



TL50 Process Data AOI Guide, v4

October 19th, 2023

This document covers the installation and use of an Add-On Instruction (AOI) for the Logix Designer software package from Rockwell Automation. This AOI handles cyclic IO-Link Process Data Out to a Banner TL50 device via an IO-Link Master connected to an Allen-Bradley PLC. The AOI covers parsing and display of the TL50 device Process Data Out. The AOI has one User Defined Tag data type.

Components

Banner_TL50_PD_v4_AOI.L5X

UDT Packaged with the AOI

Banner_TL50_PDO_v4

Other AOIs Available Separately

Banner has AOI files for controlling other Banner IO-Link devices and for a variety of IO-Link Masters. Banner also has AOI files for easily handling Banner device Process Data.

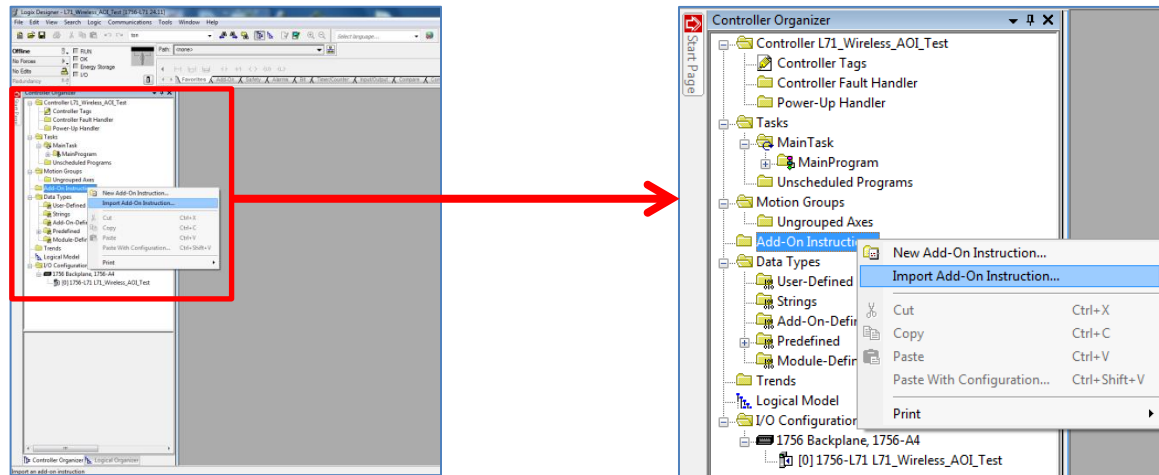
Contents

- 1. Installation Process 1
- 2. Configuring the IO-Link Master 3
- 3. Configuring the AOI..... 4
- 4. Using the AOI..... 7
- Appendix A K50 Pro L2 Process Data 8
- Appendix B IO-Link Master Cheat Sheet 9

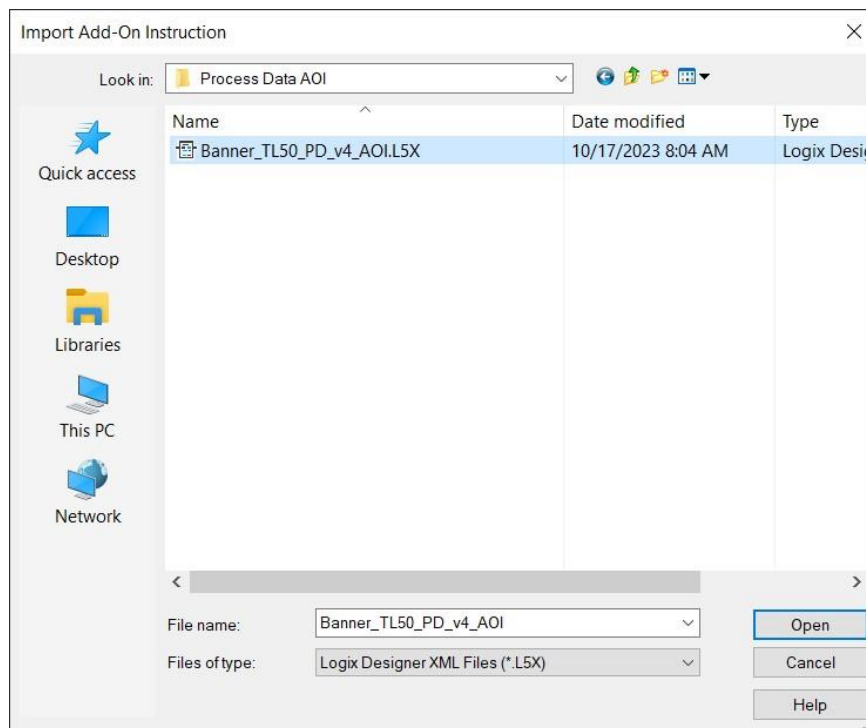
1. Installation Process

This section describes how to install the AOI in Logix Designer software.

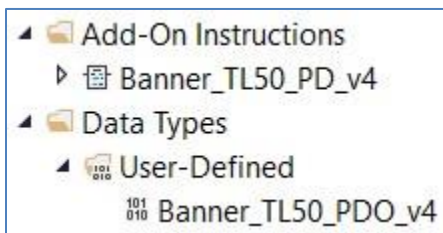
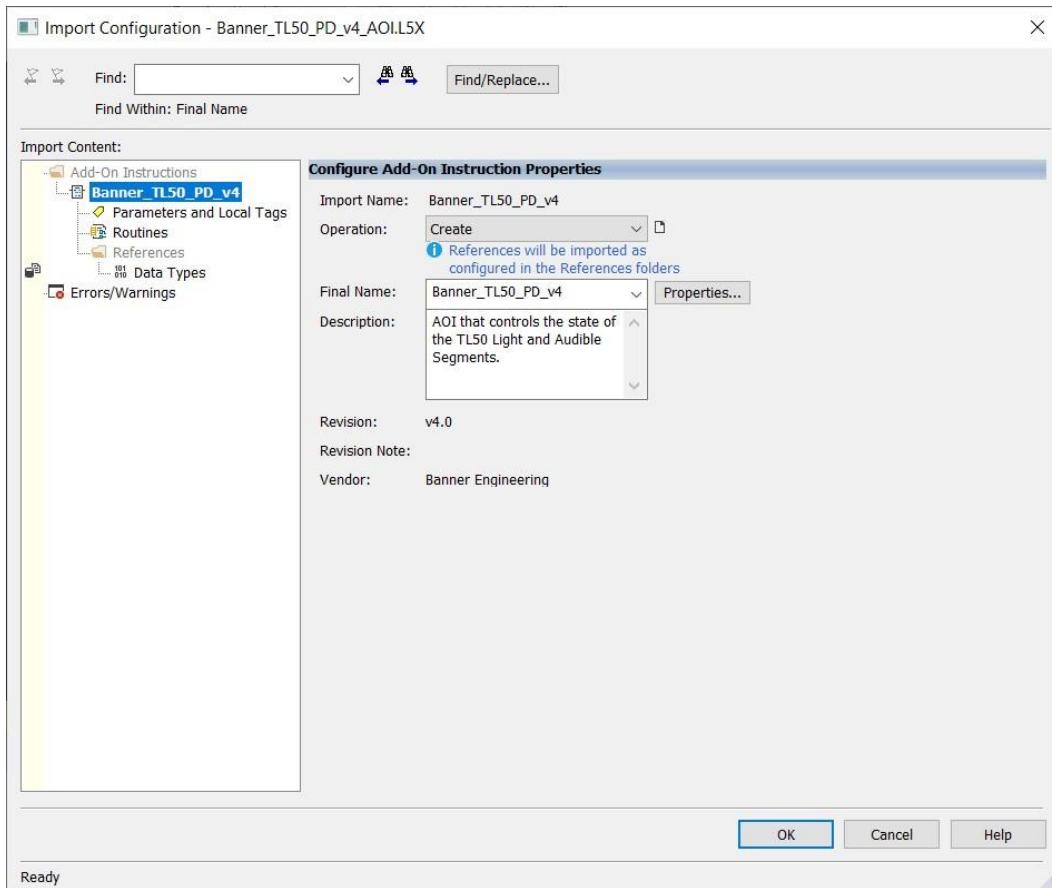
1. Open a project.
2. In the Controller Organizer window, right-click on the Add-On Instruction folder. Select the Import Add-On Instruction option.



3. Navigate to the correct file location and select the AOI to be installed. In this example the "Banner_TL50_PD_v4_AOI.L5X" file will be selected. Click the Open button.



4. The Import Configuration window will pop up. The default selection will create all the necessary items for the AOI. Click the OK button to complete the import process.



5. The AOI is added to the Controller Organizer window and should look like the picture at left.
6. AOI installation into the Logix Designer software complete.

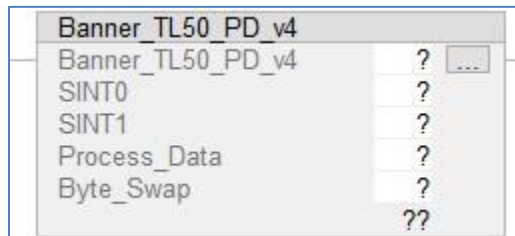
2. Configuring the IO-Link Master

Make an EtherNet/IP connection to the IO-Link Master.

Create an Ethernet communications module for the IO-Link Master device. The controller tags generated include Input (I) and Output (O) Assembly Instances. Each Assembly has a corresponding tag array. Creating this Class 1 EtherNet/IP implicit IO connection will provide the PLC access to the IO-Link device Process Data. Each port on the IO-Link Master is given a dedicated group of I and O registers. See the relevant IO-Link Master User's Guide for more information.

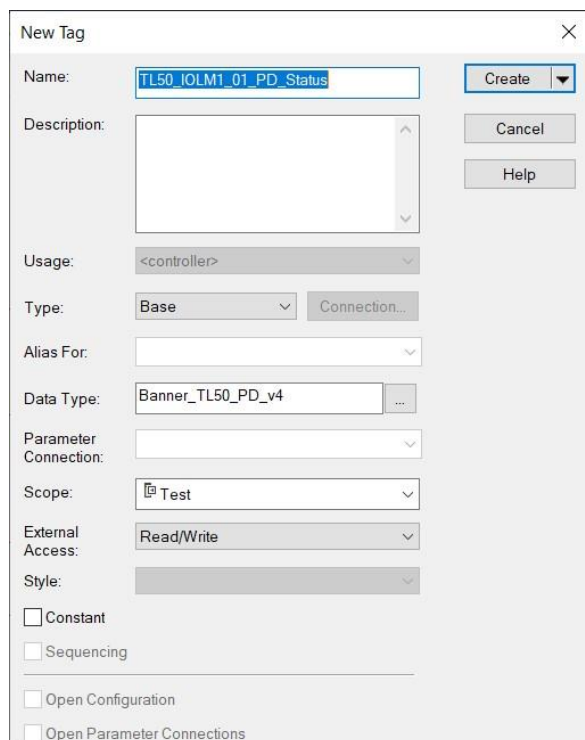
3. Configuring the AOI

1. Add the “Banner_TL50_PD_v4” AOI to your ladder logic program. For each of the question marks shown in the instruction we need to create and link a new tag array. The AOI includes a new type of User Defined Tags (UDT): a custom array of tags meant specifically for this AOI.



2. In the AOI, right-click on the question mark on the line labeled “Banner_TL50_PD_v4”. Click New Tag. Name the new tag. This example uses the name “TL50_IOLM1_01_PD_Status”. The example naming convention accounts for this being a TL50 connected to IO-Link Master #1, port #1, in our program. More masters could be named IOLM2, IOLM3, and different sensors could be connected at other port numbers, etc.

Note that the Data Type is the User-Defined Data Type (UDT) entitled “Banner_TL50_PD_v4”. This custom-made array of registers is specially built to handle the memory needs of this AOI. Click Create to make the tag array.



New Tag

Name:

Description:

Usage:

Type:

Alias For:

Data Type:

Parameter Connection:

Scope:

External Access:

Style:

☐ Constant

☐ Sequencing

☐ Open Configuration

☐ Open Parameter Connections

- Next, we will link the lines in the AOI labeled “SINT0” and “SINT1” with the group of registers the IO-Link Master has reserved for the TL50 in question. These registers are where to place the Process Data Out that this AOI will make from the labeled component pieces. In this case, the TL50 is connected to port 1. According to the documentation for the IO-Link Master used in this example, the Process Data Out for a device connected to port 1 is begins at byte 182 in the Output Assembly Instance.

See the relevant IO-Link Master documentation or Appendix B for more information on which registers contain the raw Process Data you need.

| Banner_TL50_PD_v4 | |
|-------------------|-------------------------|
| Banner_TL50_PD_v4 | TL50_IOLM1_01_PD_Status |
| SINT0 | IOLM1:O.Data[182] |
| SINT1 | IOLM1:O.Data[183] |
| Process_Data | TL50_IOLM1_01_PD |
| Byte_Swap | 0 |

- Now we will right-click on the question mark on the line labeled “Process_Data” in the AOI. Click on “New Tag”. Give the tag a name. This example uses the name “TL50_IOLM1_01_PD”. Notice that the Data Type is “Banner_TL50_PDO_v4”. Click Create.

New Tag

Name:

TL50_IOLM1_01_PD

Create

Description:

Cancel

Usage:

<controller>

Help

Type:

Base

Connection...

Alias For:

Data Type:

Banner_TL50_PDO_v4

Parameter Connection:

Scope:

Test

External Access:

Read/Write

Style:

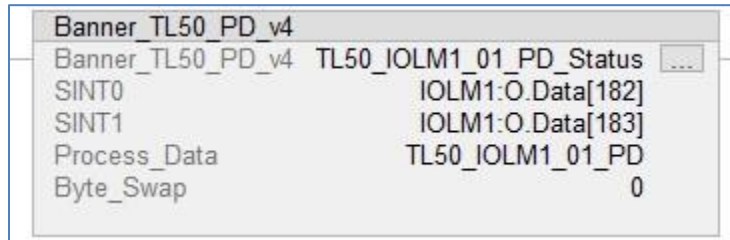
☐ Constant

☐ Sequencing

☐ Open Configuration

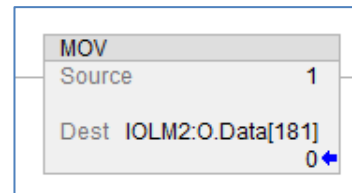
☐ Open Parameter Connections

5. The last line in the AOI is a setting to account for byte swapping. In the case of the TL50, the Process Data is two bytes long. IO-Link Masters may read those two bytes in either order, so this AOI must be ready to perform a byte swap. Enter a “0” or a “1” to toggle this setting. See Appendix B for more information.



If a Banner IO-Link Master is being used, setup a Move block. Send a 1 to the Activate Outputs array value (see table for each port’s value). As an example, if port 1 needs the process data outputs active then send a 1 to 181.

| IO-Link Master Port | Activate Outputs |
|---------------------|------------------|
| 1 | 181 |
| 2 | 215 |
| 3 | 249 |
| 4 | 283 |
| 5 | 317 |
| 6 | 351 |
| 7 | 385 |
| 8 | 419 |



The “Banner_TL50_PD_v4” AOI is now ready for use.

4. Using the AOI

The “Banner_TL50_PD_v4” Add-On Instruction has created a group of tags representing the TL50 Process Data Out, broken out into its component parts.

Look in the Controller Tags to find the name you used above. This example used the name “TL50_IOLM1_01_PD”. The tag array, seen below, has individual pieces of information instead of 16 unlabeled bits. By changing the Segment_# values you can control the lights. If a value of 1 is used the segment will turn on solid. Value of 2 flashes the segment. A value of 0 turns the segment off.

| | |
|------------------------------|-------|
| ▲ TL50_IOLM1_01_PD | {...} |
| ▶ TL50_IOLM1_01_PD.Segment_1 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_2 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_3 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_4 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_5 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_6 | 0 |
| ▶ TL50_IOLM1_01_PD.Segment_7 | 0 |

Appendix A TL50 Process Data

The TL50 has 2 bytes of Process Data Out, as shown below.

| ProcessDataOut "Process Data Out" id=V_Pd_OutSegState | | | | | | | | | |
|---|------------|----------------|----------------------------|---------------|-------------|-----------------|---------------|--------------------------|-------------|
| bit length: 16 data type: 16-bit Record | | | | | | | | | |
| subindex | bit offset | data type | allowed values | default value | acc. restr. | mod. other var. | excl. from DS | name | description |
| 1 | 0 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 1 - off/on/flash | |
| 2 | 2 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 2 - off/on/flash | |
| 3 | 4 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 3 - off/on/flash | |
| 4 | 6 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 4 - off/on/flash | |
| 5 | 8 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 5 - off/on/flash | |
| 6 | 10 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 6 - off/on/flash | |
| 7 | 12 | 2-bit UInteger | 0 = Off, 1 = On, 2 = Flash | | | | | Segment 7 - off/on/flash | |

This Process Data is mapped to a specific group of EtherNet/IP registers. The 16-bits of Process Data encode 7 separate pieces of information, as shown above.

This AOI intelligently parses this Process Data into its component pieces.

Appendix B IO-Link Master Cheat Sheet

Different IO-Link Masters behave differently in several ways. For one, the register locations where Process Data is stored varies. For another, some IO-Link Masters require byte-swapping and/or word-swapping. The tables below aim to define some of these differences. Note that these numbers are when using all default settings. IO-Link Masters can change the register locations to which Process Data is mapped in response to non-default, optional settings. See relevant IO-Link Master documentation for more information.

PDI (Process Data In) is found in the IO-Link Master's T->O (PLC "Input") Assembly Instance.

PDO (Process Data Out) is found in the IO-Link Master's O->T (PLC "Output") Assembly Instance.

Table 1. First Register of Process Data "SINT0"

| Port | Allen-Bradley* | | Comtrol | | Balluff | | Turck | | ifm | | Banner | |
|------|----------------|--------------|---------|-----|---------|-----|-------|-----|-----|-----|--------|-----|
| | PDI | PDO | PDI | PDO | PDI | PDO | PDI | PDO | PDI | PDO | PDI | PDO |
| 1 | I.Ch0Data[0] | O.Ch0Data[0] | 4 | 0 | 8 | 6 | 6 | 4 | 190 | 46 | 184 | 182 |
| 2 | I.Ch1Data[0] | O.Ch1Data[0] | 40 | 32 | 56 | 38 | 38 | 36 | 222 | 78 | 218 | 216 |
| 3 | I.Ch2Data[0] | O.Ch2Data[0] | 76 | 64 | 104 | 70 | 70 | 68 | 254 | 110 | 252 | 250 |
| 4 | I.Ch3Data[0] | O.Ch3Data[0] | 112 | 96 | 152 | 102 | 102 | 100 | 286 | 142 | 286 | 284 |
| 5 | I.Ch4Data[0] | O.Ch4Data[0] | 148 | 128 | 200 | 134 | 134 | 132 | 318 | 174 | 320 | 318 |
| 6 | I.Ch5Data[0] | O.Ch5Data[0] | 184 | 160 | 248 | 166 | 166 | 164 | 350 | 206 | 354 | 352 |
| 7 | I.Ch6Data[0] | O.Ch6Data[0] | 220 | 192 | 296 | 198 | 198 | 196 | 382 | 238 | 388 | 386 |
| 8 | I.Ch7Data[0] | O.Ch7Data[0] | 256 | 224 | 344 | 230 | 230 | 228 | 414 | 270 | 422 | 420 |

*see relevant Banner Allen-Bradley IO-Link Master AOI Guide and Allen-Bradley User Guides for more information on using device IODD files to aid in integration.

Note: Murr IO-Link Masters have configurable process data. Refer to the Murr IO-Link Master Instruction Manual for Process Data mappings.

Table 2. Byte-Swap

| IO-Link Master | Byte Swap |
|----------------|-----------|
| Allen-Bradley | 0 |
| Comtrol | 1 |
| Balluff | 0 |
| Turck | 1 |
| ifm | 1 |
| Murr | 0 |
| Banner | 0 |

Specific hardware used in both tables (all default settings):

- Allen-Bradley Armor Block I/O IO-Link Master (1732E-8IOLM12R)
- Comtrol 8-EIP IO-Link Master (99608-8)
- Balluff BNI006A (BNI EIP-508-105-Z015)
- Turck TBEN-L5-8IOL
- ifm AL1122
- Murr Impact67 E DIO 12 DIO4/IOL4 4P (Art.-No. 55144)

Banner IO-Link Masters (DXMR90-4K) have a port status register. The register gives the status of the port. It gives information on if the port has an IO-Link device connected and if Process Data is valid. This is optional information but is useful for troubleshooting. The data comes into the PLC as bytes while the literature shows the value as a word. The table below gives the upper- and lower-byte data location in the PLC. The upper byte includes bits 15 through 8, while the lower byte has bits 7 through 0.

| IO-Link Master Port | Upper Bits 15 - 8 | Lower Bits 7 - 0 |
|------------------------|----------------------|---------------------|
| 1 | 182 | 183 |
| 2 | 216 | 217 |
| 3 | 250 | 251 |
| 4 | 284 | 285 |
| 5 | 318 | 319 |
| 6 | 352 | 353 |
| 7 | 386 | 387 |
| 8 | 420 | 421 |

Port Status:

Bit0 = Connected?
Bit1 = Process Data Valid?
Bit2 = Event Pending?
Bit3 = Ready for ISDU?
Bit4 = Pin4 SIO State
Bit5 = Pin2 SIO State

Bit6-7 = Pin4 Mode:

SDCI Mode = 0
 SIO Input Mode = 1
 SIO Output Mode = 2

Bit8-10 = Pin2 Mode:

Disabled = 0
 Input Normal = 1
 Output = 2
 Diagnostic Input = 3
 Inverted Input = 4