IO-Link

Ethernet IP Add-On-Instruction Startup Guide

Turck IOLC AOI v3.0 07/23 Rev.A



[1 General Information 3](#_Toc139276604)

[1.1 About these instructions 3](#_Toc139276605)

[1.2 Explanation of symbols used 3](#_Toc139276606)

[1.3 Feedback about these instructions 3](#_Toc139276607)

[2 Introduction 4](#_Toc139276608)

[3 IO-Link AOI Programming 5](#_Toc139276609)

[3.1 AOI Instances and IOLService Integration 5](#_Toc139276610)

[3.2 IOL Operation Status 6](#_Toc139276611)

[3.3 Specifying IOLink Operations 7](#_Toc139276612)

[3.4 IOL\_ReadString 8](#_Toc139276613)

[3.5 IOL\_WriteString 9](#_Toc139276614)

[3.6 IOL\_ReadData 10](#_Toc139276615)

[3.7 IOL\_WriteData 12](#_Toc139276616)

[3.8 IOL\_Service\_ISDU - Single Ethernet Node Support 13](#_Toc139276617)

[3.9 IOLService\_ISDU – Multiple Ethernet Node Support 15](#_Toc139276618)

[4 Configuring IO-Link using Generic Ethernet/IP CIP Bridge 17](#_Toc139276619)

[4.1 Cycle Time 18](#_Toc139276620)

[4.2 IO Data Mapping 18](#_Toc139276621)

[4.3 IO Data Length and Input Data Mapping 19](#_Toc139276622)

[4.4 IO –Link Input Information 20](#_Toc139276623)

[4.5 IO Data Length and Output Data Mapping 21](#_Toc139276624)

[4.6 IO –Link Input Information 22](#_Toc139276625)

[5 Status Error Code Descriptions 23](#_Toc139276626)

[6 Appendix A – Slot and Port Designations Explained 24](#_Toc139276627)

[6.1 BL20 and BL67 Modular Platforms 24](#_Toc139276628)

[6.2 TBEN-Lx-8IOL 25](#_Toc139276629)

[6.3 TBIP-Lx-FDIO1-2IOL 26](#_Toc139276630)

[6.4 TBEN-S2-4IOL 27](#_Toc139276631)

[6.5 BLCEN-xM12MT-4IOL-xxx 28](#_Toc139276632)

[6.6 BLCEN-xM12MT-4IOL-4IOL 29](#_Toc139276633)

[7 Appendix B – Updating Older Versions of the IO-Link Call AOI 30](#_Toc139276634)

# General Information

## About these instructions

The following user manual describes the setup, functions, and use of the system. It helps you to plan, design, and implement the system for its intended purpose.

Note\*: Please read this manual carefully before using the system. This will prevent the risk of personal injury or damage to property or equipment. Keep this manual safe during the service life of the system. If the system is passed on, be sure to transfer this manual to the new owner as well.

## Explanation of symbols used

Action-related warnings are placed next to potentially dangerous work steps and are marked by graphic symbols. Each warning is initiated by a warning sign and a signal word that expresses the gravity of the danger. The warnings have absolutely to be observed:



DANGER!

DANGER indicates an immediately dangerous situation, with high risk, the death or severe injury, if not avoided.



WARNING!

WARNING indicates a potentially dangerous situation with medium risk, the death or severe injury, if not avoided.





ATTENTION!

ATTENTION indicates a situation that may lead to property damage, if it is not avoid-ed.



NOTE

In NOTES you find tips, recommendations and important information. The notes facilitate work, provide more information on specific actions and help to avoid overtime by not following the correct procedure.



⮚ CALL TO ACTION

This symbol identifies steps that the user has to perform.

🡺 RESULTS OF ACTION

This symbol identifies relevant results of steps

## Feedback about these instructions

We make every effort to ensure that these instructions are as informative and as clear as possible. If you have any suggestions for improving the design or if some information is missing in the document, please send your suggestions to [**techdoc@turck.com**](mailto:techdoc@turck.com).

# Introduction

This startup guide describes the IOLink Add-On-Instructions (AOIs) providing IOLink acyclic read and write capability when used with RSLogix 5000 programmed ControlLogix or CompactLogix (CLX) PLC’s. The AOIs allow simple implementation of acyclic IOLink read and write operations using modular Turck BL20 in-cabinet and BL67 on-machine IO, or BLCompact and TBEN hardened on-machine IO stations.



IOLink provides digital communication to the device and includes cyclic data (the state of the device process variable) and acyclic data (device configuration and status information). Cyclic data transfer can be implemented using either Generic Ethernet/IP CIP Bridge or Generic Ethernet Module. Acyclic data transfer is implemented using the IOLink AOI’s.

IOLink Add-On-Instruction Family

* IOL\_ReadString – read string type data
* IOL\_WriteString – write string type data
* IOL\_ReadData – read 8,16 and 32-bit precision numeric data
* IOL\_WriteData – write 8,16 and 32-bit precision numeric data
* IOLService – execute the Ethernet IP communication required to execute requested transfers

Each IOLink read and write request is instantiated uniquely and the number of reads and writes is limited only by controller memory. Pending requests are executed serially in the order they scanned.

The IOLink AOI’s can be used with the following Turck IOLink masters:

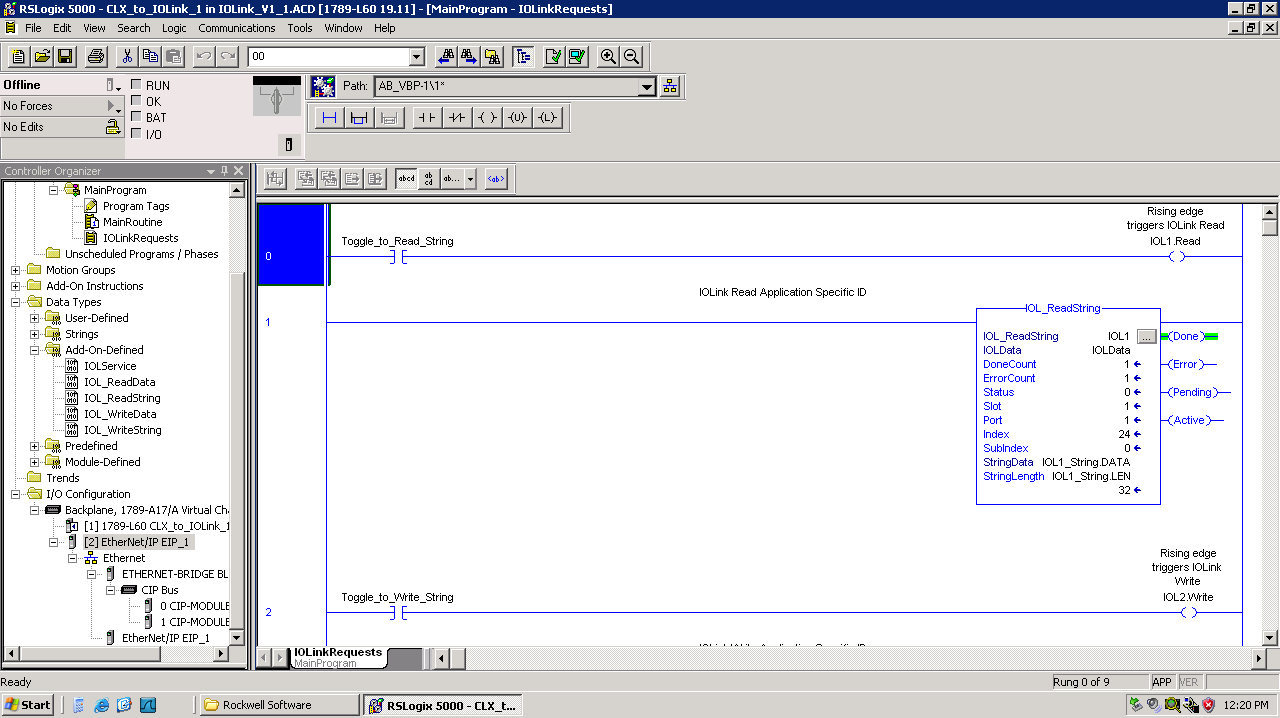
* BL20-E-4IOL(s) used with multiprotocol BL20-E-GW-EN
* BL67-4IOL(s) used with multiprotocol BL67-GW-EN
* TBEN-S2-4IOL
* BLCEN-4M12MT-4IOL
* BLCEN-8M12LT-4IOL-4IOL
* BLCEN-6M12LT-4IOL-2RFID-S
* BLCEN-8M12LT-4IOL-8XSG-P
* BLCEN-8M12LT-4IOL-4AI-VI
* BLCEN-8M12LT-4IOL-4AI4AO-VI
* TBIP-L4-FDIO1-2IOL
* TBEN-L5-8IOL
* TBEN-L4-8IOL

IOLink applications range from simple discrete IO expansion to sophisticated continuously monitored sensor applications. For a simple IO expansion application the IOLink AOI’s are not required and all configuration occurs in the PLC controller tags. As application grow in sophistication the AOI’s can be added to a program at any time.

# IO-Link AOI Programming

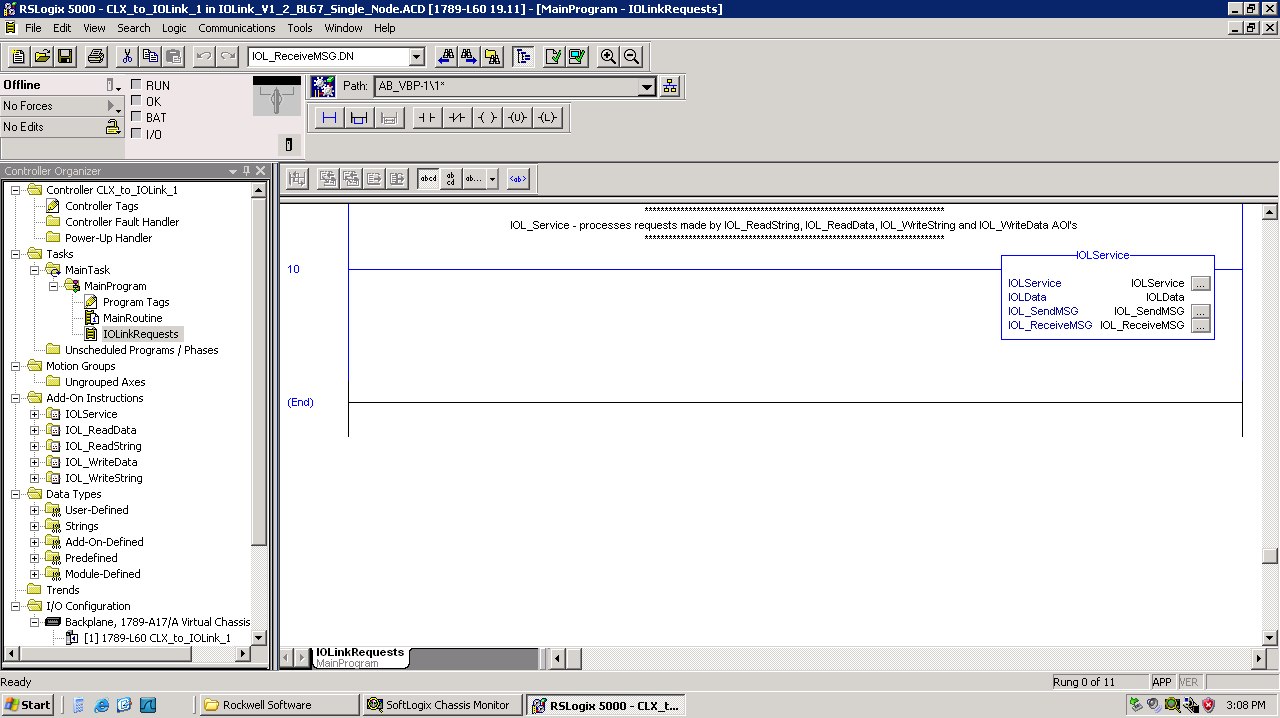
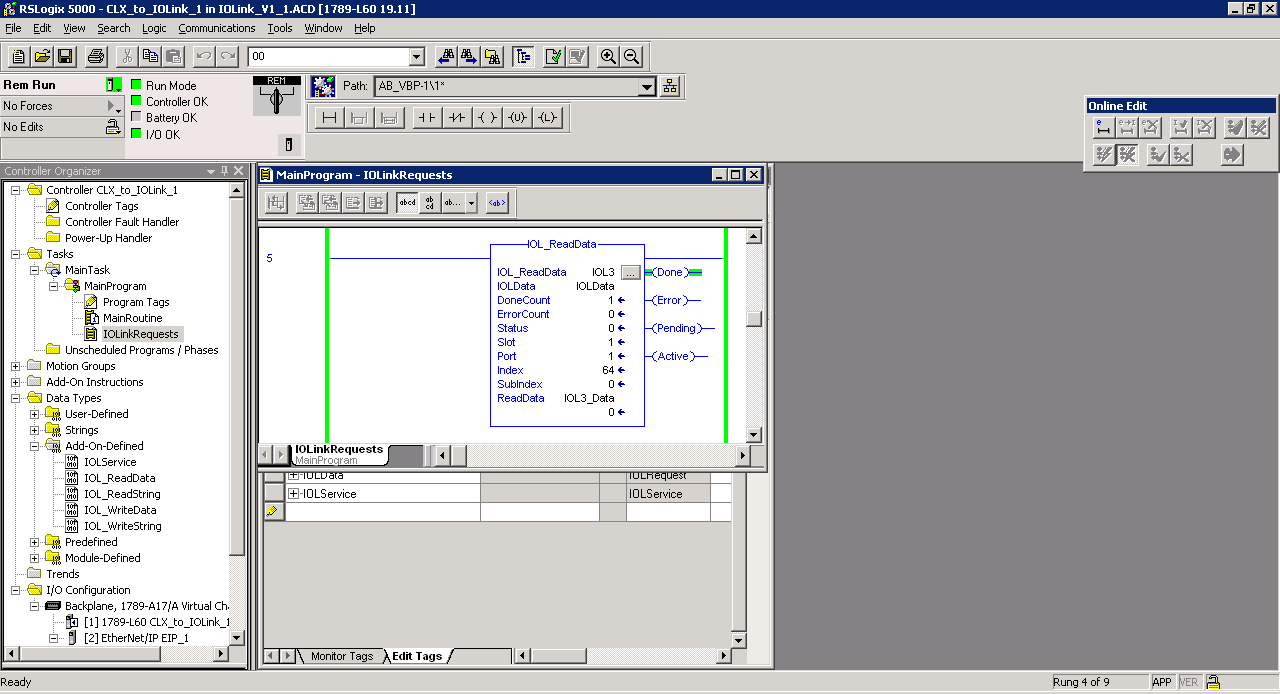
## AOI Instances and IOLService Integration

* Each IOLink transfer request must have a unique instance tag
* All IOLink transfer requests must use the same IOLData tag to share data with IOLService.



Each IOL AOI must have a unique instance tag

All IOL AOI’s must use the same IOLData tag



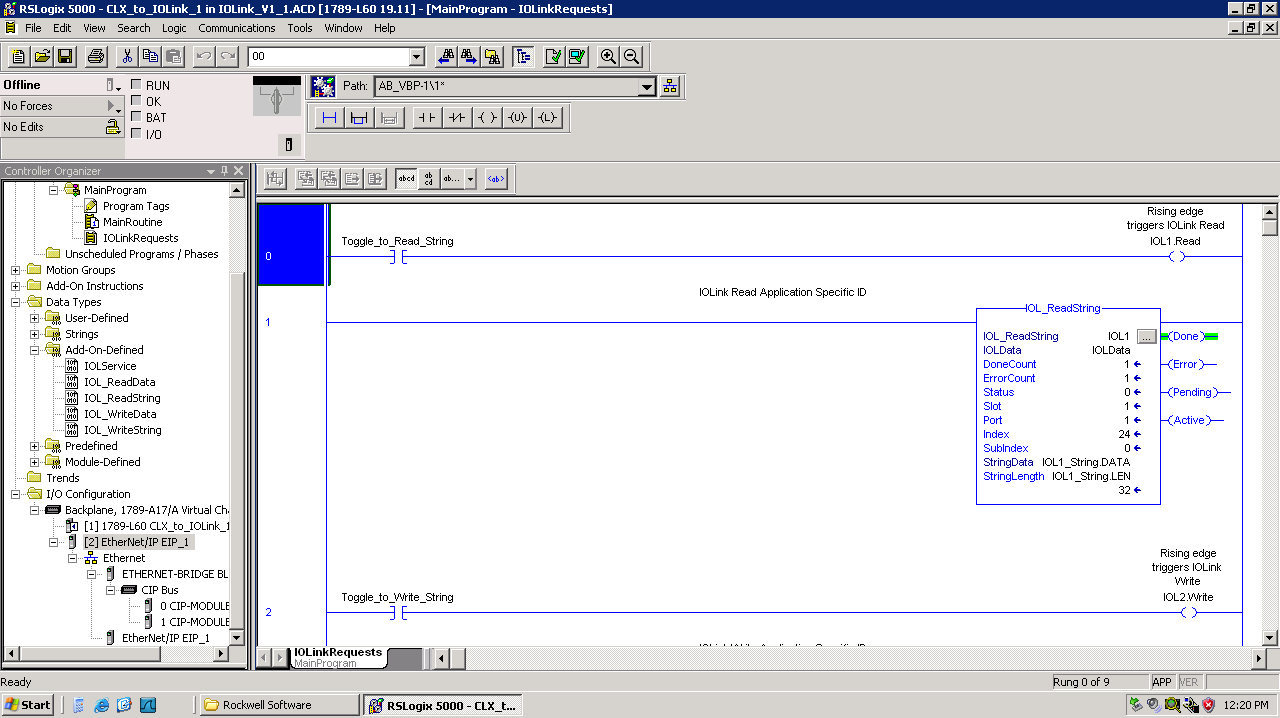
Each IOL AOI must have a unique instance tag

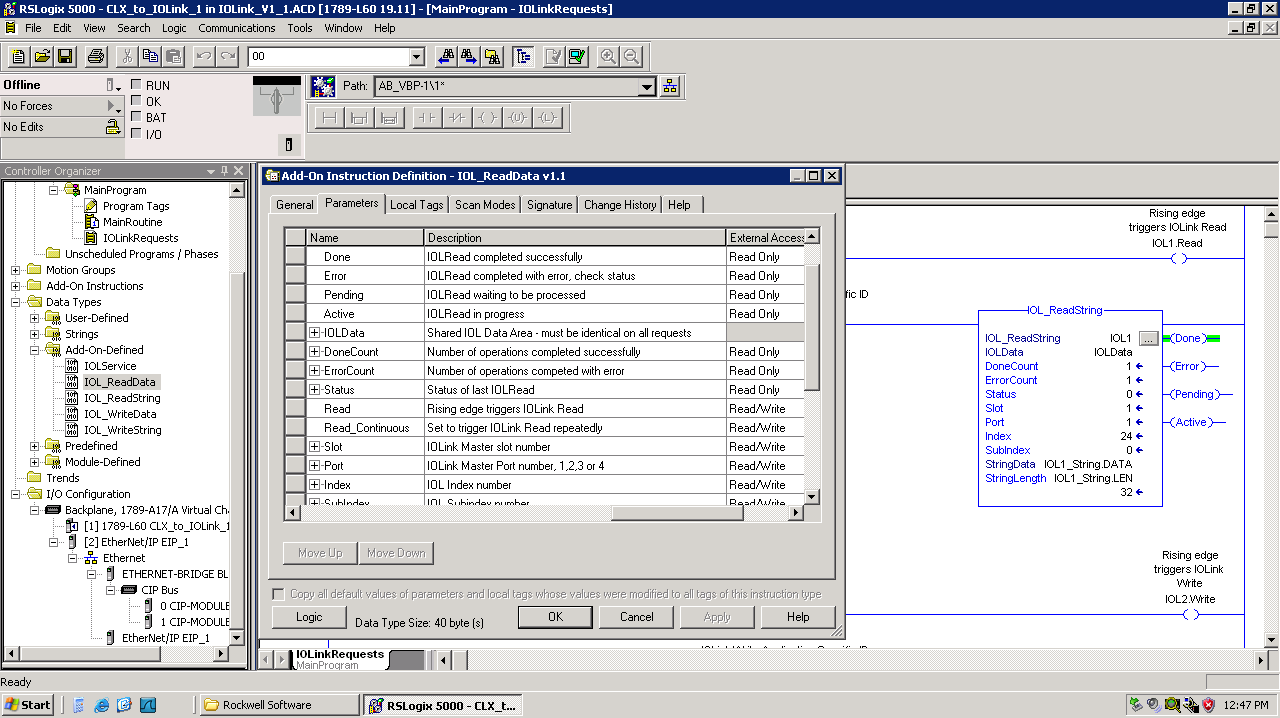
IOLService processes all IOL Read and Write requests

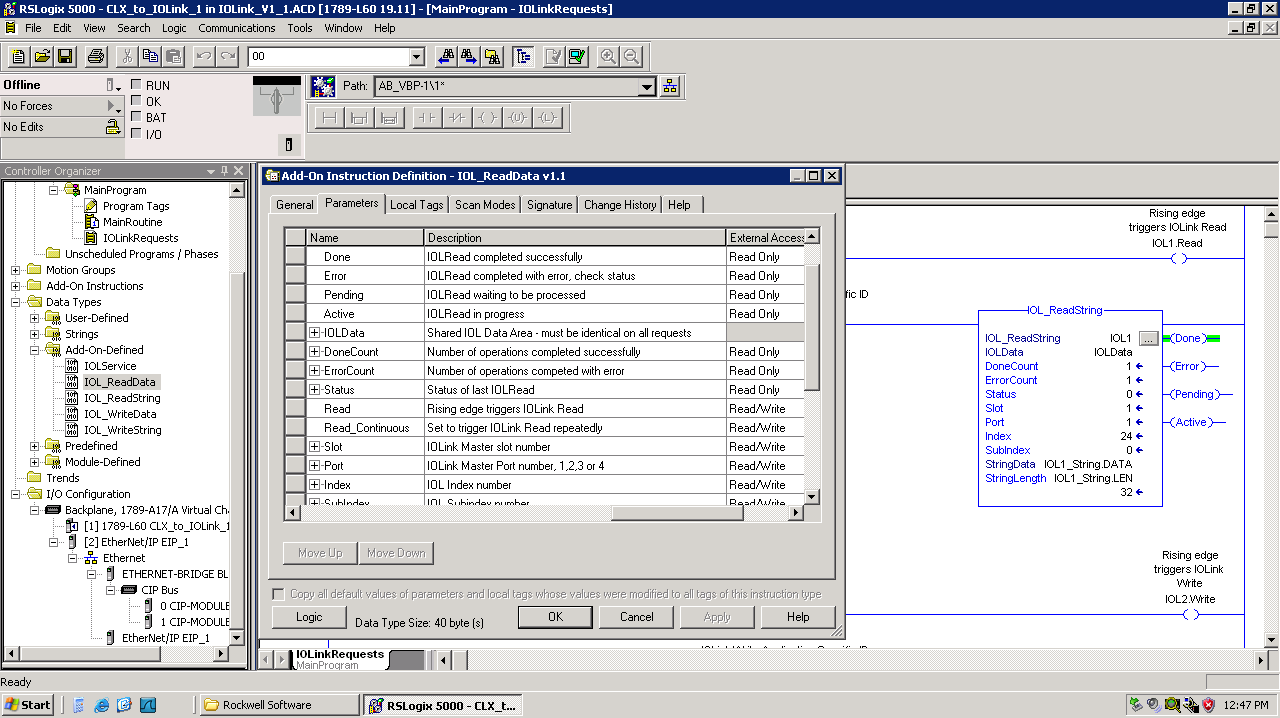
The IOLService AOI appears a single time in the application program and processes all IOLink read and write requests. The number of read and write requests in an application is limited only by controller memory. Multiple read and write requests can be triggered simultaneously and will be processed in the order they appear in the program.

## IOL Operation Status

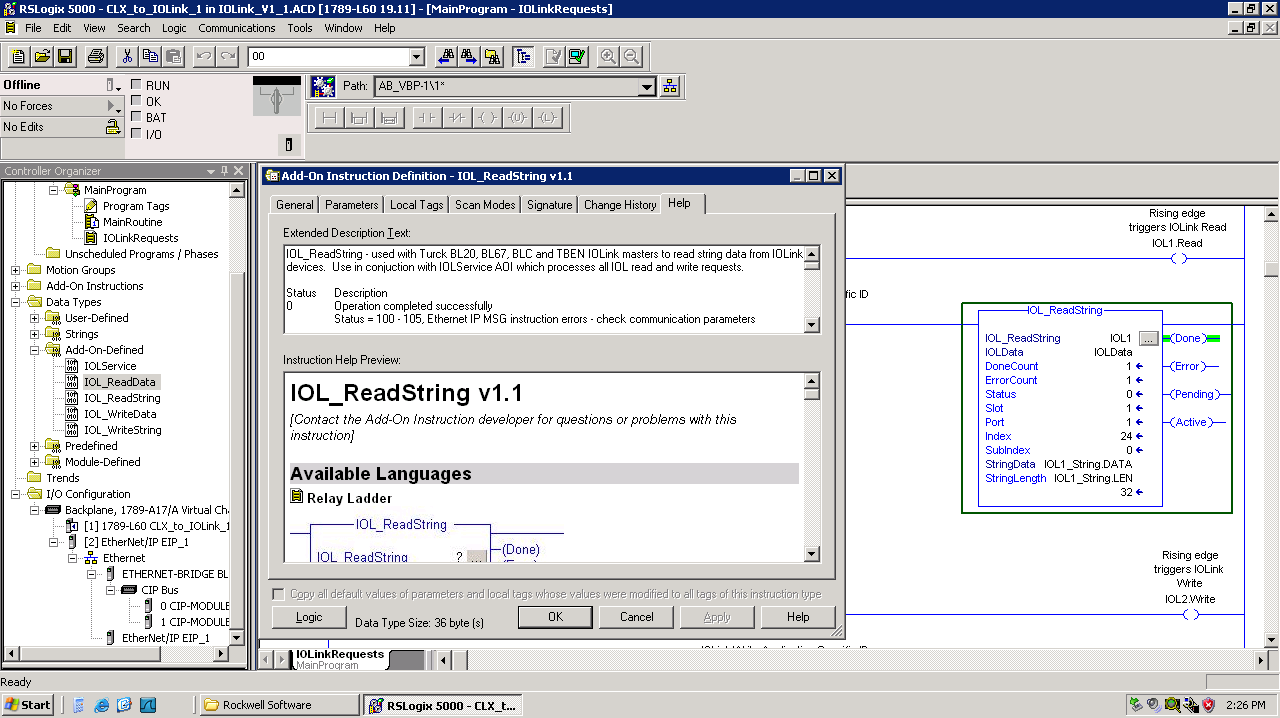
Current operation status is indicated by Done, Error, Pending and Active coils to the right of the AOI. Cumulative DoneCount and ErrorCount and last operation Status are presented on the AOI template.





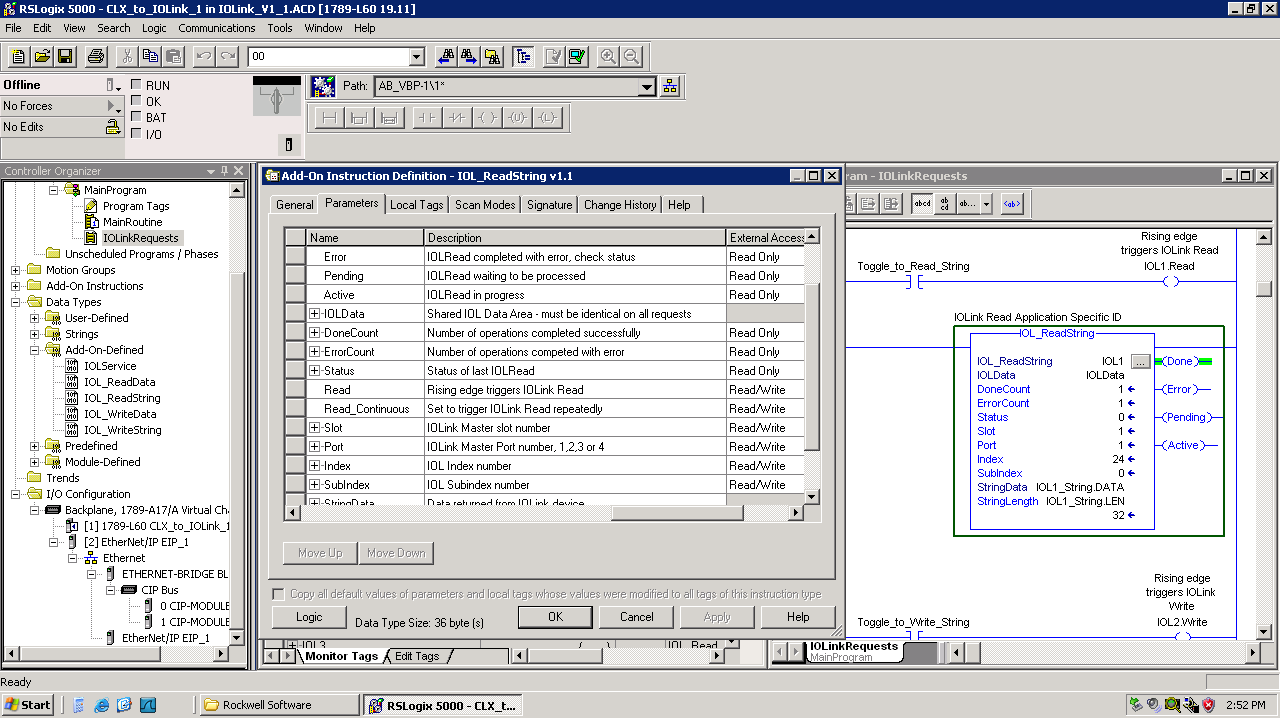
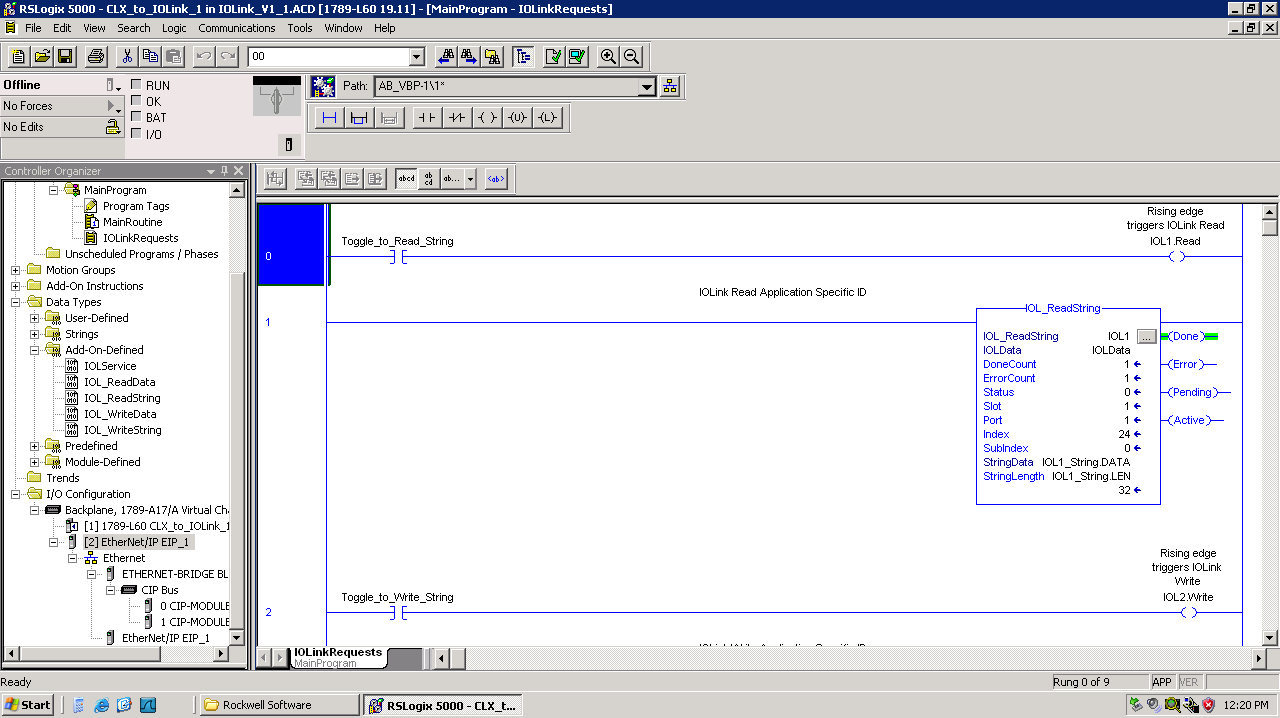


Select the AOI template and press F1 to access help which contains the description of the values presented in Status. Scroll down to view a description of the Status value when an error occurs.



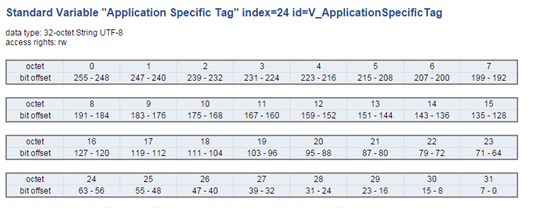
## Specifying IOLink Operations

Slot and Port specify where the IOLink device is connected. Index and Subindex specify which parameter in the IOLink device is to be read or written. When using BL20 or BL67 slot specifies the position in the rack. Using BLCEN slot is set to 1 or 2 specifying the upper or lower position in the block. Set slot to 1 for TBEN-S2-4IOL



The available parameters for each IOLink device varies and the device data dictionary is usually provide in the manufacturer’s datasheet, or an html file or an IODD (IO Link data definition file).

Shown below is the Turck-TBIL-M1-16DI-20130704-IODD1.1-en.html after dropping it in Google Chrome.

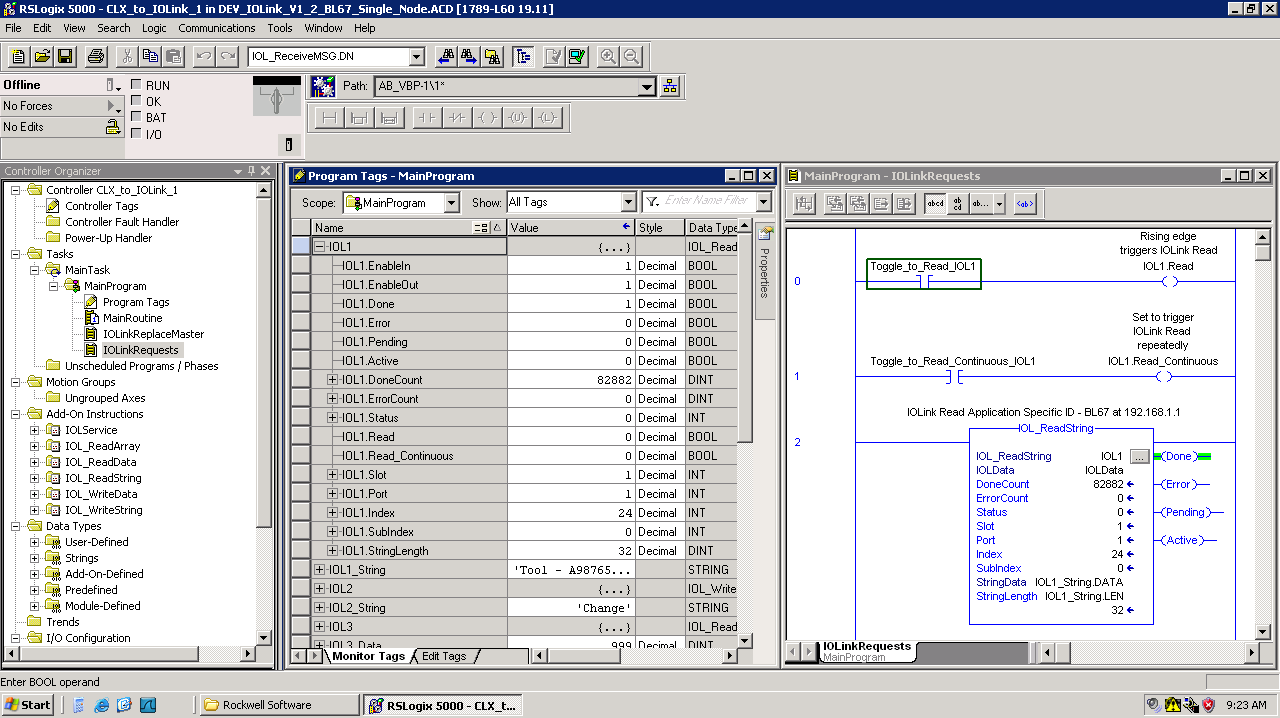


As shown, the Application Specific Tag can be read and written and is a 32-character string. This is often used for Tooling Identification when using the Turck TBIL IOLink hubs.

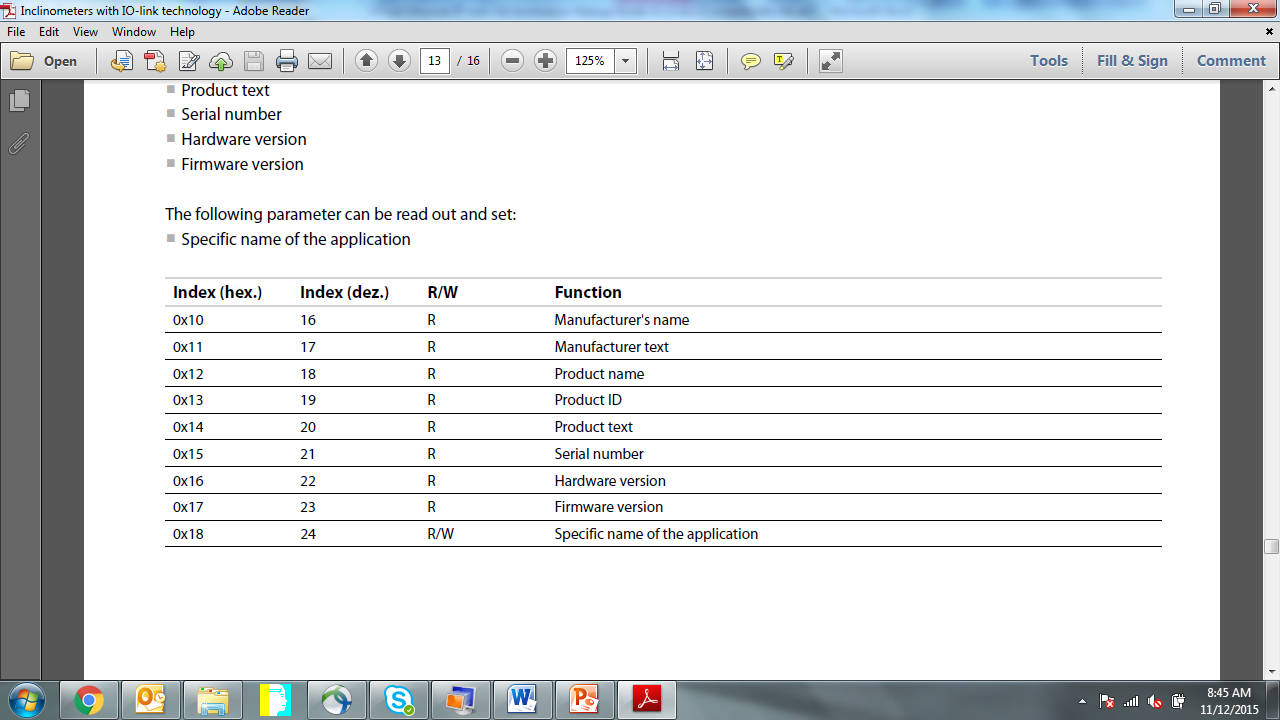
## IOL\_ReadString

Use IOL\_ReadString to read ASCII strings up to 82 characters long. As shown below, IOL\_ReadString instance IOL1 reads the IOLink Device Application Specific ID found at Index 24 Subindex 0. An off-to-on transition of IOL1.Read will trigger the operation a single time. Setting IOL1.Read\_Continuous to 1 will trigger repeated reads.

The target variable for the read operation must be a standard RSL5000 string. The array of characters (DATA) and string length (LEN) must be specified separately as shown. The length of the string returned by the IOLink Device is stored by the AOI in the variable specified at StringLength.



The table below shows a list of the string type data available from most IOLink devices:

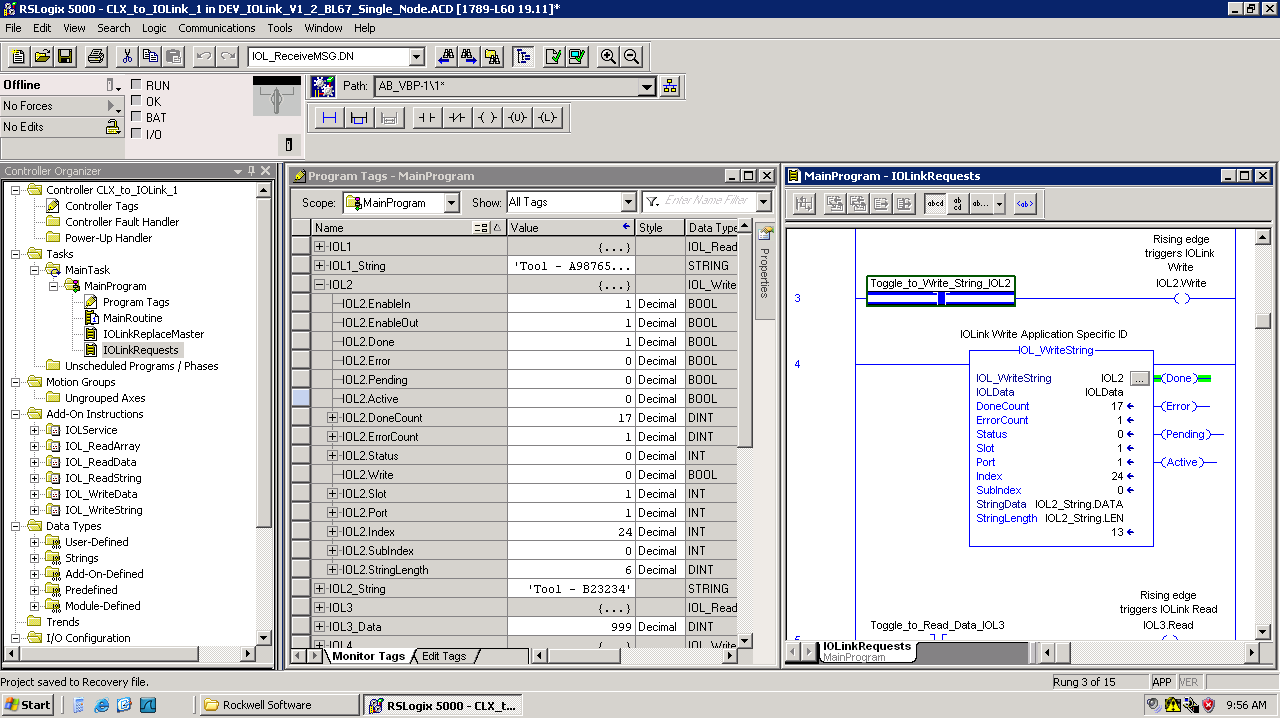


The Application Specific ID at Index 24 Subindex 0 is particularly useful for tooling identification when using the Turck TBIL-M1-16DXP IOLink hub.

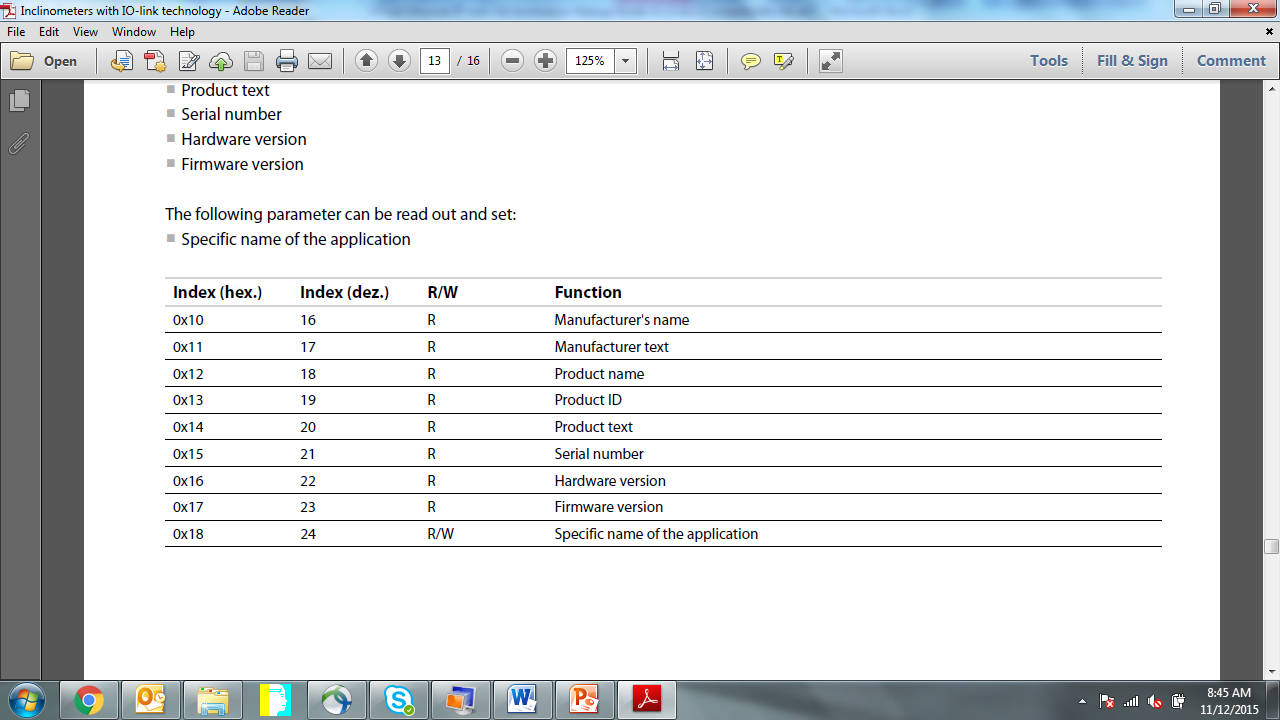
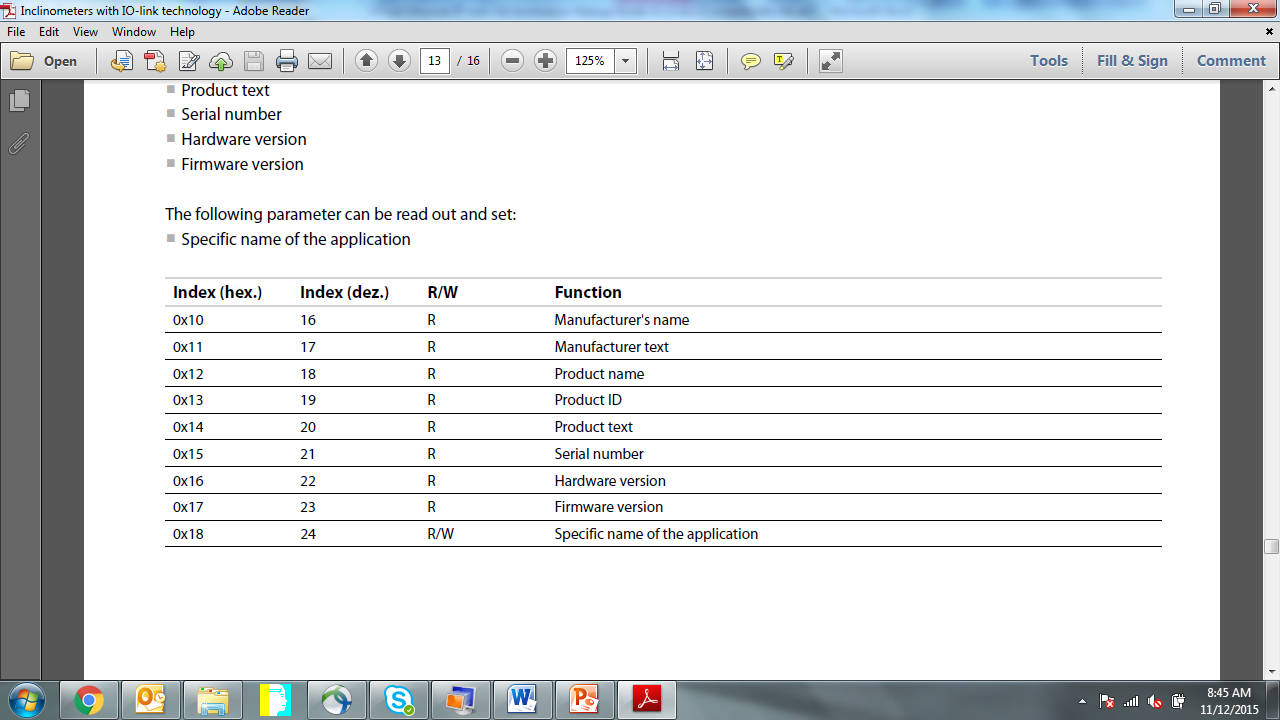
## IOL\_WriteString

Use IOL\_WriteString to write ASCII strings up to 82 characters long. As shown below, IOL\_WriteString instance IOL2 writes the IOLink Device Application Specific ID found at Index 24 Subindex 0. An off-to-on transition of IOL2.Write will trigger the operation a single time.

The source variable for the write operation must be a standard RSL5000 string. The array of characters (DATA) and string length (LEN) must be specified separately as shown. The number of characters specified by StringLength are written to the IOLink device.

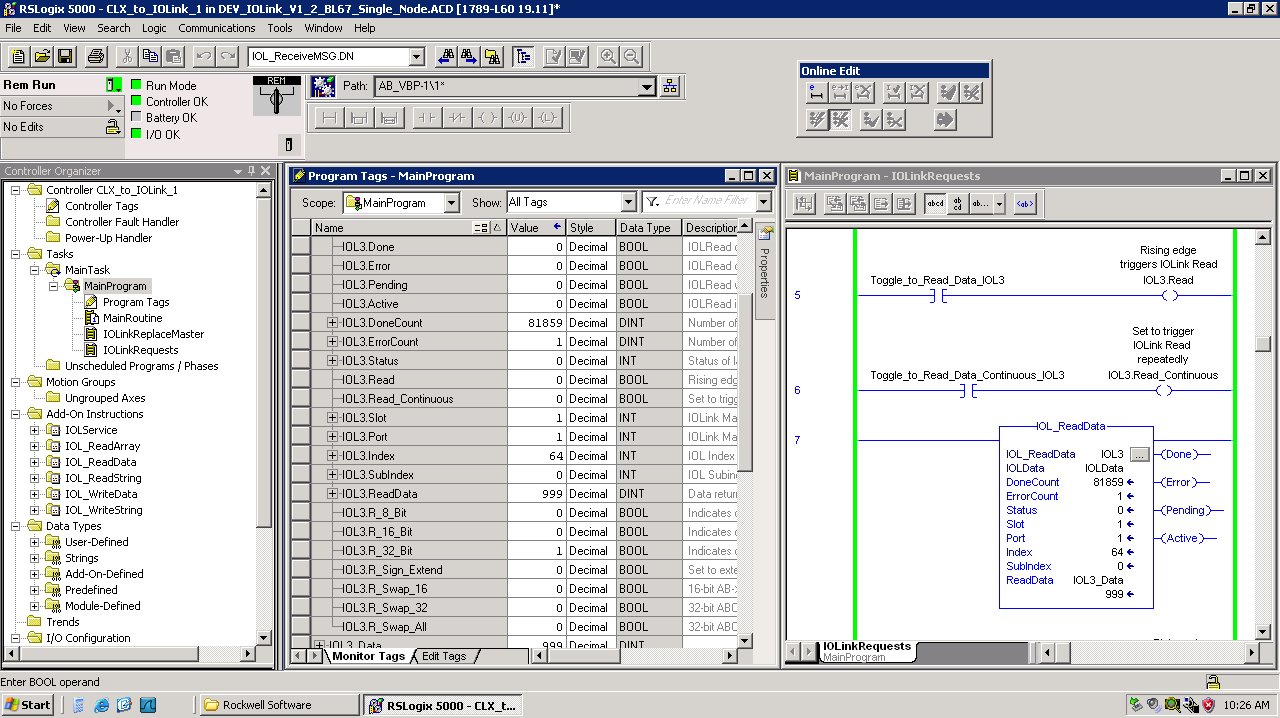


As shown above, the value specified by IOL2\_String, “Tool – B23234” will be written to the IOLink device Application Specific ID at Index 24 Subindex 0.

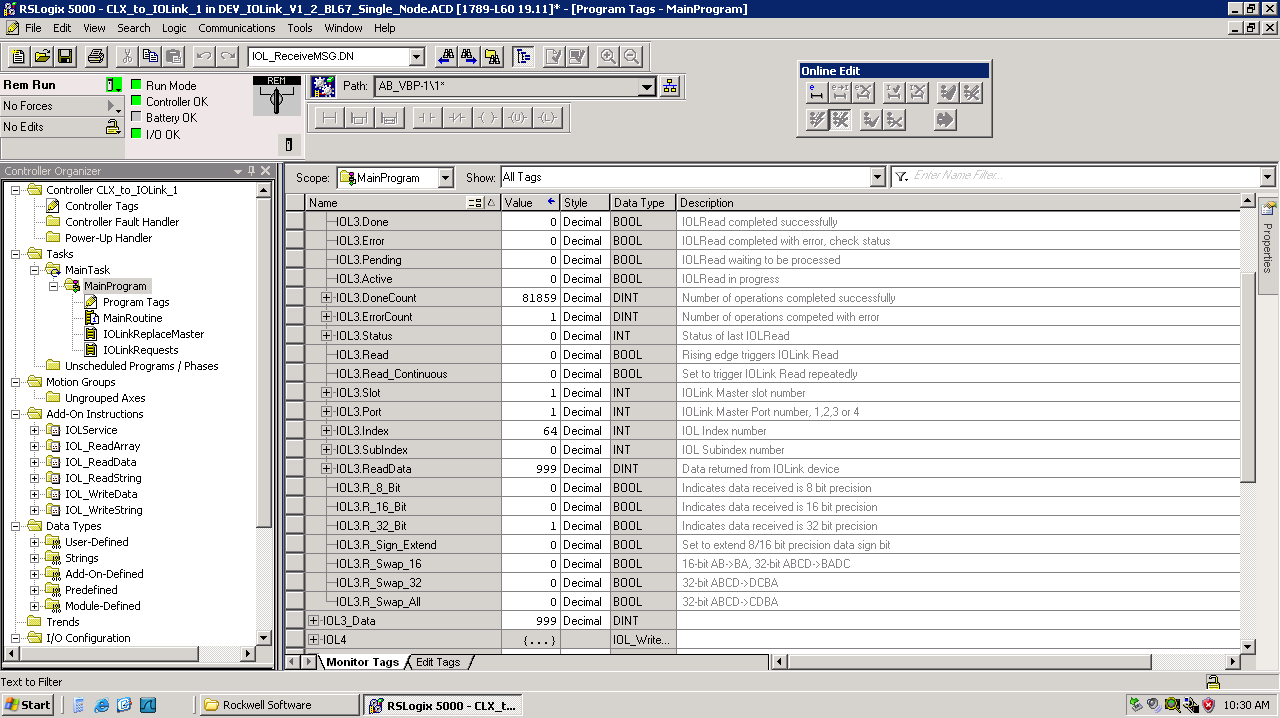


## IOL\_ReadData

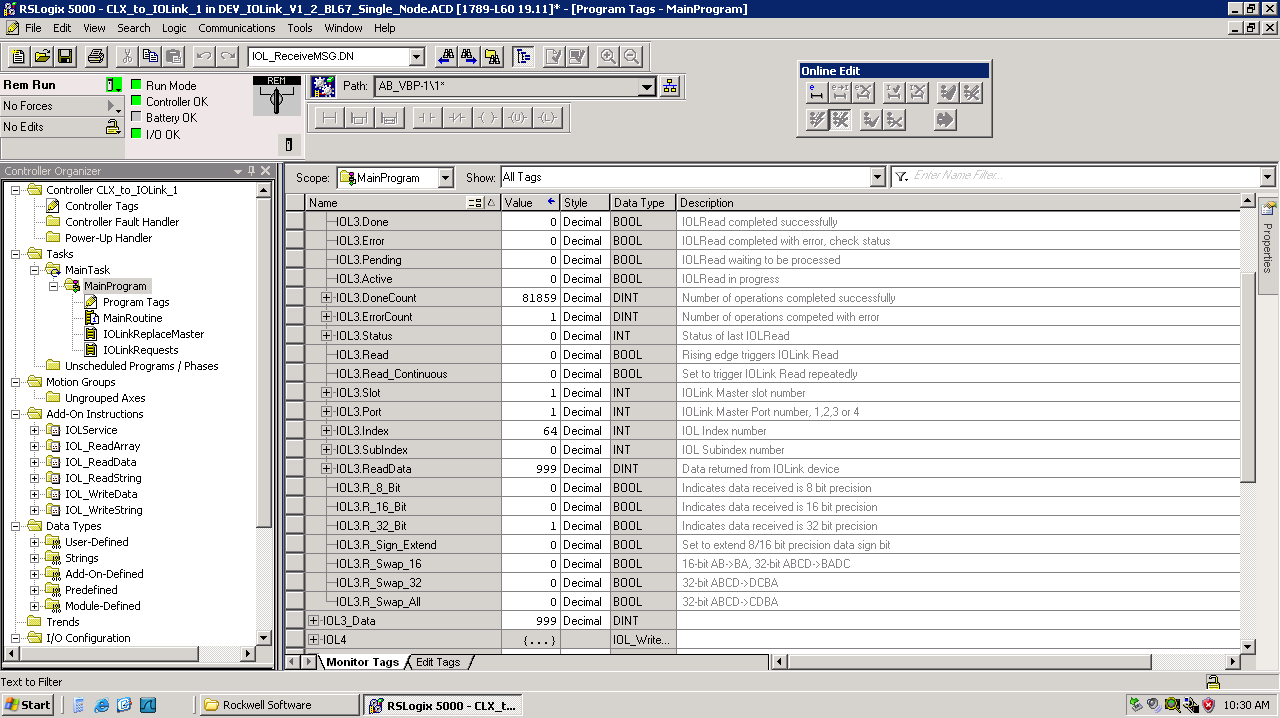
Use IOL\_ReadData to read 8, 16 or 32-bit numeric data. As shown below, IOL\_ReadData instance IOL3 reads the IOLink Parmeter ID found at Index 64 Subindex 0. An off-to-on transition of IOL3.Read will trigger the operation a single time. Setting IOL3.Read\_Continuous on will trigger repeated reads



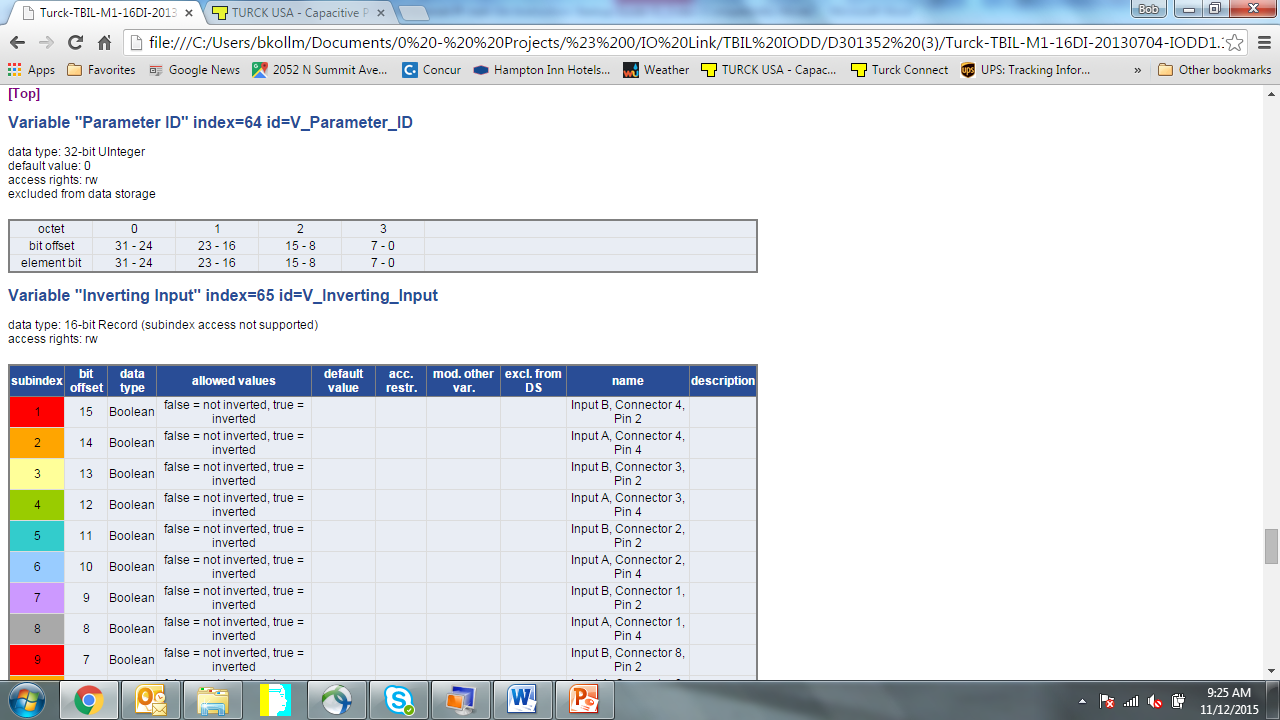
The target variable specified at ReadData must be a DINT type. When the read operation completes the precision of the data read, 8, 16 or 32-bit, is reported back to the application. This information is provided as a convenience since you’ll need to specify the precision to use IOL\_WriteData.



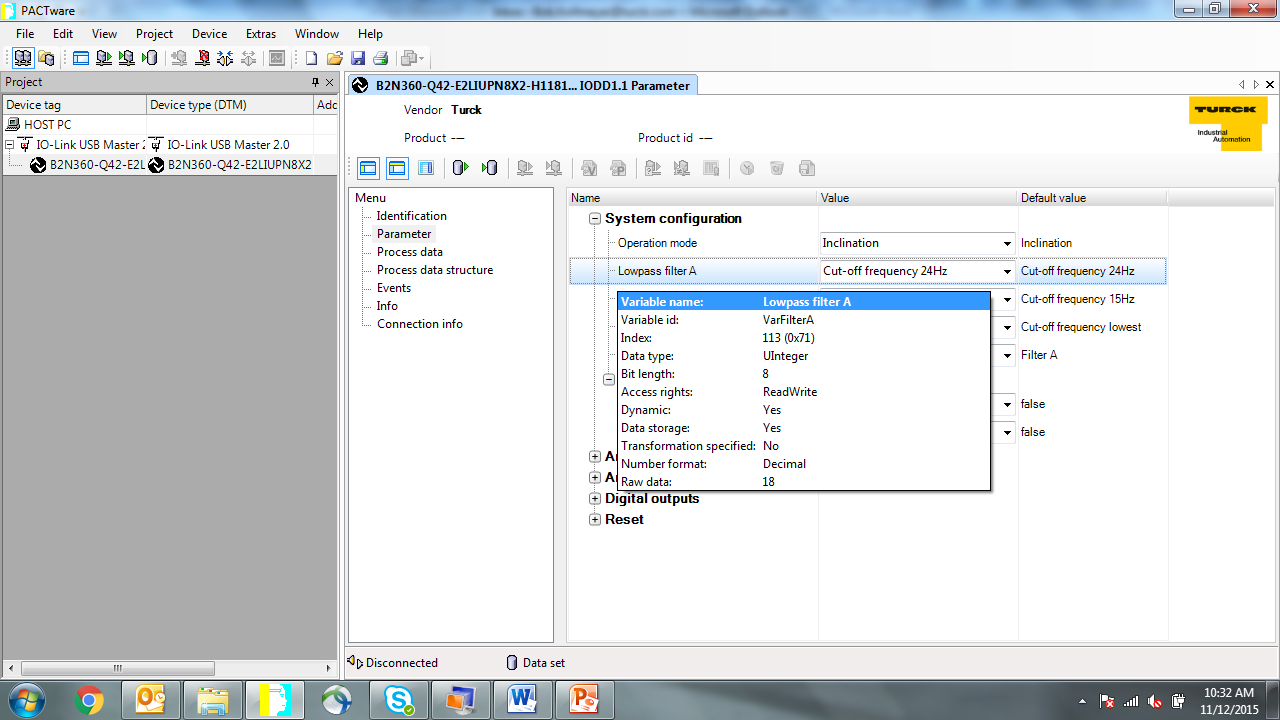
The data translation settings shown below are provided to perform any data manipulation required when reading signed data of less than 32-bit precision or when the device byte order differs from RSL5000.



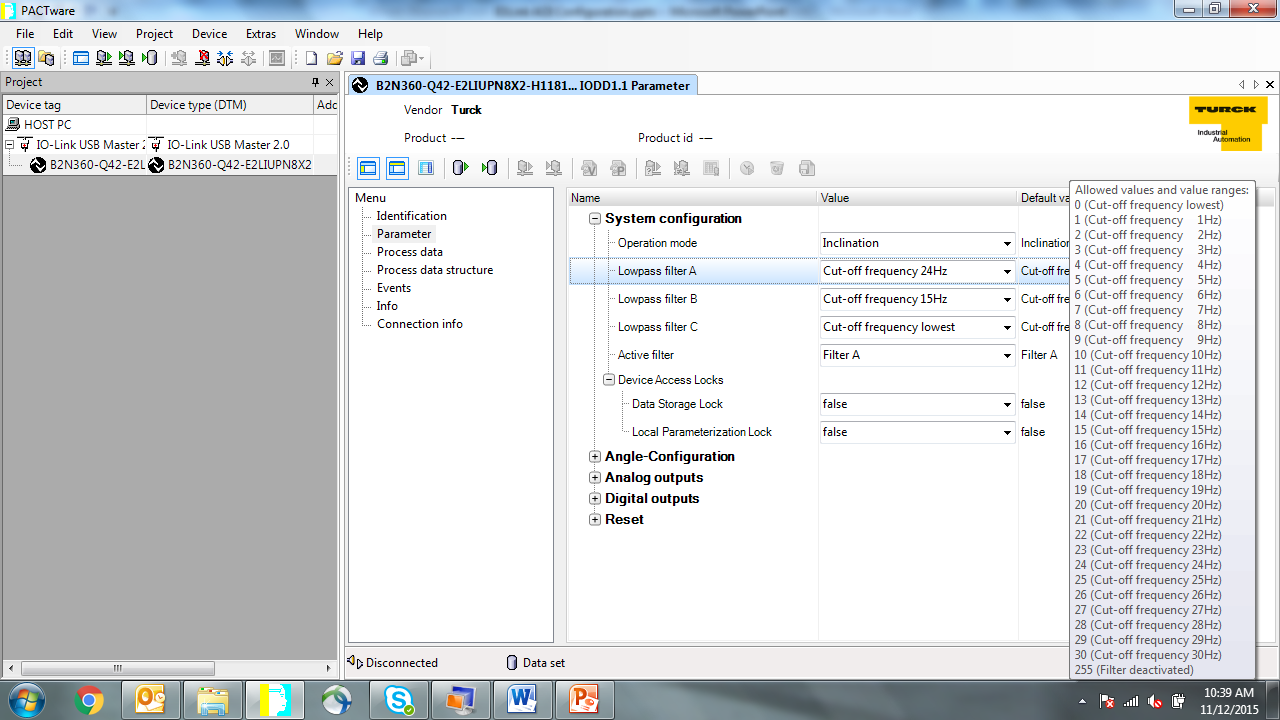
Shown below is the Turck-TBIL-M1-16DI-20130704-IODD1.1-en.html after dropping it in Google Chrome.



Depending on the device and the manufacturer finding the IOLink Index and Subindex value and data precision will be documented either in the device datasheet or in the IODD file. Using PACTware you can view information about the IOLink parameters. As shown below, if you hover over “Lowpass filter A” and right-click the IOLink Index, data precision and access rights are displayed.

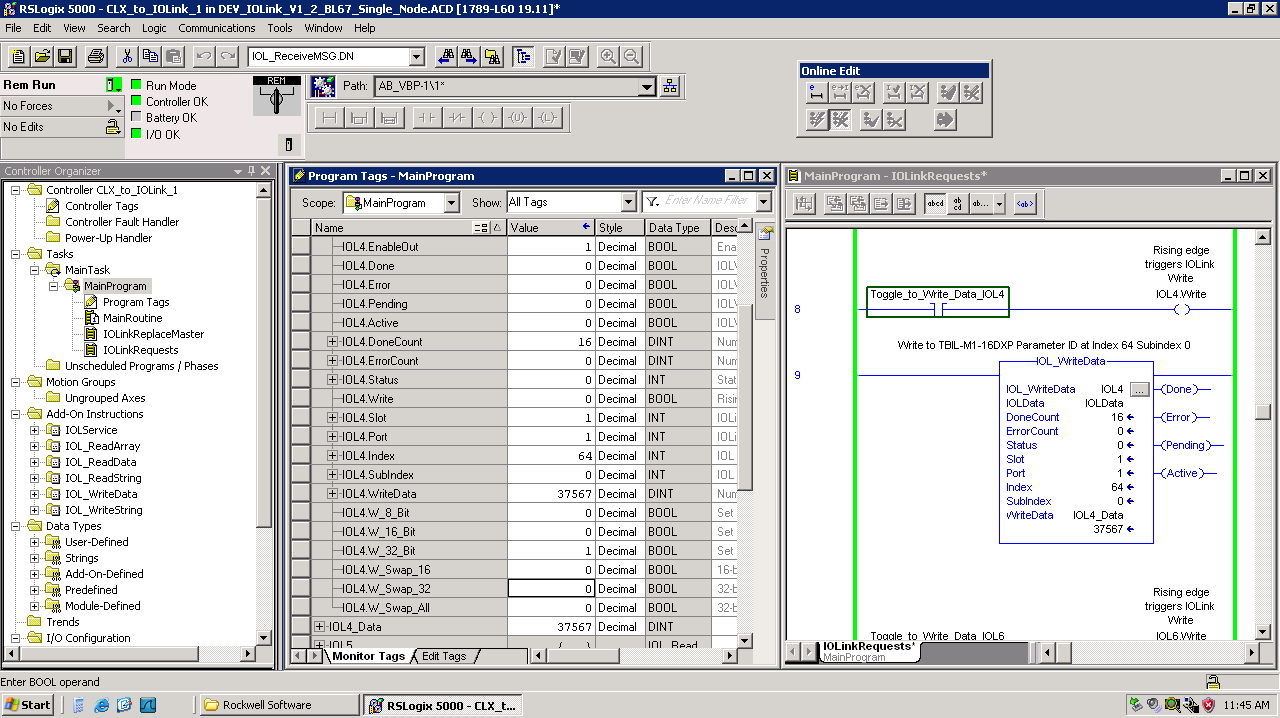


Similarly, if you hover over “Cut-off frequency 24Hz” in the Default Value column and right-click then a description of values that can be written to the device is shown.

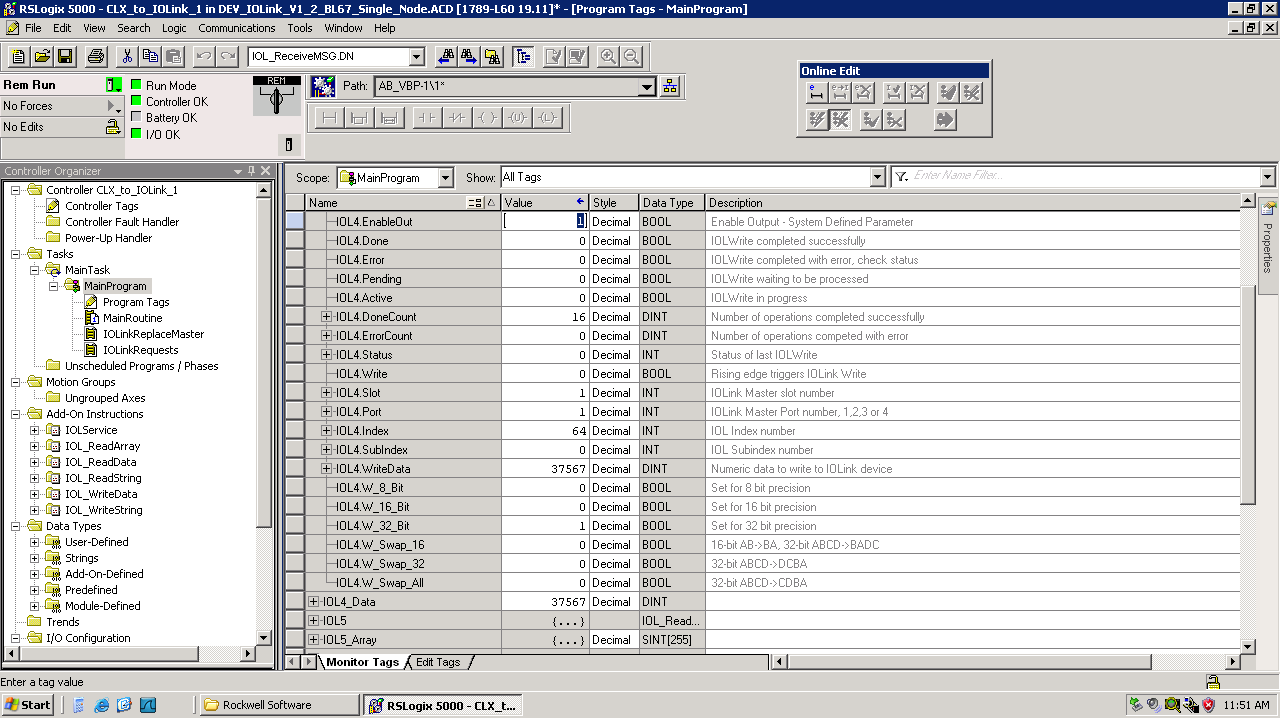


## IOL\_WriteData

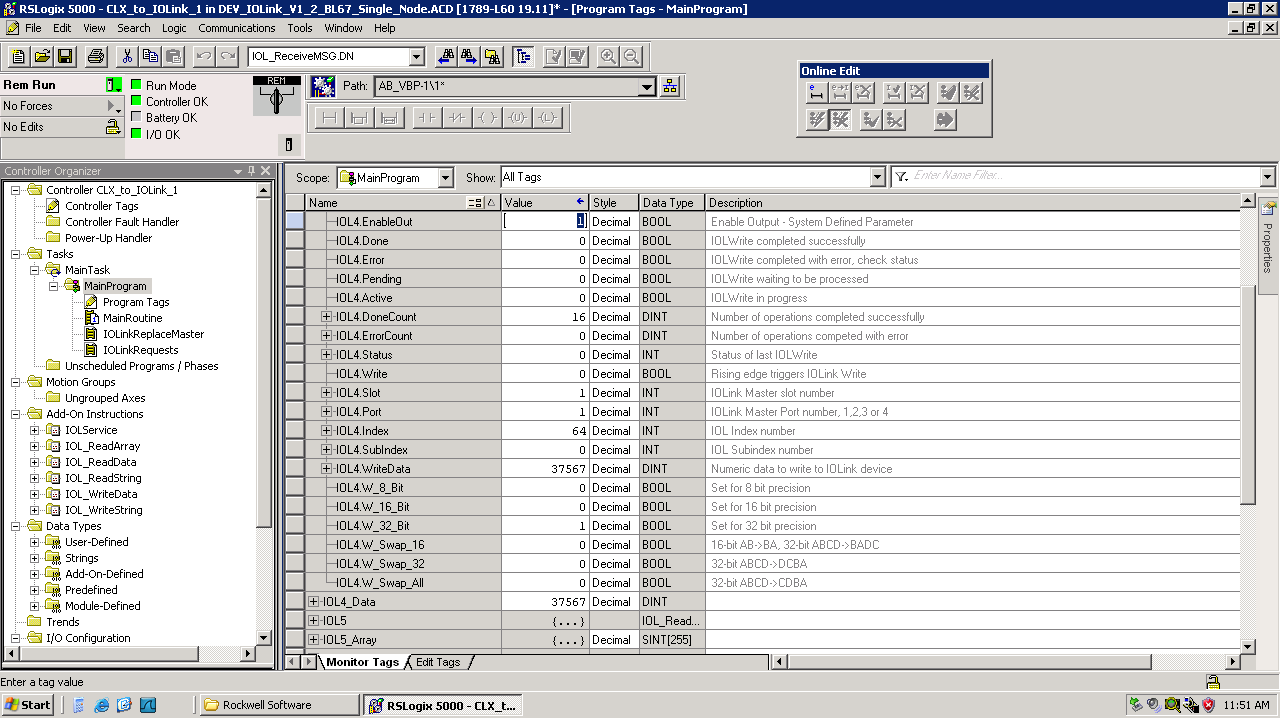
Use IOL\_WriteData to write 8, 16 or 32-bit precision numeric data. As shown below, IOL\_WriteData instance IOL4 writes the 32-bit precision TBIL Parameter ID found at Index 64 Subindex 0. An off-to-on transition of IOL4.Write will trigger the operation a single time.



Unlike read operations that return the precision of the data Write operations require that the application specify whether the target parameter is 8, 16 or 32 bit precision. This is done by setting the appropriate bit in the IOL4 instance tags.



The data translation settings shown below are provided to perform any data manipulation required when the device byte order differs from RSL5000.



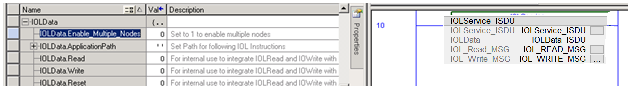
## IOL\_Service\_ISDU - Single Ethernet Node Support

For the IO-Link Call AOI v3.0 a new strategy has been implemented into Turck products to implement the IO-Link Call over the IO-Link Parameter Object via Vendor Specific Class 103 (VSC 103). Further reading on the ‘ISDU’ interface can be done in the *Instructions for Use TBEN-Lx-8IOL* (d301407.pdf) or in the Instructions for Use manual that corresponds to your IO-Link Master.

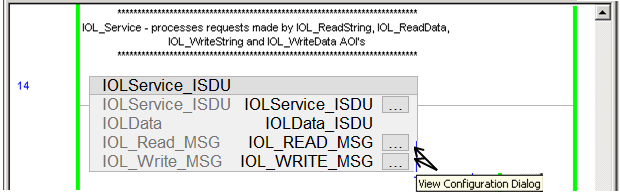
IOL\_Service\_ISDU can replace IOL\_Service in your existing project or it can be used alongside of IOL\_Service – there is no need to replace IOL\_Service with IOL\_Service\_ISDU in an existing system. IOL\_Service should be considered superseded and IOL\_Service\_ISDU should be used for all new project or for the addition on new I/O stations into existing systems.

IOL\_Service\_ISDU performs all of the Ethernet IP communications required to execute the acyclic read and write operations requested by IOLink AOI’s. The application program must contain a single instance of IOL\_Service\_ISDU. The IOLData UDT is used to share information between IOL\_Service\_ISDU and all of the IOL AOI’s. When interfacing to a single BLxx or TBEN IOLink master the setting of the RSL5000 MSG instruction communication path is done in the controller tags as shown below.

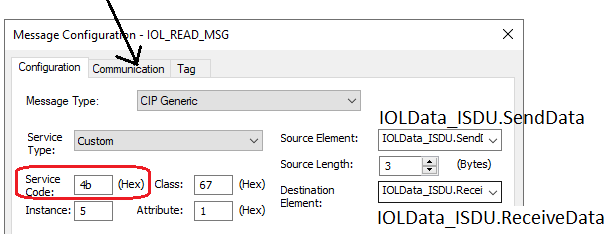
For single Ethernet node applications IOLData.Enable\_Multiple\_Nodes must be set to 0.



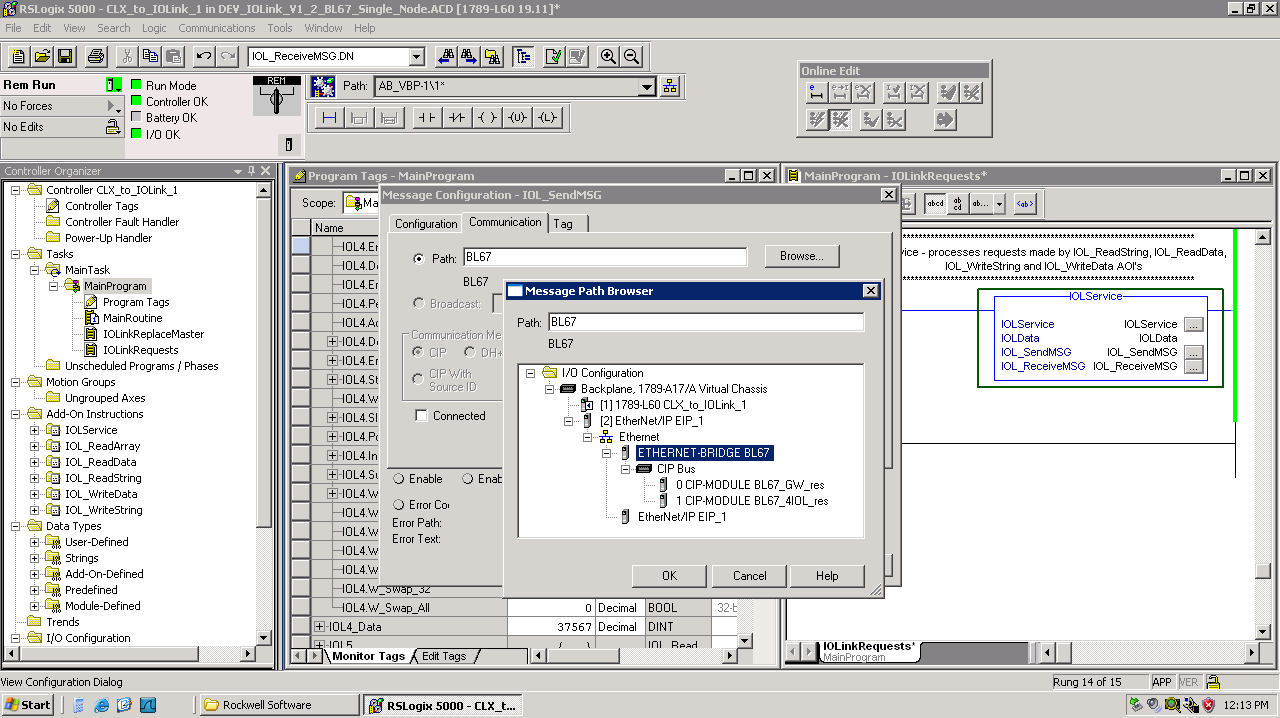
To set the communication path for your application click on the ellipsis next to IOL\_READ\_MSG and IOL\_WRITE\_MSG to access their parameters.



Select the Communication tab.



Select the Communication tab, click on Browse then highlight the station with the IOLink master.



Click on Apply then Okay to complete setting the IOL\_READ\_MSG communication path. Repeat these steps for IOL\_WRITE\_MSG.

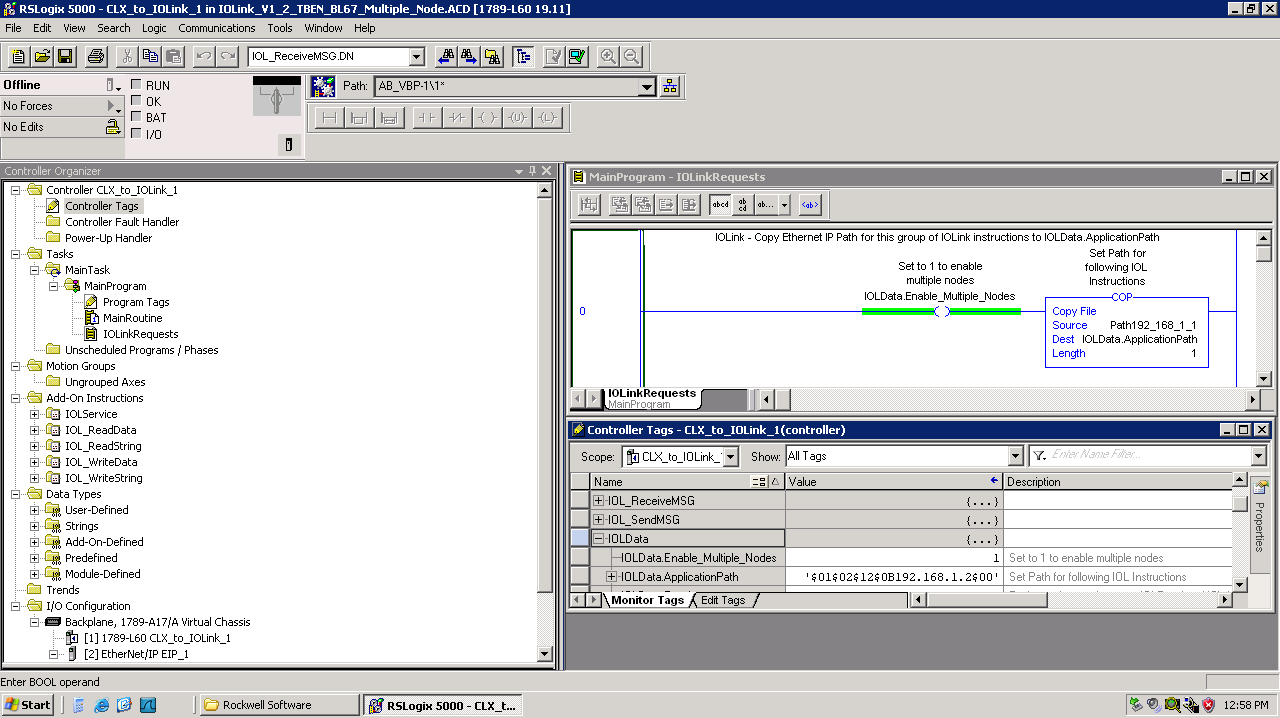
A screenshot of a computer

Description automatically generated with medium confidence

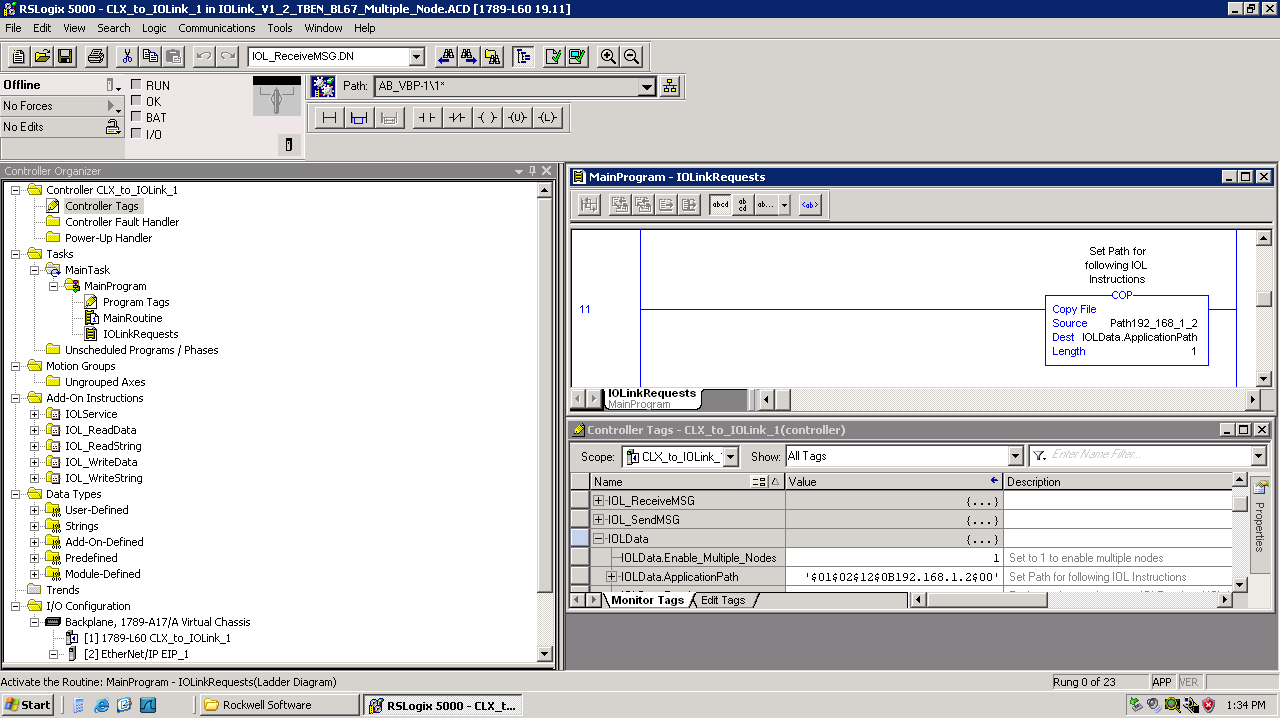
## IOLService\_ISDU – Multiple Ethernet Node Support

IOL\_Service\_ISDU performs all of the Ethernet IP communications required to execute the acyclic read and write operations requested by IOLink AOI’s. The application program must contain a single instance of IOL\_Service\_ISDU. The IOLData UDT is used to share information between IOL\_Service\_ISDU and all of the IOL AOI’s. When interfacing to multiple BLxx or TBEN IOLink masters the setting of the RSL5000 MSG instruction communication path needs to be set prior to each group of read and write AOI’s corresponding to a specific Ethernet node.

For multiple Ethernet node applications IOLData.Enable\_Multiple\_Nodes must be set to 1. At the beginning of the program containing IOL read and write AOI’s simply copy the nodes path into IOLData.ApplicationPath.

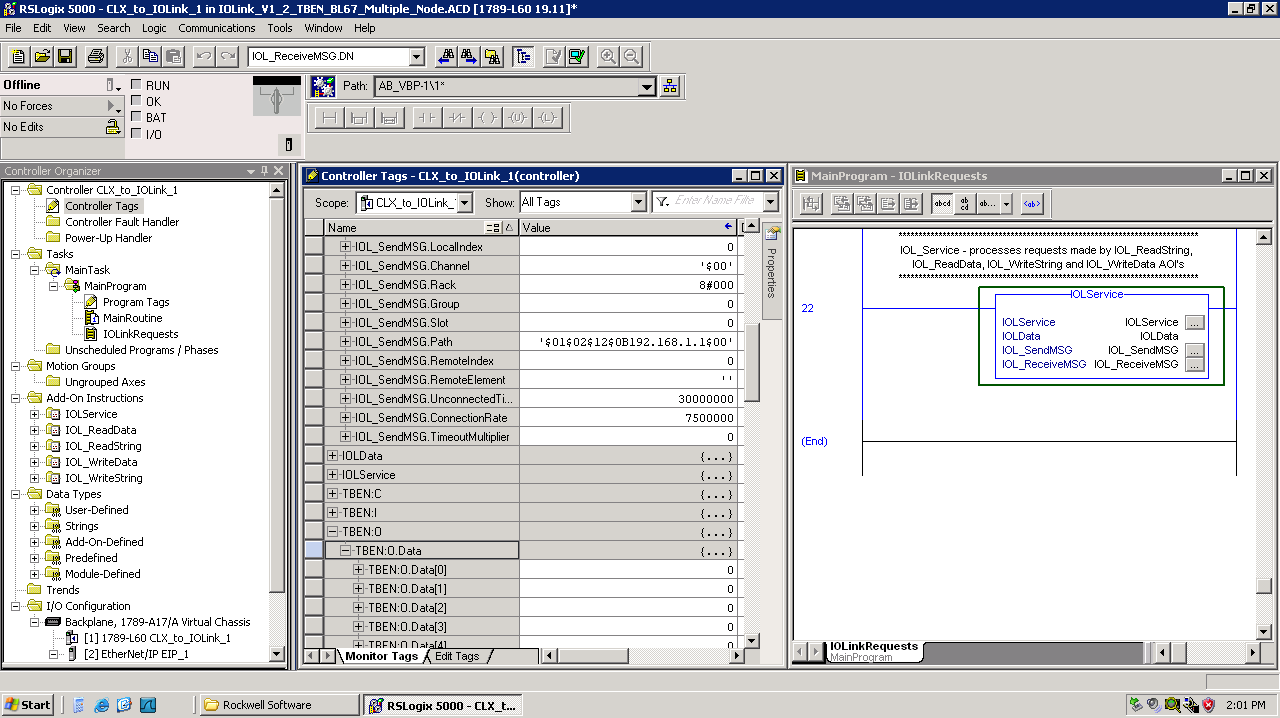


At the beginning of the program containing the IOL reads and write for the next Ethernet node you simply copy that nodes Ethernet communication path into IOLData.ApplicationPath

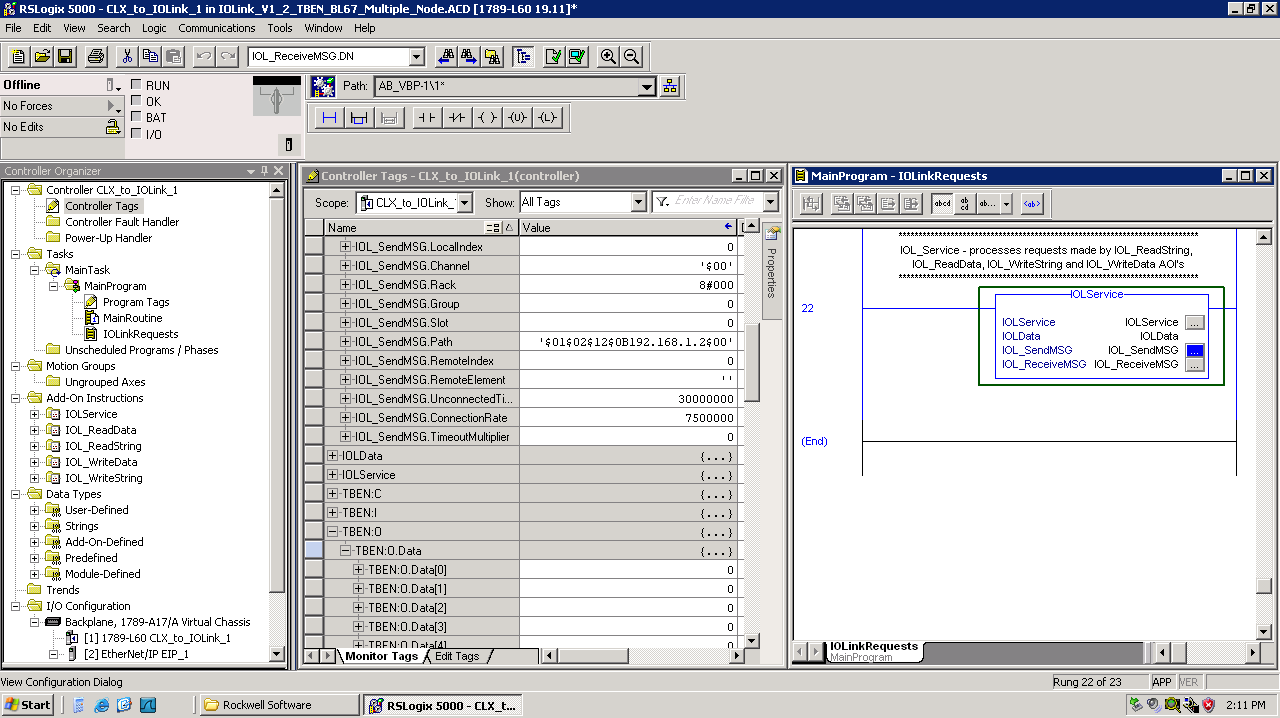


All communications is still processed through a single IOL\_Service\_ISDU and IOL\_Read and IOL\_Write requests for all of the Ethernet nodes can be triggered simultaneously with no concerns.

To determine the communication path string for each node follow the steps described in section 2.8 for setting the communication path for a single node. Once the path is set go to the controller tags and expand IOL\_SendMSG. As shown below the path for this node is “$01$02$12$0B192.168.1.2$00” – this can be entered into a standard RSL5000 string variable exactly as shown.



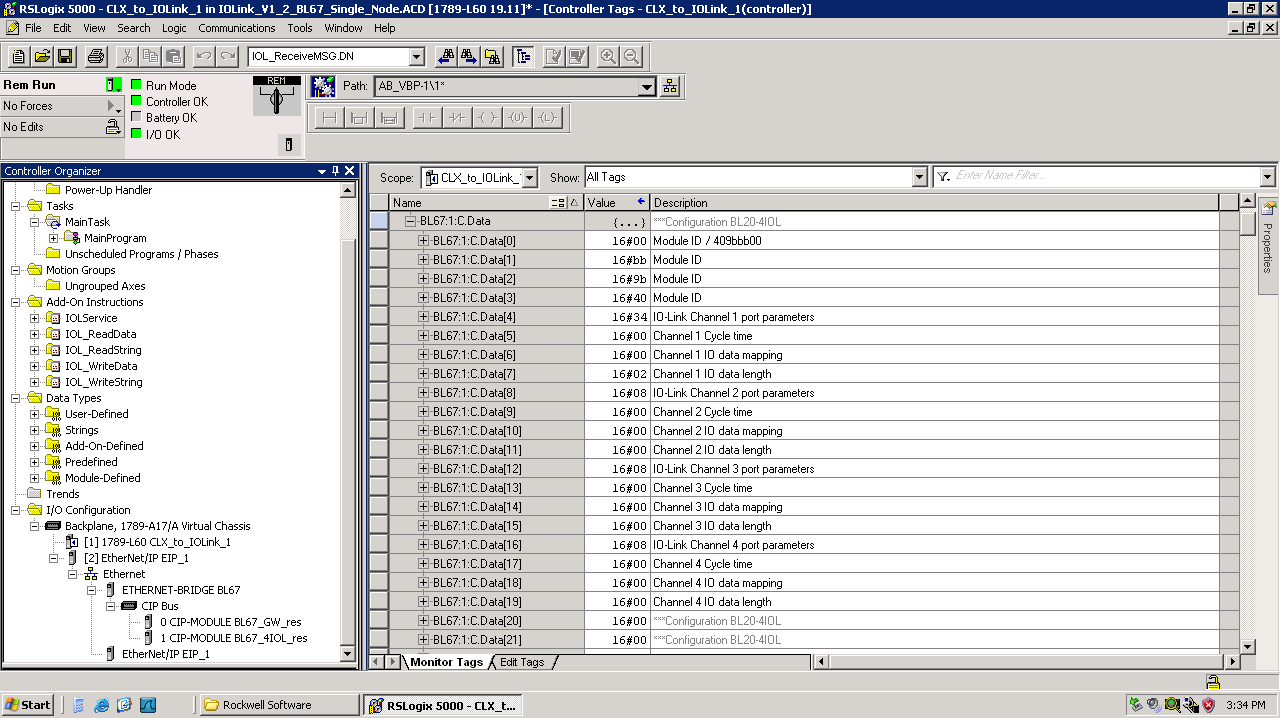
Repeat for each node and create a string variable for each that can be copied into IOLData.ApplicationPath.



# Configuring IO-Link using Generic Ethernet/IP CIP Bridge

Using the Generic Ethernet/IP CIP Bridge method of integrating the IOLink master has the great advantage that the IOLink module configuration data is saved as part of the RSL5000 program and the module configuration is applied every time the Ethernet connection is activated. If the BLxx or TBEN needs to be replaced no configuration of the IOLink channels is required – just set the units IP address. When the Ethernet connection is established the configuration data saved in the RSL5000 program will be applied.

After adding Generic Ethernet/IP CIP Bridge to your program simply drag and drop the gateway and IOLink master modules from an existing application or the Turck supplied catalog file.



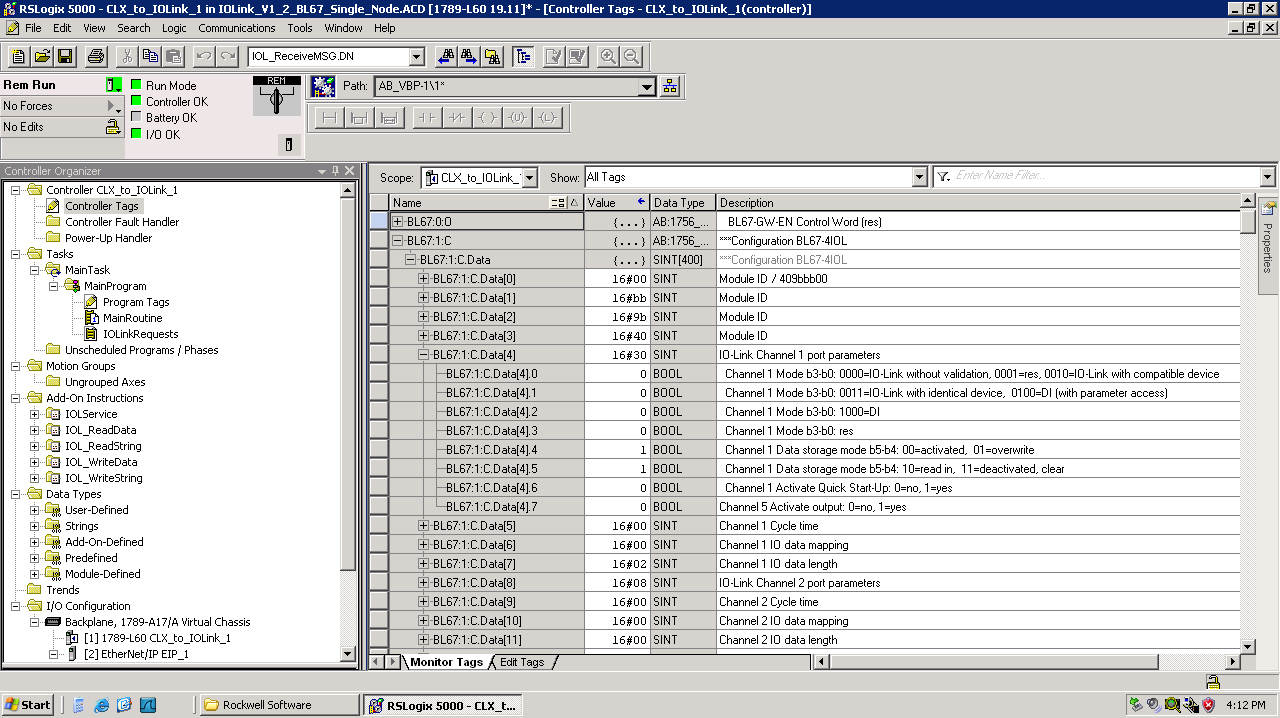
When the BL67\_4IOL is dropped on the CIP Bus the fully commented input, output and configuration tags are automatically inserted in the Controller Tags.



Expand BL67:1.C to set up the initial values for each IOLink channel. A brief overview of the configuration follows. For in-depth information refer to the D301333 IOLink Master for BL20 and BL67 User manual.



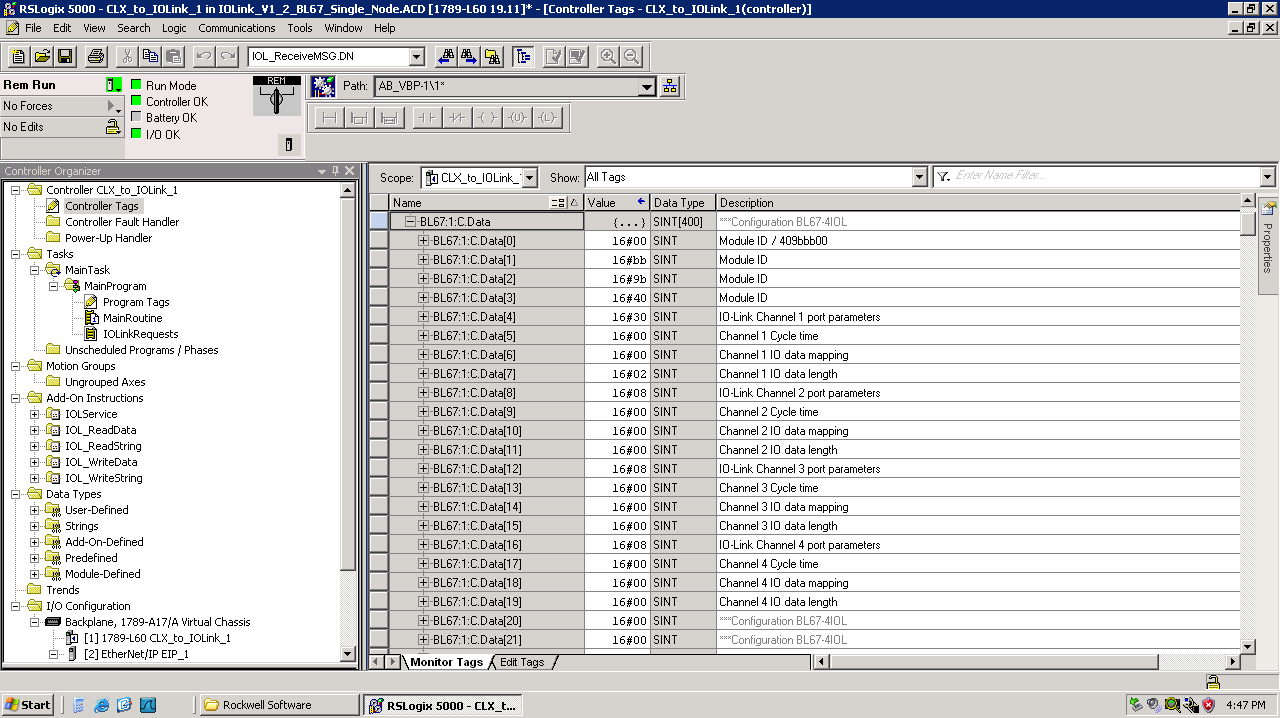
Port Parameters to see the operating mode, data storage mode and quick start configuration.



A simple place to start is with BL67:1:C.Data[4] bits .3, .2, .1 and .0 all 0 corresponding to *Channel 1 Mode   
b3-b0: 0000=IO-Link without validation*. It’s also convenient to configure unused channels as 24VDC inputs by setting b3-b0 *to Channel x Mode b3-b0: 1000=DI* this eliminates the diagnostic indication for missing device.

An important benefit of IOLink is simplified device replacement. This requires the device be an IOLink Version 1.1 device and typically uses *Channel 1 Data storage mode b5-b4: 00=activated*. While you’re getting started a simple configuration to start is with *Channel 1 Data storage mode b5-b4: 11=deactivated*, clear, as shown above. See the manual for a very clear description of this feature.

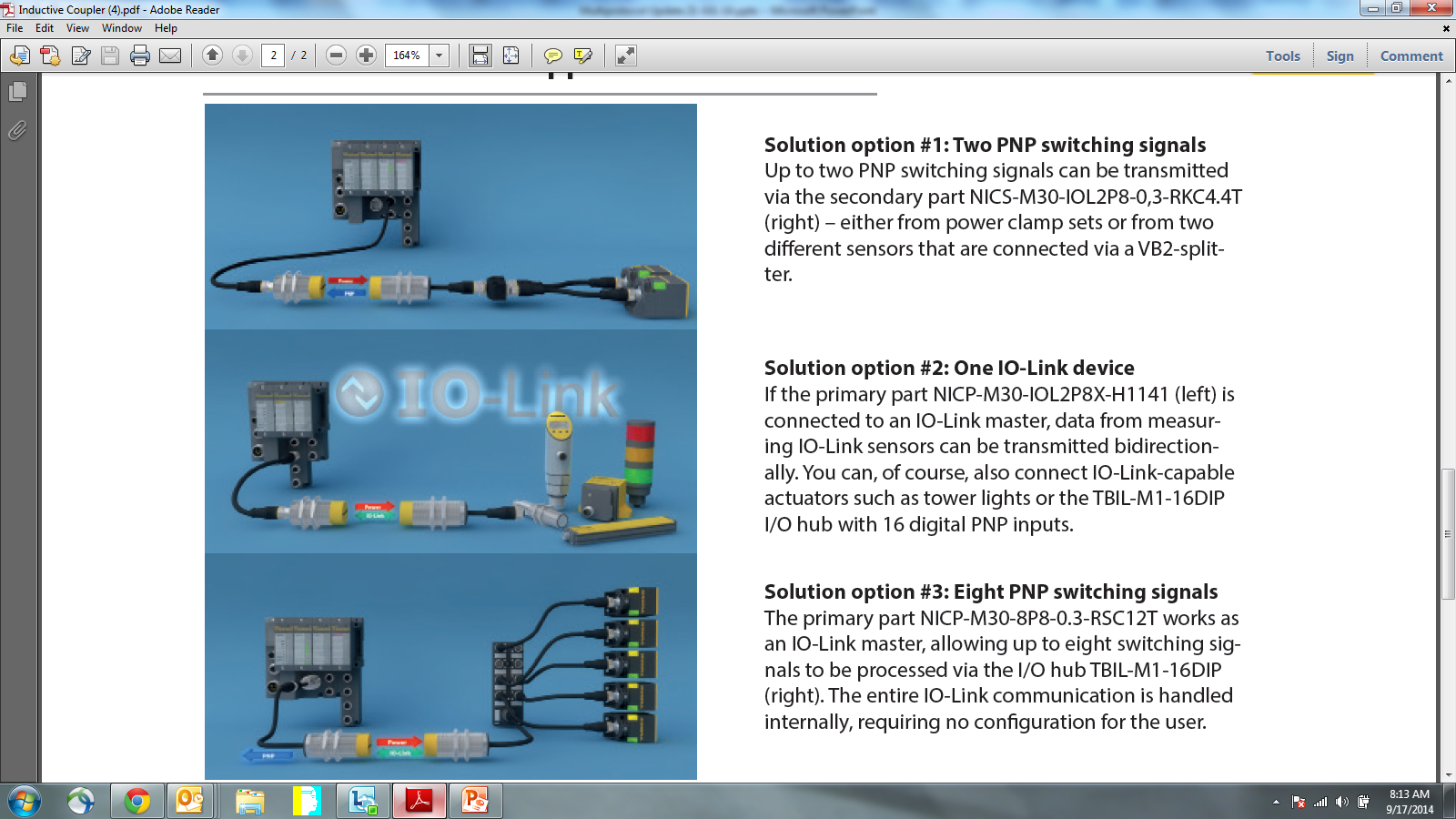
## Cycle Time



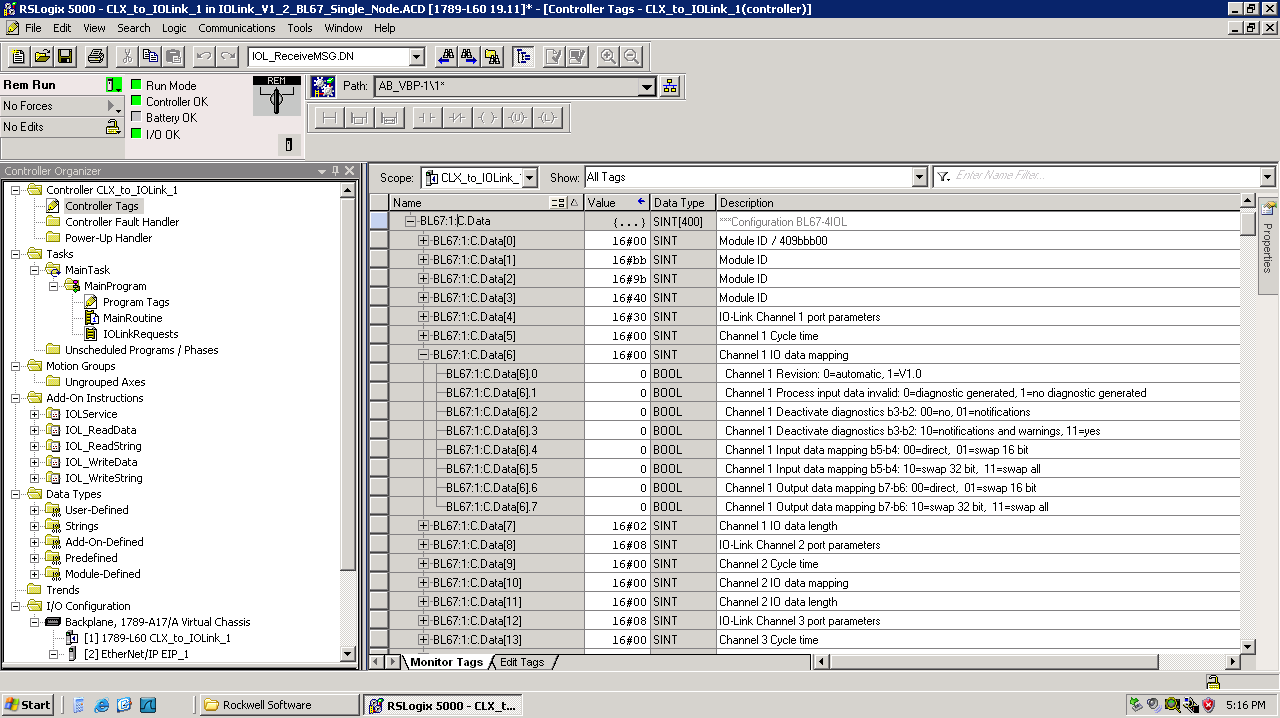
In the majority of applications the Cycle Time is left at 16#00 which results in the IOLink cyclic data update time being automatically set to the value specified by the device.

As an example the TBIL-M1-16DXP Cycle Time is automatically set to 3.2ms.

An exception would be when using IOLink through an inductive coupler as shown below. The inductive coupler transfers 0.5 amps of 24V power as well as IOLink communication and is an effective means of controlling IO though and air gap when compared to slip rings. When using the inductive coupler set the time delay to 16#44 which corresponds to an 8ms cycle time.



## IO Data Mapping



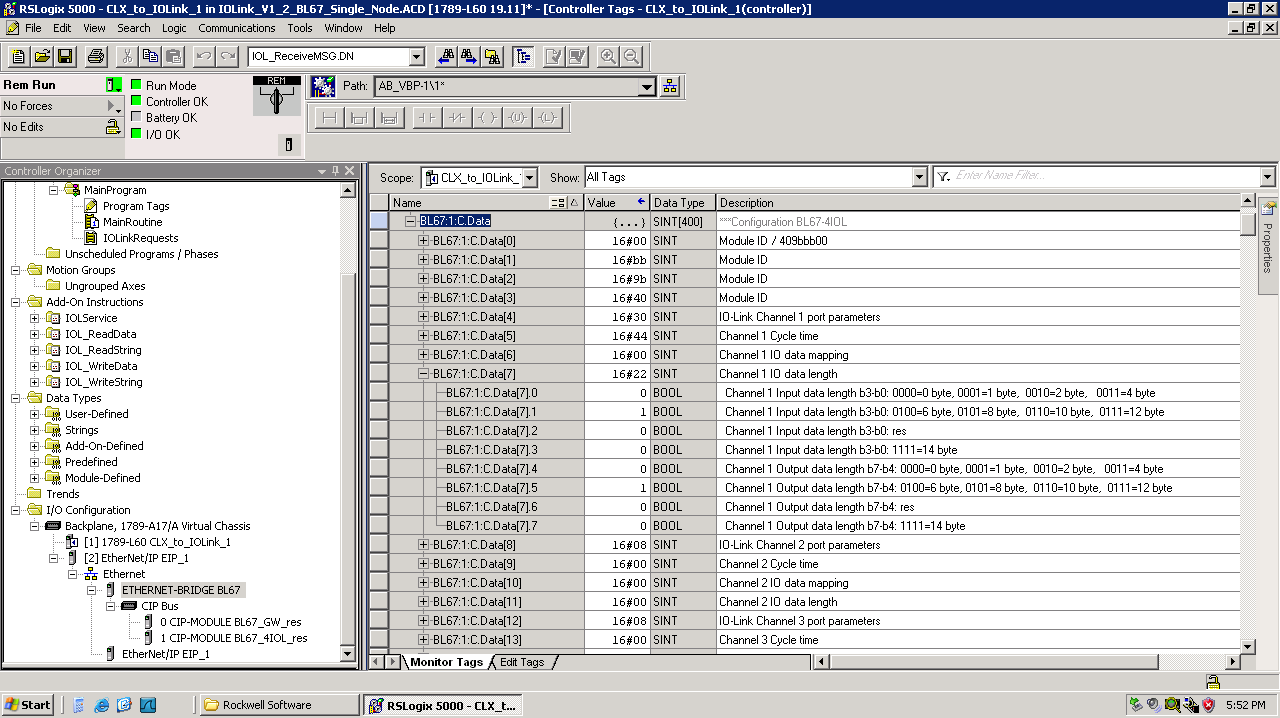
Typically b3 thru b0 are left 0 however when using the inductive coupler you can set *Channel 1 Deactivate diagnostics b3-b2: 11=yes* to eliminate the diagnostic error annunciation during the time when the couplers are not aligned. An example would be a dial index application during station change.

The input and output data mapping selections provide the necessary byte order swapping when interfacing devices using a different byte order than RSL5000.

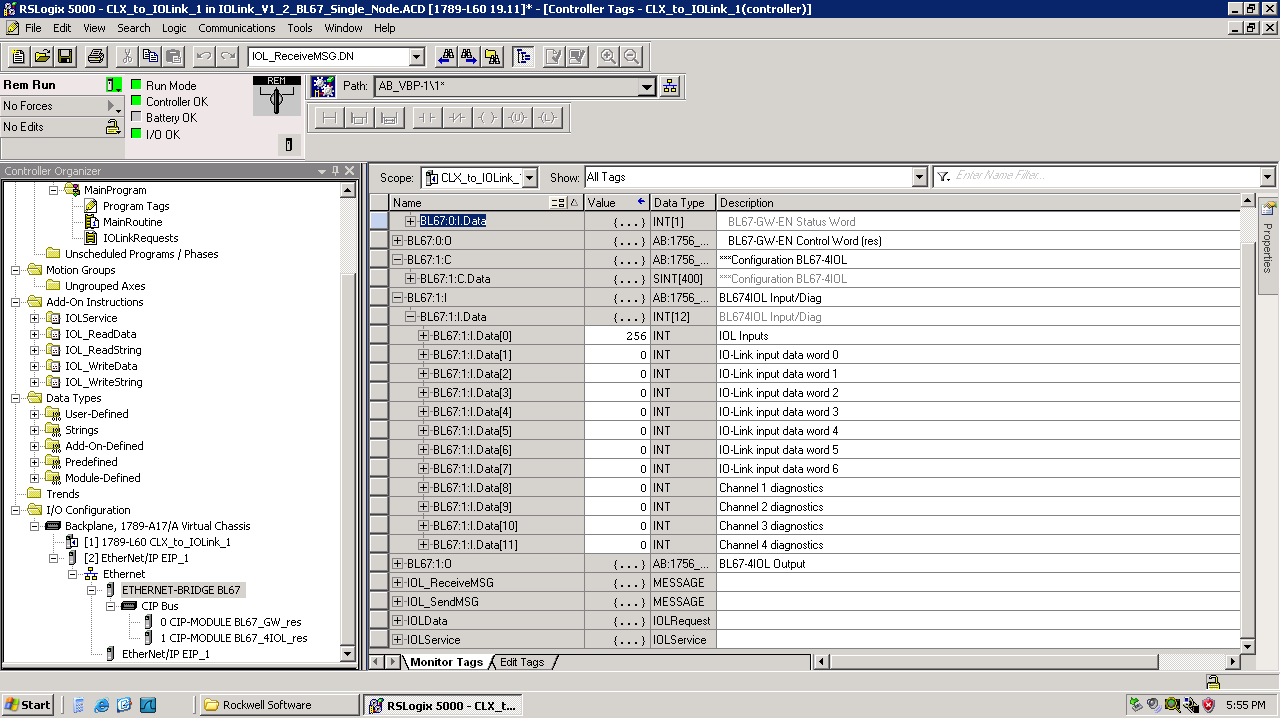
## IO Data Length and Input Data Mapping

Typical IOLink devices have 16-bits of cyclic process data that be transferred from the IOLink device to the IOLink master. The cyclic data will be mapped into the Ethernet nodes input data tags. The configuration data below shows *Channel 1 Input data length 0010=2 byte* which results in the state of the TBIL-M1-16DXP inputs being mapped to BL67:I:Data[1] IO-Link input data word 0

**C**onfiguration data BL67:1:**C**.Data



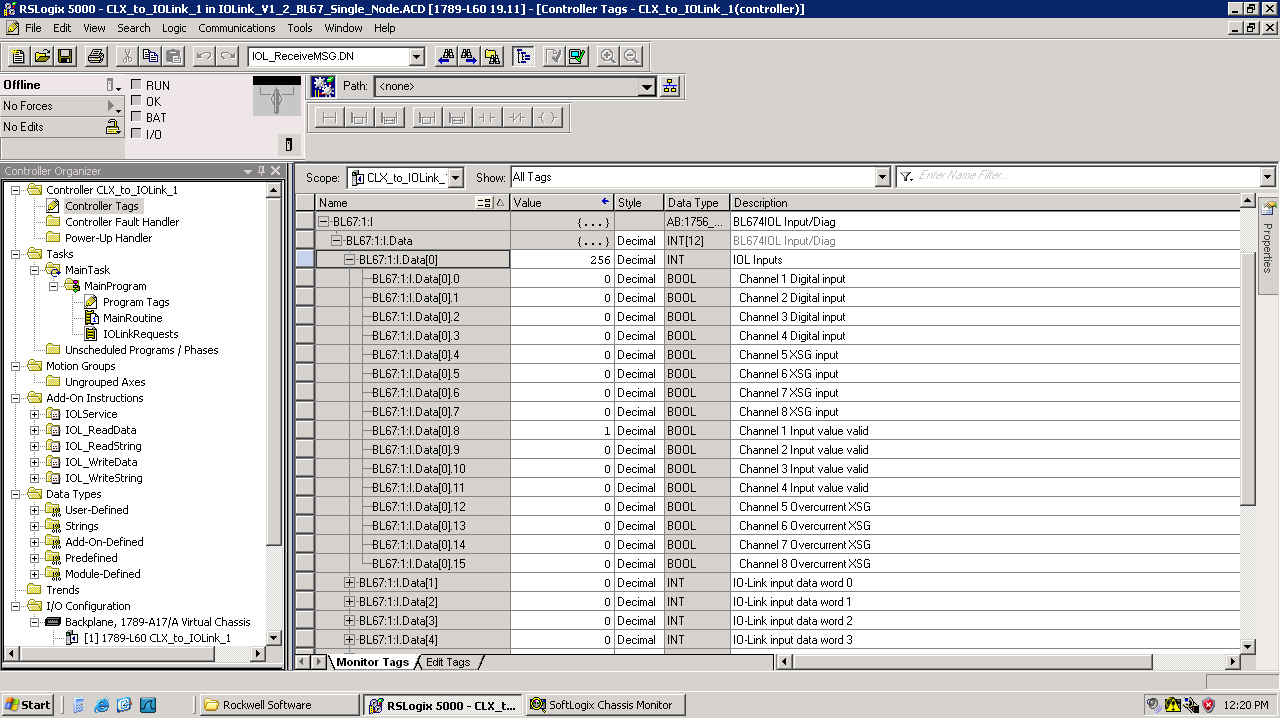
**I**nput data BL67:1:**I**.Data



IOLink channel input data is mapped onto IO-Link input data words 0 thru 7 in ascending order. In the example above the TBIL-M1-16DXP input states will be available at BL67:1:I.Data[1].0 to .15 *= IO-Link input data word 0*. Subsequent IOLink device input data will appear in the words that follow.

## IO –Link Input Information

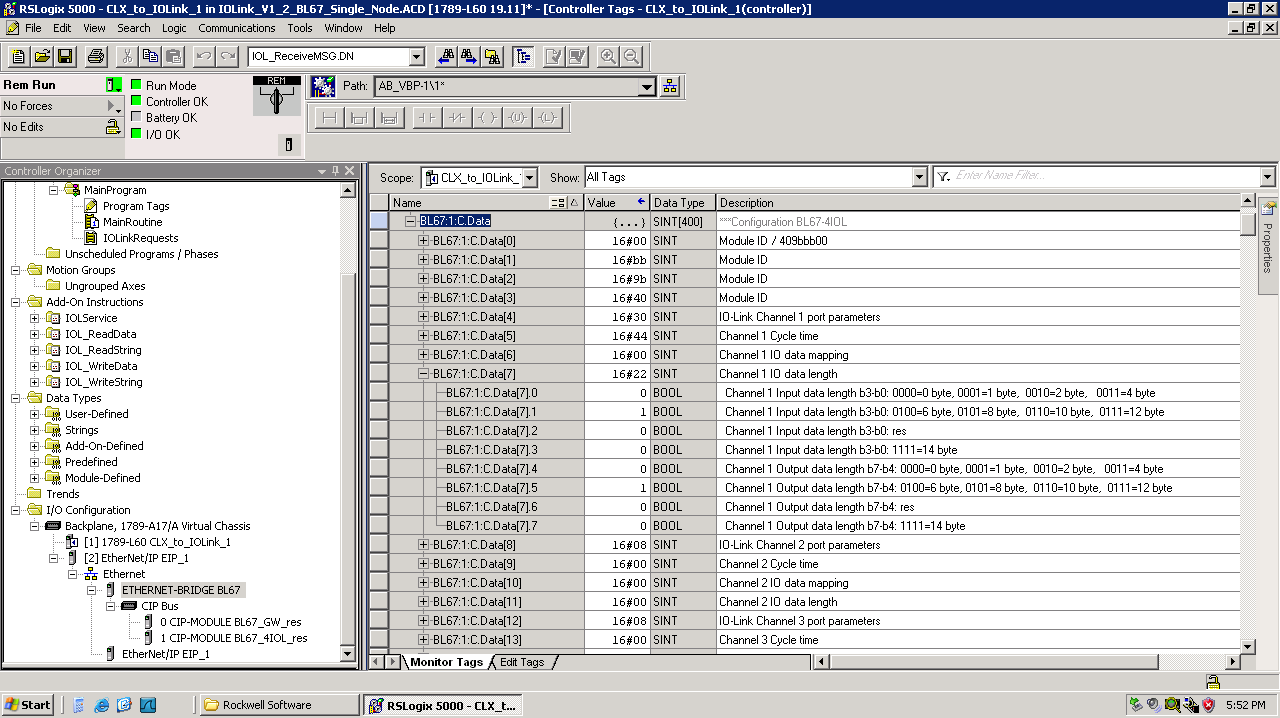
The Turck IOLink master provides four channels that can be used as either IOLink masters or standard 24V inputs and four channels that can be used as either 24V inputs or 24V outputs. Channels 1 to 4 of the IOLink master are the IOLink/24V input channels. Channels 5 through 8 are the configurable (XSG) standard 24V IO channels. The current state of the 24V inputs as well as the state of the IOLink channel is available in the first word of the IOLink master input data – BL67:1:I.Data[0]. In applications using inductive coupling the *IOLink status Channel x Input value valid* is used to confirm the IOLink connection has been established.



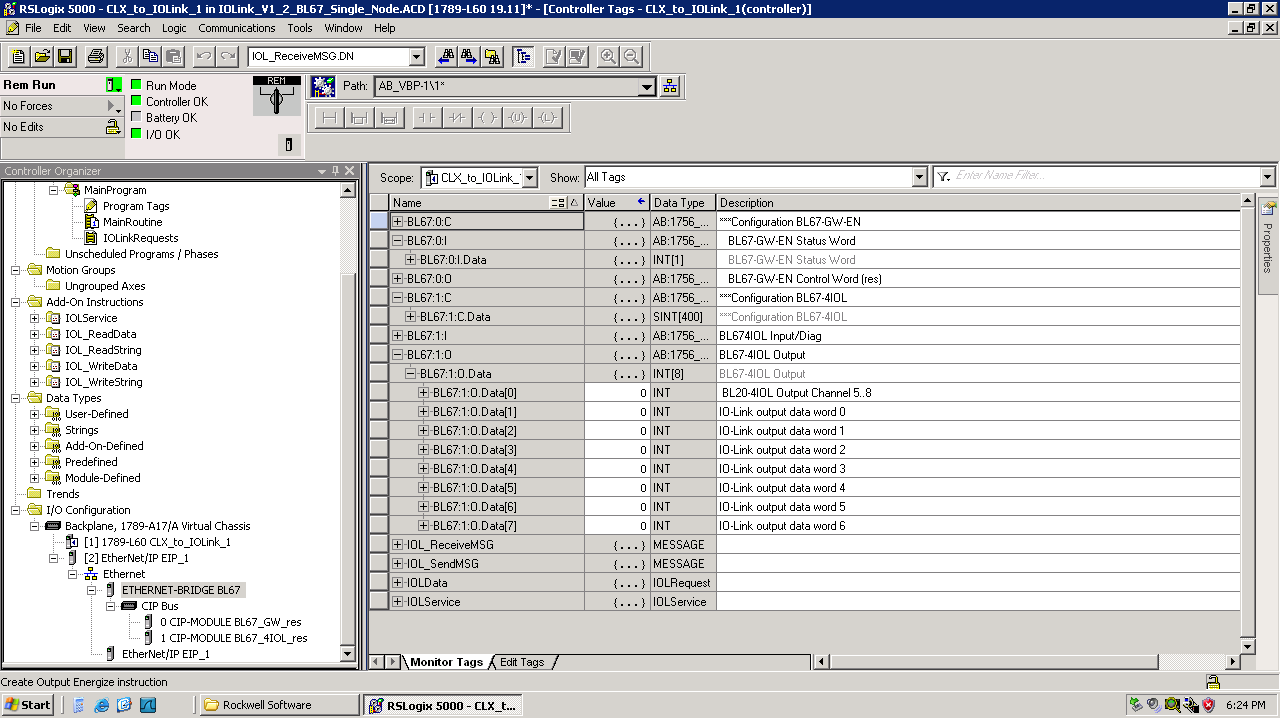
## IO Data Length and Output Data Mapping

An example of an IOLink device using Output Data would be the TBIL-M1-16DXP which provides 16 points of configurable 24V IO points. In this case 16-bits of cyclic process data will be transferred from the IOLink master to the TBIL-M1-16DXP. The cyclic data will be sourced from the Ethernet nodes output data tags. The configuration data below shows *Channel 1 Output data length b7-b4: 0010=2 byte* which results in the output states of the TBIL-M1-16DXP being sourced from BL67:1:O.Data[1].

**C**onfiguration data BL67:1:**C**.Data



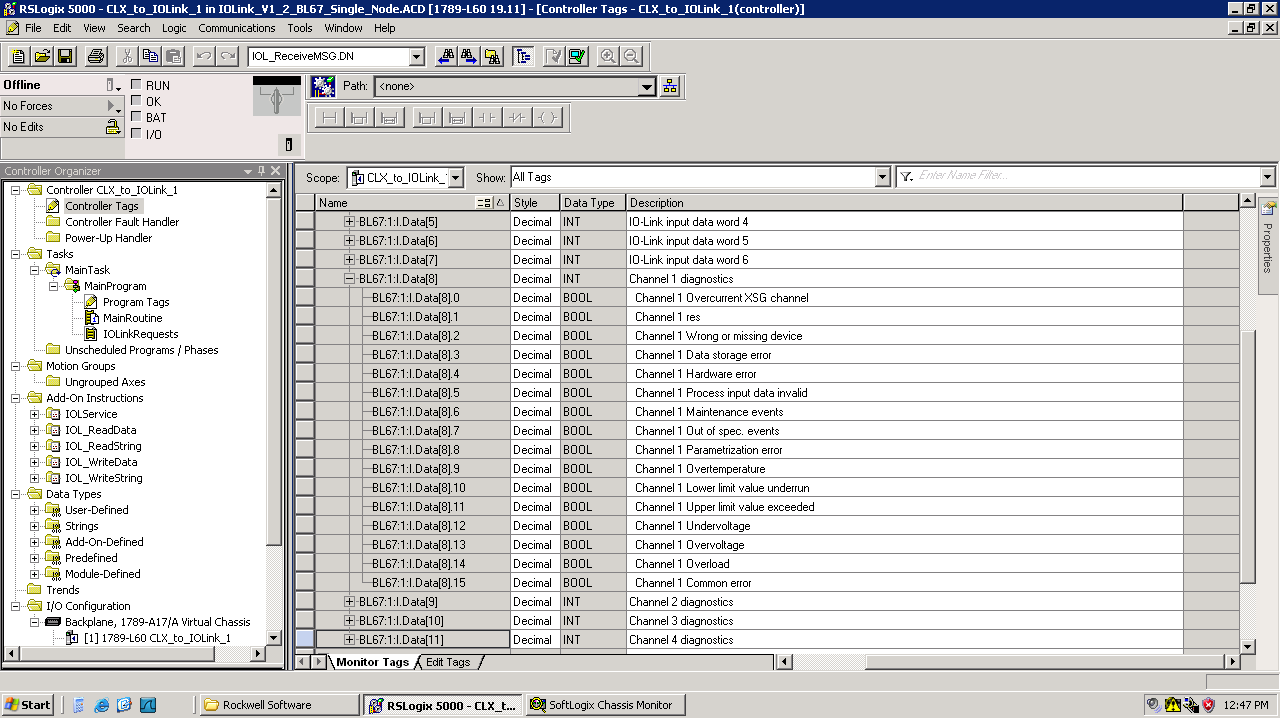
**I**nput data BL67:1:**O**.Data



IOLink channel output data is sourced from IO-Link output words 0 thru 7 in ascending order. In the example above the TBIL-M1-16DXP output states will be source from BL67:1:O.Data[1].0 to .15 – *IO-Link output data word 0*. Subsequent IO-Link device output data will appear in the words that follow. No additional configuration of the TBIL-M1-16DXP is required – for IO points used as inputs simply leave the corresponding output bits turned off.

## IO –Link Input Information

Diagnostic information for each of the IOLink channels is available following the cyclic input data at BL67:I:Data[8] through [11]. Typical errors include *Channel x Data storage error* when the configuration activates data storage but the device is a V1.0 instead of V1.1 or *Channel x Process input data invalid* when the device is not connected or the inductive coupler set is misaligned.



# Status Error Code Descriptions

Errors are typically caused by incorrect Index or Sub-index values, data out of range or write data precision errors.

Status Description

0 Operation completed successfully

Status = 100 - 101, Ethernet IP MSG instruction errors - check communication parameters

100 IOL\_SendMSG completed with error, check communication parameters

101 IOL\_RecevieMSG completed with error, check communication parameters

110 IOL\_Service Retry Limit exceeded

111 Slot specified not an IOLink master

Status = 1000 to 1255, no response from device, connection error

1016 Device not connected or Port not configured as IOLink port

Status = 2000 to 2255, device response indicates error

2000 Device error, no details

2017 Device error, Index not available

2018 Device error, Subindex not available

2032 Device error, Service temporarily not available

2033 Device error, Service temporarily not available - local control

2034 Device error, Service temporarily not available - device control

2035 Device error, Index not writeable

2048 Device error, Parameter value out of range

2049 Device error, Parameter value above limit

2050 Device error, Parameter value below limit

2051 Device error, Parameter value length overrun (check IOLWrite precision/string length)

2052 Device error, Parameter value length underrun (check IOLWrite precision/string length)

2053 Device error, Function not available

2054 Device error, Function temporarily not available

2064 Device error, Parameter set invalid

2065 Device error, Inconsistent parameter set

2130 Device error, Application not ready

Status = 3000 to 3255, device response indicates vendor specific error

3000 Vendor specific error, no details

3001 Vendor specific error, vendor specific error number

3255 Vendor specific error, vendor specific error number

Status = 7000 to 7255, IOL-M Error Codes

7000 Unexpected write request instead of read request

7001 Decode error

7002 Port occupied by another task

Status = 8000 to 8255, IOL-M Error Codes

8000 Timeout when IOL-Ds or IOL-M ports are busy

8001 IOL Index out of range

8002 Port address greater than maximum

8003 Port function not supported

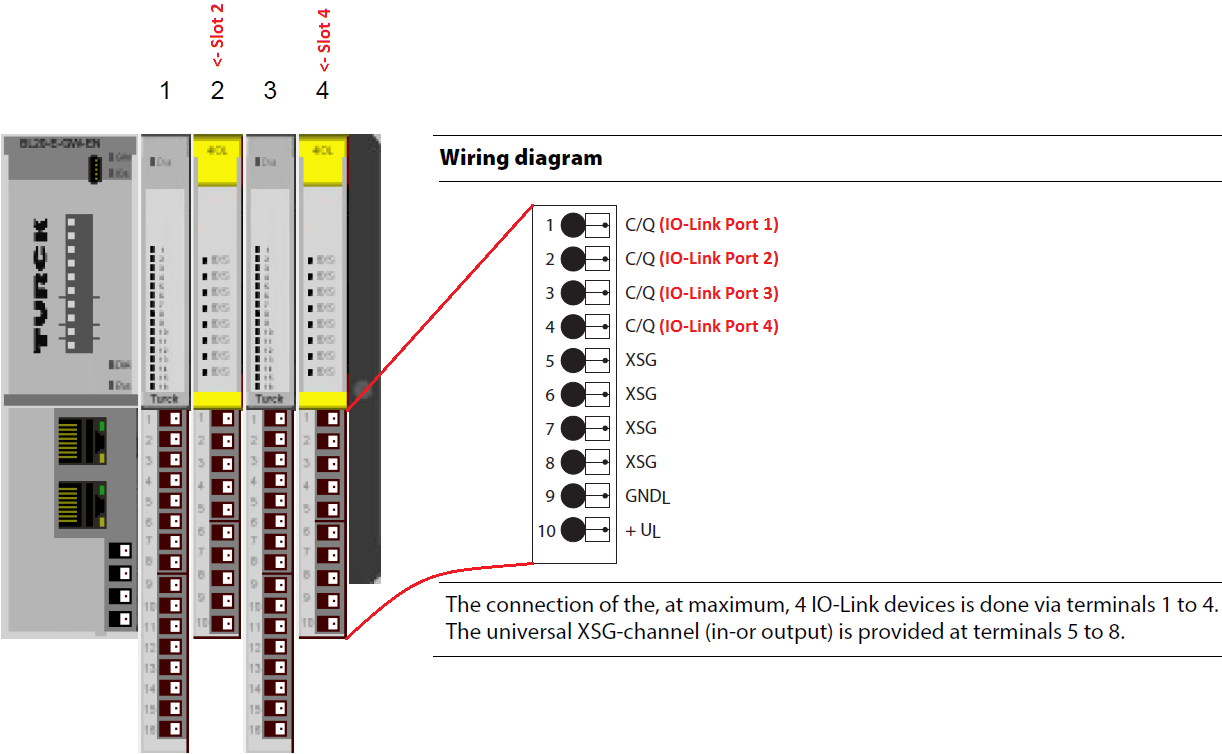
# Appendix A – Slot and Port Designations Explained

Each physical IO-Link Master Port on a station is addressed with the IO-Link Call AOI by Slot and Port. The following diagrams show how to determine Slot and Port on your Turck station.

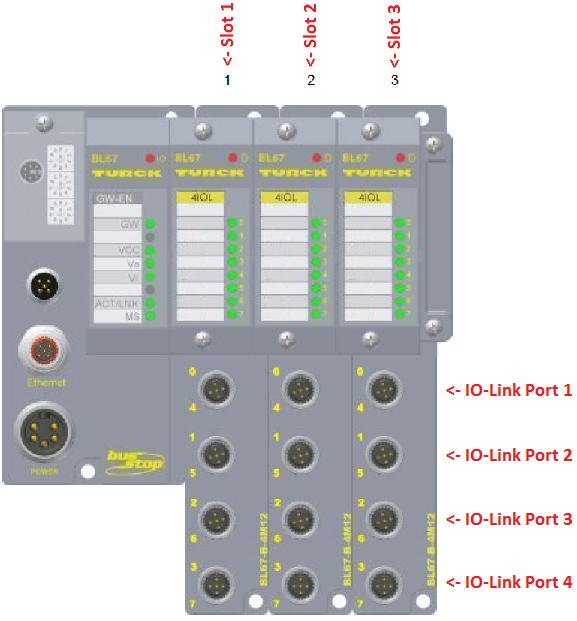
## BL20 and BL67 Modular Platforms

The BLxx Modular platforms are Slot based, each I/O slice represents 1 Slot. Slots are numbered 1 through 32 from left to right.

As an example, the BL20 Station below with IO-Link Master cards in Slots 2 and 4 would support an IO-Link Call through the IO-Link Call AOI on Slots 2 or 4, Ports 1 through 4.

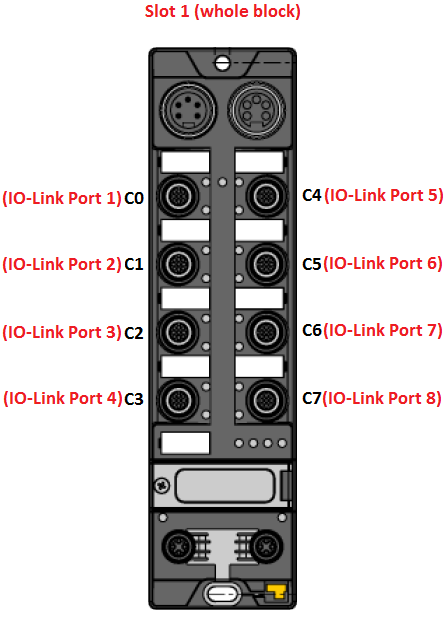


As an example, the BL67 Station below with IO-Link Master cards in Slots 1, 2 and 3 IO-Link would support an IO-Link Call through the IO-Link Call AOI on Slots 1, 2 or 3, Ports 1 through 4.



## TBEN-Lx-8IOL

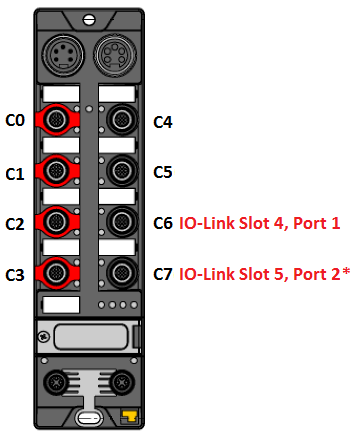
The TBEN-Lx-8IOL blocks are Slot based, all ports on this block are in Slot 1. Ports are numbered 1 through 8. Note that IO-Link Port 1 is on the physical connector designated C0, this pattern follows for all 8 ports.



## TBIP-Lx-FDIO1-2IOL

The TBIP-Lx-FDIO1-2IOL blocks are Slot based, IO-Link ports on this block are in Slots 4 and 5. Ports are numbered 1 and 2. Note that IO-Link Port 1 is on the physical connector designated C6 and IO-Link Port 2 is on the physical connector designated C7\*.

\*For the TBIP-Lx-FDIO1-2IOL power must be supplied to the IO-Link Master on Port C7 via Safety Bit FSO 1 before the port will respond to any IO-Link Call requests from the IO-Link Call AOI function, Port C6 is powered at all times by V1.



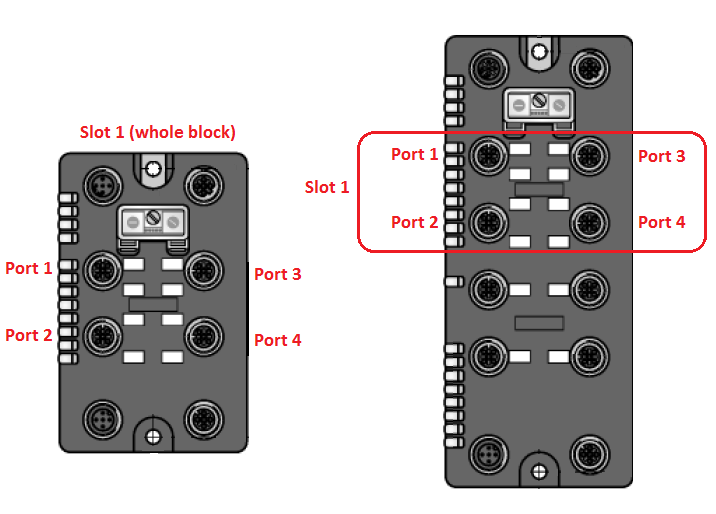
## TBEN-S2-4IOL

The TBEN-S2-4IOL blocks are Slot based, all ports on this block are in Slot 1. Ports are numbered 1 through 4. Note that IO-Link Port 1 is on the physical connector designated C0, this pattern follows for all 4 ports.



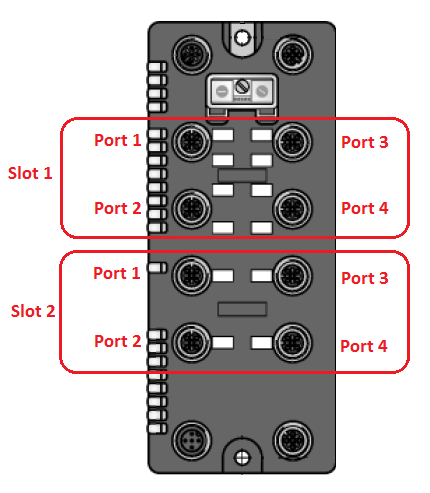
## BLCEN-xM12MT-4IOL-xxx

The BLCEN blocks are Slot based, all ports on this block are in Slot 1. Ports are numbered 1 through 4. Note that IO-Link Port 1 is on the physical connector designated C0, this pattern follows for all 4 ports.



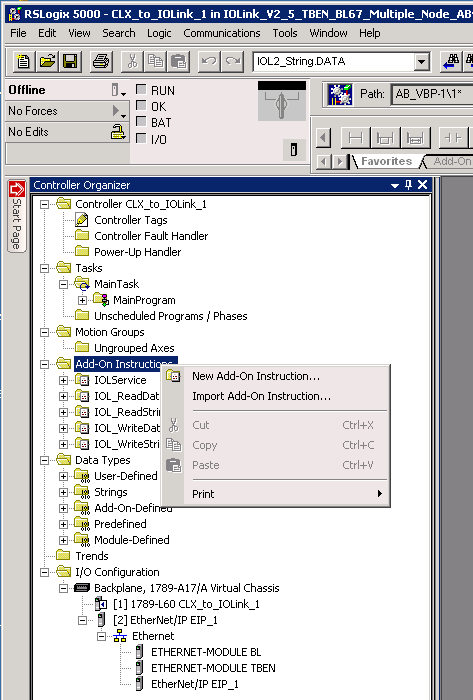
## BLCEN-xM12MT-4IOL-4IOL

The BLCEN blocks are Slot based, all ports on the block with 2 IO-Link Masters are in Slot 1 or Slot 2. Ports are numbered 1 through 4.

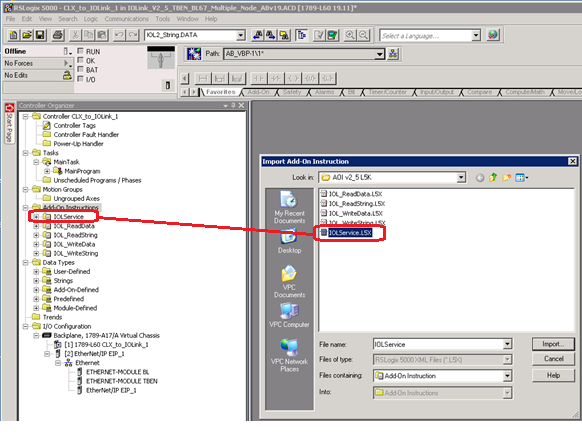


# Appendix B – Updating Older Versions of the IO-Link Call AOI

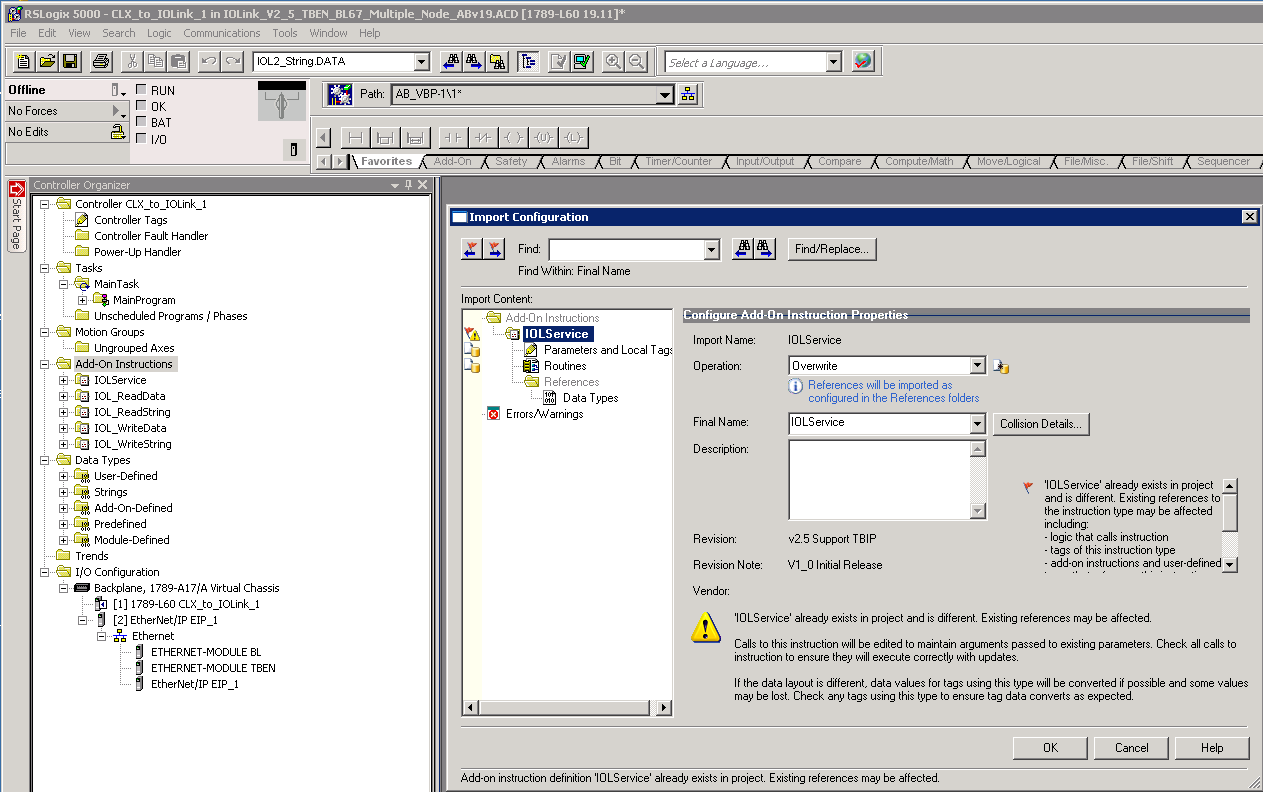
To update the versions of the AOI to the latest version, right click on Add-On Instructions and click Import Add-On Instruction…



Select the Add-On instruction you want to update, note the names MUST be the same:



RSLogix will warn you that you are about to replace an existing AO, you can verify you are importing the correct revision on this screen:



Once imported you can verify you have the correct version of each AOI, it is recommended all AOIs for the IO-Link call be the same revision:

