

# CTL50 Pro Process Data Function

---

September 24<sup>th</sup>, 2026

This document covers the installation and use of a function for Siemens's TIA Portal software package. This function handles cyclic IO-Link Process Data Out to a Banner CTL50 Pro light via an IO-Link Master from Siemens PLC. The function covers parsing and display of the CTL50 Pro sensor Process Data Out.

## **Components**

Banner CTL50 Library v16.zal16

There are two methods for the process data. The first is used when creating a connection to Banner's IO-Link masters. The second set of instructions are for systems using other manufacturer's IO-Link masters.

### **Installation Instructions**

1. Open a project.
2. Go to the Open Global Library option in the Libraries tab in TIA Portal v16 or greater.



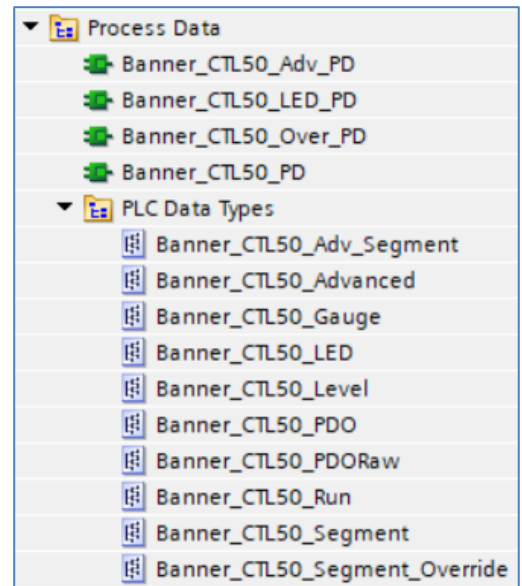
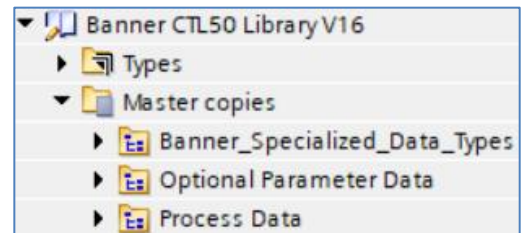
3. Switch the “Files of type” to Compressed libraries. Go to the location of the compressed library.
4. Press the Open button and the library will be uncompressed and opened.
5. The library is now accessible in the Libraries tab in v16 or greater.

**Setup of CTL50 with a Banner DXMR**

1. Go to Device and Networks to configure the DXMR. Add the DXMR if it has yet to be added to the system.
2. Add Banner IO-Link Master Info to Slot 1. This sets the DXMR for IO-Link mode.
3. Open the IO-Link Generic Devices and select the proper module. The 32/32 byte is required for CTL50. Make note of the Q address for Slot 2 which represents Port 1. Slot 2 starts at Q1 for outputs. The data for the port starts at that point (Q3). The previous two bytes are Port Control.

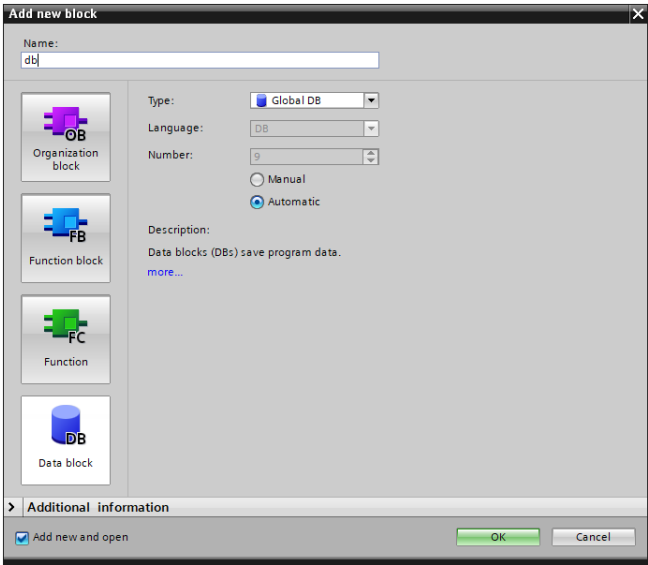
|                                      |   |   |         |        |                                    |
|--------------------------------------|---|---|---------|--------|------------------------------------|
| IO-Link In/Out 32/32 Byte + Status_1 | 0 | 2 | 10...45 | 1...46 | IO-Link In/Out 32/32 Byte + Status |
|--------------------------------------|---|---|---------|--------|------------------------------------|

4. Drag the necessary tag from Banner\_Specialized\_Data\_Types. The tag used in this example is “Banner\_32out”. This tag represents the full raw process data along with port status information.
5. Drag the necessary files from the Process Data folder.
  - a. Move all the files in the PLC Data Types to the PLC Data Types area in TIA Portal.
  - b. Move Banner\_CTL50\_Adv\_PD, Banner\_CTL50\_LED\_PD, Banner\_CTL50\_Over\_PD, and Banner\_CTL50\_PD to the Program Blocks area.
6. Go to PLC Tags. Create two tags. One tag is for the full data structure while the second creates a tag to represent the raw Process Data from the IO-Link Master. In this example, Tag table\_1 was created, then the tag “CTL50 IOLM1 01 PDO” was created using a Data Type of “Banner\_32out”. This naming convention calls out the type of device in question as well as the specific IO-Link Master and port number to which the sensor is connected. A different IO-Link Master might be named IOLM2 or IOLM3, for instance, and other specific sensors may be connected to different port numbers. The “Q” address found in step 3 (%Q1) is tied to this new tag. The second is “CTL50 IOLM1 01 outRaw” and uses the “Q” address found in step 2 (%Q3). This is the tag that will be used in the Function block.



|                         |                       |       |
|-------------------------|-----------------------|-------|
| ▶ CTL50 IOLM1 01 outRaw | "Banner_CTL50_PDORaw" | %Q3.0 |
| ▶ CTL50 IOLM1 01 PDO    | "Banner_32Out"        | %Q1.0 |

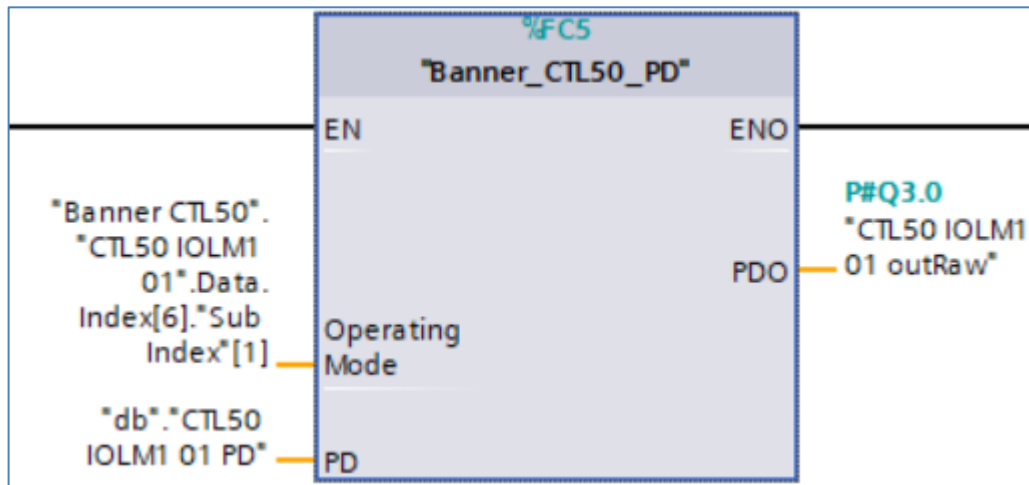
7. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named “db”.



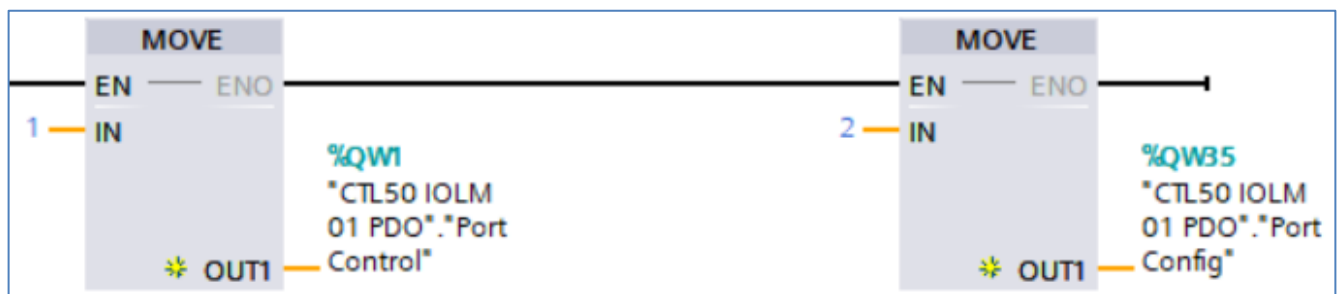
8. In the new data block, we will create a new tag to represent the parsed Process Data In for our CTL50. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type “Banner\_CTL50\_PDO” for the new tag.

|                     |                                 |
|---------------------|---------------------------------|
| ▼ CTL50 IOLM1 01 PD | "Banner_CTL50_PDO"              |
| ■ ▶ 0-Run           | "Banner_CTL50_Run"              |
| ■ ▶ 1-Segment       | "Banner_CTL50_Segment"          |
| ■ ▶ 2-Override      | "Banner_CTL50_Segment_Override" |
| ■ ▶ 3-Advanced      | "Banner_CTL50_Advanced"         |
| ■ ▶ 4-Level         | "Banner_CTL50_Level"            |
| ■ ▶ 5-Gauge         | "Banner_CTL50_Gauge"            |
| ■ ▶ 6-LED           | "Banner_CTL50_LED"              |

9. Add the “Banner\_CTL50\_PD” function to an OB ladder. Link the “PDO” to the raw process data variable from step 6. The tag name again calls out the type of device, IO-Link Master, and the port number. Use the variable was called “CTL50 IOLM1 01 outRaw” in this example. The “CTL50 PD” needs to be linked to the variable created in step 8. It was called “CTL50 IOLM1 01 PD” for this example.



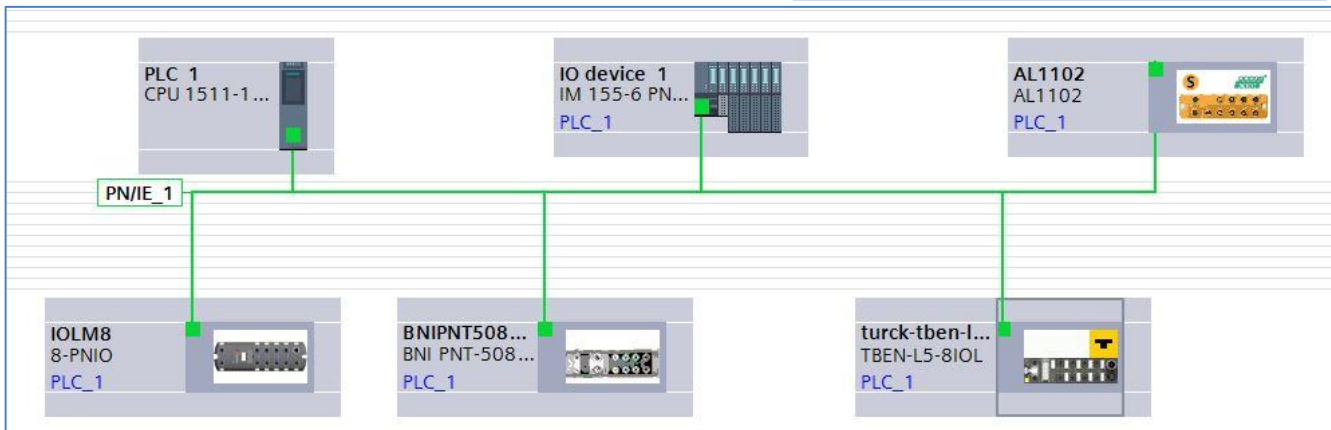
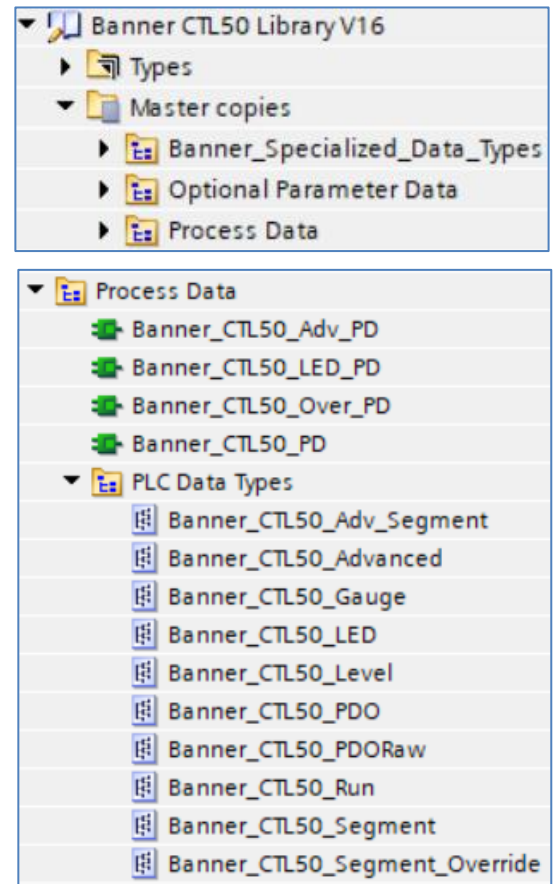
10. The final step is to configure the IO-Link output control. This is done by sending a 1 to Port Control and a 2 to Port Config. Both parameters are part of the tag created in step 6 “CTL50 IOLM1 01 PDO”.



11. Process Data Setup is complete.  
 12. Compile and download the configuration to the PLC, then go online. Open the “db” data block and click Monitor all. The CTL50 Select can be controlled now.

### **Setup of CTL50 Select with other IO-Link Masters**

1. The Banner CTL50 Library will now be in the Global Library List. Expand the Master copies section. The CTL50 folder contains elements for both Process Data and Parameter Data connections to a CTL50 device.
2. Drag Banner\_CTL50Pro\_Adv\_PD, Banner\_CTL50\_LED\_PD, Banner\_CTL50\_Over\_PD, and Banner\_CTL50\_PD to the Program Blocks area under your PLC.
3. Drag all item in the PLC Data Types folder to the PLC Data Types area under your PLC.
4. Go to Devices and networks to configure the system as necessary. Below is an example of what a configuration might look like. This example shows 5 different IO-Link Masters connected to the same PLC.



5. Click on the relevant device and configure the IO-Link Master as necessary. Refer to the documentation for the IO-Link Master. Recall that a CTL50 requires 32 bytes of space for the Process Data.
6. Record the "Q" address where this CTL50 Process Data is to be stored, as the address will be required in the next step.

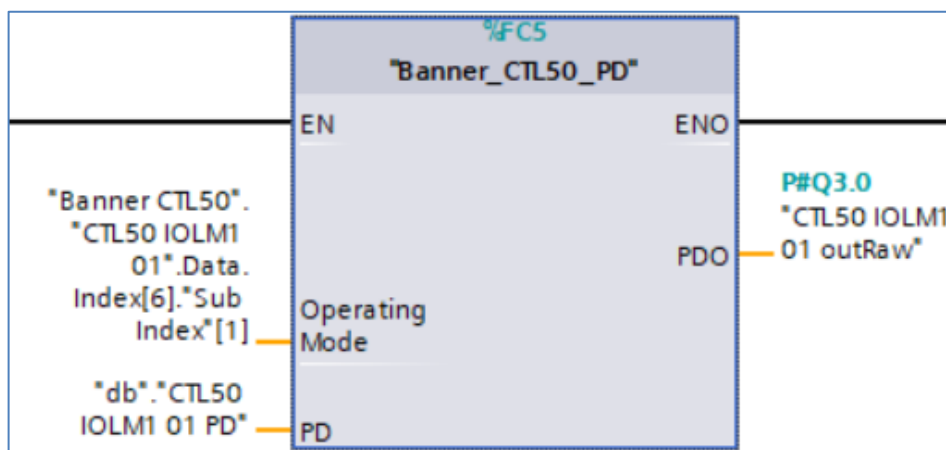
7. Go to PLC Tags. Add a new tag table, then create a new tag to represent the raw Process Data Out to be sent to the IO-Link Master. In this example, Tag table\_1 was created, then the tag “CTL50 IOLM1 01 outRaw” was created using a Data Type of “Banner\_CTL50\_PDORaw”. This naming convention calls out the type of sensor in question as well as the specific IO-Link Master and port number where the sensor is connected. A different IO-Link Master might be named IOLM2 or IOLM3, for instance, and other specific sensors may be connected to different port numbers. The “Q” address found in step 6 is tied to this new tag.

► CTL50 IOLM1 01 outRaw "Banner\_CTL50\_PDORaw" %Q3.0

8. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named “db”.
9. In the new data block, create a new tag to represent the parsed Process Data In for our CTL50 Pro. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type “Banner\_CTL50\_PDO” for the new tag.

|                     |                                 |
|---------------------|---------------------------------|
| ▼ CTL50 IOLM1 01 PD | "Banner_CTL50_PDO"              |
| ► 0-Run             | "Banner_CTL50_Run"              |
| ► 1-Segment         | "Banner_CTL50_Segment"          |
| ► 2-Override        | "Banner_CTL50_Segment_Override" |
| ► 3-Advanced        | "Banner_CTL50_Advanced"         |
| ► 4-Level           | "Banner_CTL50_Level"            |
| ► 5-Gauge           | "Banner_CTL50_Gauge"            |
| ► 6-LED             | "Banner_CTL50_LED"              |

10. Add the “Banner\_CTL50Pro\_PDO” function to an OB ladder. Link the “PDO” to the raw Process Data variable from step 7. Link “CTL50 Pro PD” to the parsed Process Data variable from step 9.



11. Process Data setup is complete.

**Using CTL50 Process Data**

The CTL50 has 7 different Process Data Modes. The default mode is Segment Override (2). Set the unit to the correct mode for your application (Parameter Data controls this).

The Override mode allows for each segment to be controlled individually. The Segment Array allows for control of up to 20 segments. Parameter Data is used to configure these segments. Use Advanced mode for full control of the segments.

|                            |                        |   |                                                                                  |
|----------------------------|------------------------|---|----------------------------------------------------------------------------------|
| ▼ CTL50 IOLM1 01 PD        | *Banner_CTL50_PDO*     |   |                                                                                  |
| ■ ▶ 0-Run                  | *Banner_CTL50_Run*     |   |                                                                                  |
| ■ ▶ 1-Segment              | *Banner_CTL50_Segme... |   |                                                                                  |
| ■ ▼ 2-Override             | *Banner_CTL50_Segme... |   |                                                                                  |
| ■ ▶ Segment                | Array[1..20] of USInt  |   | 0=Off, 1=On, 2=Flash, 3=Animation                                                |
| ■ Audio Type               | USInt                  | 0 | 0=Off, 1=Pulse, 2=Wobble, 3=Strobe, 4=Whoop, 5=Stacatto, 6=Siren, 7=Continuo     |
| ■ Audio Volume             | USInt                  | 0 | 0=Low, 1=Medium Low, 2=Medium High, 3=High                                       |
| ■ Animation Override       | USInt                  | 0 | 0=Override Not Enabled, 1=Override Enabled                                       |
| ■ Animation                | USInt                  | 0 | 0=Off, 1=Steady, 2=Flash, 3=2 Color Flash, 4=2 Color Shift, 5=50/50 Rotate, 6=Ch |
| ■ Color 1                  | USInt                  | 0 | 0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Sprin |
| ■ Color 1 Intensity        | USInt                  | 0 | 0=High, 1=Low, 2=Medium, 3=Off, 4=Custom                                         |
| ■ Speed                    | USInt                  | 0 | 0=Medium, 1=Fast, 2=Slow, 3=Custom Flash Rate                                    |
| ■ Pulse Pattern            | USInt                  | 0 | 0=Normal, 1=Strobe, 2=Three Pulse, 3=SOS, 4=Random                               |
| ■ Color 2                  | USInt                  | 0 | 0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Sprin |
| ■ Color 2 Intensity        | USInt                  | 0 | 0=High, 1=Low, 2=Medium, 3=Off, 4=Custom                                         |
| ■ Scroll/Bounce Style      | USInt                  | 0 | 0=Solid, 1=Tail, 2=Ripple                                                        |
| ■ Percent Width of Color 1 | USInt                  | 0 | 1 to 100 %                                                                       |
| ■ Direction                | USInt                  | 0 | 0=Up, 1=Down                                                                     |