|  |
| --- |
| AutomationDirect.com |
| GS30A-CM-ENETIP |
| LS Electric Function Block Walk-Through |

## Revision: 1.0

## 03-19-2024

Contents

[Introduction 2](#_Toc161729259)

[Register the GS30 EDS File 3](#_Toc161729260)

[Setup EtherNet/IP Communication to GS30 4](#_Toc161729261)

[Import User Function Block and User Data Type 7](#_Toc161729262)

[Use Function Block in LD 8](#_Toc161729263)

[Use Function Block in ST 10](#_Toc161729264)

[GS30\_EIP\_StatusAndControl\_UDT Explanation 13](#_Toc161729265)

# Introduction

The Function Block for the GS30 removes the cumbersome task of having to map the EtherNet/IP IO Messaging data from generic arrays into specific and meaningful variables. This helps YOU get integrated faster.

This walk-through will cover:

* Registering the GS30 EDS File
* Setting up communication to GS30 in XG5000
* Importing the GS30 Function Block
* Using the GS30 Function Block in your project.

**Files Included in Folder:**

|  |  |
| --- | --- |
| File Name | Description |
| GS30\_EIP\_FB.xgwx | Sample program that shows use of the function block. |
| GS30\_Drive\_EIP\_FB.fun | Export file of the function block. |
| GS30\_EIP\_StatusAndControl\_UDT.udt | Export File of the User Data Type used by the function block. |
| GS30A\_CM\_EIP1\_Rev1\_0.eds | EDS file for GS30 with 1 Ethernet port |
| GS30A\_CM\_EIP2\_Rev1\_0.eds | EDS file for GS30 with 2 Ethernet ports |

A screen shot of a computer

Description automatically generated



# Graphical user interface, application Description automatically generatedRegister the GS30 EDS File

Step 1: Open XG5000. Register EDS file from the Tools Menu. Register the GS30 EDS file.

A screenshot of a computer

Description automatically generatedStep 2: Find the EDS file. Press Open.

A screenshot of a computer

Description automatically generatedStep 3: Open the EDS information window from the View menu and make sure it was imported successfully.

# Setup EtherNet/IP Communication to GS30

Graphical user interface

Description automatically generated with medium confidenceStep 1: In the XG5000 project, open the I/O parameter screen. Add your XBL-EIPT to the slot it is in on the PLC rack. Press Apply.

Graphical user interface, application

Description automatically generatedStep 2: Right click on the XBL-EIPT card under the Network Configuration portion of the Project screen. Select Add Item -> P2P Communication.

Graphical user interface, application, Teams

Description automatically generated

Step 3: Select P2P 03 if it isn’t being used. Select P2P 04 if 03 is not available.

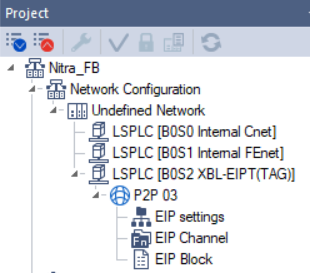
Graphical user interface, application

Description automatically generated

Step 4: Configure the IP address for the XBL-EIPT. Make sure it is on the same network as the GS30. Make sure to check Use Tag to allow variables to be used in Ethernet/IP setup.

A picture containing text

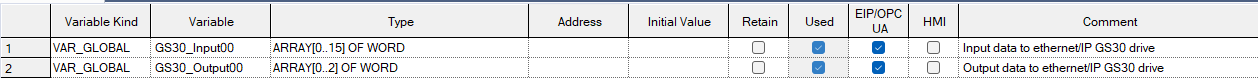
Description automatically generatedStep 5: Open EIP settings under P2P 03. Drag the GS30A EtherNet/IP Card Ver1.1 file from EDS information screen.



Graphical user interface, application

Description automatically generatedStep 6: Setup Channel number and IP address for the GS30. Press OK when done.

Step 7: Open Global/Direct Variables.

 Create the variables shown in the Image. Make sure to Check EIP/OPC UA.

A picture containing text, clock, screenshot

Description automatically generatedStep 8: Open EIP Block under P2P 03. Enter the GS30 Channel number in the first Column. It is 0 from our example.

Table

Description automatically generatedStep 9: Choose ‘0.Drive Control and Status’ as the I/O type. Choose Point to Point as Connection Type.

Table

Description automatically generatedText

Description automatically generated with medium confidenceStep 10: Click the Parameter button. Click OK button in the Parameter Settings window. Choose RPI x4 in Timeout.

Table

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated Step 11: Select GS30\_Input00 in the upper Local tag. Select GS30\_Output00 in the lower Local tag. Double Click the cell in the column Local tag to open the Select Ethernet/IP Tag Window.

# Import User Function Block and User Data Type

Graphical user interface, application, table

Description automatically generatedStep 1: From the Project Screen, Right click on User Data Type. Select Import From File -> Data Type…

A screenshot of a computer

Description automatically generated

Step 2: Select the directory where the *GS30\_EIP\_StatusAndControl\_UDT.udt* file is located. Press Open to import it.

Graphical user interface, application, table

Description automatically generatedStep 3: From the Project Screen, Right click on User Function/Function Block. Select Import From File -> Function/FB…

A screenshot of a computer

Description automatically generated

Step 4: Select the directory where the *GS30\_Drive\_EIP\_FB.fun* file is located. Press Open to import it.

# Use Function Block in LD

A screenshot of a computer

Description automatically generated

Step 1: From the Scan Program, add the GS30\_Drive\_EIP\_FB from the Function/Function Block Selection screen. Press F10 in LD or ST program to get to this screen. The user function block is located in the category <’PLC name’>. The PLC name is ‘LSPLC’ in this example.

A white grid with black lines

Description automatically generated Step 2: 5 fields within the Function Block are required to be completed:

* Instance name for the function block. This example has ‘GS30\_INST00’ as the instance name.
* The Enable for the Function Block.
* The source of InputData will be the global variable connected to EIP Block settings.
* The source OutputData will be the global variable connected to EIP Block settings. It is used on both sides of the FB.
* The source of GS30\_User\_UDT created by the user in the next step.

It is used on both sides of the FB.

Step 3: Create a UDT variable for the FB. It can be Global or Local. In this example we’ve created a local variable called ‘GS30\_User\_UDT.

Step 4: Enter the variables in the required fields. In this example, we’ve used the variable names shown in the image. You might use a different naming convention. Put appropriate variables in their designated locations.

A screen shot of a computer

Description automatically generated

Step 5: Press Check Program to Check for error. Write to XEM CPU if no errors are present.

# Use Function Block in ST

A screenshot of a computer

Description automatically generatedStep 1: From the Scan Program, add the GS30\_Drive\_EIP\_FB from the Function/Function Block Selection screen. Press F10 in LD or ST program to get to this screen. The user function block is located in the category <’PLC name’>. The PLC name is ‘LSPLC’ in this example.

Step 2: fields within the Function Block are required to be completed:

* Instance name for the function block. This example has ‘INST\_GS30\_00’ as the instance name.
* The Enable for the Function Block.
* The source of InputData will be the global variable connected to EIP Block settings.
* The source of OutputData will be the global variable connected to EIP Block settings.
* The GS30\_User\_UDT created by the user in the next step.
* The Active output is optional. It can be accesses at <instance\_name>.Active.

Or added to the function block call in the example picture.

Step 3: Create a UDT variable for the FB. It can be Global or Local. In this example we’ve created a local variable called ‘GS30\_User\_UDT’.

Step 4: Enter the variables in the required fields. In this example, we’ve used the variable names shown in the image. You might use a different naming convention. Put appropriate variables in their designated locations.



Step 5: Press Check Program to Check for error. Write to XEM CPU if no errors are present.

# GS30\_EIP\_StatusAndControl\_UDT Explanation

|  |  |  |
| --- | --- | --- |
| Variable Name | Data Type | Description |
| Direction\_Command\_REV | BOOL | Forward = FALSE or 0. Reverse = TRUE or 1 |
| Enable\_bit\_function\_6\_11 | BOOL | Enables the Nth Accel/Decel and Nth Step Speed Frequency function |
| External\_Fault\_Trigger | BOOL | Triggers an External Fault |
| External\_Interruption | BOOL | Does an external interruption |
| Reset\_Command | BOOL | Sends Reset Command to drive |
| Run | BOOL | Puts drive in RUN mode |
| Stop | BOOL | Puts drive in STOP mode. Takes precedence over Run or Enable\_JOG commands. |
| Enable\_JOG | BOOL | Puts drive in JOG mode. |
| Frequency\_Command | REAL | Frequency for drive to run at. 60.00 value will be 60 Hz. |
| Nth\_Accel\_Decel | SINT | Value for Nth Accel/Decel functionality |
| Nth\_Step\_Speed\_Frequency | SINT | Value for Nth Step Speed Frequency functionality |
| Status\_Actual\_Motor\_Speed | INT | Actual motor speed (RPM) . XXXX RPM. |
| Status\_Command\_from\_Comm\_Keypad | BOOL | Status that commands from keypad are enabled. |
| Status\_DC\_Bus\_Voltage | REAL | DC Bus Voltage. 6000 = 600.0 Vdc |
| Status\_Decel\_During\_Stop | BOOL | Status of deceleration during a stop |
| Status\_Digital\_Input\_Counter\_Value | DINT | Digital input counter from drive |
| Status\_Drive\_Running | BOOL | Status that drive is in running. |
| Status\_Drive\_Stopped | BOOL | Status that drive is stopped. |
| Status\_Freq\_From\_Analog\_Discrete | BOOL | Command frequency from analog input. |
| Status\_Freq\_from\_Comm | BOOL | Command frequency from communication |
| Status\_FreqCommand\_PIDSetpoint | REAL | Command frequency from PID setpoint |
| Status\_FWD | BOOL | Status of direction. 1 = Forward; 0 = Reverse; |
| Status\_JOG\_Active | BOOL | JOG mode is active. |
| Status\_MultiSpeed\_PIDStepNumber | INT | PID step number status. |
| Status\_Output\_Current | REAL | Output Current. XXXX = XX.XX A. |
| Status\_Output\_Frequency | REAL | Output Frequency. XXXX = XX.XX Hz |
| Status\_Output\_Torque | REAL | Output Torque . XXXX = XXX.X % |
| Status\_Output\_Voltage | REAL | Output Voltage. XXXX = XXX.X VAC |
| Status\_Parameters\_Locked | BOOL | Parameters are locked on drive |
| Status\_Power\_Factor\_Angle | REAL | Power Factor Angle . XXXX = XXX.X degrees |
| Status\_REV | BOOL | Status of reverse movement |
| Status\_Power\_Output | REAL | power output. XXXX = X.xxx kW |
| Status\_Run | BOOL | Drive is in RUN mode |
| Status\_Standby | BOOL | Drive is in Standby |
| Status\_Stop | BOOL | Drive is in STOP mode |
| Status\_Transition\_FWD\_to\_REV | BOOL | Drive is transitioning from forward to reverse |
| Status\_Transition\_REV\_to\_FWD | BOOL | Drive is transitioning from reverse to forward. |
| Status\_Warning\_Fault\_Code | INT | Drive fault code |