

IND360

PK Controller EtherNet/IP Integration



METTLER TOLEDO

© METTLER TOLEDO 2025

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of METTLER TOLEDO.

U.S. Government Restricted Rights: This documentation is furnished with Restricted Rights.

Copyright 2025 METTLER TOLEDO. This documentation contains proprietary information of METTLER TOLEDO. It may not be copied in whole or in part without the express written consent of METTLER TOLEDO.

COPYRIGHT

METTLER TOLEDO® is a registered trademark of Mettler-Toledo, LLC. All other brand or product names are trademarks or registered trademarks of their respective companies.

**METTLER TOLEDO RESERVES THE RIGHT TO MAKE REFINEMENTS OR CHANGES
WITHOUT NOTICE.**

Contents

1	Integration Manual – EtherNet/IP	1-1
1.1.	Hardware.....	1-1
1.2.	Configuration.....	1-2
1.2.1.	Open and Configure DeltaV Explorer.....	1-2
1.2.2.	Configure P01 Ethernet IO Port	1-3
1.2.3.	Add a New Ethernet I/O Physical Device (PDT1)	1-6
1.2.4.	Configure IND360 for 2-Block Format	1-9
1.2.5.	IND360 Memory Mapping for 2Block Format	1-12
1.2.6.	Add New IND360 Ethernet I/O Signals.....	1-13
1.2.7.	IND360 Memory Mapping for 8 Block Format	1-19
1.2.8.	Configure 8 Block Format Settings	1-20
1.2.9.	IND360 Memory Mapping of 8Block Format	1-23
1.3.	Delta V Diagnostics: Overall Integrity.....	1-25
1.3.1.	DeltaV Diagnostics – Device Data Value	1-26
1.4.	Export/Import Signals	1-28
1.4.1.	Editing the Exported Text File Using the Bulk-Edit Excel Add-On.....	1-33
1.4.2.	Importing Signals	1-35
1.5.	Troubleshooting.....	1-37
1.5.1.	De-Commissioning the PK Controller.....	1-37
1.5.2.	Commissioning the PK Controller.....	1-37
1.5.3.	Export/Import	1-38

Notice: Version and Format Information

DeltaV Explorer	v14.3.1	Build 7282
SAI Data Format	8-Block Format 2-Block Format	
IND360 Device Firmware Version	IND360_WebUpgrade_Analog_Method2_Step1_2.07.0000.mtb IND360_WebUpgrade_Analog_Method2_Step2_Stack_1.02.0003.mtb	

1 Integration Manual – EtherNet/IP

1.1. Hardware



Figure 1-1: DeltaV PK Hardware Layout



Figure 1-2: IND360 Indicators and Transmitter

1.2. Configuration

Perform the following step-by-step procedures to configure the IND360 to work natively with the DeltaV PK Controller.

1.2.1. Open and Configure DeltaV Explorer

1. To integrate the IND360, use the DeltaV Software Application **DeltaV Explorer**. To open **DeltaV Explorer**, click **Start | DeltaV Engineering | DeltaV Explorer**.

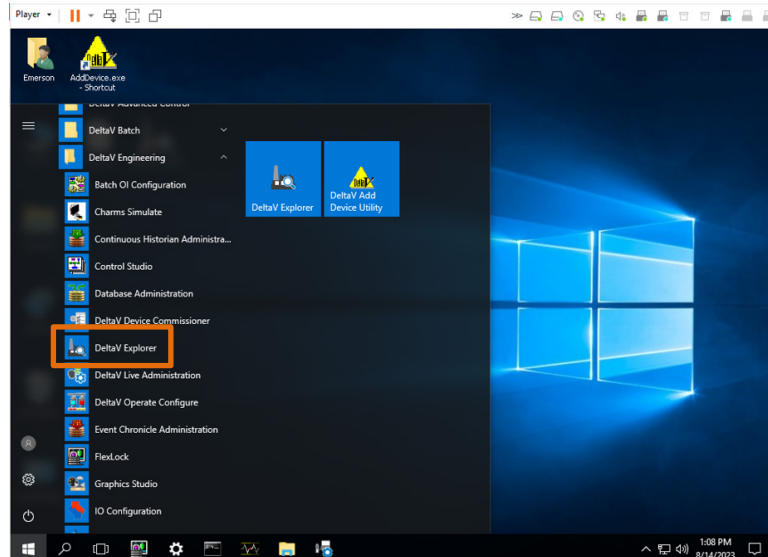


Figure 1-3: Starting DeltaV Explorer

2. The DeltaV Explorer will display its initial view.

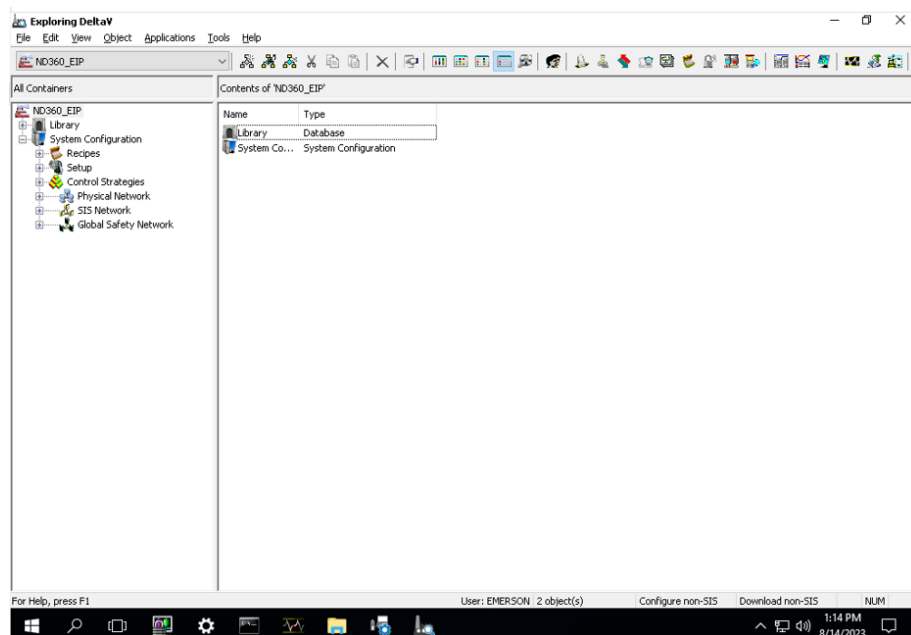


Figure 1-4: DeltaV Explorer Initial View

3. To navigate the DeltaV Explorer, do the following – the step numbers correspond to the indicators in Figure 1-5.
 - 1 Select and expand the **IND360 Demo** project **System Configuration** object.
 - 2 Select and expand the **Physical Network** object.
 - 3 Select and expand the **Control Network** object.
 - 4 Select **PKCTRL** (the PK Controller).
 - 5 Select the **P01** Ethernet I/O Physical device, to add the IND360 configuration.

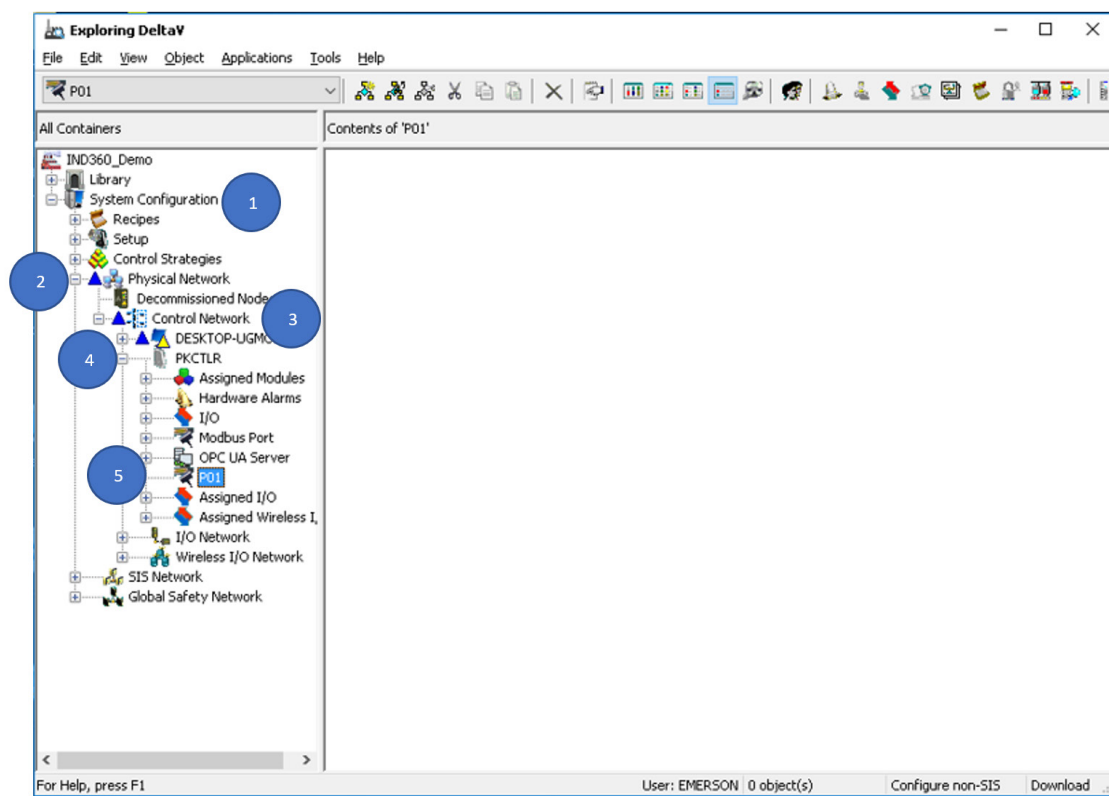


Figure 1-5: Using DeltaV Explorer

1.2.2. Configure P01 Ethernet IO Port

1. Navigate to P01 of PKCTRL – at **System Configuration > Physical Network > Control Network > PKCTRL > P01**. This node represents a port on the Ethernet I/O card or PK controller. The port contains physical devices, logical devices, and signals.
2. Rick-click the P01 node and select **Properties** from the context menu which displays.

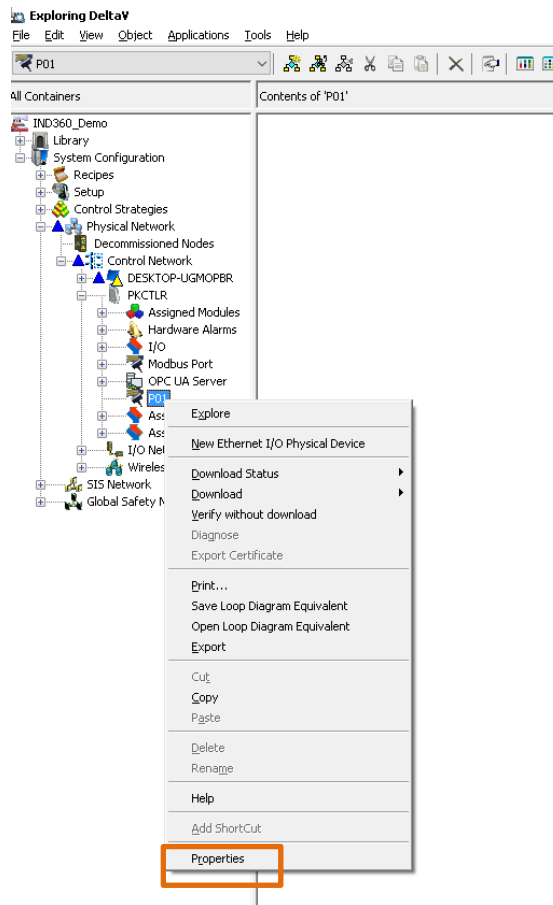


Figure 1-6: P01 Node Properties

3. Set the Interface EtherNet/IP Control Tag Integration. In the **Properties** dialog which appears, click the **General** tab.

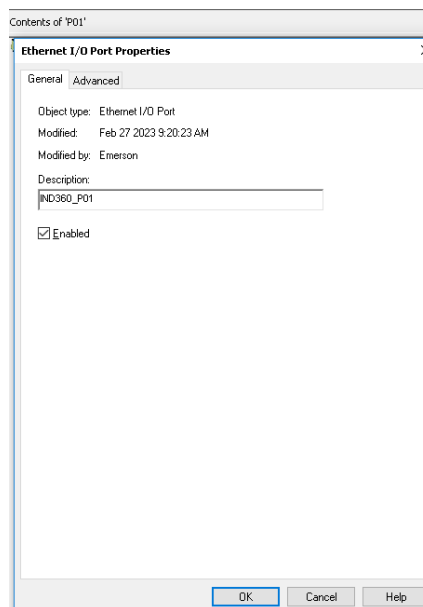


Figure 1-7: General Tab of Ethernet I/O Port Properties

Description	Enter text to describe this port. In the example above, IND360-P01 has been entered.
Enabled	Select this check box to enable the PK controller's Ethernet IO port. When enabled, the IP address specified on the Advanced tab (see below) is applied to the DeltaV PK Controller and any configured devices. When disabled, the port and any configured devices are not enabled, and IP addressing is not applied to the DeltaV PK Controller. In the disabled state, the OINTEG of the port does not contribute to the overall integrity of the DeltaV PK Controller.

4. In **Properties**, click the **Advanced** tab. Configure the interface protocol Ethernet/IP and the device networks used by the PK Controller to connect to the devices.

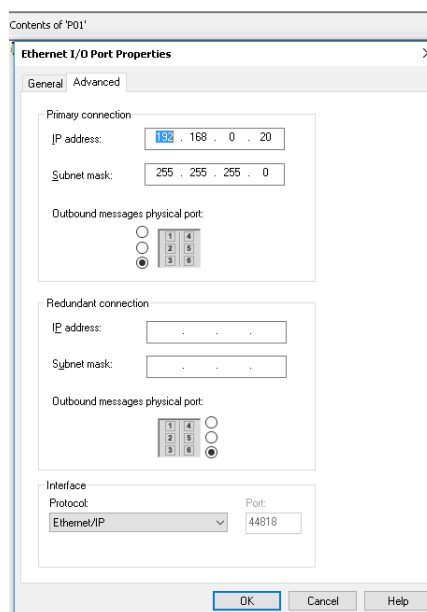


Figure 1-8: Advanced Tab of Ethernet I/O Port Properties

Primary Connection	This is the IP address and subnet mask on the primary device network used to connect to devices. Communication to devices connected to the IOP (the external Ethernet ports) is through this port, using the IP address entered here. Therefore, the IP address subnet entered here must match the subnet of the device network in use.
Outbound messages physical port	This binds the primary connection's IP address to this physical Ethernet port.
Redundant connection	The IP address and subnet mask on the secondary device network to connect to devices. Communication to devices connected to the IOP (the external Ethernet ports) is through this port using the IP address entered here. Therefore, the IP address subnet entered here must match the subnet of your device network
Outbound messages physical port	This binds the redundant connection's IP address to this physical Ethernet port.

Protocol

Use this drop-down list to specify the protocol to use with the configured Ethernet/IP.

1.2.3. Add a New Ethernet I/O Physical Device (PDT1)

1. Navigate to P01 at **System Configuration > Physical Network > Control Network > PKCTRL > P01**.
2. Right-click P01 and select **New Ethernet I/O Physical Device**.

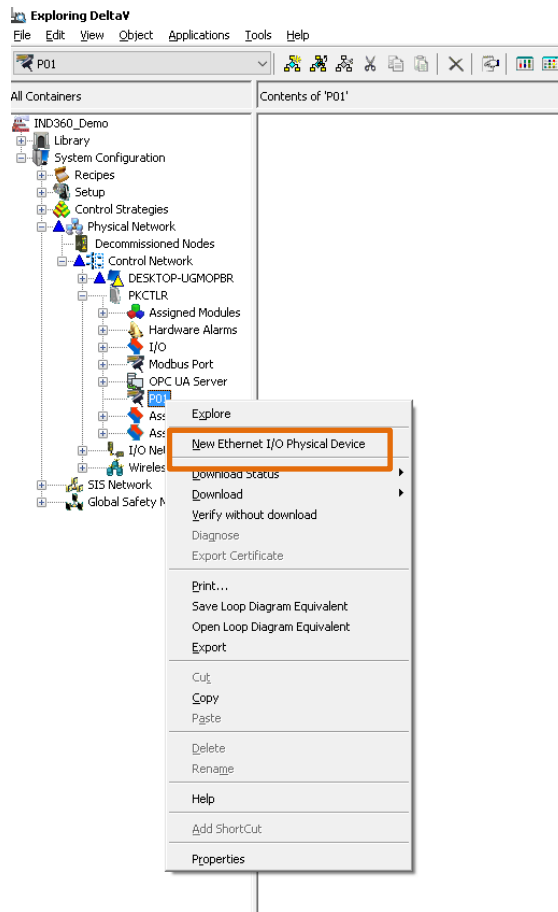


Figure 1-9: Adding an Ethernet I/O Physical Device

3. In the PDT1 **IND360_2BLOCK Properties** dialog, click the **General** tab. This tab provides details about the IND360 device. Use the **Description** field to enter identifying text to associate with the device.

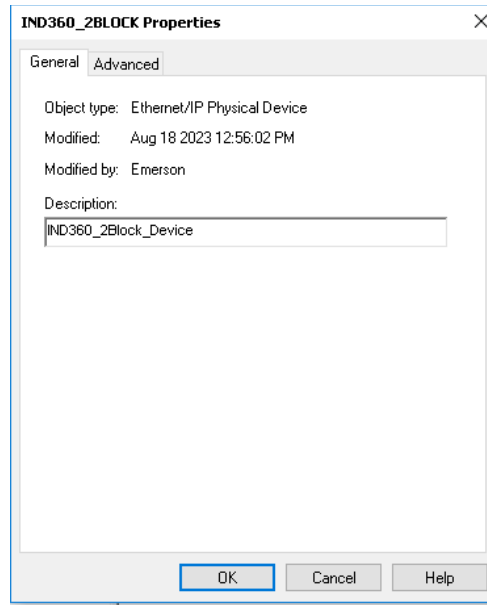


Figure 1-10: Physical Device Properties, General Tab

4. Click on the **Advanced** tab. The **Primary Connection** fields are used to configure the primary device network the IND360 uses to connect.

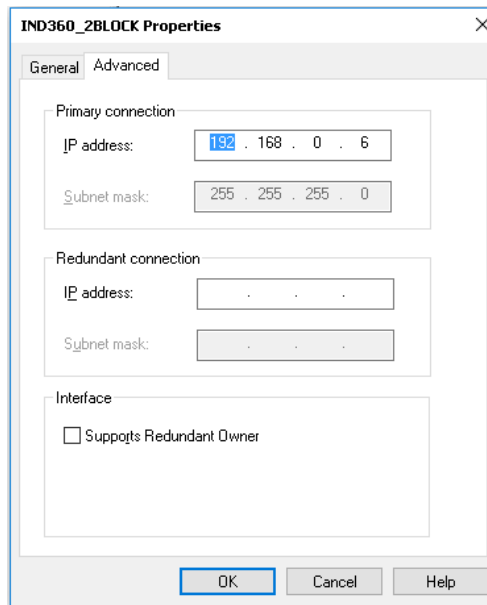


Figure 1-11: Physical Device Properties, Advanced Tab

5. Click **OK**. Now, in the DeltaV Explorer the new physical device will be displayed.

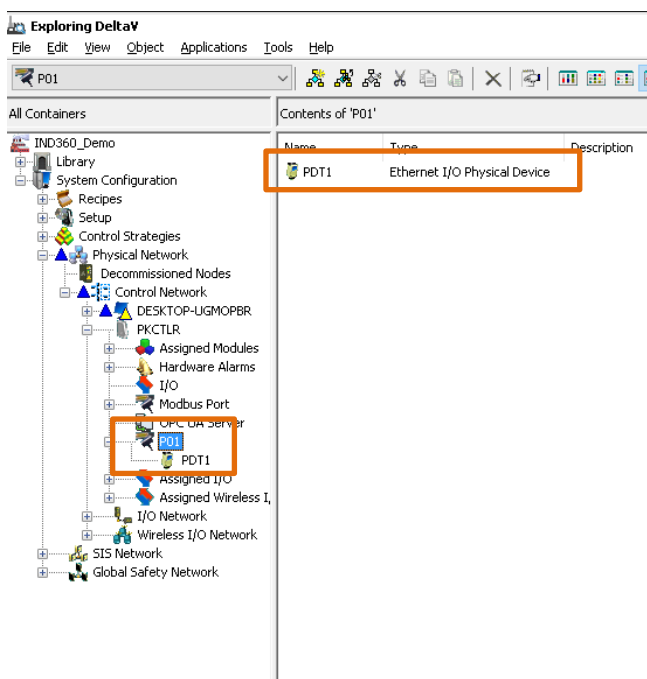


Figure 1-12: DeltaV Explorer Showing Newly-Configured Device

6. Under **P01**, Right-click on the **PDT1** and rename it as **IND360_2BLOCK**.

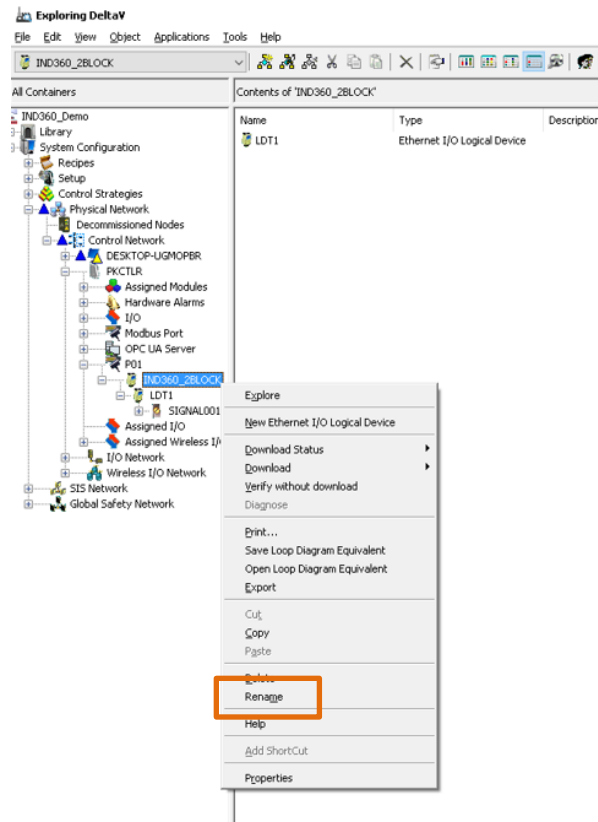


Figure 1-13: Renaming the Device Node

1.2.4. Configure IND360 for 2-Block Format

In this step, the IND360 is added as new Ethernet I/O logical device.

1. Navigate to PDT1 – **System Configuration > Physical Network > Control Network > PKCTRL > P01 > PDT1**.
2. Right-click PDT1 and select **New Ethernet I/O Logical Device**.

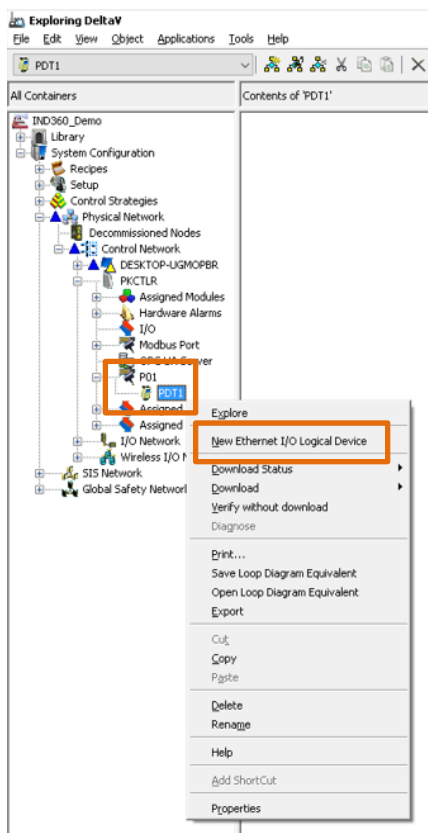


Figure 1-14: Adding a New Logical Device

3. In LDT1 (the IND360_2Block device) logical device properties, select the **General** tab. In the **Description** field, enter text to identify the device.

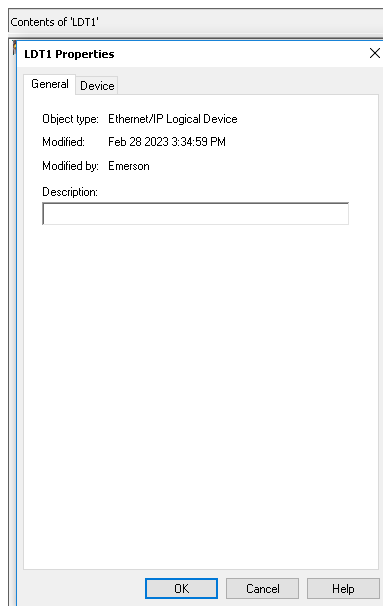


Figure 1-15: LDT1 Properties, General Tab

4. Now select the **Device** tab.

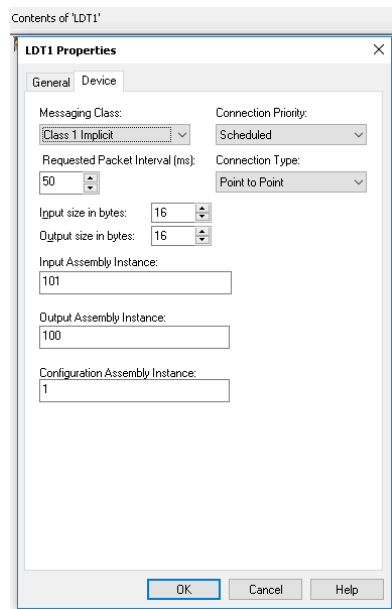


Figure 1-16: LDT1 Properties, Device Tab

Messaging Class	From the drop-down list, select Class 1 Implicit .
Input size in bytes	For Class 1 Implicit messaging, enter the total size of the input data to the device. Set a 2-block format (16 bytes) input size.
Output size in bytes	For Class 1 Implicit messaging, enter the total size of the output data from the device. Set a 2-block format (16 bytes) output size.
Requested packet interval (ms)	This defines the periodic update rate (in 50 millisecond increments) for reading the writing data in the device.
Connection Priority	For Class 1 Implicit messaging, this field determines the CIP connection priority for this device according to the Common Industrial Protocol specification. Accept the default value unless the target device requires a different value.
Connection Type	For Class 1 Implicit messaging, this field defines the transport type to use in the Target to Originator direction -- either Point to Point or Multicast .
Input Assembly Instance	For Class 1 Implicit messaging, enter the device-specific input assembly that this logical device reads and writes to. Input assembly Instances for IND360 is 101.
Output Assembly Instance	For Class 1 Implicit messaging, enter the device specific output assembly that this logical device reads and writes to. Output assembly Instances for IND360 is 100.
Configuration Assembly Instance	For Class 1 Implicit messaging, enter the device-specific configuration assembly that this logical device reads and writes to.

5. In the right pane of the DeltaV Explorer, the newly-created IND360 configuration will display.

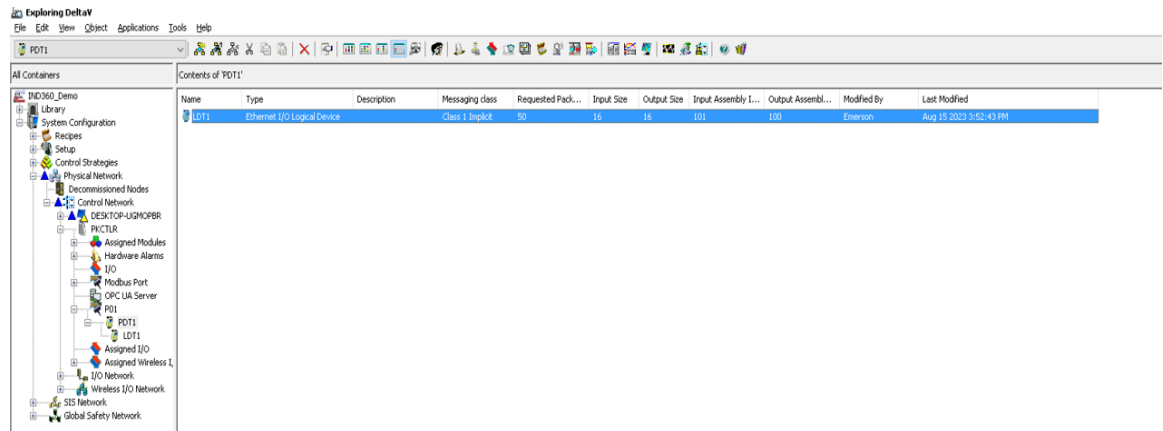


Figure 1-17: IND360 Logical Device Configuration Displayed in DeltaV Explorer

1.2.5. IND360 Memory Mapping for 2Block Format

Add controls data signals as specified for the 2 Block Format, indicated in the following tables:

IND360 2 Block	
Floating Point Data (Block 1)	
Word 0	Requested floating point value (32 Bit)
Word 1	
Word 2	16 Bit Scale status
Word 3	16 Bit Response
Status (Block 2)	
Word 0	Status Group 1
Word 1	Status Group 2
Word 2	Status Group 3
Word 3	Response

	Word Position	Field Name	Memory Position	Little Endian	Byte Swap	Word Swap	Big Endian	Byte Order Byte Count
Measuring Block	Word 0	Floating Point Word	M+0	Byte 0	Byte 1	Byte 2	Byte 3	4 Byte
			M+1	Byte 1	Byte 0	Byte 3	Byte 2	
	Word 1		M+2	Byte 2	Byte 3	Byte 0	Byte 1	
			M+3	Byte 3	Byte 2	Byte 1	Byte 0	
	Word 2	Scale Status	M+4	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+5	Byte 1	Byte 0	Byte 1	Byte 0	
	Word 3	Command Response	M+6	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+7	Byte 1	Byte 0	Byte 1	Byte 0	

	Word Position	Field Name	Memory Position	Little Endian	Byte Swap	Word Swap	Big Endian	Byte Order Byte Count
Status Block	Word 4	Status Group 1	M+8	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+9	Byte 1	Byte 0	Byte 1	Byte 0	
	Word 5	Status Group 2	M+10	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+11	Byte 1	Byte 0	Byte 1	Byte 0	
	Word 6	Status Group 3	M+12	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+13	Byte 1	Byte 0	Byte 1	Byte 0	
	Word 7	Response	M+14	Byte 0	Byte 1	Byte 0	Byte 1	2 Byte
			M+15	Byte 1	Byte 0	Byte 1	Byte 0	

SB1			
Status Group 1	16 bits, unsigned int	Input	Split into individual bits
Status Group 2	16 bits, unsigned int	Input	Split into individual bits
Status Group 3	16 bits, unsigned int	Input	Split into individual bits
Response	16 bits, signed int	Input	
Command	16 bits, signed int	Output	

1.2.6. Add New IND360 Ethernet I/O Signals

1. Navigate to LDT1 of the PKCTRL at **System Configuration > Physical Network > Control Network > PKCTRL > P01 > PDT1 > LDT1**.
2. Right-click on LDT1 and select **New Ethernet I/O Signal** from the context menu.

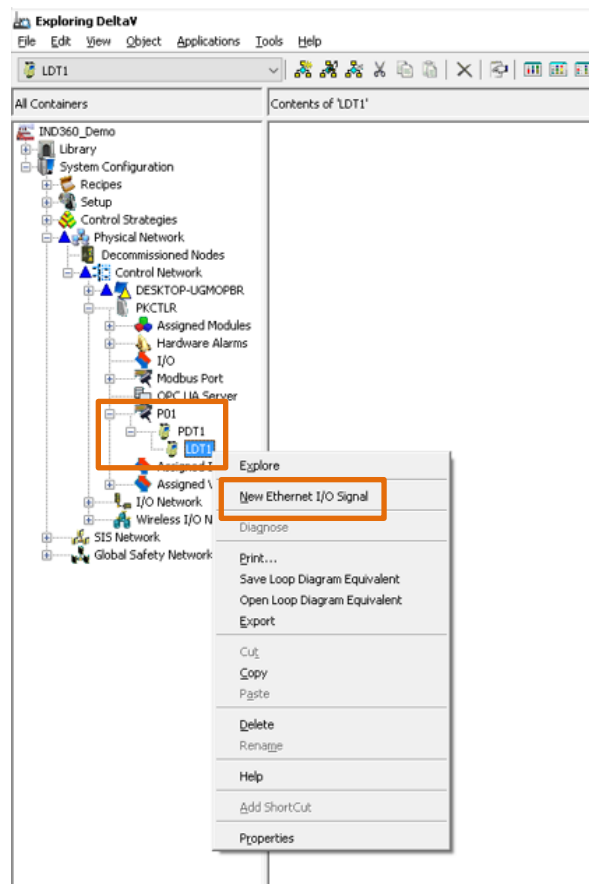


Figure 1-18: Creating a New Ethernet I/O

- Enter each signal tag data point information from the device document. The example shown below is a **Data OK** signal.

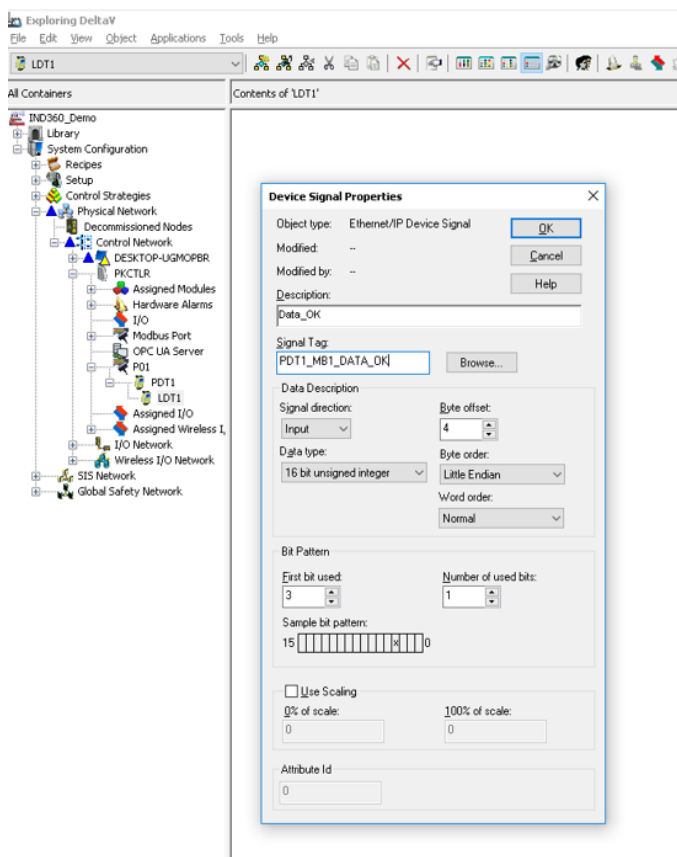


Figure 1-19: Configuring a Data OK Device Signal

Description	Enter a description of how this signal is used in the system.
Signal Tag	Enter the Device Signal Tag to associate with this signal. Click Browse to see if the tag is already created and available.
Signal direction	Selects either an input or output signal.
Byte offset	Selects the byte offset within the data group of the associated logical device. Refer to the IND360 device documentation for this information.
Data type	Selects the signal's data type.
Byte order	Selects the signal's byte order. Little Endian describes the order in which a sequence of bytes is stored in computer memory. With Little Endian, the little end or least significant value in the sequence is stored first.
Word order	Selects the signal's word order. Word order defines the order of the two 16-bit words within 32-bit data and has different meanings depending upon the protocol used. For example, 32-bit data in a physical device connected to Ethernet IO Card of 0x1234 5678.
Ethernet/IP	Word Order: Normal; read into the DeltaV system as 0x1234 5678.
First bit used	Enter the first bit for the DeltaV system to read from or write to for this signal. Refer to the IND360 device documentation for this information.

Number of used bits

Enter the number of bits that the DeltaV system reads from or writes to for this signal. Refer to the device documentation for this information.

- The figure below shows the first device signal “Data OK” created in both panes of the DeltaV Explorer window.

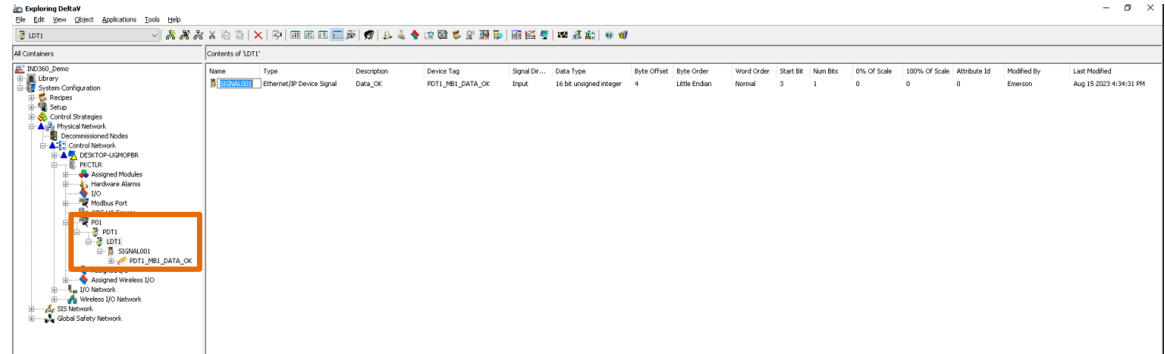


Figure 1-20: Device Signal Created

- Name the signal according to the data naming structure in use.
- Continue creating data point signals for the 2 Block format “Measuring Block” and “Status Block” SB1. The figures below show examples of Data OK, Heartbeat and Measuring Properties signals for the 2 Block format.

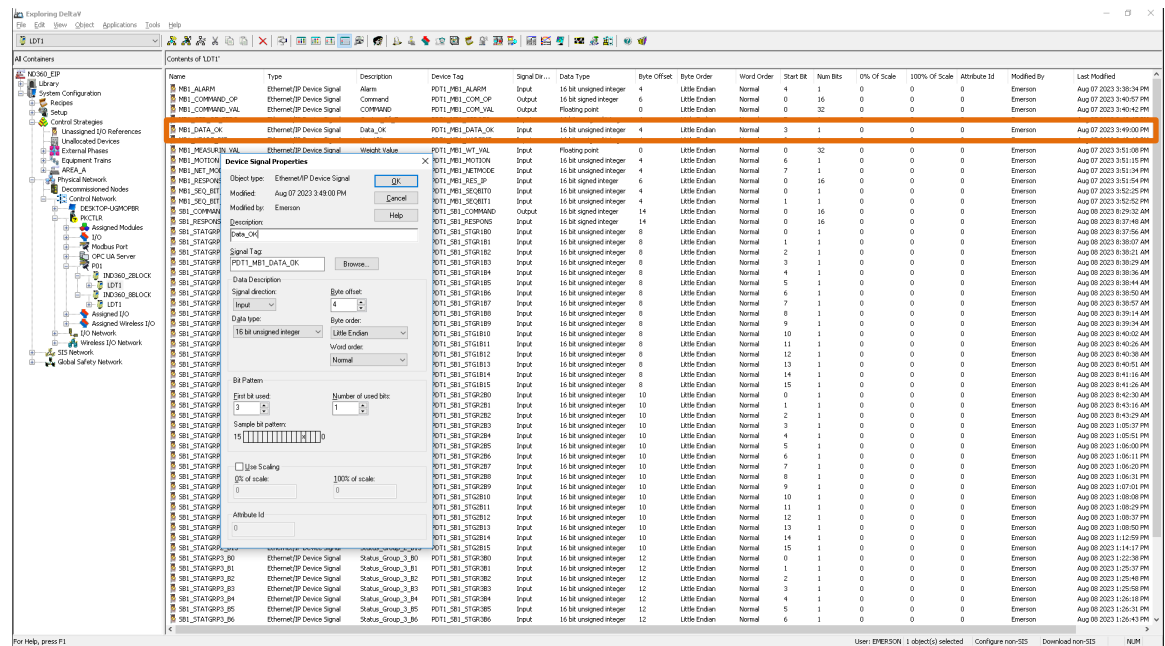


Figure 1-21: “Data OK” Signal Properties



The following three figures illustrate data mapping for the IND360 2-Block Format, with 62 device signals created.

[illegible]

Exploring DeltaV

File Edit View Object Applications Tools Help

LD1

All Containers

Contents of LD1

Name	Type	Description	Device Tag	Signal Dr...	Data Type	Byte Offset	Word Order	Start Bit	Num Bits	0% Of Scale	100% Of Scale	Attribute Id	Modified By	Last Modified
MEL_ALARM	Ethernet/IP Device Signal	Alarm	RT1_MEL_ALARM	Input	16 bit unsigned integer	4	Little Endian	Normal	4	1	0	0	Emerson	Aug 07 2023 3:38:24 PM
MEL_COMMAND_OP	Ethernet/IP Device Signal	Command	RT1_MEL_COMMAND_OP	Output	16 bit signed integer	6	Little Endian	Normal	0	16	0	0	Emerson	Aug 07 2023 3:40:57 PM
MEL_COMMAND_VAL	Ethernet/IP Device Signal	COMMAND	RT1_MEL_COMMAND_VAL	Output	Floating point	0	Little Endian	Normal	0	32	0	0	Emerson	Aug 07 2023 3:40:42 PM
MEL_CTR_OP_ZERO	Ethernet/IP Device Signal	Center_OP_Zero	RT1_MEL_CTR_OP	Input	16 bit unsigned integer	4	Little Endian	Normal	5	1	0	0	Emerson	Aug 07 2023 3:40:47 PM
MEL_DATA_OK	Ethernet/IP Device Signal	Data_OK	RT1_MEL_DATA_OK	Input	16 bit unsigned integer	4	Little Endian	Normal	3	1	0	0	Emerson	Aug 07 2023 3:40:00 PM
MEL_HEART_BTT	Ethernet/IP Device Signal	HeartBeat	RT1_MEL_HEARTBTT	Input	16 bit unsigned integer	4	Little Endian	Normal	2	1	0	0	Emerson	Aug 07 2023 3:40:40 PM
MEL_MEASURING_VAL	Ethernet/IP Device Signal	Weight Value	RT1_MEL_WT_VAL	Input	Floating point	0	Little Endian	Normal	0	32	0	0	Emerson	Aug 07 2023 3:55:08 PM
MEL_MOTION	Ethernet/IP Device Signal	Motion	RT1_MEL_MOTION	Input	16 bit unsigned integer	4	Little Endian	Normal	6	1	0	0	Emerson	Aug 07 2023 3:55:15 PM
MEL_NET_INDC	Ethernet/IP Device Signal	Net_Inode	RT1_MEL_NETWORK	Input	16 bit unsigned integer	4	Little Endian	Normal	7	1	0	0	Emerson	Aug 07 2023 3:55:14 PM
MEL_RESPONSE_IP	Ethernet/IP Device Signal	Response	RT1_MEL_IP	Input	16 bit signed integer	6	Little Endian	Normal	0	16	0	0	Emerson	Aug 07 2023 3:55:16 PM
MEL_SEQ_BIT_0	Ethernet/IP Device Signal	Sequence_BIT_0	RT1_MEL_SEQUENCE	Input	16 bit unsigned integer	4	Little Endian	Normal	0	1	0	0	Emerson	Aug 07 2023 3:52:25 PM
MEL_SEQ_BIT_1	Ethernet/IP Device Signal	Sequence_BIT_1	RT1_MEL_SEQUENCE	Input	16 bit unsigned integer	4	Little Endian	Normal	1	1	0	0	Emerson	Aug 07 2023 3:52:52 PM
MEL_COMMAND	Ethernet/IP Device Signal	Seq_Command	RT1_MEL_COMMAND	Output	16 bit signed integer	14	Little Endian	Normal	0	16	0	0	Emerson	Aug 07 2023 3:52:32 AM
MEL_RESPONSE	Ethernet/IP Device Signal	Seq_Response	RT1_MEL_SEQUENCE	Input	16 bit signed integer	14	Little Endian	Normal	0	16	0	0	Emerson	Aug 07 2023 3:57:48 AM
MEL_STATGPI_0	Ethernet/IP Device Signal	Status_Group_1_0	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	0	1	0	0	Emerson	Aug 07 2023 3:57:56 AM
MEL_STATGPI_1	Ethernet/IP Device Signal	Status_Group_1_1	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	1	1	0	0	Emerson	Aug 07 2023 3:58:07 AM
MEL_STATGPI_2	Ethernet/IP Device Signal	Status_Group_1_2	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	2	1	0	0	Emerson	Aug 07 2023 3:58:21 AM
MEL_STATGPI_3	Ethernet/IP Device Signal	Status_Group_1_3	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	3	1	0	0	Emerson	Aug 07 2023 3:58:29 AM
MEL_STATGPI_4	Ethernet/IP Device Signal	Status_Group_1_4	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	4	1	0	0	Emerson	Aug 07 2023 3:58:36 AM
MEL_STATGPI_5	Ethernet/IP Device Signal	Status_Group_1_5	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	5	1	0	0	Emerson	Aug 07 2023 3:58:44 AM
MEL_STATGPI_6	Ethernet/IP Device Signal	Status_Group_1_6	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	6	1	0	0	Emerson	Aug 07 2023 3:58:50 AM
MEL_STATGPI_7	Ethernet/IP Device Signal	Status_Group_1_7	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	7	1	0	0	Emerson	Aug 07 2023 3:58:57 AM
MEL_STATGPI_8	Ethernet/IP Device Signal	Status_Group_1_8	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	8	1	0	0	Emerson	Aug 07 2023 3:59:14 AM
MEL_STATGPI_9	Ethernet/IP Device Signal	Status_Group_1_9	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	9	1	0	0	Emerson	Aug 07 2023 3:59:24 AM
MEL_STATGPI_10	Ethernet/IP Device Signal	Status_Group_1_10	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	10	1	0	0	Emerson	Aug 07 2023 3:40:02 AM
MEL_STATGPI_11	Ethernet/IP Device Signal	Status_Group_1_11	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	11	1	0	0	Emerson	Aug 07 2023 3:40:26 AM
MEL_STATGPI_12	Ethernet/IP Device Signal	Status_Group_1_12	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	12	1	0	0	Emerson	Aug 07 2023 3:40:38 AM
MEL_STATGPI_13	Ethernet/IP Device Signal	Status_Group_1_13	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	13	1	0	0	Emerson	Aug 07 2023 3:40:51 AM
MEL_STATGPI_14	Ethernet/IP Device Signal	Status_Group_1_14	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	14	1	0	0	Emerson	Aug 07 2023 3:41:16 AM
MEL_STATGPI_15	Ethernet/IP Device Signal	Status_Group_1_15	RT1_MEL_STATGPI	Input	16 bit unsigned integer	8	Little Endian	Normal	15	1	0	0	Emerson	Aug 07 2023 3:41:26 AM
MEL_STATGPI_16	Ethernet/IP Device Signal	Status_Group_2_0	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	0	1	0	0	Emerson	Aug 07 2023 3:42:30 AM
MEL_STATGPI_17	Ethernet/IP Device Signal	Status_Group_2_1	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	1	1	0	0	Emerson	Aug 07 2023 3:43:16 AM
MEL_STATGPI_18	Ethernet/IP Device Signal	Status_Group_2_2	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	2	1	0	0	Emerson	Aug 07 2023 3:43:29 AM
MEL_STATGPI_19	Ethernet/IP Device Signal	Status_Group_2_3	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	3	1	0	0	Emerson	Aug 07 2023 3:43:37 PM
MEL_STATGPI_20	Ethernet/IP Device Signal	Status_Group_2_4	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	4	1	0	0	Emerson	Aug 07 2023 3:43:51 PM
MEL_STATGPI_21	Ethernet/IP Device Signal	Status_Group_2_5	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	5	1	0	0	Emerson	Aug 07 2023 3:44:00 PM
MEL_STATGPI_22	Ethernet/IP Device Signal	Status_Group_2_6	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	6	1	0	0	Emerson	Aug 07 2023 3:44:11 PM
MEL_STATGPI_23	Ethernet/IP Device Signal	Status_Group_2_7	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	7	1	0	0	Emerson	Aug 07 2023 3:44:20 PM
MEL_STATGPI_24	Ethernet/IP Device Signal	Status_Group_2_8	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	8	1	0	0	Emerson	Aug 07 2023 3:44:31 PM
MEL_STATGPI_25	Ethernet/IP Device Signal	Status_Group_2_9	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	9	1	0	0	Emerson	Aug 07 2023 3:44:51 PM
MEL_STATGPI_26	Ethernet/IP Device Signal	Status_Group_2_10	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	10	1	0	0	Emerson	Aug 07 2023 3:45:08 PM
MEL_STATGPI_27	Ethernet/IP Device Signal	Status_Group_2_11	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	11	1	0	0	Emerson	Aug 07 2023 3:45:29 PM
MEL_STATGPI_28	Ethernet/IP Device Signal	Status_Group_2_12	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	12	1	0	0	Emerson	Aug 07 2023 3:45:37 PM
MEL_STATGPI_29	Ethernet/IP Device Signal	Status_Group_2_13	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	13	1	0	0	Emerson	Aug 07 2023 3:45:50 PM
MEL_STATGPI_30	Ethernet/IP Device Signal	Status_Group_2_14	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	14	1	0	0	Emerson	Aug 07 2023 3:45:59 PM
MEL_STATGPI_31	Ethernet/IP Device Signal	Status_Group_2_15	RT1_MEL_STATGPI	Input	16 bit unsigned integer	10	Little Endian	Normal	15	1	0	0	Emerson	Aug 07 2023 3:46:17 PM
MEL_STATGPI_32	Ethernet/IP Device Signal	Status_Group_3_0	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	0	1	0	0	Emerson	Aug 07 2023 3:46:38 PM
MEL_STATGPI_33	Ethernet/IP Device Signal	Status_Group_3_1	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	1	1	0	0	Emerson	Aug 07 2023 3:46:57 PM
MEL_STATGPI_34	Ethernet/IP Device Signal	Status_Group_3_2	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	2	1	0	0	Emerson	Aug 07 2023 3:47:08 PM
MEL_STATGPI_35	Ethernet/IP Device Signal	Status_Group_3_3	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	3	1	0	0	Emerson	Aug 07 2023 3:47:19 PM
MEL_STATGPI_36	Ethernet/IP Device Signal	Status_Group_3_4	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	4	1	0	0	Emerson	Aug 07 2023 3:47:31 PM
MEL_STATGPI_37	Ethernet/IP Device Signal	Status_Group_3_5	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	5	1	0	0	Emerson	Aug 07 2023 3:47:43 PM
MEL_STATGPI_38	Ethernet/IP Device Signal	Status_Group_3_6	RT1_MEL_STATGPI	Input	16 bit unsigned integer	12	Little Endian	Normal	6	1	0	0	Emerson	Aug 07 2023 3:47:54 PM

For Help, press F1

User: EMERSON 62-object(s) Configure non-SES Download non-SES N/A

1.2.7. IND360 Memory Mapping for 8 Block Format

Add a new Ethernet I/O Physical device (PDT2) in the same way PDT1 was added.

1. Navigate to P01 of the PKCTRL at **System Configuration > Physical Network > Control Network > PKCTRL > P01**.
2. Right-click on P01 and select **New Ethernet I/O Physical Device** from the context menu.

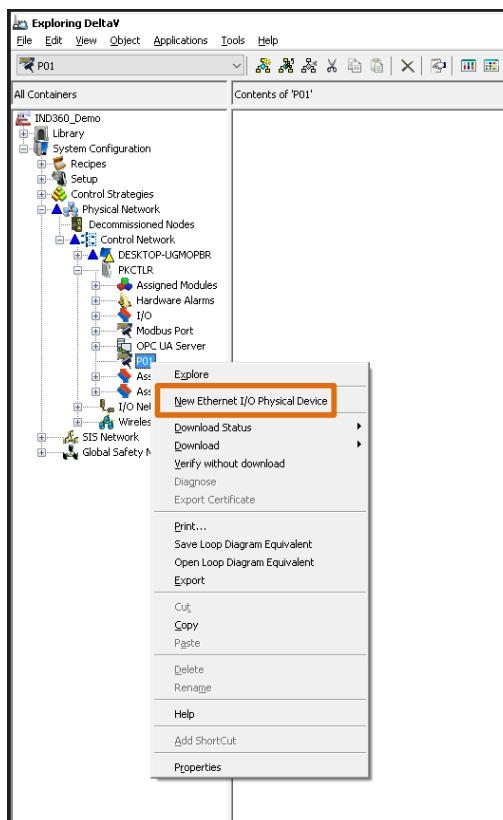


Figure 1-24: Adding an I/O Physical Device

- Right click to access the PDT2 (IND360_8Block) Physical Device **Properties** dialog and select the **General** Tab. Enter text to describe the device in the **Description** field.

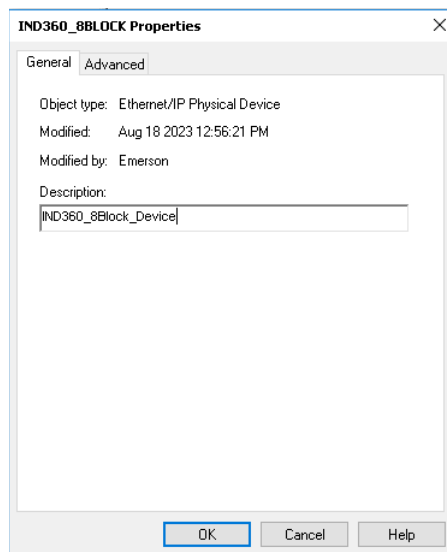


Figure 1-25: IND360_8Block Properties, General Tab

- Now click the **Advanced** tab. Configure the **Primary connection** with IP address and subnet mask on the primary device connection used by the IND360 to connect.

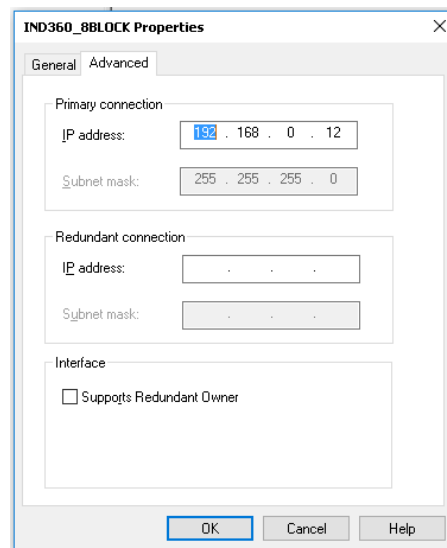


Figure 1-26: IND360_8Block Properties, Advanced Tab

1.2.8. Configure 8 Block Format Settings

- Navigate to the PDT2 (IND360_8Block) of the controller, at **Configuration > physical network > control network > PKCTRL > P01 > PDT2**.
- Right-click **PDT2** and select **New Ethernet Logical Device** from the context menu.

3. Rename PDT2 as IND360_8Block.

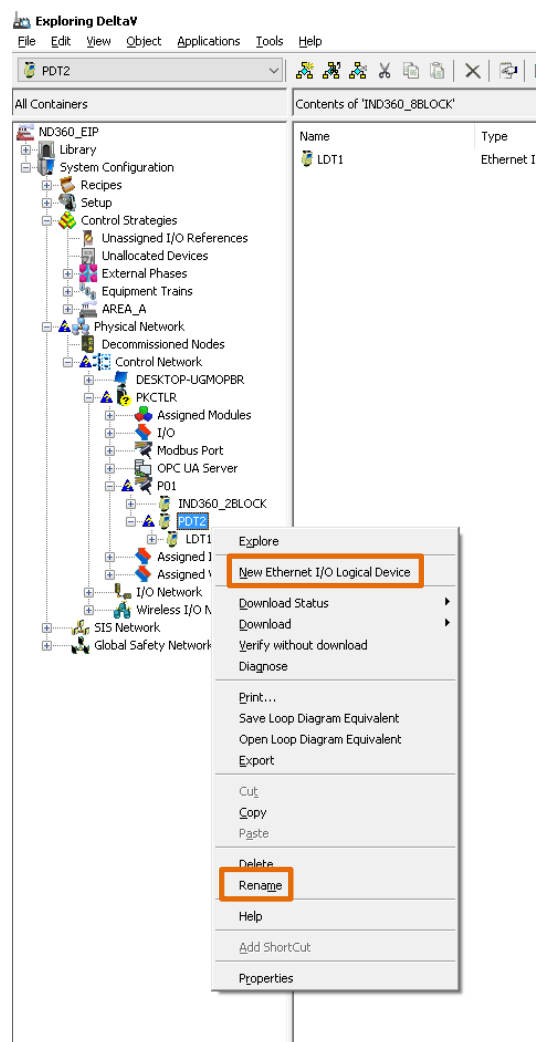


Figure 1-27: Creating a New Ethernet I/O Logical Device

4. In LDT1 (IND360_8Block) Logical Device Properties, select the **General** tab. This tab provides details about the last modification of the logical device, and includes a **Description** field in which text to describe the device can be entered.

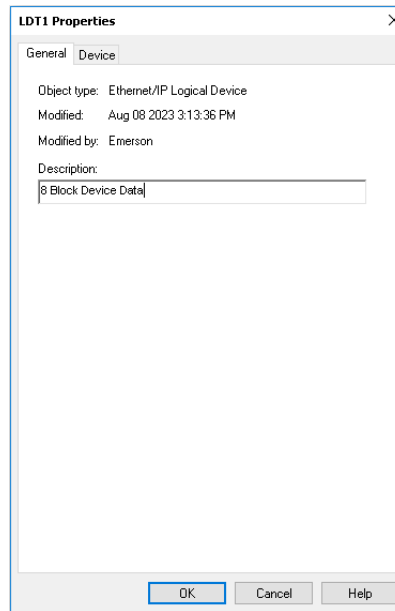


Figure 1-28: LDT1 Properties, General Tab

5. Now select the **Device** tab.

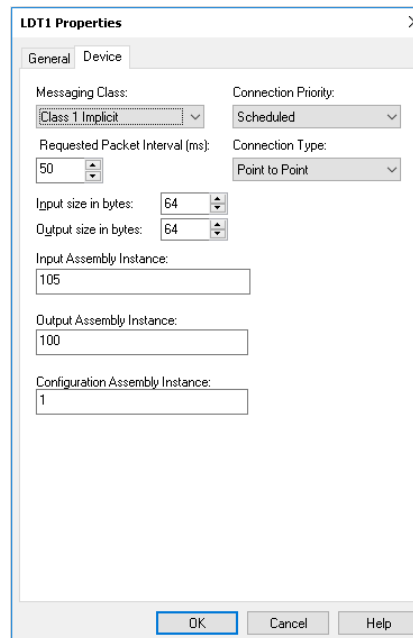


Figure 1-29: LDT1 Properties, Device Tab

Messaging Class

From the drop-down list, select **Class 1 Implicit**.

Input size in bytes

For Class 1 Implicit messaging, enter the total size of the input data to the device. Set a 8-block format (64 bytes) input size.

Output size in bytes

For Class 1 Implicit messaging, enter the total size of the output data from the device. Set a 8-block format (64 bytes) output size.

Requested packet interval (ms)	This defines the periodic update rate (in 50 millisecond increments) for reading the writing data in the device.
Connection Priority	For Class 1 Implicit messaging, this field determines the CIP connection priority for this device according to the Common Industrial Protocol specification. Accept the default value unless the target device requires a different value.
Connection Type	For Class 1 Implicit messaging, this field defines the transport type to use in the Target to Originator direction -- either Point to Point or Multicast .
Input Assembly Instance	For Class 1 Implicit messaging, enter the device-specific input assembly that this logical device reads and writes to. Input assembly Instances for IND360 is 105.
Output Assembly Instance	For Class 1 Implicit messaging, enter the device specific output assembly that this logical device reads and writes to. Output assembly Instances for IND360 is 100.
Configuration Assembly Instance	For Class 1 Implicit messaging, enter the device-specific configuration assembly that this logical device reads and writes to.

1.2.9. IND360 Memory Mapping of 8Block Format

Add controls data signals as specified for the 8 Block Format, indicated in the following table:

IND360 8Block		Floating Point Data (Block 5)	
Floating Point Data (Block 1)		Word 16	Requested floating point value (32 Bit)
Word 0	Requested floating point value (32 Bit)	Word 17	Requested floating point value (32 Bit)
Word 1		Word 18	16 Bit Scale status
Word 2	16 Bit Scale status	Word 19	16 Bit Response
Word 3	16 Bit Response		
Status (Block 2)		Floating Point Data (Block 6)	
Word 0	Status Group 1	Word 16	Requested floating point value (32 Bit)
Word 1	Status Group 2	Word 17	Requested floating point value (32 Bit)
Word 2	Status Group 3	Word 18	16 Bit Scale status
Word 3	Response	Word 19	16 Bit Response
Floating Point Data (Block 3)		Floating Point Data (Block 7)	
Word 8	Requested floating point value (32 Bit)	Word 16	Requested floating point value (32 Bit)
Word 9	Requested floating point value (32 Bit)	Word 17	Requested floating point value (32 Bit)
Word 10	16 Bit Scale status	Word 18	16 Bit Scale status
Word 11	16 Bit Response	Word 19	16 Bit Response
Floating Point Data (Block 4)		Floating Point Data (Block 8)	
Word 16	Requested floating point value (32 Bit)	Word 16	Requested floating point value (32 Bit)
Word 17	Requested floating point value (32 Bit)	Word 17	Requested floating point value (32 Bit)
Word 18	16 Bit Scale status	Word 18	16 Bit Scale status
Word 19	16 Bit Response	Word 19	16 Bit Response

- An LDT FHX file exported from an existing project can be imported. Refer to section 1.4, Export/Import Signals, on page 1-28.

Data Mapping for the IND360 8-Block Format

Exploiting IoT1

FileViewOpenCloseApplicationsToolsHelp

LOTT1

Contents of LOTT1

	Name	Type	Description	Device Tag	Signal Dir.	Data Type	Byte Offset	Byte Order	Word Order	Start Bit	Num Bits	0% of Scale	100% of Scale	Attributes	Modified By	Last Modified
Library	MSI_ALARM	Ethernet Device Signal	Alarm	POT2_MSI_ALARM	Input	16-bit unsigned integer	4	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:18 PM
	MSI_COMMAND	Ethernet Device Signal	Command	POT2_MSI_COMMAND	Output	16-bit unsigned integer	6	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:42 PM
	MSI_COMMAND_VAL	Ethernet Device Signal	Command	POT2_MSI_COMMAND	Output	Floating point	4	Little Endian	Normal	0	32	0	0	0	Emerson	Aug 08 2023 13:42 PM
	MSI_CTR_ZERO	Ethernet Device Signal	Center_Ctr_Zero	POT2_MSI_CTR_ZERO	Input	16-bit unsigned integer	4	Little Endian	Normal	5	1	0	0	0	Emerson	Aug 08 2023 13:59 PM
	MSI_DATA_OK	Ethernet Device Signal	Data_OK	POT2_MSI_DATA_OK	Input	16-bit unsigned integer	4	Little Endian	Normal	3	1	0	0	0	Emerson	Aug 08 2023 10:41 PM
	MSI_HEART_BTT	Ethernet Device Signal	HeartBtt	POT2_MSI_HEART_BTT	Input	16-bit unsigned integer	4	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:40 PM
	MSI_MEASUR_VAL	Ethernet Device Signal	Weight Value	POT2_MSI_MEASUR_VAL	Input	Floating point	0	Little Endian	Normal	0	32	0	0	0	Emerson	Aug 08 2023 13:46 PM
	MSI_PROTON	Ethernet Device Signal	Proton	POT2_MSI_PROTON	Input	16-bit unsigned integer	4	Little Endian	Normal	6	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_NET_IPCODE	Ethernet Device Signal	Net_IPCode	POT2_MSI_NET_IPCODE	Input	16-bit unsigned integer	4	Little Endian	Normal	7	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Response	POT2_MSI_RES_IP	Input	16-bit signed integer	6	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:54 PM
Physical Network	MSI_SEQ_BTT_0	Ethernet Device Signal	Sequence_BTT_0	POT2_MSI_SEQ_BTT_0	Input	16-bit unsigned integer	4	Little Endian	Normal	0	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_SEQ_BTT_1	Ethernet Device Signal	Sequence_BTT_1	POT2_MSI_SEQ_BTT_1	Input	16-bit unsigned integer	4	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_ALARM	Ethernet Device Signal	Alarm	MSI_ALARM	Input	16-bit unsigned integer	20	Little Endian	Normal	4	1	0	0	0	Emerson	Aug 08 2023 13:36 PM
	MSI_CTR_ZERO	Ethernet Device Signal	Center_Ctr_Zero	POT2_MSI_CTR_ZERO	Input	16-bit unsigned integer	20	Little Endian	Normal	5	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_DATA_OK	Ethernet Device Signal	Data_OK	POT2_MSI_DATA_OK	Input	16-bit unsigned integer	20	Little Endian	Normal	3	1	0	0	0	Emerson	Aug 08 2023 10:20 PM
	MSI_HEART_BTT	Ethernet Device Signal	HeartBtt	POT2_MSI_HEART_BTT	Input	16-bit unsigned integer	20	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:40 PM
	MSI_MEASUR_VAL	Ethernet Device Signal	Weight Value	POT2_MSI_MEASUR_VAL	Input	Floating point	16	Little Endian	Normal	0	32	0	0	0	Emerson	Aug 08 2023 13:42 PM
	MSI_NET_IPCODE	Ethernet Device Signal	Net_IPCode	POT2_MSI_NET_IPCODE	Input	16-bit unsigned integer	20	Little Endian	Normal	7	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Response	POT2_MSI_RES_IP	Input	16-bit signed integer	22	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:54 PM
	MSI_SEQ_BTT_0	Ethernet Device Signal	Sequence_BTT_0	POT2_MSI_SEQ_BTT_0	Input	16-bit unsigned integer	20	Little Endian	Normal	0	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
IO Network	MSI_SEQ_BTT_1	Ethernet Device Signal	Sequence_BTT_1	POT2_MSI_SEQ_BTT_1	Input	16-bit unsigned integer	20	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_ALARM	Ethernet Device Signal	Alarm	POT2_MSI_ALARM	Input	16-bit unsigned integer	28	Little Endian	Normal	4	1	0	0	0	Emerson	Aug 08 2023 13:36 PM
	MSI_CTR_ZERO	Ethernet Device Signal	Center_Ctr_Zero	POT2_MSI_CTR_ZERO	Input	16-bit signed integer	30	Little Endian	Normal	6	16	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_DATA_OK	Ethernet Device Signal	Data_OK	POT2_MSI_DATA_OK	Input	16-bit unsigned integer	28	Little Endian	Normal	3	1	0	0	0	Emerson	Aug 08 2023 13:29 PM
Physical Network	MSI_HEART_BTT	Ethernet Device Signal	HeartBtt	POT2_MSI_HEART_BTT	Input	16-bit unsigned integer	28	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:40 PM
	MSI_MEASUR_VAL	Ethernet Device Signal	Weight Value	POT2_MSI_MEASUR_VAL	Input	Floating point	24	Little Endian	Normal	0	32	0	0	0	Emerson	Aug 08 2023 13:42 PM
	MSI_PROTON	Ethernet Device Signal	Proton	POT2_MSI_PROTON	Input	16-bit unsigned integer	28	Little Endian	Normal	6	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_NET_IPCODE	Ethernet Device Signal	Net_IPCode	POT2_MSI_NET_IPCODE	Input	16-bit unsigned integer	28	Little Endian	Normal	7	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Response	POT2_MSI_RES_IP	Input	16-bit signed integer	30	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:54 PM
	MSI_SEQ_BTT_0	Ethernet Device Signal	Sequence_BTT_0	POT2_MSI_SEQ_BTT_0	Input	16-bit unsigned integer	28	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_SEQ_BTT_1	Ethernet Device Signal	Sequence_BTT_1	POT2_MSI_SEQ_BTT_1	Input	16-bit unsigned integer	28	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_ALARM	Ethernet Device Signal	Alarm	POT2_MSI_ALARM	Input	16-bit signed integer	36	Little Endian	Normal	4	1	0	0	0	Emerson	Aug 08 2023 13:29 PM
	MSI_COMMAND	Ethernet Device Signal	Command	POT2_MSI_COMMAND	Output	16-bit signed integer	36	Little Endian	Normal	5	1	0	0	0	Emerson	Aug 08 2023 13:42 PM
	MSI_CTR_ZERO	Ethernet Device Signal	Center_Ctr_Zero	POT2_MSI_CTR_ZERO	Input	16-bit unsigned integer	36	Little Endian	Normal	5	1	0	0	0	Emerson	Aug 08 2023 13:59 PM
Physical Network	MSI_DATA_OK	Ethernet Device Signal	Data_OK	POT2_MSI_DATA_OK	Input	16-bit unsigned integer	36	Little Endian	Normal	3	1	0	0	0	Emerson	Aug 08 2023 10:41 PM
	MSI_HEART_BTT	Ethernet Device Signal	HeartBtt	POT2_MSI_HEART_BTT	Input	16-bit unsigned integer	36	Little Endian	Normal	2	1	0	0	0	Emerson	Aug 08 2023 13:40 PM
	MSI_MEASUR_VAL	Ethernet Device Signal	Weight Value	POT2_MSI_MEASUR_VAL	Input	Floating point	32	Little Endian	Normal	0	32	0	0	0	Emerson	Aug 08 2023 13:46 PM
	MSI_PROTON	Ethernet Device Signal	Proton	POT2_MSI_PROTON	Input	16-bit unsigned integer	36	Little Endian	Normal	6	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_NET_IPCODE	Ethernet Device Signal	Net_IPCode	POT2_MSI_NET_IPCODE	Input	16-bit unsigned integer	36	Little Endian	Normal	7	1	0	0	0	Emerson	Aug 08 2023 13:51 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Response	POT2_MSI_RES_IP	Input	16-bit signed integer	38	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:54 PM
	MSI_SEQ_BTT_0	Ethernet Device Signal	Sequence_BTT_0	POT2_MSI_SEQ_BTT_0	Input	16-bit unsigned integer	36	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_SEQ_BTT_1	Ethernet Device Signal	Sequence_BTT_1	POT2_MSI_SEQ_BTT_1	Input	16-bit unsigned integer	41	Little Endian	Normal	7	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Response	MSI_RESPONSE_IP	Input	16-bit signed integer	46	Little Endian	Normal	0	16	0	0	0	Emerson	Aug 08 2023 13:54 PM
	MSI_SEQ_BTT_0	Ethernet Device Signal	Sequence_BTT_0	MSI_SEQ_BTT_0	Input	16-bit unsigned integer	44	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
IO Network	MSI_SEQ_BTT_1	Ethernet Device Signal	Sequence_BTT_1	MSI_SEQ_BTT_1	Input	16-bit unsigned integer	44	Little Endian	Normal	1	1	0	0	0	Emerson	Aug 08 2023 13:55 PM
	MSI_COMMAND	Ethernet Device Signal	Seq_Command	MSI_COMMAND	Output	16-bit signed integer	14	Little Endian	Normal	0	16	0	0	0	DelSysNet	Jul 29 2019 11:42 PM
	MSI_RESPONSE_IP	Ethernet Device Signal	Seq_Response	MSI_RESPONSE_IP	Input	16-bit signed integer	14	Little Endian	Normal	0	16	0	0	0	DelSysNet	Jul 29 2019 11:42 PM
	MSI_STATISTICS_00	Ethernet Device Signal	Status_Group_1_00	STATISTICS_GRP_1_00	Input	16-bit unsigned integer	8	Little Endian	Normal	0	1	0	0	0	DelSysNet	Jul 29 2019 11:42 PM

FileViewOpenCloseApplicationsToolsHelp

For Help, press F1

User: EMPSR01 1 object(s) selected

Configure new SS

Download new SS

N/A

File Explorer

OpenCloseRefreshBackForwardHomeAddress BarSearchViewToolsHelp

Content Area

Contents of 'LOT1'

MONOP

System Configuration

Routines

Setpoints

Control Strategies

Unassigned I/O References

IO Modules

External Phases

Equipment Trans

Physical Network

Control Network

Detector Output

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

IO Modules

Download the configuration to the PK Controller. To download, go to **System Configuration > Physical Network > Control Network > PKCTRL > Download > PK Controller**.

■ To export or import signals, refer to section 1.4, Export/Import Signals, on page 1-28.

1.3. Delta V Diagnostics: Overall Integrity

To check the overall integrity of the devices, access Delta V Diagnostics:

1. Navigate to P01 in DeltaV Explorer – at **System Configuration > Physical Network > Control Network > PKCTLR > P01**.
2. Right-click **P01** and select **Diagnose**.

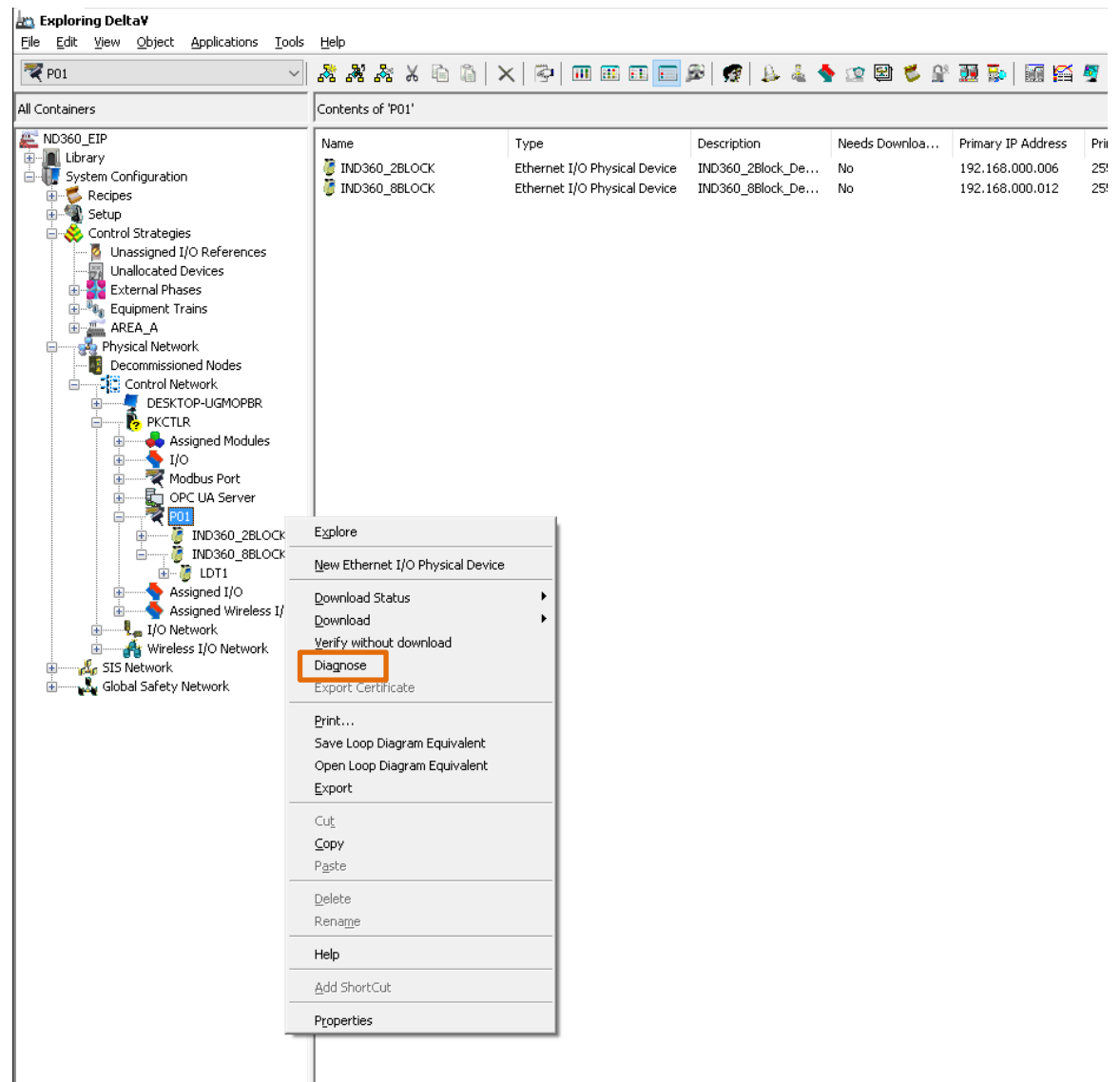


Figure 1-30: Starting DeltaV Diagnostics

3. The DeltaV **Diagnostics** window will open, displaying the contents of **P01**.

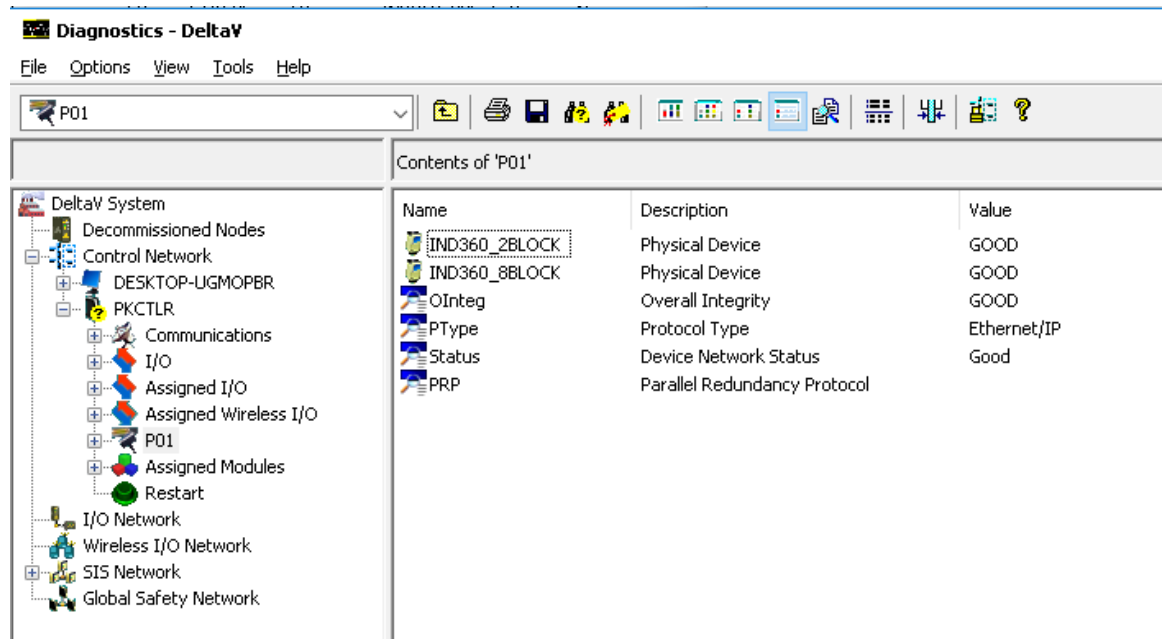


Figure 1-31: DeltaV Diagnostics Window

1.3.1. DeltaV Diagnostics – Device Data Value

To view the device data value:

1. Navigate to LDT1 in DeltaV Diagnostics – at **Control Network > PKCTRL > P01 > IND360_2Block > LDT1**.
2. Right-click LDT1 and select **View Device Data**.

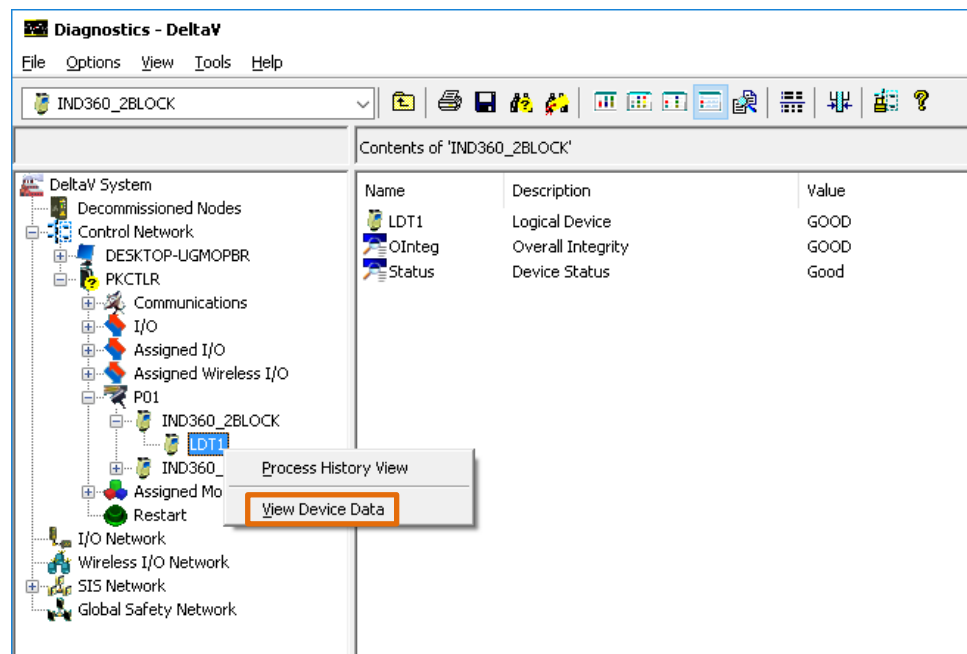


Figure 1-32: Viewing Device Data

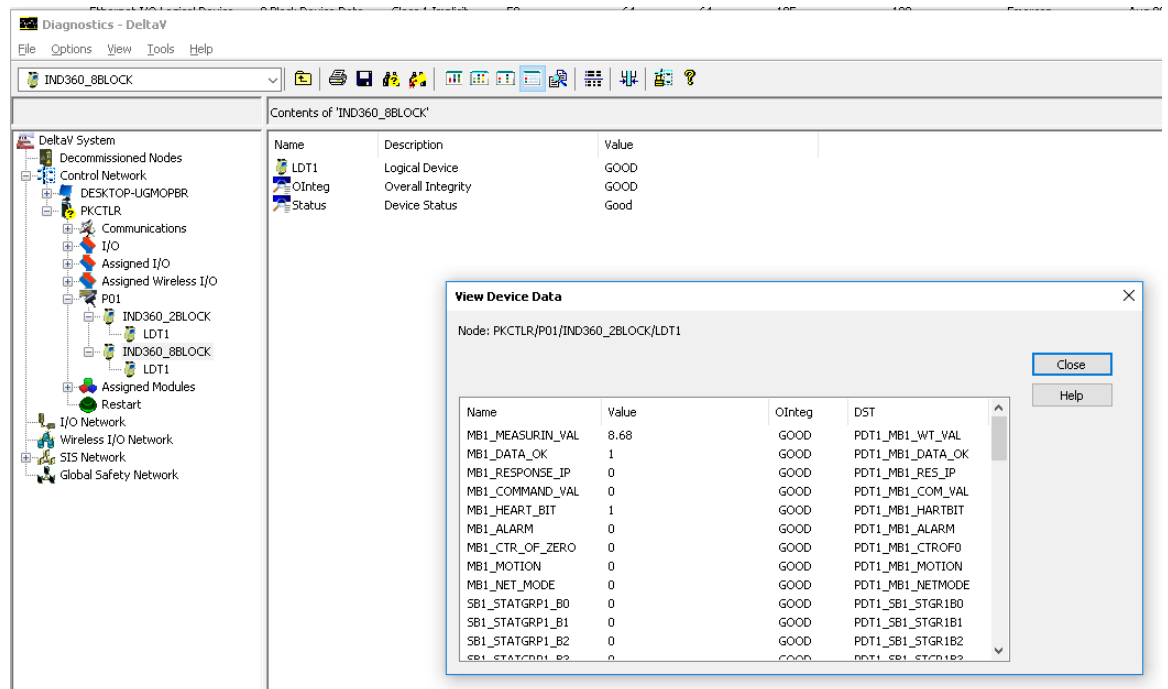


Figure 1-33: Viewing Device Data for IND360 2-Block Format

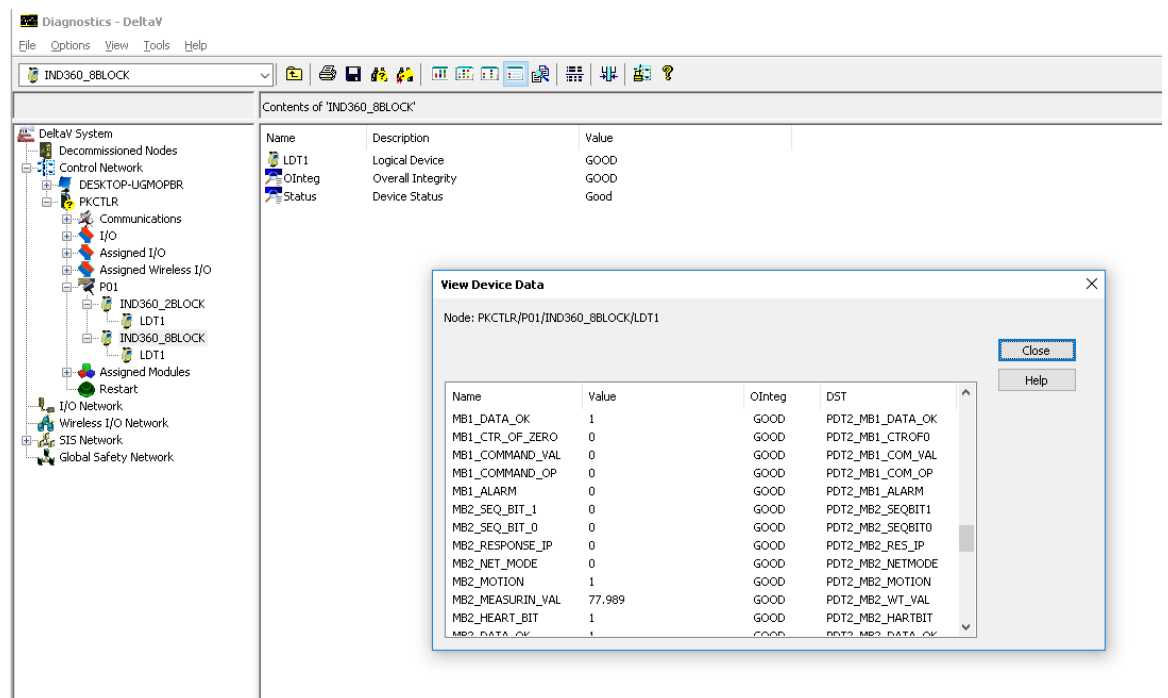
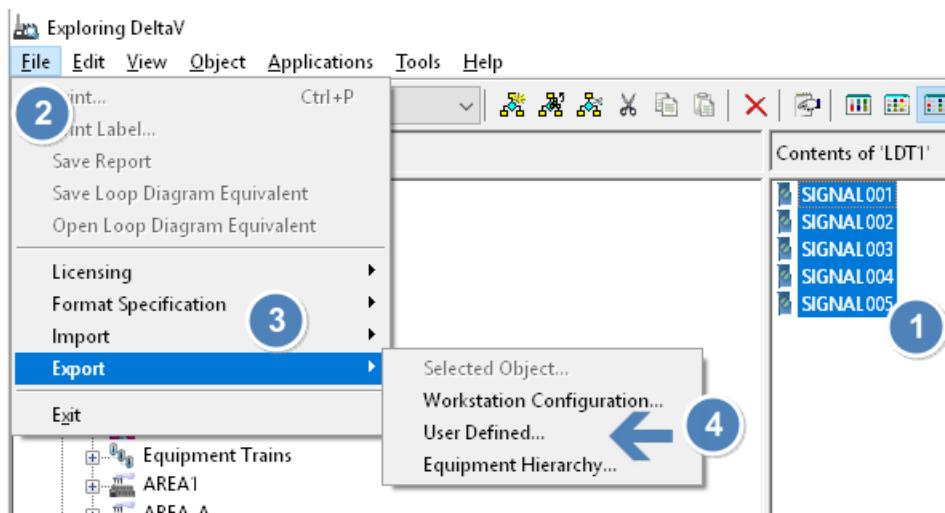


Figure 1-34: Viewing Device Data for IND360 8-Block Format

1.4. Export/Import Signals

1. In DeltaV Explorer, select all the signals to be exported [1], then access **File** [2] | **Export** [3] | **User Defined...** [4], to create a text file that can be imported into the bulk edit Excel add-in.



2. The **User Defined Export** window will display. Click **Next**.

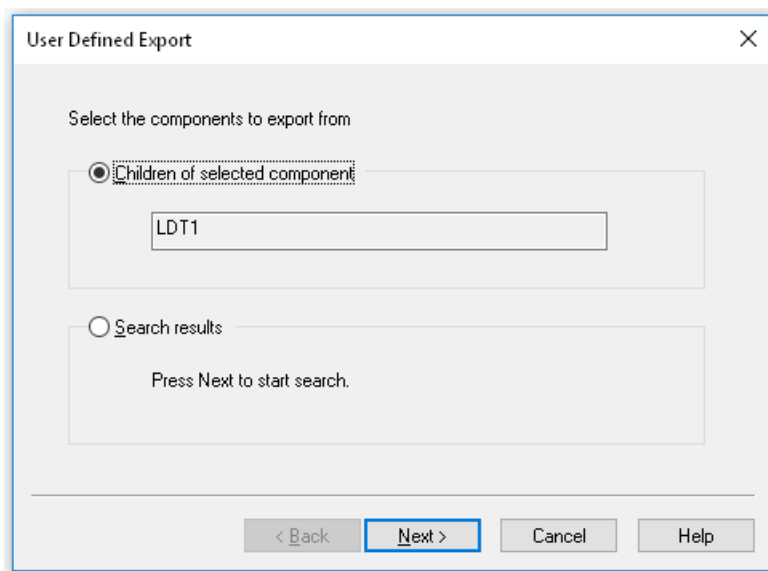
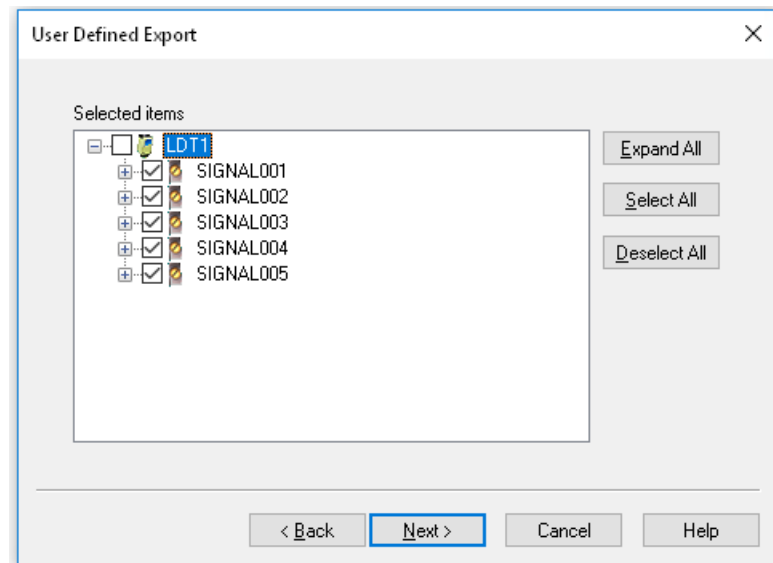


Figure 1-35: User Defined Export Window

3. A list of signals will display. Select the ones to export by clicking on the check boxes or, to export all signals, click on the check box at top left.



4. Click **Next**. The **Format source** dialog will appear. Click on Browse and then search for the correct **Signal_EIP_###.fmt** format.

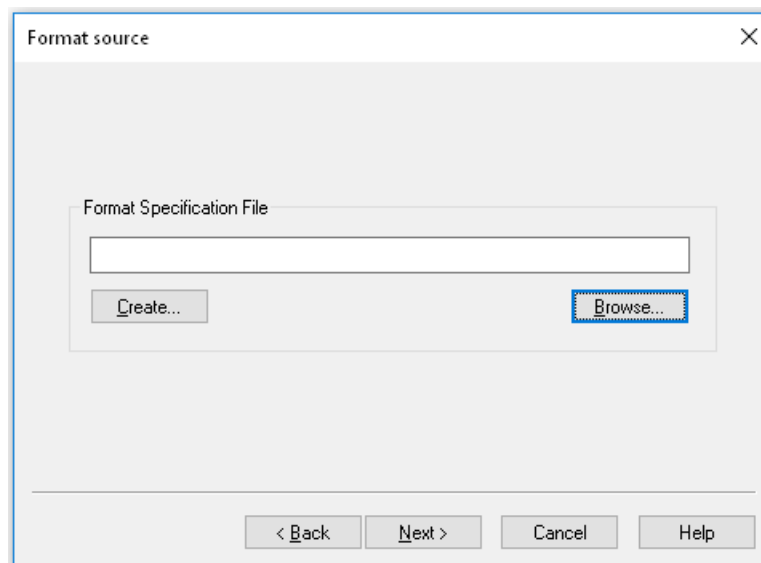


Figure 1-36: Format Source Dialog

- Check that the format has the same **Logical Device Messaging Class** by accessing the **LDT Properties > Device** tab [1]. For instance, **Class 1 Implicit** [2] should use **Signal_EIP_Class1.fmt** [3].

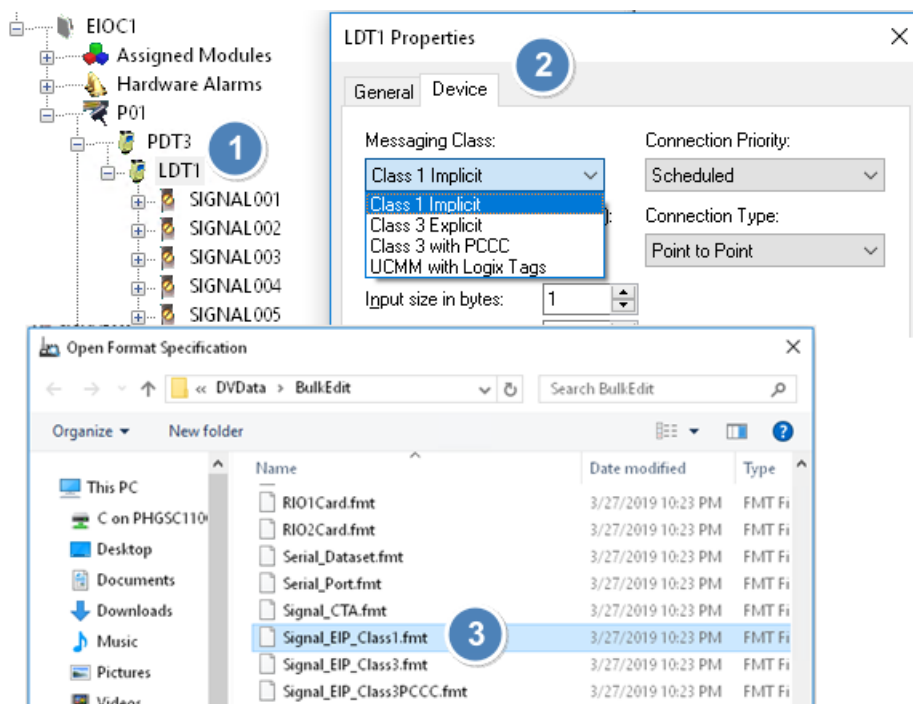


Figure 1-37: Checking the Messaging Class

- In the **Export target** window, choose **Data File** and click on **Browse**.

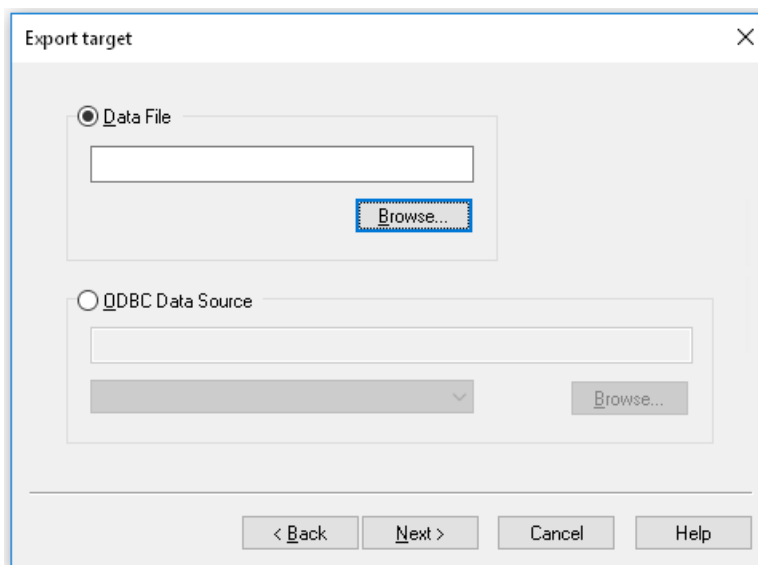


Figure 1-38: Export Target Dialog

7. Enter a name for the export file, click **Open**.

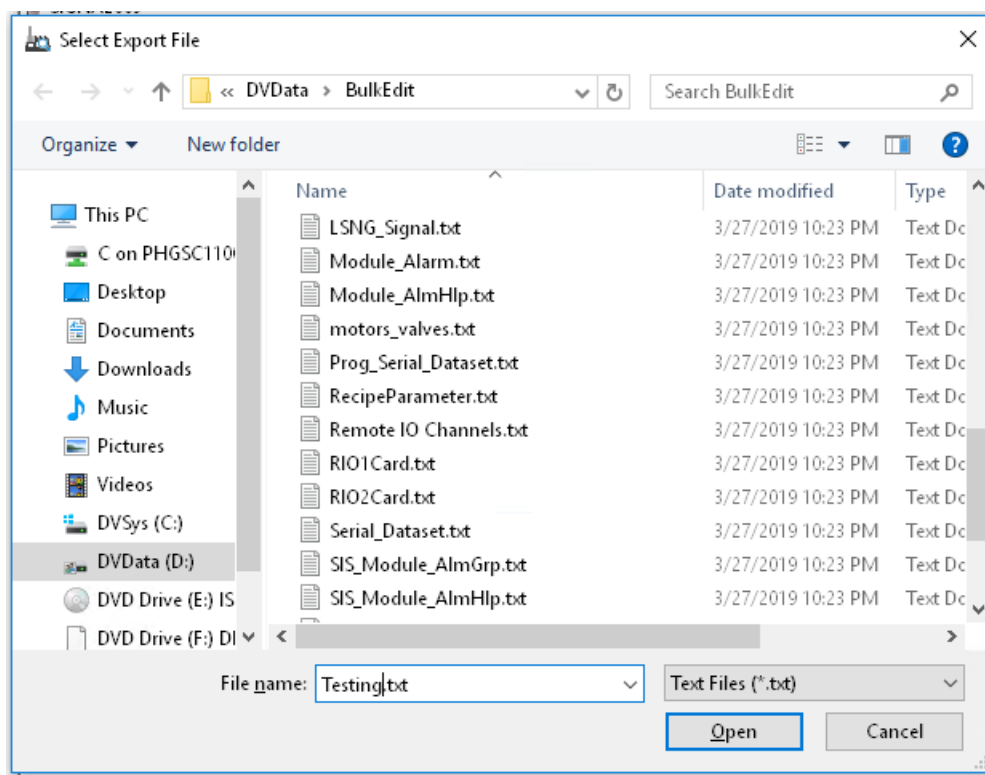


Figure 1-39: Selecting Target Location for Export File

8. In the **Export target** dialog which display, click **Next**.

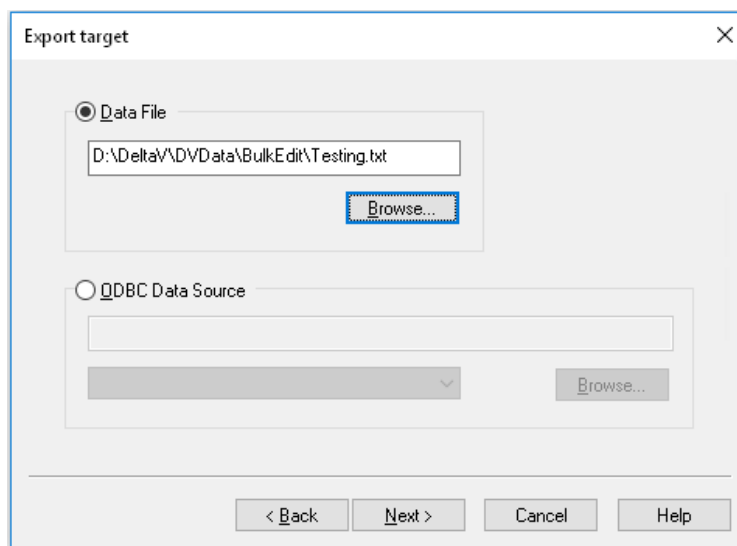


Figure 1-40: Confirming the Export

9. The **Export results** dialog will display. Click on **Export** [1] then **Finish** [2].

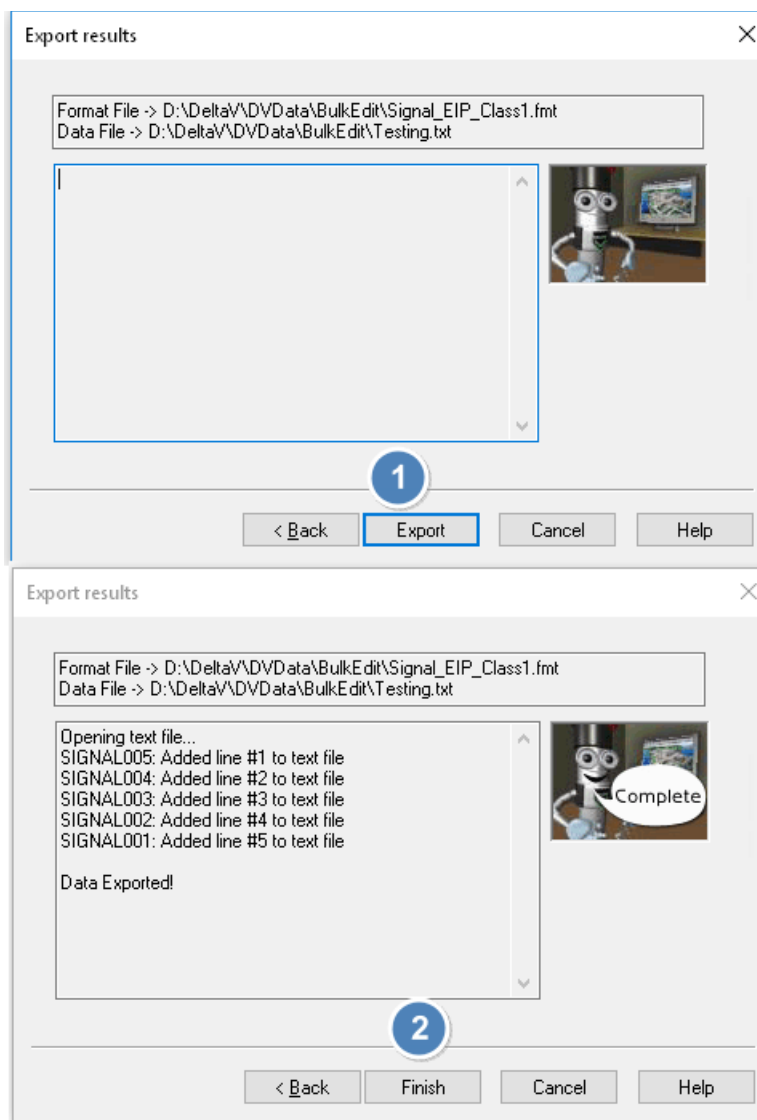


Figure 1-41: Completing the Export Process

1.4.1. Editing the Exported Text File Using the Bulk-Edit Excel Add-On

1. Go to **D: > DeltaV > DVDData > BulkEdit** and then open **BulkEdit_Support.xla**. If a Microsoft Excel Security Notice pop-up window appears, simply select **Enable Macros**.

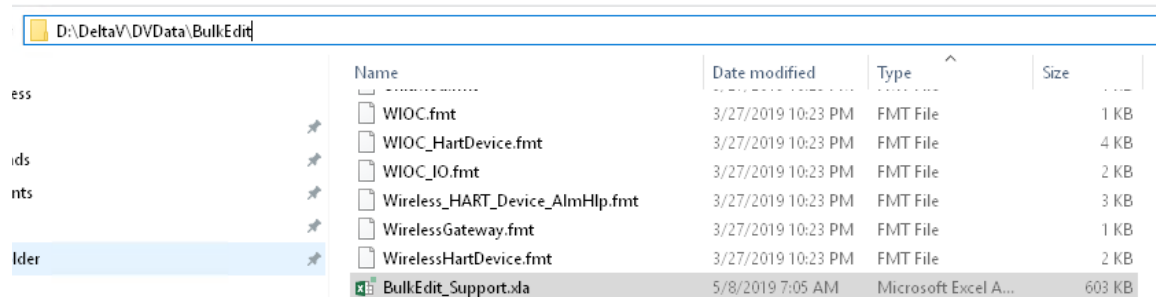


Figure 1-42: BulkEdit_Support.xla Shown in File List

2. On the **Add-Ins** tab [1], select **DeltaV Bulk Edit: Open data file** [2], then look for the exported file to edit [3]. Click OK to dismiss the **Unicode Detected** pop-up dialog [4].

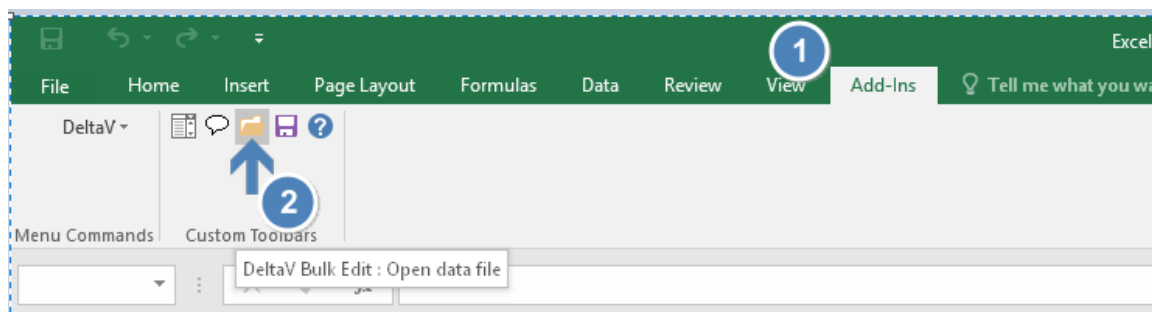


Figure 1-43: Access to File to Open

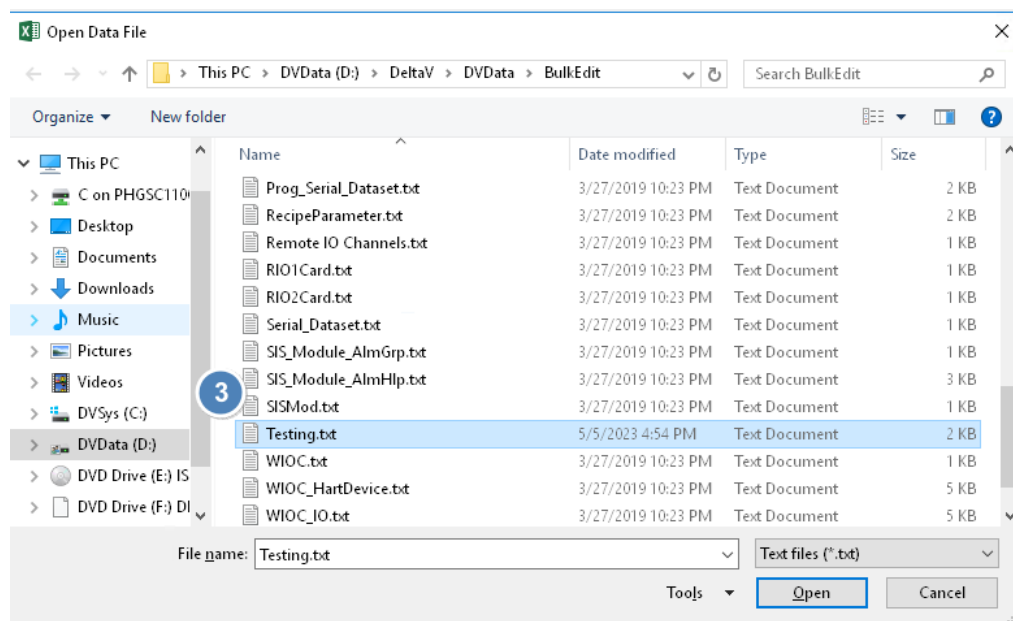


Figure 1-44: Select and Open the Desired File

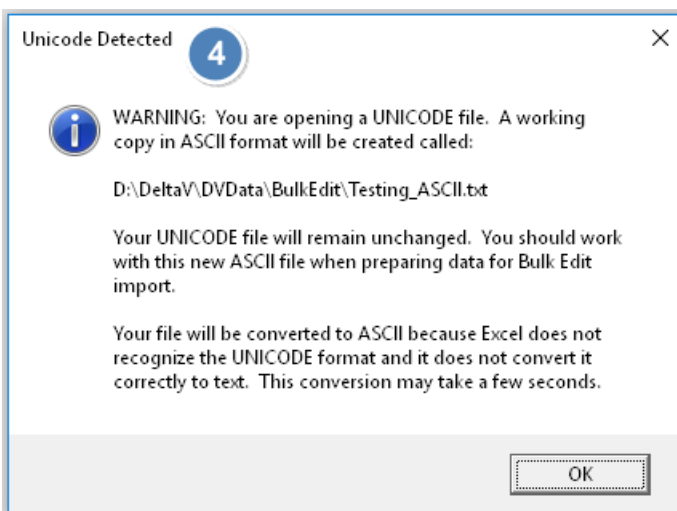


Figure 1-45: Dismiss the Unicode Detected Dialog

3. Change the **Physical Device Node Name** and **Signal Tag** based on the newly-created PDT name (e.g. from PDT3 to PDT4).

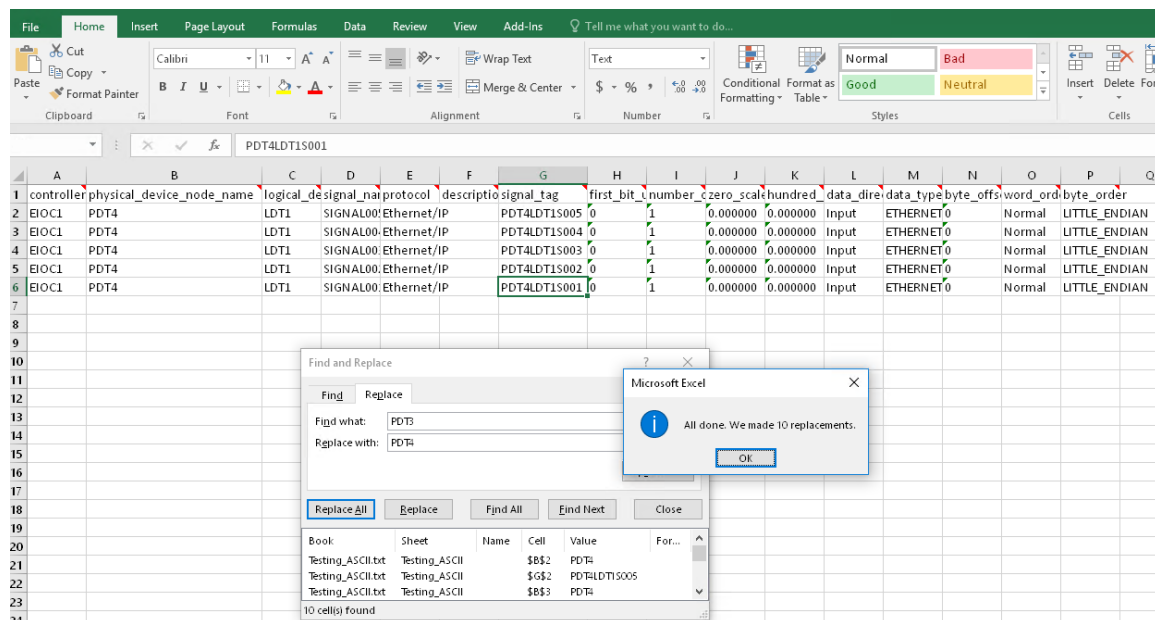


Figure 1-46: Modifying the Excel File Contents

4. Save the edited file – the original can be overwritten, or a new file created. Select **Yes** to “keep using” text format.

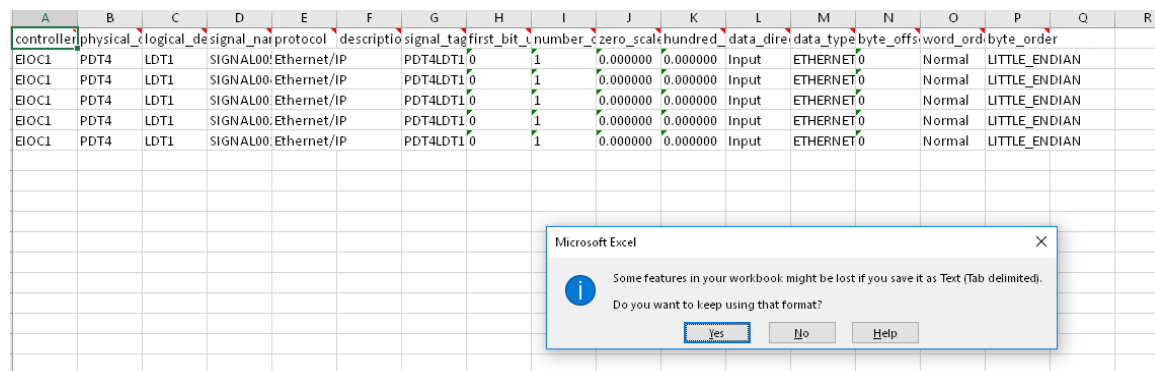


Figure 1-47: Confirming File Format

1.4.2. Importing Signals

To import signals from a file:

1. In DeltaV Explorer, select the newly-created Logical Device [1], then access **File [2] > User Defined Format... [3]**.

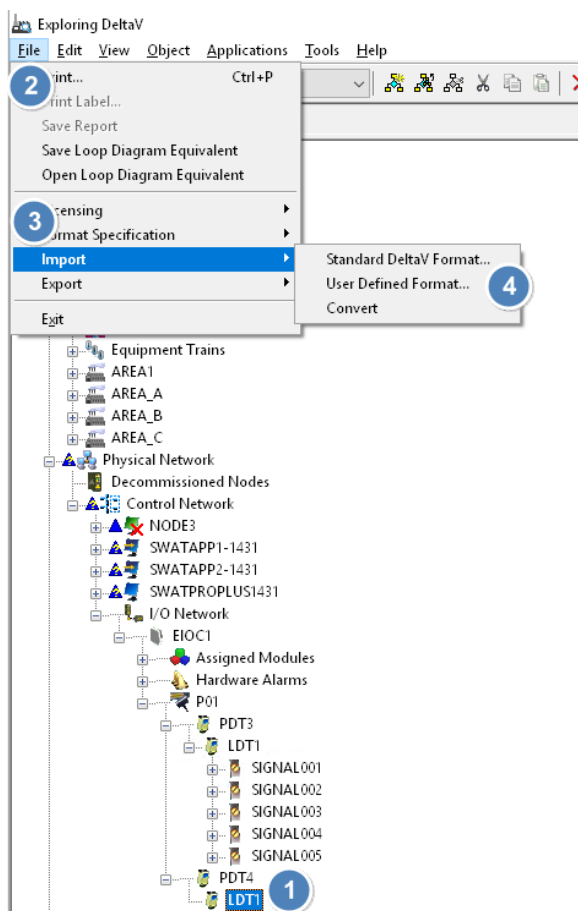


Figure 1-48: Selecting a File to Import

2. A **Format source** dialog will display. Click on **Browse** and search for the correct **Signal_EIP_###.fmt** file.

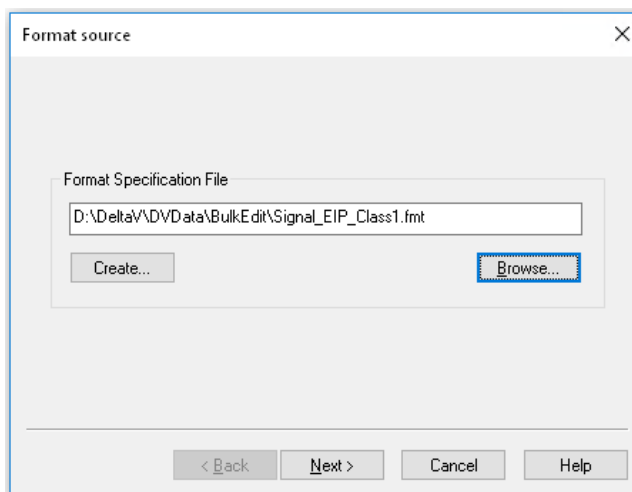


Figure 1-49: Format Source Dialog

- Important: Refer to the format checking note in section 1.4, step 5, on page 1-30.
3. Click **Next**. In the **Import data source** dialog which appears, click **Browse** [1] and search for the edited exported file [2]. Select the file and click **Open**.

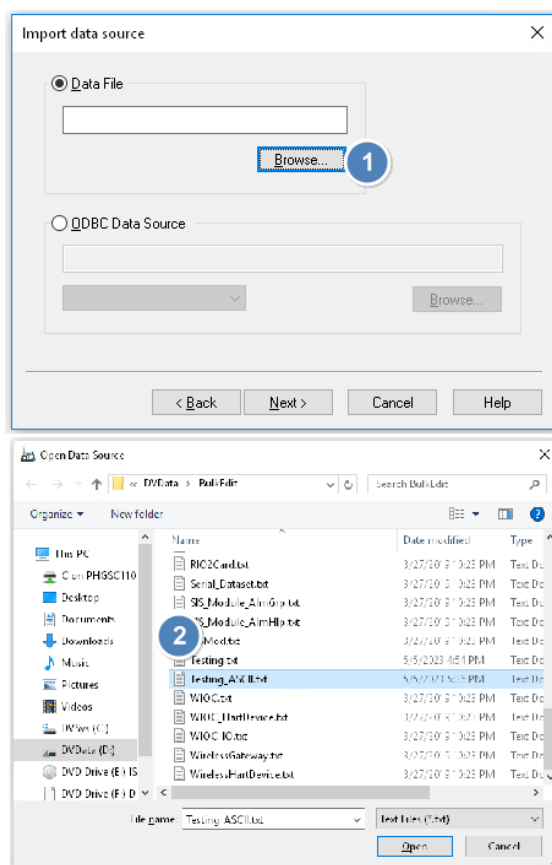


Figure 1-50: Selecting and Opening the Source File

4. Click on **Import**, and click **Close** once the import is complete.

1.5. Troubleshooting

1.5.1. De-Commissioning the PK Controller

For a stand-alone PK Controller:

1. Unplug both primary and secondary cables.
2. Remove the Ethernet ports from the backplane. There is a small release button under Primary and Secondary that must be pushed in to get the ports out.
3. Move the controller from the primary to the secondary slot.
4. Wait one or two minutes (it may take longer) for the red LED to flash, indicating that the controller is decommissioned.
5. Restore everything to its initial state - the controller should now appear in the DeltaV Explorer list of decommissioned nodes.

1.5.2. Commissioning the PK Controller

This process is the same as if a new DeltaV controller was being commissioned:

1. Right click on the PK Controller placeholder
2. Select **Commission**. The PK Controller should be recognized and listed in the **System Configuration > Physical Network > Decommissioned Nodes** in the DeltaV Explorer hierarchy.
3. Run DeltaV PK Controller Administration and create a password for the controller.

Notes:

- Best practice is to create a unique password for each controller when maintaining multiple projects on the Engineering station.
 - Do not forget the unique password.
4. Enable the controller's Ethernet ports that will be used and configure the port-to-port communications as appropriate.
 - In the DeltaV Explorer, select the controller, right click it, and select **Properties** from the dialog.
 - Enable the external Ethernet ports to be used.
 - Click the **Port-to-Port Communications** button.
 - Select the appropriate configuration of communications/isolation of ports.
 - Click **OK** on each dialog.
 5. Configure the P01 subsystem on the PK Controller
 - From the DeltaV Explorer, with the controller object expanded, right-click P01 and select properties.
 - On the **General** tab, select **Enabled**.
 - Click the **Advanced** tab on the **Properties** dialog.

- Enter a valid IP address for your Primary and Redundant (if using the secondary network) connections.
- Select the protocol for this Controller's P01 subsystem.
- Click **OK**.

1.5.3.**Export/Import**

Before importing, make sure the file to be imported contains the correct name – LDT1, LDT2, etc.

METTLER TOLEDO Service

To protect your product's future:

Congratulations on choosing the quality and precision of METTLER TOLEDO. Proper use according to these instructions and regular calibration and maintenance by our factory-trained service team ensure dependable and accurate operation, protecting your investment. Contact us about a service agreement tailored to your needs and budget.

We invite you to register your product at www.mt.com/productregistration so we can contact you about enhancements, updates and important notifications concerning your product.

www.mt.com/IND360

For more information

Mettler-Toledo, LLC
1900 Polaris Parkway
Columbus, OH 43240

© 2025 Mettler-Toledo, LLC
Rev. 00, 10/2025