Intelligent Actuator Super SEL Driver

Table of Contents

Intelligent Actuator Super SEL Driver

Help version 1.016

CONTENTS

Overview

The Intelligent Actuator Super SEL Driver provides a reliable way to connect IA Super SEL controllers to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It is intended for use with Intelligent Actuator Super SEL Controllers (Type E & G).

Device Setup

Supported Devices

IA Super SEL Controllers (Type E & G)
X-Sel Controllers (Protocol Version 1 only)

Communication Protocol

Super SEL Serial Protocol
X-Sel Controllers Serial Protocol (Version 1 only)

Supported Communication Properties

Baud Rate: 9600 (19200, 38400 X-Sel only)
Parity: None
Data Bits: 8
Stop Bits: 1
Flow Control: None

Ethernet Encapsulation

This driver supports Ethernet Encapsulation, which allows the driver to communicate with serial devices attached to an Ethernet network using a terminal server. Ethernet Encapsulation mode can be enabled through the COM ID property group in Channel Properties. For more information, refer to the OPC server's help file.

Flow Control

When using an RS232/RS485 converter, the type of flow control that is required will depend on the needs of the converter. Some converters do not require any flow control and others will require RTS flow. Consult the converter's documentation to determine its flow requirements. An RS485 converter that provides automatic flow control is recommended.

Note: When using the manufacturer's supplied communications cable, it is sometimes necessary to choose a flow control setting of **RTS** or **RTS Always** under the Channel Properties.

Modem Setup

This driver supports modem functionality. For more information, please refer to the topic "Modem Support" in the OPC Server Help documentation.

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
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
Float	32 bit floating point value.
Double	64 bit floating point value.
String	Null terminated character array

Address Descriptions

Click on the links below for information regarding the IA Super SEL protocol address specifications.

Super SEL
X-Sel

Note: The actual number of addresses of each type depends on the IA Super SEL device in use. For address ranges, refer to the device documentation.

Super SEL Addressing

For more information on a specific Super SEL model, click a link from the list below.

Input Port
Output Port
Flag
Axis Status and Control
Point Data
Variable
Program Execution
Test Controller
Reset Controller
Error Messaging

Note: The actual number of addresses of each type depends on the IA Super SEL device in use. Refer to the device documentation for address ranges.

Input Port (Super SEL)

The syntax for accessing any input is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
I<number>	**Boolean**	0-287	Read Only
<number>	**Boolean**	0-287	Read Only

Examples

I0	Input 0
30	Input 30
I287	Input 287

Output Port (Super SEL)

The syntax for accessing any output is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
Q<number>	**Boolean**	300-587	Read/Write
O<number>	**Boolean**	300-587	Read/Write
<number>	**Boolean**	300-587	Read/Write

Examples

Q300	Output 300
400	Output 400
O587	Output 587

Flag (Super SEL)

The syntax for accessing any flag is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
F<number>	**Boolean**	600-887	Read/Write
<number>	**Boolean**	600-887	Read/Write

Examples

F600	Flag 600
700	Flag 700
F887	Flag 887

Axis Status and Control (Super SEL)

The default data types are shown in **bold**.

Accessing Axis Status-Reading Data

Syntax	Data Type	Axis	Access	Functionality
A<axis>s	**Boolean**	1-8	Read/Write	Axis servo status.
A<axis>h	**Boolean**	1-8	Read/Write	Axis homing status.
A<axis>m	**Boolean**	1-8	Read/Write	Axis moving status.
A<axis>e	**Byte**	1-8	Read Only	Axis error code.
A<axis>p	**Float**	1-8	Read Only	Axis position.

Accessing Axis Control-Writing Data

Syntax	Data Type	Axis	Access	Functionality
A<axis>s	**Boolean**	1-8	Read/Write	Turn axis servo on or off.
A<axis>h	**Boolean**	1-8	Read/Write	Home an axis.
A<axis>m	**Boolean**	1-8	Read/Write	Halt any axis.
A<axis>mf	**Boolean**	1-8	Write Only	Jog any axis forward.
A<axis>mb	**Boolean**	1-8	Write Only	Jog any axis backward.
H<axis>v	**Float**	1-8	Write Only	Velocity to home an axis.
J<axis>v	**Float**	1-8	Write Only	Velocity to jog an axis.
J<axis>a	**Float**	1-8	Write Only	Acceleration to jog an axis.
ABS<axis>	**Float**	1-8	Write Only	Move axis to a real position.
M<axis>v	**Float**	1-8	Write Only	Velocity to move to a real position.
M<axis>a	**Float**	1-8	Write Only	Acceleration to move to a real position.

Note: J<axis>v and J<axis>a tags must be present and set in the client application before jog commands can be issued by the driver. Likewise, M<axis>v and M<axis>a tags must be present and set before absolute axis movement commands can be issued.

Examples

1. To show the current servo status for axis 1:
A1s

2. To turn on axis 1 servo:
A1s-write 1

3. To jog axis 1 forward at a velocity of 50 mm/sec and acceleration of 1/100g:
J1v-write 50
J1a-write .01
A1mf-write any value

4. To halt the previous jog command:
A1m-write 1

5. To home axis 1 at 200 mm/sec:
H1v-write 200
A1h-write 1

6. To move to real position 150.25 (1/1000mm. on axis 1 at a velocity of 300 mm/sec and 1/100g:
M1v-write 300
M1a-write .01
ABS1-write 150.25

Point Data (Super SEL)

The syntax for accessing point data is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Axis	Point	Access	Functionality
P<axis>a,<point>	**Float**	1-8	0-9999	Read/Write	Axis acceleration point data.
P<axis>v,<point>	**Float**	1-8	0-9999	Read/Write	Axis velocity point data.
P<axis>p,<point>	**Float**	1-8	0-9999	Read/Write	Axis position point data.

The syntax for moving to point data is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Axis	Point	Access	Functionality
PM<axis>a	**Float**	1-8	N/A	Write Only	Axis acceleration to move.
PM<axis>v	**Float**	1-8	N/A	Write Only	Axis velocity to move.
PM<axis>,<point>	**Boolean**	1-8	0-9999	Write Only	Axis point position to move.

Examples

1. To show the current position in the point data table for axis 2 point 3:
P2p,3

2. To show the current acceleration in the point data table for axis 1 point 17:
P1a,17

3. To move to the position in the point data table for axis 1 point 17, write a value to:
PM1,17

Note: PM1,v and PM1,a can be set to override the specified point tables velocity and acceleration.

Variable (Super SEL)

The syntax for accessing any variable is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Program	Number	Access
V<program>,<number>	**DWord,** Long, Float	0-99	0-999	Read Only

Example

To read variable 200 for program 0:
V0,200

Program Execution (Super SEL)

The syntax for executing or stopping a program is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Number	Access
EXE<number>	**Boolean**	0-99	Write Only

Examples

1. To start program 2:
EXE2-write 1

2. To stop the program:
EXE2-write 0

Test Controller (Super SEL)

The syntax for testing the controller is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Read Write
TST	**Boolean**	Read Only

Example

To test the controller:
TST-The controller is responding properly if the value displayed is 1.

Reset Controller (Super SEL)

The syntax for resetting the controller is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Read Write
RST	**Boolean**	Write Only

Example

To reset the controller:
RST-Writing any value to this address will reset the controller.

Error Messaging (Super SEL)

The syntax for accessing an error message from an error code is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Read Write	Functionality
MSG	**String**	Read Only	Displays the error message for the error code stored in EC.
EC	**Word**, Short	Read/Write	Holds the error code for the displayed error code.

Note: Initially EC holds 0 and no error message will be displayed. Entering in an error code for EC will display the appropriate message for MSG. Writing a 0 to EC will clear the current error message.

Example

To display the error message associated with error code 0xA3:
EC-write 0xA3 (163 decimal)
MSG-will display the error message "(A3) DEV_ERR"

X-Sel Addressing

For more information on a specific X-SEL model, click a link from the list below.

Input Port
Output Port
Flag
Axis Status and Control
Point Data
Variable
Program Execution
Test Controller
Reset Controller
Error Messaging

Note: The actual number of addresses of each type depends on the IA Super SEL device in use. Refer to the device documentation for address ranges.

Input Port (X-Sel)

The syntax for accessing any input is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
I<number>	**Boolean**	0-287	Read Only
<number>	**Boolean**	0-287	Read Only

Examples

I0	Input 0
30	Input 30
I287	Input 287

Output Port (X-Sel)

The syntax for accessing any output is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
Q<number>	**Boolean**	300-587	Read/Write
O<number>	**Boolean**	300-587	Read/Write
<number>	**Boolean**	300-587	Read/Write

Examples

Q300	Output 300
400	Output 400
O587	Output 587

Flag (X-Sel)

The syntax for accessing global flags is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Number	Access
F<number>	**Boolean**	600-899	Read/Write
<number>	**Boolean**	600-899	Read/Write

Examples

F600	Flag 600
700	Flag 700
F887	Flag 887

Axis Status and Control (X-SEL)

The default data types are shown in **bold**.

Accessing Axis Status-Reading Data

Syntax	Data Type	Axis	Access	Functionality
A<axis>s	**Boolean**	1-8	Read/Write	Axis servo status.
A<axis>h	**Boolean**	1-8	Read/Write	Axis homing status.
A<axis>m	**Boolean**	1-8	Read/Write	Axis moving status.
A<axis>e	**Word**, Short	1-8	Read Only	Axis error code.
A<axis>p	**Float**	1-8	Read Only	Axis position.
A<axis>mc	**Boolean**	1-8	Read Only	Motion complete.
A<axis>pf	**Boolean**	1-8	Read Only	Push force.
A<axis>cr	**Boolean**	1-8	Read Only	Creep.
A<axis>ov	**Boolean**	1-8	Read Only	Overrun.
A<axis>hs	**Boolean**	1-8	Read Only	Home.
A<axis>os	**Boolean**	1-8	Read Only	Overspeed.
A<axis>fs	**Boolean**	1-8	Read Only	Full absolute status.
A<axis>ce	**Boolean**	1-8	Read Only	Counter error.
A<axis>of	**Boolean**	1-8	Read Only	Counter overflow.
A<axis>me	**Boolean**	1-8	Read Only	Multiple rotation error.
A<axis>be	**Boolean**	1-8	Read Only	Battery error.
A<axis>ba	**Boolean**	1-8	Read Only	Battery alarm.

Accessing Axis Control-Writing Data

Syntax	Data Type	Axis	Access	Functionality
A<axis>s	**Boolean**	1-8	Read/Write	Turn axis servo on or off.
A<axis>h	**Boolean**	1-8	Read/Write	Home an axis.
A<axis>m	**Boolean**	1-8	Read/Write	Halt any axis.
A<axis>mf	**Boolean**	1-8	Write Only	Jog any axis forward.
A<axis>mb	**Boolean**	1-8	Write Only	Jog any axis backward.
A<axis>if	**Boolean**	1-8	Write Only	Incremental move forward.
A<axis>ib	**Boolean**	1-8	Write Only	Incremental move backward.
H<axis>v	**Float**	1-8	Write Only	Velocity to home an axis.
J<axis>v	**Float**	1-8	Write Only	Velocity to jog an axis.
J<axis>a	**Float**	1-8	Write Only	Acceleration to jog an axis.
J<axis>d	**Float**	1-8	Write Only	Deceleration to jog an axis.
ABS<axis>	**Float**	1-8	Write Only	Move axis to a real position.
REL<axis>	**Float**	1-8	Write Only	Relative move forward.
M<axis>v	**Float**	1-8	Write Only	Velocity to move to a real or relative position.
M<axis>a	**Float**	1-8	Write Only	Acceleration to move to a real or relative position.
M<axis>d	**Float**	1-8	Write Only	Deceleration to move to a real or relative position.

Notes: J<axis>v and J<axis>a tags must be present and set in the client application before jog commands can be issued by the driver. Likewise, M<axis>v and M<axis>a tags must be present and set before absolute and relative axis movement commands can be issued. The maximum incremental jog movement is 1 mm.

Examples

1. To show the current servo status for axis 1:
A1s

2. To turn on axis 1 servo:
A1s – write 1

3. To jog axis 1 forward at a velocity of 50 mm/sec, and acceleration of 1/100 g, and deceleration of 2/100 g:
J1v-write 50
J1a-write 0.01
J1d-write 0.02
A1mf-write 1

4. To halt the previous jog command:
A1m-write 1

5. To home axis 1 at 200 mm/sec:
H1v-write 200
A1h-write 1

6. To move axis 1 to absolute position 150.25 mm at a velocity of 300 mm/sec, and with acceleration of 1/100 g, and deceleration of 2/100 g:
M1v-write 300
M1a-write 0.02
M1d-write 0.01
ABS1-write 150.25

7. To jog axis 1 backward 0.5 mm from current position then stop:
A1ib-write 0.5

8. To move axis 1 forward 0.5 mm from current position then stop:
A1if-write 0.5, or
REL1-write 0.5

9. To move axis 1 forward 50 mm from current position, users must utilize a relative movement tag since the incremental jog movements have 1 mm limit.
REL1-write 50.0

10. To turn off axis 1 servo:
A1s-write 0

Point Data (X-SEL)

The syntax for accessing point data is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Axis	Point	Access	Functionality
P<axis>a,<point>	**Float**	1-8	0-9999	Read/Write	Axis acceleration point data.
P<axis>d,<point>	**Float**	1-8	0-9999	Read/Write	Axis deceleration point data.
P<axis>v,<point>	**Float**	1-8	0-9999	Read/Write	Axis velocity point data.
P<axis>p,<point>	**Float**	1-8	0-9999	Read/Write	Axis position point data.

The syntax for moving to point data is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Axis	Point	Access	Functionality
PM<axis>a	**Float**	1-8	N/A	Write Only	Axis acceleration to move.
PM<axis>d	**Float**	1-8	N/A	Write Only	Axis deceleration to move.
PM<axis>v	**Float**	1-8	N/A	Write Only	Axis velocity to move.
PM<axis>,<point>	**Boolean**	1-8	0-9999	Write Only	Axis point position to move.
PCL<point>	**Boolean**	N/A	0-9999	Write Only	Point data clear.

Note: Because the driver ignores the axis number for Speed, Acceleration and Deceleration, an arbitrary number can be used.

Examples

1. To show the current position in the point data table for axis 2 point 3:
P2p,3

2. To show the current acceleration in the point data table for axis 1 point 17:
P1a,17

3. To move to the position in the point data table for axis 1 point 17, write a value to:
PM1,17

Note: PM1,v PM1,a and PM1,d can be set to override the specified point tables velocity, acceleration and deceleration.

4. To clear point 3 from the device memory, write a value to:
PCL3

Variable (X-Sel)

The syntax for accessing any variable is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Program	Number	Access	Functionality
VI<program>,<number>	**DWord**, Long, Float	0-99	200-1299	Read/Write	Integer variable.
VR<program>,<number>	**Double**, Float	0-99	300-1399	Read/Write	Real variable.
VS<program>,<number>.<max length> (max length must be less than 256)	**String**	0-99	300-990	Read/Write	String variable.

Note: In order to access global variables, use a program number of zero ('0').

Examples

1. To read variable 200 for program 0:
V0,200

2. To write the string "Hi there!" to string variables 300 to 308 (one character per location), write the string to:
VS0,300.10. This address is used to write strings up to 10 characters long.

Program Execution (X-SEL)

The syntax for executing or stopping a program is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Number	Access	Functionality
EXE<number>	**Boolean**	0-99	Write Only	Run or stop a program.
PSE<number>	**Boolean**	0-99	Write Only	Pause or continue a program.
STP<number>	**Boolean**	0-99	Write Only	Step a paused program.

Examples

1. To start program 2:
EXE2-write 1

2. To stop the program:
EXE2-write 0

3. To pause the program:
PSE2 – write 1

4. To execute the next step in the program:
STP – write any value

5. To continue execution of the program from current step:
PSE2 – write 0

Test Controller (X-Sel)

The syntax for testing the controller is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Read Write
TST	**Boolean**	Read Only

Example

To test the controller:
TST-The controller is responding properly if the value displayed is 1.

Reset Controller (X-Sel)

The syntax for resetting the controller is as follows. The default data type is shown in **bold**.

Syntax	Data Type	Read Write	Functionality
RSE	**Boolean**	Write Only	Reset errors.
RSS	**Boolean**	Write Only	Reset controller software.
RSD	**Boolean**	Write Only	Reset driver (controller).
RP	**Boolean**	Write Only	Release pause.

Examples

1. RSS- Writing any value to this address will reset the controller's software.
2. RSE- Writing any value to this address will clear all errors from the device.

Error Messaging (X-Sel)

The syntax for accessing an error message from an error code is as follows. The default data types are shown in **bold**.

Syntax	Data Type	Read Write	Functionality
ERS	**String**	Read Only	Displays highest level system error message.
ERA<axis>	**String**	Read Only	Displays the last axis error message.
ERP<program>	**String**	Read Only	Displays the last program error message.

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

Serial Communications

COMn does not exist
Error opening COMn
COMn is in use by another application
Unable to set comm properties on COMn
Communications error on '<channel name>' [<error mask>]

Device Status Messages

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

IA Super SEL Device Specific Messages

Bad address in block [<start address> to <end address>] on device '<device name>'

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 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 the address is correct; if it is not, re-enter it in the client application. Also verify 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.

Serial Communications

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

Serial Communications
COMn does not exist
Error opening COMn
COMn is in use by another application
Unable to set comm properties on COMn
Communications error on '<channel name>' [<error mask>]

COMn does not exist

Error Type:
Fatal

Possible Cause:
The specified COM port is not present on the target computer.

Solution:
Verify that the proper COM port has been selected.

Error opening COMn

Error Type:
Fatal

Possible Cause:
The specified COM port could not be opened due an internal hardware or software problem on the target computer.

Solution:
Verify that the COM port is functional and may be accessed by other Windows applications.

COMn is in use by another application

Error Type:
Fatal

Possible Cause:
The serial port assigned to a device is being used by another application.

Solution:
Verify that the correct port has been assigned to the channel.

Unable to set comm properties on COMn

Error Type:
Fatal

Possible Cause:
The serial properties for the specified COM port are not valid.

Solution:
Verify the serial properties and make any necessary changes.

Communications error on '<channel name>' [<error mask>]

Error Type:
Serious

Error Mask Definitions:
B= Hardware break detected.
F = Framing error.
E = I/O error.
O = Character buffer overrun.
R = RX buffer overrun.
P = Received byte parity error.
T = TX buffer full.

Possible Cause:

1. The serial connection between the device and the Host PC is bad.
2. The communications properties for the serial connection are incorrect.

Solution:

1. Verify the cabling between the PC and the device.
2. Verify that the specified communications properties match those of the device.

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 serial connection between the device and the Host PC is broken.
2. The communications properties for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.
4. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device property.

Solution:

1. Verify the cabling between the PC and the device.
2. Verify that the specified communications properties match those of the device.
3. Verify that the Network ID given to the named device matches that of the actual device.
4. Increase the Request Timeout property so that the entire response can be handled.

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

Error Type:
Serious

Possible Cause:

1. The serial connection between the device and the Host PC is broken.
2. The communications properties for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.

Solution:

1. Verify the cabling between the PC and the device.
2. Verify that the specified communications properties match those of the device.
3. Verify that the Network ID given to the named device matches that of the actual device.

IA Super SEL Device Specific Messages

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

IA Super SEL Device Specific Messages

Bad address in block [<start address> to <end address>] on device '<device name>'

Bad address in block [<start address> to <end address>] on device '<device name>'

Error Type:
Serious

Possible Cause:
An attempt has been made to reference a nonexistent location in the specified device.

Solution:
Verify the tags assigned to addresses in the specified range on the device and eliminate ones that reference invalid locations.

Index

F

I

L

M

O

P

R

S

T

U

V

W

X