K50 Pro Indicator Process Data Function

11/17/2022

This document covers the installation and use of a function for Siemen's TIA Portal software package. This function handles cyclic IO-Link Process Data Out to a Banner K50 Pro Indicator light via an IO-Link Master from a Siemens PLC. The function covers parsing and display of the K50 Pro Indicator sensor Process Data Out.

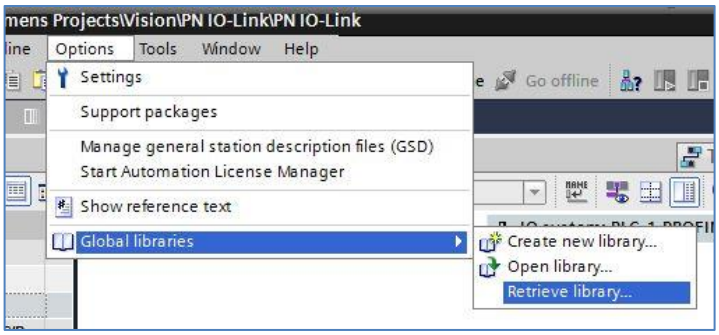
Components

Banner K50 Indicator.zal14

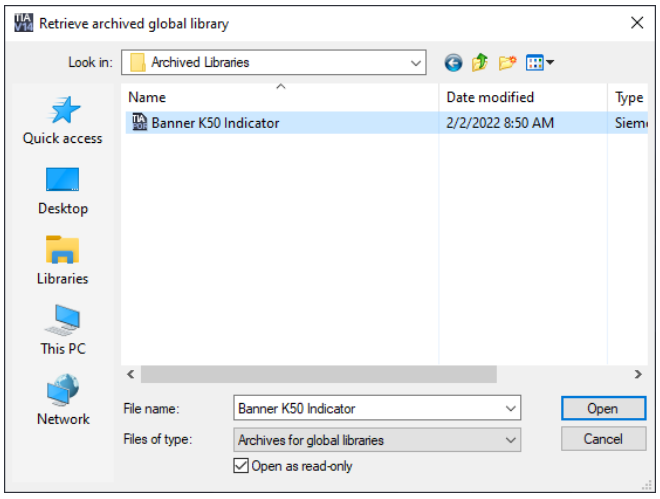
There are two methods for the process data. The first is used when creating a connection to Banner's IO-Link masters. The second set of instructions are for systems using other manufacturer's IO-Link masters.

Installation Instructions

1. Open a project.
2. Go to Options > Global Libraries > Retrieve Library.



3. Select the Banner K50 Indicator. Click Open.



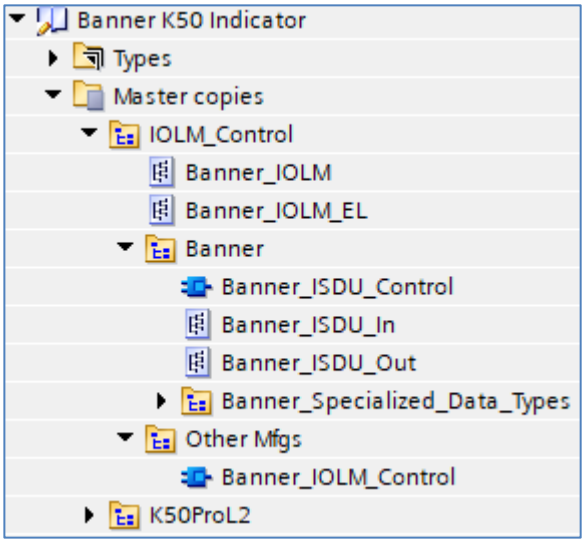
4. The library is now accessible in the Libraries tab.
5. Go to page 3 for Banner IO-Link Masters, to page 6 for all other IO-Link Masters.

Setup of K50 Pro L2 with a Banner DXMR90-4K

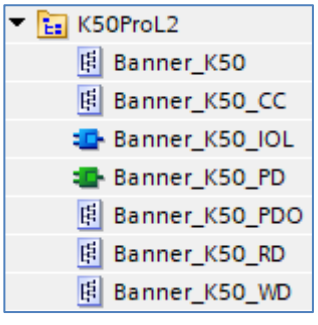
1. Go to Device and Networks to configure the DXMR90-4K. Add the DXMR90-4K if it has yet to be added to the system.
2. Add Banner IO-Link Master Info to Slot 1. This sets the DXMR90-4K for IO-Link mode.
3. Open the IO-Link Generic Devices and select the proper module. The 4/4 byte is required for K50 Pro Indicator. Make note of the I address for the Slot 2 which represents Port 1. Slot 2 starts are 1 for outputs. The other number needed is I3. The data for the port start at that point (I3). The previous two bytes Port Control.

Module	Rack	Slot	I address	Q address	Type
dxm	0	0			1-port Device
Interface	0	0 X1			dxm
Banner IO-Link Master Info_1	0	1	1...9		Banner IO-Link Master Info
IO-Link In/Out 4/ 4 Byte + Status_1	0	2	10...17	1...18	IO-Link In/Out 4/ 4 Byte + Status

4. Drag the necessary tag from IOLM_Control > Banner > Banner_Specialized_Data_Types. The tag used in this example is “Banner_4out”. This tag represents the full raw process data along with port status information.

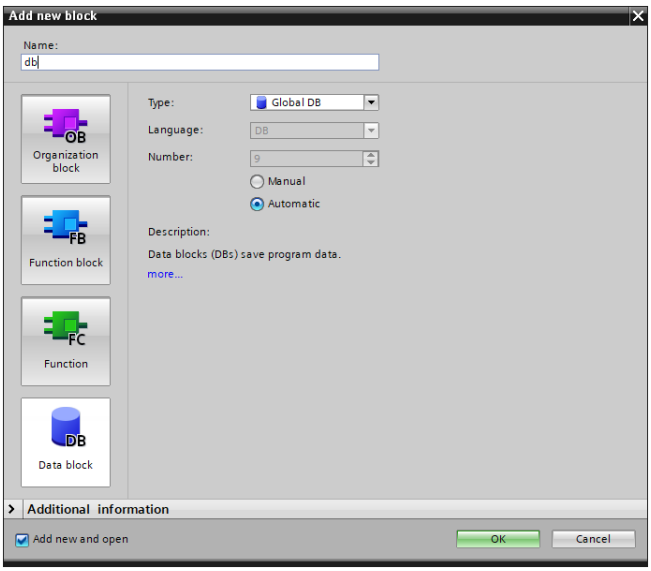


5. Drag the necessary files from the K50ProL2 Folder.
   a. Move Banner_K50_PDO to the PLC Data Types area.
   b. Move Banner_K50_PD to the Program Blocks area.
6. Go to PLC Tags. Create two tags. The first tag is for the full data structure while the second tag represents the raw Process Data from the IO-Link Master. In this example, Tag table_1 was created, then the tag “K50Pro IOLM1 01 PDO” was created using a Data Type of “Banner_4out”. This naming convention calls out the type of device 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 2 (%Q1) is tied to this new tag. The tag that represents the raw data is “K50Pro IOLM1 01 outRaw” and uses the “Q” address found in step 2 (%Q3).



Name	Data type	Address
K50Pro IOLM1 01 outRaw	DInt	%QD3
K50Pro IOLM1 01 PDO	"Banner_4Out"	%Q1.0

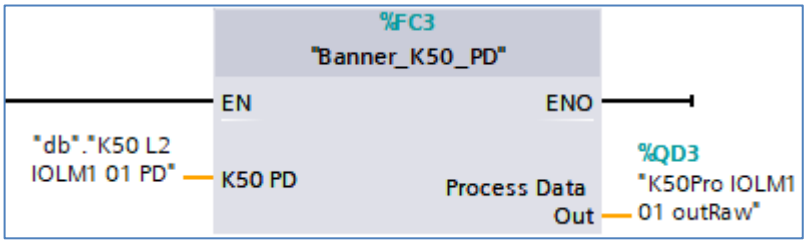
7. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named “db”.



8. In the new data block, create a new tag to represent the parsed Process Data Output for our K50 Pro L2. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type “Banner_K50_PDO” for the new tag.

Name	Data type
Static	
K50 L2 IOLM1 01 PD	"Banner_K50_PDO"
Color1	USInt
Color Flash Rate	USInt
Color2	USInt
Audible Mode	USInt
Rotation Direction	Bool
Animation Type	USInt
Color 1 Intensity	USInt
Color 2 Intensity	USInt

8. Add the “Banner_K50_PD” function to an OB ladder. Link the “Process Data Out” to the raw process data variable from step 5. The tag name again calls out the type of device, IO-Link Master, and the port number. Use the variable called “K50 L2 IOLM1 01 outRaw” in this example. The “K50 PD” needs to be linked to the variable created in step 7. It was called “K50Pro IOLM1 01 PD” for this example.

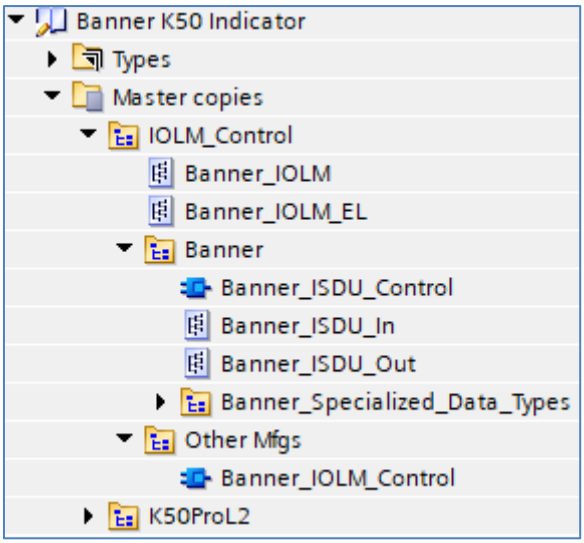


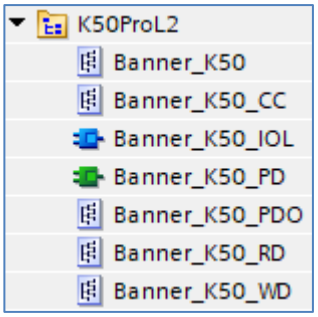
9. The final step is to configure the IO-Link output control. This is done by sending a 1 to Port Control and a 2 to Port Config. Both parameters are part of the tag created in step 6 “TLF100 IOLM1 01 PDO”.

10. Process Data Setup is complete.
11. Compile and download the configuration to the PLC, then go online. Open the “db” data block and click Monitor all. The K50 Pro Indicator can be controlled now.

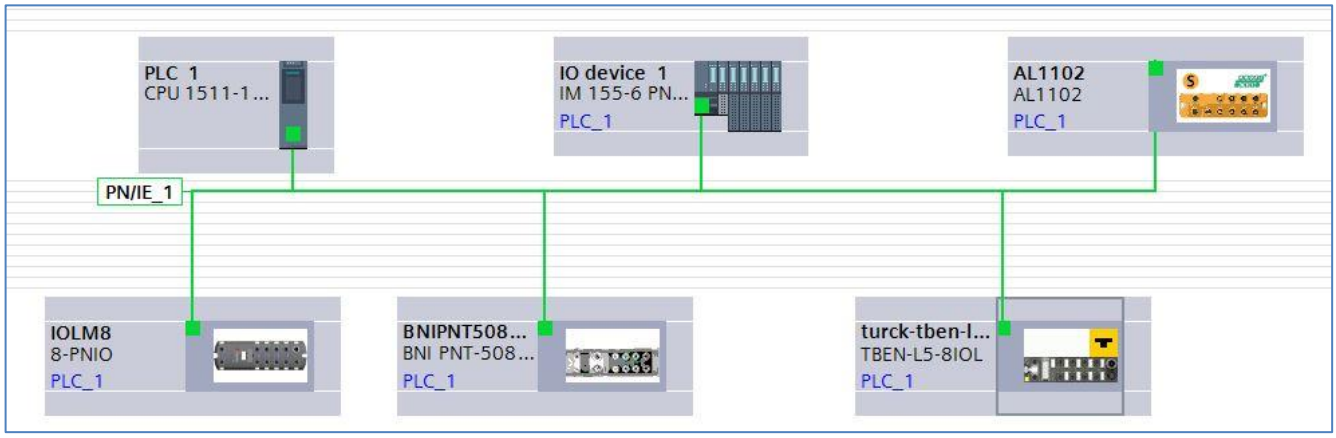
Setup of K50 Pro Indicator with other IO-Link Masters

1. The Banner K50 Indicator library will now be in the Global Library List. Expand the Master copies section. The K50 Pro L2 folder contains elements for both Process Data and Parameter Data connections to a K50 Pro L2 device. As Process Data is the focus of this paper, we will concern ourselves with these two items: Banner_K50_PD and Banner_K50_PDO.
2. Drag Banner_K50_PD to the Program Blocks area under your PLC.
3. Drag the Banner_K50_PDO to the PLC Data Types area under your PLC.



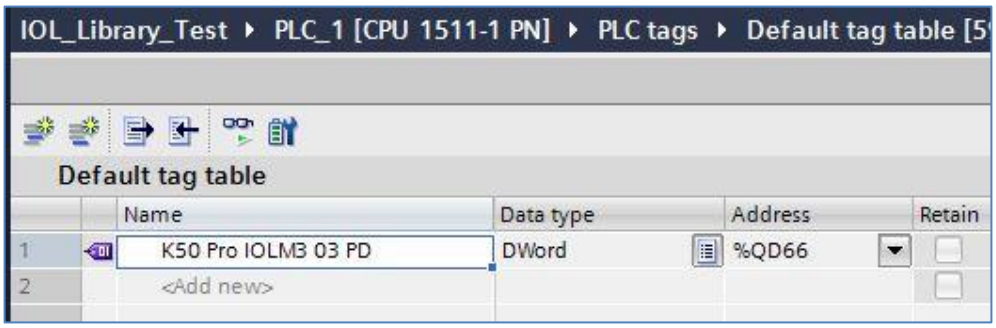


4. Go to Devices and networks to configure the system as necessary. Below is an example of what a configuration might look like. This example shows 5 different IO-Link Masters connected to the same PLC.

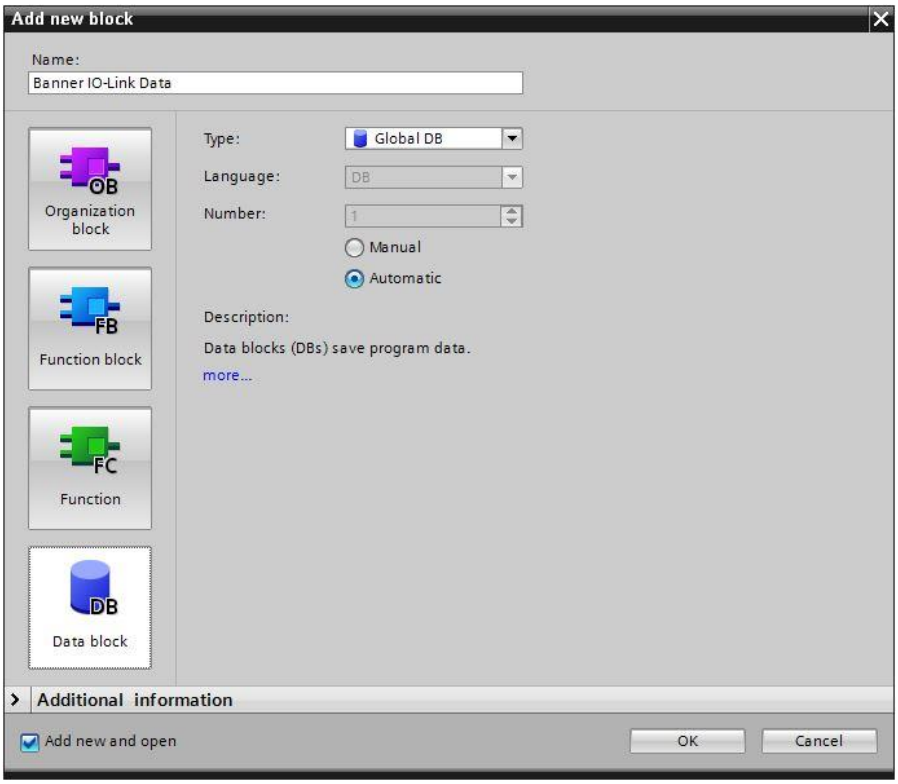


5. Click on the relevant device and configure the IO-Link Master as necessary. Refer to the documentation for the IO-Link Master. Recall that a K50 Pro L2 requires 4 bytes of space for the Process Data.
6. Record the "Q" address where this K50 Pro L2 Process Data is to be stored, as the address will be required in the next step. In this example, 4 bytes of Process Data Out for port 3 on the IO-Link Master will be stored in Q66 through Q69.

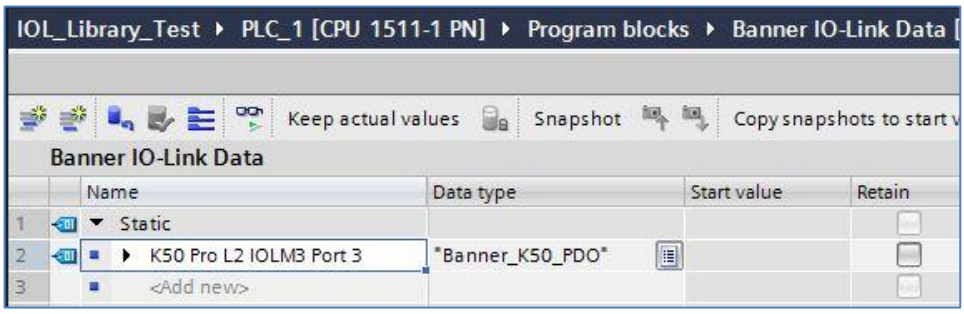
7. Go to PLC Tags. Add a new tag table, then create a new tag to represent the raw Process Data Out to be sent to the IO-Link Master. In this example, Tag table_1 was created, then the tag "K50 Pro IOLM3 03 PD" was created using a Data Type of "DWord". This naming convention calls out the type of sensor 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 IOLM1 or IOLM2, for instance, and other specific sensors may be connected to different port numbers. The "Q" address found in step 9 is tied to this new tag.



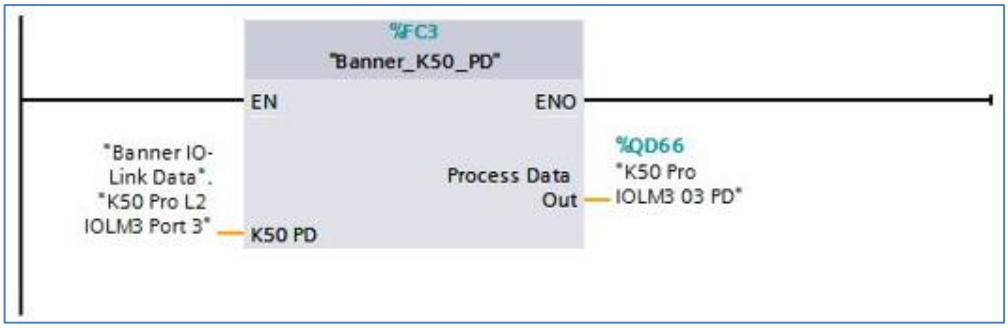
8. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named "Banner IO-Link Data".



9. In the new data block, create a new tag to represent the parsed Process Data In for our K50 Pro L2. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type “Banner_K50_PDO” for the new tag.

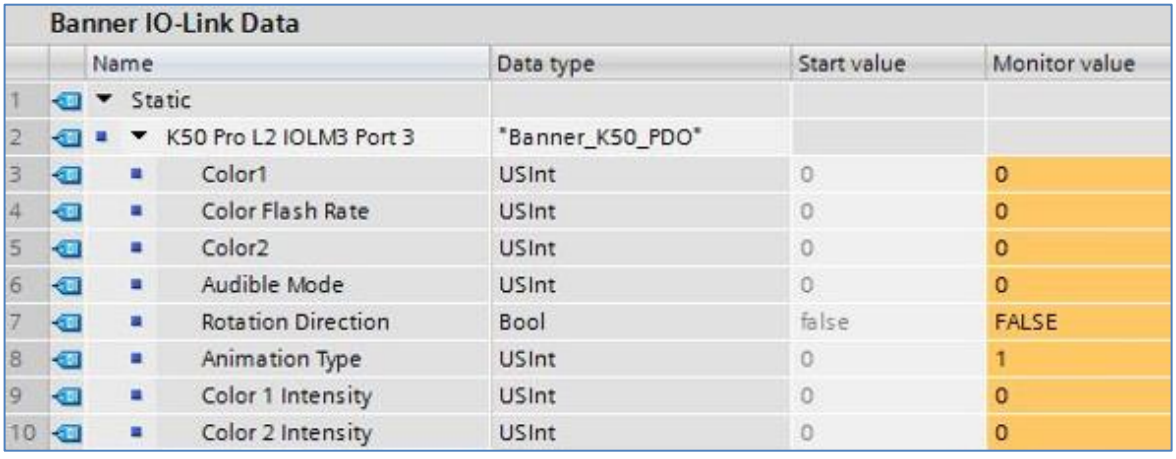


10. Add the “Banner_K50_PD” function to an OB ladder. Link the “Process Data Out” to the raw Process Data variable from step 10. Link “K50 PD” to the parsed Process Data variable from step 12.



11. Process Data setup is complete.

12. Compile and download the configuration to the PLC, then go online. Open the “Banner IO-Link Data” data block and click Monitor all. You should see parsed K50 Pro L2 Process Data In, like that shown below.



Appendix A K50 Pro L2 Process Data

The K50 Pro L2 has 4 bytes of Process Data Out, as shown below.

ProcessDataOut "Process Data Out" id=V_Pd_OutConfigurationT

bit length: 32
data type: 32-bit Record

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	5-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Yellow, 4 = Lime Green, 5 = Spring Green, 6 = Cyan, 7 = Sky Blue, 8 = Blue, 9 = Violet, 10 = Magenta, 11 = Rose, 12 = White, 13 = Custom 1, 14 = Custom 2, 15 = Custom 3, 16 = Custom 4, 17 = Custom 5					Color 1	
2	5	3-bit UInteger	0 = 1.5, 1 = 0.5, 2 = 3.0, 3 = 6.0, 4 = 9.0, 5 = 12.0, 6 = Custom					Color Flash Rate (Hz)	
3	8	5-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Yellow, 4 = Lime Green, 5 = Spring Green, 6 = Cyan, 7 = Sky Blue, 8 = Blue, 9 = Violet, 10 = Magenta, 11 = Rose, 12 = White, 13 = Custom 1, 14 = Custom 2, 15 = Custom 3, 16 = Custom 4, 17 = Custom 5					Color 2	
4	13	2-bit UInteger	0 = Off, 1 = On, 2 = Pulse					Audible Mode	
5	15	Boolean	false = Counter Clockwise, true = Clockwise					Rotation Direction	
6	16	4-bit UInteger	0 = Off, 1 = Steady, 2 = Flash, 3 = Two Color Flash, 4 = Strobe, 5 = Half/Half, 6 = Half/Half Rotate, 7 = Chase, 8 = Demo Mode					Animation Type	
7	20	4-bit UInteger	0 = High, 1 = Low, 2 = Medium, 3 = Custom, 4 = Off					Color 1 Intensity	
8	24	4-bit UInteger	0 = High, 1 = Low, 2 = Medium, 3 = Custom, 4 = Off					Color 2 Intensity	
9	28	4-bit UInteger	0					Reserved	

This Process Data is mapped to a specific group of PROFINET addresses. The 32-bits of Process Data actually encode 8 separate pieces of information, as shown above.

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