



Scripting with the JI-300 I2C Bus Adapter

Description

Scripting is now available on the JI-300 I2C bus adapter via software version 4.1. This new feature provides an alternate method of creating and editing I2C messages outside the JI-300 application environment by the use of a text editor such as Windows Notepad. The JI-300 scripting language includes both I2C commands and JI-300 instructions. Script files are loaded into the JI-300 application and executed as a conventional Message List session.

Procedure

Two script functions are provided: **Import** and **Export**. Import reads a script file into the I2C Message List and Export writes I2C messages from the I2C Message List to a script file. These functions are located at the **File** tab at the main window bar.

The best way to get started creating scripts is to create sample I2C messages using the standard I2C message entry method (i.e. Add Message Window) and then exporting to a script file. Use these messages as prototypes or models of other I2C messages that need to be created. Using your text editor, simply cut, paste, and edit I2C messages to create the desired functions. Script statements are listed below.

Script Statements

I2C Commands

1. I2C Write

Description: This command is used to write data to a slave device. A maximum of 32 and a minimum 0 bytes can be transmitted. The first two characters following the write character (w) specifies the slave address. The remaining characters constitute the data payload. A carriage-return and line-feed terminates the statement.

Syntax: `waaaa.....aa<CR><LF>`

Address & Direction: `aa` Upper 7-bits = slave address. LSB = direction (0 = write)
Range = *even bytes* from 00h to FEh.

Data: `aa.....aa` Write Data 32 bytes maximum.

Example:

| Command | Description |
|--|---|
| <code>wa031abc34892<CR><LF></code> | Write 31h, abh, c3h, 48h, and 92h to slave address 50h. |
| <code>w2c1235ab<CR><LF></code> | Write 12h, 35h, and abh to slave address 16h. |

2. I2C Write without a Stop

Description: This command is identical to the "I2C Write" command, except that it does not issue an I2C stop event .

Syntax: `yaaaa.....aa<CR><LF>`

Address
& Direction: `aa` Upper 7-bits = slave address. LSB = direction (0 = write)
Range = *even bytes* from 00h to FEh.

Data: `aa.....aa` Write Data 32 bytes maximum.

Example:

| <u>Command</u> | <u>Description</u> |
|--|---|
| <code>ya031abc34892<CR><LF></code> | Write 31h, abh, c3h, 48h, and 92h to slave address 50h. |
| <code>y2c1235ab<CR><LF></code> | Write 12h, 35h, and abh to slave address 16h. |

3. I2C Read

Description: This command is used to read data from a slave device. A maximum of 32 and a minimum 0 bytes can be read. The first two characters following the read character (q) specifies the number of bytes to read. The next two characters specifies the slave address. A carriage-return and line-feed terminates the statement.

Syntax: `qaaaa<CR><LF>`

Read Count: `aa` Range 00h to 20h.

Address
& Direction: `aa` Upper 7-bits = slave address. LSB = direction (1 = read)
Range = *odd bytes* from 01h to FFh.

Example:

| <u>Command</u> | <u>Description</u> |
|--|---------------------------------------|
| <code>q14a3<CR><LF></code> | Read 20 bytes from slave address 51h. |
| <code>q0ab1<CR><LF></code> | Read 10 bytes from slave address 58h. |

4. I2C Read without Stop

Description: This command is identical to the "I2C Read" command, except that it does not issue an I2C stop event .

Syntax: `daaaa<CR><LF>`

Read Count: `aa` Range 00h to 20h.

Address
& Direction: `aa` Upper 7-bits = slave address. LSB = direction (1 = read)
Range = *odd bytes* from 01h to FFh.

Example:

| <u>Command</u> | <u>Description</u> |
|----------------|---------------------------------------|
| d14a3<CR><LF> | Read 20 bytes from slave address 51h. |
| d0ab1<CR><LF> | Read 10 bytes from slave address 58h. |

JI-300 Instructions

1. Stop

Description: This instruction stops message execution during an I2C session.

Syntax: *S<CR><LF>

Address: N/A

Data: N/A

Example:

| <u>Command</u> | <u>Description</u> |
|----------------|-------------------------|
| *S<CR><LF> | Stop message execution. |

2. NOP

Description: This instruction performs no operation during an I2C session. It is typically used to temporarily remove an I2C command while saving the location in the message list.

Syntax: *N<CR><LF>

Address: N/A

Data: N/A

Example:

| <u>Command</u> | <u>Description</u> |
|----------------|-----------------------|
| *N<CR><LF> | Perform no operation. |

3. Comment

Description: Use this instruction to add comments to the message list.

Syntax: *C~aaa...aaa~<CR><LF>

Address: N/A

Data: N/A

Example:

| <u>Command</u> | <u>Description</u> |
|--------------------------|---|
| * C~Hello world~<CR><LF> | Adds the comment "Hello world" to the message list. |

