electrical & safety · industrial connectivty · software & electronics · advanced connectivity

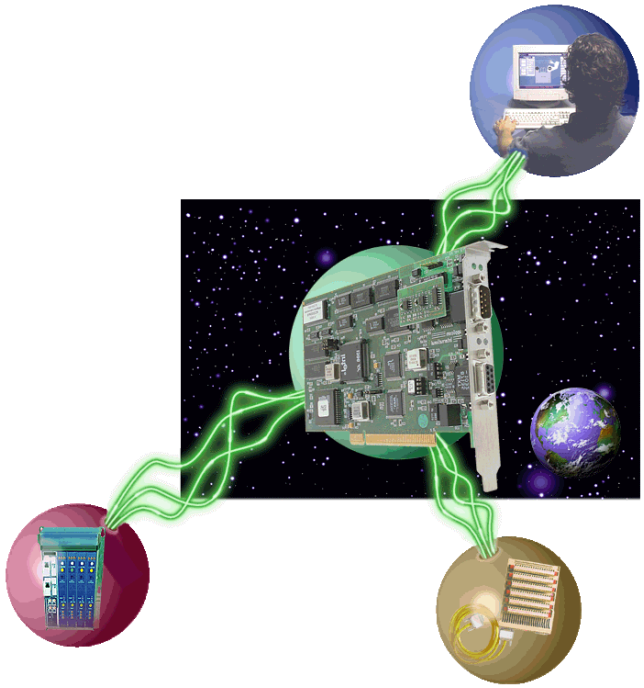
Woodhead

Protocol manual

SAIA-SBUS

applicom® 3.7

a product of Woodhead Software & Electronics



http://www.applicom-int.com

Table of contents

1. - Functionality

- Introduction

The ***applicom*®** interface channel configured in SAIA SBUS is SBUS master. The interface is configurable in 'break without parity' mode or in 'parity mark and space' mode.

PCD PLC Variable	***applicom*® Addressing (Addr)**	**Exchange type (cyclic mode)**	**Corresponding *applicom*® function (library/DLL access)**
Flag Fx	x	Readbits Write bits	READPACKBIT, READDIFBIT WRITEPACKBIT, WRITEDIFPACKBIT
Input Ix	x	Read input bits	READPACKIBIT, READDIFIBIT
Output Ox	x	Read output bits Write output bits	READPACKQBIT, READDIFQBIT WRITEPACKQBIT, WRITEDIFPACKQBIT
Register Rx	x	Read double words Write double words Read floating words Write floating words	READDWORD, READDIFDWORD WRITEDWORD, WRITEDIFDWORD READFWORD, READDIFFWORD WRITEFWORD, WRITEDIFFWORD
Data block DByEz	y * 65536 + z + 65536	Read double words Write double words Read floating words Write floating words	READDWORD, READDIFDWORD WRITEDWORD, WRITEDIFDWORD READFWORD, READDIFFWORD WRITEFWORD, WRITEDIFFWORD
Timer Tx	x		READTIMER WRITETIMER
Counter Cx	x		READCOUNTER WRITECOUNTER

x : Variable number.
y : Data Block number.
z : Element number in the DB.

Remark:
The PLC counters and timers being on 32 bits and the ***applicom*®** primitives handling 16 bit counters and timers, the value 65535 is returned if a counter or timer exceeds this value.

The counters and timers are only available with the ***applicom*®** library.

- Maximum number of variables per frame with the library

Object	Max. quantity in read	Max. Quantity in write
Bit	128	128
Double word	32	32
Floating word	32	32
Timer	32	32
Counter	32	32

- Maximum number of variables per frame with PCDDE

Object	Max. quantity in read	Max. Quantity in write
Bit	128	1
Double word	32	1
Floating word	32	1

The number given for read frames corresponds to the maximum number of points (as imposed by the server and/or the protocol) which can be grouped together during dynamic optimization of the frames carried out by the server. However, this number can be reduced to suit a specific item of equipment by configuring the length of frames in the topic (see chapter "Implementation/Topics configuration/Advanced options").

Where write operations are concerned, a variable automatically entails the formation of a frame.

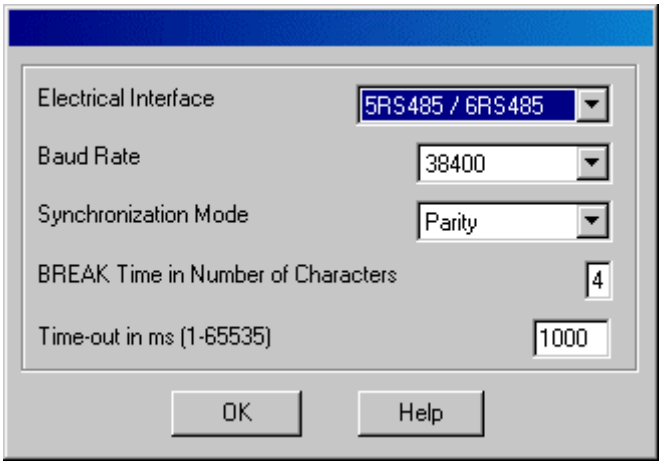
- Maximum number of variables per frame with OPC server

Object	Max. quantity in read	Max. Quantity in write
Bit	128	128
Double word	32	32
Floating word	32	32

The number given for read frames corresponds to the maximum number of points (as imposed by the server and/or the protocol) which can be grouped together during dynamic optimization of the frames carried out by the server. However, this number can be reduced to suit a specific item of equipment by configuring the length of frames in the topic (see chapter "Implementation/Topics configuration/Advanced options").

For the write frames, see chapter "OPC Server/Optimization of synchronous and asynchronous requests".

2. - Configuration



Electrical interface :
The interface module type is determined by the reference of the module installed on the ***applicom***® interface or on the galvanic isolation option (BX4010, BX4010R).

You have then the following options:
RS232 (2 signals) (3RS232*, 5RS232*, 6RS232)
RS232 (6 signals) (3RS232*, 5RS232*, 6RS232)
BC20 mA*
3RS485*
3RS485-4*
5RS485* or 6RS485

*** : <u>Not galvanically isolated. Isolation only ensured with option BX4010 and BX4010R</u>.**

For the module compatibility, you can refer to the Wiring/Output module compatibility section of « hardware installation » manual.

RS232 (2 signals)(3RS232*, 5RS232*, 6RS232)
Means that only the TxD and RxD signals are exploited by the protocol task installed on the channel. RTS, DTR, CTS and DCD signals can be accessed from the application by using the ***applicom***® library GETMODEM and SETMODEM functions.

RS232 (6 signals)(3RS232, 5RS232, 6RS232)
Means that the protocol task installed at the channel level manages the signals.
- TxD and RxD for data transmission and reception.
- RTS, an output signal positioned by the ***applicom***® interface before each transmission.
- CTS, an input signal, must be active to allow transmission.
- DCD, an input signal, must be active to validate reception.

You can position the DTR output signal from the applicative program by using the ***applicom***® library SETMODEM function.
You can query the CTS and DCD input signal status from the application program by using the ***applicom***® library GETMODEM function.
The 6RS232 module ensures the galvanic isolation.

BC20 mA
Conductive 20 mA current loop. The active or passive operation is determined by wiring. If active, see "5BC20 module characteristics" in wiring section of the Hardware installation manual. The active operation should only be used if BX4010 and BX4010R boxes are used.

3RS485
2-wire RS485 link. The line prepolarization as well as the load resistance are wiring-selectable.
Caution: Wiring is different from modules 3RS485-4 and 5RS485.

3RS485-4
4-prepolarized wire RS422 / RS485 link. The load resistance is wiring-selectable.

5RS485 or 6RS485
2 or 4-prepolarized wire RS422 / RS485 link. Load resistances are wiring-selectable. In addition, the 6RS485 module ensures the galvanic isolation. Your interface must be equipped for this type of PROM module V3.0 or higher.

Baud rate :
You must select a transmission speed comprised between **50 and 38400 bauds**, independently on each channel. This speed must be consistent with that of the equipments connected on the channel.

Synchronization Mode :
The field « Synchronization Mode » is used to define the communication mode. There are two choices:

- protocol operating in the Mark and Space parity mode.
- protocol operating in Break mode.

BREAK time in Characters
The field « BREAK time in Characters » will represent the Break transmission time expressed in number of characters.

❑ **Time-out in ms**
Time-out in milliseconds.

3. - *applicom*® functions usable on the master channel

- Wait mode

- Standard functions

readpackbit	**writepackbit**
readpackibit	
readpackqbit	**writepackqbit**
readdword	**writedword**
readfword	**writefword**
readtimer	**writetimer**
readcounter	**writecounter**
readident	
readmes	**writemes**
	writereadmes
getmodem	**setmodem**

- Specific functions

iocounter	
resetiocounter	
manual	**automatic**

- mode

readdifbit	**writedifpackbit**	
readdifibit		
readdifqbit	**writedifpackqbit**	
readdifdword	**writedifdword**	
readdiffword	**writediffword**	
readdifmes	**writedifmes**	**writereaddifmes**
testtransdif	**transdif**	**transdifpack**

- Cyclic mode

createcyc
startcyc **stopcyc** **actcyc**
transcyc **transcycpack**

Type of variable in equipment	Function type	
	Reading	Writing
Packed bits	X	X
Input packed bits	X	
Output packed bits	X	X
32 bit double words	X	X
32 bit IEEE floating words	X	X

4. - Item of image variables

- Presentation

The "item of image variables" are the syntaxes which allow to access to the variables through the DDE server "pcdde" or the OPC server.

Report you to sections "DDE server/Principles regarding access to Data" or "OPC server/Data Access Principle

According to the configuration of the equipment, the descriptor is determined by default by the server. In case of SBUS protocol, the default descriptor is SAIA descriptor.

You however have the possibility of using another descriptor (in particular the ***applicom***® standard descriptor) by the means of the advanced options.

- Standard descriptor

The standard descriptor can be used for access to the equipments which have not specific descriptors. The address field of the item name may be 10 digits long. It allows to compose a linear address from 0 to 4 giga.

	Single Mode	Table Mode, Matrix Mode
Internal bits	**Bx**	**Bx_n, Bx_n_l**
Input bits	**BIx**	**BIx_n, BIx_n_l**
Output bits	**BOx**	**BOx_n, BOx_n_l**
Internal double words	**Dx**	**Dx_n, Dx_n_l**
Internal floating words	**Fx**	**Fx_n, Fx_n_l**

Note : Limit values for n and l parameters are depending on the protocol. However, in case of PCDDE, limits cannot never be superior than 128 for bits and bytes, 64 for words, 32 for double words and floating words.

For more information on the limits in read and write, see :
- Maximum number of variables per frame with PCDDE on page 2
- Maximum number of variables per frame with OPC server on page 2

For the variable addressing, refer you to the chapter - Introduction on page 1.

- Internal bits => Bx (BIT type)

x : First bit number.
Example : B4

- Internal bits => Bx_n, Bx_n_l

n : Number of bits.
l : Number of bits per line (Matrix mode only).
Examples : B4_10, B4_10_5

- Input bits => Blx (BIT type)

x : First bit number
Example : Bl4

- Input bits => Blx_n, Blx_n_l

n : Number of bits.
l : Number of bits per line (Matrix mode only).
Examples : Bl4_10, Bl4_10_5

- Output bits => BOx (BIT type)

x : First bit number.
Example : BO4

- Output bits => BOx_n, BOx_n_l

n : Number of bits.
l : Number of bits per line (Matrix mode only).
Examples : BO4_10, BO4_10_5

- Internal double words => Dx (32 bits WORD type)

x : First double word number.
Example : D4

- Internal double words => Dx_n, Dx_n_l

n : Number of double words.
l : Number of double words per line (Matrix mode only).
Examples : D4_10, D4_10_5

- Internal floating words => Fx (32 bits IEEE REAL type)

x : First floating word number.
Example : F4

- Internal floating words => Fx_n, Fx_n_l

n : Number of floating words.
l : Number of floating words per line (Matrix mode only).
Examples : F4_10, F4_10_5

- SAIA descriptor

This descriptor is usable only for access to the **SAIA** PLCs through the ***applicom*®** interfaces with **SAIA-SBUS** protocol.

This descriptor is the default descriptor determined by the OPC and DDE server.

	Single Mode	Table Mode, Matrix Mode
Flags	**Fx**	**Fx_n, Fx_n_l**
Input	**Ix**	**Ix_n, Ix_n_l**
Output	**Ox**	**Ox_n, Ox_n_l**
Binary registers	**Rx**	**Rx_n, Rx_n_l**
Bits in binary registers	**Rx.b**	
Floating registers	**Qx**	**Qx_n, Qx_n_l**
Binary Data Block	**DBxRy**	**DBxRy_n, DBxRy_n_l**
Bits in binary Data Block	**DBxRy.b**	
Floating Data Block	**DBxQy**	**DBxQy_n, DBxQy_n_l**

Note : Limit values for n and l parameters are depending on the protocol. However, in case of PCDDE, limits cannot never be superior than 128 for bits and bytes, 64 for words, 32 for double words and floating words.

For more information on the limits in read and write, see :
- Maximum number of variables per frame with PCDDE on page 2
- Maximum number of variables per frame with OPC server on page 2

For the variable addressing, refer you to the chapter - Introduction on page 1.

- Flags=> Fx (BIT type)

x : First bit number (0 to 8091).
Example : F4

- Flags => Fx_n, Fx_n_l

n : Number of bits.
l : Number of bits per line (Matrix mode only).
Examples : F4_10, F4_10_5

- Input => Ix (BIT type)

x : First bit number (0 to 8191).
Example : I4

- Input => Ix_n, Ix_n_l

n : Number of bits
l : Number of bits per line (Matrix mode only).
Examples : I4_10, I4_10_5

- Output => Ox (BIT type)

x : First bit number (0 to 8191).
Example : O4

- Output => Ox_ n, Ox_n_l

n : Number of bits.
l : Number of bits per line (Matrix mode only).
Examples : O4_10, O4_10_5

- Binary registers => Rx (32 bits WORD type)

x : First register number (0 to 4095).
Example : R4

- Binary registers => Rx_n, Rx_n_l

n : Number of registers.
l : Number of registers per line (Matrix mode only).
Examples : R4_10, R4_10_5

- Bits in binary registers => Rx.b (BIT type)

x : Register number (0 to 4095)
b : Bit range in binary register (0 to 31)
Example : R4.5

- Floating registers => Qx (32 bits IEEE REAL type)

x : First register number (0 to 4095).
Example : Q4

- Floating registers => Qx_n, Qx_n_l

n : Number of registers.
l : Number of registers per line (Matrix mode only).
Examples : Q4_10, Q4_10_5

- Binary Data Block => DBxRy (32 bits WORD type)

x : Data Block number (0 to 7999).
y : First element number in Data Block (0 to 7999).
Example : DB4R30

- Binary Data Block => DBxRy_n, DBxRy_n_l

n : Number of double word elements.
l : Number of double word elements per line (Matrix mode only).
Examples : DB4R30_10, DB4R30_10_5

- Bits in binary Data Block => DBxRy.b (BIT type)

x : Data Block number (0 to 7999).
y : Element number in Data Block (0 to 7999).
b : Bit range in binary Data Block (0 to 31).
Example : DB4R30.5

- Floating Data Block => DBxQy (32 bits IEEE REAL)

x : Data block number (0 to 7999).
y : First element number in Data Block (0 to 7999).
Example : DB4Q30

- Floating Data Block => DBxQy_n, DBxQy_n_l

n : Number of floating word elements.
l : Number of floating word elements per line (Matrix mode only).
Examples : DB4Q30_10, DB4Q30_10_5

5. - Appendices

- List of extra files for this protocol

MASTSBUS.	SAIA-SBUS master task

- Evolution / compatibility

This protocol necessitates on ***applicom*®** interfaces a version EPROM minimum V3.3.

6. - Return status of *applicom*® functions

- Introduction

The various ***applicom*®** functions return a status word that:
- Guarantees the request quality.
- Can be used to diagnose the cause of a failure.

The significance of the status word value is given in the following table. As well as the general significance, « Further details » allow you to guide your diagnostic according to the protocol used.

- *applicom*® general statuses

-6	The **TRANSCYC** (or **TRANSCYCPACK**) function is used with a cyclic function number that is no longer activated.
-5	The user program tries to perform a **TRANSDIF** (or **TRANSDIFPACK**) deferred transfer although the deferred request in progress is not completed.
-1	**TRANSDIF** (or **TRANSDIFPACK**) deferred transfer request related to a write that took place correctly.
0	No anomaly detected. The function took place correctly.
1	**Unknown function.** The requested function is not supported.
2	**Incorrect address.** The address of the variable you are soliciting is incorrect.
3	**Incorrect data.** Further details Function: BINBCD, BCDBIN. - At least one of the accessed values is not in BCD format (0 ≤ value ≤ 9999).
4	**Irretrievable data.**
32	**Bad parameter passed into the function.** Incorrect number of variables.
35	**Data not available in cyclic read.** Attempt to transfer data with **TRANSCYC** (or **TRANSCYCPACK**) before it has been read in the equipment.
40	Deferred read or write attempt when the deferred request register is full. Another task must free the resources by making an exitbus
41	Deferred read or write attempt when the deferred request register is full Perform deferred request transfers with **TRANSDIF** (or **TRANSDIFPACK**) in order to release the register (64 positions).
42	Deferred request transfer attempt with **TRANSDIF** (or **TRANSDIFPACK**) when the latter is empty (no deferred requests in progress).
45	**Non-resident communication software.** Initialize the ***applicom*®** interface before using it by typing command **applicom** (or **PCINIT** under Windows).
46	Board number not configured, or Master/client ***applicom*®** function aiming at a channel configured as slave/server, or vice versa.
47	No ***applicom*®** interface.
51	Driver system problem.
59	Protection key missing on the ***applicom*®** interface.. Using ***applicom*®** function without INITBUS function.
66	Insufficient ***applicom*®** interface memory.
255	Used by the « PCDDE » MS-Windows server. Initial value of « STATUS_READ » and « STATUS_WRITE ». This value indicates that no transaction has been made between « PCDDE » and ***applicom*®** interface.

Comments :
Negative status codes are information codes.

- Statuses according to the protocol

-8	End of reception on inter-character Time-out.
0	No anomaly detected. The function took place correctly.
1	**Unknown function.** The requested function is not supported.
3	**Incorrect data.** - Inconsistent frame content. - Function: WRITEFWORD, WRITEDIFFWORD, at least one of the write values is not in real format.
4	**Irretrievable data.** Negative acknowledgment from the equipment.
32	**Bad parameter passed into the function.** Incorrect number of variables.
33	**Response time fault (Time-Out).** - No response to the request within time limit. - Check wiring. Check the channel and slave configuration.
34	**Parity fault in reception or of control word (CRC16, BCC).** Check the configuration of the channel and target equipment. Check the wiring (RC at end of line for the RS485).
36	**Equipment not configured.** Equipment number > 255. Define the equipment configuration with **PCCONF** and start again the ***applicom*®** product.
48	The ***applicom*®** interface **RS232** channel configured as "**6 signals**" cannot transmit since it waited for the Modem CTS for more than 3 seconds during a transmission.

Comments :
Negative status are information codes.

7. - Glossary of terms

***applicom*® interface**
Communication card, ISA or PCI type, with the ***applicom*®** real time multi-task kernel.(PC1000, PC2000, PC4000, PC1500PFB, ...)

break
A logical 1 is generated on the bus.

BX4010
4 channels distribution box, with galvanic insulation.

BX4010R
4 channels rack distribution box, with galvanic insulation.

channel
Physical output of an ***applicom*®** card.

CTS
Clear To Send

Data Block
Internal memory bloc of the PLC

DCD
Data Carrier Detect

DTR
Data Terminal Ready

OPC
Ole for Process Control.

PCDDE
***applicom*®** DDE server.

RTS
Ready to transmit

RxD
Receipt signal

SBUS master
Node of SBUS network, which has the initiative of exchange

TxD
Transmit signal.

8. - Index