arrêt tâche utilisateur 3	V/F — 4	
Task_4	TRUE arrêt tâche utilisateur 4	V/F — 8	
TaskPri1 - 4	Priorité requise tâche utilisateur 1 - 4 (<i>non disponible</i>)	Entier	

Table 10-10 Paramètres du bloc T940

BrownOut (Baisse de tension). Définit la durée (minutes) nécessaire d'une coupure de courant pour déclencher une alarme *BrownOut*.

Relays (Relais). Les entrées affectées aux bits de ce champ contrôlent l'activation des trois relais du T940. Le bit *WdogRly* a priorité sur le paramétrage normal du relais du chien de garde (à savoir activé/fermé = 'bon fonctionnement', désactivé/ouvert = 'défaut base de données') *et ne doit être utilisé qu'à des fins de test*.

NOTA. La LED du chien de garde de la face avant n'est pas liée au paramétrage de *WdogRly*.

Alarms (Alarmes). Voir la description générale du champ Alarms à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

- **BrownOut.** Indique une panne de courant dont la durée dépasse celle définie dans le champ *BrownOut*. Cette alarme s'affiche un "événement" non acquitté, mais n'a pas d'état acquitté.
- **ComS/W.** Définit si une alarme logicielle se produit dans les blocs de la base de données en cours d'exécution.
- **PSU.** Perte de courant dans les alimentations A ou B.
- **OverTemp.** Passe à 1 si le capteur interne détecte une température suffisamment élevée pour endommager ou perturber le bon fonctionnement de l'électronique.
- **ExtBat.** Passe à 1 si la batterie externe est absente ou défectueuse.
- **IntBat.** Passe à 1 si la batterie interne rechargeable est absente ou défectueuse.
- **MainFan.** Passe à 1 si le détecteur d'arrêt du ventilateur d'enceinte signale une défaillance.
- **CPUFan.** Passe à 1 si le détecteur d'arrêt du ventilateur de l'UC signale une défaillance.
- **Chngovr.** Passe à 1 en cas de basculement du régulateur primaire dans un système redondant.
- **I/OComms.** Passe à 1 en cas de défaillance d'une ou de plusieurs unités de communication E/S.
- **WdogLoom.** Passe à 1 si le faisceau de câblage de commande du chien de garde est déconnecté — après entretien de l'unité, par exemple. *Si cette alarme est déclenchée, retournez l'unité à l'usine pour inspection.*
- **CPFfail.** Indique que le fichier de paramètres de démarrage à froid n'a pas pu être exécuté. Ce fichier contient des valeurs de paramètres à utiliser en cas de démarrage à froid.
- **Combined.** TRUE (VRAI) si une alarme est active dans le bloc. Adopte le même message d'état et le numéro de priorité que l'alarme la plus haute du bloc.

Features. (*Ce champ binaire est inutilisé à l'heure actuelle.*)

Node. L'adresse du noeud ALIN de l'instrument, lue en fonction des commutateurs d'adresse du T940 en conduite.

Status. Champ binaire indiquant l'état des communications et du matériel du T940.

- **ColdStrt.** TRUE indique que le commutateur rotatif "Restart" de la face avant du T940 est soit en position 'cold' ou 'hot/cold' (froid ou chaud/froid)
- **HotStrt.** TRUE indique que le commutateur "Restart" est soit en position 'hot' ou 'hot/cold' (chaud ou chaud/froid).

NOTA. En fonction des 4 combinaisons TRUE/FALSE possible de ces deux bits, vous pouvez inférer la position du commutateur Restart — par ex.: les deux bits TRUE indique la position 'hot/cold'.

- **ExtBat.** TRUE indique une batterie externe absente ou défectueuse.

- **IntBat.** TRUE indique une batterie interne rechargeable absente ou défectueuse.
- **Red.** TRUE si le T940 est une machine redondante.
- **MainFan.** TRUE si le ventilateur principal de l'enceinte du T940 est défectueux.
- **CPUFan.** TRUE si le ventilateur de l'UC du T940 est défectueux.
- **WdogLoom.** TRUE si le faisceau de câble de commande du chien de garde est déconnecté de l'UC (voir le paragraphe *Alarms* ci-dessus).
- **Chngovr.** TRUE s'il y a eu basculement du régulateur primaire dans un système redondant.
- **CommsAlm.** Ce bit est mis à 1 si *un* bloc en mémoire cache dans le T940 est en alarme logicielle, en raison d'une défaillance de communication (pas une erreur de total de contrôle). Le signal *CommsAlm* permet donc à un instrument de supervision de contrôler le bon fonctionnement de toutes les communications entre T940, même lorsque le superviseur ne voit pas les blocs concernés. Il suffit de mettre en mémoire cache les blocs de communication des *T940* pour rendre leurs bits *CommsAlm* accessibles.

Ce bit peut être utilisé en liaison avec l'alarme *ComS/W* pour déterminer si un défaut de communication OU une erreur de total de contrôle se sont produits.
- **PwrFail.** Passe à 1 si la base de données a été lancée à la mise sous tension (plutôt que par une demande utilisateur, par ex., par l'intermédiaire du configurateur de terminal ou d'une commande LIN/ALIN déportée). Notez que *PwrFail* peut être réinitialisé par l'utilisateur.
- **TmpPFail.** Ce bit est mis à 1 de la même manière que *PwrFail*, mais est automatiquement remis à 0 à la seconde itération de la base de données.

Options. Champ binaire définissant certaines options d'état des communications et du matériel du T940.

- **CommsDis.** Une entrée TRUE désactive les écritures entrantes du champ LIN/ALIN.
- **Protectd.** TRUE fait que les bases de données sont encryptées à l'enregistrement et décryptées au chargement. (*Cette fonction n'est pas gérée à l'heure actuelle*).
- **CONFspd.** Sélectionne la vitesse du configurateur. TRUE sélectionne le 'mode Configurateur' et FALSE le 'mode Conduite'. Le mode Configurateur fait que la régulation tourne à efficacité réduite (~85%), mais n'est pas affectée lorsque le configurateur résident est exécuté. Le mode Conduite permet à la régulation de fonctionner à efficacité maximale, lorsque le Configurateur n'est pas utilisé, mais la ralentit à ~53% à chaque fois que le configurateur est exécuté.
- **FullSave.** TRUE enregistre la base de données active dans le fichier .DBF actif.
- **PartSave.** TRUE enregistre la base de données active dans un fichier appelé *filename.SBF*, où *filename* est le nom racine du fichier de la base de données active. Notez que l'exécution de la base de données est interrompue au cours du processus d'enregistrement.

- **StallMB1 à StallMB4.** Au lancement de la base de données, TRUE fait que la ligne de communication Modbus correspondante (1-4) interroge tous les paramètres et les met à jour avant de permettre de lancement de la base de données, ce qui permet de s'assurer — par exemple — que toutes les valeurs E/S sont à jour. Ces bits sont également utilisés dans les systèmes redondants au cours du basculement, lorsque le nouveau primaire prend le relais. Ils doivent être configurés pour que le basculement redondant fonctionne correctement.
- **StallPB1 à StallPB2.** Ces bits fonctionnent comme les bits *StallMBn* (voir ci-dessus), mais avec les lignes de communication Profibus correspondantes (1-2).

CldStPri. TRUE indique que cette UC de T940 est le primaire de démarrage à froid, autrement dit qu'elle tentera de démarrer comme régulateur primaire après une panne de courant.

T800: BLOC DE CONFIGURATION T800

T2900: BLOC DE CONFIGURATION T2900

Fonction du bloc

Ces blocs "en-tête" sont identiques à l'exception du nom. L'un ou l'autre de ces blocs doivent figurer dans un schéma de boucles pour être exécutés dans un T800 ou T2900. Ils permettent d'accéder à l'horloge temps réel de l'unité de base (qui est réglée indépendamment pour chaque unité), de spécifier les unités de température pour un seul T800/2900, une temporisation de démarrage à froid (réinitialisation système) et contient une alarme de baisse de tension qui est mémorisée lorsque l'alimentation est coupée pendant une durée spécifiée. Le bloc permet également de définir différents paramètres de communication/matériel des T800/2900, et de sélectionner différentes options utilisateur.

Paramètres du bloc

Les symboles utilisés dans la table 10-11 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Ces informations complètent celles de la table 10-11.

Dbase, Block, Type. Voir les détails sur ces blocs "en-tête" à la page 2-3.

Time, Date. Horloge temps réel et calendrier indépendants. Ces champs permettent de modifier l'heure et la date d'un T800/2900, lorsque le système tourne "en ligne".

IP_type. Champ binaire permettant de définir la fréquence du secteur et le système/unités de température. Les spécifications s'appliquent à l'ensemble d'un T800/2900 et affectent les tables de linéarisation internes, ainsi que la compensation de soudure froide. Les unités ne peuvent être mélangées dans un T800/2900 individuel, mais différents T800/2900 sur le réseau LIN/ALIN peuvent être paramétrés différemment.

- **60Hz.** Sélectionner TRUE (VRAI) pour le secteur à 60Hz ou FALSE (FAUX) pour un secteur à 50Hz.
- **Imperial.** TRUE sélectionne le système impérial — Fahrenheit (°F)/Rankine (R). FALSE sélectionne le système métrique — Celsius (°C)/Kelvin (K).
- **Absolute.** TRUE sélectionne les unités de température absolues — Kelvin/Rankine. FALSE sélectionne les unités relatives — Celsius/Fahrenheit.

ColdStrt. Définit la durée minimale (minutes) d'une interruption de courant qui se traduira par un démarrage à froid du T800/2900. Un "démarrage à chaud" se produit après des interruptions de courant d'une durée inférieure à *ColdStrt*; autrement dit. le schéma de boucles actifs est rechargé avec la base de données de paramètres et les modes de fonctionnement existants. Après un démarrage à froid, par contre, la base de données de paramètres est entièrement réinitialisée et le schéma de boucle défini dans le fichier à l'extension .RUN est chargé et exécuté (si possible).

Paramètre	Fonction	Unités	Etat
Time	Heure horloge temps réel	hh:mm:ss	
Date	Date horloge temps réel	dd/mm/yy	
IP_type	Fréquence secteur, unités température	(ABC)D hex	
60Hz	Fréquence secteur (TRUE = 60Hz, FALSE = 50Hz)	V/F	1
Imperial	Système temp. (TRUE = Impérial, FALSE = SI)	V/F	2
Absolute	Unités temp. (TRUE=Absolues, FALSE=Relatives)	V/F	4
			8
ColdStrt	Coupure mini. pour déclencher démarrage à froid	Min	
BrownOut	Durée coupure pour déclencher alarme BrownOut	Min	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
BrownOut	Coupure plus longue que BrownOut	V/F	
S6000	Alarme logicielle (comm.) dans bloc S6000 (<i>non disponible</i>)	V/F	
BadBat	Tension batterie faible	V/F	
RTCinit	Horloge temps réel non initialisée	V/F	
CPFail	Erreur exécution fichier paramètres Coldstart	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Features	(<i>Champ binaire non utilisé à l'heure actuelle</i>)	ABCD hex	
Node	Adresse noeud SLIN/ALIN, lue à partir source	(01-FE hex)	
Options	Etat communication/matériel	(AB)CD hex	
CommsDis	TRUE désactive écritures ent. champs SLIN/ALIN	V/F	1
Protectd	TRUE code/décode BdD si enregistrée/chargée	V/F	2
CONFspd	TRUE=Configurateur, FALSE=Conduite	V/F	4
FullSave	TRUE enregistre BdD active dans fichier .DBF	V/F	8
PartSave	TRUE enregistre partiel. BdD dans fichier .SBF	V/F	1
BLOCKspd	TRUE accélère l'exécution des blocs de fonction	V/F	2
PANELspd	TRUE améliore la réaction de la face avant	V/F	4
		V/F	8
		V/F	1
		V/F	2
		V/F	4
		V/F	8
ClrBrOut	TRUE désactive auto-effacement alarme brownout	V/F	1
ColdStart	TRUE active démarrage à froid	V/F	2
WarmStart	TRUE active démarrage à chaud (<i>non disponible</i>)	V/F	4
HotStart	TRUE active démarrage à chaud	V/F	8

Table 10-11 suite...

Lorsqu'un fichier de base de données *dbase.DBF* est téléchargé d'un noeud déporté ou chargé en utilisant le terminal local du T800/2900, un fichier appelé *dbase.RUN* est créé, et tous les autres fichiers avec une extension *.RUN* sont supprimés. Au cours d'un démarrage à froid, le T800/2900 recherche un fichier dont l'extension est *.RUN* dans son système d'archivage. En cas d'échec, le T800/2900 ne lance pas sa conduite.

...Table 10-11 suite




Paramètre	Fonction	Unités	Etat
Status	Etat communications/matériel	ABCD hex	
		V/F — 1	D
		V/F — 2	
		V/F — 4	
		V/F — 8	
BattBad	Batterie absente ou défectueuse	V/F — 1	C
		V/F — 2	
		V/F — 4	
		V/F — 8	
		V/F — 1	B
		V/F — 2	
		V/F — 4	
		V/F — 8	
CommsAlm	Défaut comm. bloc cache (<i>non disponible</i>)	V/F — 1	A
PwrFail	BdD lancée mise sous tension (réinit. utilisateur)	V/F — 2	
TmpPFail	Comme PwrFail, auto-réinit. 2ème itération BdD	V/F — 4	
		V/F — 8	
Relays	Etat relais face arrière et sélection priorité	(C)D hex	
Wdg	TRUE = Relais 'Health' face arrière désactivé	V/F — 1	D
Run	TRUE = Relais 'Run' face arrière désactivé	V/F — 2	
		V/F — 4	
		V/F — 8	
TaskRpt1 - 4	Intervalle requis répétition tâche (<i>seul TaskRpt1 disponible</i>)	Sec	
TaskHalt	Paramètres d'arrêt tâche utilisateur (<i>non disponible</i>)	(ABC)D hex	
Task_1	TRUE arrête la tâche utilisateur 1	V/F — 1	D
Task_2	TRUE arrête la tâche utilisateur 2	V/F — 2	
Task_3	TRUE arrête la tâche utilisateur 3	V/F — 4	
Task_4	TRUE arrête la tâche utilisateur 4	V/F — 8	
TaskPri1 - 4	Priorité requise de la tâche utilisateur 1 - 4. (<i>non disponible</i>)	Entier	

Table 10-11 Paramètres du bloc T800/T2900

BrownOut (Baisse de tension). Définit la durée (minutes) nécessaire d'une coupure de courant pour déclencher une alarme *BrownOut*.

Options. Champ binaire définissant certaines options d'état des communications et du matériel du T800/2900.

- **CommsDis.** Une entrée TRUE désactive les écritures entrantes du champ LIN/ALIN.
- **Protectd.** TRUE fait que les bases de données sont encryptées à l'enregistrement et décryptées au chargement. (*Cette fonction n'est pas gérée à l'heure actuelle*).

- **CONFspd.** Sélectionne la vitesse du configurateur. TRUE sélectionne le 'mode Configurateur' et FALSE le 'mode Conduite'. Le mode Configurateur fait que la régulation tourne à efficacité réduite (~85%), mais n'est pas affectée lorsque le configurateur résident est exécuté. Le mode Conduite permet à la régulation de fonctionner à efficacité maximale, lorsque le Configurateur n'est pas utilisé, mais la ralentit à ~53% à chaque fois que le configurateur est exécuté.
- **FullSave.** TRUE enregistre la base de données active dans le fichier .DBF actif.
- **PartSave.** TRUE enregistre la base de données active dans un fichier appelé *filename.SBF*, où *filename* est le nom racine du fichier de la base de données active. Notez que l'exécution de la base de données est interrompue au cours du processus d'enregistrement.
- **BLOCKspd.** TRUE accroît les ressources UC affectées à l'exécution des blocs de fonction. Ceci améliore les temps entrées-sorties pour les systèmes Profibus, mais ralentit les temps de réponse de la face avant.
- **PANELspd.** TRUE réduit les ressources UC affectées à l'exécution des blocs de fonction (à l'inverse du paramètre ci-dessus). Ceci améliore les temps de réponse de la face avant, et sur les systèmes Modbus permet d'améliorer légèrement les performances.

NOTAS. Il n'est pas recommandé de sélectionner TRUE pour *PANELspd* pour les systèmes Profibus haute performance. Aucun effet, si TRUE est sélectionné pour *BLOCKspd* et *PANELspd*.

- **ClrBrOut.** TRUE active l'auto-effacement de l'alarme brownout.
- **ColdStart.** TRUE active le démarrage à froid.
- **WarmStart.** TRUE active le démarrage à chaud (*non disponible*).
- **HotStart.** TRUE active le démarrage à chaud.

Status. Champ binaire indiquant l'état des communications et du matériel du T800/2900.

- **BattBad.** Batterie défectueuse ou absente.
- **CommsAlm.** Ce bit est mis à 1 si un bloc en mémoire cache dans le T800/2900 est en alarme logicielle, en raison d'une défaillance de communication (pas une erreur de total de contrôle). Le signal *CommsAlm* permet donc à un instrument de supervision de contrôler le bon fonctionnement de toutes les communications entre les T800/2900, même lorsque le superviseur ne voit pas les blocs concernés. Il suffit de mettre en mémoire cache les blocs de communication des T800/2900 pour rendre leurs bits *CommsAlm* accessibles.

Ce bit peut être utilisé en liaison avec l'alarme *ComS/W* pour déterminer si un défaut de communication OU une erreur de total de contrôle se sont produits..

- **PwrFail.** Passe à 1 si la base de données a été lancée à la mise sous tension (plutôt que par une demande utilisateur, par ex., par l'intermédiaire du configurateur de terminal ou d'une commande SLIN/ALIN déportée). Notez que *PwrFail* peut être réinitialisé par l'utilisateur.
- **TmpPFail.** Ce bit est mis à 1 de la même manière que *PwrFail*, mais est automatiquement remis à 0 à la seconde itération de la base de données.

Relays. Champ binaire indiquant l'état des relais 'Health' et 'Run' (bon fonctionnement et conduite) de la face arrière de l'instrument. Les entrées reliées à *Wdg* et *Run* permettent de prendre le pas sur le fonctionnement normal de ces relais, si nécessaire — mais uniquement lorsque la base de données tourne. (Voir les détails sur le fonctionnement normal de ces relais dans le manuel produit).

TaskRpt1 - 4. Définit un intervalle de répétition pour les tâches 1 à 4. Les valeurs admissibles se situent dans la plage de 10 à 60 ms, et sont arrondies aux 10 ms les plus proches.

[Page laissée intentionnellement blanche]

Chapitre 11 BLOCS DE FONCTION HIST

HIST : BLOC HISTORIQUE

Fonction du bloc

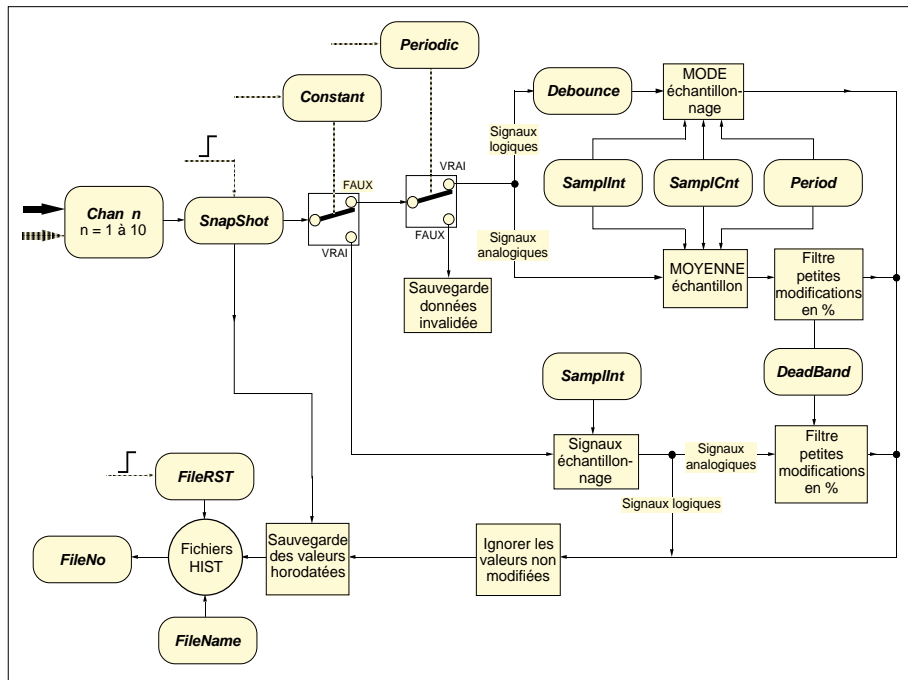


Figure 11-1 Schéma du bloc HIST

Voir le schéma de la figure 11-1. Le bloc Historique collecte les valeurs analogiques et logiques de dix points maximum dans un schéma de boucles, filtrant, traitant et comprimant les données avant de les enregistrer dans des fichiers RAM spéciaux. Les données historiques archivées peuvent être récupérées et visualisées sur une page de tendance de conduite du T1000.

Modes d'échantillonnage - Constant & Périodique

Le bloc peut être défini pour échantillonner les données de deux manières différentes. En mode Constant, les données sont échantillonnées une fois par intervalle de temps spécifié, *SampInt*, et les échantillons individuels filtrés et stockés. En mode périodique, un groupe

de ces échantillons individuels (quantité *SamplCnt*) est pris à intervalles réguliers (période) et sa valeur *moyenne* est consignée.

La figure 11-2 illustre les modes d'échantillonnage périodique et constant. Il faut noter qu'en mode périodique, seuls les points d'échantillonnage en "noir" sont traités, tandis qu'en mode constant, tous les points d'échantillonnage sont consignés (soumis au filtrage).

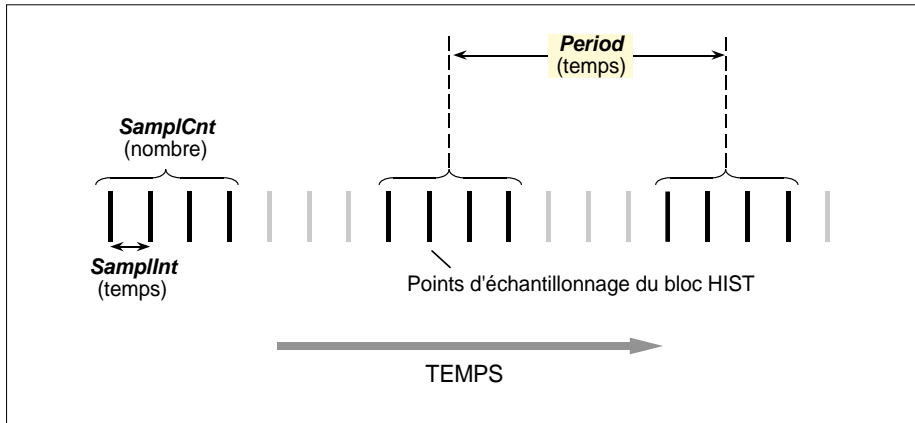


Figure 11-2 Echantillonnage du bloc HIST

Dans les deux modes d'échantillonnage, le bloc peut être déclenché à tout moment par l'intermédiaire d'une entrée sélective (*Snapshot*) pour archiver une "image instantanée" de toutes les valeurs des points à la dernière mise à jour du bloc HIST. L'image instantanée (*Snapshot*) contourne l'action de moyenne et de filtrage.

Points logiques & analogiques

Ces points sont traités différemment dans les deux modes d'échantillonnage. En mode périodique, les signaux logiques sont d'abord soumis à un "anti-rebond" pour rejeter les pointes - les transitions courtes - puis la moyenne du groupe est effectuée, ce qui pour les signaux logiques signifie qu'ils prennent l'état du mode (majoritaire). En ce qui concerne les signaux analogiques, la moyenne du groupe est effectuée (moyenne arithmétique) et le résultat soumis au filtrage de la bande morte pour rejeter les changements de faible pourcentage considérés comme du bruit. En mode constant, la moyenne n'est pas effectuée ; les échantillons logiques ne sont pas traités, mais les signaux analogiques subissent un filtrage de bande morte.

Compression des données

Pour économiser l'espace d'archivage, dans les deux modes (mais pas en mode sélectif - snapshot), une valeur qui après traitement est identique à celle archivée précédemment n'est pas archivée à nouveau. Comme les enregistrements sont horodatés, aucune donnée n'est perdue.

Fichiers historiques

Les échantillons traités et horodatés sont enregistrés dans un fichier spécial sur le disque RAM du T1000, qui peut contenir jusqu'à 4 Ko (environ 300 enregistrements). Lorsqu'il est plein, le fichier suivant est automatiquement créé et ouvert pour que la tendance puisse continuer. Un fichier peut être fermé avant d'être plein et pour créer le suivant, il suffit d'appliquer un front montant à l'entrée *FileRST*. Les fichiers historiques sont désignés par *FileName.HFileNo*, où *FileName* est la désignation racine spécifiée par le paramètre *FileName* et *FileNo* est un nombre séquentiel qui va de 00 à ZZ (base-36). Lorsque seize fichiers historiques ont été remplis, les nouveaux fichiers créés provoquent l'effacement séquentiel des fichiers historiques les plus anciens pour laisser la place aux nouveaux. (Les opérateurs sont avertis dans ce cas par une alarme "device full"). Cet ensemble - archives - de seize fichiers dans la RAM peut contenir jusqu'à 4800 enregistrements. Les opérateurs doivent s'assurer que les fichiers historiques à conserver ont été copiés sur une carte DTU avant d'être écrasés automatiquement, ce qui peut être fait au cours de la conduite grâce à une page d'archivage historique spéciale du T1000 (Page d'archivage de tendance en conduite).

NOTA. Il est recommandé d'utiliser un maximum de cinq blocs HIST par T1000, afin de minimiser l'effet de la consignation des données sur les performances de mise à jour du bloc/affichage/clavier. Il vaut également mieux *échelonner* les périodes de sauvegarder des blocs HIST, par ex. en exécutant chacun d'eux séparément par l'intermédiaire d'un bloc TIMEDATE. (Les sorties du bloc TIMEDATE peuvent être reliées aux entrées *Constant* et *Periodic* des blocs HIST pour les maintenir à FAUX, ce qui invalide la sauvegarde des données pendant les périodes appropriées).

Paramètres du bloc

Les symboles utilisés dans la table 11-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Periodic (Périodique). VRAI valide le mode d'échantillonnage périodique, dans lequel un groupe d'échantillons est pris périodiquement et dont la moyenne est effectuée. Voir les détails dans la section *Fonction du bloc* ci-dessus. Il faut noter que le paramètre *Constant* a une priorité plus élevée et qu'il doit être FAUX pour que le mode périodique puisse fonctionner (voir figure 11-1).

Constant (Constant). VRAI permet de sélectionner le mode d'échantillonnage constant qui annule le mode périodique. En mode constant, des échantillons sans moyenne sont pris à chaque intervalle d'échantillonnage, et ainsi beaucoup plus de détails sont

consignés. Lorsque périodique et constant sont FAUX, aucune donnée n'est consignée et le bloc est invalidé.










Paramètre	Fonction	Unités	Etat
Periodic	Validation mode échantillonnage périodique	V/F	
Constant	Validation mode échantillonnage constant	V/F	
SnapShot	Enregistrement instantané des entrées non filtrées	V/F	
FileRST	Déclenchement fermeture fichier historique	V/F	
FileName	Nom racine des fichiers historiques	Alphanumérique	
FileNo	Dernier fichier historique actif (00-ZZ)	Alphanumérique	
Chan 1-Chan 10	Liaisons d'entrée de données	Alphanumérique	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
FileFull	Fichier historique actif plein	V/F	
DevCFull	Disque RAM (unité M) plein	V/F	
WriteErr	Erreur écriture RAM	V/F	
ReadErr	Erreur lecture RAM	V/F	
Combined	Alarme combinée du bloc	V/F	
Period	Intervalle d'échantillonnage de la moyenne (secs.)	Entier	
SamplInt	Intervalle d'échantillonnage (secs.)	Entier	
SamplCnt	Nombre d'échantillons compris dans la moyenne	Entier	
DeadBand	% mini. de changements d'échantillonnage pour la consignation	Entier	
Debounce	Nbre de signaux logiques répétés pour un changement d'état	Entier	

Table 11-1 Paramètres du bloc HIST

Snapshot (Image instantanée). Un front montant (FAUX à transition VRAI) entré à n'importe quel moment active une copie immédiate de toutes les valeurs d'entrée active (non filtrées) dans le fichier historique. *Snapshot* fonctionne quel que soit les états des paramètres périodique, constant, et des autres paramètres de filtrage /échantillonnage.

FileRST (Remise à zéro fichier). Un front montant (FAUX à transition VRAI) entré à n'importe quel moment ferme un fichier historique (avant qu'il ne soit plein) et crée et ouvre le fichier suivant dans la séquence.

FileName (Nom de fichier). Un nom racine alphanumérique à 8 caractères pour les fichiers créés par le bloc Historique. Les fichiers tendance historique ont le format *FileName.HFileNo*. Le numéro de séquence, *FileNo*, est créé automatiquement. "H" renvoie à "Historique".

FileNo (N° de fichier). Le numéro de séquence du fichier historique actif (ou le plus récemment actif). *FileNo* est exprimé en base 36 et va de 00, 01 ...09, 0A...YZ,ZZ.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.

■ **FileFull (Fichier plein).** Définit lorsque le fichier historique actif est plein, qu'il est fermé et que le fichier suivant dans la séquence est ouvert.

- **DevcFull (Disque plein).** Mis à 1 lorsque le bloc HIST a ouvert le nombre maximum (16) de fichiers autorisés sur le disque RAM (unité M), et a commencé de remplir le dernier fichier. Lorsque celui-ci est plein, les nouveaux fichiers commencent à effacer et écraser les plus anciens. Cette alarme peut être utilisée comme un signal d'avertissement qui demande aux utilisateurs de transférer les données sur une DTU.
- **WriteErr (Erreur d'écriture).** Erreur d'écriture RAM. Défaillance du système d'archivage.
- **ReadErr (Erreur de lecture).** Erreur de lecture RAM. Défaillance du système d'archivage.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Period (Période). (mode périodique uniquement). Permet de spécifier l'intervalle de temps (secondes) entre les moyennes successives des groupes d'échantillons. La période est également l'intervalle entre les enregistrements de données dans les fichiers historiques, à moins que l'action de filtrage n'intervienne.

SamplInt (Intervalle d'échantillonnage). Permet de spécifier l'intervalle de temps entre tous les échantillons en mode constant et entre les échantillons formant la moyenne en mode périodique.

SampleCnt (Nombre d'échantillons). (mode périodique uniquement). Permet de spécifier le nombre d'échantillons utilisés pour calculer la moyenne une fois par période. Si mise à 1, la moyenne est invalidée et SamplInt est ignoré. Il faut noter que $SampleCnt \times SamplInt \leq Period$. *SampleCnt* s'applique à la fois aux signaux logiques et analogiques ; pour les signaux logiques, la "moyenne" est prise comme étant l'état (majoritaire) du mode.

DeadBand (Bande morte). Valeur analogique de filtrage ou de "bruit" (pas appliquée aux signaux logiques). Permet de spécifier la valeur (en unités de pourcentage) par laquelle un échantillon ou une moyenne d'échantillons doit différer de la dernière valeur enregistrée pour qu'elle soit enregistrée à son tour. Si le changement est inférieur à la bande morte, la valeur est considérée comme du bruit et est ignorée. Il faut noter que pour éviter les enregistrements redondants, même si la bande morte est mise à 0 %, un échantillon n'est enregistré que si sa valeur a effectivement changé.

Debounce (Anti-rebond). Valeur logique de filtrage ou de "bruit" (pas appliquée aux valeurs analogiques). Utilisée uniquement en mode périodique. Permet de spécifier le nombre d'échantillons homogènes consécutifs nécessaire pour constituer un changement d'état. Si le nouvel état persiste pendant moins d'échantillons que *Debounce*, les échantillons sont traités comme des "pointes" et sont ignorés. La figure 11-3 montre un exemple de la manière dont l'anti-rebond fonctionne sur un groupe de 14 échantillons logiques, *Debounce* étant mis à 3. L'état initial était BAS.

L'anti-rebond (*Debounce*) est appliqué avant que la moyenne des signaux logiques ne soit effectuée, de sorte que ces pointes filtrées n'affectent pas l'état enregistré. Pour économiser l'espace sur le disque, seuls les changements par rapport à l'état précédent sont envoyés dans le fichier (comme pour les signaux analogiques).

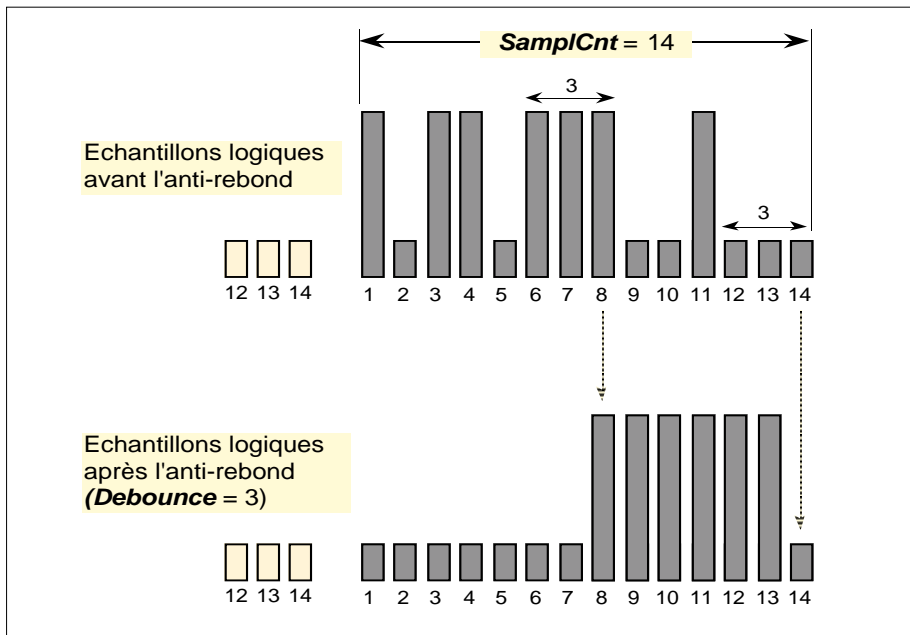


Figure 11-3 Action anti-rebond du bloc HIST - Exemple

Chan 1 à Chan 10 (Voie 1 à Voie 10). Liaisons d'entrées analogiques et/ou logiques du bloc HIST ; les points qui sont les sources des données consignées. Il faut noter que ces champs ne sont pas affichés dans le *Menu de spécifications du bloc* en conduite.

NOTA1. Dans la page de configuration des schémas de boucles du T1000 ou de LINtools, si une entrée *logique* est reliée (par l'intermédiaire de la touche de fonction WIRE - FIL), le menu de liaisons déroulant contient l'élément "TO FILE" (VERS FICHIERS). La sélection de TO FILE permet d'afficher les entrées *Chan 1 à Chan 10*. (Les liaisons aux voies de consignation de données ne sont pas visibles sur la page de conception).

NOTA2. Pour supprimer ou modifier une liaison filaire dans les voies de consignation des données du bloc HIST, interrogez (touche de fonction ??) un paramètre *Chan n* configuré dans le menu de spécifications du bloc. Un sous-menu avec les options **Change** et **Clear** s'affiche à l'écran. Sélectionnez **Change** pour modifier la liaison ou **Clear** pour la supprimer.

Chapitre 12 BLOCS DE FONCTION DIAG

La catégorie DIAG comprend les blocs de diagnostic T100/T1000 qui sont principalement destinés à l'usage d'EUROTHERM SYSTEMES ; ils peuvent donc être modifiés sans préavis. Dans ce cas, les tailles des blocs seront maintenues pour pouvoir charger les anciennes bases de données, bien que la signification de certains champs ait pu changer. Les blocs DIAG n'ont pas besoin d'être présents pour le fonctionnement correct du LIN.

LRA : BLOC ALGORITHME DE REDONDANCE LIN

Fonction du bloc

Le bloc LRA (Algorithme de redondance LIN), lorsqu'il est exécuté dans une base de données de schéma de boucles, permet de contrôler le mode de fonctionnement du réseau LIN (Réseau Local) et détermine également son état. Il faut noter que le LRA fonctionne indépendamment de la présence d'un bloc LRA, et il n'est pas nécessaire d'inclure le bloc dans un schéma de boucles pour le fonctionnement du réseau LIN en mode *automatique* (sélection par défaut), ou si le contrôle n'est pas nécessaire.

Définition d'un réseau redondant

Un réseau redondant est un groupe d'instruments qui sont tous connectés par leurs voies A et B du réseau LIN, et qui peuvent tous prendre en charge les voies LIN au niveau du matériel et logiciel. Même si l'un des instruments n'est pas connecté aux deux voies, ou ne dispose pas du matériel ou logiciel nécessaire, alors le réseau n'est *pas* un réseau redondant. (Les voies LIN sont décrites ci-dessous dans la section *Matériel LIN* et suivantes).

Mauvaise utilisation de logiciels LIN redondants sur un réseau non-redondant

Attention

Si un instrument tourne sous le logiciel V3.1 ou une version ultérieure, mais n'est pas équipé du matériel nécessaire pour supporter un réseau redondant - ou s'il est équipé du matériel, mais ne dispose pas de liaison voie B Lin - une mauvaise utilisation du bloc LRA peut provoquer la perte de communication sur ce noeud.

Les communications peuvent être perdues en forçant l'instrument à recevoir la voie B, ce qui peut se produire de plusieurs manières:

- Localement, en sélectionnant *HoldMode* (Mode Maintien) VRAI, puis le mode de fonctionnement *FORCE_B*. Dans ce cas, les communications seront perdues tant que la configuration n'est pas modifiée.

- Si un message de mode *FORCE_B/HoldMode* VRAI est reçu depuis une station active. Dans ce cas, la perte de communication persiste tant que soit l'instrument n'est pas reconfiguré localement ou est mis HORS/SOUS tension pendant qu'une configuration réalisable est diffusée (voir figure 12-3).
- Si un message de mode *FORCE_B/HoldMode* FAUX est reçu depuis une station active. Dans ce cas, la perte de communication persiste jusqu'à ce que l'instrument repasse en AUTO (c'est à dire voie A du réseau LIN) après 30 secondes environ (dépassement du temps NO_MASTER) en l'absence d'autres diffusions.

Si les diffusions en mode *FORCE_B* se poursuivent, les communications sont à nouveau perdues au cours de chaque diffusion. Sinon, le réseau rectifiera de lui-même en repassant en AUTO. C'est pour cette raison qu'il est en général conseillé de sélectionner *HoldMode* FAUX pour la station de diffusion. (Voir figure 12-2).

NOTA. Les descriptions dans les paragraphes ci-après, ainsi que les figures 12-2 et 12-3 expliquent les termes utilisés dans ATTENTION ci-dessus, et pourquoi ces pertes de communication se produisent.

Matériel LIN

Chaque station sur un réseau LIN redondant dispose de deux voies de communication - LIN A et LIN B. Le matériel est conçu pour qu'une station transmette toujours sur les deux voies LIN, mais n'en reçoivent qu'une. Le but principal de l'algorithme de redondance LIN - contrôlé par le bloc LRA - est de décider sur laquelle des deux voies il faut recevoir.

Commutation voie LIN

La commutation des voies LIN indique quelle voie, LIN A ou LIN B, est sélectionnée pour une station en réception - avec le minimum d'interférences dans les communications. Afin de réduire les interruptions sur le réseau, une station change de voie LIN uniquement lorsqu'aucune transmission n'est effectuée vers la station par l'intermédiaire de la voie qui est déconnectée. Physiquement, la commutation se fait par l'intermédiaire d'un commutateur à circuits intégrés et ne prend que quelques microsecondes. Il n'est pas nécessaire pour une station de se déconnecter du réseau LIN au cours de la commutation de voie LIN.

La voie sur laquelle une station est en réception est indiquée par la paramètre *Using* (Utilisation) du bloc LRA, qui peut indiquer LIN_A, LIN_B ou DISCONN (si les deux voies sont déconnectées).

Détection erreur LIN

Les erreurs sur le réseau LIN - des ruptures dans les câbles, des courts-circuits, des interférences radio importantes - sont détectées par un *module comparateur* matériel qui compare en permanence les signaux sur les deux voies LIN, A et B, qui devraient être identiques en situation idéale. Une erreur est considérée comme s'étant produite, si ces

signaux diffèrent pendant plus de 10 microsecondes. (Cette marge tient compte du désalignement entre les signaux sur les différentes voies en raison de longueurs de câbles différentes, par exemple). Lorsqu'une erreur se produit, des bits signalent l'erreur LIN et indiquent laquelle des voies était silencieuse (en défaut) par l'intermédiaire du paramètre correspondant - *FaultOnA* ou *FaultOnB* - du bloc LRA.

Suppression d'une erreur LIN

La détection de la disparition d'un défaut de câble est plus compliquée que la détection de son avènement. Dans la figure 12-1, James détectera la rupture sur LIN B chaque fois que Henry, Thomas ou Gordon transmettront. Mais les comparateurs dans les trois autres stations, ne détecteront le défaut que lorsque James transmettra. Donc, avant de décider qu'un défaut a été corrigé, l'algorithme doit laisser suffisamment de temps jusqu'à ce que tous les noeuds aient transmis. Normalement, ce temps est mesuré par le nombre de jetons reçus et est d'environ une seconde. Lorsque, pour une raison ou pour une autre, les jetons ne sont pas reçus, ce temps est d'environ 2 secondes.

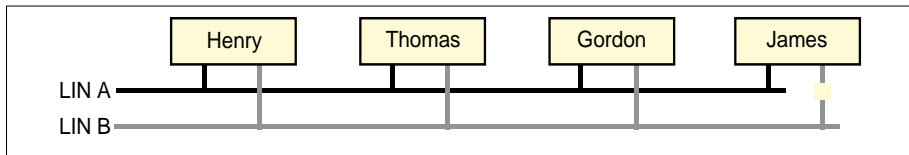


Figure 12-1 Exemple de défaut de câble LIN

Détection d'un défaut matériel

Les défauts dans l'électronique LIN interne de l'instrument - puces, composants, cartes, etc. - sont détectés par le module *auto-test*, qui utilise la capacité d'une station à se déconnecter et à "s'écouter parler". Chaque voie effectue un test automatique à la mise sous tension. Si les deux voies sont défectueuses, le test est répété périodiquement. Sinon, si elle peut transmettre et recevoir sur au moins une voie, la station se connecte sur le réseau LIN et le test automatique n'est pas répété.

Il faut noter que le test automatique est un test actif et qu'il n'est donc pas effectué lorsque le noeud est connecté au réseau LIN, ce qui interromperait les communications ailleurs sur le réseau. Les incidents de test automatique sont indiqués dans le bloc LRA par l'intermédiaire des paramètres *FailTstA* et *FailTstB*.

Modes de fonctionnement du réseau LIN

Le réseau LIN peut fonctionner en quatre modes supportés directement par l'algorithme de redondance LIN - AUTO, FORCE_A, FORCE_B ou FREEZE (GEL) - avec un bit supplémentaire de qualification appelé *HoldMode*. Un autre paramètre - le bit Actif - détermine si la station peut "activement" forcer son propre mode de fonctionnement et son état *HoldMode* sur d'autres stations LIN ou accepter "passivement" le mode diffusé par les autres. Le paramètre *Mode* du bloc LRA indique le mode de fonctionnement actif de la station.

- **AUTO.** En mode AUTO, la commutation se produit automatiquement sur les voies LIN, c'est à dire que l'algorithme décide quelle est la voie qui doit recevoir (A ou B) et connecte automatiquement la station à la voie sélectionnée. L'algorithme de commutation favorise la voie A. C'est à dire que, soit *sans* défaut de câble LIN ou en cas de défaut sur A et B, (détection par le module comparateur), la liaison se fait sur LIN A.

NOTA. Le réseau LIN fonctionne au mieux de son efficacité, lorsque tous les noeuds sont à leur état par défaut, c'est à dire Mode AUTO, *HoldMode* et *Active FAUX*. (Le paramètre *Active* est expliqué ci-dessous).

- **FORCE_A, FORCE_B.** Lorsque le mode FORCE_A (ou B) est sélectionné, l'algorithme de commutation LIN est invalidé, et LIN A (ou LIN B) est sélectionné comme la voie de réception, quels que soient les défauts détectés.
- **FREEZE (GEL).** Lorsque le mode FREEZE est sélectionné, l'algorithme de commutation LIN est invalidé, et la station continue de recevoir par l'intermédiaire de la voie LIN qui est connectée.
- **Active Bit (Bit Actif).** Les stations ayant des paramètres *Active FAUX* sont des non-diffuseurs "passifs" dont les modes de fonctionnement sont forcés de suivre le mode diffusé régulièrement par une station "active" (*Active* = TRUE - VRAI)
- **HoldMode Bit (Bit Mode Maintien).** Le but du paramètre *HoldMode* est de définir le mode auquel une station passive revient, lorsqu'aucune station ne diffuse ou si la station de diffusion est hors ligne. Si *HoldMode* est FAUX, une station passive revient (ou reste en AUTO), s'il n'y a pas eu de diffusion au cours de la temporisation NO_MASTER de 30 secondes. Si *HoldMode* est VRAI, le mode en cours est maintenu.

Les effets sur le mode de fonctionnement des paramètres *HoldMode* et *Active* à la mise sous tension de la station, et ultérieurement, sont représentés schématiquement dans les figures 12-2 et 12-3.

NOTA. La transmission LIN est expliquée ci-après dans le paragraphe *Diffusion sur le LIN*.

Configuration en fonctionnement automatique

AUTO est le mode de fonctionnement par défaut de l'algorithme de redondance LIN, même lorsqu'aucune base de données ou bloc LRA n'est exécuté. Cependant, pour contrôler l'état du réseau LIN, un bloc LRA doit être exécuté dans la base de données du schéma de boucles, FAUX étant sélectionné pour son entrée *Active*. Dans cette configuration, toutes les stations sur le réseau LIN commutent automatiquement entre LIN A et LIN B en cas de défaut - c'est à dire qu'elles fonctionnent en mode AUTO.

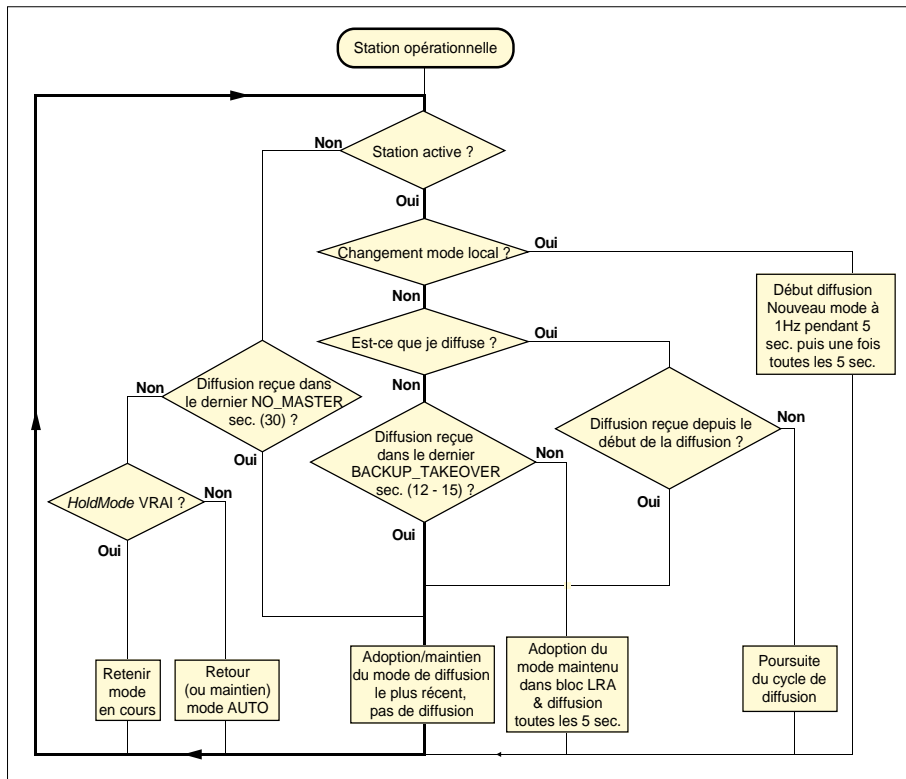


Figure 12-2 Fonctionnement de l'algorithme de redondance du LIN - Station opérationnelle

Configuration en fonctionnement forcé

Pour forcer toutes les stations sur le réseau à utiliser une voie LIN particulière, une base de données contenant un bloc LRA dont l'entrée *Active* est VRAIE devrait être exécutée. Le fonctionnement de toutes les stations peut alors être forcé en sélectionnant *FORCE_A*, *FORCE_B* ou *FREEZE* pour la station "active". Le mode actif est alors diffusé à d'autres stations pour s'assurer qu'elles adoptent le même mode.

Lorsqu'il n'y a pas de blocs LRA ayant des champs *Active* VRAI, il est possible de contrôler le fonctionnement d'une ou de stations individuelle(s). Il faut alors inclure un bloc LRA - son champ *Active* étant FAUX - dans la base de données de la station. Après avoir sélectionné VRAI pour *HoldMode*, l'écriture dans le champ *Mode* de la station permet de sélectionner laquelle des voies LIN doit être en réception. (Ces actions sont représentées schématiquement dans la figure 12-2 et expliquées plus en détail ci-après).

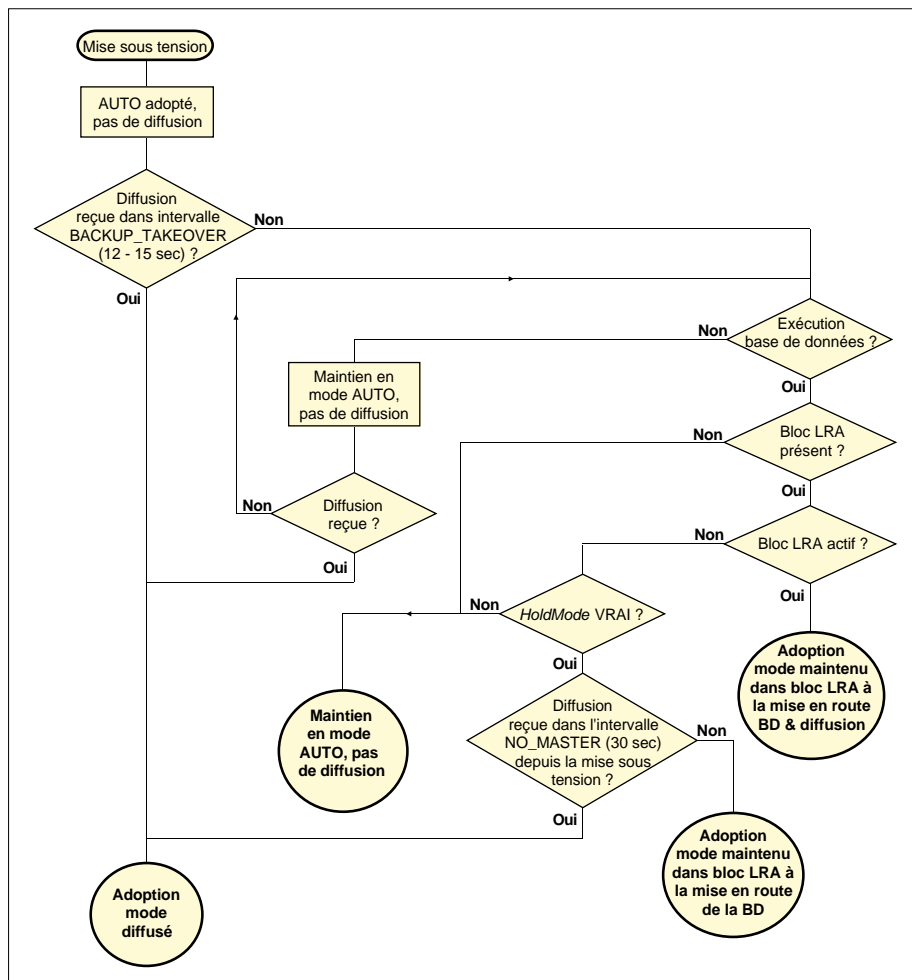


Figure 12-3 Fonctionnement de l'algorithme de redondance LIN - Mise sous tension de la station

Diffusion sur le réseau LIN

“Diffusion” renvoie aux messages contrôlés par l’algorithme de redondance LIN intégré dans chaque station, ce qui détermine le mode de fonctionnement qui doit être adopté par tous les noeuds sur le réseau LIN. Ces diffusions LRA sont différentes des données transmises sur le LIN - valeurs de paramètres, etc.

Seule une station exécutant une base de données contenant un bloc LRA dont le champ *Active* est VRAI - une station “active” - est potentiellement capable de diffuser son mode

de fonctionnement (AUTO, FORCE_A, FORCE_B ou FREEZE) et l'état *HoldMode* à toutes les autres stations sur le réseau LIN. Elle ne pourra toutefois diffuser si une autre station active la devance. Comme il ne peut y avoir de confirmation d'une diffusion, le message est répété périodiquement (toutes les cinq secondes) - ce qui permet également aux nouvelles stations sur le LIN d'adopter le mode de fonctionnement correct avec le minimum de retard.

La diffusion permet à l'opérateur d'une station de contrôler le mode de fonctionnement de l'ensemble du réseau LIN, sans qu'un grand nombre de blocs image soit nécessaire. Lorsqu'un opérateur change le mode du LIN de n'importe quelle station active, la station diffuse son nouveau mode et l'état *HoldMode* à la fréquence accélérée d'une fois par seconde pendant les cinq premières secondes, pour revenir à la fréquence de répétition normale d'une diffusion toutes les cinq secondes ; ce qui a pour effet de permettre à la station de "dicter" son nouveau mode de fonctionnement à toutes les autres stations sur le réseau LIN, qu'elles soient actives ou non. La figure 12-2 montre schématiquement comment l'algorithme procède dans ce cas.

Diffuseurs multiples

Bien qu'il puisse y avoir plusieurs stations actives sur le réseau LIN, l'algorithme ne permet qu'à une seule d'entre elles de diffuser à la fois - normalement la station où le mode a été modifié en dernier. Si cette station se met hors ligne, une autre station peut prendre la relève après un dépassement du temps imparti (BACKUP_TAKEOVER - RESERVE_RELEVE). Afin de réduire les chances que deux stations n'essaient de prendre la relève en même temps, la valeur de dépassement du temps imparti est différente pour chaque adresse de station LIN dans la plage de 12 à 15 secondes environ. La station dont la valeur de BACKUP_TAKEOVER est la plus courte est la première à "remarquer" que la diffusion s'est arrêtée, et est donc la première à diffuser le mode contenu dans son bloc LRA. Les autres stations qui sont plus lentes à répondre sont alors forcées d'adopter ce mode, et ne peuvent faire leur propres diffusions. Voir figure 12-2.

NOTA. Les temps de BACKUP_TAKEOVER sont calculés comme suit :

$$12 + (0.01 \times \text{adresse réseau [décimal]}) \text{ secondes.}$$

Par ex. le noeud à l'adresse FE hex (254 dec.) a un dépassement du temps imparti de $12 + 2.54 = 14.54$ secondes.

Actions de mise sous tension

Ces actions sont schématisées à la figure 12-3. Lorsqu'un noeud est mis sous tension, il s'initialise en mode AUTO sans diffusion. S'il reçoit alors une diffusion dans l'intervalle de son dépassement de temps imparti BACKUP_TAKEOVER, il adopte simplement le mode diffusé et l'état *HoldMode*, et continue dans ce mode jusqu'à ce que la situation change (voir figure 12-2). Ce qui se produit lorsque la station qui vient d'être mise sous tension ne reçoit pas rapidement une diffusion dépend de plusieurs facteurs, voir figure 12-3. Après la mise sous tension, la station adopte l'un des quatre états de fonctionnement illustrés dans la figure (entourés d'un *cercle* dans la figure).

Paramètres du bloc

Les symboles utilisés dans la table 12-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.















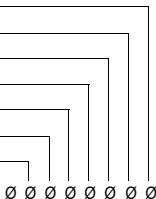
Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement actif LIN	Menu	
HoldMode	Invalidation mode reprise (stations passives)	V/F	
Active	Validation diffuseur (station active)	V/F	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
LIN_A	Incident câble ou auto-test sur voie A LIN	V/F	
LIN_B	Incident câble ou auto-test sur voie B LIN	V/F	
Combined	Alarme combinée du bloc	V/F	
Using	Voie LIN en utilisation	Menu	
FaultOnA	Défaut câble sur voie A LIN	V/F	
FaultOnB	Défaut câble sur voie B LIN	V/F	 
FailTstA	Incident auto-test sur voie A LIN	V/F	 
FailTstB	Incident auto-test sur voie B LIN	V/F	 
LINstate	Etat de fonctionnement LIN	Champ binaire	 
Auto	Mode de fonctionnement AUTO	V/F	
A_Forced	Mode de fonctionnement FORCE_A	V/F	
B_Forced	Mode de fonctionnement FORCE_B	V/F	
Frozen	Mode de fonctionnement FREEZE (GEL)	V/F	
A_Select	Voie A en réception	V/F	
Connectd	Station s'est connectée sur le LIN	V/F	
Fault	FaultOnA OU FaultOnB	V/F	

Table 12-1 Paramètres du bloc TPO

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Mode. (AUTO/FORCE_A/FORCE_B/FREEZE) Le mode de fonctionnement actif de la station sur le réseau LIN. Le mode peut être modifié localement ou par une diffusion à partir d’une autre station.

- **AUTO.** (Mode par défaut). Commutation automatique sur le réseau LIN entre les voies A et B contrôlée par l’algorithme de redondance LIN.
- **FORCE_A, FORCE_B.** La commutation LIN est invalidée, et la station est forcée en réception sur la voie A ou la voie B indépendamment des incidents détectés.
- **FREEZE (GEL).** La commutation LIN est invalidée, et la station continue de recevoir par la voie LIN connectée.

Voir les détails dans la section *Fonction du bloc*.

HoldMode (Mode maintien). Détermine le mode de reprise d'une station passive (*Active* = FAUX), lorsqu'elle ne reçoit aucune diffusion après les 30 secondes de temporisation NO_MASTER. Lorsque FAUX est sélectionné pour *HoldMode*, le mode de reprise est AUTO ; lorsque VRAI est sélectionné pour *HoldMode*, la station maintient son mode en cours. *HoldMode* peut être modifié soit localement ou par une diffusion d'une autre station.

Active (Actif). Lorsque VRAI est sélectionné pour *Active* (dans un bloc LRA en cours d'exécution), une station peut - si elle en a l'occasion - être le diffuseur sur le réseau LIN et forcer les autres stations à adopter son mode de fonctionnement. Si FAUX est sélectionné pour *Active* (ou sans bloc LRA en cours d'exécution), la station est "passive" et ne peut diffuser.

Alarms (Alarmes). Voir la description générale du champ Alarmes à la page 2-5.

- **Software (Logiciel).** Anomalie mémoire dans les données RAM du bloc.
- **LIN_A.** VRAI si la voie A du LIN souffre d'un défaut câble détecté par le module comparateur de la station, et/ou si l'auto-test échoue à la mise sous tension.
- **LIN_B.** VRAI si la voie B du LIN souffre d'un défaut câble détecté par le module comparateur de la station, et/ou si l'auto-test échoue à la mise sous tension.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Using (Utilisation). (LIN_A/LIN_B/DISCONN) Indique la voie LIN utilisée par la station (A, B ou déconnectée sur les deux voies).

FaultOnA, FaultOnB (Défaut sur A/B). VRAI indique un défaut de câble sur la voie A ou B du réseau LIN, détecté par le module comparateur.

FailTstA, FailTstB (Echec auto-test A/B). VRAI indique un défaut de l'électronique dans le matériel LIN de la voie A ou B, détecté par l'auto-test à la mise sous tension.

LINstate (Etat LIN). Champ binaire de 7 bits indiquant l'état de fonctionnement du réseau LIN de la station.

- **Auto, A_Forced, B_Forced, Frozen (Gelé).** VRAI indique le mode de fonctionnement correspondant.
- **A_Select (A sélectionné).** VRAI indique que la station est en réception sur la voie A du réseau LIN. FAUX indique que la voie B du réseau LIN est en réception ou que la station est déconnectée du LIN.
- **Connectd (Connecté).** VRAI indique la station s'est connectée sur le réseau LIN.
- **Fault (Défaut).** VRAI si soit *FaultOnA* ou *FaultOnB* est VRAI ou les deux.

LIN_DIAG : BLOC DE DIAGNOSTIC LIN

Fonction du bloc

Le bloc LIN_DIAG collecte des statistiques sur le fonctionnement du réseau local instrument (LIN). Les informations concernent la couche contrôle accès support (MAC) et la couche contrôle liaison logique (LLC). Le bloc peut être mis en image dans d'autres instruments.

Paramètres du bloc

Les symboles utilisés dans la table 12-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
MACstate	Mode de fonctionnement de la machine état MAC	Menu	
ThisNode	Adresse LIN de cette station	Hex	
NextNode	Adresse LIN de la station logique suivante	Hex	
PrevNode	Adresse LIN de la station logique précédente	Hex	
ToknHldr	Adresse du détenteur de jeton LIN (échantillonnée)	Hex	
XmtFault	Comptage défaut de transmission (pas de transmission jeton)	Entier	
Intr_sol	Horloge associée à la maintenance de l'anneau	Entier	
ToknTime	Temps de rotation du jeton (approximatif)	Entier	
Rep_pass	Acceptation d'une nouvelle station dans l'anneau logique Résolution contentieux entre station au démarrage	Entier	
Who_fol		Entier	
Rep_who		Entier	
Sol_any		Entier	
Tot_fail	Jeton perdu & comptage panne anneau logique	Entier	
Xmt_back	Pannes station à transmettre sur le comptage LIN	Entier	
Xmt_blok	Triple panne poste à transmettre sur le comptage LIN	Entier	
Xmt_Urtn	Comptage pannes logiciel à matériel	Entier	
LLCstate	Mode de fonctionnement de la machine état LLC	Menu	
SAPsfree	Nombre de SAP libres	Entier	
SAPsbusy	Nombre de SAP en utilisation	Entier	
Tx_free	Nombre de tampons de transmission de réserve	Entier	
Tx_alloc	Nombre de tampons de transmission en service	Entier	
Tx_busy	Nombre de tampons de transmission occupés	Entier	
Tx_ready	Nombre de tampons de transmission en attente	Entier	
Rx_free	Nombre de tampons de réception de réserve	Entier	
Rx_alloc	Nombre de tampons de réception en service	Entier	
Rx_busy	Nombre de tampons de réception occupés	Entier	
Rx_bufav	Nombre de tampons de réception disponibles	Entier	
Rx_ready	Nombre de tampons de réception prêts	Entier	

Table 12-2 Paramètres du bloc LIN_DIAG

Menu de spécifications du bloc

Les informations suivantes sont données en plus de la table 12-2.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

MACstate. (OFFLINE/IDLE/DEMAND_IN/DEMAND_DL/CLAIM_TOK/AWAIT_IFM/CHK_TOK_P/AWAIT_RES). Mode de fonctionnement actif du contrôle accès support (MAC).

XmtFault. Défaut de transmission. Est incrémenté chaque fois que le poste ne peut transmettre le jeton - peut-être qu’il s’agit de la seule station sur l’anneau.

Tot_fail. Est incrémenté chaque fois que le jeton est perdu et que l’anneau logique est défaillant. Ne devrait pas s’incrémenter au cours du fonctionnement normal.

Xmt_back. Est incrémenté chaque fois que le poste ne transmet pas sur le réseau. Peut être incrémenté au cours du fonctionnement normal.

Xmt_blok. Est incrémenté chaque fois que le poste ne transmet pas sur le réseau trois fois de suite. Ne devrait pas s’incrémenter au cours du fonctionnement normal.

Xmt_Urun. Est incrémenté chaque fois qu’il y a une panne entre le logiciel et le matériel dans l’unité. Ne devrait pas s’incrémenter au cours du fonctionnement normal.

LLCstate. (INIT/START/UP/STOP/DOWN/ABORT). Mode de fonctionnement actif du contrôle liaison logique (LLC).

SAPsbusy. Nombre de services d’accès aux points en réserve (SAP). Egal à 1 pour SAPzéro, + 1 pour le service (SAP) connu de l’extérieur, + 1 pour chaque base de données déportée.

Tx_alloc. Nombre de tampons de transmission en service. Egal à 1 pour SAP + 1 pour chaque base de données déportée.

Rx_alloc. Nombre de tampons de reception en service. Egal à 1 pour SAP + 1 pour chaque base de données déportée.

UCTUNE : BLOC PERFORMANCES DU T1000


Fonction du bloc

Le bloc UCTUNE (“réglage de l’unité de régulation”) permet de contrôler l’utilisation UC des processeurs principal et de communication, et les performances des trois serveurs de blocs - Local (S1), Image (S2) et S6000 (S3).

La moitié gauche du menu de spécifications affiché indique les données du processeur principal, et le côté droit les alarmes et données du processeur de communication (IOP). Les temps du cycle du serveur mesurent la période entre les exécutions consécutives du premier bloc sur chaque liste de mise à jour. Les chiffres de l’utilisation UC représentent des parties approximatives de temps passé pour chaque tâche, exprimées sur une échelle de 0 à 1000, c’est à dire x 0,1 %. Les tâches en haut de liste ont la priorité la plus basse et celles en bas de liste la plus haute.

Paramètres du bloc

Les symboles utilisés dans la table 12-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
Bgnd	Utilisation tâches de fond (en %)		
Config	Utilisation tâches configurateur/affichage conduite (en %)		
Trend	Utilisation logiciel tendance (en %)		
AltTask	Utilisation tâches alarmes (Page d’alarmes, barre message, Acq.) (en %)		
S6000Srv	Serveur pour les instruments S6000 RS422, Utilisation (en %)		
CacheSrv	Serveur pour les blocs image, utilisation (en %)		
LocalSrv	Serveur pour les blocs locaux, utilisation (en %)		
TickTask	Mise à jour horloge & utilisation tâche contrôle clavier (en %)		
IOPSlave	TAN & utilisation tâche interface communications S6000 (en %)		
NetSlave	Utilisation tâche interface communications LIN (en %)		
S6000_B	Temps de cycle serveur bloc S6000 (ms)		
Cached_B	Temps de cycle serveur bloc image (ms)		
Local_B	Temps de cycle serveur bloc local (ms)		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
IOP_Bgnd	Utilisation tâches de fond IOP (en %)		
IOP_Tan	Utilisation tâche TAN IOP (en %)		
IOP_LLC	Utilisation tâche bas niveau communications LIN IOP (en %)		
IOP_Net	Utilisation tâche haut niveau communications LIN IOP (en %)		
IOP_ComS	Utilisation tâche communications S6000 IOP (en %)		
IOP_RX	Utilisation tâche bas niveau réception LIN IOP (en %)		

* Tous les paramètres ont un format entier et un statut de lecture uniquement au cours de la conduite.

Table 12-3 Paramètres du bloc UCTUNE

MDTUNE : BLOC PERFORMANCES DU T100




Fonction du bloc

Le bloc MDTUNE contrôle les performances des trois serveurs de blocs dans le T100 - E/S (S1), Local (S2) et Image (S3) - et également le temps utilisé par chaque tâche dans le système.

La moitié gauche de menu de spécifications du bloc affiché montre les données du serveur. Une 'période' est la durée du cycle du serveur, c'est à dire, le temps entre deux exécutions du premier bloc sur la liste de mise à jour. Une période n'est jamais inférieure à 100 ms et peut atteindre 1000 ms, lorsque les schémas de boucles sont très importants. La moitié droite du menu de spécifications montre les données des tâches ; les proportions sont exprimées sur une échelle de 0 à 1000, c'est à dire x 0,1 %. Les tâches en haut de la liste ont la priorité la plus basse et celles en bas de liste la priorité la plus haute.

Paramètres du bloc

Les symboles utilisés dans la table 12-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
S1used	Activité serveur (S1) bloc E/S (ms par période)		
S1period	Durée du dernier cycle S1 (ms)		
S1delay	Suspension forcée S1 (ms par période)		
S2used	Activité serveur (S2) bloc local (ms par période)		
S2period	Durée du dernier cycle S2 (ms)		
S2delay	Suspension forcée S2 (ms par période)		
S3used	Activité serveur (S3) blocs image (ms par période)		
S3period	Durée du dernier cycle S3 (ms)		
S3delay	Suspension forcée S3 (ms par période)		
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Bgnd	Utilisation tâches de fond (en ‰)		
Config	Utilisation tâches configurateur (en ‰)		
Load	Utilisation tâche chargement base de données (en ‰)		
Sumcheck	Utilisation tâche contrôle des données (en ‰)		
CacheSrv	Serveur pour les blocs image, utilisation (en ‰)		
LocalSrv	Serveur pour les blocs locaux, utilisation (en ‰)		
BatLoad	Utilisation tâche chargement BATCH/RECORD (en ‰)		
Network	Utilisation tâche réseau (en ‰)		
LLC	Utilisation tâche liaison contrôle logique (en ‰)		
RX	Utilisation tâche traitement tampons réception LIN (en ‰)		

* Tous les paramètres ont un format entier et un statut de lecture uniquement au cours de la conduite.

Table 12-4 Paramètres du bloc MDTUNE

Menu de spécifications du bloc

Les informations suivantes sont données en plus de la table 12-4. Pour plus de détails, voir également les paramètres correspondants du bloc UCTUNE (à la section précédente).

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Config. Temps utilisé par la tâche Configurateur (interface au terminal).

Sumcheck. Temps utilisé par la tâche contrôle de données qui vérifie la cohérence de la base de données et met à jour le sommaire d’alarmes.

BatLoad. Temps utilisé par la tâche chargement batch/record (lot/enregistrement)

LLC. Temps utilisé par la tâche contrôle de la liaison logique, qui est responsable des fonctions réseau de niveau bas.




DB_DIAG : BLOC DIAGNOSTIC DE LA BASE DE DONNEES

Fonction du bloc

Le bloc DB_DIAG indique le niveau de ressource *utilisé* par la base de données active (dans la moitié gauche du menu de spécifications) et également les niveaux de ressource maximum *autorisés* par la version du logiciel (dans moitié droite). Noter que les valeurs affichées des paramètres ne sont valables qu'au cours de la conduite dans l'instrument cible.

Paramètres du bloc

Les symboles utilisés dans la table 12-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
Block	Blocs utilisés dans la base de données en service		
Tmpl	Modèles utilisés dans base de données en service		
Libs	Bibliothèques de modèles utilisées		
Ent†	Tables de noms externes en service		
Edb	Bases de données externes utilisées (càd T100/T1000 image)		
Latt†	Connexions locales utilisées		
Featt	Nbre de blocs dans base de données locale mis en image ailleurs		
Teatt	Nbre de blocs bases de données déportées mis en image localement		
Sert	Tâches serveur actives		
Connect	Nbre de connexions champ à champ		
DBSize	Taille de la base de données active en octets		
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Hardware	Défauts module E/S/Alim. transmetteur	V/F	
MaxBlock	Nbre maximum de blocs dans la base de données		
MaxTmpl	Nbre maximum de modèles dans la base de données		
MaxLibs	Nbre maximum de bibliothèques de modèles		
MaxEnt†	Nbre maxi. de tables de noms externes (<i>pas T100/T1000</i>)		
MaxEdb	Nbre maxi. de bases de données externes		
MaxLatt†	Nbre maximum de connexions locales (<i>pas T100/T1000</i>)		
MaxFeatt	Nbre maxi. de blocs dans base de données locale en image ailleurs		
MaxTeatt	Nbre maxi. de blocs bases données déportées en image localement		
MaxSert	Nbre maximum de tâches serveur		
MaxConn	Nbre maxi. de connexions champ à champ		
MaxDBSiz	Taille maximale de la base de données en octets		

* Tous les paramètres ont un statut de lecture uniquement au cours de la conduite.

† Ent et Latt ne sont pas utilisés à l'heure actuelle dans les T100 ou T1000, et donc
MaxEnt et MaxLatt ont des valeurs de zéro.

Table 12-5 Paramètres du bloc DB_DIAG

Menu de spécifications du bloc

Les informations suivantes sont données en plus de la table 12-5.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Featt. Depuis les connexions externes. Nombre de blocs dans la base de données locale qui sont en image ailleurs - comptent comme un pour chaque copie du bloc local.

Teatt. Vers les connexions externes. Nombre de blocs dans la base de données déportée qui sont en image dans la base de données locale.

LIN_DEXT : BLOC D'EXTENSION DIAGNOSTIC HAUT NIVEAU LIN

Fonction du bloc


Le bloc LIN_DEXT recueille des statistiques de haut niveau (application) sur le fonctionnement du réseau local (LIN) - ce qui complète l'activité bas niveau du bloc LIN_DIAG.

Le menu de spécifications du bloc est divisé en deux parties principales : les compteurs des messages de service dans la colonne de gauche, et les compteurs de messages de la base de données à droite (plus certains paramètres généraux). La partie supérieure de chaque colonne compte les messages entrants et la partie inférieure les messages sortants. Les compteurs peuvent être remis à zéro ou suspendus par l'intermédiaire du champ *Mode*.

Le réseau local (LIN) utilise deux types de messages :

- *messages non-confirmés*, qui exigent une réponse immédiate du noeud récepteur sans autre acquittement. (Les messages non-confirmés sont marqués d'une épée (†) dans la table 12-6 ci-après.)
- *messages confirmés*, qui sont acquittés immédiatement par le noeud récepteur. Un autre traitement a alors lieu avant qu'une confirmation avec des informations supplémentaires ne soit envoyée.

Paramètres du bloc

Paramètre	Fonction	Unités	Etat*
Sparen	Affichage ThisNode, NextNode, PrevNode	Hex	
wkCReqIn	Réception des demandes de confirmations du service	Entier	
wkCRspln	Réception des confirmations du service	Entier	
wkRjctIn	Réception des rejets du service	Entier	
wkUReqIn	Réception des demandes non-confirmées du service	Entier	
wkUknwIn	Réception des messages non-reconnus du service	Entier	
wkConfirm	Nombre de confirmations générales envoyées	Entier	
wkEDBCRq	Envoi demandes de liaison à la BD externe	Entier	
wkFileRq	Envoi demandes système archivage déporté	Entier	
wkIdntRq	Envoi demandes identification déportée	Entier	
wkStStRq	Envoi demandes BD déportée lancement/arrêt	Entier	
wkLoadRq	Envoi demandes chargement BD déportée	Entier	
wkSaveRq	Envoi demandes sauvegarde BD déportée	Entier	
wkLRBcst	Nbre de diffusions LRA envoyées	Entier	
wkNRBcst	Nbre de diffusions tables de routage envoyées	Entier	
wkTxOK	Envoi avec succès du total des transmissions	Entier	
wkTxFail	Envoi sans succès du total des transmissions	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Mode	Mode de fonctionnement du bloc	Menu	
dbCReqIn	Reception demandes de confirmation du service BD	Entier	
dbCRspln	Reception confirmations du service BD	Entier	
dbRjctIn	Reception rejets du service BD	Entier	
dbUReqIn	Reception demandes non-confirmées du service BD	Entier	
dbUknwIn	Reception messages non-reconnus du service BD	Entier	
dbConfirm	Envoi messages de confirmation du service BD	Entier	
dbDiscRq	Envoi demandes déconnexion BD	Entier	
dbUpdtRq†	Envoi messages de mise à jour bloc image	Entier	
dbVrfyRq	Envoi demandes de vérification BD	Entier	
dbAtchRq	Envoi demandes de rattachement bloc image	Entier	
dbChAtRq	Envoi demandes de modification rattachement bloc image	Entier	
dbWFldRq†	Envoi messages de champ déporté écriture (liaison mise à jour)	Entier	
dbAAImRq	Envoi messages d'alarme acquittée déportée	Entier	
dbWAlmRq	Envoi messages d'alarme écriture déportée	Entier	
dbHeartB†	Envoi pulsations base de données	Entier	
dbTxOK	Envoi avec succès total des transmissions BD	Entier	
dbTxFail	Envoi sans succès total des transmissions BD	Entier	

* Tous les paramètres (Mode excepté) ont un statut de lecture uniquement au cours de la conduite.

† Messages non-confirmés.

Table 12-6 Paramètres du blocLIN_DEXT

EDB_DIAG : BLOC DIAGNOSTIC DE LA BASE DE DONNEES EXTERNE

Fonction du bloc

Le bloc EDB_DIAG affiche des informations sur les liaisons aux bases de données externes exécutées dans les instruments déportés. Une base de données déportée est contrôlée à la fois, sélectionnée par l'intermédiaire du champ *EDBindex*. Les entrées des bases de données déportées peuvent être examinées pour vérifier que le nom et l'adresse sont comme prévus et que la communication est établie. Le bloc contrôle également l'algorithme de mise au point de la période de mise à jour du bloc image.

Paramètres du bloc

Les symboles utilisés dans la table 12-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Les informations suivantes sont données en plus de la table 12-7.

My ID, His ID. Les identificateurs utilisés par un poste pour suivre l'ordre des messages et les réponses aux demandes confirmées. Ces valeurs devraient s'incrémenter pour chaque message reçu ou envoyé.

ErrRate, TxCount, TxBias. Donnent des informations sur l'algorithme de mise au point de la période de mise à jour de l'état du bloc. *ErrRate* donne une mesure du taux d'erreur de transmission vers le noeud déporté.

Paramètre	Fonction	Unités	Etat*
EDBIndex	Entrée dans table EDB - Echelle 1 à MaxEdb (bloc DB_DIAG)	Hex	
NodeNo	Adresse du noeud de l'instrument déporté	Hex	
NodeName	Nom de la base de données de l'instrument déporté	Alphanumérique	
RetrySta	Numéro état relance liaison EBD	Entier	
RetryTim	Dépassement temps imparti relance liaison EBD (tops)	Entier	
DSap	Point d'accès service du noeud déporté	Hex	
LSap	Point d'accès service du noeud local	Hex	
TimeStmp	Heure du dernier message envoyé	Entier	
Status	Drapeaux état EDB	ABCD hex	
Status0	Etat EDB active	V/F — 1	D
Status1	Etat EDB active	V/F — 2	
Status2	Etat EDB active	V/F — 4	
Spare		V/F — 8	
Spare		V/F — 1	C
Spare		V/F — 2	
Spare		V/F — 4	
Spare		V/F — 8	
1st_time	EDB connectée à la première tentative	V/F — 1	B
Remote	Entrée EDB créée par un noeud déporté	V/F — 2	
T_alarm	Alarme Teatt (Teatt sans Featt déporté correspondant)	V/F — 4	
F_alarm	Featt en alarme	V/F — 8	
Verify	Serveur de réseau vérifie Featt	V/F — 1	A
Sent_OK	Dernière modification envoyée avec succès	V/F — 2	
Spare		V/F — 4	
Spare		V/F — 8	
My ID	Identificateur message local	Hexadécimal	
His ID	Identificateur message déporté	Hexadécimal	
ErrRate	Comptage erreurs de transmission vers EDB active (Nbre d'erreurs dans dernières transmissions TxCount)	Entier	
TxCount	Nbre transmissions depuis dernier ajustement de TxBias	Entier	
TxBias	Décalage dépassement temps imparti transmission (calculé à partir de ErrRate, TxCount; ajusté pour donner 0.5-1.0 % taux d'erreur objectif)	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 12-7 Paramètres du bloc EDB_DIAG





DTU_DIAG : BLOC DE DIAGNOSTIC DTU

Fonction du bloc

Ce bloc est principalement destiné à être utilisé par Eurotherm Systèmes. Il présente des statistiques de diagnostic sur les performances des deux unités de transfert de données (DTU).

Paramètres du bloc

Les symboles utilisés dans la table 12-8 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat*
Init	Initialisation du bloc	V/F	
StatusA	Etat du lecteur A DTU	Menu	
StatusB	Etat du lecteur B DTU	Menu	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
HRCOUNTA		Entier	
HRErrorA		Entier	
RRCOUNTA		Entier	
RRErrorA		Entier	
HWCOUNTA		Entier	
HWEErrorA		Entier	
RWCOUNTA		Entier	
RWEErrorA		Entier	
OWCOUNTA		Entier	
OWEErrorA		Entier	
HRCOUNTB		Entier	
HREErrorB		Entier	
RRCOUNTB		Entier	
RREErrorB		Entier	
HWCOUNTB		Entier	
HWEErrorB		Entier	
RWCOUNTB		Entier	
RWEErrorB		Entier	
OWCOUNTB		Entier	
OWEErrorB		Entier	

*Lecture uniquement sauf indication contraire

Table 12-8 Paramètres du bloc DTU_DIAG

T231TUNE : BLOC PERFORMANCES DU T231


Fonction du bloc

Le bloc T231TUNE contrôle les performances des serveurs de blocs dans le T231 - E/S, Local et Image - ainsi que le pourcentage de temps utilisé par chaque tâche dans le système.

La moitié gauche du menu de spécifications du bloc affiché montre les données serveur exprimées en millisecondes par “période”. Une période est la durée du cycle du serveur, c’est à dire la durée entre des exécutions consécutives du premier bloc sur la liste de mise à jour. La durée n’est jamais inférieure à 100 ms et peut atteindre 1000 ms environ pour des stratégies très importantes. Le côté droit du menu de spécifications montre les données des tâches; les proportions sont exprimées sur une échelle de 0 à 1000, c’est à dire x 0,1%. Les tâches du haut de la liste ont la priorité la plus basse, et celles du bas de la liste, la priorité la plus haute.

Paramètres du bloc

Les symboles utilisés dans la table 12-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
S6000_B	Durée du cycle du serveur bloc S6000 (ms)		
Cached_B	Durée du cycle du serveur bloc image (ms)		
Local_B	Durée du cycle du serveur bloc local (ms)		
I/O_B	Durée du cycle du serveur bloc E/S (ms)		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Bgnd	Utilisation tâche de fond (en %)		
Config	Utilisation tâche configurateur (en %)		
Load	Utilisation tâche chargement base de données (en %)		
Sumcheck	Utilisation tâche contrôle par totalisation (en %)		
S6000Srv	Serveur de blocs pour les blocs S6000, utilisation (en %)		
CacheSrv	Serveur de blocs pour les blocs image, utilisation (en %)		
LocalSrv	Serveur de blocs pour les blocs locaux, utilisation (en %)		
S6000Scn	Utilisation tâche échantillonnage communicat. S6000 (en %)		
Network	Utilisation tâche réseau (en %)		
LLC	Utilisation tâche contrôle liaison logique (en %)		
RX	Utilisation tâche traitement tampon réception LIN (en %)		
NFS	Utilisation tâche du serveur fichiers réseau (en %)		
BatLoad	Utilisation tâche chargement BATCH/enregistrement (en %)		

* Tous les paramètres ont un format entier et sont en état de lecture uniquement au cours de la conduite.

Table 12-9 Paramètres du bloc T231TUNE

Menu de spécifications du bloc

Les informations suivantes sont données en plus de celles de la table 12-9.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Config. Pourcentage de temps utilisé par la tâche configurateur (interface à terminal).

Sumcheck. Pourcentage de temps utilisé par la tâche contrôle par totalisation, qui vérifie la cohérence de la base de données et met à jour le sommaire d’alarmes.

BatLoad. Pourcentage de temps utilisé par la tâche chargement des lots/enregistrements.

LLC. Pourcentage de temps utilisé par la tâche contrôle des liaisons logiques qui est responsable des fonctions de réseau de niveau bas.

NFS. Pourcentage de temps utilisé par la tâche serveur de fichiers réseau, qui est responsable du traitement des demandes de copies de fichiers sur le réseau.

S6_DIAG : BLOC DIAGNOSTIC S6000

Fonction du bloc

Le bloc S6_DIAG contient des informations de configuration et de diagnostic sur le pilote de communication binaire asynchrone (RS422).


Paramètres du bloc

Les symboles utilisés dans la table 12-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans le paragraphe ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Line	Sélection du n° de ligne auquel les BSC sont rattachées	Entier	
Speed	Sélection de la vitesse en bauds	Menu	
Parity	Sélection type de contrôle de parité	Menu	
DataBits	Sélection du nombre de bits de données par caractère	Menu	
StopBits	Sélection du nombre de bits d'arrêt par caractère	Menu	
Timeout	Durée d'attente tâche pour message complet (80-65535 ms)		
Retries	Nbre maxi. de relances transaction INO (1-10)	Entier	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Config	Pilote communicat. ne peut se connecter à tous les INO	V/F	
Combined	Alarme combinée du bloc	V/F	
PrtYErr	Nbre erreurs de parité détectées par pilote communicat. †	Entier	
Overrun	Nbre erreurs de dépassemt. détectées par pilote communicat. †		Entier
Framing	Nbre erreurs de cadrage détectées par pilote communicat. †		Entier
TimedOut	Nbre dépassemt. temps imparti messages détectées par pilote de communicat. †	Entier	
NAK	Nbre réponses NAK détectées par pilote de communicat. †	Entier	
CAN	Nbre réponses CAN détectées par pilote de communicat. †	Entier	
BCC	Nbre réponses mauvais octet BCC reçues †	Entier	
GDnEnbl (n=0-7)	Etat échantillonnage communications INO†	ABCD hex	
UID0	INO(16*n + 0) échantillonné par le réseau	V/F — 1	D
UID1	INO(16*n + 1) échantillonné par le réseau	V/F — 2	
UID2	INO(16*n + 2) échantillonné par le réseau	V/F — 4	
UID3	INO(16*n + 3) échantillonné par le réseau	V/F — 8	
UID4	INO(16*n + 4) échantillonné par le réseau	V/F — 1	C
UID5	INO(16*n + 5) échantillonné par le réseau	V/F — 2	
UID6	INO(16*n + 6) échantillonné par le réseau	V/F — 4	
UID7	INO(16*n + 7) échantillonné par le réseau	V/F — 8	
UID8	INO(16*n + 8) échantillonné par le réseau	V/F — 1	B
UID9	INO(16*n + 9) échantillonné par le réseau	V/F — 2	
UID10	INO(16*n + 10) échantillonné par le réseau	V/F — 4	
UID11	INO(16*n + 11) échantillonné par le réseau	V/F — 8	

suite...

...suite

Paramètre	Fonction	Unités	Etat
UID12	INO($16 \cdot n + 12$) échantillonné par le réseau	V/F — 1	A
UID13	INO($16 \cdot n + 13$) échantillonné par le réseau	V/F — 2	
UID14	INO($16 \cdot n + 14$) échantillonné par le réseau	V/F — 4	
UID15	INO($16 \cdot n + 15$) échantillonné par le réseau	V/F — 8	
GIDnAlm ($n=0-7$) Etat hors ligne communicat. INO		ABCD hex	
UID0	INO($16 \cdot n + 0$) hors ligne (erreur MAJ communicat.)	V/F — 1	D
UID1	INO($16 \cdot n + 1$) hors ligne (erreur MAJ communicat.)	V/F — 2	
UID2	INO($16 \cdot n + 2$) hors ligne (erreur MAJ communicat.)	V/F — 4	
UID3	INO($16 \cdot n + 3$) hors ligne (erreur MAJ communicat.)	V/F — 8	
UID4	INO($16 \cdot n + 4$) hors ligne (erreur MAJ communicat.)	V/F — 1	C
UID5	INO($16 \cdot n + 5$) hors ligne (erreur MAJ communicat.)	V/F — 2	
UID6	INO($16 \cdot n + 6$) hors ligne (erreur MAJ communicat.)	V/F — 4	
UID7	INO($16 \cdot n + 7$) hors ligne (erreur MAJ communicat.)	V/F — 8	
UID8	INO($16 \cdot n + 8$) hors ligne (erreur MAJ communicat.)	V/F — 1	B
UID9	INO($16 \cdot n + 9$) hors ligne (erreur MAJ communicat.)	V/F — 2	
UID10	INO($16 \cdot n + 10$) hors ligne (erreur MAJ communicat.)	V/F — 4	
UID11	INO($16 \cdot n + 11$) hors ligne (erreur MAJ communicat.)	V/F — 8	
UID12	INO($16 \cdot n + 12$) hors ligne (erreur MAJ communicat.)	V/F — 1	A
UID13	INO($16 \cdot n + 13$) hors ligne (erreur MAJ communicat.)	V/F — 2	
UID14	INO($16 \cdot n + 14$) hors ligne (erreur MAJ communicat.)	V/F — 4	
UID15	INO($16 \cdot n + 15$) hors ligne (erreur MAJ communicat.)	V/F — 8	

* Ecriture uniquement au moment de la configuration

† Peut être remis à zéro au cours de la conduite

Table 12-10 Paramètres du bloc S6_DIAG

Menu de spécifications du bloc

Les informations suivantes sont données en plus de celles de la table 12-10.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Line (Ligne). Permet de sélectionner le numéro de ligne auquel les communications binaires synchrones (BSC) sont rattaché. Line n’est pas configurable que ce soit en conduite ou au moment de la configuration, et n’est donné qu’à titre d’information.

Speed (Vitesse). (9600/4800/2400/1200/600/300). Sélecteur de la vitesse en bauds des communications (valeur par défaut 9600). En lecture uniquement au cours de la conduite.

Parity (Parité). (Even/Odd/None - Paire/Impaire/Aucune). Permet de sélection le type de contrôle de parité nécessaire sur le réseau (parité paire par défaut). En lecture uniquement au cours de la conduite.

DataBits (Bits de données). (8/7). Permet de sélectionner le nombre de bits de données dans chaque caractère (8 par défaut). En lecture uniquement au cours de la conduite.

StopBits (Bits d'arrêt). (1/1.5/2). Permet de sélectionner le nombre de bits d'arrêt pour un caractère (1 par défaut). En lecture uniquement au cours de la conduite.

Timeout (Dépassement temps imparti). Permet de spécifier le dépassement de temps imparti que la tâche de communication applique pendant l'attente d'un message complet. La plage est de 80 à 65535 millisecondes (100 ms par défaut).

Retries (Relances). Permet de spécifier le nombre de relances que le pilote de communication tentera pour une transaction sur un INO avant de déclarer l'INO hors ligne. La plage est de 1 à 5 (5 par défaut). Modifiable au cours de la configuration et de la conduite.

GID0Enbl à GID7Enbl. Huit champs binaires de 16 bits, chaque bit correspondant à un UID dans le GID en question. Les bits - *UID0* à *UID15* - passent à VRAI si l'INO correspondant est échantillonné par la communication. FAUX dans les autres cas. Pour *GIDgEnbl*, le INO correspondant au bit *UIDu* est donné par:

$$\text{INO} = 16xg + u$$

CAN. Permet de compter le nombre de réponses CAN détectées par le pilote de communication. Une réponse CAN correspond au 6255 ou 6445 informant le pilote de communication que l'instrument sur la ligne des installations est hors ligne. Peut être remis à zéro au cours de la conduite.

BCC. Permet de compter le nombre de réponses reçues avec un mauvais octet BCC. Peut être remis à zéro au cours de la conduite.

GID0Alm à GID7Alm. Huit champs binaires à 16 bits, chaque bit correspondant à un UID dans le GID en question. Les bits *UID0* à *UID15* - passent à VRAI si l'INO correspondant est hors ligne en raison d'une erreur de mise à jour des communications, le bit passe à FAUX dans les autres cas. Le bit passe également à FAUX pour les INO qui ne sont pas associés à un bloc. Pour *GIDgAlm*, l'INO correspondant au bit *UIDu* est donné par:

$$\text{INO} = 16xg + u$$

ALINDIAG : BLOC DIAGNOSTIC MAC/LCC ALIN

Fonction du bloc

Ce bloc fournit des informations sur les performances bas niveau MAC et LCC ALIN.

Paramètres du bloc

Les symboles utilisés dans la table 12-11 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
MACstate	Etat MAC puce Arcnet		
ThisNode	Adresse MAC ALIN de ce noeud		
NextNode	Adresse MAC du noeud suivant (peut être inconnu)		
PrevNode	Adresse MAC du noeud précédent (peut être inconnu)		
DiagStat	Etat diagnostic puce Arcnet		
TENT_ID	Pas valable en fonctionnement normal		
EXC_NAK	Mise à 1 indique trop de NAK		
Token	Mise à 1 indique qu'un jeton a été reçu depuis un autre noeud		
RCV_ACT	Mise à 1 indique une activité récepteur		
DUP-ID	Pas valable en fonctionnement normal		
MyRecon	Mise à 1 indique que ce noeud a provoqué la dernière reconfiguration Arcnet		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
LLCstate	Mode de fonctionnement de la machine état LLC	Menu	
SAPsfree	Nombre de points d'accès de maintenance libres (SAP)	Entier	
SAPsbusy	Nombre de SAP utilisés	Entier	
Tx_free	Nombre de tampons de transmission libres	Entier	
Tx_alloc	Nombre de tampons de transmission utilisés	Entier	
Tx_busy	Nombre de tampons de transmission occupés	Entier	
Tx_ready	Nombre de tampons de transmission en attente	Entier	
Rx_free	Nombre de tampons de réception libres	Entier	
Rx_alloc	Nombre de tampons de réception utilisés	Entier	
Rx_busy	Nombre de tampons de réception occupés	Entier	
Rx_bufav	Nombre de tampons de réception disponibles	Entier	
Rx_ready	Nombre de tampons de réception prêts	Entier	

Table 12-11 Paramètres du bloc ALINDIAG

EDB_TBL : BLOC TABLE DE LA BASE DE DONNEES EXTERNE

Fonction du bloc

Ce bloc récapitule toutes les bases de données externes utilisées.

Paramètres du bloc

Les symboles utilisés dans la table 12-12 sont expliqués au chapitre 1.






Paramètre	Fonction	Etat
Page	Sélection d'un groupe de 10 bases de données externes	
Noden (<i>n</i> = 1-10)	N° de noeud - segment, MAC de la base de données externe	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
ENamen (<i>n</i> = 1-10)	Nom correspondant de la base de données externe	

Table 12-12 Paramètres du bloc EDB_TBL

T600TUNE : BLOC PERFORMANCES DU T600

Fonction du bloc

Le bloc T600TUNE contrôle les performances du serveur de blocs pour les instruments de la série T600 et le pourcentage de temps utilisé par chaque tâche dans le système. Les données serveur sont indiquées en millisecondes par période (durée du cycle du serveur). Les données tâche sont exprimées sur une échelle de 0 à 1000, c'est à dire x 0,1 %.

Paramètres du bloc

Les symboles utilisés dans la table 12-13 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
T1used	Durée d'exécution tâche utilisateur 1	ms	
T1period	Période de répétition tâche utilisateur 1	ms	
T2used	Durée d'exécution tâche utilisateur 2	ms	
T2period	Période de répétition tâche utilisateur 2	ms	
T3used	Durée d'exécution tâche utilisateur 3	ms	
T3period	Période de répétition tâche utilisateur 3	ms	
T4used	Durée d'exécution tâche utilisateur 4	ms	
T4period	Période de répétition tâche utilisateur 4	ms	
CacheUse	Durée d'exécution blocs image	ms	
CachePer	Période de répétition blocs image	ms	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Bgnd	Utilisation UC tâche de fond	0,1%	
Scan	Utilisation UC tâche échantillonnage	0,1%	
Modbus	Utilisation UC tâche Modbus	0,1%	
NFS	Utilisation UC tâche NFS	0,1%	
Load	Utilisation UC tâche chargement	0,1%	
LLC	Utilisation UC tâche LLC	0,1%	
CacheSrv	Utilisation UC tâche serveur blocs image	0,1%	
UserT4	Utilisation UC tâche utilisateur 4	0,1%	
UserT3	Utilisation UC tâche utilisateur 3	0,1%	
UserT2	Utilisation UC tâche utilisateur 2	0,1%	
UserT1	Utilisation UC tâche utilisateur 1	0,1%	
Fpanel	Utilisation UC tâche face avant	0,1%	
Network	Utilisation UC tâche réseau	0,1%	
BinModRx	Utilisation UC tâche communications binaires	0,1%	
RX	Utilisation UC tâche réception réseau	0,1%	

Table 12-13 Paramètres du bloc T600TUNE

La nature prioritaire des tâches utilisateur et la signification des champs "utilisé" et "période" sont illustrés par la figure 12-14

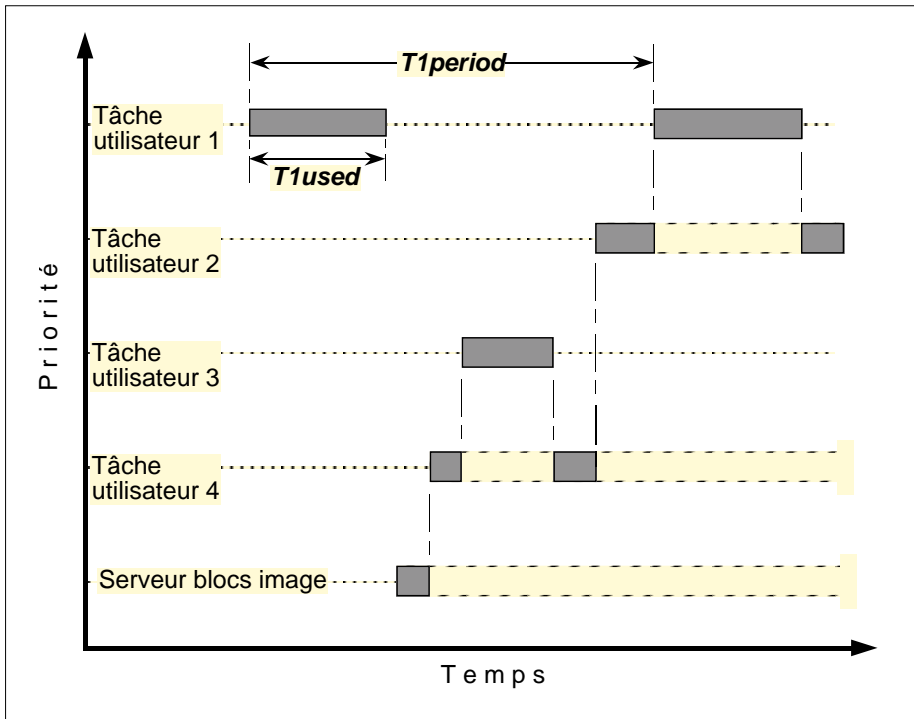


Figure 12-4 Priorités des tâches utilisateur dans le T600

ROUTETABL : BLOC TABLE D'ACHEMINEMENT

Fonction du bloc

Ce bloc permet d'afficher la table d'acheminement.

Paramètres du bloc

Les symboles utilisés dans la table 12-14 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.






Paramètre	Fonction	Etat
Page	Sélection d'un groupe de 16 entrées de table d'acheminement	
SegCost n ($n=0-F$)	Numéro de segment de l'entrée de la tache d'acheminement, coût pour arriver au segment	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
PortMAC n ($n=0-F$)	Numéro de port de la table d'acheminement, adresse de destination MAC	

Table 12-14 Paramètres du bloc ROUTETBL

Menu de spécifications du bloc

Les informations suivantes sont données en plus de celles de la table 12-14.

SegCost0 à SegCostF. Permet de montrer le numéro de segment de l'entrée de la table d'acheminement (8 bits hauts) et le coût pour arriver à ce segment (8 bits bas).

PortMAC0 à PortMACF. Les 4 bits supérieurs sont mis à 1, s'il s'agit d'une entrée de sauvegarde, c'est à dire qu'il y a une autre entrée pour le même segment utilisé. Les 4 bits suivants montre le numéro de port de l'entrée de la table d'acheminement (0=LIN, 1=ALIN sur passerelle) et les derniers 8 bits donnent l'adresse MAC où il faut envoyer les messages pour ce segment (FF=entrée permanente).

RTB_DIAG : BLOC DIAGNOSTICS TABLE D'ACHEMINEMENT

Fonction du bloc

Ce bloc permet d’afficher des informations de diagnostic sur la table d’acheminement.

Paramètres du bloc

Les symboles utilisés dans la table 12-15 sont expliqués au chapitre 1.




Paramètre	Fonction	Etat
Entries	Nombre d’entrées de table d’acheminement utilisées	
Search	Durée de recherche moyenne pour les consultations	
SearchF	Nombre de comparaisons sur échec recherche	
Lookup	Nombre de consultations	
LookupF	Nombre d’échecs en consultation	
Delete	Nombre de suppressions	
DeleteF	Nombre d’échecs de suppression	
Add	Nombre d’additions	
AddF	Nombre d’échecs d’addition	
Garbage	Nombre de récupérations de place	
Reclaim	Nombre d’entrées récupérées	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
Searchx	Index des recherches par durée de recherche	
SearchFx	Index des échecs de recherche par durée de recherche	

Table 12-15 Paramètres du bloc RTB_DIAG

ISB_DIAG : BLOC PERFORMANCES DU BUS SERIE INTERNE

Fonction du bloc

Le but de ce bloc est de permettre l'analyse des performances de l'ISB (bus série interne).

Paramètres du bloc

Les symboles utilisés dans la table 12-16 sont expliqués au chapitre 1.








Paramètre	Fonction	Unités	Etat
TimeSlot	Comptage des erreurs de dépassement des tranches de temps		
Usage n ($n=1-16$)	Trafic ISB sur ce noeud	%	
Spare	Largeur de bande ISB inutilisé	%	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Errors n ($n=1-16$)	Erreurs sur ce noeud	comptage	
Mode	AUTO/INIT/HOLD	Menu	

Table 12-16 Paramètres du bloc ISB_DIAG

NOTA. Les totalisateurs d'erreur fonctionnent en octets (c'est à dire que la valeur de comptage maximale est de 255) et reviennent ensuite à zéro.

NOTA. *Mode* permet de geler (HOLD - MAINTIEN) les valeurs Usage & Errors (Utilisation & Erreurs) ou de remettre à zéro (INIT).

ISB_DEXT : BLOC D’EXTENSION DE DIAGNOSTIC ISB

Fonction du bloc

Ce bloc est une extension aux diagnostics fournis par le bloc ISB_DIAG, et permet de décomposer les erreurs ISB en type d’erreur plutôt qu’en erreurs par noeud comme dans le bloc ISB_DIAG.

Paramètres du bloc

Les symboles utilisés dans la table 12-17 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
TimeSlot	Comptage des erreurs de dépassement tranches de temps ISB		
Timeout	Comptage des erreurs de dépassement temps imparti		
Parity	Comptage des erreurs de parité en réception		
Framing	Comptage des erreurs d’encadrement en réception		
Overrun	Comptage des erreurs de dépassement ISB		
DIC	Comptage des erreurs DIC en réception		
Address	Comptage des erreurs d’adresse en réception		
Sequence	Comptage des erreurs de séquence en réception		
MRB_NAK	Comptage des réponses NAK		
MRB_CAN	Comptage des réponses CAN		
ACK	Comptage des réponses ACK incorrectes		
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	
Mode	AUTO/INIT/HOLD	Menu	
FreeRAM	RAM sauvegarde de données ISB non utilisée		

Table 12-17 Paramètres du bloc ISB_DEXT

NOTA. Les totalisateurs d’erreur fonctionnent en octets, et ont donc une valeur de comptage maximale de 255, ils reviennent ensuite à zéro.

NOTA. *Mode* permet de geler (HOLD - MAINTIEN) les valeurs d’erreurs ou de remettre à zéro (INIT).

T221TUNE : BLOC DE MISE AU POINT T221

Fonction du bloc

Ce bloc donne des informations sur les performances de la passerelle T221.

Paramètres du bloc

Les symboles utilisés dans la table 12-18 sont expliqués au chapitre 1.





















Paramètre	Fonction	Etat
LocalUse	Temps utilisé dans le dernier cycle par la tâche serveur de blocs locaux	
LocalPer	Période de répétition de la tâche serveur de blocs locaux	
CacheUse	Temps utilisé dans le dernier cycle par la tâche serveur de blocs image	
CachePer	Période de répétition de la tâche serveur de blocs image	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
Bgnd	Temps UC utilisé par la tâche de fond	
Main	Temps UC utilisé par la tâche configurateur de terminaux	
PollPlnt	Temps UC utilisé par la tâche scrutation des installations	
NFS	Temps UC utilisé par la tâche système d'archivage du réseau	
Load	Temps UC utilisé par la tâche chargement de la base de données	
Sumcheck	Temps UC utilisé par la tâche contrôle par totalisation de la base de données	
CacheSrv	Temps UC utilisé par la tâche serveurs de blocs image	
LocalSrv	Temps UC utilisé par la tâche serveur de blocs locaux	
ALIN_LLC	Temps UC utilisé par la tâche LLC ALIN	
LIN_LLC	Temps UC utilisé par la tâche LLC LIN	
Network	Temps UC utilisé par la tâche application réseau (système d'archivage, base de données déportée, service général, acheminement)	
ALI_N_RX	Temps UC utilisé par la tâche réception ALIN	
LIN_RX	Temps UC utilisé par la tâche réception LIN	

Table 12-18 Paramètres du bloc T221TUNE

NET_DIAG : BLOC DIAGNOSTIC RESEAU

Fonction du bloc

Ce bloc récapitule le mode de fonctionnement d'un réseau multi-segments.

Paramètres du bloc

Les symboles utilisés dans la table 12-19 sont expliqués au chapitre 1.

Paramètre	Fonction	Etat
LNode	Adresse du noeud LIN	
LMode	Mode de fonctionnement du LIN (Complet/Compatible)	
LFault	Défaut LIN	
LDupSeg	Plus d'un n° de segment défini sur la câble LIN (mode complet)	
LSpltSeg	Plus d'un câble défini le même n° de segment	
LSister	Au moins 1 passerelle redondante soeur	
LoListen	Adresse d'écoute basse pour les instruments anciens	
HiListen	Adresse d'écoute haute pour les instruments anciens	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
ANode	Adresse du noeud ALIN	
AMode	Mode de fonctionnement ALIN (Complet/Compatible)	
AFault	Défaut ALIN	
ADupSeg	Plus d'un n° de segment défini sur la câble ALIN (mode complet)	
ASpltSeg	Plus d'un câble défini le même n° de segment	
ASister	Au moins 1 passerelle redondante soeur	

Table 12-19 Paramètres du bloc NET_DIAG

FWD_LOG : BLOC CONSIGNATION D'ACHEMINEMENT

Fonction du bloc

Ce bloc affiche les informations sur le résultat des tentatives d'acheminement de paquets.

Paramètres du bloc

Les symboles utilisés dans la table 12-20 sont expliqués au chapitre 1.






Paramètre	Fonction	Etat
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
<i>Pour les transmissions sur le réseau LIN:</i>		
Ltx_ok	Transmission ok	
Lno_resp	Aucun encadrement de réponse reçu	
Lacm_err	Jeton volé	
Lresp_lo	Encadrement de réponse perdu	
Lsetup	Tampon transmission dans mauvais segment	
Lllc_nav	LLC en défaut	
Ltx_full	Il ne reste aucune entrée dans l'alignement TX_CONTRL_T	
Ltx_busy	Transmission déjà en cours à partir de ce SAP vers destination donnée	
Lbuf_nav	Il ne reste pas de tampons de transmission MAC	
Loffline	Le port n'achemine pas	
Lbuf_len	Le tampon est trop long pour être transmis	
Lsap_all	Le SAP utilisé n'est pas encore affecté	
Llin_nav	Le LIN demandé n'est pas installé physiquement	
Lmac_nav	Le MAC demandé n'est pas installé physiquement	
Lsap_ill	Le SAP utilisé est incorrect	
Lrem_llc	Le LLC sur le noeud déporté n'a pas accepté la transmission	
<i>Pour les transmissions sur le réseau ALIN:</i>		
Atx_ok	Transmission ok	
Ano_resp	Aucun encadrement de réponse reçu	
Aacm_err	Jeton volé	
Aresp_lo	Encadrement de réponse perdu	
Asetup	Tampon transmission dans mauvais segment	
Allc_nav	LLC en défaut	
Atx_full	Il ne reste aucune entrée dans l'alignement TX_CONTRL_T	
Atx_busy	Transmission déjà en cours à partir de ce SAP vers destination donnée	
Abuf_nav	Il ne reste pas de tampons de transmission MAC	
Aoffline	Le port n'achemine pas	
Abuf_len	Le tampon est trop long pour être transmis	
Asap_all	Le SAP utilisé n'est pas encore affecté	
Alin_nav	Le LIN demandé n'est pas installé physiquement	
Amac_nav	Le MAC demandé n'est pas installé physiquement	
Asap_ill	Le SAP utilisé est incorrect	
Arem_llc	Le LLC sur le noeud déporté n'a pas accepté la transmission	

Table 12-20 Paramètres du bloc FWD_LOG

CON_TBL : BLOC TABLE DES LIAISONS

Fonction du bloc

Ce bloc affiche un récapitulatif des liaisons des fichiers ou des bases de données utilisées sur une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-21 sont expliqués au chapitre 1.






Paramètre	Fonction	Etat
Page	Sélection d'un groupe de 16 entrées de table d'acheminement	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
SNodeX	Adresse du noeud origine de la liaison (source)	
DNodeX	Adresse du noeud destiné à recevoir la liaison (dest.)	

Table 12-21 Paramètres du bloc CON_TBL

CON_DIAG : BLOC DIAGNOSTIC DES LIAISONS

Fonction du bloc

Ce bloc permet d'afficher des statistiques sur les liaisons d'une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-22 sont expliqués au chapitre 1.




Paramètre	Fonction	Etat
Entries	Nombre d'entrées de la table de liaison en utilisation	
Search	Durée de recherche moyenne pour les consultations	
SearchF	Nombre de comparaisons sur les échecs en recherche	
Lookup	Nombre de consultations	
LookupF	Nombre d'échecs de consultation	
Delete	Nombre de suppressions	
DeleteF	Nombre d'échecs de suppression	
DeleteB	Nombre de suppressions de connexions orphelines	
Add	Nombre d'additions	
AddF	Nombre d'échecs d'addition	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
Searchx	Index des recherches par durée de recherche	
SearchFx	Index des échecs de recherche par durée de recherche	

Table 12-22 Paramètres du bloc CON_DIAG














CON_ENT : BLOC ENTREE LIAISONS

Fonction du bloc

Ce bloc affiche des informations sur une liaison particulière sur une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-23 sont expliqués au chapitre 1.

Paramètre	Fonction	Etat
Index	Sélection d'une entrée de table de liaison particulière	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
<i>Pour la source de la liaison:</i>		
ONode	Adresse du noeud source	
OName	Nom de la base de données ou "***Fichier***" du noeud source	
OPort	Port du noeud source	
OStatus	Etat liaison	
SevError	Mise à 1 sur erreur grave	
NoHops	Comptage connexion directe entre 2 noeuds épuisé	
LostMsg	Mise à 1 sur message perdu	
ChngHops	Mise à 1 si comptage connexion directe entre 2 noeuds a changé	
UsedSAP	Mise à 1 si un SAP particulier a été interdit sur une liaison de répétition	
HasBuff	Cette liaison a un tampon valable	
HasHand	Cette liaison a un module LLC valable	
OHops	Nbre de connexions entre noeuds de cette passerelle au noeud source	
OLMAC	Adresse locale MAC sur le port source	
OLSAP	SAP local sur port source	
ORMAC	Adresse déportée MAC sur le port source	
ORSAP	SAP déporté sur port source	
OFwdCnt	Comptage des messages acheminés vers le noeud source	
ORtryCnt	Comptage des transmissions relancées des messages vers noeud source	
OFailCnt	Comptage des échecs de transmission des messages vers noeud source	
OTstamp	Horodatage de la dernière transmission vers noeud source	
<i>Pour la destination des liaisons:</i>		
TNode	Adresse du noeud source	
TName	Nom de la base de données ou "***Fichier***" du noeud source	
TPort	Port du noeud source	
TStatus	Etat liaison	
SevError	Mise à 1 sur erreur grave	
NoHops	Comptage connexion directe entre 2 noeuds épuisé	
LostMsg	Mise à 1 sur message perdu	
ChngHops	Mise à 1 si comptage connexion directe entre 2 noeuds a changé	
UsedSAP	Mise à 1 si un SAP particulier a été interdit sur une liaison de répétition	
HasBuff	Cette liaison a un tampon valable	
HasHand	Cette liaison a un module LLC valable	

suite...

...suite






Paramètre	Fonction	Etat
THops	Nbre de connexions entre noeuds de cette passerelle au noeud source	
TLMAC	Adresse locale MAC sur le port source	
TLSAP	SAP local sur port source	
TRMAC	Adresse déportée MAC sur le port source	
TRSAP	SAP déporté sur port source	
TFwdCnt	Comptage des messages acheminés vers le noeud source	
TRtryCnt	Comptage des transmissions relancées des messages vers noeud source	
TFailCnt	Comptage des échecs de transmission des messages vers noeud source	
TTstamp	Horodatage de la dernière transmission vers noeud source	

Table 12-23 Paramètres du bloc CON_ENT

POL_TBL : BLOC TABLE D’INTERROGATION

Fonction du bloc

Ce bloc affiche la table d’interrogation des installations sur une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-24 sont expliqués au chapitre 1.






Paramètre	Fonction	Etat
Page	Sélection d’un groupe de 10 entrées de la table d’interrogation des installations	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
Nodex	Adresse instrument: port (8 bits hauts) (0=LIN, 1=ALIN pour passerelle)	
	Adresse MAC (8 bits bas)	
Namex	Nom de base de données ou *Arrêt* si l’instrument ne fonctionne pas	

Table 12-24 Paramètres du bloc POL_TBL














POL_DIAG : BLOC DIAGNOSTIC DES INTERROGATIONS DES INSTALLATIONS

Fonction du bloc

Ce bloc permet d'afficher des statistiques sur le processus d'interrogation des installations sur une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-25 sont expliqués au chapitre 1.

Paramètre	Fonction	Etat
LStatus	Etat de l'interrogation LIN	
Loop	Chaîne corrompue de noeuds	
Changed	Un noeud est apparu ou a disparu depuis le dernier échantillonnage	
NodeLost	Un noeud a disparu depuis le dernier échantillonnage	
NodeGain	Un noeud est apparu depuis le dernier échantillonnage	
FastScan	Echantillonnage rapide (quelque chose a changé au cours du dernier échantillonnage)	
Stopped	Arrêt de l'exécution d'une base de données par un noeud depuis le dernier échantillonnage	
Started	Début d'exécution d'une base de données par un noeud depuis le dernier échantillonnage	
LHiNode	Adresse la plus haute du noeud sur le LIN	
LLoNode	Adresse la plus basse du noeud sur le LIN	
LPollLen	Nombre d'interrogations requises pour une interrogation complète sur le LIN	
LINcnt	Nombre de noeuds sur le LIN	
LINold	Nombre de noeuds de type ancien sur le LIN	
LINnew	Nombre de noeuds de type nouveau sur le LIN	
LINrun	Nombre de noeuds exécutés sur le LIN	
LINhalt	Nombre de noeuds arrêtés sur le LIN	
AStatus	Etat de l'interrogation ALIN	
Loop	Chaîne corrompue de noeuds	
Changed	Un noeud est apparu ou a disparu depuis le dernier échantillonnage	
NodeLost	Un noeud a disparu depuis le dernier échantillonnage	
NodeGain	Un noeud est apparu depuis le dernier échantillonnage	
FastScan	Echantillonnage rapide (quelque chose a changé au cours du dernier échantillonnage)	
Stopped	Arrêt de l'exécution d'une base de données par un noeud depuis le dernier échantillonnage	
Started	Début d'exécution d'une base de données par un noeud depuis le dernier échantillonnage	
AHiNode	Adresse la plus haute du noeud sur le ALIN	
ALoNode	Adresse la plus basse du noeud sur le ALIN	
APollLen	Nombre d'interrogations requises pour une interrogation complète sur le ALIN	

suite...

...suite

Paramètre	Fonction	Etat
ALINcnt	Nombre de noeuds sur le ALIN	
ALINold	Nombre de noeuds de type ancien sur le ALIN	
ALINnew	Nombre de noeuds de type nouveau sur le ALIN	
ALINrun	Nombre de noeuds exécutés sur le ALIN	
ALINhalt	Nombre de noeuds arrêtés sur le ALIN	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
PollAddr	Adresse active de l'interrogation	
PollPort	Port actif de l'interrogation	
FailCnt	Comptage du nombre d'échecs d'interrogation	
OKCnt	Comptage du nombre d'interrogations réussies	
TotalCnt	Comptage du nombre total d'interrogations	
AddCnt	Comptage du nombre d'entrées ajoutées	
DelCnt	Comptage du nombre d'entrées supprimées	
PassCnt	Nombre total de passes d'interrogation des installations	

Table 12-25 Paramètres du bloc POL_DIAG

POL_ENT : BLOC ENTREE DES INTERROGATIONS DES INSTALLATIONS

Fonction du bloc

Ce bloc affiche des informations sur une entrée particulière dans la table d'interrogation des installations sur une passerelle.

Paramètre du bloc

Les symboles utilisés dans la table 12-26 sont expliqués au chapitre 1.









Paramètre	Fonction	Etat
MACAddr	Adresse MAC de l'instrument interrogé	
Port	Port sélectionné	
Type	Type d'instrument trouvé à cette adresse	
Version	Numéro de version indiqué par l'instrument	
DBName	Nom de la base de données indiqué par l'instrument	
Status	Etat de la base de données (Exécutée/Arrêtée)	
LastErr	Dernière erreur signalée par l'instrument (pas pris en compte sur tous les instruments)	
Alarms		
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
Search	Passe à VRAI pour la recherche des entrées et en trouver une valable	

Table 12-26 Paramètres du bloc POL_ENT

NODE_MAP : BLOC PROTOCOLE NOEUD LIN

Fonction du bloc

Ce bloc affiche des informations sur le protocole supporté par chaque noeud sur le réseau LIN (mais ne s'applique pas au réseau ALIN).

Paramètres du bloc

Les symboles utilisés dans la table 12-27 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
Node _{nx} (n=0-F)	Adresse du groupe d'instruments sur le LIN	ABCD hex	
n0	Protocole pris en charge par noeud n0 (LIN/XLIN)	V/F	D
n1	Protocole pris en charge par noeud n1 (LIN/XLIN)	V/F	
n2	Protocole pris en charge par noeud n2 (LIN/XLIN)	V/F	
n3	Protocole pris en charge par noeud n3 (LIN/XLIN)	V/F	
n4	Protocole pris en charge par noeud n4 (LIN/XLIN)	V/F	C
n5	Protocole pris en charge par noeud n5 (LIN/XLIN)	V/F	
n6	Protocole pris en charge par noeud n6 (LIN/XLIN)	V/F	
n7	Protocole pris en charge par noeud n7 (LIN/XLIN)	V/F	
n8	Protocole pris en charge par noeud n8 (LIN/XLIN)	V/F	B
n9	Protocole pris en charge par noeud n9 (LIN/XLIN)	V/F	
nA	Protocole pris en charge par noeud nA (LIN/XLIN)	V/F	
nB	Protocole pris en charge par noeud nB (LIN/XLIN)	V/F	
nC	Protocole pris en charge par noeud nC (LIN/XLIN)	V/F	A
nD	Protocole pris en charge par noeud nD (LIN/XLIN)	V/F	
nE	Protocole pris en charge par noeud nE (LIN/XLIN)	V/F	
nF	Protocole pris en charge par noeud nF (LIN/XLIN)	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 2-27 Paramètres du bloc NODE_MAP

Protocoles pris en charge

Les protocoles pris en charge à l'heure actuelle par les différents instruments sont les suivants:

Instrument	Révision	Protocole
T100/T1000	Jusqu'à 3/x	LIN
T231	Jusqu'à 4/1	LIN
PCLIN	Jusqu'à 4/x	LIN
QLIN	Jusqu'à 2p0x	LIN
T221	De 1/1	XLIN
Tous les instruments ALIN	De 1/1	XLIN

FWD_DIAG : BLOC STATISTIQUES D'ACHEMINEMENT

Fonction du bloc

Ce bloc permet d'afficher des statistiques sur le processus d'acheminement sur une passerelle.

Paramètres du bloc

Les symboles utilisés dans la table 12-28 sont expliqués au chapitre 1.






























Paramètre	Fonction	Etat
Requests	Nombre de demandes de connexion traitées	
Response	Nombre de réponses de connexion	
R EDBConn	Nombre de demandes de connexion EDB	
FileConn	Nombre de demandes de connexion de fichiers	
OthrConn	Nombre de demandes de connexion inconnues (devrait être 0)	
LINUFwd	Nombre d'acheminements basés sur des non-connexions (LIN)	
LINApp	Nombre de messages appl acheminés avec succès (LIN)	
LINRtry	Nombre de relances effectuées (LIN)	
LINFail	Nombre de pertes complètes de messages, défaut transmission (LIN)	
LINRate	Vitesse des mises à jour (LIN)	
ALINUFwd	Nombre d'acheminements basés sur des non-connexions (ALIN)	
ALINApp	Nombre de messages application acheminés avec succès (ALIN)	
ALINRtry	Nombre de relances effectuées (ALIN)	
ALINFail	Nombre de pertes complètes de messages, défaut transmission (ALIN)	
ALINRate	Vitesse des mises à jour (ALIN)	
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
CTimeout	Nombre de chemins de connexion en dépassement temps imparti	
CRouteF	Nombre d'échec d'acheminement de demandes de connexion	
CLoop	Tentatives d'acheminement par le même port	
CCreateF	Echec dans création d'entité de connexion	
NoSAPs	Nombre d'échecs de demandes SAP	
NoBufs	Nombre d'échecs de demandes tampon	
ExtraRsp	Nombre de réponses sans connexion	
TimeStmp	Horodatage pour la dernière passe	
PassTime	Durée d'une passe	
Period	Période	
HoldOff	Nombre de suspensions effectuées	

Table 12-28 Paramètres du bloc FWD_DIAG

BCS_DIAG : BLOC DIAGNOSTIC DE DIFFUSION DU ROUTAGE

Fonction du bloc

Ce bloc permet d'afficher des statistiques sur le processus de diffusion du routage.

Paramètres du bloc

Les symboles utilisés dans la table 12-29 sont expliqués au chapitre 1.






Paramètre	Fonction	Etat
Alarms		  
Software	Données corrompues/Défaut de communication	
Combined	Alarme combinée du bloc	
<i>Pour le port LIN:</i>		
LSMTime	Base de temps pour mesurer le comptage de la passerelle soeur	
LTimeDue	Durée avant la diffusion suivante	
LTxTime	Horodatage pour la diffusion de transmission	
LTxSlip	Glissement sur diffusion transmission	
LTxPrd	Période pour la diffusion de transmission	
LPhTime	Temps entre ma diffusion et la diffusion soeur suivante	
LPhErr	Erreur sur la dernière phase mesurée	
LBcstCnt	Comptage diffusion du routage	
LSRxCnt	Nombre de diffusions de routage soeur reçues	
LRxCnt	Nombre total de diffusions de routage reçues	
LSMCnt	Nombre de périodes de mesure	
LSCnt	Nombre de passerelles soeur	
LSRxFnd	Nombre de soeurs trouvées jusqu'à présent	
LSMode	En mode soeur (booléen)	
LMyPhase	J'étais le dernier noeud soeur à transmettre (booléen)	
LSplSeg	Segment divisé détecté	
<i>Pour le port ALIN:</i>		
ASMTime	Base de temps pour mesurer le comptage de la passerelle soeur	
ATimeDue	Durée avant la diffusion suivante	
ATxTime	Horodatage pour la diffusion de transmission	
ATxSlip	Glissement sur diffusion transmission	
ATxPrd	Période pour la diffusion de transmission	
APhTime	Temps entre ma diffusion et la diffusion soeur suivante	
APhErr	Erreur sur la dernière phase mesurée	
ABcstCnt	Comptage diffusion du routage	
ASRxCnt	Nombre de diffusions de routage soeur reçues	
ARxCnt	Nombre total de diffusions de routage reçues	
ASMCnt	Nombre de périodes de mesure	
ASCnt	Nombre de passerelles soeur	
ASRxFnd	Nombre de soeurs trouvées jusqu'à présent	
ASMode	En mode soeur (booléen)	
AMyPhase	J'étais le dernier noeud soeur à transmettre (booléen)	
ASplSeg	Segment divisé détecté	

Table 12-29 Paramètres du bloc BCS_DIAG

XEC_DIAG : BLOC DIAGNOSTIC DES TACHES

Fonction du bloc

Ce bloc permet d'afficher des informations sur les tâches dans le système d'exploitation de l'instrument (XEC).

Paramètres du bloc

Les symboles utilisés dans la table 12-30 sont expliqués au chapitre 1.
















Paramètre	Fonction	Etat
LocITUID	Sélection de la tâche	
CurrPrtY	Priorité active de la tâche	
BasePrtY	Priorité de base de la tâche	
TCB_Lock	Verrouillages de la tâche	
TUID	ID (Identité) de la tâche	
SuspMask	Masque de suspension de la tâche	
TSuspMask	Masque de suspension de la tâche temporisé	
PendMask	Réveils en attente	
AlrmTime	Durée de réveil suivante	
CPUTime	Compteur utilisation UC	
WDCount	Compteur chien de garde	
Alarms		  
Software	Données corrompues/Défaut de communication	
StackErr	Alarme dépassement pile	
Combined	Alarme combinée du bloc	
CurrFree	Pile libre active	
MinFree	Pile libre minimale	

Table 12-30 Paramètres du bloc XEC_DIAG

SUM_DIAG: BLOC DE DIAGNOSTIC RÉCAPITULATIF

Fonction du bloc

Ce bloc regroupe et récapitule les données sur les défauts réseau et de la base de données d'un instrument actif. La plupart des données peuvent également être affichées par d'autres blocs de diagnostic, mais SUM_DIAG met en évidence les informations les plus importantes et les présente sous forme simplifiée. Le bloc effectue une analyse "experte" pour ne retenir que les indications de défaut significatives, ce qui permet de diagnostiquer très rapidement les défauts du réseau LIN et de la base de données.

Paramètres du bloc

Les symboles utilisés dans la table 12-31 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
InstType	Type de l'instrument	Alphanumérique	
Version	Version du logiciel de l'instrument	Alphanumérique	
Database	Nom de la base de données active	Alphanumérique	
SegNode	Segment (AB) & adresse du noeud (CD)	ABCD hex	
LastErr	Dernier code d'erreur de la BD (voir manuel)	Hex	
LRA	Etat de l'algorithme de redondance LIN	Menu	
MAC	Etat de la couche de contrôle d'accès au support	Menu	
LLC	Etat de la couche de contrôle des liens logiques	Menu	
NET	Etat de la couche NET (<i>non-encore utilisé</i>)	Menu	
EDB	Etat de la couche EDB (BD externe)	Menu	
EDBName	Nom de la BD externe en défaut (voir EDB)	Alphanumérique	
EDBNode	N° du noeud de la BD externe en défaut (voir EDB)	Hex	
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
LRA	Alarme état LRA. VRAI si état LRA pas "OK"	V/F	
MAC	Alarme état MAC. VRAI si état MAC pas "OK"	V/F	
LLC	Alarme état LLC. VRAI si état LLC pas "OK"	V/F	
NET	Alarme état NET. VRAI si état NET pas "OK"	V/F	
EDB	Alarme état EDB. VRAI si état EDB pas "OK"	V/F	
Status	Alarme d'état. VRAI si bits VRAIS	V/F	
Spare (4)	(<i>Alarmes en réserve</i>)	V/F	
Combined	Alarme combinée du bloc	V/F	
Mode	Mode de fonctionnement de la base de données	Menu	
Status	Informations générales d'état	(ABC)D hex	
Memory	VRAI si < 2% de la mémoire BD libre	V/F	<div style="border: 1px solid black; padding: 2px; display: inline-block;"> <div style="display: flex; align-items: center;"> <div style="width: 10px; height: 10px; margin-right: 5px;"></div> <div style="text-align: center;"> 1 2 4 8 </div> </div> <div style="margin-left: 10px; font-weight: bold;">D</div> </div>
ScanRate	Tps scrutation serveur blocs local (tâche utilisateur 1) ms		
Spare1 à Spare7	(<i>non-encore utilisé</i>)		
Spare8	(<i>non-encore utilisé</i>)	Alphanumérique	

Table 12-31 Paramètres du bloc SUM_DIAG

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-31.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

SegNode. Quatre chiffres hexadécimaux indiquant le segment LIN (deux premiers chiffres) et l'adresse de noeud (deux derniers chiffres) de ce noeud.

LastErr. Quatre chiffres hexadécimaux indiquant le code de la dernière erreur de la base de données. Voir la liste des codes d'erreur et de leur signification (pour les cas non-évidents) dans le manuel spécifique à l'instrument.

LRA. (OK/'N/A'/SelfTstB/SelfTstA/FaultOnB/FaultOnA). Indique l'état du LRA (algorithme de redondance LIN). Voir les détails au début de ce chapitre.

- **OK.** Indique l'absence de défauts LRA
- **N/A (Sans objet).** Indique que le LRA est inapproprié (instruments ALIN).
- **SelfTstB.** Indique l'échec du test automatique LRA sur LIN B, si LIN B est activé. Indique un défaut physique du poste local.
- **SelfTstA.** Indique l'échec du test automatique LRA sur LIN A, si LIN A est activé. Indique un défaut physique du poste local.
- **FaultOnB.** Indique un défaut LRA sur LIN B, si LIN B est activé. Indique qu'un message a été détecté sur LIN A qui n'est pas apparu sur LIN B. Le défaut peut être causé par un câblage ou un connecteur défectueux, un bruit excessif ou un problème physique sur l'un des postes.
- **FaultOnA.** Indique un défaut LRA sur LIN A, si LIN A est activé. Indique qu'un message a été détecté sur LIN B qui n'est pas apparu sur LIN A. Le défaut peut être causé par un câblage ou un connecteur défectueux, un bruit excessif ou un problème physique sur l'un des postes.

NOTA. Les éléments du menu sont classés par ordre de priorité ascendant, autrement dit, l'indication *FaultOnA* aura priorité sur toute autre indication d'état.

MAC. (OK/'N/A'/LoclFail/ToknPass/ToknRotn/ALINConn/LINConn/ALINOffl/LINOffl). Indique l'état de la couche MAC (contrôle d'accès au support). Les éléments du menu sont classés par ordre de priorité ascendant, comme ci-dessus.

- **OK.** Indique l'absence de défauts MAC.
- **N/A.** Indique que MAC est inapproprié (*non-encore utilisé*).
- **LoclFail.** (*Sans objet pour ALIN*) Indique un défaut physique sur le poste local.
- **ToknPass.** (*Sans objet pour ALIN.*) Indique un problème au moment du passage du jeton au poste suivant sur le réseau LIN. Indique un éventuel défaut physique dans la synchronisation de l'équipement local, l'équipement du poste logique suivant ou du câblage. Cet élément du menu ne doit pas apparaître en fonctionnement normal, sauf si les postes sont connectés et déconnectés physiquement.

ToknRotn. (*Sans objet pour ALIN*) Indique que la rotation du jeton pose problème sur le réseau LIN, c'est à dire en cas de problème pour maintenir l'anneau logique. Indique que les postes apparaissent et disparaissent sur le réseau LIN, ce qui peut être causé par la connexion et déconnexion physiques des noeuds ou par un éventuel défaut de câblage. Cet élément du menu ne doit pas apparaître en fonctionnement normal, sauf si les postes sont connectés et déconnectés physiquement.

ALINConn. Indique que le réseau ALIN n'est pas connecté, autrement dit, que l'état MAC est IDLE (inactif), et qu'aucune rotation du jeton n'est possible. Indique que le câblage est incorrect ou interrompu ou qu'aucun autre noeud ALIN n'est connecté.

LINConn. Indique que le réseau LIN n'est pas connecté, autrement dit, que l'état MAC est IDLE (inactif), et qu'aucune rotation du jeton n'est possible. Indique que le câblage est incorrect ou interrompu ou qu'aucun autre noeud ALIN n'est connecté.

ALINOffl. Indique que l'état MAC ALIN est "hors ligne".

LINOffl. Indique que l'état MAC ALIN est "hors ligne", en raison, par exemple, d'une référence de noeud erronée (00 ou FF), une référence de noeud dupliquée ou éventuellement un transmetteur défectueux.

LLC. (OK/'N/A'/ALINBuff/LINBuff/ALINIdle/LINIdle) Indique l'état de la couche LLC (contrôle des liens logiques). Ces éléments de menu sont classés par ordre de priorité ascendant, comme précédemment.

■ **OK.** Indique qu'aucun défaut LLC n'a été détecté.

■ **N/A.** Indique que LLC est inapproprié (*non-encore utilisé*).

■ **ALINBuff.** Indique que les tampons LLC ALIN sont saturés, autrement dit que *SAPsfree*, *Tx_Free* ou *Rx_Free* sont à zéro. C'est ce qui se passe en général, lorsque vous tentez de connecter trop de bases de données externes (EDB).

■ **LINBuff.** Indique que les tampons LLC LIN sont saturés, autrement dit que *SAPsfree*, *Tx_Free* ou *Rx_Free* sont à zéro. C'est ce qui se passe en général, lorsque vous tentez de connecter trop de bases de données externes (EDB).

■ **ALINIdle.** Indique que la couche LLC ALIN ne fonctionne pas. (Hautement improbable, mais indique un défaut sérieux).

■ **LINIdle.** Indique que la couche LLC LIN ne fonctionne pas. (Hautement improbable).

NET. (OK/'N/A') Indique l'état de la couche (???). Les éléments du menu sont classés par ordre de priorité ascendant, comme précédemment.

■ **OK.** Indique qu'aucun défaut NET n'a été détecté.

■ **N/A.** Indique que NET est inapproprié (*non-encore mise en oeuvre*).

NOTA. Le champ *NET* est destiné à une expansion future, mais n'est pas encore utilisé.

EDB. (OK/'N/A'/Teatt/Featt/Verify/Disconn/Idle). Indique l'état de la couche EDB (base de données externe). *EDB* vérifie les bases de données externe et affiche les informations de défaut de la première base de données externe qui pose problème. Les éléments de menu sont classés par ordre de priorité ascendant, comme précédemment.

- **OK.** Indique qu'il n'y a aucun défaut dans aucune des bases de données externes.
- **N/A.** Indique que les EDB sont inappropriées (*non-encore mis en oeuvre*).
- **Teatt.** Indique qu'au moins une alarme *Teatt* est active dans la base de données externe, c'est à dire que les blocs locaux en mémoire cache ne sont pas mis à jour. C'est ce qui se produit en général en cas de désignation erronée des blocs en mémoire cache. Le ou les blocs qui provoquent une alarme *Teatt* sont eux-mêmes en alarme logiciel.
- **Featt.** Indique qu'une alarme *Featt* est active, autrement dit que le noeud local ne transmet pas les mises à jour des blocs en mémoire cache. C'est ce qui se produit en général lorsqu'un noeud dont les blocs locaux sont en mémoire cache se met hors ligne. Sinon, le défaut peut être causé par un noeud déporté qui demande un nom de bloc qui n'existe pas dans cet instrument. Le noeud déporté signalera une alarme logiciel.
- **Verify.** Indique que l'EDB n'a pas établi le rattachement, autrement dit que l'état EDB est INIT ou VERIFYING. Il s'agit en général d'un état transitoire qui se produit pendant la connexion à des bases de données déportées.
- **Disconn.** Indique que l'EDB n'est pas connectée, autrement dit que l'état de l'EDB est DORMANT, DISCONNECTED ou DISCONNECTING (en sommeil, déconnecté ou en cours de déconnexion). C'est ce qui se produit en général lorsqu'une base de données déportée est inexistante ou arrêtée. Si l'EDB a été créée par un noeud déporté, il s'agit d'un état transitoire qui prend fin lorsque l'EDB locale est supprimée. Il s'agit d'un état non-transitoire si le noeud déporté est déconnecté ou ne tourne pas.
- **Idle.** Indique que la couche EDB ne fonctionne pas, autrement dit que l'état EDB est HORS LIGNE (Défaut qui se produit rarement).

Mode. (Auto/Init/Latch/Hold) Permet de définir et d'indiquer le mode de fonctionnement de la base de données.

- **Auto.** Mode de fonctionnement automatique normal.
- **Init.** Réinitialise les valeurs du bloc et repasse en mode *Auto*.
- **Latch.** Mémorise l'état de priorité le plus élevé de chaque champ jusqu'à ce qu'il soit supprimé en mettant *Mode* sur *Auto*.
- **Hold.** Gèle les valeurs actives.

Status. Champ binaire donnant des informations générales d'état (voir table 12-31).

NOTA. Seul un bit est utilisé à l'heure actuelle — les 15 autres sont réservés à un usage ultérieur.

ScanRate. Temps d'itération en millisecondes du serveur de blocs local ou de la tâche utilisateur 1 pour les instruments qui gèrent cette fonction.

TOD_DIAG: BLOC DE DIAGNOSTIC DE L'HEURE DU JOUR

Fonction du bloc

Ce bloc permet de disposer d'un mécanisme pour contrôler et superviser la synchronisation de l'heure réseau. Les trois modes du bloc déterminent le fonctionnement de la synchronisation de l'heure, en vous permettant de définir si un instrument doit diffuser (mode MASTER - maître) ou recevoir (mode SLAVE - esclave) l'heure du jour ou doit être isolé des efforts de synchronisation du réseau (mode ISOLATED - isolé).

Paramètres du bloc

Les symboles utilisés dans la table 12-32 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.










Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement actif.	Menu	
Brdcasts	Nbre de diffusions heure effectuées sur le réseau	Entier	
RcvdOk	Nbre de diffusions heure valables reçues du réseau	Entier	
RvcdRjct	Nbre de diffusions heure reçues du réseau & rejetées	Entier	
Alarms			  
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
Own Time	Instrument n'a pas reçu la diffusion régulière du maître	V/F	
NoTime	Horloge temps réel de l'instrument non-configurée	V/F	
Muffled	Instrument maître a été neutralisé	V/F	
Combined	Alarme combinée du bloc	V/F	
NeedTime	Noeud diffusé requête d'heure, réponse attendue	V/F	
ReqsIn	Nbre de requêtes reçues par ce noeud	Entier	
ReqsOut	Nbre de requêtes diffusées par ce noeud	Entier	

Table 12-32 Paramètres du bloc TOD_DIAG

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-32.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Mode. (MASTER/SLAVE/ISOLATED - maître/esclave/isolé) Mode de fonctionnement actif.

■ **MASTER.** Lorsque le mode MASTER est sélectionné, l'instrument diffuse une requête d'heure initiale. S'il n'y a aucune réponse d'un noeud de référence inférieure dans un temps donné (à l'heure actuelle ~2 sec.), l'heure est diffusée. Les diffusions sont transmises à intervalles réguliers (à l'heure actuelle ~15 min.) et également en réponse à des requêtes d'heure. Si une réponse à une requête d'heure ou à une diffusion régulière de l'heure est reçue d'un noeud de référence inférieure, le noeud local est "neutralisé" et l'horloge est mise à l'heure locale. Si un noeud maître est neutralisé, il

agit comme un noeud esclave, sauf si la diffusion régulière de l'heure est tardive (à l'heure actuelle ~20 sec.), le noeud se déneutralise et reprend le statut de maître. Un maître se neutralise également si son horloge temps réel n'a pas été configurée.

- **SLAVE.** Ce mode est le mode par défaut. Lorsque le mode esclave est sélectionné, l'instrument diffuse une requête d'heure initiale. A réception d'une diffusion d'heure, l'horloge est mise à l'heure. Des diffusions sont attendues à intervalles réguliers (~15 min) et une requête d'heure est diffusée si la diffusion régulière de l'heure tarde (~20 sec.). Les requêtes d'heure reçues en mode esclave sont ignorées.
- **ISOLATED.** Le noeud ne diffuse ni l'heure, ni répond aux diffusions et requêtes d'heure transmises par les autres noeuds.

Dans tous les modes sauf en mode isolé, la modification de l'heure de l'horloge locale entraîne une diffusion de l'heure, et les noeuds de réception mettent à jour leur horloge - sauf s'ils sont en mode isolé. Il n'en résulte aucun changement de mode. Notez que l'intervalle entre les diffusions régulières n'est pas affecté par ces diffusions de changement de l'horloge locale.

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **Muffled.** Cet instrument est configuré comme maître, mais a été neutralisé par un maître sur un noeud de référence inférieure ou par suite d'une alarme *NoTime* (horloge temps réel non-configurée). Voir les explications sous la rubrique *Mode* de la page précédente.

RED_CTRL: BLOC DE CONTRÔLE DE REDONDANCE

Fonction du bloc

Il s'agit du bloc de contrôle de redondance des T102/302, qui permettent de contrôler les états de redondance de deux UC T920 fonctionnant en mode duplex. Le bloc qui montre différents paramètres de diagnostic, permet également de déclencher la synchronisation et désynchronisation du processeur, ainsi que l'échange primaire/secondaire.

La gestion de la synchronisation des deux processeurs dans les T920 est traitée par une tâche dédiée, appelée tâche de gestion de la redondance des processeurs (PRMT). Elle contrôle toutes les actions entre les deux processeurs et conserve des statistiques auxquelles les blocs de diagnostic peuvent accéder.

NOTA. Il ne faut configurer qu'un seul bloc RED_CTRL par base de données.

Paramètres du bloc

Les symboles utilisés dans la table 12-33 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-33.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Refresh (rafraîchissement). Ce paramètre permet de réduire l'utilisation mémoire du processeur pour l'exécution des blocs, et également de définir la durée pour examiner les valeurs intermédiaires des champs. Pour visualiser les modifications des paramètres après moins de 10 secondes (valeur par défaut) réduisez *Refresh* en conséquence.

HW_IntLk. La tâche principale du registre des interverrouillages est de vérifier quel processeur a ou demande accès à l'interface SDX/IDLC pour les modules entrées/sorties.

PrIntLck, SeIntLck. Il s'agit d'interverrouillages logiciels calculés par PRMT.

Les préfixes **A_** et **B_** des paramètres renvoient à ce processeur et à l'autre processeur. Ces préfixes associés à **Pr** et **Se** permettent de déterminer ce que chaque processeur croit qu'il fait et ce qu'il croit que l'autre fait. *PrIntLck* donne le point de vue de l'UC primaire et *SeIntLck* celui de l'UC secondaire.

Par exemple,

- Si le sous-champ *PrIntLck.A_IsPrim* est VRAI, alors vous savez que le primaire croit qu'il est le primaire.
- Si *PrIntLck.B_IsPrim* est FAUX, alors vous savez que le primaire croit que le secondaire n'est pas le primaire (aussi évident que cela paraît).
- En examinant le champ *SeIntLck*, vous pouvez déterminer ce que le secondaire perçoit comme étant vrai. À l'état normal, *SeIntLck.A_IsPrim* doit être FAUX, ce qui indique que le secondaire sait qu'il est le secondaire.























Paramètre	Fonction	Unités	Etat
Refresh	Intervalle de mise à jour des blocs (0-10s)	Secs	
HW_IntLk	Registre des interverrouillages matériel	(ABC)D Hex	 
PrIntLk	Interverrouillages logiciel; point de vue UC primaire	ABCD Hex	 
PrHWstat	Bits d'état matériel UC primaire	ABCD Hex	 
PrSWstat	Bits d'état logiciel UC primaire	ABCD Hex	 
PrFilSt	Etat de synchronisation fichiers du primaire	Menu	
PrEdbSt	Etat synchron. ALIN; point de vue primaire	Menu	
PrSYNCst	Etat synchronisation globale; point de vue primaire	Menu	
PrMCStat	Etat fonctionnement global UC primaire	Menu	
PrPRM_DB	Etat fonctionnement base de données primaire	Menu	
PrRedMod	Mode redondance; point de vue UC primaire	Menu	
SYNC	VRAI lance processus synchronisation (auto-réinit)	V/F	
DeSYNC	VRAI lance processus désynchronisation (auto-réinit)	V/F	
Alarms			  
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
PrIntLk	Bit(s) d'interverrouillage primaire à 1	V/F	
SeIntLk	Bit(s) d'interverrouillage secondaire à 1	V/F	
PrHard	Bit(s) d'état matériel primaire remis à 0	V/F	
SeHard	Bit(s) d'état matériel secondaire remis à 0	V/F	
PrSoft	Bit(s) d'état logiciel primaire à 1	V/F	
SeSoft	Bit(s) d'état logiciel secondaire à 1	V/F	
PrMCfail	Etat fonctionnement machine primaire = FAILED	V/F	
SeMCfail	Etat fonctionnement machine primaire = FAILED	V/F	
PrSync	Etat synchronisation primaire = erreur	V/F	
SeSync	Etat synchronisation secondaire = erreur	V/F	
PrEdbSyn	Echec synchronisation EDB primaire (ALIN)	V/F	
SeEdbSyn	Echec synchronisation EDB secondaire (ALIN)	V/F	
Combined	Alarme combinée du bloc	V/F	
PwrData	Données démarrage dans puce horloge temps réel	(AB)CD Hex	 
SeIntLk	<i>Correspond aux paramètres "Pr" ci-dessus</i>		
SeHWstat			
SeSWstat			
SeFilSt			
SeEdbSt			
SeSYNCst			
SeMCStat			
SePRM_DB			
SeRedMod			
Change	VRAI lance processus échange primaire/secondaire	V/F	
DB_Path	Chemin/unité fichier spécifié dans New_DB (<i>inutilisé</i>)	Alphanumérique	
New_DB	DB chargée par nouveau primaire après changt (<i>inutilisé</i>)	Alphanumérique	

Table 12-33 Paramètres du bloc RED_CTRL

PrFilSt, SeFilSt. (unSYNCd/SYNCrqst/SYNCing/in_SYNC/SYNCfail/unSYNCrq).

L'état de synchronisation des fichiers. Pour le contrôle redondant, il est essentiel que tous les fichiers présents sur le primaire le soient également dans le système de fichiers (EEPROM E:) du secondaire. Ces paramètres indiquent le point atteint par le processus de synchronisation des fichiers.

PrEdbSt, SeEdbSt. (unSYNCd/SYNCing/in_SYNC/SYNCfail). Etat de synchronisation ALIN. Pour maintenir les liaisons cache aux noeuds déportés, le secondaire doit dupliquer toutes les bases de données externes (EDB), les TEATT et FEATT qui se trouvent sur le primaire. Ce champ l'état normal ou non du processus de synchronisation de ces ressources.

PrSYNCst, SeSYNCst. (NotSYNCd/SyncStrt/FileSync/FileDone/DBLoadSt/DBLoadDn/DBRunSt/DBRunDn/DBSyncSt/DBSyncDn/SyncDone/SyncErr/FileErr/DBLoadEr/DBRunEr/DBSyncEr/DSyncing). Ces champs indiquent le point atteint par le processus de synchronisation globale. Si la synchronisation échoue, vous devez examiner ce champ pour déterminer précisément les raisons de l'échec de la synchronisation.

NOTA. En raison du mécanisme de reconnexion déjà présent sur le réseau ALIN (pour la mise en mémoire cache des blocs, un échec de la synchronisation des ressources EDB entre les deux processeurs n'est pas considéré comme fatal et n'empêchera donc pas la synchronisation globale. C'est pourquoi l'état de synchronisation EDB ne fait pas partie de ce champ.

PrMCStat, SeMCStat. (POST/INIT/OPERATNL/FAILED). Etat de la machine. Etat de fonctionnement global de l'instrument.

SYNC. Une entrée VRAIE dans ce champ représente une demande de synchronisation des bases de données de deux processeurs T920. Notez que la mise à 1 de *SYNC* supprime toute entrée dans le champ *New_DB*, que la commande synchronisation ait été exécutée normalement ou non.

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **PrIntlck, SeIntlck.** Les bits d'interverrouillage suivants ont été mis à 1 dans le processeur correspondant: *A_FrcdPr, A_MajFlt, A_MinFlt, B_MajFlt, B_MinFlt*.
- **PrHard, SeHard.** Les bits d'état matériel suivants ont été remis à zéro dans le processeur correspondant: *Alin_Ok, ICM_Ok, RTC_Bat, Ext_Bat*.
- **PrSoft, SeSoft.** Les bits d'état logiciel suivants ont été mis à 1 dans le processeur correspondant: *SW_Fault, FngrFlt*.
- **PrMcfail, SeMcfail.** L'état de fonctionnement machine du processeur correspondant est FAILED (défaut).
- **PrSync, SeSync.** L'état de synchronisation du processeur correspondant est: *SyncErr, FileErr, DBLoadEr, DBRunEr, ou DBSyncEr*.
- **PrEdbSyn, SeEdbSyn.** Echec de la synchronisation Edb (ressource ALIN).

Change. Si le paramètre est VRAI, les processeurs changent de rôles, autrement dit l'UC T920 primaire devient secondaire et *vice versa*. Cette requête ne peut aboutir que si les processeurs sont synchronisés. Notez qu'après le basculement, la synchronisation n'est *pas* lancée automatiquement. Pour resynchroniser les T920, une entrée VRAIE dans le champ SYNC ou une intervention sur le panneau avant) est nécessaire.

New_DB. (*Inutilisé dans la version actuelle*) Si un nom de base de données est défini dans ce champ, alors après un basculement normal des processeurs, le nouveau primaire tente d'exécuter cette nouvelle base de données. Notez que *New_DB* est remis à zéro après l'exécution d'une commande *SYNC* (qu'elle aboutisse ou non).

SDX_RSRC: BLOC DE RESSOURCES D'ÉCHANGE DE DONNÉES SÉRIE

Fonction du bloc

Ce bloc indique l'état de l'interface entre un contrôleur d'unités T102 et l'ensemble de seize modules maximum qui peuvent lui être associés - "fingers ou module". L'interface comprend une unité SDX (échange de données série) pour chaque module entrées/sorties et un bus IDLC (supervision de la liaison de données entrées/sorties) par UC T920.

Le bloc SDX_RSRC montre l'état de chaque SDX présent, de chaque IDLC du T920, ainsi que les types de modules configurés dans la base de données par rapport à ceux physiquement présents sur chacun des seize sites.

Paramètres du bloc

Les symboles utilisés dans la table 12-34 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.














Paramètre	Fonction	Unités	Etat
Refresh	Intervalle de mise à jour du bloc (0-60,00 s)	Sec	
PrSDX	Etat SDX pour les sites configurés (Finger_1 à 16)	ABCD Hex	 
PrIDLCOK	VRAI = Etat IDLC du T920 primaire OK	V/F	 
PrFTyp_1 à 16	Types modules E/S configurés dans BD sur les sites	Menu	
Alarms			  
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
SeSDX	Etat SDX pour les 16 sites (Finger_1 à Finger_16)	ABCD Hex	 
SeIDLCOK	VRAI = Etat IDLC du T920 secondaire OK	V/F	 
SeFTyp_1 à 16	Types modules E/S configurés dans BD sur les sites	Menu	

Table 12-34 Paramètres du bloc SDX_RSRC

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-34.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Refresh (Rafraîchissement). Ce paramètre permet de réduire la mémoire utilisée pour l'exécution des blocs par le processeur. La réduction de cette valeur augmente l'intervalle de mise à jour du bloc (limité à l'intervalle de mise à jour du serveur de blocs local).

PrSDX. Etat de SDX du point de vue du T920 primaire. Ce paramètre comprend seize bits — *Finger_1* à *Finger_16* — qui correspond aux sites des modules entrées/sorties du châssis du T102. *PrSDX* n'indique que les sites configurés dans la base de données qui tourne sur le T920 primaire, tous les autres sites sont ignorés.

Un *Finger_n* à l'état bas signifie qu'un module entrées/sorties non-spécifié est connecté à un site configuré *n*, et que son interface SDX fonctionne correctement. Notez que l'état de bon fonctionnement ne signifie pas que le module entrées/sorties connecté est **correct**, mais simplement que les communications SDX fonctionnent bien au niveau de ce site.

Un *Finger_n* à l'état haut signifie soit que le site configuré *n* est vacant ou la présence d'un module entrées/sorties non-spécifié avec un dysfonctionnement des communications SDX ("défaut").

Notez que tous les bits *PrSDX* correspondant à des sites non-configurés sont signalés comme étant UNHLTHY (état haut) quelque soit la présence ou l'absence de SDX fonctionnant correctement ou non au niveau de ces sites.

La partie supérieure de la figure 12-5 montre comment l'IDLC du T920 primaire n'accède qu'aux sites configurés dans la base de données, c'est à dire uniquement les sites 2, 14 et 15 dans cet exemple. Les sites 2 et 14 sont indiqués comme étant HEALTHY (OK) (*Finger₂* et *Finger₁₄* = "bas"). Le site 15 est indiqué comme étant UNHLTHY (défaut) (*Finger₁₅* = "haut"), dans la mesure où le SDX attendu est manquant ou ne fonctionne pas correctement. Tous les autres sites sont indiqués comme étant UNHLTHY par défaut, puisqu'ils n'ont pas été configurés dans la base de données.

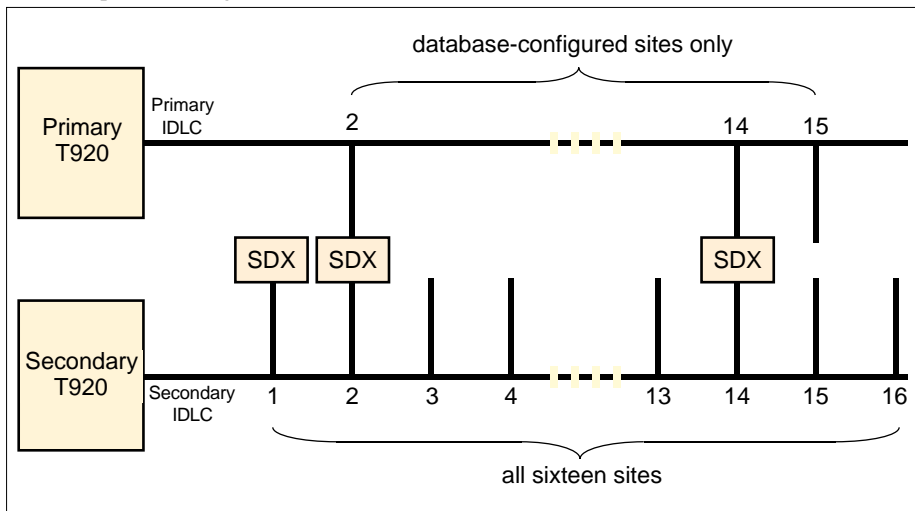


Figure 12-5 Exemple de communications IDLC & SDX (schéma)

NOTA. Si l'IDLC du T920 primaire est en défaut indiqué par le paramètre *PrIDLCOk* = FAUX, alors tous les autres sites sont indiqués comme étant UNHLTHY (en défaut).

SeSDX. Etat de SDX du point de vue du T920 secondaire. Ce paramètre comprend seize bits — *Finger₁* à *Finger₁₆* — qui correspond aux sites des modules entrées/sorties du châssis du T102. Notez qu'à l'inverse du paramètre *PrSDX*, *SeSDX* indique l'ensemble des seize sites, et non pas simplement ceux configurés dans la base de données.

Un *Finger_n* à l'état bas signifie qu'un module entrées/sorties non-spécifié est connecté au site *n*, et que son interface SDX fonctionne correctement. Notez que l'état de bon fonctionnement ne signifie pas que le module entrées/sorties connecté est **correct**, mais simplement que les communications SDX fonctionnent bien au niveau de ce site.

Un *Finger_n* à l'état haut signifie soit que le site configuré *n* est vacant ou la présence d'un module entrées/sorties non-spécifié avec un dysfonctionnement des communications SDX ("défaut").

La partie inférieure de la figure 12-5 montre comme l'IDLC du T920 secondaire accède à tous les sites entrées/sorties, et non pas simplement aux sites configurés. Les sites 1, 2 et 14 sont indiqués comme étant HEALTHY (OK) (*Finger_1*, *_2* et *14* = "bas"). Les sites inoccupés restants sont indiqués comme étant UNHLTHY (en défaut) par défaut.

NOTA. Si l'IDLC du T920 secondaire est en défaut indiqué par le paramètre *SelDLCOk* = FAUX, alors tous les autres sites sont indiqués comme étant UNHLTHY (en défaut).

PrFTyp_1 à PrFTyp16. (INVALID/T110/T111/T112/T120/T121/T122/T130/T140/T150/T180/T151). Ces seize paramètres indiquent les types de modules entrées/sorties configurés dans la base de données qui tourne dans le T920 primaire pour les sites correspondants. Autrement dit, ils indiquent quels modules *doivent* être présents au niveau de ces sites, en fonction de la base de données chargée.

INVALID indique qu'aucun module entrées/sorties n'a été configuré pour ce site.

SeFTyp_1 à SeFTyp16. (INVALID/T110/T111/T112/T120/T121/T122/T130/T140/T150/T180/T151). Ces seize paramètres indiquent les types de modules entrées/sorties présents au niveau des sites correspondants dans le châssis du T102 pour tous les sites. Autrement dit, ils indiquent les modules *connectés physiquement* sur ces sites, quelque soit la configuration de la base de données.

Pour ces paramètres, *INVALID* indique un site inoccupé.

FT_TUNE: BLOC PERFORMANCES DU T102


Fonction du bloc

Le bloc FT_TUNE supervise les performances des quatre blocs serveurs du T102 — mises à jour des entrées/sorties (S1), synchronisation et connexions sortantes des blocs locaux (S2) et des blocs en mémoire cache (S3), ainsi que les connexions entrantes des blocs en mémoire cache (S4) — ainsi que la durée de chaque tâche dans le système. Notez qu'un "serveur" est un processus spécifique qui met à jour les blocs d'un schéma de boucles (en lisant, par exemple, les entrées/sorties). Un ou plusieurs serveurs peuvent être prévus pour l'exécution de chaque tâche, et les tâches sont elles-mêmes planifiées par le programme de supervision de la conduite (XEC).

La moitié gauche du menu de spécifications du bloc montre les données du serveur. Une période représente le temps de cycle du serveur, c'est à dire l'intervalle entre les exécutions ultérieures du premier bloc sur la liste de mise à jour. Il n'est jamais inférieur à 100 ms et peut atteindre 1500 ms dans les schémas de boucles importants. La partie droite du menu de spécifications montre les données des tâches, les proportions sont exprimées sur une échelle de 0 à 1000, c'est à dire $\times 0,1\%$. les tâches en haut de liste ont la priorité la plus basse, tandis que celles en bas de liste ont la priorité la plus haute.

Paramètres du bloc

Les symboles utilisés dans la table 12-35 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
S1used	Activité du serveur de blocs E/S (S1) (ms par période)		
S1period	Dernier temps de cycle S1 (ms)		
S1delay	Suspension forcée S1 (ms par période)		
S2used	Activité serveur blocs local (S2) (ms par période)		
S2period	Dernier temps de cycle S2 (ms)		
S2delay	Suspension forcée S2 (ms par période)		
S3used	Activité serveur de synchronisation des blocs (S3) (ms par période)		
S3period	Dernier temps de cycle S3 (ms)		
S3delay	Suspension forcée S3 (ms par période)		
S4used	Activité liaisons entrantes blocs cache (S4) (ms par période)		
S4period	Dernier temps de cycle S4 (ms)		
S4delay	Suspension forcée S4 (ms par période)		
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
Idle	Inactivité — aucune tâche exécutée (pour 1000)		
Bgnd	Utilisation par la tâche de fond (pour 1000)		
BatLoad	Utilisation par la tâche chargement des LOTS:enregistrements (pour 1000)		
Config	Utilisation par la tâche Configurateur (pour 1000)		
FileSync	Utilisation par la tâche synchronisation des fichiers (pour 1000)		
Load	Utilisation par la tâche chargement de la base de données (pour 1000)		
LLC	Utilisation par la tâche de supervision des liaisons logiques (pour 1000)		

suite ...

...suite

Paramètre	Fonction	Unités*	Etat*
CacheCon	Utilisation par la tâche liaisons cache (pour 1000)		
CacheSrv	Serveur de blocs pour les blocs cache, utilisation (pour 1000)		
LocalSrv	Serveur de blocs pour les blocs locaux, utilisation (pour 1000)		
NFS	Utilisation par la tâche système de fichiers réseau (pour 1000)		
Network	Utilisation par la tâche réseau (pour 1000)		
PRMT	Utilisation par la tâche PRMT (pour 1000)		
RX_ALIN	Utilisation par la tâche de traitement des tampons de réception ALIN (pour 1000)		
RX_ICM	Utilisation par la tâche de traitement des tampons de réception ICM (pour 1000)		
FngrChck	Utilisation par la tâche de vérification des modules E/S ("doigt") (pour 1000)		

*Tous les paramètres sont au format entier, et en lecture seule en conduite.

Table 12-35 Paramètres du bloc FT_TUNE

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-35.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

BatLoad. Durée utilisée par la tâche de chargement des lots/enregistrements.

Config. Durée utilisée par la tâche du configurateur (interface au terminal).

LLC. Durée utilisée par la tâche de supervision des liaisons logiques, qui est responsable des fonctions réseau de base.

CacheCon. Durée utilisée par le serveur responsable de la génération des écritures ALIN, lorsque les données sur des liaisons aux blocs en mémoire cache changent.

CacheSrv. Durée utilisée par le serveur de blocs en mémoire cache. En raison du fonctionnement de la gestion de la redondance, les blocs en mémoire cache sont synchronisés entre les processeurs T920 primaire et secondaire par ce serveur dédié séparé.

NFS. Durée utilisée par la tâche serveur de fichiers réseau, qui est responsable du traitement des requêtes de copie de fichiers sur le réseau.

FngrChck. Durée utilisée par la tâche de vérification des doigts, qui est responsable de la vérification de l'accès par le processeur T920 secondaire aux modules entrées/sorties ("doigts"), autrement dit du fonctionnement de l'équipement, qui est désynchronisé en cas de défaillance du matériel.

NOTA. Cette tâche ne tourne que sur le T920 secondaire, la valeur du champ doit donc toujours être zéro.

FTQ_DIAG: BLOC DE DIAGNOSTIC DES FILES D'ATTENTE PRMT

Fonction du bloc

Chaque procédé dans un contrôleur d'unités T102 ou un superviseur d'unités T302 "parle" au PRMT (tâche de gestion de la redondance des processeurs) par l'intermédiaire d'un ensemble de files d'attente. Le bloc FTQ_DIAG permet d'accéder aux statistiques de diagnostic de base de chaque file d'attente maintenue par le PRMT.

Paramètres du bloc

Les symboles utilisés dans la table 12-36 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

















Paramètre	Fonction	Unités	Etat
Refresh	Intervalle de mise à jour du bloc (0-60.00 sec)	Sec	
PrQueue	Sélection file d'attente processeur primaire, par nom	Menu	
PrMode	Mode d'affichage stats file d'attente sélectionnée	Menu	
PrIdent, Seldent	} Réservé à Eurotherm Automation TCS Systèmes		
PrSender, SeSender			
PrSndEvt, SeSndEvt			
PrSndRsn, SeSndRsn			
PrRcvr, SeRcvr			
PrRcvEvt, SeRcvEvt			
PrRcvRsn, SeRcvRsn			
PrTx_Cnt, SeTx_Cnt			
PrNo_Buf, SeNo_Buf			
PrTx_Err, SeTx_Err			
PrRx_Cnt, SeRx_Cnt			
PrNo_Msg, SeNo_Msg			
PrRx_Err, SeRx_Err			
Alarms			  
Software	Anomalie données RAM du bloc /défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
SeQueue	Sélect. file d'attente processeur secondaire, par nom	Menu	
SeMode	Mode d'affichage stats file d'attente sélectionnée	Menu	

Table 12-36 Paramètres du bloc FTQ_DIAG

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-36.

PrMode, SeMode. Champs d'entrée utilisateur, qui permettent une mie à jour continue (AUTO) ou l'initialisation (INIT) des données de la file d'attente sélectionnée et également le gel des affichages (HOLD) pour aider à la synchronisation des événements.

PrQueue, SeQueue. Champs d'entrée utilisateur, qui permettent de sélectionner les statistiques de la file d'attente à afficher.

PrIdent to SeRcvRsn. Ces champs fournissent des détails complexes du fonctionnement du mécanisme de mise en file d'attente et ne sont d'aucune aide pour la plupart des utilisateurs. En cas de problèmes persistants, les membres de l'assistance technique d'Eurotherm peuvent demander l'examen de ces valeurs comme aide aux mesures de diagnostic prises par Eurotherm. Si des schémas réguliers et/ou cohérents dans les valeurs de ces champs devaient apparaître dans le temps, ces informations seront intégrées dans les mises à jour à venir de la documentation.

PrTx_Cnt à SeRx_Err. Ces champs affichent un ensemble particulier de décomptes, de la manière spécifiée par les paramètres *PrMode* et *SeMode*.

ICM_DIAG: BLOC DE STATISTIQUES DU MODULE DE COMMUNICATION INTERPROCESSEUR

Fonction du bloc

Le bloc ICM_DIAG fournit des statistiques de diagnostic ICM (module de communication interprocesseur) sur le nombre et les types de messages transmis par chaque processeur de deux T920 redondants. L'ICM est le module responsable du transfert physique des messages entre deux processeurs.

Paramètres du bloc

Les symboles utilisés dans la table 12-37 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Refresh	Intervalle de mise à jour du bloc (0-60,00 secs)	Sec	
Alarms			
Software	Anomalie données RAM du bloc / défaut réseau	V/F	
Combined	Alarme combinée du bloc	V/F	
PrMode	Mode affichage stats du "Pr" (primaire) sélectionné	Menu	
PrRx_Cnt	Nbre de messages reçus		
PrTx_Cnt	Nbre de messages transmis		
PrNo_Buf	Nbre de tampons utilisés		
PrFail	Nbre de messages qui n'ont pas abouti		
PrTx_Bsy	Nbre de fois où ICM trop occupé pour transmettre		
PrTx_Err	Nbre de fois où transmission ICM a échoué		
SeMode	Correspond aux paramètres "Pr" ci-dessus		
SeRx_Cnt			
SeTx_Cnt			
SeNo_Buf			
SeFail			
SeTx_Bsy			
SeTx_Err			
Class	Classe des statistiques de messages transmis et reçus (ci-après)		
Cls_Mode	Mode affichage stats transmission & réception sélect.	Menu	
TxBusy	Nbre de messages émis perdus - ICM occupé		
TxFail	Nbre de messages émis perdus - autres défauts		
RxMissng	Nbre de messages émis restés sans réponse		
RxDuplct	Nbre de réponses reçues en double		

Table 12-37 Paramètres du bloc ICM_DIAG

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-37.

Dbase, Block, Type. Voir les détails sur ces blocs "en-tête" à la page 2-3.

PrMode, SeMode. (AUTO/INIT/HOLD/INITBOTH/HOLDBOTH) Champs d'entrée utilisateur, définissant la manière dont les paramètres "*Pr*" et "*Se*" affichent leurs statistiques.

- **AUTO** est le mode normal, les statistiques étant mises à jour régulièrement (limitées par le champ *Refresh* - rafraîchissement).
- **HOLD** gèle les affichages de statistiques, pour pouvoir les examiner, par exemple. Si vous repassez en mode AUTO après HOLD, les statistiques font un bond en avant pour afficher les valeurs atteintes lorsque HOLD était actif.
- **INIT** réinitialise les statistiques.
- **HOLDBOTH** (*PrMode* uniquement) gèle simultanément les affichages des statistiques "*Pr*" et "*Se*".
- **INITBOTH** (*PrMode* uniquement) réinitialise simultanément les affichages des statistiques "*Pr*" et "*Se*".

Class. (EXCEPTION/DB/DBMNGR/LINSYNC/FILING/STATUS/SYNC/CLASS_7 à CLASS_15) Champ d'entrée utilisateur qui permet de sélectionner la classe de messages affichés par les champs *Txbusy*, *TxFail*, *RxMissng* et *RxDuplct*.

Cls_Mode. (AUTO/INIT/HOLD) Champ d'entrée utilisateur qui s'applique aux champs *TxBusy*, *TxFail*, *RxMissng* et *RxDuplct*. Sélectionnez AUTO pour une mise à jour continue des statistiques de la classe active, INIT pour les réinitialiser et HOLD pour geler leurs affichages.

TxBusy, TxFail, RxMissng, RxDuplct. Les messages affichés par ces champs sont de la classe sélectionnée par le paramètre *Class*, et ne sont disponibles que pour l'UC T920 primaire.

MDBDIAG: BLOC DE DIAGNOSTIC MODBUS

Fonction du bloc

Ce bloc affiche des informations qui sont normalement disponibles dans la table de diagnostic d'une configuration Modbus. (Voir les informations détaillées sur les tables de diagnostic dans le *Guide d'utilisation LINtools T500*).

L'utilisation d'un bloc MDBDIAG constitue un moyen plus efficace d'accéder aux données de diagnostic Modbus que par l'intermédiaire d'une table, et permet également de disposer d'une table supplémentaire pour la configuration. Notez que la configuration d'un bloc MDBDIAG désactive la table de diagnostic correspondante, ce qui évite des conflits de valeurs binaires.

Dans le bloc, les champs *Diag0* à *Diag15* correspondent aux registres de diagnostic 0 à 15 de la table, et les champs *Table1* à *Table16* aux registres de diagnostic 16 à 31 de la table. Un champ supplémentaire à 16 bits, *TableOf1*, comprend le complément du bit (bit 0), un bit par table.

Paramètres du bloc

Les symboles utilisés dans la table 12-38 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Les informations suivantes sont complémentaires de celles de la table 12-38.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

TableOf1. Champ binaire à 16 bits, *Table1* à *Table16*, qui renvoie aux tables correspondantes dans la configuration Modbus. Pour les esclaves, ce bit est VRAI, si la table n'a pas fait l'objet d'écritures ou de lectures pendant la période définie par le paramètre *Time out* dans le menu *Setup* (Configurer) de la configuration Gateway (Passerelle) — voir le *Guide d'utilisation LINtools T500*. Pour les maîtres, le bit est VRAI, si l'esclave ne répond pas aux sous-programmes de scrutation dans les secondes du *Time out*, c'est à dire qu'il est effectivement "offline" (hors ligne).

Notez que pour les esclaves le bit *TableOf1* est l'inverse du bit *Online* du champ *Tablen*. Pour les maîtres, le bit *TableOf1* n'est vrai que si la référence de l'esclave est différente de zéro et que le bit *DisScan* de la *Tablen* est FAUX.

Chaque bit de ce champ peut être relié au champ *UserAlrm* d'un bloc TOT_CON (ou à toute autre entrée appropriée) pour indiquer les défauts de communication Modbus.

Table1 à Table16. Ensemble de 16 champs binaires comprenant des bits de supervision et de contrôle pour la table Modbus correspondante. L'utilisation de ces bits dépend de ce que la table est associée à un maître ou un esclave - voir les détails dans *Guide d'utilisation LINtools T500*.








Paramètre	Fonction	Unités	Etat
TableOfI	Etat en/hors ligne de la table	ABCD hex	 
Table1	VRAI si Table 1 hors ligne	V/F — 1	D
Table2	VRAI si Table 2 hors ligne	V/F — 2	
Table3	VRAI si Table 3 hors ligne	V/F — 4	
Table4	VRAI si Table 4 hors ligne	V/F — 8	
Table5	VRAI si Table 5 hors ligne	V/F — 1	C
Table6	VRAI si Table 6 hors ligne	V/F — 2	
Table7	VRAI si Table 7 hors ligne	V/F — 4	
Table8	VRAI si Table 8 hors ligne	V/F — 8	
Table9	VRAI si Table 9 hors ligne	V/F — 1	B
Table10	VRAI si Table 10 hors ligne	V/F — 2	
Table11	VRAI si Table 11 hors ligne	V/F — 4	
Table12	VRAI si Table 12 hors ligne	V/F — 8	
Table13	VRAI si Table 13 hors ligne	V/F — 1	A
Table14	VRAI si Table 14 hors ligne	V/F — 2	
Table15	VRAI si Table 15 hors ligne	V/F — 4	
Table16	VRAI si Table 16 hors ligne	V/F — 8	
Table1 à Table16	Bits de supervision & contrôle de la table	ABCD hex	
Online	VRAI si accès/réponse à la table dans temps imparti	V/F — 1	D
DisScan	VRAI arrête la scrutation de l'esclave par le maître	V/F — 2	
Single	VRAI laisse séquence initialiser scrutation unique	V/F — 4	
Complete	VRAI si maître a terminé scrutation de l'esclave	V/F — 8	
DisWr	VRAI désactive les écritures du maître à l'esclave	V/F — 1	C
ForceWr	VRAI force une écriture	V/F — 2	
Spare1	(inutilisé)	V/F — 4	
Spare2	(inutilisé)	V/F — 8	
ScanErr1	Code d'erreur lecture	V/F — 1	B
ScanErr2		V/F — 2	
ScanErr3		V/F — 4	
ScanErr4		V/F — 8	
WrErr1	Code d'erreur écriture	V/F — 1	A
WrErr2		V/F — 2	
WrErr3		V/F — 4	
WrErr4		V/F — 8	
Alarms			  
Software	Anomalie données RAM du bloc/ défaut réseau	V/F	
TableOfI	VRAI si TableOfI ≠ 0 (càd table hors ligne)	V/F	
Config32	VRAI si Diag1 ≠ 0 (config 32 bit erronée)	V/F	
Combined	Alarme combinée du bloc	V/F	
Diag0 à Diag15	Registres de diagnostic internes	ABCD hex	

Table 12-38 Paramètre du bloc MDBDIAG

Alarms. Voir la description générale du champ Alarms à la page 2-5.

- **TableOfI.** VRAI si l'un des bits du champ *TableOfI* est à 1, autrement dit, si une table est hors ligne. Cet alarme indique une erreur de communication Modbus.
- **Config32.** Indique une erreur de configuration 32 bits. Un registre 32 bits peut être configuré correctement (voir le *Guide d'utilisation LINtools T500*), mais ensuite la configuration peut être corrompue par inadvertance en modifiant la taille de la table, le nombre de scrutations, etc. Le bit d'alarme *Config32* est mis à 1, si de tels problèmes sont détectés au moment du chargement de la table, mais la communication Modbus sera néanmoins établie.

NOTA. Si aucun bloc MDBDIAG n'est configuré, une erreur de configuration 32 bits met le bit (bit 1) "Disable continuous scan" (désactiver scrutation continue) à 1, **et la table Modbus ne pourra communiquer.**

Diag0 à Diag15. Ensemble de 16 champs hexadécimaux correspondant aux registres de diagnostic 0 à 15 de la table Modbus. Les fonctions de ces champs sont décrits ci-après dans la table 12-39. Voir les détails dans le *Guide d'utilisation LINtools T500*.

Champ	Fonction
Diag0	(Inutilisé)
Diag1	Erreurs de configuration 32 bits (bits 0 à15 correspondent aux tables 1-16)
Diag2	Registre de diagnostic, bits affectés: Bit 5 — Esclave en mode écoute seule
Diag3	Données d'interrogation transmises par le code de fonction 8 sous-code 0
Diag4	Délimiteur d'entrée transmis par le code de fonction 8 sous-code 3
Diag5	(Inutilisé)
Diag6	(Inutilisé)
Diag7	Nombre de messages d'erreur transmis par l'esclave
Diag8	(Inutilisé)
Diag9	(Inutilisé)
Diag10	(Inutilisé)
Diag11	Tâche de scrutation maître: cycle de tops de 4 ms
Diag12	Tâche de scrutation: durée pour vérifier toutes les tables en tops de 4 ms
Diag13	Tâche de scrutation: temps écoulé depuis la dernière périodicité en tops de 4 ms
Diag14	Tâche de scrutation: temps écoulé depuis la dernière temporisation en tops de 4 ms
Diag15	(Inutilisé)

Table 12-39 Fonctions des champs Diag0 à Diag15




SFC_DIAG: BLOC DE DIAGNOSTIC DES SEQUENCES

Fonction du bloc

Le bloc SFC_DIAG indique les niveaux de ressources réels *utilisés* par la séquence active (partie gauche du menu de spécifications comme sur l'écran du configurateur LINtools T500), ainsi que les niveaux de ressources maximum autorisés par la version du logiciel (dans la moitié droite). Notez que les valeurs affichées des paramètres ne sont valables qu'en conduite dans l'instrument cible.

Paramètres du bloc

Les symboles utilisés dans la table 12-40 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités*	Etat*
SFC	Nombre total de SFC (racines & sous-SFC) chargés		
ROOT	Nombre total de SCF racines chargés		
STEP	Nombre total d'étapes chargées		
STEPLINK	Nombre total de "fils" entrant et sortant des étapes		
TRAN	Nombre total de transitions chargées		
TRANLINK	Nombre total de "fils" entrant dans les transitions		
ASSOCIAT	Nombre total de associations d'actions chargées		
ACTION	Nombre total d'actions référencées		
EVENT	Nombre total d'événement fixes planifiés (par les étapes actives)		
POINT	Nbre total de points sélectionnés pour l'affichage dans VIEW Séquence T1000		
SFCsize	Espace total ST utilisé (octets)		
Alarms			  
Software	Anomalie données RAM du bloc /défaut réseau	V/F	
NonSeq	Instrument cible ne gère pas les SFC	V/F	
Combined	Alarme combinée du bloc	V/F	
MaxSFC	Nombre maximum de SFC (roots & sous-SFC) chargeables		
MaxROOT	Nombre maximum de SFC racines chargeables		
MaxSTEP	Nombre maximum d'étapes chargeables		
MaxSTLK	Nombre maximum autorisé de "fils" entrant et sortant des étapes		
MaxTRAN	Nombre maximum de transitions chargeables		
MaxTRLK	Nombre maximum autorisé de "fils" entrant dans les transitions		
MaxASSOC	Nombre maximum d'associations d'action chargeables		
MaxACTN	Nombre maximum d'actions référençables		
MaxEVENT	Nombre maximum d'événements fixes planifiables (par les étapes actives)		
MaxPOINT	Nbre maxi. de points sélectionnables à l'affichage dans VIEW Séquence T1000		
MaxSFCsz	Espace ST maximal disponible (octets)		

*Tous les paramètres sont au format entier, et en lecture seule en conduite

Table 12-40 Paramètres du bloc SFC_DIAG

ISE_DIAG: BLOC DE DIAGNOSTIC DU MOTEUR DE STRATÉGIE INDUSTRIELLE

Fonction du bloc

Le bloc ISE_DIAG indique les options et fonctionnalités disponibles dans le T2900 (moteur de stratégie industrielle) qui exécute le bloc.



Paramètres du bloc

Les symboles de la table 12-41 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
SRAM_Fit	Total SRAM installée	Ko	
SRAM_Use	Total SRAM utilisée	Ko	
DRAM_Fit	Total DRAM installée	Ko	
DRAM_Use	Total DRAM utilisée	Ko	
Flash_Fit	Total Flash installée	Ko	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Comms	Récapitulatif options de communication installée	ABCD hex	
ProfIDPM	Profibus-DP option maître installée	V/F	D
ProfIFMS	Profibus-FMS option installée	V/F	
ALIN	Option ALIN installée	V/F	
Ethernet	Option Ethernet installée	V/F	
GPB	Option GPB IEEE488 installée	V/F	C
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
COM5	Communications série extension en COM5	V/F	B
COM6	Communications série extension en COM6	V/F	
COM7	Communications série extension en COM7	V/F	
COM8	Communications série extension en COM8	V/F	
LPT1	Option port parallèle LPT1 gérée	V/F	A
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
Panel	Récapitulatif type de face avant installée	(AB)CD hex	
Panel	TRUE si face avant installée	V/F	D
VGA	TRUE si face avant type VGA installée, FALSE si autre	V/F	
FullVGA*	TRUE si full VGA, FALSE si 1/4 VGA	V/F	
Colour*	TRUE si couleur, FALSE si monochrome/échelle gris	V/F	
Touch*	TRUE si écran tactile installé	V/F	C
Keypad*	TRUE si clavier numérique installé	V/F	
Réserve		V/F	
Réserve		V/F	

Table 12-41 suite ...

...Table 12-41 suite

Paramètre	Fonction	Unités	Etat
Media	Liste des supports installés	(ABC)D hex	
Floppy	TRUE si lecteur de disquettes installé	V/F — 1	D
PCMCIA	TRUE si PCMCIA installé	V/F — 2	
LS120	TRUE si LS120 installé	V/F — 4	
Réserve		V/F — 8	
Software	Récapitulatif des fonctions logicielles	(AB)CD hex	
Unicode	TRUE = caractères unicodes 16 bits gérés	V/F — 1	D
Réserve		V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	
Archive	TRUE = Archivage enregistreur données disponible	V/F — 1	C
Réserve		V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	

*Applicable uniquement si Panel TRUE

Table 12-41 Paramètres du bloc ISE_DIAG

PNL_DIAG: BLOC DEDIAGNOSTIC DE LA FACE AVANT

Fonction du bloc

Le bloc PNL_DIAG fournit des informations sur la face avant installée sur le T2900 (moteur de stratégie industrielle) dans lequel le bloc est exécuté.

Paramètres du bloc

Les symboles utilisés dans la table 12-42 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
State	Etat de la face avant	Menu	
MemUse	Pourcentage de mémoire face avant utilisée (%)	Eng	
Access	Niveau d'accès	Entier	
StatusId	Agent actif dans la fenêtre d'état	Entier	
MainId	Agent actif dans la fenêtre principale	Entier	
PopupId	Agent actif dans la fenêtre en incrustation	Entier	
DialogId	Agent actif dans la fenêtre de dialogue	Entier	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Active	Etat fenêtre en incrustation & de dialogue	(ABC)D hex	
PopUp	TRUE = fenêtre en incrustation active	V/F — 1	D
Dialog	TRUE = fenêtre de dialogue active	V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	

Table 12-42 Paramètres du bloc PNL_DIAG

Menu de spécifications du bloc

Les informations ci-dessous complètent celles de la table 12-42.

State. (UNINIT/NONE/BARE/APP). Etat de la face avant.

ALH_DIAG: BLOC DE DIAGNOSTIC DE L'HISTORIQUE DES ALARMES

Fonction du bloc

Le bloc ALH_DIAG fournit des statistiques sur la fonction d'historique des alarmes du T2900 (moteur de stratégie industrielle) dans lequel le bloc est exécuté.

Paramètres du bloc

Les symboles utilisés dans la table 12-43 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.








Paramètre	Fonction	Unités	Etat
SeqNo	N° de séquence de l'historique des alarmes	Entier	
NumAct	Nombre d'alarmes actives	Entier	
NumUnack	Nombre d'alarmes non acquittées	Entier	
ACK_ALL	TRUE = acquittement de toutes les alarmes	V/F	
Alarms			  
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 12-43 Paramètres du bloc ALH_DIAG

Menu de spécifications du bloc

Les informations ci-dessous complètent celles de la table 12-43.

SeqNo. Le numéro de séquence de l'historique des alarmes. Est incrémenté lorsqu'un changement se produit.

DDR_DIAG: BLOC DE DIAGNOSTIC DE LA FONCTION D'ENREGISTREMENT DES DONNÉES






























Fonction du bloc

Le bloc de diagnostic DDR_DIAG fournit des statistiques sur la fonction d'enregistrement des données du T2900 (moteur de stratégie industrielle) dans lequel le bloc est exécuté.

Une grande partie du système DDR appartient ou est contrôlé par les blocs Group dans la base de données ORG. Il y a donc deux niveaux d'informations DDR — à l'échelle du système et spécifiques aux groupes. Pour traiter un nombre arbitraire de groupes sans que cela devienne trop complexe, le bloc DDR_DIAG affiche les données d'un seul groupe, sélectionné en utilisant les champs d'entrée *ShowArea* et *ShowGrp*.

Paramètres du bloc

Les symboles utilisés dans la table 12-44 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
Areas	Nbre de zones détectées dans base de données LIN	Entier	 
Groups	Nbre de groupes détectés dans base de données LIN	Entier	 
Points	Nbre points d'enregistrement dans base données LIN	Entier	 
ShowArea	La zone du groupe à afficher	Entier	 
ShowGrp	Le groupe à afficher	Entier	 
SampRate	Intervalle d'échantillonnage du groupe (secondes)	Eng	 
Samples	Nbre d'échantillons pris pour le groupe actif	Entier	 
Mem_Used	Affectation maxi. mémoire T2900 (octets)	Entier	 
Mem_Aloc	Nbre de blocs affectés à l'utilisation maximale	Entier	 
FlashReq	*Nbre de blocs flash affectés au groupe actif	Entier	 
FlashRcv	*Nbre blocs flash récupérés démarrage chaud groupe	Entier	 
FlashRsz	*Nbre blocs flash redimensionnés démarrage chaud groupe	Entier	 
DataWdth	Taille de chaque échantillon pour groupe actif (octets)	Entier	 
Alarms			  
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

*Unités de 64 Ko

Table 12-44 Paramètres du bloc DDR_DIAG

AMC_DIAG: BLOC DE DIAGNOSTIC DES COMMUNICATIONS MAÎTRE DES APPLICATIONS

Fonction du bloc

Le bloc AMC_DIAG fournit des statistiques de communication pour une ligne données de communications maître des applications et pour un noeud sélectionné sur cette ligne.

Paramètres du bloc

Les symboles utilisés dans la table 12-45 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

















































Paramètre	Fonction	Unités	Etat
Port	Numéro du port	Menu	
L_Update	Total des mises à jour cycliques	Entier	 
L_MisUpd	Total des mises à jours cycliques manquées	Entier	 
L_Req	Total des requêtes sur la ligne	Entier	 
L_Resp	Total des bonnes réponses	Entier	 
L_Reject	Total des requêtes rejetées sur la ligne	Entier	 
L_Lost	Total des messages perdus dus à la défaillance ligne	Entier	 
L_LkTimO	Total dépassement temps imparti liaison sur ligne	Entier	 
L_LkErrs	Total des erreurs de liaison sur la ligne	Entier	 
L_ApNak	Total des NACQ esclaves sur la ligne	Entier	 
L_Stat	Etat de la ligne	(ABC)D hex	 
NodeSusp	Communicat. cycl. un/plusieurs noeuds suspendues	V/F	 
LineSusp	Communicat. cycl. tous noeuds sur ligne suspendues	V/F	
Réserve		V/F	
Réserve		V/F	
L_Reset	TRUE = Remise à zéro statistiques ligne	V/F	  
MemCfg	Pourcentage de mémoire de configuration utilisée	Eng	 
MemTrs	Pourcentage de mémoire transitoire utilisée	Eng	 
MemTrsHI	Pourcentage maximum de mémoire transitoire	Eng	 
Alarms			  
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Node	Numéro du noeud (0 - 255, 0 = OFF)	Entier	
N_Req	Total des requêtes sur le noeud	Entier	 
N_Resp	Total des bonnes réponses sur le noeud	Entier	 
N_Reject	Total des requêtes rejetées sur la ligne	Entier	 
N_Lost	Total des messages perdus dus à la défaillance ligne	Entier	 
N_LkTimO	Total dépassement temps imparti liaison sur ligne	Entier	 
N_LkErrs	Total des erreurs de liaison sur la ligne	Entier	 
N_ApNak	Total des NACQ esclaves sur la ligne	Entier	 

Table 12-45 suite...

...Table 12-45 suite





Paramètre	Fonction	Unités	Etat
N_Stat	Etat du noeud	(ABC)D hex	 
NodeSusp	Communications cycliques sur ce noeud suspendues	V/F -1	D
Réserve		V/F -2	
Réserve		V/F -4	
Réserve		V/F -8	
N_Reset	TRUE = Remise à zéro des statistiques du noeud	V/F	 
N_Filter	Sélection des classes de lecture/écriture	(AB)CD hex	
CyclicR	TRUE sélectionne les lectures cycliques	V/F -1	D
CyclicW	TRUE sélectionne les écritures cycliques	V/F -2	
AcyclicR	TRUE sélectionne les lectures acycliques	V/F -4	
AcyclicW	TRUE sélectionne les écritures acycliques	V/F -8	
AdHocR	TRUE sélectionne les lectures ponctuelles	V/F -1	C
AdHocW	TRUE sélectionne les écritures ponctuelles	V/F -2	
Réserve		V/F -4	
Réserve		V/F -8	

Table 12-45 Paramètres du bloc AMC_DIAG

Menu de spécifications du bloc

Les informations ci-dessous complètent celles de la table 12-45.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFIDP_1/PROFIDP_2). Définition du port de communication et référence du protocole. Ceci permet d'utiliser le même protocole pour plusieurs ports de communication. Le protocole par défaut est MODBUS_1 (qui est la seule option disponible à l'heure actuelle).

PS_TUNE: BLOC DE PERFORMANCES T940

Fonction du bloc

Le bloc PS_TUNE contrôle les surformances des trois blocs serveurs dans le T940 — synchronisation et connexions sortantes des blocs locaux (S2), des blocs cache (S3), connexions entrantes des blocs cache (S4) — ainsi que le pourcentage de temps utilisé par chaque tâche dans le système. Notez qu'un "serveur" est un procédé spécifique qui met à jour les blocs des schémas de boucles (par ex.: synchronisation des blocs cache). Un ou plusieurs serveurs peuvent être prévus pour être exécutés par chaque tâche, et les tâches sont elles-mêmes prévues pour être exécutées par le gestionnaire d'exécution (XEC).

La moitié gauche du menu de spécifications du bloc affiché montre les données du serveur. Une "période" représente le temps de cycle du serveur, c'est à dire le temps entre les différentes exécutions du premier bloc sur la liste de mise à jour. Une période n'est jamais inférieure à 100 ms et peut atteindre 1500 ms dans des schémas de boucles importants. La partie droite du menu de spécifications montre les données des tâches, les proportions sont exprimées sur une échelle de 0 à 1000, c'est à dire $\times 0,1$ %. Les tâches en haut de la liste ont la priorité la plus faible et celle en bas de liste la plus haute.




Paramètres du bloc

Les symboles utilisés dans la table 12-46 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
S2used	Activité du serveur de blocs locaux (S2) (ms par période)		
S2surioid	Dernier temps de cycle S2 (ms)		
S2delay	Suspension forcée S2 (ms par période)		
S3used	Activité du serveur de synchronisation des blocs cache (S3) (ms par période)		
S3surioid	Dernier temps de cycle S3 (ms)		
S3delay	Suspension forcée S3 (ms par période)		
S4used	Activité du serveur connexions entrantes blocs cache (S4) (ms par période)		
S4surioid	Dernier temps de cycle S4 (ms)		
S4delay	Suspension forcée S4 (ms sur surioid)		
OpSystem	Utilisation du système d'exploitation (sur 1000)		
Idle	Temps d'inactivité — aucune tâche exécutée (sur 1000)		
Bgnd	Utilisation tâche de fond (sur 1000)		
BatLoad	Utilisation tâche de charge BATCH/enregistrement (sur 1000)		
Config	Utilisation tâche Configurateur (sur 1000)		
PMC1	Utilisation tâche 1 communications maître Profibus (sur 1000)		
PMC2	Utilisation tâche 2 communications maître Profibus (sur 1000)		
AMC1	Utilisation tâche 1 communications maître applications (sur 1000)		
AMC2	Utilisation tâche 2 communications maître applications (sur 1000)		
AMC3	Utilisation tâche 3 communications maître applications (sur 1000)		
AMC4	Utilisation tâche 4 communications maître applications (sur 1000)		

Table 12-46 suite...

...Table 12-46 suite

Paramètre	Fonction	Unités*	Etat*
Alarms			  
Software	Anomalie mémoire données RAM bloc/défaut réseau V/F		
Combined	Combinaison de tous les bits d'alarmes	V/F	
Load	Utilisation tâche chargement base de données (sur 1000)		
LLC	Utilisation tâche de contrôle lien logique (sur 1000)		
CacheCon	Utilisation tâche connexions cache (sur 1000)		
CacheSrv	Serveur de blocs pour blocs cache, utilisation (sur 1000)		
NFS	Utilisation tâche système archivage réseau (sur 1000)		
LocalSrv	Serveur de blocs pour blocs locaux, utilisation (sur 1000)		
ModServ	Tâche serveur Modbus esclave, utilisation (sur 1000)		
Mod_Rx	Tâche réception Modbus esclave, utilisation (sur 1000)		
FileSync	Utilisation tâche synchronisation fichier (sur 1000)		
Network	Utilisation tâche réseau (sur 1000)		
PRMT	Utilisation tâche PRMT (sur 1000)		
Unused1	(Inutilisé)		
Rx_ALIN	Utilisation tâche traitement tampons réception ALIN (sur 1000)		
Rx_ICM	Utilisation tâche traitement tampons réception ICM (sur 1000)		
ARCirp	Utilisation tâche traitement interruptions ALIN (sur 1000)		
ICMirp	Utilisation tâche traitement interruptions ICM (sur 1000)		
Unused2	(Inutilisé)		
LIOleds	Utilisation tâche affichage LED E/S locales (sur 1000)		
TickTask	Utilisation tâche top horloge (sur 1000)		

*Tous les paramètres sont au format entier, et en lecture seule en conduite.

Table 12-46 Paramètres du bloc PS_TUNE

Menu de spécifications du bloc

Les informations suivantes complètent celles de la htable 12-46.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

BatLoad. Temps utilisé par la tâche de chargement batch/enregistrement.

Config. Temps utilisé par la tâche Configurateur (interface au terminal).

LLC. Temps utilisé par la tâche de contrôle des liens logiques, qui prend en charge les fonctions réseau de niveau inférieur.

CacheCon. Temps utilisé par le serveur chargé de générer les écritures, lorsque les données des connexions aux blocs cache évoluent.

CacheSrv. Temps utilisé par le serveur de blocs cache. En raison du mode de fonctionnement de la gestion de la redondance, les blocs cache sont synchronisés entre les processeurs T940 primaires et secondaires par ce serveur dédié.

NFS. Temps utilisé par la tâche serveur de fichiers du réseau, chargé de traiter les requêtes de copie de fichiers sur l'ensemble du réseau.

PMC_DIAG: BLOC DE DIAGNOSTIC DE LA LIGNE PROFIBUS

Fonction du bloc

Le bloc PMC_DIAG block permet d'accéder aux informations de diagnostic d'une ligne Profibus. Le système Profibus permet de disposer d'un diagnostic par l'intermédiaire d'un simple appel direct qui renseigne un certain nombre de structures de diagnostic. Cet appel est effectué au moment de la mise à jour, et les données retournées sont copiées dans les champs du bloc. Le paramètre *Port* du bloc définit la ligne Profibus, qui doit être soumise au diagnostic.

NOTA. A l'heure actuelle, seule une ligne Profibus est disponible (PROFDP_1).

Paramètres du bloc

Les symboles utilisés dans la table 12-47 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port	Port Profibus à diagnostiquer (PROFDP_1/PROFDP_2)	Menu	
Status	Etat du port	Menu	
Go_NoGo	'Go' = port OK; 'NoGo' = Problème	Menu	
COM	TRUE = port recevant données cycl. de ≥ 1 noeud configuré	V/F	
CyclRate	Intervalle cycle réel en % cycle demandé	Entier	
LkErrCnt	Nbre d'erreurs sur la liaison	Entier	
BdMsgCnt	Nbre de messages rejetés sur le bus	Entier	
MstState	Etat de l'interface Profibus maître de l'instrument	(AB)CD hex	
NoError	Etat normal sans erreur	V/F	<div><div>1</div><div>2</div><div>4</div><div>8</div><div>D</div></div>
NoAccess	Diagnostic inaccessible	V/F	
HwFault	Défaut matériel	V/F	
SlvFault	Défaut esclave	V/F	
SPmFault	Configuration erronée paramètres esclave	V/F	<div><div>1</div><div>2</div><div>4</div><div>8</div><div>C</div></div>
MPmFault	Configuration erronée paramètres maître	V/F	
UnkFault	Défaut inconnu	V/F	
Spare		V/F	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Mode	Mode mémorisé des champs binaires du bloc	Menu	
Slave_No	N°du noeud esclave appliqué à State_1, State_2 Lecture/Ecriture	Entier	

Table 12-47 suite...

...Table 12-47 suite





Paramètre	Fonction	Unités	Etat
State_1	Diagnostic pour l'esclave spécifié par Slave_No	ABCD hex	
Configd	Maître a paramétré normalement l'esclave	V/F — 1	D
CycleRun	Maître en communication cyclique avec l'esclave	V/F — 2	
DiagAval	Diagnostic esclave disponible (message au maître)	V/F — 4	
MstrLock	Esclave lié à ce maître	V/F — 8	
ParamFlt	Erreur paramètres (rejetés par esclave/maître)	V/F — 1	C
BadResps	Réponse erronée de l'esclave	V/F — 2	
UnknwnCd	Esclave ne gère pas la commande	V/F — 4	
GotExtDg	Diagnostic étendu disponible	V/F — 8	
ConfigFlt	Défaut de configuration	V/F — 1	B
NotReady	Esclave pas prêt	V/F — 2	
NoResps	Aucune réponse/réponse erronée	V/F — 4	
Deactvtd	Esclave désactivé	V/F — 8	
SynMdAct	Mode synchro actif	V/F — 1	A
FreMdAct	Mode gel actif (esclave prié de geler sorties)	V/F — 2	
WdogAct	Chien de garde activé	V/F — 4	
DgOnly	Seules données diagnostic extraites de l'esclave	V/F — 8	
State_2	Diagnostic pour l'esclave spécifié par Slave_No(suite)(ABC)D hex		
ParamReq	Esclave nécessite paramétrage	V/F — 1	D
ExDiagOv	Esclave plus de diagnostics qu'il ne peut envoyer	V/F — 2	
		V/F — 4	
		V/F — 8	
Read	Nbre d'octets lecture données cycliques pour Slave_No	Entier	
Write	Nbre d'octets écriture données cycliques pour Slave_No	Entier	

Table 12-47 Paramètres du bloc PMC_DIAG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 12-47.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Status. (OK/Not_Prof/Not_Init/No_Diag/NoSiDiag). Indique l'état du port Profibus défini par le paramètre *Port*.

- **OK.** Le port fonctionne normalement.
(Les paramètres *State_1* and *State_2* ne sont applicables que si *Status* est 'OK'.)
- **Not_Prof.** N'est pas un port Profibus.
- **Not_Init.** Le port n'est pas initialisé.
- **No_Diag.** Les diagnostics ne sont pas disponibles.
- **NoSiDiag.** Les diagnostics esclave ne sont pas disponibles.

NOTA. Les champs du blocs ne sont activés que si *Status* est soit 'OK' ou 'NoSiDiag'.

Mode. (Auto/Hold). Définit si les champs binaires du bloc sont ou non verrouillés. En mode "Hold" (maintien), les champs binaires verrouillent les valeurs des bits qui passent à l'état TRUE (Vrai). Ceci permet d'enregistrer les situations d'erreur non persistantes. Si l'option "Auto" est sélectionnée pour *Mode*, les bits verrouillés existants sont déverrouillés et tout verrouillage ultérieur est inhibé.

AGA8DIAG: BLOC DE DIAGNOSTIC AGA8

Fonction du bloc

Le bloc AGA8DIAG fournit un ensemble de diagnostics sur le fonctionnement détaillé d'un bloc AGA8DATA (calcul AGA8) utilisé dans le schéma de boucles. Pour comprendre la signification des différents champs de ce bloc, consultez les équations publiées dans le rapport American Gas Association Report #8 (rapport n° 8 de l'association américaine du gaz).

Paramètres du bloc

Les symboles utilisés dans la table 12-48 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
AGA8blk	Nom du bloc AGA8DATA à soumettre au diagnostic	Menu	
AGA8name	Chaîne caractères AGA8blk vue pour blocs cache	Alphanumérique	
Search	TRUE recherche le bloc AGA8DATA suivant	V/F	
Pwork	Pression utilisée	Eng	
Twork	Température utilisée	Eng	
Pbase	Pression de base utilisée	Eng	
Tbase	Température de base utilisée	Eng	
Dbase	Densité de base utilisée	Eng	
uu	} Valeur de calcul intermédiaire (voir rapport AGA8)		
rk3p0			
ww			
q2p0			
hh			
bmix			
Alarms			(<))
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
AGA8Blck	Champ AGA8blk erroné, bloc non mis à jour	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
n	Index matrice à utiliser pour B _n (0 - 9)	Entier	
Bn	Une entrée de la matrice B[n]	Eng	
m	Index matrice à utiliser pour F _m (0 - 9)	Entier	
Fm	Une entrée de la matrice F[m]	Eng	
State	Partie du calcul AGA8 en cours d'exécution	Menu	
Statel	Index matrice indiquant n° de passe cette phase de calcul	Entier	
StateJ	Index matrice indiquant n° de passe cette phase de calcul	Entier	
ConcTime	Temps réact. à changement dans concentration gaz	Secs	
TempTime	Temps réaction à changement dans température gaz	Secs	
PresTime	Temps réaction à changement dans pression gaz	Secs	
ExecTime	Temps d'exécution bloc par cycle base de données	Msec	
MaxTime	Temps exécution maxi. bloc par cycle base données	Msec	
MaxState	Etat calculs lorsque MaxTime atteint	Menu	

Table 12-48 Paramètres du bloc AGA8DIAG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 12-48.

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

AGA8blk. Nom du bloc AGA8DATA à soumettre au diagnostic. Sélectionnez-le dans le menu déroulant des noms de bloc.

NOTA. Si le bloc AGA8DIAG en cours de configuration est ne mémoire cache dans une autre base de données, son champ affichera uniquement un symbole '*', au lieu du nom du bloc AGA8DATA déporté. Dans ce cas, utilisez le paramètre *Search* pour sélectionner le nom du bloc AGA8DATA, qui apparaîtra dans son intégralité dans le champ *AGA8name*.

AGA8name. Nom du bloc AGA8DATA à soumettre au diagnostic. Celui-ci est en général le même que celui du champ *AGA8blk*, sauf lorsque le bloc AGA8DIAG est en mémoire cache (voir Nota ci-dessus).

State. (Init/Poll/Chardl/SiBase/BrktBase/DdtlBase/ZdtlBase/NewTemp/Temp/NewPres/Bracket/Ddetail/Zdetail). Partie des calculs AGA8 en cours d'exécution.

- **Init.** Bloc en cours d'initialisation.
- **Poll.** Interrogation pour les changements d'entrée.
- **Chardl.** Nouveau calcul de composition.
- **SiBase.** Conversion de la référence de base en unités métriques.
- **BrktBase.** Parenthèse pour les calculs P pour D (calculs Tbase et Pbase).
- **DdtlBase.** Détermination de D pour références.
- **ZdtlBase.** Détermination de Z pour références.
- **NewTemp.** Collecte de la nouvelle température.
- **Temp.** Exécution des calculs de température.
- **NewPres.** Collecte de la nouvelle pression.
- **Bracket.** Parenthèse P pour les calculs D (calculs T et P).
- **Ddetail.** Détermination de D pour la pression.
- **Zdetail.** Détermination de Z pour la pression.

MaxState. (Init/Poll/Chardl/SiBase/BrktBase/DdtlBase/ZdtlBase/NewTemp/Temp/NewPres/Bracket/Ddetail/Zdetail). Etat des calculs au point où le temps d'exécution du bloc a atteint sa valeur maximale (maintenue dans *MaxTime*). Les éléments du menu ont la même signification que ceux du paramètre *State* (voir ci-dessus).

RSRCDIAG: BLOC DE DIAGNOSTIC DES RESSOURCES











Fonction du bloc

Le bloc RSRCDIAG permet d'accéder en lecture à toutes les ressources de la base de données. Vous pouvez sélectionner une ressource par l'intermédiaire du champ *Resource*, et le champ *SubType* permet de sélectionner une partie spécifique de cette ressource. L'adresse de début, ainsi que les 32 premiers octets de la ressource sélectionnée sont affichés respectivement dans les champs *Segment*, *Offset*, *Base_0* à *Base_1E*. Le champ *Timer* permet de faire défiler automatiquement toutes les ressources affectées du type sélectionné à un intervalle défini par *Timer* (secondes).

Le bloc permet également d'explorer toute zone de mémoire de l'instrument — en saisissant MEM_DIAG dans le champ *Resource* et en spécifiant les valeurs adéquates dans *Segment* et *Offset*.

Paramètres du bloc

Les symboles utilisés dans la table 12-49 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Resource	Sélectionne la ressource en question	Menu	
SubType	Sélectionne le sous-type de la ressource en question	Menu	
Index	Sélectionne la référence de la ressource en question	ABCD hex	
Timer	Sélectionne intervalle défil.(sec) ressources affectées(Odésac)	ABCD hex	
Segment	Segment contenant la ressource sélectionnée	ABCD hex	 _[1]
Offset	Décalage (dans <i>Segment</i>) de la ressource sélectionnée	ABCD hex	 _[1]
Max ^[2]	Nbre maxi. de ressources spécifiées affectables	ABCD hex	
Alloc ^[2]	Nbre de ressources spécifiées affectées	ABCD hex	
HWM ^[2]	Nbre maxi. de ressources affectées depuis le début	ABCD hex	
Free	Nbre de ressources spécifiées encore disponibles	ABCD hex	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Region	Zone mémoire avec ressource/mémoire sélectionnées	Menu	
String	8 octets ressource/mémoire en chaîne caractères	Alphanumérique	
Base_0 - 1E	32 octets ressource sélectionnée (ou zone mémoire)	ABCD hex	

[1] Lecture/écriture si Resource = MEM_DIAG [2] Sans objet si Resource = MEM_DIAG

Table 12-49 Paramètres du bloc RSRCDIAG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 12-49.

Resource. (BLOCK/TEMPLATE/LIBRARY/ENT/EDB/LATT/FEATT/TEATT/USERT/SERT/CONNECT/MEM_DIAG) Sélectionne la ressource (ou l'emplacement mémoire) à examiner. Les commandes du menu sont les suivantes:

- **BLOCK.** Blocs de la base de données.
- **TEMPLATE.** Modèles de la base de données.
- **LIBRARY.** Bibliothèques de la base de données.
- **ENT.** (*Non mis en oeuvre*)
- **EDB.** Liaisons de la base de données externe.
- **LATT.** (*Non mis en oeuvre*)
- **FEATT.** Liaisons provenant de blocs de la base de données externe.
- **TEATT.** Liaisons vers des blocs de la base de données externe.
- **USERT.** (*Non mis en oeuvre*).
- **SERT.** Serveurs.
- **CONNECT.** Liaisons de la base de données ("liaisons filaires").
- **MEM_DIAG.** Permet d'utiliser les champs *Segment* et *Offset* pour définir une adresse de début de zone mémoire à afficher dans les champs *Base_0* à *Base_1E*.
Notez que les champs *Segment* et *Offset* sont normalement en lecture seule.

SubType. (RESOURCE/HEADER/DYNAMIC/LOCAL/POOL/MEM_DIAG)

Sélectionne le sous-type de la ressource en question. Les commandes du menu sont les suivantes:

- **RESOURCE.** La structure réelle de la ressource.
- **HEADER.** L'en-tête du bloc/modèle.
- **DYNAMIC.** Zone de données dynamiques d'un bloc.
- **LOCAL.** Zone de données locales d'un bloc.
- **POOL.** Zone de données communes pour Block/Template.
- **MEM_DIAG.** Sélectionné automatiquement si *Resource* = 'MEM_DIAG'.

Index. Permet de sélectionner la *référence* d'un type de ressource particulier, qui a été sélectionné dans le champ *Resource*. Si, par exemple, vous sélectionnez **BLOCK** pour *Resource* et **4** pour *Index*, le contenu de l'indice 4 des ressources du bloc est affiché.

Timer. Si vous sélectionnez une valeur différente de zéro pour *Timer*, l'affichage fait défiler toutes les ressources affectées du type sélectionné dans les champs *Resource* et *SubType*, à un intervalle de secondes *Timer* par ressource. La remise à zéro de *Timer* arrête le processus de défilement.

Segment, Offset. Ces deux champs affichent l'adresse de début de la ressource sélectionnée. Si *Resource* = MEM_DIAG, ces champs passent en lecture/écriture et permettent de sélectionner une adresse de début pour une zone mémoire à afficher dans les champs *Base_0* à *Base_1E*.

Region. (ROM/RAM/E2ROM/UNUSED) Montre l'unité de mémoire dans laquelle la ressource sélectionnée réside. Lorsque le champ *Resource* a été sélectionné en tant que **MEM_DIAG**, *Region* montre l'emplacement de la zone mémoire sélectionnée. Les commandes du menu sont les suivantes:

- **ROM.** ROM (mémoire morte).
- **RAM.** RAM (mémoire vive).
- **E2ROM.** E²ROM (mémoire morte effaçable électriquement).
- **UNUSED.** Aucune unité de mémoire utilisée.

ISE_TUNE: BLOC DE MISE AU POINT DU MOTEUR DE STRATÉGIE INDUSTRIELLE

Fonction du bloc

Le bloc ISE_TUNE donne une utilisation approximative du processeur pour chaque fonction de l'ISE (Industrial Strategy Engine - moteur de stratégie industrielle, c'est à dire Visual Supervisor). Il donne également les temps de cycle des serveurs: S1 (Serveur 1) est le serveur de blocs locaux, S2 et S3 ne sont pas utilisés à l'heure actuelle, et S4 est le serveur des blocs en mémoire cache.

Paramètres du bloc

Les symboles utilisés dans la table 12-50 sont décrits au Chapitre 1.

Paramètre	Fonction	Unités	Etat
S1used-S4used	Millisec. utilisées dans cycles S1-S4 (normalement 100 ms)	Entier	
S1period-S4period	Périodes de mise à jour S1-S4 (millisecondes)	Entier	
S1delay-S4delay	Millisec. S1-S4 suspendus dernier cycle	Entier	
System	Utilisation UC tâches système (unités de 0,1 %)	Entier	
Other	Utilisation UC autres tâches (unités de 0,1 %)	Entier	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	T/F	
Combined	Combinaison de tous les bits d'alarme	T/F	
Bgnd	Utilisation UC tâche de fond (0,1 %)	Entier	
BatLoad	Utilisation UC chargeur batch (SFC+RECORD) (0,1 %)	Entier	
Config	Utilisation UC tâche configurateur terminal (0,1 %)	Entier	
Panel	Utilisation UC tâche face avant (unités de 0,1 %)	Entier	
Logging	Utilisation UC tâche de consignation (unités de 0,1 %)	Entier	
MODBUS_1 à 2	MODBUS_1, 2 utilisation UC tâches blocs DCM (0,1 %)	Entier	
PROFDP_1	PROFDP_1 utilisation UC tâche blocs DCM (unités de 0,1 %)	Entier	
Aux	(Non utilisé)	Entier	
CacheSrv	Utilisation UC serveur blocs cache (unités de 0,1 %)	Entier	
LocalSrv	Utilisation UC serveur blocs locaux (unités de 0,1 %)	Entier	
MODBUS_S	Utilisation UC tâches MODBUS esclave (unités de 0,1 %)	Entier	
NFS	Utilisation UC système archivage réseau (0,1 %)	Entier	
Load	Utilisation UC chargeur base données réseau (0,1 %)	Entier	
Network	Utilisation UC tâche réseau (unités de 0,1 %)	Entier	
xLIN	Utilisation UC tâche(s) SLIN ou ALIN (unités de 0,1 %)	Entier	
Tick	Utilisation UC tâche top XEC (unités de 0,1 %)	Entier	

Table 12-50 Paramètres du bloc ISE_TUNE

Chapitre 13 BLOCS DE FONCTION TAN

RS485_NODE : BLOC DE CONFIGURATION DU NOEUD TAN

Fonction du bloc

Le TAN (réseau asynchrone TCS) est une interface RS232/422 utilisant un bus semi-duplex RS485 pour transférer les données codées en format binaire entre un maximum de 31 noeuds de différents types. Le TAN comprend un contrôle de bus tournant piloté par un noeud de commande principal, un support modem et une communication entre homologues. Les dépassements de temps imparti configurables assurent la reprise du réseau si un paquet de données est perdu, et un noeud de commande secondaire de secours prend la relève en cas de défaillance du noeud principal.

Le bloc RS485_NODE permet de définir les informations de configuration - fonctionnement du noeud et dépassement du temps imparti - pour le TAN. Un bloc RS485_NODE unique doit d'abord être créé dans chaque instrument T1000 connecté au TAN avant de configurer les blocs de transmission et de réception. Il n'y a pas de limite prédéfinie pour le nombre de blocs de transmission et de réception par T1000, en dehors des considérations de performance. (Ces blocs - TX_AN, RX_AN, TX_DG et RX_DG sont décrits plus loin dans d'autres sections). Au cours de la configuration du bloc RS485_NODE, les champs du menu de spécifications sont en lecture/écriture, mais certains passent en lecture uniquement à l'exécution du bloc (sauf pour le champ alarmes).

Au cours de la conduite, le bloc active la tâche de communication dans le T1000, en configurant la vitesse du port, l'adresse du noeud, la parité, etc. Le bloc est également responsable de l'arrêt en séquence ordonnée du noeud de communication à la fin de la conduite.

Paramètres du bloc




Les symboles utilisés dans la table 13-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

Address. Adresse du noeud TAN du T1000 (1 à 31).

Online (En ligne). VRAI valide le T1000 pour les communications TAN.

Paramètre	Fonction	Unités	Etat*
Address	Adresse du noeud TAN T1000 (1 à 31)	Entier	
Online	Validé pour les communications TAN	V/F	
Class	Type administratif du noeud	Menu	
Max_Node	Adresse la plus haute du noeud interrogé (1 à 31)	Entier	
Node_Off	(Réservé)	Entier	
Speed	Vitesse en bauds du réseau (300 à 19200)	Menu	
Parity	Parité du réseau	Menu	
Retries	Nombre de relances pour acquittement (0 à 255)	Entier	
Alarms			  
Software	(Non mis en oeuvre)		
Hardware	(Non mis en oeuvre)		
Combined	(Non mis en oeuvre)		
Bus_Actv	Dépassement temps imparti actif du bus (sec.)	Entier	
Ctrl_Mon	Dépassement temps imparti actif par gestion contrôle (sec.)	Entier	
Poll_New	Temps de répétition interrogation nouveau venu (sec.)	Entier	
Mstr_Tak	Noeud prenant contrôle du bus, dépassement temps (sec.)	Entier	
Mstr_Kpt	Noeud tenant contrôle du bus, dépassement temps (sec.)	Entier	
Mstr_Rtn	Tributaire tenant contrôle du bus, dépassement temps (sec.)	Entier	
Rsp_Strt	Début attente réponse, dépassement du temps (sec.)	Entier	
Rsp_End	Fin attente réponse, dépassement du temps (sec.)	Entier	

*En lecture uniquement au cours de l'exécution du bloc

Table 13-1 Paramètres du bloc RS485_NODE

Class (Classe). (Tributary/Primary/Secondary - Tributaire/Primaire/Secondaire). Permet de spécifier le type administratif du noeud. Le noeud de commande primaire unique peut prendre le contrôle du bus RS485, (c'est à dire devenir le *noeud de commande actif*), établir l'existence de noeuds sur le réseau, et leur allouer le contrôle du bus. Le noeud de commande secondaire (unique) agit comme un noeud de secours pour le noeud primaire en cas de défaillance. Le noeud primaire a priorité sur le noeud secondaire en cas de contentieux pour le contrôle du bus. Un noeud tributaire est un noeud autre qu'un noeud de commande. Il ne peut activer la transmission des données jusqu'à ce qu'il y soit autorisé par le noeud de commande actif. Un noeud peut accéder au bus uniquement lorsqu'il détient le jeton - c'est à dire est le maître du bus - ou lorsqu'il répond à une demande.

Max_Node. (1 à 31). Permet de spécifier l'adresse la plus haute du noeud qui sera interrogé par le noeud de commande actif (au démarrage, à la remise à zéro du bus ou après une interrogation nouveau venu) pour déterminer quels noeuds sont présents sur le TAN. Mettre *Max_Node* à l'adresse la plus haute du noeud considéré (elle ne peut être modifiée en cours de conduite) pour empêcher une interrogation inutile et ne pas gaspiller le temps réseau. Il faut noter que *Max_Node* devrait être mis à la même valeur dans les noeuds primaires et secondaires, mais sa valeur pour les noeuds tributaires est "indifférente".

Node_Off. (Champ réservé).

Speed (Vitesse). Permet de spécifier la vitesse en bauds du réseau.

Parity (Parité). (None/Odd/Even - Aucune/Impaire/Paire). Permet de spécifier la parité du réseau. Utiliser de préférence la parité paire.

Retries (Relances). (0 à 255). Permet de spécifier le nombre maximum de relances autorisées à un noeud maître pour obtenir un acquittement d'un noeud esclave.

Alarms (Alarmes). Voir la description général du champ alarmes à la page 2-5.

■ **Software (Logiciel).** (*Non mis en oeuvre*)

■ **Hardware (Matériel).** (*Non mis en oeuvre*)

■ **Combined (Combinée).** (*Non mis en oeuvre*)

Bus_Actv. Bus actif. Le noeud de commande non-actif essaie de prendre le contrôle du bus, si aucun caractère n'a été transmis après *Bus_Actv* secondes.

Ctrl_Mon. Gestion du contrôle. Le noeud de commande non-actif essaie de prendre le contrôle du bus, si aucun caractère n'a été transmis après *Ctrl_Mon* secondes.

Poll_New. Interrogation nouveaux venus. Permet de spécifier la fréquence à laquelle le noeud de commande actif émet les demandes d'interrogation nouveaux venus, lorsqu'un noeud nouvellement connecté au TAN est reconnu.

Mstr_Tak. Prise de contrôle. Le noeud de commande actif, après avoir passé le statut de maître à un tributaire, va essayer de reprendre le contrôle du bus, si le tributaire n'a pas transmis de caractère dans l'intervalle *Mstr_Tak*. (Le caractère peut être XSYN). Dans ce cas, le noeud de commande actif passe le jeton au noeud suivant et met le tributaire qui ne répond pas hors ligne.

Mstr_Kpt. Captation du contrôle. Le noeud de commande actif, après avoir passé le statut de maître à un tributaire, va essayer de reprendre le contrôle du bus, si le tributaire a gardé le contrôle pendant plus de *Mstr_Kpt* secondes.

Mstr_Rtn. Conservation du contrôle. Après avoir reçu le statut de maître, le noeud active sa propre horloge pour limiter son temps de contrôle pour qu'il ne dépasse pas *Mstr_Rtn* secondes. (Si ce temps est dépassé, l'horloge *Mstr_Kpt* du noeud de commande actif peut se déclencher et ôter le contrôle du bus).

Rsp_Strt. Début de la réponse. Un esclave (c'est à dire tout noeud qui n'est pas le maître) qui a reçu une demande doit commencer à transmettre la réponse dans l'intervalle *Rsp_Strt* secondes. Il faut noter que la valeur par défaut est 1 seconde, ce qui n'est peut-être pas suffisant pour les configurations importantes. Pour les noeuds T1000, il est recommandé de mettre la valeur à 2 secondes.

Rsp_End. Fin de la réponse. Un esclave (c'est à dire tout noeud qui n'est pas le maître) qui a commencé à transmettre une réponse doit achever sa réponse dans l'intervalle *Rsp_End* secondes. Dans le cas contraire, le maître supposera que l'esclave est hors ligne.







TX_AN : BLOC DE TRANSMISSION ANALOGIQUE

Fonction du bloc

Chaque bloc TX_AN permet de transmettre un maximum de huit signaux analogiques d'un schéma de boucles sur le réseau asynchrone TCS RS485 (TAN) à des paramètres de bloc spécifiés en un noeud de destination unique. Les blocs TX_AN peuvent uniquement être configurés dans un schéma qui contient déjà un bloc RS485_NODE configuré. Il faut noter que les blocs TX_AN ne nécessitent pas de blocs spéciaux de "réception" correspondants sur le TAN.

Paramètres du bloc

Les symboles utilisés dans la table 13-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat*
XNode	Adresse du noeud de la base de données de destination	Entier	
Conn_1 à Conn_8	Drapeaux communication TAN établie	V/F	
Value_1 à Value_8	Valeurs des paramètres transmises sur le TAN	Eng	
XBlock_1 à XBlock_8	Noms de repère des blocs de destination	Alphanumérique	
XField_1 à XField_8	Noms des points dans les blocs de destination	Alphanumérique	
Alarms			  
Software	Déconnecté du réseau TAN (défaut communicat.)		
Offline	(Non mis en oeuvre)		
Timeout	(Non mis en oeuvre)		
Combined	(Non mis en oeuvre)		
Scan	Période de mise à jour TAN du bloc (0 à 65535 secondes)	Entier	

* Au cours de la conduite

Table 13-2 Paramètres du bloc TX_AN

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs "en-tête" à la page 2-3.

XNode. Adresse du noeud TAN (1 à 31) de la base de données de destination.

Conn_1 à Conn_8. VRAI lorsque la communication TAN a été établie avec succès pour la transmission des paramètres *Value_1* à *Value_8*. FAUX indique un défaut de communication sur cette voie quelle qu'en soit la raison (par ex. nom de repère ou nom de point dans le bloc de destination mal orthographié, noeud hors ligne, ligne installations très bruyante, etc.).

Value_1 à Value_8. Valeurs des paramètres transmises sur le TAN, calculées soit à partir des liaisons du schéma de boucles ou de valeurs entrées directement (point flottant analogique à 32 bits).

XBlock_1 à XBlock_8. Noms de repère (champs *Bloc*) du bloc de destination déporté auquel les paramètres *Value_1* à *Value_8* sont transmis sur le TAN.

XField_1 à XField_8. Mnémoniques des paramètres de destination déportée auxquels les paramètres *Value_1* à *Value_8* sont transmis sur le TAN. (Par ex. *PV*, *HR_SP*, etc.).

Alarms (Alarmes). Voir la description général du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Bloc non connecté au TAN.

■ **Offline (Hors ligne).** (*Non mis en oeuvre*)

■ **Timeout (Dépassement temps imparti).** (*Non mis en oeuvre*)

■ **Combined (Combinée).** (*Non mis en oeuvre*)

Scan (Echantillonnage). Intervalle de mise à jour du bloc, c'est à dire transmet les valeurs actives au noeud déporté sur le TAN (0 à 65535 secondes). Si *Scan* est mis à zéro, le temps d'échantillonnage est maximal.

TX_DG : BLOC DE TRANSMISSION LOGIQUE

Fonction du bloc

Chaque bloc TX_DG permet de transmettre un maximum de huit signaux logiques d'un schéma de boucles sur le réseau asynchrone TCS RS485 (TAN) à des paramètres de bloc spécifiés en un noeud de destination unique. Les blocs TX_DG peuvent uniquement être configurés dans un schéma qui contient déjà un bloc RS485_NODE configuré. Il faut noter que les blocs TX_DG ne nécessitent pas de blocs spéciaux de "réception" correspondants sur le TAN.

Paramètres du bloc

Les paramètres du bloc TX_DG sont identiques à ceux du bloc TX_AN, détaillés à la page 13-4, sauf pour les liaisons et les formats d'entrée du paramètre *Value_1* à *Value_8*. Dans le bloc TX_DG, ces paramètres ont des liaisons d'entrée logiques et les formats VRAI/FAUX.

Menu de spécifications du bloc

Le menu de spécifications du bloc TX_DG est identique à celui du bloc TX_AN, détaillé à la page 13-4 (aux exceptions près ci-dessus).

RX_AN : BLOC DE RECEPTION ANALOGIQUE

Fonction du bloc

Chaque bloc RX_AN permet de recevoir un maximum de huit signaux analogiques par le réseau asynchrone TCS RS485 (TAN) de paramètres de bloc spécifiés en un noeud source unique. Les signaux sont reçus dans les paramètres *Value_1* à *Value_8* du bloc RX_AN qui peuvent faire passer des valeurs dans le schéma de boucles local par l’intermédiaire des liaisons du bloc. Les blocs RX_AN peuvent uniquement être configurés dans un schéma de boucles contenant déjà un bloc RS485_NODE configuré. Noter que les blocs RX_AN ne nécessitent pas de blocs spéciaux de “transmission” correspondants sur le TAN.

Paramètres du bloc

Les symboles utilisés dans la table 13-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat*
XNode	Adresse du noeud de la base de données source (1 - 31)	Entier	
Conn_1 à Conn_8	Drapeaux communication TAN établie	V/F	
Value_1 à Value_8	Valeurs des paramètres reçues sur le TAN	Eng	
XBlock_1 à XBlock_8	Noms de repère des blocs source	Alphanumérique	
XField_1 à XField_8	Noms des points dans les blocs source	Alphanumérique	
Alarms			
Software	Déconnexion du TAN (défaut communicat.)		
Offline	(Non mis en oeuvre)		
Timeout	(Non mis en oeuvre)		
Combined	(Non mis en oeuvre)		
Scan	Période de mise à jour TAN du bloc (0 à 65535 secondes)	Entier	

* Au cours de la conduite

Table 13-3 Paramètres du bloc RX_AN

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails sur ces champs “en-tête” à la page 2-3.
- XNode.** Adresse du noeud TAN (1 à 31) de la base de données source.
- Conn_1 à Conn_8.** VRAI lorsque la communication TAN a été établie avec succès pour la réception des paramètres *Value_1* à *Value_8*. FAUX indique un défaut de communication sur cette voie quelle qu'en soit la raison (par ex. nom de repère ou nom de point dans le bloc source mal orthographié, noeud hors ligne, ligne installations très bruyante, etc.).

Value_1 à Value_8. Valeurs des paramètres reçues sur le TAN, ou de valeurs entrées directement, si aucune liaison TAN n'est établie (point flottant analogique à 32 bits).

XBlock_1 à XBlock_8. Noms de repère (champs *Bloc*) du bloc source déporté à partir duquel les paramètres *Value_1* à *Value_8* sont reçus sur le TAN.

XField_1 à XField_8. Mnémoniques des paramètres de source déportée à partir desquels les paramètres *Value_1* à *Value_8* sont reçus sur le TAN. (Par ex. *PV*, *HR_SP*, etc.).

Alarms (Alarmes). Voir la description général du champ alarmes à la page 2-5.

■ **Software (Logiciel).** Bloc non connecté au TAN.

■ **Offline (Hors ligne).** (*Non mis en oeuvre*)

■ **Timeout (Dépassement temps imparti).** (*Non mis en oeuvre*)

■ **Combined (Combinée).** (*Non mis en oeuvre*)

Scan (Echantillonnage). Intervalle de mise à jour du bloc, c'est à dire reçoit les valeurs actives au noeud déporté sur le TAN (0 à 65535 secondes). Si *Scan* est mis à zéro, le temps d'échantillonnage est maximal.

RX_DG : BLOC DE RECEPTION LOGIQUE

Fonction du bloc

Chaque bloc RX_DG permet de recevoir un maximum de huit signaux logiques par le réseau asynchrone TCS RS485 (TAN) de paramètres de bloc spécifiés en un noeud source unique. Les signaux sont reçus dans les paramètres *Value_1* à *Value_8* du bloc RX_DG qui peuvent faire passer des valeurs dans le schéma de boucles local par l'intermédiaire des liaisons du bloc. Les blocs RX_DG peuvent être configurés uniquement dans un schéma de boucles contenant déjà un bloc RS485_NODE. Noter que les blocs RX_DG ne nécessitent pas de blocs spéciaux de "transmission" correspondants sur le TAN.

Paramètres du bloc

Les paramètres du bloc RX_DG sont identiques à ceux du bloc RX_AN, détaillés à la page 13-7, sauf pour les liaisons et les formats d'entrée du paramètre *Value_1* à *Value_8*. Dans le bloc RX_DG, ces paramètres ont des liaisons d'entrée logiques et les formats VRAI/FAUX.

Menu de spécifications du bloc

Le menu de spécifications du bloc RX_DG est identique à celui du bloc RX_AN, détaillé à la page 13-7 (aux exceptions près ci-dessus).

TAN_DIAG : BLOC DE DIAGNOSTIC TAN

Fonction du bloc

Le bloc TAN_DIAG contrôle les performances du TAN, en recueillant différentes statistiques de fonctionnement et d'erreurs. Il n'a ni liaisons d'entrée ni liaisons de sortie, et n'a pas besoin d'être présent pour le fonctionnement du TAN.

Paramètres du bloc

Les symboles utilisés dans la table 13-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités*	Etat*
Parity	Comptage des erreurs de parité		
Overrun	Comptage des erreurs de surcharge		
Framing	Comptage des erreurs d'encadrement		
Csum	Comptage des erreurs de contrôle de données		
Protocol	Comptage des erreurs internes du logiciel		
Transact	Comptage des transactions noeud (<i>Non mis en oeuvre</i>)		
Act_Node	Adresses du noeud actif (en ligne)		
Err_Node	(<i>Réservé</i>)		
BA_Timer	Comptage des dépassements temps imparti bus actif		
CM_Timer	Comptage des dépassements temps imparti par la gestion du contrôle		
PN_Timer	} Comptage des dépassements temps imparti (<i>Non mis en oeuvre</i>)		
MT_Timer			
MK_Timer			
MR_Timer			
RS_Timer	Comptage des dépassements temps imparti début attente de réponse		
RC_Timer	Comptage des dépassements temps imparti fin attente de réponse		

* Tous les paramètres du bloc TAN_DIAG ont le format entier et le statut de lecture uniquement.

Table 13-4 Paramètres du bloc TAN_DIAG

Menu de spécifications du bloc

Les informations suivantes sont données en plus de la table 13-4.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Act_Node. Indique l'ensemble des adresses de noeud vues par ce noeud comme contribuant au réseau (c'est à dire en ligne). *Act_Node* est affiché comme une valeur décimale donnée par :

$$\sum_n 2^{n-1}$$

où n représente toutes les adresses de noeud en ligne (dans la plage de 1 à 31). Par ex., si les adresses en ligne sont $n = 1, 4$ et 5 , alors $Act_Node = 2^0 + 2^3 + 2^4 = 25$ (décimal).

BA_Timer. Compte le nombre de dépassements de temps imparti du bus actif. (Voir les détails des paramètres de dépassement de temps imparti du TAN dans le *Menu de spécification du bloc RS485_NODE*). Les comptages *BA_Timer* ne sont pertinents que pour les noeuds primaires ou secondaires, ils peuvent être ignorés pour les noeuds tributaires.

[Page laissée intentionnellement blanche]

Chapitre 14 BLOCS DE FONCTION LOT

SFC_CON : BLOC DE COMMANDE SFC

Fonction du bloc

Les blocs de commande SFC (graphe de fonctions séquentielles) permettent de sélectionner et d'exécuter les séquences au cours de la conduite du schéma de boucles principal.

Les signaux logiques liés au bloc à partir du schéma de boucles peuvent charger un fichier de séquence (spécifié dans *FileName* - Nom de Fichier) dans la RAM (*Load* - Charger), exécuter la séquence chargée (*Run* - Exécuter), interrompre la séquence en permettant aux actions en cours de se poursuivre (*Hold* - Maintenir) ou la ré-initialiser (*Init* - Initialisation). Le paramètre *Filepath* (chemin du fichier) permet de spécifier un noeud LIN et une unité de disque, lorsque la source du fichier de la séquence est déportée.

Le bloc SFC_CON fournit des sorties d'alarmes qui signalent les erreurs de chargement de fichiers (*LoadFile*), des demandes d'exécution d'actions qui sont déjà en cours d'exécution (*Actions*), signale des surcharges de mémoire vive (RAM) potentiellement gênantes dues au traitement de la séquence (*Overload*), et également un bit de "validation" (*Clear*) qui passe uniquement à l'état bas lorsqu'un fichier de séquence est chargé avec succès.

Priorités d'entrée

(Voir les détails sur les paramètres d'entrée dans la section *Menu de spécifications du bloc* ci-après).

- *Load* (Charger) a la priorité 1 (la plus haute). Si Load passe à FAUX, la séquence s'arrête et est effacée de la RAM quels que soient les états des autres entrées.
- *Run* (Exécuter) a la priorité 2 et son état détermine si *Hold* (Maintenir) et *Init* (Initialisation) ont un effet ou non. *Init* n'a pas d'effet lorsque *Run* est VRAI ; *Hold* n'a pas d'effet lorsque *Run* est FAUX.
- *Hold* (Maintenir) a la priorité 3. Son état contrôle l'exécution des transitions qui ne peuvent se produire que lorsque la séquence est en cours d'exécution. Lorsque la séquence est arrêtée, *Hold* n'a aucun effet.
- *Init* (Initialiser) a également la priorité 3. Son état contrôle la ré-initialisation de la séquence, mais uniquement lorsque *Run* est FAUX.

Contrôle de la ré-initialisation : le paramètre Init

Le maintien de *Init* à VRAI signifie que chaque fois que la séquence est arrêtée, elle est ré-initialisée ; c'est à dire que toutes les étapes sont désactivées, les actions qualifiées "F" sont exécutées et les horloges mises à zéro. Si la séquence est relancée, elle recommence à l'étape ou aux étapes initiale(s).

Le maintien de *Init* à FAUX signifie que les étapes actives restent actives et les horloges ne sont pas remises à zéro lorsque la séquence est arrêtée, mais sont “gelées”. Les actions qualifiées “F” ne sont pas exécutées lorsque la séquence s’arrête. Si la séquence est relancée, elle reprend là où elle était restée, les mêmes étapes restant actives et les horloges continuant tout simplement. Les étapes relancées ne ré-exécutent pas les actions qualifiées “P”.

Le passage de *Init* de FAUX à VRAI - la séquence étant arrêtée - ré-initialise la séquence, mais les actions qualifiées “F” dans les étapes désactivées ne sont pas exécutées cette fois. Le retour de *Init* de VRAI à FAUX après une ré-initialisation ne ramène pas la séquence là où elle était restée. Elle reprend toujours à l’étape ou aux étapes initiale(s), si *Init* est VRAI tandis que *Run* est FAUX.

Paramètres du bloc

Les symboles utilisés dans la table 14-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.








Paramètre	Fonction	Unités	Etat
Run	Lance/continue l’exécution de la séquence	V/F	
Hold	Maintient la séquence à l’étape/aux étapes active(s)	V/F	
Init	Retour à l’étape/aux étapes initiale(s) (si Run FAUX)	V/F	
Load	Charge la séquence spécifiée dans la RAM	V/F	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Clear	Bit “Validation” chargement fichier (bas=chargement OK)		V/F
LoadFile	Erreur de chargement du fichier (matériel/logiciel)	V/F	
Actions	Demande d’exécution d’une action en cours d’exécution	V/F	
Overload	Surcharge de la RAM par traitement de la séquence	V/F	
Combined	Alarme combinée du bloc	V/F	
FileName	Nom du fichier de la séquence à charger		Alphanumérique
Filepath	Localisation LIN du fichier spécifié dans Filename		Alphanumérique
DispBlk	Nom du bloc SFC_DISP lié		Alphanumérique

Table 14-1 Paramètres du bloc SFC_CON

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

Run (Exécution). Lorsque *Run* est VRAI, la séquence chargée dans la RAM est exécutée - c’est à dire a une ou des étapes active(s) - ou reprend l’exécution. (Le paramètre *Init* détermine si la séquence est ré-initialisée ou non). Lorsque *Run* est FAUX, l’exécution de la séquence est arrêtée. Il faut noter que l’entrée *Load* peut annuler *Run*.

Hold (Maintien). Lorsque *Hold* est VRAI, une séquence en cours d’exécution poursuit l’exécution, mais les transitions d’une étape à l’autre sont bloquées, même si l’état de la transition est VRAI. (*Hold* n’a aucun effet lorsque *Run* est FAUX).

Init (Initialisation). Le passage de *Init* à VRAI ramène la séquence à l'étape ou aux étapes initiale(s) si *Run* est ou passe à FAUX. Si *Init* et *Run* sont VRAIS, l'initialisation est validée, mais pas exécutée (sauf si *Run* passe à FAUX).

Load (Chargement). Un front montant - une transition FAUX à VRAI - entré dans *Load* copie la séquence spécifiée par les paramètres *FileName* et *Filepath* dans la RAM, où elle peut être contrôlée par d'autres entrées de bloc. *Load* doit rester VRAI pour conserver la séquence chargée dans la RAM ; le passage à FAUX décharge la séquence et efface la mémoire. L'entrée *Load* annule toutes les autres.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** VRAI si la mémoire est corrompue ou si le bloc image est hors ligne.
- **Clear (Effacement).** Passe à l'état haut, lorsqu'aucune séquence n'est chargée dans la RAM, mais est ré-initialisé lorsqu'une séquence est chargée avec succès. Peut être utilisé comme un bit de "validation" pour l'opération de chargement.
- **LoadFile (Chargement fichier).** Indique les erreurs de chargement de fichiers provoquées soit par des défauts du matériel (par ex. une unité défectueuse) ou du logiciel (par ex. nom de fichier ou chemin inexistants).
- **Actions (Actions).** Indique une demande interdite pour lancer une action qui est déjà en cours d'exécution. (Par ex. une action de qualificatif S répétée dans une étape). Il faut noter qu'il n'est pas interdit de demander la fin d'une action déjà arrêtée.
- **Overload (Surcharge).** Indique que la RAM est surchargée par le traitement de la séquence, c'est à dire que la séquence est trop "active". Dans ces conditions, le comportement peut être affecté, comme par ex. des actions ou transitions non exécutées.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

FileName (Nom de fichier). Nom de fichier de l'utilisateur (huit caractères maximum) spécifiant le fichier de séquence à charger dans la RAM au cours de la conduite. *FileName* peut être modifié au cours de la conduite, à condition que la séquence ne soit pas chargée (*Load* = FAUX).

Filepath (Chemin du fichier). Spécifie l'emplacement du fichier de séquence désigné dans le paramètre *FileName*. *Filepath* passe par défaut à "local, unité par défaut" (champ vierge), qui est l'unité M: pour un T1000 et E: pour un T100. *Filepath* peut comprendre la lettre d'une unité, un numéro de noeud ou les deux, par ex.

M: Unité virtuelle de l'instrument local,
ED: : Noeud ED LIN, unité par défaut,
ED: : B: Noeud ED LIN, unité DTU du bas.

DispBlk (Bloc affichage). Nom de repère du bloc SFC_DISP (le cas échéant) lié au bloc SFC_CON. Le bloc SFC_DISP permet d'afficher/de superviser/contrôler une séquence exécutée dans un instrument déporté sur l'écran de séquence de conduite du T1000 local.

Le champ *DspBlk* est chargé automatiquement avec le nom de repère du bloc SFC_DISP lié, lorsque la liaison est effectuée depuis le bloc SFC_DISP, et effacé automatiquement lorsque la liaison est rompue. En outre, la saisie d'un nom de repère dans *DispBlk* introduit automatiquement le nom de repère correspondant dans le champ *SfcBlock* du bloc SFC_DISP pour compléter la liaison. (Voir les détails dans la section Bloc SFC Display à la page 14-7).

SFC_MON : BLOC DE SUPERVISION SFC

Fonction du bloc

Les blocs de supervision SFC (graphe de fonctions séquentielles) permettent la supervision de séquence au cours de la conduite du schéma de boucles principal. Chaque bloc SFC_MON peut superviser un maximum de huit étapes dans la séquence contrôlée par le bloc SFC_CON correspondant (désigné dans le paramètre *SfcBlock*). Il n’y a pas de limite autre que celle de la RAM disponible en ce qui concerne le nombre de blocs SFC_CON et SFC_MON dans un schéma de boucles. Pour les séquences de plus de huit étapes, des blocs SFC_MON supplémentaires peuvent être créés, si nécessaire.

Le bloc SFC_MON dispose de huit trios de paramètres qui indiquent le nom alphanumérique de l’étape (*Step_A* à *Step_H*), que l’étape soit active ou non (valeurs logiques *A_X* à *H_X*), et le temps (secondes) que l’étape a été active (valeurs analogiques *A_T* à *H_T*). Les huit valeurs logiques et analogiques sont disponibles comme des sorties de lectures uniquement dans le schéma de boucles. Le bloc n’a pas de liaisons d’entrée.

Le bloc dispose d’une sortie d’alarme *SfcBlock* spéciale qui signale si le nom de repère du bloc SFC_CON défini dans le champ *SfcBlock* du bloc SFC_MON est incorrect pour une raison ou pour une autre (par ex. type inexistant ou référence à un type de bloc erroné).

Paramètres du bloc

Les symboles utilisés dans la table 14-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
SfcBlock	Nom de repère bloc SFC_CON correspondant	Alphanumérique	
Step_A à Step_H	Noms des étapes supervisées	Alphanumérique	
A_X à H_X	Etats d’activité des étapes A à H (VRAI=actif)	V/F	
A_T à H_T	Temps d’exécution (sec) des étapes A à H	Eng	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
SfcBlock	Spécification de champ SfcBlock incorrect	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 14-2 Paramètres du bloc SFC_MON

Menu de spécifications du bloc

- Dbase, Block, Type.** Voir les détails de ces champs “en-tête” à la page 2-3.
- SfcBlock.** “Nom de repère” (champ *bloc*) du bloc SFC_CON qui charge et contrôle la séquence supervisée.

Step_A à Step_H (Etape_A à Etape_H). Noms des huit étapes (maximum) dans la séquence supervisée à laquelle les paramètres d'activité (*A_X* à *H_X*) et de temps correspondants (*A_T* à *H_T*) renvoient.

A_X à H_X. VRAI si l'étape correspondante (*Etape_A* à *Etape_H*) est active. Il faut noter que, pour une étape active, le paramètre reste VRAI lorsqu'une séquence est arrêtée (*Init FAUX*, *Run FAUX*), mais passe à FAUX lorsque la séquence est RE-INITIALISEE (*Init VRAI*, *Run FAUX*).

A_T à H_T. Indique la période en secondes pendant laquelle l'étape correspondante (*Etape_A* à *Etape_H*) est restée active. Il faut noter que, pour une étape active, le paramètre est "gelé" lorsque la séquence est ARRETEE (*Init FAUX*, *Run FAUX*), mais est mis à zéro lorsque la séquence est RE-INITIALISEE (*Init VRAI*, *Run FAUX*).

NOTA. Comme alternative à ces paramètres, "T" peut être utilisé dans une instruction de texte structuré ou dans une transition pour indiquer la période d'activité de l'étape sans référence à un bloc SFC_MON. Par ex. **REACT.T** représente la période d'activité en secondes pour l'étape appelée "REACT".

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** VRAI si la mémoire est corrompue ou si le bloc image est hors ligne.
- **SFcBlock.** Indique que le champ *SfcBlock* est incorrect, par ex. désigne un bloc SFC_CON inexistant.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

SFC_DISP : BLOC SFC DISPLAY (AFFICHAGE)








Fonction du bloc

Le bloc SFC Display permet d’afficher graphiquement, de superviser et de contrôler une séquence exécutée sur un instrument déporté sur un T1000 local.

Le bloc SFC_DISP et le bloc SFC_CON sont exécutés dans l’instrument déporté - reliés par leurs paramètres *DispBlk* et *SfcBlock* respectifs. Dans le T1000 local, seul le bloc SFC_CON déporté doit être mis en image. Lorsque la séquence déportée est sélectionnée en visualisation au cours de la conduite, le bloc SFC_DISP déporté commence l’exécution et est automatiquement mis en image dans le T1000 local, ce qui permet aux graphiques de la séquence d’être visualisés à l’écran. Lorsque l’écran de séquence de conduite n’est plus affiché, l’exécution du bloc SFC_DISP est arrêtée.

Paramètres du bloc

Les symboles utilisés dans la table 14-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
SfcBlock	Nom de repère du bloc SFC_CON lié	Alphanumérique	
ActStepA à E	Noms des étapes actives (5 maxi. affichés)	Alphanumérique	
StepName*	Spécifie le nom de l’étape existante	Alphanumérique	
SAtt Out*	Spécifie les attributs à modifier (masque bit)	Numérique	
SAtt In*	Spécifie les valeurs d’attributs nécessaires (masque bit)	Numérique	
SDone*	Fin drapeaux VRAI de l’opération d’écriture	V/F	
Token	Nom de l’étape maintenant le jeton TRAK	Alphanumérique	
Alarms			  
Software	Données corrompues/Défaut de communication	V/F	
Bad_Seq	<i>(Non mis en oeuvre dans cette version)</i>		
More_Sts	Plus de 5 étapes actives	V/F	
More_Act	Plus de 4 actions actives	V/F	
Clear	Pas de séquence chargée	V/F	
Combined	Alarme combinée du bloc	V/F	
AAActionA à D	Noms des actions actives (4 maxi. affichées)		
TrActIx*	} Spécifie la transition à adresser		
TrTrIx*			
TAtt Out*	Spécifie les attributs à modifier (masque bit)	Numérique	
TAtt In*	Spécifie les valeurs d’attributs nécessaires (masque bit)	Numérique	
TDone*	Fin drapeaux VRAI de l’opération d’écriture	V/F	
ClrHlds	VRAI efface tous les HOLDS, ENABLES, DISABLES	V/F	
Running	VRAI exécute sous-programme de MAJ bloc SFC_DISP	V/F	

* A usage EUROTHERM SYSTEMES uniquement

Table 14-3 Paramètres du bloc SFC_DISP

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails de ces champs “en-tête” à la page 2-3.

SfcBlock (Bloc Sfc). Nom de repère (champ *bloc*) du bloc SFC_CON lié au bloc SFC_DISP. A tout moment, uniquement un seul bloc SFC_CON peut être lié à un bloc SFC_DISP donné et vice-versa. Le logiciel s’en assure automatiquement au moment de la configuration de la manière suivante. Lorsqu’un nom de repère est saisi dans le champ *SfcBlock*, tout lien conflictuel existant est effacé (avec des champs nuls) et le champ *DispBlk* du bloc SFC_CON désigné est écrit en établissant une liaison à deux voies. Des actions automatiques correspondantes se produisent également si la liaison est effectuée à partir du champ *DispBlk* du bloc SFC_CON.

ActStepA à ActStepE (Etape action A à Etape action E). Noms de cinq (maximum) étapes actives dans la séquence supervisée affichés sans ordre précis. Pour une séquence exécutée à distance, un maximum de cinq étapes peuvent être affichées à la fois par l’intermédiaire de la touche multifonctions FIND [ACTIVE STEP (ETAPE ACTIVE)] de l’écran de séquence de conduite. Cette limitation ne s’applique pas à une séquence exécutée localement.

Token (Jeton). Nom de l’étape (le cas échéant) qui détient le jeton TRAK, c’est à dire positionné sous le curseur dans l’écran de séquence de conduite, la fonction TRAK étant activée.

NOTA. La fonction TRAK permet au curseur de suivre l’étape active à mesure qu’il progresse dans le graphe de fonctions séquentielles affiché.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** VRAI si la mémoire est corrompue ou si le bloc image est hors ligne.
- **Bad_Seq (Mauvaise séquence).** *(Non mis en oeuvre dans cette version).*
- **More_Sts.** VRAI si plus de cinq étapes sont actives, et par conséquent, toutes ne peuvent être affichées dans les champs *ActStepA* à *E*.
- **More_Act.** VRAI, si plus de quatre actions sont actives, et par conséquent, ne peuvent toutes être affichées dans les champs *AActionA* à *D*.
- **Clear (Effacement).** VRAI, si aucune séquence n’est chargée (par l’intermédiaire du bloc SFC_CON).
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d’état et numéro de priorité que l’alarme active du bloc qui a la priorité la plus haute.

AActionA à AActionD. Noms de quatre actions actives (maximum y compris ROOT (RACINE)) dans la séquence supervisée, listés sans ordre précis. Pour une séquence exécutée à distance, un maximum de quatre actions peuvent être affichées à la fois par l’intermédiaire de la touche multifonctions FIND [ACTIVE ACTION (ACTION ACTIVE)] de l’écran de séquence de conduite. Cette limitation ne s’applique pas à une séquence exécutée localement.

ClrHlds. Si *ClrHlds* passe à VRAI, toutes les commandes HOLD STEP, ENABLE ou DISABLE TRANSITION (MAINTIEN ETAPE, VALIDE OU INVALIDE TRANSITION) appliquées aux étapes ou transitions par l'intermédiaire de l'écran de séquence de conduite sont effacées. *ClrHlds* est ré-initialisé automatiquement après usage.

Running (Exécution). Doit être VRAI pour le bloc SFC_DISP à mettre à jour. *Running* passe automatiquement à VRAI, chaque fois qu'une séquence déportée est sélectionnée pour la visualisation en conduite, et repasse à FAUX, lorsque l'écran de séquence de conduite n'est plus actif.

RECORD : BLOC ENREGISTREMENT

Fonction du bloc

Le bloc RECORD enregistre et récupère des ensembles de valeurs analogiques et logiques à utiliser dans les schémas de boucles et les séquences - comme des recettes par exemple. Il peut contenir un maximum de huit valeurs analogiques et seize valeurs logiques (dans les paramètres *A0 - A7* et *Bit0 - Bit15*). Le bloc peut être déclenché (par l'entrée *Get*) - au cours de la conduite ou de la configuration - pour charger de nouveaux ensembles de valeurs dans ses paramètres à partir de fichiers d'enregistrements spéciaux, et également pour sauvegarder des ensembles complets de valeurs dans ces fichiers (par l'entrée *Put*).

Fichiers d'enregistrement

Les fichiers d'enregistrement sont créés automatiquement par le bloc RECORD (nommé par son paramètre *Filename* plus l'extension *.RCD*) et peuvent contenir un maximum de 50 ensembles complets de valeurs - les 'enregistrements' (records). Le paramètre *RecordNo* qui peut être une entrée d'un schéma spécifie le numéro de l'enregistrement à charger/sauvegarder. Les enregistrements sont numérotés de 0 à 49 dans chaque fichier d'enregistrement, et chaque enregistrement comprend une légende texte de huit caractères maximum.

Un fichier d'enregistrement préparé avec une série d'enregistrements peut être utilisé pour mettre en oeuvre un mécanisme de contrôle par lots. Une partie des données de chaque enregistrement peut être interprétée comme un numéro de procédé (qui définit la séquence à exécuter), le reste des données étant des paramètres pour ce procédé particulier. Le déroulement des enregistrements en série (par le paramètre *GetNext*) exécutera alors une série prédéfinie de lots.

Collecte des données de conduite

Il faut noter que le bloc peut à la fois entrer des données depuis le schéma de boucles et les sortir dans le schéma de boucles - une fonction importante qui permet au bloc RECORD de collecter, de sauvegarder et d'utiliser des valeurs à mesure que le schéma de boucles est exécuté (par ex. nouvelles recettes, résultats et statistiques du procédé).

Le bloc dispose d'une sortie d'alarme spéciale, *File* (fichier), qui signale les noms de fichiers, les numéros d'enregistrement erronés et les erreurs de chargement/sauvegarde matériel/logiciel.

Paramètres du bloc

Les symboles utilisés dans la table 14-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Caption	Identificateur Enregistrement défini (sauvegardé) par l'utilisateur	Alphanumérique	
A0 à A7	Valeurs analogiques	Eng	
D	Valeurs logiques	ABCD hex	
Bit0	Valeur logique 1	V/F	1
Bit1	Valeur logique 2	V/F	2
Bit2	Valeur logique 3	V/F	4
Bit3	Valeur logique 4	V/F	8
Bit4	Valeur logique 5	V/F	1
Bit5	Valeur logique 6	V/F	2
Bit6	Valeur logique 7	V/F	4
Bit7	Valeur logique 8	V/F	8
Bit8	Valeur logique 9	V/F	1
Bit9	Valeur logique 10	V/F	2
Bit10	Valeur logique 11	V/F	4
Bit11	Valeur logique 12	V/F	8
Bit12	Valeur logique 13	V/F	1
Bit13	Valeur logique 14	V/F	2
Bit14	Valeur logique 15	V/F	4
Bit15	Valeur logique 16	V/F	8
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
File	Nom de fichier/enregistrement erroné ou lecteur défectueux/manquant	V/F	
Combined	Alarme combinée du bloc	V/F	
FileName	Nom de fichier de l'enregistrement pour les opérations charge/sauvegarde	Alphanumérique	
FilePath	Adresse du noeud LIN et/ou code du lecteur du fichier d'enregistrement	Alphanumérique	
RecordNo	Numéro d'enregistrement du nom de fichier (0-49)	Eng*	
Get	Charge les valeurs d'un bloc Record depuis un enregistrement spécifié	V/F	
GetNext	Incrémente RecordNo & Get Record	V/F	
Put	Copie les valeurs du bloc Record dans un enregistrement spécifié	V/F	
GetOK	Lit drapeau OK (sinon alarme fichier)	V/F	
PutOK	Ecrit drapeau OK (sinon alarme fichier)	V/F	

* Les nombres fractionnaires sont arrondis à des valeurs entières

Table 14-4 Paramètres du bloc RECORD

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Caption (Légende). Légende de huit caractères maximum pour identifier un enregistrement de données sauvegardé dans chaque enregistrement.

A0 à A7. Ensemble de 8 valeurs analogiques contenues dans un bloc RECORD à utiliser dans un schéma de boucles et/ou des séquences comme valeurs de recettes, etc. Ces valeurs peuvent être entrées depuis un schéma de boucles ou sorties dans un schéma de boucles et peuvent être chargées ou sauvegardées dans des enregistrements de données spécifiés au cours de la conduite ou de la configuration, déclenchées par les entrées de bloc *Get* et *Put*.

D. Champ binaire de 16 valeurs logiques - *Bit0* à *Bit15* - contenues dans le bloc RECORD, utilisées comme les paramètres *A0* à *A7*.

Alarms (Alarmes). Voir la description générale du champ Alarmes à la page 2-5.

- **Software (Logiciel).** VRAI si la mémoire est corrompue ou si un bloc image se met hors ligne.
- **File (Fichier).** VRAI si *Put* ou *Get* tentent d'utiliser un nom de fichier erroné (par ex. inexistant) ou un numéro d'enregistrement erroné (par ex. vierge ou hors échelle).
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

Filename (Nom de fichier). Un nom de fichier de 8 caractères (maxi) pour la sauvegarde de 50 enregistrements maximum. Le nom de fichier spécifié devient le nom de fichier racine d'un fichier d'enregistrement, l'extension .RCD étant ajoutée automatiquement. Les noms de fichiers peuvent être modifiés au cours de la conduite par l'intermédiaire du menu de spécifications du bloc pour accéder à d'autres enregistrements.

FilePath (Chemin du fichier). Spécifie l'emplacement du fichier d'enregistrement dans le paramètre *FileName*. *FilePath* prend par défaut le "lecteur par défaut local" (champ vierge) qui est l'unité M: pour un T1000 et E: pour un T100.

FilePath peut comprendre la lettre d'une unité, le numéro de noeud ou les deux, par ex.:

M: Unité de disque RAM sur l'instrument local
 ED: : Noeud ED LIN, unité par défaut
 ED: : B: Noeud ED LIN, lecteur DTU inférieur.

NOTA. Avec la version 3/1 du logiciel, il ne faut pas utiliser *Get* ou *Put* sur le réseau dans un instrument qui fonctionne comme un serveur de fichiers, c'est à dire qu'il pourrait répondre à la demande d'opérations de transfert de fichiers d'un instrument déporté. Dans ce cas, l'instrument serait un client, et la version 3/1 du logiciel ne permet pas d'avoir simultanément un statut serveur/client.

Le transfert de fichier sur le réseau est plus lent que l'accès aux fichiers locaux et est soumis à un maximum de huit fichiers ouverts simultanément au niveau du serveur, et (Version 3/1) à des risques de conflits serveur/client. C'est pour ces raisons que vous devriez lorsque c'est possible maintenir les fichiers localement.

RecordNo (N° d'enregistrement). Le numéro d'enregistrement individuel (0 à 49) auquel il faut accéder dans un fichier d'enregistrement (spécifié par le paramètre

FileName), lorsque les opérations *Get*, *GetNext* ou *Put* sont déclenchées. Un fichier d'enregistrement peut contenir un maximum de 50 enregistrements, chaque enregistrement contenant un ensemble complet de 8 valeurs analogiques et 16 valeurs logiques, ainsi qu'une légende.

Get (Lire). L'entrée d'un front montant (transition FAUX à VRAI) dans *Get* permet de copier les valeurs analogiques et logiques, ainsi que la légende sauvegardées dans l'enregistrement spécifié par *FileName*, *FilePath* et *RecordNo* dans les paramètres correspondants du bloc RECORD (*A0* à *A7*, *D* et *Caption*). Les valeurs existantes sont écrasées. *Get* se remet automatiquement à zéro au cours de la conduite s'il n'est pas relié.

NOTA 1. Pour lire (*Get*) des données (ou écrire (*Put*)) dans un fichier d'enregistrement sauvegardées dans un instrument déporté, un bloc RECORD devrait être local dans l'instrument où le fichier d'enregistrement réside, et mis en image dans l'instrument qui doit lire ou écrire (*Get* ou *Put*) les données (Voir figure 14-1). De cette manière, le transfert de données se fait par mise à jour du bloc, ce qui est plus rapide et moins restrictif que le transfert de fichiers serveur/client.

NOTA 2. Ne jamais lier les entrées *Get* ou *Put* d'un bloc RECORD image, dans la mesure où le bloc réel correspondant ne peut détecter les liaisons et donc remet automatiquement à zéro ces entrées, en conflit avec les entrées qui ne se remettent pas à zéro (liées) dans le bloc image. Il vaut mieux utiliser *Get* ou *Put* manuellement ou par l'intermédiaire d'une instruction SFC ou en transmettant des valeurs logiques en utilisant un bloc DG_CONN image dans l'instrument déporté et lié à ce niveau au bloc RECORD local - voir l'exemple de la figure 14-1.

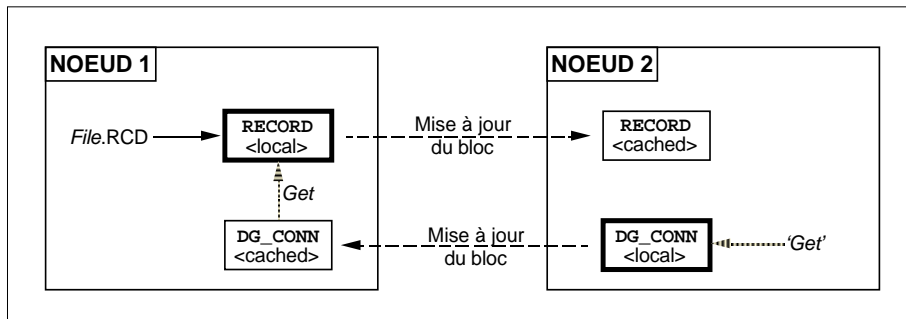


Figure 14-1 "Get" utilisé sur le réseau avec des blocs image

GetNext (Lire suivant). Si *GetNext* passe à VRAI, l'enregistrement suivant du fichier est chargé, c'est à dire incrémenter *RecordNo* et ensuite fait un "Get". Après le numéro d'enregistrement 49, on repasse à 0. *RecordNo* peut être présélectionné à 49 (ou - 1) pour permettre à *GetNext* de charger le numéro d'enregistrement 0. *GetNext* se remet automatiquement à zéro après utilisation.

Put (Ecrire). L'entrée d'un front montant (transition FAUX à VRAI) dans *Put* permet de copier les valeurs analogiques et logiques, ainsi que la légende sauvegardées dans le bloc RECORD (en *A0* à *A7*, *D* et *Caption*) dans l'enregistrement spécifié par *FileName*, *FilePath* et *RecordNo*. Les valeurs existantes sont écrasées. *Put* repasse automatiquement à FAUX.

NOTA 1. Voir les Notas 1 et 2 de la section *Get* ci-dessus.

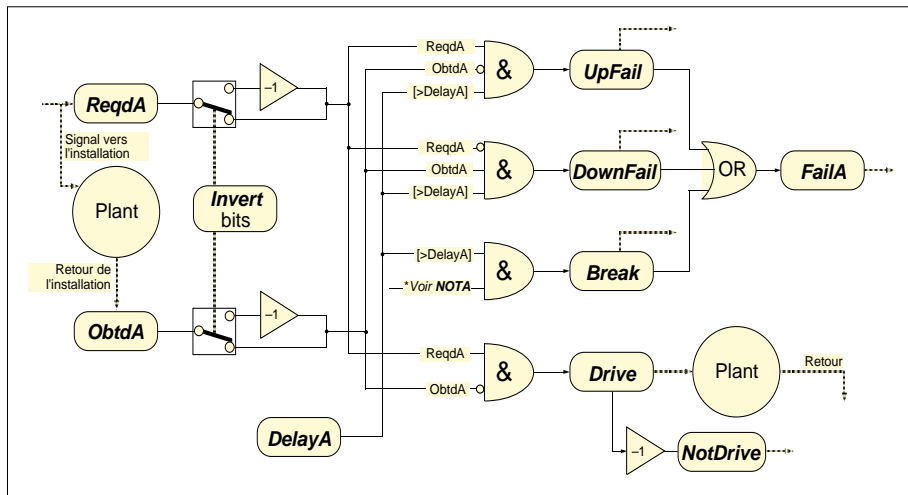
NOTA 2. L'utilisation de *Put* appliqué à l'unité E: d'un T100 est restreinte au cours de la conduite et devrait être évitée.

GetOK (Lire OK). Passe à VRAI après la réussite d'une opération *Get* ou *GetNext*. Sinon *GetOK* passe à FAUX et l'alarme *fichier* se déclenche.

PutOK (Ecrire OK). Passe à VRAI après la réussite d'une opération *Put*. Sinon *PutOK* passe à FAUX et l'alarme *fichier* se déclenche.

DISCREP : BLOC DISCORDANCE

Fonction du bloc



*NOTA. Voir les détails sur le paramètre *Break* dans la section *Menu de spécifications du bloc*

Figure 14-2 Schéma du bloc DISCREP - Voie A

Voir le schéma de la figure 14-2. Le bloc DISCREP permet de vérifier que les instruments à commande tout ou rien d'une installation répondent correctement. La vérification se fait principalement par comparaison des signaux transmis à l'installation - les états "requis" - avec les signaux correspondants renvoyés par l'installation - les états "obtenus". Si ces signaux ne correspondent pas aux critères spécifiés (en tenant compte des temporisations définies), les bits de sortie appropriés du bloc et des alarmes sont définis, et peuvent être utilisés comme avertissements, diagnostics, et comme liaisons à un schéma de boucles.

Entrées

Le bloc comprend quatre voies identiques et indépendantes, A à D, qui peuvent être validées/invalidées individuellement si nécessaire (par les entrées *DisableA* à *DisableD*). Pour des raisons de clarté - seule l'une d'elles - la voie A - est représentée sur la figure 14-2. Chaque voie dispose également d'une entrée qui contrôle l'état logique requis (*ReqdA* à *ReqdD*), et une entrée pour l'état de l'instrument renvoyé et effectivement obtenu (*ObtdA* à *ObtdD*). Toutes les entrées 'Reqd' et 'Obtd' peuvent être inversées individuellement par l'intermédiaire du champ binaire *Invert* du bloc.

Sorties

Chaque voie comprend cinq sorties logiques (groupées dans les champs binaires *OutA* à *OutD*) contenant les sorties *Drive* et *NotDrive*, *UpFail*, *DownFail* et *Break*.

Ces sorties définies lorsque certaines discordances se produisent entre les entrées “requis” et les entrées “obtenues” sont schématisées dans la figure 14-2 et détaillées dans la section *Menu de spécifications du bloc* ci-après. La sortie *Drive* peut être utilisée pour le contrôle ‘On/Off’ d’un actionneur. Un paramètre de temporisation pour chaque voie (*DelayA* à *DelayD*) permet de définir une temporisation convenable - par ex. “temps d’excursion” - entre le changement d’état d’un signal “Reqd” et la commutation en conséquence du signal “Obtd”, avant qu’une erreur ne soit indiquée. *UpFail* est défini si l’instrument ne se met pas en route, et *DownFail* est défini si l’instrument ne se met pas hors tension après la temporisation. *Break* est déclenché lorsque “Obtd” ayant égalé “Reqd” change en conséquence d’état (après la temporisation pour tenir compte du ‘rebond de contact’).

Les bits de sortie alarme (*FailA* à *FailD*) indiquent pour chaque voie si ses paramètres *UpFail*, *DownFail* ou *Break* ont été définis.

Paramètres du bloc

Les symboles utilisés dans la table 14-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Invert	Bits d’inversion entrée	Champ binaire	
ReqdA	Inversion entrée ReqdA	V/F	
ObtdA	Inversion entrée ObtdA	V/F	
ReqdB	Inversion entrée ReqdB	V/F	
ObtdB	Inversion entrée ObtdB	V/F	
ReqdC	Inversion entrée ReqdC	V/F	
ObtdC	Inversion entrée ObtdC	V/F	
ReqdD	Inversion entrée ReqdD	V/F	
ObtdD	Inversion entrée ObtdD	V/F	
ReqdA à ReqdD	Etat éléments requis, voies A à D	V/F	
ObtdA à ObtdD	Etat éléments obtenu, voies A à D	V/F	
DelayA à DelayD	Temporisation, voies A à D	Eng	
DisableA à DisableD	Invalidation voie, voies A à D	V/F	
OutA à OutD	Sorties, voies A à D	Champ binaire	
Drive	Sortie contrôle “ON/OFF”	V/F	
NotDrive	Inverse de Drive	V/F	
UpFail	Indique défaut de Obtd à la mise à 1 (in < delay)	V/F	
DownFail	Indique défaut de Obtd à RAZ (in < Delay)	V/F	
Break	Indique interruption dans état Obtd (après Delay)	V/F	
Alarms			
Software	Données corrompues/Défaut de communication	V/F	
FailA à FailD	Défaut sur voie A à D (UpFail, DownFail ou Break)	V/F	
Combined	Alarme combinée du bloc	V/F	

Table 14-5 Paramètres du bloc DISCREP

Menu de spécifications du bloc

Dbase, Block, Type. Voir les détails sur ces champs “en-tête” à la page 2-3.

Invert (Inversion). Un champ binaire de 8 bits qui permet d’inverser l’entrée correspondante du bloc. Les opérations de vérification de discordance sont effectuées sur les entrées *après* toute inversion, comme le montre la figure 14-1.

ReqdA à ReqdD. Chacune de ces quatre entrées logiques (avec ses bits *Invert*) indique l’état requis d’un élément d’installation pour le comparer à l’état réel obtenu de son signal de retour (*ObtdA* à *ObtdD*). Les états des paramètres “Reqd” et “Obtd” devraient s’égaliser lorsque l’élément de l’installation a atteint l’état requis.

ObtdA à ObtdD. Chacune de ces quatre entrées logiques (y compris les bits *Invert*) permettent de renvoyer un signal indiquant l’état réel obtenu d’un élément d’une installation.

DelayA à DelayD (TemporisationA à TemporisationD). Les paramètres analogiques définissant pour chaque voie la temporisation autorisée entre le changement d’état de “Reqd” et la commutation “Obtd” en conséquence, avant qu’une erreur (*UpFail* ou *DownFail*) ne soit indiquée. La sortie *Break* (Interruption) utilise également le paramètre *Delay*. Les horloges de temporisation se remettent à zéro et recommencent chaque fois que le paramètre “Reqd” correspondant change d’état.

NOTA. Dans la figure 14-1, [>DelayA] signifie “temps défini par *DelayA* dépassé”, qui peut être VRAI ou FAUX.

DisableA à DisableD (InvalidationA à InvalidationD). Lorsque ces entrées passent à VRAI, les voies correspondantes du bloc sont invalidées, c’est à dire qu’aucune sortie n’est générée quel que soit l’état des entrées. Ces paramètres passent par défaut à BAS (c’est à dire validés) lorsqu’ils ne sont pas reliés.

OutA à OutD (SortieA à SortieD). Les champs binaires de cinq sorties indiquant pour chaque voie le résultat de la vérification de discordance en associant ses paramètres ‘Reqd’, ‘Obtd’ et ‘Delay’. (Voir le schéma du bloc à la figure 14-1 ci-dessus).

■ **Drive (Entraînement).** VRAI lorsque “Reqd” est VRAI et “Obtd” est FAUX. *Drive* peut être utilisé en mode de commande ‘ON/OFF’ pour piloter un actionneur jusqu’à ce que “Obtd” renvoyé indique que la position requise “Reqd” est atteinte.

■ **NotDrive (Non-entraînement).** L’inverse du paramètre *Drive* à utiliser avec une logique négative.

■ **UpFail (Défaut marche).** VRAI lorsque “Reqd” est VRAI, mais “Obtd” reste FAUX plus longtemps que le temps autorisé par “Delay”. Voir figure 14-3.

■ **DownFail (Défaut arrêt).** VRAI lorsque “Reqd” est FAUX, mais “Obtd” reste VRAI plus longtemps que le temps autorisé par “Delay”.

La figure 14-3 montre le fonctionnement de *UpFail* et *Downfail*. Dans la figure, *ReqdA* passe à l’état haut au temps A, ce qui déclenche l’horloge *DelayA*. *ObtdA* tente de suivre, mais ne réussit pas avant le temps C - après un certain rebond et au-delà de

la temporisation autorisée. *UpFail* est mis à 1 en B, à la fin de la temporisation, mais est remis à zéro au temps C - dans la mesure où *ObtdA* est maintenant mis en route. En D, *ReqdA* est remis à zéro, mais *ObtdA* est toujours à l'état haut à la fin de la nouvelle temporisation, donc *DownFail* est mis à 1 en E. *ObtdA* repasse à l'état souhaité en F et rebondit à ce moment-là, ce qui fait que *DownFail* copie le schéma de défaut.

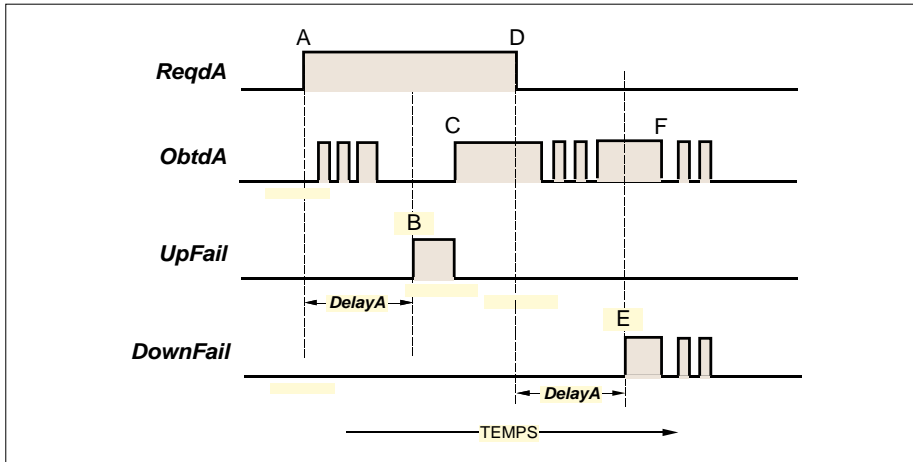


Figure 14-3 Action des sorties UpFail & DownFail - Voie A

- **Break (Interruption).** Break passe à VRAI, “Obtd” ayant égalé “Reqd” (depuis la dernière commutation de “Reqd”), et change d’état en conséquence. Pour empêcher les contacts à rebond de déclencher un *Break* intempestif, cette sortie n’est pas activée tant que le temps spécifié par “Delay” ne s’est pas écoulé après le dernier changement d’état de “Reqd”. La figure 14-4 montre le fonctionnement de *Break*.

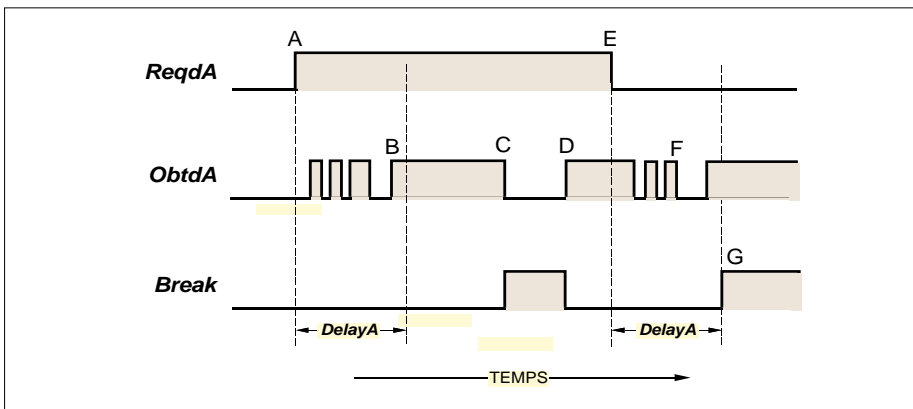


Figure 14-4 Action de la sortie Break - Voie A

Dans la figure, *ReqdA* passe à l'état haut au temps A, en déclenchant l'horloge *DelayA*. *ObtdA* tente de suivre *ReqdA* et y réussit au temps B - après un certain rebond. *Break* n'est pas défini, dans la mesure où B se trouve dans l'intervalle de temporisation spécifié. Toutefois, *ObtdA* est ultérieurement en défaut entre les temps C et D, en définissant *Break* pour cette période. Il est à noter que *Break* ne se verrouille pas. En E, *ReqdA* est remis à zéro, mais *ObtdA* - en dépit d'une remise à zéro temporaire au temps F - est à nouveau en défaut et reste à l'état haut. Donc, *Break* est défini au temps G, à la fin de la nouvelle période de temporisation.

Alarms (Alarmes). Voir la description générale du champ alarmes à la page 2-5.

- **Software (Logiciel).** VRAI si la mémoire est corrompue ou si un bloc image se met hors ligne.
- **FailA à FailD (DéfautA à D).** Défaut sur les voies A à D - c'est à dire *UpFail*, *DownFail* ou *Break* VRAIS.
- **Combined (Combinée).** VRAI si une alarme est active dans le bloc. Adopte le même message d'état et numéro de priorité que l'alarme active du bloc qui a la priorité la plus haute.

RCP_SET: BLOC ENSEMBLE DE RECETTES

Fonction du bloc

Le bloc de fonction RCP_SET contrôle l'ensemble de recettes — c'est à dire le fichier à utiliser — pour un certain nombre de lignes de recettes.

Paramètres du bloc

Les symboles utilisés dans la table 14-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Load	Front montant charge fichier .UYR	V/F	
Filepath	Chemin du nom de fichier	Alphanumérique	
Filename	Nom de base du fichier .UYR	Alphanumérique	
InUse	TRUE lorsque la ligne est téléchargée, càd fichier utilisé	V/F	
Options	(Options réservées à un usage ultérieur)	(ABCD)	
Active	TRUE lorsqu'un rapport est en cours de génération	V/F	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Config	Trop de blocs RCP_SET dans la BdD ou aucune ligne attachée	V/F	
File	Erreur dans le fichier <i>FormFile.UYF</i> au chargement	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
FileErr	Code d'erreur fichier	Menu	
FileErLn	N° de ligne où <i>FileErr</i> s'est produit	Entier	
Line1 - Line16	Noms des blocs RCP_LINE associés	Alphanumérique	

Table 14-6 Paramètres du bloc RCP_SET

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 14-6.

InUse. TRUE lorsqu'une ligne est téléchargée, pour indiquer que le fichier est utilisé. Lorsque *InUse* est TRUE, un nouveau fichier ne peut être chargé. FALSE indique qu'aucune ligne n'est en cours de téléchargement.

FileErr. (OK /FORMAT /TOO_BIG /BAD_REF /ACTIVE /NO_FILE /OTHER)
Code d'erreur fichier, c'est à dire la raison de l'erreur fichier. Les options signifient:

- **OK.** Aucune erreur.
- **FORMAT.** Format de fichier erroné.
- **TOO_BIG.** Recette trop grande, c'est à dire trop de lignes, trop de recettes ou trop de variables.
- **BAD_REF.** Référence de base de données LIN incorrecte ou inexistante.

■ **ACTIVE.** Fichier recette en cours d'utilisation.

■ **NO_FILE.** Fichier introuvable.

■ **OTHER.** Erreur indéfinie.

Line1 - Line16. Nom des blocs RCP_LINE associés. (*Seuls Line1 à Line4 sont disponibles à l'heure actuelle.*)

RCP_LINE: BLOC LIGNE DE RECETTE

Fonction du bloc

Le bloc de fonction RCP_LINE contrôle le téléchargement de recettes d'un fichier .UYR dans le bloc RCP_SET associé. Un maximum de seize blocs RCP_LINE peuvent être rattachés à un seul bloc RCP_SET, par l'intermédiaire des champs *Line1* à *Line16* du bloc RCP_SET.

Paramètres du bloc

Les symboles utilisés dans la table 14-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
RecipeNo	Numéro de la recette	Entier	
Download	Front montant télécharge recette spécifiée par <i>RecipeNo</i>	V/F	
Abort	Front montant suspend recette spécifiée par <i>RecipeNo</i>	V/F	
Capture	Front montant acquière valeurs act. & enregistre fichier	V/F	
Options	(Options réservées à un usage ultérieur)	(ABCD)	
RprtName	Nom du fichier pour le rapport de recette	Alphanumérique	
TrigRprt	Front montant génère rapport texte de la recette	V/F	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Config	Bloc non rattaché à un bloc RCP_SET block	V/F	
Failed	Certains réf. BdD LIN non vérifiées dans recette	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Status	Etat actif de la recette	Menu	
Active	TRUE lorsque téléchargement en cours	V/F	
Failed	TRUE si échec dernier téléchargement	V/F	

Table 14-7 Paramètre du bloc RCP_LINE

Menu de spécifications du bloc

Les informations suivantes complètes celles de la table 14-7.

TrigRprt. Une entrée de front montant dans *TrigRprt* génère un rapport texte de la recette. Le rapport généré par le bloc RCP_LINE est un fichier de variables séparées par des virgules, dont l'extension est .TXT. Le format est le suivant:

- **Line 1.** Détails d'analyse du fichier .UYR
- **Line 2.** Date et heure du rapport
- **Line 3.** Nom du bloc RCP_SET, bloc RCP_LINE, nom du fichier .UYR, nom de la ligne, nom de la recette
- **Line 4+.** Nom de la variable, valeur de la recette, valeur active (valeur de contrôle si spécifiée).

Exemple. Le fichier TREND.UYR suivant pour le bloc RCP_SET '**SET1**' exécutant la recette '**Run Prog**' sur le bloc RCP_LINE '**LINE1**' pourrait générer le rapport texte suivant:

```
UYR,1
2,24/08/00,10:55:34,Mike Fox
,30
,Setpoint:1,Monitor,Run Prog
File Name,SppCtrl.RqNxtPrg,SppCtrl.CurrProg,TREND
Ready,SppCtrl.NxtRdy,SppCtrl.State,TRUE
Duration,,SppCtrl.ProgDur,
```

Le rapport généré serait le suivant:

```
2,24/08/00,10:55:34,A N Other
,20/09/00,16:34:25,
SET1      ,LINE1      ,TREND.UYR,1,Run Prog
File Name,TREND,TREND,TREND
Ready,TRUE,TRUE,RUN
Duration,,00:37:10
```

Alarms.

■ **Failed.** Certaines des références de la base de données LIN dans la recette n'ont pas été reconnues.

Status. (RESET /DOWNLOAD /COMPLETE /FAILED) Etat actif de la recette. Les options signifient:

- **RESET.** Aucune recette chargée.
- **DOWNLOAD.** Téléchargement en cours.
- **COMPLETE.** Téléchargement s'est déroulé normalement.
- **FAILED.** Echec du dernier téléchargement.

BAT_CTRL: BLOC DE CONTRÔLE DES LOTS

Fonction du bloc

Le bloc de fonction BAT_CTRL régit le chargement (d'un fichier .UYB) et le contrôle d'un lot. Il fournit un environnement de base de données LIN avec une interface de commande dans le moteur de lots correspondant, ainsi que les informations d'état et d'assistance correspondantes.

Paramètres du bloc

Les symboles utilisés dans la table 14-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Mode	Mode de fonctionnement par lots (SeuSlave mis en oeuvre)	Menu	
Filepath	Chemin du fichier pour Filename	Alphanumérique	
Filename	Nom de base du fichier .UYB	Alphanumérique (ABCD)	
Options	(Options réservées à un usage ultérieur)	Menu	
Command	Commandes pour le contrôleur de lots	Menu	
CmndFlg	Version booléenne de Command	(AB)CD hex	
* [6] LOAD	Chargement lot & transition vers état repos	V/F — 1	D
[7] RESET	Transition vers état Réinit (ou repos en option)	V/F — 2	
[5] START	Initialisation lot & transition vers état Démar.	V/F — 4	
[2] HOLD	Transition vers état [VRAI] Maintien, [FAUX] Redémar.	V/F — 8	
[4] ABORT	Transition vers état Abandon	V/F — 1	C
†[3] STOP	Transition vers état Arrêt	V/F — 2	
†[1] PAUSE	Transition vers état [VRAI] Pause, [FAUX] Reprise	V/F — 4	
		V/F — 8	
CmndHshk	Drapeaux protocole transfert Command	(AB)CD hex	
STARTED	Transition Starting → Started	V/F — 1	D
HELD	Transition Holding → Held	V/F — 2	
RESTART	Transition Restarting → Restarted	V/F — 4	
ABORTED	Transition Aborting → Aborted	V/F — 8	
†STOPPED	Transition Stopping → Stopped	V/F — 1	C
†PAUSED	Transition Pausing → Paused	V/F — 2	
†RESUMED	Transition Resuming → Resumed	V/F — 4	
		V/F — 8	
Alarms			
Software	Anomalie mémoire données RAM bloc / défaut réseau	V/F	
Config	Erreur de configuration	V/F	
File	Erreur de traitement fichier	V/F	
Failed	Défaut conduite lot (Voir CondFlg)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
FileErr	Raison erreur fichier	Menu	
FileErLn	0 si pas d'erreur, sinon n° ligne 1ère erreur analyse trouvée	Entier	

* *Priorité de commande (voir ci-dessous)* † *Non mis en oeuvre*

Table 14-8 suite...

...Table 14-8 suite

Paramètre	Fonction	Unités	Etat
StrtDate	Date démarrage lot	Date	
StrtTime	Heure démarrage lot	Heure	
EndDate	Date fin lot	Date	
EndTime	Heure fin lot	Heure	
BatchNo	Numéro unique du lot	Entier	
BatchId	Chaîne ID du lot	Alphanumérique	
State	Etat actif état machine moteur des lots	Menu	
StateFlg	Version booléenne de <i>State</i>	ABCD hex	
RESET	Aucun lot chargé	V/F — 1	D
IDLE	Lot chargé & prêt à démarrer	V/F — 2	
STARTING	Transition de <i>Idle</i> à <i>Running</i>	V/F — 4	
RUNNING	Exécution normale du lot	V/F — 8	
COMPLETE	Fin automatique normale du lot	V/F — 1	C
HOLDING	Transition de <i>Running</i> à <i>Held</i>	V/F — 2	
HELD	Lot suspendu manuellement et indéfiniment	V/F — 4	
RESTART	Transition de <i>Held</i> à <i>Running</i>	V/F — 8	
†PAUSING	Transition de <i>Running</i> à <i>Paused</i>	V/F — 1	B
†PAUSED	Lot suspendu manuellement, brièvement seulement	V/F — 2	
†RESUMING	Transition de <i>Paused</i> à <i>Running</i>	V/F — 4	
†STOPPING	Transition de <i>Running</i> à <i>Stopped</i>	V/F — 8	
†STOPPED	Fin manuelle normale du lot	V/F — 1	A
ABORTING	Transition à <i>Aborted</i>	V/F — 2	
ABORTED	Fin manuelle normale du lot, avec une tempo mini.	V/F — 4	
FAILED	Exécution du lot terminée suite à défaut irrécupérable	V/F — 8	
CondFlg	Drapeaux condition du lot	(AB)CD hex	
LOADED	Lot chargé	V/F — 1	D
ACTIVE	Commande Start acceptée, lot pas encore au stade final	V/F — 2	
	(<i>Inutilisé</i>)	V/F — 4	
	(<i>Inutilisé</i>)	V/F — 8	
TIMEOUT	Logique de phase n'a pas répondu avant fin tempo.	V/F — 1	C
RCP_FAIL	Défaut causé par le système de recettes	V/F — 2	
LOG_FAIL	Défaut causé par le système de consignation	V/F — 4	
	(<i>Inutilisé</i>)	V/F — 8	
CurPhase	Phase de lot en cours d'exécution	Entier	

†Etats non mis en oeuvre à l'heure actuelle

Table 14-8 Paramètres du bloc BAT_CTRL

Menu de spécifications du bloc

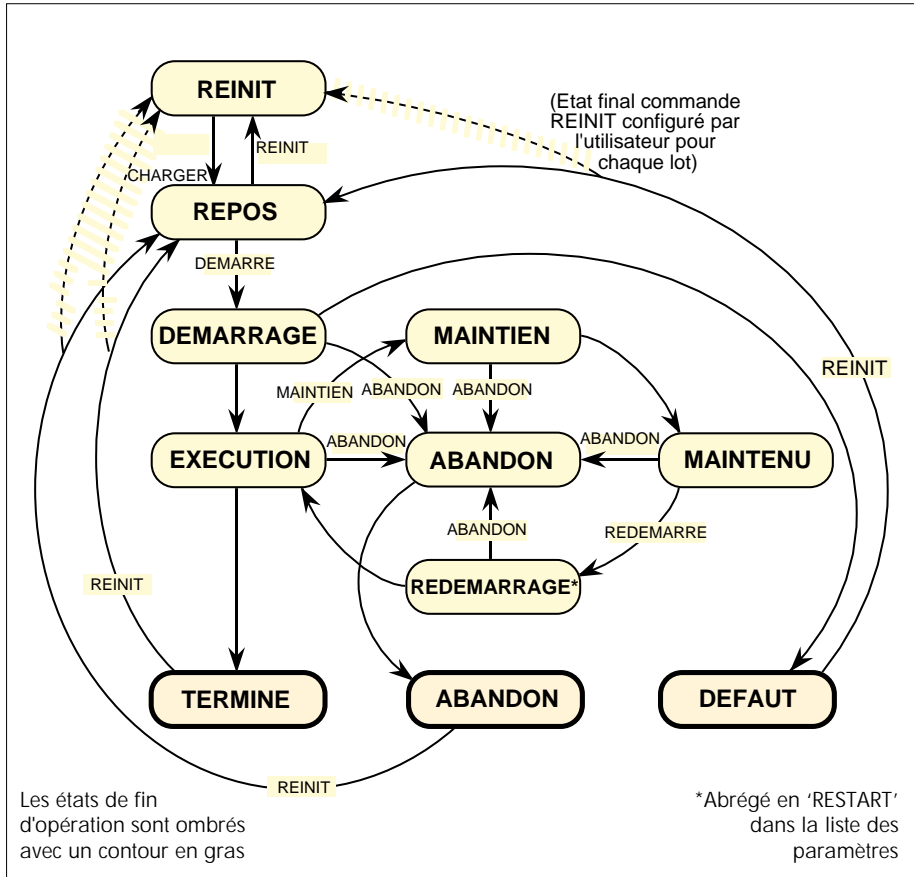
Les informations suivantes complètent celles de la table 14-8.

Command. (LOAD/RESET/START/HOLD/RESTART/PAUSE/RESUME/STOP/ABORT/UNLOAD) Commandes dans le contrôleur de lots. Voir le paramètre *CmndFlg* de la table 14-8 pour de plus amples informations sur ces commandes. La figure 14-5 montre comment certaines des commandes fonctionnent dans le schéma d'état du moteur

des lots.

CmndFlg. Version booléenne du paramètre *Command*. Chaque champ *CmndFlg* est sensible au niveau, mais est ignoré, sauf si l'état du moteur des lots est applicable à la commande (voir figure 14-5).

L'acceptation d'une commande évite une acceptation répétée de la même commande.



Lorsque plusieurs drapeaux de commande sont actifs, ils sont traités en ordre de priorité décroissant. (Les priorités sont données dans la table 14-8.)

FileErr. (OK /FORMAT /TOO_BIG /BAD_REF /BAD_ID /BAD_PHAS /BAD_STAT /NO_FILE /FILE_WRI /SHAR_BLK /BAD_DICT /OTHER) Code d'erreur fichier, c'est à dire la raison de l'erreur dans le fichier. Les options signifient:

- **OK.** Aucune erreur.
- **FORMAT.** Format de fichier erroné.

- **TOO_BIG.** Définition du lot trop importante, ex. trop de phases.
- **BAD_REF.** Référence de base de données LIN non valable ou inexistante.
- **BAD_ID.** Problème au niveau de l'ID du contrôleur de lots.
- **BAD_PHAS.** (*Non mis en oeuvre*).
- **BAD_STAT.** Erreur dans l'état du lot
- **NO_FILE.** Fichier introuvable.
- **FILE_WRI.** (*Non mis en oeuvre*).
- **SHAR_BLK.** Référence de base de données LIN illégalement partagée par plusieurs contrôleurs de lots.
- **BAD_DICT.** Référence au dictionnaire incorrecte ou inexistante.
- **OTHER.** Erreur indéfinie.

State. (RESET/IDLE/STARTING/RUNNING/COMPLETE/HOLDING/HELD/RE-START/PAUSING/PAUSED/RESUMING/STOPPING/STOPPED/ABORTING/ABORTED/FAILED) Etat actif dans la machine d'état du moteur de lots. Voir le paramètre *StateFlg* dans la table 14-8 pour de plus amples informations sur ces états, et également la figure 14-5.

[Page laissée intentionnellement blanche]

Chapitre 15 BLOCS DE FONCTION DCM

Les blocs de fonction du module de contrôle dévolu (DCM) qui tournent dans les T2900/T800/T940 permettent de communiquer sur le réseau avec des instruments cibles 2500 déportés (système d'entrée/sortie) — et d'autres de la série 2000.

Chaque bloc DCM présente une vue sur des valeurs de données particulières dans l'instrument cible, et permet de configurer les paramètres de communication. Les données des 2500 déportés/des instruments de la série 2000 sont affichées dans le bloc DCM local sous la forme de champs d'entrée et/ou de sortie qui peuvent être reliés au schéma de boucles qui tournent dans les T2900/T800/T940, et ainsi établir un dialogue.

SOUS-CATÉGORIES DES BLOCS DE FONCTION DCM

Les blocs de fonction dans la catégorie DCM peuvent être divisés en huit catégories, voir la liste de la table 15-1. La table récapitule la fonction de chaque bloc et vous renvoie à la page où commence la description du bloc dans le présent chapitre.

Sous-catégorie DCM	Bloc DCM	Fonction	Voir page...
Blocs boucle	D2X_LOOP	Accès à la boucle de régulation PID dans les 2200/2400/2500	15-8
	D2X_TUNE	Mise au point de la boucle PID dans les 2200/2400/2500	15-10
	D25_LOOP	Accès à la boucle de régulations PID dans le 2500C	15-12
	D25eLOOP	Accès à la boucle de régulations PID dans le 2500E	15-12
	D25_TUNE	Mise au point de la boucle PID dans le 2500C	15-15
	D25eTUNE	Mise au point de la boucle PID dans le 2500E	15-15
Blocs rampe	D25_RAMP	Point de consigne déporté rampe (2500C)	15-17
	D25eRAMP	Point de consigne déporté rampe (2500E)	15-17
Blocs modules E/S	D25_MOD	Accès à un seul module E/S physique dans le 2500	15-19
	D25_AI2	Accès module d'entrée analogique 2 voies dans le 2500	15-21
	D25_AI3	Accès module d'entrée analogique 3 voies dans le 2500	15-21
	D25_AI4	Accès module d'entrée analogique 4 voies dans le 2500	15-21
	D25_AO2	Accès module de sortie analogique 2 voies dans le 2500	15-23
	D25_AO4	Accès module de sortie analogique 4 voies dans le 2500	15-23
	D25_DI4	Accès module d'entrée logique 4 voies dans le 2500	15-25
	D25_DI6	Accès module d'entrée logique 6 voies dans le 2500	15-25
	D25_DI8	Accès module d'entrée logique 8 voies dans le 2500	15-25
	D25_DO4	Accès module de sortie logique 4 voies dans le 2500	15-27
Blocs voies E/S	D25_AICH	Accès voie d'entrée analogique unique dans le 2500	15-29
	D25_AOCH	Accès voie de sortie analogique unique dans le 2500	15-32
	D25_DICH	Accès voie d'entrée logique unique dans le 2500	15-34
	D25_DOCH	Accès voie de sortie logique unique dans le 2500	15-36
	D25_AI	Accès voie d'entrée analogique unique dans le 2500	15-48
	D25_AO	Accès voie de sortie analogique unique dans le 2500	15-52
	D25_DI	Accès voie d'entrée logique unique dans le 2500	15-55
	D25_DO	Accès voie de sortie logique unique dans le 2500	15-58

Table 15-1 suite...

...Table 15-1 suite

Sous-catégorie DCM	Bloc DCM	Fonction	Voir page...
Blocs filaires	D25_R_CV	Accès à 8 valeurs réelles calculées liaisons filaires dans le 2500	15-38
	D25_B_CV	Accès à 8 valeurs booléen. calculées liaisons filaires dans 2500	15-39
	D25_R_UV	Accès à 8 valeurs réelles utilisateur dans le 2500	15-40
Blocs alarmes	D25_UALM	Accès alarmes analogiques ou logiques utilisateur dans 2500	15-41
Bloc paramètres	DCM_I8	Accès à 8 paramètres entiers avec signe (16 bits) dans le 2500	15-42
	DCM_S8	Accès à 8 paramètres entiers avec signe (8 bits) dans le 2500	15-42
	DCM_UI8	Accès à 8 paramètres entiers sans signe (16 bits) dans le 2500	15-42
	DCM_US8	Accès à 8 paramètres entiers sans signe (8 bits) dans le 2500	15-42
	DCM_R8	Accès à 8 paramètres nombres réels (32 bits) dans le 2500	15-42
	DCM_T8	Accès à 8 paramètres de durée (32 bits) dans le 2500	15-42
	DCM_W8	Accès à 8 paramètres mots hex 'ABCD' (16 bits) dans le 2500	15-42
	DCM_Y8	Accès à 8 paramètres octets hex 'AB' (8 bits) dans le 2500	15-42
	DCM_B8	Accès à 8 paramètres booléens dans le 2500	15-42
Blocs instruments	DCM_D8	Accès 8 paramètres entiers double précision (32 bits) dans 2500	15-42
	D2500	Fournit une vue globale d'un instrument 2500 déporté	15-44
	D2000	Fournit une vue globale d'un instrument 2200 ou 2400 déporté	15-46

Table 15-1 Sous-catégories des blocs DCM

UTILISATION DES BLOCS DE FONCTION E/S DCM

La table 15-1A compare les propriétés de la gamme de modules E/S et les blocs DCM à voies E/S, et vous guide sur leur utilisation.

Type DCM	Bloc DCM	Accède à un 2500 ...	Notes d'utilisation
Module E/S	D25_MOD	Module physique à E/S unique	Permet surtout d'identifier un module
	D25_AI2	Module entrée analogique à 2 voies	Les blocs de modules E/S ont des temps d'exécution de la base de données et des communications légèrement plus rapides que les blocs à voies E/S, et utilisent moins de mémoire de base de données. Mais, ils ne fournissent qu'un repère pour toutes les voies du module, alors que les blocs à voies E/S disposent d'un repère pour chaque voie.
	D25_AI3	Module entrée analogique à 3 voies	
	D25_AI4	Module entrée analogique à 4 voies	
	D25_AO2	Module sortie analogique à 2 voies	
	D25_AO4	Module sortie analogique à 4 voies	
	D25_DI4	Module entrée logique à 4 voies	
	D25_DI6	Module entrée logique à 6 voies	
	D25_DI8	Module entrée logique à 8 voies	
	D25_DO4	Module sortie logique à 4 voies	
			Voir nota [1]

Tableau 15-1A suite...

...Tableau 15-1A suite

Type DCM	Bloc DCM	Accède à un 2500 ...	Notes d'utilisation
Voie E/S	D25_AICH	Voie entrée analogique unique	Gère les fonctionnalités 2500 sans "extra". Exécution & communication légèrement plus rapides que le D25_AI.
	D25_AI	Voie entrée analogique unique	Sur-ensemble du bloc D25_AICH avec des fonctionnalités supplémentaires — par ex.: champs alarmes & modes — mais, exécution & communications légèrement plus lentes.
	D25_AI_T	Voie entrée analogique unique	Comme le bloc D25_AI, mais avec des alarmes temporisées.
	D25_AOCH	Voie sortie analogique unique	Gère les fonctionnalités 2500 sans "extra". Exécution & communication légèrement plus rapides que le D25_AO. Voir nota [1].
	D25_AO	Voie sortie analogique unique	Sur-ensemble du bloc D25_AOCH avec des fonctionnalités supplémentaires — par ex.: champs alarmes & modes — mais, exécution & communications légèrement plus lentes. Voir nota [2].
	D25_DICH	Voie entrée logique unique	Gère les fonctionnalités 2500 sans "extra". Exécution & communication légèrement plus rapides que le D25_DI.
	D25_DI	Voie entrée logique unique	Sur-ensemble du bloc D25_DICH avec des fonctionnalités supplémentaires — par ex.: champs alarmes & modes — mais, exécution & communications légèrement plus lentes.
	D25_DI_T	Voie entrée logique unique	Comme le bloc D25_DI, mais avec une alarme d'entrée temporisée.
	D25_DOCH	Voie sortie logique unique	Gère les fonctionnalités 2500 sans "extra". Exécution & communication légèrement plus rapides que le D25_DO. Voir nota [1].
	D25_DO	Voie sortie logique unique	Sur-ensemble du bloc D25_DOCH avec des fonctionnalités supplémentaires — par ex.: champs alarmes & modes — mais, exécution & communications légèrement plus lentes. Voir nota [2].

[1] Non recommandé pour les écritures continues de SFC. [2] Recommandé pour des écritures continues de SFC, sans surcharge des communications.

Table 15-1A Notes d'utilisation des blocs E/S DCM

FICHIERS DE DEFINITION DE PROFIL

Chaque bloc de fonction dans la catégorie DCM comprend un champ de paramètre appelé *Profile* — normalement laissé à son état par défaut qui est vierge. Si vous voulez remplacer la correspondance des adresses de communication par défaut fournie avec un bloc particulier, il faut créer un *fichier de définition de profil* et saisir son nom dans ce champ (sans l'extension du fichier). Alors, lorsque le bloc est exécuté, c'est votre profil qui est adopté au lieu du profil par défaut.

NOTA. L'ensemble des blocs paramètres DCM_*8 sont exceptionnels, dans la mesure où aucun d'entre eux n'est fourni avec un profil par défaut. Vous **devez** donc pour chacun de ces blocs créer un fichier de définition de profil et saisir son nom dans le champ *Profile*.

Un fichier de définition de profil spécifie la correspondance des adresses d'un bloc et est au format texte à variables séparées par des virgules (CSV). La syntaxe détaillée dépend du protocole de communication utilisé — Modbus ou Profibus.

NOTA. Un nombre illimité de blocs DCM peuvent utiliser le même fichier de définition de profil, à condition que leur utilisation du registre (page d'adresses) soit la même dans l'instrument cible.

Fichiers de définition de profil Modbus

Les fichiers de définition de profil Modbus se distinguent par l'extension de fichier **.UYM**. Ces fichiers contiennent des lignes au format suivant, d'une longueur maximale de 60 caractères.

[...] signifie 'optionnel' et {...} indique 'zéro ou plus':

```
<Field>,<Register>[,<Type>[,"{<FnCode>,<FnCode>}"][,,"Comment"]]
```

où:

- **<Field>** est le nom du champ du bloc de base de données LIN qui fait l'objet de la projection, par ex. **PV**
- **<Register>** est la référence du registre Modbus pour le champ. Il peut s'agir d'un entier (par ex. **1024**) ou encore d'expressions sous la forme:

```
<Constant>[{+<Constant>*(<Fieldname>±<Constant>)}] ou  
<Constant>[{+<Constant>*(<Fieldname>±<Fieldname>±<Constant>)}]
```

où:

<Constant> est un entier.

<Fieldname> peut être n'importe quel nom de champ du bloc ayant une valeur entière de 16 bits, par ex.

Chan_No. Cette valeur est utilisée dans l'expression.

Exemple d'expression: **200+10*(Slot_No-1)+10*(Chan_No-1).**

Le résultat de l'évaluation de l'expression est un entier de 16 bits sans signe, mais tous les paramètres d'entrée sont des entiers de 16 bits avec signe, sauf le paramètre de décalage principal qui est un entier de 16 bits sans signe.

NOTA. Dans le menu *Parameter List > Address Display*, iTools affiche deux variantes d'adresse - *canonical et actual*. L'adresse canonique doit être utilisée pour la correspondance d'un registre à 16 bits (entier) dans l'instrument cible, et l'adresse réelle doit être utilisée pour la correspondance des champs réels (à point flottant). Notez que le doublement de l'adresse canonique et le fait de lui ajouter 32768 donne l'adresse réelle.

- **<Type>** définit le format des données pour les registres reçus sur Modbus, par ex. **REAL**, **TIME** ou **UINT**. La table 15-2 donne la liste de tous les formats de données possible et la table 15-3 indique les formats équivalents affichés dans le configurateur de terminal, ainsi que dans les colonnes Unités des tables des paramètres des blocs dans ce manuel.

Si **<Type>** est omis, n'importe quelle entrée commune existante est adoptée. Sinon, le format par défaut **UINT** est adopté.

Format de données	Description
BOOL	Valeur 0/1 en bit de poids faible
UINT	Entier à 16 bits sans signe
INT	Entier à 16 bits avec signe
USINT	Entier à 8 bits sans signe
SINT	Entier à 8 bits avec signe
UDINT	Entier à 32 bits sans signe
UDINT_X	Entier à 32 bits sans signe (*voir NOTA)
DINT	Entier à 32 bits avec signe
DINT_X	Entier à 32 bits avec signe (*voir NOTA)
TIME	Durée à 32 bits avec signe en millisecondes
STIME_ds	Durée à 16 bits en décisecondes (0.1 s)
STIME_dm	Durée à 16 bits en déciminutes (0.1 m)
STIME_dh	Durée à 16 bits en déciheures (0.1 h)
REAL	Valeur virgule flottante 32 bits IEEE dans 2 registres
REAL_X	Valeur virgule flottante 32 bits IEEE dans 2 registres (*voir NOTA)
SREAL_p1	Nombre à 16 bits en unités de 0.1
SREAL_p2	Nombre à 16 bits en unités de 0.01
SREAL_p3	Nombre à 16 bits en unités de 0.001
SREAL_p4	Nombre à 16 bits en unités de 0.0001

Table 15-2 Formats de données

*NOTA. Les deux mots de 16 bits de ces formats de données sont en ordre inverse par rapport aux formats non ‘_X’. Ils permettent en particulier de communiquer avec d’autres instruments LIN par l’intermédiaire de la passerelle esclave Modbus.

Configurateur	“Unités” manuel réf.	Types de DCM acceptables
Bool	V/F	BOOL
UInt8	(0-x)	BOOL, USINT, UINT
Int8	Entier	BOOL, SINT, INT
UInt16	(0-x)	BOOL, UINT, INT
Int16	Entier	BOOL, UINT, INT, REAL
UInt32	(0-x)	UDINT, DINT, TIME, UDINT_X, DINT_X
Int32	Entier	UDINT, DINT, TIME, UDINT_X, DINT_X
Time	hh:mm:ss	UDINT, DINT, TIME, UDINT_X, DINT_X
Subfield8*	CD hex	USINT
Subfield16*	ABCD hex	UINT, INT
Real32	Eng	REAL, SREAL_p1 à SREAL_p4, REAL_X
Enumeration*	Menu	UINT, INT
Alarms*	Alarmes V/F	(Ne peut être transféré en utilisant des blocs DCM)
Date	jj:mm:aa	(Ne peut être transféré en utilisant des blocs DCM)
BlockR	Alphanumérique	(Ne peut être transféré en utilisant des blocs DCM)

*Nom du type non affiché dans le configurateur de terminal

Table 15-3 Formats de données équivalents

- **<FnCode>** définit un code de fonction Modbus acceptable comme partie d'une liste, par ex. "3,4,6,16". Si aucune valeur **<FnCode>** n'est spécifiée, toute entrée commune existante est adoptée. Sinon, la liste des codes de fonctions prend la valeur par défaut de "3,4" (registres en lecture seule).
- **"Comment"** peut être une chaîne de caractères de commentaire, par ex. le paramètre particulier auquel la ligne renvoie ou tout autre texte utile. Ne dépassez pas le nombre maximum de caractères pour l'ensemble de la ligne (60 caractères).

Les profils par défaut fournis, ainsi que tout profil personnalisé requis, sont tous créés en utilisant la syntaxe qui vient d'être décrite. A titre d'exemple, voici le fichier de définition de profil Modbus par défaut pour le bloc D2X_LOOP (boucle série 2000):

```
PV,1+1024*(Slot_No-1),REAL,"3,4"
SP,2+1024*(Slot_No-1),REAL,"3,4,6,16"
Out,3+1024*(Slot_No-1),REAL,"3,4,6,16"
Mode_Blk,273+1024*(Slot_No-1),UINT,"3,4"
```

Fichiers de définition de profil Profibus

Les fichiers de définition de profil Profibus se distinguent par l'extension **.UYP**. Ces fichiers contiennent des lignes au format suivant d'une longueur maximale de 60 caractères.

[...] signifie 'optionnel' et {...} 'zéro ou plus':

<Field>,**<Address>**[,**<Type>**[,"{<Operation>,<Operation>"}][,,"Comment"]

où

- **<Field>** est le nom du champ du bloc de base de données LIN qui fait l'objet d'une projection, par ex. **PV**
- **<Address>** est la référence du registre Profibus pour le champ. Il peut soit s'agir d'un entier (par ex. **1024**) ou encore d'expressions sous la forme:

```
<Constant>[{+<Constant>*(<Fieldname>±<Constant>)}] or
<Constant>[{+<Constant>*(<Fieldname>±<Fieldname>±<Constant>)}]
```

où:

<Constant> est un entier.

<Fieldname> peut être n'importe quel nom de champ du bloc ayant une valeur entière de 16 bits, par ex.

Chan_No. Cette valeur est utilisée dans l'expression.

Exemple d'expression: **200+10*(Slot_No+Chan_No-1).**

Le résultat de l'évaluation de l'expression est un entier de 16 bits sans signe, mais tous les paramètres d'entrée sont des entiers de 16 bits avec signe, sauf le paramètre de décalage principal qui est un entier de 16 bits sans signe.

- **<Type>** définit le format des données pour les registres reçus sur Profibus, par ex. **REAL**, **TIME** ou **UINT**. La table 15-2 donne la liste de tous les formats de données

possible et la table 15-3 indique les formats équivalents affichés dans le configurateur de terminal, ainsi que dans les colonnes Unités des tables des paramètres des blocs dans ce manuel.

Si **<Type>** est omis, n'importe quelle entrée commune existante est adoptée. Sinon, le format par défaut **UINT** est adopté.

- **<Operation>** spécifie une opération de lecture/écriture autorisée comme partie d'une liste, par ex.: **"RC,WC"**. Les opérations disponibles sont **RC**, **WC**, **RA** et **WA**, où **R** = lecture, **W** = écriture, **C** = cyclique et **A** = acyclique. Si aucune **<Operation>** n'est spécifiée, l'opération adoptée par défaut est **RC** (lecture cyclique).
- **"Comment"** peut être une chaîne de caractères de commentaire, par ex. le paramètre particulier auquel la ligne renvoie ou tout autre texte utile. Ne dépassez pas le nombre maximum de caractères pour l'ensemble de la ligne (60 caractères).

Les profils par défaut fournis, ainsi que tout profil personnalisé requis, sont tous créés en utilisant la syntaxe qui vient d'être décrite.

Voici un exemple de fichier .UYP:

```
MV,16,UINT,"RC,WC",,"Recorder 4, channel 17"
```












D2X_LOOP: BLOC BOUCLE SÉRIE 2000

Fonction du bloc

Le bloc D2X_LOOP donne une vue d'un bloc de schéma de boucles correspondant à une boucle de régulation PID, exécutée dans l'un des instruments de la série 2000 (2200/2400/2500).

Paramètres du bloc

Les symboles de la table 15-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont donnés dans la section *Menu de spécifications du bloc* qui suit.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port communication & n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° de révision statique Fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (par défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° empl. boucle PID instrument (1-2, défaut=1)	Entier	
PV	Valeur variable procédé	Eng	 
SP	Valeur point de consigne	Eng	 
OP	Valeur sortie	Eng	 
Mode	Mode fonctionnement bloc (AUTO/MANUEL)	Menu	 
Alarms			  
Software	Anomalie mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur communication vers 2500 (ex. tempo)	V/F	
Config	Erreur config. commun. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-4 Paramètres du bloc D2X_LOOP

Menu de spécifications du bloc

Les informations suivants complètent celles de la table 15-4.

Dbase, Block, Type. Voir les détails sur ces champs “en-tête au chapitre 2; bloc ANIN.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFIDP_1/PROFIDP_2) Définition du port de communication et référence du protocole. Plusieurs ports peuvent être ainsi utilisés avec le même protocole. Le port par défaut est ‘MODBUS_1’. (Tous les éléments du menu ne peuvent être utilisés à l'heure actuelle).

Profile. Le nom de base d'un fichier de définition de profil. Tout bloc DCM block peut avoir un profil optionnel qui peut être utilisé pour remplacer la projection de l'adresse par défaut du bloc.

Voir les détails sur les *Fichiers de définition de profil* à la page 15-2.

ST_REV. Numéro de révision statique de la fondation Fieldbus. Chaque fois que *ST_REV* change ou est égal à 0, toutes les données statiques dans le module de contrôle dévolu — marquées [S] dans la table — sont relues.

Instr_No. Numéro de l'instrument en fonction de l'instrument. Pour les communications Modbus, *Instr_No* se situe dans la plage de 1 à 247. Pour les communications Profibus-DP, la plage est de 1 à 125. La valeur par défaut est de 0, ce qui signifie “hors ligne”. (*Notez que pour Profibus dans le T940, les valeurs 1 et 2 sont illégales*).

Slot_No. N° de l'emplacement de la boucle PID dans l'instrument. *Slot_No* peut prendre la valeur de “1” ou “2” dans un 2500, mais uniquement “1” pour les instruments de la série 2200/2400.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, bloc ANIN. Ces informations complètent celles de la table 15-4:

- **Config.** VRAI s'il y a une erreur de configuration des communications — par ex. adresse de communication dupliquée ou noeud erroné sur le réseau — ou si le fichier de définition du profile est erroné ou manquant.
- **Reset.** VRAI si le module de contrôle dévolu est réinitialisé — par ex. à la mise sous tension ou en basculant ou en sortant du mode de configuration, etc. Dans ce cas, les paramètres de fonctionnement statiques (marqués [S] dans la table 15-4) sont relus. (*Non mis en oeuvre à l'heure actuelle*).

D2X_TUNE: BLOC DE MISE AU POINT DES BOUCLES SÉRIE 2000




Fonction du bloc

Le bloc D2X_TUNE permet de mettre au point une boucle PID exécutée dans l'un des instruments de la série 2000 (2200/2400/2500).

NOTA. PID_1 implique la mise au point active de l'ensemble des paramètres d'un 2500, et est en lecture seule.

Paramètres du bloc

Les symboles utilisés dans la table 15-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont donnés dans la section *Menu de spécifications du bloc* qui suit.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port communicat. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique Fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (par défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° empl. boucle PID dans instrum. (1-4, défaut=1)	Entier	
Chan_No ^[S]	N° voie dans emplacement (1-2, défaut=1)	Entier	
XP ^[S]	Bande proportionnelle	Eng	
TI ^[S]	Heure intégrale	hh:mm:ss	
TD ^[S]	Heure dérivée	hh:mm:ss	
RelCgain ^[S]	Gain refroidissement relatif	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur communication vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
User_1	Alarme configurée par l'utilisateur	V/F	
User_2	Alarme configurée par l'utilisateur	V/F	
User_3	Alarme configurée par l'utilisateur	V/F	
User_4	Alarme configurée par l'utilisateur	V/F	
SnsBrk	Rupture capteur	V/F	
LoopBrk	Rupture boucle	V/F	
HtrFail	Défaut réchauffeur	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
CB_HI ^[S]	Cutback à l'état haut	Eng	
CB_LO ^[S]	Cutback à l'état bas	Eng	
ManReset ^[S]	Réinitialisation manuelle	Eng	
ActSet ^[S]	Sélect. paramètres mise point active (PID_1-2)	Menu	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-5 Paramètres du bloc D2X_TUNE

Menu de spécifications du bloc

Reportez à la section *D2X_LOOP Bloc boucle série 2000* à la page 15-6, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-5.

Chan_No. Définit quel ensemble de paramètres de mise au point PID du bloc sont mis en correspondance.

Ex.: Un *Chan_No* de '2' projette l'ensemble PID_2. La valeur par défaut est 1.

(Dans un 2500, seul PID_1 peut être utilisé.)

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, bloc ANIN. Ces informations complètent celles de la table 15-5:

■ **User_1 à User_4.** Ces paramètres représentent les quatre alarmes standard (maximum) qui peuvent être affectées dans l'instrument déporté pour chaque boucle. Vous pouvez les sélectionner parmi les six types d'alarmes suivantes:

Type d'alarme	Description
HighAbs	Alarme haute absolue
LowAbs	Alarme basse absolue
HighDev	Alarme d'écart haute
LowDev	Alarme d'écart basse
DevBand	Alarme de bande d'écart
Rate	Alarme de vitesse de variation

Dans le 2500 et dans le bloc DCM, chacune de ces quatre alarmes n'est identifiée que par le chiffre qui lui est affecté (1-4).

D25_LOOP: BLOC BOUCLE 2500

D25eLOOP: BLOC BOUCLE 2500

Fonction du bloc

Ces deux blocs donnent une vue d'une boucle de régulation PID exécutée dans un 2500. Le bloc D25_LOOP est utilisé avec le régulateur plus ancien 2500C (2 boucles) et le bloc D25eLOOP avec le régulateur E/S 2500E (8 boucles). Ces blocs sont identiques, à l'exception de la plage du paramètre *Slot_No*.

Paramètres du bloc

Les symboles utilisés dans la table 15-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont donnés dans la section *Menu de spécifications du bloc* qui suit.
















Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port communic. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplac. boucle PID dans instrument (défaut=1)	Entier	
PV	Valeur variable procédé	Eng	 
SP	Valeur point de consigne	Eng	 
OP	Valeur sortie	Eng	 
Mode	Mode fonctionnement bloc (AUTO/MANUEL)	Menu	
RemEnabl	VRAI active & FAUX désactive mode déporté	V/F	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Alm1	Alarme définie par l'utilisateur	V/F	
Alm2	Alarme définie par l'utilisateur	V/F	
Alm3	Alarme définie par l'utilisateur	V/F	
Alm4	Alarme définie par l'utilisateur	V/F	
Snsr_Brk	Rupture capteur	V/F	
LoopBrk	Rupture boucle	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme boucle	(AB)CD hex	 
Alm1_Act	Alarme 1 active	V/F —1	D
Alm1_Ack	Alarme 1 acquittée	V/F —2	
Alm2_Act	Alarme 2 active	V/F —4	
Alm2_Ack	Alarme 2 acquittée	V/F —8	
Alm3_Act	Alarme 3 active	V/F —1	C
Alm3_Ack	Alarme 3 acquittée	V/F —2	
Alm4_Act	Alarme 4 active	V/F —4	
Alm4_Ack	Alarme 4 acquittée	V/F —8	

Table 15-6 suite...

...Table 15-6 suite

Paramètre	Fonction	Unités	Etat
LoopSW	Mot état génération alarmes (<i>Non mis en oeuvre</i>)	ABCD hex	 
CtrlFrz	Gel régulation	V/F	D
Snsr_Brk	Rupture capteur PV	V/F	
SRL_Act	Limiteur vitesse consigne actif	V/F	
RemSpAct	Consigne déportée active	V/F	
Servo	Signal asservi PID	V/F	C
Debump	Signal anti-à-coups PID	V/F	
LoopBrk	Rupture boucle	V/F	
IntFrz	Gel intégrale	V/F	
RemFault	Défaut déporte	V/F	B
DirRev	Action directe (VRAI)/Inverse (FAUX)	V/F	
<i>Réserve</i>		V/F	
OP_Lim	Sortie limitée	V/F	
AutoTune	Auto mise au point active	V/F	A
AdapTune	Mise au point adaptative activée	V/F	
AutDroop	Compensation affaissement auto. activée	V/F	
Man_ModeMode.	VRAI = Manuel, FAUX = Auto	V/F	
SP1 ^[S]	Consigne 1	Eng	
SP1_HI ^[S]	Consigne 1 limite haute	Eng	
SP1_LO ^[S]	Consigne 1 limite basse	Eng	
SP2 ^[S]	Consigne 2	Eng	
SP2_HI ^[S]	Consigne 2 limite haute	Eng	
SP2_LO ^[S]	Consigne 2 limite basse	Eng	
SSEL ^[S]	Sélection consigne	Entier	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-6 Paramètres du bloc D25_LOOP & D25eLOOP

Menu de spécifications du bloc

Reportez à la section *D2X_LOOP Bloc boucle série 2000* à la page 15-6, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-6.

Slot_No. Il s'agit du "n° d'emplacement" de la boucle PID dans l'instrument 2500. *Slot_No* peut prendre une valeur de 1 à 4* dans le bloc D25_LOOP (ce qui correspond au régulateur E/S 2500C plus ancien) et de 1 à 8 dans le bloc D25eLOOP (pour le régulateur E/S 2500E).

*NOTA. Le régulateur E/S 2500C plus ancien permet de ne gérer que *deux* boucles PID à l'heure actuelle, et vous ne pouvez donc saisir qu'un *Slot_No* de 1 ou 2.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, bloc ANIN. Ces informations complètent celles de la table 15-6.

- **Alm1 à Alm4.** Ces paramètres représentent les quatre alarmes standard (maxi.) qui peuvent être affectées dans l'instrument déporté pour chaque boucle. Vous pouvez les sélectionner parmi les six types d'alarmes suivantes:

Type d'alarme	Description
HighAbs	Alarme haute absolue
LowAbs	Alarme basse absolue
HighDev	Alarme d'écart haute
LowDev	Alarme d'écart basse
DevBand	Alarme de bande d'écart
Rate	Alarme de vitesse de variation




Dans le 2500 et dans le bloc DCM, chacune de ces quatre alarmes n'est identifiée que par le chiffre qui lui est affecté (1-4).

D25_TUNE: BLOC DE MISE AU POINT DES BOUCLES 2500**D25eTUNE: BLOC DE MISE AU POINT DES BOUCLES 2500****Fonction du bloc**

Ces blocs permettent de mettre au point une boucle PID exécutée dans l'instrument 2500.
Le bloc D25_TUNE est utilisé avec le régulateur E/S 2500C plus ancien (2 boucles), et le bloc D25eTUNE avec le régulateur E/S 2500E (8 boucles). Les blocs sont identiques, sauf pour la plage du paramètre *Slot_No*.

Paramètres du bloc

Les symboles utilisés dans la table 15-7 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres dont les noms sont similaires sont données dans la section *Bloc D2X_TUNE* à la page 15-8, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplac. boucle PID dans instrument (défaut=1)	Entier	
Chan_No ^[S]	N° voie dans emplacement (1-3, défaut=1)	Entier	
XP ^[S]	Bande proportionnelle	Eng	
T ^[S]	Heure intégrale	hh:mm:ss	
TD ^[S]	Heure dérivée	hh:mm:ss	
RelCgain ^[S]	Gain relatif refroidissement	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. ou profil erroné	V/F	
Reset	<i>(Non mis en oeuvre)</i>	V/F	
User_1	Alarme configurée par l'utilisateur	V/F	
User_2	Alarme configurée par l'utilisateur	V/F	
User_3	Alarme configurée par l'utilisateur	V/F	
User_4	Alarme configurée par l'utilisateur	V/F	
SnsBrk	Rupture capteur	V/F	
LoopBrk	Rupture boucle	V/F	
HtrFail	Défaut réchauffeur	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
CB_HI ^[S]	Cutback à l'état haut	Eng	
CB_LO ^[S]	Cutback à l'état bas	Eng	
ManReset ^[S]	Réinitialisation manuelle	Eng	
ActSet ^[S]	Sélect. paramètres mise point act. (PID_1-3)	Menu	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-7 Paramètres du bloc D25_TUNE & D25eTUNE

Menu de spécifications du bloc

Reportez à la section *D2X_TUNE Bloc mise au point boucles série 2000* à la page 15-8, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-7.

Slot_No. Il s'agit du "n° d'emplacement" de la boucle PID dans l'instrument 2500.

Slot_No peut prendre une valeur de 1 à 4* dans le bloc D25_TUNE (ce qui correspond au régulateur E/S 2500C plus ancien) et de 1 à 8 dans le bloc D25eTUNE (pour le régulateur E/S 2500E).

*NOTA. Le régulateur E/S 2500C plus ancien permet de ne gérer que *deux* boucles PID à l'heure actuelle, et vous ne pouvez donc saisir qu'un *Slot_No* de 1 ou 2.

D25_RAMP: BLOC RAMPE POUR PROGRAMMEUR DE CONSIGNES

D25eRAMP: BLOC RAMPE POUR PROGRAMMEUR DE CONSIGNES

Fonction du bloc

Ces blocs sont utilisés par le programmeur de consignes du T2900 pour incrémenter ou décrémenter la rampe d'un point déporté du réseau. Le bloc D25_RAMP est utilisé avec le régulateur E/S 2500C plus ancien (2 boucles) et le bloc D25eRAMP avec le régulateur E/S 2500E (8 boucles). Les blocs sont identiques, à l'exception de la plage du paramètre *Slot_No*.




Paramètres du bloc

Les symboles de la table 15-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres dont les noms sont similaires sont données dans la section *D2X_LOOP Bloc boucle série 2000* à la page 15-6, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port communicat. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplac. de RAMP dans instrument (défaut=1)	Entier	
PV	Variable procédé	Eng	
TgtSP ^[D]	Consigne cible	Eng	
Rate ^[D]	Vitesse de variation demandée	Eng	
RampUnit ^[D]	Unités de la rampe (SEC/MIN/ HOUR)	Menu	
OutValeur active	consigne	Eng	
NewTgtSP ^[D]	Consigne cible suivante	Eng	
NewRate ^[D]	Vitesse segment suivant	Eng	
Sync ^[D]	Charge valeurs de NewTgtSP & NewRate	V/F	
SyncPV ^[N]	Asservi à PV au début de la rampe	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur communication vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Holdback	Rampe en rétention	V/F	
NotRamp	Rétention irrécupérable (càd pas de rampe)	V/F	
Combined	Combinaison de tous les bits d'alarmes	V/F	
Complete	Rampe terminée	V/F	
Active	Rampe en cours d'exécution	V/F	
NotActiv	Rampe heure pas active	Heure	
Holdback	Rampe en rétention	V/F	
Hold	Rampe en HOLD (maintien)	V/F	
HB_Mode ^[D]	Type de rétention (NONE/LOW/HIGH/DEV)	Menu	
HB_Value ^[D]	Valeur de rétention	Eng	

Table 15-8 suite...

...Table 15-8 suite

Paramètre	Fonction	Unités	Etat
HB_Dis	Rétention désactivée	V/F	
MinHback ^{[S][N]}	(Fonction réservée)	Eng	
OorHback ^[N]	Rampe rétention & 'déf. vitesse' (Non mis en oeuvre)	V/F	
RemEnabl	Mis à 1 par SPP début programme, raz fin	V/F	
StLocRmt ^[S]	Paramètres de Local/déporté au démarrage	Menu	
StWspCh ^[S]	Paramètres de WSP au démarrage	Menu	
StHold ^[S]	Paramètres de Hold au démarrage	Menu	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

^[N] Aucune communication avec ce champ n'est mis en oeuvre à l'heure actuelle.

^[D] Ecriture directe par le programmeur de consignes.

Table 15-8 Paramètres des blocs D25_RAMP & D25eRAMP

Menu de spécifications du bloc

Reportez à la section *D2X_LOOP Bloc boucle série 2000* à la page 15-6, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-8.

Slot_No. Il s'agit du "n° d'emplacement" de la boucle PID dans l'instrument 2500. *Slot_No* peut prendre une valeur de 1 à 4* dans le bloc D25_RAMP (ce qui correspond au régulateur E/S 2500C plus ancien) et de 1 à 8 dans le bloc D25eRAMP (pour le régulateur E/S 2500E).

*NOTA. Le régulateur E/S 2500C plus ancien permet de ne gérer que *deux* boucles PID à l'heure actuelle, et vous ne pouvez donc saisir qu'un *Slot_No* de 1 ou 2.

D25_MOD: BLOC MODULE ANONYME

Fonction du bloc

Le bloc D25_MOD fournit une vue d'un seul module E/S physique dans un 2500. Sa principale fonction est d'identifier le module. Tous les blocs module spécifiques sont des sur-ensembles de ce bloc.

Paramètres du bloc

Les symboles de la table 15-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplacement module dans instrument (1-16)	Entier	
Req_ID	Type de module demandé	ABCD hex	
Act_ID	Type de module réel	ABCD hex	
Version ^[S]	N° de version du module	Entier	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex tempo)	V/F	
Config	VRAI si erreur config comm ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
ID Err	VRAI si Mod_Stat = No_Mod Bad_ID Unk_ID	V/F	
Mod Comm	VRAI si Mod_Stat = Comm_Err	V/F	
Ch1_HI	Voie 1 HIGH Voie 2 HIGH Voie 1 LOW Voie 2 LOW (source: AlarmSW)	V/F	
Ch2_HI		V/F	
Ch1_LO		V/F	
Ch2_LO		V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot d'état alarme 2500	(ABC)D hex	
Chan1_HI	Voie 1 HIGH	V/F — 1	D
Chan2_HI	Voie 2 HIGH	V/F — 2	
Chan1_LO	Voie 1 LOW	V/F — 4	
Chan2_LO	Voie 2 LOW	V/F — 8	
Mod_Stat	Etat module	(ABC)D hex	
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur module communication	V/F — 8	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-9 Paramètre du bloc D25_MOD

Menu de spécifications du bloc

Reportez à la section *D2X_LOOP Bloc boucle série 2000* à la page 15-6, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom également utilisés par ce bloc.

Les informations suivantes complètent celle de la table 15-9.

Req_ID. Type de module demandé. La table 15-10 donne la liste des types de modules définis à l'heure actuelle.

Type de module	Valeur hexadécimale 'ABCD'
Vide	0000
DI4	0001
DO4, Alim. externe — 10 mA	0010
DO4, Alim. externe — 100 mA	0011
DO4, Alim. interne — 10 mA	0012
RL4	0020
AI2	0040
AO2	0050

Table 15-10 Types de modules sélectionnables par l'intermédiaire du paramètre Req_ID

Act_ID. Type de module réel. Voir table 15-10.

D25_AI2: BLOC MODULE ENTREE ANALOGIQUE A DEUX VOIES**D25_AI3: BLOC MODULE ENTREE ANALOGIQUE A TROIS VOIES****D25_AI4: BLOC MODULE ENTREE ANALOGIQUE A QUATRE VOIES****Fonction du bloc**

Le bloc D25_AI2 (ou D25_AI3, D25_AI4) exécuté dans un T2900 permet de communiquer avec un module d'entrée analogique à 2 voies (ou 3/4 voies) fonctionnant dans un 2500 déporté. Les paramètres du bloc sont projetés sur les valeurs PV déportées et leurs états. Les PV analogiques entrées dans le 2500 déporté apparaissent comme des sorties rattachables dans le bloc local.

Paramètres du bloc

Les symboles de la table 15-11 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres dont les noms sont similaires sont données dans la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*.








Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Version ^[S]	N° version module	Entier	
PV1-PV4*	Valeurs PV entrées voies (sorties du bloc)	Eng	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
ID Err	VRAI si Mod_Stat = No_Mod Bad_ID Unk_ID	V/F	
Mod Comm	VRAI si Mod_Stat = Comm_Err	V/F	
PV1_HI	Voie 1 HIGH	V/F	
PV2_HI	Voie 2 HIGH	V/F	
PV1_LO	Voie 1 LOW	V/F	
PV2_LO	Voie 2 LOW	V/F	
PV3_HI	Voie 3 HIGH (pas D25_AI2)	V/F	
PV4_HI	Voie 4 HIGH (D25_AI4 seult)	V/F	
PV3_LO	Voie 3 LOW (pas D25_AI2)	V/F	
PV4_LO	Voie 4 LOW (D25_AI4 seult)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme 2500	ABCD hex	
PV1_HI	Voie 1 HIGH	V/F	
Réserve		V/F	
PV2_HI	Voie 2 HIGH	V/F	
Réserve		V/F	

Table 15-11 suite...

...Table 15-11 suite

Paramètre	Fonction	Unités	Etat
PV1_LO	Voie 1 LOW	V/F —1	C
Réserve		V/F —2	
PV2_LO	Voie 2 LOW	V/F —4	
Réserve		V/F —8	
PV3_HI	Voie 3 HIGH (<i>pas bloc D25_AI2</i>)	V/F —1	B
Réserve		V/F —2	
PV4_HI	Voie 4 HIGH (<i>bloc D25_AI4 seult</i>)	V/F —4	
Réserve		V/F —8	
PV3_LO	Voie 3 LOW (<i>pas bloc D25_AI2</i>)	V/F —1	A
Réserve		V/F —2	
PV4_LO	Voie 4 LOW (<i>bloc D25_AI4 seult</i>)	V/F —4	
Réserve		V/F —8	
Mod_Stat	Etat module	(ABC)D hex	
No_Mod	Module manquant	V/F —1	D
Bad_ID	ID module erronée	V/F —2	
Unk_ID	ID module non reconnue	V/F —4	
Comm_Err	Erreur communication module	V/F —8	
Req_ID	Type de module demandé	ABCD hex	
Act_ID	Type de module réel	ABCD hex	
PV1_Stat-PV4_Stat*	Etat voies 1 to 4*	(AB)CD hex	 
Snsr_Brk	Rupt. capteur détectée (ou PV hors table linéar.)	V/F —1	D
CJC_Fail	Déf. compens. soudure froide (<i>D25_AI2 seult</i>)	V/F —2	
Not_Used	Voie non utilisée	V/F —4	
Réserve		V/F —8	
Init	Initialisation — valeur non mise à jour	V/F —1	C
Inv_Cal	Etalonnage entrée analogique incorrect	V/F —2	
Réserve		V/F —4	
Mod_Fail	Déf. module (combin. tous bits Mod_Stat)	V/F —8	

[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

*Voies 1 - 4 pour le bloc D25_AI4, 1 - 3 pour le bloc D25_AI3, 1 - 2 pour le D25_AI2 bloc.

Table 15-11 Paramètres du bloc D25_AI2, D25_AI3, & D25_AI4

Menu de spécifications du bloc

Reportez-vous à la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*, pour obtenir des informations supplémentaires sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_AO2: BLOC MODULE SORTIE ANALOGIQUE A2 VOIES**D25_AO4: BLOC MODULE SORTIE ANALOGIQUE A 4 VOIES****Fonction du bloc**

Les blocs D25_AO2 (ou D25_AO4) exécutés dans un T2900 permettent de communiquer avec un module de sortie analogique à deux voies (ou à 4 voies) fonctionnant dans un 2500 déporté. Les paramètres du bloc sont projetés sur les valeurs PV déportées et leurs états. Les PV analogiques générées localement sont rattachées au bloc et apparaissent comme des sorties dans le module 2500 déporté.

NOTA. Les blocs peuvent également être utilisés comme modules de sortie logique configurés comme sorties basées sur le temps. C'est le seul moyen d'utiliser le bloc D25_AO4 à l'heure actuelle, parce qu'un module de sortie analogique à 4 voies n'est pas encore disponible.

Paramètres du bloc

Les symboles de la table 15-12 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*.








Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° de révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Version ^[S]	N° de version module	Entier	
OP1-OP4*	Valeurs sorties voies (entrées dans bloc)	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm ou profile erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
ID Err	VRAI si Mod_Stat = No_Mod Bad_ID Unk_ID	V/F	
Mod Comm	VRAI si Mod_Stat = Comm_Err	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme 2500	ABCD hex	
Mod_Stat	Etat module	(ABC)D hex	
No_Mod	Module manquant	V/F	1
Bad_ID	ID module erronée	V/F	2
Unk_ID	ID module non reconnue	V/F	4
Comm_Err	Erreur comm module	V/F	8

Table 15-12 suite...

...Table 15-12 suite

Paramètre	Fonction	Unités	Etat
Req_ID	Type de module demandé	ABCD hex	
Act_ID	Type de module réel	ABCD hex	
OP1_Stat-OP4_Stat*	Etat voie 1 à 4*	(AB)CD hex	
Réserve		V/F —1	D
Réserve		V/F —2	
Not_Used	Voie non utilisée	V/F —4	
OP_Sat	Sortie analogique saturée	V/F —8	
Init	Initialisation — valeur non mise à jour	V/F —1	C
Inv_Cal	Etalonnage entrée analogique incorrect	V/F —2	
Réserve		V/F —4	
Mod_Fail	Déf. module (combin. tous bits Mod_Stat)	V/F —8	

[5] ‘Données ‘Statiques’, uniquement lues en cas de réinitialisation du maître ou de l’esclave. Les autres données sont lues/écrites à chaque cycle de communication.

*Voies1 - 4 pour le bloc D25_AO4 , 1 - 2 pour le bloc D25_AO2.

Table 15-12 Paramètres des blocs D25_AO2 & D25_AO4

Menu de spécifications du bloc

Reportez-vous à la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*, pour obtenir des informations supplémentaires sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_DI4: BLOC MODULE ENTREE LOGIQUE A 4 VOIES

D25_DI6: BLOC MODULE ENTREE LOGIQUE A 6 VOIES

D25_DI8: BLOC MODULE ENTREE LOGIQUE A 8 VOIES

Fonction du bloc

Les blocs D25_DI4 (ou D25_DI6, D25_DI8) exécutés dans un T2900 permettent de communiquer avec un module d’entrée logique à 4 voies (ou à 6/8 voies) fonctionnant dans un 2500 déporté. Les paramètres du bloc sont projetés sur les valeurs PV déportées et leurs états. Les valeurs logiques entrées dans le 2500 déportés apparaissent comme des sorties rattachables dans le bloc local.

Paramètres du bloc

Les symboles de la table 15-13 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*.











Paramètre	Fonction	Unité	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Version ^[S]	N° version module	Entier	
In1 à In8*	Valeurs entrées voies 1 à 8* (sorties du bloc)	V/F	 
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
ID Err	VRAI si Mod_Stat = No_Mod Bad_ID Unk_ID	V/F	
Mod Comm	VRAI si Mod_Stat = Comm_Err	V/F	
In1 to In8*	Voies 1 à 8* en alarm (source: Mod_Stat)	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	
AlarmSW	Mot état alarme 2500	ABCD hex	
In1_Act	Voie 1 alarme active	V/F	<div><div>1</div><div>2</div><div>4</div><div>8</div></div> <div>D</div>
In1_Ack	Voie 1 alarme acquittée	V/F	
In2_Act	Voie 2 alarme active	V/F	
In2_Ack	Voie 2 alarme acquittée	V/F	
In3_Act	Voie 3 alarme active	V/F	<div><div>1</div><div>2</div><div>4</div><div>8</div></div> <div>C</div>
In3_Ack	Voie 3 alarme acquittée	V/F	
In4_Act	Voie 4 alarme active	V/F	
In4_Ack	Voie 4 alarme acquittée	V/F	

Table 15-13 suite...

...Table 15-13 suite

Paramètre	Fonction	Unités	Etat
In5_Act	Voie 5 alarme active	V/F —1	B
In5_Ack	Voie 5 alarme acquittée	V/F —2	
In6_Act	Voie 6 alarme active	V/F —4	
In6_Ack	Voie 6 alarme acquittée	V/F —8	
(pas D25_DI4)			
In7_Act	Voie 7 alarme active	V/F —1	A
In7_Ack	Voie 7 alarme acquittée	V/F —2	
In8_Act	Voie 8 alarme active	V/F —4	
In8_Ack	Voie 8 alarme acquittée	V/F —8	
(D25_DI8 seule)			
Mod_Stat	Etat module	(ABC)D hex	
No_Mod	Module manquant	V/F —1	D
Bad_ID	ID module erroné	V/F —2	
Unk_ID	ID module non reconnue	V/F —4	
Comm_Err	Erreur comm. module	V/F —8	
Req_ID	Type de module demandé	ABCD hex	
Act_ID	Type de module réel	ABCD hex	
In1_Stat-In8_Stat*	Etat voies 1 à 8*	(AB)CD hex	 
Réserve		V/F —1	D
Réserve		V/F —2	
Not_Used	Voie non utilisée	V/F —4	
Réserve		V/F —8	
Init	Initialisation — valeur non mise à jour	V/F —1	C
Réserve		V/F —2	
Réserve		V/F —4	
Mod_Fail	Déf. module (combin. tous bits Mod_Stat)	V/F —8	

[5] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

*Voies 1 - 8 pour le bloc D25_DI8, 1 - 6 pour le bloc D25_DI6, 1 - 4 pour le bloc D25_DI4.

Table 15-13 Paramètres des blocs D25_DI4, D25_DI6, & D25_DI8

Menu de spécifications du bloc

Reportez-vous à la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*, pour obtenir des informations supplémentaires sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_DO4: BLOC MODULE SORTIE LOGIQUE A 4 VOIES

Fonction du bloc

Le bloc D25_DO4 exécuté dans un T2900 permet de communiquer avec un module de sortie logique à quatre voies fonctionnant dans un 2500 déporté. Les paramètres du bloc sont projetés sur les valeurs PV déportées et leurs états. Les valeurs logiques générées localement sont rattachées au bloc et apparaissent comme des sorties dans le module 2500 déporté.

Paramètres du bloc

Les symboles de la table 15-14 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*.








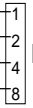




Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondaton Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Version ^[S]	N° version module	Entier	
Out1 to Out4	Valeurs sorties voies 1 à 4 (entrées bloc)	V/F	 
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm ou profil erroné	V/F	
Reset	<i>(Non mis en oeuvre)</i>	V/F	
ID Err	VRAI si Mod_Stat = No_Mod Bad_ID Unk_ID	V/F	
Mod Comm	VRAI si Mod_Stat = Comm_Err	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme 2500	ABCD hex	
Mod_Stat	Etat module	(ABC)D hex	
No_Mod	Module manquant	V/F	
Bad_ID	ID module erronée	V/F	
Unk_ID	ID module non reconnue	V/F	
Comm_Err	Erreur comm. module	V/F	
Req_ID	Type de module demandé	ABCD hex	
Act_ID	Type de module réel	ABCD hex	

Table 15-14 suite...

...Table 15-14 suite

Paramètre	Fonction	Unités	Etat
Out1Stat à Out4Stat	Etat voies 1 à 4	(AB)CD hex	  
Réserve		V/F — 1	D
Réserve		V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
Réserve		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
Réserve		V/F — 2	
Réserve		V/F — 4	
Mod_Fail	Déf. module (combin. tous bits Mod_Stat)	V/F — 8	

[5] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-14 Paramètres du bloc D25_DO4

Menu de spécifications du bloc

Reportez-vous à la section *D25_MOD Bloc module anonyme* à la page 15-17, sous *Menu de spécifications du bloc*, pour obtenir des informations supplémentaires sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_AICH: BLOC D’ENTREE ANALOGIQUE

Fonction du bloc

Le bloc D25_AICH exécuté dans un T2900 permet de communiquer avec une seule voie d’entrée analogique fonctionnant dans un module 2500 déporté. Les paramètres des blocs sont projetés sur les états déportés et génèrent des alarmes d’état locales. Une PV analogique entrée en mode déporté apparaît comme une sortie rattachable dans le bloc local.

Paramètres du bloc

Les symboles de la table15-15 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.










Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défault 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie dans module (1-4, défaut=1)	Entier	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Snsr_Brk	VRAI si PV_Stat.Snsr_Brk	VRAI	V/F
CJC_Fail	VRAI si PV_Stat.CJC_Fail	VRAI	V/F
Not_Used	VRAI si PV_Stat.Not_Used	VRAI	V/F
Init	VRAI si PV_Stat.Init	VRAI	V/F
Inv_Cal	VRAI si PV_Stat.Inv_Cal	VRAI	V/F
Mod_Fail	VRAI si PV_Stat.Mod_Fail	VRAI	V/F
HiAbs	VRAI si PV en alarme haute absolue	V/F	
LoAbs	VRAI si PV en alarme basse absolue	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	
AlarmSW	Mot d’état alarme pour module	ABCD hex	 
PV	Valeur entrée analogique (sortie du bloc)	Eng	 

Table 15-15 suite...

...Table 15-15 suite

Paramètre	Fonction	Unités	Etat
PV_Stat	Etat voie entrée	(AB)CD hex	 
Snsr_Brk	Rupture capteur détectée (ou PV table linéaris.)	V/F — 1	D
CJC_Fail	Défaut compensation soudure froide	V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
Réserve		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
Inv_Cal	Etalonnage entrée analogique incorrecte	V/F — 2	
Réserve		V/F — 4	
Mod_Fail	Déf. module (combin. bits mots état module)	V/F — 8	

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-15 Paramètres du bloc D25_AICH

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 15-15.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFIDP_1/PROFIDP_2) Définition du port de communication et du numéro du protocole. Ceci permet d'utiliser plusieurs ports sur le même protocole. La valeur par défaut est 'MODBUS_1'. *(Tous les éléments du menu ne sont pas gérés à l'heure actuelle).*

Profile. Le nom de base d'un fichier de définition de profil. Tout bloc DCM peut avoir un profil optionnel qui peut être utilisé pour remplacer la correspondance d'adresse par défaut du bloc. Les profils sont définis dans un fichier au format CSV et son contenu dépend du protocole utilisé.

ST_REV. N° de révision statique de la fondation Fieldbus. Chaque fois que *ST_REV* change ou est égal à 0, toutes les données statiques dans le module de contrôle dévolu — marquées d'un [S] dans la table — sont relues.

Instr_No. N° de l'instrument, en fonction du protocole. Pour les communications Modbus, *Instr_No* se situe dans la plage de 1 à 247. Pour les communications Profibus-DP, la plage est de 1 à 125. La valeur par défaut est de 0, ce qui signifie "hors ligne". *(Notez que pour Profibus dans le T940, les valeurs de 1 et 2 ne sont pas autorisées).*

Slot_No. N° de l'emplacement du module dans l'instrument, c'est à dire le numéro d'instance du module de contrôle dévolu dans la plage de 1 à 16. La valeur par défaut est de 1.

Chan_No. N° de voie dans le module, utilisée pour identifier une voie E/S dans un module multi-voies. *Chan_No* accepte des entrées dans la plage de 1 à 4. La valeur par défaut est de 1.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, Bloc ANIN.

NOTA. Ces alarmes sont activées/désactivées voie par voie dans le 2500 et doivent être activées/désactivées de la même manière dans les T2900/T800/T940 pour qu'elles correspondent très exactement. L'état activé/désactivé peut être lu dans le 2500, mais n'est pas importé dans le T2900/T800/T940.

Ces informations complètent celles de la table 15-15:

- **Config.** VRAI s'il y a une erreur de configuration des communications - par ex. adresse de communication dupliquée ou noeud non valable sur le réseau — ou si le fichier de définition de profil n'est pas valable ou est manquant.
- **Reset.** VRAI si le module de contrôle dévolu est "réinitialisé" — par ex. mise sous tension ou passage en mode config ou sortie du mode config, etc. Dans ce cas, les paramètres de fonctionnement statique (marqués d'un [S] dans la table 15-15) sont relus. (*Non mis en oeuvre à l'heure actuelle*).

D25_AOCH: BLOC SORTIE ANALOGIQUE




Fonction du bloc

Le bloc D25_AOCH exécuté dans un T2900 permet de communiquer avec une seule voie de sortie analogique fonctionnant dans un module 2500 déporté. Les paramètres des blocs sont projetés sur les états déportés et génèrent des alarmes d'état locales. Une PV analogique rattachée au bloc local apparaît comme une sortie au niveau du module déporté.

NOTA. Le bloc D25_AOCH peut également être utilisé avec des modules de sortie logique configurés comme des sorties basées sur le temps.

Paramètres du bloc

Les symboles de la table 15-16 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant un nom similaire sont données dans la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[5]	Définit port comm. et n° protocole	Menu	
Profile ^[5]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[5]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[5]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[5]	N° voie dans module (1-4, défaut=1)	Entier	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)V/F		
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	<i>(Non mise en oeuvre)</i>	V/F	
Not_Used	VRAI si OP_Stat.Not_Used VRAI	V/F	
OP_Sat	VRAI si OP_Stat.OP_Sat VRAI	V/F	
Init	VRAI si OP_Stat.Init VRAI	V/F	
Inv_Cal	VRAI si OP_Stat.Inv_Cal VRAI	V/F	
Mod_Fail	VRAI si OP_Stat.Mod_Fail VRAI	V/F	
Combined	Combinaison de tous les bits d'alarmes	V/F	
OP	Valeur sortie analogique (entrée bloc)	Eng	
OP_Stat	Etat voie sortie	(AB)CD hex	
Réserve		V/F	—1
Réserve		V/F	—2
Not_Used	Voie non utilisée	V/F	—4
OP_Sat	Sortie analogique saturée	V/F	—8
Init	Initialisation — valeur non mise à jour	V/F	—1
Inv_Cal	Etalonnage sortie analogique incorrect	V/F	—2
Réserve		V/F	—4
Mod_Fail	Déf. module (combin. tous bits mot état module)V/F	V/F	—8

^[5] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-16 Paramètres du bloc D25_AOCH

Menu de spécifications du bloc

Reportez-vous à la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*, pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_DICH: BLOC ENTREE LOGIQUE

Fonction du bloc

Le bloc D25_DICH exécuté dans un T2900 permet de communiquer avec une seule voie d'entrée logique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur les états déportés et génèrent des alarmes d'état locales. Une entrée logique au niveau du module déporté apparaît comme une sortie rattachable dans le bloc local.

Paramètres du bloc

Les symboles de la table 15-17 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie dans module (1-8)	Entier	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mise en oeuvre)	V/F	
Not_Used	VRAI si In_Stat.Not_Used VRAI	V/F	
Init	VRAI si In_Stat.Init VRAI	V/F	
Mod_Fail	VRAI si In_Stat.Mod_Fail VRAI	V/F	
In	VRAI si alarmé configurée dans 2500 VRAI	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme pour module	ABCD hex	
In	Valeur entrée logique (sortie du bloc)	V/F	
In_Stat	Etat voie entrée	(AB)CD hex	
Réserve		V/F	D
Réserve		V/F	
Not_Used	Voie non utilisée	V/F	
Réserve		V/F	
Init	Initialisation — valeur non mise à jour	V/F	C
Réserve		V/F	
Réserve		V/F	
Mod_Fail	Déf. module fault (combin. bits mot état module)	V/F	

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-17 Paramètres du bloc D25_DICH

Menu de spécifications du bloc

Reportez-vous à la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*, pour obtenir de plus amples informations sur des paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivants complètent celles de la table 15-17:

Alarms. Voir la description générale du champs *Alarms* au chapitre 2, Bloc ANIN.

- **In.** Ce bit d'alarme passe à l'état VRAI, si l'alarme de la voie d'entrée logique unique configurée dans le 2500 déporté est déclenchée. L'alarme du 2500 peut être configurée en fonction de l'un des cinq types de la table 15-18.

Type d'alarme ...	est déclenché si ...
ISTRUE	l'entrée logique est VRAIE
ISFALSE	l'entrée logique est FAUSSE
GOFALSE	l'entrée logique passe de VRAIE à FAUSSE
GOTRUE	l'entrée logique passe de FAUSSE à VRAIE
CHANGE	l'entrée logique change d'état dans un sens ou dans l'autre

Table 15-18 Types d'alarmes logiques du 2500







D25_DOCH: BLOC SORTIE LOGIQUE

Fonction du bloc

Le bloc D25_DOCH exécuté dans un T2900 permet de communiquer avec une voie de sortie logique unique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur les états déportés et génèrent des alarmes d'état locales. Un signal logique rattachés au bloc local apparaît comme une sortie au niveau du module déporté.

Paramètres du bloc

Les symboles de la table 15-19 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie dans module (1-8)	Entier	
Alarms   			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Not_Used	VRAI si Out_Stat.Not_Used VRAI	V/F	
Init	VRAI si Out_Stat.Init VRAI	V/F	
Mod_Fail	VRAI si Out_Stat.Mod_Fail VRAI	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Out	Valeur sortie logique (entrée bloc)	V/F	
Out_Stat	Etat voie de sortie	(AB)CD hex	 
Réserve		V/F	←1
Réserve		V/F	←2
Not_Used	Voie non utilisée	V/F	←4
Réserve		V/F	←8
Init	Initialisation — valeur non mise à jour	V/F	←1
Réserve		V/F	←2
Réserve		V/F	←4
Mod_Fail	Déf. module (combinaison bits mot état module)	V/F	←8

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-19 Paramètres du bloc D25_DOCH

Menu de spécifications du bloc

Reportez-vous à la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*, pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.






D25_R_CV: BLOC VALEURS CALCULEES REELLES A LIAISONS FILAIRES UTILISATEUR

Fonction du bloc

Le bloc D25_R_CV exécuté dans un T2900 permet d'accéder aux valeurs calculées réelles (CV) à liaisons filaires utilisateur dans un module 2500 déporté. Chaque bloc D25_R_CV peut accéder à un bloc de huit valeurs calculées maximum. Les valeurs calculées déportées apparaissent comme des sorties rattachables en lecture seule dans le bloc D25_R_CV local.

Paramètres du bloc

Les symboles de la table 15-20 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° de révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° du bloc des CV à accéder (1 ou 2)	Entier	
CV_1 à CV_8	Valeurs ≤ 8 CV (Slot_No définit 1er ou 2ème bloc)	Eng	
Alarms			 
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	  
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-20 Paramètres du bloc D25_R_CV

Menu de spécifications du bloc

Reportez-vous à la section *D25_AICH Bloc entrée analogique* à la page 15-27, sous *Menu de spécifications du bloc*, pour obtenir des informations supplémentaires sur des paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-20:

Slot_No. Ce paramètre permet de diviser les valeurs calculées en deux blocs de huit valeurs consécutives pour réduire l'activité de communication, lorsque pas plus de huit valeurs ne sont nécessaires. Un bloc de fonction D25_R_CV accède à chaque bloc CV. Un *Slot_No* de '1' définit l'accès aux huit premières valeurs calculées dans le 2500 déporté (CV1 à CV8), qui sont écrites dans les champs *CV_1* à *CV_8*. Un *Slot_No* de '2' permet d'accéder au second bloc de valeurs calculées (CV9 et CV16), qui sont écrites dans les champs *CV_1* à *CV_8*.

D25_B_CV: BLOC VALEURS CALCULEES BOOLEENNES A LIAISONS
FILAIRES UTILISATEUR

Fonction du bloc

Le bloc D25_B_CV exécuté dans un T2900 permet d’accéder aux valeurs calculées (CV) booléennes à liaisons filaires utilisateur dans un module 2500 déporté. Chaque bloc D25_B_CV peut accéder à un bloc de huit valeurs calculées maximum. Les CV déportées apparaissent comme des sorties rattachables en lecture seule dans le bloc D25_B_CV local.

Paramètres du bloc

Les symboles de la table 15-21 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_R_CV Bloc valeur calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° bloc de CV à accéder (1 ou 2)	Entier	
CV_1 à CV_8	Valeurs ≤ 8 CV (Slot_No définit 1er ou 2ème bloc)	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	

^[S] ‘Données ‘Statiques’, uniquement lues en cas de réinitialisation du maître ou de l’esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-21 Paramètres du bloc D25_B_CV

Menu de spécifications du bloc

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.





D25_R_UV: BLOC VALEURS REELLES UTILISATEUR

Fonction du bloc

Le bloc D25_R_UV exécuté dans un T2900 permet d’accéder aux huit valeurs réelles utilisateur (UV) dans un 2500 déporté. Les UV déportées apparaissent dans le bloc D25_R_UV local comme des entrées et sorties rattachables.

Paramètres du bloc

Les symboles de la table 15-22 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	(Fixe à '1')	Entier	
UV_1 to UV_8	Huit valeurs utilisateur (maxi)	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

^[S] 'Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-22 Paramètres du bloc D25_R_UV

Menu de spécifications du bloc

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.






D25_UALM: BLOC ALARME UTILISATEUR 2500

Fonction du bloc

Le bloc D25_UALM exécuté dans un T2900 permet d’accéder aux alarmes utilisateur analogique ou logique dans un 2500 déporté. Les bits d’état des alarmes déportées peuvent être utilisés par le bloc pour générer des alarmes T2900 locales correspondantes.

Paramètres du bloc

Les symboles de la table 15-23 sont expliqués au chapitre 1. Des informations supplémentaires sur des paramètres portant des noms similaires sont données dans la section *D25_R_CV Bloc valeur calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No	Type alarme utilis. (1=analog., 2=booléen.)	Entier	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (eX. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
User1-User4	Alarmes générées par AlarmSW	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	
AlarmSW	Bits alarme utilisateur (huit maxi.)	(AB)CD hex	 

^[S] ‘Données ‘Statiques’, uniquement lues en cas de réinitialisation du maître ou de l’esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-23 Paramètres du bloc D25_UALM

Menu de spécifications du bloc

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.

- DCM_I8: BLOC PARAMETRE ENTIER 16 BITS AVEC SIGNE A 8 VOIES
- DCM_S8: BLOC PARAMETRE ENTIER 8 BITS AVEC SIGNE A 8 VOIES
- DCM_UI8: BLOC PARAMETRE ENTIER 16 BITS SANS SIGNE A 8 VOIES
- DCM_US8: BLOC PARAMETRE ENTIER 8 BITS SANS SIGNE A 8 VOIES
- DCM_R8: BLOC PARAMETRE NOMBRE REEL À 8 VOIES
- DCM_T8: BLOC PARAMETRE DUREE A 8 VOIES
- DCM_W8: BLOC PARAMETRE MOT/SOUS-CHAMP A 8 VOIES
- DCM_Y8: BLOC PARAMETRE OCTET HEX 'AB' A 8 VOIES
- DCM_B8: BLOC PARAMETRE BOOLEEN A 8 VOIES
- DCM_D8: BLOC PARAMETRE DOUBLE ENTIER A 8 VOIES

Fonction du bloc

Chacun des dix types de blocs de la série DCM_*8, exécuté dans un T2900/T800/T940, fournit une vue d'un ensemble de paramètres ou d'un paramètre unique dans un instrument déporté. La table 15-24 donne la liste des types de paramètres auxquels chaque bloc paramètre peut accéder.




Bloc paramètres	Accès au paramètre
DCM_I8	Entier avec signe (16 bits)
DCM_S8	Entier avec signe (8 bits)
DCM_UI8	Entier sans signe (16 bits)
DCM_US8	Entier sans signe (8 bits)
DCM_R8	Nombre réel (32 bits)
DCM_T8	Valeur de durée (32 bits)
DCM_W8	Mot hexadécimal ABCD (16 bits)
DCM_Y8	Octet hexadécimal AB (8 bits)
DCM_B8	Booléen
DCM_D8	Entier à double précision (32 bits)

Table 15-24 Types de paramètres auxquels accèdent les blocs paramètre

L'ensemble des dix blocs paramètre fonctionnent de la même manière et ont le même ensemble de champs.

Paramètres des blocs

Les symboles de la table 15-25 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications des blocs* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[5]	Définit port comm. et n° protocole	Menu	
Profile ^[5]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[5]	N° instrument (défaut 0=hors ligne)	Entier	
P_1 à P_8	Valeurs paramètres 1 à 8 (spécifié par profil)	[† Voir nota]	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	

^[5] Données ‘Statiques’, uniquement lues en cas de réinitialisation du maître ou de l’esclave. Les autres données sont lues/écrites à chaque cycle de communication.
† Les unités dépendent du type de bloc paramètre. Voir table 15-24.

Table 15-25 Paramètres des blocs DCM_*8

Menu de spécifications des blocs

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-25:

Profile. Le nom de base du fichier de définition de profil. En général, les blocs DCM peuvent avoir un profil optionnel, qui peut être utilisé pour remplacer la projection par défaut des adresses du bloc. Mais, aucun bloc de la série DCM_*8 ne dispose de projections de paramètres par défaut, et il **faut** donc saisir les noms des fichiers de définition de profil dans leur champs *Profile*.

Voir les détails à la section *Définition du profil*, page 15-2 for more information.

P_1 à P_8. Les valeurs des paramètres 1 à 8, spécifiées par la projection d’adresses dans le fichier de définition de profil désigné dans le paramètre *Profile* . Les unités de *P_1* à *P_8* dépendent du type de bloc utilisé — voir table 15-24.

D2500: BLOC INSTRUMENT 2500

Fonction du bloc

Le bloc D2500 exécuté dans un T2900 fournit une vue globale d'un instrument 2500 déporté (système E/S).

Paramètre du bloc

Les symboles de la table 15-26 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.




















Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Version ^[S]	Version micrologiciel	ABCD hex	 
IO_Sites ^{[S]*}	Nbre de sites pour modules E/S (1 à 16)	Entier	
Mode	Mode instrument (OPERATE/STANDBY/CONFIG)	Menu	
StStdBy ^[S]	VRAI lance 2500 en mode veille	V/F	
StIOFail ^[S]	Passe en mode veille si défaut E/S démarrage	Menu	
WdgION ^[S]	Pér. chien garde rés.E/S, ms. (0=désactivé)	Entier	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	VRAI si erreur comm. vers 2500 (ex. tempo)	V/F	
Config	VRAI si erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
IO_Stat	Alarme mot état E/S (pas voie non utilisée)	V/F	
Ins_Stat	Alarme mot état 2500	V/F	
Mode	Instrument 2500 pas en mode OPERATE	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
IO_Stat	Mot état E/S	(AB)CD hex	
Snsr_Brk	Rupture capteur (combin. octets état voie)	V/F	
CJC_Fail	Défaut compensation soudure froide	V/F	
Not_Used	Voie non utilisée	V/F	
AO_Sat	Sortie analogique saturée	V/F	
Init	Initialisation — valeur non mise à jour	V/F	
Inv_Cal	Etalonnage entrée analogique incorrecte	V/F	
Réserve		V/F	
Mod_Fail	Défaut module (combin. octets état module E/S)	V/F	

Table 15-26 suite...

...Table 15-26 suite

Paramètre	Fonction	Unités	Etat
Ins_Stat	Mot état instrument 2500	(AB)CD hex	
Config	En mode configuration	V/F	D
Overrun		V/F	
IO_Orun		V/F	
Cksum	Défaut total de contrôle NVRAM	V/F	
BadLin	Mauvaise linéarisation personnalisée	V/F	C
BadSize	Taille base erronée	V/F	
OutConfig	Eté en configuration	V/F	
Restart	Schéma relancé	V/F	
OR_Cnt	Nbre dépassements depuis mise sous tension	Entier	
IOOR_Cnt	Nbre dépassements E/S mise sous tension	Entier	
Cfg_Cnt ^{[S]*}	Nbre fois 2500 a quitté mode CONFIG	Entier	
St_Cnt ^{[S]*}	Nbre fois 2500 a été relancé	Entier	

^[S] Données ‘Statiques’, uniquement lues en cas de réinitialisation du maître ou de l’esclave. Les autres données sont lues/écrites à chaque cycle de communication.

^{*}Non mis en oeuvre.

Table 15-26 Paramètres du bloc D2500

Menu de spécifications du bloc

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la table 15-26.

StIOFail. (None/Sby_IO/SbyExit) Instruction pour repasser en mode veille en cas de défaut E/S au démarrage.

D2000: BLOC INSTRUMENT SÉRIE 2000

Fonction du bloc

Le bloc D2000 exécuté dans un T2900 fournit une vue globale d'un instrument déporté de la série 2000 (2400/2200).

Paramètres du bloc

Les symboles de la table 15-27 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.




















Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Inst_Id ^[S]	Type instr. série 2000 (0=pas série 2000)	ABCD hex	 
InstMode	Mode instrument (Normal/Standby/Config)	Menu	 
PV	Variable procédé	Eng	 
SP	Consigne	Eng	 
OP	Sortie	Eng	 
Mode	Mode bloc (AUTO/MANUEL)	Menu	 
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. temporisation)	V/F	
Config	Erreur config. comm. ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
Alarm_1-4	Alarme configurée par l'utilisateur (1 à 4)	V/F	
Snsr_Brk	Rupture capteur	V/F	
Loop_Brk	Rupture boucle	V/F	
Htr_Fail	Défaut réchauffeur	V/F	
PV_OOR	PV hors échelle	V/F	
Rmt_Brk	Rupture déporté	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 15-27 suite...

...Table 15-27 suite

Paramètre	Fonction	Unités	Etat
OpStatus	Mot état sortie sommaire	ABCD hex	 
Alarm_1	Alarme configurée par l'utilisateur	V/F —1	D
Alarm_2		V/F —2	
Alarm_3		V/F —4	
Alarm_4		V/F —8	
Man_Mode	Mode. VRAI = Manuel, FAUX = Auto	V/F —1	C
Snsr_Brk	Rupture capteur	V/F —2	
Loop_Brk	Rupture boucle	V/F —4	
Htr_Fail	Défaut réchauffeur	V/F —8	
Tune_Act	(=LoadFail dans un 2200)	V/F —1	B
RampComp		V/F —2	
PV_OOR	PV hors échelle	V/F —4	
DC_Fault	(=SSR_Fail dans un2200)	V/F —8	
Prog_Run	(=New_Alm dans un 2200)	V/F —1	A
RmSnsBrk	Rupture capteur déporté	V/F —2	
IP1Fault	Défaut entrée 1	V/F —4	
Réservé		V/F —8	
CtStatus	Mot état contrôle	ABCD hex	 
Freeze	Gel contrôle	V/F —1	D
PV_Brokn	Rupture capteur PV	V/F —2	
PV_OOR	PV hors échelle	V/F —4	
TuneFail		V/F —8	
Servo	Signal asservi PID	V/F —1	C
Debump	Signal anti-à-coups PID	V/F —2	
Loop_Brk	Rupture boucle	V/F —4	
AccFreez		V/F —8	
Tune_Ok		V/F —1	B
Dir_Rev	Direct (VRAI)/Inverse (FAUX)	V/F —2	
Init	Initialisation — valeur non mise à jour	V/F —4	
Dem_Lim		V/F —8	
AutoTune	Mise au point auto active (<i>pas dans 2200</i>)	V/F —1	A
AdapTune	Mise au point adaptative activée	V/F —2	
AutDroop	Compensation auto. affaïssement activée	V/F —4	
Man_Mode	Mode. VRAI = Manuel, FAUX = Auto	V/F —8	

^[5] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-27 Paramètres du bloc D2000

Menu de spécifications du bloc

Reportez-vous à la section *D25_R_CV Bloc valeurs calculées réelles à liaisons filaires utilisateur* à la page 15-36, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur les paramètres portant le même nom, également utilisés par ce bloc.

D25_AI: BLOC ENTRÉE ANALOGIQUE À VOIE UNIQUE

Fonction du bloc

Le bloc D25_AI exécuté dans un T800 ou T940 permet de communiquer avec une voie d'entrée analogique unique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur les états déportés et génèrent des alarmes d'état locales. Une PV analogique entrée dans le module déporté apparaît comme une sortie rattachable dans le bloc local.

Paramètres du bloc

Les symboles de la table 15-28 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.














Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV ^[S]	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplac. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie module E/S (1 - 4)	Entier	
Method	Emplac. prog. mise à jour ST bloc (défaut=modèle)	Menu	
Mode	Mode (défaut = Auto)	Menu	
PV	Val. PV entrée, lecture seule mode auto (sortie bloc)	Eng	
HR	Echelle graphique haute (défaut=100.0)	Eng	
LR	Echelle graphique basse (défaut = 0.0)	Eng	
AI	Valeur entrée instal. réel. Champ Val iTools	Eng	 
HiHi	Niveau alarme absolue <i>HiHi</i> (défaut = 100.0)	Eng	
Hi	Niveau alarme absolue <i>Hi</i> (défaut = 100.0)	Eng	
Lo	Niveau alarme absolue <i>Lo</i> (défaut = 0.0)	Eng	
LoLo	Niveau alarme absolue <i>LoLo</i> (défaut = 0.0)	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. (ex. noeud) ou profil erroné	V/F	
Reset	<i>(Non mis en oeuvre)</i>	V/F	
OpenCct	Cct ouv. entrée bloc installations (suit ChanStat.Snsr_Brk)	V/F	
CJC_Fail	Déf. comp. soudure froide (suit ChanStat.CJC_Fail)	V/F	
Not_Used	Voie non utilisée (suit ChanStat.Not_Used)	V/F	
Override	Mode pas Auto	V/F	
Hardware	Déf. module E/S (ChanStat.Mod_Fail ChanStat.Inv_Cal)	V/F	
OutRange	Déclenché si PV 5% hors HR/LR (0.5% bande hyst.)	V/F	
HiHi	Déclenché si PV > HiHi (hyst.=0.5% pleine échelle)	V/F	
Hi	Déclenché si PV > Hi (hyst.=voir ci-dessous)	V/F	
Lo	Déclenché si PV < Lo (hyst.=voir ci-dessous)	V/F	
LoLo	Déclenché si PV < LoLo (hyst.=0.5% pleine échelle)	V/F	

Table 15-28 suite...

...Table 15-28 suite

Paramètre	Fonction	Unités	Etat
EvalFail	Défaut évaluation logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme module E/S. Champ AlmSW iTools	ABCD hex	
Mod_Stat	Mot état alarme module E/S. Champ ModSta iTools	(ABC)D hex	
No_Mod	Module manquant	V/F	1
Bad_ID	ID module erronée	V/F	2
Unk_ID	ID module non reconnue	V/F	4
Comm_Err	Erreur comm. module	V/F	8
ChanStat	Mot état voie entrée. Champ ChStat iTools	(AB)CD hex	
Snsr_Brk	Rupt. capteur détect./PV hors table linéar.	V/F	1
CJC_Fail	Défaut compensation soudure froide	V/F	2
Not_Used	Voie non utilisée	V/F	4
Réserve		V/F	8
Init	Initialisation — valeur non mise à jour	V/F	1
Inv_Cal	Etalonnage entrée analogique incorrect	V/F	2
Réserve		V/F	4
Mod_Fail	Défaut module (combin. bits Mod_Stat)	V/F	8
Failed	Combinaison certains bits d'alarme	V/F	

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-28 Paramètres du bloc D25_AI

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 15-28.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFIDP_1/PROFIDP_2) Définition du port de communication et du numéro du protocole. Ceci permet d'utiliser plusieurs ports sur le même protocole. La valeur par défaut est 'MODBUS_1'. (Tous les éléments du menu ne sont pas gérés à l'heure actuelle).

Profile. Le nom de base d'un fichier de définition de profil. Tout bloc DCM peut avoir un profil optionnel qui peut être utilisé pour remplacer la correspondance d'adresse par défaut du bloc. Les profils sont définis dans un fichier au format CSV et son contenu dépend du protocole utilisé. Le fichier de définition de profil (extension .UYM pour Modbus) doit résider sur le lecteur E: de l'instrument cible.

ST_REV. N° de révision statique de la fondation Fieldbus. Chaque fois que *ST_REV* change ou est égal à 0, toutes les données statiques dans le module de contrôle dévolu — marquées d'un [S] dans la table — sont relues.

Instr_No. N° de l'instrument, en fonction du protocole. Pour les communications Modbus, *Instr_No* se situe dans la plage de 1 à 247. Pour les communications Profibus-DP, la plage est de 1 à 125. La valeur par défaut est de 0, ce qui signifie "hors ligne". *Instr_No* identifie l'adresse esclave (ex. 2500). (Notez que pour Profibus dans le T940, les valeurs de 1 et 2 ne sont pas autorisées).

Slot_No. N° de l'emplacement du module dans l'instrument, c'est à dire le numéro d'instance du module de contrôle dévolu dans la plage de 1 à 16.

Chan_No. N° de voie dans le module, utilisée pour identifier une voie E/S dans un module multi-voies, dans la plage de 1 à 4.

Method. (Template/Block/Native) Indique l'emplacement du sous-programme de mise à jour ST du bloc. Par défaut, la méthode est intégrée dans le modèle, mais le sous-programme de mise à jour du modèle peut être remplacé par une méthode différente qui se trouve dans les ressources communes du bloc. (*A l'heure actuelle, seule l'option 'Template' peut être sélectionnée pour ce champ*)

Mode. (Auto/Manual/Local) Mode de fonctionnement actif.

- **Auto.** Il s'agit du mode de conduite par défaut. *PV* est égal à la valeur des installations *AI* obtenue du 2500.
- **Manual.** Permet de forcer *PV* à prendre une valeur spécifique en conduite, quelle que soit la lecture réelle du 2500 (*PV* déconnectée de *AI*).
- **Local.** Le mode local permet d'effectuer des tests et simulations, lorsqu'un 2500 n'est pas installé. Le bloc ne peut être mis à jour depuis le 2500, mais exécute du code supplémentaire pour simuler les fonctions du 2500, par ex. limiter les alarmes. L'alarme de communication du 2500 est supprimée. (*PV*, limites des alarmes, échelles, *AlarmSW*, *Mod_Stat* et *ChanStat* déconnectés du 2500).

HR, LR. Echelles graphiques haute et basse projetées sur les champs *VALH* et *VALL* de iTools. En lecture seule en mode Auto et Manuel.

AI. Valeur d'entrée réelle des installations, projetée sur le champ *Val* de iTools. *PV* est calculée à partir de *AI* en mode Auto. En lecture seule en mode Auto et Manuel (mais, pas utilisé en mode Local).

HiHi, Hi, Lo, LoLo. Niveaux d'alarme absolue définissant à quel moment les alarmes correspondantes sont déclenchées. *HiHi* et *LoLo* sont soumis à une hystérésis de 0.5 % de la pleine échelle dans tous les modes. *Hi* et *Lo* sont soumis à une valeur d'hystérésis 0.5 % de la pleine échelle en mode Manual et Local, mais en mode Auto la valeur de l'hystérésis est paramétrée dans le 2500 (par iTools). *Hi* et *Lo* sont projetés sur les champs *ALSP_n* de iTools, où $n = 1$ et 3 pour la voie 1, $n = 2$ et 4 pour la voie 2, $n = 5$ et 7 pour la voie 3, etc.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, Bloc ANIN.

Ces informations complètent celles de la table 15-28:

- **HiHi, LoLo.** Déclenchés si $PV > HiHi$ et $PV < LoLo$. Les alarmes ne disparaissent pas tant que *PV* n'est pas revenue à ces niveaux dans les limites de la valeur d'hystérésis — 0.5 % de la pleine échelle dans tous les modes.
- **Hi, Lo.** Déclenchés si $PV > Hi$ et $PV < Lo$. Les alarmes ne disparaissent pas tant que *PV* n'est pas revenue à ces niveaux dans les limites de la valeur d'hystérésis. En mode Auto, l'hystérésis est spécifiée dans le 2500 (par iTools) et les alarmes sont signalées par le champ *AlarmSW* de iTools. En mode Manual et Local, les alarmes sont générées localement, l'hystérésis étant fixée à 0.5 % de la pleine échelle.

ChanStat. Champ binaire projeté sur le champ *ChStat* de iTools, et indiquant l'état de la voie d'entrée. Les informations ci-dessous complètent celle de la table 15-28.

- **Snsr_Brk.** Rupture capteur détectée ou *PV* hors table de linéarisation. L'alarme *OpenCct* est déclenchée, si ce bit est VRAI.

- **CJC_Fail.** Défaut compensation soudure froide. L'alarme *CJC_Fail* est déclenchée, si ce bit est VRAI.
- **Not_Used.** Voie E/S non utilisée. L'alarme *Not_Used* est déclenchée, si ce bit est VRAI.
- **Inv_Cal.** Etalonnage de l'entrée analogique incorrecte. L'alarme *Hardware* est déclenchée, si ce bit est VRAI.
- **Mod_Fail.** Défaut module E/S (combinaison de tous les bits *Mod_Stat*). L'alarme *Hardware* est déclenchée, si ce bit est VRAI.

Failed. VRAI si l'une des alarmes suivantes est déclenchée: *Software*, *Comms*, *Config*, *OpenCct*, *Not_Used*, *Hardware*, ou *EvalFail*. Notez que pour être prise en compte la priorité d'une alarme doit avoir une valeur *différente de zéro*. *Failed* définit les conditions standard pour mettre une boucle PID en mode manuel forcé.

Notes de mise en oeuvre

Les blocs LIN D25_AI personnalisés sont inclus dans la base de données de configuration comme modèle étranger. La taille de la base de données d'exécution occupe 1660 octets supplémentaires par modèle et 192 octets pour chaque instance supplémentaire d'un bloc LIN personnalisé.

D25_AO: BLOC SORTIE ANALOGIQUE A VOIE UNIQUE

Fonction du bloc

Le bloc D25_AO exécuté dans un T800 ou T940 permet de communiquer avec une voie de sortie analogique unique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur la valeur de sortie déportée et son état et génèrent des alarmes d'état locales. Une sortie analogique générée localement et rattachée au bloc apparaît comme une sortie au niveau du module 2500 déporté.

Paramètres du bloc

Les symboles de la table 15-29 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.















Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV ^[S]	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie module E/S (défaut=0, plage 1-4)	Entier	
Method	Emplace. prog. mise à jour ST bloc (défaut=modèle)	Menu	
Mode	Mode (défaut = Auto)	Menu	
OP	Valeur de sortie (entrée dans bloc)	Eng	
HR_OP	Echelle graphique haute (défaut=100.0)	Eng	
LR_OP	Echelle graphique basse (default = 0.0)	Eng	
AO	Sortie réelle installation. Champ Val iTools	Eng	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. (ex. noeud) ou profil erroné	V/F	
Reset	(<i>Non mis en oeuvre</i>)	V/F	
OpenCct	Cct ouv./court sortie bloc installations (suit ChanStat.Snsr_Brk)	V/F	
OP_Sat	Sortie saturée (suit ChanStat.OP_Sat)	V/F	
Not_Used	Voie non utilisée (suit ChanStat.Not_Used)	V/F	
Override	Mode pas Auto	V/F	
Hardware	Déf. module E/S (ChanStat.Mod_Fail ChanStat.Inv_Cal)	V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
EvalFail	Défaut évaluation logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 15-29 suite...

...Table 15-29 suite

Paramètre	Fonction	Unités	Etat
Mod_Stat	Mot état alarme module E/S. Champ <i>ModSta</i> iTools	(ABC)D hex	 
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur comm. module	V/F — 8	
ChanStat	Mot état voie sortie. Champ <i>ChStat</i> iTools	(AB)CD hex	 
Snsr_Brk	Rupt. capteur détect./PV hors table linéar.	V/F — 1	D
<i>Réserve</i>		V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
OP_Sat	Sortie saturée	V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
Inv_Cal	Etalonnage sortie analogique incorrect	V/F — 2	
<i>Réserve</i>		V/F — 4	
Mod_Fail	Défaut module (combin. bits Mod_Stat)	V/F — 8	
Failed	Combinaison certains bits d'alarme	V/F	 

^[5] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-29 Paramètres du bloc D25_AO

Menu de spécifications du bloc

Reportez à la section *D25_AI Bloc entrée analogique à voie unique* aux pages 15-46, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la Table 15-29 :

Mode. (Auto/Manual/Local) Mode de fonctionnement actif.

- **Auto.** Il s'agit du mode de conduite par défaut. *AO* (sortie réelle installation/module 2500) est obtenue de *OP*.
- **Manual.** Permet de forcer *AO* à prendre une valeur spécifique en conduite, quelle que soit la valeur réelle de *OP* (*AO* déconnecté de *OP*).
- **Local.** Le mode local permet d'effectuer des tests et simulations, lorsqu'un 2500 n'est pas installé. Le bloc ne met pas à jour le 2500, et exécute du code supplémentaire pour simuler les fonctions du 2500. (*OP* et *AO* déconnectés du 2500). L'alarme de communication du 2500 est supprimée.

HR_OP, LR_OP. Echelles graphiques haute et basse projetées sur les champs *VALH* et *VALL* respectivement de iTools. En lecture seule en mode Auto et Manuel.

AO. Valeur de sortie réelle des installations, projetée sur le champ *Val* de iTools. En lecture seule, *AO* est calculé à partir de *OP* en mode Auto.

Failed. VRAI si l'un des bits des champs *Alarms* suivants est VRAI : *Software*, *Comms*, *Config*, *Not_Used*, *CctFault*, *Hardware*, et *EvalFail*.

Notes de mise en oeuvre

Les blocs LIN D25_AO personnalisés sont inclus dans la base de données de configuration comme modèle étranger. La taille de la base de données d'exécution occupe 1040 octets supplémentaires par modèle et 174 octets pour chaque instance supplémentaire d'un bloc LIN personnalisé.

D25_DI: BLOC ENTREE LOGIQUE A VOIE UNIQUE

Fonction du bloc

Le bloc D25_DI exécuté dans un T800 ou T940 permet de communiquer avec une voie d'entrée analogique unique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur les états déportés et génèrent des alarmes d'état locales. Une entrée logique dans le module 2500 déporté apparaît comme une sortie rattachable dans le bloc local.







Paramètres du bloc

Les symboles de la table 15-30 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV ^[S]	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie module E/S (défaut=0, plage 1-4)	Entier	
Method	Emplace. prog. mise à jour ST bloc (défaut=modèle)	Menu	
Mode	Mode (défaut = Auto)	Menu	
In	Etat entrée, à partir de <i>Val</i> en mode Auto (sortie bloc)	V/F	
NotIn	Complément état entrée, c.à.d. NOT <i>In</i>	V/F	
Val	Etat entrée de l'installation. Champ <i>Val</i> iTools	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. (ex. noeud) ou profil erroné	V/F	
Reset	(<i>Non mis en oeuvre</i>)	V/F	
<i>Réserve</i>		V/F	
<i>Réserve</i>		V/F	
Not_Used	Voie non utilisée (suit ChanStat.Not_Used)	V/F	
Override	Mode pas Auto	V/F	
Hardware	Déf. module E/S (ChanStat.Mod_Fail)	V/F	
<i>Réserve</i>		V/F	
Input	Calculé à partir de <i>AlarmSW</i>	V/F	
<i>Réserve</i>		V/F	
<i>Réserve</i>		V/F	
<i>Réserve</i>		V/F	
EvalFail	Défaut évaluation logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme module E/S. Champ <i>AlmSW</i> iTools	ABCD hex	

Table 15-30 suite...

...Table 15-30 suite

Paramètre	Fonction	Unités	Etat
Mod_Stat	Mot état alarme module E/S. Champ <i>ModSta</i> iTools (ABC)D hex		 
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur comm. module	V/F — 8	
ChanStat	Mot état voie entrée. Champ <i>ChStat</i> iTools	(AB)CD hex	 
Snsr_Brk	Rupt. capteur détect./PV hors table linéar.	V/F — 1	D
CJC_Fail	Défaut compensation soudure froide	V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
<i>Réserve</i>		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
Inv_Cal	Etalonnage entrée analogique incorrect	V/F — 2	
<i>Réserve</i>		V/F — 4	
Mod_Fail	Défaut module (combin. bits Mod_Stat)	V/F — 8	
Failed	Combinaison certains bits d'alarme	V/F	 

^[S] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-30 Paramètres du bloc D25_DI

Menu de spécifications du bloc

Reportez à la section *D25_AI Bloc entrée analogique à voie unique* aux pages 15-46, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom également utilisés par ce bloc.

Les informations suivantes complètent celles de la Table 15-30 :

In. Etat entrée (sortie du bloc). En lecture seule en mode Auto.

Val. Etat entrée depuis l'installation, après inversion dans le 2500, à partir duquel *In* est calculé — en mode Auto uniquement. Projeté sur le champ *Val* de iTools.

Mode. (Auto/Manual/Local) Mode de fonctionnement actif.

- **Auto.** Il s'agit du mode de conduite par défaut. *AO* (sortie réelle installation/module 2500) est obtenue de *OP*.
- **Manual.** Permet de forcer *AO* à prendre une valeur spécifique en conduite, quelle que soit la valeur réelle de *OP* (*AO* déconnectée de *OP*).
- **Local.** Le mode local permet d'effectuer des tests et simulations, lorsqu'un 2500 n'est pas installé. Le bloc ne met pas à jour le 2500, et exécute du code supplémentaire pour simuler les fonctions du 2500. (*OP* et *AO* déconnectés du 2500). L'alarme de communication du 2500 est supprimée.

Failed. VRAI si l'un des bits des champs *Alarms* suivants est VRAI : *Software*, *Comms*, *Config*, *Not_Used*, *Hardware*, et *EvalFail*.

Notes de mise en oeuvre

Les blocs LIN D25_DI personnalisés sont inclus dans la base de données de configuration comme modèle étranger. La taille de la base de données d'exécution occupe 1114 octets supplémentaires par modèle et 162 octets pour chaque instance supplémentaire d'un bloc LIN personnalisé.

D25_DO: BLOC SORTIE LOGIQUE A VOIE UNIQUE

Fonction du bloc

Le bloc D25_DO exécuté dans un T800 ou T940 permet de communiquer avec une voie de sortie logique unique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont projetés sur les états déportés et génèrent des alarmes d'état locales. Un signal logique rattaché au bloc local apparaît comme une sortie au niveau du module déporté.

Paramètres du bloc

Les symboles de la table 15-31 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.











Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit port comm. et n° protocole	Menu	
Profile ^[S]	Nom base fichier CSV profil adresse-projection	Alphanumérique	
ST_REV ^[S]	N° révision statique fondation Fieldbus	Entier	
Instr_No ^[S]	N° instrument (défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° emplace. module dans instrument (1-16)	Entier	
Chan_No ^[S]	N° voie module E/S (défaut=0, page 1-4)	Entier	
Method	Emplace. prog. mise à jour ST bloc (défaut=modèle)	Menu	
Mode	Mode (défaut = Auto)	Menu	
Out	Etat sortie (entrée dans bloc)	V/F	
Val	Etat sortie à l'installation. Champ iTools <i>Val</i>	V/F	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Erreur comm. vers 2500 (ex. tempo)	V/F	
Config	Erreur config. comm. (ex. noeud) ou profil erroné	V/F	
Reset	(<i>Non mis en oeuvre</i>)	V/F	
Réserve		V/F	
Réserve		V/F	
Not_Used	Voie non utilisée (suit ChanStat.Not_Used)	V/F	
Override	Mode pas Auto	V/F	
Hardware	Déf. module E/S (ChanStat.Mod_Fail)	V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
Réserve		V/F	
EvalFail	Défaut évaluation logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlarmSW	Mot état alarme module E/S. Champ <i>AlmSW</i> iTools	ABCD hex	

Table 15-31 suite...

...Table 15-31 suite

Paramètre	Fonction	Unités	Etat
Mod_Stat	Mot état alarme module E/S. Champ <i>ModSta</i> iTools	(ABC)D hex	
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur comm. module	V/F — 8	
ChanStat	Mot état voie sortie. Champ <i>ChStat</i> iTools	(AB)CD hex	
<i>Réserve</i>		V/F — 1	D
<i>Réserve</i>		V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
<i>Réserve</i>		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
<i>Réserve</i>		V/F — 2	
<i>Réserve</i>		V/F — 4	
Mod_Fail	Défaut module (combin. bits Mod_Stat)	V/F — 8	
Failed	Combinaison certains bits d'alarme	V/F	

^[5] Données 'Statiques', uniquement lues en cas de réinitialisation du maître ou de l'esclave. Les autres données sont lues/écrites à chaque cycle de communication.

Table 15-31 Paramètres du bloc D25_DO

Menu de spécifications du bloc

Reportez à la section *D25_AI Bloc entrée analogique à voie unique* à la page 15-46, sous *Menu de spécifications du bloc* pour obtenir de plus amples informations sur des paramètres portant le même nom, également utilisés par ce bloc.

Les informations suivantes complètent celles de la Table 15-31 :

Val. Etat sortie à l'installation, avant inversion dans le 2500. Calculé à partir de *Out* en mode Auto — lorsque *Val* est en lecture seule. Projeté sur le champ *Val* de iTools.

Mode. (Auto/Manual/Local) Mode de fonctionnement actif.

- **Auto.** Il s'agit du mode de conduite par défaut. *Val* (sortie réelle à l'installation/module 2500) est obtenue de *Out*.
- **Manual.** Permet de forcer *Val* à prendre une valeur spécifique en conduite, quelle que soit la valeur réelle de *Out* (*Val* déconnectée de *Out*).
- **Local.** Le mode local permet d'effectuer des tests et simulations, lorsqu'un 2500 n'est pas installé. Le bloc ne met pas à jour le 2500, et exécute du code supplémentaire pour simuler les fonctions du 2500. (*Out* et *Val* déconnecté du 2500). L'alarme de communication du 2500 est supprimée.

Failed. VRAI si l'un des bits des champs *Alarms* suivants est VRAI : *Software*, *Comms*, *Config*, *Not_Used*, *Hardware*, et *EvalFail*.

Notes de mise en oeuvre

Les blocs LIN D25_DO personnalisés sont inclus dans la base de données de configuration comme modèle étranger. La taille de la base de données d'exécution occupe 860 octets supplémentaires par modèle et 160 octets pour chaque instance supplémentaire d'un bloc LIN personnalisé.

D25_AI_T: BLOCS D'ALARMES TEMPORISEES D'UNE ENTREE ANALOGIQUE A UNE SEULE VOIE

Fonction du bloc

Le bloc D25_AI_T exécuté sur un T800 ou T940 permet de communiquer avec une entrée analogique à une seule voie fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont appliqués aux états déportés et génèrent des alarmes d'état locales. Une PV analogique appliquée à l'entrée du module déporté est présentée comme une sortie filaire dans le bloc local.

Le bloc est similaire au bloc D25_AI, mais comprend des temporisations configurables pour le déclenchement et la suppression de toutes les huit alarmes liées au procédé — *OpenCct*, *OutRange*, *HiHi*, *Hi*, *HiDev*, *LoDev*, *Lo*, et *LoLo*. Les alarmes temporisées permettent d'ignorer des perturbations transitoires dans le procédé supervisé/contrôlé, ce qui permet d'éviter les alarmes intempestives.

NOTA. Les alarmes *CJC_Fail* et *Not_Used* du bloc D25_AI ne sont pas disponibles comme alarmes séparées dans le bloc D25_AI_T. Dans les deux cas, c'est l'alarme *Hardware* (matériel) qui est déclenchée.

Paramètres du bloc

Les symboles utilisés dans la table 15-32 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.




















Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit le port comm et le n° de protocole	Menu	
Profile ^[S]	Nom de base fichier CSV définissant profil adresse-corresp.	Alphanumérique	
ST_REV ^[S]	N° de révision statique Fieldbus Foundation	Entier	
Instr_No ^[S]	N° d'instrument (par défaut 0=hors ligne)	Entier	
Slot_No ^[S]	N° alvéole du module dans l'instrument (1-16)	Entier	
Chan_No ^[S]	N° voie du module E/S (1 - 4)	Entier	
Method	Emplacement routine m2j ST du bloc (défaut = modèle)	Menu	
Mode	Mode (par défaut = Auto)	Menu	
PV	Valeur entrée PV, lecture seule mode Auto (sortie bloc)	Eng	
HR ^[S]	Echelle graphique haute (défaut=100.0)	Eng	
LR ^[S]	Echelle graphique basse (par défaut=0.0)	Eng	
AI	Valeur entrée réelle installations. Appliquée à iTools Val	Eng	 
SetPoint ^[S]	Valeur consigne	Eng	
Hyst ^[S]	Bande hystérésis pour supprimer alarmes abs & écart	Eng	
HiHi ^[S]	Niveau alarme absolue HiHi (défaut=100.0)	Eng	
Hi ^[S]	Niveau alarme absolue Hi (défaut=100.0)	Eng	
HiDev ^[S]	Niveau alarme écart HiDev (défaut=100.0)	Eng	
LoDev ^[S]	Niveau alarme écart LoDev (défaut = 100.0)	Eng	
Lo ^[S]	Niveau alarme absolue (défaut = 0.0)	Eng	
LoLo ^[S]	Niveau alarme absolue LoLo (défaut = 0.0)	Eng	

Table 15-32 suite...

...Table 15-32 suite

Paramètre	Fonction	Unités	Etat
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Comms	Toute erreur comm vers 2500 (ex.: temporisation)	V/F	
Config	Erreur config. comm. (ex.: noeud erroné) ou profil erroné	V/F	
Reset	(Non mis en oeuvre)	V/F	
OpenCct*	Circuit ouvert entrée bloc du champ (suit ChanStat.Snsr_Brk)	V/F	
Override	Mode non Auto	V/F	
Hardware	Défaut module E/S (suit certains champs ChanStat)	V/F	
OutOfRange*	Déclenchée si PV 5 % hors HR/LR (0,5 % bande hyst.)	V/F	
HiHi*	Déclenchée si $PV > HiHi$ (suppr. si $PV + Hyst < HiHi$)	V/F	
Hi*	Déclenchée si $PV > Hi$ (supprimée si $PV + Hyst < Hi$)	V/F	
HiDev*	Déclenchée si $PV - SP > HiDev$ (suppr. si $PV - SP + Hyst < HiDev$)	V/F	
LoDev*	Déclenchée si $SP - PV > LoDev$ (suppr. si $SP - PV + Hyst < LoDev$)	V/F	
Lo*	Déclenchée si $PV < Lo$ (supprimée si $PV > Lo + Hyst$)	V/F	
LoLo*	Déclenchée si $PV < LoLo$ (suppr. si $PV > LoLo + Hyst$)	V/F	
EvalFail	Défaut éval. conduite dans logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Mod_Stat	Mot d'état module E/S. Appliqué au champ iTools ModSta	(ABC)D hex	
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur comm. module	V/F — 8	
ChanStat	Mot d'état voie d'entrée. Appliqué au champ iTools ChStat	(AB)CD hex	
Snsr_Brk	Rupture capteur détectée ou PV hors table linéarisation	V/F — 1	D
CJC_Fail	Défaut compensation soudure froide	V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
Réserve		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
Inv_Cal	Etalonnage entrée analogique erroné	V/F — 2	
Réserve		V/F — 4	
Mod_Fail	Défaut module (combinaison tous les bits Mod_Stat)	V/F — 8	
Deviatn	Valeur d'écart, à savoir $PV - SetPoint$	Eng	
DisableA, B	Champs de désact. alarmes (DisableA OU DisableB)	ABCD hex	
OpenCct		V/F — 1	D
OutOfRange		V/F — 2	
HiHi		V/F — 4	
Hi		V/F — 8	
HiDev	Bit DisableA OU DisableB = VRAI désactive l'alarme correspondante	V/F — 1	C
LoDev		V/F — 2	
Lo		V/F — 4	
LoLo		V/F — 8	
All		V/F — 1	B
Réserve		V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	

*Ces bits d'alarme sont soumis aux temporisations AlmOnTim & AlmOffTim

Table 15-32 suite...

Table 15-32 suite

Paramètre	Fonction	Unités	Etat
<i>Réserve</i>		V/F — 1	A
<i>Réserve</i>		V/F — 2	
<i>Réserve</i>		V/F — 4	
Hardware	<i>DisableA</i> OU <i>DisableB</i> = VRAI désactive l'alarme	V/F — 8	
AlmOnTim ^[S]	Temporisation alarme ON (secondes)	Eng	
AlmOffTim ^[S]	Temporisation alarme OFF (secondes)	Eng	
TrueCond	Conditions d'alarme vraies (ignore tempo/désact.)	(AB)CD hex	
OpenCct	} Bit VRAI = condition d'alarme présente	V/F — 1	D
OutRange		V/F — 2	
HiHi		V/F — 4	
Hi		V/F — 8	
HiDev	} Bit VRAI = condition d'alarme présente & activée	V/F — 1	C
LoDev		V/F — 2	
Lo		V/F — 4	
LoLo		V/F — 8	
CurrCond	Conditions d'alarme actuelles (ignore uniq. tempo)	(AB)CD hex	
OpenCct	} Bit VRAI = condition d'alarme présente & activée	V/F — 1	D
OutRange		V/F — 2	
HiHi		V/F — 4	
Hi		V/F — 8	
HiDev	} Bit VRAI = condition d'alarme présente & activée	V/F — 1	C
LoDev		V/F — 2	
Lo		V/F — 4	
LoLo		V/F — 8	
Status	Etat de traitement de l'alarme	(AB)CD hex	
NewAlarm	VRAI = nouvelle alarme déclenchée (après tempo)	V/F — 1	D
StartUp	VRAI pendant 1ère itération base de données, ensuite FAUX	V/F — 2	
Bit2	(<i>Réserve</i>)	V/F — 4	
Bit3	(<i>Réserve</i>)	V/F — 8	
Bit4	(<i>Réserve</i>)	V/F — 1	C
Bit5	(<i>Réserve</i>)	V/F — 2	
Bit6	(<i>Réserve</i>)	V/F — 4	
Bit7	(<i>Réserve</i>)	V/F — 8	
Options	Options de traitement des alarmes	(AB)CD hex	
AckNewAl	Remet à zéro <i>Status.NewAlarm</i>	V/F — 1	D
Bit1	(<i>Réserve</i>)	V/F — 2	
Bit2	(<i>Réserve</i>)	V/F — 4	
InitLast	VRAI remet à 0 tous les temporisateurs(bit remis à 0auto)	V/F — 8	
Bit4	(<i>Réserve</i>)	V/F — 1	C
Bit5	(<i>Réserve</i>)	V/F — 2	
Bit6	(<i>Réserve</i>)	V/F — 4	
Bit7	(<i>Réserve</i>)	V/F — 8	
Failed	Défaut E/S ou de communication	V/F	

^[S] 'Données 'statiques', seulement lues si le maître ou l'esclave sont réinitialisés. Les autres données sont lues/écrites au cours de chaque cycle de communication.

Table 15-32 Paramètres du bloc D25_AI_T

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 15-32.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFDP_1/PROFDP_2)
Définition du port comm et du numéro de protocole. Ceci permet d'utiliser le même protocole sur plusieurs ports. 'MODBUS_1' est la valeur par défaut. *(Toutes les commandes de menu ne sont pas prises en charge à l'heure actuelle).*

Profile. Le nom de base d'un fichier de définition de profil. Tout bloc DCM peut avoir un profil optionnel qui peut se substituer à la mise en correspondance des adresses par défaut du bloc. Un profil est défini dans un fichier au format CSV, et son contenu dépend du protocole auquel il s'applique. Le fichier de définition de profil (avec l'extension .UYM pour Modbus) doit se trouver sur l'unité E: de l'instrument cible.

ST_REV. Numéro de révision statique de la fondation Fieldbus. Chaque fois que *ST_REV* change ou est égal à 0, toutes les données statiques dans le module de contrôle dévolu - marquées d'un [S] dans la table — sont relues.

Instr_No. Numéro d'instrument ou de noeud, suivant le protocole. Pour les communications Modbus, *Instr_No* se situe dans la plage de 1 à 247. Pour les communications Profibus-DP, la plage est de 1 à 125. La valeur par défaut est de 0, ce qui signifie "hors ligne". *Instr_No* identifie l'adresse de l'esclave (par ex. 2500). *(Notez que pour Profibus dans le T940, les valeurs 1 et 2 ne peuvent pas être utilisées).*

Slot_No. N° de l'avéole E/S du module dans l'instrument, c'est à dire le numéro d'instance du module de contrôle dévolu dans la plage de 1 à 16.

Chan_No. N° de voie dans une alvéole permettant d'identifier une voie E/S dans un module multi-voie, dans la plage de 1 à 4.

Method. (Template/Block/Native) Indique l'emplacement du sous-programme de mise à jour de ST du bloc. Par défaut, il est intégré dans le modèle, le sous-programme de mise à jour du modèle peut être remplacé par une méthode différente qui se trouve dans le paquet du bloc. *(A l'heure actuelle, seul 'Template' peut être sélectionné pour ce champ).*

Mode. (Auto/Manual/Local) Mode de fonctionnement utilisé.

- **Auto.** Il s'agit du mode de conduite par défaut. *PV* est égal à la valeur des installations *AI* obtenue du 2500.
- **Manual.** Permet de forcer *PV* à une valeur spécifique en conduite quelle que soit la lecture réelle du 2500 (*PV* déconnecté de *AI*). Toutes les alarmes sont évaluées et aucune n'est automatiquement supprimée.
- **Local.** Le mode local est utilisé pour des tests et simulations, lorsqu'aucun 2500 n'est installé. Le bloc ne peut être mis à jour par le 2500, mais exécute du code supplémentaire pour répliquer les fonctions du 2500, à savoir les alarmes de limites. L'alarme de communication du 2500 est supprimée. (*PV*, les limites des alarmes, les échelles, *AlarmSW*, *Mod_Stat*, et *ChanStat* sont déconnectés du 2500).

HR, LR. Les échelles graphiques hautes et basses appliquées respectivement aux champs *iTools VALH* et *VALL*.. En lecture seule dans les modes Auto et Manual.

NOTA. *HR* et *LR* permettent de limiter *PV*, *SetPoint*, *HiHi*, *Hi*, *HiDev*, *LoDev*, *Lo*, *LoLo*, et *AI*, afin qu'un signal proportionnel puisse être généré pour l'établissement de tendances, le remplissage de bargraphes, etc. Aucune limite n'est appliquée à aucun champ qui limiterait sa valeur.

AI. Valeur d'entrée réelle des installations, appliquée au champ *Val* de iTools. *PV* est calculé en fonction de *AI* en mode Auto. En lecture seule dans les modes Auto et Manual (mais n'est pas utilisé en mode Local).

HiHi, Hi, Lo, LoLo. Niveaux d'alarme haute et basse absolue définissant quand les alarmes correspondantes activées sont déclenchées, soumises à d'éventuelles temporisations définies par *AlmOnTim*. Les alarmes sont supprimées sous réserve de l'hystérésis configurée dans le champ *Hyst* et d'éventuelles temporisations définies par *AlmOfTim*.

NOTA. Ces champs d'alarmes sont calculés localement dans la base de données LIN et ne sont pas transmis au 2500. Ce cas est différent du bloc D25_AI où ils sont appliqués aux champs *ALSP_n* de iTools.

HiDev, LoDev. Niveaux d'alarme d'écart haut et bas définissant quand les alarmes correspondantes activées sont déclenchées, soumises à d'éventuelles temporisations définies par *AlmOnTim*. Les alarmes sont supprimées sous réserve de l'hystérésis configurée dans le champ *Hyst* et d'éventuelles temporisations définies par *AlmOfTim*.

NOTA. Ces champs d'alarmes sont calculés localement dans la base de données LIN et ne sont pas transmis au 2500.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, Bloc ANIN.

Ces informations complètent celles de la table 15-32:

- **Hardware.** Défaut dans le module E/S. — soit *ChanStat.Mod_Fail* OU *ChanStat.Inv_Cal* OU *ChanStat.CJC_Fail* OU *ChanStat.Not_Used* VRAIS. Notez la différence par rapport au bloc D25_AI où *CJC_Fail* et *Not_Used* sont des alarmes séparées.
- **OutOfRange.** Déclenché dès que *PV* atteint une valeur de 5 % au-dessus ou en-dessous de *HR* ou *LR*, (sous réserve d'éventuelles temporisations définies par *AlmOnTim*). Une bande d'hystérésis de 0,5 % intégrée est appliquée asymétriquement à chaque valeur d'échelle pour supprimer l'alarme (sous réserve d'éventuelles temporisations définies par *AlmOfTim*).

DisableA, DisableB. Champs de désactivation (masquage) des alarmes. VRAI sur toute entrée désactive l'alarme correspondante. *DisableA* et *DisableB* fonctionnent en mode 'OU', à savoir que le passage à VRAI de l'un ou l'autre désactive l'alarme. Notez que le champ *TrueCond* est toujours mis à jour en fonction des résultats réels de l'évaluation des alarmes, quel que soit le paramétrage de *DisableA/B*, mais les bits *CurrCond* correspondants sont forcés à l'état de désactivation des alarmes (FAUX).

NOTA. Si une alarme non acquittée est désactivée, elle le reste après avoir été désactivée.

CurrCond. Ce champ binaire montre quelles conditions d'alarme sont définies, après que toutes les désactivations (*DisableA* OU *DisableB*) ont été prises en compte, mais en ignorant toute temporisation configurée (*AlmOnTim*, *AlmOfTim*). L'affectation de bit est la même que pour le champ binaire *TrueCond*.

Chaque fois que le bloc est traité, *CurrCond* est mis à jour avec les états d'alarme actifs basés sur les champs *TrueCond* et *DisableA/B*. Si un bit *CurrCond* a une valeur différente de sa dernière valeur dans *TrueCond*, le temporisateur interne correspondant est chargé avec la valeur *AlmOnTim* (si *CurrCond* est VRAI) ou avec *AlmOfTim* (si *CurrCond* est FAUX). Notez qu'il y a un temporisateur indépendant pour chacune des huit alarmes temporisées - marquées d'un astérisque dans la table 15-32. Lorsqu'un temporisateur atteint zéro, l'état du champ *Alarms* correspondant est mis à 1 par *CurrCond*.

NOTA. Si *CurrCond* change en permanence dans les limites de durée spécifiées pour *AlmOnTim* et *AlmOfTim*, l'état de la ou des alarmes signalées ne change pas.

Status. Champ binaire d'état de traitement des alarmes.

Ces informations complètent celles de la table 15-32:

- **NewAlarm.** Chaque fois qu'une nouvelle alarme est déclenchée (après une temporisation configurée, le cas échéant), ce bit passe à l'état VRAI. Il peut être remis à l'état initial, en sélectionnant VRAI pour *Options.AckNewAl*. Pour une remise automatique à l'état initial après une scrutation de la base de données, reliez *Status.NewAlarm* à *Options.AckNewAl*.

Cette fonctionnalité peut être utilisée dans un plan de gestion des alarmes, par exemple, pour générer un signal d'avertisseur externe, lorsqu'une nouvelle alarme se produit.

- **StartUp.** Ce bit est VRAI au cours de la première itération de la base de données après le démarrage, et repasse ensuite à l'état FAUX. Lorsque *StartUp* est VRAI, tous les temporisateurs du bloc sont mis à zéro. (Cette action a le même effet que de mettre *Options.InitLast* à l'état VRAI).

Failed. Défaut E/S ou de communication. VRAI si l'une des alarmes suivantes est déclenchée: *Software*, *Comms*, *Config*, *OpenCct*, *Hardware*, ou *EvalFail*. Notez que pour être prise en compte, la priorité d'une alarme doit avoir une valeur *différente de zéro*. La sortie *Failed* définit les conditions standard pour mettre une boucle de régulation en mode manuel forcé.

Notes de mise en oeuvre

Chaque bloc D25_AI_T inclus dans la base de données augmente sa taille d'au moins 280 octets (plus 8 octets, si des UC redondantes sont utilisées).

D25_DI_T: BLOCS D’ALARMES TEMPORISEES D’UNE ENTREE LOGIQUE A UNE SEULE VOIE

Fonction du bloc

Le bloc D25_DI_T exécuté dans un T800 ou T940 permet de communiquer avec une seule voie d’entrée logique fonctionnant dans un module 2500 déporté. Les paramètres du bloc sont appliqués sur les états déportés et génèrent des alarmes locales d’état. Une entrée logique sur le module 2500 déporté apparaît comme une sortie filaire dans le bloc local.

Le bloc est similaire au bloc D25_DI, mais comprend une temporisation configurable pour le déclenchement et la suppression du champ *Alarms.Input* calculé localement. Les alarmes temporisées permet d’ignorer des perturbations transitoires du procédé supervisé/contrôlé, ce qui permet d’éviter les alarmes intempestives.

NOTA. L’alarme *Not_Used* du bloc D25_DI n’est pas présent comme alarme séparée dans le bloc D25_DI_T. L’alarme *Not_Used* est intégrée à l’alarme *Hardware*, dans ce cas.
















Paramètres du bloc

Les symboles de la table 15-33 sont expliqués au chapitre 1. Des informations supplémentaires sur ces paramètres sont données dans le *Menu de spécifications de bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Port ^[S]	Définit le port comm et le n° de protocole	Menu	
Profile ^[S]	Nom de base fichier CSV définissant profil adresses-corresp.	Alphanumérique	
ST_REV ^[S]	N° de révision statique Fondation Fieldbus	Entier	
Instr_No ^[S]	N° d’instrument (par défaut 0 = hors ligne)	Entier	
Slot_No ^[S]	N° d’alvéole du module dans l’instrument (1-16)	Entier	
Chan_No ^[S]	N° de voie du module E/S (1 - 4)	Entier	
Method	Emplacement routine māj ST du bloc (par déf. = Template)	Menu	
Mode	Mode (par défaut = Auto)	Menu	
In	Etat d’entrée calculé à partir de Val dans Auto (sortie du bloc)	V/F	
NotIn	Complément état d’entrée, c’est à dire NOT <i>In</i>	V/F	
Val	Etat entrée installations. Appliqué champ Val iTools	V/F	
InactStr ^[S]	Description chaîne appliquée à In si FAUX	Alphanumérique	
ActStr ^[S]	Description chaîne appliquée à In si VRAI	Alphanumérique	
AlmWhen	Sélectionne valeur de In déclenchant alarme d’entrée	Menu	
InStr	Montre <i>InactStr</i> ou <i>ActStr</i> suivant la valeur de In	Alphanumérique	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Comms	Toute erreur de comm. vers 2500 (ex. temporisation)	V/F	
Config	Erreur config. comm. (ex.: noeud erroné) ou profil erroné	V/F	
Reset	(<i>Non mis en oeuvre</i>)	V/F	
Override	Mode pas Auto	V/F	
Hardware	Défaut module E/S (ChanStat.Mod_Fail OU ChanStat.Not_Used)	V/F	

Table 15-33 suite...

...Table 15-33 suite.

Paramètre	Fonction	Unités	Etat
Input	Alarme entrée calculée à partir de <i>In & AlmWhen</i>	V/F	
EvalFail	Déf. évaluation logique interne bloc	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Mod_Stat	Mot état module E/S. Appliqué champ <i>ModSta</i> iTools	(ABC)D hex	 
No_Mod	Module manquant	V/F — 1	D
Bad_ID	ID module erronée	V/F — 2	
Unk_ID	ID module non reconnue	V/F — 4	
Comm_Err	Erreur de communication du module	V/F — 8	
ChanStat	Mot état voie entrée. Appliqué champ <i>ChStat</i> iTools	(AB)CD hex	 
<i>Réserve</i>		V/F — 1	D
<i>Réserve</i>		V/F — 2	
Not_Used	Voie non utilisée	V/F — 4	
<i>Réserve</i>		V/F — 8	
Init	Initialisation — valeur non mise à jour	V/F — 1	C
<i>Réserve</i>		V/F — 2	
<i>Réserve</i>		V/F — 4	
Mod_Fail	Défaut module (combinaison tous bits Mod_Stat)	V/F — 8	
DisableA, B	Champs désactiv. alarmes (<i>DisableA</i> OU <i>DisableB</i>)	V/F	
AlmOnTim ^[5]	Temporisation alarme ON (secondes)	Eng	
AlmOffTim ^[5]	Temporisation alarme OFF (secondes)	Eng	
TrueCond	VRAI si entrée en alarme (tempo/désact. ignorées)	V/F	 
CurrCond	VRAI si entrée en alarme (tempo ignorée seulement)	V/F	 
Status	Etat de traitement des alarmes	(AB)CD hex	 
NewAlarm	VRAI = nouvelle alarme déclenchée (après tempo)	V/F — 1	D
StartUp	VRAI première itération base de données, ensuite FAUX	V/F — 2	
Bit2	(<i>Réserve</i>)	V/F — 4	
Bit3	(<i>Réserve</i>)	V/F — 8	
Bit4	(<i>Réserve</i>)	V/F — 1	C
Bit5	(<i>Réserve</i>)	V/F — 2	
Bit6	(<i>Réserve</i>)	V/F — 4	
Bit7	(<i>Réserve</i>)	V/F — 8	
Options	Options de traitement des alarmes	(AB)CD hex	
AckNewAl	Réinitialise <i>Status.NewAlarm</i>	V/F — 1	D
Bit1	(<i>Réserve</i>)	V/F — 2	
Bit2	(<i>Réserve</i>)	V/F — 4	
InitLast	VRAI met à zéro le temporisateur (raz auto bit)	V/F — 8	
Bit4	(<i>Réserve</i>)	V/F — 1	C
Bit5	(<i>Réserve</i>)	V/F — 2	
Bit6	(<i>Réserve</i>)	V/F — 4	
Bit7	(<i>Réserve</i>)	V/F — 8	
Failed	Défaut E/S ou de communication	V/F	 

^[5] 'Données 'statiques', seulement lues si le maître ou l'esclave sont réinitialisés. Les autres données sont lues/écrites au cours de chaque cycle de communication.

Table 15-33 Paramètres du bloc D25_DI_T

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 15-33.

Port. (MODBUS_1/MODBUS_2/MODBUS_3/MODBUS_4/PROFDP_1/PROFDP_2)

Définition du port comm et du numéro de protocole. Ceci permet d'utiliser le même protocole sur plusieurs ports. 'MODBUS_1' est la valeur par défaut. *(Toutes les commandes de menu ne sont pas prises en charge à l'heure actuelle).*

Profile. Le nom de base d'un fichier de définition de profil. Tout bloc DCM peut avoir un profil optionnel qui peut se substituer à la mise en correspondance des adresses par défaut du bloc. Un profil est défini dans un fichier au format CSV, et son contenu dépend du protocole auquel il s'applique. Le fichier de définition de profil (avec l'extension .UYM pour Modbus) doit se trouver sur l'unité E: de l'instrument cible.

ST_REV. Numéro de révision statique de la fondation Fieldbus. Chaque fois que *ST_REV* change ou est égal à 0, toutes les données statiques dans le module de contrôle dévolu - marquées d'un [S] dans la table — sont relues.

Instr_No. Numéro d'instrument ou de noeud, suivant le protocole. Pour les communications Modbus comms, *Instr_No* se situe dans la plage de 1 à 247. Pour les communications Profibus-DP, la plage est de 1 à 125. La valeur par défaut est de 0, ce qui signifie "hors ligne". *Instr_No* identifie l'adresse de l'esclave (par ex. 2500). *(Notez que pour Profibus dans le T940, les valeurs 1 et 2 ne peuvent pas être utilisées).*

Slot_No. N° de l'avéole E/S du module dans l'instrument, c'est à dire le numéro d'instance du module de contrôle dévolu dans la plage de 1 à 16.

Chan_No. N° de voie dans une alvéole permettant d'identifier une voie E/S dans un module multi-voie, dans la plage de 1 à 4.

Method. (Template/Block/Native) Indique l'emplacement du sous-programme de mise à jour de ST du bloc. Par défaut, il est intégré dans le modèle, le sous-programme de mise à jour du modèle peut être remplacé par une méthode différente qui se trouve dans le paquet du bloc. *(A l'heure actuelle, seul 'Template' peut être sélectionné pour ce champ).*

Mode. (Auto/Manual/Local) Mode de fonctionnement utilisé.

■ **Auto.** Il s'agit du mode de conduite par défaut. *In* est égal à la valeur des installations *Val* fournie par le 2500.

■ **Manual.** Permet de forcer *In* à une valeur spécifique en conduite, quel que soit la lecture réelle du 2500 (*In* déconnecté de *Val*).

■ **Local.** Le mode local est utilisé pour des tests et simulations, lorsqu'aucun 2500 n'est installé. Le bloc ne peut être mis à jour par le 2500, mais exécute du code supplémentaire pour répliquer les fonctions du 2500, à savoir les alarmes de limites. L'alarme de communication du 2500 est supprimée.

In. Etat d'entrée (sortie du bloc). En lecture seule et calculé à partir de *Val* en mode Auto.

Val. Etat d'entrée des installations, après inversion dans le 2500 qui calcule *In* — en mode Auto uniquement. Appliqué au champ *Val* de iTools.

AlmWhen. (InactStr/ActStr/Neither) Sélectionne la valeur de *In* qui déclenche l'alarme *Input*. Le fait de sélectionner **Neither** (ni l'un, ni l'autre) revient à sélectionner VRAI pour *DisableA* ou *DisableB* — c'est à dire que l'alarme est désactivée.

Alarms. Voir la description générale du champ *Alarms* au chapitre 2, Bloc ANIN.

Ces informations complètent celles de la table 15-33:

- **Hardware.** Défaut dans le module E/S — soit *ChanStat.Mod_Fail* OU *ChanStat.Not_Used* VRAI. Notez la différence par rapport au bloc D25_DI où *Not_Used* est une alarme séparée.
- **Input.** Alarme d'état de l'entrée. Déclenchée si *In* adopte la valeur sélectionnée par le champ *AlmWhen* — sauf si *AlmWhen* = **Neither**, ce qui désactive l'alarme. L'alarme est également désactivée si *DisableA* ou *DisableB* sont VRAIS.

NOTA. Ces champs d'alarmes sont calculés localement dans la base de données LIN et ne sont pas transmis au 2500.

DisableA, DisableB. Champs de désactivation (masquage) des alarmes. VRAI dans soit *DisableA* ou *DisableB* désactive l'alarme *Input*. Notez que le champ *TrueCond* est toujours mis à jour en fonction des résultats réels de l'évaluation des alarmes, quel que soit le paramétrage de *DisableA/B*, mais les bits *CurrCond* correspondants sont forcés à l'état de désactivation des alarmes (FAUX). VRAI dans *DisableA* OU *DisableB* OU (*AlmWhen* = **Neither**) désactive l'alarme *Input*.

NOTA. Si une alarme non acquittée est désactivée, elle le reste après avoir été désactivée.

TrueCond. VRAI si l'entrée est en alarme, FAUX dans le cas contraire. Affiche la condition réelle, quelles que soient les temporisations de désactivations configurées.

CurrCond. VRAI si l'entrée est en alarme et n'est pas désactivée de quelque manière que ce soit, mais ignore toute temporisation configurée (*AlmOnTim*, *AlmOfTim*).

Chaque fois que le bloc est traité, *CurrCond* est mis à jour avec les états d'alarme actifs basés sur les champs *TrueCond* et *DisableA/B* et *AlmWhen*. Si *CurrCond* a une valeur différente de sa dernière valeur dans *TrueCond*, un temporisateur interne est chargé avec la valeur *AlmOnTim* (si *CurrCond* est VRAI) ou avec *AlmOfTim* (si *CurrCond* est FAUX). Lorsqu'un temporisateur atteint zéro, l'état du champ *Alarms.Input* est mis à 1 par *CurrCond*.

NOTA. Si *CurrCond* change en permanence dans les limites de durée spécifiées pour *AlmOnTim* et *AlmOfTim*, l'état de la ou des alarmes signalées ne change pas.

Status. Champ binaire d'état de traitement des alarmes.

Ces informations complètent celles de la table 15-33:

- **NewAlarm.** Chaque fois qu'une nouvelle alarme est déclenchée (après une temporisation configurée, le cas échéant), ce bit passe à l'état VRAI. Il peut être remis à l'état initial, en sélectionnant VRAI pour *Options.AckNewAl*. Pour une remise automatique à l'état initial après une scrutation de la base de données, reliez *Status.NewAlarm* à *Options.AckNewAl*.

Cette fonctionnalité peut être utilisée dans un plan de gestion des alarmes, par exemple, pour générer un signal d'avertisseur externe, lorsqu'une nouvelle alarme se produit.

- **StartUp.** Ce bit est VRAI au cours de la première itération de la base de données après le démarrage, et repasse ensuite à l'état FAUX. Lorsque *StartUp* est VRAI, le temporisateur du bloc est mis à zéro. (Cette action a le même effet que de mettre *Options.InitLast* à l'état VRAI).

Failed. Défaut E/S ou de communication. VRAI si l'une des alarmes suivantes est déclenchée: *Software*, *Comms*, *Config*, *OpenCct*, *Hardware*, ou *EvalFail*. Notez que pour être prise en compte, la priorité d'une alarme doit avoir une valeur *différente de zéro*.

Notes de mise en oeuvre

Chaque bloc D25_DI_T inclus dans la base de données augmente sa taille d'au moins 206 octets (plus 8 octets, si des UC redondantes sont utilisées).

[Page laissée intentionnellement blanche]

Chapitre 16 BLOCS DE FONCTION ORGANISE

Les six blocs de fonction de la catégorie ORGANISE peuvent être classés en deux groupes — les blocs *zone/groupe* qui se composent des blocs AREA et GROUP et des blocs de *consignation de données* comprenant les blocs LOGDEV, LGROUP, LPTDEV et PGROUP.

Blocs Zone/Groupe

Les blocs AREA et GROUP permettent de configurer la base de données de zones et de groupes nécessaire pour la navigation de l’affichage et l’enregistrement des données de la mémoire flash dans le T2900 et T800.

Les blocs forment une hiérarchie à deux niveaux de 16 groupes maximum dans quatre* zones maximum. Chaque groupe comprend 16 éléments d’affichage maximum et 16 canaux d’enregistrement. Les groupes et zones peuvent se chevaucher, c’est à dire qu’un point peut apparaître dans plusieurs groupes et même plus d’une fois dans un même groupe. Un groupe peut également apparaître dans plusieurs zones.

Les blocs comprennent des sorties de sommaire d’alarmes, afin que l’état des alarmes du groupe/zone puisse être contrôlé.

La figure 16-1 illustre cette hiérarchie. L’exemple montre deux zones — *Area1* et *Area2*. *Area1* comprend deux groupes (*Zone1* et *Zone2*), chacune avec deux PID dans le groupe affichage et enregistrant SP et PV des deux PID. *Area2* comprend un seul groupe — *Réchauffeurs* — de quatre PID dans le groupe affichage, chacun enregistrant les PV.

**Une seule zone mise en oeuvre à l’heure actuelle.*

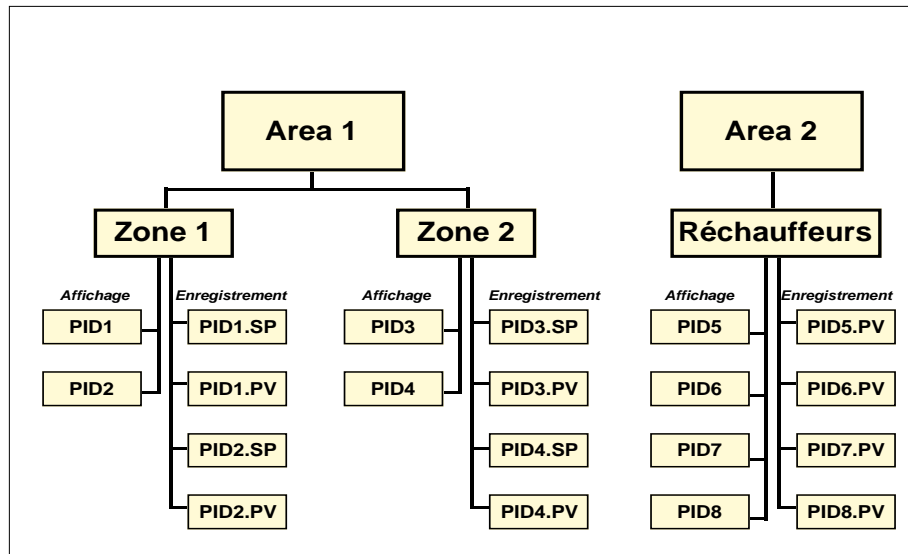


Figure 16-1 Exemple de hiérarchie groupe/zone

Prétracés T2900

L'une des fonctions importantes des blocs AREA et GROUP est de regrouper les points de consigne pour les afficher dans la vue de "prétracé" disponible dans le programmeur multi-consigne du T2900. (Voir les détails sur l'utilisation de cet affichage dans *T2900 Multi-Setpoint Programmer & Data Logger Customers' Manual*, réf. HA 260 754 U005). La configuration des blocs pour cette tâche est décrite dans le présent chapitre à la page 16-6 sous la rubrique *Utilisation des voies des blocs GROUP pour constituer le prétracé du T2900*.

Blocs de consignation des données

Les blocs LOGDEV et LGROUP constituent les blocs de points analogiques et logiques en groupe de consignation pour consigner les données sur un support d'archivage amovible. Un maximum de 16 points peuvent être associés à chaque groupe de consignation, et un maximum de 16 groupes de consignation peuvent être consignés sur une seule unité de consignation.

La figure 16-3 (page 16-8) montre schématiquement comment les blocs ORGANISE sont disposés pour collecter des données des blocs point — DR_DGCHP et DR_ANCHP — et comment ces données sont acheminées vers l'unité de consignation.

AREA: BLOC AREA

Fonction du bloc

Le bloc AREA permet d'associer un maximum de 16 blocs GROUP à une zone, de définir les modes* d'affichage de groupes disponibles, et d'associer un synoptique à la zone*. Le bloc génère également un ensemble d'alarmes de zone. La figure 16-1 montre un exemple de hiérarchie zone/groupe, et la figure 16-2 montre plus en détail comment le bloc est relié aux autres blocs dans les catégories ORGANISE et RECORDER.

Affichages. Si plus d'un bloc AREA est configuré*, la face avant du T2900 permet d'accéder à un écran 'System Overview' (Vue d'ensemble système), qui montre ces zones. L'écran utilisateur 'Area Overview' (Vue d'ensemble zone) est une représentation de tous les blocs GROUP associés aux champs *Group1* à *Group16* du bloc AREA sélectionné. Au bas de la hiérarchie d'affichage, les écrans 'Group Overview' (Vue d'ensemble Group) 'Point/Loop View' (Vue Point/Boucle) sont déterminés par les types de blocs d'affichage liés aux blocs GROUP par l'intermédiaire des champs* *Disp1* à *Disp16*. Voir figure 16-2.

*NOTA. La définition des modes d'affichage des groupes, l'association d'un synoptique à la zone, la configuration de plus d'une zone, les champs *Disp1* à *Disp16*, (et le champ *Mimic*) ne peuvent être utilisés à l'heure actuelle.

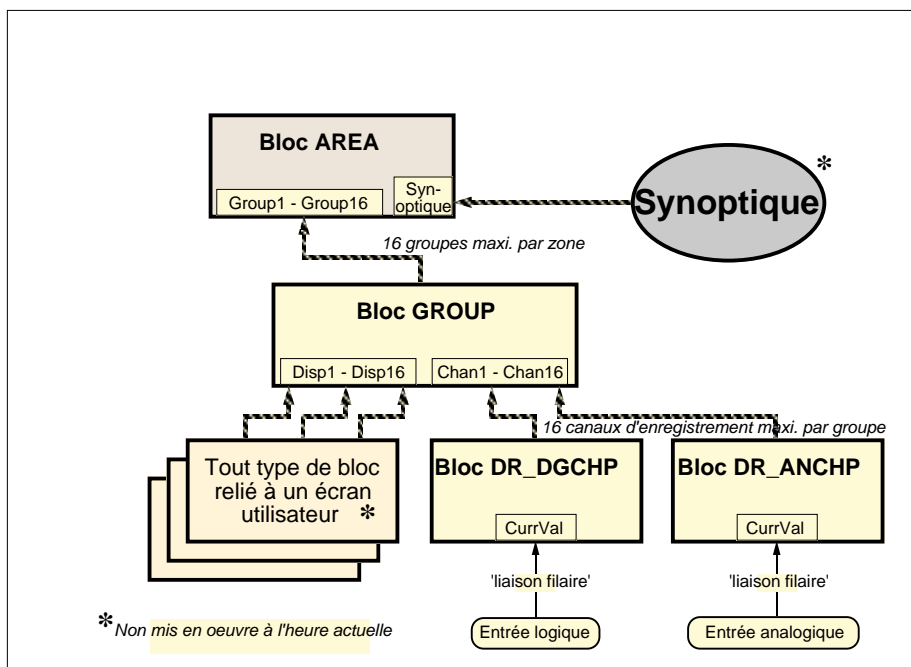


Figure 16-2 Enregistrement & affichage de blocs dans des groupes et zones

Paramètres du bloc

Les symboles de la table 16-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.







Paramètre	Fonction	Unités	Etat
Id	N° zone (1 à 4, 0=désact., par déf.)	Entier	
Group1-Group16	Nom bloc GROUP associé	Alphanumérique	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAI=erreur configuration (ex. Id ou Group <i>n</i> erronés)	V/F	
Area	VRAI=bloc GROUP est dans alarme Group	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlmAct	VRAI=bloc GROUP en alarme	V/F	 
AlmUnack	VRAI=alarme non acquittée dans bloc GROUP	V/F	 
DispMode	Définit modes affichage groupe disponible	(A)BCD hex	
V_Trend	Affichage vertical tendance (par déf.=VRAI)	V/F —1	D
H_Trend	Affichage horizonral tendance (par déf.=FAUX)	V/F —2	
FV_Trend	Affichage vert. tendance pl. larg. (par déf.=FAUX)	V/F —4	
V_Bars	Affichage barres verticales (par déf.=FAUX)	V/F —8	
H_Bars	Affichage barres horizontales (par déf.=VRAI)	V/F —1	C
Numeric	Affichage face avant numérique (par déf.=FAUX)	V/F —2	
Faceplat	Affichage face avant instrument (par déf.=VRAI)	V/F —4	
Mimic	Affichage synoptique utilisateur (par déf.=FAUX)	V/F —8	
FH_Trend	Affichage horiz. tendance pl. larg. (par déf.=FAUX)	V/F —1	B
		V/F —2	
		V/F —4	
		V/F —8	
Mimic	Synoptique associé à zone (1-999, 0=aucun par déf.)	Entier	
TrendOpt	Définit options présentation affichage tendance	(AB)CD hex	
DkBgLive	Fond sombre si tendance réelle (par déf.=FAUX)	V/F —1	D
DkBgRevw	Fond sombre si révision (par déf.=FAUX)	V/F —2	
Markers	Affiche marqueurs heure/date (par déf.=FAUX)	V/F —4	
SmoothSc	Défil. lent pour vue panoramique (par déf.=FAUX)	V/F —8	
Messages	Affiche messages des graphiques (par déf.=VRAI)	V/F —1	C
		2	
		4	
		8	
RecrdOpt	Définit options d'enregistrement	(ABC)D hex	
AlarmCol	Enregist. alarmes codés couleurs (par déf.=VRAI)	V/F —1	D
		V/F —2	
		V/F —4	
		V/F —8	
RecrdAlm	Bloc DR_ALARM identifier alarmes/évén. à enregist.	Alphanumérique	

Table 16-1 Paramètres du bloc AREA

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 16-1.

Id. Affecte un numéro de zone (1-4) à cette zone. Si plus d'une zone est configurée, la face avant du T2900 permet d'accéder à un écran 'System Overview', qui représente tous les blocs AREA. (*Seule une zone — Id 1 — est mise en oeuvre à l'heure actuelle*).

Group1 à Group16. Définit les blocs GROUP à associer à cette zone et qui seront affichés dans l'écran 'Area Overview' sur le T2900.

Alarms. L'état d'alarme zone est disponible comme sortie *AlmAct*, ainsi que comme alarme *Alarms.Area*. Si vous voulez superviser l'état sans générer d'alarme, désactivez *Area*.

DispMode. Définit quels modes d'affichage du groupe sont disponibles.








GROUP: BLOC GROUP

Fonction du bloc

Le bloc GROUP permet d'associer un maximum de 16 blocs de type affichage et 16 blocs de canaux enregistreurs à un groupe, de définir la vitesse et le mode d'échantillonnage de l'enregistrement et d'associer un synoptique au groupe. Le bloc permet également de disposer d'un ensemble d'alarmes de groupe. La figure 16-2 montre comment le bloc est associé à d'autres blocs dans les catégories de blocs ORGANISE et RECORDER.

Paramètres du bloc

Les symboles de la table 16-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Update	Vit. échant. enregist. (1-60 sec, par déf. = 0 sec)	Eng	
RecMode	Mode enregist. (par déf. = NORMAL)	Menu	
Mimic	Synoptique associé au groupe (1-999, 0 = aucun par déf.)	Entier	
Disp1-Disp16*	Nom bloc fournissant point affichage	Alphanumérique	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAI = erreur configuration (ex. Chan <i>n</i> ou Disp <i>n</i> erroné)	V/F	
Group	VRAI = Bloc source affichage en alarme	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlmAct	VRAI = Bloc source affichage en alarme	V/F	 
AlmUnack	TRUE = Bloc source affichage a une alarme non acquittée	V/F	 
Chan1-Chan16	Nom bloc canal enregistreur formant groupe enregistrement	Alphanumérique	

*Non mis en oeuvre à l'heure actuelle

Table 16-2 Paramètres du bloc GROUP

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 16-2.

Update. Valeurs en unités d'une seconde. A la valeur par défaut de zéro, il n'y a aucun enregistrement.

RecMode. (NORMAL/ADAPTIVE*) Mode d'enregistrement à utiliser.

NOTA. *L'option ADAPTIVE n'est pas disponible à l'heure actuelle.

Disp1 à Disp16. Définit les blocs de type affichage à associer à ce groupe, qui seront affichés sur les écrans 'Group Overview' et 'Loop/Point View' du T2900.

NOTA. Les champs *Disp1* à *Disp16* ne sont pas mis en oeuvre à l'heure actuelle.

Alarms. L'état d'alarme du bloc affichage est disponible comme sortie *AlmAct* et également comme alarme *Alarms.Group*. Si vous voulez contrôler l'état sans générer d'alarme, désactivez *Group*.

Chan1 à Chan16. Définit les blocs de canaux enregistreurs — DR_ANCHP et DR_DGCHP — à associer à ce groupe. Ces blocs transmettent les données analogiques et/ou logiques à enregistrer dans la zone de mémoire flash de l'ISE.

Utilisation des voies des blocs GROUP pour créer le prétracé des T2900

L'une des fonctions importantes des paramètres *Chan1* à *Chan16* est de regrouper les points de consigne pour les afficher dans la vue de "prétracé" du programmeur multi-consignes du T2900. (Voir les détails sur l'utilisation de cet affichage dans *T2900 Multi-Setpoint Programmer & Data Logger Customers' Manual*, réf. HA 260 754 U005).

La figure 16-3 montre comment un groupe de deux points de consigne analogiques et un point de consigne logique doivent être associés à un bloc GROUP — par l'intermédiaire des blocs DR_ANCHP et DR_DGCHP — pour créer un affichage de prétracé valable (Voir les détails sur les blocs DR_ANCHP et DR_DGCHP au chapitre 17, *Blocs de fonction Recorder*).

Pour que l'affichage de prétracé soit valable, les voies du bloc GROUP doivent être associées en ordre numérique strict aux points de consigne générés par le programme de points de consigne. La procédure est la suivante:

- 1 D'abord, les PV analogiques doivent être associées dans l'ordre numérique défini par le programme de points de consigne. Autrement dit, PV1 analogique (provenant du point de consigne analogique 1) doit être associé à *Chan1*, PV2 analogique à *Chan2*, etc., jusqu'à ce que toutes les PV analogiques aient été traitées - dans cet exemple, il n'y en a que deux.
- 2 Ensuite, tout point logique doit être associé dans l'ordre du programme aux voies suivantes du bloc GROUP. Dans cet exemple, le point de consigne logique 1 unique doit être associé à *Chan3*.

NOTA. Dans l'affichage de prétracé, les points de consigne logiques sont toujours affichés comme des tendances uniques, à l'inverse des points analogiques affichés comme des paires PV/SP dans la région "passé". Vous pouvez en fait choisir d'afficher n'importe quels point(s) logique(s) par l'intermédiaire des voies logiques soit calculés localement ou à distance.

- 3 Lorsque toutes les PV analogiques et logiques ont été traitées, les SP analogiques sont associés aux voies restantes dans l'ordre. Dans cet exemple, SP1 analogique est associé à *Chan4*, et SP2 analogique est associé à *Chan5* pour compléter l'ensemble.

NOTA. Les PV sont calculées sur la base des paramètres PV des blocs modules de contrôle dévolu (voir figure 16-3) et des SP des paramètres *Out* des blocs. (Voir les détails au chapitre 15, *Blocs de fonction DCM*).

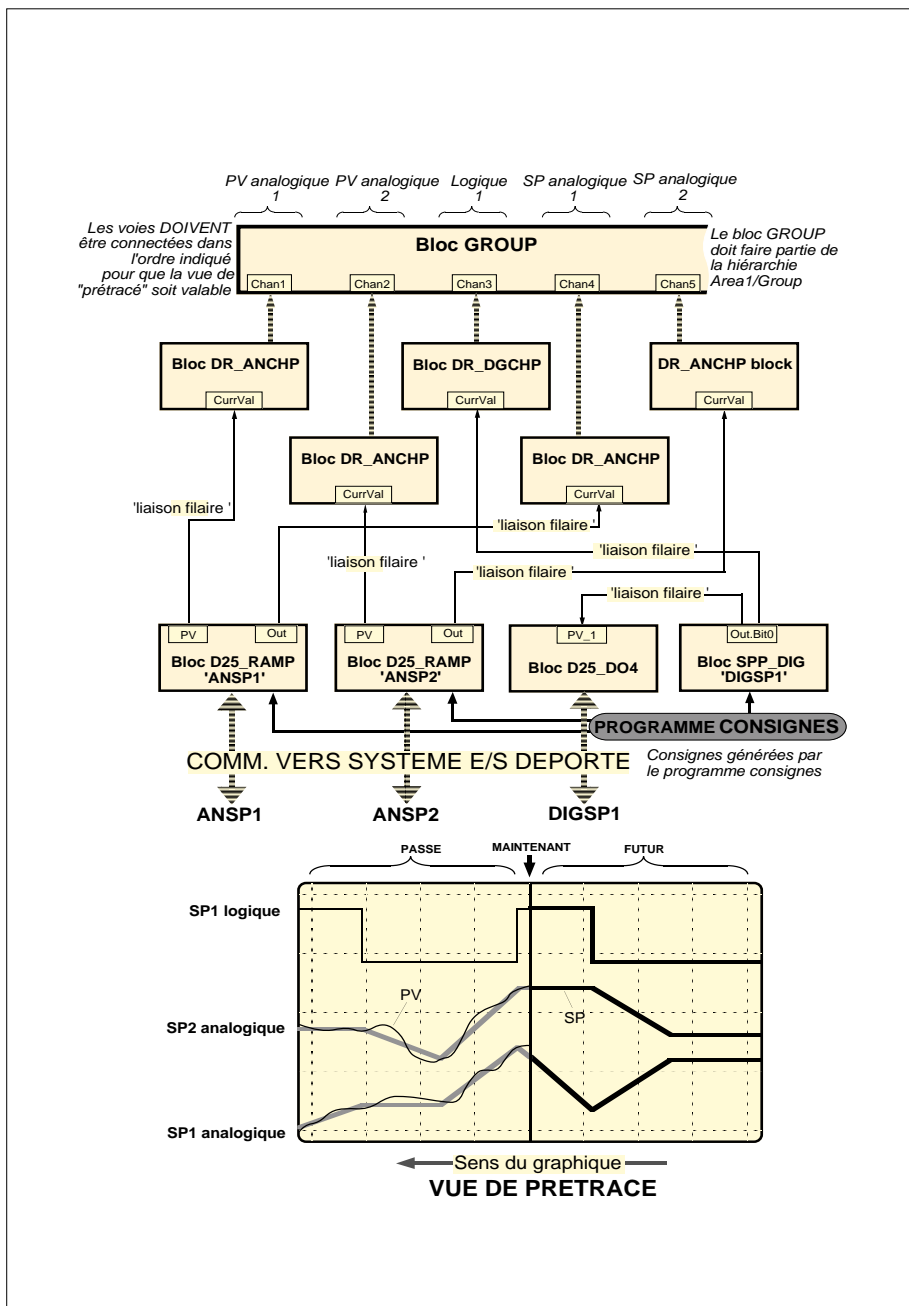


Figure 16-3 Structure des voies des points de consigne pour le "prétracé" du T2900

LOGDEV: BLOC UNITÉ DE CONSIGNATION

Fonction du bloc

Le LOGDEV permet de définir et de contrôler l'accès à un support d'archivage et de définir un maximum de 16 groupes de points (groupes de consignation) qui doivent être consignés sur ce support.

Chaque groupe de consignation est à son tour défini par un bloc LGROUP, qui collecte les données de 16 blocs point DR_DGCHP, DR_ANCHP, DR_ALARM et DR_REPRT.

La figure 16-4 montre comme les blocs ORGANISE (grisés) interconnectent les données point et l'unité de consignation.

NOTA. La hiérarchie de la figure 16-4 coexiste avec, mais n'est pas la même que la hiérarchie Zone/Groupe utilisée par la catégorie de blocs RECORDER pour l'enregistrement flash des données (voir figure 16-2).

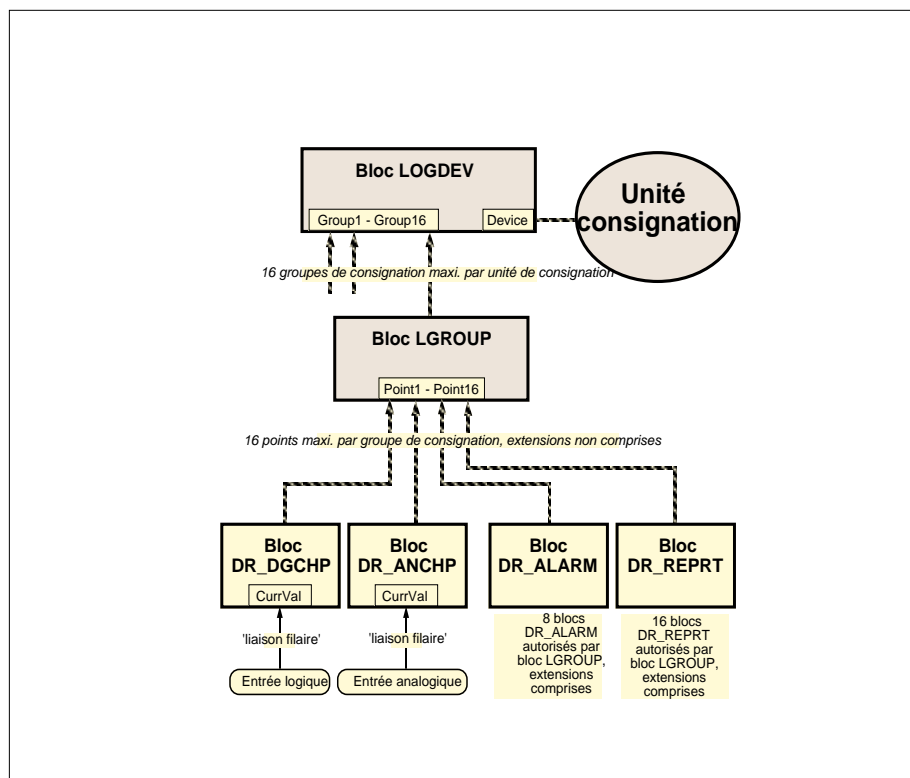


Figure 16-4 Organisation des blocs point pour la consignation des données

Paramètres du bloc

Les symboles de la table 16-3 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.













Paramètre	Fonction	Unités	Etat
State	Définit quand on peut accéder au support	Menu	
Status	Indique état de la consignation	Menu	 
Trigger	VRAI active consignation si State = TRIGGER	V/F	
Options	Définit les options de consignation	(AB)CD hex	
Del_ASC	VRAI = Supprime .ASC - stratégie suppression (par déf.)	V/F — 1	D
Del_PKD	VRAI = Supprime .PKD - stratégie suppression (par déf.)	V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	
New_File	VRAI arrête suppression fichier consign. démar. froid	V/F — 1	C
Réserve		V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	
Device	Définit nom unité de consignation (par déf. = 'A')	Alphanumérique	
Group1-Group16	Définit noms des blocs LGROUP	Alphanumérique	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAI=erreur config. (ex. Id groupe consign. ou nom unité erronés)	V/F	
DiskFull	VRAI = espace insuffisant sur support archivage	V/F	
No Disk	VRAI = pas de support archivage	V/F	
Disk Err	VRAI = tout autre erreur disque	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
FreeSpac	Espace disponible sur le support (%)	Eng	 
FreeTime	Durée consign. restante à vitesse actuelle (heures)	Eng	 
DiskFull	% disque plein av. décl. alarme DiskFull (par déf. = 95%)	Eng	

Table 16-3 Paramètres du bloc LOGDEV

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 16-3.

State. (ON/OFF/TRIGGER) Détermine quand on peut accéder au support, en spécifiant à quel moment la consignation est permise. 'ON' (par déf.) active et 'OFF' désactive la consignation. 'TRIGGER' (déclenchement) permet à l'entrée logique *Trigger* d'activer/désactiver la consignation.

Status. (INACTIVE/ACTIVE/FLUSHING/OFF_LINE) Indique l'état en cours de la consignation sur disque. La table décrit les options.

Option...	...Signifie
INACTIVE	Aucun fichier ouvert
ACTIVE	Des fichiers sont ouverts
FLUSHING	Copie sur disque
OFF_LINE	Consignation terminée

Table 16-4 Signification des options de Status

Options. Champ binaire permettant de définir les options de consignation.

■ **New_File.** La mise à 1 de ce bit empêche la suppression du fichier de consignation .ASC ou .PKD en cas de démarrage à froid. Dans ce cas, les fichiers sont fermés et de nouveaux fichiers de consignation sont ouverts au démarrage à froid, les extensions étant modifiées en AS1 ... AS9, A10 ... A99 et PKD en PK1 ... PK9, P10 ... P99.

Group1 à Group16. Définit les seize blocs LGROUP maximum à associer à l'unité de consignation spécifiée dans le champ *Device*. Saisissez les noms des blocs dans les champs en question. (La figure 16-4 montre la manière dont les blocs LGROUP et LOGDEV sont reliés entre eux).

LGROUP: BLOC GROUPE DE CONSIGNATION

Fonction du bloc

Le bloc LGROUP collecte les données d'un maximum de seize blocs point DR_DGCHP et DR_ANCHP — extensible à un total de 80 — et permet de les archiver sur un support spécifié par le bloc LOGDEV associé. Un total de huit blocs DR_ALARM et de seize blocs DR_REPRT peuvent également être rattachés à un seul bloc LGROUP, y compris toute extension de bloc LOGRPEX.

Le figure 16-4 montre comment le bloc intervient entre les données point dans le groupe de consignation et l'unité de consignation.

NOTA. La hiérarchie de la figure 16-4 coexiste avec, mais n'est pas la même que la hiérarchie Zone/Groupe utilisée par la catégorie de blocs RECORDER pour l'enregistrement flash des données (voir figure 16-2).

Paramètres du bloc

Les symboles de la table 16-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.









Paramètre	Fonction	Unités	Etat
State	Définit quand le groupe doit être consigné	Menu	
UpdateA	Intervalle consignation en sec. (par déf.=0.0s)	Eng	
UpdateB	Intervalle consignation alternatif (par déf.=0.0s)	Eng	
UpdB_On	VRAI active UpdateB (par déf.=FAUX)	V/F	
LogNow	VRAI consigne échant. unique, puis réinit.	V/F	 
Trigger	VRAI active consignation si State = TRIGGER	V/F	
Extend	Nom bloc LOGRPEX ajoutant points consignés	Alphanumérique	
Point1-Point16	Spécifie noms blocs point rattachés	Alphanumérique	
Alarms			  
Software	AnoM mém. données RAM bloc/défaut réseau	V/F	
Config	VRAI=erreur config. (ex. Id groupe consign. erroné, etc.)	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
NameType	Définit le nom du fichier	Menu	
FileType	Définit format fichier généré par groupe consign.	Menu	
FileName	Définit nom de base fichier à générer	Alphanumérique	
ColTitle	Définit si titres colonnes ASCII affichés	Menu	
DateFmt	Définit format date et/ou heure ASCII	Menu	
ComRatio	Définit si données sont compressées ou normales	Menu	
DataRate	Vitesse consignation (Ko/heure)	Eng	

Table 16-5 Paramètres du bloc LGROUP

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 16-5.

State. (ON/OFF/TRIGGER) Détermine quand le groupe de consignation doit être consigné. ‘ON’ (par défaut) active et ‘OFF’ désactive la consignation. ‘TRIGGER’ permet à l’entrée logique *Trigger* d’activer/désactiver la consignation.

Extend. Nom d’un bloc d’extension de groupe de consignation (LOGGRPEX) qui permet de consigner 16 points supplémentaires. Vous pouvez ajouter davantage de points consignés - jusqu’à un total de 80 - en renvoyant à un autre bloc d’extension dans chaque bloc LOGGRPEX (par l’intermédiaire du champ *Next* du bloc).

NameType. (Text/Hourly/Daily) Définit le nom du fichier.

FileType. (ASCII/Packed) Définit le type de format de fichier généré par le groupe de consignation.

FileName. Chaîne de caractères spécifiant le nom de base du fichier qui doit être généré par la consignation. Si *NameType* est autre que texte, seuls les deux premiers caractères peuvent être modifiés.

ColTitle. (Standard/None) Avec des titres de colonnes ASCII, si vous sélectionnez “Standard” pour *ColTitle*, les titres seront ajoutés aux fichiers de consignation (par défaut). “None” signifie que les titres sont omis.

DateFmt. (DateTime/Sprdsht/Integer/Duration/Days/D,H,M,S) Avec des colonnes date et/ou heure ASCII, *DateFmt* détermine leur présentation.

La table 16-6 liste les exemples des options.

Option	Exemple
DateTime	01/01/88 00:02:26
Sprdsht	32143.001690
Integer	980605123030 (=5/6/98 @ 12:30:30)
Duration	00:02:26
Days	08
D,H,M,S	08:00:02:26

Table 16-6 Exemples d’options de DateFmt

ComRatio. (Normal/High) ‘facteur de compression’. Si vous sélectionnez “Packed” pour *FileType* et “Normal” pour *ComRatio*, les données ne sont pas compressées (par défaut), mais elles le seront si vous sélectionnez “High” pour *ComRatio*.

LOGGRPEX: BLOC D’EXTENSION DE GROUPE DE CONSIGNATION

Fonction du bloc

Le bloc LOGGRPEX permet de consigner 16 points supplémentaires dans un bloc LGROUP. Vous pouvez consigner davantage de points - jusqu’à un maximum de 80 - en reliant en série d’autres blocs d’extension par l’intermédiaire de leur champs *Next*.

Paramètres du bloc

Les symboles de la table 16-7 sont expliqués au chapitre 1.




Paramètre	Fonction	Unités	Etat
Next	Nom autres blocs LOGGRPEX ajout. points consignés	Alphanumérique	
Point1-Point16	Définit les noms des blocs point rattachés	Alphanumérique	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAl=erreur config. (ex. Id gr. consign. erroné, etc.)	V/F	
Combined	Combinaison de tous les bits d’alarme	V/F	

Table 16-7 Paramètres du bloc LOGGRPEX

LPTDEV: BLOC IMPRIMANTE

Fonction du bloc

Le bloc LPTDEV définit l'accès à une imprimante et spécifie quels groupes d'impression s'appliquent à cette imprimante. La figure 16-5 montre comment les blocs ORGANISE (grisés) relient les points de données et l'imprimante.

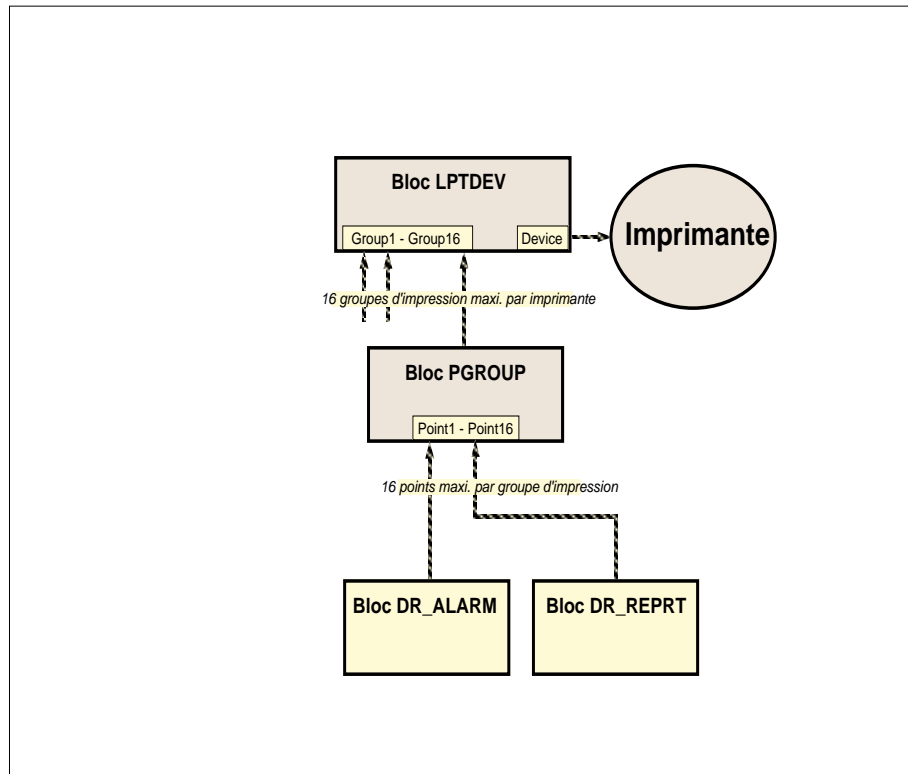


Figure 16-5 Configuration des blocs point DR_ALARM & DR_REPT pour l'impression

Paramètre du bloc

Les symboles de la table 16-8 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
State	Définit quand on peut accéder à l'imprimante	Menu	
Status	Rapport état impression en cours	Menu	
Trigger	VRAI active impression si State = TRIGGER	V/F	
Options	Définit options d'impression (<i>non encore utilisé</i>)	ABCD hex	
Device	Définit nom imprimante (par déf. = 'PRINTER1')	Alphanumérique	
AlmForm	Nom fichier formul. imprim. .UYT opt. sort. alarme	Alphanumérique	
Group1-Group16	Définit noms blocs PGROUP (gr. impr.)	Alphanumérique	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAI=erreur config. (ex. groupe ou nom imprim. erronés)	V/F	
Form	VRAI = erreurs dans <i>AlmForm</i> (.UYT)	V/F	
PaperEnd	VRAI = plus de papier dans imprimante	V/F	
ErrorLPT	VRAI = erreur imprimante	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Tx_Cnt	Nbre octets envoyés imprimante - mise en route	Entier	
FormErr	Code erreur formulaire	Menu	
FormErLn	N° ligne où erreur <i>FormErr</i> est survenue	Entier	
FormErCh	Pos. caract. erreur dans ligne signalée par <i>FormErLn</i>	Entier	

Table 16-8 Paramètres du bloc LPTDEV

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 16-8.

State. (ON/OFF/TRIGGER) Détermine quand on peut accéder à l'imprimante, en spécifiant quand l'impression est autorisée. 'ON' (par défaut) active et 'OFF' désactive l'impression. 'TRIGGER' permet à l'entrée logique *Trigger* d'activer/désactiver l'impression.

Status. (INACTIVE/ACTIVE/PRINTING/OFF_LINE) Indique l'état de l'impression en cours. La table 16-9 décrit les options.

Option...	...Signifie
INACTIVE	Impression non autorisé
ACTIVE	Impression autorisée
PRINTING	Impression en cours
OFF_LINE	Imprimante hors ligne

Table 16-9 Signification des options de Status

Device. (PRINTER1/PRINTER2) L'imprimante à utiliser pour l'impression (uniquement PRINTER1 à l'heure actuelle).

AlmForm. Nom d'un fichier de formulaire d'imprimante en option (extension .UYT), qui peut être utilisé pour personnaliser la présentation du texte de la sortie des alarmes.

Group1 à Group16. Définit les 16 blocs (PGROUP) à associer à l'imprimante définie dans le champ *Device*. Saisissez les noms des blocs dans les champs en question. (La figure 16-5 montre comment les blocs the PGROUP et LPTDEV sont reliés entre eux).

FormErr. (OK /BAD_FILE /LINE_LEN /NEWLINE /MEMORY /SYNTAX /RANGE /NAME /DICTNRY /TYPE /ACTION /FORM /OTHER) Code d'erreur formulaire. La

signification des options est la suivante:

- **OK.** Aucune erreur.
- **BAD_FILE.** Fichier introuvable ou ne peut être lu.
- **LINE_LEN.** Une ligne du fichier trop longue (limite 255, RC et/ou CL non compris).
- **NEWLINE.** Fichier ne se termine pas par une nouvelle ligne (probablement corrompu).
- **MEMORY.** Formulaire trop grand pour la mémoire affectée.
- **SYNTAX.** Erreur de syntaxe.
- **RANGE.** Une valeur numérique est en dehors de la plage admissible.
- **NAME.** Objet désigné introuvable (sans doute mal orthographié).
- **DICTNRY.** Entrée de dictionnaire introuvable.
- **TYPE.** Type d'objet inapproprié dans le contexte où il est utilisé.
- **ACTION.** (*Non mis en oeuvre*)
- **FORM.** Un construit a été utilisé, ce qui n'est pas valable dans un fichier de formulaire.
- **OTHER.** Autre erreur.

PGROUP: BLOC GROUPE IMPRIMANTE

Fonction du bloc

Le bloc PGROUP collecte les données de seize blocs point DR_ALARM et DR_REPRT et permet de les sortir sur une imprimante, définie dans le bloc LPTDEV associé.









Ce bloc peut être utilisé pour regrouper des valeurs consignées périodiquement (similaire à l'établissement de tendances et à la consignation*) et également de spécifier des rapports qui peuvent être générés sur l'imprimante.

La figure 16-5 montre comme le bloc PGROUP intervient entre les données point dans le groupe d'impression et l'imprimante.

**Non pris en charge à l'heure actuelle.*

Paramètres du bloc

Les symboles de la table 16-10 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
State	Définit quand le groupe doit être imprimé	Menu	
UpdateA*	Intervalle impression en sec. (par déf. = 0.0s)	Eng	
UpdateB*	Intervalle impression altern. (par déf. = 0.0s)	Eng	
UpdB_On*	VRAI active UpdateB (par déf. = FAUX)	V/F	
PrintNow*	VRAI imprime échant. unique, et réinit.	V/F	 
Trigger	VRAI active impression si State = TRIGGER	V/F	
Point1-Point16	Définit noms blocs points rattachés	Alphanumérique	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	VRAI=erreur config. (ex. Id groupe impression erronée, etc)		V/F
Combined	Combinaison de tous les bits d'alarme	V/F	
FrqTitle*	Fréquence impression titre (valeurs imprimées périod.)		Menu
ColTitle*	Sélect. si titres colonnes ASCII imprimés	Menu	
ColWidth*	Largeur colonnes en caractères (5 - 16; par déf. = 8)		Entier
DateFmt*	Définit format date &/ou heure ASCII	Menu	
DataRate	Vitesse impression (Ko/heure)	Eng	

**Non mis en oeuvre à l'heure actuelle*

Table 16-10 Paramètres du bloc PGROUP

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table16-10.

State. (ON/OFF/TRIGGER) Détermine quant le groupe d'impression doit être imprimé. 'ON' (par défaut) active et 'OFF' désactive l'impression. 'TRIGGER' permet à l'entrée logique *Trigger* d'activer/désactiver l'impression.

Alarms. Ces informations sont données en plus de celles de la table 16-10:

■ **Config.** VRAI en cas d'erreur de configuration — par ex. Id de groupe d'imprimante erronée, référence de bloc point erronée ou caractères non autorisés dans les noms de fichier.

FrqTitle. (Never/Start/Always) La fréquence à laquelle les titres sont imprimés pour des valeurs imprimées périodiquement. (*Non mis en oeuvre à l'heure actuelle*).

ColTitle. (Standard/None) Avec des titres de colonnes ASCII, si vous sélectionnez "Standard" pour *ColTitle*, les titres seront ajoutés aux sorties papier (par défaut). "None" signifie que les titres sont omis. (*Non mis en oeuvre à l'heure actuelle*).

DateFmt. (DateTime/Sprdsht/Integer/Duration/Days/D,H,M,S) Avec des colonnes date et/ou heure ASCII, *DateFmt* détermine leur présentation. (*Non mis en oeuvre à l'heure actuelle*).

La table 16-11 liste des exemples d'options.

Option	Exemple
DateTime	01/01/88 00:02:26
Sprdsht	32143.001690
Integer	980605123030 (=5/6/98 @ 12:30:30)
Duration	00:02:26
Days	08
D,H,M,S	08:00:02:26

Table 16-11 Exemples d'options DateFmt

[Page laissée intentionnellement blanche]

Chapitre 17 BLOCS DE FONCTION RECORDER

Les blocs de fonction de la catégorie RECORDER — DR_ANCHP et DR_DGCHP — permettent d'enregistrer en mémoire flash des points analogiques et logiques dans des enregistreurs graphiques vidéo. Les points analogiques et logiques sont “rattachés” comme entrées aux blocs depuis la base de données. Pour que l'enregistrement flash se produise, les blocs doivent être disposés en groupes et zones par l'intermédiaire des blocs GROUP et AREA (dans la catégorie ORGANISE), voir figure 17-1.

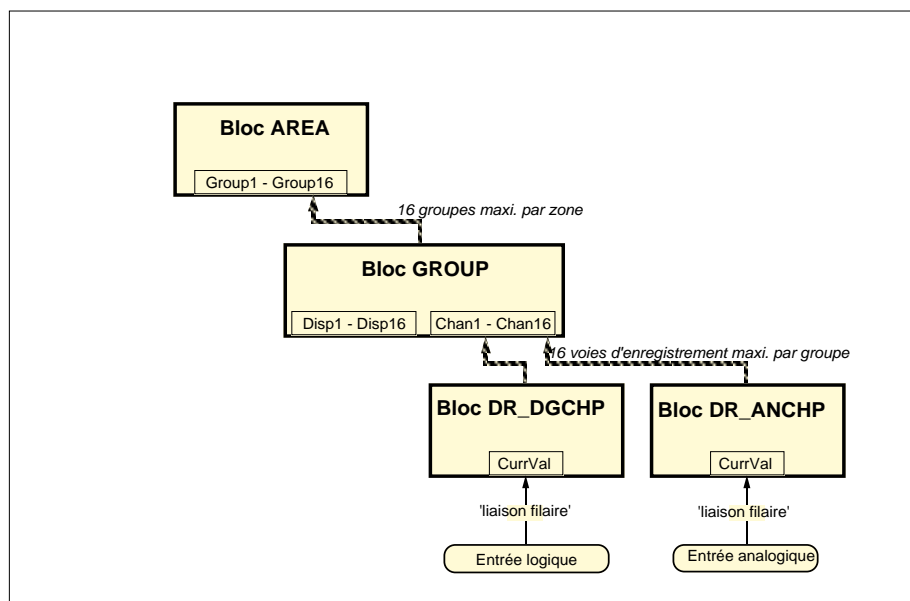


Figure 17-1 Disposition des blocs recorder en groupes et zones

NOTA. Les blocs point DR_ANCHP et DR_DGCHP permettent également d'archiver de manière permanente des données sur un support amovible pour pouvoir les repasser hors ligne. Il s'agit là de consignation de données qui est différent de l'enregistrement flash où les données sont repassées dans l'instrument. Pour la consignation des données, les blocs point doivent être disposés en *groupes de consignation* par l'intermédiaire de blocs spéciaux de la catégorie ORGANISE — LOGDEV, LGROUP et LOGGRPEX. Voir les détails au chapitre 16.

DR_ANCHP: BLOC POINT VOIE ANALOGIQUE D'ENREGISTREMENT DE DONNÉES

Fonction du bloc

Le bloc de fonction DR_ANCHP permet d'enregistrer l'historique d'un ensemble de valeurs de données analogiques sur le système d'archivage flash local du T2900 qui exécute le bloc. L'enregistrement peut ensuite être repassé dans l'instrument.

Le bloc spécifie comment la valeur analogique doit être analysée et enregistrée. La fonctionnalité des voies analogiques mesurées et calculées est assurée par le bloc.

Le point analogique à enregistrer doit être "rattaché" dans la base de données au champ d'entrée *CurrVal* du bloc. Le bloc doit être rattaché à un bloc GROUP — par l'intermédiaire d'un champ *Chann* — pour que l'enregistrement puisse avoir lieu. Un total de 16 blocs d'enregistrement de données (blocs DR_ANCHP et DR_DGCHP) peuvent être rattachés à un bloc GROUP donné.

Paramètres du bloc

Les symboles de la table 17-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
CurrVal	Valeur point active	Eng	
Quality	Qualité de CurrVal — Définitions enregistreur	(A)BCD hex	
GOOD	GOOD	V/F — 1	D
PV_OFF	PV_OFF	V/F — 2	
O_RANGE	OVER_RANGE	V/F — 4	
U_RANGE	UNDER_RANGE	V/F — 8	
HW_ERROR	HARDWARE_ERROR	V/F — 1	C
RANGING	RANGING	V/F — 2	
OVERFLOW	OVERFLOW	V/F — 4	
BAD_PV	BAD_PV	V/F — 8	
HW_EXCED	HARDWARE_EXCEEDED	V/F — 1	B
NO_DATA	NO_DATA	V/F — 2	
LIN TABL	TOO_MANY_LIN_TABLES	V/F — 4	
<i>Réserve</i>		V/F — 8	
Trace_On	VRAI=trace activée (par déf.), FAUX=désactivée	V/F	
Format	Format affichage et consignation de CurrVal	Menu	
ColourA	Couleur à utiliser pour la trace	Menu	
ColourB	Couleur alternative à utiliser pour la trace	Menu	
ColB_On	VRAI sélectionne couleur B, FAUX A (par déf.)	V/F	
ZoneA_HI	Zone A val. haute (0-100% largeur graphique, 100%=par déf.)	%	
ZoneA_LO	Zone A val. basse (0-100% largeur graphique, 100%=par déf.)	%	
ZoneB_HI	Zone B val. haute (0-100% largeur graphique, 100%=par déf.)	%	
ZoneB_LO	Zone B val. basse (0-100% largeur graphique, 100%=par déf.)	%	
ZoneB_On	VRAI sélectionne zone B, FAUX A (par déf.)	V/F	

Table 17-1 suite...

...Table 17-1 suite



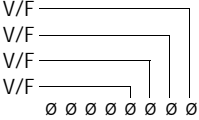
Paramètre	Fonction	Unités	Etat
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
<AlmTyp_1>	Alarme utilisateur définie par AlmTyp_1	V/F	
<AlmTyp_2>	Alarme utilisateur définie par AlmTyp_2	V/F	
<AlmTyp_3>	Alarme utilisateur définie par AlmTyp_3	V/F	
<AlmTyp_4>	Alarme utilisateur définie par AlmTyp_4	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlmAct			
Alm_1	VRAI = Alarme AlmTyp_1	V/F	
Alm_2	VRAI = Alarme AlmTyp_2	V/F	
Alm_3	VRAI = Alarme AlmTyp_3	V/F	
Alm_4	VRAI = Alarme AlmTyp_4	V/F	
AlmTyp_1-AlmTyp_4	Types d'alarme pour Alm_1 - 4	Menu	
SpanA_HI	Valeur maxi. zone état haut (par déf.=100.0)	Eng	
SpanA_LO	Valeur maxi. zone état bas (par déf.=0.0)	Eng	
SpanB_HI	Valeur maxi. zone état haut (par déf.=100.0)	Eng	
SpanB_LO	Valeur maxi. zone état bas (par déf.=0.0)	Eng	

Table 17-1 Paramètres du bloc DR_ANCHP

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 17-1.

Quality. Ce paramètre correspond à l'énumération des états PV utilisée dans les enregistreurs Eurotherm. Vous pouvez rattacher toute valeur logique appropriée au champ *Quality*

Format. Définit le format de nombre utilisé pour l'affichage et la consignation de l'entrée analogique *CurrVal*. La table 17-2 liste les options disponibles avec des exemples.

Option	Exemple
XXXXX	12345
XXXX.X	1234.5
XXX.XX	123.45
XX.XXX	12.345
X.XXXX	1.2345
XXXXXXXX	12345678
XXXXXX.X	123456.7
XXXXX.XX	12345.67
XXXX.XXX	1234.567
XXX.XXXX	123.4567
XX.XXXXX	12.34567
X.XXXXXX	1.234567
X.XXXE+XX	1.234E+56

Table 17-2 Options du paramètre Format avec des exemples

ColourA. (Default/Violet/Blue/Green/Brown/Red/Black/DkViolet/DkBlue/DkGreen/DkBrown/DkRed/DkGrey/LiViolet/LiBlue/LiGreen/LiBrown/LiRed/LiGrey) La couleur à utiliser pour la trace. Les préfixes ‘Dk’ et ‘Li’ représentent les versions sombres et claires des couleurs de base, et gris sombre et gris clair. Si vous spécifiez *Default*, la couleur est déterminée par le numéro de *Chann* du bloc dans le bloc GROUP.

ColourB. (Default/Violet/Blue/Green/Brown/Red/Black/DkViolet/DkBlue/DkGreen/DkBrown/DkRed/DkGrey/LiViolet/LiBlue/LiGreen/LiBrown/LiRed/LiGrey/FlViolet/FlBlue/FlGreen/FlBrown/FlRed/FlBlack) Comme pour *ColourA* ci-dessus, mais les versions clignotantes des couleurs de base et le noir sont également sélectionnables (représentées par le préfixe ‘Fl’).

ZoneA_LO, ZoneA_HI. La zone du “graphique” (perpendiculaire à la base de temps) occupée par une tendance est appelée *zone*. Spécifiée comme une paire de valeurs marquant le début et la fin de la zone, chacune dans une plage de 0% à 100%. *ZoneA_LO* et *ZoneA_HI* spécifient les valeurs haute et basse pour la zone A.

ZoneB_LO, ZoneB_HI. (*Spécifications de la Zone B non mises en oeuvre à l’heure actuelle*).

ZoneB_On. (*Spécifications de la Zone B non mises en oeuvre à l’heure actuelle*)

Alarms. Les alarmes <AlmTyp_1> to <AlmTyp_4> sont spécifiées par les paramètres correspondants (voir suivant) et déclenchées par les entrées respectives *Alm_1* à *Alm_4* dans le champ binaire *AlmAct*.

AlmTyp_1 à AlmTyp_4. (ABS LOW /ABS HIGH /DEV IN /DEV OUT /ROC RISE /ROC FALL) Ces paramètres permettent de spécifier quatre alarmes utilisateur à apparaître dans le champ *Alarms*, qui doivent correspondre aux entrées d’alarme rattachées au champ binaire *AlmAct*. La table 17-3 liste les options possible.

Option	Type d’alarme
ABS LOW	Basse absolue
ABS HIGH	Haute absolue
DEV IN	Ecart bas
DEV OUT	Ecart haut
ROC RISE	Vitesse de variation croissante
ROC FALL	Vitesse de variation décroissante

Table 17-3 Options du type d’alarme (AlmTyp_1 à AlmTyp_4)

SpanA_HI, SpanA_LO. ‘Span’ définit la plage des valeurs qui doivent être affichées dans la tendance. (La plage de valeurs *enregistrées* n’est pas affectée). Span est spécifié pour la zone A sous la forme d’une paire de valeurs en unités physiques pour la plage haute — *SpanA_HI* — basse — *SpanA_LO* — de la zone. La tendance est en fait écrêtée entre ces deux valeurs.

SpanB_HI, SpanB_LO. (*Spécifications de la Zone B non mises en oeuvre à l’heure actuelle*)

DR_DGCHP: BLOC POINT VOIE LOGIQUE D'ENREGISTREMENT DE DONNEES

Fonction du bloc

Le bloc de fonction DR_DGCHP permet d'enregistrer l'historique d'un ensemble de valeurs de données logiques sur le système d'archivage flash local du T2900 qui exécute le bloc. L'enregistrement peut ensuite être repassé dans l'instrument.

Le bloc spécifie comment la valeur logique doit être analysée et enregistrée.

Le point logique à enregistrer doit être "rattaché" dans la base de données au champ d'entrée *CurrVal* du bloc. Le bloc doit être rattaché à un bloc GROUP — par l'intermédiaire d'un champ *Chann* — pour que l'enregistrement puisse avoir lieu. Un total de 16 blocs d'enregistrement de données (blocs DR_DGCHP et DR_ANCHP) peuvent être rattachés à un bloc GROUP donné.

Paramètres du bloc

Les symboles de la table 17-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
CurrVal	Valeur point active	V/F	☐▶
Quality	Qualité de CurrVal — Définitions enregistreur	(A)BCD hex	☐▶
PV_OFF	PV_OFF	V/F — 1	D
O_RANGE	OVER_RANGE	V/F — 2	
U_RANGE	UNDER_RANGE	V/F — 4	
HW_ERROR	HARDWARE_ERROR	V/F — 8	
RANGING	RANGING	V/F — 1	C
OVERFLOW	OVERFLOW	V/F — 2	
BAD_PV	BAD_PV	V/F — 4	
HW_EXCED	HARDWARE_EXCEEDED	V/F — 8	
NO_DATA	NO_DATA	V/F — 1	B
LIN TABL	TOO_MANY_LIN_TABLES	V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	
Trace_On	VRAI=trace activée (par déf.), FAUX=désactivée	V/F	☐▶
Format	Format affichage et consignation de CurrVal	Menu	
ColourA	Couleur à utiliser pour la trace	Menu	
ColourB	Couleur alternative à utiliser pour la trace	Menu	
ColB_On	VRAI sélectionne couleur B, FAUX A (par déf.)	V/F	☐▶
ZoneA_HI	Zone A val. haute (0-100% largeur graphique, 100%=par déf.) %		
ZoneA_LO	Zone A val. basse (0-100% largeur graphique, 100%=par déf.) %		
ZoneB_HI	Zone B val. haute (0-100% largeur graphique, 100%=par déf.) %		
ZoneB_LO	Zone B val. basse (0-100% largeur graphique, 100%=par déf.) %		
ZoneB_On	VRAI sélectionne zone B, FAUX A (par déf.)	V/F	☐▶

Table 17-4 suite...

...Table 17-4 suite



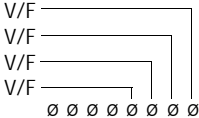
Paramètre	Fonction	Unités	Etat
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
DIGITAL1	Alarme logique utilisateur 1	V/F	
DIGITAL2	Alarme logique utilisateur 2	V/F	
DIGITAL3	Alarme logique utilisateur 3	V/F	
DIGITAL4	Alarme logique utilisateur 4	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
AlmAct	Alarmes logiques utilisateur actives		
Alm_1	VRAI = Alarme DIGITAL1	V/F	
Alm_2	VRAI = Alarme DIGITAL2	V/F	
Alm_3	VRAI = Alarme DIGITAL3	V/F	
Alm_4	VRAI = Alarme DIGITAL4	V/F	

Table 17-4 Paramètres du bloc DR_DGCHP

Menu de spécifications du bloc

Les informations suivantes complètent celle de la table 17-4.

Quality. Ce paramètre correspond à l'énumération des états PV utilisée dans les enregistreurs Eurotherm. Vous pouvez rattacher toute valeur logique appropriée au champ *Quality*.

ColourA. (Default/Violet/Blue/Green/Brown/Red/Black/DkViolet/DkBlue/DkGreen/DkBrown/DkRed/DkGrey/LiViolet/LiBlue/LiGreen/LiBrown/LiRed/LiGrey) La couleur à utiliser pour la trace. Les préfixes 'Dk' et 'Li' représentent les versions sombres et claires des couleurs de base, et gris sombre et gris clair. Si vous spécifiez *Default*, la couleur est déterminée par le numéro de *Chann* du bloc dans le bloc GROUP.

ColourB. (Default/Violet/Blue/Green/Brown/Red/Black/DkViolet/DkBlue/DkGreen/DkBrown/DkRed/DkGrey/LiViolet/LiBlue/LiGreen/LiBrown/LiRed/LiGrey/FiViolet/FiBlue/FiGreen/FiBrown/FiRed/FiBlack) Comme pour *ColourA* ci-dessus, mais les versions clignotantes des couleurs de base et le noir sont également sélectionnables (représentées par le préfixe 'Fi').

ZoneA_LO, ZoneA_HI. La zone du "graphique" (perpendiculaire à la base de temps) occupée par une tendance est appelée *zone*. Spécifiée comme une paire de valeurs marquant le début et la fin de la zone, chacune dans une plage de 0% à 100%. *ZoneA_LO* et *ZoneA_HI* spécifient les valeurs haute et basse pour la zone A.

ZoneB_LO, ZoneB_HI. (*Spécifications de la Zone B non mises en oeuvre à l'heure actuelle*).

ZoneB_On. (*Spécifications de la Zone B non mises en oeuvre à l'heure actuelle*)

Alarms. Les alarmes *DIGITAL_1* à *DIGITAL_4* sont déclenchées par les entrées *Alm_1* à *Alm_4* dans le champ binaire *AlmAct*.





DR_ALARM: BLOC POINT ALARME

Fonction du bloc

Le bloc de fonction DR_ALARM permet de filtrer les alarmes au cours de l'enregistrement dans des fichiers de consignment et/ou de la sortie sur des imprimantes.

Paramètres du bloc

Les symboles de la table 17-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Disable	VRAI désactive le filtre, FAUX l'active (par déf.)	V/F	
FiltType	Définit les alarmes à inclure	(AB)CD hex	
AlmAct	Inclut alarme active (par déf. =VRAI)	V/F —1	D
AlmClr	Inclut alarme supprimée (par déf. =VRAI)	V/F —2	
AlmAck	Inclut acquittements alarme (par déf. =VRAI)	V/F —4	
(Rserve)	(par déf.=FAUX)	V/F —8	
EvtSys	Inclut événements système (par déf. =VRAI)	V/F —1	C
EvtBlk	Inclut événement bloc (par déf. =VRAI)	V/F —2	
EvtNote	Inclut note opérateur (par déf. =VRAI)	V/F —4	
(Réserve)	(Par défaut=FAUX)	V/F —8	
FiltBlk	Nom du bloc pour filtrer alarmes	Alphanumérique	
FiltPriA	Priorité alarme mini. à inclure (1 - 15; par déf. = 1)	Entier	
FiltPriE	Priorité évén. mini. à inclure (1 - 15; par déf. = 1)*	Entier	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

*Fixée à 1 à l'heure actuelle

Table 17-5 Paramètre du bloc DR_ALARM

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 17-5.

Disable. VRAI désactive le filtre, FAUX l'active (par déf.). Lorsque le filtre est désactivé, aucune alarme n'est filtrée.

FiltBlk. Nom du bloc pour filtrer les alarmes. Si le champ est laissé vierge, *toutes* les alarmes/événements sont filtrés. Si un bloc est désigné, seuls les alarmes/événements du bloc spécifié sont filtrés. Si le bloc désigné est un bloc GROUP, seuls les alarmes/événements dans le bloc GROUP ou dans un bloc associé à l'un de ses champs *Disp1-16* ou *Chan1-16* sont filtrés. Si le bloc désigné est un bloc AREA, seuls les alarmes/événements dans le bloc AREA ou dans l'un de ses blocs GROUP ou groupes sont filtrés.

NOTA. Les blocs AREA/GROUP spécifiés dans *FiltBlk* peuvent ou non être utilisés à des fins d'affichage.

DR_REPRT: BLOC POINT REPORT

Fonction du bloc

Le bloc de fonction DR_REPRT génère des rapports texte définis par l'utilisateur et les envoie sur des unités texte (*uniquement des imprimantes à l'heure actuelle*). Notez qu'un bloc DR_REPRT ne peut être partagé entre des blocs PGROUP ou LGROUP, autrement dit, on ne peut y renvoyer qu'une fois dans la base de données.

Paramètres du bloc

Les symboles de la table 17-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
FormFile	Nom fichier formulaire (extension .UYF)	Alphanumérique	
FormId	Id dans <i>FormFile</i> .UYF utilisé pour formul. rapport	Entier	
Trigger	Front montant génère rapport	V/F	
Triggerd	VRAI pend. dem. rapport accept., non terminée	V/F	
Active	VRAI pendant génération rapport	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Form	Erreur dans fichier formulaire <i>FormFile</i> .UYF	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
FormErr	Code erreur formulaire	Menu	
FormErLn	Ligne sur laquelle <i>FormErr</i> s'est produit	Entier	
FormErCh	Position caractère de l'erreur à la ligne <i>FormErLn</i>	Entier	

Table 17-6 Paramètres du bloc DR_REPRT

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 17-6.

FormFile. Un fichier de formulaire permet de définir la sortie textuelle. Le fichier .UYF définit une présentation de texte — voir les détails sur ces fichiers dans votre manuel produit. Ce formulaire est ensuite “remplit” à la demande pour générer la sortie textuelle demandée.

Triggerd. VRAI pendant que la demande de rapport a été acceptée, mais pas menée à bien. Un autre rapport ne peut être lancé (par l'intermédiaire de l'entrée *Trigger*) jusqu'à ce que *Triggerd* passe de nouveau à l'état FAUX. *Triggerd* repasse à l'état FAUX, lorsque le rapport a été généré — dans le même temps le champ *Active* passe à l'état FAUX.

FormErr. (OK /BAD_FILE /LINE_LEN /NEWLINE /MEMORY /SYNTAX /RANGE /NAME /DICTNRY /TYPE /ACTION /FORM /OTHER) Code d'erreur formulaire. La signification des options est la suivante:

- **OK.** Aucune erreur.
- **BAD_FILE.** Fichier introuvable ou ne peut être lu.

- **LINE_LEN.** Une ligne du fichier trop longue (limite 255, RC et/ou CL non compris).
- **NEWLINE.** Fichier ne se termine pas par une nouvelle ligne (probablement corrompu).
- **MEMORY.** Formulaire trop grand pour la mémoire affectée.
- **SYNTAX.** Erreur de syntaxe.
- **RANGE.** Une valeur numérique est en dehors de la plage admissible.
- **NAME.** Objet désigné introuvable (sans doute mal orthographié).
- **DICTNRY.** Entrée de dictionnaire introuvable.
- **TYPE.** Type d'objet inapproprié dans le contexte où il est utilisé.
- **ACTION.** (*Non mis en oeuvre*)
- **FORM.** Un construit a été utilisé, ce qui n'est pas valable dans un fichier de formulaire.
- **OTHER.** Autre erreur.

[Page laissée intentionnellement blanche]

Chapter 18 BLOCS DE FONCTION PROGRAMMER

Les blocs de fonction de la catégorie PROGRAMMER — SPP_CTRL, SPP_DIG, SPP_RAMP et SPP_EXT — sont destinés à être utilisés avec le programmeur de points de consigne multi-boucle du T2900.

- Le bloc SPP_CTRL (contrôle programmeur points de consigne) permet de contrôler le programme chargé, de planifier l'exécution du programme suivant et de contrôler l'automate fini du programmer.
- Le bloc SPP_DIG (contrôle logique) permet de créer des liaisons filaires sortantes pour les points de consigne logiques du programmeur de points de consigne.
- Le bloc SPP_RAMP (rampe locale) permet d'incrémenter/décrémenter en rampe locale les points de consigne du T2900.
- Le bloc SPP_EXT (action programmeur) permet un contrôle simple du programme sans séquences (SFC).

SPP_CTRL: BLOC CONTROLE PROGRAMMEUR CONSIGNES

Fonction du bloc

Le bloc SPP_CTRL (contrôle programmeur consignes) permet d'accéder aux fonctionnalités d'état et de programmation/planification du programmeur de consignes pour être utilisées par des écrans utilisateur, SFC, FBD et au niveau des communications. Ce bloc doit être exécuté si tous les aspects de la fonctionnalité programmeur sont requis.

Il dispose des mêmes fonctions que l'agent du panneau du programmeur, sauf qu'il ne peut générer des descriptions de texte intégral (autres que les noms à 8 caractères de la base de données LIN), en raison de la limitation à 8 caractères de la chaîne de caractères des bases de données LIN.

Paramètres du bloc

Les symboles de la table 18-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.




























Paramètre	Fonction	Unités	Etat
Id	N° ID du programme (toujours '1')	Entier	 
State	Etat actif du programme	Menu	 
Hold	Programme en HOLD (maintien)	V/F	 
Manual	Programmeur en mode manuel	V/F	 
NxtRdy	Programme suivant demandé prêt	V/F	
RqNxtPrg	Nom fichier demandé programme suivant à exécuter	Alphanumérique	
RqStrtDt	Date au plus tôt pour lancer programme suivant	Date	
RqStrtTm	Heure au plus tôt pour lancer programme suivant	Heure	
RqNumIt	Nbre itérations demandées (0-999; 0=continu)	Entier	
Alarms			  
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Holdback	Programme en retenue	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
CurrProg	Nom du fichier .UYS actif (programme consignes)	Alphanumérique	 
EndDate	Date de fin itération programme en cours d'exécution	Date	 
EndHeure	Heure de fin itération programme en cours d'exécution	Heure	 
NumIt	Nbre original d'itérations demandées (0-999; 0=continu)	Entier	 
ItRemain	Nbre itér. program. restantes, y comp. en cours (0-999; 0=cont.)	Entier	 
CurrSeg	Exécution segment n° (0=aucun segment actif)	Entier	 
SegHeure	Durée restant segment actif	Heure	 
ProgDur	Durée totale prévue itération programme	Heure	 

Table 18-1 Paramètres du bloc SPP_CTRL

Menu de spécification du bloc

Les informations suivantes complètent celles de la table 18-1.

Id. Numéro d'identification du programme à utiliser avec plusieurs programmeurs (*non mis en oeuvre à l'heure actuelle*). Prend la valeur par défaut de '1'.

State. (RUN/HELDBACK/RESET/IDLE/COMPLETE) Etat actif du programme.

Manual. VRAI signifie que le programme est en mode manuel. En manuel, le bloc SPP_CTRL ne permet pas d'utiliser les fonctions planification et contrôle.

NxtRdy. VRAI signifie que le programme suivant demandé est prêt. Vous devez mettre *NxtRdy* à FAUX lorsque vous modifiez les champs '*Rq...*' fields, puis le remettre à VRAI, lorsque tous les champs sont configurés correctement. *NxtRdy* et '*Rq...*' doivent être paramétrés en utilisant une séquence (SFC) ou une définition de démarrage à froid.

RqStrtDt. La date au plus tôt demandée, à laquelle lancer le programme suivant au format de date jj/mm/aa. Saisissez '??/??/??' dans ce champ signifie 'aujourd'hui'.

NOTA. Toute entrée dans ce champ est remplacée si '??:??:??' a été saisi dans le champ *RqStrtTm* (voir suivant).

RqStrtTm. L'heure du jour au plus tôt demandée pour lancer le programme suivant au format d'heure hh:mm:ss. Saisissez '??:??:??' dans ce champ signifie "commencement immédiat, en ignorant l'entrée du champ *RqStrtDt*" (voir précédent).

RqNumIt. Le nombre d'itérations demandé (0 = 'continu').

NOTA. L'utilisation principale des champs *NxtRdy*, *RqStrtDt*, *RqStrtTm* et *RqNumIt* est de forcer l'exécution d'un programme ou un démarrage à froid.

EndDate. Date à laquelle l'itération du programme en cours d'exécution prendra fin. *EndDate* ne change que si le programme passe en HOLD ou HOLDBACK (maintien ou retenue).

EndHeure. Heure du jour à laquelle l'itération du programme en cours d'exécution prendra fin. *EndTime* ne change que si le programme passe en HOLD ou HOLDBACK (maintien ou retenue).

ProgDur. Durée totale prévue de l'itération du programme. *ProgDur* ne change pas, une fois que le programme a été lancé.

SPP_DIG: BLOC LOGIQUE PROGRAMMEUR DE POINTS DE CONSIGNE

Fonction du bloc

Le bloc de fonction SPP_DIG regroupe les points de consigne logiques générés par le programme de points de consigne exécutés dans le T2900, et fournit les sorties logiques correspondantes qui peuvent être rattachées aux instruments cibles déportés par l'intermédiaire de blocs du module de contrôle dévolu. Les sorties logiques peuvent également être utilisées, par ex. en les reliant à des blocs DR_DGCHP pour des enregistrements en mémoire flash dans le T2900.

La figure 18-1 montre un exemple de bloc SPP_DIG relié à d'autres blocs dans un T2900. Cette architecture permet de contrôler un point de consigne déporté "Ventilateur activé" sur le réseau (par l'intermédiaire du bloc D25_D04), ainsi que l'enregistrement local du point de consigne (par l'intermédiaire du bloc DR_DGCHP). Notez que le bloc DR_DGCHP doit faire partie d'une hiérarchie 'Group/Area' pour que l'enregistrement ait lieu — non représentée sur la figure. (Voir Chapitre 17, *Blocs de fonction Recorder*).

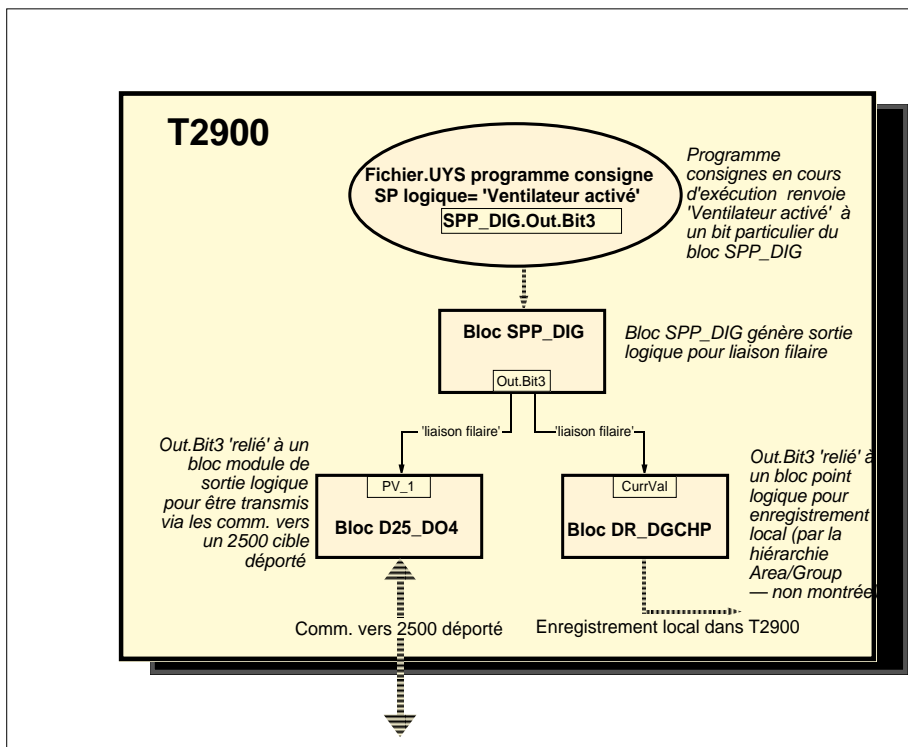


Figure 18-1 Utilisation du bloc SPP_DIG pour transmettre des consignes logiques — exemple

NOTA. Il n’y a pas de bloc équivalent pour traiter les points de consigne analogiques générés par un programme de points de consigne. Ceux-ci sont transmis directement à leurs cibles par l’intermédiaire des blocs D25_RAMP (module de contrôle dévolu). Voir les détails sur l’utilisation de ce bloc au chapitre 15, *Blocs de fonction DCM*.

Paramètres du bloc

Les symboles de la table 18-2 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Out	Valeurs consignes logiques	ABCD hex	
Bit0	VRAI = Consigne logique affectée 'ON'	V/F — 1	D
Bit1		V/F — 2	
Bit2		V/F — 4	
Bit3		V/F — 8	
Bit4		V/F — 1	C
Bit5		V/F — 2	
Bit6		V/F — 4	
Bit7		V/F — 8	
Bit8		V/F — 1	B
Bit9		V/F — 2	
Bit10		V/F — 4	
Bit11		V/F — 8	
Bit12		V/F — 1	A
Bit13		V/F — 2	
Bit14		V/F — 4	
Bit15		V/F — 8	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 18-2 Paramètres du bloc SPP_DIG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 18-2.

Out. Les bits dans ce champ sont associés aux points de consigne logiques générés par le programme des points de consigne. Ceci peut être réalisé pour chaque point de consigne, en saisissant le bit *Out* du bloc SPP_DIG dans le champ *Hardware reference* de la boîte de dialogue *Setpoint Properties* de l’éditeur du programme des points de consigne — en utilisant le format **Blockname.Out.Bitn**. (Voir les détails dans *Setpoint Program Editor Handbook*, réf. HA 261 134 U005).

SPP_RAMP: BLOC RAMPE LOCALE PROGRAMMEUR CONSIGNES

Fonction du bloc

Le bloc SPP_RAMP duplique la fonctionnalité disponible dans les unités du système E/S; en permettant d'incrémenter/décroémenter localement la rampe d'un point de consigne local (dans le T2900) comme alternative à l'exécution de la rampe dans le 2500 déporté.

Notez qu'en dehors des champs marqués d'un "*" dans la table 18-3, ce bloc se comporte exactement comme les blocs D25_RAMP et D25eRAMP utilisés dans un 2500.

Paramètres du bloc

Les symboles de la table 18-3 sont expliqués au chapitre 1.

Paramètre	Fonction	Unités	Etat
ResetSP*	Consigne utilisée en 'réinit', càd si RemEnabl FAUX	Eng	
HL_SP*	Limite supérieure de SP (dans toutes les formes)	Eng	
LL_SP*	Limite inférieure de SP (dans toutes les formes)	Eng	
Track*	Poursuite PV	V/F	
RampDis*	Rampe désactivée	V/F	
PV	Variable procédé	Eng	
TgtSP	Consigne cible	Eng	
Rate	Vitesse de variation demandée	Eng	
RampUnit	Unités de la rampe (SEC/MIN/HEURE)	Menu	
Out	Valeur consigne active	Eng	
NewTgtSP	Consigne cible suivante	Eng	
NewRate	Vitesse segment suivant	Eng	
Sync	Déclenche chargement valeurs NewTgtSP et NewRate	V/F	
SyncPV	Asservi à PV au début de la rampe	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Holdback	Rampe en retenue	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Complete	Rampe terminée	V/F	
Active	Rampe en cours	V/F	
NotActiv	Rampe temps pas active	Heure	
Holdback	Rampe retenue	V/F	
Hold	Rampe en HOLD (maintien)	V/F	
HB_Mode	Type de retenue (NONE/LOW/HIGH/DEV)	Menu	
HB_Value	Valeur de retenue	Eng	
HB_Dis	Retenue désactivée	V/F	
MinHback ^[N]	<i>(Fonction réservée)</i>	Eng	
OorHback ^[N]	Rampe en retenue & 'déf. vitesse' <i>(Non mise en oeuvre)</i>	V/F	
RemEnabl	Mis à 1 par SPP début programme, réinit. à la fin	V/F	
StLocRmt	Définit Local/Déporté au démarrage	Menu	
StWspCh	Définit WSP au démarrage	Menu	
StHold	Définit Hold au démarrage	Menu	
SP_Servo*	VRAI = SP asservi. (réinitialisable par utilisateur)	V/F	

^[N] Aucune communication mise en oeuvre avec ce champ à l'heure actuelle.

Table 18-3 Paramètres du bloc SPP_RAMP

SPP_EXT: BLOC EXTENSION PROGRAMMEUR CONSIGNES

Fonction du bloc

Le bloc SPP_EXT est un bloc optionnel qui peut être utilisé pour assurer un contrôle supplémentaire sur un programme en cours d'exécution ou pour exécuter des actions en des points donnés dans l'exécution du programmeur. La même fonctionnalité peut être réalisée, en utilisant une séquence (SFC), mais pour des applications qui ne nécessitent pas d'autres séquences, ce bloc représente une solution plus simple.

Le bloc fournit comme entrées un certain nombre de conditions de programmeur: le programme en cours d'exécution (*CurrProg*), une gamme de références de segments (*FirstSeg* à *LastSeg*), un ensemble d'états (*Inhibit*) qui peuvent être sélectionnés individuellement pour inhiber l'action demandée du programmeur, et un stimulus externe (*Trigger*). Si toutes les conditions sont satisfaites et que l'entrée *Trigger* est déclenchée, alors la sortie *Triggerd* est mise à 1 et l'action demandée du programmeur — *Action* ou *GotoSeg* — est exécutée.

Paramètres du bloc

Les symboles de la table 18-4 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section *Menu de spécifications du bloc* ci-après.

Paramètre	Fonction	Unités	Etat
Id	N° ID du programme	Entier	
CurrProg	Nom fichier .SPP sur lequel agir	Alphanumérique	
FirstSeg	Réf. 1er segment qui doit correspondre	Entier	
LastSeg	Réf. der. seg. qui doit correspondre (0=seg. final)	Entier	
Inhibit	Etat inhibant action programmeur	(AB)CD hex	
RUNNING	} VRAI inhibe action si cet état est actif	V/F — 1	D
HOLD		V/F — 2	
HELD BACK		V/F — 4	
COMPLETE		V/F — 8	
ERROR	} Réserve	V/F — 1	C
Réserve		V/F — 2	
Réserve		V/F — 4	
Réserve		V/F — 8	
Trigger	VRAI = condition externe satisfaite	V/F	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Config	Id erronée	V/F	
Segment	Segment GOTO non valable AND <i>Triggerd</i> = VRAI	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Triggerd	VRAI = conditions action satisfaites	V/F	
Action	Action programme exécutée si <i>Triggerd</i> = VRAI	Menu	
GoToSeg	Réf. segment à atteindre si <i>Action</i> = GOTO	Entier	
NumIt	(Non utilisé)	Entier	

Table 18-4 Paramètres du bloc SPP_EXT

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 18-4.

Id. Numéro d'identification du programme à utiliser avec plusieurs programmeurs.

Prend la valeur par défaut de '1'. *Id* doit correspondre à la valeur *Id* dans le bloc SPP_CTRL approprié.

CurrProg. Nom du fichier .SPP (sans l'extension) sur lequel l'action doit être exécutée. Laisser ce champ vierge (chaîne de caractères nulle), signifie "n'importe quel" programme.

FirstSeg, LastSeg. Ces champs définissent le bloc de références de segments de programmes consécutifs, sur lequel l'action doit être exécutée. Les valeurs sont les suivantes:

- 0, 0 signifie tous les segments
- *n*, *n* signifie uniquement le segment *n*
- *m*, *n* signifie les segments *m* à *n* inclus
- 0, *n* signifie tous les segments jusqu'à et y compris *n*
- *n*, 0 signifie tous les segments de *n* jusqu'à la fin du programme.

Alarms. L'alarme *Segment* est déclenchée si la référence du segment spécifiée dans *GoToSeg* n'est pas valable, lorsque les autres conditions d'entrée sont satisfaites (c'est à dire lorsque *Triggerd* est VRAI).

Triggerd. Ce bit ne passe à l'état VRAI que lorsque toutes les conditions du programmeur sont satisfaites, et l'action est alors exécutée. *Triggerd* n'est exécuté que si:

- Le nom du programme correspond à celui spécifié dans *CurrProg*
- Le segment en cours d'exécution est inclus dans le bloc spécifié par *FirstSeg* et *LastSeg*.
- Les états spécifiés comme étant VRAI dans le paramètre *Inhibit* ne sont pas actifs.
- L'entrée *Trigger* est VRAIE.

Action. (NONE/HOLD/ABORT/SKIP/GOTO) Action à exécuter sur le programme, lorsque les conditions sont satisfaites, c'est à dire lorsque *Triggerd* = VRAI. Lorsque HOLD est sélectionné, l'exécution du programme est reprise, lorsque *Triggerd* repasse à l'état FAUX. Si GOTO est spécifié, le programme passe à la référence du segment spécifié par *GoToSeg*.

Chapitre 19 BLOCS DE FONCTION CONVERT

Les blocs de la catégorie CONVERT permettent de convertir des types de champs de base de données LIN dissemblables, en particulier des valeurs énumérées. Autrement dit, ces types de champs peuvent être reliés entre eux dans un schéma de boucles par l'intermédiaire de blocs CONVERT.

ENUMENUM: BLOC CONVERTISSEUR ENUMERATIF A ENUMERATIF

Fonction du bloc

Le bloc ENUMENUM convertit une valeur énumérée d'une énumération à l'autre. Le bloc nécessite que la source et la destination soient reliées.

Un champ énuméré comprend une chaîne de caractères associée à un nombre. Les champs énumérés dans différents types de blocs LIN ayant les mêmes chaînes de caractères peuvent être associés à différents ensembles de nombres. Le bloc ENUMENUM lit la chaîne de caractères associée à la valeur numérique de l'énumération du bloc source (connectée aux champs *SrcEnum1* à 8), en recherchant une correspondance exacte de la chaîne de caractères dans l'énumération du bloc de destination (connectée aux champs *DstEnum1* à 8), et sort ensuite la valeur numérique associée trouvée dans ce bloc dans le bloc de destination. Des alarmes (*Wire1* à 8) sont déclenchées si une connexion n'est pas établie ou si aucune correspondance n'est trouvée.

NOTA. Ce bloc ne doit pas être utilisé pour des liaisons en éventail à différentes énumérations, il ne risque de ne pas fonctionner comme prévu dans ces circonstances. Chaque liaison filaire sortante ne doit être reliée qu'à des instances d'un seul type de bloc.

Paramètres du bloc

Les symboles de la table 19-1 sont expliqués au chapitre 1.





Paramètre	Fonction	Unités	Etat
SrcEnum1 à 8	Énumération source (valeurs 0 à 20)	ENUM	
DstEnum1 à 8	Énumération destination (valeurs 0 à 20)	ENUM	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Wire1 à 8	VRAI si auc. correspond. texte ou connex. manquante	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 19-1 Paramètres du bloc ENUMENUM

UINTENUM: BLOC CONVERTISSEUR ENTIER A ENUMERATIF

Fonction du bloc

Le bloc UINTENUM convertit en entier en une énumération. Le bloc nécessite que l'énumération de destination soit reliée.

Un champ énuméré comprend une chaîne de caractères associée à un nombre. Le bloc UINTENUM lit l'entier source (entré dans les champs *SrcUint1* à 8), et écrit ce nombre dans l'énumération du bloc de destination (par l'intermédiaire des champs de sortie *DstEnum1* à 8), sélectionnant ainsi la chaîne de caractères associée (le cas échéant) dans le bloc de destination. Les alarmes (*Wire1* à 8) sont déclenchées si une connexion de destination n'est pas établie ou si l'entier source ne correspond à aucune valeur d'énumération admissible dans le bloc de destination.

Paramètres du bloc

Les symboles de la table 19-2 sont expliqués au chapitre 1.




Paramètre	Fonction	Unités	Etat
SrcUint1 à 8	Entier source (valeurs 0 à 20)	UINT	
DstEnum1 à 8	Enumération de destination (valeurs 0 à 20)	ENUM	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Wire1 à 8	VRAl si entier hors échelle ou destination non reliée	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 19-2 Paramètres du bloc UINTENUM

ENUMUINT: BLOC CONVERTISSEUR ENUMERATIF A ENTIER

Fonction du bloc

Le bloc ENUMUINT convertit une énumération en entier. Le bloc nécessite que l'énumération source soit relié.

Un champ énuméré comprend une chaîne de caractères associée à un nombre. Le bloc ENUMUINT lit la valeur numérique de l'énumération du bloc source (entrée par les champs *SrcEnum1* à 8), et écrit ce nombre dans l'entier de destination (par l'intermédiaire des champs de sortie *DstUint1* à 8). Les alarmes (*Wire1* à 8) sont déclenchées si une connexion de destination n'est pas établie ou si l'entier source ne correspond à aucune des valeurs d'énumération permmissibles dans le bloc de destination.

Paramètres du bloc

Les symboles de la table 19-3 sont expliqués au chapitre 1.









Paramètre	Fonction	Unités	Etat
SrcEnum1 à 8	Énumération source (valeurs 0 à 20)	ENUM	       
DstUint1 à 8	Entier destination (valeurs 0 à 20)	UINT	
Alarms			
Software	Anom. mém. données RAM bloc/déf. réseau	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	

Table 19-3 Paramètres du bloc ENUMUINT

[Page laissée intentionnellement blanche]

Chapitre 20 BLOCS DE FONCTION OPERATEUR

Les cinq blocs dans la catégorie OPERATEUR permettent de contrôler les activités liées à l'interface opérateur dans le ou les instruments pris en charge — par ex. Visual Supervisor.

- Le bloc PNL_CMD (commande panneau) permet la prise en main du panneau, force, par exemple, un saut vers une page spécifique.
- Le bloc PNL_MSG (message panneau) permet de créer un message.
- Le bloc PNL_DLG (boîte de dialogue panneau) permet de créer une boîte de dialogue.
- Le bloc PNL_ACC (accès panneau) valide et permet de contrôler les ouvertures de session des utilisateurs.
- Le bloc **READER** permet d'utiliser des lecteurs de codes à barres et dispositifs similaires.

PNL_CMD: BLOC DE COMMANDE DE PANNEAU

Fonction du bloc

Le bloc PNL_CMD permet en général de forcer le panneau du Visual Supervisor à sauter à une page spécifique. Il peut s'agir d'une *page utilisateur* (Id 1-999) ou d'un *agent* intégré (Id > 999).

La paramètre *Action* du bloc comprend les options suivantes:

- **GOTO.** Pour aller à la page spécifiée.
- **DESCEND.** 'Descend' jusqu'à la page spécifiée, avec une touche 'échappement' pour revenir à la page d'origine.
- **ESCAPE.** Annule une commande 'descend' ou supprime une fenêtre en incrustation. (Equivalent à la touche "page vers le haut").

D'autres actions sont également disponibles (voir le paragraphe *Menu de spécifications du bloc* ci-dessous) pour créer, par exemple, des démonstrations roulantes.

Exécution d'une commande interface utilisateur

Une séquence particulière de changements de valeurs des paramètres doit être suivie pour une interaction d'interface utilisateur réussie. Les changements requis dans le bloc PNL_CMD en question peuvent être effectués en utilisant une séquence (SFC), une liste d'actions utilisateur ou par des entrées filaires du schéma de boucles en cours d'exécution.

Un changement de page unique type, en s'assurant que la nouvelle page est atteinte, peut être réalisé comme suit:

- 1 Sélectionnez VRAI pour **Option.Lock** pour supprimer la navigation commandée directement par le panneau lui-même, ainsi que les temporisations.
- 2 Sélectionnez VRAI pour **Acquire** pour lancer l'acquisition de commande de l'interface utilisateur.
- 3 Attendez que **Acquired** passe à l'état VRAI. Ceci confirme qu'aucun autre bloc de fonction PNL_CMD ne commande déjà l'interface.

NOTA. Vous pouvez également rendre l'acquisition conditionnelle par rapport au niveau d'accès et/ou du statut de l'utilisateur par l'intermédiaire des sous-champs *Access*.

- 4 Spécifiez *GoTo* pour **Action** et sélectionnez la valeur appropriée pour **Id** et afficher la page voulue.
- 5 Faites passer **Trigger** de l'état FAUX à VRAI pour déclencher l'action sélectionnée.
- 6 Attendez que **Triggerd** passe à 1, et vérifiez (éventuellement) qu'il n'y a pas d'alarmes.
- 7 Remettez **Acquire** et **Trigger** à l'état initial pour abandonner le contrôle et préparer d'autres déclenchements.

Dans les applications à un seul bloc PNL_CMD et sans verrouillage, il est plus simple de laisser le contrôle en permanence au bloc (*Acquire* maintenu à VRAI), de préconfigurer *Action* et *Id*, et de faire passer *Trigger* à VRAI, puis à FAUX, à chaque fois qu'une action est requise. Vous pouvez également laisser *Trigger* à l'état VRAI et faire en sorte que *Acquire* déclenche l'action.

Paramètres du bloc

Les symboles utilisés dans la table 20-1 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PanelId	Id du panneau (<i>par défaut = 1. Non encore mis en oeuvre</i>)	Entier	
Acquire	VRAI acquiert le contrôle interface utilisateur, FAUX l'abandonne	V/F	
Trigger	VRAI exécute l'action sélectionnée (verrouillage)	V/F	
Action	Sélectionne l'action en question	Menu	
Id	Id de la page ou agent de destination	Entier	
Context	Context du bloc, le cas échéant	Nom bloc	
DispMode	Mode affichage, le cas échéant (0, 1-255)	Entier	
Access	Statut accès/utilisateur	(AB)CD hex	
LOCKED	VRAI active l'acquisition en niveau d'accès LOCKED	V/F — 1	D
OPERATOR	VRAI active l'acquisition en niveau d'accès OPERATOR	V/F — 2	
COMMISSION	VRAI active l'acquisition en niveau d'accès COMMISSION	V/F — 4	
ENGINEER	VRAI active l'acquisition en niveau d'accès ENGINEER	V/F — 8	
ADMIN	VRAI active l'acquisition en niveau d'accès ADMIN	V/F — 1	C
UserRef	VRAI active l'acquisition <i>UserRef</i> = réf utilisateur actif	V/F — 2	
		V/F — 4	
		V/F — 8	
UserRef	Limitation référence utilisateur en option (0-9999)	Entier	
Options	Options de navigation	(ABC)D hex	
Lock	Supprime navigation commandée par panneau & temporisations	V/F — 1	D
		V/F — 2	
		V/F — 4	
		V/F — 8	
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
BadId	Id nouvelle page erronée	V/F	
Access	Niveau accès actuel insuffisant pour nouvelle page	V/F	
Lock	Erreur changement page, car page active verrouillée	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Acquired	VRAI = contrôle acquis; FAUX = abandonné	V/F	
Triggerd	VRAI = action terminée, FAUX = <i>Trigger</i> FAUX	V/F	
StatusId	Id active fenêtre état	Entier	
MainId	Id active fenêtre principale	Entier	
PopUpId	Id active fenêtre en incrustation	Entier	

Table 20-1 Paramètre du bloc PNL_CMD

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 20-1.

Acquire. Une entrée VRAIE lance l'acquisition de commande de l'interface utilisateur. Après la mise à 1 de l'entrée *Acquire*, il y a une légère temporisation. Ensuite, à condition qu'aucun autre bloc PNL_CMD n'ait pris le contrôle, la sortie *Acquired* passe à 1 pour confirmer que l'acquisition s'est déroulée normalement. Ceci permet de s'assurer que seul un bloc PNL_CMD block a le contrôle à un moment donné. *Acquire* est verrouillé à VRAI, et doit ensuite être remis à l'état FAUX pour abandonner le contrôle. Dans ce cas, *Acquired* repasse également à l'état FAUX.

NOTA. Une action en attente risque d'être perdue si *Triggerd* n'est pas encore passé à l'état VRAI, lorsque *Acquire* est remis à l'état FAUX.

Trigger. Les changements de page ou d'autres actions sont déclenchées, lorsque l'entrée *Trigger* passe à l'état VRAI, à condition que *Acquired* soit également VRAI — autrement dit, le bloc PNL_CMD en question contrôle l'interface utilisateur. Après le déclenchement, il y a une légère temporisation (suivant l'activité de panneau en cours) avant que l'action requise ne soit exécutée, ensuite la sortie *Triggerd* passe à l'état VRAI pour confirmer que le déclenchement s'est déroulé normalement. *Trigger* est verrouillé à VRAI, et doit ensuite être remis à l'état FAUX par une entrée basse, avant de pouvoir être réutilisé pour déclencher une action.

Il peut être quelquefois plus pratique de déclencher une action en maintenant *Trigger* à l'état VRAI, lorsque le bloc PNL_CMD block n'assure pas le contrôle. Ensuite, le passage de *Acquire* à l'état VRAI agit comme déclencheur — à condition qu'aucun autre bloc PNL_CMD n'assure déjà le contrôle.

NOTA. Sous certaines conditions, une alarme peut être activée, lorsque *Triggerd* passe à 1. Ceci se produit toujours si une commande GOTO ou DESCEND n'aboutit pas pour une raison ou autre. Mais, cela peut également se produire si une action demandée n'est pas pertinente dans le contexte actif. Dans ce cas, l'action est tout simplement ignorée — de la même manière qu'un appui sur une touche sur le panneau de l'instrument proprement dit est ignoré.

Action. (GOTO/DESCEND/ESCAPE/CYCLE/NEXT/PREV/HOME/ROOT)
Sélectionne l'action à exécuter, lorsque [*Trigger* ET *Acquired*] passent à l'état VRAI.

- **GOTO.** Affiche la page dont l'*Id* a été spécifiée.
- **DESCEND.** 'Descend' à la page spécifiée, avec une touche "échappement" pour revenir à la page originelle.
- **ESCAPE.** Annule une action 'descend' ou supprime une fenêtre en incrustation. (Equivalent à la touche "page précédente").
- **CYCLE.** Equivalent à la touche "défilement écrans".
- **NEXT.** Equivalent à la touche "page vers la droite", lorsque cela entraîne un changement de page.
- **PREV.** Equivalent à la touche "page vers la gauche", lorsque cela entraîne un changement de page.
- **HOME.** Affiche la page d'accueil.
- **ROOT.** Affiche le menu racine. Equivalent à la touche (menu) "racine".

Id. Id (numéro d'identité) de la page ou de l'agent de destination. Les pages utilisateur — que vous créez avec l'éditeur d'écrans utilisateur — ont une *Id* dans la plage de 1-999; les agents préconfigurés "intégrés" ont une *Id* >999. Pour les agents intégrés, il y a certains cas où des informations contextuelles (ex.: nom du bloc de fonction, mode d'affichage) peuvent s'avérer nécessaires pour définir parfaitement la page cible. (Voir les paramètres *Context* et *DispMode*, ci-dessous.)

Context. Contexte du bloc (le cas échéant). Ce champ de nom de bloc permet de mieux définir la page de destination, en plus de l'*Id* de l'écran sélectionné.

La table 20-2 donne la liste des éléments de la zone/groupe/agent point de Visual Supervisor area/group/point agent auxquels le paramètre *Context* peut être appliqué en général, ainsi que les types de blocs de fonctions disposant d'un contexte.

Id	Agent	Valeur du paramètre Context
1500	Vue système	(Sans objet)
1501	Vue zone	Nom de bloc AREA (Sans objet à l'heure actuelle)
1502	Vue groupe*	Nom de bloc GROUP
1503	Vue	Point à afficher

**DispMode* également applicable

Table 20-2 Valeurs du paramètre Context pour les agents zone/groupe/point

Le paramètre *Context* peut également être appliqué aux agents messages opérateur pour contrôler quels messages doivent être intégrés dans les affichages. La table 20-3 donne la liste des valeurs *Id* et les formats de message correspondants, et montre le fonctionnement de *Context*.

Id	Format d'affichage de l'agent messages opérateur
9020	Format de liste ordonnée. Ne liste que les messages définis par <i>Context</i>
9021	Format de liste ordonnée. Liste tous les messages (<i>Context</i> sans objet)
9022	Format de tableau. N'affiche que les messages définis par <i>Context</i>
9023	Format de tableau. Affiche tous les messages (<i>Context</i> sans objet)

Table 20-3 Valeurs Id et formats des agents de messages opérateur

■ Exemple. Si l'*Id* = **9022** et *Context* = **FIC-010** (nom du bloc), la page de destination n'affichera sous forme de tableau que les messages opérateur associés au bloc de fonction LIN appelé 'FIC-010'. Si le bloc spécifié par *Context* est un bloc AREA ou GROUP, les messages associés à tous les blocs inclus dans la zone/groupe sont affichés.

DispMode. Mode d'affichage requis (le cas échéant) pour l'*Id* et le *Context* spécifiés. Les modes d'affichages correspondent aux différentes vues d'un même ensemble de données, par ex. les différents formats de tendance de Visual Supervisor.

La table 20-4 donne la liste de codes de modes d'affichage disponibles de Visual Supervisor utilisés à l'heure actuelle, ainsi que leur signification. (*DispMode* peut prendre des valeurs de 1 à 255. '0' indique 'vue par défaut'.)

NOTA. A l'heure actuelle, le paramètre *DispMode* ne s'applique qu'à la page *Group View* de Visual Supervisor, dont l'*Id* par défaut est de 1502.

DispMode	Affichage
0	Vue par défaut
1	Synoptique
2	Face avant
3	Numérique
4	Bargraphes verticaux
5	Bargraphes horizontaux
6	Tendance verticale avec faces avant
7	Tendance verticale sans faces avant
8	Tendance horizontale avec faces avant
9	Tendance horizontale sans faces avant

Table 20-4 Valeurs du paramètre DispMode

- **Exemple.** Si *Id* = **1502** (page vue groupe), *Context* = **preplot** (nom de bloc catégorie GROUP) et *DispMode* = **8**, la page de destination affiche une tendance horizontale avec des faces avant pour les points dans le groupe défini par le bloc GROUP appelé 'preplot'.

Access. Champ binaire contrôlant l'accès utilisateur pour l'acquisition de l'interface utilisateur. Les informations suivantes complètent celles de la table 20-1.

- **UserRef.** Si VRAI, l'acquisition n'est activée que si la valeur du champ *UserRef* est égale à 'User Reference' de l'utilisateur qui a ouvert une session. Cette condition doit être satisfaite, **en plus** de toute limitation de niveau d'accès imposée par les autres sous-champs *Access*.

NOTA. Le Visual Supervisor dispose de deux systèmes d'accès sécurisés — *Standard Access* et *User ID Access*. Dans le système *User ID Access*, un code 'User Reference' (non unique) est affecté à chaque utilisateur. C'est le code auquel on renvoie ici.

Options. Champ binaire contrôlant les options de navigation de l'interface utilisateur. Les informations suivantes complètent celles de la table 20-1.

- **Lock.** Si VRAI, lorsque *Acquire* est également VRAI, *Lock* supprime la navigation commandée directement par le panneau proprement dit, par ex. par l'intermédiaire du menu racine. Il supprime également les temporisations de panneau, par ex. la temporisation pour afficher la page d'accueil. Il peut y avoir un délai supplémentaire dans l'acquisition dans ce cas, parce qu'un certain nombre de pages intégrées utilisent également le verrouillage (par ex. au cours du clonage ou d'une mise à jour).

NOTA. Cette option doit être utilisée avec beaucoup de prudence pour éviter de verrouiller l'interface utilisateur de manière irrécupérable! Notez également que *Option.Lock* ne s'applique qu'au moment de l'acquisition — un basculement ultérieur n'a aucun effet.

PNL_MSG: BLOC MESSAGES DE PANNEAU

Fonction du bloc

Le bloc PNL_MSG permet de configurer et de mettre en file d'attente des messages que l'application peut soumettre à l'opérateur pour les visualiser sur la face avant de Visual Supervisor. Ces messages permettent, par exemple, de solliciter une réaction particulière de l'opérateur ou simplement de l'informer qu'un élément ne nécessite qu'un acquittement. Le schéma de boucles proprement dit peut être configuré pour répondre à des messages en l'absence de réaction de l'opérateur. Le bloc permet au schéma de boucles d'enregistrer — et, si nécessaire, d'agir sur — toutes les réactions opérateur.

Configuration et mise en file d'attente des messages. Les messages sont configurés par l'intermédiaire du champ *Title* du bloc, qui définit le bandeau du message, et du champ *Body*, qui définit le contenu du message, tout deux en tant qu'entrées de dictionnaire.

NOTA. Tous les messages texte pour ce bloc sont consignés dans un *dictionnaire de messages* spécial ('M'). Une valeur d'entrée de dictionnaire de '0' est interprétée comme "absence de texte". Ceci n'empêche pas des références à d'autres dictionnaires dans un "formulaire" associé.

Une entrée VRAIE pour *Trigger* ajoute le message configuré à la file d'attente des messages où il reste actif et disponible, afin que l'opérateur puisse le voir et y réagir, jusqu'à ce qu'il soit automatiquement supprimé, après qu'une temporisation en secondes (*Timeout*) se soit écoulée. (*Timeout* peut être désactivé pour permettre la mise en file d'attente indéfinie du message). Le message est immédiatement supprimé, si l'opérateur y répond en appuyant sur un "bouton" message ou si l'application elle-même y répond automatiquement.

Les messages sont mis en file d'attente (par défaut) dans l'ordre chronologique de création, le message le plus ancien étant le premier dans la file d'attente à être soumis à l'attention de l'opérateur. Le paramètre *Priority* permet cependant de modifier cet état de fait et de remonter un message dans la file d'attente.

Boutons opérateur. Un maximum de quatre "boutons" opérateur peuvent être affichés au bas du message. Ces boutons ('OK', 'CANCEL', 'YES', 'NO', etc) sont sélectionnés par l'intermédiaire du paramètre *Buttons*, plus quatre boutons de format libre définis par l'utilisateur avec des légendes définies dans un dictionnaire (pour les applications multilingues). Le paramètre *Response* enregistre le bouton sur lequel l'opérateur a appuyé ou si l'application a répondu automatiquement, le ou lesquels des boutons a ou ont été activé(s). Le paramètre *SelfResp* définit le ou lesquels des boutons sera(ont) activé(s) par l'application, lorsqu'une entrée VRAIE est déclenchée dans le champ *TrigSelf*.

Contexte des messages. Un "contexte" de bloc de fonction LIN peut être affecté à chaque message, en utilisant le paramètre *Context*. Ceci permet à l'opérateur de visualiser — et à l'application d'utiliser — sélectivement les messages en file d'attente, filtrés par contexte, comme alternative à un affichage global. Notez que d'autres blocs de fonction — le bloc PNL_CMD, par exemple — permettent également d'utiliser ce contexte pour filtrer les messages.

NOTA. Plusieurs blocs PNL_MSG peuvent être exécutés dans une application, chaque bloc permettant la prise en charge d'un message actif dans la file d'attente des messages à un moment donné.

Paramètres du bloc

Les symboles utilisés dans la table 20-5 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.






Paramètre	Fonction	Unités	Etat
PanelId	Id du panneau (par défaut = 1. Non encore mis en oeuvre)	Entier	
Trigger	VRAI ajoute le message à la file d'attente (verrouillage)	V/F	
Title	Entrée dictionnaire pour titre message (20 car. maxi.)	Entier	
Body	Entrée dictionnaire pour message texte	Entier	
FormFile	(Non encore mis en oeuvre)	Alphanumérique	
FormId	(Non encore mis en oeuvre)	Entier	
Options	Fonctions en option	(ABC)D hex	
Panel	VRAI (par déf.) active génération boîte message à l'écran	V/F — 1	D
Event	VRAI (par déf.) active consignation événement message/réaction	V/F — 2	
TimeStmp	VRAI inclut horodatage dans message affiché	V/F — 4	
		V/F — 8	
Buttons	Sélectionne les boutons à afficher dans le message, active tempo. (A)BCD hex		
OK	Bouton 'OK' (par défaut = VRAI)	V/F — 1	D
CANCEL	Bouton 'CANCEL' (Annuller)	V/F — 2	
YES	Bouton 'YES' (Oui)	V/F — 4	
NO	Bouton 'NO' (Non)	V/F — 8	
ABORT	Bouton 'ABORT' (Abandon)	V/F — 1	C
UButton1	Bouton utilisateur 1	V/F — 2	
UButton2	Bouton utilisateur 2	V/F — 4	
UButton3	Bouton utilisateur 3	V/F — 8	
	Légendes spécifiées dans UButton1-4		
UButton4	Bouton utilisateur 4	V/F — 1	B
Timeout	Activation temporisation (par défaut = VRAI)	V/F — 2	
		V/F — 4	
		V/F — 8	
UButton1 - 4	Légendes dictionnaire pour boutons utilisateur 1 à 4	Entier	
Timeout	Message chronol. reste file d'attente (sec); 0 = désact.	Entier	
Priority	Priorité ordre file attente message, 1 - 15. (Default = 1)	Entier	
Emphasis	Mise en évidence appliquée titre message (attirer attention)	Menu	
Context	Contexte du message pour le bloc de fonction (éventuel)	Nom bloc	
TrigSelf	Déclenche 'réponse auto' & supprime message (verrouillage).	V/F	
SelfResp	Définit réponse auto (tous boutons) si TrigSelf VRAI	(A)BCD hex	
OK	Bouton 'OK' (par défaut = VRAI)	V/F — 1	D
CANCEL	Bouton 'CANCEL' (Annuller)	V/F — 2	
YES	Bouton 'YES' (Oui)	V/F — 4	
NO	Bouton 'NO' (Non)	V/F — 8	
ABORT	Bouton 'ABORT' (Abandon)	V/F — 1	C
UButton1	Bouton utilisateur 1	V/F — 2	
UButton2	Bouton utilisateur 2	V/F — 4	
UButton3	Bouton utilisateur 3	V/F — 8	

Table 20-5 suite....

...Table 20-5 suite













Paramètre	Fonction	Unités	Etat
UButton4	Bouton utilisateur 4	V/F — 1	B
Timeout	Force temporisation message (si <i>TrigSelf</i> activé)	V/F — 2	
		V/F — 4	
		V/F — 8	
Alarms			  
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Nbuttons	Trop de boutons demandés (limite = 4)	V/F	
Dictnry	Erreur dictionnaire	V/F	
TextLen	Texte dans dictionnaire trop long pour contexte	V/F	
Form	Erreur fichier formulaire (voir <i>FormErr</i> , <i>FormErLn</i> , <i>FormErCh</i>)	V/F	
Failure	Echec mise en file d'attente message	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Triggerd	Acquittement déclencheur	V/F	 
Response	Signale boutons enfoncés opérateur ou bits réponse auto	(A)BCD hex	 
OK	Bouton 'OK' bouton 'enfoncé'	V/F — 1	D
CANCEL	Bouton 'CANCEL' 'enfoncé'	V/F — 2	
YES	Bouton 'YES' 'enfoncé'	V/F — 4	
NO	Bouton 'NO' 'enfoncé'	V/F — 8	
ABORT	Bouton 'ABORT' 'enfoncé'	V/F — 1	C
UButton1	Bouton utilisateur1 'enfoncé'	V/F — 2	
UButton2	Bouton utilisateur 2 'enfoncé'	V/F — 4	
UButton3	Bouton utilisateur 3 'enfoncé'	V/F — 8	
UButton4	Bouton utilisateur 4 'enfoncé'	V/F — 1	B
Timeout	Message supprimé après une temporisation	V/F — 2	
		V/F — 4	
		V/F — 8	
Status	Statut du message	(ABC)D hex	 
Active	VRAI = message actif, visualisé ou non	V/F — 1	D
OnView	VRAI = message actif & visualisé à l'écran	V/F — 2	
		V/F — 4	
		V/F — 8	
FormErr	Code erreur formulaire	Menu	
FormErLn	Ligne sur laquelle erreur signalée par <i>FormErr</i> s'est produite	Entier	
FormErCh	Position caractère de l'erreur sur la ligne signalée par <i>FormErLn</i>	Entier	

Table 20-5 Paramètre du bloc PNL_MSG

Menu de spécifications du bloc

Les informations suivantes complètent celle de la table 20-5.

Trigger. La mise à 1 de l'entrée *Trigger* génère un message (en fonction de vos spécifications) et l'ajoute à la file d'attente des messages actifs, qui attendent l'attention de l'opérateur. *Trigger* se verrouille et doit être remis à l'état FAUX avant d'être réutilisé.

Notez que le passage à l'état VRAI remet à zéro tous les bits dans le champ binaire *Response*.

Title. Entrée de dictionnaire pour le titre du message, 20 caractères maxi. Ceci s'affiche comme le "bandeau" du message et peut plus ou moins être mis en évidence pour attirer l'attention de l'opérateur (en utilisant le paramètre *Emphasis*).

Buttons. Sélectionne les boutons qui seront affichés au bas de la fenêtre de message (et active également la temporisation du message). Les informations suivantes complètent celles de la table 20-5.

■ **UButton1-4.** Ces bits permettent de sélectionner un maximum de quatre boutons utilisateur de format libre avec des légendes définies par les champs dictionnaire correspondants *UButton1* à *UButton4*.

■ **Timeout.** Mettez ce bit à l'état VRAI (par défaut) pour activer la temporisation de message, à savoir la période pendant laquelle le message non acquitté reste dans la file d'attente — définie par le champ *Timeout*. Si le bit *Timeout* est FAUX, la temporisation est désactivée et le message reste indéfiniment dans la file d'attente, jusqu'à l'acquiescement (par l'opérateur ou l'application).

NOTA. Ce bit permet de désactiver la temporisation sans supprimer la valeur de temporisation dans le paramètre *Timeout*.

Timeout. La durée (secondes) pendant laquelle le message reste dans la file d'attente pour que l'opérateur (ou l'application) puisse l'acquiescer. A l'écoulement de cette durée, le message est automatiquement supprimé. Notez que le message est immédiatement supprimé, si l'opérateur réagit en appuyant sur un "bouton" message (ou si l'application elle-même réagit au message). Pour désactiver la temporisation, mettez *Timeout* à zéro. Le message reste alors indéfiniment dans la file d'attente jusqu'à l'acquiescement.

Priority. Priorité dans l'ordre de la file d'attente des messages (1 - 15, par défaut = 1, la priorité la plus basse). Les messages sont normalement mis en file d'attente chronologiquement, le message le plus ancien étant le premier dans la file d'attente soumis à l'attention de l'opérateur. Le paramètre *Priority* permet cependant de modifier cet état de fait et de remonter un message plus récent dans la file d'attente, parce que la priorité prévaut sur l'ancienneté. Si les messages ont les mêmes priorités (cas par défaut), le message le plus ancien prévaut sur les autres.

Emphasis. (NONE/WEAK/STRONG/ATTN) Sélectionne la mise en évidence appliquée au titre du message pour attirer l'attention de l'opérateur.

■ **NONE.** Aucune mise en évidence (titre "vert Eurotherm").

■ **WEAK.** Légère mise en évidence (titre vert).

■ **STRONG.** Mise en évidence soutenue (titre rouge).

■ **ATTN.** Mise en évidence maximale pour attirer l'attention (titre rouge clignotant).

Context. Permet de définir un nom de repère de bloc de fonction LIN comme contexte du message — autrement dit, de l'associer au message. Un champ vierge indique "aucun contexte particulier". Ceci permet à l'opérateur de visualiser — et à l'application d'utiliser — sélectivement les messages en file d'attente, filtrés par contexte comme alternative à un affichage global. Notez que d'autres blocs de fonction — comme le bloc PNL_CMD — peuvent également utiliser ce contexte pour filtrer les messages.

TrigSelf. Cette entrée permet à l'application (schéma de boucles) de réagir aux boutons dans un message, au lieu de tabler sur les réactions de l'opérateur. En particulier, une entrée VRAIE pour *TrigSelf* copie le profil binaire du paramètre *SelfResp* sur le paramètre *Response*. Ceci revient à appuyer automatiquement sur tous les boutons définis dans *SelfResp*. Lorsque *TrigSelf* est mis à 1, le message est supprimé de la file d'attente (comme lorsqu'un opérateur y répond).

NOTA. *TrigSelf* se verrouille, et doit être remis à l'état FAUX, avant d'être réutilisé.

SelfResp. Définit la réponse automatique du message — autrement dit, le ou lesquels des boutons seront effectivement activés, lorsque l'application met l'entrée *TrigSelf* à l'état VRAI. En particulier, une entrée VRAIE pour *TrigSelf* copie le profil binaire du paramètre *SelfResp* sur le paramètre *Response*. Ceci revient à appuyer automatiquement sur tous les boutons définis dans *SelfResp*.

La réponse automatique peut être utilisée par l'application comme alternative à l'attente d'une réaction opérateur. Notez que, à l'inverse d'une réaction opérateur, une réponse automatique permet d'appuyer simultanément sur plusieurs boutons et également d'activer des boutons qui n'ont pas été validés par le paramètre *Buttons*, et n'apparaissent donc pas dans la boîte de message.

Triggerd. Cette sortie passe à l'état VRAI, après que *Trigger* a été mis à 1 et le nouveau message qui en résulte a été ajouté à la file d'attente. Il peut être utilisé comme protocole de transfert pour confirmer la création d'un message.

Response. Ce champ binaire en lecture seule enregistre les réactions aux boîtes de message opérateur (ou automatique). Le ou les bits correspondants passe(nt) à l'état VRAI, lorsque l'opérateur appuie sur un bouton ou lorsqu'une réponse automatique est déclenchée par une entrée dans *TrigSelf*. Les bits sont verrouillés à l'état VRAI, jusqu'à ce que l'entrée *Trigger* soit mise à 1, lorsque tous les bits sont remis à l'état FAUX (et un nouveau message est généré).

NON ENCORE MIS EN OEUVRE: Un formulaire peut être utilisé comme alternative pour définir le contenu du corps. Si un dictionnaire et un formulaire sont utilisés ensemble, alors ils sont présentés dans cet ordre, comme deux paragraphes séparés. Si un formulaire est utilisé, il est traité au moment où le message est déclenché, afin que toutes les variables soient affichées avec les valeurs qu'elles avaient à ce moment-là. Il s'en suit que le formulaire est internationalisé à ce moment-là. Notez que ceci a pour conséquence que si la langue est modifiée, lorsqu'un message est actif, alors lorsque ce message est visualisé ultérieurement, le texte de ce formulaire sera affiché dans la langue antérieure, alors que tout le reste est affiché dans la nouvelle langue. (Cette considération ne s'applique pas au cas des boîtes de dialogue, parce que celles-ci réquisitionnent l'interface utilisateur lorsqu'elle est active, ce qui empêche tout changement de langue).

NOTAS: Toutes les entrées de dictionnaire pour ce bloc (et pour le bloc PNL_DLG) sont prises dans un nouveau dictionnaire 'M', et dans tous les cas, une valeur d'entrée de dictionnaire de 0 est interprétée comme "absence de texte". (Ceci n'empêche pas bien sûr des références à d'autres dictionnaires dans un formulaire associé).

NON ENCORE MIS EN OEUVRE:

FormErr. (OK /BAD_FILE /LINE_LEN /NEWLINE /MEMORY /SYNTAX /RANGE /NAME /DICTNRY /TYPE /ACTION /CONTEXT/OTHER) Code d'erreur formulaire.

La signification des options est la suivante:

- **OK.** Aucune erreur.
- **BAD_FILE.** Fichier introuvable ou illisible.
- **LINE_LEN.** Une ligne dans le fichier était trop longue (la limite est 255, retour chariot et/ou interligne non compris).
- **NEWLINE.** Le fichier ne se termine pas par une nouvelle ligne (indique éventuellement un fichier corrompu).
- **MEMORY.** Formulaire trop gros pour la mémoire qui lui est affecté.
- **SYNTAX.** Il y a une erreur de syntaxe.
- **RANGE.** Une valeur numérique est en dehors de la plage admissible.
- **NAME.** Un objet désigné est introuvable (peut être mal orthographié).
- **DICTNRY.** Une entrée de dictionnaire est introuvable.
- **TYPE.** Un type d'objet est inapproprié dans le texte dans lequel il est utilisé.
- **ACTION.** (*Non mis en oeuvre*).
- **CONTEXT.** Une construction OIFL a été utilisée, ce qui est incorrect dans le contexte des formulaires.
- **OTHER.** Autre erreur.

PNL_DLG: BLOC DE DIALOGUE PANNEAU

Fonction du bloc

Les blocs PNL_DLG permettent de configurer et de mettre en file d'attente des boîtes de dialogue que l'application force l'opérateur à visualiser et auxquelles il doit réagir sur la face avant du Visual Supervisor. Ces boîtes de dialogue peuvent, par exemple, être utilisées pour solliciter une réaction particulière de l'opérateur ou pour simplement l'informer de quelque chose qui n'a besoin que d'un acquittement. Le schéma de boucles lui-même peut être configuré pour réagir à des boîtes de dialogue, en l'absence de réaction de la part de l'opérateur. Le bloc permet au schéma de boucles d'enregistrer — et, si nécessaire, d'agir sur — toutes les réactions opérateur.

Configuration & mise en file d'attente des boîtes de dialogue. La configuration d'une boîte de dialogue est réalisée par l'intermédiaire du champ *Title* du bloc, qui définit le bandeau de la boîte de dialogue et le champ *Body*, qui définit le contenu de la boîte de dialogue, tout deux comme des entrées de dictionnaire.

NOTA. Tous les messages texte pour ce bloc sont stockés dans un *dictionnaire de messages* spécial ('M'). Une valeur d'entrée de dictionnaire de 0 est interprétée comme "absence de texte". Ceci n'empêche pas bien sûr des références à d'autres dictionnaires dans un formulaire associé.

Une entrée VRAIE dans *Trigger* ajoute la boîte de dialogue configurée à la file d'attente des boîtes de dialogue où elle reste active et prête à être affichée à l'écran pour que l'opérateur y réagisse. La boîte de dialogue affichée est immédiatement supprimée, si l'opérateur y réagit en appuyant sur un bouton de la boîte de dialogue ou si l'application proprement dite y réagit automatiquement.

Les boîtes de dialogue sont mises en file d'attente (par défaut) dans l'ordre chronologique de création, la boîte de dialogue la plus ancienne (le cas échéant) étant affichée à l'écran. Le paramètre *Priority* permet cependant de modifier cet état de fait et de remonter une boîte de dialogue dans la file d'attente. Pour les boîtes de dialogue de même priorité, la plus ancienne prévaut sur l'autre. Si l'opérateur ne réagit pas à la boîte de dialogue active (ou en l'absence de réaction automatique), **aucune autre action opérateur** ne peut être effectuée. Lorsqu'une boîte de dialogue est supprimée, la boîte de dialogue suivante dans la file d'attente (s'il y en a une) s'affiche.

Boutons opérateur. Un maximum de quatre "boutons" opérateur peuvent être affichés au bas de la boîte de dialogue. Ceux-ci sont sélectionnés par le paramètre *Buttons* et il s'agit de 'OK', 'CANCEL', 'YES', 'NO', etc. plus quatre boutons de format libre définis par l'utilisateur avec des légendes définies dans un dictionnaire (ce qui permet de gérer des applications multilingues). Le paramètre *Response* consigne le bouton sur lequel l'opérateur a appuyé ou si l'application a réagi automatiquement, le ou lesquels des boutons ont été activés. Le paramètre *SelfResp* définit le ou lesquels des boutons sera(ont) activés par l'application, lorsqu'il est déclenché par une entrée VRAIE dans le champ *TrigSelf*.

NOTA. Plusieurs blocs PNL_DLG peuvent tourner dans une application, chaque bloc permettant de gérer une boîte de dialogue active dans la file d'attente des boîtes de dialogue à un moment donné.

Paramètres du bloc

Les symboles utilisés dans la table 20-6 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PanelId	Id du panneau (<i>par défaut = 1. Non encore mis en oeuvre</i>)	Entier	
Trigger	VRAI ajoute boîte de dialogue file attente (verrouillage)	V/F	
Title	Entrée dictionnaire pour titre boîte dialogue (20 car. maxi)	Entier	
Body	Entrée dictionnaire pour texte boîte dialogue	Entier	
FormFile	(<i>Non encore mis en oeuvre</i>)	Alphanumérique	
FormId	(<i>Non encore mis en oeuvre</i>)	Entier	
Options	Fonctions en option	(ABC)D hex	
PlsWait	VRAI ajoute texte clignotant 'Please wait...' à la boîte de dialogue	V/F — 1	
Progress	VRAI ajoute barre progression, contrôlée par entrée <i>Progress</i>	V/F — 2	D
		V/F — 4	
		V/F — 8	
		V/F — 8	
Buttons	Sélectionne boutons à afficher dans boîte dialogue	(A)BCD hex	
OK	Bouton 'OK' (par défaut = VRAI)	V/F — 1	D
CANCEL	Bouton 'CANCEL'	V/F — 2	
YES	Bouton 'YES'	V/F — 4	
NO	Bouton 'NO'	V/F — 8	
ABORT	Bouton 'ABORT'	V/F — 1	C
UButton1	Bouton utilisateur 1	V/F — 2	
UButton2	Bouton utilisateur 2	V/F — 4	
UButton3	Bouton utilisateur 3	V/F — 8	
Légendes définies dans <i>UButton1-4</i>			
UButton4	Bouton utilisateur 4	V/F — 1	B
		V/F — 2	
		V/F — 4	
		V/F — 8	
UButton1 - 4	Légendes dictionnaire pour boutons utilisateur 1 - 4	Entier	
Progress	Entrée indicateur progression en option (<i>Options.Progress</i> active)	Eng	
Priority	Priorité ordre file attente boîte dialogue, 16 - 31. (par déf. = 16)	Entier	
Emphasis	Mise en évidence appliquée titre boîte dialogue (indique urgence)	Menu	
TrigSelf	Déclenche "réaction auto" & supprime boîte dialogue (verrouillage)	V/F	
SelfResp	Définit réaction auto (tous boutons) si <i>TrigSelf</i> VRAI	(A)BCD hex	
OK	Bouton 'OK' (par défaut = VRAI)	V/F — 1	D
CANCEL	Bouton 'CANCEL'	V/F — 2	
YES	Bouton 'YES'	V/F — 4	
NO	Bouton 'NO'	V/F — 8	
ABORT	Bouton 'ABORT'	V/F — 1	C
UButton1	Bouton utilisateur 1	V/F — 2	
UButton2	Bouton utilisateur 2	V/F — 4	
UButton3	Bouton utilisateur 3	V/F — 8	

Table 20-6 suite...

...Table 20-6 suite

Paramètre	Fonction	Unités	Etat
UButton4	Bouton utilisateur 4	V/F — 1 V/F — 2 V/F — 4 V/F — 8	B
Alarms			
Software	Anomalie mémoire données RAM bloc/défaut réseau	V/F	
Nbuttons	Trop de boutons demandés (limite = 4)	V/F	
Dictnry	Erreur dictionnaire	V/F	
TextLen	Texte dans dictionnaire trop long pour contexte	V/F	
Form	Erreur fichier formulaire (voir <i>FormErr</i> , <i>FormErLn</i> , <i>FormErCh</i>)	V/F	
Failure	Echec mise en file attente boîte de dialogue	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Triggerd	Acquittement déclencheur	V/F	
Response	Signale boutons "activés" par opérateur ou bits réaction auto (A)BCD hex		
OK	Bouton 'OK' activé	V/F — 1	D
CANCEL	Bouton 'CANCEL' activé	V/F — 2	
YES	Bouton 'YES' activé	V/F — 4	
NO	Bouton 'NO' activé	V/F — 8	
ABORT	Bouton 'ABORT' activé	V/F — 1	C
UButton1	Bouton utilisateur 1 activé	V/F — 2	
UButton2	Bouton utilisateur 2 activé	V/F — 4	
UButton3	Bouton utilisateur 3 activé	V/F — 8	
UButton4	Bouton utilisateur 4 activé	V/F — 1 V/F — 2 V/F — 4 V/F — 8	B
Status	Statut boîte de dialogue	(ABC)D hex	
Active	VRAI = boîte dialogue active, visualisée ou non	V/F — 1	D
OnView	VRAI = boîte dialogue & visualisée à l'écran	V/F — 2	
		V/F — 4	
		V/F — 8	
FormErr	Code erreur formulaire	Menu	
FormErLn	Ligne où erreur signalée par <i>FormErr</i> est survenue	Entier	
FormErCh	Position caractère erreur sur ligne signalée par <i>FormErLn</i>	Entier	

Table 20-6 Paramètres du bloc PNL_DLG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 20-6.

Trigger. La mise à 1 de l'entrée *Trigger* génère une boîte de dialogue (en fonction de vos spécifications) et l'ajoute à la file d'attente des boîtes de dialogue actives, qui attendent l'attention de l'opérateur. *Trigger* se verrouille et doit être remis à l'état FAUX avant d'être réutilisé. Notez que le passage à l'état VRAI remet à zéro tous les bits dans le champ binaire *Response*.

Title. Entrée de dictionnaire pour le titre de la boîte de dialogue, 20 caractères maxi. Celle-ci s'affiche comme "bandeau" de la boîte de dialogue et peut plus ou moins être mise en évidence pour indiquer le niveau d'urgence à l'opérateur (en utilisant le paramètre *Emphasis*).

Options. Fonction en option des boîtes de dialogue.

■ **PlsWait.** VRAI ajoute une légende clignotante '**Please wait...**' (Veuillez patienter...) à la boîte de dialogue — ou éventuellement le texte défini dans l'entrée 89 (S89) du dictionnaire système. Vous pouvez, par exemple, utiliser cette fonction dans une boîte de dialogue pour forcer l'opérateur(trice) à attendre la fin d'une procédure particulière avant de l'autoriser à poursuivre. Aucune autre action opérateur — autre que d'activer un bouton de boîte de dialogue (si activés) — ne peut être effectuée, lorsqu'une boîte de dialogue est affichée à l'écran. Une barre de progression peut être intégrée à la boîte de dialogue (voir ci-après), ainsi que l'option *PlsWait*, si c'est possible.

■ **Progress.** VRAI ajoute une barre de progression horizontale, contrôlée par l'entrée *Progress*. Si vous voulez disposer de cette fonction, créez une liaison filaire sur une entrée appropriée du schéma de boucles. L'indication affichée va de 0.0 à 100.0.

Buttons. Sélectionne quels boutons seront affichés au bas de la boîte de dialogue. Les informations suivantes complètent celle de la table 20-6.

■ **UButton1-4.** Ces bits permettent de sélectionner un maximum de quatre boutons utilisateur de format libre avec des légendes définies dans les champs dictionnaire correspondants *UButton1* à *UButton4*.

Priority. Priorité dans l'ordre de la file d'attente des boîtes de dialogue (16 - 31, par défaut = 16, la priorité la plus basse). Les boîtes de dialogue sont normalement mises en file d'attente chronologiquement, la boîte de dialogue la plus ancienne étant la première dans la file d'attente soumise à l'attention de l'opérateur. Le paramètre *Priority* permet cependant de modifier cet état de fait et de remonter une boîte de dialogue plus récente dans la file d'attente, parce que la priorité prévaut sur l'ancienneté. Si les boîtes de dialogue ont les mêmes priorités (cas par défaut), la boîte de dialogue la plus ancienne prévaut sur l'autre.

Emphasis. (NONE/WEAK/STRONG/ATTN) Sélectionne la mise en évidence appliquée au titre de la boîte de dialogue pour attirer l'attention de l'opérateur et indiquer le niveau d'urgence de la boîte de dialogue.

■ **NONE.** Aucune évidence (titre 'Eurotherm vert').

■ **WEAK.** Légère mise en évidence (titre vert).

■ **STRONG.** Mise en évidence soutenue (titre rouge).

■ **ATTN.** Mise en évidence maximale pour attirer l'attention (titre rouge clignotant).

TrigSelf. Cette entrée permet à l'application (schéma de boucles) de réagir aux boutons dans une boîte de dialogue, au lieu de tabler sur les réactions de l'opérateur. En particulier, une entrée VRAIE pour *TrigSelf* copie le profil binaire du paramètre *SelfResp* sur le paramètre *Response*. Ceci revient à appuyer automatiquement sur tous les boutons définis dans *SelfResp*. Lorsque *TrigSelf* est mis à 1, le message est supprimé de la file d'attente (comme lorsqu'un opérateur y répond).

NOTA. *TrigSelf* se verrouille et doit être remis à l'état FAUX avant d'être réutilisé.

SelfResp. Définit la réponse automatique de la boîte de dialogue — autrement dit, le ou lesquels des boutons seront effectivement activés, lorsque l'application met l'entrée *TrigSelf* à l'état VRAI. En particulier, une entrée VRAIE pour *TrigSelf* copie le profil binaire du paramètre *SelfResp* sur le paramètre *Response*. Ceci revient à appuyer automatiquement sur tous les boutons définis dans *SelfResp*.

La réponse automatique peut être utilisée par l'application comme alternative à l'attente d'une réaction opérateur. Notez que, à l'inverse d'une réaction opérateur, une réponse automatique permet d'appuyer simultanément sur plusieurs boutons et également d'activer des boutons qui n'ont pas été validés par le paramètre *Buttons*, et n'apparaissent donc pas dans la boîte de dialogue.

Triggerd. Cette sortie passe à l'état VRAI, après que *Trigger* a été mis à 1 et la nouvelle boîte de dialogue qui en résulte a été ajoutée à la file d'attente. Elle peut être utilisée comme protocole de transfert pour confirmer la création d'une boîte de dialogue.

Response. Ce champ binaire en lecture seule enregistre les réactions opérateur aux boîtes de dialogue (ou automatique). Le ou les bits correspondants passe(nt) à l'état VRAI, lorsque l'opérateur appuie sur un bouton ou lorsqu'une réponse automatique est déclenchée par une entrée dans *TrigSelf*. Les bits sont verrouillés à l'état VRAI, jusqu'à ce que l'entrée *Trigger* soit mise à 1, lorsque tous les bits sont remis à l'état FAUX (et une nouvelle boîte de dialogue est générée).

PNL_ACC: BLOC D'ACCÈS AU PANNEAU

Fonction du bloc

Le bloc PNL_ACC permet à une application de contrôler la procédure d'ouverture/fermeture de session, à la fois pour le système d'accès standard de Visual Supervisor et son système d'accès multi-utilisateur.

Pour le système standard, l'ouverture de session est configurée, en utilisant le paramètre *ReqLevel* pour sélectionner le niveau d'accès requis, ex. ENGINEER, et le paramètre *Password* pour enregistrer le mot de passe en question. Pour le système multi-utilisateur, *ReqID* et *Password* consistent l'identité de l'utilisateur et son mot de passe. (Le niveau d'accès n'est pas requis pour une ouverture de session multi-utilisateur).

Une entrée VRAIE dans le champ *LogOn* ouvre alors la session de l'utilisateur préconfiguré ou définit le niveau d'accès requis. L'entrée *LogOff* permet de fermer la session de l'utilisateur connecté et remet le niveau d'accès à l'état LOCKED (verrouillé).

Voir les détails des systèmes d'accès dans le manuel produit de l'instrument.

Paramètres du bloc

Les symboles utilisés dans la table 20-7 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, *Menu de spécifications du bloc*.

Paramètre	Fonction	Unités	Etat
PanelId	Id du panneau (<i>par défaut = 1. Non encore mis en oeuvre</i>)	Entier	
LogOn	VRAI déclenche ouverture de session (verrouillage)	V/F	
LogOff	VRAI déclenche fermeture de session (verrouillage)	V/F	
ReqID	ID utilisateur exigée (stratégie multi-utilisateur) (2 - 8 car.)	Alphanumérique	
ReqLevel	Niveau d'accès exigé (stratégie standard)	Menu	
Password	Mot de passe nouvel utilisateur (affiché *) (0 - 8 car)	Alphanumérique	
Options	Fonctions en option	(ABC)D hex	
LOnHome	Retour à page accueil ouverture de session	V/F	<div>1 2 4 8</div> D
LOffHome	Retour à page accueil fermeture de session	V/F	
		V/F	
		V/F	
Alarms			
Software	Anomalie données RAM bloc/défaut réseau	V/F	
InvLogon	Tentative ouverture session rejetée, ex. mot de passe incorrect	V/F	
Conflict	Ouverture session rejetée car conflit	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
LoggedOn	VRAI si connecté (niveau > LOCKED), sinon FAUX	V/F	
CurID	ID utilisateur connecté ou vierge si déconnecté	Alphanumérique	
CurLevel	Nom du niveau d'accès actif	Menu	
CurLvlNo	Référence du niveau d'accès actif (1 - 5)	Entier	
CurRef	N° de référence de l'utilisateur connecté (0 - 65535)	Entier	

Table 20-7 Paramètres du bloc PNL_DLG

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 20-7.

ReqLevel, CurLevel. (LOCKED/OPERATOR/COMMISSI/ENGINEER/ADMIN)

Niveau d'accès exigé (système d'accès standard) et niveau d'accès en cours (pour les deux systèmes).

- **LOCKED.** Niveau d'accès verrouillé.
- **OPERATOR.** Niveau d'accès opérateur.
- **COMMISSI.** Niveau d'accès mise en service.
- **ENGINEER.** Niveau d'accès ingénieur.
- **ADMIN.** Niveau d'accès administrateur (système d'accès multi-utilisateur uniquement).

Options. Fonctions d'accès en option.

- **LOnHome.** Retour à la page d'accueil à l'ouverture de session. Dans la stratégie standard, l'ouverture de session signifie passer du niveau d'accès LOCKED à un niveau supérieur.
- **LOffHome.** Retour à la page d'accueil à la fermeture de session. Dans la stratégie standard, la fermeture de session signifie passer d'un niveau supérieur à LOCKED.

CurLvIno. Sort le niveau d'accès actif — *CurLevel* — sous la forme d'un entier dans la plage de 1 à 5. La table 20-8 donne la liste des valeurs *CurLvIno* et les désignations *CurLevel* correspondantes.

CurLvIno	CurLevel
1	LOCKED
2	OPERATOR
3	COMMISSI(ON)
4	ENGINEER
5	ADMIN

Table 20-8 Valeurs CurLvIno et désignations des niveaux d'accès correspondants

CurRef. Sort le n° de référence de l'utilisateur connecté (système d'accès multi-utilisateur uniquement) sous la forme d'un entier.

READER: BLOC DE LECTURE

Fonction du bloc

Le bloc READER permet d'utiliser des unités de lecture, qui gèrent des périphériques d'entrée basés sur des enregistrements — par ex. des lecteurs de codes à barres.

Un enregistrement d'entrée d'une telle unité est interprété comme suit:

- 1 Un nombre (éventuellement 0) de caractères préfixes, dont le premier peut être vérifié en option.
- 2 Le corps de l'enregistrement auquel est appliqué un filtrage.
- 3 Un nombre (éventuellement 0) de caractères suffixes, dont le dernier peut être vérifié en option.
- 4 Un caractère de fin, dont la détection déclenche le processus de correspondance.

Paramètres du bloc

Les symboles utilisés dans la table 20-9 sont expliqués au chapitre 1. Des informations supplémentaires sur les paramètres sont données dans la section ci-après, Menu de spécifications du bloc.

Paramètre	Fonction	Unités	Etat
PanelId	Id du panneau (<i>par défaut=1. Non encore mis en oeuvre</i>)	Entier	
Disable	VRAI désactive bloc. (VRAI à FAUX vide le tampon entrée)	V/F	
Options	Fonctions en option	(ABC)D hex	
ChkPrefix	VRAI = vérifier caractère préfixe initial (par déf. FAUX)	V/F — 1	D
ChkSuffix	VRAI = vérifier caractère suffixe final (par déf. FAUX)	V/F — 2	
WildSpc	VRAI = utiliser 'espace' comme car. générique dans filtrage	V/F — 4	
WildQuMk	VRAI = utiliser '?' comme car. générique dans filtrage	V/F — 8	
Device	L'unité à lire (READER1/READER2)	Menu	
PattFile	Fichier définissant matrices & actions d'entrée	Alphanumérique	
PrefChrs	Nombre total de caractères préfixes	Entier	
Prefix	Caractère préfixe initial, si vérifié (par déf=02 hex)	Entier	
SuffChrs	Nombre total de caractères suffixes	Entier	
Suffix	Caractère suffixe final, si vérifié (par déf = 03 hex)	Entier	
Termintr	Caractère de fin enregistrement d'entrée (par déf=0D hex)	Entier	
Alarms			
Software	Anomalie données RAM bloc/défaut réseau	V/F	
Config	Erreur de configuration	V/F	
Pattern	Erreur fichier matrice (détaillé dans <i>PattErr</i>)	V/F	
Comms	Erreur liée au port de communication	V/F	
Combined	Combinaison de tous les bits d'alarme	V/F	
Matched	Passe à VRAI si l'entrée correspond (verrouillage)	V/F	
MatchNo	Réf. matrice (n° de ligne dans <i>PattFile</i>) qui correspond	Entier	
MatchPne	Fenêtre où correspondance a été trouvée (<i>Non mis en oeuvre</i>)	Entier	
MatchId	Id page utilisateur où correspondance trouvée (<i>Non mis en oeuvre</i>)	Entier	

Table 20-9 suite...

...Table 20-9 suite






Paramètre	Fonction	Unités	Etat
MatchCnt	Nombre de correspondances jusqu'à présent (à remise à 0)	Entier	
UnmatCnt	Nombre de non correspondances jusqu'à présent (à remise à 0)	Entier	
PattErr	Code d'erreur matrice	Menu	
PattErLn	Ligne où l'erreur signalée par <i>PattErr</i> est survenue	Entier	
PattErCh	Position caractère de l'erreur sur ligne signalée par <i>PattErLn</i>	Entier	

Table 20-9 Paramètres du bloc READER

Menu de spécifications du bloc

Les informations suivantes complètent celles de la table 20-9.

PattFile. Nom du fichier de langage d'interface du lecteur définissant les matrices et actions d'entrée. Voir les détails sur ces fichiers dans le manuel produit de l'instrument.

PattErr. (OK /BAD_FILE /LINE_LEN /NEWLINE /MEMORY /SYNTAX /RANGE /NAME /DICTNRY /TYPE /ACTION /CONTEXT/OTHER) Code d'erreur de la matrice. La signification des options est la suivante:

- **OK.** Aucune erreur.
- **BAD_FILE.** Fichier introuvable ou illisible.
- **LINE_LEN.** Une ligne dans le fichier était trop longue (la limite est de 255, retour chariot et/ou interligne non compris).
- **NEWLINE.** Fichier ne se termine pas par une nouvelle ligne (indique éventuellement qu'il est corrompu).
- **MEMORY.** Fichier trop gros pour la mémoire qui lui est affectée.
- **SYNTAX.** Erreur de syntaxe.
- **RANGE.** Une valeur numérique est en dehors de la plage admissible.
- **NAME.** Un objet désigné est introuvable (peut-être mal orthographié).
- **DICTNRY.** Une entrée de dictionnaire est introuvable.
- **TYPE.** Un type d'objet est inapproprié dans le contexte, dans lequel il est utilisé.
- **ACTION.** (*Non mis en oeuvre*).
- **CONTEXT.** Une construction a été utilisée et n'est pas valable dans le contexte des fichiers de langage d'interface de lecteur.
- **OTHER.** Autre erreur.

[Page laissée intentionnellement blanche]

Annexe A

MODES DE FONCTIONNEMENT DE LA BOUCLE DE REGULATION

Le bloc de régulation PID peut fonctionner suivant différents types de *modes de régulation*, chacun ayant son propre mode de contrôle de la boucle. Les caractéristiques de ces modes sont décrites au Chapitre 5 dans le paragraphe sur le bloc PID. La présente annexe explique comment les modes de régulation sont sélectionnés et comment ils dialoguent entre eux. Cette annexe décrit également l’algorithme de régulation PID à 3 termes utilisé dans le bloc PID, et explique les techniques d’équilibrage et de désaturation de l’intégrale, ainsi que leur mise en oeuvre.

MODE DE RÉGULATION ACTIF

Les modes de la boucle de régulation sont sélectionnés, mais pas activés, par l’intermédiaire du paramètre *SelMode* ou (dans certains cas) par l’intermédiaire du champ *Mode*. Plusieurs modes peuvent être sélectionnés à la fois, mais un seul est actif. Un mode “sélectionné” est donc uniquement *potentiellement* actif. Le mode actif est le mode sélectionné qui a la priorité la plus élevée. Si le mode actif est désélectionné, le mode sélectionné avec la priorité suivante la plus haute devient actif.

Priorités du mode de régulation

L’ordre du mode de priorité est le suivant :

Priorité	Mode	Mode de reprise*
1 (la plus haute)	HOLD	—
2	TRACK	—
3	FORCED MANUAL	MANUAL
4	MANUAL	MANUAL
5	AUTO	AUTO
6	REMOTE	REMOTE
7 (la plus basse)	FORCED AUTO	FORCED AUTO

* Voir le paragraphe suivant

Modes de reprise (Fallback)

Le mode de reprise est le mode de régulation qui prend la relève du mode actif en cours, si tous les modes sont désélectionnés par le champ binaire *SelMode*, c’est à dire si tous les bits se mettent en FALSE (FAUX) (*SelMode* = 00000000), ce qui n’est pas nécessairement

la même chose que le mode qui prend la relève si le mode actif en cours est désélectionné; dans ce cas, ce sont les priorités du mode qui sont déterminantes.

Chaque mode de régulation a son propre mode de reprise (en général lui-même). Voir la liste des modes de reprise dans la table ci-dessus. MAINTIEN (HOLD) & POURSUITE (TRACK) sont les seules exceptions - lorsque l'un ou l'autre de ces modes devient actif, le mode de reprise reste à sa valeur précédente.

NOTA. Si tous les bits du *SelMode* passent à zéro *simultanément*, le mode de régulation que la boucle "reprend" n'est pas défini, dans la mesure où en réalité chaque bit est remis à zéro dans un ordre d'échantillonnage qui n'est généralement pas connu de l'utilisateur, et donc le tout dernier mode actif, ainsi que sa reprise, sont imprévisibles.

Sélection des modes de régulation

Les modes de régulation peuvent être sélectionnés/activés soit en reliant les liaisons logiques aux bits du paramètre *SelMode* à partir d'autres sections du schéma de boucles ou en écrivant directement dans ces bits ou en "appuyant" sur les boutons-poussoirs de la face avant synoptique de conduite (T1000). Les entrées câblées du mode *SelMode* prennent toujours le pas sur les entrées écrites, qui à leur tour prennent le pas sur les boutons-poussoirs. Les boutons-poussoirs agissent directement sur le paramètre *Mode*, et représentent une alternative à l'écriture directe dans le *Mode*.

Mode Déporté (Remote)

Le mode déporté est inhabituel, dans la mesure où il ne peut devenir actif que s'il est validé (*EnaRem* = TRUE (VRAI)) et à condition qu'il soit le mode sélectionné avec la priorité la plus haute (*SelRem* = TRUE). Les modes qui peuvent être adoptés par la boucle, déterminés par ces deux bits du paramètre *SelMode* sont :

SelRem	EnaRem	Mode
TRUE	TRUE	REMOTE (si priorité la plus haute)
TRUE	FALSE	FORCED AUTO (si priorité la plus haute)
FALSE	TRUE	Mode en cours
FALSE	FALSE	Mode en cours

ALGORITHME DE RÉGULATION PID À 3 TERMES

Equation de régulation analogique à 3 termes

L'équation de régulation (PID) classique à 3 termes mise en oeuvre par les régulateurs analogiques traditionnels en utilisant des amplificateurs opérationnels s'écrit généralement comme suit :

$$OP = -\frac{100}{XP} \left[ER + \frac{1}{TI} \int ER dt + TD \frac{dER}{dt} \right] \quad (1)$$

où :

- OP = sortie du régulateur
- XP = bande proportionnelle
- TI = temps d'action intégrale
- TD = temps d'action dérivée
- ER = erreur (PV-SP)

Cette équation peut être réécrite suivant la terminologie Y(s) de la transformée de Laplace:

$$\frac{OP(s)}{ER} = -\frac{100}{XP} \left(1 + \frac{1}{sTI} + sTD \right) \quad (2)$$

La limitation de la réponse de haute fréquence introduit un filtre, choisi typiquement pour avoir une constante de temps égale à un quart du temps de la dérivée. La fonction complète de transfert est alors la suivante :

$$\frac{OP(s)}{ER} = -\frac{100}{XP} \left(1 + \frac{1}{sTI} + sTD \right) \left(\frac{1}{1 + sTD/4} \right) \quad (3)$$

Algorithme de régulation numérique

Dans les instruments pilotés par microprocesseur comme le T1000/T100, les techniques d'échantillonnage doivent être utilisées pour calculer les termes de l'équation de régulation. Il est également plus pratique de réécrire la fonction de transfert en terme d'équations différentielles plutôt que suivant la terminologie Y(s) de la transformée de Laplace. Ainsi, la sortie calculée à 3 termes après n échantillons est donnée par :

$$OP_n = -\frac{100}{XP} \left[ER_n + \frac{TS}{TI} \sum_{r=1}^{r=n} ER_r + \frac{TD}{TS} \Delta PV_n \right] + FF \quad (4)$$

où, (en plus) :

- FF = tendance
- OP_n = sortie du régulateur après n échantillons
- ER_n = valeur d'erreur pour l'échantillon n
- ER_r = valeur d'erreur pour l'échantillon r
- ΔPV_n = modification de la valeur de la variable procédé filtrée entre les échantillons n et n-1.

ΔPV_n est obtenu après un filtrage du premier ordre avec une constante de temps effective TD/4, ainsi :

$$\Delta PV_n = \Delta PV_{n-1} + \frac{4TS}{TD} (dN - \Delta PV_{n-1}) \quad (5)$$

où, (en plus) :

$$dN = PV_n - PV_{n-1}$$

La mesure PV est elle-même une version filtrée de la valeur échantillonnée de l'entrée MV.

$$PV_n = PV_{n-1} + \frac{TF}{IF} (MV_n - PV_{n-1}) \quad (6)$$

où, (en plus) :

MV_n	=	valeur de l'entrée analogique pour l'échantillon n
TF	=	constante de temps effective du premier ordre
IF	=	constante filtre voie d'entrée

NOTA. Le décalage FF est apparent à zéro erreur sous l'action de la régulation proportionnelle, le terme de l'intégrale étant invalidé, ce qui permet à la sortie de répondre à la fois aux erreurs négatives et positives, si nécessaire.

Equivalence entre les équations analogiques et numériques

Lorsque le point de consigne est constant, l'algorithme numérique de l'équation (4) peut se transcrire comme fonction de transfert continue équivalente suivante :

$$\frac{OP(s)}{ER} = - \frac{100}{XP} \left(1 + \frac{1}{sTI} + sTD \right) \left(\frac{1}{1 + sTD/4} \right) \quad (7)$$

On peut alors comparer cette équation à la version classique des $Y(s)$ du régulateur analogique représenté dans l'équation (2). Les termes proportionnels (P) et de l'intégrale (I) sont identiques, mais le terme de la dérivée (D) est légèrement différent. C'est parce que le filtrage supplémentaire du premier ordre est appliqué à la valeur de la dérivée ΔPV , plutôt que directement à l'erreur.

NOTA. La réponse aux modifications du point de consigne local (SL) est déterminée par la valeur du bit *IntBalSL* dans le paramètre d'options du bloc de régulation, qui peut être positionné pour invalider l'équilibrage du terme de l'intégrale en cas de modifications de SL . De même, le bit *IntBalXP* détermine la réponse en cas de modifications de XP .

L'équilibrage de l'intégrale est effectuée *automatiquement*, chaque fois que le mode de la boucle passe en AUTO, DEPORTE ou AUTO FORCE.

Voir la description plus détaillée de l'algorithme à 3 termes dans la section 3 de la publication EUROTHERM SYSTEMES *Manuel des applications du régulateur Système 6000*.

EQUILIBRAGE DE L'INTÉGRALE & DÉSATURATION DE L'INTÉGRALE

Il y a deux techniques de calcul conditionnées par des situations particulières et appliquées au *terme de l'intégrale* de l'algorithme PID pour améliorer le comportement de sortie du régulateur lorsque ces situations se produisent. Bien que les techniques soient généralement bénéfiques dans une boucle de régulation, elles modifient certainement les caractéristiques de réponse - un point qui doit être pris en considération lors de la conception et de la mise au point d'un système de régulation.

Equilibrage de l'intégrale

L'équilibrage de l'intégrale est appliquée pour empêcher les changements brusques - "à-coups" - de la sortie qui pourraient se produire après un changement de mode de régulation, et les changements de pas dans certains paramètres de régulation (par ex. la consigne). Les à-coups de la sortie sont indésirables, dans la mesure où ils peuvent endommager les vannes et déstabiliser le procédé.

Par principe, l'équilibrage de l'intégrale fonctionne par le réglage de l'accumulateur de l'intégrale (TS/TI. ΣER_i) dans le calcul PID pour maintenir la nouvelle sortie (presque) égale à la valeur qui précède le déclenchement de l'événement. Les détails mathématiques sont donnés dans une section ultérieure.

Effet de l'équilibrage de l'intégrale sur la dynamique

Les figures 15-1 et 15-2 montrent un exemple des réponses de la sortie (*OP*) et de la variable procédé (*PV*) à un changement brusque du point de consigne (*SP*), sans et avec l'application de l'équilibrage de l'intégrale. Dans ce cas, l'application de l'équilibrage de l'intégrale réduit nettement la pente et la valeur de crête de la réponse *OP*, tout en ramenant *PV* à la nouvelle valeur *SP* presque tout aussi rapidement, mais avec la moitié du dépassement seulement.

Il est clair que l'équilibrage de l'intégrale modifie la réponse de la boucle aux changements de points de consigne, et il faut prendre en considération ce fait, lorsqu'une boucle est mise au point en utilisant ces changements. La réponse d'une boucle bien mise au point avec l'équilibrage de l'intégrale a en général un dépassement inférieur à la boucle sans équilibrage de l'intégrale.

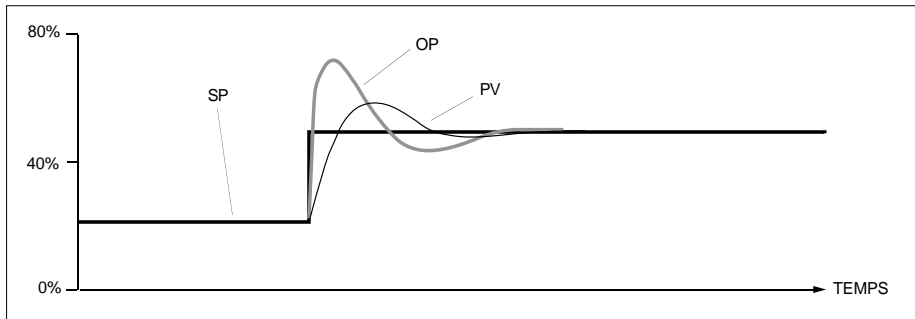


Figure 15-1 Réponse au changement du point de consigne du pas - sans équilibrage de l'intégrale

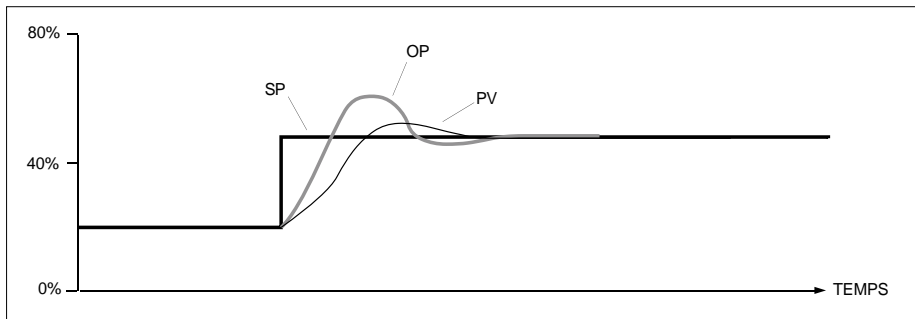


Figure 15-2 Réponse au changement du point de consigne du pas - avec équilibrage de l'intégrale

Conditions de déclenchement de l'équilibrage de l'intégrale

- Modification du point de consigne local (*SL*) en mode AUTO. (En option, par l'intermédiaire du paramètre du bloc de régulation *Options.IntBalSL*).
- Modification de la bande proportionnelle (*XP*) en mode AUTO ou DEPORTE. (En option, par l'intermédiaire du paramètre *Options.IntBalXP*)
- Modification du mode de fonctionnement en n'importe quel mode automatique (AUTO, DEPORTE, AUTO FORCE).
- Entrée d'un front montant dans le bit *IntBal* du paramètre *Options* du bloc PID.

Équilibrage de l'intégrale & contrôle adaptatif du gain

L'équilibrage de l'intégrale est particulièrement utile dans les usines où plusieurs ensembles de constantes de mise au point sont nécessaires pour traiter les différentes dynamiques du procédé à différents niveaux de fonctionnement (par ex. dosage pH). Dans ce cas, l'équilibrage de l'intégrale empêche les à-coups de la sortie, chaque fois que les constantes sont modifiées pour une limite.

Dans certaines applications de régulation, la valeur de la bande proportionnelle XP doit être modifiée *continuellement* ("gain adaptatif"). Dans ces cas, $IntBalXP$ devrait être mis à FALSE (FAUX) pour empêcher un équilibrage de l'intégrale à chaque itération de l'algorithme PID, ce qui générerait une régulation correcte.

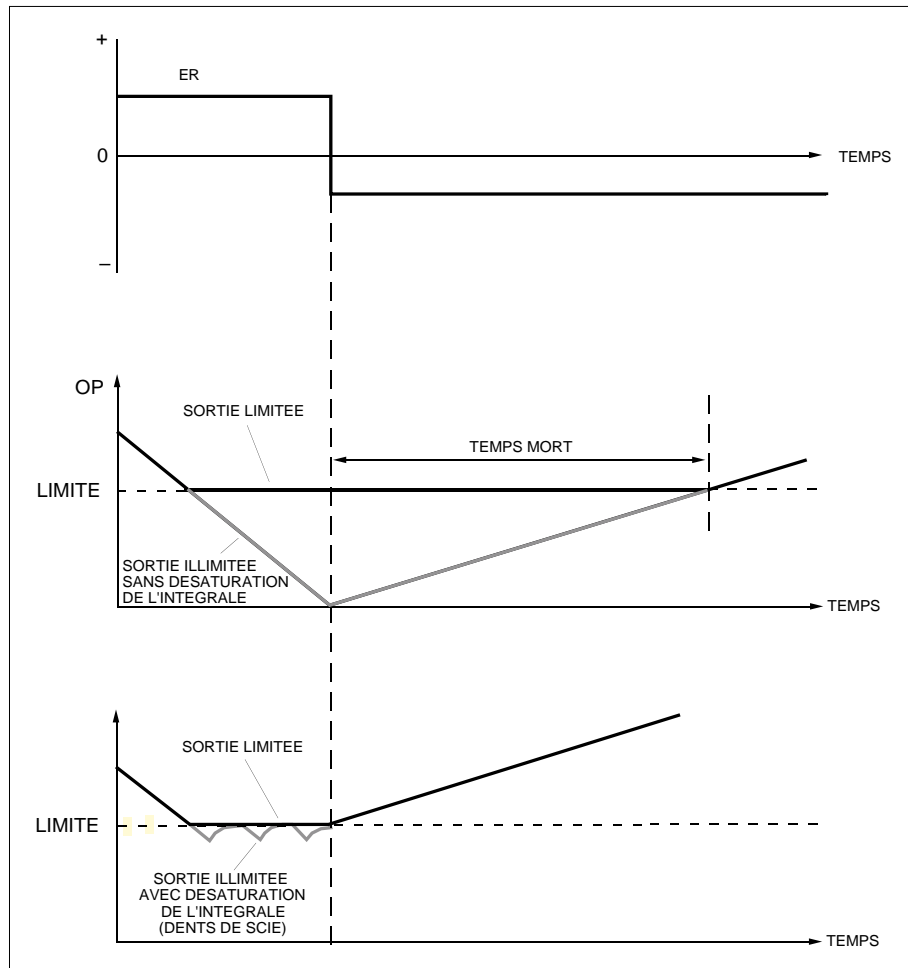


Figure 15-3 Effet de la désaturation de l'intégrale (action de l'intégrale)

Désaturation de l'intégrale

Dans le cas d'une erreur persistante (ER), le terme de l'intégrale de l'algorithme PID ($TS/ TI. \Sigma ER_i$) peut accumuler une valeur importante ("saturation du terme de l'intégrale"), qui

ne commence à diminuer que lorsque ER s'inverse. Lorsque l'accumulateur de l'intégrale est assez grand, il maintient la sortie du régulateur à une limite de sortie, et même au cours de la réduction, il continue le maintien pendant une période de *temps mort* supplémentaire jusqu'à ce qu'il se soit suffisamment vidé. La partie supérieure de la figure 15-3 montre cet effet schématiquement.

La désaturation de l'intégrale élimine ce temps mort, en permettant à la sortie de s'écarter de la limite dès que l'erreur s'inverse, et évite ainsi les dépassements importants de PV . L'action de l'intégrale est suspendue, lorsqu'une limite de sortie est dépassée, et l'accumulateur de l'intégrale est "déchargé" jusqu'à ce que la sortie calculée revienne à la limite. A ce stade, l'action de l'intégrale reprend. La désaturation de l'intégrale est activée de façon répétée, si nécessaire, pour maintenir la sortie près de la limite, jusqu'à ce que l'erreur s'inverse. La partie inférieure de la figure 15-3 montre cet effet schématiquement.

Dans cet exemple, l'action de régulation est *uniquement l'intégrale*, pour des raisons de clarté. L'effet de "dents de scie" de la sortie illimitée provient de l'application répétée de la désaturation de l'intégrale pour la ramener à la limite. Ce phénomène ne peut être constaté dans la sortie limitée envoyée à l'installation.

Détection de la limite

L'approche d'une limite est détectée en comparant (à chaque itération) la valeur de sortie calculée OP et la valeur de contre-réaction correspondante FB , qui a été limitée en station manuelle. Si ces valeurs diffèrent de plus d'une marge étroite, il est considéré qu'une limite a été atteinte et la désaturation de l'intégrale commence. La figure 15-4 en donne une représentation schématique. (Les détails mathématiques seront donnés par la suite).

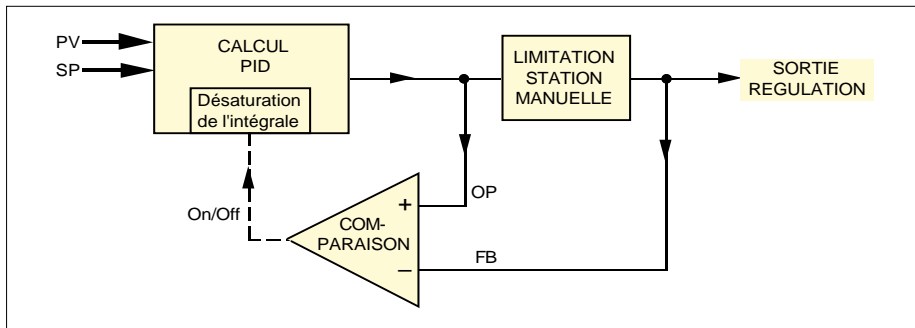


Figure 15-4 Application de la désaturation de l'intégrale (schématiquement)

La marge entre OP et FB nécessaire pour déclencher la désaturation de l'intégrale est très étroite (0,006 % approximatif.), donc le signal de contre-réaction devrait venir directement de la sortie de la station manuelle, chaque fois que c'est possible. FB peut provenir d'ailleurs, à condition que les temporisations excessives dans la boucle de contre-réaction soient évitées. Les temporisations supérieures au temps d'échantillonnage (TS) peuvent faire que la contre-réaction diffère suffisamment de la dernière sortie pour déclencher une désaturation de l'intégrale intempestive, ce qui ralentit l'action du régulateur.

Effet de la désaturation de l'intégrale sur la dynamique

La figure 15-5 montre un exemple plus réaliste où la désaturation de l'intégrale empêche un dépassement excessif dans une application de régulation PID complète. En comparaison, la figure 15-6 montre la même configuration *sans* la désaturation de l'intégrale.

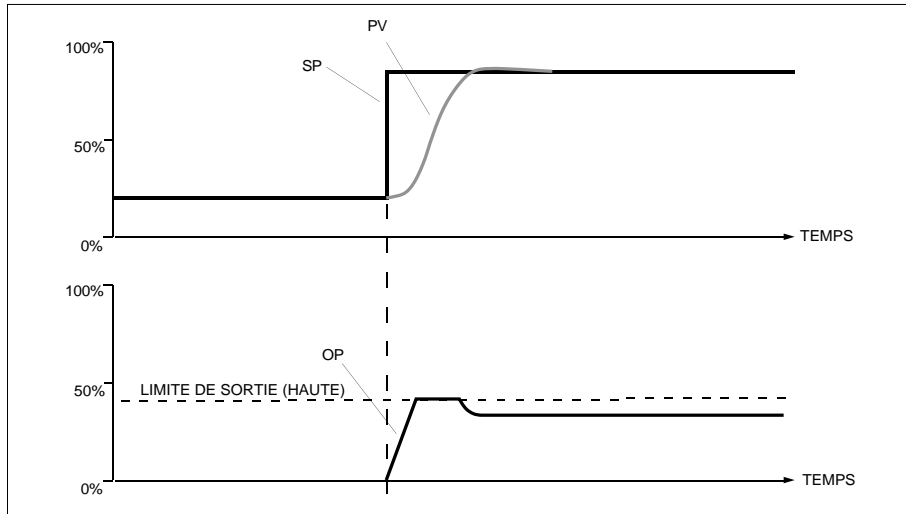


Figure 15-5 Application avec désaturation de l'intégrale (action PID)

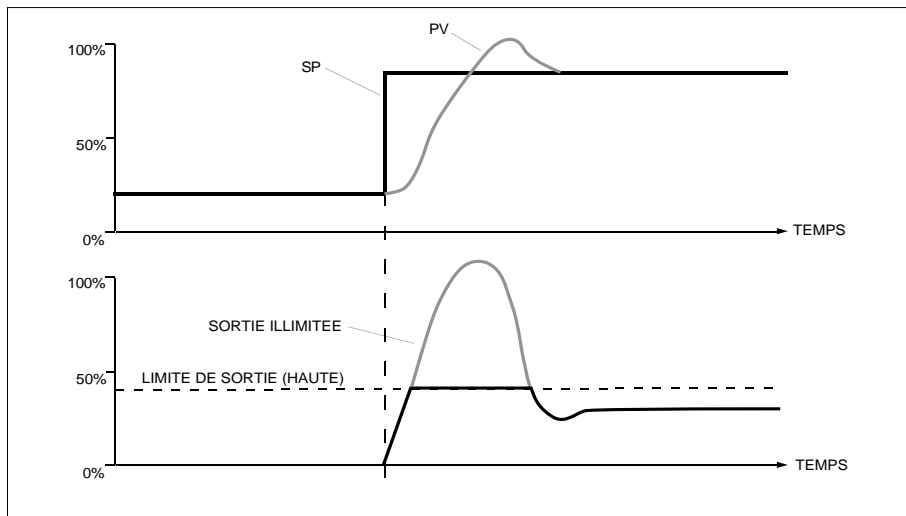


Figure 15-6 Application sans désaturation de l'intégrale (action PID)

Mise en oeuvre mathématique

Définitions des termes

OP	=	sortie PID
XP	=	bande proportionnelle
ER	=	erreur (PV-SP)
TS	=	temps d'échantillonnage (période d'itération de l'algorithme)
TI	=	temps d'action intégrale
TD	=	temps d'action dérivée
FF	=	valeur de tendance
FB	=	valeur de contre-réaction (normalement $FB_n = OP_{n-1}$)
ΔPV	=	$PV_n - PV_{n-1}$ (filtré)
PD	=	terme combiné proportionnelle et dérivée
I	=	terme de l'intégrale

(Les caractères en *indice* indiquent le nombre d'itérations de l'algorithme)

Dans les instruments EUROTHERM SYSTEMES (TCS), l'algorithme de sortie PID est mis en oeuvre en général comme suit :

$$OP = \frac{-100}{XP} \left[ER + \frac{TS}{TI} \sum ER_i + \frac{TD}{TS} \Delta PV \right] + FF$$

L'organigramme de la figure 15-7 montre comment l'équilibrage de l'intégrale et la désaturation de l'intégrale sont effectivement appliqués à l'algorithme PID à chaque itération. Dans le schéma, les termes proportionnelle et dérivée sont combinés pour des raisons de clarté en un terme unique :

$$PD_n = ER_n + \frac{TD_n}{TS_n} \Delta PV_n$$

et le terme standard de l'intégrale est appelé:

$$I_n = I_{n-1} + \frac{TS_n}{TI_n} \cdot ER_n$$

Equilibrage du terme de l'intégrale

La figure 15-7 montre "l'équilibrage" du terme de l'intégrale comme étant:

$$I_n = \frac{-XP_n}{100} [FB_n - FF_n] - PD_n + \frac{TS_n}{TI_n} ER_n$$

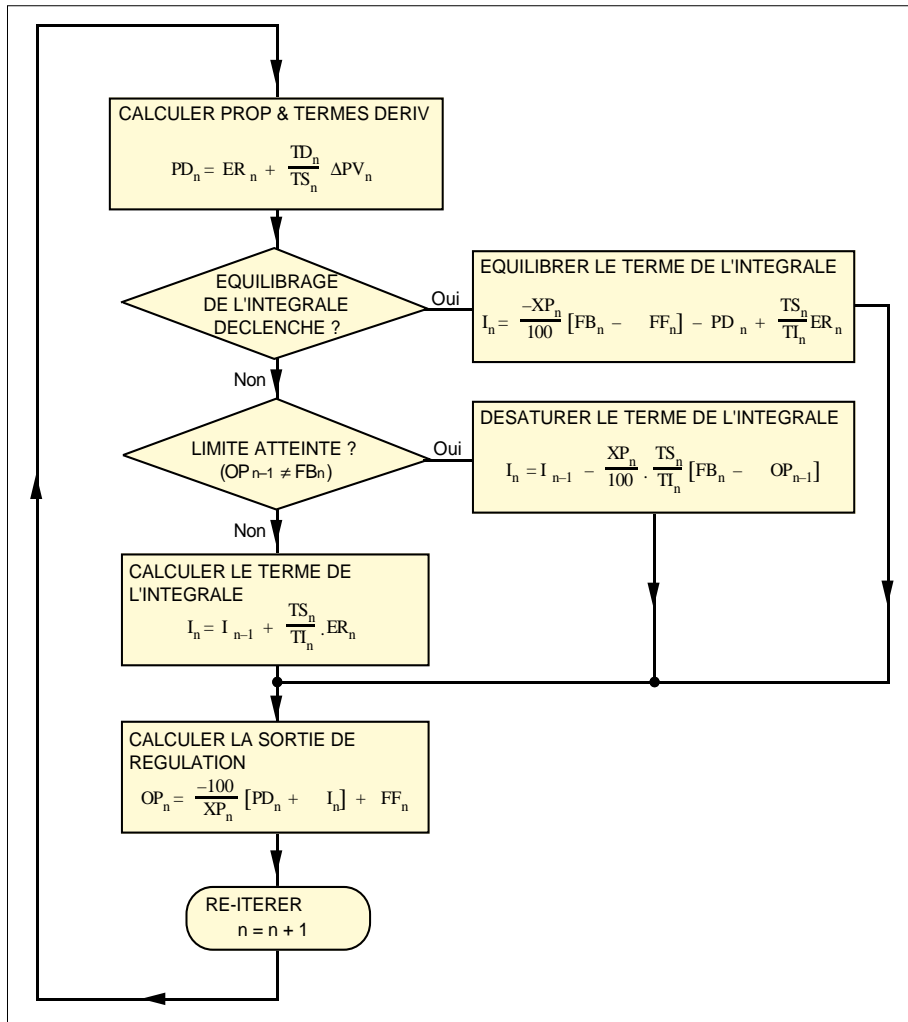


Figure 15-7 Mise en oeuvre de l'équilibrage de l'intégrale & désaturation de l'intégrale

Pour voir l'effet produit sur la sortie calculée, I_n équilibré peut être remplacé dans l'équation définissant OP_n , ce qui donne :

$$OP_n = FB_n - \left[\frac{100}{XP_n} \frac{TS_n}{TI_n} ER_n \right]$$

c'est à dire, OP_n (équilibré) = OP_{n-1} (limité) - l'incrément de l'intégrale

(En raison de la manière dont l'algorithme est exécuté, la contre-réaction à la nième itération, FB_n est calculée à partir de la sortie de l'itération *précédente*, OP_{n-1}). Ainsi, l'équilibrage du terme de l'intégrale produit une nouvelle sortie qui ne diffère de la sortie précédente (limitée) que par l'incrément du terme de l'intégrale, sans contribution des termes proportionnelle ou dérivée.

"Désaturation" du terme de l'intégrale

La figure 15-7 montre que chaque fois que la désaturation de l'intégrale est appliquée, le terme de l'intégrale est réduit de la quantité suivante :

$$\frac{XP_n}{100} \cdot \frac{TS_n}{TI_n} [FB_n - OP_{n-1}]$$

c'est à dire, par une valeur proportionnelle à $FB_n - OP_{n-1}$, c'est à dire la marge par laquelle la sortie calculée dépasse la limite. Dans ce cas, la sortie approche la limite exponentiellement jusqu'à ce que la marge soit suffisamment étroite (<0.006 % approximatif.) pour ne pas déclencher la désaturation de l'intégrale.

Annexe B LA COUCHE D'APPLICATION LIN

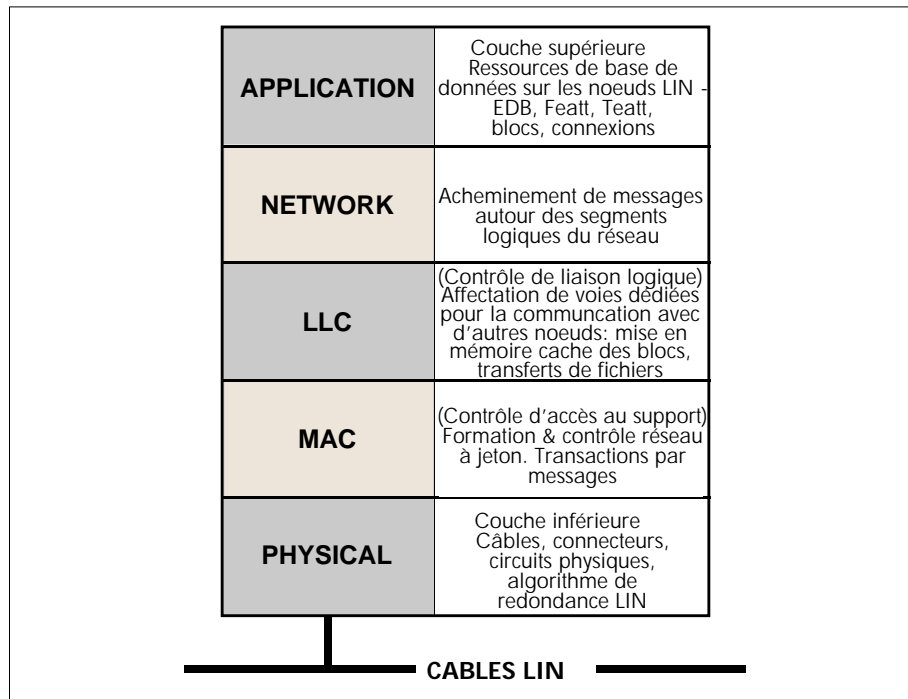


Figure B-1 Structure des couches du réseau LIN

Le réseau LIN (Local Instrument Network - réseau local instruments) est un système complexe dans son intégralité, mais il peut être décomposé en “couches” plus simples. Chaque couche est essentielle pour le bon fonctionnement du LIN, les couches supérieures tablant sur le bon fonctionnement des couches inférieures pour une exploitation normale. La figure B-1 schématise ces couches et récapitule leurs fonctions.

La présente annexe est axée sur la couche supérieure (Application) du modèle, en utilisant une série de schémas qui expliquent ce qui se passe lorsque des blocs de fonction utilisés à titre d'exemple sont mis en mémoire cache sur le réseau LIN pour permettre la communication entre eux. Les étapes illustrées sont les suivantes:

- 1 Configuration d'un bloc en mémoire cache — Etapes initiales (figure B-2)
- 2 Mise en mémoire cache d'un bloc en conduite (figure B-3)
- 3 Mise en mémoire cache d'un second bloc entre les mêmes bases de données (figure B-4)
- 4 Mise en mémoire cache dans la direction opposée (figure B-5)
- 5 Mise en mémoire cache de blocs entre trois noeuds (figure B-6).

1 Configuration d'un bloc en mémoire cache

La figure B-2 montre deux noeuds (instruments) connectés sur le réseau LIN — appelés 'T3000_80' et 'T100_20', respectivement aux adresses 80 et 20. Le but de cette étape est de mettre en mémoire cache le bloc de fonction 'LIC001', qui tourne sur le noeud T100_20, dans la base de données du noeud T3000_80. Il s'agit de créer effectivement sur le noeud T3000_80 une 'image' du bloc déporté pour leur permettre de communiquer sur le réseau LIN.

Les blocs locaux sont représentés avec des contours en gras, alors que les blocs locaux (en mémoire cache) ont des contours moins marqués sur les schémas. Une table **Teatt** (To External Attachment - vers un rattachement externe) et une table **LEDB** (Locally Requested External Database - base de données externe requise localement) sont créées dans la base de données du noeud 80, qui sont enregistrées, lorsque la base de données est sauvegardée. Rien ne se produit encore dans la base de données T100_20.

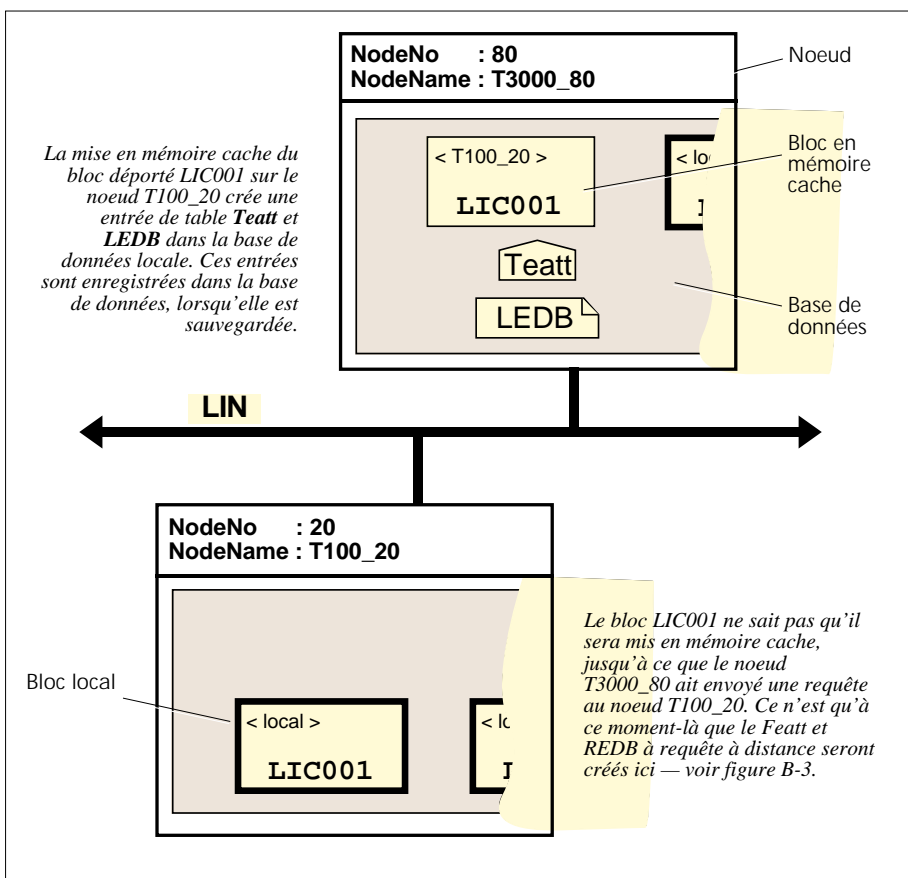


Figure B-2 Configuration d'un bloc en mémoire cache — étapes initiales

2 Mise en mémoire cache d'un bloc en conduite

La figure B-3 montre ce qui se produit, lorsque les bases de données configurées sont exécutées.

La base de données T100 reçoit une requête du T3000 pour mettre en mémoire cache le bloc LIC001. Le T100 réagit en créant une entrée de table **REDB** (Remotely Requested External Database - base de données externe requise à distance) et une entrée de table **Featt** (From External Attachment) dans sa base de données. Ces structures permettent aux données de circuler du bloc LIC001 réel vers son "image" déportée en mémoire cache dans le T3000.

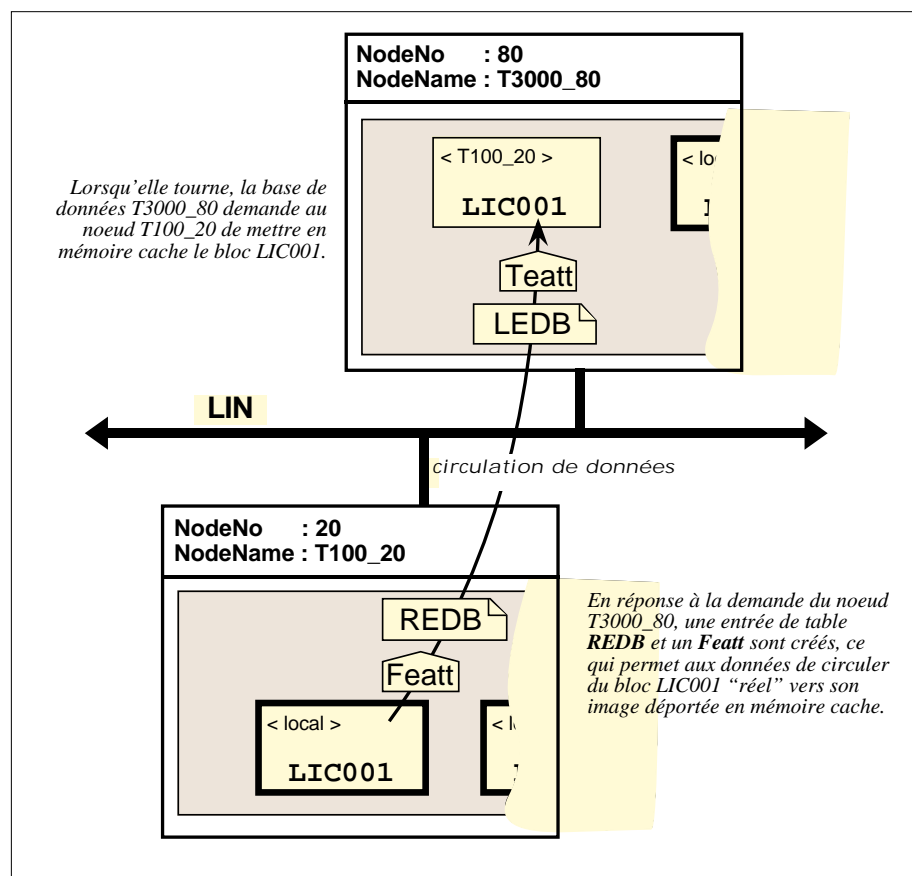


Figure B-3 Mise en mémoire cache d'un bloc en conduite

3 Mise en mémoire cache d'un second bloc

La figure B-4 montre que la base de données du T3000 a été configurée pour mettre un autre bloc en mémoire cache — FIC002 — sur le même noeud déporté T100. Ceci crée un second Teatt, mais aucune autre entrée LEDB n'est nécessaire, dans la mesure où il en existe déjà une vers la base de données déportées du T100. Cette LEDB peut être partagée par tous les blocs en mémoire cache entre les deux noeuds.

En conduite, le T100 réagit à la requête de mettre le bloc FIC002 en mémoire cache, en créant un nouveau Featt pour permettre à FIC002 de transmettre des données. Notez que la liaison EDB traite toutes les données des blocs en mémoire cache entre les deux noeuds, mais plusieurs Featt et Teatt sont nécessaires.

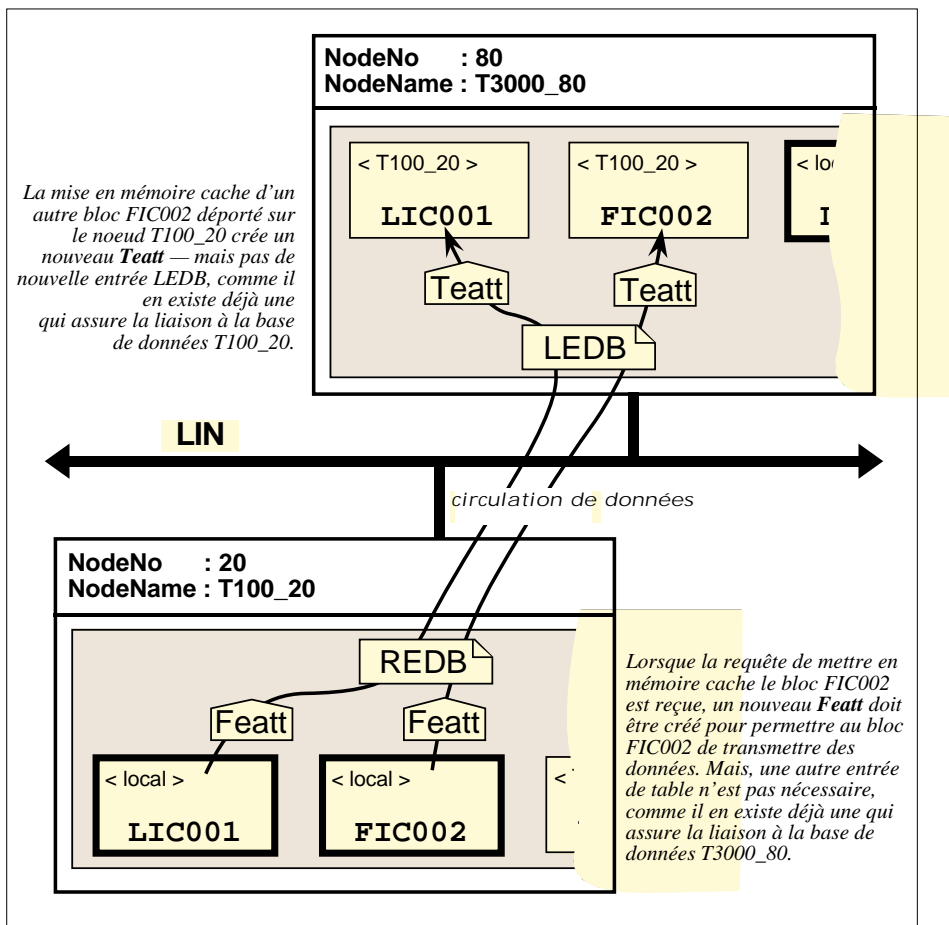


Figure B-4 Mise en mémoire cache d'un second bloc entre les mêmes bases de données

4 Mise en mémoire cache d'un autre bloc dans la direction opposée

La figure B-5 montre ce qui se produit, lorsqu'un bloc de la base de données du T3000_80 est en mémoire cache sur le noeud T100_20 — autrement dit, dans la direction opposée.

Dès que la base de données du T100 est configurée avec le bloc en mémoire cache LRA_80, une entrée Teatt et LEDB est créée comme d'habitude et sauvegardée dans la base de données. Les LEDB peuvent traiter la circulation des données des blocs en mémoire cache dans les deux directions, un REDB n'est donc pas nécessaire en conduite.

De même, un REDB n'est pas nécessaire sur le noeud T3000 (mais, il faut un Featt) en réponse à la requête de mettre en mémoire cache LRA_80, parce que la LEDB existante peut traiter la circulation bidirectionnelle des données.

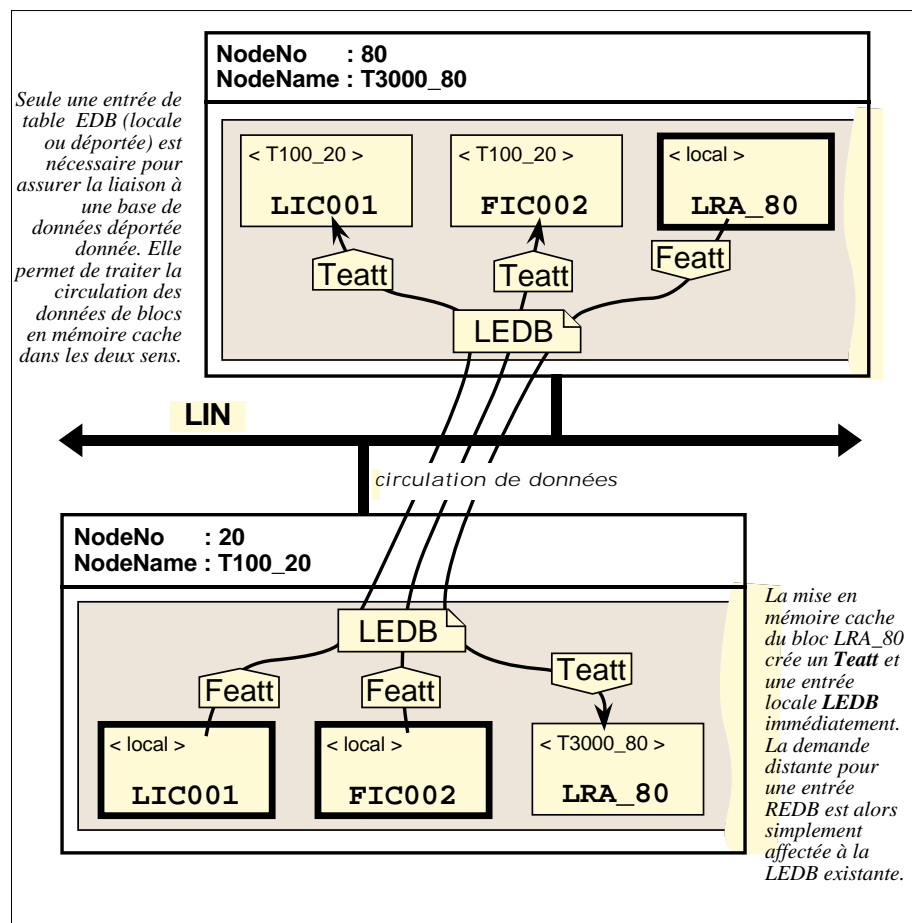


Figure B-5 Mise en mémoire cache d'un autre bloc dans la direction opposée

5 Mise en mémoire cache de blocs entre trois noeuds

La figure B-6 montre qu'un troisième noeud a été connecté au réseau LIN — le T100_60 à l'adresse 60. Ce noeud a été configuré pour mettre en mémoire cache le bloc LIC001 sur le noeud T100_20, autrement dit, ce bloc est mis en mémoire cache simultanément sur deux noeuds différents. Comme d'habitude, une nouvelle entrée de table Teatt et LEDB est créée au moment de la configuration et sauvegardée dans la base de données du T100_60.

En conduite, la requête du T100_60 de mettre en mémoire cache le bloc entraîne la création d'une nouvelle Featt et REDB dans le T100_20. Une Featt est nécessaire pour chaque "image" cache du bloc demandé, et l'entrée REDB est nécessaire pour permettre la circulation des données vers le nouveau noeud.

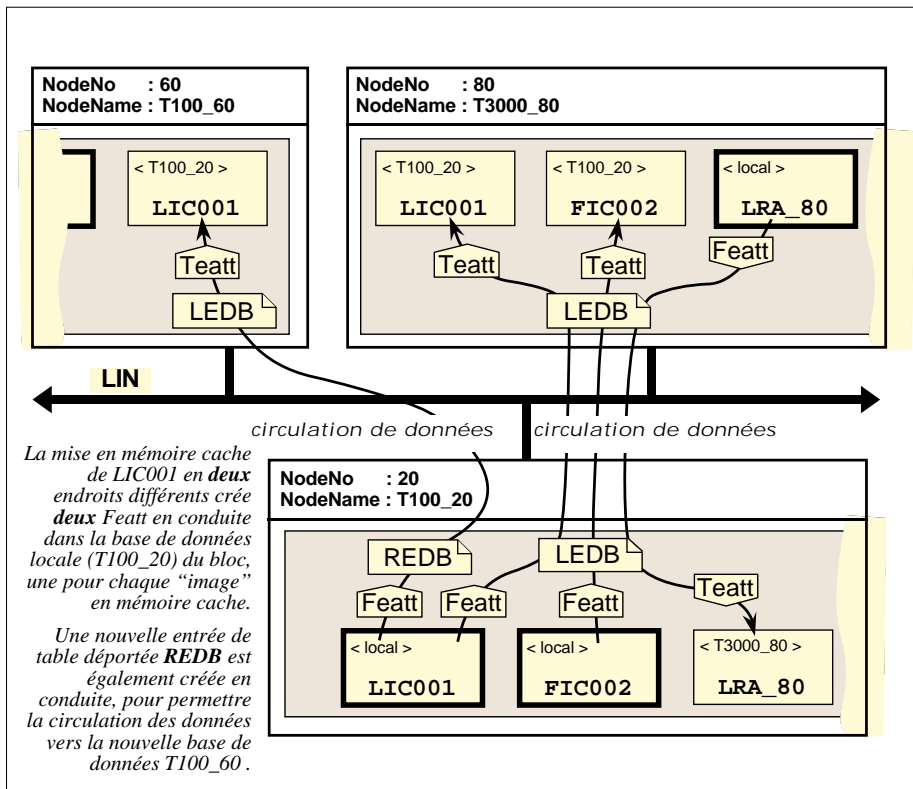


Figure B-6 Mise en mémoire cache de blocs entre trois noeuds

Symboles

.STO, fichier	9-10
.UYM	15-4
.YUP	15-6
0832	
bloc modèle de pseudo-instrument	3-45
0832 pseudo-instruments	3-31
2 voies	
bloc module entrée analogique	15-21
bloc module sortie analogique	15-23
2000	
instruments série	15-8
221TUNE, bloc	12-35
2500	
bloc instrument	15-44
bloc boucle	15-12
bloc mise au point boucle	15-15
types de modules	15-20
bloc alarme utilisateur	15-41
2OF3VOTE, bloc	7-9
3_TERM, bloc	5-30
32 bits	
valeurs de paramètre	6-35
4 voies,	
bloc module entrée analogique	15-21
bloc module sortie analogique	15-23
bloc module entrée logique	15-25
bloc module sortie logique	15-27
bloc impulsions logiques	2-54
6 voies,	
bloc module d'entrée logique	15-25
6255	
multiplexeur de communication	3-1
6360	
bloc modèle de régulateur	3-55
6366, bloc ext. modèle régulateur	3-62
6366, bloc modèle de régulateur	3-49
6432 AI, bloc	3-13
6432 AO, bloc	3-18
6432 DI, bloc	3-22

6432 DO, bloc	3-26
6437	
bloc modèle de pseudo-instrument	3-47
8 voies,	
bloc paramètre octet hex AB	15-42
bloc sortie analogique	2-68
bloc paramètre booléen	15-42
bloc entrée logique	2-12, 2-42
bloc module entrée logique	15-25
bloc sortie logique	2-15, 2-46
bloc paramètre double entier	15-42
bloc paramètre entier	15-42
bloc paramètre nombre réel	15-42
bloc paramètre durée	15-42
bloc paramètre entier sans signe	15-42
bloc paramètre mot/sous-champ	15-42

A

AB, octet hex (8 bits)	
ABCD, mot hex (16 bits)	
A-coups	A-5
Acheminement	
bloc diagnostic diffusion	12-48
bloc diagnostic table	12-32
bloc table	12-31
Acquittement des alarmes	2-5
Action	9-10
ACTION, bloc	9-7
ADD2, bloc	9-1
AGA8DATA, bloc	4-29
AGA8DIAG, bloc	12-84
AI_CALIB, bloc	2-62
Alarm(e)s	2-5
Alarme combinée	2-6
Alarme logiciel	2-5
Alarmes	
auto-acquittées	2-5
Alarmes à "acquitter"	2-5
ALC, bloc	7-5

Algorithme de mise au point		
période de mise à jour	12-19	
ALH_DIAG, bloc	12-75	
ALIN	12-52	
ALIN MAC/LCC	12-27	
ALINDIAG, bloc	12-27	
AMC_DIAG	12-27	
AN_ALARM, bloc	4-11	
AN_CONN, bloc	5-18	
AN_IP, bloc	2-32	
AN_OUT, bloc	2-42	
AN8_OUT, bloc	2-72	
Analogique(s)		
bloc modèle d'entrée	3-13	
bloc de liaison de sortie	3-69	
bloc de liaison d'entrée	3-65	
bloc d'alarme	4-11	
bloc de liaisons	5-18	
bloc de réception	13-7	
bloc de sorties	2-9, 2-42	
bloc de transmission	13-4	
bloc d'entrées .. 2-1, 2-33, 2-79, 15-29		
bloc d'étalonnage des entrées	2-62	
bloc d'étalonnage des sorties	2-68	
bloc d'inversion	4-1	
bloc modèle de sortie	3-18	
bloc modèle d'entrée	3-33	
bloc modèle sortie	3-36	
bloc station manuelle	5-11	
AND4, bloc	8-4	
ANIN, bloc	2-1	
ANIN_6, bloc	2-79	
ANMS, bloc	5-11	
Anonyme, bloc module	15-19	
AO_CALIB, bloc	2-68	
AREA, bloc	16-3	
AREA, blocs	16-1	
AUTO	A-1	
B		
Bande proportionnelle	A-6	
Base de données, bloc de diagnostic	12-15	
Bases de données externes		
bloc de diagnostic	12-19	
bloc table	12-28	
BAT_CTRL, bloc	14-21	
BATCH, blocs de fonction	14-1	
BCS_DIAG, bloc	12-48	
Binaire/hexadécimal/décimal		
tableau de conversion	1-1	
Blocs	2-3	
catégories	1-1	
Bloc de calcul AGA8	4-29	
Bloc de consignation d'acheminement	12-37	
Bloc de contrôle de redondance	12-56	
Bloc de débit corrigé	4-21	
Bloc diagnostic files d'attente PRMT ..	12-64	
Bloc de diagnostic DTU	12-21	
Bloc de diagnostic Modbus	12-68	
Bloc ressources échge données série	12-59	
Bloc statistiques module communicat.	12-66	
Bloc performances bus série interne	12-33	
Bloc statistiques d'acheminement	12-47	
Blocs de fonction		
catégories		
prise en charge instruments	1-1	
Blocs de fonction E/S	2-1	
Blocs de régulation	A-1	
Blocs "esclaves"	3-1	
Blocs "maîtres"	3-1	
Blocs, prise en charge instruments	1-1	
BOOL	15-5	
Booléen	15-42	
bloc valeurs calculées	15-39	
bloc paramètre	15-42	
C		
Caractérisation		
bloc	4-3	
type	2-23	
Catégories	1-1	
CHAR, bloc	4-3	
Clapet restricteur	2-56	
Communications	3-2	
COMPARE, bloc	8-12	
CON_DIAG, bloc	12-39	
CON_ENT, bloc	12-40	
CON_TBL, bloc	12-38	
Concentration de gaz naturel		
bloc de données	4-24	
CONDITN		
blocs de fonctions	4-1	
CONFIG, blocs de fonction	10-1	
CONN32AI, bloc	3-65	

CONN32AO, bloc	3-69
Connexion	
bloc de diagnostic	12-39
bloc entrée	12-40
bloc table	12-38
Contrôle adaptatif du gain	A-6
CONVERT, blocs de fonction	19-1
COUNT, bloc	8-9

D

D2000, bloc	15-46
D25_AI, bloc	15-48
D25_AI_T, bloc	15-61
D25_AI2, bloc	15-21
D25_AI3, bloc	15-21
D25_AI4, bloc	15-21
D25_AICH, bloc	15-29
D25_AO, bloc	15-52
D25_AO2, bloc	15-23
D25_AO4, bloc	15-23
D25_AOCH, bloc	15-32
D25_B_CV, bloc	15-39
D25_DI, bloc	15-55
D25_DI_T, bloc	15-67
D25_DI4, bloc	15-25
D25_DI6, bloc	15-25
D25_DI8, bloc	15-25
D25_DICH, bloc	15-34
D25_DO, bloc	15-58
D25_DO4, bloc	15-27
D25_DOCH, bloc	15-36
D25_LOOP, bloc	15-12
D25_MOD, bloc	15-19
D25_R_CV, bloc	15-38
D25_R_UV, bloc	15-40
D25_RAMP, bloc	15-17
D25_TUNE, bloc	15-15
D25_UALM, bloc	15-41
D2500, bloc	15-44
D25eLOOP, bloc	15-12
D25eRAMP, bloc	15-17
D25eTUNE, bloc	15-15
D2X_LOOP, bloc	15-8
D2X_TUNE, bloc	15-10
DB_DIAG, bloc	12-15

DCM, blocs de fonction	
DCM_B8, bloc	15-42
DCM_D8, bloc	15-42
DCM_I8, bloc	15-42
DCM_R8, bloc	15-42
DCM_S8, bloc	15-42
DCM_T8, bloc	15-42
DCM_UI8, bloc	15-42
DCM_US8, bloc	15-42
DCM_W8, bloc	15-42
DCM_Y8, bloc	15-42
DDR_DIAG, bloc	12-76
Décalage FF	A-4
Décimal/hexadéciml/binaire	
tableau de conversion	1-1
DELAY, bloc	6-25
Dents de scie	
effet	A-8
Dépassements de PV	A-8
“Désaturation” du terme de l’intégrale	A-12
DEVICES sous-répertoire	9-10
DG_CONN, bloc	5-20
DG_IN, bloc	2-46
DG_IN1, bloc	2-79
DG_IN8, bloc	2-12, 5-19
DG_OUT, bloc	2-50
DGIN_8, bloc	2-46
DGMS, bloc	5-13
DGOUT_8, bloc	2-15
DGPULS_4, bloc	2-54
DIAG, blocs de fonction	12-1
Diagnostic réseau, bloc	12-36
DIGACT, bloc	9-10
DIGALARM, bloc	4-14
Digital control algorithm	A-3
DINT	15-5
DINT_X	15-5
DISCREP, bloc	14-15
DIV2, bloc	9-1
DR_ALARM, bloc	17-7
DR_ANCHP, bloc	17-2
DR_DGCHP, bloc	17-5
DR_REPT, bloc	17-9
DTIME, bloc	6-12
DTU, bloc de diagnostic	12-21
DTU_DIAG, bloc	12-21

E

E2PROM	5-22
EDB_DIAG, bloc	12-19
EDB_TBL, bloc	12-28
Entrées en millivolts de faible niveau	2-4
ENUMENUM, bloc	19-1
ENUMUINT, bloc	19-3
Equilibrage de l'intégrale	A-4
Equilibrage du terme de l'intégrale	A-10
ER	A-3
Erreur	A-7
Etalonnage	
bloc	2-62
EXPR, bloc	9-3

F

Face avant, bloc de diagnostic	12-74
Featt	B-3
FF, décalage	A-4
Files d'attente	12-64
FILTER, bloc	4-7
Filtrage du premier ordre	A-3
FLOWCOMP, bloc	4-21
Fonction, blocs	
catégories	1-1
prise en charge instruments	1-1
Forced Auto (auto forcé)	A-1
Forced Manual (manuel forcé)	A-1
FREQIN, bloc	2-26
Fréquence	
bloc d'entrée	2-26
séquence étal. entrées analogiques ..	2-67
FT_TUNE, bloc	12-62
FTQ_DIAG, bloc	12-64
FULL_TC8, bloc	2-21
FWD_DIAG, bloc	12-47
FWD_LOG, bloc	12-37

G

GASCONC, bloc	4-24
GEN_COMM, bloc	3-28
Gestion de la redondance	
des processeurs	12-56
GROUP, bloc	16-6
voies pour le prétracé T2900	16-7
GROUP, blocs	16-1, 16-3

Group, modes d'affichage	16-3
Group, vue d'ensemble	16-3

H

Haut niveau	
séquence d'étalonnage des entrées ..	2-65
séquence d'étalonnage des sorties ..	2-71
Heure du jour	12-54
Hexadécimal/décimal/binaire	
tableau de conversion	1-1
Hex, octet (8 bits)	15-42
Hex, mot (16 bits)	15-42
HIST, bloc	11-1
HOLD (maintien)	A-1
Horloge temps réel	12-54

I

ICM (module de communication	
interprocesseur)	12-66
ICM_DIAG, bloc	12-66
IDLC (supervision de la liaison de données	
entrées	12-59
Instruments du système 6000	3-1
INT	15-5
Interrogation	
bloc diagnostic des installations ..	12-43
bloc entrées - installations	12-45
bloc table	12-42
INVERT, bloc	4-1
ISB	
bloc d'extension de diagnostic	12-34
ISB_DEXT, bloc	12-34
ISB_DIAG, bloc	12-33
ISE_DIAG, bloc	12-72
ISE_TUNE, bloc	12-88

L

Laplace, transformation	A-3
LATCH, bloc	8-7
LEAD_LAG, bloc	4-16
LEADLAG, bloc	4-9
LEDB	B-2
LGROUP, bloc	16-2, 16-12
Liaisons	
bloc de diagnostic	12-39
bloc entrée	12-40
bloc table	12-38

Limite, détection	A-8
LIN	12-52
& défaut base de données	12-50
adresse de station	12-7
bloc algorithme de redondance	12-1
bloc de diagnostic	12-10
bloc protocole noeud	12-46
commutation des voies	12-2
diagnostic haut niveau	
bloc d'extension	12-17
diffusion	12-6
état	12-1
état de fonctionnement	12-9
matériel	12-2
modes de fonctionnement	12-3
segment	12-51
statistiques	12-17
suppression d'une erreur	12-3
voie utilisée	12-9
LIN, couche d'application	B-1
LIN_DEXT, bloc	12-17
LIN_DIAG, bloc	12-10
Local	
accès aux fichiers	14-12
bloc	2-3
réseau instruments (LIN)	12-10
LOG, bloc groupe	16-12
LOG, bloc d'extension du groupe	16-14
LOGDEV, bloc	16-2, 16-9
LOGIC, blocs de fonction	8-1
Logiciel	
alarme	2-5
Logique	
bloc d'alarme	4-14
bloc de liaisons	5-20
bloc de réception	13-9
bloc de transmission	13-6
bloc modèle de sortie	3-26, 3-42
bloc modèle d'entrée	3-22, 3-39
bloc station manuelle	5-13
bloc d'alarme	4-14
LOGRPEX, bloc	16-14
LPTDEV, bloc	16-15
LRA, bloc	12-1

M

MAINTIEN (HOLD)	A-2
MAN_STAT, bloc	5-38
MANUAL (manuel)	A-1
MDBDIAG, bloc	12-68
MDTUNE, bloc	12-13
Message, opérateur	20-5
Messages	20-7
Mise au point	A-5
Modbus	
bloc de diagnostic	12-68
fichier de définition de profil	15-4
MODE, bloc	5-43
Mode de fonctionnement du LIN	12-1
Mode de régulation actif	A-1
Mode Déporté (Remote)	A-2
mode ISOLATED	
isolé	12-54
Mode MASTER	
maître	12-54
Mode "sélectionné"	A-1
Mode SLAVE	
esclave	12-54
Modes de reprise (Fallback)	A-1
Module, bloc	
entrée analogique	15-21
sortie analogique	15-23
anonyme	15-19
entrée logique	15-25
sortie logique	15-27
MUL2, bloc	9-1

N

NAK (accusé de réception négatif)	3-39
NET_DIAG, bloc	12-36
Neutralisé	
noeud local	12-54
NODE_MAP, bloc	12-46
Noeud esclave	12-55
Noeud maître	12-54
NOT, bloc	8-6

O

OP	A-3
OP, réaction	A-5
OPERATOR, blocs de fonction	20-1
Opérateur, message	20-5
OR, bloc	8-4
OR4, bloc	8-4
ORGANISE, blocs de fonction	16-1

P

PAGE, bloc	7-7
Paramètre	
bloc	15-42
blocs (DCM)	15-2
matrices	3-1
état	1-1
PGROUP, bloc	16-18
PID	
algorithme	5-30
bloc	5-1
bloc connexion	5-45
bloc de liaison	5-48
équation de contrôle	A-2
PID_CONN, bloc	5-48
PID_LINK, bloc	5-51
PMC_DIAG, bloc	12-81
PNL_ACC, bloc	20-1, 20-17
PNL_CMD, bloc	20-1, 20-2
PNL_DIAG, bloc	12-74
PNL_DLG, bloc	20-1, 20-12
PNL_MSG, bloc	20-1, 20-7
Point/boucle, vue	16-3
POL_DIAG, bloc	12-43
POL_ENT, bloc	12-45
POL_TBL, bloc	12-42
POURSUITE (TRACK)	A-2
Priorité	A-1
PRMT	12-56
Procédures d'étalonnage	2-64, 2-70
Processeur, redondance	
tâche de gestion	12-56
Profibus	
fichiers de définition de profil	15-6
bloc de diagnostic de ligne	12-81
PROGRAMMER, blocs de fonction	18-1
PS_TUNE, bloc	12-79
PULSE, bloc	8-1
PV, dépassements	A-7

Q

Qualificateur de l'action	9-12
---------------------------------	------

R

Rampe, blocs (DCM)	15-1
RANGE, bloc	4-19
Rate	
bloc alarme	6-28
bloc limite	6-32
RATE_ALM, bloc	6-28
RATE_LMT, bloc	6-32
RCP_LINE, bloc	14-19
RCP_SET, bloc	14-17
READER, bloc	20-1, 20-19
REAL	15-5
REAL_X	15-5
RECORD, bloc	14-10
RECORDER, blocs de fonction	17-1
RED_CTRL, bloc	12-56
REDB	B-3
Régulation	
blocs de fonction	5-1
Régulation proportionnelle	
action de la	A-4
REMOTE (déporté)	A-1
Requêtes d'heure	12-54
Réseau asynchrone TCS	13-1
Réseau non-redondant	12-1
ROUTETBL, bloc	12-31
RS485_NODE, bloc	13-1
RSRCDIAG, bloc	12-86
RTB_DIAG, bloc	12-32
RTD, bloc	2-1
RTD_6, bloc	2-75
RX_AN, bloc	13-7
RX_DG, bloc	13-9

S

S6_DIAG, bloc	12-24
S6000	
bloc de diagnostic	12-24
blocs	3-1
communications	3-2
S6360, bloc	3-4
S6366, bloc	3-8
S6OvrLd	
alarme	3-2

Saturation du terme de l'intégrale	A-7
SDX_RSRC, bloc	12-59
SELECT, bloc	7-1
Sélection des modes de régulation	A-2
SELECTOR, blocs de fonction	7-1
SEQ, bloc	6-1
SEQE, bloc d'extension	6-8
Séquence	
bloc de diagnostic	12-71
SETPOINT, bloc	5-26
SFC	
bloc d'affichage	14-7
bloc de commande	14-1
bloc de supervision	14-5
SFC_CON, bloc	14-1
SFC_DIAG, bloc	12-71
SFC_DISP, bloc	14-7
SFC_MON, bloc	14-5
SIM, bloc	5-15
SINT	15-5
SL0832, bloc	3-45
SL6360, bloc	3-55
SL6360R, bloc	3-77
SL6366, bloc	3-49
SL6366EX, bloc d'extension	3-62
SL6366R, bloc	3-72
SL6432AI, bloc	3-33
SL6432AO, bloc	3-36
SL6432DI, bloc	3-39
SL6432DO, bloc	3-42
SL6437, bloc	3-47
Sonde à résistance,	
bloc d'entrée	2-1
SPP_CTRL, bloc	18-1, 18-2
SPP_DIG, bloc	18-1, 18-4
SPP_EXT, bloc	18-1, 18-7
SPP_RAMP, bloc	18-1, 18-6
SREAL_p1 à p4	15-5
STIME_dh/dm/ds	15-5
SUB2, bloc	9-1
SUM_DIAG, bloc	12-50
SWITCH, bloc	7-3
Symboles	
de l'état du paramètre	1-1
Synchronisation de l'heure	12-54

T

T100	
bloc	10-1
bloc de caractérisation	4-5
bloc de configuration	10-1
bloc performance du	12-13, 12-22
T1000	
bloc de configuration	10-6
bloc performance du	12-12
T102	
bloc performances du	12-62
T102, bloc	10-23
T221	
bloc de mise au point	12-35
T221, bloc	10-21
T221TUNE, bloc	12-35
T231	
bloc performances du	12-22
T231, bloc	10-8
T231TUNE, bloc	12-22
T2900	12-76, 15-1
alarmes	15-41
bloc de diagnostic	12-72
navigation affichage	16-1
enregistrement des données en mémoire flash	16-1
hiérarchie groupe/zone, exemple ..	16-1
programmeur multi points de consigne & consignateur de données	16-1
T2900	
bloc	10-40
bloc de configuration	10-40
T302, bloc	10-28
T600	
bloc de configuration	10-13
bloc performances du	12-29
T600TUNE, bloc	12-29
T800	
bloc	10-40
bloc de configuration	10-40
T932, bloc de configuration	10-33
T940	
bloc	10-35
bloc de configuration	10-35
bloc de performances	12-79

Tableau de conversion	
binaire/hexadécimal/décimal	1-1
TAG, bloc	7-12
TAN	
bloc de configuration du noeud	13-1
bloc de diagnostic	13-10
blocs de fonctions	13-1
TAN_DIAG, bloc	13-10
TCOUPLE, bloc	2-1
TD	A-3
Teatt	B2
Temps mort	A-8
Thermocouple	
bloc d'entrée	2-1, 2-21
TI	A-3
TIME	15-5
TIMEDATE, bloc	6-17
TIMER, bloc	6-15
TIMING, blocs de fonction	
TOD_DIAG, bloc	12-54
TOT_CON, bloc	6-35
TOTAL, bloc	6-9
TOTAL2, bloc	6-37
Totalisation, bloc	6-37
TP_CONN, bloc	5-23
TPO, bloc	6-22
TRACK (poursuite)	A-1
TX_AN, bloc	13-4
TX_DG, bloc	13-6

U

UCHAR, bloc	4-5
UCTUNE, bloc	12-12
UDINT	15-5
UDINT_X	15-5
UNINT	15-5
UINTENUM, bloc	19-2
USINT	15-5

V

Valeur de contre-réaction FB	A-8
Vidéo, enregistreur de données	17-1
Visual Supervisor, agents	20-5

X

XEC_DIAG, bloc	12-49
XOR 4, bloc	8-4
XP	A-3

Z

Zone	17-4
------------	------