

# Add-on Instructions for Configuration, Reading and Writing Data of IQT1... IO-Link RFID read/write heads

## Introduction

Add-on instructions are easy to use function blocks designed for Rockwell Automation Studio 5000 software. The instructions allow all IO-Link RFID parameters to be read or written all at one time. The Process data is also available in either EASY or EXPERT mode. Process data is the Input/Output interface that allows you to read and write to an RFID tag.

## Available Instructions

Instruction	Description
<b>PF_IQT_IOLINK_P2</b>	Reads and Writes parameters to an IQT1 IO-Link head.
<b>PF_IOLink_Param_ReadWrite3</b>	Used by PF_IQT_IOLINK_P2
<b>PF_IQT_IOLINK_ICE2</b>	Reads and Writes parameters from an ICE2 IO-Link master to an IQT1 IO-Link head.
<b>PF_IOLink_Param_ICE2</b>	Used by PF_IQT_IOLINK_ICE2
<b>PF_IQT_IOLINK_EASY</b>	Reads and Writes Data from an IQT1 head to a tag (Set MODE to EASY)
<b>PF_IQT_IOLINK_EXPERT2</b>	Reads and Writes Data from an IQT1 head to a tag (Set Mode to EXPERT)
<b>PF_IQT_IOLINK_EXPERT_TP</b>	Reads and Writes Data from an IQT1 head to a tag (Set Mode to EXPERT), Includes Tag Present bit that tells you if the tag is in front of the read head at any time

## Supported IO-Link Masters

These are the supported IO-Link masters for use with the PF\_IQT\_IOLINK\_P2 Add-on to read and write parameters. Any IO-Link master can be used with the EASY or EXPERT mode Add-ons but make sure that all byte swapping is disabled. Only IO-Link masters starting with ICE2... can use the PF\_IQT\_IOLINK\_ICE2 Add-on instruction.

IO-Link Master	Description
<b>ICE1-8IOL-G60L-V1D</b>	8 port Pepperl+Fuchs IO-Link master in metal housing
<b>ICE1-8IOL-G30L-V1D</b>	8 port Pepperl+Fuchs IO-Link master in compact metal housing
<b>ICE11-8IOL-G60L-V1D</b>	8 port Pepperl+Fuchs IO-Link master in metal housing
<b>ICE2-8IOL-G65L-V1D</b>	8 port Pepperl+Fuchs IO-Link master in plastic housing, Class-A
<b>ICE2-8IOL1-G65L-V1D</b>	8 port Pepperl+Fuchs IO-Link master in plastic housing, Class-A and Class B
<b>ICE2-8IOL-K45S-RJ45</b>	8 port Pepperl+Fuchs Panel Mount IO-Link master, Screw Terminals
<b>ICE2-8IOL-K45P-RJ45</b>	8 port Pepperl+Fuchs Panel Mount IO-Link master, Spring Terminals
<b>1734-4IOL</b>	Allen Bradley Point IO IO-Link Master
<b>1732E-8IOLM12R</b>	Allen Bradley ArmorBlock IO-Link master

## When to use Easy or Expert mode

Easy mode allows up to 28bytes of data to be read or written. Expert mode allows up to 192 bytes of data to be read or written at one time (Check memory availability of the tag).

### Easy mode Features

- 28 bytes maximum read and write
- No Add-On Instructions required
- Continuous reading and writing can be turned on and off on-demand
- The Tag Type, Start Address, Memory Area, and Length for reading and Writing setup beforehand using IO-Link parameters
- Easy programming

### Expert mode Features

- 192 bytes maximum read and write
- Single and Enhanced Read/Write commands possible
- On the fly changes to Start Address, Length, and Memory area possible
- Add-on Instructions required
- Configuration of Tag Type done through the Add-on
- Advanced programming

## Version

All add-ons were created in Studio 5000 V24 and tested in Studio 5000 V32.

# Reading and Writing IO-Link Parameters

This instructions can read and write all IO-Link parameters and Reset all Parameters to default.

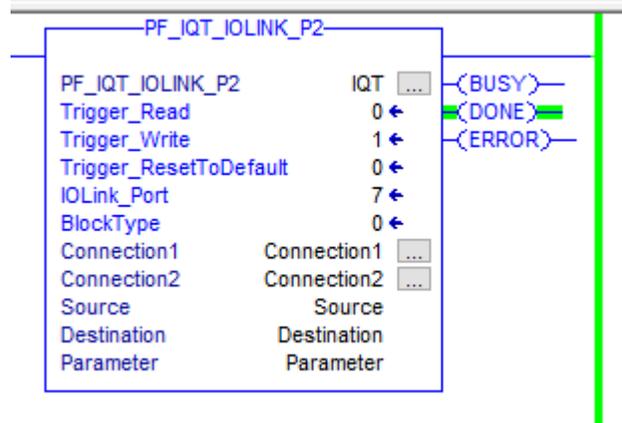


Figure 1: Read/Write parameter add-on for use with all IO-Link masters

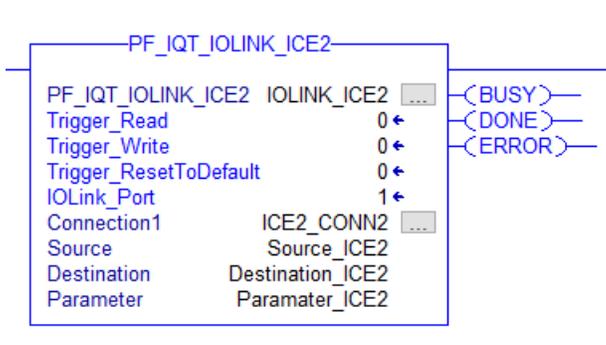


Figure 2: Read/Write parameter add-on for use with the ICE2 IO-Link masters only

When all triggers are low then the “Busy, Done and Error” bits will go low. Only one trigger can be issued at a time. The command is executed on the off-to-on transition and is complete when either a DONE or ERROR bit turns on. When complete the parameter data is displayed in the **Parameter** variable.

-	Parameter	{...}	PF_IQT1
+	Parameter.Identification	{...}	PF_Identification2
+	Parameter.Configuration	{...}	PF_IQTConfiguration
+	Parameter.Diagnostics	{...}	PF_IQTDiagnostics

Suggested order of operation to change parameters.

1. Read Parameters
2. Make changes

### 3. Write Parameters

#### Identification Variables

[-] Parameter	{...}		PF
[-] Parameter.Identification	{...}		PF
+ Parameter.Identification.Vendor_Name	{...}	ASCII	SINT
+ Parameter.Identification.Vendor_Text	{...}	ASCII	SINT
+ Parameter.Identification.Product_Name	{...}	ASCII	SINT
+ Parameter.Identification.Product_ID	{...}	ASCII	SINT
+ Parameter.Identification.Product_Text	{...}	ASCII	SINT
+ Parameter.Identification.Serial_Number	{...}	ASCII	SINT
+ Parameter.Identification.Hardware_Revision	{...}	ASCII	SINT
+ Parameter.Identification.Firmware_Revision	{...}	ASCII	SINT
+ Parameter.Identification.App_Specific_Tag	{...}	ASCII	SINT
+ Parameter.Identification.Minimum_Cycle_Time	40	Decimal	SINT
+ Parameter.Identification.Master_Cycle_Time	47	Decimal	SINT
+ Parameter.Identification.IOLink_Version	16#11	Hex	SINT
+ Parameter.Configuration	{...}		PF
+ Parameter.Diagnostics	{...}		PF

#### Configuration Parameters

[-] Parameter	{...}		PF_IQT1
+ Parameter.Identification	{...}		PF_Identific
[-] Parameter.Configuration	{...}		PF_IQTCon
+ Parameter.Configuration.Mode	16#00	Hex	SINT
+ Parameter.Configuration.Easy_ReadTask_Autostart	16#80	Hex	SINT
+ Parameter.Configuration.Easy_ReadTask_Command	16#80	Hex	SINT
+ Parameter.Configuration.Easy_ReadTask_ByteLength	8	Decimal	SINT
+ Parameter.Configuration.Easy_ReadTask_StartAddress	8	Decimal	INT
+ Parameter.Configuration.Easy_WriteTask_Command	16#00	Hex	SINT
+ Parameter.Configuration.Easy_WriteTask_ByteLength	16	Decimal	SINT
+ Parameter.Configuration.Easy_WriteTask_StartAddress	16	Decimal	INT
+ Parameter.Configuration.Easy_TagType	20	Decimal	SINT
+ Parameter.Configuration.DeviceAccessLocks	0	Decimal	INT
+ Parameter.Diagnostics	{...}		PF_IQTDiag

## Diagnostic Parameters

-	Parameter	{ ... }		PF
+	Parameter.Identification	{ ... }		PF
+	Parameter.Configuration	{ ... }		PF
-	Parameter.Diagnostics	{ ... }		PF
+	Parameter.Diagnostics.Status	0	Decimal	SII
+	Parameter.Diagnostics.Status_Detail	{ ... }	Decimal	SII
+	Parameter.Diagnostics.Indication_Control	0	Decimal	SII

## Trigger Reset to Default

Resets all the sensor parameters to default. Read the parameters after to see the default settings, then make changes as necessary.

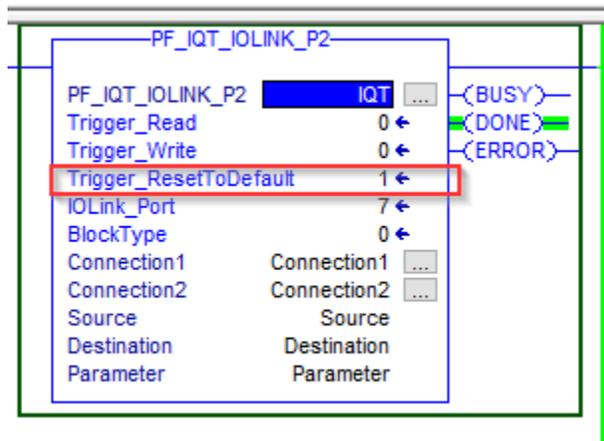


Figure 3: Reset to default for use with all IO-Link masters

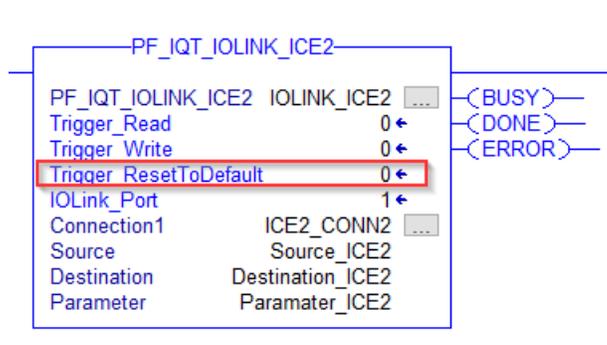


Figure 4: Reset to default for use with ICE2... IO-Link masters

## Port number and module type

The port number is the port that the sensor is connected to.

The BlockType defines what type of IO-Link master is being used. The BlockType parameter is not used for PF\_IQT\_IOLINK\_ICE2 add-on instruction

- BlockType = 0 > Pepperl+Fuchs ICE1 IO-Link Masters
- BlockType = 1 > Allen Bradley IO-Link Masters
- BlockType = 2 > Pepperl+Fuchs ICE2 IO-Link Masters
- BlockType = 11 > Pepperl+Fuchs ICE11 IO-Link Masters

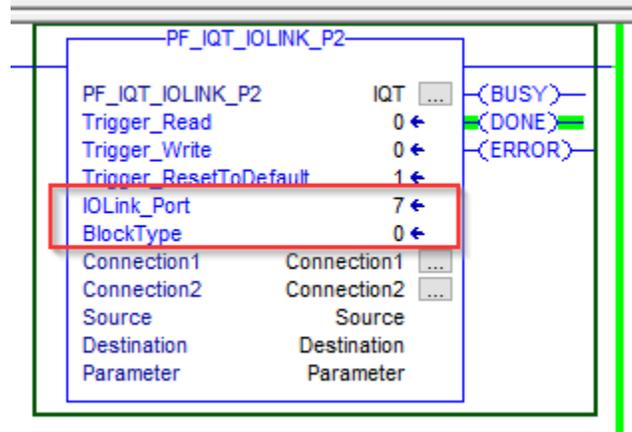


Figure 5: IO-Link port and BlockType configuration for all IO-Link masters

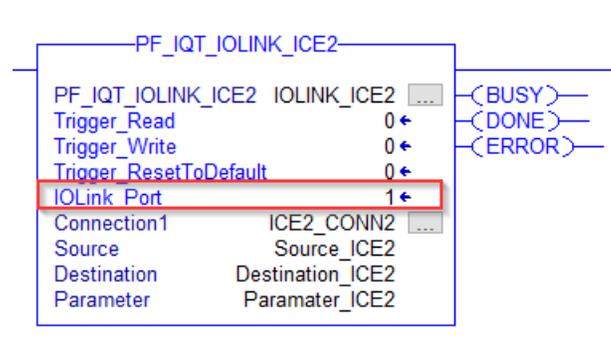


Figure 6: IO-Link port configuration for ICE2... IO-Link masters

## Connections

These are the connection messages used to read and write the parameters. There are only 3 items in each that must be set.

1. Select the correct Service Code
2. Point to the correct Source and Destination variables
3. Select the communication path

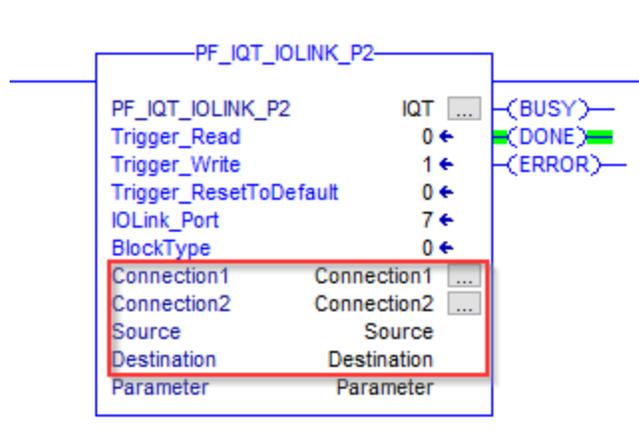


Figure 7: Two connections to setup for all IO-Link masters

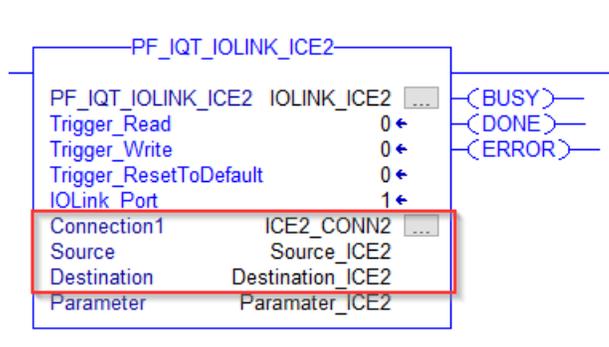
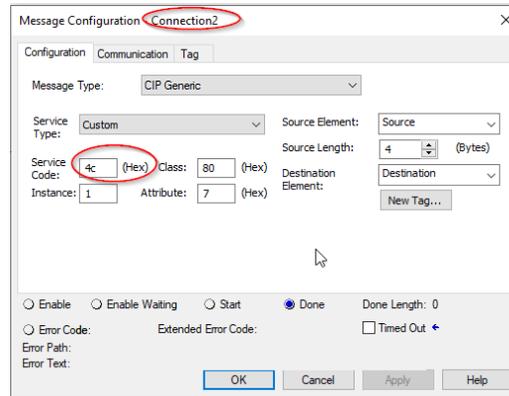
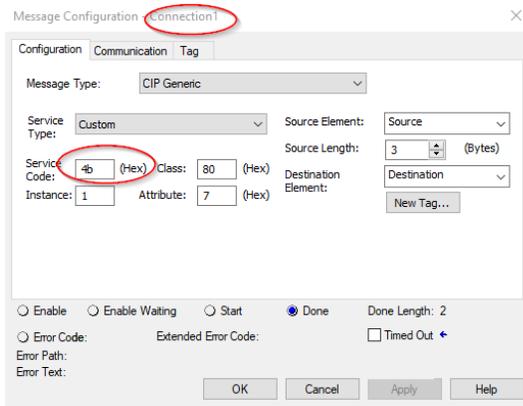


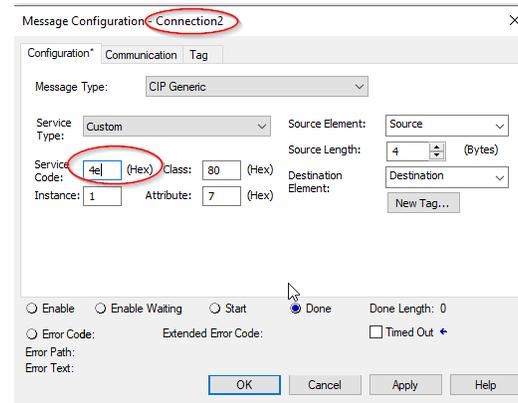
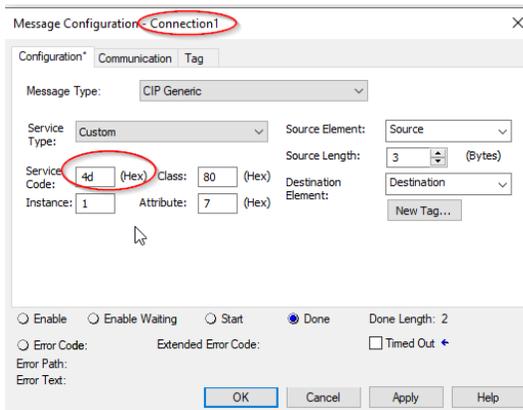
Figure 8: One connection to setup for ICE2... IO-Link masters

## Service Codes

Service Codes 4b, 4c for Block\_Type 0, 2, and 11. See screen shots. The connection1 used for PF\_IQT\_IOLINK\_ICE2 is always 4b

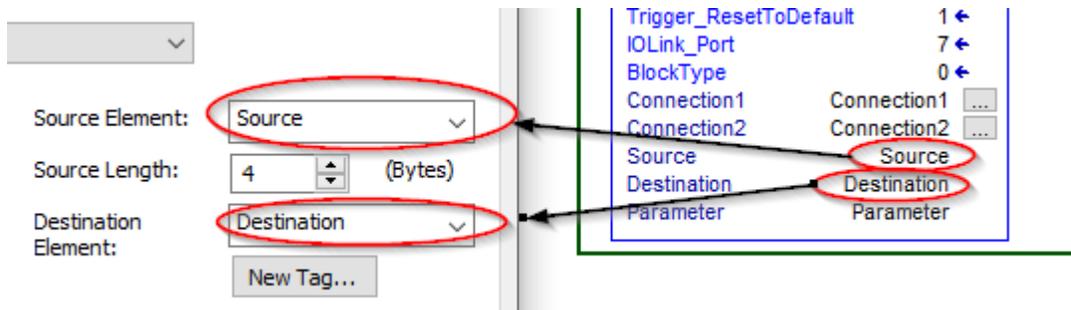


Service Codes 4D, 4E for Block\_Type 1. See screen shots.



## Source and Destination Variables

The Source and Destination on main Add-on screen must match and be used in the Connection Message instructions.



The Communication Path must point to the IO-Link master.

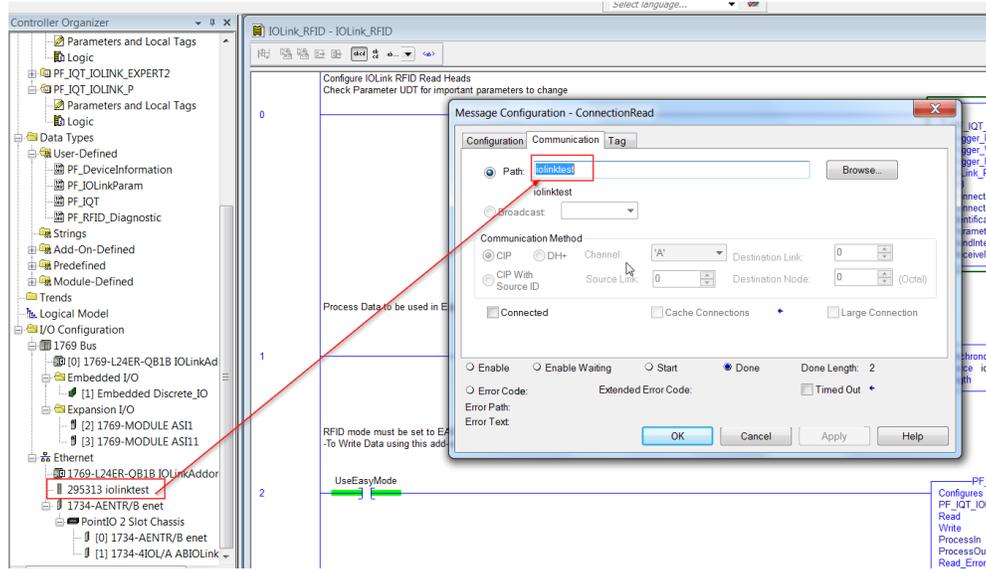


Figure 9: Path to IO-Link master

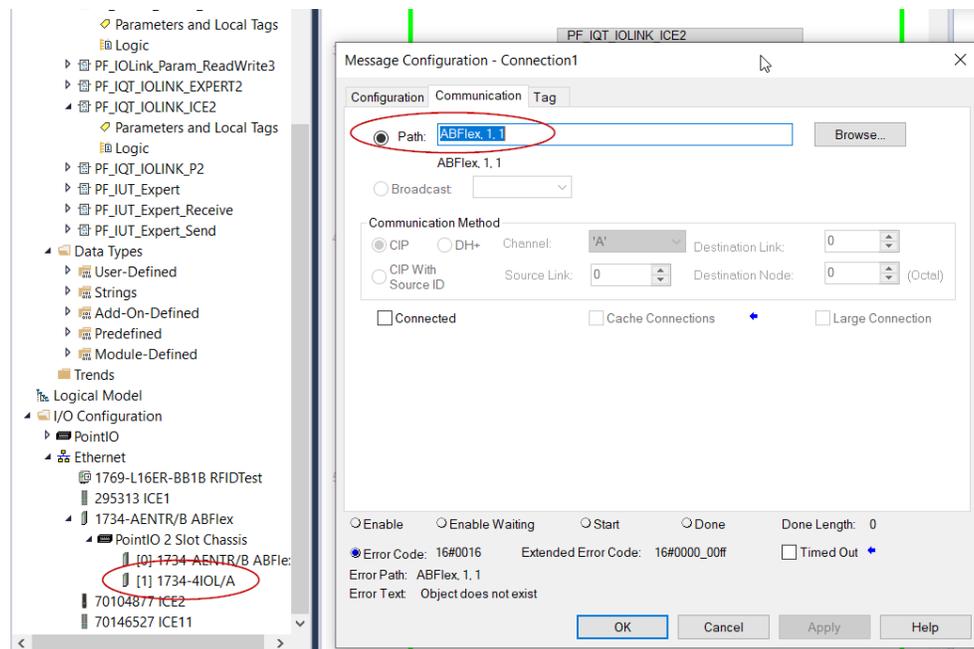


Figure 10: Path to AB PointIO IO-Link master [2,IP\_address,backplane,slot] backplane usually 1]

## Easy and Expert mode Add-on Setup

In Easy mode no Add-on instruction is required at all but if you want to use either the Easy or Expert mode Add-on then you need to do 3 things.

1. Configure the IO-Link port for IO-Link
2. Configure the IO-Link master to read and write 32bytes of IO-Link data
3. Copy the Process Data to variables to be used in the Add-on instruction

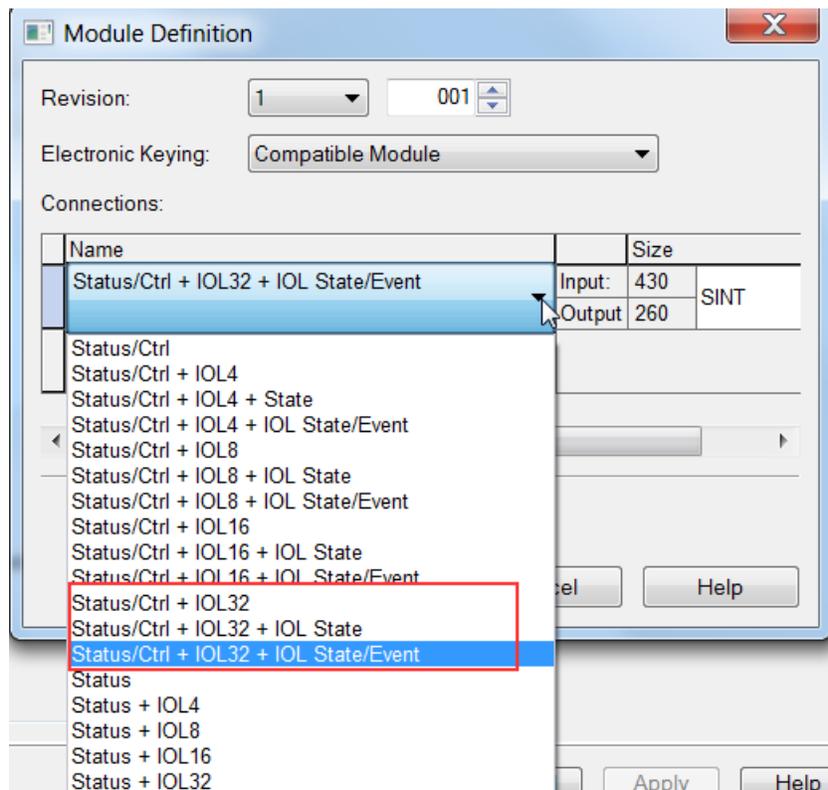
## Configure the IO-Link port for IO-Link

Every IO-Link master is different as far as configuring the ports for IO-Link. The ICE1, and ICE11 IO-Link master's ports are configured using the C file. Set each port to be used for IO-Link to '4'. When the IO-Link port is configured properly the green power LED on the head will blink .5 sec off and 2 seconds on.

iolinktestC.IOL_Port_1_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_2_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_3_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_4_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_5_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_6_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_7_Configuration	4	Decimal	SINT
iolinktestC.IOL_Port_8_Configuration	4	Decimal	SINT

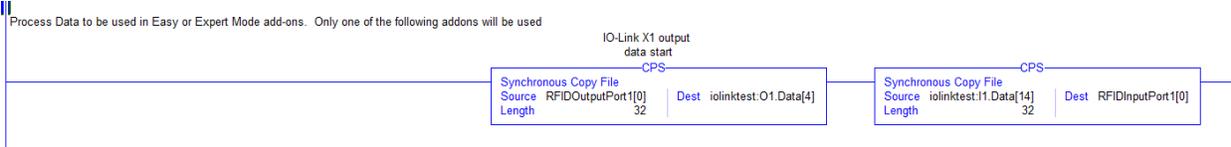
## Configure the IO-Link master to read and write 32bytes of IO-Link data

The read head has 32bytes of Process data so configure the IO-Link master to map 32bytes as well.



## Copy the Process Data to variables to be used in the Add-on instruction

In order to pass the RFID data to the Add-on instruction copy it to and from sint[32] byte variables first. The data can easily be found if the description files are imported first. See the [Pepperl+Fuchs web site](#) and the appropriate IO-Link master then download and import. If you use the ICE11 IO-Link masters then the Add-on instructions provided with these masters will do this for you.

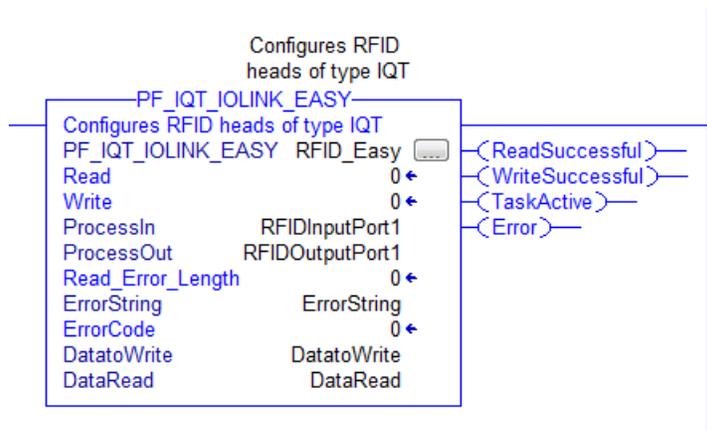


## Using Easy mode

Before using the Easy mode Add-on instruction make sure to parameterize the sensor properly.

IQTPParameter	{ . . . }		PF_IQT	
IQTPParameter.Mode	16#80	Hex	SINT	0=Expert, 16#80=Easy(Default)
IQTPParameter.Easy_ReadTask_Autostart	16#80	Hex	SINT	0=Off, 16#80=Read enabled on power-up(D...
IQTPParameter.Easy_ReadTask_Command	16#00	Hex	SINT	0=User(Default), 16#80=UID
IQTPParameter.Easy_ReadTask_ByteLength	8	Decimal	SINT	Number of Bytes to read (Default=8)
IQTPParameter.Easy_ReadTask_StartAddress	0	Decimal	INT	Byte to Start Reading (Default=0)
IQTPParameter.Easy_WriteTask_Command	16#00	Hex	SINT	0=User (Default)
IQTPParameter.Easy_WriteTask_ByteLength	8	Decimal	SINT	Number of Bytes to Write (Default=8)
IQTPParameter.Easy_WriteTask_StartAddress	0	Decimal	INT	Byte to Start Writing (Default=0)
IQTPParameter.Easy_TagType	20	Decimal	SINT	Tag to read and Write (Default=Auto)
IQTPParameter.DeviceAccessLocks	0	Decimal	INT	0=unlocked(default), 1=lock write parameter...

Mode must be set to 16#80 to use Easy mode. If you want to write data set “Autostart” to 0. Reading and writing the start address and length is defined by the Easy mode parameters so set accordingly.

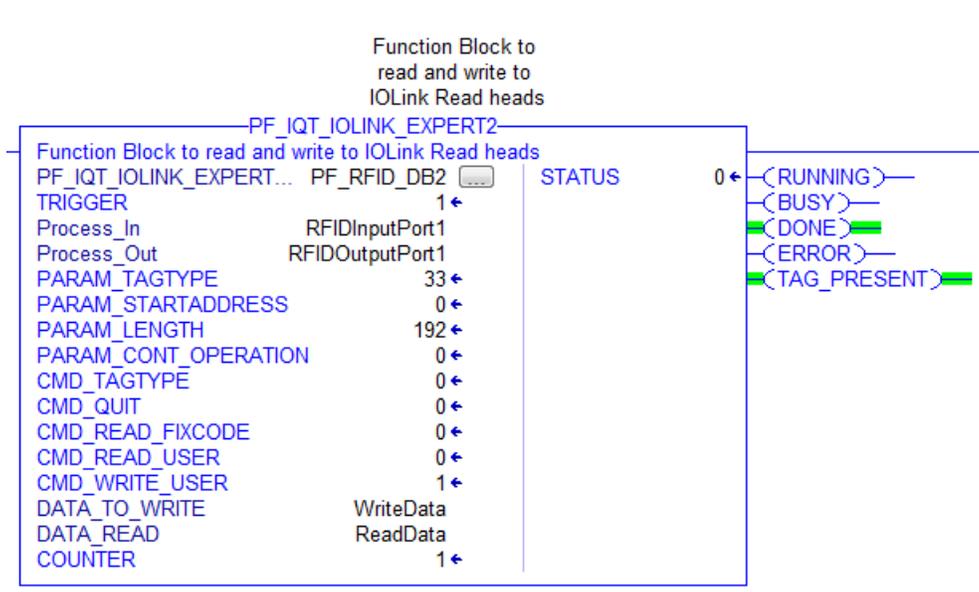


If Autostart=16#80 then the instruction can only be used to read the data and the Read and Write trigger variables will not do anything. If Autostart=0 then the Read and Write variables must be set to 1 in order to activate the command. If a read is running and a tag is in front of the head then the “ReadSuccessful” will be on. When on data is available in the “DataRead” variable. If a write is running and a tag is in front of the head then the “WriteSuccessful” will be on. When a write happens then the data is taken from “DatatoWrite” variable. If a read or write command is running then the “TaskActive” will be on.

## Using Expert mode

Expert mode can be used to read and write up to 192bytes of data to a tag. Make sure to set the Mode to 0. You can only use Expert mode if you are not using Easy mode. Once the instruction is setup follow these steps to run it.

1. Make sure "Trigger" is low and wait until all outputs bits are off
2. Configure the instructions parameters (PARAM\_...) and fill in the "DatatoWrite" variable if you plan to write data to the tag.
3. Flip one of the Command bits (CMD\_...) on. Only one command can be executed at one time.
4. Make the TRIGGER bit "1" to execute the command. Wait for "Running" before evaluating the result of the command.



## Parameters

**PARAM\_TAGTYPE** – Only used with the command CMD\_TAGTYPE.

20 – AutoDetect the ISO 15693 tags

21 – 37 – Specific RFID tag that is being used. Match to model number to the tag type. For example IQC21 = 21...

**PARAM\_STARTADDRESS** – Start Byte where to read or write. 0 is the first byte on the tag. Certain tags require specific byte offsets

IQC22, 22, 23, 24, 31 > 0,4,8,12,16...

IQC33 > 0,8,16,24,32,40...

IQC37 > 0,32,64...

**PARAM\_LENGTH** – Length of bytes to read or write. Make the number of bytes equal to the block size.

IQC22, 22, 23, 24, 31 > 4,8,12,16...

IQC33 > 8,16,24,32,40...

IQC37 > 32,64...

**PARAM\_CONT\_OPERATION** – Make 1 if you want to run an enhanced read or write command and keep the instruction.

### Commands

**CMD\_TAGTYPE** – Send a command to change the tag type

**CMD\_QUIT** – Send a command to quit an enhanced read command

**CMD\_READ\_USER** – Send a command to read User data

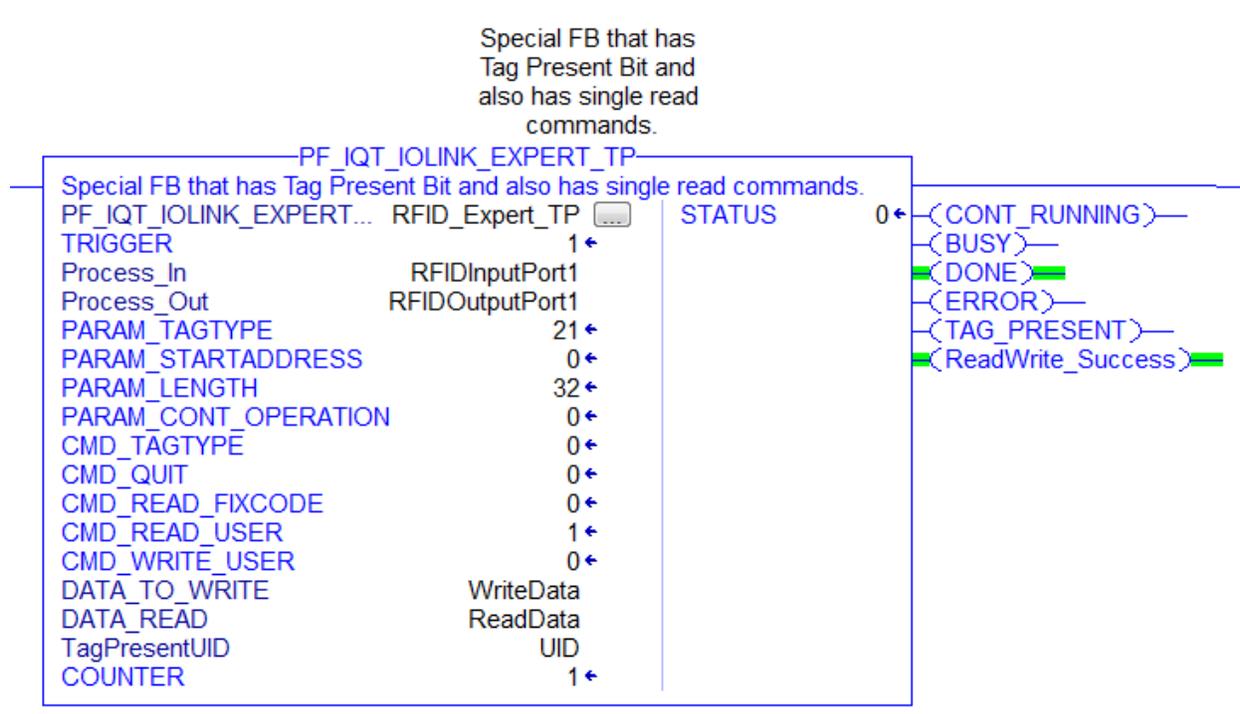
**CMD\_READ\_FIXCODE** – Send a command to read the read only code from the tag

**CMD\_WRITE\_USER** – Send a command to write User data

## Using Expert mode with TP (Tag Present Bit)

This Add-on is exactly the same as the standard Expert mode Add-On instruction except for Three things.

1. The TAG\_PRESENT will always show if there is a tag in front of the read head or not. If on a tag is present and if off a tag is not present
2. The ReadWrite\_Success bit tells you if a tag was read or written successfully
3. The TagPresentUID will tell the value of the fixcode on the tag if the Trigger bit is off.



### Example using Pepperl+Fuchs IO-Link block

See video attached: