



## **R90C-B21 or R95C-B21 Process Data AOI Guide, v4**

### **October 23<sup>rd</sup>, 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 from a Banner R90C-B21 or R95C-8B21 device via an IO-Link Master to an Allen-Bradley PLC.

#### **Components**

Banner\_R90C\_R95C\_PD\_v4\_AOI.L5X

#### **UDTs Packaged with the AOI**

Banner\_R90C\_R95\_PDIO\_v4  
Banner\_R90C\_R95\_PDI\_Discrete\_v4  
Banner\_R90C\_R95\_PDI\_v4  
Banner\_R90C\_R95\_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 Parameter Data.

**Contents**

1. Installation Process ..... 1

2. Configuring the IO-Link Master ..... 3

3. Configuring the AOI ..... 4

4. Using the AOI ..... 8

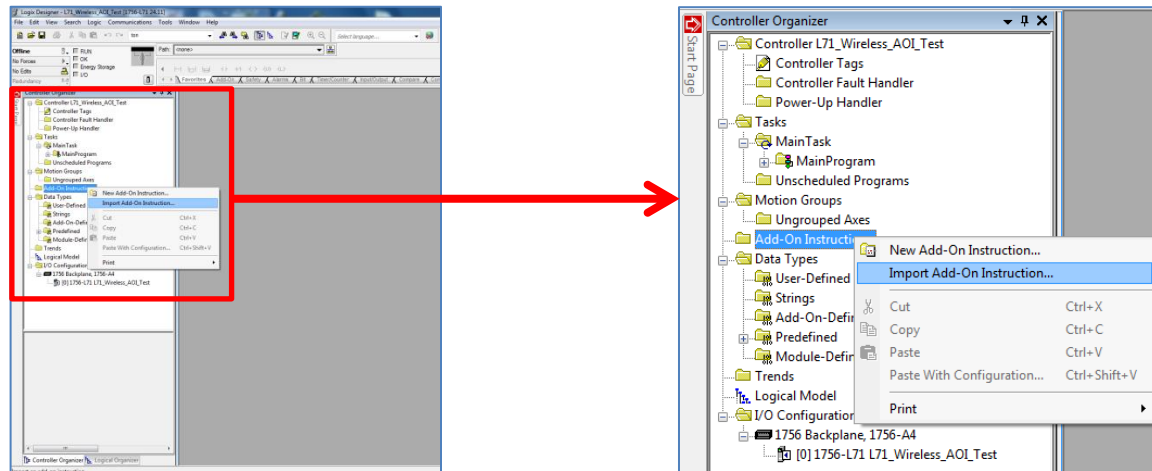
Appendix A R90C-B21 R95C-8B21 Process Data ..... 10

Appendix B IO-Link Master Cheat Sheet ..... 13

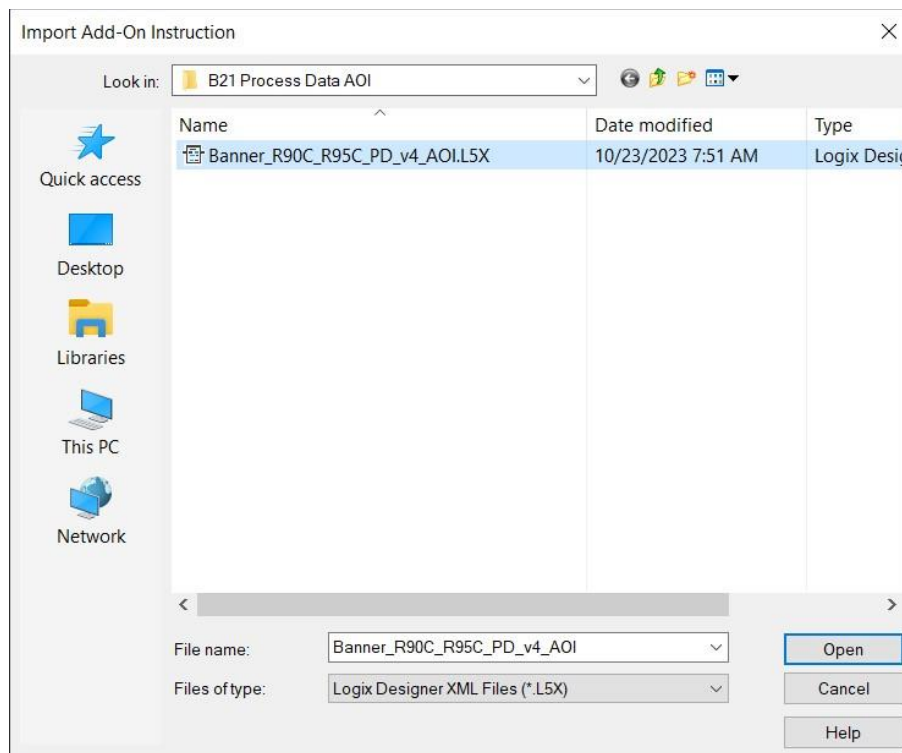
## 1. Installation Process

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

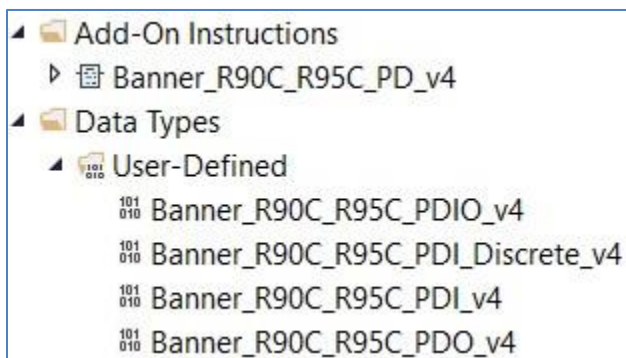
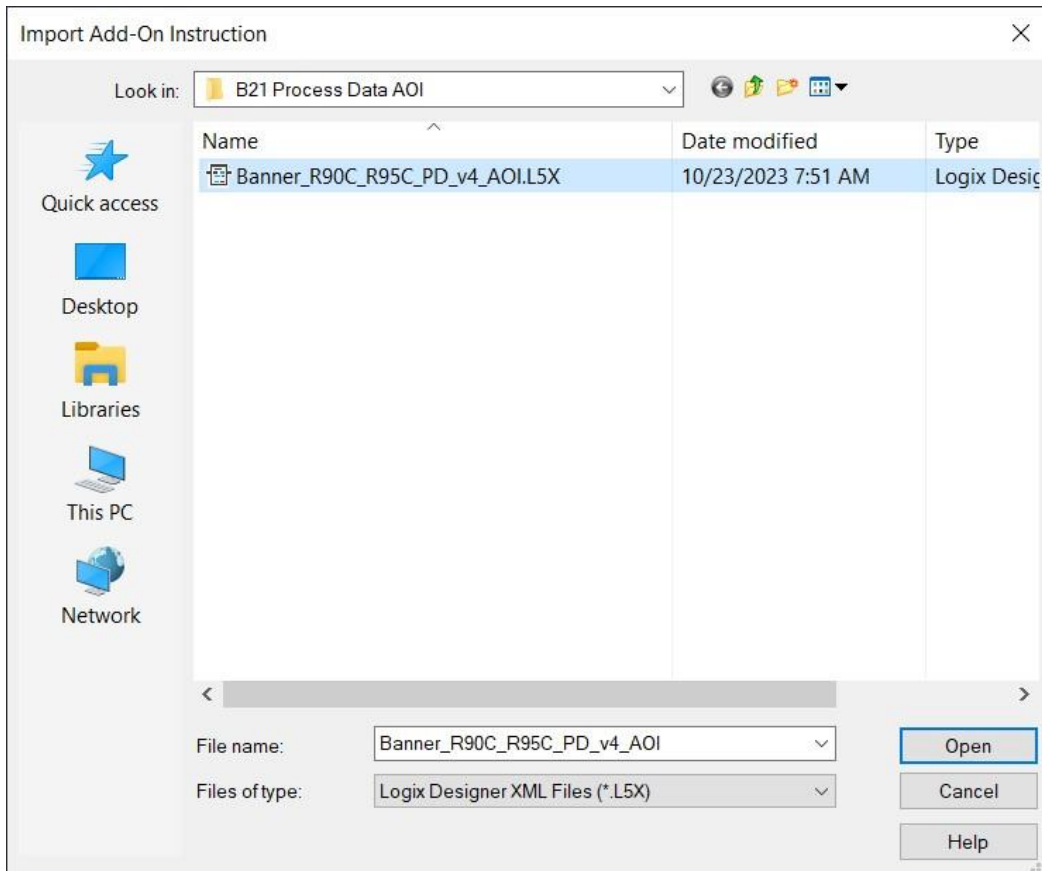
1. Open a project.
2. In the Controller Organizer window, right-click on the Add-On Instruction folder. Select the Import Add-On Instruction option.



3. Navigate to the correct file location and select the AOI to be installed. In this example the "Banner\_R90C\_R95C\_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.

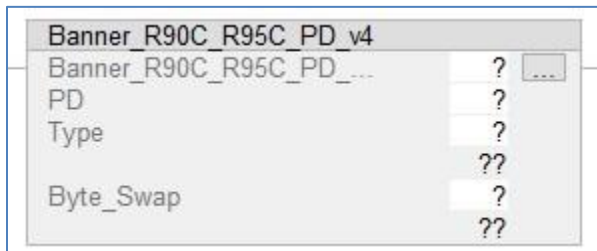
## 2. Configuring the IO-Link Master

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.

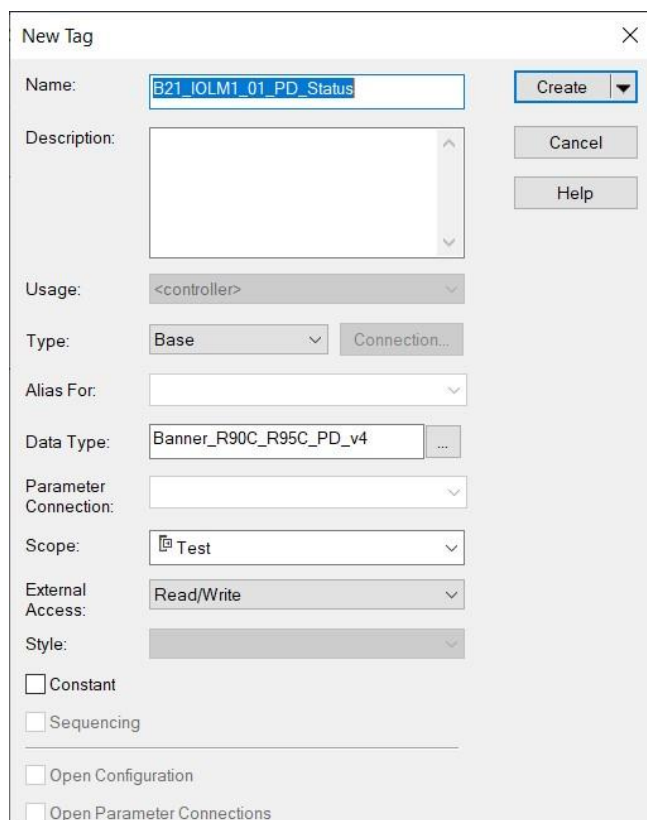
### 3. Configuring the AOI

1. Add the “Banner\_R90C\_R95C\_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\_R90C\_R95\_PD\_v4”. Click New Tag. Name the new tag. This example uses the name “B21\_IOLM1\_01\_PD\_Status”. The example naming convention accounts for this being a R90C-4B21 device 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\_R90C\_R95\_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.



- 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 "B21\_IOLM1\_01\_PD". Notice that the Data Type is "Banner\_R90C\_R95\_PDIO\_v4". Click Create.

This array will handle the displaying of the parsed Process Data for the R90C-4B21 or R95C-8B21.

**New Tag**

Name:

Description:

Usage:

Type:

Alias For:

Data Type:

Parameter Connection:

Scope:

External Access:

Style:

☐ Constant

☐ Sequencing

☐ Open Configuration

☐ Open Parameter Connections

- The next line in the AOI is a setting for which type of B21 is being used. A 4 or an 8 should be entered depending on if a R90C-4B21 or a R95C-8B21 is being used. The example shows an R90C-4B21 being used.

Banner_R90C_R95C_PD_...	B21_IOLM1_01_PD_Status
PD	B21_IOLM1_01_PD
Type	4
Byte_Swap	0

5. The final line in the AOI is a setting to account for byte swapping. In the case of the R90C-4B21, the Process Data In is 18 bytes long and the Process Data Out is 1 byte. IO-Link Masters may read each pair of 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.

**NOTE:** If the IO-Link Master you are using requires byte swapping be set to “1”, the single byte of Process Data Out the R90C-4B21 or R95C-8B21 uses will show up in one register higher than that listed on table 1 in Appendix B.

6. The final step required before we download and run the R90C-4B21 or R95C-8B21 Process Data AOI involves adding two File Synchronous Copy (CPS) instructions. These instructions allow the AOI to read from and write to the raw Process Data values found in the register tags of the IO-Link Master.

Add a CPS instruction before the AOI on the ladder rung that looks like the one seen below. Refer to Appendix B for which byte to start with in the “Source” area. In this case, the IO-Link Master in question has the raw Process Data In values for a device connected to port 1 starting at byte 6. For the “Destination”, we will enter the “DTI[0]” location, as seen below. Finally, the length will be 18 bytes, as that is the size of the S15C-B21 Process Data In.

CPS	
Source	IOLM1:I.Data[184]
Dest	B21_IOLM1_01_PD.DTI[0]
Length	18

The second CPS instruction sends the Process Data Output to the correct location for the IO-Link Master. It requires a size of 2 bytes. The Source will be the DTO[0] and is linked the to an output location as shown in Appendix B.

CPS	
Source	B21_IOLM1_01_PD.DTO[0]
Dest	IOLM1:O.Data[182]
Length	2

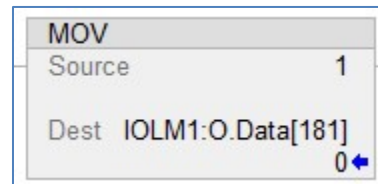
Here is what the entire rung looks like when completed.

<div>CPS</div> <div>Source IOLM1:I.Data[184]</div> <div>Dest B21_IOLM1_01_PD.DTI[0]</div> <div>Length 18</div>	<div>Banner_R90C_R95C_PD_v4</div> <div>Banner_R90C_R95C_PD_... B21_IOLM1_01_PD_Status</div> <div>PD B21_IOLM1_01_PD</div> <div>Type 8</div> <div>Byte_Swap 0</div>	<div>CPS</div> <div>Source B21_IOLM1_01_PD.DTO[0]</div> <div>Dest IOLM1:O.Data[182]</div> <div>Length 2</div>
--	--	---



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\_R90C\_R95\_PD\_v4" AOI is now ready for use.

## 4. Using the AOI

The “Banner\_R90C\_R95\_PD\_v4” Add-On Instruction has created a group of tags representing the R90C-4B21 or R95C-8B21 Process Data In, broken out into its component parts.

Look in the Controller Tags to find the name you used in Step 4 above. This example used the name “B21\_IOLM1\_01\_PD”. The tag array, seen below, has individual pieces of information instead of a group of unlabeled bits.

▲ B21_IOLM1_01_PD
▸ B21_IOLM1_01_PD.PDI
▸ B21_IOLM1_01_PD.PDO
▸ B21_IOLM1_01_PD.DTI
▸ B21_IOLM1_01_PD.DTO

This is the basic process data breakout. The PDI is the Process Data In for the system while the PDO is the Process Data Out.

▲ B21_IOLM1_01_PD.PDI	{...}
▸ B21_IOLM1_01_PD.PDI.Discrete_Info	{...}
▸ B21_IOLM1_01_PD.PDI.Measurement_1	2435
▸ B21_IOLM1_01_PD.PDI.Measurement_2	12
▸ B21_IOLM1_01_PD.PDI.Measurement_3	34
▸ B21_IOLM1_01_PD.PDI.Measurement_4	56

Expanding the PDI gives the following information. The Discrete Info can also be expanded. It gives the current On/OFF state of the discrete inputs 1 or 2. Measurements 1 through 4 give specific information. These can be configured as either a Counter, On Duration Timer, or Events per Minute value.

▲ B21_IOLM1_01_PD.PDO	{...}
▶ B21_IOLM1_01_PD.PDO.P1_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P2_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P3_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P4_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P5_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P6_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P7_Output_2	0
▶ B21_IOLM1_01_PD.PDO.P8_Output_2	0

The PDO allows the user to turn the discrete output on or off. This can only be done if discrete 2 is configured as an output via parameter data.

## Appendix A R90C and R95C Process Data

The R90C-4B21 or R95C-8B21 both have 18 bytes of Process Data In and 1 byte of Process Data Out.

### R90C-4B21

#### ProcessDataIn "Process Data Input" id=PD\_ProcessDataIn

bit length: 144

data type: 144-bit Record (subindex access not supported)

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	Boolean	false = Inactive, true = Active					Port1 Discrete1 Input State	true (1) = Discrete1 Input Active
2	1	Boolean	false = Inactive, true = Active					Port1 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
3	2	Boolean	false = Inactive, true = Active					Port2 Discrete1 Input State	true (1) = Discrete1 Input Active
4	3	Boolean	false = Inactive, true = Active					Port2 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
5	4	Boolean	false = Inactive, true = Active					Port3 Discrete1 Input State	true (1) = Discrete1 Input Active
6	5	Boolean	false = Inactive, true = Active					Port3 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
7	6	Boolean	false = Inactive, true = Active					Port4 Discrete1 Input State	true (1) = Discrete1 Input Active
8	7	Boolean	false = Inactive, true = Active					Port4 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
9	16	32-bit Integer						Process Data Measurement 1	Process Data Measurement 1 Value
10	48	32-bit Integer						Process Data Measurement 2	Process Data Measurement 2 Value
11	80	32-bit Integer						Process Data Measurement 3	Process Data Measurement 3 Value
12	112	32-bit Integer						Process Data Measurement 4	Process Data Measurement 4 Value

#### ProcessDataOut "Process Data Output" id=PD\_ProcessDataOut

bit length: 8

data type: 8-bit Record (subindex access not supported)

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	Boolean	false = Off, true = On					Port1 Discrete2 Output State	true (1) = Discrete2 Output Active
2	1	Boolean	false = Off, true = On					Port2 Discrete2 Output State	true (1) = Discrete2 Output Active
3	2	Boolean	false = Off, true = On					Port3 Discrete2 Output State	true (1) = Discrete2 Output Active
4	3	Boolean	false = Off, true = On					Port4 Discrete2 Output State	true (1) = Discrete2 Output Active

## R95C-8B21

ProcessDataIn "Process Data Input" id=PD_ProcessDataIn									
bit length: 144									
data type: 144-bit Record (subindex access not supported)									
subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	Boolean	false = Inactive, true = Active					Port1 Discrete1 Input State	true (1) = Discrete1 Input Active
2	1	Boolean	false = Inactive, true = Active					Port1 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
3	2	Boolean	false = Inactive, true = Active					Port2 Discrete1 Input State	true (1) = Discrete1 Input Active
4	3	Boolean	false = Inactive, true = Active					Port2 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
5	4	Boolean	false = Inactive, true = Active					Port3 Discrete1 Input State	true (1) = Discrete1 Input Active
6	5	Boolean	false = Inactive, true = Active					Port3 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
7	6	Boolean	false = Inactive, true = Active					Port4 Discrete1 Input State	true (1) = Discrete1 Input Active
8	7	Boolean	false = Inactive, true = Active					Port4 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
9	8	Boolean	false = Inactive, true = Active					Port5 Discrete1 Input State	true (1) = Discrete1 Input Active
10	9	Boolean	false = Inactive, true = Active					Port5 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
11	10	Boolean	false = Inactive, true = Active					Port6 Discrete1 Input State	true (1) = Discrete1 Input Active
12	11	Boolean	false = Inactive, true = Active					Port6 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
13	12	Boolean	false = Inactive, true = Active					Port7 Discrete1 Input State	true (1) = Discrete1 Input Active
14	13	Boolean	false = Inactive, true = Active					Port7 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
15	14	Boolean	false = Inactive, true = Active					Port8 Discrete1 Input State	true (1) = Discrete1 Input Active
16	15	Boolean	false = Inactive, true = Active					Port8 Discrete2 Input State	true (1) = Discrete2 Input Active. Note - even if Discrete2 is configured as an output, the active state will be reflected at the input
17	16	32-bit Integer						Process Data Measurement 1	Process Data Measurement 1 Value
18	48	32-bit Integer						Process Data Measurement 2	Process Data Measurement 2 Value
19	80	32-bit Integer						Process Data Measurement 3	Process Data Measurement 3 Value
20	112	32-bit Integer						Process Data Measurement 4	Process Data Measurement 4 Value

**ProcessDataOut "Process Data Output" id=PD\_ProcessDataOut**

bit length: 8

data type: 8-bit Record (subindex access not supported)

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	Boolean	false = Off, true = On					Port1 Discrete2 Output State	true (1) = Discrete2 Output Active
2	1	Boolean	false = Off, true = On					Port2 Discrete2 Output State	true (1) = Discrete2 Output Active
3	2	Boolean	false = Off, true = On					Port3 Discrete2 Output State	true (1) = Discrete2 Output Active
4	3	Boolean	false = Off, true = On					Port4 Discrete2 Output State	true (1) = Discrete2 Output Active
5	4	Boolean	false = Off, true = On					Port5 Discrete2 Output State	true (1) = Discrete2 Output Active
6	5	Boolean	false = Off, true = On					Port6 Discrete2 Output State	true (1) = Discrete2 Output Active
7	6	Boolean	false = Off, true = On					Port7 Discrete2 Output State	true (1) = Discrete2 Output Active
8	7	Boolean	false = Off, true = On					Port8 Discrete2 Output State	true (1) = Discrete2 Output Active

## 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