SattBus Serial Driver Help

© 2012 Kepware Technologies

Table of Contents

SattBus Serial Driver Help

Help version 1.020

CONTENTS

Overview

The SattBus Serial Driver provides an easy and reliable way to connect SattBus Serial devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It is intended for use with SattCon devices communicating via the SattBus interface.

The SattBus PC-board is used as the hardware interface for connecting PCs to a SattBus system. The SattBus PC-board is implemented as a standard I/O channel board. It is equipped with a dual port RAM chip (which is mapped into the host processor memory area).

Device Setup

Supported Devices
SattCon 05
SattCon 15
SattCon 31
SattCon 35
SattCon 200
Satt Control OP45

Communication Protocol
SattBus

Device IDs
This parameter specifies the unique ID that will be used in order to communicate with other devices. The valid range is from 2 to 127. The default setting is 2.

Request Timeout
This parameter specifies the amount of time that the driver will wait on a response from the device before retrying or giving up and moving on to the next request. The valid range is from 2000 to 30000 ms.

Fail After xxx Successive Timeouts
Valid numbers range from 2 to 10.

Select Bit Ordering for Memory Cell (M, MW, XW)
This setting is used to configure the bit ordering that will be used when reading or writing Word memory cell tags (M, MW and XW). The default setting is MSBit 7| 6| 5| 4| 3| 2| 1| 0| 15| 14| 13| 12| 11| 10| 9| 8 LSBit. For more information, refer to **Settings - Bit Ordering for Memory Cells**.

Settings - Bit Ordering for Memory Cells
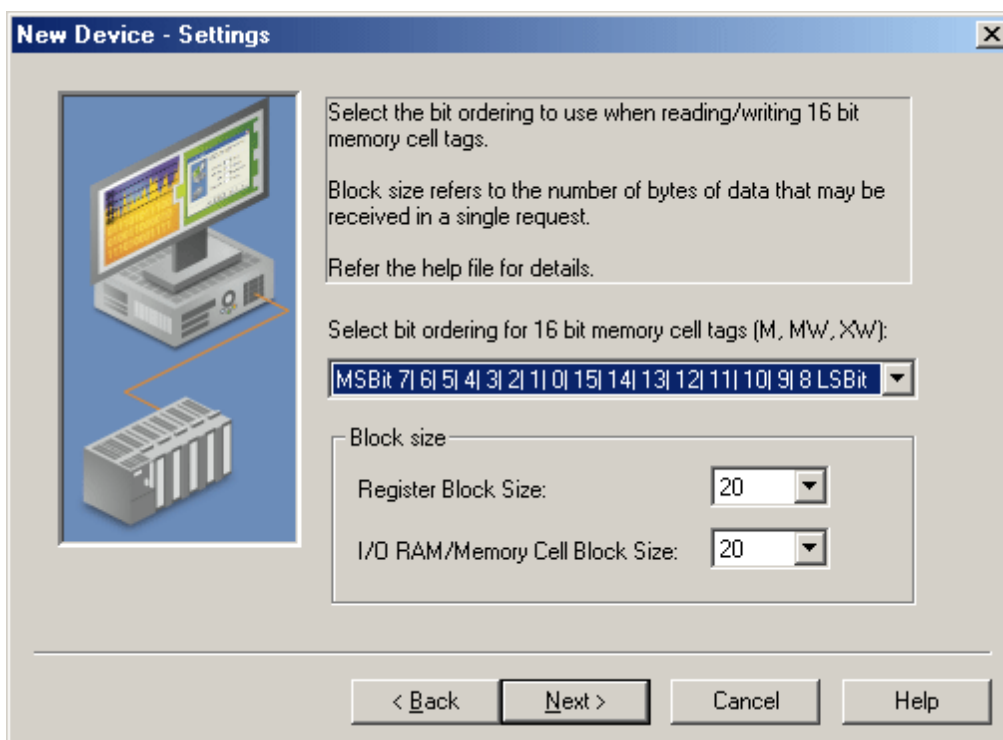
The **Select bit ordering for 16 bit memory cell tags (M, MW, XW)** setting allows users to configure the bit ordering to be used when reading or writing Word memory cell tags (M, MW and XW). There are three possible selections: select a link from the list below in order to jump to the full description. The default setting is MSBit 7| 6| 5| 4| 3| 2| 1| 0| 15| 14| 13| 12| 11| 10| 9| 8 LSBit.

MSBit 0| 1| 2| 3| 4| 5| 6| 7| 8| 9| 10| 11| 12| 13| 14| 15 LSBit
MSBit 15| 14| 13| 12| 11| 10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0 LSBit

MSBit 7| 6| 5| 4| 3| 2| 1| 0| 15| 14| 13| 12| 11| 10| 9| 8 LSBit

Note: The addresses are in octal.

MSBit 0| 1| 2| 3| 4| 5| 6| 7| 8| 9| 10| 11| 12| 13| 14| 15 LSBit

This is how the DOX10 programming software represents the memory cells. In this condition, the most significant bit (MSBit) of the tag being read or written is equal to the address of the tag. The following examples will illustrate this condition for reads and writes:

1. For the Word tag 'MW00' (address '0'), the most significant bit (MSBit) is the memory cell address '0'. The least significant bit (LSBit) is memory cell address '17'.

<-------------------------- MW00 -------------------------->																
MSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**LSBit**

Read
If memory addresses '2' and '13' are ON, the rest are OFF.

<-------------------------- MW00 -------------------------->																
State	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	=>16416
MSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**LSBit**

The value that the tag 'MW00' will read is 16416.

Write
Writing a value '1' to the tag 'MW00' will set the bit memory addresses as shown below:

<-------------------------- MW00 -------------------------->																
State	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	=>1
MSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**LSBit**

2. For the Word tag 'MW11' (address '11'), the most significant bit (MSBit) is the memory bit '11'. The least significant bit (LSBit) is memory bit '31'.

   Note: Word access at non-byte boundaries is not allowed in the DOX10 programing software.

<-------------------------- MW11 -------------------------->																
MSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**LSBit**

Read
If memory addresses '14' and '27' are ON, the rest are OFF.

<-------------------------- MW11 -------------------------->																
State	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	=>4100
MSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**LSBit**

The value that the tag 'MW11' will read is 4100.

Write
Writing a value '258' to the tag 'MW11' will set the bit memory addresses as shown below:

<-------------------------- MW11 -------------------------->																
State	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	=>258
MSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**LSBit**

Note: Due to the nature of this bit ordering option, care is needed when addressing 8-bit memory that is close to 16-bit addresses.

MSBit 15| 14| 13| 12| 11| 10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0 LSBit

In this condition, the least significant bit (LSBit) of the tag being read or written is equal to the address of the tag. The following examples will illustrate this condition for reads and writes.

1. For the Word tag 'MW00' (address '0') the most significant bit (MSBit) is the memory cell address '17'. The least significant bit (LSBit) is memory cell address '0'.

<---------------------------- **MW00** ---------------------------->																	
LSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**MSBit**

Read
If memory addresses '2' and '13' are ON, the rest are OFF.

<---------------------------- **MW00** ---------------------------->																	
State	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	=>2052
LSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**MSBit**

The value that the tag 'MW00' will read is 2052.

Write
Writing a value '1' to the tag 'MW00' will set the bit memory addresses as shown below:

<---------------------------- **MW00** ---------------------------->																	
State	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	=>1
LSBit	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	**MSBit**

2. For the Word tag 'MW11' (address '11'), the most significant bit (MSBit) is the memory bit '31'. The least significant bit (LSBit) is memory bit '11'.

Note: Word access at non-byte boundaries is not allowed in the DOX10 programing software.

<---------------------------- **MW11** ---------------------------->																	
LSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**MSBit**

Read
If memory addresses '14' and '27' are ON, the rest are OFF.

<---------------------------- **MW11** ---------------------------->																	
State	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	=>8200
LSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**MSBit**

The value that the tag 'MW11' will read is 8200.

Write
Writing a value '258' to the tag 'MW11' will set the bit memory addresses as shown below:

<---------------------------- **MW11** ---------------------------->																	
State	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	=>258
LSBit	11	12	13	14	15	16	17	20	21	22	23	25	26	27	30	31	**MSBit**

MSBit 7| 6| 5| 4| 3| 2| 1| 0| 15| 14| 13| 12| 11| 10| 9| 8 LSBit

This condition is similar to the second selection described above, but with the bytes swapped. It is the default condition. The following examples will illustrate this condition for reads and writes.

1. For the Word tag 'MW00' (address '0'), the most significant bit (MSBit) is the memory cell address '7'. The least significant bit (LSBit) is memory cell address '10'.

<---------------------------- **MW00** ---------------------------->																	
LSBit	10	11	12	13	14	15	16	17	0	1	2	3	4	5	6	7	**MSBit**

Read

If memory addresses '2' and '13' are ON, the rest are OFF.

<---------------------------- MW00 ---------------------------->																
State	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	=>1032
LSBit	10	11	12	13	14	15	16	17	0	1	2	3	4	5	6	7	**MSBit**

The value that the tag 'MW00' will read is 1032.

Write
Writing a value '1' to the tag 'MW00' will set the bit memory addresses as shown below:

<---------------------------- MW00 ---------------------------->																
State	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	=>1
LSBit	10	11	12	13	14	15	16	17	0	1	2	3	4	5	6	7	**MSBit**

2. For the Word tag 'MW11' (address '11'), the most significant bit (MSBit) is the memory bit '20'. The least significant bit (LSBit) is memory bit '21'.

Note: Word access at non-byte boundaries is not allowed in the DOX10 programing software.

<---------------------------- MW11 ---------------------------->																
LSBit	21	22	23	25	26	27	30	31	11	12	13	14	15	16	17	20	**MSBit**

Read
If memory addresses '14' and '27' are ON, the rest are OFF.

<---------------------------- MW11 ---------------------------->																
State	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	=>2080
LSBit	21	22	23	25	26	27	30	31	11	12	13	14	15	16	17	20	**MSBit**

The value that the tag 'MW11' will read is 2080.

Write
Writing a value '258' to the tag 'MW11', will set the bit memory addresses as shown below:

<---------------------------- MW11 ---------------------------->																
State	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	=>258
LSBit	21	22	23	25	26	27	30	31	11	12	13	14	15	16	17	20	**MSBit**

Note: Due to the nature of this bit ordering option, care is needed when addressing 8-bit memory that is close to 16-bit addresses.

Block Size

Register Block Size
This parameter specifies the number of bytes of Register data (R tags) that may be requested from a device at one time. To refine the performance of this driver, configure the block size to one of the following settings: 20, 32, 64, 128, 256 or 510. The default setting is 20 bytes.

I/O RAM/Memory Cell Block Size
This parameter specifies the number of bytes of I/O RAM/Memory Cell data (X, XB, XW, M, MW, I, O and IO tags) that may be requested from a device at one time. To refine the performance of the driver, configure the block size to one of the following settings: 20, 32, 64, 128 or 255. The default setting is 20 bytes.

Interface Board Settings

These channel properties must be set to match the SattBus Interface Board's configuration.

Memory Segment (hex)
The dual port RAM of the interface board is mapped onto the host processor RAM area as a 2 KB memory segment. The start address of the dual port memory segment can be selected from a range between 0x00000 and 0xFF000

in increments of 0x1000. This driver only supports addresses between 0xC0000 and 0xEF000, however, with 0xD0000 being the recommended value.

SattBus Node

This parameter specifies the network address that will be assigned to the server channel. This address may be between 2 and 127. If 0 is specified, the network address configured with the interface board DIP switches will be used.

Cable Diagrams

Linking strips looping Sattbus network to next station

Termination Resistance
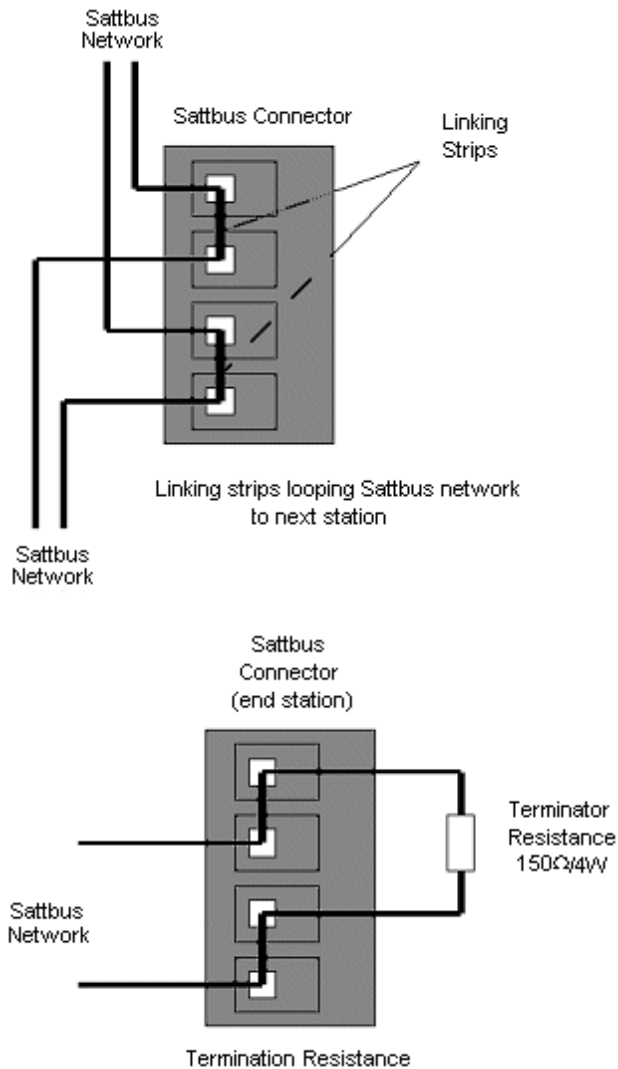
Data Types Description

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8 bit value bit 0 is the low bit bit 7 is the high bit
Char	Signed 8 bit value bit 0 is the low bit bit 6 is the high bit bit 7 is the sign bit
Word	Unsigned 16 bit value bit 0 is the low bit bit 15 is the high bit
Short	Signed 16 bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
DWord	Unsigned 32 bit value bit 0 is the low bit bit 31 is the high bit
Long	Signed 32 bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit

Address Descriptions

Default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Data Type	Access
Register*	R0-R32767	**Word**, Short	Read/Write
	R0.00-R32767.00 … R0.15-R32767.15	**Boolean**	Read Only
	RW0-RW32767	**Word**, Short	Read/Write
	RW0.00-RW32767.00 … RW0.15-RW32767.15	**Boolean**	Read Only
	RD0-RD32766	**DWord**, Long	Read/Write
I/O RAM bits**	0-77777	**Boolean**	Read/Write
	IO0-IO77777	**Boolean**	
	I0-I77777	**Boolean**	
	O0-O77777	**Boolean**	
	X0-X77777	**Boolean**	
I/O RAM***	XB0-XB77770	**Byte**, Char	Read/Write
	XW0.00-XW77760.00 … XW0.15-XW77760.15	**Boolean**	
	XW0-XW77760	**Word**, Short	
Memory Cell in I/O RAM****	M0-M77760	**Word**, Short	Read/Write
	M0.00-M77760.00 … M0.15-M77760.15	**Boolean**	
	MW0-MW77760	**Word**, Short	
	MW0.00-MW77760.00 … MW0.15-MW77760.15	**Boolean**	

*Addresses are in decimal.
**Addresses are in octal.
***Addresses are in octal, bit numbers are in decimal. Addresses can be on a non-byte boundary (non-multiples of 10 octal).
****Bit numbers are in decimal.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation
Missing address
Device address '<address>' contains a syntax error
Address '<address>' is out of range for the specified device or register
Device address '<address>' is not supported by model '<model name>'
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' is Read Only

Device Status Messages
Device '<device name>' is not responding
Unable to write to '<address>' on device '<device name>'

Device Specific Messages
Unable to read '<number>' bytes starting at address '<address>' on device '<device name>'. <reason>
Unable to write to tag '<tag name>' on device '<device name>' <reason>
Failed to start Board '<channel name>'. Possible resource conflicts
Failed to connect to Board '<channel name>'
Failed to map memory for Board '<channel name>'
Failed to allocate Board '<channel name>'
Failed to allocate memory for Board '<channel name>'
Failed to stop Board '<channel name>'
Failed to initialize Board '<channel name>'
Failed to initialize Communications for Board '<channel name>'
Failed to get Version for Board '<channel name>'
Board '<channel name>' not present at location specified
Failed to set Communication Parameters for Board '<channel name>'
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. Memory segment value has to be between '<start address>' and '<end address>'
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. Memory segment value has to be in increments of 1000 (hex values)
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. The Sattbus Node ID cannot be 1. Valid Node ID values are 0 and 2-127

See Also:
Error Codes

Error Codes

Error Code	Description
0	No error.
1	General error.
2	Checksum error.
3	Limit error.
4	Message length error.
5	Illegal function.
6	Illegal address.
7	Illegal type.
8	Illegal operation.
10	Queue full.
11	Queue ready.
17	Function occupied.
18	No such variable.
19	Buffer error.
20	Module restart.

21	Device not responding.
22	Temporarily inoperable.

Address Validation

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation
Missing address
Device address '<address>' contains a syntax error
Address '<address>' is out of range for the specified device or register
Device address '<address>' is not supported by model '<model name>'
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' is Read Only

Missing address

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically has no length.

Solution:
Re-enter the address in the client application.

Device address '<address>' contains a syntax error

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically contains one or more invalid characters.

Solution:
Re-enter the address in the client application.

Address '<address>' is out of range for the specified device or register

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically references a location that is beyond the range of supported locations for the device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application.

Device address '<address>' is not supported by model '<model name>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically references a location that is valid for the communications protocol but not supported by the target device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application. Also verify that the selected model name for the device is correct.

Data Type '<type>' is not valid for device address '<address>'

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has been assigned an invalid data type.

Solution:

Modify the requested data type in the client application.

Device address '<address>' is Read Only

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has a requested access mode that is not compatible with what the device supports for that address.

Solution:

Change the access mode in the client application.

Device Status Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Status Messages
Device '<device name>' is not responding
Unable to write to '<address>' on device '<device name>'

Device '<device name>' is not responding

Error Type:

Serious

Possible Cause:

1. The SattBus Serial connection between the device and the Host PC is broken.
2. The SattBus Interface board parameters are incorrect.
3. The SattBus board may not be configured correctly.
4. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified parameters match those of the SattBus card and device.
3. Verify the configuration of the SattBus card.
4. Increase the Request Timeout setting so that the entire response can be handled.

Unable to write to '<address>' on device '<device name>'

Error Type:

Serious

Possible Cause:

1. The SattBus Serial connection between the device and the Host PC is broken.
2. The SattBus Interface board parameters are incorrect.
3. The SattBus board may not be configured correctly.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified parameters match those of the SattBus card and device.
3. Verify the configuration of the SattBus card.

Device Specific Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Specific Messages

Unable to read '<number>' bytes starting at address '<address>' on device '<device name>'. <reason>
Unable to write to tag '<tag name>' on device '<device name>' <reason>
Failed to start Board '<channel name>'. Possible resource conflicts
Failed to connect to Board '<channel name>'
Failed to map memory for Board '<channel name>'
Failed to allocate Board '<channel name>'
Failed to allocate memory for Board '<channel name>'
Failed to stop Board '<channel name>'
Failed to initialize Board '<channel name>'
Failed to initialize Communications for Board '<channel name>'
Failed to get Version for Board '<channel name>'
Board '<channel name>' not present at location specified
Failed to set Communication Parameters for Board '<channel name>'
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. Memory segment value has to be between '<start address>' and '<end address>'
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. Memory segment value has to be in increments of 1000 (hex values)
Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel'<channel name>'. The Sattbus Node ID cannot be 1. Valid Node ID values are 0 and 2-127

Unable to read '<number>' bytes starting at address '<address>' on device '<device name>'. <reason>

Error Type:
Serious

Possible Cause:
The driver was unable to complete a read operation because of the reason specified.

Solution:
Refer to the reasons below.

Reasons:
1. The frame returned contains errors.
2. The board response timed-out.
3. The board returned incomplete data.
4. An unsupported message was used.
5. The board returned error code '<code>'.
6. An unexpected error occurred.
7. The device is busy.
8. No buffers were available to transmit request.

See Also:
Error Codes

Unable to write to tag '<tag name>' on device '<device name>' <reason>

Error Type:
Serious

Possible Cause:
The driver was unable to complete a write operation because of the reason specified.

Solution:
Refer to the reasons below.

Reasons:
1. The frame returned contains errors.
2. The board response timed-out.
3. The board returned incomplete data.
4. An unsupported message was used.
5. The board returned error code '<code>'.
6. An unexpected error occurred.
7. The device is busy.

8. No buffers were available to transmit request.

See Also:
Error Codes

Failed to start Board '<channel name>'. Possible resource conflicts

Error Type:
Serious

Possible Cause:
The driver was unable to start the SattBus interface board.

Solution:
Check for conflicting devices installed on the computer.

Failed to connect to Board '<channel name>'

Error Type:
Serious

Possible Cause:
The driver was unable to connect to the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to map memory for Board '<channel name>'

Error Type:
Warning

Possible Cause:
The SattBus kernel mode interface failed to provide a card memory mapping to the driver.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to allocate Board '<channel name>'

Error Type:
Warning

Possible Cause:
Resources could not be allocated for kernel mode interface to the SattBus board.

Solution:
Shut down any unused applications and restart the server.

Failed to allocate memory for Board '<channel name>'

Error Type:
Warning

Possible Cause:
Resources could not be allocated for required memory map of SattBus card.

Solution:
Shut down any unused applications and restart the server.

Failed to stop Board '<channel name>'

Error Type:
Serious

Possible Cause:

The driver was unable to stop the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to initialize Board '<channel name>'

Error Type:
Serious

Possible Cause:
The driver was unable to initialize the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to initialize Communications for Board '<channel name>'

Error Type:
Serious

Possible Cause:
The driver was unable to initialize communications for the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to get Version for Board '<channel name>'

Error Type:
Serious

Possible Cause:
The driver was unable get version information from the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Board '<channel name>' not present at location specified

Error Type:
Serious

Possible Cause:
The SattBus interface card may not be configured or installed correctly.

Solution:
Verify that the interface board is configured and installed correctly.

Failed to set Communication Parameters for Board '<channel name>'

Error Type:
Serious

Possible Cause:
The driver was unable to set communications parameters for the SattBus interface board.

Solution:
Verify that the interface board is configured and installed correctly.

Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. Memory segment value has to be in increments of 1000 (hex values)

Error Type:

Serious

Possible Cause:
The memory segment address is not a value that is divisible by 0x1000.

Solution:
Make sure the memory segment address is divisible by 0x1000 (i.e last three digits are zeros).

Invalid XML document. Reason: Error loading 'InterfaceBoard Configuration' for channel '<channel name>'. Memory segment value has to be between '<start address>' and '<end address>'

Error Type:
Serious

Possible Cause:
The memory segment address is outside the driver supported range of 0xC000-0xEF000.

Solution:
Make sure the memory segment address is in the range 0xC000-0xEF000.

See Also:
Interface Board Settings

Invalid XML document. Reason: Error loading 'Interface Board Configuration' for channel '<channel name>'. The Sattbus Node ID cannot be 1. Valid Node ID values are 0 and 2-127

Error Type:
Serious

Possible Cause:
The Sattbus Node ID is set to 1.

Solution:
Make sure the Sattbus Node ID is either 0 or 2-127.

See Also:
Interface Board Settings

Index

A

B

C

D

E