

**3100/3150 - EMC
Emerson FX Drive
Master Module**
Revision 1.2

USER MANUAL

November 1997

ProSoft Technology, Inc.
(661) 716-5100

(661) 716-5101 (fax)

E-mail address: prosoft@prosoft-technology.com

Product Revision History

06/09/96	Revision 1.0 Initial release of product
11/02/97	Revision 1.2 Added support for ML and FM commands

Implementation Guide

Integration of the EMC module into a PLC or SLC application is easier the first time if a series of steps are followed. In order to assist the first time users of our products in getting the EMC operational quickly, we have come up with this step-by-step implementation guide.

- a) Starting with one of the ladder logic programs provided on disk with the EMC complete the following steps:

PLC 5	EMC5
SLC 5/03	EMC503

- b) Edit the ladder logic provided on disk as needed for the application
Verify rack and slot location in program
Modify ladder instruction addresses as needed
Reference Appendix for tips in the SLC platform
- c) Setup the Communication Configuration parameters (See Section 2)
Determine each port's communication configuration requirements:
Baud Rate, RTS delay requirements
(Parity is fixed at None and Stop Bits are fixed at 1)
(Drive must be configured in Full Duplex Mode, with Auto CR/LF Enabled and Axis Identifier)
- d) Setup the Command List for each port (See Section 2)
- e) Identify the jumper requirements (See Appendix)
- f) Make up the communication cables (See Section 5)
- g) Place processor into the run mode
- h) Monitor the data table Error Status values (See Section 2)

Table of Contents

Revision History	i
Implementation Guide	ii
1 Product Specifications	1
General Specifications	1
2 EMC Theoretical Operation	2
2.1 Block Transferring Data to the Module.....	2
2.1.1 Communications Configuration [BTW Block ID 255].....	2
2.1.2 Command Block [BTW Block ID Code 0 and 1].....	4
2.2 Block Transferring data from the module	6
2.2.1 Command Block Results [BTR Block ID and 1].....	7
2.2.2 Module Information Table [BTW Block ID 255].....	8
3 Protocol Commands.....	10
4 Hardware Diagnostics.....	11
4.1 3100 PLC Platform	11
4.2 3150 SLC Platform	12
5 Cable Connection	14
6 Technical Support	15

Appendix

Support, Service and Warranty
Jumper Configurations
FX Drive Op Code List
SLC Programming Considerations
Example Ladder Logic
PLC-5
SLC-5/03

1 Product Specifications

The 3100/3150-EMC ("Emerson FX Drive Master Module") product family allows Allen-Bradley 1771 and 1746 I/O compatible processors to easily interface as a host with up to 8 Emerson FX Series Drives (4 per port).

The EMC product includes the following standard features:

General Specifications

- Two fully configurable serial ports, each capable of supporting the Emerson FX Drive Master functionality
- Support for the storage and transfer of up to 12 Commands (6 per port) to the PLC /SLC data tables
- RS-232C communications
- Software configuration (From processor ladder logic)
 - VFP and DDP : Power of 10 selectable per Drive Addr
 - Drive Addr : 1 to 4 are valid
 - Baud Rate : 300 TO 38,400
 - RTS to TxD : 0-65535 ms, 1 ms resolution
- Communication parameters default to :
 - 1 Start Bit
 - 8 Data Bits
 - 1 Stop Bit
 - Full Duplex
- Response time
 - The protocol drivers are written in Assembly and in a compiled higher level language. As such, the interrupt capabilities of the hardware are fully utilized to minimize delays, and to optimize the product's performance
- A full listing of the supported EMC Function codes is available in the Appendix
- Supports up to 4 slave devices per port
- Error Codes returned to the ladder processor per Drive Addr

Hardware Specifications

- Backplane Current Load :
 - 3100 : 0.65 A
 - 3150 : 0.15 A at 5 V
 - 0.04 A at 24 V
- Operating Temperature : 0 to 60 °C
- Storage Temperature : -40 to 85 °C
- Connections :
 - 3100 : 2 - DB25 Female Connectors
 - 3150 : 2 - DB9 Male Connectors

2 EMC Theoretical Operation

Data transfers between the processor and the ProSoft Technology module occur using the Block Transfer commands, in the case of the PLC, and M0/M1 data transfer commands, in the case of the SLC. These commands transfer up to 64 physical registers per transfer. The logical data length changes depending on the data transfer function.

The following discussion details the data structures used to transfer the different types of data between the ProSoft Technology module and the processor. The term 'Block Transfer' is used generically in the following discussion to depict the transfer of data blocks between the processor and the ProSoft Technology module. Although a true Block Transfer function does not exist in the SLC, we have implemented a pseudo-block transfer command in order to assure data integrity at the block level. Examples of the PLC and SLC ladder logic are included in Appendix A.

In order for the ProSoft Technology module to function, the PLC must be in the RUN mode, or in the REM RUN mode. If in any other mode (Fault/PGM), the block transfers between the PLC and the module will stop, and communications will halt until block transfers resume.

2.1 Block Transferring Data to the Module

Data transfer to the module from the processor is executed through the Block Transfer Write function. The different types of data which are transferred require slightly different data block structures, but the basic data structure is:

<u>Word</u>	<u>Description</u>
0	BTW Block ID code
1-63	Data

Although the full physical 64 words of the data buffer may not be used, the BTW and M0 lengths must be configured for 64 words otherwise module operation will be unpredictable.

Where:

BTW Block ID Code: A block identifier code between 0 and 255 in value. This code is used by the ProSoft module to determine what to do with the data block. Valid codes are:

<u>BTW Code</u>	<u>Description</u>
0,1	Command Block
255	Module Communication Configuration

Data: The data to be written to the module. The structure of the data is dependent on the Block ID code. The following sections provide details on the different structures.

2.1.1 Communications Configuration [BTW Block ID 255]

The ProSoft Technology firmware communication parameters must be configured at least once when the card is first powered up, and any time thereafter when the parameters must be changed.

On power up, the module enters into a logical loop waiting to receive configuration data from the processor. While waiting, the module sets the second word of the BTR buffer to 255, telling the processor that the module must be configured before anything else will be done. The module will continuously perform block transfers until the communications configuration

parameters block is received. Upon receipt, the module will begin execution of the command list if present, or begin looking for the command list from the processor.

Transferring the Communications Configuration Parameters to the module will force a reset of the communication ports

The configuration data block structure which must be transferred from the processor to the module is as follows:

BTW Block ID 255

<u>Word</u>	<u>Description</u>
0	BTW Block ID = 255
1-10	Port 1 Configuration parameters
11-20	Port 2 Configuration parameters
21-30	System Configuration parameters

<u>BTW Buffer</u>	<u>Data Word</u>	<u>Description</u>
0		Block ID Header = 255

Port 1 Configuration

1	N[]:0	Baud Rate
2	N[]:1	RTS to TXD Delay
3	N[]:2	VDP Drive Addr 1
4	N[]:3	DDP Drive Addr 1
5	N[]:4	VDP Drive Addr 2
6	N[]:5	DDP Drive Addr 2
7	N[]:6	VDP Drive Addr 3
8	N[]:7	DDP Drive Addr 3
9	N[]:8	VDP Drive Addr 4
10	N[]:9	DDP Drive Addr 4

Port 2 Configuration

11	N[]:10	Baud Rate
12	N[]:11	RTS to TXD Delay
13	N[]:12	VDP Drive Addr 1
14	N[]:13	DDP Drive Addr 1
15	N[]:14	VDP Drive Addr 2
16	N[]:15	DDP Drive Addr 2
17	N[]:16	VDP Drive Addr 3
18	N[]:17	DDP Drive Addr 3
19	N[]:18	VDP Drive Addr 4
20	N[]:19	DDP Drive Addr 4

System Configuration

21	N[]:20	N/A
22	N[]:21	N/A
23	N[]:22	N/A
24	N[]:23	N/A
25	N[]:24	N/A
26	N[]:25	N/A
27	N[]:26	N/A
28	N[]:27	N/A
29-30		N/A

Configuration Memory map for Example Application

	0	1	2	3	4	5	6	7	8	9	
N7:0	1	0	0	0	0	0	0	0	0	0	Port #1 Configuration
N7:10	1	1	0	0	0	0	0	0	0	0	Port #2 Configuration
N7:20	0	0	0	0	0	0	0	0	0	0	System Configuration (Not currently in use)

Baud Rate: 0
 RTS to TxD Delay: 1
 VDP/DDP Pairs (1 to 4): 2, 3

IMPORTANT

The FX Drive must be configured in the full-duplex mode with Auto CR/LF Enabled to operate with the EMC module. Additionally, the Drive must be configured with an Axis Identifier of 1 or more (This is the Slave address). This is true regardless of whether the FX drives are in multi-drop mode or not

Where:

Baud Rate: The baud rate at which the module is to operate. The baud rate is configured as follows:

<u>Value</u>	<u>Baud Rate</u>
0	300 Baud
1	600 Baud
2	1200 Baud
3	2400 Baud
4	4800 Baud
5	9600 Baud
6	19200 Baud
7	38400 Baud

RTS To TXD Delay: This value represents the time in 1 ms increments to be inserted between asserting RTS, and the actual transmission of data. The delay, if greater in duration than the hardware time delay associated with CTS, will override the CTS line until the time-out is complete.

This configurable parameter is useful when interfacing with modem based devices, or anytime line noise must be allowed to subside before data is transmitted.

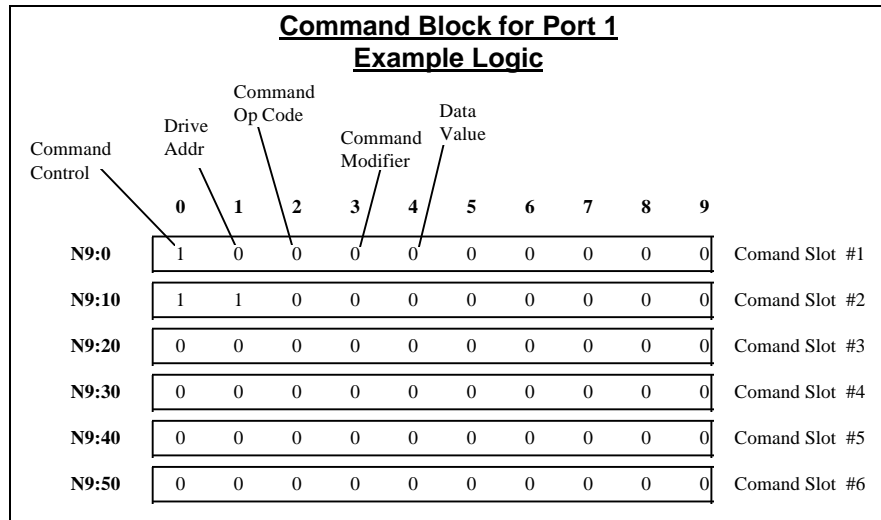
VDP/DDP: These values are entered on a per active slave address on each port. The values entered here are used with selected commands when reading and writing data value between the drive and the EMC module. See the FX Protocol Manual and the Appendix for a listing of commands which use these values for scaling purposes.

2.1.2 Command Block [BTW Block ID Code 0 and 1]

An EMC Master port establishes communications and performs various communications functions based on the data which the user has placed in the Command Block. The Command Block is a 60 word data block containing 6 Command Slots. Each slot is capable of performing a separate command to any Drive Addr.

This command data, entered into the processor Data Table, is transferred to the module's memory using Block IDs 0 and 1, depending on the port to be commanded.

<u>Word</u>	<u>Description</u>
0	BTW Block ID Code 0 - Port 1 1 - Port 2
1 to 10	Command Slot #1
11 to 20	Command Slot #2
21 to 30	Command Slot #3
31 to 40	Command Slot #4
41 to 50	Command Slot #5
51 to 60	Command Slot #6



where:

BTW Block ID: The block identifier number allows the EMC Module to identify which port the Command Block is to be assigned to.

Command Slot #x: The Command Block is broken down into 6 separate Command Slots. Each Command Slot is capable of executing an individual command, allowing a total of 6 commands to be executed per Block Transfer sequence. The structure of the Command Slot is as follows:

Position	Name	Description
0	Command Select	Provides the ability for the ladder logic to select the way the Op Code selected in the Command Op Code field is to be executed. The choices are as follows: 0 - Null (do nothing) 1 - Assign 2 - Query 3 - Execute
1	Drive Address	Selects the individual drive to which the command will be sent. Valid entries in this field are 1 to 4, inclusive
2	Command Op Code	Selects the FX Drive Op Code which is to be executed. A numerical value has been assigned to each supported Op Code (See Appendix). The Op Code

		is selected by entering a decimal value for the desired command
3	Command Modifier	If an Op Code allows a modifier to be sent then enter the desired modifier value in this field
4	Data Value	If the Command is to be sent as an Assign command then enter the data value which is to be assigned to the drive
5-9	Spare	Not used at this time

2.2 Block Transferring data from the module

When the Master port driver reads data from a slave, the resulting data is placed into the ProSoft module's data space. This Module Data space is then transferred to the PLC/SLC ladder program.

The transfer of data from the ProSoft Technology module to the processor is executed through the Block Transfer Read function. The data structure for the block transfer depends on the type of block data. The following sections detail the different types of data.

Although the full physical 64 words of the data buffer may not be used, the BTR and M1 lengths must be configured for a length of 64 words, otherwise module operation will be unpredictable

The ladder logic must be programmed to look at the BTR buffer, decode several words, and then take action. The BTR buffer definition is:

<u>Word</u>	<u>Description</u>
0	BTR Block ID
1	BTW Block ID
2-62	Data

Where:

BTR Block ID Number: The ladder logic uses this value to determine the contents of the data portion of the BTR buffer. With some conditional testing in ladder logic, the data from the module can be placed into the PLC/SLC data table.

<u>BTR Code</u>	<u>Description</u>
0,1	Command Block Results

BTW Block ID Number: The module returns this value to the processor to be used to enable the movement of the Command Blocks to the module. The BTW Block ID number is developed by the module. This value is intended to only be a suggestion and to ease the ladder logic programming requirements. If it is desired to develop a different data transfer series, this may be easily accomplished in ladder logic.

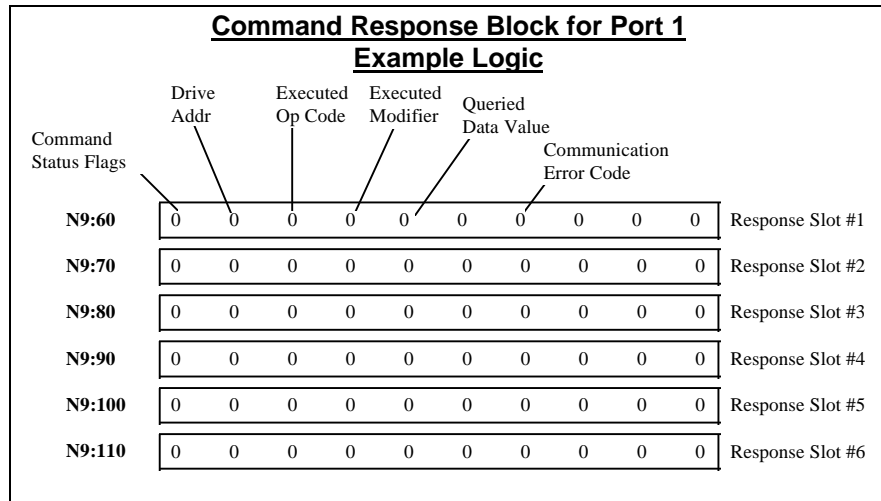
<u>BTW Code</u>	<u>Description</u>
0,1	Command Block Results
255	Module Configuration and Information

Data: Values being transferred from the module to the ladder logic. These will consist of either Command Block results or Module Information data.

2.2.1 Command Block Results [BTR Block ID and 1]

This command data, entered into the processor Data Table, is transferred to the module's memory using Block IDs 0 and 1, depending on the port to be commanded.

<u>Word</u>	<u>Description</u>
0	BTR Block ID Code 0 - Port 1 1 - Port 2
1	BTW Block ID
2 to 11	Command Response Slot #1
12 to 21	Command Response Slot #2
22 to 31	Command Response Slot #3
32 to 41	Command Response Slot #4
42 to 51	Command Response Slot #5
52 to 61	Command Response Slot #6



where:

BTR Block ID: The block identifier number allows the EMC Module to identify which port the Command Block is to be assigned to.

BTW Block ID: The module returns this value to the processor to be used to enable the movement of the Command Blocks to the module. Reference Section 2.1

Command Response Slot #x: The Command Response Slot represents the status and data results of each of the individual Commands executed by the module. The results correspond to the Command Slot directly (i.e., Command Slot #1 results will be returned in Command Response Slot #1). The structure of the Command Response Slot is as follows:

Position	Name	Description								
0	Command Status Flags	The completion status of the command is returned in this register in a high/low byte fashion. The high byte contains a bit mapped status: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Done - Successful</td> </tr> <tr> <td>4</td> <td>Error - Cmd Failed</td> </tr> <tr> <td></td> <td>See Comm Status field</td> </tr> </tbody> </table>	Bit	Description	7	Done - Successful	4	Error - Cmd Failed		See Comm Status field
Bit	Description									
7	Done - Successful									
4	Error - Cmd Failed									
	See Comm Status field									

		The low byte contains the command type which was executed: 1 Assign 2 Query 3 Execute
1	Drive Address	The individual drive to which the command was sent. Valid entries in this field are 1 to 4, inclusive
2	Executed Op Code	The FX Drive Op Code which was executed. A numerical value has been assigned to each supported Op Code (See Appendix). The Op Code is selected by entering a decimal value for the desired command
3	Executed Modifier	The modifier which was sent
4	Data Value	If the Command which was completed was a Query command then the data value which was received from the drive will be returned here.
5	Spare	Not used at this time
6	Communication Status Code	If the command was executed successfully (Bit 15 of Command Status Flags set) then this field will be 0. If Error Bit is set in the Flags, then the value in this field will indicate the reason for failure. The status codes are: 1 Bad Query '=' not received 2 Command Not Ack ** not received to Assign or Execute commands 3 Error in Response The echo did not match the value transmitted 4 Bad Response A '?' was received 5 Bad Op Code The Op Code is not supported by the module 8 Response Timeout 255 Transmition Timeout Check RTS to CTS jumper on the module
7-9	Spare	Not used at this time

2.2.2 Module Information Table [BTW Block ID 255]

The EMC Module provides product data to the ladder logic during power up through the BTR data buffer whenever the BTW Block ID is set to 255. This data is useful for determining revision information and can be useful should support be necessary from the factory. This 10 word block of data is returned at the bottom of the BTR data.

BTR	Description
Word 51-52	Product Name (ASCII)
53-54	Revision (ASCII)
55-56	Operating System Rev(ASCII)
57-58	Production Run Number (ASCII)
59-60	Spare

Where:

Product Name: These two words represent the product name of the module in an ASCII representation. In the case of the EMC product, the letters ' EMC '

should be displayed when placing the programming software in the ASCII data representation mode.

Revision : These two words represent the product revision level of the firmware in an ASCII representation. An example of the data displayed would be '1.01' when placing the programming software in the ASCII data representation mode.

Operating System Revision : These two words represent the module's internal operating system revision level in an ASCII representation.

Production Run Number: This number represents the 'batch' number that your particular chip belongs to in an ASCII representation.

3 Protocol Commands

The ProSoft Technology EMC module Master driver supports a reasonably extensive subset of the command available for the FX drive. A full list of the supported commands and any limitations is available in the Appendix.

4 Hardware Diagnostics

Several hardware diagnostics capabilities have been implemented using the LED indicator lights on the front of the module. The following sections explain the meaning of the individual LEDs for both the PLC and the SLC platforms.

4.1 3100 PLC Platform

The following table documents the LEDs on the hardware platform and explains the operation of the LEDs.

ProSoft CIM Card		
ACTIVE	○ ○	FLT
CFG	○ ○	BPLN
ERR1	○ ○	ERR2
TXD1	○ ○	TXD2
RXD1	○ ○	RXD2

Table 4.1 : PLC Platform LED Indication

ProSoft CIM	Color	Status	Indication
ACT	Green	Blink (Fast)	<u>Normal state</u> : The module is operating normally and successfully Block Transferring with the PLC
		On	The module is receiving power from the backplane, but there may be some other problem
		Off	The module is attempting to Block Transfer with the PLC and has failed. The PLC may be in the PGM mode or may be faulted
FLT	Red	Off	<u>Normal State</u> : No system problems are detected during background diagnostics
		On	A system problem was detected during background diagnostics. Please contact factory for technical support
CFG	Green	Off	<u>Normal state</u> : No configuration related activity is occurring at this time
		Blink	This light blinks every time a Module Configuration block (ID = 255) is received from the processor ladder logic
		On	The light is on continuously whenever a configuration error is detected. The error could be in the Port Configuration data or in the System Configuration data. See Section 4 for details
BPLN	Red	Off	<u>Normal State</u> : When this light is off and the ACT light is blinking quickly, the module is actively Block Transferring data with the PLC
		On	Indicates that Block Transfers between the PLC and the module have failed.(Not activated in the initial release of the product)
ERR1 ERR2	Amber	Off	<u>Normal State</u> : When the error LED is off and the related port is actively transferring data, there are no communication errors
		Blink	Periodic communication errors are occurring during data communications. See Section 4 to determine the error condition

		On	This LED will stay on under several conditions: <ul style="list-style-type: none"> • CTS input is not being satisfied • Port Configuration Error • System Configuration Error • Recurring error condition on master
Tx1 Tx2	Green	Blink	The port is transmitting data.
Rx1 Rx2	Green	Blink	The port is receiving data

4.2 3150 SLC Platform

The following table documents the LEDs on the hardware platform and explains the operation of the LEDs.

3150-EMC









COMMUNICATIONS			
	ACT		FAULT
	CFG		BPLN
	PRT1		ERR1
	PRT2		ERR2

Table 4.2 : SLC Platform LED Indication

LED Name	Color	Status	Indication
ACT	Green	Blink (Fast)	<u>Normal state</u> : The module is operating normally and successfully Block Transferring with the SLC
		On	The module is receiving power from the backplane, but there may be some other problem
		Blink (1/Sec)	Indicates the module has somehow entered the Basic Programming Mode. Verify jumper JW3 (BAS only) configuration. If all are correct, then contact the factory
FLT	Red	Off	The module is attempting to Block Transfer with the SLC and has failed. The SLC may be in the PGM mode or may be faulted (<i>Not in initial release</i>)
		On	<u>Normal State</u> : No system problems are detected during background diagnostics
		On	A system problem was detected during background diagnostics. Please contact factory for technical support
CFG	Green	Off	<u>Normal state</u> : No configuration related activity is occurring at this time
		Blink	This light blinks every time a Module Configuration block (ID = 255) is received from the processor ladder logic
		On	The light is on continuously whenever a configuration error is detected. The error could be in the Port Configuration data or in the System Configuration data. See Section 4 for details
BPLN	Red	Off	<u>Normal State</u> : When this light is off and the ACT light is blinking quickly, the module is actively Block Transferring data with the SLC
		On	Indicates that Block Transfers between the SLC and the module have failed
ERR1 ERR2	Amber	Off	<u>Normal State</u> : When the error LED is off and the related port is actively transferring data, there are no communication errors
		Blink	Periodic communication errors are occurring during data communications. See Section 4 to determine the error condition

		On	<p>This LED will stay on under several conditions:</p> <ul style="list-style-type: none"> • CTS input is not being satisfied • Port Configuration Error • System Configuration Error • Unsuccessful comm on EMC slave • Recurring error condition on EMC master
TxRx1 TxRx2	Green	Blink	The port is communicating, either transmitting or receiving data

5 Cable Connection

The connection between the 3100/3150-EMC module and the Emerson FX Drives is by RS-232 to the DB connections on the front of the module. The cable connections for both the 3100 and the 3150 units are shown below:

<u>ProSoft Module</u>			<u>FX Cable</u>
<u>3100</u>	<u>3150</u>		<u>9-Pin</u>
<u>25-Pin</u>	<u>9-Pin</u>		
2 TxD	3	-----	2 RxD
3 RxD	2	-----	3 TxD
4 RTS	7	----	
5 CTS	8	-----	
			Must intall jumper between RTS and CTS on
		-----	ProSoft module
7 GND	5	-----	5 GND

6 Technical Support

Technical support for the EMC product will be provided by both Emerson and ProSoft Technology. Emerson is much better suited to provide support on the drive configuration issues which ProSoft Technology will support ladder logic and module configuration issues.

Contact information for both companies is as follows:

ProSoft Technology, Inc.

(805) 664-7208

(800) 326-7066

(805) 664-7233 (fax)

E-mail address: prosoft@prosoft-technology.com

Emerson Drives

(800) 397-3786

(612) 474-1116

(612) 474-8711 fax

8 AM to 5 PM Central Time

Support, Service and Warranty

Technical Support

ProSoft Technology survives on its ability to provide meaningful support to its customers. Should any questions or problems arise, please feel free to contact us at:

Factory/Technical Support

ProSoft Technology, Inc.
9801 Camino Media, Suite 105
Bakersfield, CA 93311
(805) 664-7208
(800) 326-7066
(805) 664-7233 (fax)

E-mail address: prosoft@prosoft-technology.com

Before calling for support, please prepare yourself for the call. In order to provide the best and quickest support possible, we will most likely ask for the following information (you may wish to fax it to us prior to calling):

1. Product Version Number
2. Configuration Information
 - Communication Configuration
 - Jumper positions
3. System hierarchy
4. Physical connection information
 - RS-232, 422 or 485
 - Cable configuration
5. Module Operation
 - Block Transfers operation
 - LED patterns

An after-hours answering service (on the Bakersfield number) allows pager access to one of our qualified technical and/or application support engineers at any time to answer the questions that are important to you.

Module Service and Repair

The EMC card is an electronic product, designed and manufactured to function under somewhat adverse conditions. As with any product, through age, misapplication, or any one of many possible problems, the card may require repair.

When purchased from ProSoft Technology, the module has a one year parts and labor warranty according to the limits specified in the warranty. Replacement and/or returns should be directed to the distributor from whom the product was purchased. If you need to return the card for repair, it is first necessary to obtain an RMA number from ProSoft Technology. Please call the factory for this number and display the number prominently on the outside of the shipping carton used to return the card.

General Warranty Policy

ProSoft Technology, Inc. (Hereinafter referred to as ProSoft) warrants that the Product shall conform to and perform in accordance with published technical specifications and the accompanying written materials, and shall be free of defects in materials and workmanship, for the period of time herein indicated, such warranty period commencing upon receipt of the Product.

This warranty is limited to the repair and/or replacement, at ProSoft's election, of defective or non-conforming Product, and ProSoft shall not be responsible for the failure of the Product to perform specified functions, or any other non-conformance caused by or attributable to: (a) any misapplication or misuse of the Product; (b) failure of Customer to adhere to any of ProSoft's specifications or instructions; (c) neglect of, abuse of, or accident to, the Product; or (d) any associated or complementary equipment or software not furnished by ProSoft.

Limited warranty service may be obtained by delivering the Product to ProSoft and providing proof of purchase or receipt date. Customer agrees to insure the Product or assume the risk of loss or damage in transit, to prepay shipping charges to ProSoft, and to use the original shipping container or equivalent. Contact ProSoft Customer Service for further information.

Limitation of Liability

EXCEPT AS EXPRESSLY PROVIDED HEREIN, PROSOFT MAKES NO WARRANTY OF ANY KIND, EXPRESSED OR IMPLIED, WITH RESPECT TO ANY EQUIPMENT, PARTS OR SERVICES PROVIDED PURSUANT TO THIS AGREEMENT, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NEITHER PROSOFT OR ITS DEALER SHALL BE LIABLE FOR ANY OTHER DAMAGES, INCLUDING BUT NOT LIMITED TO DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION IN CONTRACT OR TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), SUCH AS, BUT NOT LIMITED TO, LOSS OF ANTICIPATED PROFITS OR BENEFITS RESULTING FROM, OR ARISING OUT OF, OR IN CONNECTION WITH THE USE OR FURNISHING OF EQUIPMENT, PARTS OR SERVICES HEREUNDER OR THE PERFORMANCE, USE OR INABILITY TO USE THE SAME, EVEN IF PROSOFT OR ITS DEALER'S TOTAL LIABILITY EXCEED THE PRICE PAID FOR THE PRODUCT.

Where directed by State Law, some of the above exclusions or limitations may not be applicable in some states. This warranty provides specific legal rights; other rights that vary from state to state may also exist. This warranty shall not be applicable to the extent that any provisions of this warranty is prohibited by any Federal, State or Municipal Law that cannot be preempted.

Hardware Product Warranty Details

Warranty Period : ProSoft warranties hardware product for a period of one (1) year.

Warranty Procedure : Upon return of the hardware Product ProSoft will, at its option, repair or replace Product at no additional charge, freight prepaid, except as set forth below. Repair parts and replacement Product will be furnished on an exchange basis and will be either reconditioned or new. All replaced Product and parts become the property of ProSoft. If ProSoft determines that the Product is not under warranty, it will, at the Customer's option, repair the Product using current ProSoft standard rates for parts and labor, and return the Product freight collect.

Jumper Configurations

When purchasing the EMC product, there are two available configurations. These choices are as follows:

Description	ProSoft Cat Num	
	<u>PLC</u>	<u>SLC</u>
Module provided by ProSoft	3100	3150

When purchasing the module from ProSoft Technology, the jumper configurations will have been factory set to default positions for testing prior to shipment..

Module Jumper Configurations

The following section details the available jumper configurations for the 1771 and 1746 platform solutions. As needed, differences between the module based solutions and the firmware based solutions are highlighted.

3100 for the 1771 Platform

Following are the jumper positions for the ProSoft Technology 3100 module:

<u>Jumper</u>	<u>3100</u>
JW1	N/A
JW2	N/A
JW3	N/A
JW4	Flash Pgm/Run Mode
JW5	8 Pt
JW6	Not Used
JW7	Enabled
JW8	Port 2 RS232/422/485 config
JW9	Port 1 RS232/422/485 config

JW4	Flash Pgm/Run Mode Select	Run Position
	The position of this jumper should only be changed if needing to reprogram the FLASH memory. This will only need to be done if the module is to be upgraded in the field to a later version of firmware.	
JW5	Backplane 8/16 point	8 Point
	The module should be operated in the 8 point configuration unless specifically directed otherwise by the factory.	
JW7	Battery Enable / Disable	Enabled
	This jumper should be placed in the Enabled position when the module is powered up. Although not critical to the operation of the module, this will back up some data registers in the module during a power failure or reset.	
JW8/9	RS Configuration for Port 1 and 2	RS-232,422,485
	The default from factory is RS-232, but all options are supported by the firmware	

3150 for the 1746 Platform

Following are the jumper positions for the ProSoft Technology 3150 module :

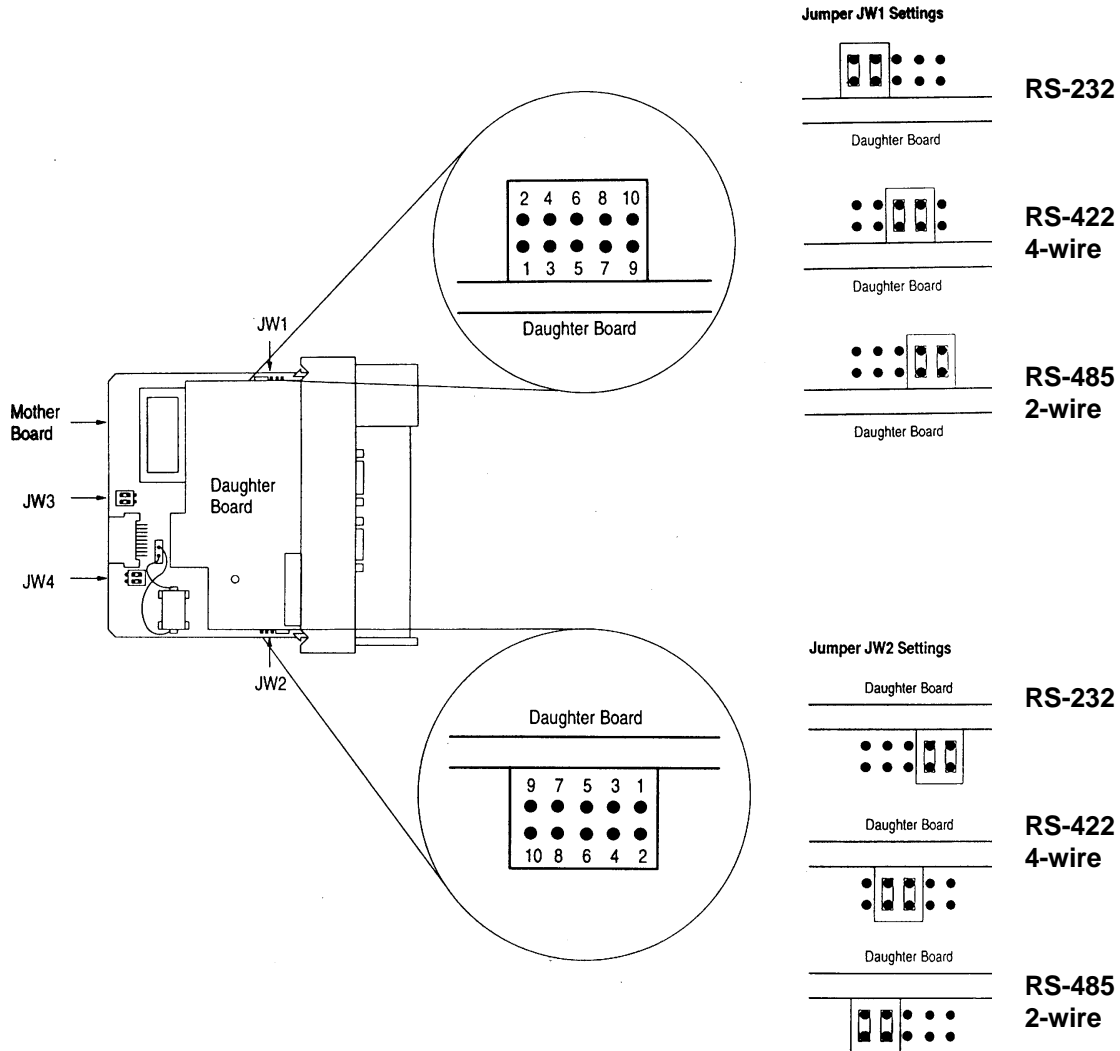
<u>Jumper</u>	<u>3150</u>
JW1	As Needed
JW2	As Needed
JW3	N/A
JW4	N/A

JW1/2 RS configuration for port 1 and 2

RS-232 Position

The default from factory is RS-232, but RS-422 and RS-485 are supported by the firmware and hardware. See the following diagram:

**Communication Port
Jumper Settings for 3150/3151 Modules - JW1 & JW2**



SLC Programming Considerations

The 3150-EMC is also very easy to get operational. After the IEMC cover and firmware are installed and the jumpers have been configured, the module is ready to be configured.

In order to implement the sample logic, the user must make sure that the correct processor and rack size match up. Also, should it be necessary to re-locate the EMC module, the user should be certain to configure the correct slot as a 1746-BAS 5/02 Configuration.

When initially setting up the SLC program file, or when moving the module from one slot to another, the user must configure the slot to accept the EMC module.

It is important that the slot containing the ProSoft module be configured as follows:

- 1746-BAS module or enter 13106 for the module code
- Configure the M0/M1 files for 64 words
- Configure I/O for 8 words

The following is a step by step on how to configure these files using Allen-Bradley APS software. ICOM software users should follow similar steps.

From the Main Menu:

- 1) Select the correct processor program and F3 for Offline programming
- 2) F1 for Processor Functions
- 3) F1 for Change Processor
 Modify the processor here if necessary (Note the EMC will only work with 5/02 or greater processors)
- 4) F5 for Configure I/O
 Select 1746-BAS module for SLC 5/02 or greater, or enter 13106 for module code
- 5) F9 for SPIO Config when the correct slot is highlighted
- 6) F5 Advanced Setup
- 7) F5 for M0 file length - type in 64 and Enter
- 8) F6 for M1 file length - type in 64 and Enter

Esc out and save configuration

FX Drive Command OpCodes

FX Serial Command	Command Opcode (Dec)	Command Modifier	Command Title	DDP, VDP or None	Data Type	Base of Data	Data Range	Query	Assignment	Execution
AT	19	0-255	Index Accel. Time (msec)	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
BK	36	N/A	Brake Override	None	8 bit uns. int	decimal	0 / 1	1	1	0
BR	43	N/A	Brake Inhibit	None	8 bit uns. int	decimal	0 / 1	1	1	0
CF	57	0-5	Analog Channel Full Scale Value (hex)	None	16 bit uns. int.	hex	0 - 3FFF	1	1	0
CH	59	0-5	Analog Channel Reading	None	16 bit uns. int.	hex	0 - 3FFF	1	0	0
CO	66	0-255	Index Repeat Count	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
CP	67	0-1	Command Position - absolute(0), and display(1)	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
CV	73	0-1	Current Command Velocity scaled see VC0, VC1	VDP	16 bit uns. int.	decimal	0 - 50000	1	0	0
DS	96	0-255	Index Distance or Position (signed)	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
DT	97	0-255	Index Decel Time	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
DW	100	0-255	Index Dwell Time	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
EP	119	0-1	External Encoder Position - 32 bit(0), user units(1)	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
FA	130	0-255	Index Feed Accel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
FD	133	0-255	Index Feed Decel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
FE	134	N/A	Following Error	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	0	0
FH	137	N/A	Hardware Fault Status	None	ASCII	N/A	1 char, 0 - 9 or A - F	1	0	0
FM	142	0-15	Follower/Master Ration array	None	16 bit uns. Int	Decimal	0 - 65535	1	1	0
FP	145	N/A	Feedback Position	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	0	0
FS	148	N/A	Fault Status	None	16 bit uns. int.	hex	0 - FFFF	1	0	0
FV	151	0-255	Index Feed Velocity	VDP	16 bit uns. int.	decimal	0 - max Velocity	1	1	0
HA	182	0-1	Home Accel Ramp Time	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
HD	185	0-1	Home Decel Ramp Time	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
HF	187	N/A	Serial Hold Command	None	8 bit uns. int	decimal	0 - 255	1	1	0
HI	190	0-1	Home Feed On Velocity	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	1	0
HL	193	0-1	Home Distance Limit	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
HM	194	N/A	Execute Home When in Serial Control	None	N/A	N/A	N/A	0	0	1
HO	196	0-1	Home Feed Off Velocity	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	1	0
HP	197	0-1	Home Position	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
HR	199	0-1	Home Resolver Velocity	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	1	0
IC	210	0-255	Index Feed Torque Limit	None	8 bit uns. int	decimal	0 - 200	1	1	0
ID	211	0-4	Drive Identifier	None	ASCII	N/A	up to 25 char	1	0	0
IN	221	0-255	Default index to be run using input (IN=5)	None	8 bit uns. int	decimal	0 - 255	1	1	1
IO	222	0-19	Input/Output Monitoring (IO15=inputs, IO16=outputs)	None	32 bit uns. int.	hex	depends on modifier	1	1	0

JA	234	N/A	Jog Accel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
JD	237	N/A	Jog Decel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
JF	239	N/A	Jog Fast Velocity	VDP	16 bit uns. int.	decimal	0 - max velocity	1	1	1
JG	240	N/A	Jog Serial Velocity (+/- may be added for direction)	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	1	1
JS	252	N/A	Jog Slow Velocity	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	1	1
LD	289	0-255	Index Feed Limit Distance	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
LS	303	0-15	PLS Positions	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
LW	307	0-15	PLS Pulse Width	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
LX	308	0-15	PLS PGO Pattern Mask	None	16 bit uns. int.	hex	0 - 65535	1	1	0
LY	309	0-15	PLS PGO Pattern	None	16 bit uns. int.	hex	0 - 65535	1	1	0
ML	323	0-10	Master Cycle Length	None	32 bit sgn, int	Decimal	-214.... - 2147483647	1	1	0
PC	392	0-255	Program Count	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
PF	395	N/A	Power-up Flags	None	8 bit uns. int	hex	0 - FF	1	1	0
PL	401	N/A	Position Limits	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
PR	407	0-1	Program Execute/Sets or Queries Def/Spnd Progs	None	8 bit uns. int	decimal	0 - 255	1	1	1
RA	442	0-3	Current Resolver Changes, Web fault limits	DDP	16 bit uns. int.	decimal	0 - 65535	1	0	0
RD	445	0-8	PCM-18 Commands	DDP	16 bit uns. int.	decimal	0 - 65535	1	1	0
RO	456	0-255	Home Resolver Offset	DDP	16 bit uns. int.	decimal	0 - 4096	1	1	0
RP	457	N/A	Rotary Absolute Position	DDP	32 bit uns. int.	decimal	0 - 2147483647	1	1	0
RR	459	0	Initialize New Roll Ratio	None	N/A	N/A	N/A	0	0	1
RS	460	N/A	Reset Drive	None	N/A	N/A	N/A	0	0	1
RT	461	N/A	Real-Time Clock	None	32 bit sgn. int	decimal	-214.... - 2147483647	1	0	0
RW	464	0-3	Web Ratio Commands	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
SC	470	N/A	Serial Control	None	8 bit uns. int	decimal	0 - 255	1	1	0
SD	471	N/A	Stop Decel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
SJ	477	N/A	Stop Jog	None	N/A	N/A	N/A	0	0	1
SP	483	N/A	Suspend/Resume Program Flag	None	8 bit uns. int	decimal	0 - 255	1	1	0
SR	485	0-3	User Steps per Rev(0)/Ext. Steps per Rev(1),slip(2,3)	DDP	16 bit uns. int.	decimal	50 - 65535	1	1	0
SS	486	N/A	Steps/sec. max. expected from master sync source	None	32 bit uns. int	decimal	0 - 2147483647	1	1	0
ST	487	N/A	Stop All Motion	None	N/A	N/A	N/A	0	0	1
SV	489	N/A	Sync Velocity	VDP	16 bit uns. int.	decimal	0 - max Velocity	1	1	0
TD	497	N/A	Travel Limit Decel Ramp	None	16 bit uns. int.	decimal	0 - 65535	1	1	0
TL	505	0-1	Travel Limit - CCW(0), CW(1)	DDP	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
TN	507	0-14	Loop Parameters	None	16 bit sig. int	decimal	- 1000 to + 1000	1	1	0
TQ	510	0-1	Torque Limit Hit Reset/Sets,Queries Set(0), mon(1)	None	8 bit uns. int	decimal	0 - 255	1	1	1
TV	516	0-1	Drive Velocity in VC0/VC1 User Units	VDP	16 bit sig. int	decimal	- max vel to + max vel	1	0	0
UR	538	0-55	User Register Data	*****	32 bit sgn. int	decimal	-214.... - 2147483647	1	1	0
VC	548	0-1	Velocity Conversion Number (3000 RPM = XXXX.XX)	VDP	16 bit uns. int.	decimal	0 - 65535	1	1	0

VE	550	0-255	Index Velocity	VDP	16 bit uns. int.	decimal	0 - max velocity	1	1	1
VL	557	0-2	Velocity User Entry Limit	VDP	16 bit uns. int.	decimal	0 - max velocity	1	1	0
WL	583	0	Global External Input Timeout	None	32 bit uns. int.	decimal	0 - 2147483.647	1	1	0
WP	587	0-1	In-Position Window Enable(0), Time Limit(1)	DDP	16 bit uns. int.	decimal	0/1	1	1	0
WR	589	N/A	In-Position Range	DDP	16 bit uns. int.	decimal	0 - 65535	1	1	0
WT	591	N/A	In-Position Time for FE to be within WR	None	16 bit uns. int.	decimal	0 - 65535	1	1	0

Example PLC and SLC Ladder Logic

Overview

The following ladder logic provides an example for the ladder logic necessary to integrate the 3100-EMC and the 3150-EMC modules into their respective processor platforms. This logic can be incorporated directly as is, or if desired modified as needed for the application.

Data Files

The examples use the same memory map for both of the platforms, with the exception of the actual block transfer data and control files.

The memory map for the example application has been detailed in the attached data table listing. Please reference the right hand side of the data table listing for details.

Communication Configuration

	Baud Rate		RTS to TxD Delay		VDP/DDP Pairs (1 to 4)						
	0	1	2	3	4	5	6	7	8	9	
N7:0	1	0	0	0	0	0	0	0	0	0	Port #1 Configuration
N7:10	1	1	0	0	0	0	0	0	0	0	Port #2 Configuration
N7:20	0	0	0	0	0	0	0	0	0	0	System Configuration (Not currently in use)

Command Slot Configuration - Port 1

Port 2 is located at N10 with same image

	Command Control		Drive Addr		Command Op Code		Command Modifier		Data Value		
	0	1	2	3	4	5	6	7	8	9	
N9:0	1	0	0	0	0	0	0	0	0	0	Comand Slot #1
N9:10	1	1	0	0	0	0	0	0	0	0	Comand Slot #2
N9:20	0	0	0	0	0	0	0	0	0	0	Comand Slot #3
N9:30	0	0	0	0	0	0	0	0	0	0	Comand Slot #4
N9:40	0	0	0	0	0	0	0	0	0	0	Comand Slot #5
N9:50	0	0	0	0	0	0	0	0	0	0	Comand Slot #6

Command Results Block - Port 1

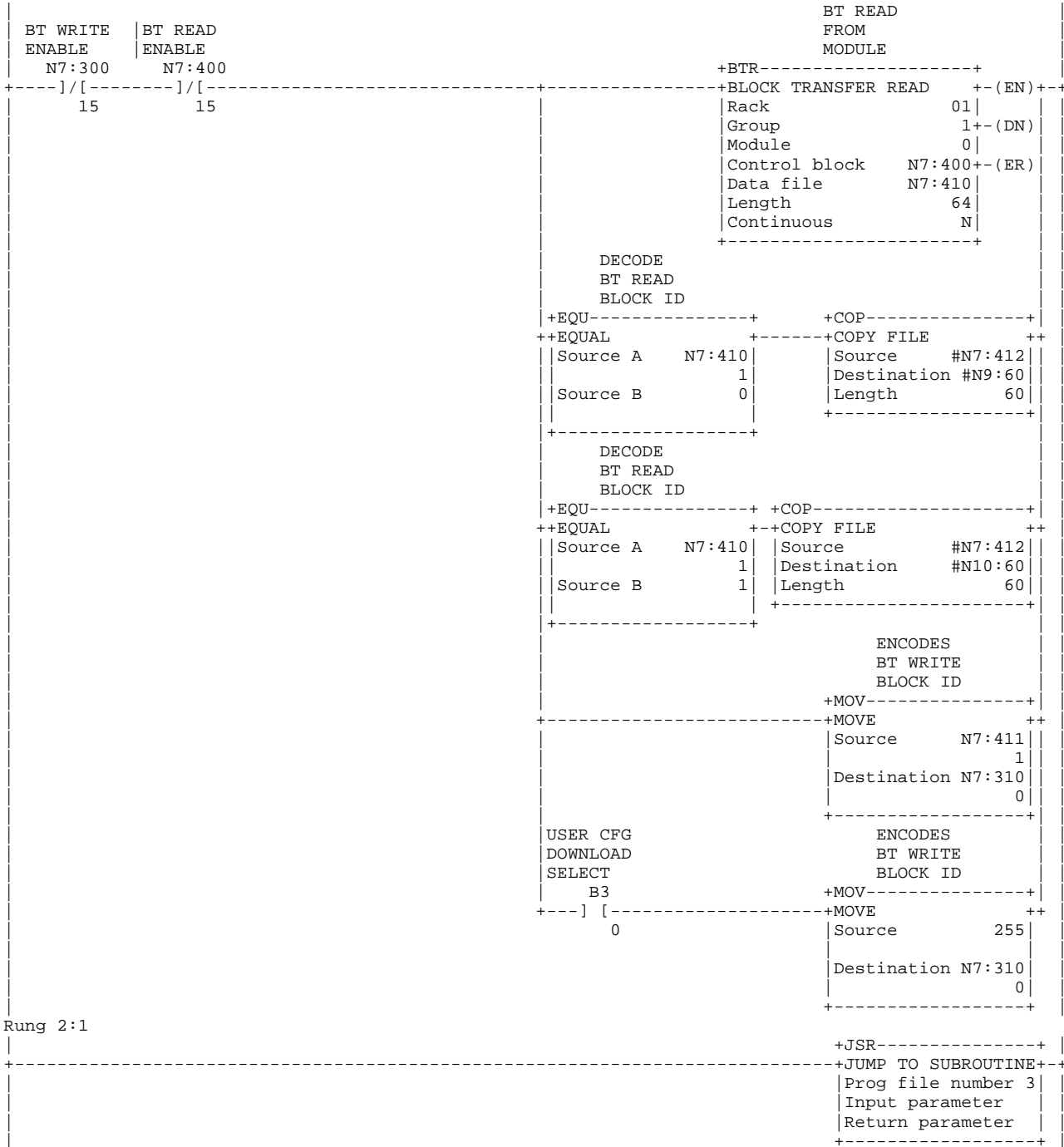
Port 2 is located in similar position in N10 with same image

Command Status Flags	Drive Addr	Executed Op Code	Executed Modifier	Queried Data Value	Communication Error Code	
N9:60	6	0	2	0	0	Response Slot #1
N9:70	0	0	0	0	0	Response Slot #2
N9:80	8	0	2	0	16382	Response Slot #3
N9:90	0	0	0	0	0	Response Slot #4
N9:100	0	0	0	0	0	Response Slot #5
N9:110	0	0	0	0	0	Response Slot #6

PLC Example Ladder Logic

Rung 2:0

BT READ AND REGISTER TRANSFER FROM MODULE DECODING
 BT READ from module. If BT READ Block ID is 1, then transfers the module's registers 50 - 99 into the PLC data table starting at N10:50. To add additional data blocks, simply add additional decoding logic.



Rung 2:1

PLC Example Ladder Logic

Rung 3:0

```

N9:60      +MOV-----+
+--] [-----+MOVE      +-+
      |7      |Source      0|
      |      |Destination N9:0|
      |      |              0|
      +-----+
|N9:60      +CTU-----+
+--] [-----+COUNT UP  +- (CU)++
      |4      |Counter      C5:0|
      |      |Preset        3+- (DN)|
      |      |Accum          0|
      +-----+
|C5:0      +MOV-----+
+--] [-----+MOVE      +-+
      |DN     |Source      0|
      |      |Destination N9:0|
      |      |              0|
      +-----+
      |C5:0|
      +-----+
      |----- (RES)+
  
```

Rung 3:1

```

N9:70      +MOV-----+
+--] [-----+MOVE      +-+
      |7      |Source      0|
      |      |Destination N9:10|
      |      |              0|
      +-----+
|N9:70      +CTU-----+
+--] [-----+COUNT UP  +- (CU)++
      |4      |Counter      C5:1|
      |      |Preset        3+- (DN)|
      |      |Accum          0|
      +-----+
|C5:1      +MOV-----+
+--] [-----+MOVE      +-+
      |DN     |Source      0|
      |      |Destination N9:10|
      |      |              0|
      +-----+
      |C5:1|
      +-----+
      |----- (RES)+

      +MOV-----+
+-----+MOVE      +-+
      |Source      0|
      |Destination N9:70|
      |              0|
      +-----+
  
```

Rung 3:2

[END OF FILE]

PLC Example Ladder Logic

File N7

Address	0	1	2	3	4	5	6	7	8	9
N7:0	6	0	1	1	1	1	1	1	1	1
N7:10	6	25	1	1	1	1	1	1	1	1
N7:20	0	0	0	0	0	0	0	0	0	0
N7:30	0	0	0	0	0	0	0	0	0	0
N7:40	0	0	0	0	0	0	0	0	0	0
N7:50	0	0	0	0	0	0	0	0	0	0

File N9

Address	0	1	2	3	4	5	6	7	8	9
N9:0	0	1	19	0	567	0	0	0	0	0
N9:10	0	2	19	0	1	0	0	0	0	0
N9:20	0	1	19	0	0	0	0	0	0	0
N9:30	0	4	19	0	0	0	0	0	0	0
N9:40	0	1	19	0	0	0	0	0	0	0
N9:50	0	2	19	0	0	0	0	0	0	0
N9:60	0	0	0	0	0	0	0	0	0	0
N9:70	0	0	0	0	0	0	0	0	0	0
N9:80	0	0	0	0	0	0	0	0	0	0
N9:90	0	0	0	0	0	0	0	0	0	0
N9:100	0	0	0	0	0	0	0	0	0	0
N9:110	0	0	0	0	0	0	0	0	0	0

File N10

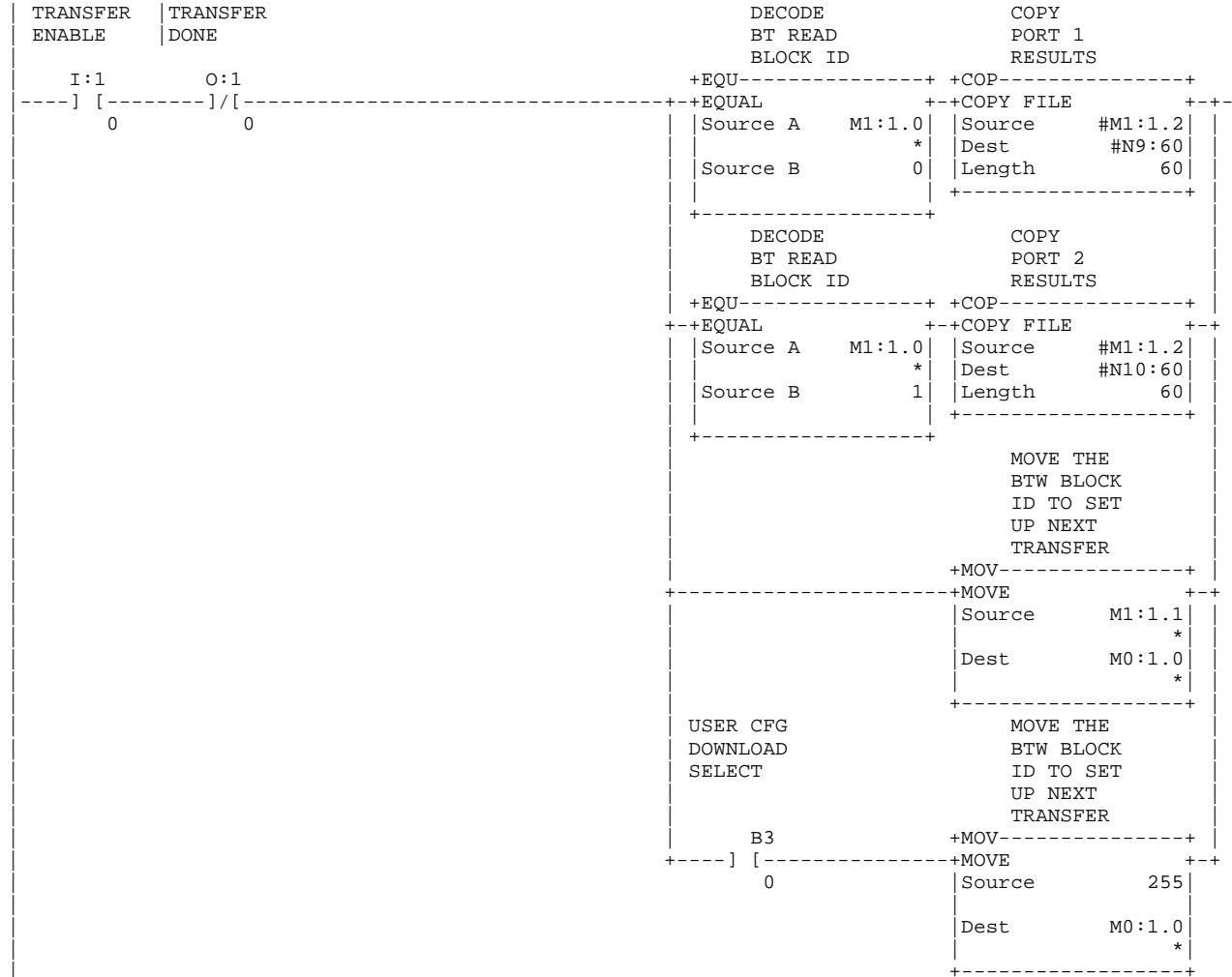
Address	0	1	2	3	4	5	6	7	8	9
N10:0	1	2	19	0	100	0	0	0	0	0
N10:10	0	2	19	0	0	0	0	0	0	0
N10:20	2	2	19	0	500	0	0	0	0	0
N10:30	0	1	460	0	0	0	0	0	0	0
N10:40	0	1	19	0	0	0	0	0	0	0
N10:50	0	1	19	0	0	0	0	0	0	0
N10:60	129	2	19	0	0	0	0	0	0	0
N10:70	0	0	0	0	0	0	0	0	0	0
N10:80	130	2	19	0	100	0	0	0	0	0
N10:90	0	0	0	0	0	0	0	0	0	0
N10:100	0	0	0	0	0	0	0	0	0	0
N10:110	0	0	0	0	0	0	0	0	0	0

SLC Example Ladder Logic

Rung 2:0

READ DATA FROM MODULE

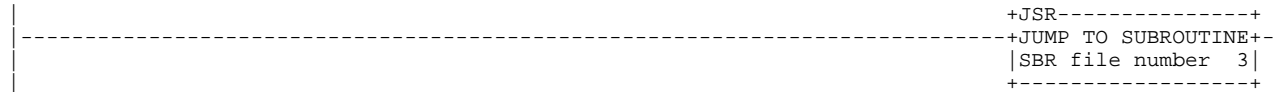
This rung transfers the results of the last Command Block for each of the ports. When the BTR Block ID number is 0, the data is for port 1 and when it is 1 the data is for port 2.



Rung 2:1

COMMAND ENABLE TOGGLE

The subroutine handles clearing the Command Enable word if the done bit is set. The User must enter logic to handle if the error bit is set or if the command does not need to be one shoted

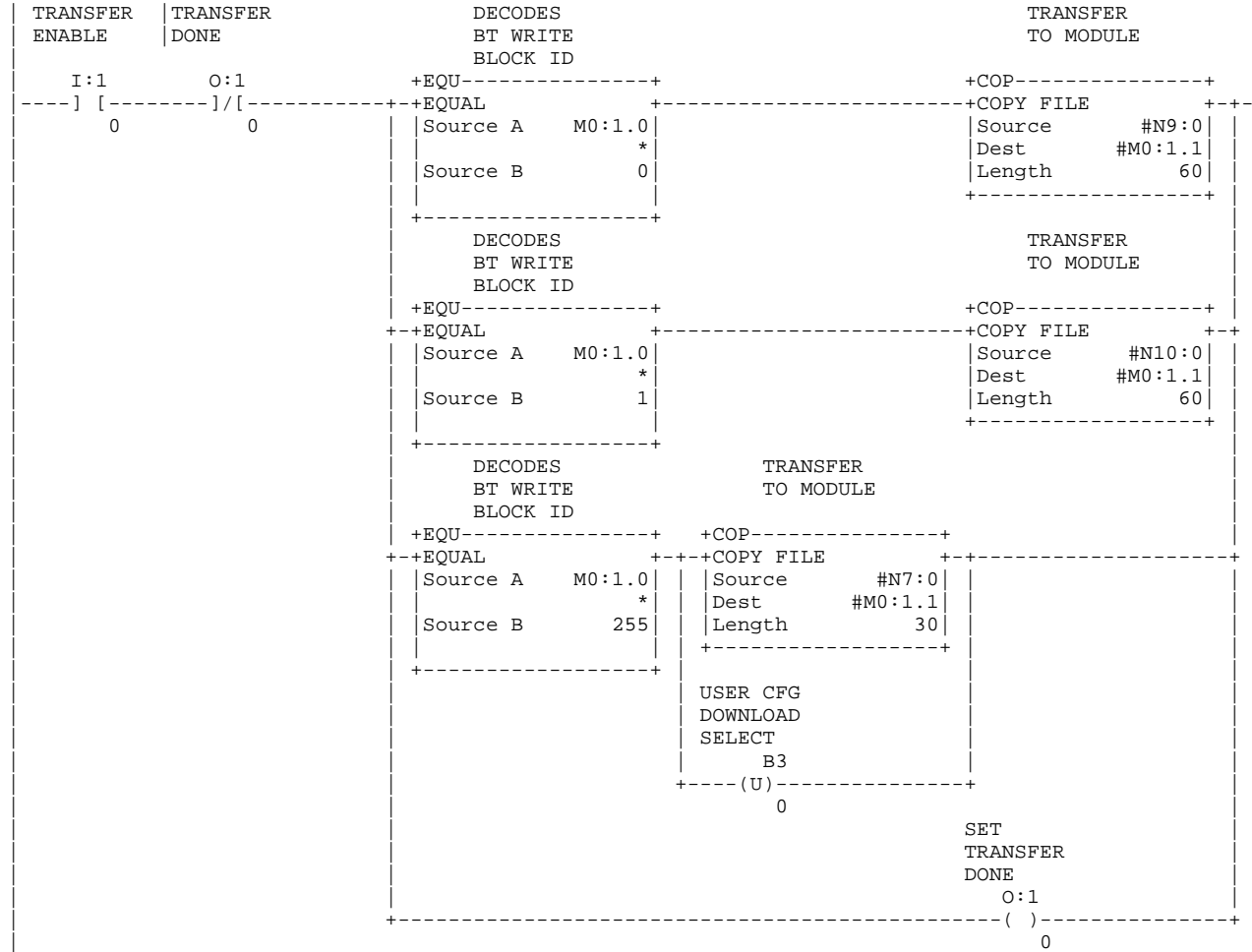


SLC Example Ladder Logic

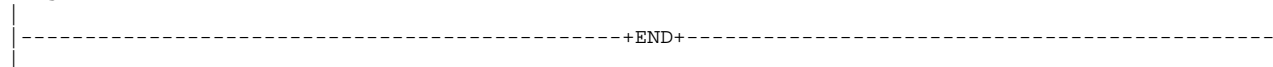
Rung 2:2

WRITES DATA OR CONFIGURATION BLOCK TO EMC

This rung moves data from the ladder logic data space to the module. When the BTW Block ID number is 0 the data for port 1 is moved to the module and when the value is 1 the data for port 2 is moved.



Rung 2:3



SLC Example Ladder Logic

|

```
+++
| N10:80 +MOV-----+
+--] [---+MOVE      +-+
|   7   |Source      0|
|       |Dest        N10:20|
|       |              0|
|       +-----+
| N10:90 +MOV-----+
+--] [---+MOVE      +-+
|   7   |Source      0|
|       |Dest        N10:30|
|       |              0|
|       +-----+
| N10:100 +MOV-----+
+--] [---+MOVE      +-+
|   7   |Source      0|
|       |Dest        N10:40|
|       |              0|
|       +-----+
| N10:110 +MOV-----+
+--] [---+MOVE      +-+
|   7   |Source      0|
|       |Dest        N10:50|
|       |              0|
|       +-----+
```

Rung 3:2

-----+END+-----

SLC Example Ladder Logic

File N7

N7:0	6	0	1	1	1	1	1	1	1	1	6	0
N7:12	1	1	1	1	1	1	1	1	0	0	0	0
N7:24	0	0	0	0	0	0						

File N9

N9:0	0	1	19	0	100	0	0	0	0	0	0	1
N9:12	19	0	0	0	0	0	0	0	0	0	0	0
N9:24	0	0	0	0	0	0	0	0	0	0	0	0
N9:36	0	0	0	0	0	0	0	0	0	0	0	0
N9:48	0	0	0	0	0	0	0	0	0	0	0	0
N9:60	0	1	19	0	567	0	1	0	0	0	0	1
N9:72	19	0	100	0	0	0	0	0	0	0	0	0
N9:84	0	0	0	0	0	0	0	0	0	0	0	0
N9:96	0	0	0	0	0	0	0	0	0	0	0	0
N9:108	0	0	0	0	0	0	0	0	0	0	0	0

File N10

N10:0	0	0	0	0	0	0	0	0	0	0	0	0
N10:12	0	0	0	0	0	0	0	0	0	0	0	0
N10:24	0	0	0	0	0	0	0	0	0	0	0	0
N10:36	0	0	0	0	0	0	0	0	0	0	0	0
N10:48	0	0	0	0	0	0	0	0	0	0	0	0
N10:60	0	0	0	0	0	0	0	0	0	0	0	0
N10:72	0	0	0	0	0	0	0	0	0	0	0	0
N10:84	0	0	0	0	0	0	0	0	0	0	0	0
N10:96	0	0	0	0	0	0	0	0	0	0	0	0
N10:108	0	0	0	0	0	0	0	0	0	0	0	0