

# S22 Indicator Process Data Function

---

March 19<sup>th</sup>, 2026

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 S22 Indicator light via an IO-Link Master from a Siemens PLC. The function covers parsing and display of the S22 Indicator sensor Process Data Out.

## **Components**

Banner S22 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.



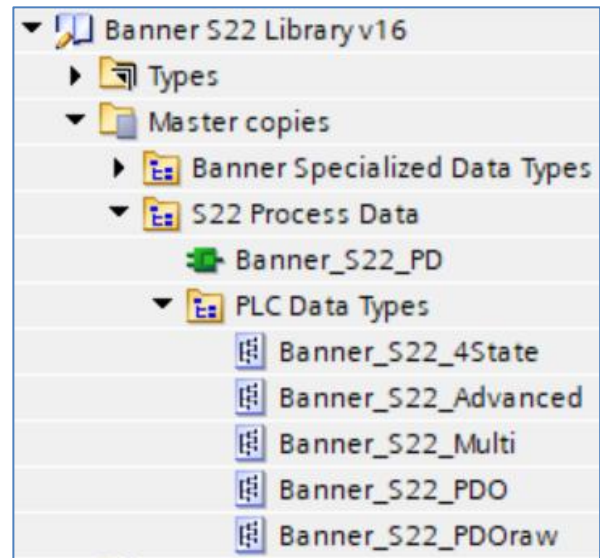
3. Switch the “Files of type” to Compressed libraries. Go to the location of the compressed library.
4. Press the Open button and the library will be uncompressed and opened.
5. The library is now accessible in the Libraries tab in v16 or greater.

**Setup of S22 Indicator 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 8/8 byte is required for K30 Pro Touch. 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 8/ 8 Byte + Status_1	0	2	10...21	1...22	IO-Link In/Out 8/ 8 Byte + Status

4. Drag the necessary tag from Banner\_Specialized\_Data\_Types. The tag used in this example is "Banner\_8out".
5. Drag the necessary files from the S22 Indicator Folder.
  - a. Banner\_S22\_4State, Banner\_S22\_Advanced, Banner\_S22\_Multi, Banner\_S22\_PDO, and Banner\_S22\_PDOraw to the PLC Data Types area.
  - b. Move Banner\_S22\_PD to the Program Blocks area.
6. Go to PLC Tags. Create two tags. In this example, Tag table\_1 was created, then the tag "S22PT IOLM1 01 PDO" was created using a Data Type of "Banner\_8out". 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 "S22PT IOLM1 01 oRaw" and uses the "Q" address found in step 2 (%Q3).



▶ S22PT IOLM1 01 PDO	"Banner_8Out"	%Q1.0
▶ S22PT IOLM1 01 oRaw	"Banner_S22_PDOraw"	%Q3.0

- 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 S22 Indicator. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type "Banner\_S22\_PDO" for the new tag.

▼ S22 IOLM1 01 PD	"Banner_S22_PDO"
■ ▶ 0-MultiColor	"Banner_S22_Multi"
■ ▶ 1-4State	"Banner_S22_4State"
■ ▶ 2-Advanced	"Banner_S22_Advanced"

- 9. Add the "Banner\_S22\_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 called "S22PT IOLM1 01 oRaw" in this example. The "S22" needs to be linked to the variable created in step 8. It was called "S22 IOLM1 01 PD" for this example.

The last variable, "Operational Mode", allow the function to correctly interpret the Process Data Out. In the case of the S22 Indicator, there are four user-selected modes for the Process Data Out. This function needs to know what choice has been made in the S22 Indicator for this Operational Mode variable.

There are two ways to achieve this goal. We can simply type in the correct number for Operational Mode (see Fig. 1), or we can link this S22 Process Data Function to the S22 Parameter Data Function Block (see Fig. 2). See Appendix A for more information about K30 Pro Touch Process Data Out.

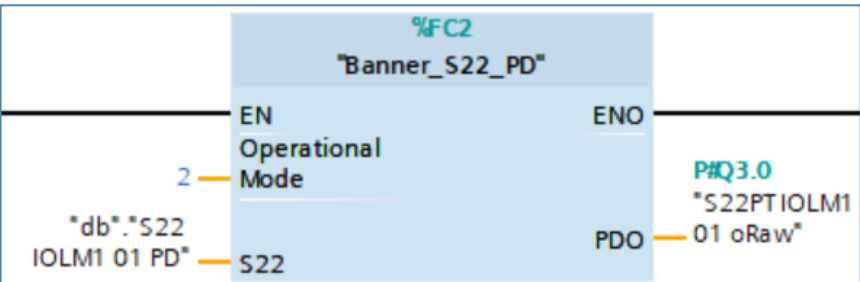


Figure 1: Hand type correct number for Operational Mode

**NOTE:** if you type in the incorrect number (i.e. it does not match the tower light’s current Operational Mode configuration) you will get incorrectly displayed Process Data Out information.

**Operational Mode:** the options here are “0” (MultiColor Mode), “1” (Four State Mode), and “2” (Advanced Mode); where the entire tower light behaves as a level indicator). The default is “2”.

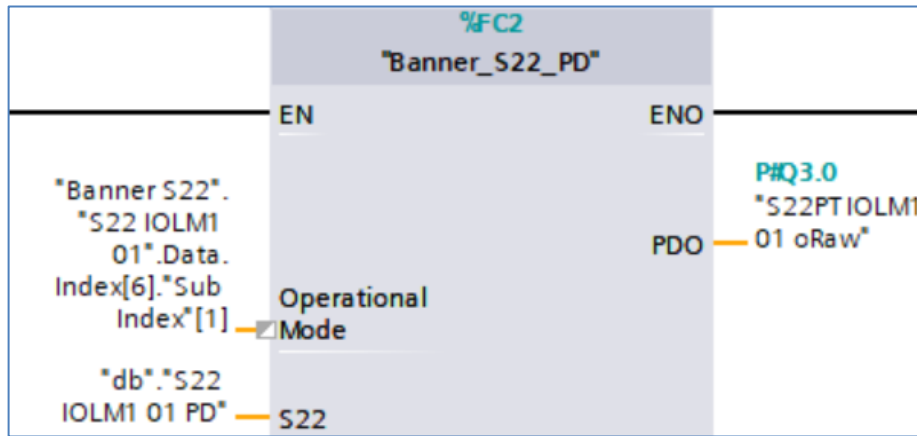
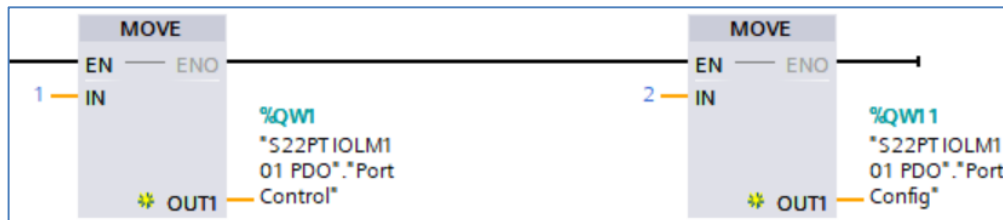


Figure 2: Linking Operational Mode variable to K30 Pro Touch Parameter Data Function Block

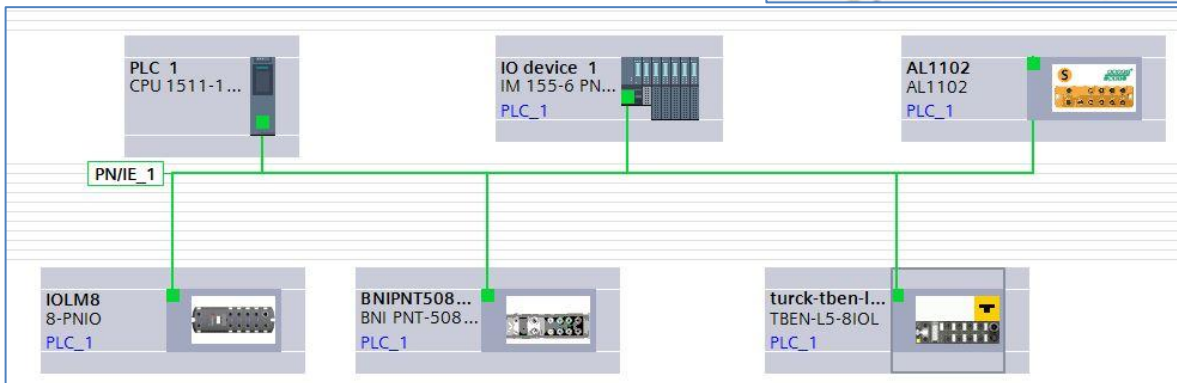
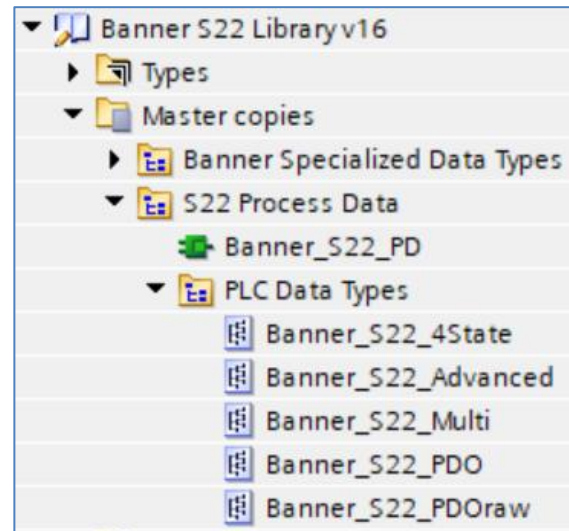
10. The final step is to configure the IO-Link output control. This is done by sending a 1 to Port Control and a 2 to Port Config. Both parameters are part of the tag created in step 6 “S22PT IOLM1 01 PDO”.



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 S22 Indicator can be controlled now.

**Setup of S22 Indicator with other IO-Link Masters**

1. The Banner S22 library will now be in the Global Library List. Expand the Master copies section.
2. Drag Banner\_S22\_PD to the Program Blocks area under your PLC.
3. Drag Banner\_S22\_4State, Banner\_S22\_Advanced, Banner\_S22\_Multi, Banner\_S22\_PDO, and Banner\_S22\_PDORaw 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 S22 Indicator requires 6 bytes of space for the Process Data Out.
6. Record the “Q” addresses where this S22 Indicator Process Data is to be stored, as these addresses will be required in the next step. In this example, 8 bytes of Process Data Out for port 1 on the IO-Link Master will be stored starting at Q68.
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 from the IO-Link Master. In this example, Tag table\_1 was created, then the tag “S22PT IOLM1 01 PDO” was created using a Data Type of “Banner\_S22\_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.

► S22PT IOLM1 01 oRaw	"Banner_S22_PDORaw"	%Q68.0
-----------------------	---------------------	--------

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 Out for our S22 Indicator. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type “Banner\_S22\_PDO” for the new tag.

▼ S22 IOLM1 01 PD	"Banner_S22_PDO"
■ ▶ 0-MultiColor	"Banner_S22_Multi"
■ ▶ 1-4State	"Banner_S22_4State"
■ ▶ 2-Advanced	"Banner_S22_Advanced"

10. Add the “Banner\_S22\_PD” function to an OB ladder. Link the “PDO” to the raw Process Data Out variable from step 7. Link “S22 PD” to the parsed Process Data variable from step 9.

The last variable, “Operational Mode”, allows the function to correctly interpret the Process Data Out. In the case of the S22 Indicator, there are five user-selected modes for the Process Data Out. This function needs to know what choice has been made in the S22 Indicator for this Operational Mode variable.

There are two ways to achieve this goal. We can simply type in the correct number for Operational Mode (see Fig. 1), or we can link this S22 Indicator Process Data Function to the S22 Parameter Data Function Block (see Fig. 2). See Appendix A for more information about S22 Indicator Process Data Out.

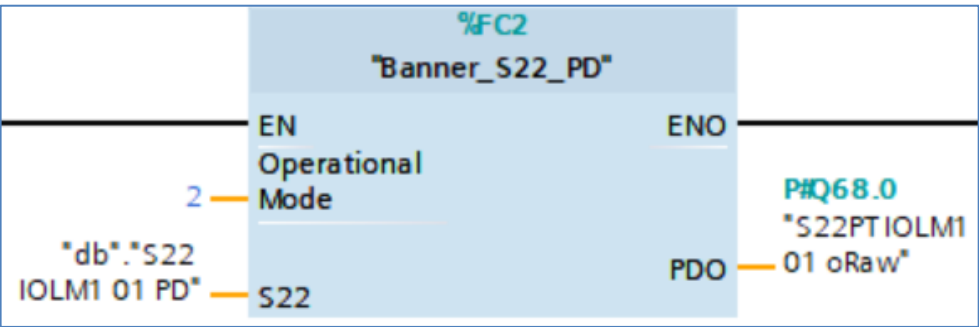


Figure 3: Hand type correct number for Operational Mode

**NOTE:** if you type in the incorrect number (i.e., it does not match the S22’s current Operational Mode configuration) you will get incorrectly displayed Process Data Out information.

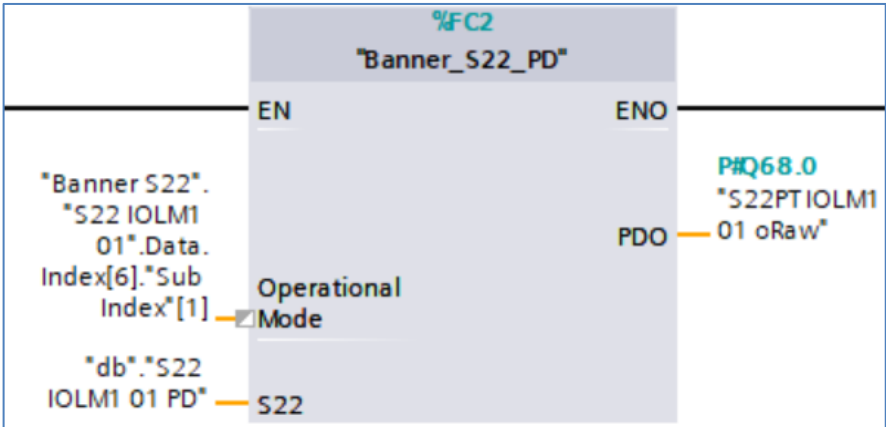


Figure 4: Linking Operational Mode variable to K30 Pro Touch Parameter Data Function Block

- 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.
- 13. Process Data is broken up into three different types. Operational Modes controls which of three types is used to parse the raw byte data.

▼ S22 IOLM1 01 PD	"Banner_S22_PDO"
■ ▶ 0-MultiColor	"Banner_S22_Multi"
■ ▶ 1-4State	"Banner_S22_4State"
■ ▶ 2-Advanced	"Banner_S22_Advanced"



**Appendix A****S22 Indicator Process Data**

The S22 Indicator has 8 bytes of Process Data Out. There are three modes for displaying this data, as shown below. This Process Data is mapped to a specific group of PROFINET addresses. This function intelligently parses this Process Data into its component pieces.

The first is mode 0, “Multicolor”.

ProcessDataOut "Process Data Out" id=V_Pd_OutMulticolor									
bit length: 48 data type: 48-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	40	2-bit UInteger	0 = State1, 1 = State2, 2 = State3, 3 = State4					State	Animation State. Related parameters defined in Four State Full Logic/Multicolor parameter data.

The next mode, “1”, is “Four State Full Logic”.

ProcessDataOut "Process Data Out" id=V_Pd_OutFourStateFullLogic									
bit length: 48 data type: 48-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	40	Boolean	false = Off, true = On					Job Input	Job Input for Four State Full Logic mode.
2	41	Boolean	false = Off, true = On					User Input (Indicator only)	User Input for Four State Full Logic mode (Indicator only).

Mode 2 is “Advanced”.

### ProcessDataOut "Process Data Out" id=V\_Pd\_OutAdvanced

bit length: 48

data type: 48-bit Record

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	40	3-bit UInteger	0 = Off, 1 = Steady, 2 = Flash, 3 = Two Color Flash, 4 = Intensity Sweep, 5 = Color Sweep					Animation Type	The Animation type
2	43	3-bit UInteger	0 = Flash, 1 = Strobe, 2 = Three Pulse, 3 = SOS, 4 = Random					Animation Pattern	The pattern of Animation
3	46	2-bit UInteger	0 = Slow, 1 = Medium, 2 = Fast, 3 = Custom					Animation Speed	The speed of the Animation
4	8	5-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Spring Green, 7 = Cyan, 8 = Sky Blue, 9 = Blue, 10 = Violet, 11 = Magenta, 12 = Rose, 13 = White, 14 = Custom1, 15 = Custom2					Color 1	The main color of the Animation, Custom Colors are defined in Parameter data
5	13	3-bit UInteger	0 = High, 1 = Medium, 2 = Low, 3 = Off, 4 = Custom					Color 1 Intensity	The Intensity of Color 1, Custom Intensity defined in Parameter Data
6	0	5-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Spring Green, 7 = Cyan, 8 = Sky Blue, 9 = Blue, 10 = Violet, 11 = Magenta, 12 = Rose, 13 = White, 14 = Custom1, 15 = Custom2					Color 2	The secondary color of the Animation, Custom Colors are defined in Parameter data
7	5	3-bit UInteger	0 = High, 1 = Medium, 2 = Low, 3 = Off, 4 = Custom					Color 2 Intensity	The Intensity of Color 2, Custom Intensity defined in Parameter Data