

Design Document

EtherNet/IP Explicit Messaging Instruction *Function Block and Sample Ladder Logic Program*

Version 1.5

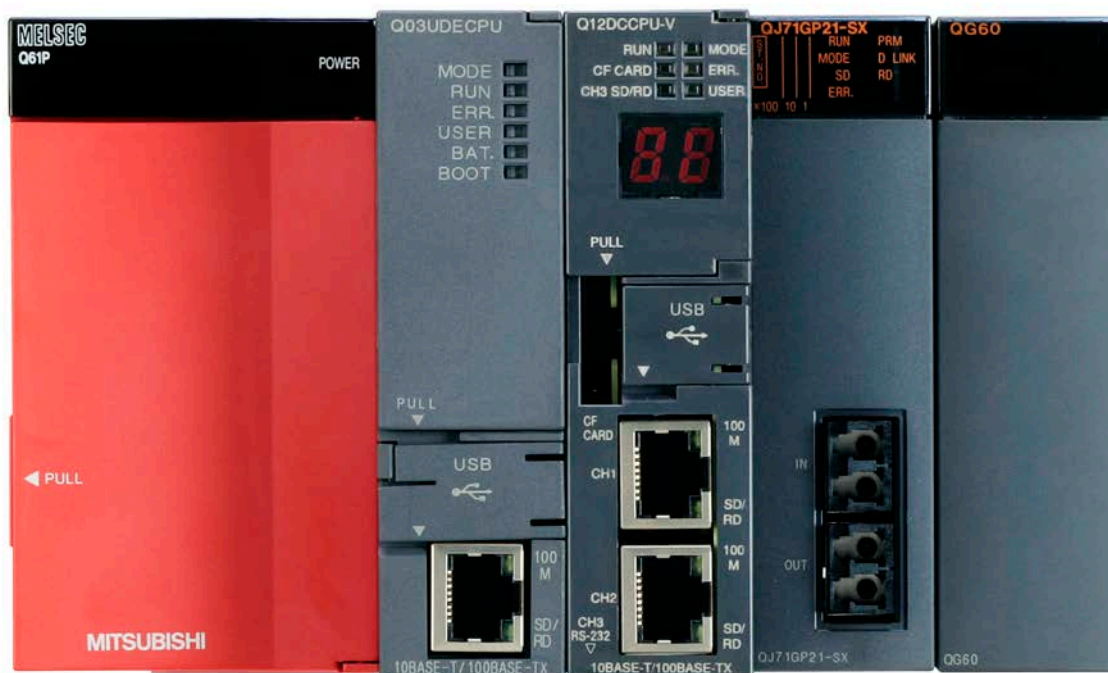


Table of Content

1	Introduction	1
2	Explicit Messaging Instruction Function Block.....	1
2.1	Description	1
2.2	ExplicitMSG FB Time Out	1
2.3	Function Block Variables	2
2.4	Command Block and the Associated Structured Data Type	6
3	Installing the ExplicitMSG FB	9
4	Examples of Using ExplicitMSG FB	10
4.1	Set Attribute Single Operation	10
4.2	Data Table Read Operation	11
4.3	Typed Write Operation.....	12
5	Sample Ladder Logic Program	13
5.1	Description	13
5.2	The Sample Ladder Program	14

1 Introduction

This document describes the design of an EtherNet/IP Explicit Messaging Instruction Function Block, named as ExplicitMSG in the implementation. ExplicitMSG is implemented as a function block using Structure Text programming language in GX Works 2.

The key function of ExplicitMSG is to send and receive EtherNet/IP Explicit Messages and Rockwell CSP messages through an EtherNet/IP scanner module that is based on the Mitsubishi Q12DCCPU-V Module, with the product name of EIP4CCPU. The EtherNet/IP Scanner firmware is developed in cooperation with **Industrial Control Communications, Inc.**

2 Explicit Messaging Instruction Function Block

2.1 Description

ExplicitMSG gathers input parameters and send/receive data location and length from a user and determines whether these input parameters are configured properly. It then packages the information in a “command block” as defined in the “C CPU EtherNet/IP Scanner Specifications.”

ExplicitMSG then interacts with the EtherNet/IP Scanner Module using interrupt through the backplane. Once the message transfer is completed, it determines the transfer results and provides the proper status and data (when used to receive messages from the network) to the designated locations in the PLC.

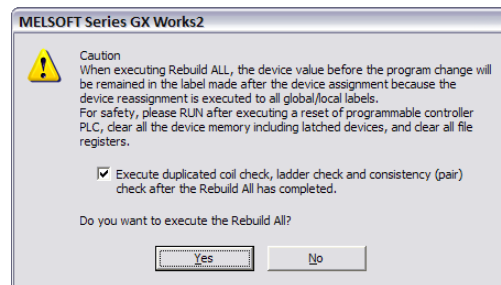
2.2 ExplicitMSG FB Time Out

A use-defined time out value is used by the MSG Instruction FB to detect the failure of calling the interrupt and also the execution of interrupt service routine of the EIP4CCPU. The timer used by the FB is determined by the ExplicitMSGChannel which defines the interrupt channel that is being used. For example, if Interrupt Channel 1 is selected for the MSG FB, Timer 1 (i.e. TC1) will be used by the MSG FB. If the timer times out before the interrupt call responded or before the EIP4CCPU completing the sending and receiving the message, then the error flag will be activated.

It is important to note that the Timer used by the MSG Instruction FB should **NOT** be used anywhere else in the program. For example, when Interrupt Channel 1 is used for MSG FB, Timer 1 (TC1) is used by the MSG FB. Thus, TC1 and its associated coil (TS1) and timer register (TN1) should NOT be used anywhere else in the program.

The minimum time out value is 5 seconds and the maximum is 100 seconds. If a user input a value less than 5 seconds or more than 100 seconds, the time out value will be set at 5 seconds and 100 seconds respectively.

Since the Time Out timer is used in two locations within the FB, one for clearing the Timer value when reset and another one for the actual timing of the messaging operation, one may get two warning messages when compiling the project where the MSG Instruction FB is called with the “Duplicated Coil Check” box selected as shown below:



No.	Result	Data Name	Class	Content	Error Code
1	CheckWarning	ExplicitMSG	Duplicated coil/ladder/consistency check	A double-coil error has occurred.	C9300
2	CheckWarning	ExplicitMSG	Duplicated coil/ladder/consistency check	A double-coil error has occurred.	C9300

The user should simply ignore the CheckWarning messages or uncheck the “Execute duplicated coil check” box while compiling the project.

2.3 Function Block Variables

The local variables that are used by the FB to interact with a user are described in this section. Three types of local variables are used:

- Input Variables – The values of these variables need to be properly set or defined by connecting proper logic or variable inputs to the FB before execution.
- Output Variables – The values of these variables will be properly set or defined after the execution of the FB is completed. A user of the FB can chose not to use the results of the output variables.
- Input_Output Variables – The values of these variables need to be properly set or defined. The values will then be modified by the logic of the FB and provided after the execution of the FB is completed.

The ExplicitMSG Function Block is shown in the figure below:

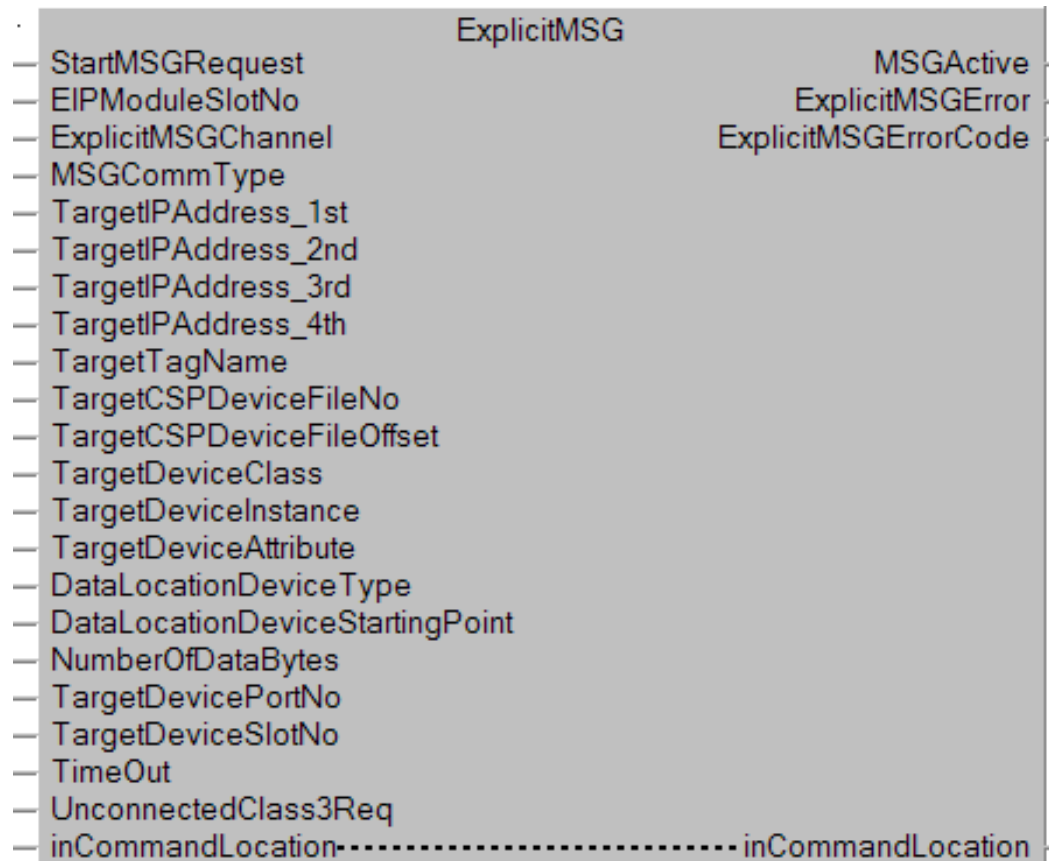


Figure 1 ExplicitMSG Function Block

Variable Type	Variable Label	Data Type	Description
VAR_INPUT	StartMSGRequest	Bit	When set, start the Explicit Messaging functions
VAR_INPUT	EIPModuleSlotNo	Word [Signed]	The slot number where the EtherNet/IP scanner is mounted in the rack. The Slot No. has to be between 2 and 4.
VAR_INPUT	ExplicitMSGChannel	Word [Signed]	The C CPU interrupt channel that is used by the ExplicitMSG. The valid value is between 0 and 15.

EtherNet/IP Explicit Messaging Instruction Function Block

Variable Type	Variable Label	Data Type	Description
VAR_INPUT	MSGCommType	Word [Unsigned]	<p>The Explicit Messaging type for this message transfer. Valid values are:</p> <ul style="list-style-type: none"> 0x0E: Get Attribute Single Read a single attribute value. Must specify the Class, Instance, and Attribute fields. 0x10: Set Attribute Single Write a single attribute value. Must specify the Class, Instance, and Attribute fields. 0x4C: Data Table Read Must specify the Tag name. Tag/File offset is optional and will be ignored if left blank or 0. 0x4D: Data Table Write Must specify the Tag name. Tag/File offset is optional and will be ignored if left blank or 0. 0x68: Typed Read Must specify the File number and Tag/File offset. It is highly recommended to specify a Port value and a Slot value. A Port value of 1 and Slot value of 0 is recommended if these fields are unknown. 0x67: Typed Write Must specify the File number and Tag/File offset. It is highly recommended to specify a Port value and a Slot value. A Port value of 1 and Slot value of 0 is recommended if these fields are unknown.
VAR_INPUT	TargetIPAddress_1st	Word [Unsigned]	First byte of the IP address of the target EtherNet/IP server device
VAR_INPUT	TargetIPAddress_2nd	Word [Unsigned]	Second byte of the IP address of the target EtherNet/IP server device
VAR_INPUT	TargetIPAddress_3rd	Word [Unsigned]	Third byte of the IP address of the target EtherNet/IP server device
VAR_INPUT	TargetIPAddress_4th	Word [Unsigned]	Fourth byte of the IP address of the target EtherNet/IP server device
VAR_INPUT	TargetTagName	String(32)	The target tag name on an EIP server device. Use of the tag name is mutually-exclusive of use of the class, instance and attribute members.
VAR_INPUT	TargetCSPDeviceFileNo	Word [Unsigned]	A PLC5 file number. Valid values are 0 to 255.
VAR_INPUT	TargetCSPDeviceFileOffset	Word [Unsigned]	An offset for the Tag name or File number. Valid values are any 16-bit value.

EtherNet/IP Explicit Messaging Instruction Function Block

Variable Type	Variable Label	Data Type	Description
VAR_INPUT	TargetDeviceClass	Word [Unsigned]	The targeted EIP server device class.
VAR_INPUT	TargetDeviceInstance	Word [Unsigned]	The targeted EIP server device instance.
VAR_INPUT	TargetDeviceAttribute	Word [Unsigned]	The targeted EIP server device attribute.
VAR_INPUT	DataLocationDeviceType	String(2)	For set/write requests, this field specifies the source device type in the Q PLC. For get/read requests, this field is the destination device type in the Q PLC. Used in conjunction with the DataLocationDeviceStartingPoint below. Valid Device Types are: D, W, and ZR
VAR_INPUT	DataLocationDeviceStartingPoint	Double Word [Unsigned]	For set/write requests, this parameter is the starting device point of the source in the PLC. For get/read requests, this field is the starting device point of the destination in the PLC. Used in conjunction with the DataLocationDeviceType above.
VAR_INPUT	NumberOfDataBytes	Word [Unsigned]	During “set/write” operations, it indicates the number of bytes to be written to the EIP server device. During “get/read” operations, this value indicates the number of bytes that are received from the EIP server device and stored in the device type & point location. Valid values are 1 to 240 bytes.
VAR_INPUT	TargetDevicePortNo	Word [Unsigned]	This field specifies the port to be used by the EIP <u>server</u> adaptor module to route the packet to the “target” device on its rack. The port values are specific to the adapter device on a rack-based chassis. Typically, a port value of 1 will correspond to the chassis backplane. Valid values are 0 to 14. A value of 0 must be set if not used.
VAR_INPUT	TargetDeviceSlotNo	Word [Unsigned]	This field should only be used when addressing a modular EIP server device on a rack-based chassis. The slot value will indicate the location of the target device in the rack. A Port value other than 0 must be specified if the Slot value is used. Otherwise, a Port value of 0 must be set if the Slot value is not used. Valid values are 0 to 255.

EtherNet/IP Explicit Messaging Instruction Function Block

Variable Type	Variable Label	Data Type	Description
VAR_INPUT	TimeOut	Word [Signed]	This value defines the timeout of the instruction in Seconds. The user can define how long the FB should wait for a response from the EIP4CCPU. The minimum value is 5 seconds and the maximum is 100 seconds. If a user input a value < 5, this value will be set to 5. If a user input a value >100 this value will be set at 100 seconds.
VAR_INPUT	UnconnectedClass3Req	Bit	0: Connected Class 3 Messaging is used 1: Unconnected Class 3 Messaging is used. Message routing is not supported.
VAR_IN_OUT	inCommandLocation	MSG_SDT	The structured data of the command block that is used by the ExplicitMSG.
VAR_OUTPUT	MSGActive	Bit	When this bit is on, it indicates the ExplicitMSG FB is active in sending or receiving messages
VAR_OUTPUT	ExplicitMSGError	Bit	When this bit is set, it indicates there is an error in one of the input parameters to the Function Block or the operation of the function block returns an error.

Variable Type	Variable Label	Data Type	Description
VAR_OUTPUT	ExplicitMSGErrorCode	Word [Unsigned]	<p>When ExplicitMSGError is set, the error code is output here:</p> <ol style="list-style-type: none"> 1: Request Explicit Messaging operation (MSGCommType) is not valid 2: Invalid number of data bytes (NumberOfDataBytes) is specified. 3: Invalid Explicit Messaging Channel (ExplicitMSGChannel) is specified. 4: Invalid EIP module slot number (EIPModuleSlotNo) is specified. 5: Invalid source/destination device type (DataLocationDeviceType) is specified. 6: The EIP explicit messaging operation resulted in an error. The Response General Status word and the Response Extended Status word of the command block (<i>the location of the command block is specified by the user in the inCommandLocation</i>) should be checked for the actual error code returned. 7: The CSP operation resulted in an error. The CSP Response General Status word and the CSP Response Extended Status word of the command block (<i>the location of the command block is specified by the user in the inCommandLocation</i>) should be checked for the actual error code returned. 8: There is an EIP Module interrupt error thus the messaging request is not completed. 9: Illegal Target Device Port and/or Slot numbers 10: The interrupt operation time out occurs. The EIP4CCPU did not complete the explicit messaging operation before the time out occurs.

2.4 Command Block and the Associated Structured Data Type

The command block is used for each ExplicitMSG FB instance to communicate with the EIP Scanner module. The size of each command block is 64 words. *The responsibility of a user of the ExplicitMSG FB is to specify where the command block should be located in the PLC memory and pass the label or device location to the ExplicitMSG FB through the inCommandLocation parameter.*

A user does not have to be concerned about the operation of the command block and populating the command block content. The ExplicitMSG FB will take the user inputs and configure the command block properly. However, *a user may have to read the Response General Status, Response Extended Status, CSP Response General Status, and CSP Response Extended Status* when there is an error as described in the section above.

The structured data type MSG_SDT is used to define a command block for a specific ExplicitMSG instance. Most the elements in the data structure have been explained in the section above and the definition of each element of the structured data type is documented below for completeness.

EtherNet/IP Explicit Messaging Instruction Function Block

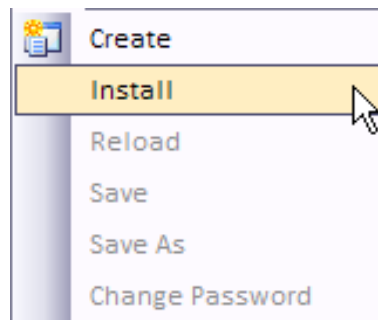
Variable Label	Data Type	Description
Version	Word [Unsigned]	The version of the command block structure, the initial version is "0"
Active	Word [Unsigned]	<ul style="list-style-type: none"> While zero, any of the other members of the structure (class, instance, etc.) may be freely changed by the PLC (the EM object remains dormant). When non-zero, the values contained in the other structure members will be used to generate the EM client request to the EIP server device. When the C CPU has fully processed the EIP server's response and populated the other members of the structure with the response data, the "active" member will be set to 0 to inform the PLC that the EM response is ready.
ResponseGeneralStatus	Word [Unsigned]	As defined by the EIP specification. 0=service completed successfully, nonzero=error.
ResponseExtendedStatus	Word [Unsigned]	As defined by the EIP specification (can be ignored if response general status is 0).
CSPGeneralStatus	Word [Unsigned]	As defined by the CSP specification. 0=service completed successfully, nonzero=error, 0xF0=see CSP response extended status.
CSPExtendedStatus	Word [Unsigned]	As defined by the CSP specification (can be ignored if CSP response general status is not 0xF0).
ServiceCode	Word [Unsigned]	Messaging operation types (see MSGCommType in the FB variable section above).
IPAddress	Double Word [Unsigned]	Target server device IP address in IPV4 format.
Port	Word [Unsigned]	<p>This field specifies the port to be used by the EIP <u>server</u> adaptor module to route the packet to the "target" device on its rack. The port values are specific to the adapter device on a rack-based chassis. Typically, a port value of 1 will correspond to the chassis backplane.</p> <p>Valid values are 0 to 14. A value of 0 must be set if not used.</p>
Slot	Word [Unsigned]	<p>This field should only be used when addressing a modular EIP server device on a rack-based chassis. The slot value will indicate the location of the target device in the rack.</p> <p>A Port value other than 0 must be specified if the Slot value is used. Otherwise, a Port value of 0 must be set if the Slot value is not used.</p> <p>Valid values are 0 to 255.</p>
TagName	String(32)	The target tag name on an EIP server device. Use of the tag name is mutually-exclusive of use of the class, instance and attribute members.

EtherNet/IP Explicit Messaging Instruction Function Block

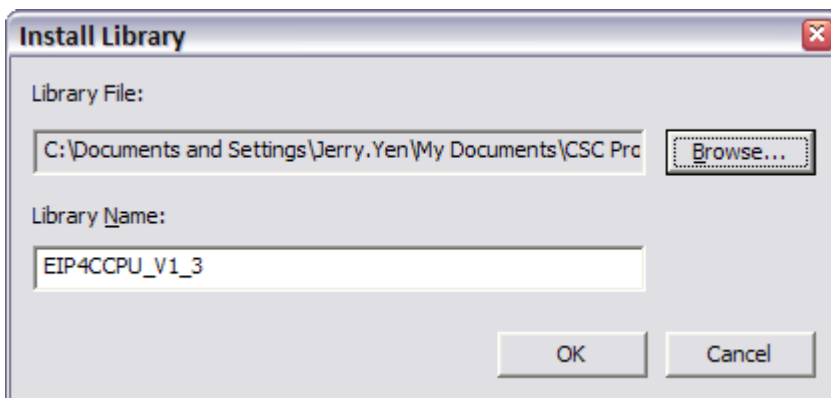
Variable Label	Data Type	Description
FileNo	Word [Unsigned]	A PLC5 file number. Valid values are 0 to 255.
TagFileOffset	Word [Unsigned]	An offset for the Tag name or File number. Valid values are 0 to 255.
Class	Word [Unsigned]	The targeted EIP server device class.
Instance	Word [Unsigned]	The targeted EIP server device instance.
Attribute	Word [Unsigned]	The targeted EIP server device attribute.
DeviceType	Word [Unsigned]	For set/write requests, this field specifies the source device type in the Q PLC. For get/read requests, this field is the destination device type in the Q PLC. Used in conjunction with the DataLocationDeviceStartingPoint below. Valid Device Types are: D, W, and ZR
DevicePoint	Double Word [Unsigned]	For set/write requests, this parameter is the starting device point of the source in the PLC. For get/read requests, this field is the starting device point of the destination in the PLC. Used in conjunction with the DataLocationDeviceType above.
PayloadSize	Word [Unsigned]	During “set/write” operations, it indicates the number of bytes to be written to the EIP server device. During “get/read” operations, this value indicates the number of bytes that are received from the EIP server device and stored in the device type & point location. Valid values are 1 to 240 bytes.
Unconnected	Word [Unsigned]	0: Connected Class 3 Messaging is used 1: Unconnected Class 3 Messaging is used. Message routing is not supported.
Reserved	Word [Unsigned] (0..26)	Not Used.

3 Installing the ExplicitMSG FB

- Open the User Library Window
- Click the “New Library” icon and select “Install” as shown below

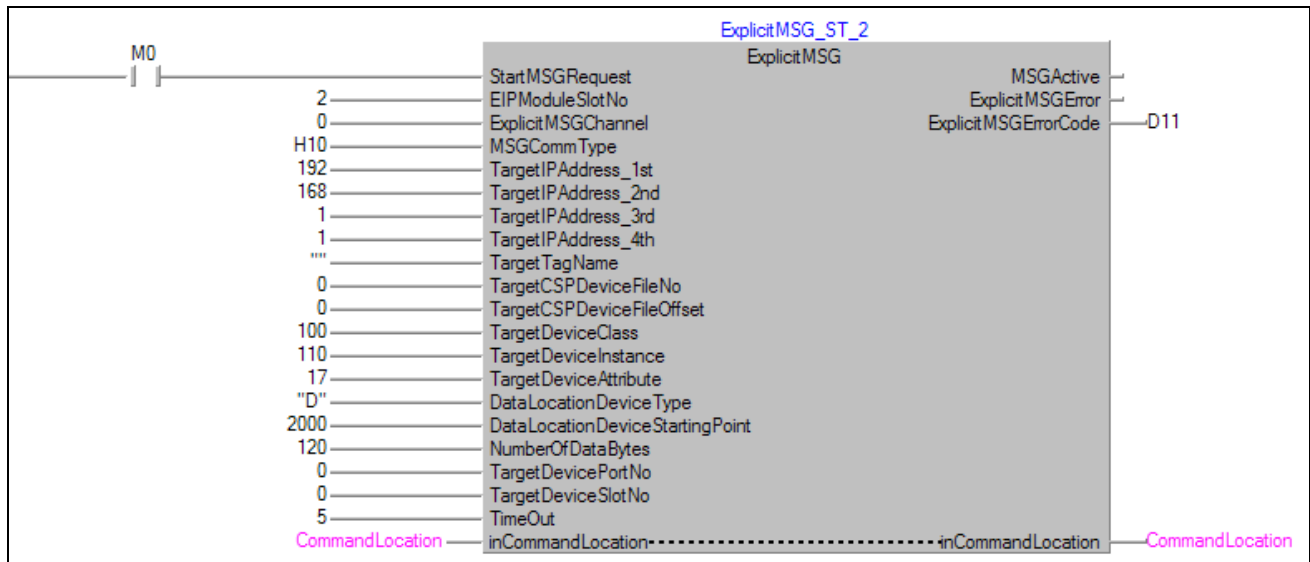


- When the “Install Library” pop-up window appears, “Browse” to the directory where the library is stored and select the latest version of the EIP4CCPU_Vx_x. Below is an example showing the installation of library EIP4CCPU_V1_3



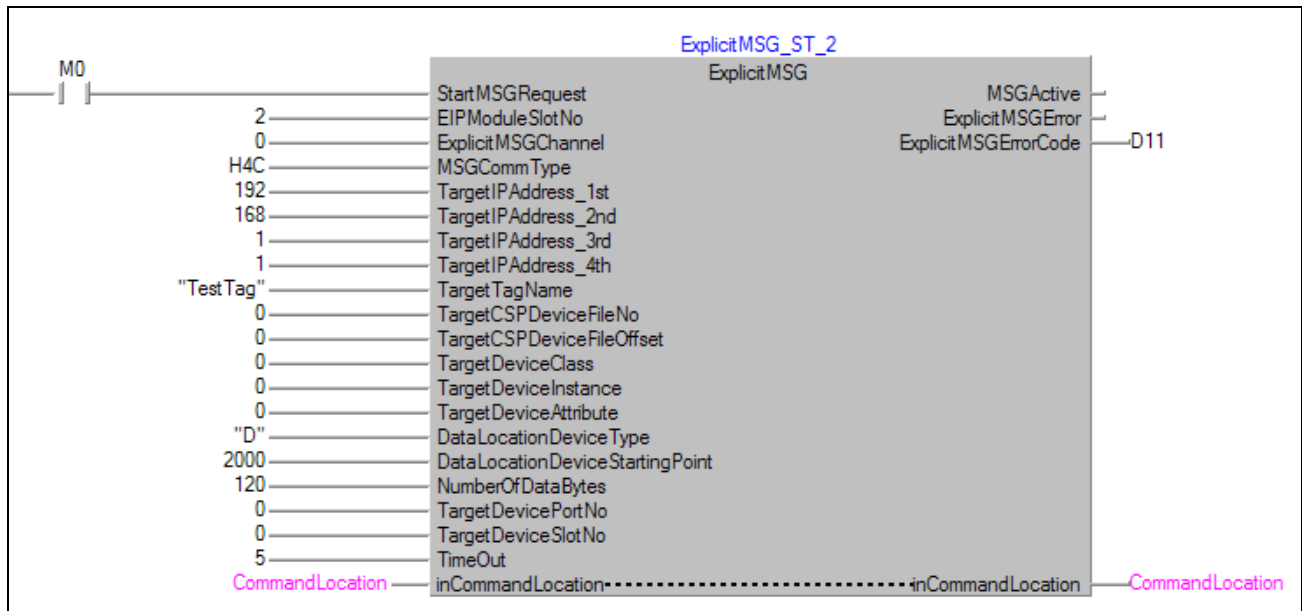
4 Examples of Using ExplicitMSG FB

4.1 Set Attribute Single Operation



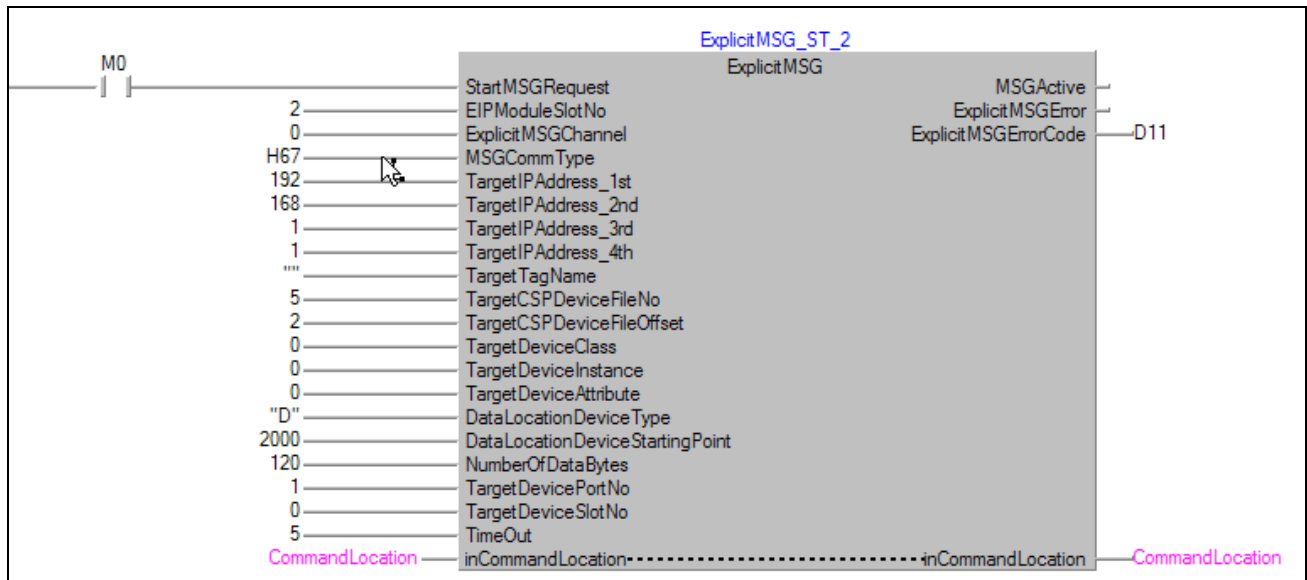
- The Command Block of this message instruction is at the label location "CommandLocation."
- The FB initiates a "Set Attribute Single" message that will be sent to a server device at IP address 192.168.1.1.
- The EIP Scanner module is installed in Slot #2
- The interrupt channel that will be used is 0
- The Target Device Class is 100
- The Target Device Instance is 110
- The Target Device Attribute is 17
- The source of the data is located in D2000 and the length of data is 120 bytes.
- The Time Out is set at 5 seconds.
- When M0 transitions from 0 to 1, the message instruction will be initiated.

4.2 Data Table Read Operation



- The Command Block of this message instruction is at the label location "CommandLocation."
- The FB initiates a "Data Table Read" operation that from a server device at IP address 192.168.1.1.
- The EIP Scanner module is installed in Slot #2
- The interrupt channel that will be used is 0
- The Tag in the server device to be read is "TestTag"
- The destination of the data is located in D2000 and the length of data is 120 bytes.
- The Time Out is set at 5 seconds.
- When M0 transitions from 0 to 1, the message instruction will be initiated.

4.3 Typed Write Operation



- The Command Block of this message instruction is at the label location "CommandLocation."
- The FB initiates a "Typed Write" operation that to a server device using CSP protocol at IP address 192.168.1.1.
- The Scanner module is installed in Slot #2
- The interrupt channel that will be used is 0
- The file in the server device to be written is 5 with offset of 2
- The destination of the data is located in D2000 and the length of data is 120 bytes.
- The Time Out is set at 5 seconds.
- When M0 transitions from 0 to 1, the message instruction will be initiated.

5 Sample Ladder Logic Program

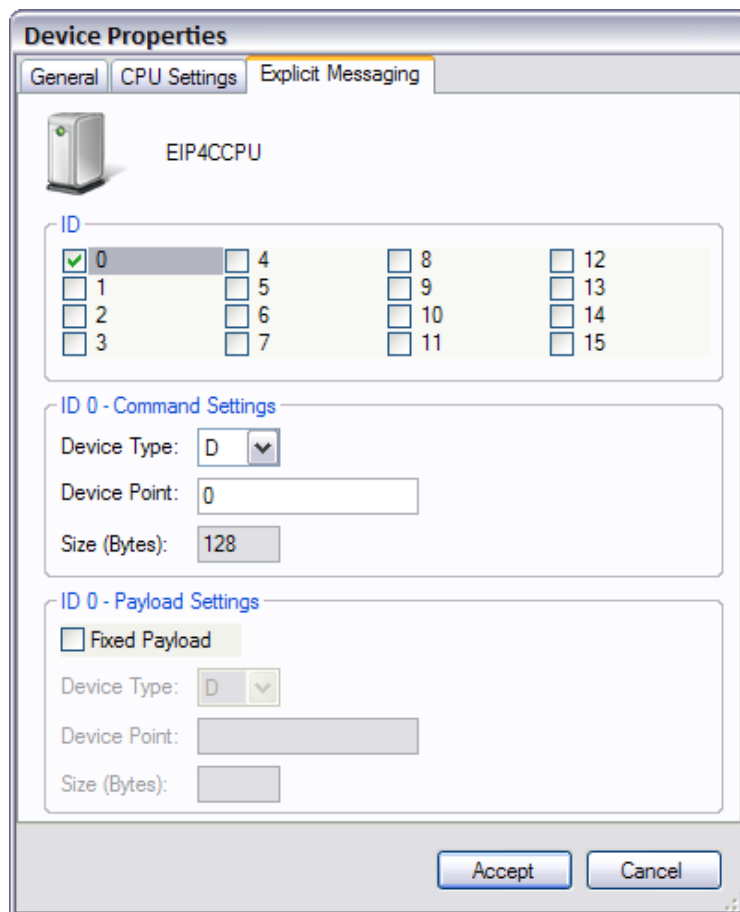
The design of a sample ladder program that performs the EtherNet/IP Explicit Messaging function using GX Developer is described in this section.

The key functions of this program are:

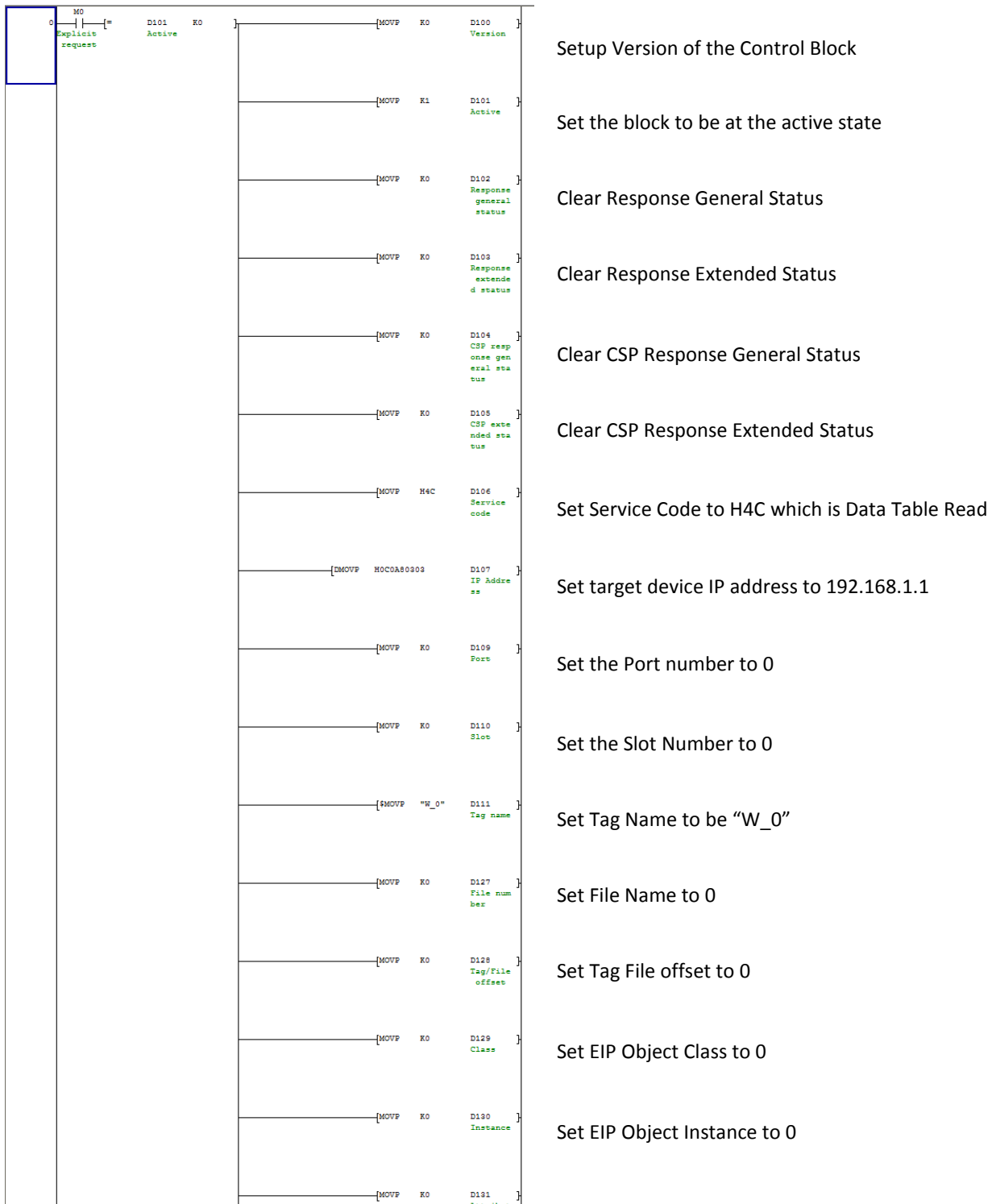
- Issuing “Data Table Read” command to a target device with IP address 192.168.1.1;
- Getting 5 words (10bytes) of data from W0 of another MELSEC Q PLC through the EtherNet/IP scanner EIP4CCPU;
- Storing the received data in the area starting with register D500.
- Turning on M31 when the Explicit Messaging function was executed successfully;
- Turning on M22 or M32 when the Explicit Messaging function execution failed to execute properly.

5.1 Description

The following sample ladder program sets up the 64-word Command Data area starting from register D0 when “M0” transitions from 0 to 1. When configuring the EIP4CCPU scanner, the Interrupt channel 0 is used for the Explicit Messaging, thus the address D0 should be configured as shown in the figure below.



5.2 The Sample Ladder Program



EtherNet/IP Explicit Messaging Instruction Function Block

