Chapter 13

SLAVE_VARS

Edition 3

OVERVIEW

Contents (continued)

Overview

This chapter describes the SLAVE-VARS class of function blocks.

SLAVE parameters are used in association with a suitable SLAVE communications driver. They are used when the PC3000 is the SLAVE device. The blocks may be viewed as fixed addresses or 'letterboxes' into which data may be written or read by a suitable master device e.g. a PC based supervisor. The blocks provide diagnostics, read/write access control etc.

The SLAVE multi-element parameters allow more efficient memory usage since one block can be used to control the transactions associated with several fifferent values. Additionally, there are benefits in speed of initialisation since only one address has to be set to its' cold start value compared with several when using individual parameters.

Slave Parameter
Slv_Bool
Slv_Real
Slv_Dint
Slv_Time
Slv_Str
Slv_Bool_8
Slv_Real_8
Slv_Dint_8
Slv_Bool_64
Slv_Real_64
Slv_Dint_64
Slv_SW

The status word data type is a special data type which packs 16 booleans into a single block. It is used for status type information.

This class is used in conjunction with the COMMS Function Block class.

SLV_BOOL FUNCTION BLOCK

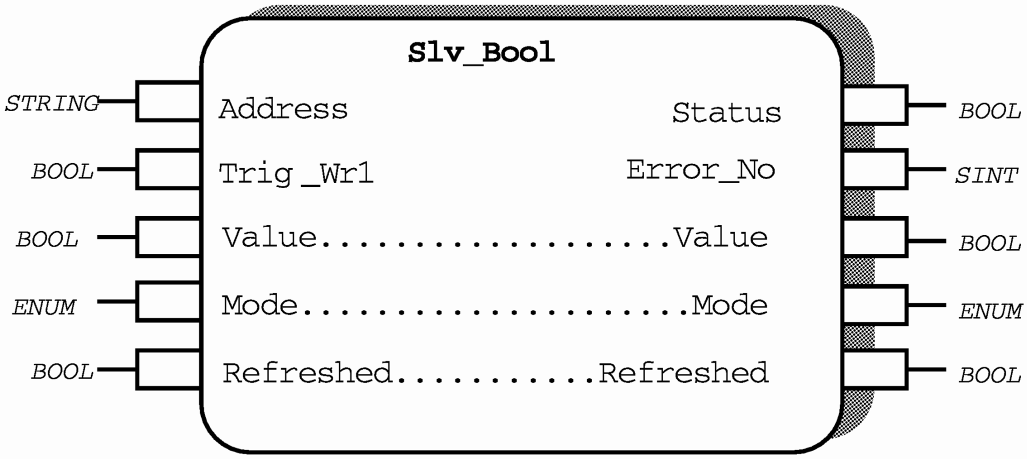
Slv_Bool
STRING
BOOL
BOOL
ENUM
BOOL
Address
Trig_Wr1
Value....................Value
Mode.....................Mode
Refreshed...........Refreshed
Status
Error_No
BOOL
SINT
BOOL
ENUM
BOOL

Figure 13-1 Slv_Bool Function Block

Functional Description

The Slv_Bool function block allows the reading and / or writing of the value of a boolean parameter in the PC3000 by an external device via a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slv_Vars function blocks, see the Communications Overview in chapter 3.

Function Block Attributes

Type:.................................. 3c 10

Class: .... SLV_VARS

Default_Task: .... Task_2

Short List: .... Value, Mode, Refreshed, Status

Memory Requirements: ...... 100 Bytes

Execution Time: .... 122 µ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to access the function block. The rest of the address is protocol specific and is used to by a remote device to identify the variable. Refer to the associated slave communications driver in chapter 3 for details of slave variable addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value (VAL)

Value is the boolean parameter to be read from or written via the communications ports associated with the protocol selected by the Address parameter.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0): All remote read or write access is freely allowed.

R_Cont (1): All remote reads are freely allowed, but remote writes are blocked.

Wr_once (2): Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

This is set to Yes (1) each time the Value is written to via the communications channel. It should be reset by the user program to detect a change in value.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

It is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	**Type**	**Cold Start**	**Read Access**	**Write Access**	**Type Specific Information**	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once(2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value	**BOOL**	Value (0)	Oper	Super	Senses	Value (0) Value (1)

Table 13-1 Slv_Bool Parameter Attributes

SLV_REAL FUNCTION BLOCK

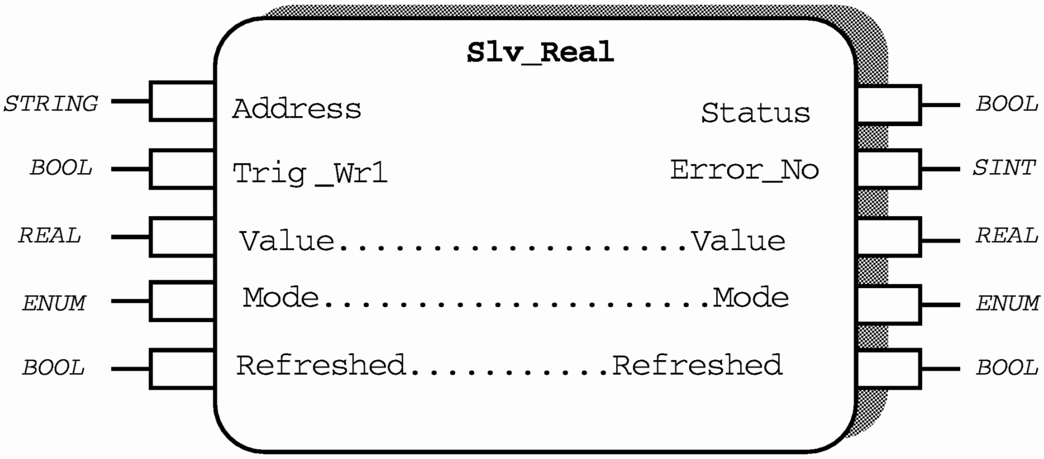
Slv_Real
STRING
BOOL
REAL
ENUM
BOOL
Address
Trig _Wr1
Value..................Value
Mode....................Mode
Refreshed..........Refreshed
Status
Error_No
BOOL
SINT
REAL
ENUM
BOOL

Figure 13-2 Slv_Real Function Block

Functional Description

The Slv_Real function block allows the reading and / or writing of the value of a floating point parameter in the PC3000 by an external device via a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slv_Vars function blocks, see the Communications Overview in chapter 3.

Function Block Attributes

Type: 3c 20

Class:...................................SLV_VARS

Default_Task: ....Task_2

Short List: ....Value, Mode, Refreshed, Status

Memory Requirements: ......106 Bytes

Execution Time; ....128 µ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to access the function block. The rest of the address is protocol specific and is used to by a remote device to identify the variable, etc. Refer to the associated slave communications driver in chapter 3 for details of slave variable addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value (VAL)

Value is the floating point (Real) parameter to be read from or written via the associated communication ports, etc.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0):	All remote read or write access is freely allowed.
R_Cont (1):	All remote reads are freely allowed, but remote writes are blocked.
Wr_Once (2):	Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

Refreshed is set to Yes (1) each time the Value is written to via the communications channel.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

Error_No is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	Type	Cold Start	Read Access	Write Access	Type Specific Information	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once (2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value	**REAL**	0	Oper	Super	High Limit Low Limit	1,000,000 -1,000,000

Table 13-2 Slv_Real Parameter Attributes

SLV_DINT FUNCTION BLOCK

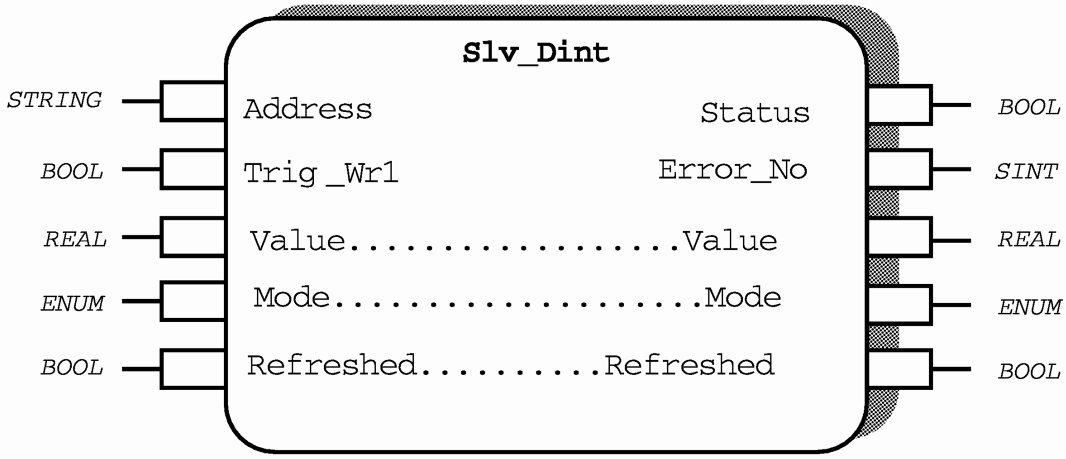
Slv_Dint
STRING
BOOL
REAL
ENUM
BOOL
Address
Trig _Wrl
Value..................Value
Mode...................Mode
Refreshed.........Refreshed
Status
Error_No
BOOL
SINT
REAL
ENUM
BOOL

Figure 13-3 Slv_Dint Function Block

Functional Description

The Slave_Dint function block allows the reading and / or writing of the value of a 32-bit signed integer parameter in the PC3000 by an external devicevia a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slv_Vars function blocks, see the Communications Overview in Chapter 3.

Function Block Attributes

Type:................................. 3c 30

Class: .... SLV_VARS

Default_Task: .... Task_2

Short List: .... Value, Mode, Refreshed, Status

Memory Requirements: ...... 110 Bytes

Execution Time: .... 123 µ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to access the function block. The rest of the address is protocol specific and is used to by a remote device to identify the variable, etc. Refer to chapter 3 for details of slave varaible addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value (VAL)

Value is the integer parameter to be read from or written via the associated communication ports, etc.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0): All remote read or write access is freely allowed.

R_Cont (1): All remote reads are freely allowed, but remote writes are blocked.

Wr_Once (2): Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

Refreshed is set to Yes (1) each time the Value is written to via the communications channel.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

Error_No is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	Type	Cold Start	Read Access	Write Access	Type Specific Information	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once (2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value	**DINT**	0	Oper	Super	High Limit Low Limit	1,000,000 -1,000,000

Table 13-3 Slv_Dint Parameter Attributes

SLV_TIME FUNCTION BLOCK

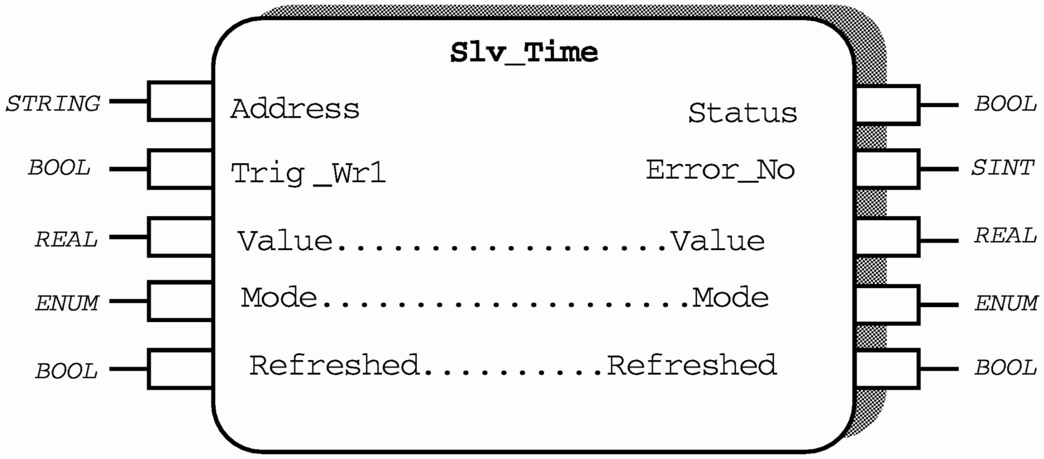
Slv_Time
STRING
BOOL
REAL
ENUM
BOOL
Address
Trig _Wr1
Value..................Value
Mode...................Mode
Refreshed.........Refreshed
Status
Error_No
BOOL
SINT
REAL
ENUM
BOOL

Figure 13-4 Slv_Time Function Block

Functional Description

The Slv_Time function block allows the reading and / or writing of the value of a duration parameter in the PC3000 by an external device via a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slave_Vars function blocks, see the Communications Overview in Chapter 3.

Function Block Attributes

Type: 3c 40

Class:..................................SLV_VARS

Default_Task: ....Task_2

Short List: ....Value, Mode, Refreshed, Status

Memory Requirements: ......106 Bytes

Execution Time: ....123μ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to access the function block. The rest of the address is protocol specific and is used to by a remote device to identify the variable, etc. Refer to the slave communications driver in chapter 3 for details of slave variable addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value (VAL)

Value is the duration (TIME) parameter to be read from or written via the associated communication ports, etc.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0):	All remote read or write access is freely allowed.
R_Cont (1):	All remote reads are freely allowed, but remote writes are blocked.
Wr_Once (2):	Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

Refreshed is set to Yes (1) each time the Value is written to via the communications channel.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

Error_No is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	Type	Cold Start	Read Access	Write Access	Type Specific Information	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once (2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value	**TIME**	0	Oper	Super		

Table 13-4 Slv_Time Parameter Attributes

SLV_STR FUNCTION BLOCK

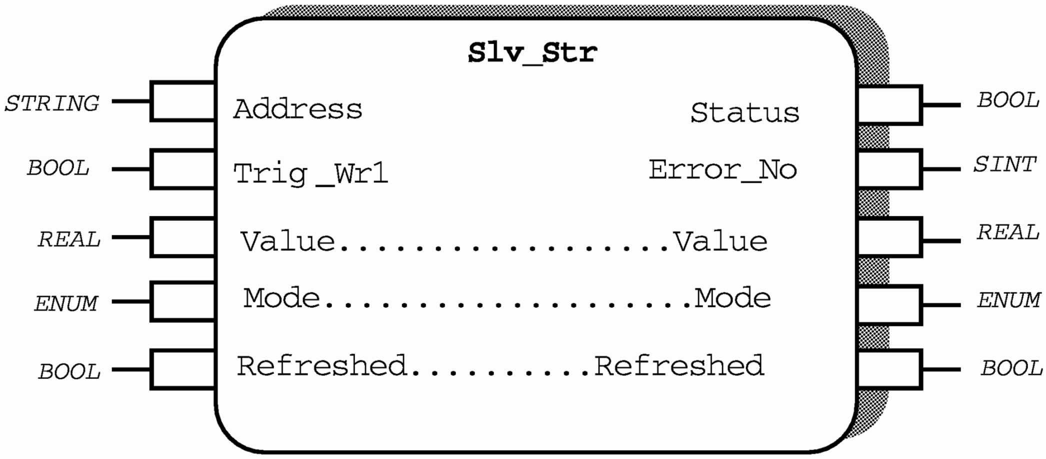
Slv_Str
STRING
Address
BOOL
Trig_Wr1
REAL
Value..................Value
ENUM
Mode...................Mode
BOOL
Refreshed.........Refreshed
Status
BOOL
Error_No
SINT
REAL
ENUM
BOOL

Figure 13-5 Slv_Str Function Block

Functional Description

The Slv_Str function block allows the reading and / or writing of the value of a string parameter in the PC3000 by an external device via a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slave_Vars function blocks, see the Communications Overview in Chapter 3.

Function Block Attributes

Type:	3c 80
Class:	SLV_VARS
Default_Task:	Task_2
Short List:	Value, Mode, Refreshed, Status
Memory Requirements:	486 Bytes
Execution Time:	514 µ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to access the function block. The rest of the address is protocol specific and is used to by a remote device to identify the vaiable, etc. Refer to the associated slave communications driver in chapter 3 for details of slave variable addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value (VAL)

Value is the string (STRING) parameter to be read from or written via the associated communication ports, etc.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0): All remote read or write access is freely allowed.

R_Cont (1): All remote reads are freely allowed, but remote writes are blocked.

Wr_Once (2): Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

Refreshed is set to Yes (1) each time the Value is written to via the communications channel.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

Error_No is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	Type	Cold Start	Read Access	Write Access	Type Specific Information	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once (2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value	**STRING**	' '	Oper	Super		

Table 13-5 Slv_Str Parameter Attributes

SLV_SW FUNCTION BLOCK

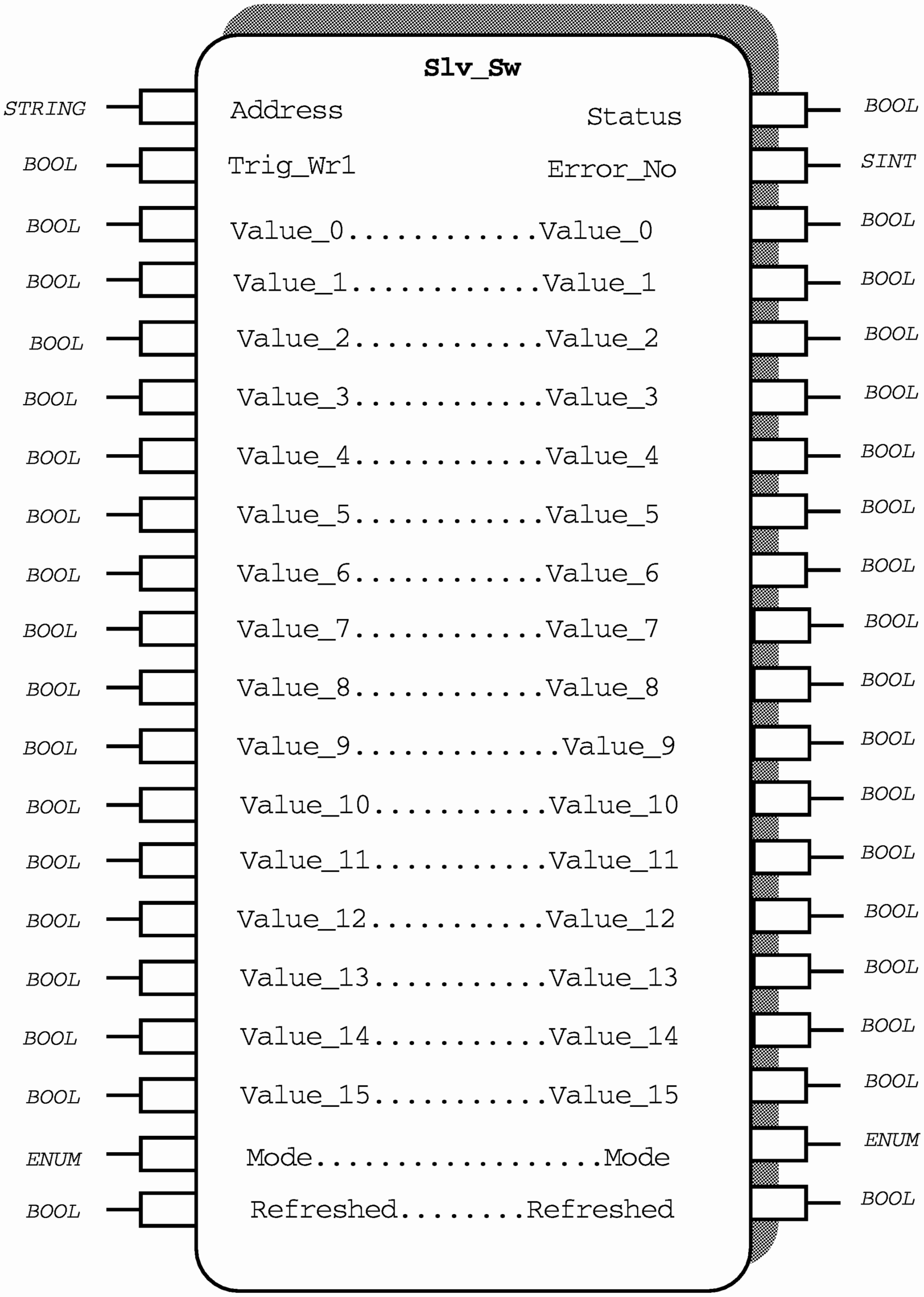
Slv_Sw
STRING
Address
Status
BOOL
BOOL
Trig_Wr1
Error_No
SINT
BOOL
Value_0............Value_0
BOOL
BOOL
Value_1............Value_1
BOOL
BOOL
Value_2............Value_2
BOOL
BOOL
Value_3............Value_3
BOOL
BOOL
Value_4............Value_4
BOOL
BOOL
Value_5............Value_5
BOOL
BOOL
Value_6............Value_6
BOOL
BOOL
Value_7............Value_7
BOOL
BOOL
Value_8............Value_8
BOOL
BOOL
Value_9.............Value_9
BOOL
BOOL
Value_10...........Value_10
BOOL
BOOL
Value_11...........Value_11
BOOL
BOOL
Value_12...........Value_12
BOOL
BOOL
Value_13...........Value_13
BOOL
BOOL
Value_14...........Value_14
BOOL
BOOL
Value_15...........Value_15
BOOL
ENUM
Mode..................Mode
ENUM
BOOL
Refreshed........Refreshed
BOOL

Figure 13-6 Slv_SW Function Block

Functional Description

The Slave Status Word function block , Slv_SW, allows the reading and / or writing of the value of a 16-bit parameter in the PC3000 by an external device via a communications port on the PC3000. The function block is not communications protocol specific, but can only be used when associated with a slave protocol function block, e.g. EI_Bisync_S.

For a full description of the use of Slave_Vars function blocks, see the Communications Overview in Chapter 3.

Function Block Attributes

Type:..................................	3c f0
Class:	SLV_VARS
Default_Task:	Task_2
Short List:	Value_1, Mode, Refreshed, Status
Memory Requirements:	126 Bytes
Execution Time:	136 µ Secs

Parameter Descriptions

Address (A)

The parameter Address is a string which is used to associate the function block with a parameter in the remote device to which the PC3000 is connected. The first two characters of Address defines the protocol which can be used to update the function block. The rest of the address is protocol specific and is used to identify the instrument and parameter that the function block is communicating with. Refer to the associated slave communications driver in chapter 3 for details of slave variable addresses.

Trig_Wr1 (TRW)

On the leading edge of changing Trig_Wr1 from Off (0) to On (1), the Mode is changed to Wr_Once (2). This may be used in soft wiring to create an inter-lock condition. On receipt of a new value the data cannot be overwritten until Trig_Wr1 has been switched back to the Off (0) state.

Value_0 to Value_15 (V0 to V15)

Value_0 to Value_15 are the parameters to be read from or written via the comm's channel, representing bits 0 to 15 respectively.

Mode (M)

The Mode specifies mode of operation of the function block. It can have three possible values:

Rd_Wr (0): All remote read or write access is freely allowed.

R_Cont (1): All remote reads are freely allowed, but remote writes are blocked.

Wr_Once (2): Remote reads are always allowed, but following a remote write the mode will change to Rd_Only (1). This ensures that newly received values are not overwritten.

Refreshed (R)

Refreshed is set to Yes (1) each time the Value is written to via the communications channel.

Status (ST)

If Status is set to Go (1), the previous communications request has been completed successfully. If Status is set to NOGO (0), an error has occured.

Error_No (ERR)

Error_No is a diagnostic parameter which provides an indication to the type of communications error which has occured. Refer to the appropriate communications slave driver in chapter 3 for error number explanation.

Parameter Attributes

Name	Type	Cold Start	Read Access	Write Access	Type Specific Information	
Address	**STRING**		Oper	Config		
Error_No	**SINT**	0	Oper		High Limit Low Limit	255 0
Mode	**ENUM**	Rd_Wr (0)	Oper	Oper	Senses	Rd_Wr (0) Rd_Only (1) Wr_Once (2)
Refreshed	**BOOL**	No (0)	Oper	Super	Senses	No (0) Yes (1)
Status	**BOOL**	NOGO (0)	Oper		Senses	NOGO (0) Go (1)
Trig_Wr1	**BOOL**	Off (0)	Oper	Oper	Senses	Off (0) On (1)
Value_0 to Value_15	**BOOL**	Value (0)	Oper	Super	Senses	Value (0) Value (1)

Table 13_6 Slv_SW Parameter Attributes

SLV_****_8 FUNCTION BLOCK

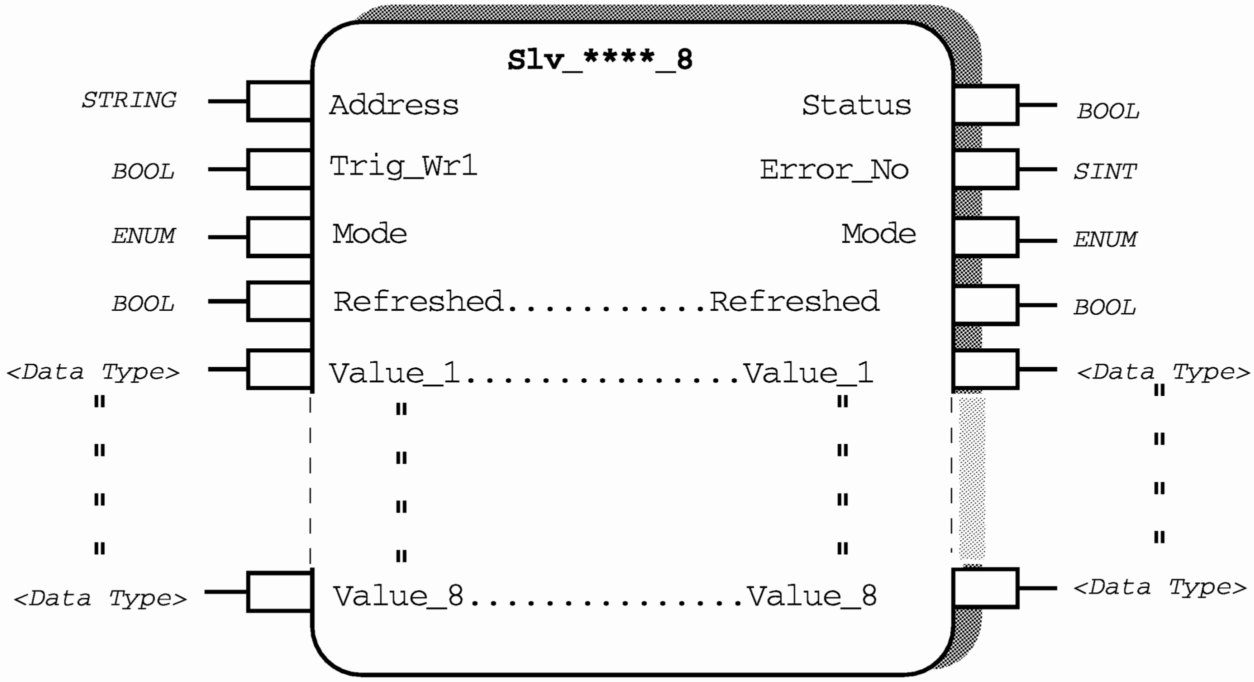

Figure 13-7 Slv_****_8 Function Block

Functional Description

All blocks in this class are described as Slv_****_8 where **** is BOOL, REAL, DINT or STR.

Usage and facilities provided are identical to the Slv_*** function blocks described earlier in this chapter. The primary difference is the provision of multiple Value inputs (Value_1 through Value_8 [Mnemonics V1 - V8].

These function blocks provide an efficient method for accessing multiple parameters using a single read or write transaction. The blocks can access 8 contiguous locations. The data to be read or written is accessed using a single composite parameter. The order of data within the composite parameter is the same as the displayed order i.e. Value_1 to Value_8.

The blocks allow more efficient use of memory, since only one address is defined and there is only one set of control and status parameters for 8 data points.

These blocks may be used as an alternative to packing and unpacking data into strings using the COMPACT functions provided in the Structured Text language.

SLV_****_64 FUNCTION BLOCK

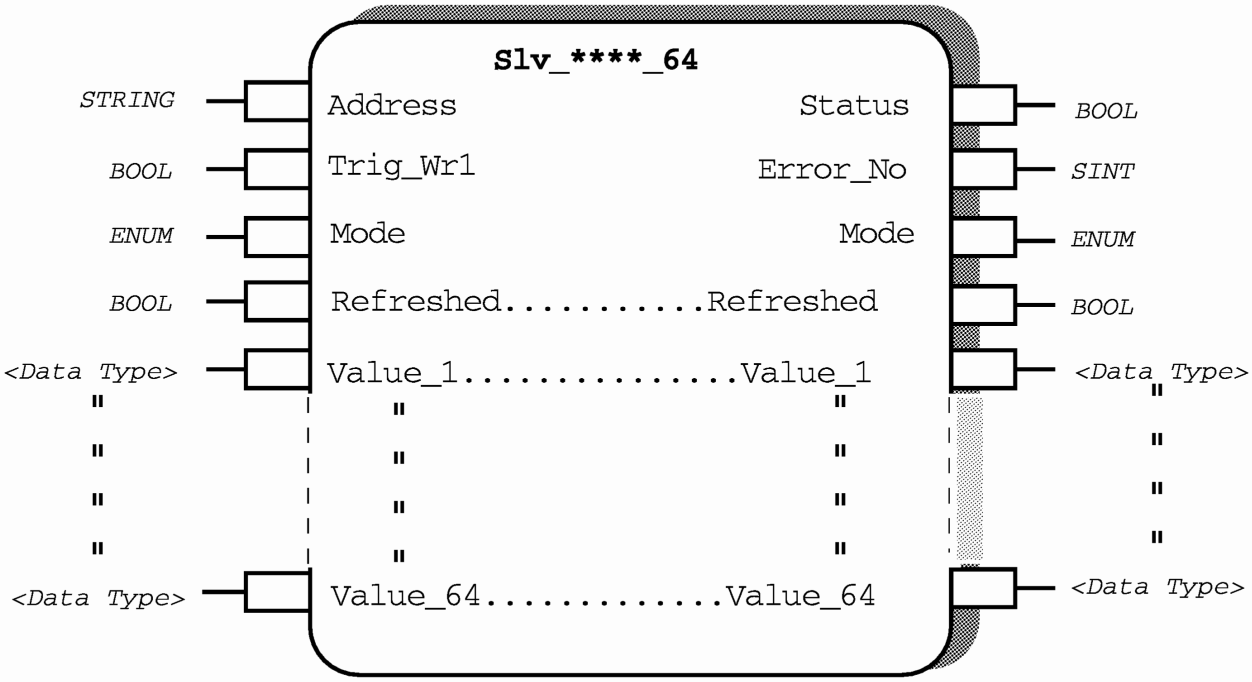

Figure 13-8 Slv_****_64 Function Block

Functional Description

All blocks in this class are described as Slv_****_64 where **** is BOOL, REAL, DINT or STR.

Usage and facilities provided are identical to the Slv_*** function blocks described earlier in this chapter. The primary difference is the provision of multiple Value inputs (Value_1 through Value_64 [Mnemonics V1 - V64].

These function blocks provide an efficient method for accessing multiple parameters using a single read or write transaction. The blocks can access 64 contiguous locations. The data to be read or written is accessed using a single composite parameter. The order of data within the composite parameter is the same as the displayed order i.e. Value_1 to Value_64.

The blocks allow more efficient use of memory, since only one address is defined and there is only one set of control and status parameters for 64 data points.

These blocks may be used as an alternative to packing and unpacking data into strings using the COMPACT functions provided in the Structured Text language.