

# IND780

## Weighing Terminal



© METTLER TOLEDO 2021

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 2021 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.**

### **FCC Notice**

This device complies with Part 15 of the FCC Rules and the Radio Interference Requirements of the Canadian Department of Communications. Operation is subject to the following conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his or her expense.

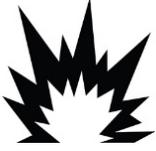
- Declaration of Conformity may be found at <http://glo.mt.com/us/en/home/search/compliance.html/compliance/>.

### **Statement regarding harmful substances**

We do not make direct use of harmful materials such as asbestos, radioactive substances or arsenic compounds. However, we purchase components from third party suppliers, which may contain some of these substances in very small quantities.

## Warnings and Cautions

- READ this manual BEFORE operating or servicing this equipment and FOLLOW these instructions carefully.
- SAVE this manual for future reference.

	<p style="text-align: center;"> <b>WARNING</b></p> <p><b>FOR CONTINUED PROTECTION AGAINST SHOCK HAZARD CONNECT TO PROPERLY GROUNDED OUTLET ONLY. DO NOT REMOVE THE GROUND PRONG.</b></p>
	<p style="text-align: center;"> <b>WARNING</b></p> <p><b>NOT ALL VERSIONS OF THE IND780 ARE DESIGNED FOR USE IN HAZARDOUS (EXPLOSIVE) AREAS. REFER TO THE DATA PLATE OF THE IND780 TO DETERMINE IF A SPECIFIC TERMINAL IS APPROVED FOR USE IN AN AREA CLASSIFIED AS HAZARDOUS BECAUSE OF COMBUSTIBLE OR EXPLOSIVE ATMOSPHERES.</b></p>
	<p style="text-align: center;"> <b>WARNING</b></p> <p><b>WHEN THIS EQUIPMENT IS INCLUDED AS A COMPONENT PART OF A SYSTEM, THE RESULTING DESIGN MUST BE REVIEWED BY QUALIFIED PERSONNEL WHO ARE FAMILIAR WITH THE CONSTRUCTION AND OPERATION OF ALL COMPONENTS IN THE SYSTEM AND THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY HARM AND/OR PROPERTY DAMAGE.</b></p>
	<p style="text-align: center;"> <b>CAUTION</b></p> <p><b>BEFORE CONNECTING/DISCONNECTING ANY INTERNAL ELECTRONIC COMPONENTS OR INTERCONNECTING WIRING BETWEEN ELECTRONIC EQUIPMENT ALWAYS REMOVE POWER AND WAIT AT LEAST THIRTY (30) SECONDS BEFORE ANY CONNECTIONS OR DISCONNECTIONS ARE MADE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN DAMAGE TO OR DESTRUCTION OF THE EQUIPMENT AND/OR BODILY HARM.</b></p>
	<p style="text-align: center;"><b><i>NOTICE</i></b></p> <p><b>OBSERVE PRECAUTIONS FOR HANDLING ELECTROSTATIC SENSITIVE DEVICES.</b></p>

# Contents

<b>1</b>	<b>A-B RIO [Discontinued, January 2021].....</b>	<b>1-1</b>
1.1.	Overview.....	1-1
1.2.	Sharing a PLC Interface.....	1-3
1.3.	Data Definition.....	1-4
1.4.	Hardware Setup.....	1-24
1.5.	Software Setup.....	1-25
1.6.	Troubleshooting.....	1-31
1.7.	Allen-Bradley RIO Option Kit.....	1-32
1.8.	Programming Examples.....	1-32
<b>2</b>	<b>Analog Output.....</b>	<b>2-1</b>
2.1.	Specifications.....	2-1
2.2.	Analog Output Operation.....	2-2
2.3.	Installation.....	2-3
2.4.	Wiring.....	2-4
2.5.	Setup in the IND780 Terminal.....	2-4
2.6.	Status LEDs.....	2-8
2.7.	Analog Output Option Kit.....	2-9
<b>3</b>	<b>ControlNet.....</b>	<b>3-1</b>
3.1.	Overview.....	3-1
3.2.	Sharing a PLC Interface.....	3-4
3.3.	Data Definition.....	3-5
3.4.	Hardware Setup.....	3-27
3.5.	Software Setup.....	3-28
3.6.	Troubleshooting.....	3-34
3.7.	ControlNet Option Kit.....	3-34
3.8.	Programming Examples.....	3-35
<b>4</b>	<b>DeviceNet.....</b>	<b>4-1</b>
4.1.	Preface.....	4-1
4.2.	Overview.....	4-1
4.3.	DeviceNet Characteristics.....	4-1
4.4.	Data Definition.....	4-3
4.5.	Hardware Setup.....	4-18
4.6.	Software Setup.....	4-19

4.7.	Troubleshooting .....	4-22
4.8.	DeviceNet Option Kit.....	4-23
4.9.	DeviceNet Commissioning and Configuration Examples.....	4-23
<b>5</b>	<b>EtherNet/IP .....</b>	<b>5-1</b>
5.1.	Overview.....	5-1
5.2.	Sharing a PLC Interface.....	5-4
5.3.	Data Definition.....	5-5
5.4.	Software Setup.....	5-25
5.5.	Troubleshooting.....	5-31
5.6.	EtherNet / IP Option Kit .....	5-32
5.7.	Programming Examples .....	5-32
<b>6</b>	<b>Modbus TCP .....</b>	<b>6-1</b>
6.1.	Preface .....	6-1
6.2.	Overview.....	6-1
6.3.	Modbus TCP Characteristics .....	6-2
6.4.	Modbus TCP Board .....	6-2
6.5.	Sharing a PLC Interface.....	6-3
6.6.	Data Definition.....	6-4
6.7.	Software Setup.....	6-23
6.8.	Troubleshooting.....	6-29
6.9.	Modbus TCP Option Kit.....	6-30
6.10.	Modbus TCP Configuration Example.....	6-30
<b>7</b>	<b>PROFIBUS .....</b>	<b>7-1</b>
7.1.	Overview.....	7-1
7.2.	Sharing a PLC Interface.....	7-4
7.3.	Data Definition.....	7-5
7.4.	Hardware Setup .....	7-27
7.5.	Software Setup.....	7-28
7.6.	Troubleshooting.....	7-35
7.7.	PROFIBUS Option Kit .....	7-35
<b>8</b>	<b>PROFINET .....</b>	<b>8-1</b>
8.1.	Overview.....	8-1
8.2.	Sharing a PLC Interface.....	8-3
8.3.	Data Definition.....	8-5
8.4.	Shared Data Access .....	8-25

8.5.	Software Setup.....	8-25
8.6.	PROFINET GSDML File .....	8-32
8.7.	Assigning the IP Address and Device Name.....	8-33
8.8.	Troubleshooting .....	8-35
8.9.	PROFINET Option Kit .....	8-37
8.10.	Siemens S7-300 Programming Examples.....	8-37
<b>9</b>	<b>Indirect Shared Data Access .....</b>	<b>1</b>
9.1.	Overview.....	1
9.2.	Indirect Access Data Structure .....	1
9.3.	Command Codes .....	3
9.4.	EtherNet/IP Example .....	3
9.5.	PROFINET Example .....	6

# 1 A-B RIO [Discontinued, January 2021]

- The Allen Bradley RIO interface was discontinued in January, 2021. Information provided in this manual is provided to support legacy installations only.

## 1.1. Overview

The A-B RIO Kit option enables the IND780 terminal to communicate to Allen-Bradley Programmable Logic Controllers (PLCs) through direct connection to the A-B RIO network. The option consists of a backplane-compatible I/O module, and software that resides in the IND780 terminal to implement the data exchange. Figure 1-1 shows an A-B Rio interface module.



Figure 1-1: A-B RIO Interface Module

The A-B RIO Kit option has the following features:

- A-B RIO Node Adapter Chip Set (licensed from Allen-Bradley) and termination for the A-B network cable (blue hose) on a three-position removable terminal block.
- User-programmable RIO communication parameters that are configured in software set up through the terminal keyboard/display. The parameters are:
  - 57.6K, 115.2K, or 230.4K baud rate
  - 1/4, 1/2, 3/4, full rack (depends upon the number of scales/interface)
  - Rack address
  - Starting quarter
  - Last rack designation

- Capability for bi-directional discrete mode communications of weight, display increments, status, and control data between the PLC and the terminal.
- Capability for bi-directional block transfer communication of many IND780 terminal data variables.

### 1.1.1. Communications

The IND780 terminal utilizes component parts that are provided by Allen-Bradley to ensure complete compatibility with the Allen-Bradley RIO network. An IND780 terminal is recognized as a RIO (Allen-Bradley) device by the PLC.

Each option connected to the Allen-Bradley RIO network represents a physical node. The connection is facilitated by a three-position removable terminal block on the option card. These terminals correspond to the terminals on the A-B PLC RIO connector.

The wiring between the PLC and the RIO connector uses the standard RIO cable used by Allen-Bradley (Figure 1-4). This cable is often referred to as the “blue hose.” The cable installation procedures and specification including distance and termination requirements are the same as recommended by Allen-Bradley for the RIO network – refer to the A-B RIO scanner card documentation for further information.

The IND780 terminal’s baud rate is programmed via setup at **Communication > PLC Interface > A-B RIO**.

### 1.1.2. Node/Rack Address

Although each RIO option represents one physical node, the addressing of the node is defined as a logical rack address. This address is chosen by the system designer, and then programmed into the IND780 terminal and PLC. The IND780 terminal’s address is programmed in setup at **Communication > PLC Interface > A-B RIO**. IND780 address entry can be in either decimal or octal, while most PLC address entry is in octal.

The IND780 terminal’s setup capabilities allow selection of the logical rack address, starting quarter, and designation of the last rack and the number of quarters (Message Slots). Quarters must be contiguous in a single, logical rack, so the starting quarter must be low enough to accommodate all of the required data for the scales in a single, logical rack. The IND780 will determine the number of quarters needed for the chosen data format and number of configurable Message Slots. It only allows selection of the possible starting quarters and maximum Message Slots. Note that floating point data format uses a complete rack address regardless of the number of message slots used (max. 2).

### 1.1.3. Data Formats

The A-B RIO Kit option has two types of data exchanges: discrete data and block transfer data.

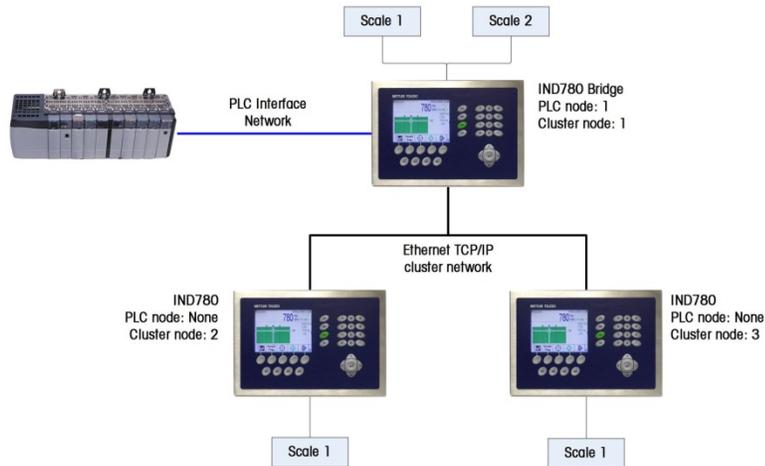
Discrete data is continuously available. The A-B RIO Kit option has its own logical rack address to send and receive information to and from the PLC. Discrete data is always sent even when the optional block transfer data is used.

Block transfer data is available when the option is enabled in setup at **Communication > PLC Interface > A-B RIO**. This data is used to pass information that cannot be sent by the discrete data because of size or process speed limitations. Refer to section 1.3, **Data Definition**, for more information.

# 1.2. Sharing a PLC Interface

## 1.2.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. The following example (Figure 1-2) shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 1-2: A-B RIO Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate chapter in this manual for further details on message slots' limitations.

To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure 1-3 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to section 1.5, **Software Setup**, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

PLC Message Slots View		
Message	Scale	Terminal
1	1	Local
2	1	6
3	- End -	

**Figure 1-3: PLC Message Slots View Screen**

## 1.3. Data Definition

The A-B RIO Kit option uses two types of data for its communication with PLCs: discrete data and block transfer data. Discrete data is always available. The data transfer is accomplished via the PLC's I/O messaging. Block transfer data is only available if this data option is enabled in setup at **Communication > PLC Interface > A-B RIO**. If the block transfer data option is enabled, it is provided in addition to the discrete data. Block transfer data requires "block transfer" ladder sequence programming to accomplish the data transfer between the IND780 and PLC.

### 1.3.1. Data Integrity

The IND780 has specific bits to allow the PLC to confirm that data was received without interrupt and the IND780 is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received from the IND780. Refer to the data charts for specific information regarding the Data OK, Update in Progress, Data Integrity bits and their usage.

### 1.3.2. Discrete Data

Five formats of discrete data are available with the A-B RIO Kit option: Integer, divisions, floating point, template and application. Only one type of data format may be selected and used by IND780's sharing the same A-B RIO logical rack address.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word (refer to Table 1-7 and Table 1-8 for explanation) numerical values. The IND780 provides one quarter rack of data per Message Slot.

The floating-point format allows bi-directional communication of discrete bit encoded information or numeric data encoded in IEEE 754, single precision floating point format. The IND780 provides one-half rack of data per Message Slot. In floating point format, the IND780 will populate a complete rack of data regardless if one or two Message Slots are used.

The format of discrete data will affect the amount of rack space required. Integer and division formats require one-quarter rack per IND780 (two 16-bit words of input and two 16-bit words of output data) Message Slot. One IND780, with 1 Message Slot, would use a quarter rack; two IND780's, with 1

Message Slot, would use a half rack; three IND780's, with 1 Message Slot, would use three-quarters of a rack; and four IND780's, with 1 Message Slot, would use a full rack.

The floating-point format requires more space per IND780 because floating point data uses two 16-bit words of data to represent just the numeric data alone. The floating point format uses one-half rack per IND780 (four 16-bit words of input and four 16-bit words of output data) Message Slot.

Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions (increments). The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Floating point is the only data format that includes decimal point information. All other formats ignore decimal points. Accommodation of decimal point location must take place in the PLC logic, when it is needed with these formats.

Another issue is the type of information communicated between the IND780 and PLC for the application. Because the floating point format has more space for its data, it has additional information that can be sent or received without using the optional block transfer data. Please see each format's detailed description of the data available, to determine which is most suitable for the specific application.

Changing the Data Format to be used by the IND780 will cause all Message Slots to be cleared. Data format is set up in the **Communication > PLC Interface > Data Format** screen.

1.3.2.1. Examples

<b>250 x .01 scale</b>				
IND780 Displays:	0	2.00	51.67	250.00
Format sent:				
Integer	0	200	5167	25000
Division	0	200	5167	25000
Floating Point	0	2.0	51.67	250.0

Any of the formats could be used in this case.

<b>50,000 x 10 scale</b>				
IND780 Displays:	0	200	5160	50000
Format sent:				
Integer	0	200	5160	-(15536)
Division	0	20	516	5000
Floating Point	0	200.0	5160.0	50000.0

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

150 x .001 scale				
IND780 Displays:	0	2.100	51.607	150.000
Format sent:				
Integer	0	2100	-(13929)	18928
Division	0	2100	-(13929)	18928
Floating Point	0	2.1	51.607	150.0

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32.767.

### 1.3.3. Byte Order

**Word Swap** Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** Makes the floating point format compatible with S7 PROFIBUS.

**Historic** Makes the floating point format compatible with PLC 5.

### 1.3.4. Message Slots

The integer and division formats provide one-quarter rack (two 16-bit words of input and two 16-bit words of output data) per Message Slot. Each message slot is assigned to a local or remote scale and scales may be repeated in additional message slots. Remote scales must reside in an IND780 Ethernet TCP clustered to the IND780 containing the A-B RIO interface. Depending upon the starting quarter there may be up to four Message Slots provided. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. Table 1-1 and Table 1-2 provide input and output words and word usage information.

**Table 1-1: PLC Input Words and Word Usage (Integer and Division), BT Disabled**

PLC Input Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
0	Message Slot 1 Weight Data	-	-	-
1	Message Slot 1 Scale Status	-	-	-
2	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-	-
3	Message Slot 2 Scale Status	Message Slot 1 Scale Status	-	-
4	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-
5	Message Slot 3 Scale Status	Message Slot 2 Scale Status	Message Slot 1 Scale Status	-

PLC Input Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
6	Message Slot 4 Weight Data	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data
7	Message Slot 4 Scale Status	Message Slot 3 Scale Status	Message Slot 2 Scale Status	Message Slot 1 Scale Status

**Table 1-2: PLC Output Words and Word Usage (Integer and Division), BT Disabled**

PLC Output Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
0	Message Slot 1 Weight Data	-	-	-
1	Message Slot 1 Scale Command	-	-	-
2	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-	-
3	Message Slot 2 Scale Command	Message Slot 1 Scale Command	-	-
4	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-
5	Message Slot 3 Scale Command	Message Slot 2 Scale Command	Message Slot 1 Scale Command	-
6	Message Slot 4 Weight Data	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data
7	Message Slot 4 Scale Command	Message Slot 3 Scale Command	Message Slot 2 Scale Command	Message Slot 1 Scale Command

The floating point format provides one-half rack (four 16-bit words of input and up to four 16-bit words of output data) per Message Slot. See Table 1-5 and Table 1-6 for details.

The number of Message Slots is set up at **Communications > PLC Interface > Data Format** (Figure 1-8).

**Table 1-3: PLC Input Words and Word Usage (Integer and Division), BT Enabled**

PLC Input Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
0	Not Valid	Reserved	Reserved	Reserved
1	Not Valid	Reserved	Reserved	Reserved
2	Not Valid	Message Slot 1 Weight Data	-	-
3	Not Valid	Message Slot 1 Scale Status	-	-

PLC Input Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
4	Not Valid	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-
5	Not Valid	Message Slot 2 Scale Status	Message Slot 1 Scale Status	-
6	Not Valid	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data
7	Not Valid	Message Slot 3 Scale Status	Message Slot 2 Scale Status	Message Slot 1 Scale Status

**Table 1-4: PLC Output Words and Word Usage (Integer and Division), BT Enabled**

PLC Output Words	Word Usage Start Quarter 1 (Group 0)	Word Usage Start Quarter 2 (Group 2)	Word Usage Start Quarter 3 (Group 4)	Word Usage Start Quarter 4 (Group 6)
0	Not Valid	Reserved	Reserved	Reserved
1	Not Valid	Reserved	Reserved	Reserved
2	Not Valid	Message Slot 1 Weight Data	-	-
3	Not Valid	Message Slot 1 Scale Command	-	-
4	Not Valid	Message Slot 2 Weight Data	Message Slot 1 Weight Data	-
5	Not Valid	Message Slot 2 Scale Command	Message Slot 1 Scale Command	-
6	Not Valid	Message Slot 3 Weight Data	Message Slot 2 Weight Data	Message Slot 1 Weight Data
7	Not Valid	Message Slot 3 Scale Command	Message Slot 2 Scale Command	Message Slot 1 Scale Command

**Table 1-5: PLC Floating Point Input Words**

PLC Output Words	Bits 0 - 7	Bits 8 - 15
0	Message Slot 1, Reserved	Message Slot 1, Command Response, See Message Slot Table
1	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
2	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
3	Message Slot 1, Scale Status, See Message Slot Table	Message Slot 1, Scale Status, See Message Slot Table
4	Message Slot 2, Reserved	Message Slot 2, Command Response, See Message Slot Table
5	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data

PLC Output Words	Bits 0 - 7	Bits 8 - 15
6	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
7	Message Slot 2, Scale Status, See Message Slot Table	Message Slot 2, Scale Status, See Message Slot Table

**Table 1-6: PLC Floating Point Output Words**

PLC Output Words	Usage
0	Reserved
1	Message Slot 1 Command
2	Message Slot 1 Floating Point data
3	Message Slot 1 Floating Point data
4	Message Slot 2 Command
5	Message Slot 2 Floating Point data
6	Message Slot 2 Floating Point data
7	Not Used

### 1.3.5. Integer and Division

When one of these formats is selected, the IND780 will have one quarter rack of data: two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC's input data will contain one 16-bit word for the scale's weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC's output data. The PLC's output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 1-7 and Table 1-8 provide detailed information on the integer (int) and division (div) data formats. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 1-7: Discrete Read Integer or Division – IND780 >> PLC**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

**Notes for Table 1-7:**

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
5. **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) and when in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 1-8: Discrete Write Integer or Division – PLC >> IND780**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

#### Notes for Table 1-8:

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **Bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- 2 A binary value in **Bit 0**, **Bit 1**, and **Bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 and 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- 3 A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- 4 A transition from "0" to "1" initiates a CLEAR command.
- 5 A transition from "0" to "1" initiates a TARE command.
- 6 A transition from "0" to "1" initiates a PRINT command.
- 7 A transition from "0" to "1" initiates a ZERO command.
- 8 If **Bit 8** is set to "0", all of the scale's target logic is aborted. Setting **Bit 8** to "1" again restarts the terminal's target logic. Note: The PLC must start the target logic before the PLC can abort the target logic.
- 9 **Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.
- 10 **Bit 12**, **Bit 13**, and **Bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- 11 When **Bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and transferred into the target logic.

## 1.3.6. Floating Point

### 1.3.6.1. Operational Overview

The IND780 uses integer commands from the PLC to select the floating point weight output data. The IND780 recognizes a command when it sees a new value in the Message Slot command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the IND780 recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. The IND780 also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC should wait until it receives the command acknowledgment from the IND780 before sending another command.

The IND780 can report two types of values to the PLC: real-time and static. When the PLC requests a real-time value, the IND780 acknowledges the command from the PLC once but sends and updates the value at every interface cycle update. Note that PLC RIO scanner baud rate and the number of racks in the scanner configuration will have an impact on the PLC input data update

rate. If the PLC requests a static value, the IND780 acknowledges the command from the PLC once and updates the value once. The IND780 will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, feed, and tolerance values are examples of static data.

The IND780 can send a rotation of up to nine different real-time values. The PLC sends commands to the IND780 to add a value to the rotation. Once the rotation is established, the PLC must instruct the IND780 to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the IND780 to advance to the next value. Note that once the rotation is established the rotation will be present on all message slots. If the IND780 is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next Interface cycle update. (The Interface cycle update has an update rate of up to 17 Hz or 60 milliseconds.) Note that PLC RIO scanner baud rate and the number of racks in the scanner configuration will have an impact on the PLC input data update rate.

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command the IND780 switches to the next value in the rotation order. Each message slot can be controlled separately. The IND780 stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point command examples in Table 1-11 through Table 1-14 for additional information. The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the order entered in Data Format setup.

Table 1-9 to Table 1-12 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 1-9: Discrete Read Floating Point – IND780 >> PLC**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
17	Cmd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	15

**Notes for Table 1-9:**

- 1 The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- 4 **Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- 5 **Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- 6 **Bit 6** and **Bit 7** are user defined, and corresponds to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- 7 The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- 8 **Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- 9 **Bit 9, Bit 10, and Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 10 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 11 The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- 12 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) and when in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 1-10: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **

Dec	Data
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency, instance by scale
12	Notch filter frequency, instance by scale
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size, instance by scale
19–28	Not used
29	Last IND780 error code, instance by scale
30	No data response – command successful
31	No data response – command failed

- \* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.
- \*\* SDV means Shared Data Variable.

**Table 1-11: Discrete Write Floating Point – PLC >> IND780**

Octal Address	WORD 0	WORD 1	WORD 2 <sup>1</sup> FP load value	WORD 3 <sup>1</sup> FP load value	WORD 4 <sup>2</sup>	WORD 5 <sup>1,2</sup> FP load value	WORD 6 <sup>1,2</sup> FP load value	WORD 7	Bit number
0	Reserved	Command Word Message Slot 1 <sup>3</sup>	X	X	Command Word Message Slot 2 <sup>2,3</sup>	X	X	Not Used	<b>0</b>
1			X	X		X	X		<b>1</b>
2			X	X		X	X		<b>2</b>
3			X	X		X	X		<b>3</b>
4			X	X		X	X		<b>4</b>
5			X	X		X	X		<b>5</b>
6			X	X		X	X		<b>6</b>
7			X	X		X	X		<b>7</b>
10			X	X		X	X		<b>8</b>
11			X	X		X	X		<b>9</b>
12			X	X		X	X		<b>10</b>
13			X	X		X	X		<b>11</b>
14			X	X		X	X		<b>12</b>
15			X	X		X	X		<b>13</b>
16			X	X		X	X		<b>14</b>
17			X	X		X	X		<b>15</b>

**Notes for Table 1-11:**

- 1 The Bits in WORD 2 and WORD 3 (and WORD 5 and WORD 6) are a single-precision floating point value. This value is used with the command in WORD 1 (or WORD 4) to instruct the terminal to download the floating point value into the field specified in the command.
- 2 These words are only used if a second Message Slot is desired.
- 3 The command words WORD 1 and WORD 4 (for the second Message Slot) are used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 1-12, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.

Not all commands will require a value in the floating point load value words.

**Table 1-12: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName
0	00	Report next rotation field @ next Interface cycle update <sup>1</sup>	
1	01	Report next rotation field <sup>1,2</sup>	
2	02	Report next rotation field <sup>1,2</sup>	
3	03	Reset rotation	
10	0a	Report gross weight <sup>1,3</sup>	
11	0b	Report net weight <sup>1,3</sup>	
12	0c	Report tare weight <sup>1,3</sup>	
13	0d	Report fine gross weight <sup>1,3</sup>	
14	0e	Report fine net weight <sup>1,3</sup>	
15	0f	Report fine tare weight <sup>1,3</sup>	
16	10	Report Rate <sup>1,3</sup>	
17	11	Report SDV ALxx01 <sup>6</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>	
20	14	Report notch filter frequency <sup>3</sup>	
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12
25	19	Report spill value <sup>3,10</sup>	SPxx09
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>
29	1d	Report last error <sup>3</sup>	
30	1e	Report primary increment size <sup>3</sup>	
40	28	Add gross weight to rotation <sup>7</sup>	
41	29	Add net weight to rotation <sup>7</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>	
46	2e	Add rate to rotation <sup>7</sup>	
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>
60	3c	Load numeric tare value <sup>4</sup>	
61	3d	Pushbutton tare command <sup>7</sup>	
62	3e	Clear command <sup>7</sup>	
63	3f	Print command <sup>7</sup>	
64	40	Zero command <sup>7</sup>	
65	41	Select scale 1 <sup>7</sup>	
66	42	Select scale 2 <sup>7</sup>	
67	43	Select next scale <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>

Dec	Hex	Command	SDName
76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
78	4e	Disable LCD display <sup>7</sup>	
79	4f	Enable LCD display <sup>7</sup>	
80	50	Set normal display mode <sup>7,9</sup>	PD0119
81	51	Display message 1 <sup>7,9</sup>	PD0119
82	52	Display message 2 <sup>7,9</sup>	PD0119
83	53	Display message 3 <sup>7,9</sup>	PD0119
84	54	Display message 4 <sup>7,9</sup>	PD0119
85	55	Display Message 5 <sup>7,9</sup>	PD0119
86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
87	57	Display SDV PDO118 <sup>7,9</sup>	PD0119
88	58	Disable weight display <sup>7</sup>	
89	59	Enable weight display <sup>7</sup>	
90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
104	68	Master control relay OFF <sup>7, 12</sup>	
105	69	Master control relay ON <sup>7, 12</sup>	
110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
111	6f	Set target fine feed value <sup>4,10</sup>	
112	70	Set - tolerance value <sup>4,10</sup>	
113	71	Set target value and apply <sup>4, 10, 14</sup>	
114	72	Start target logic <sup>7,10, 14</sup>	
115	73	Abort target logic <sup>7,10, 15</sup>	
116	74	Target use gross weight <sup>7,10</sup>	SPxx04
117	75	Target use net weight <sup>7,10</sup>	SPxx04
118	76	Target use rate <sup>7,10</sup>	SPxx04
119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
121	79	Enable target latching <sup>7,10</sup>	SPxx06
122	7a	Disable target latching <sup>7,10</sup>	SPxx06
123	7b	Reset target latch <sup>7</sup>	SPxx07
124	7c	Set target spill value <sup>4, 10</sup>	
131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
160	a0	Apply scale setup <sup>7</sup>	QC0149
161	a1	Write Calibration to EEPROM <sup>7</sup>	
162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
163	a3	Enable Tare on IND780 console <sup>7</sup>	

Dec	Hex	Command	SDName
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>
73	49	Set low-pass filter corner frequency <sup>4</sup>	
74	4a	Set notch filter frequency <sup>4</sup>	
75	4b	Reset ENTER key <sup>7</sup>	

Dec	Hex	Command	SDName
164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
168	a8	Select scale 3 <sup>7</sup>	
169	a9	Select scale 4 <sup>7</sup>	

### Notes for Table 1-12:

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

### 1.3.6.2. Floating Point Data Format and Compatibility

In Floating Point Data Format, the PLC and terminal exchange weight, target, and tare data in single-precision floating-point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

Some Allen-Bradley PLCs require special integrity checking to communicate floating point numbers across the Remote I/O link. The Allen-Bradley PLC-5 and KTX Scanner Card programs must check two data integrity bits to verify the integrity of the floating point data it reads from the terminal. Allen-Bradley SLC programs always read valid floating-point data from the terminal and do not have to make special checks to guarantee the validity of the floating-point data. The Allen-Bradley PLC-3 and PLC-5/250 cannot support terminals in floating point mode as they cannot guarantee the integrity of the floating-point data.

There are two data integrity bits that the terminal uses to maintain data integrity when communicating with the Allen-Bradley PLC-5 Remote I/O Scanner or KTX Scanner Card. One bit is in the beginning word of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC-5. When the program detects this condition, it should send a new command to the terminal.

The Allen-Bradley SLC PLC programs do not have to make special checks to guarantee the validity of the floating-point data.

The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the word order set up by the user.

1.3.6.3.

Floating Point Command Examples

Table 1-13 to Table 1-16 provide floating point command examples.

**Table 1-13: Data Requirement: Only Net Weight Sent (Continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word 0	none required		
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word, the IND780 terminal will update the net value every interface update cycle.				

**Table 1-14: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2 (PLC sends command to set target 1 cutoff value)	110 (dec) loaded into command word 0	floating point value = 21.75		
3 (IND780 terminal sees new command, loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word 0			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	(null value)
The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.				

**Table 1-15: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	0.0
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0
At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.				
7 (PLC sends the command to begin the rotation at next interface update)	0 (dec) loaded into command word 0			
8 (IND780 terminal sends gross weight at Interface cycle update ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at next interface update)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 6	Floating point value = rate
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value next interface update)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at next interface update)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 1-16: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	0.0
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0
At this point, the rotation has been set up. Now the PLC needs to send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word 0			
8 (IND780 terminal acknowledges the command and sends net weight at every Interface cycle update until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word 0			
10 (IND780 terminal acknowledges the command and sends rate at every Interface cycle update until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
11 (PLC sends the command to report the next field in the rotation.)	1 (dec) loaded into command word 0			
12 (IND780 terminal acknowledges the command and sends net weight at every Interface cycle update until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
13 (PLC sends the command to report the next field.)	2 (dec) loaded into command word 0			
14 (IND780 terminal acknowledges the command and sends rate at every Interface cycle update until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate

At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value

#### 1.3.6.4. Shared Data Mode

The Shared Data mode PLC communications is not available in Allen-Bradley PLC's. Block Transfer communications is used instead.

#### 1.3.7. Block Transfer

Block Transfer mode is much less efficient than the discrete data modes, which are optimized for real time communications of weight and status data. Block Transfer mode accesses the IND780 "Shared Data" directory structure each time a data item is accessed. By contrast, the discrete mode communications has a direct interface to a limited number of real time terminal data fields.

- Do not use Block Transfer mode for real-time communications.
- When Block Transfer is enabled in integer and divisions, the first quarter (Word0, Word1) is reserved.

## 1.3.7.1.

## Floating Point and String Data Field Codes for BTW/BTR

Table 1-15 and Table 1-16 describe some of the floating point and string data fields that the IND780 terminal can access. String data fields are ASCII character strings that identify an IND780 Shared Data Variable. Each table contains the following information:

<b>Field Code</b>	The ASCII field that must be loaded into the Block Transfer write buffer. It identifies the data that is written to the terminal or returned by the terminal in a Block Transfer read.  The field code must be expanded to eight ASCII bytes by filling with two leading spaces. The field code structure is CCIIAA; where CC is the Shared Data Class consisting of two ASCII alpha characters; where II is the Shared Data Class Instance consisting of two ASCII numeric characters; where AA is the Shared Data Class Instance Attribute consisting of two ASCII numeric characters. The Shared Data Class Instance is typically associated to scale number; '01' = scale 1, '02' = scale 2, etc. There are only a few Shared Data Classes that have more than scale number Instance.  Block Transfer Shared Data Variables are either an ASCII string or a number. Numbers are written and read as a 32-bit floating point value. If the Shared Data Variable is not a string then it is a number.
<b>Description</b>	A description of the field.
<b>Read/Write</b>	Indicates whether the PLC can read and/or write to the field.
<b>Length</b>	The number of bytes (length) of the field. All floating point values are 4 bytes (2 words) long. Strings are the length specified.

**Table 1-17: Floating Point Data Fields**

Field Code	Description	Read/Write	Length
Wtxx10	Gross Weight	R	4
Wtxx11	Net Weight	R	4
Wtxx12	Auxiliary Gross Weight	R	4
Wtxx13	Auxiliary Net Weight	R	4
Wsxx02	Tare Weight	R	4
Wsxx04	Auxiliary Tare Weight	R	4
Spxx05	Target Coincidence Value	R/W	4
Wxxx31	Scale Motion (0 or 1)	R	4
Wxxx32	Center of Zero (0 or 1)	R	4
Wxxx33	Over Capacity (0 or 1)	R	4
Wxxx34	Under Zero (0 or 1)	R	4
Wxxx35	Net Mode (0 or 1)	R	4
Wsxx01	Current Scale Mode	R	4
Wsxx02	Tare Weight	R	4
Wsxx04	Auxiliary Tare Weight	R	4
Wsxx05	Current Units	R	4
Wsxx06	Tare Source	R	4
Csxx04	Auxiliary Weight Units	R/W	4
Spxx04	Target Data Stream Type	R/W	4
Xpxx01	Transaction Counter	R/W	4

**Table 1-18: String Data Fields**

Field Code	Description	Read/Write	Length
wfxx01	Gross Weight	R	13
wfxx02	Net Weight	R	13
wfxx03	Weight Units	R	4
wfxx04	Auxiliary Gross Weight	R	13
wfxx05	Auxiliary Net Weight	R	13
wfxx06	Auxiliary Weight Units	R	7
csxx12	Custom Units Name	R/W	13
csxx03	Scale ID	R/W	21
spxx01	(Target) Descriptor Name	R/W	21
xdxx03	Current Date	R	12
xdxx04	Time of Day	R	12
csxx03	Software Part Number	R	15
ptxx01	Print Template 1	R/W	40
ptxx11	Print Literal 1	R/W	40

- Refer to the **IND780 Shared Data Reference** provided on the IND780 documentation CD, part number 64057241.

1.3.7.2. Block Transfer Data

Block transfer allows the IND780 terminal and PLC to exchange many types of data in blocks of up to 80 bytes.

Block transfer works concurrently with discrete data. Discrete mode communicates continuously in the background and a block transfer occurs only when the PLC program executes a block transfer read or write instruction. Data transfer is controlled by the PLC.

**1.3.8. Block Transfer Formats**

Table 1-19 and Table 1-20 provide block transfer format information for block transfer write to the terminal and block transfer read from the terminal.

**Table 1-19: Block Transfer Write (Words 0–62) to Terminal**

Base #	0	1	2	3	4	5	6	7	8	9
<b>N#:0</b>	Display Mode*	16 Byte Display String: sent from PLC to terminal shared data if preceding word is non-zero value and discrete display bits are set to 7								8 Byte>> ASCII
<b>N#:10</b>	<<Floating Point Write Field Code: shows where next value will be loaded			Floating Point Write Value		8 Byte ASCII String Write Field Code: shows where the next value will be loaded			40 Byte>>	
<b>N#:20</b>	<<40 Byte String Data. Note: if string is shorter than 40 bytes it must be left justified and null-terminated >>									
<b>N#:30</b>	<< 40 Byte String Data. Note: if string is shorter than 40 bytes it must be left-justified (and null-terminated)>>									8 Byte>> ASCII
<b>N#:40</b>	<<Floating Point Read Field Code: requests FP value for BTR			8 Byte (ASCII) String Read Field Code: requests string value for BTR			Reserved			

Base #	0	1	2	3	4	5	6	7	8	9
N#:50	Reserved									
N#:60	Reserved									

**Table 1-20: Block Transfer Read (Words 0–62) from Terminal**

Base #	0	1	2	3	4	5	6	7	8	9
N#:0	8 Byte (ASCII) Floating Point Read Field Code: name of value sent in next field				Floating Point Read Value		8 Byte (ASCII) String Read Field Code: name of string sent in next field			
N#:10	40 Byte Data String>>									
N#:20	<<40 Byte String Data. Note: if string is shorter than 40 bytes it must be left-justified (and null-terminated)>>									
N#:30	Reserved									
N#:40	Reserved									
N#:50	Reserved									
N#:60	Reserved									

\* Display Mode: The integer value of this word determines how the IND780 display operates: 0 = reset display to normal mode, 1 = display until overwritten by PLC or ENTER is pressed, 2 = display for 30 seconds, 3 = display for 60 seconds, any value > 3 = reserved.

All Field Codes are six right-justified bytes expanded to eight with two leading spaces.

Example SD = wt0101 . Hex value of field code = 2020 7774 3031 3031

### 1.3.9. Controlling the Discrete I/O Using a PLC Interface

The IND780 terminal provides the ability to directly control its discrete outputs and read its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the terminal's discrete I/O updates are synchronized with the terminal's interface update rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the IND780 Terminal Technical Manual for discrete I/O wiring.

## 1.4. Hardware Setup

### 1.4.1. Wiring

The IND780 terminal's A-B RIO option card uses a three-position removable terminal strip to connect to the A-B RIO network interface (Figure 1-4). Cable distance, type, and termination are specified by Allen-Bradley (See Allen-Bradley documentation for reference on cable design guidelines for the various PLCs). Mettler-Toledo recommends Belden 9463 cable.

The connector comes with the RIO option kit.

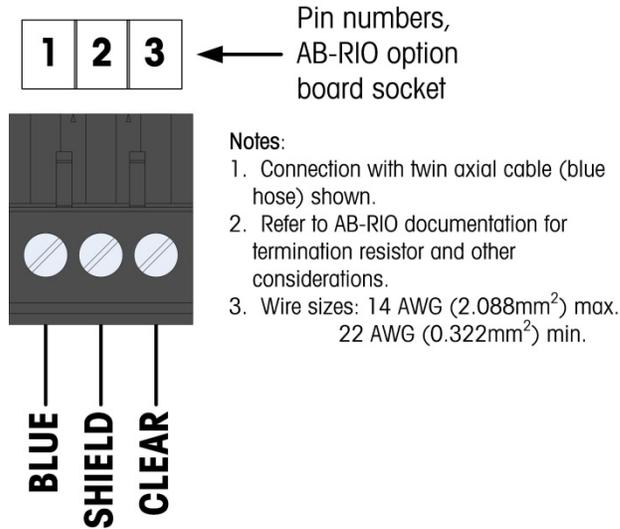


Figure 1-4: Three-Position Terminal Strip Wiring

## 1.5. Software Setup

The IND780 terminal automatically detects the presence of an A-B RIO Kit option board if one is installed. When an A-B Rio PLC module is detected, the IND780 enables the Allen-Bradley parameters in setup at **Communication > PLC Interface > A-B RIO**. Figure 1-5 shows the A-B RIO program block and Figure 1-6 the PLC data format setup block. Default settings are indicated with an asterisk.

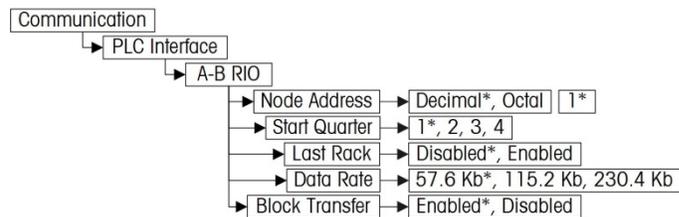
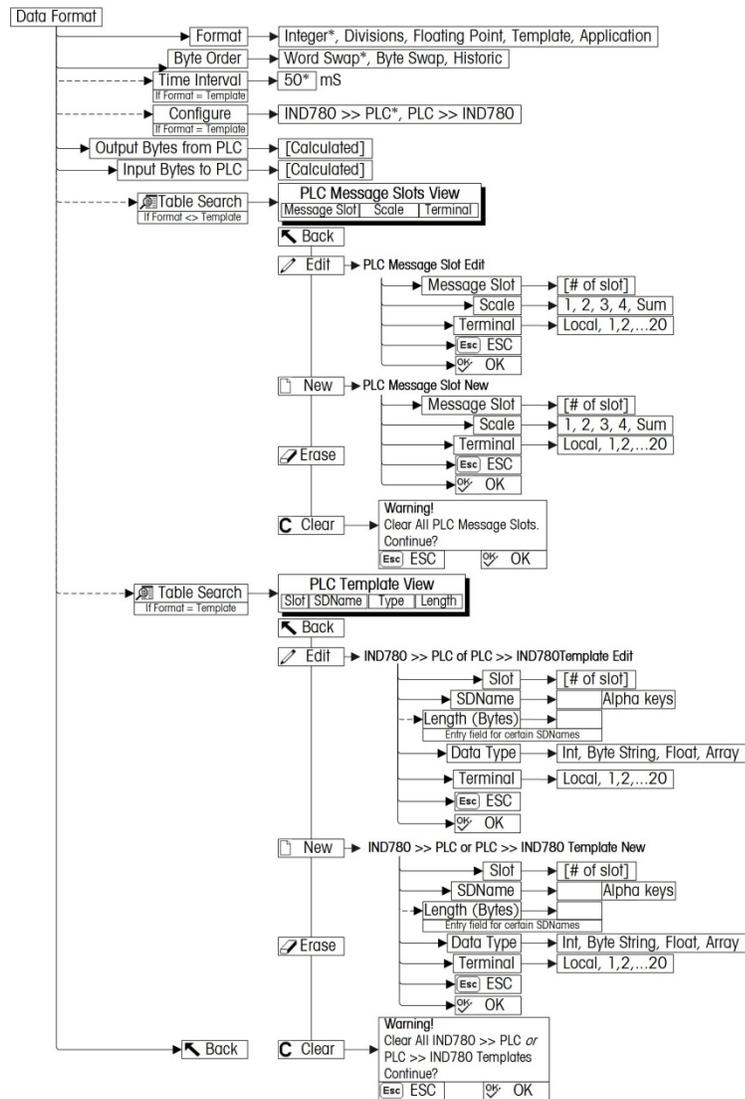


Figure 1-5: Allen-Bradley RIO Program Block



**Figure 1-6: PLC Data Format Setup Block**

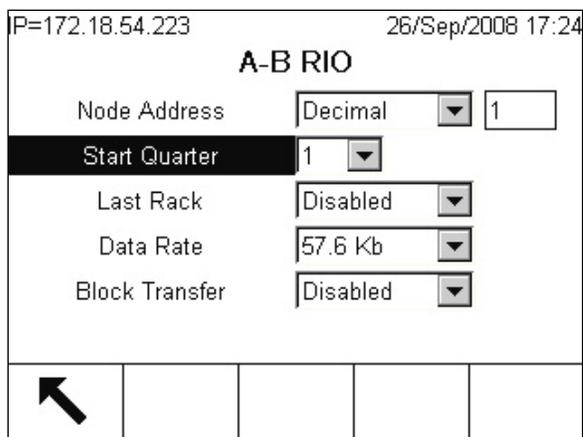
### 1.5.1. A-B RIO and Data Format Setup Blocks

The Scale Setup block lets you specify how the RIO interface is used. Several options are available to correspond with your system setup.

#### 1.5.1.1. A-B RIO setup

To configure the block:

1. From Setup select **Communication > PLC > A-B RIO** (Figure 1-7):



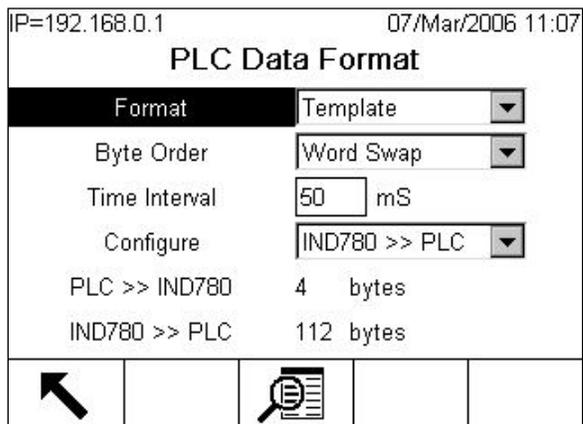
**Figure 1-7: A-B Rio Setup Screen**

2. Select the **Node Address** (0–59 Dec or 0-73 Octal)
3. Select **Start Quarter** (1-4)
4. Select **Last Rack** (Enabled, Disabled).
5. Select **Data Rate** (57.6K, 115.2K, 230.4K).
6. Select **Block Transfer** (Enabled, Disabled).

1.5.1.2.

Data Format setup

1. In setup, select **Communication > PLC Interface > Data Format**. Figure 1-8 shows the screen with Template selected as the format, revealing the Time Interval, Configure, and directional displays. When Application format is selected, only the Format box appears on this screen.



**Figure 1-8: PLC Data Format Screen**

2. Select the Format (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the Format will delete any existing Message Slots.

- Integer** Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).
- Divisions** Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.
- Floating** Displays weight in floating point data format

**Point**

**Template** Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed “on the fly” by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the Shared Data Reference Manual (document number 64059110) provided on the CD-ROM supplied with your terminal.

**Application** Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

3. Select **Byte Order** (see Table 1-21): [default Word Swap]

**Table 1-21: PLC Data Byte-Ordering**

	Byte Swap	Word Swap	Historic
Integer	1 2	2 1	2 1
Float	1 2 3 4	4 3 2 1	2 1 4 3
String	A B C D	ABCD	N/A

**Word Swap** Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

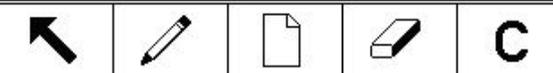
**Byte Swap** Makes the floating point format compatible with S7 PROFIBUS.

**Historic** Makes the floating point data format compatible with PLC 5

4. Set the **Time Interval** in milliseconds. (Only when the Format is Template)
5. The bottom of this screen (Figure 1-8) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.
6. Use the **Configure** drop-down list to select the direction of data flow – IND780 >> PLC or PLC >> IND780. (Only when the Format is Template)

7. Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of **Message Slots** (1, 2, 3, or 4. In Block Transfer mode, maximum number is 3.) [default is 0], or to configure the Template for communications between the IND780 and the PLC. Figure 1-9 shows the Message Slots View, which is viewed whenever the Format is **not** Template.

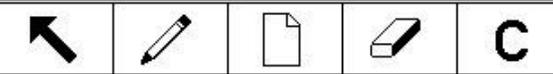
IP=172.18.54.111 06/Dec/2007 16:25		
PLC Message Slots View		
Message	Scale	Terminal
▶ 1	1	Local
2	- End -	



**Figure 1-9: PLC Message Slots View Screen**

Figure 1-10 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1 07/Mar/2006 11:08			
PLC >> IND780 Template View			
Slot	SDName	Type	Length
▶ 1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		



**Figure 1-10: PLC Template View**

Press the EXIT softkey  to exit the view screen.

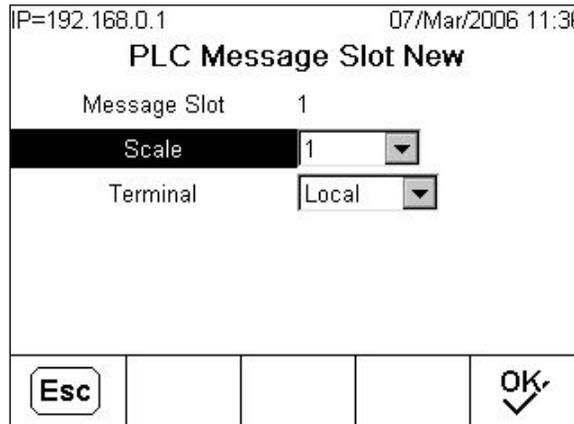
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey  to delete all existing slots.

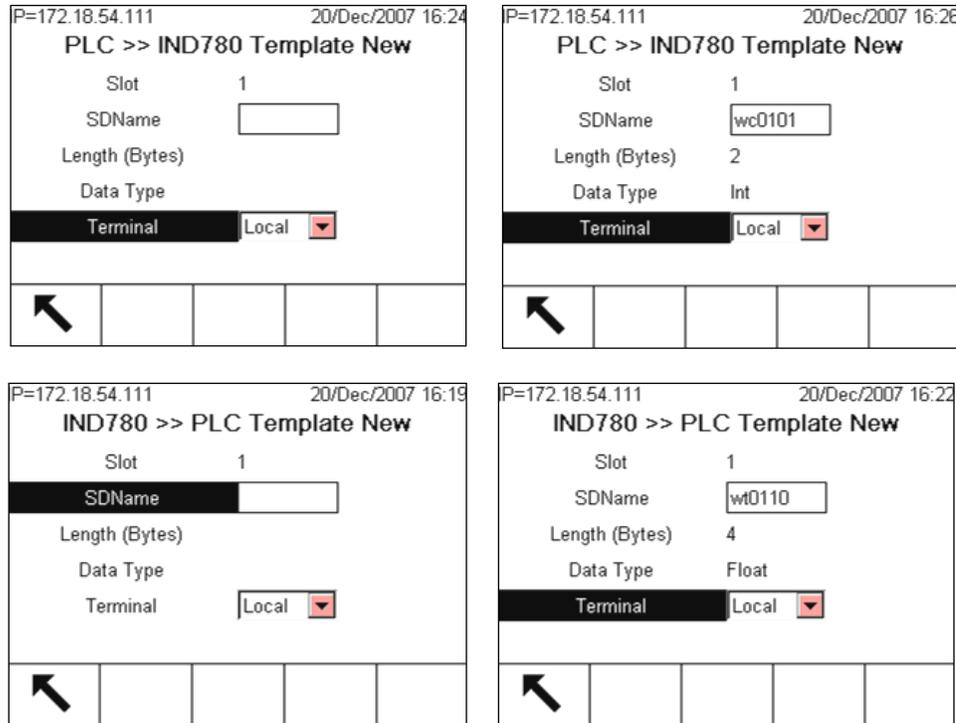
Figure 1-11 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 1-9).



**Figure 1-11: New Message Slot Screen**

- Scale** Enter the scale number to be associated with the new Message Slot.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 1-12 shows a series of Template New screens, as accessed from the Template View screen (Figure 1-10).



**Figure 1-12: New Template Screens, PLC to IND780 and IND780 to PLC**

- Slot** Automatically assigned.

<b>SDName</b>	An alphabetic entry field used to define the Shared Data variable associated with the slot.
<b>Length (Bytes) and Data Type</b>	Automatically displayed once the SDName has been entered. The maximum length of a Shared Data variable cannot exceed 16 bytes when Block Transfer is disabled, or 14 bytes when Block transfer is enabled.
<b>Terminal</b>	Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

- Refer to the Table 1-7 (Discrete Read) and Table 1-8 (Discrete Write) in this manual for additional information on mapping of discrete read data to the PLC.

## 1.6. Troubleshooting

If the IND780 does not communicate with PLC do the following:

- Check wiring and network termination.
- Confirm that the IND780 settings for data type and rack assignment match those in the PLC.
- Replace the RIO interface kit if the problem persists.
- If the communication kit was changed from another type, like PROFIBUS, ControlNet or EtherNet I/P, a master reset of the IND780 must be performed.

### 1.6.1. Status LEDs

The RIO card has an LED indicator to indicate communication status of the card. Figure 1-13 indicates the location of the LED, and Table 1-22 explains its indications.



Figure 1-13: A-B RIO Card Status LED

Table 1-22: A-B RIO Status LED Indications

LED Indications	Meaning
No LED	Bad card or no power. Check cable wiring or PLC power.
Solid Green	Online and communicating
Blinking LED	Power to network but node not communicating to PLC. Check for rack size or data size mismatch.

## 1.7. Allen-Bradley RIO Option Kit

There are no associated spare parts with the RIO option kit. The kit CIMF part number is 71209098. Table 1-23 shows what comes in the kit.

**Table 1-23: A-B RIO Option Kit**

Description	Qty.
Installation Instructions	1
PCB Package	1
Cylindrical ferrites	3
M3 x 8 screws	3

## 1.8. Programming Examples

The following five Figures show sample screen images from ladder logic programming examples for RSLogix 5000 software (version 13). These examples only demonstrate scanner and adapter configuration, plus conversion of data to real numbers. The floating point example includes Block Transfer functions.

- **Note:** The Utilities folder of the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.

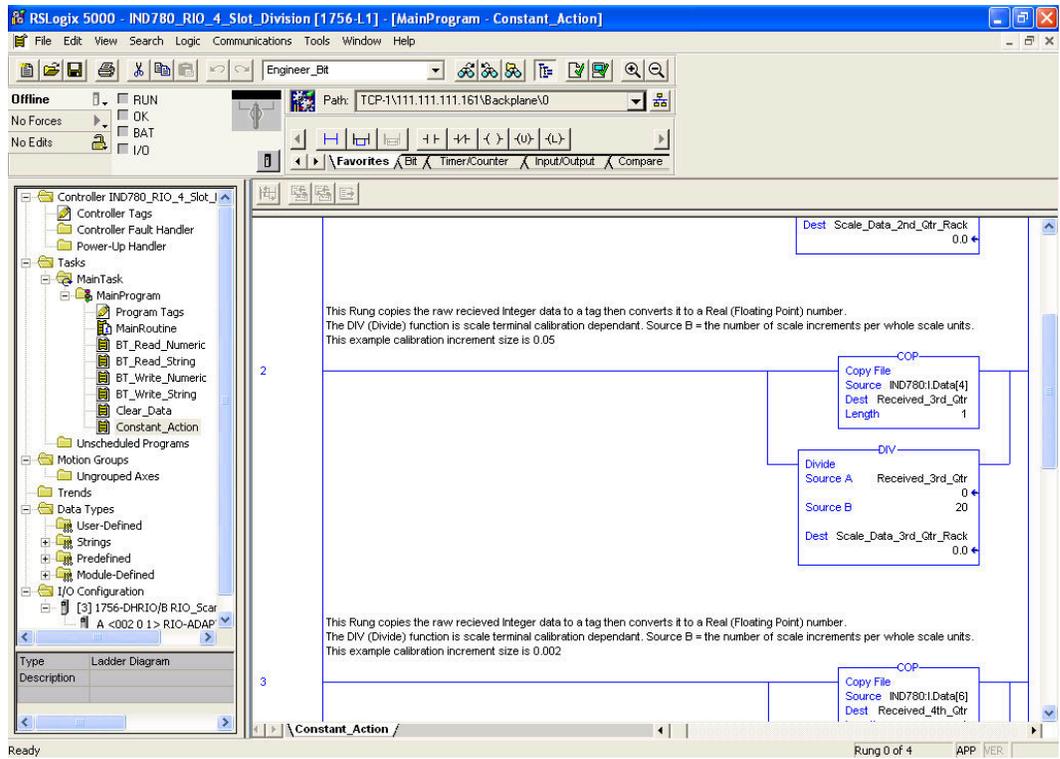


Figure 1-14: IND780 Divisions, 4 Message Slots

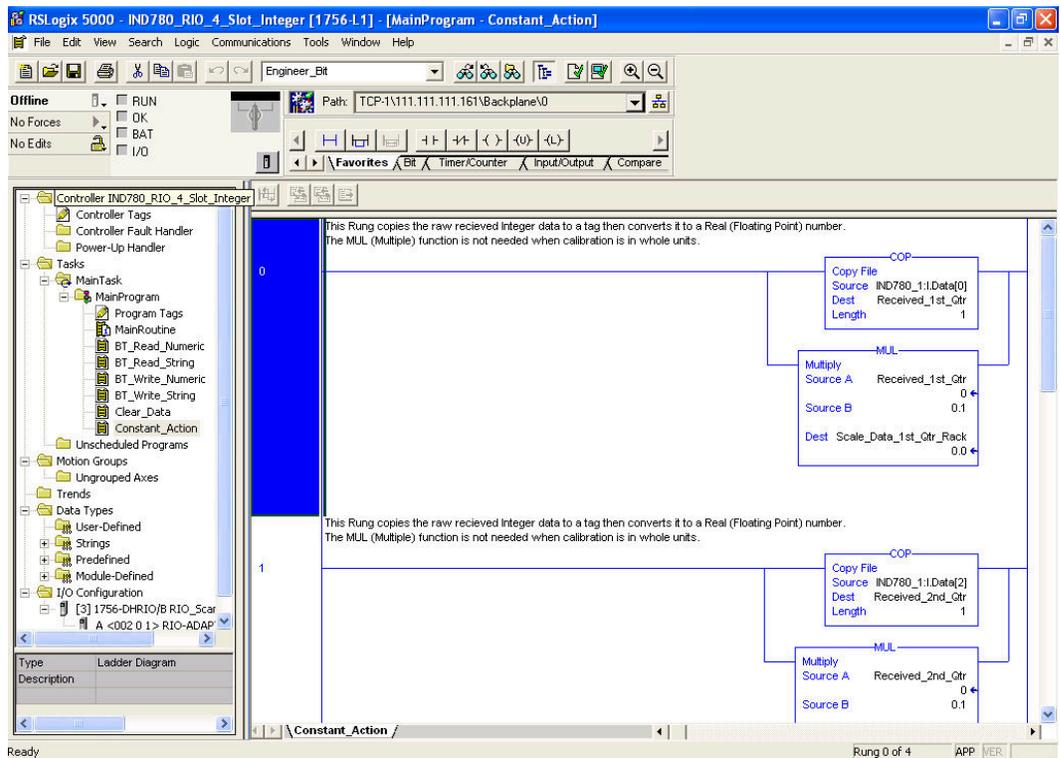
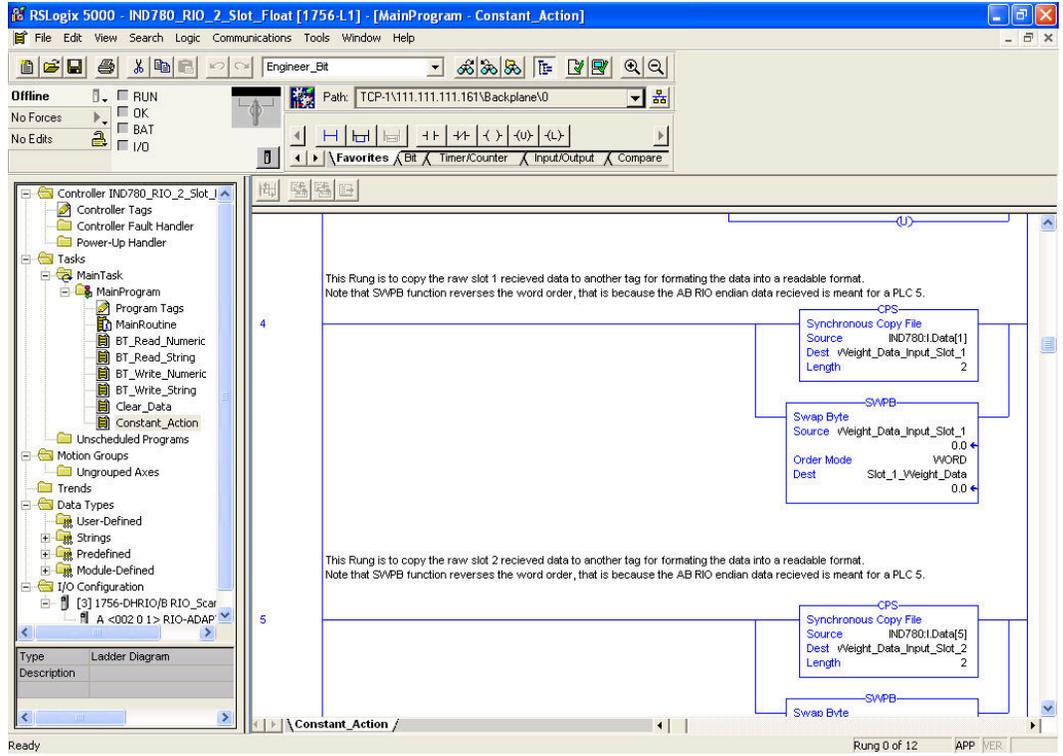
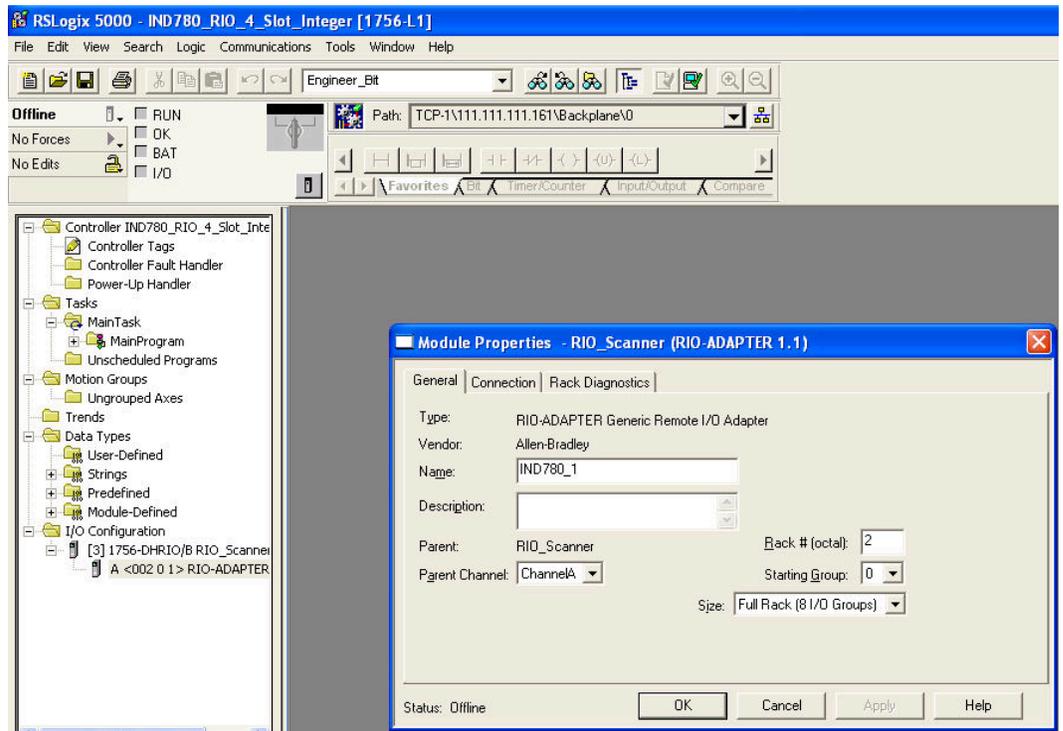


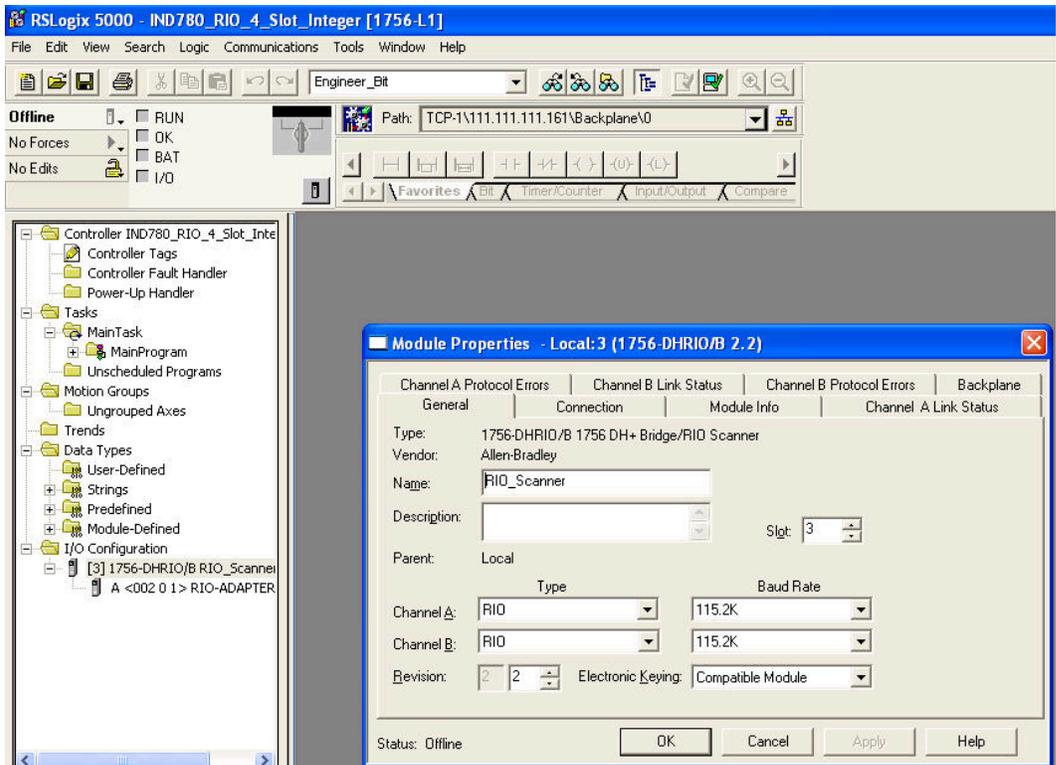
Figure 1-15: IND780 Integer, 4 Message Slots



**Figure 1-16: IND780 Floating Point, Historic Format**



**Figure 1-17: IND780 Adapter Configuration, Integer, 4 message slots**



**Figure 1-18: RIO Scanner Configuration**

# 2 Analog Output

The Analog Output option kit provides two channels of an isolated 4-20 mA analog signal output for displayed weight, gross weight, rate or application variables (selected in setup). The analog option uses a 16-bit D/A converter for a very precise output. The output signals will be at the lower limit (4 mA) when the value represented is at zero. When the value reaches its maximum limit, the output signal will increase to the higher limit (20 mA). Any value between zero and the maximum limit will be represented as a percentage of the output proportional to the percentage of the value.

The setup block at **Communication > PLC Interface > Analog Output** permits the selection of the data source for the analog signal and provides a method to calibrate the analog zero and high limit values. The IND780 terminal must be calibrated to the desired scale capacity before making Analog Output adjustments. If rate is to be used as the source for the analog output signal, the rate function must be enabled in setup at **Scale > Rate**. One or two Analog Output boards can be installed in the IND780 terminal, to support up to four channels of 4-20 mA analog output.

## 2.1. Specifications

<b>Maximum Cable Length</b>	300 m (1000 ft.)
<b>Max Load Resistance</b>	500 $\Omega$ maximum
<b>Outputs</b>	2 channels capable of supplying 4-20 mA
<b>Resolution</b>	16 bit resolution – 65536 levels across entire range
<b>Error Indication</b>	Two open collector discrete output error signals per channel to indicate out-of-range conditions of the terminal, labeled Under and Over.

- Note that if the maximum load resistance value is exceeded, the analog output will not operate properly.
- Figure 2-1 shows the Analog Output Option Board with its connector at top right.

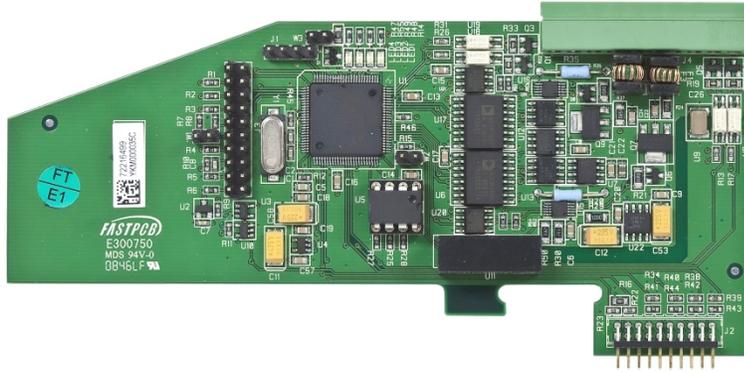


Figure 2-1: Analog Output Option Board

## 2.2. Analog Output Operation

The source for the analog output can be set to **Displayed Weight**, **Gross Weight**, **Rate** or custom **Application Variables**. Regardless of the source selected, the analog output signal operates as described below.

- In order to use Rate as the source, it must be enabled in setup at **Scale > Rate**.

- Under Zero** When the source value drops below zero, the analog signal continues to decrease below 4 mA. This continues until approximately 1 mA. When the under zero display blanking point is reached, or the analog signal negative range is exceeded (< 1 mA), the analog output immediately switches to 0 mA and remains there until the display is no longer blanked or the analog signal returns to within range. The Under error output will also be turned on.
- Normal Range** The analog output will reflect the source value from 4 mA (zero setting) to 20 mA (high limit, Full Scale setting).
- Over High Limit** When the source value exceeds the Full Scale value setting, the analog signal continues to increase. This continues until approximately 23 mA. When the overcapacity display blanking point is reached, or the analog signal positive range is exceeded (> 23 mA), the analog output immediately switches to approximately 24 mA and remains there until the weight display is no longer blanked or the analog signal returns to within range. The Over error output will also be turned on.

Table 2-1 summarizes the function of the analog output under zero and over the high limit, and the operation of the Under and Over discrete output error signals.

Table 2-1: Analog Signal Operation

Error Outputs and Analog Signal Operation			
Condition of Weight or Rate	Analog Signal	Over Error	Under Error
Blanked under zero (blanking range 5d)	0 mA	OFF	ON
Under zero with < 1 mA of analog signal (blanking range disabled)	0 mA	OFF	ON
Under zero but not blanked (blanking range 5d)	3.95 - 4.0 mA	OFF	OFF

Error Outputs and Analog Signal Operation			
Condition of Weight or Rate	Analog Signal	Over Error	Under Error
Under zero with $\geq 1$ mA of analog signal and analog output still functional (blinking range disabled)	1 – 4.0 mA	OFF	OFF
Normal operating range	4 - 20 mA	OFF	OFF
Above full scale value but with $\leq 23$ mA of analog signal (display not blanked)	20 – 23 mA	OFF	OFF
Above full scale value with $> 23$ mA of analog signal, but not over capacity blanked	24 mA	ON	OFF
Blanked over capacity, analog out of range or weight data unavailable	24 mA	ON	OFF

## 2.3. Installation

	<b>! WARNING</b>
	<b>DISCONNECT ALL POWER TO THIS UNIT BEFORE SERVICING. DO NOT APPLY POWER TO THE TERMINAL UNTIL INSTALLATION OF COMPONENTS AND EXTERNAL WIRING HAS BEEN COMPLETED.</b>
	<b>! WARNING</b>
	<b>THE ANALOG OUTPUT BOARD KIT NUMBER 72230302 MUST NOT BE USED IN AN IND780 TERMINAL INSTALLED IN AN AREA CLASSIFIED AS DIVISION 2 OR ZONE 2/22. FAILURE TO COMPLY WITH THIS WARNING COULD RESULT IN BODILY HARM AND/OR PROPERTY DAMAGE.</b>
	<b>NOTICE</b>
	<b>OBSERVE PRECAUTIONS FOR HANDLING ELECTROSTATIC SENSITIVE DEVICES.</b>

The analog output option for all enclosure types can be installed at the factory or it can be ordered as a kit and installed in the field. The option kit includes detailed drawings to assist in the installation. The analog output option board can only be installed in slot locations 2 thru 6 on the terminal's main board.

The recommended wiring for the analog output is 2-conductor, 20 GA cable available from METTLER TOLEDO (part number 510220190). It is equivalent to Belden #8762.

## 2.4. Wiring

	 <b>WARNING</b>
	<b>DISCONNECT ALL POWER TO THIS UNIT BEFORE SERVICING. DO NOT APPLY POWER TO THE TERMINAL UNTIL INSTALLATION OF COMPONENTS AND EXTERNAL WIRING HAS BEEN COMPLETED.</b>
 <b>WARNING</b>	
<b>IF THIS DEVICE IS USED IN AN AUTOMATIC OR MANUAL FILLING CYCLE, ALL USERS MUST PROVIDE A HARD-WIRED EMERGENCY STOP CIRCUIT OUTSIDE THE DEVICE. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.</b>	

The maximum recommended cable length for the 4-20 mA output is 1,000 feet (300 meters). The recommended cable for use with the analog output is shielded two-conductor stranded 20-gauge cable (Belden #8762 or equivalent), which is available from METTLER TOLEDO using part number 510220190. Refer to Figure 2-2 for connection and termination information.

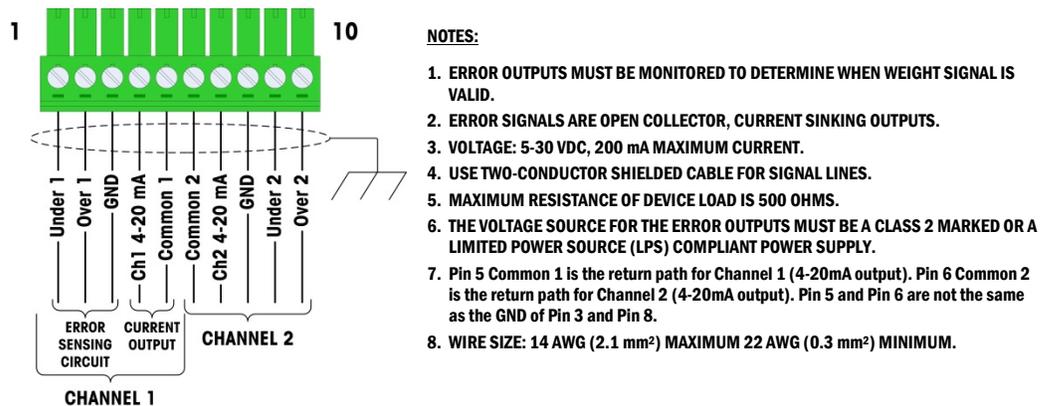
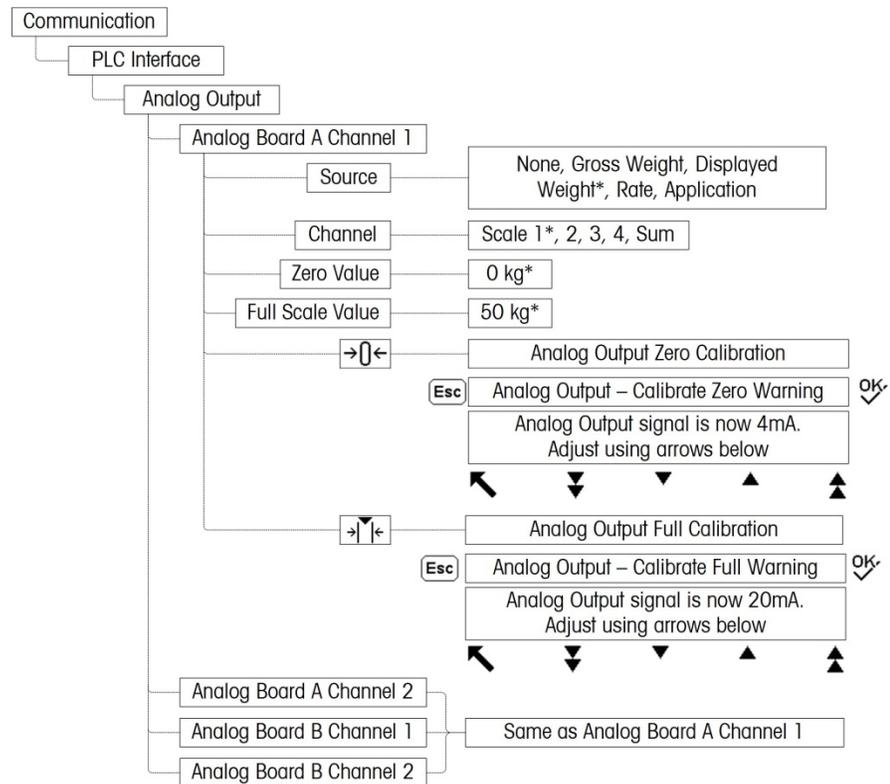


Figure 2-2: Analog Output Kit Wiring Connections

## 2.5. Setup in the IND780 Terminal

The IND780 terminal automatically detects the presence of an Analog Output option board if one is installed. When an Analog Output option is detected, the IND780 enables the Analog Output parameters at **Setup > Communication > PLC Interface > Analog Output**. Figure 2-3 illustrates the setup menu for configuring the Analog Output option and its respective channels for the IND780 terminal. Default settings are indicated with an asterisk.



**Figure 2-3: Setup Procedures for Configuring the Analog Output Option Card**

### 2.5.1. Analog Output Setup Sub-Block

To configure the Analog Output Kit Option:

1. With power to the IND780 terminal removed, connect a current meter to the appropriate 4-20mA output channel. If the customer's device is already connected, the meter is not necessary.
2. Apply power to the terminal and enter Setup. Navigate to **Communication > PLC Interface**.
3. Select the **Analog Output** branch and the respective Analog Output channel to configure. The Analog Output screen will appear, as shown in Figure 2-4.

IP=172.18.54.97		26/Aug/2009 14:32	
<b>Analog Board A Channel 1</b>			
Source	Displayed Weight ▾		
Channel	Scale 1 ▾		
Zero Value	0	kg	
Full Scale Value	50	kg	
Option Location	4 ▾		
↶		→0←	
→ ←		↵	

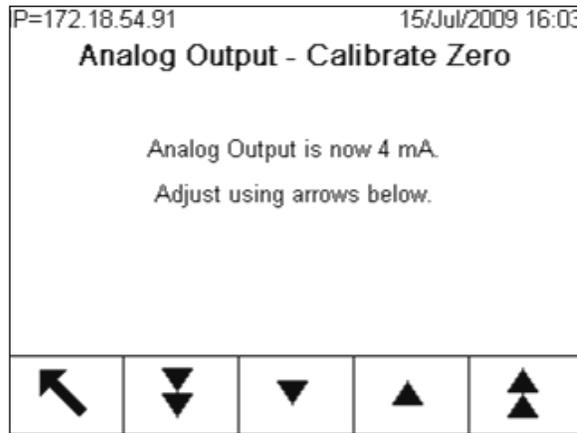
1. **Figure 2-4: Analog Output 1 Screen**

4. The following settings are made in this screen:

<b>Source</b>	Choices are <b>None</b> , <b>Displayed Weight</b> (the default), <b>Gross Weight</b> , <b>Rate</b> and <b>Application</b> . <b>None</b> disables the analog output. <b>Displayed Weight</b> outputs an analog signal based on the displayed net or gross weight. When <b>Gross Weight</b> is selected, the analog signal is based on the gross weight regardless of what the net weight might be. In order to be available as a source, <b>Rate</b> must be configured at <b>Scale &gt; Rate</b> . The <b>Application</b> selection allows a TaskExpert program to assign an application variable as the source and to control the operation of the analog output. Refer to the <b>TaskExpert Reference Manual</b> (64060431) for details.
<b>Channel</b>	Options are <b>Scale 1</b> , <b>Scale 2</b> , <b>Scale 3</b> , <b>Scale 4</b> , or <b>Sum Scale</b> (if they are enabled). This selection refers to the source channel.
<b>Zero Value</b>	Enter the desired source value at which the analog output should be at 4 mA (zero) level. Typically this would be "0" in most applications; however, any valid value below the Full Scale value can be used.
<b>Full Scale Value</b>	Enter the desired source value at which the analog output should be at its 20 mA (high limit) level. For sources of weight, this would typically be scale capacity, but it could be lower. For rate, this should be the rate value that should provide a full analog output signal.
<b>Option Location</b>	Since up to two Analog Output option boards can be installed in the IND780, the <b>Option Location</b> selection specifies which of the option board slot locations to apply to the currently selected Analog Output channel. Choices include <b>None</b> , <b>2</b> , <b>3</b> , <b>4</b> , <b>5</b> and <b>6</b> .

5. After all these parameters have been entered, the analog output signal can be adjusted to meet the customer's specific requirements using the CAPTURE ZERO softkey →0← and the CAPTURE SPAN softkey →|←.
6. To adjust the zero reference analog signal, press the CAPTURE ZERO softkey →0←. Note that a display message is shown warning that during the adjustment, the analog output signal will change value indicating that it will not represent the source value. Press the ESCAPE softkey **[Esc]** to exit the zero adjustment process or press the OK softkey **OK** to continue the adjustment process.

- At the **Analog Output - Calibrate Zero** screen (Figure 2-5), use the softkeys to adjust the analog output signal to be exactly zero on the attached device. The available softkeys are described in Table 2-2.

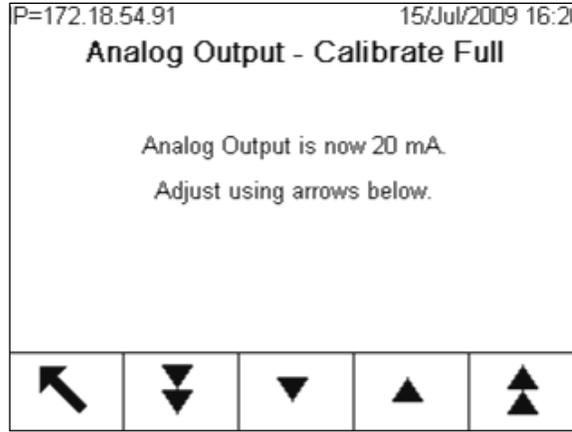


**Figure 2-5: Analog Output – Calibrate Zero Screen**

**Table 2-2: Analog Output Calibrate Softkey Descriptions**

	<b>Coarse Down</b>	Decreases the analog signal level in large steps.
	<b>Fine Down</b>	Decreases the analog signal level in small steps.
	<b>Fine Up</b>	Increases the analog signal level in small steps.
	<b>Coarse Up</b>	Increases the analog signal level in large steps.

- When the zero adjustment is complete, press the EXIT softkey  to return to the Analog Output screen.
- Now, the full scale analog output value can be adjusted by pressing the CAPTURE SPAN softkey . A similar warning message will be shown indicating the analog output will change and will not monitor changes in the source. Press the ESCAPE softkey **Esc** to exit the span adjustment process or press the OK softkey  to continue the adjustment process.
- At the **Analog Output - Calibrate Full** screen (Figure 2-6), use the softkeys to adjust the analog output signal to be exactly what the customer's device requires for its high limit. The available softkeys are described in Table 2-2.

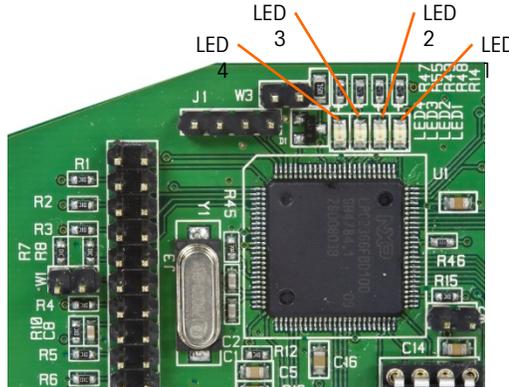


**Figure 2-6: Analog Output – Calibrate Full Screen**

11. When the full scale adjustment is complete, press the EXIT softkey  to return to the Analog Output screen.

## 2.6. Status LEDs

The Analog Output option board has four LEDs to indicate status of the card for troubleshooting purposes. Figure 2-7 indicates the location of the LEDs and Table 2-3 explains its indications.



**Figure 2-7: Analog Output LEDs**

**Table 2-3: Analog Output Status LED Indications**

LEDs	Color	Meaning
LED 1: USB	Green	Solid ON: USB communicating with IND780 main board
LED 2: Channel 1	Green	Solid ON: Channel 1 is functioning
LED 3: Channel 2	Green	Solid ON: Channel 2 is functioning
LED 4: Board OK	Green	Slow blink: Checking board communication and function Fast blink: Board communicating and functioning

## 2.7. Analog Output Option Kit

No spare parts are associated with the analog output option kit. The kit CIMF part number is 72230302. Table 2-4 shows what is included in the kit.

**Table 2-4: Analog Output Option Kit**

Description	Qty.
Installation Instructions	1
PCB Package	1
Installation Kit	1

# 3 ControlNet

## 3.1. Overview

The ControlNet Kit option enables the IND780 terminal to communicate to ControlNet Programmable Logic Controllers (PLCs) through direct connection to the ControlNet network. The kit consists of a backplane-compatible I/O module and software that resides in the IND780 terminal to implement the data exchange. Figure 3-1 shows two views of the module, and Figure 3-2 points out its components.

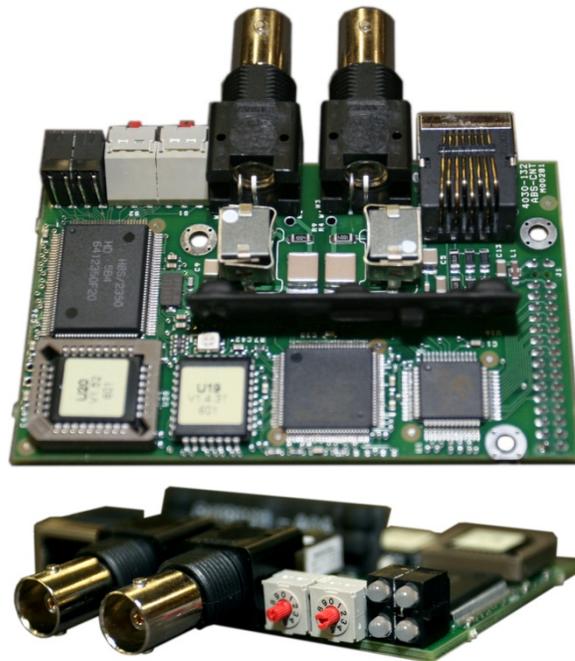


Figure 3-1: ControlNet PLC Module

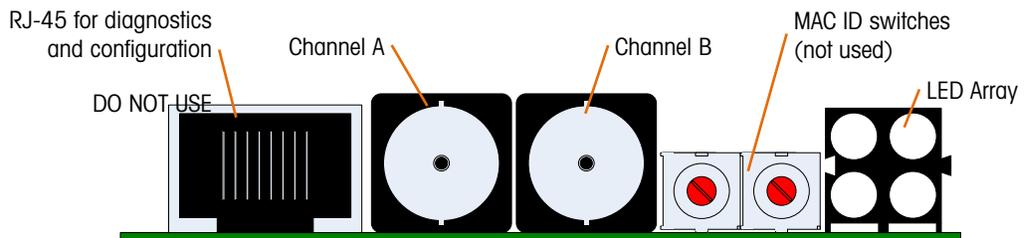


Figure 3-2: ControlNet PLC Module Components

- Do **not** plug an Ethernet cable into the RJ-45 connector shown at left in Figure 3-2. It is for diagnostic use only.

The ControlNet Kit option has the following features:

- User-programmable node (MAC ID) address.
- Capability for bi-directional discrete mode communications (Class 1 Messaging) of weight or display increments, status, and control data between the PLC and the IND780.

### 3.1.1. Definition of Terms

Some terms (such as Target) used by the ControlNet PLC application have a different sense from their use by the IND780 terminal. Table 3-1 offers definitions specific to ControlNet.

**Table 3-1: ControlNet Definition of Terms**

Term	Definition
<b>Adapter Class</b>	An Adapter Class product emulates functions provided by traditional rack-adapter products. This type of node exchanges real-time I/O data with a Scanner Class product. It does not initiate connections on its own.
<b>Class 1 Messaging</b>	In ControlNet communication protocol scheduled (cyclic) message transfer between a PLC and CIP Adapter Class device.
<b>Class 3 Messaging</b>	In ControlNet communication protocol unscheduled message transfer between a PLC and CIP Adapter Class device. This is used by the IND780 for explicit messaging.
<b>Connected Messaging</b>	A connection is a relationship between two or more application objects on different nodes. The connection establishes a virtual circuit between end points for transfer of data. Node resources are reserved in advance of data transfer and are dedicated and always available. Connected messaging reduces data handling of messages in the node. Connected messages can be Implicit or Explicit. See also <b>Unconnected Messaging</b> .
<b>Connection Originator</b>	Source for I/O connection or message requests. Initiates an I/O connection or explicit message connection.
<b>Explicit Messaging</b>	Explicit Messages can be sent as a connected or unconnected message. CIP defines an Explicit Messaging protocol that states the meaning of the message. This messaging protocol is contained in the message data. Explicit Messages provide a one-time transport of a data item. Explicit Messaging provide the means by which typical request/response oriented functions are performed (e.g. module configuration). These messages are typically point-to-point.
<b>Implicit Messaging</b>	Implicit Messages are exchanged across I/O Connections with an associated Connection ID. The Connection ID defines the meaning of the data and establishes the regular/repeated transport rate and the transport class. No messaging protocol is contained within the message data as with Explicit Messaging. Implicit Messages can be point-to-point or multicast and are used to transmit application-specific I/O data. This term is used interchangeably with the term <b>I/O Messaging</b> .
<b>I/O Client</b>	Function that uses the I/O messaging services of another (I/O Server) device to perform a task. Initiates a request for an I/O message to the server module. The I/O Client is a Connection Originator.
<b>I/O Messaging</b>	Used interchangeably with the term <b>Implicit Messaging</b> .
<b>I/O Server</b>	Function that provides I/O messaging services to another (I/O Client) device. Responds to a request from the I/O Client. I/O Server is the target of the connection request.

Term	Definition
<b>Message Client</b>	Function that uses the Explicit messaging services of another (Message Server) device to perform a task. Initiates an Explicit message request to the server device.
<b>Message Server</b>	Function that provides Explicit messaging services to another (Message Client) device. Responds to an Explicit message request from the Message Client.
<b>Scanner Class</b>	A Scanner Class product exchanges real-time I/O data with Adapter Class and Scanner Class products. This type of node can respond to connection requests and can also initiate connections on its own.
<b>Target</b>	Destination for I/O connection or message requests. Can only respond to a request, cannot initiate an I/O connection or message.
<b>Unconnected Messaging</b>	Provides a means for a node to send message requests without establishing a connection prior to data transfer. More overhead is contained within each message and the message is not guaranteed destination node resources. Unconnected Messaging is used for non-periodic requests (e.g. network "Who" function). Explicit messages only. See also <b>Connected Messaging</b> .

### 3.1.2. Communications

The IND780 terminal utilizes component parts to ensure complete compatibility with the Allen-Bradley ControlNet network. An IND780 terminal is recognized as a generic ControlNet device by the PLC.

Each ControlNet option connected to the ControlNet network represents a physical node. The connection is facilitated by BNC connectors on the option card.

The wiring between the PLC and the IND780 ControlNet connection uses RG-6 CATV cable and 75 ohm impedance matching transformer tap for each node. The cable is commonly referred to as coaxial cable. The cable installation procedures and specification including distance and termination requirements are the same as recommended by Allen-Bradley for the ControlNet network. The normal connection is to the channel A connector, the channel B connector is only used for redundant connection networks.

The IND780 terminal's communication update rate is setup by the use of Allen Bradley software Networx for ControlNet.

The IND780 only uses Class 1 cyclic data for discrete data and explicit messages for access to the IND780 Shared Data Variables. Explicit message blocks may be connected or unconnected; the PLC programmer must make this choice.

### 3.1.3. Node Address

Each ControlNet option represents one physical node. This address is chosen by the system designer, and then programmed into the IND780 terminal and PLC. The IND780 terminal's address is programmed in setup at **Communication > PLC Interface > ControlNet**. IND780 address entry is in decimal.

### 3.1.4. Data Formats

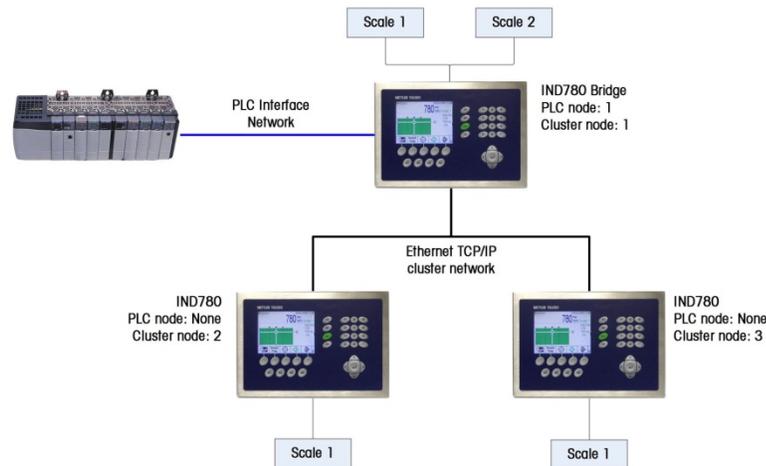
The ControlNet Kit option provides discrete data transfer, Class 1 messaging. Discrete data is continuously available. The ControlNet option has its own logical node address to send and receive

information to and from the PLC. There are five data formats: Integer, Divisions, Floating Point, Template and Application.

## 3.2. Sharing a PLC Interface

### 3.2.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. The following example (Figure 3-3) shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 3-3: Terminal Cluster Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate PLC chapters in this manual for further details on message slots' limitations.

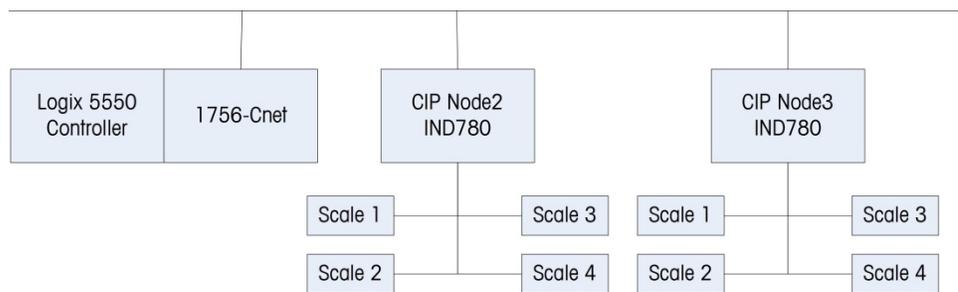
To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure 3-4 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to section 3.5, **Software Setup**, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

P=172.18.54.122		09/Oct/2007 17:20	
PLC Message Slots View			
Message	Scale	Terminal	
1	1	Local	
2	1	6	
3	- End -		

**Figure 3-4: PLC Message Slots View Screen**

Figure 3-5 shows a typical ControlNet network arrangement, with IND780 terminals connected directly to a Logix controller without the use of the terminal’s PLC interface sharing.



**Figure 3-5: Typical Network Topology**

## 3.3. Data Definition

The ControlNet Kit option uses discrete data for its communication with PLCs. Data transfer is accomplished via the PLC’s cyclic messaging.

### 3.3.1. Data Integrity

The IND780 has specific bits to allow the PLC to confirm that data was received without interrupt and the IND780 is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received by the IND780. Refer to the data charts for specific information regarding the Data OK, Update in Progress, Data Integrity bits and their usage.

### 3.3.2. Assembly Instances of Class 1 Cyclic Communications

Class 1 cyclic communications is used for transfer of Discrete Data between the PLC and the IND780.

The PLC Input Assembly Instance is 100 (decimal). This instance is used for all Data Formats and data size requirements.

The PLC Output Assembly Instance is 150 (decimal). This instance is used for all Data Formats and data size requirements.

The IND780 uses data only; no configuration data is used or required. Within the PLC ControlNet Interface setup set the Configuration Instance to 1 and the data size to zero.

The EDS file provided on the Documentation CD has no Assembly Instance or data size limitations. The IND780 programming controls the Assembly Instance and data size limitations.

### 3.3.3. Discrete Data

Five formats of discrete data are available with the ControlNet Kit option: Integer, divisions, floating point, template and application. Only one type of data format may be selected and used by the IND780.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word numerical values(see Table 3-6 and Table 3-7).

The floating-point format allows bi-directional communication of discrete bit encoded information and numeric data encoded in IEEE 754, single precision floating point format.

The data format of discrete data will affect the data size required in the configuration of the PLC. The IND780 console PLC message slot setup screen provides data size requirements in bytes, except for the Application data format.

Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions or increments. The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Floating point is the only data format that includes decimal point information. Integer and division formats ignore decimal points. Accommodation of decimal point location must occur in the PLC logic, when it is needed with these formats.

Changing the Data Format to be used by the IND780 will clear all Message Slots. Data format is set up in the **Communication > PLC Interface > Data Format** screen.

#### 3.3.3.1.1.

Examples

250 x .01 scale				
IND780 Displays:		2.00	51.67	250.00
Format sent:				
Integer	0	200	5167	25000
Division	0	200	5167	25000
Floating Point	0	2.00	51.67	250.00

Any of the formats could be used in this case.

50,000 x 10 scale				
IND780 Displays:		200	5160	50000
Format sent:				

50,000 x 10 scale				
Integer	0	200	5160	-(15536)
Division	0	20	516	5000
Floating Point	0	200	5160	50000

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

150 x .001 scale				
IND780 Displays:		2.100	51.607	150.000
Format sent:				
Integer	0	2100	-(13929)	18928
Division	0	2100	-(13929)	18928
Floating Point	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32.767.

### 3.3.4. Byte Order

**Word Swap** takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** makes the floating point format compatible with S7 PROFIBUS.

### 3.3.5. Message Slots

There may be up to 12 message slots for discrete data transfer, Class 1 messaging, in Integer, Divisions and Floating Point Data Formats. Each message slot is assigned to a local or remote scale and scales may be repeated in additional message slots. Remote scales must reside in an IND780 Ethernet TCP clustered to the IND780 containing the ControlNet interface. The integer and division formats provide (two 16-bit words of input and two 16-bit words of output data) per Message Slot. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. Table 3-2 and Table 3-3 provide input and output usage information.

**Table 3-2: ControlNet PLC Input Data and Data Usage (Integer and Division)**

PLC Input Bytes	PLC Input Words	Usage
0	0	Reserved
1		Reserved
2	1	Reserved
3		Reserved
4	2	Message Slot 1 Weight Data
5		Message Slot 1 Weight Data
6	3	Message Slot 1 Scale Status
7		Message Slot 1 Scale Status
8	4	Message Slot 2 Weight Data
9		Message Slot 2 Weight Data

PLC Input Bytes	PLC Input Words	Usage
10	5	Message Slot 2 Scale Status
11		Message Slot 2 Scale Status
12	6	Message Slot 3 Weight Data
13		Message Slot 3 Weight Data
14	7	Message Slot 3 Scale Status
15		Message Slot 3 Scale Status
16	8	Message Slot 4 Weight Data
17		Message Slot 4 Weight Data
18	9	Message Slot 4 Scale Status
19		Message Slot 4 Scale Status
20	10	Message Slot 5 Weight Data
21		Message Slot 5 Weight Data
22	11	Message Slot 5 Scale Status
23		Message Slot 5 Scale Status
24	12	Message Slot 6 Weight Data
25		Message Slot 6 Weight Data
26	13	Message Slot 6 Scale Status
27		Message Slot 6 Scale Status
28	14	Message Slot 7 Weight Data
29		Message Slot 7 Weight Data
30	15	Message Slot 7 Scale Status
31		Message Slot 7 Scale Status
32	16	Message Slot 8 Weight Data
33		Message Slot 8 Weight Data
34	17	Message Slot 8 Scale Status
35		Message Slot 8 Scale Status
36	18	Message Slot 9 Weight Data
37		Message Slot 9 Weight Data
38	19	Message Slot 9 Scale Status
39		Message Slot 9 Scale Status
40	20	Message Slot 10 Weight Data
41		Message Slot 10 Weight Data
42	21	Message Slot 10 Scale Status
43		Message Slot 10 Scale Status
44	22	Message Slot 11 Weight Data
45		Message Slot 11 Weight Data
46	23	Message Slot 11 Scale Status
47		Message Slot 11 Scale Status
48	24	Message Slot 12 Weight Data
49		Message Slot 12 Weight Data
50	25	Message Slot 12 Scale Status

PLC Input Bytes	PLC Input Words	Usage
51		Message Slot 12 Scale Status

**Table 3-3: ControlNet PLC Output Words and Word Usage (Integer and Division)**

PLC Output Bytes	PLC Output Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Command
3		Message Slot 1 Scale Command
4	2	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Command
7		Message Slot 2 Scale Command
8	4	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	5	Message Slot 3 Scale Command
11		Message Slot 3 Scale Command
12	6	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	7	Message Slot 4 Scale Command
15		Message Slot 4 Scale Command
16	8	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data
18	9	Message Slot 5 Scale Command
19		Message Slot 5 Scale Command
20	10	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	11	Message Slot 6 Scale Command
23		Message Slot 6 Scale Command
24	12	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	13	Message Slot 7 Scale Command
27		Message Slot 7 Scale Command
28	14	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	15	Message Slot 8 Scale Command
31		Message Slot 8 Scale Command
32	16	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	17	Message Slot 9 Scale Command
35		Message Slot 9 Scale Command
36	18	Message Slot 10 Weight Data

PLC Output Bytes	PLC Output Words	Usage
37		Message Slot 10 Weight Data
38	19	Message Slot 10 Scale Command
39		Message Slot 10 Scale Command
40	20	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	21	Message Slot 11 Scale Command
43		Message Slot 11 Scale Command
44	22	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	23	Message Slot 12 Scale Command
47		Message Slot 12 Scale Command

The floating point format provides four 16-bit words of input data and three 16-bit words of output data) per Message Slot. See Table 3-4 and Table 3-5 for details.

The number of Message Slots is setup in **Communication > PLC Interface > Data Format** setup screen (Figure 3-11, Figure 3-12).

**Table 3-4: ControlNet PLC Floating Point Input Words**

PLC Output Words	Bits 0 – 7	Bits 8 - 15
0	Reserved	Reserved
1	Reserved	Reserved
2	Message Slot 1, Reserved	Message Slot 1, Command Response. Refer to Message Slot Table
3	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
4	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
5	Message Slot 1, Scale Status. Refer to Message Slot Table	Message Slot 1, Scale Status. Refer to Message Slot Table
6	Message Slot 2, Reserved	Message Slot 2, Command Response. Refer to Message Slot Table
7	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
8	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
9	Message Slot 2, Scale Status. Refer to Message Slot Table	Message Slot 2, Scale Status. Refer to Message Slot Table
10	Message Slot 3, Reserved	Message Slot 3, Command Response. Refer to Message Slot Table
11	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
12	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
13	Message Slot 3, Scale Status. Refer to Message Slot Table	Message Slot 3, Scale Status. Refer to Message Slot Table
14	Message Slot 4, Reserved	Message Slot 4, Command Response. Refer to Message Slot Table
15	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
16	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data

PLC Output Words	Bits 0 – 7	Bits 8 - 15
17	Message Slot 4, Scale Status. Refer to Message Slot Table	Message Slot 4, Scale Status. Refer to Message Slot Table
18	Message Slot 5, Reserved	Message Slot 5, Command Response. Refer to Message Slot Table
19	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
20	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
21	Message Slot 5, Scale Status. Refer to Message Slot Table	Message Slot 5, Scale Status. Refer to Message Slot Table
22	Message Slot 6, Reserved	Message Slot 6, Command Response. Refer to Message Slot Table
23	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
24	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
25	Message Slot 6, Scale Status. Refer to Message Slot Table	Message Slot 6, Scale Status. Refer to Message Slot Table
26	Message Slot 7, Reserved	Message Slot 7, Command Response. Refer to Message Slot Table
27	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
28	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
29	Message Slot 7, Scale Status. Refer to Message Slot Table	Message Slot 7, Scale Status. Refer to Message Slot Table
30	Message Slot 8, Reserved	Message Slot 8, Command Response. Refer to Message Slot Table
31	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
32	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
33	Message Slot 8, Scale Status. Refer to Message Slot Table	Message Slot 8, Scale Status. Refer to Message Slot Table
34	Message Slot 9, Reserved	Message Slot 9, Command Response. Refer to Message Slot Table
35	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
36	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
37	Message Slot 9, Scale Status. Refer to Message Slot Table	Message Slot 9, Scale Status. Refer to Message Slot Table
38	Message Slot 10, Reserved	Message Slot 10, Command Response. Refer to Message Slot Table
39	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
40	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
41	Message Slot 10, Scale Status. Refer to Message Slot Table	Message Slot 10, Scale Status. Refer to Message Slot Table
42	Message Slot 11, Reserved	Message Slot 11, Command Response. Refer to Message Slot Table
43	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
44	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
45	Message Slot 11, Scale Status. Refer to Message Slot Table	Message Slot 11, Scale Status. Refer to Message Slot Table
46	Message Slot 12, Reserved	Message Slot 12, Command Response. Refer to Message Slot Table

PLC Output Words	Bits 0 – 7	Bits 8 - 15
47	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
48	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
49	Message Slot 12, Scale Status. Refer to Message Slot Table	Message Slot 12, Scale Status. Refer to Message Slot Table

**Table 3-5: ControlNet PLC Floating Point Output Words**

PLC Output Bytes	PLC Output Words	Usage
0	0	Reserved
1		Reserved
2	1	Message Slot 1 Command
3		Message Slot 1 Command
4	2	Message Slot 1 Floating Point data
5		Message Slot 1 Floating Point data
6	3	Message Slot 1 Floating Point data
7		Message Slot 1 Floating Point data
8	4	Message Slot 2 Command
9		Message Slot 2 Command
10	5	Message Slot 2 Floating Point data
11		Message Slot 2 Floating Point data
12	6	Message Slot 2 Floating Point data
13		Message Slot 2 Floating Point data
14	7	Message Slot 3 Command
15		Message Slot 3 Command
16	8	Message Slot 3 Floating Point data
17		Message Slot 3 Floating Point data
18	9	Message Slot 3 Floating Point data
19		Message Slot 3 Floating Point data
20	10	Message Slot 4 Command
21		Message Slot 4 Command
22	11	Message Slot 4 Floating Point data
23		Message Slot 4 Floating Point data
24	12	Message Slot 4 Floating Point data
25		Message Slot 4 Floating Point data
26	13	Message Slot 5 Command
27		Message Slot 5 Command
28	14	Message Slot 5 Floating Point data
29		Message Slot 5 Floating Point data
30	15	Message Slot 5 Floating Point data
31		Message Slot 5 Floating Point data
32	16	Message Slot 6 Command
33		Message Slot 6 Command

PLC Output Bytes	PLC Output Words	Usage
34	17	Message Slot 6 Floating Point data
35		Message Slot 6 Floating Point data
36	18	Message Slot 6 Floating Point data
37		Message Slot 6 Floating Point data
38	19	Message Slot 7 Command
39		Message Slot 7 Command
40	20	Message Slot 7 Floating Point data
41		Message Slot 7 Floating Point data
42	21	Message Slot 7 Floating Point data
43		Message Slot 7 Floating Point data
44	22	Message Slot 8 Command
45		Message Slot 8 Command
46	23	Message Slot 8 Floating Point data
47		Message Slot 8 Floating Point data
48	24	Message Slot 8 Floating Point data
49		Message Slot 8 Floating Point data
50	25	Message Slot 9 Command
51		Message Slot 9 Command
52	26	Message Slot 9 Floating Point data
53		Message Slot 9 Floating Point data
54	27	Message Slot 9 Floating Point data
55		Message Slot 9 Floating Point data
56	28	Message Slot 10 Command
57		Message Slot 10 Command
58	29	Message Slot 10 Floating Point data
59		Message Slot 10 Floating Point data
60	30	Message Slot 10 Floating Point data
61		Message Slot 10 Floating Point data
62	31	Message Slot 11 Command
63		Message Slot 11 Command
64	32	Message Slot 11 Floating Point data
65		Message Slot 11 Floating Point data
66	33	Message Slot 11 Floating Point data
67		Message Slot 11 Floating Point data
68	34	Message Slot 12 Command
69		Message Slot 12 Command
70	35	Message Slot 12 Floating Point data
71		Message Slot 12 Floating Point data
72	36	Message Slot 12 Floating Point data
73		Message Slot 12 Floating Point data

### 3.3.6.

In the template data format a message slot is a shared data variable. The number of message slots is limited to 496 bytes (I/O), including the 4 bytes of reserve in the IND780 output (PLC input) data.

### Integer and Division

When one of these formats is selected, the IND780 will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC's input data will contain one 16-bit word for the scale's weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC's output data. The PLC's output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 3-6 and Table 3-7 provide detailed information on the integer and division data formats. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 3-6: Discrete Read Integer or Division – IND780 >> PLC (ControlNet), for each Message Slot**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN <sup>1</sup>	WORD 2 IN <sup>2</sup>	WORD 3 IN	Bit number
0	Reserved	Reserved	X	Feed <sup>3</sup>	0
1			X	Fast Feed <sup>3</sup>	1
2			X	Tolerance OK <sup>4</sup>	2
3			X	Under low tolerance <sup>5</sup>	3
4			X	Over high tolerance <sup>5</sup>	4
5			X	Comparator 3 <sup>6</sup>	5
6			X	Comparator 2 <sup>6</sup>	6
7			X	Comparator 1 <sup>6</sup>	7
10			X	ENTER key <sup>7</sup>	8
11			X	Input 1 <sup>8</sup>	9
12			X	Input 2 <sup>9</sup>	10
13			X	Input 3 <sup>9</sup>	11
14			X	Motion <sup>9</sup>	12
15			X	Net mode <sup>10</sup>	13
16			X	Update in progress <sup>11</sup>	14
17			X <sup>13</sup>	Data OK <sup>12</sup>	15

**Notes for Table 3-6:**

1. WORD 0 and WORD 1 are reserved 16 bit words only present in message slot 1.
2. WORD 2 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
3. **Bit 0** and **Bit 1** are used only in material transfer mode.
4. **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
5. When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
6. **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
7. **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.

- 8 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 9 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 10 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 11 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 12 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) and when in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode Word 2 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 13 When number is negative, Word 2 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 3-7: Discrete Write Integer or Division – PLC (ControlNet) >> IND780, per Message Slot**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

**Notes for Table 3-7:**

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- 2 A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 or 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- 3 A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- 4 A transition from "0" to "1" initiates a CLEAR command.

- 5 A transition from "0" to "1" initiates a TARE command.
- 6 A transition from "0" to "1" initiates a PRINT command.
- 7 A transition from "0" to "1" initiates a ZERO command.
- 8 If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- 9 **Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.
- 10 **Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- 11 When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

### 3.3.7. Floating Point

#### 3.3.7.1. Operational Overview

The IND780 uses integer commands from the PLC to select the floating point weight input data. The IND780 recognizes a command when it sees a new value in the Message Slot command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the IND780 recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. The IND780 also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC should wait until it receives the command acknowledgment from the IND780 before sending another command.

The IND780 can report two types of values to the PLC: real-time and static. When the PLC requests a real-time value, the IND780 acknowledges the command from the PLC once but sends and updates the value at every Interface update cycle. If the PLC requests a static value, the IND780 acknowledges the command from the PLC once and updates the value once. The IND780 will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, feed, and tolerance values are examples of static data.

The IND780 can send a rotation of up to nine different real-time values. The PLC sends commands to the IND780 to add a value to the rotation. Once the rotation is established, the PLC must instruct the IND780 to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the IND780 to advance to the next value. If the IND780 is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next Interface update cycle. (The Interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command, the IND780 switches to the next value in the rotation order. The IND780 stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point command examples in Table 3-12 through Table 3-15 for additional information. The method of handling string and floating point data varies among Allen-Bradley PLC generations. The IND780 provides floating point data in the order entered in Data Format setup.

Table 3-8 to Table 3-11 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 3-8: Discrete Read Floating Point – IND780 >> PLC Input, per Message Slot**

Octal Address	WORD 0 <sup>0</sup>	WORD 1 <sup>0</sup>	WORD 2 Command Response	WORD 3 <sup>1</sup> FP value	WORD 4 <sup>1</sup> FP value	WORD 5 Status	Bit number		
0	Reserved	Reserved	Reserved	X	X	Feed <sup>2</sup>	<b>0</b>		
1				X	X	Fast Feed <sup>2</sup>	<b>1</b>		
2				X	X	Tolerance OK <sup>3</sup>	<b>2</b>		
3				X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>		
4				X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>		
5				X	X	Selected scale <sup>5</sup>	<b>5</b>		
6				X	X	Custom Bit <sup>6</sup>	<b>6</b>		
7				X	X	Custom Bit <sup>6</sup>	<b>7</b>		
10					FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11					FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12					FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13					FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14					FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15					Data integrity1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16					Cmnd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>
17					Cmnd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

**Notes for Table 3-8:**

- 0 WORD 0 and WORD 1 are reserved 16 bit words, only present in message slot 1.
- 1 The bits in WORD 3 and WORD 4 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- 4 **Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- 5 **Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- 6 **Bit 6** and **Bit 7** are user defined, and correspond to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.

- 7 The Floating Point Input Indication bits (WORD 2, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 3 and WORD 4). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- 8 **Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- 9 **Bit 9, Bit 10,** and **Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 10 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 11 The Data Integrity bit in WORD 2 - **bit 13** is used in conjunction with the bit in WORD 5 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- 12 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) and when in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 3-9: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency
12	Notch filter frequency
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size

Dec	Data
19–28	Not used
29	Last IND780 error code
30	No data response – command successful
31	No data response – command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* SDV means Shared Data Variable.

**Table 3-10: Discrete Write Floating Point – ControlNet PLC >> IND780, per Message Slot**

Octal Address	WORD 0 <sup>0</sup>	WORD 1	WORD 2 <sup>1</sup> FP load value	WORD 3 <sup>1</sup> FP load value	Bit number
0	Reserved	Command Word <sup>2</sup>	X	X	<b>0</b>
1			X	X	<b>1</b>
2			X	X	<b>2</b>
3			X	X	<b>3</b>
4			X	X	<b>4</b>
5			X	X	<b>5</b>
6			X	X	<b>6</b>
7			X	X	<b>7</b>
10			X	X	<b>8</b>
11			X	X	<b>9</b>
12			X	X	<b>10</b>
13			X	X	<b>11</b>
14			X	X	<b>12</b>
15			X	X	<b>13</b>
16			X	X	<b>14</b>
17			X	X	<b>15</b>

**Notes for Table 3-10:**

- 0 WORD 0 is a reserved 16 bit word, only present in message slot 1.
  - 1 The Bits in WORD 2 and WORD 3 are a single-precision floating point value. This value is used with the command in WORD 1 to instruct the terminal to download the floating point value into the field specified in the command.
  - 2 The command Word 1 is used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 3-11, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.
- Not all commands will require a value in the floating point load value words.

**Table 3-11: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName	Dec	Hex	Command	SDName
0	00	Report next rotation field @ next Interface update cycle <sup>1</sup>		76	4c	Start ID2 sequence <sup>7,9</sup>	PDO119

Dec	Hex	Command	SDName
1	01	Report next rotation field <sup>1,2</sup>	
2	02	Report next rotation field <sup>1,2</sup>	
3	03	Reset rotation	
10	0a	Report gross weight <sup>1,3</sup>	
11	0b	Report net weight <sup>1,3</sup>	
12	0c	Report tare weight <sup>1,3</sup>	
13	0d	Report fine gross weight <sup>1,3</sup>	
14	0e	Report fine net weight <sup>1,3</sup>	
15	0f	Report fine tare weight <sup>1,3</sup>	
16	10	Report Rate <sup>1,3</sup>	
17	11	Report SDV ALxx01 <sup>6</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>	
20	14	Report notch filter frequency <sup>3</sup>	
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12
25	19	Report spill value <sup>3,10</sup>	SPxx09
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>
29	1d	Report last error <sup>3</sup>	
30	1e	Report primary increment size <sup>3</sup>	
40	28	Add gross weight to rotation <sup>7</sup>	
41	29	Add net weight to rotation <sup>7</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>	
46	2e	Add rate to rotation <sup>7</sup>	
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>
60	3c	Load numeric tare value <sup>4</sup>	
61	3d	Pushbutton tare command <sup>7</sup>	
62	3e	Clear command <sup>7</sup>	

Dec	Hex	Command	SDName
78	4e	Disable LCD display <sup>7</sup>	
79	4f	Enable LCD display <sup>7</sup>	
80	50	Set normal display mode <sup>7,9</sup>	PD0119
81	51	Display message 1 <sup>7,9</sup>	PD0119
82	52	Display message 2 <sup>7,9</sup>	PD0119
83	53	Display message 3 <sup>7,9</sup>	PD0119
84	54	Display message 4 <sup>7,9</sup>	PD0119
85	55	Display Message 5 <sup>7,9</sup>	PD0119
86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
88	58	Disable weight display <sup>7</sup>	
89	59	Enable weight display <sup>7</sup>	
90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
104	68	Master control relay OFF <sup>7, 12</sup>	
105	69	Master control relay ON <sup>7, 12</sup>	
110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
111	6f	Set target fine feed value <sup>4,10</sup>	
112	70	Set - tolerance value <sup>4,10</sup>	
113	71	Set target value and apply <sup>4, 10, 14</sup>	
114	72	Start target logic <sup>7,10, 15</sup>	
115	73	Abort target logic <sup>7,10, 15</sup>	
116	74	Target use gross weight <sup>7,10</sup>	SPxx04
117	75	Target use net weight <sup>7,10</sup>	SPxx04
118	76	Target use rate <sup>7,10</sup>	SPxx04
119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
121	79	Enable target latching <sup>7,10</sup>	SPxx06
122	7a	Disable target latching <sup>7,10</sup>	SPxx06

Dec	Hex	Command	SDName
63	3f	Print command <sup>7</sup>	
64	40	Zero command <sup>7</sup>	
65	41	Select scale 1 <sup>7</sup>	
66	42	Select scale 2 <sup>7</sup>	
67	43	Select next scale <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>
73	49	Set low-pass filter corner frequency <sup>4</sup>	
74	4a	Set notch filter frequency <sup>4</sup>	
75	4b	Reset ENTER key <sup>7</sup>	

Dec	Hex	Command	SDName
123	7b	Reset target latch <sup>7</sup>	SPxx07
124	7c	Set target spill value <sup>4, 10</sup>	
131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
160	a0	Apply scale setup <sup>7</sup>	QC0149
161	a1	Write Calibration to EEPROM <sup>7</sup>	
162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
163	a3	Enable Tare on IND780 console <sup>7</sup>	
164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
168	a8	Select scale 3 <sup>7</sup>	
169	a9	Select scale 4 <sup>7</sup>	

**Notes for Table 3-11:**

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

### 3.3.7.2. Floating Point Data Format and Compatibility

In Floating Point Data Format, the PLC and IND780 terminal exchange weight, target, and tare data in single-precision floating-point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

There are two data integrity bits that the IND780 uses to maintain data integrity when communicating with the PLC. One bit is in the beginning word of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC; if the PLC program detects this condition, it should send a new command to the terminal.

The Allen-Bradley SLC PLC programs do not have to make special checks to guarantee the validity of the floating-point data.

The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the word order set up by the user.

3.3.7.3.

Floating Point Command Examples

Table 3-12 through Table 3-15 provide floating point command examples.

**Table 3-12: Data Requirement: Only Net Weight Sent (continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word 0	none required		
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word, the IND780 terminal will update the net value every interface update cycle.				

**Table 3-13: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2 (PLC sends command to set target 1 cutoff value)	110 (dec) loaded into command word 0	floating point value = 21.75		
3 (IND780 terminal sees new command, loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word 0			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
The PLC should always wait to receive a command acknowledgment before sending the next command to the				

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.				

**Table 3-14: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	0.0
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0
At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.				
7 (PLC sends the command to begin the rotation at interface update cycle)	0 (dec) loaded into command word 0			
8 (IND780 terminal sends gross weight at interface update cycle ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value at next interface update)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
cycle)				
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 3-15: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack.= 1 F.P. ind = 30	0.0
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0	RESERVED for Future Use		
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0
At this point, the rotation has been set up. Now the PLC needs to send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word 0			
8 (IND780 terminal acknowledges the command and sends net weight at every Interface update cycle until the PLC gives the command to report the next rotation)			Command ack. = 1 F.P. ind = 1	Floating point value = net weight

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
field.)				
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word 0			
10 (IND780 terminal acknowledges the command and sends rate at every Interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
11 (PLC sends the command to report the next field in the rotation.)	1 (dec) loaded into command word 0			
12 (IND780 terminal acknowledges the command and sends net weight at every Interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
13 (PLC sends the command to report the next field.)	2 (dec) loaded into command word 0			
14 (IND780 terminal acknowledges the command and sends rate at every Interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value				

#### 3.3.7.4. Shared Data Mode

The Shared Data mode PLC communications is provided using CIP explicit messages.

The **IND780 Shared Data Reference** lists the Shared Data Variables available to ControlNet. This document also includes the hex Class Code, Instance and Attribute for the shared data. The PLC must use Get Attribute Single (0e hex) to read a Shared Data Variable and Set Attribute Single (10 hex) to write a Shared Data Variable.

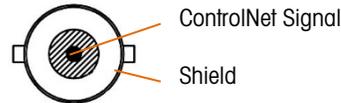
### 3.3.8. Controlling the Discrete I/O Using a PLC Interface

The IND780 terminal provides the ability to directly control its discrete outputs and read its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the IND780 terminal's discrete I/O updates are synchronized with the terminal's interface update rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the IND780 Terminal Technical Manual for discrete I/O wiring.

## 3.4. Hardware Setup

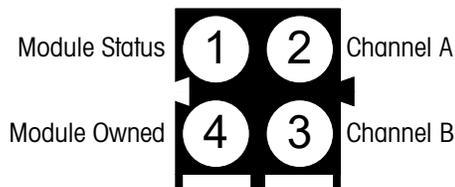
### 3.4.1. Wiring

The IND780 terminal's ControlNet option card uses one BNC coaxial connector, channel A or two BNC coaxial connectors for redundant ControlNet configurations, one for each channel (Figure 3-6). Cable distance, type, and termination are specified by Allen-Bradley (See Allen-Bradley documentation for reference on cable design guidelines for the various PLCs).



**Figure 3-6: Coaxial Connector for ControlNet**

Figure 3-7 shows the array of status indicator LEDs on the ControlNet card (see also Table 3-1). Table 3-16 explains the meaning of the indicators.



**Figure 3-7: ControlNet Status Indicator LEDs**

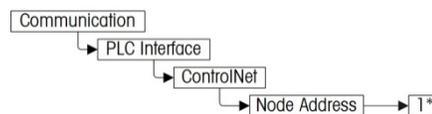
**Table 3-16: ControlNet Status Indications**

LED #	Indication	LED State	Description
1	Module Status	Green	Connection in Run state
		Green, flashing	Connecting Connection Idle
		Red	Major fault
		Red, flashing	Minor fault
2	Channel A	Off	Module not initialized

LED #	Indication	LED State	Description
2 or 3	Channel A or Channel B	Off	Channel disabled
		Green	Normal operation of channel
		Green, flashing	Temporary error (node will self-correct) or not configured
4	Module Owned	Red, flashing	No other nodes, or media fault
		Red & green, flashing	Network configuration error
and 3	and Channel B	Red	Major fault
		Alternating red/green	Self test
4	Module Owned	Off	No connection has been opened
		Green	A connection has been opened towards the module

## 3.5. Software Setup

The IND780 terminal automatically detects the presence of a ControlNet Kit option board if one is installed. When the IND780 detects that a ControlNet board is present, it enables the ControlNet parameters in setup at **Communication > PLC Interface > ControlNet**. Figure 3-8 shows the ControlNet program block and Figure 3-9 the PLC data format setup block at **Communication > PLC Interface > Data Format**, with default settings indicated by an asterisk.



**Figure 3-8: The ControlNet Program Block**

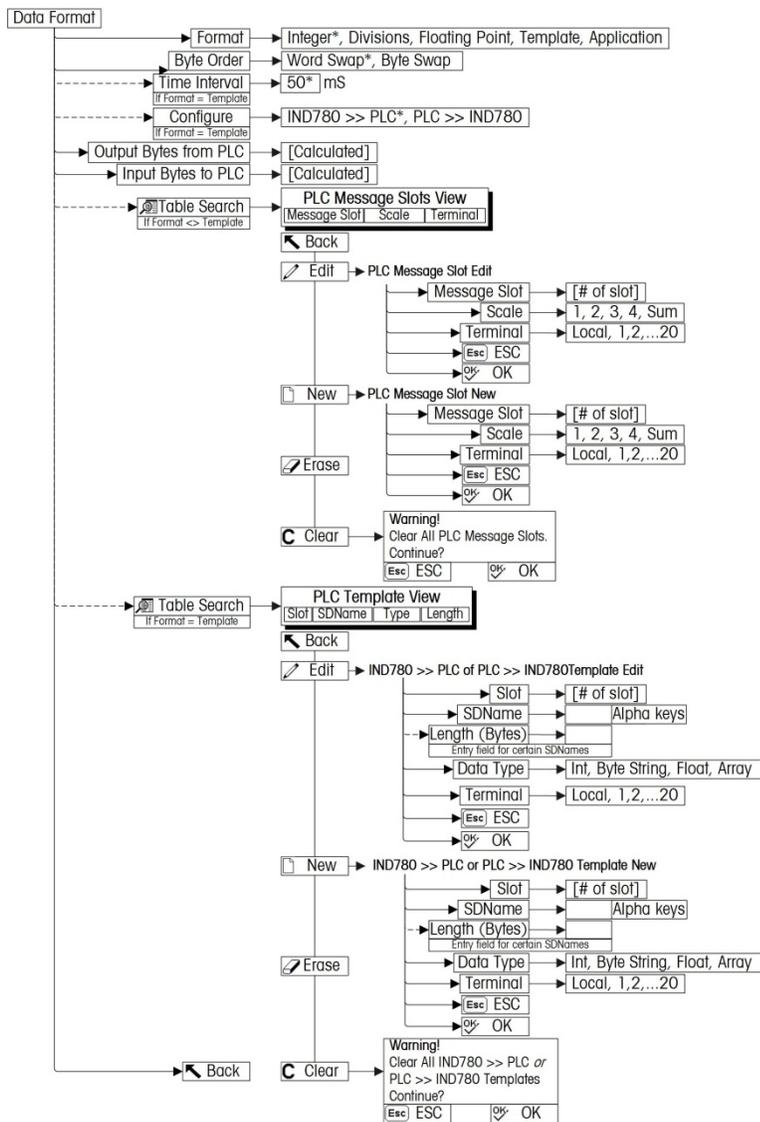


Figure 3-9: PLC Data Format Setup Block

### 3.5.1. ControlNet and Data Format Setup Blocks

#### 3.5.1.1. ControlNet Setup

The Scale Setup block lets you specify node address of the ControlNet interface. To configure the block, access **Communication > PLC Interface > ControlNet** (Figure 3-10) and set the Node Address.

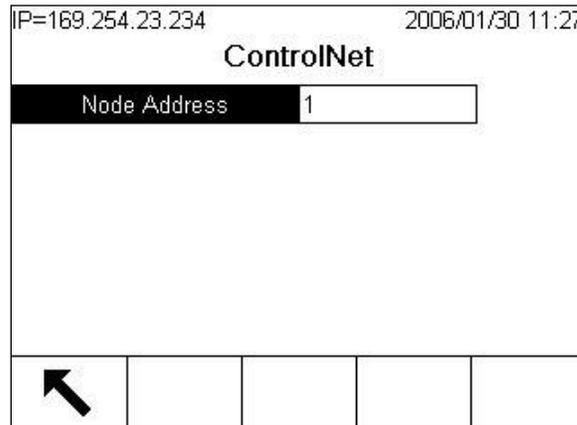


Figure 3-10: ControlNet Setup Screen

#### 3.5.1.2. Data Format Setup

1. Select **Communication > PLC Interface > Data Format** (Figure 3-9).

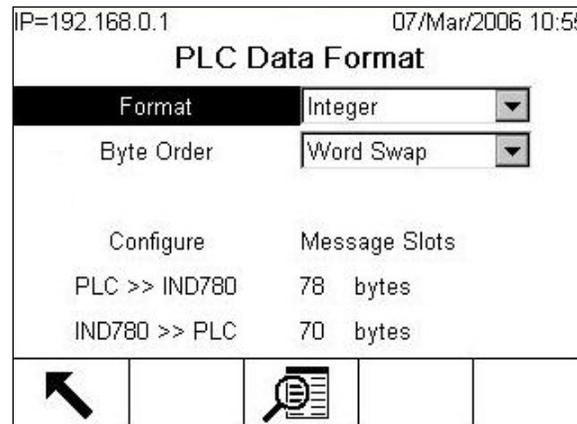
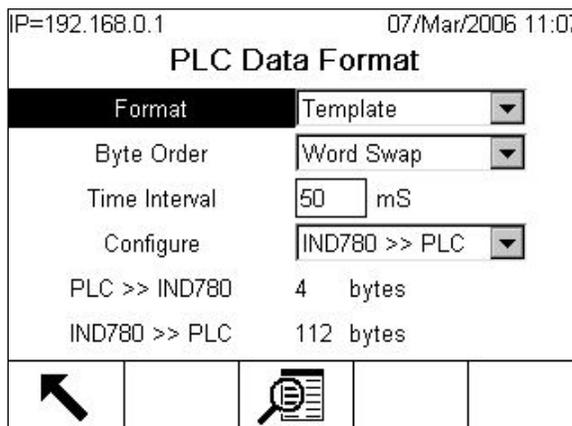


Figure 3-11: PLC Data Format Screen, Integer Selected

- Figure 3-10 shows the screen with **Template** selected as the format, revealing the **Time Interval**, **Configure**, and directional displays. When **Application** format is selected, only the **Format** box appears on this screen.



**Figure 3-12: PLC Data Format Screen, Template Selected**

- Select the **Format** (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the **Format** will delete any existing **Message Slots**.

<b>Integer</b>	reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).
<b>Divisions</b>	reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.
<b>Floating Point</b>	displays weight in floating point data format
<b>Template</b>	Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed "on the fly" by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the Shared Data Reference Manual (document number 64059110) provided on the CD-ROM supplied with your terminal.
<b>Application</b>	Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

- Select **Byte Order**: [default Word Swap]
 

<b>Word Swap</b>	Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.
<b>Byte Swap</b>	Makes the floating point format compatible with S7 PROFIBUS.
- Set the **Time Interval** in milliseconds. (Only when the **Format** is **Template**)
- Use the **Configure** drop-down list to select the direction of data flow – **IND780 >> PLC** or **PLC >> IND780**. (Only when the **Format** is **Template**)
- The bottom of this screen (Figure 3-12) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.

8. Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of Message Slots (maximum number is 12, default is 0), or to configure the Template for communications between the IND780 and the PLC. Figure 3-13 shows the Message Slots View, which is viewed whenever the Format is not Template.

IP=172.18.54.111 06/Dec/2007 16:25

**PLC Message Slots View**

Message	Scale	Terminal
1	1	Local
2	- End -	







**Figure 3-13: Message Slots View Screen**

Figure 3-14 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1 07/Mar/2006 11:08

**PLC >> IND780 Template View**

Slot	SDName	Type	Length
1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		







**Figure 3-14: Template View**

Press the EXIT softkey  to exit the view screen.

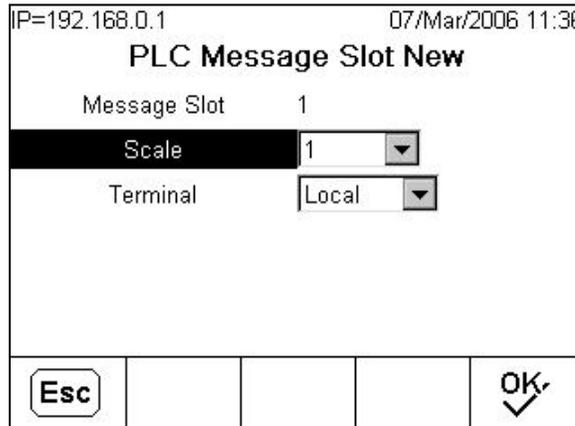
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey  to delete all existing slots.

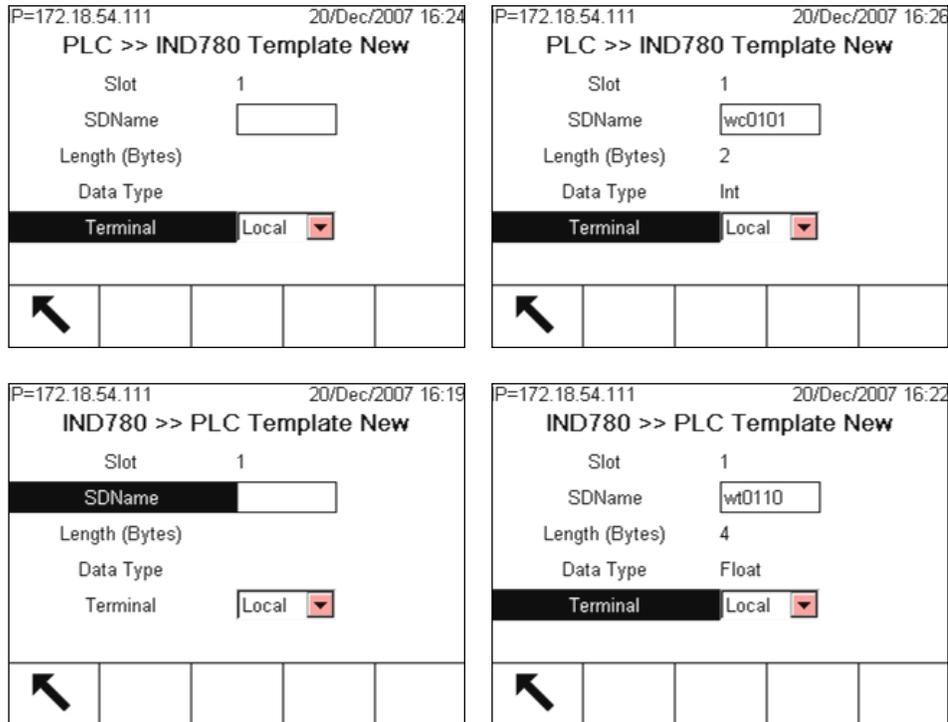
Figure 3-15 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 3-13).



**Figure 3-15: New Message Slot Screen**

- Scale** Enter the scale number to be associated with the new Message Slot.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 3-16 shows a series of Template New screens, as accessed from the Template View screen (Figure 3-14).



**Figure 3-16: New Template Screens, PLC to IND780 and IND780 to PLC**

<b>Slot</b>	Automatically assigned.
<b>SDName</b>	An alphabetic entry field used to define the Shared Data variable associated with the slot.
<b>Length (Bytes) and Data Type</b>	Automatically displayed once the SDName has been entered.
<b>Terminal</b>	Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

For a complete listing of Shared Data Fields, refer to the IND780 Shared Data Reference. Refer to Table 3-6 (Discrete Read) and Table 3-7 (Discrete Write) in this manual for additional information on mapping of discrete read data to the PLC.

## 3.6. Troubleshooting

If the IND780 does not communicate with PLC, do the following:

- Check wiring and network termination.
- Confirm that the IND780 settings for data type and node assignment match those in the PLC and that each IND780 has a unique node assignment.
- Replace the ControlNet interface kit if the problem persists.
- If the communication kit was changed from another type, like PROFIBUS or Remote I/O or EtherNet I/P, a master reset of the IND780 must be performed.

## 3.7. ControlNet Option Kit

No spare parts are associated with the ControlNet option kit. The kit CIMF part number is 64057423. Table 3-17 shows what comes in the kit.

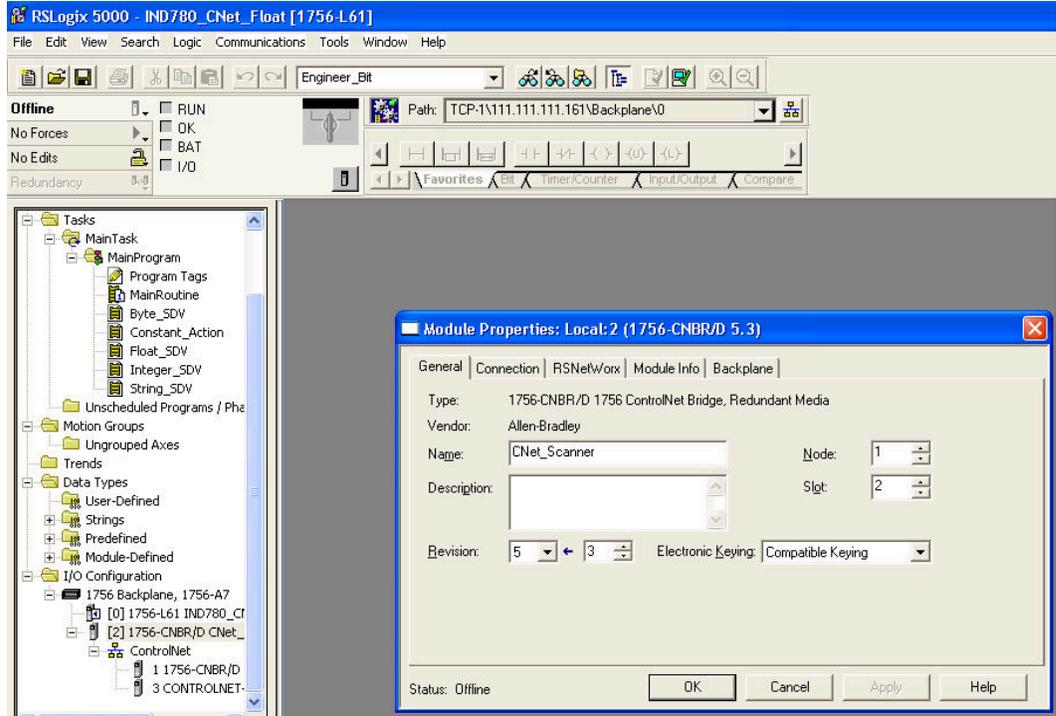
**Table 3-17: ControlNet Option Kit**

Description	Qty.
Installation Guide	1
PCB package	1
Clamp-on ferrite	1
M3 x 8 screws	3

## 3.8. Programming Examples

The following Figures show screen images of ladder logic programming examples for RSLogix 5000 software (version 13).

- **Note:** The Utilities folder of the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.



**Figure 3-17: 1756 CNB Setup**

The following RSLogix 5000 screens for Integer, Division and Floating Point data formats only show an example of a particular Input and Output size configuration. The Connection Parameters I/O sizes must be appropriately configured with reference to the number of slots assigned in the IND780 PLC Data Format Message Slots settings. Table 3-18 and Table 3-19 show the relationship between the IND780 message slots and the RSLogix 5000 I/O sizing for Integer, Division and Floating Point data formats.

**Table 3-18: Message Slot and PLC I/O Sizes (Integer/ Division)**

Message Slots	IND780 Integer/ Division Data		RSLogix 5000 Comm Format	
	Bytes (8 Bit)		INT (16 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780	Input	Output
1	8	4	4	2
2	12	8	6	4
3	16	12	8	6

IND780 Integer/ Division Data			RSLogix 5000 Comm Format	
Message Slots	Bytes (8 Bit)		INT (16 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780	Input	Output
4	20	16	10	8
5	24	20	12	10
6	28	24	14	12
7	32	28	16	14
8	36	32	18	16
9	40	36	20	18
10	44	40	22	20
11	48	44	24	22
12	52	48	26	24

Table 3-19: Message Slot and PLC I/O Sizes (Floating Point)

IND780 Floating Point Data			RSLogix 5000 Comm Format	
Message Slots	Bytes (8 Bit)		INT (16 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780	Input	Output
1	12	8	6	4
2	20	14	10	7
3	28	20	14	10
4	36	26	18	13
5	44	32	22	16
6	52	38	26	19
7	60	44	30	22
8	68	50	34	25
9	76	56	38	28
10	84	62	42	31
11	92	68	46	34
12	100	74	50	37

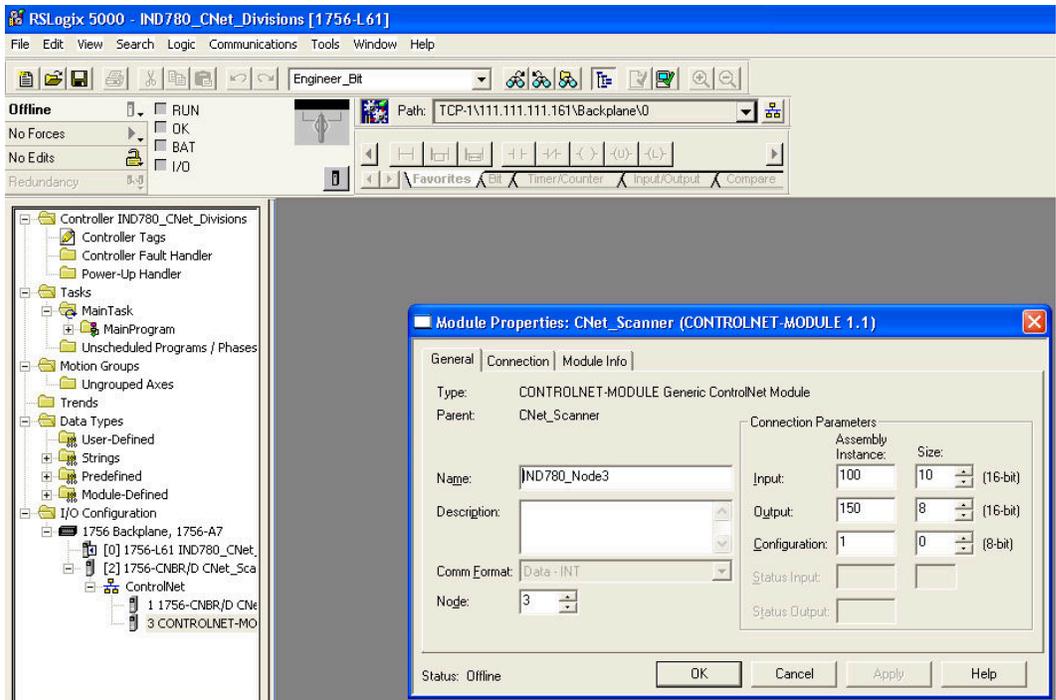


Figure 3-18: DIV ControlNet 4-Slot Module Setup

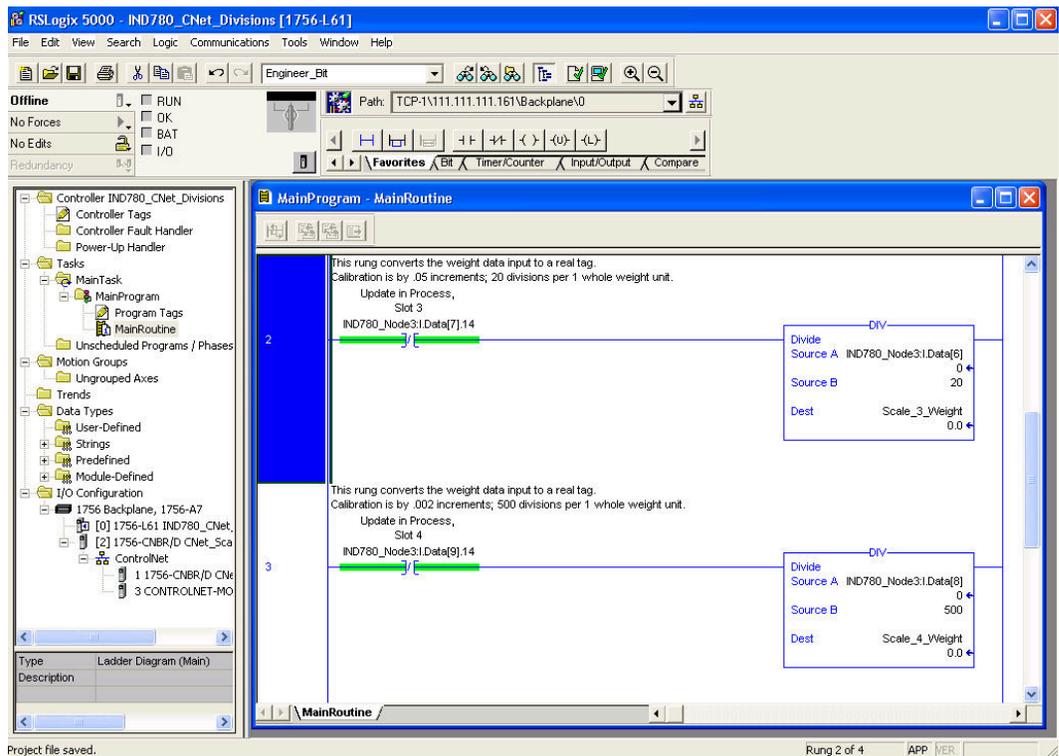
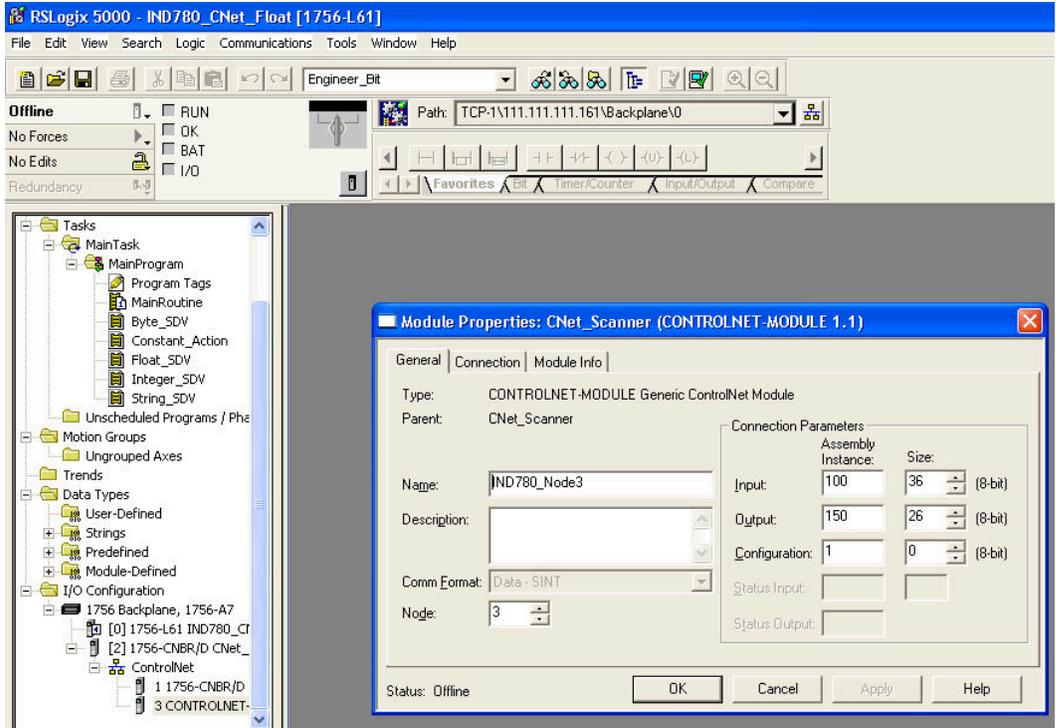
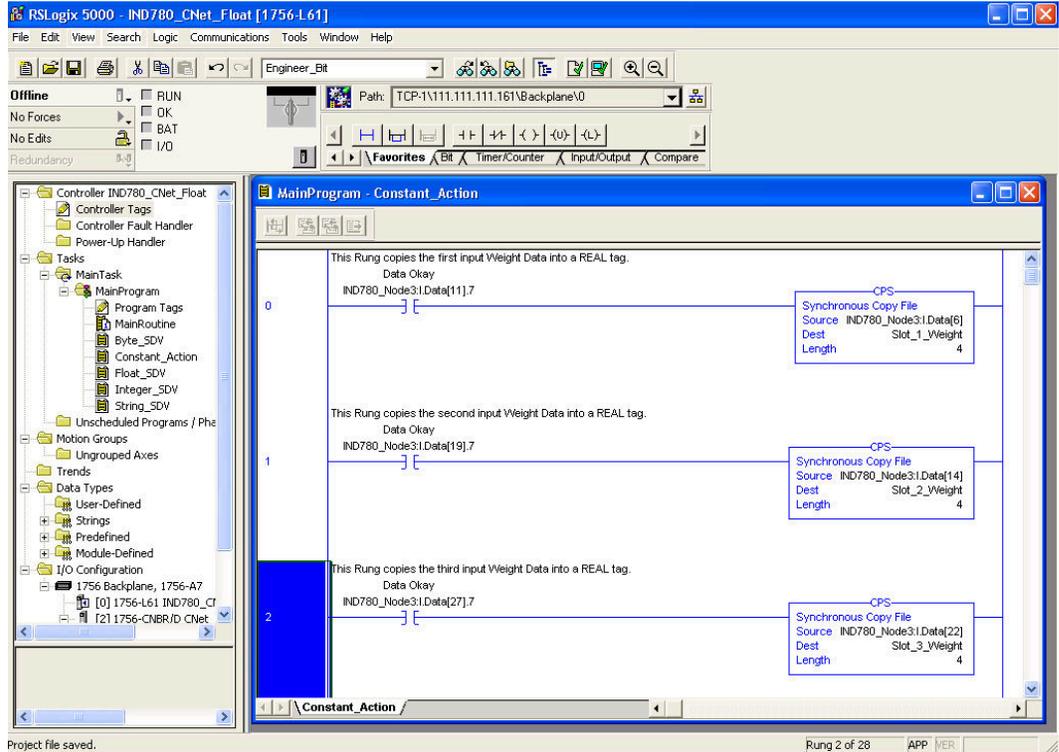


Figure 3-19: DIV ControlNet 4-Slot Weight 2 Tab Setup



**Figure 3-20: Float ControlNet 4-Slot Module Setup**



**Figure 3-21: Float ControlNet 4-Slot Weight to Tag**

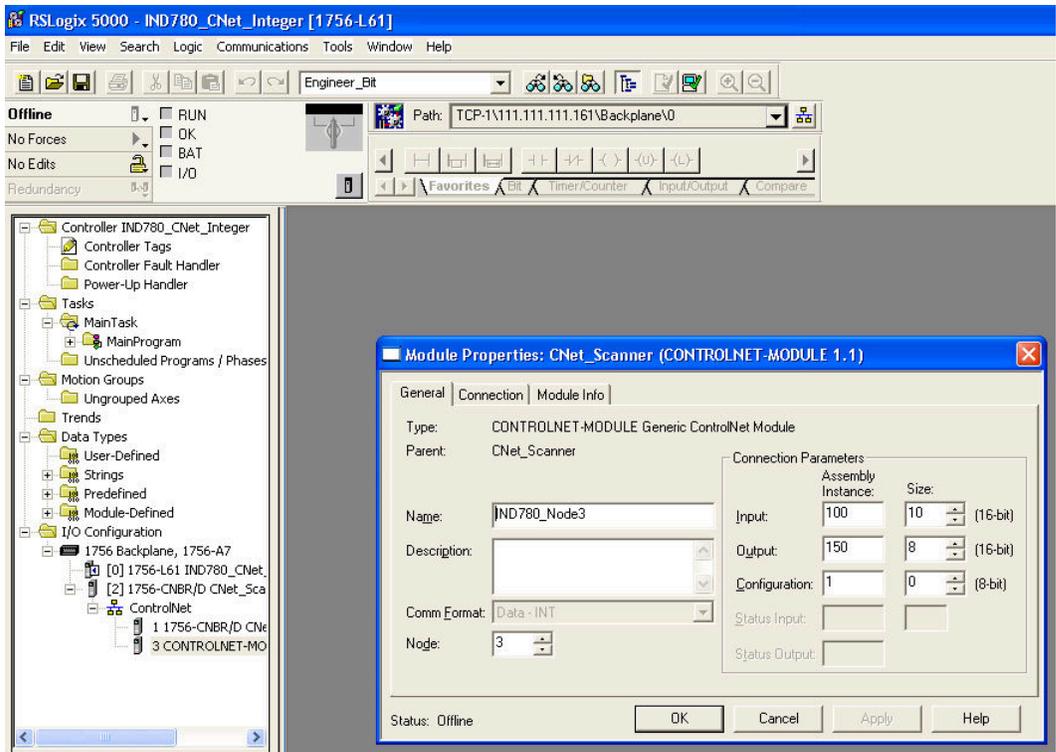


Figure 3-22: INT ControlNet 4-Slot Module Setup

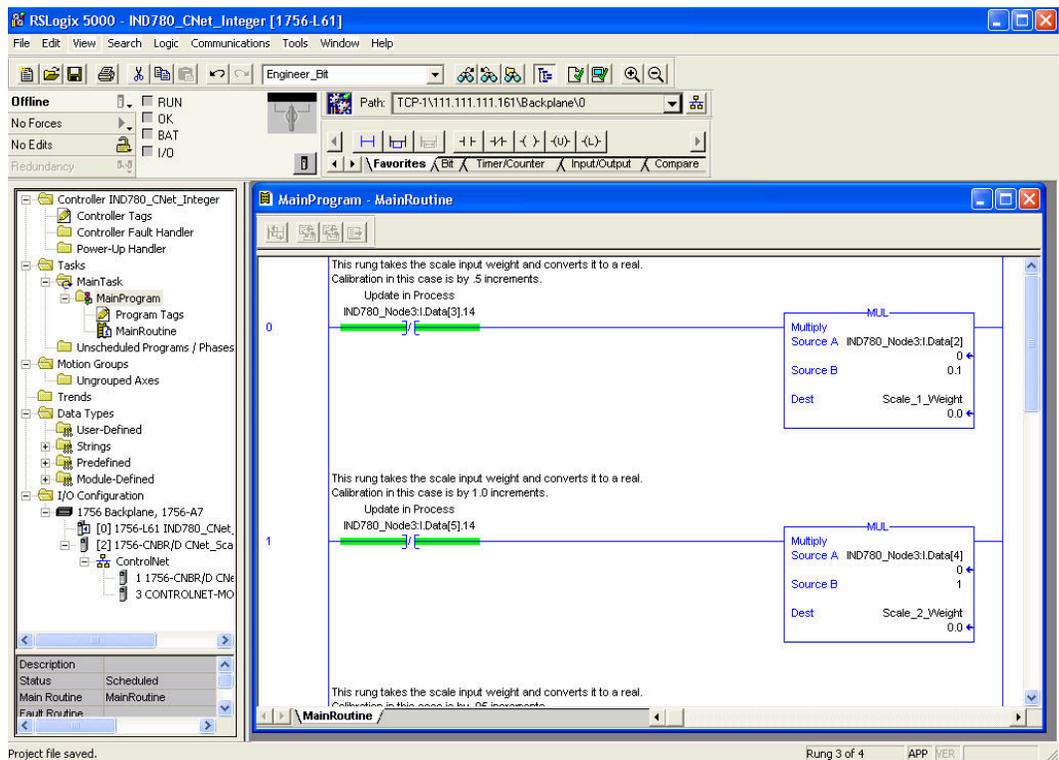


Figure 3-23: INT ControlNet 4-Slot Weight to Tag

# 4 DeviceNet

## 4.1. Preface

Users should note that the DeviceNet option is used in both the METTLER TOLEDO IND560 and IND780 terminals. There are minor differences in the Floating Point polled data between the two terminals. The terminals share the same EDS file and Icon file for use in a DeviceNet network configuration tool.

## 4.2. Overview

DeviceNet is an RS-485-based network utilizing CAN chip technology. This network was created for bit and byte-level devices. The network can be configured to run up to 500Kbits per second depending on cabling and distances. Messages are limited to 8 un-fragmented bytes. Any larger message must be broken up and sent in multiples. The IND780 implementation of DeviceNet does not support fragmented messages – all messages are 8 bytes or shorter. The network is capable of 64 nodes including the master, commonly called the scanner.

## 4.3. DeviceNet Characteristics

- DeviceNet-specific (twisted pair) cable
- Access to intelligence present in low-level devices – Master/Slave and Peer-to-Peer capabilities
- Trunkline-dropline configuration
- Support for up to 64 nodes
- Node removal without severing the network
- Simultaneous support for both network-powered (sensors) and self-powered (actuators) devices
- Use of sealed or open style connectors
- Protection from wiring errors
- Selectable baud rates of 125k baud, 250k baud, and 500k baud. Maximum trunk distance 500 meters and drop length of 156 meters at 125k baud
- Adjustable power configuration to meet individual application needs
- High current capability (up to 8 amps per supply)
- Operation with off-the-shelf power supplies

- Power taps that allow the connection of several power supplies from multiple vendors that comply with DeviceNet standards
- Built-in overload protection
- Power available along the bus: both signal and power lines contained in the trunkline
- Provisions for the typical request/response-oriented network communications
- Provisions for the efficient movement of I/O data
- Fragmentation (anything in excess of 8 bytes) for moving larger bodies of information Note: Mettler Toledo terminals do not support fragmented messaging.
- Duplicate MAC ID detection

Consult <http://www.odva.org/> to obtain additional information on DeviceNet.

Figure 4-1 shows a view of the DeviceNet Option Board, with its connector at lower left. Figure 4-2 indicates the board's connector and status lights.



**Figure 4-1: DeviceNet Option Board**



**Figure 4-2: DeviceNet Option Board Components**

### 4.3.1. Communications

The IND780 utilizes polled messages. This type of message may be referred to as scheduled or cyclic messages. It does not support explicit or unscheduled messaging.

### 4.3.2. Node Address

The IND780 can be assigned any valid DeviceNet node address. Typically 0 is reserved for scanner cards and address 63 is reserved for new devices “out of the box”. The IND780’s default MAC ID is 63.

### 4.3.3. Data Formats

While being polled, the IND780 supports floating point, integer, and divisions data formats. The Application data format is also available to allow a custom TaskExpert program to control the PLC input and output data. Shared Data cannot be accessed directly using DeviceNet.

### 4.3.4. Network Topology

The following example (Figure 4-3) shows a typical DeviceNet network topology with IND780 terminals connected directly to a Logix controller.

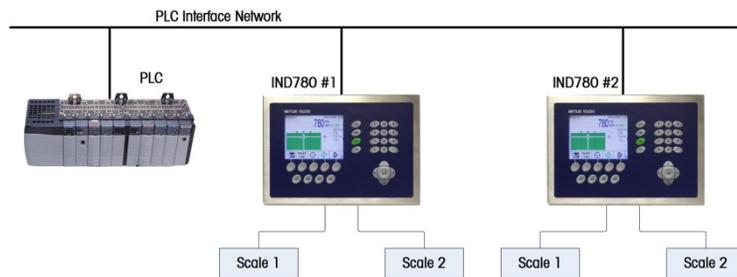


Figure 4-3: Typical Network Topology

### 4.3.5. Network Power and Current

Table 4-1 and Table 4-2 detail power and current characteristics of the DeviceNet network.

Table 4-1: Network Power Consumption

Voltage	Current
11 V	50 mA
25 V	30 mA

Table 4-2: Network Inrush Current

Voltage	Current	Duration
24 V	0.7 A	6 ms

## 4.4. Data Definition

### 4.4.1. Data Integrity

The IND780 Terminals have specific bits to allow the PLC to confirm that data was received without interruption and the IND780 Terminal is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received for the IND780

Terminal. Refer to the data charts for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage.

#### 4.4.2. Discrete Data

Four formats of discrete data are available with the DeviceNet Kit option: integer, division, floating point and application. Only one type of data format may be selected and used by IND780 Terminals sharing the same DeviceNet logical node address.

The integer and division formats allow bi-directional communication of discrete bit-encoded information or 16 bit binary word numerical values. The IND780 Terminal provides four bytes per message slot. Two slots are available in integer and division mode while only one eight byte slot is available via floating point mode.

The floating-point format allows bi-directional communication of discrete bit encoded information or numeric data encoded in IEEE 754, single precision floating point format. The floating-point format requires more space per IND780 Terminal because floating point data uses two 16-bit words of data to represent just the numeric data alone. Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions or increments. The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Changing the Data Format to be used by the IND780 will clear all Message Slots. Data format is configured in setup at **Communication > PLC Interface > Data Format**.

##### 4.4.2.1.

Examples

250 x .01 scale				
IND780 Displays:	0	2.00	51.67	250.00
Format sent:				
Int	0	200	5167	25000
Div	0	200	5167	25000
FLT	0	2.00	51.67	250.00

Any of the formats could be used in this case.

50,000 x 10 scale				
IND780 Displays:	0	200	5160	50000
Format sent:				
Int	0	200	5160	-(15536)
Div	0	20	516	5000
FLT	0	200	5160	50000

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

150 x .001 scale				
IND780 Displays:	0	2.100	51.607	150.000
Format sent:				
Int	0	2100	-(13929)	18928
Div	0	2100	-(13929)	18928
FLT	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32.767.

#### 4.4.3. Byte Order

**Word Swap** takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** makes the floating point format compatible with S7 PROFIBUS.

#### 4.4.4. Message Slots

There may be up to 2 message slots for discrete data transfer, in Integer and Divisions Data Formats and maximum 1 message slot in Floating Point Data Format. Each message slot is assigned to a local scale and scales may be repeated in additional message slots. The integer and division formats provide (two 16-bit words of input and two 16-bit words of output data) per Message Slot. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. The following two Tables provide input and output usage information.

**Table 4-3: DeviceNet PLC Input Data and Data Usage (Integer and Division)**

PLC Input Bytes	PLC Input Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Status
3		Message Slot 1 Scale Status
4	2	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Status
7		Message Slot 2 Scale Status

**Table 4-4: DeviceNet PLC Output Words and Word Usage (Integer and Division)**

PLC Output Bytes	PLC Output Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Command
3		Message Slot 1 Scale Command
4	2	Message Slot 2 Weight Data

PLC Output Bytes	PLC Output Words	Usage
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Command
7		Message Slot 2 Scale Command

The floating point format provides four 16-bit words of input data and three 16-bit words of output data per Message Slot. See Table 4-5 and Table 4-6 for details.

**Table 4-5: DeviceNet PLC Floating Point Input Words**

PLC Output Words	Bits 0 – 7	Bits 8 – 15
0	Message Slot 1, Reserved	Message Slot 1, Command Response. Refer to Message Slot Table
1	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
2	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
3	Message Slot 1, Scale Status. Refer to Message Slot Table	Message Slot 1, Scale Status. Refer to Message Slot Table

**Table 4-6: DeviceNet PLC Floating Point Output Words**

PLC Output Bytes	PLC Output Words	Usage
0	0	Reserved
1		Reserved
2	1	Message Slot 1 Command
3		Message Slot 1 Command
4	2	Message Slot 1 Floating Point data
5		Message Slot 1 Floating Point data
6	3	Message Slot 1 Floating Point data
7		Message Slot 1 Floating Point data

#### 4.4.5. Integer and Division

When one of these formats is selected, the IND780 will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC’s input data will contain one 16-bit word for the scale’s weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC’s output data. The PLC’s output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 4-7 and Table 4-8 provide detailed information on the integer and division data formats. Read data refers to the PLC’s input data and write data refers to the PLC’s output data.

**Table 4-7: Discrete Read Integer or Division – IND780 >> PLC, for each Message Slot**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

**Notes for Table 4-7:**

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
- 5 **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 4-8: Discrete Write Integer or Division – PLC >> IND780, per Message Slot**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

#### Notes for Table 4-8:

- WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 and 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- A transition from "0" to "1" initiates a CLEAR command.
- A transition from "0" to "1" initiates a TARE command.
- A transition from "0" to "1" initiates a PRINT command.
- A transition from "0" to "1" initiates a ZERO command.
- If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.
- Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

## 4.4.6. Floating Point

### 4.4.6.1. Operational Overview

The IND780 terminal uses integer commands from the PLC to select the floating point weight output data. The terminal recognizes a command when it sees a new value in the scale's command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the terminal recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. It also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC waits until it receives the command acknowledgment from the terminal before sending another command.

The terminal has two types of values that it can report to the PLC: real-time and static. When the PLC requests a real-time value, the terminal acknowledges the command from the PLC once but sends and updates the value at every interface update cycle. If the PLC requests a static value, the terminal acknowledges the command from the PLC once and updates the value once. The terminal will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, fine feed, and tolerance values are examples of static data.

The terminal can send a rotation of up to nine different real-time values for each scale. The PLC sends commands to the terminal to add a value to the rotation. Once the rotation is established, the PLC must instruct the terminal to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the terminal to advance to the next value. If the terminal is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next interface update cycle. (The interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command, the terminal switches to the next value in the rotation. The terminal stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point rotation command examples in Table 4-13 and Table 4-16 for additional information. The IND780 provides floating point data in the byte order entered in Data Format setup. The byte order follows the convention of the Controllogix platform. Note that this is word swapped compared to PLC5 and SLC500 processors.

Table 4-9 through Table 4-12 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 4-9: Discrete Read Floating Point (float) – IND780 >> PLC (DeviceNet)**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity 1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmnd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>
17	Cmnd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

**Notes for Table 4-9:**

- 1 The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- 4 **Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- 5 **Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- 6 **Bit 6** and **Bit 7** are user defined, and correspond to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- 7 The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- 8 **Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- 9 **Bit 9**, **Bit 10**, and **Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 10 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 11 The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- 12 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) or in 10x mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 4-10: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency
12	Notch filter frequency
13	Target value, instance by scale

Dec	Data
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size
19–28	Not used
29	Last IND780 error code
30	No data response – command successful
31	No data response – command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* SDV means Shared Data Variable.

**Table 4-11: Discrete Write Floating Point –PLC (DeviceNet) >> IND780**

Octal Address	WORD 0 <sup>0</sup>	WORD 1	WORD 2 <sup>1</sup> FP load value	WORD 3 <sup>1</sup> FP load value	Bit number
0	Reserved	Command Word <sup>2</sup>	X	X	<b>0</b>
1			X	X	<b>1</b>
2			X	X	<b>2</b>
3			X	X	<b>3</b>
4			X	X	<b>4</b>
5			X	X	<b>5</b>
6			X	X	<b>6</b>
7			X	X	<b>7</b>
10			X	X	<b>8</b>
11			X	X	<b>9</b>
12			X	X	<b>10</b>
13			X	X	<b>11</b>
14			X	X	<b>12</b>
15			X	X	<b>13</b>
16			X	X	<b>14</b>
17			X	X	<b>15</b>

**Notes for Table 4-11:**

- 0 WORD 0 is a reserved 16 bit word, only present in message slot 1.
- 1 The Bits in WORD 2 and WORD 3 are a single-precision floating point value. This value is used with the command in WORD 1 to instruct the terminal to download the floating point value into the field specified in the command.
- 2 The command Word 1 is used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 4-12, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.

Not all commands will require a value in the floating point load value words.

Table 4-12: PLC Output Command Table (Floating Point Only)

Dec	Hex	Command	SDName	Dec	Hex	Command	SDName
0	00	Report next rotation field @ next interface update cycle <sup>1</sup>		76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
1	01	Report next rotation field <sup>1,2</sup>		78	4e	Disable LCD display <sup>7</sup>	
2	02	Report next rotation field <sup>1,2</sup>		79	4f	Enable LCD display <sup>7</sup>	
3	03	Reset rotation		80	50	Set normal display mode <sup>7,9</sup>	PD0119
10	0a	Report gross weight <sup>1,3</sup>		81	51	Display message 1 <sup>7,9</sup>	PD0119
11	0b	Report net weight <sup>1,3</sup>		82	52	Display message 2 <sup>7,9</sup>	PD0119
12	0c	Report tare weight <sup>1,3</sup>		83	53	Display message 3 <sup>7,9</sup>	PD0119
13	0d	Report fine gross weight <sup>1,3</sup>		84	54	Display message 4 <sup>7,9</sup>	PD0119
14	0e	Report fine net weight <sup>1,3</sup>		85	55	Display Message 5 <sup>7,9</sup>	PD0119
15	0f	Report fine tare weight <sup>1,3</sup>		86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
16	10	Report Rate <sup>1,3</sup>		87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
17	11	Report SDV ALxx01 <sup>6</sup>		88	58	Disable weight display <sup>7</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>		89	59	Enable weight display <sup>7</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>		90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
20	14	Report notch filter frequency <sup>3</sup>		91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>	92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>	93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10	100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12	101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
25	19	Report spill value <sup>3,10</sup>	SPxx09	102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>	103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>	104	68	Master control relay OFF <sup>7, 12</sup>	
29	1d	Report last error <sup>3</sup>		105	69	Master control relay ON <sup>7, 12</sup>	
30	1e	Report primary increment size <sup>3</sup>		110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
40	28	Add gross weight to rotation <sup>7</sup>		111	6f	Set target fine feed value <sup>4,10</sup>	
41	29	Add net weight to rotation <sup>7</sup>		112	70	Set - tolerance value <sup>4,10</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>		113	71	Set target value and apply <sup>4, 10, 14</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>		114	72	Start target logic <sup>7,10, 15</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>		115	73	Abort target logic <sup>7,10, 15</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>		116	74	Target use gross weight <sup>7,10</sup>	SPxx04
46	2e	Add rate to rotation <sup>7</sup>		117	75	Target use net weight <sup>7,10</sup>	SPxx04
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>	118	76	Target use rate <sup>7,10</sup>	SPxx04
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>	119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
60	3c	Load numeric tare value <sup>4</sup>		120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
61	3d	Pushbutton tare command <sup>7</sup>		121	79	Enable target latching <sup>7,10</sup>	SPxx06
62	3e	Clear command <sup>7</sup>		122	7a	Disable target latching <sup>7,10</sup>	SPxx06
63	3f	Print command <sup>7</sup>		123	7b	Reset target latch <sup>7</sup>	SPxx07
64	40	Zero command <sup>7</sup>		124	7c	Set target spill value <sup>4, 10</sup>	
65	41	Select scale 1 <sup>7</sup>		131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
66	42	Select scale 2 <sup>7</sup>		160	a0	Apply scale setup <sup>7</sup>	QC0149
67	43	Select next scale <sup>7</sup>		161	a1	Write Calibration to EEPROM <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>	162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>	163	a3	Enable Tare on IND780 console <sup>7</sup>	
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>	164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>	165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>	166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
73	49	Set low-pass filter corner frequency <sup>4</sup>		167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
74	4a	Set notch filter frequency <sup>4</sup>		168	a8	Select scale 3 <sup>7</sup>	
75	4b	Reset ENTER key <sup>7</sup>		169	a9	Select scale 4 <sup>7</sup>	

#### Notes for Table 4-12:

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

### 4.4.7. Floating Point Data Format and Compatibility

#### 4.4.7.1. Operational Overview

In Floating Point Data Format, the PLC and terminal exchange weight, target, and tare data in single-precision floating point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight and rate data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

Some Allen-Bradley PLCs require special integrity checking to communicate floating point numbers across DeviceNet. The Allen-Bradley PLC-5 must check two data integrity bits to verify the integrity of the floating point data it reads from the terminal. Allen-Bradley SLC programs always read valid floating-point data from the terminal and do not have to make special checks to guarantee the validity of the floating-point data.

The two integrity bits function as follows when communicating with the Allen-Bradley PLC-5 DeviceNet scanner card. One bit is in the beginning word of the data; the second is in the ending

byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC-5 program detects this condition, it should send a new command to the terminal.

The Allen-Bradley SLC PLC programs do not have to make special checks to guarantee the validity of the floating-point data.

#### 4.4.7.2. Floating Point Command Examples

Table 4-13 through Table 4-16 provide floating point command examples.

**Table 4-13: Data Requirement: Only Net Weight Sent (continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word O:XX1	none required		
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word the IND780 terminal will update the net value every interface update cycle.				

**Table 4-14: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2 (PLC sends command to set target 1 cutoff value)	110 (dec) loaded into command word O:XX1	floating point value = 21.75		
3 (IND780 terminal sees new command, loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word O:XX1			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	(null value)

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.				

**Table 4-15: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word O:XX1			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word O:XX1			
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word O:XX1			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	(null value)
At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.				
7 (PLC sends the command to begin the rotation at interface update cycle)	0 (dec) loaded into command word O:XX1			
8 (IND780 terminal sends gross weight at interface update cycle ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 6	Floating point value = rate

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value at next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 4-16: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word O:XX1			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word O:XX1			
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word O:XX1			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	(null value)
At this point, the rotation has been set up. Now the PLC needs send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word O:XX1			

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
8 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net weight
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word O:XX1			
10 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 2 F.P. ind = 6	Floating point value = rate
At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value				

#### 4.4.8. Controlling the Discrete I/O Using a PLC Interface

The IND780 terminal provides the ability to directly control some of its discrete outputs and read some of its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the terminal's discrete I/O updates are synchronized with the terminal's interface update cycle rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the IND780 Terminal Technical Manual for discrete I/O wiring. Also note that the outputs must be unassigned in the IND780 terminal setup.

## 4.5. Hardware Setup

### 4.5.1. Wiring

The IND780 wiring is shown below. Consult <http://www.odva.org/> for additional DeviceNet wiring information.



**Figure 4-4: DeviceNet Connector Pin Numbering**

Pin Number	Description	Wire Color
1	V -	Black
2	CAN L	Green
3	Drain	
4	CAN H	White
5	V +	Red

## 4.6. Software Setup

The IND780 terminal automatically detects the presence of a DeviceNet Kit option board if one is installed. When the option is detected, the IND780 terminal enables the DeviceNet parameters in setup at **Communication > PLC Interface > DeviceNet**. Figure 4-5 graphs the DeviceNet and PLC Data Format program blocks. Default values are indicated with an asterisk (\*).

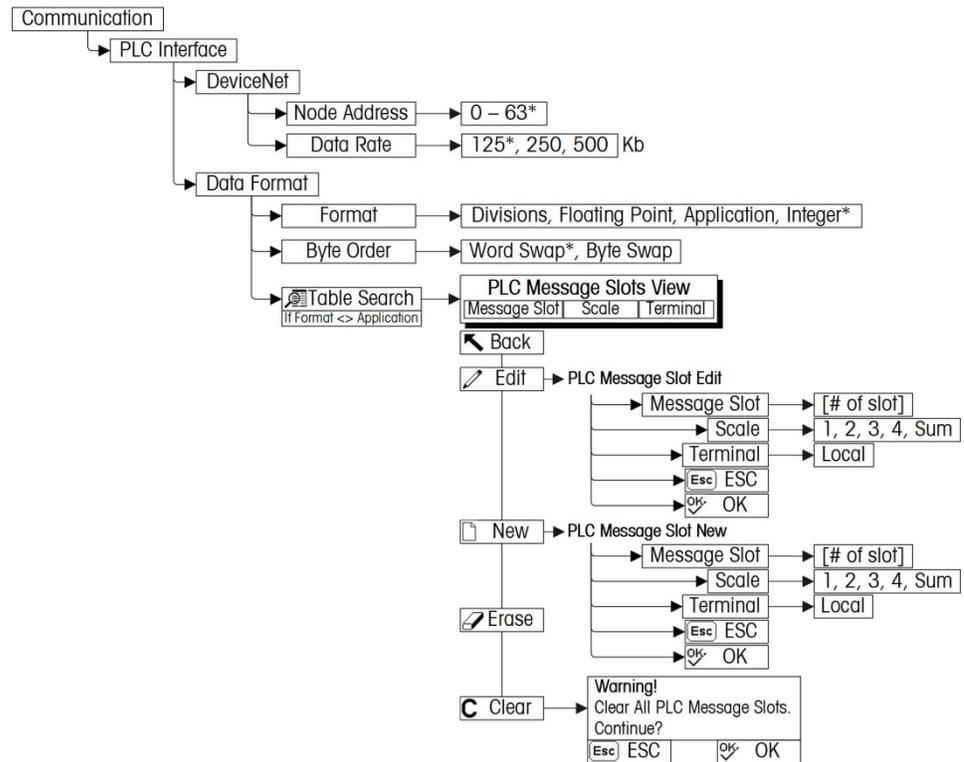
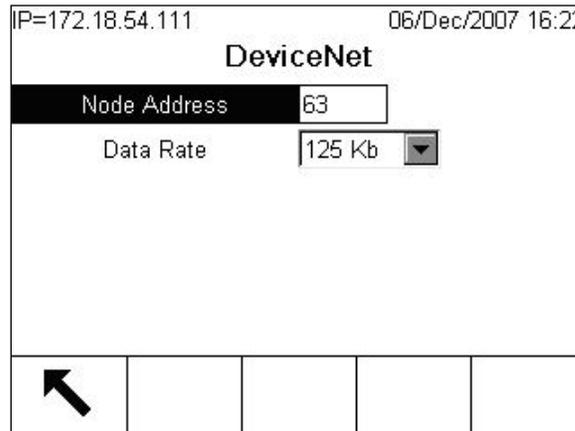


Figure 4-5: The DeviceNet Program Block and Data Format Setup Block

## 4.6.1. DeviceNet and Data Format Setup Blocks

### 4.6.1.1. DeviceNet setup

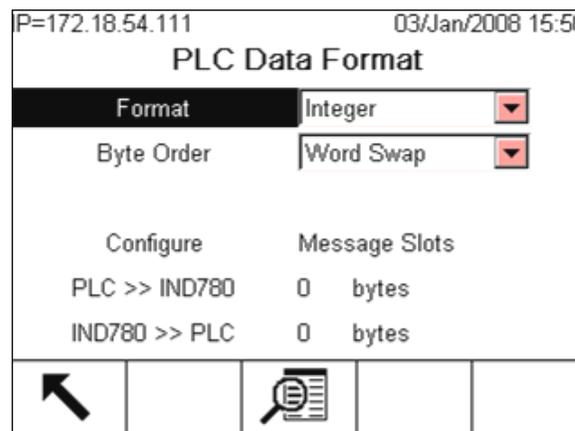
In Setup, access **Communication > PLC Interface > DeviceNet** (Figure 4-6). Select the Node Address field and set an address, from 0 to 63. The address should be unique in the DeviceNet network. Select from a list of possible Data Rates (125Kb, 250Kb or 500 Kb).



**Figure 4-6: DeviceNet Setup Screen**

### 4.6.1.2. Data Format setup

1. From Setup, access **Communication > PLC Interface > Data Format**. Format may be selected from a drop-down list as shown in Figure 4-7. When Application format is selected, only the Format box appears on this screen.



**Figure 4-7: PLC Data Format Screen**

2. Select the Format (Integer, Divisions, Floating Point or Application): [default Integer] Changing the Format will delete any existing Message Slots.

<b>Integer</b>	Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).
<b>Divisions</b>	Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.

- Floating Point** Displays weight in floating point data format
- Application** Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies (8 bytes I/O maximum). For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

3. Select Byte Order: [default Word Swap]

**Word Swap** Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** Makes the floating point format compatible with S7 PROFIBUS.

4. The bottom of this screen (Figure 4-7) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.

5. Press the VIEW TABLE softkey  to set up the number of Message Slots (maximum number is 2 if integer or Division data format is selected; default is 0), for communications between the IND780 and the PLC. Figure 4-8 shows the Message Slots View.

IP=172.18.54.111		06/Dec/2007 16:25	
<b>PLC Message Slots View</b>			
Message	Scale	Terminal	
▶ 1	1	Local	
2	- End -		
			 <b>C</b>

**Figure 4-8: PLC Message Slots View Screen**

Press the EXIT softkey  to exit the view screen.

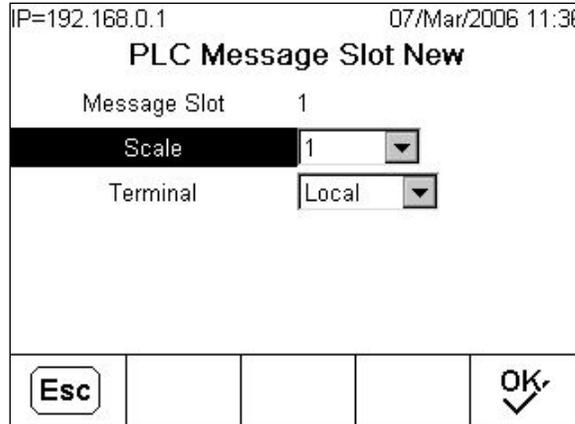
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey **C** to delete all existing slots

6. Figure 4-9 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 4-8).



**Figure 4-9: New Message Slot Screen**

**Scale** Enter the scale number to be associated with the new Message Slot.

**Terminal** Default is Local.

## 4.7. Troubleshooting

All DeviceNet nodes are required to have 2 status LEDs. These LEDs (labeled in Figure 4-2) indicate module and network status. See definitions below.

### 4.7.1.1. Module status LED (MS)

This LED displays the status of the IND780 Terminal DeviceNet board.

LED State	Meaning
Solid Green	Normal operation
Flashing Green	DeviceNet board fault
OFF	No power to the DeviceNet board
Solid Red	Unrecoverable board fault
Flashing Red	Recoverable fault
Flashing Orange	Board performing self-test

- **Note:** If the module status LED indicates anything after powering up the unit and attaching the DeviceNet cable, the IND780 Terminal must be powered down and restarted. If the LED continues to show a condition other than solid green, replace the board.

#### 4.7.1.2. Network status LED (NS)

This LED displays network status.

- Solid Green** Node is communicating to scanner
- Flashing Green** Device is connected to the network but not being scanned  
The most common reason for this is the device has not been added to the scan list. Consult DeviceNet configuration tool's help in order to commission the node and put it in the scan list.
- OFF** No DeviceNet power
- Solid Red** Critical Link error  
This error typically indicates a cable problem
- Blinking Red** Connection Timeout

## 4.8. DeviceNet Option Kit

No spare parts are associated with the DeviceNet option kit. The kit CIMF part number is 72193580. Table 4-17 shows what comes in the kit.

**Table 4-17: DeviceNet Option Kit**

Description	Qty.
Installation Instructions	1
PCB Package	1
Cylindrical ferrites	3
M3 x 8 screws	3

## 4.9. DeviceNet Commissioning and Configuration Examples

The user must register the EDS, connect the device and add the IND780 Terminal to the DeviceNet master's scanlist. Note that every vendor's software is different. Depending upon master and software, the user may have to cycle power on the master in order to complete the commissioning of any device added to the network. Consult the master's documentation for more information. The following example is for Rockwell software and Logix5000 processor.

### 4.9.1. Configuring the IND780 Terminal with RSNetworx for DeviceNet

The EDS file located on the CD-ROM supplied with the IND780 Terminal contains configuration information to allow RSNetWorx for DeviceNet to set up a single polled I/O connection between a METTLER TOLEDO IND780 Terminal and DeviceNet master/scanner. Note that this order of operations isn't the only way of configuring the IND780 terminal.

4.9.1.1. Registering the EDS file

The EDS file must first be registered into RSNetWorx for DeviceNet. This is accomplished using the EDS Wizard.

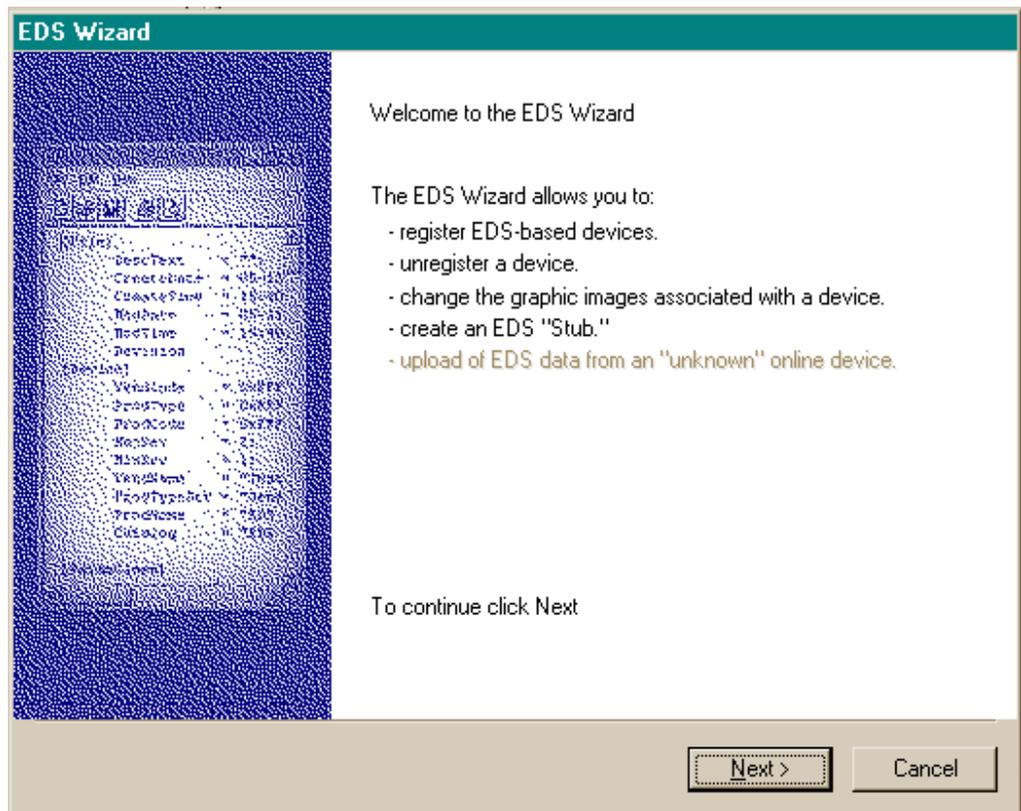
To access the EDS Wizard:

Click **T**ools then **EDS Wizard...** to begin the registration process.



**Figure 4-10: Accessing the EDS Wizard**

The EDS Wizard Welcome screen appears.



**Figure 4-11: EDS Wizard Welcome**

Click Next to begin the registration process.

In the Options screen, make sure the Register an EDS file(s) radio button is selected.

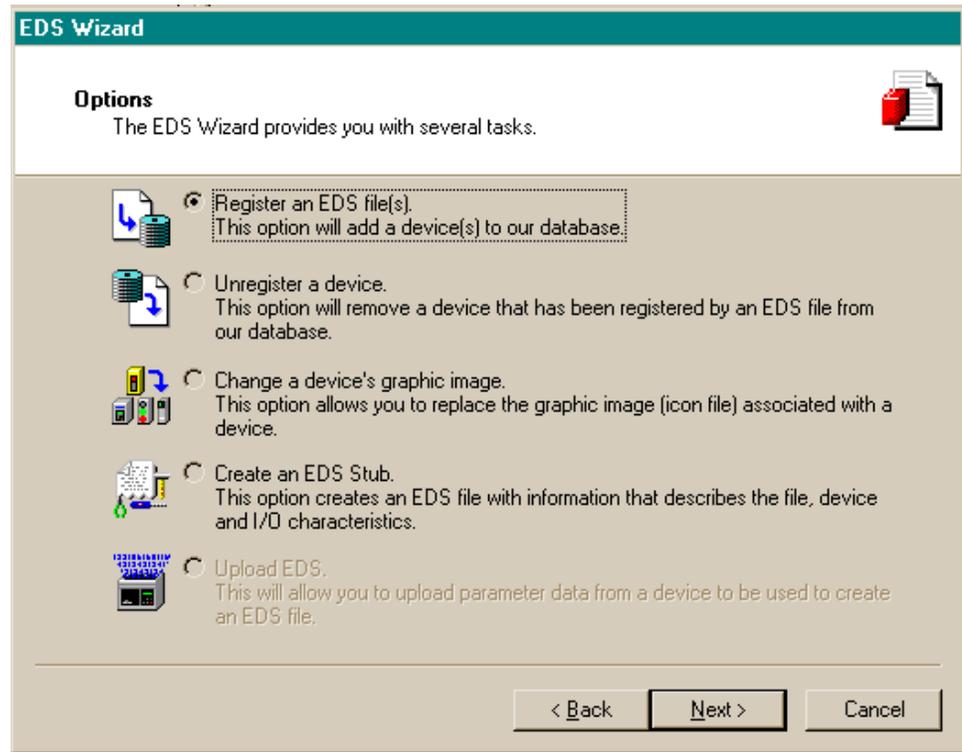


Figure 4-12: EDS Wizard Options Screen

Click Next, then Browse to select a file to register.

Browse to the appropriate location and select the file MT\_IND-DNET.eds. (The EDS file is located on the CD-ROM.) Click the Open button.

Confirm that the correct file is showing in the Named: field, then click Next.

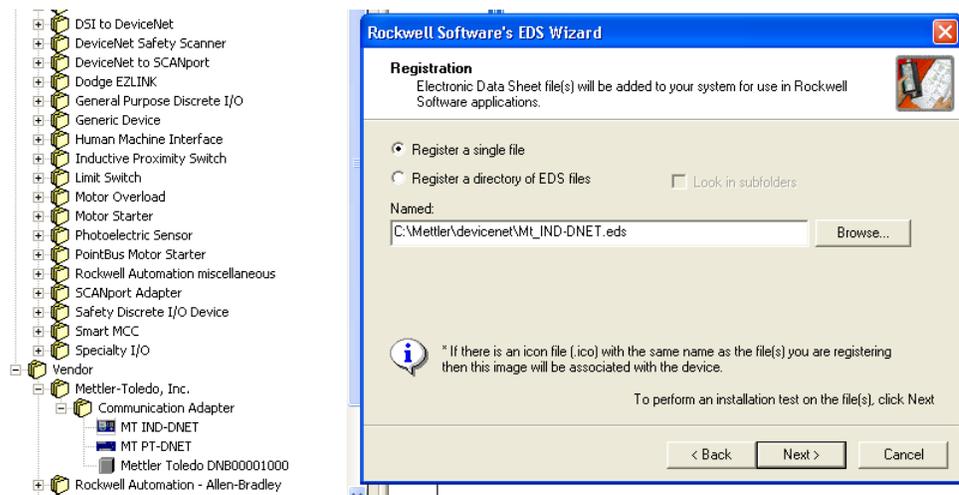
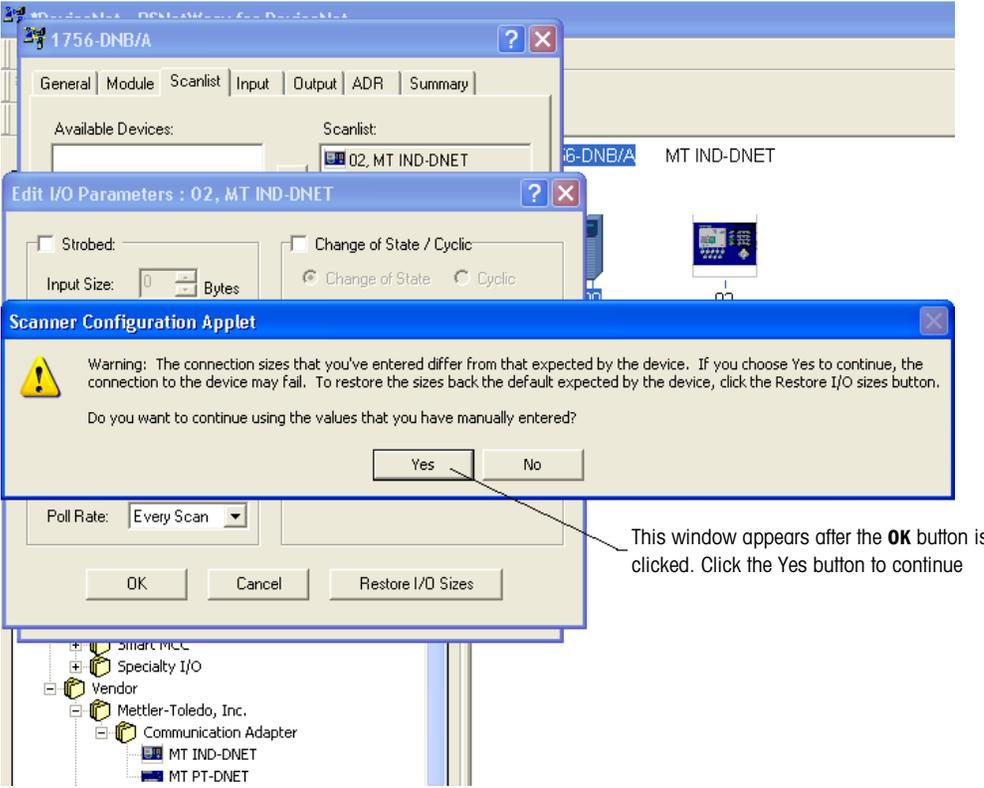


Figure 4-13: File Selected, Ready to Register

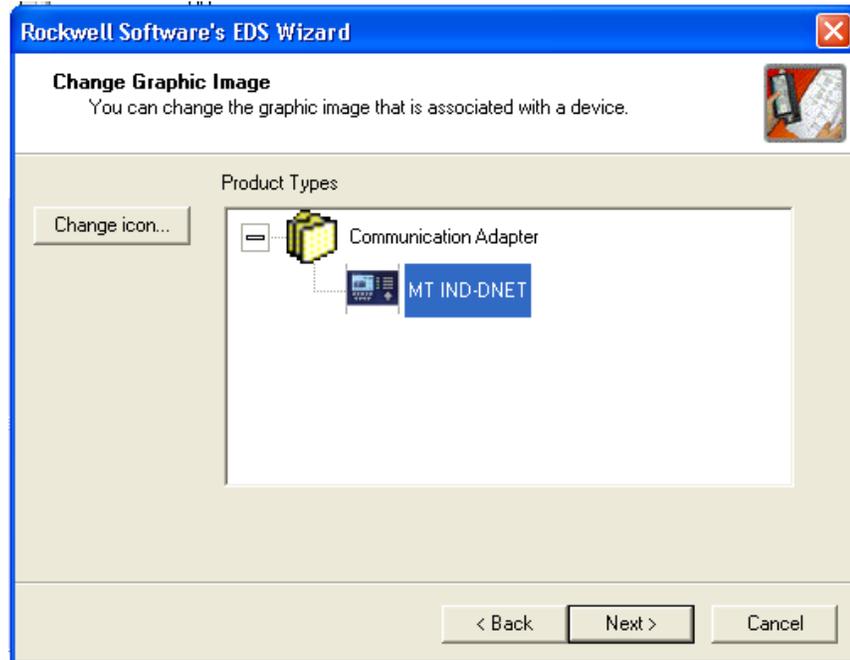
Acknowledge the error applet. This error is generated due to the EDS file being generic for other MT devices. The IO sizes will be configured later in the process.



**Figure 4-14: File Validity Confirmed**

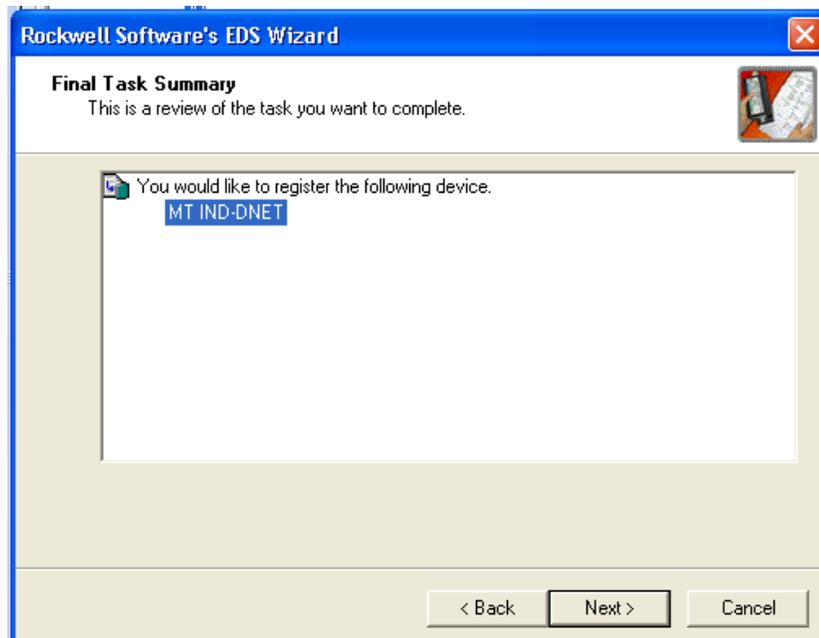
Ensure that the MT\_IND-DNET.ico icon is selected.

**Note:** RSNetWorx for DeviceNet will not be able to find the icon unless it is in the same directory as the EDS file.



**Figure 4-15: EDS Wizard Graphic Image (Icon) Selection**

The Final Task Summary screen (Figure 4-16) will appear. Click Next.



**Figure 4-16: Final Task Summary Screen**

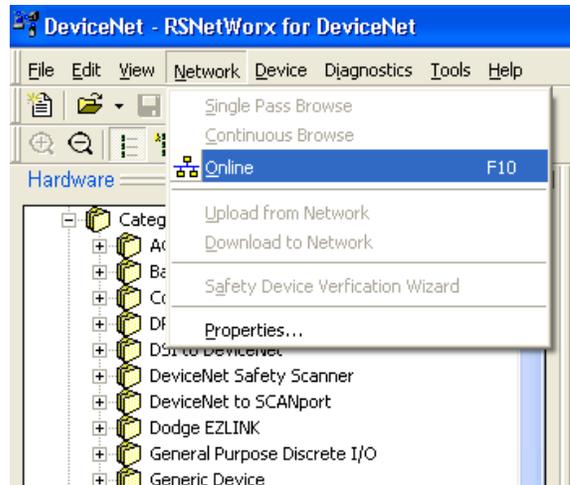
#### 4.9.1.1.1. Setting up an I/O Connection

After the EDS file has been registered, RSNetWorx is used to set up a polled connection between the METTLER TOLEDO IND Terminal and the DeviceNet master/scanner.

To set up the connection:

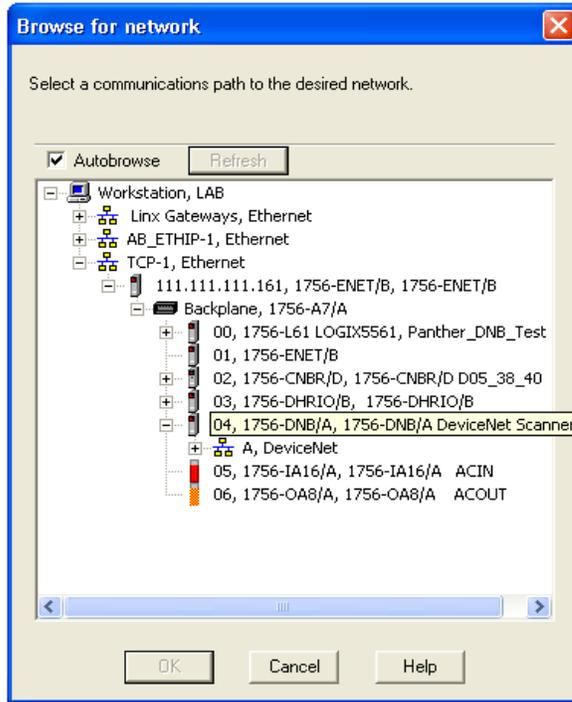
**Note:** You must add the DeviceNet scanner card and choose the proper revision before going online.

1. Select Network then Online to browse the DeviceNet network.



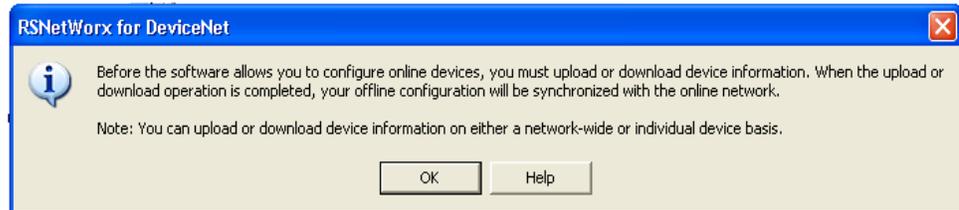
**Figure 4-17: RSNetWorx Online Browse**

2. Select the appropriate network path. In this case (Figure 4-18), 1756-DNB/A DeviceNet Scanner is selected.



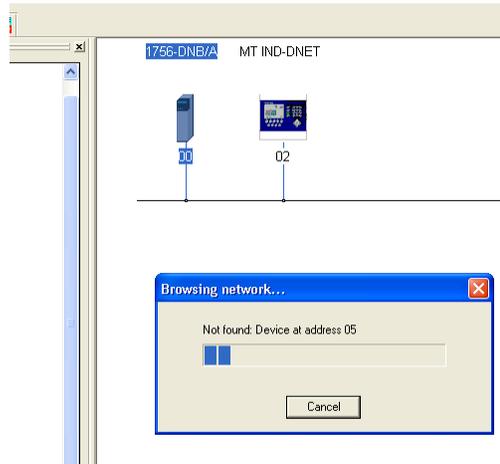
**Figure 4-18: Browse for network**

3. Click OK to continue. A dialog box like the one shown in Figure 4-19 will appear. Note that you may be asked to upload or download, depending on the version of software used.



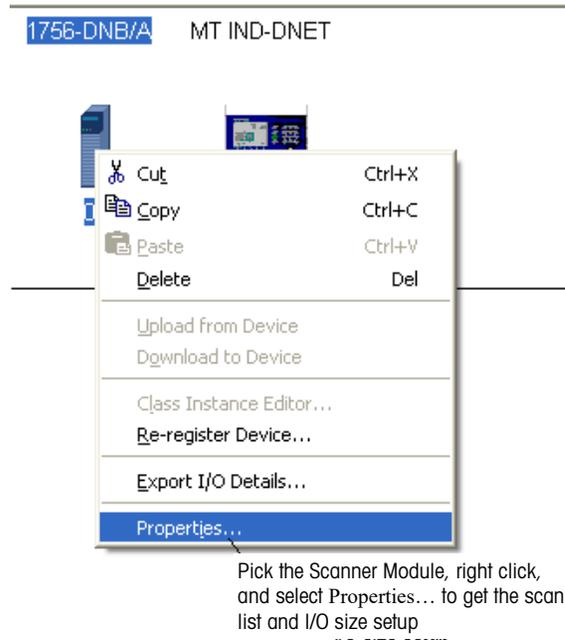
**Figure 4-19: Confirmation Dialog Box**

- Click OK in the dialog box. A Browsing network... box will display with a progress meter indicating that the process is underway.



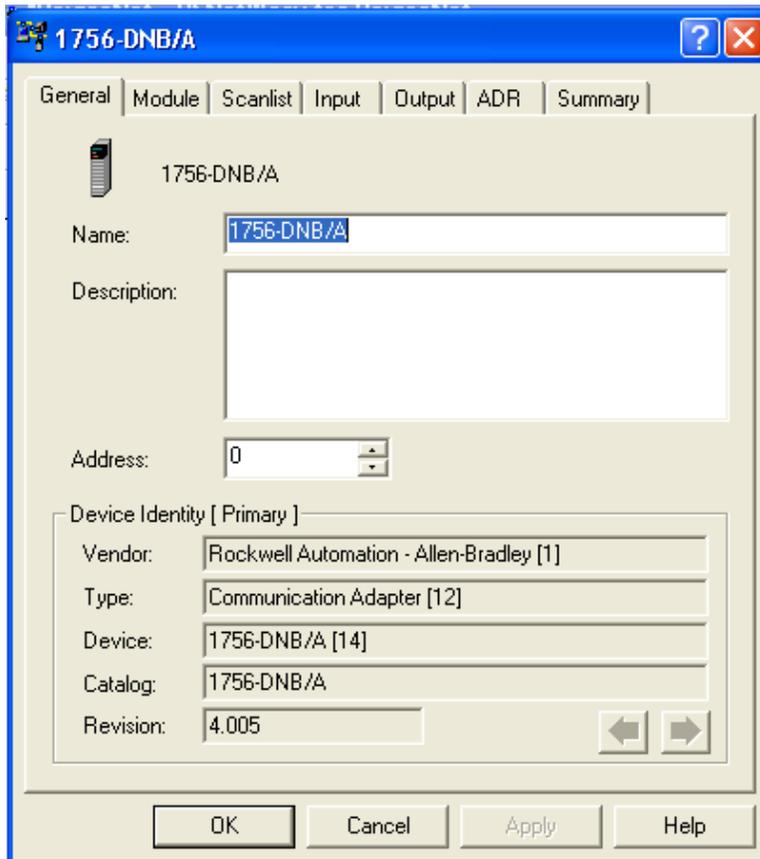
**Figure 4-20: Browsing Network Underway**

- Once the scanner has browsed the entire network, add IND Terminal to the 1756-DNB/A's scanlist by right-clicking on the scanner icon in the Graph window (note the tab at the bottom of Figure 4-21), and choosing Properties...



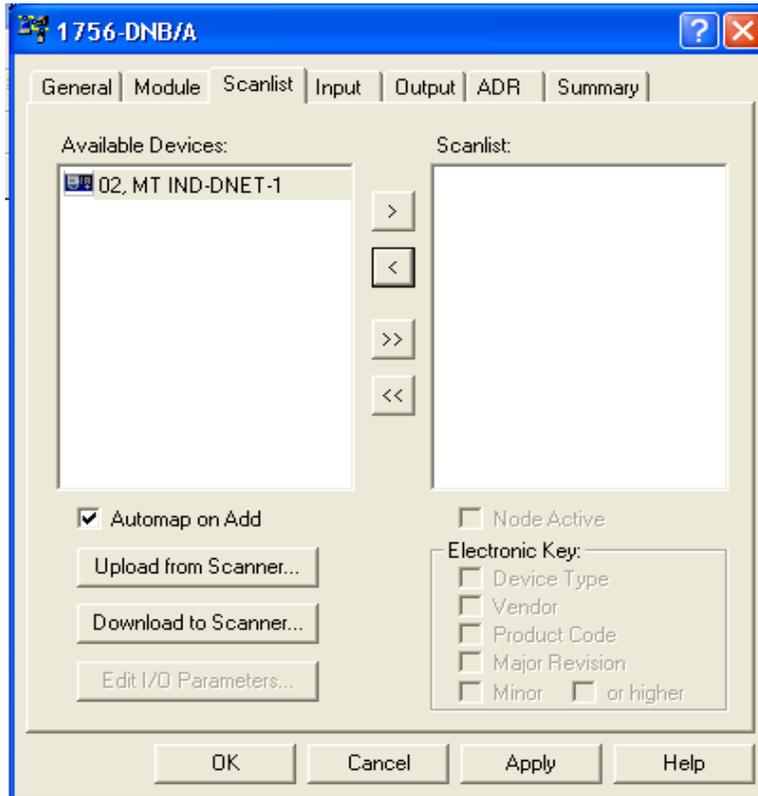
**Figure 4-21: Accessing Scanner Properties**

6. The scanner's properties dialog box will appear, as in Figure 4-22.



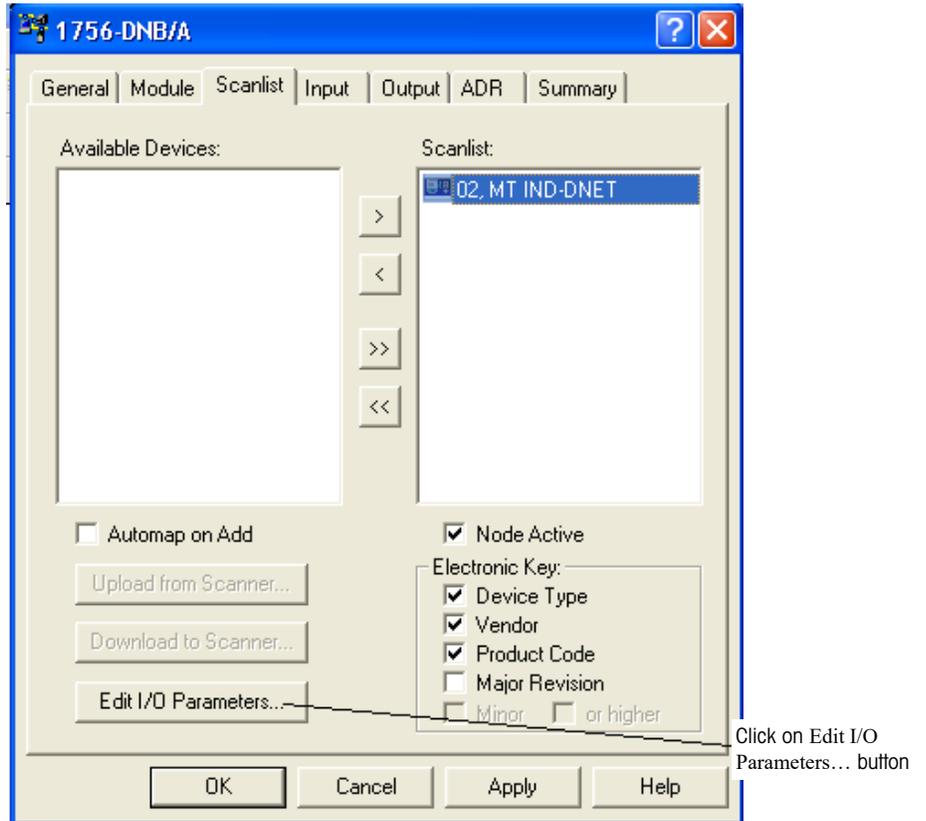
**Figure 4-22: Scanner Properties Dialog: Initial View**

7. Click the Scanlist tab in the properties dialog box. The view shown in Figure 4-23 will appear.



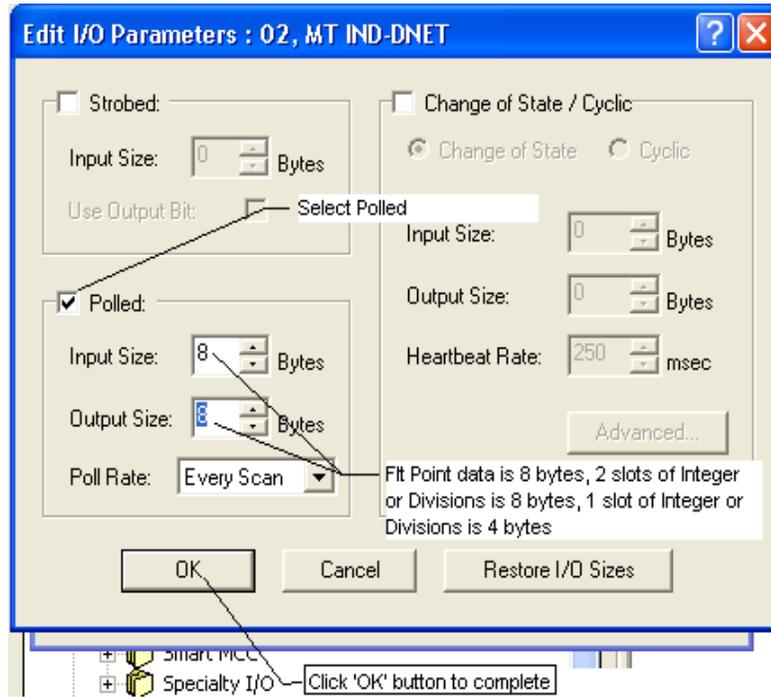
**Figure 4-23: Scanner Properties Dialog: Scanlist Tab Showing**

8. Highlight the IND Terminal (MT IND-DNET) and left-click to add it to the Scanlist. Once the IND Terminal is added, it will appear in the right pane (Figure 4-24). Click OK.



**Figure 4-24: IND Terminal Added to Scanlist**

- The next step is to edit the I/O parameters of the IND terminal. The I/O size depends on the data type and the number of slots selected in the terminal. Note that slots is a terminal phrase that isn't used in typical DeviceNet terms; it was derived from previous PLC memory mapping. It is used in the terminal setup to remain consistent across the Mettler Toledo terminal line. Integer or Divisions with one slot will be 4 bytes/in 4 bytes out. Integer or Division with 2 slots is 8 bytes in/8 bytes out. Float is always 8 bytes in and out.



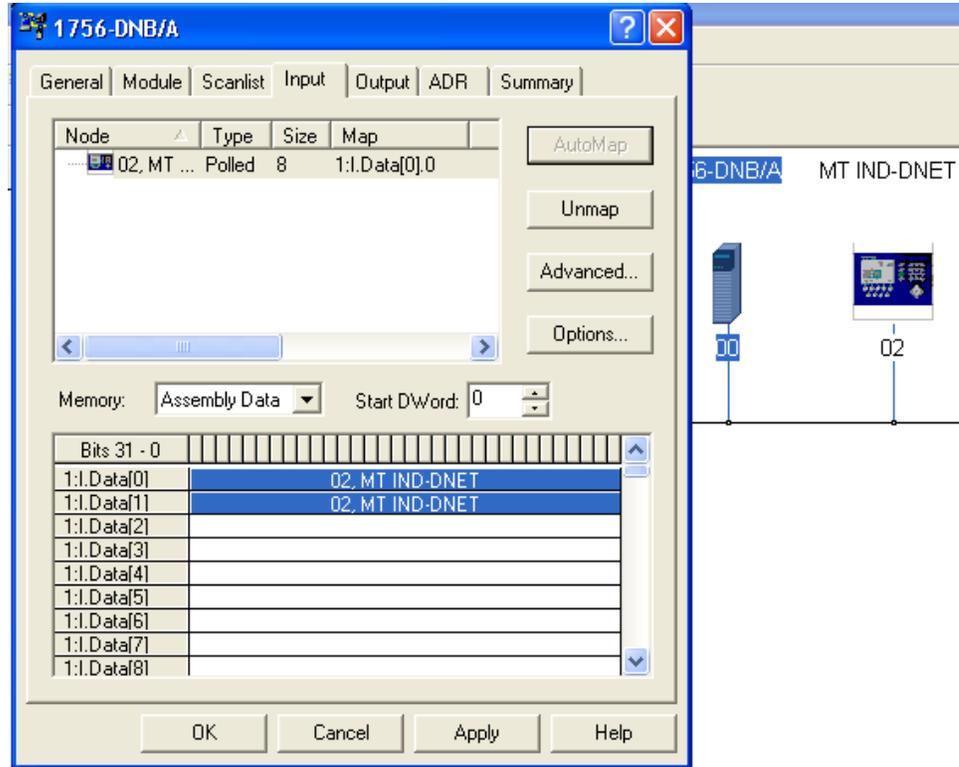
**Figure 4-25: Editing I/O Parameters**

- Download the configuration to the scanner card, in order to commission the network. In the prompt that appears (Figure 4-26), click Yes to continue. Note that some scanner cards may require power down for changes to take effect.



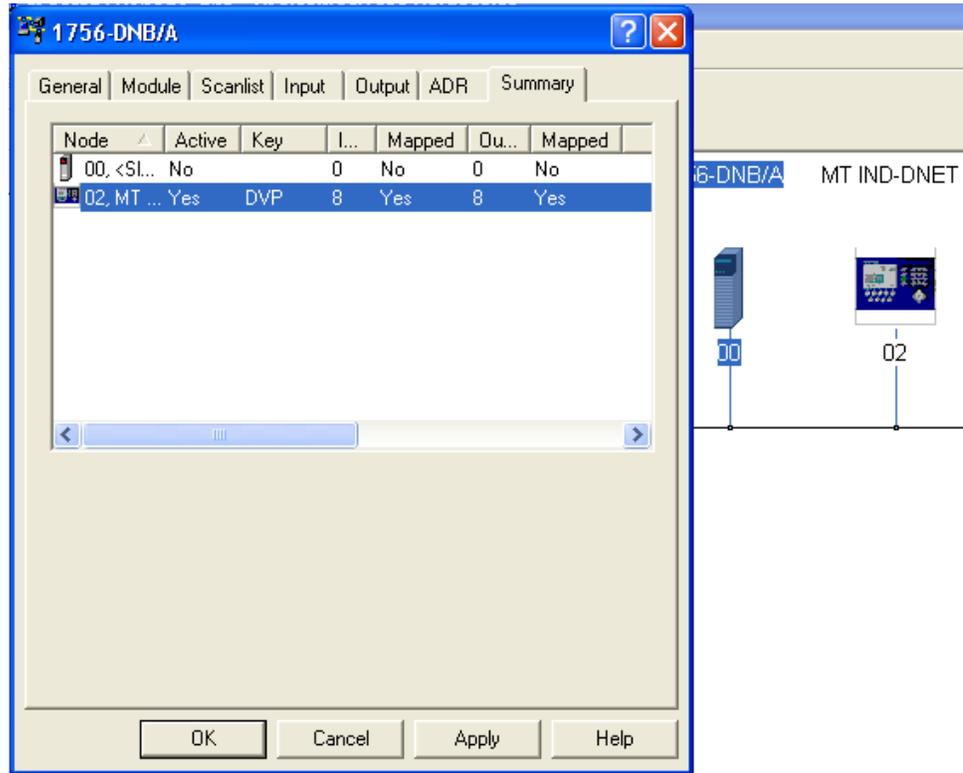
**Figure 4-26: Network Download Confirmation Prompt**

- Once the IND Terminal has been added to the scanlist, access the Properties dialog to verify its I/O mapping (Figure 4-27 and Figure 4-28) within the scanner card. Note that auto or manual mapping can be used. Consult the master's documentation for advance I/O mapping options. If manual mapping is used, Be sure to start with an unused section of memory for offset.



**Figure 4-27: IND Terminal Mapping**

12. Verify that I/O mapping is complete by choosing the summary tab. Note that the Mapped columns show Yes for the IND Terminal.



**Figure 4-28: Summary Tab**

## 4.9.2. PLC Programming

The IND Terminal is now ready to communicate to the master or DeviceNet scanner. You must configure the PLC or other master with DeviceNet scanner to send and receive data via polled messaging. For this example the RSLogix5000 software is used in conjunction with the ControlLogix5000 processor with DeviceNet scanner card. Note that these examples may or may not be applicable depending upon whether you have an existing program or different version of software.

1. Create a new project.

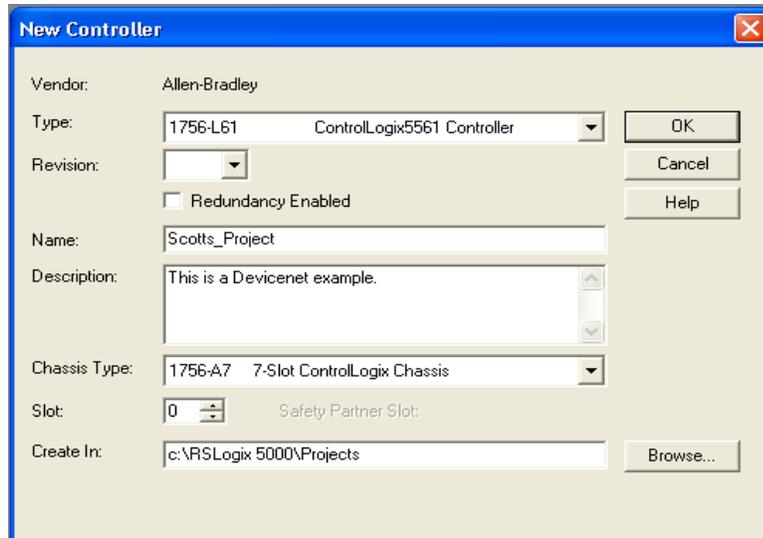


Figure 4-29: New Project Dialog

2. Add DeviceNet scanner card to existing chassis.

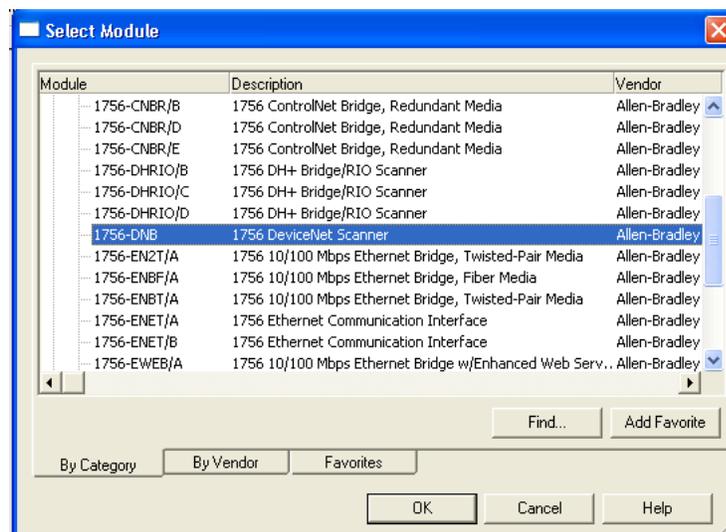
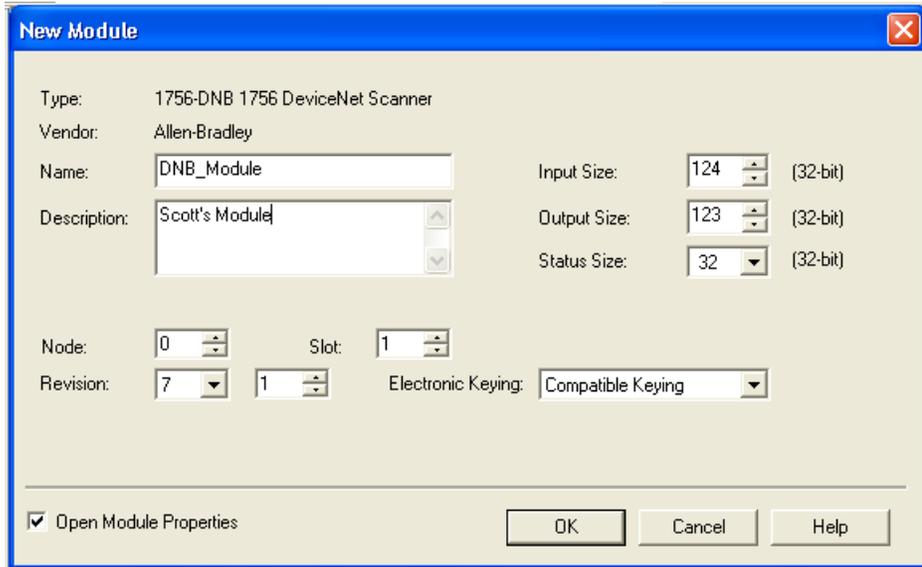


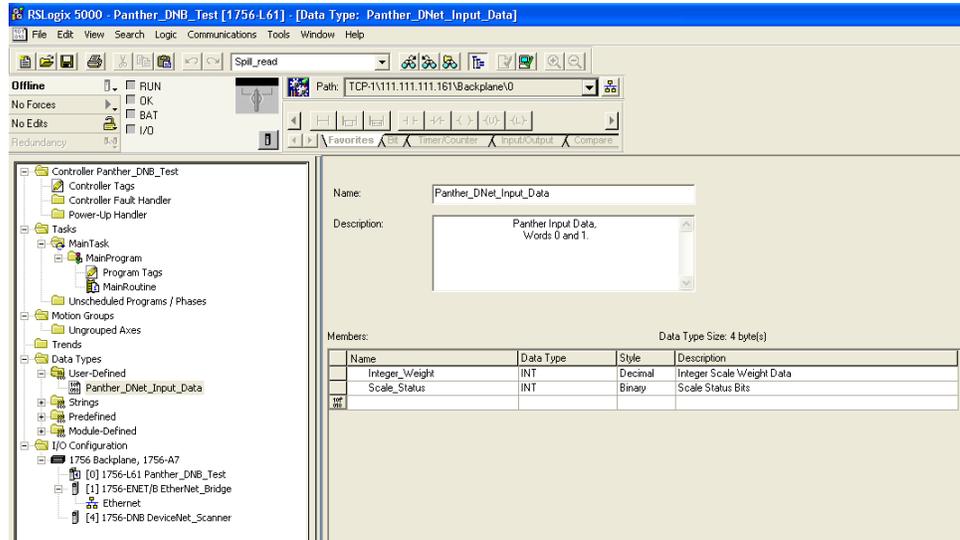
Figure 4-30: Adding DeviceNet Scanner

- Configure scanner card. Refer to Rockwell software documentation for more information.



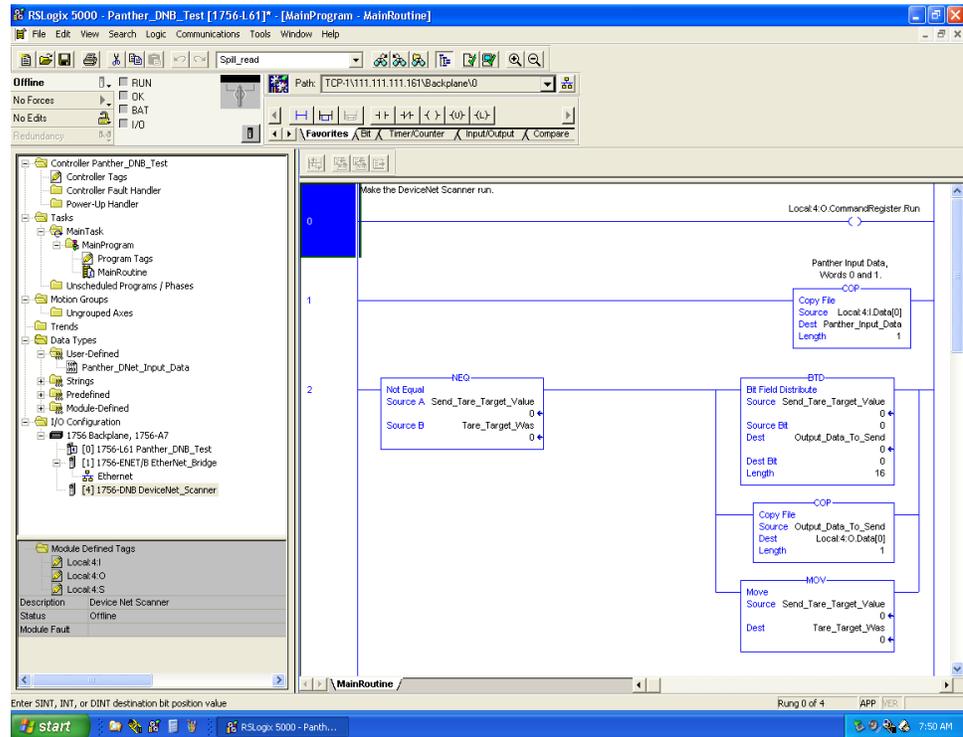
**Figure 4-31: Ready to Go Online**

- Create User-Defined tags. These are under data type. In this example the Integer Data Format is being used; create these tags in order to get the data from the scanner's memory location into a useable/readable format. Create 2 tags. One is for the weight data (single 16 bit integer) that can be read in decimal format and the other is for the status information (16 bit value) this is read in binary. User-Defined tags will be different for the Floating Point Data Format.



**Figure 4-32: Creating User-Defined Tags**

5. Create an unconditional rung of logic that activates the command register run bit. This bit is required to activate the DeviceNet scanner card. Then create another unconditional rung that activates the copy command. Copy the scanner's I/O mapped data into the user defined tags.



**Figure 4-33: Creating an Unconditional Rung**

6. At this point you can download the program and run it and, provided the wiring and configuration are correct, you will receive weight and status information back from the IND780 Terminal.
  - Note that examples, and the EDS and IND Terminal icon files, are located on the IND780 documentation CD that comes with every unit.

# 5 EtherNet/IP

## 5.1. Overview

EtherNet / IP, short for "EtherNet Industrial Protocol," is an open industrial networking standard that takes advantage of commercial, off-the-shelf EtherNet communication chips and physical media. This networking standard supports both implicit messaging (real-time I/O messaging) and explicit messaging (message exchange). The protocol is supported by ControlNet International (CI), the Industrial Ethernet Association (IEA) and the Open DeviceNet Vendor Association (ODVA).

EtherNet / IP utilizes commercial, off-the-shelf EtherNet hardware (for example, switches and routers) and is fully compatible with the Ethernet TCP/IP protocol suite. It uses the proven Control and Information Protocol (CIP) to provide control, configuration, and data collection capability.

The kit enables the IND780 terminal to communicate to EtherNet / IP Programmable Logic Controllers (PLCs) through direct connection to the EtherNet / IP network at either 10 or 100 MBPS speed. The kit consists of a backplane-compatible I/O module, mounting hardware, and a ferrite. Software to implement the data exchange resides in the IND780 terminal.

The EtherNet / IP Kit option has the following features:

- User-programmable IP addressing.
- Capability for bi-directional discrete mode communications (Class 1 Messaging) of weight or display increments, status, and control data between the PLC and the IND780.

Figure 5-1 and Figure 5-2 show an EtherNet / IP module and its components. Note that the module's address is set in software (see Figure 5-8), and the DIP switches indicated in Figure 5-2 must all be set to OFF.

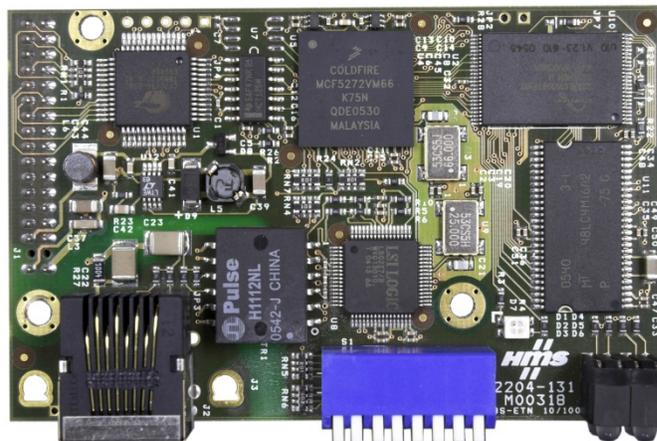


Figure 5-1: EtherNet / IP Module



**Figure 5-2: EtherNet / IP Module Components**

**5.1.1. Definition of Terms**

Some terms (such as Target) used by the EtherNet / IP PLC application have a different sense from their use by the IND780 terminal. Table 5-1 provides definitions specific to EtherNet / IP.

**Table 5-1: EtherNet / IP Definition of Terms**

Term	Definition
<b>Adapter Class</b>	An Adapter Class product emulates functions provided by traditional rack-adapter products. This type of node exchanges real-time I/O data with a Scanner Class product. It does not initiate connections on its own.
<b>Class 1 Messaging</b>	In EtherNet / IP communication protocol scheduled (cyclic) message transfer between a PLC and CIP Adapter Class device.
<b>Class 3 Messaging</b>	In EtherNet / IP communication protocol unscheduled message transfer between a PLC and CIP Adapter Class device. This is used by the IND780 for explicit messaging.
<b>Connected Messaging</b>	A connection is a relationship between two or more application objects on different nodes. The connection establishes a virtual circuit between end points for transfer of data. Node resources are reserved in advance of data transfer and are dedicated and always available. Connected messaging reduces data handling of messages in the node. Connected messages can be <b>Implicit</b> or <b>Explicit</b> . See also <b>Unconnected Messaging</b> .
<b>Connection Originator</b>	Source for I/O connection or message requests. Initiates an I/O connection or explicit message connection.
<b>Explicit Messaging</b>	Explicit Messages can be sent as a connected or unconnected message. CIP defines an Explicit Messaging protocol that states the meaning of the message. This messaging protocol is contained in the message data. Explicit Messages provide a one-time transport of a data item. Explicit Messaging provide the means by which typical request/response oriented functions are performed (e.g. module configuration). These messages are typically point-to-point.
<b>Implicit Messaging</b>	Implicit Messages are exchanged across I/O Connections with an associated Connection ID. The Connection ID defines the meaning of the data and establishes the regular/repeated transport rate and the transport class. No messaging protocol is contained within the message data as with Explicit Messaging. Implicit Messages can be point-to-point or multicast and are used to transmit application-specific I/O data. This term is used interchangeably with the term <b>I/O Messaging</b> .
<b>I/O Client</b>	Function that uses the I/O messaging services of another (I/O Server) device to perform a task. Initiates a request for an I/O message to the server module. The I/O Client is a <b>Connection Originator</b> .
<b>I/O Messaging</b>	Used interchangeably with the term <b>Implicit Messaging</b> .
<b>I/O Server</b>	Function that provides I/O messaging services to another ( <b>I/O Client</b> ) device. Responds to a request from the I/O Client. I/O Server is the target of the connection request.

Term	Definition
<b>Message Client</b>	Function that uses the Explicit messaging services of another ( <b>Message Server</b> ) device to perform a task. Initiates an Explicit message request to the server device.
<b>Message Server</b>	Function that provides Explicit messaging services to another ( <b>Message Client</b> ) device. Responds to an Explicit message request from the Message Client.
<b>Scanner Class</b>	A Scanner Class product exchanges real-time I/O data with <b>Adapter Class</b> and Scanner Class products. This type of node can respond to connection requests and can also initiate connections on its own.
<b>Target</b>	Destination for I/O connection or message requests. Can only respond to a request, cannot initiate an I/O connection or message.
<b>Unconnected Messaging</b>	Provides a means for a node to send message requests without establishing a connection prior to data transfer. More overhead is contained within each message and the message is not guaranteed destination node resources. Unconnected Messaging is used for non-periodic requests (e.g. network "Who" function). Explicit messages only. See also <b>Connected Messaging</b> .

### 5.1.2. Communications

The IND780 terminal utilizes component parts to ensure complete compatibility with the Allen-Bradley EtherNet / IP network. An IND780 terminal is recognized as a generic EtherNet / IP device by the PLC.

Each EtherNet / IP option connected to the EtherNet / IP network represents a physical IP Address. The connection is made via a RJ-45 connector on the option card (see Figure 5-2).

The wiring between the PLC and the IND780 EtherNet / IP connection uses EtherNet twisted pair cable. The cable installation procedures and specification including distance and termination requirements are the same as recommended by Allen-Bradley for the EtherNet / IP network.

The IND780 only uses Class 1 cyclic data for discrete data and explicit messages for access to the IND780 Shared Data Variables. Explicit message blocks may be connected or unconnected; the PLC programmer must make this choice.

### 5.1.3. IP Address

Each EtherNet / IP option represents one physical IP Address. This address is chosen by the system designer, and then programmed into the IND780 terminal and PLC. There is also a DHCP client feature for EtherNet / IP to use a host server to assign addresses. The IND780 terminal's address is programmed in setup at **Communication > PLC Interface > EtherNet / IP**. IND780 IP Address entry must be unique for each IND780.

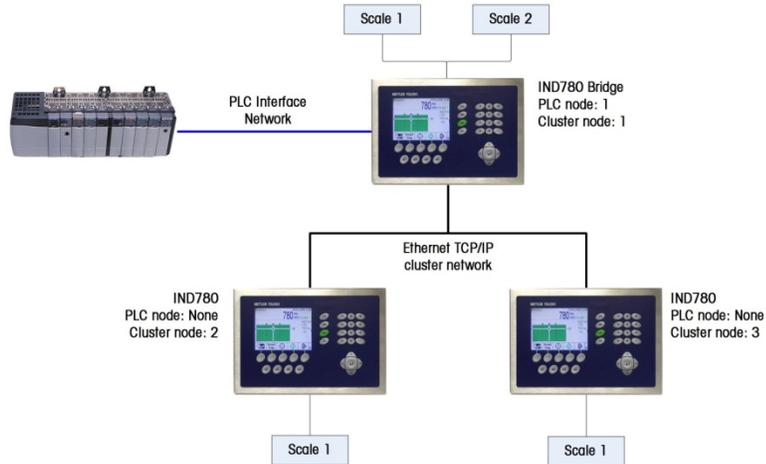
### 5.1.4. Data Formats

The EtherNet / IP Kit option provides discrete data transfer, Class 1 messaging. Discrete data is continuously available. The EtherNet / IP option has its own logical IP address to send and receive information to and from the PLC. There are five data formats: Integer, Divisions, Floating Point, Template and Application.

## 5.2. Sharing a PLC Interface

### 5.2.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. The following example (Figure 5-3) shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 5-3: Terminal Cluster Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate PLC chapters in the IND780 PLC Interface Manual for further details on message slots' limitations.

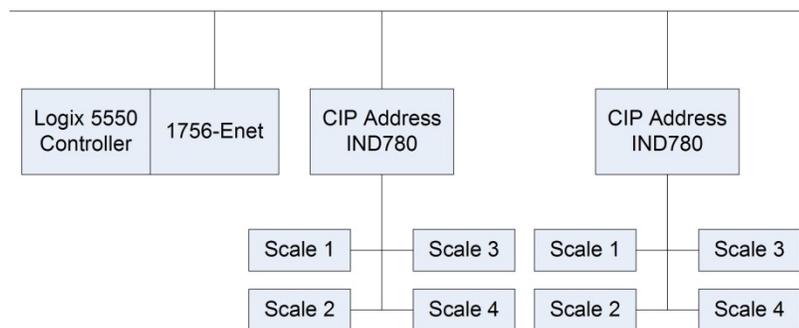
To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure 5-4 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to the Software Setup section, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

P=172.18.54.122		09/Oct/2007 17:20	
PLC Message Slots View			
Message	Scale	Terminal	
1	1	Local	
2	1	6	
3	- End -		

**Figure 5-4: PLC Message Slots View Screen**

Figure 5-5 shows a typical EtherNet/IP network arrangement, with IND780 terminals connected directly to a Logix controller without the use of the terminal’s PLC interface sharing.



**Figure 5-5: Typical Network Topology**

## 5.3. Data Definition

The EtherNet / IP Kit option uses discrete data for its communication with PLCs. Data transfer is accomplished via the PLC’s cyclic messaging.

### 5.3.1. Data Integrity

The IND780 has specific bits to allow the PLC to confirm that data was received without interruption and that the IND780 is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received by the IND780. Refer to the data charts for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage.

### 5.3.2. Assembly Instances of Class 1 Cyclic Communications

Class 1 cyclic communications is used for transfer of Discrete Data between the PLC and the IND780.

The PLC Input Assembly Instance is 100 (decimal). This instance is used for all Data Formats and data size requirements.

The PLC Output Assembly Instance is 150 (decimal). This instance is used for all Data Formats and data size requirements.

The IND780 uses data only. Configuration data is not used or required. Within the PLC EtherNet / IP Interface setup set the Configuration Instance to 1 and the data size to zero.

The EDS file provided on the Documentation CD has no Assembly Instance or data size limitations. The IND780 programming controls the Assembly Instance and data size limitations.

**5.3.3. Discrete Data**

Five formats of discrete data are available with the EtherNet / IP Kit option: Integer, divisions, floating point, template and application. Only one type of data format may be selected and used by the IND780.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word (see Table 5-6 and Table 5-7 for an explanation) numerical values.

The floating-point format allows bi-directional communication of discrete bit encoded information and numeric data encoded in IEEE 754, single precision floating point format.

The data format of discrete data will affect the data size required in the configuration of the PLC. The IND780 console PLC message slot setup screen provides data size requirements in bytes, except for the Application data format.

Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions or increments. The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Floating point is the only data format that includes decimal point information. Integer and division formats ignore decimal points. Accommodation of decimal point location must take place in the PLC logic, when it is needed with these formats.

Changing the Data Format to be used by the IND780 will clear all Message Slots. Data format is configured in setup at **Communication > PLC Interface > Data Format** (Figure 5-9).

5.3.3.1. Examples

250 x .01 scale				
IND780 Displays:	0	2.00	51.67	250.00
Format sent:				
Integer	0	200	5167	25000
Division	0	200	5167	25000
Floating Point	0	2.00	51.67	250.00

Any of the formats could be used in this case.

50,000 x 10 scale				
IND780 Displays:	0	200	5160	50000
Format sent:				
Integer	0	200	5160	-(15536)
Division	0	20	516	5000
Floating Point	0	200	5160	50000

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

150 x .001 scale				
IND780 Displays:	0	2.100	51.607	150.000
Format sent:				
Integer	0	2100	-(13929)	18928
Division	0	2100	-(13929)	18928
Floating Point	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32.767.

#### 5.3.4. Byte Order

**Word Swap** takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** makes the floating point format compatible with S7 PROFIBUS.

#### 5.3.5. Message Slots

There may be up to 12 message slots for discrete data transfer, Class 1 messaging, in Integer, Divisions and Floating Point Data Formats. Each message slot is assigned to a local or remote scale and scales may be repeated in additional message slots. Remote scales must reside in an IND780 Ethernet TCP clustered to the IND780 containing the Ethernet I/P interface. The integer and division formats provide (two 16-bit words of input and two 16-bit words of output data) per Message Slot. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. The following two Tables provide input and output usage information.

**Table 5-2: EtherNet / IP PLC Input Data and Data Usage (Integer and Division)**

PLC Input Bytes	PLC Input Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Status
3		Message Slot 1 Scale Status
4	2	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Status

PLC Input Bytes	PLC Input Words	Usage
7		Message Slot 2 Scale Status
8	4	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	5	Message Slot 3 Scale Status
11		Message Slot 3 Scale Status
12	6	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	7	Message Slot 4 Scale Status
15		Message Slot 4 Scale Status
16	8	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data
18	9	Message Slot 5 Scale Status
19		Message Slot 5 Scale Status
20	10	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	11	Message Slot 6 Scale Status
23		Message Slot 6 Scale Status
24	12	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	13	Message Slot 7 Scale Status
27		Message Slot 7 Scale Status
28	14	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	15	Message Slot 8 Scale Status
31		Message Slot 8 Scale Status
32	16	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	17	Message Slot 9 Scale Status
35		Message Slot 9 Scale Status
36	18	Message Slot 10 Weight Data
37		Message Slot 10 Weight Data
38	19	Message Slot 10 Scale Status
39		Message Slot 10 Scale Status
40	20	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	21	Message Slot 11 Scale Status
43		Message Slot 11 Scale Status
44	22	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	23	Message Slot 12 Scale Status
47		Message Slot 12 Scale Status

**Table 5-3: EtherNet / IP PLC Output Words and Word Usage (Integer and Division)**

PLC Output Bytes	PLC Output Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Command
3		Message Slot 1 Scale Command
4	2	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Command
7		Message Slot 2 Scale Command
8	4	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	5	Message Slot 3 Scale Command
11		Message Slot 3 Scale Command
12	6	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	7	Message Slot 4 Scale Command
15		Message Slot 4 Scale Command
16	8	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data
18	9	Message Slot 5 Scale Command
19		Message Slot 5 Scale Command
20	10	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	11	Message Slot 6 Scale Command
23		Message Slot 6 Scale Command
24	12	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	13	Message Slot 7 Scale Command
27		Message Slot 7 Scale Command
28	14	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	15	Message Slot 8 Scale Command
31		Message Slot 8 Scale Command
32	16	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	17	Message Slot 9 Scale Command
35		Message Slot 9 Scale Command
36	18	Message Slot 10 Weight Data
37		Message Slot 10 Weight Data
38	19	Message Slot 10 Scale Command
39		Message Slot 10 Scale Command

PLC Output Bytes	PLC Output Words	Usage
40	20	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	21	Message Slot 11 Scale Command
43		Message Slot 11 Scale Command
44	22	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	23	Message Slot 12 Scale Command
47		Message Slot 12 Scale Command

The floating point format provides four 16-bit words of input data and three 16-bit words of output data) per Message Slot. See Table 5-4 and Table 5-5 for details.

The number of Message Slots is configured in setup at **Communication > PLC Interface > Data Format** (Figure 5-9).

**Table 5-4: EtherNet / IP PLC Floating Point Input Words**

PLC Output Words	Bits 0 – 7	Bits 8 – 15
0	Message Slot 1, Reserved	Message Slot 1, Command Response. Refer to Message Slot Table
1	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
2	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
3	Message Slot 1, Scale Status. Refer to Message Slot Table	Message Slot 1, Scale Status. Refer to Message Slot Table
4	Message Slot 2, Reserved	Message Slot 2, Command Response. Refer to Message Slot Table
5	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
6	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
7	Message Slot 2, Scale Status. Refer to Message Slot Table	Message Slot 2, Scale Status. Refer to Message Slot Table
8	Message Slot 3, Reserved	Message Slot 3, Command Response. Refer to Message Slot Table
9	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
10	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
11	Message Slot 3, Scale Status. Refer to Message Slot Table	Message Slot 3, Scale Status. Refer to Message Slot Table
12	Message Slot 4, Reserved	Message Slot 4, Command Response. Refer to Message Slot Table
13	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
14	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
15	Message Slot 4, Scale Status. Refer to Message Slot Table	Message Slot 4, Scale Status. Refer to Message Slot Table
16	Message Slot 5, Reserved	Message Slot 5, Command Response. Refer to Message Slot Table
17	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data

<b>PLC Output Words</b>	<b>Bits 0 – 7</b>	<b>Bits 8 – 15</b>
18	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
19	Message Slot 5, Scale Status. Refer to Message Slot Table	Message Slot 5, Scale Status. Refer to Message Slot Table
20	Message Slot 6, Reserved	Message Slot 6, Command Response. Refer to Message Slot Table
21	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
22	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
23	Message Slot 6, Scale Status. Refer to Message Slot Table	Message Slot 6, Scale Status. Refer to Message Slot Table
24	Message Slot 7, Reserved	Message Slot 7, Command Response. Refer to Message Slot Table
25	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
26	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
27	Message Slot 7, Scale Status. Refer to Message Slot Table	Message Slot 7, Scale Status. Refer to Message Slot Table
28	Message Slot 8, Reserved	Message Slot 8, Command Response. Refer to Message Slot Table
29	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
30	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
31	Message Slot 8, Scale Status. Refer to Message Slot Table	Message Slot 8, Scale Status. Refer to Message Slot Table
32	Message Slot 9, Reserved	Message Slot 9, Command Response. Refer to Message Slot Table
33	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
34	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
35	Message Slot 9, Scale Status. Refer to Message Slot Table	Message Slot 9, Scale Status. Refer to Message Slot Table
36	Message Slot 10, Reserved	Message Slot 10, Command Response. Refer to Message Slot Table
37	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
38	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
39	Message Slot 10, Scale Status. Refer to Message Slot Table	Message Slot 10, Scale Status. Refer to Message Slot Table
40	Message Slot 11, Reserved	Message Slot 11, Command Response. Refer to Message Slot Table
41	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
42	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
43	Message Slot 11, Scale Status. Refer to Message Slot Table	Message Slot 11, Scale Status. Refer to Message Slot Table
44	Message Slot 12, Reserved	Message Slot 12, Command Response. Refer to Message Slot Table
45	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
46	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
47	Message Slot 12, Scale Status. Refer to Message Slot Table	Message Slot 12, Scale Status. Refer to Message Slot Table

**Table 5-5: EtherNet / IP PLC Floating Point Output Words**

PLC Output Bytes	PLC Output Words	Usage
0	0	Reserved
1		Reserved
2	1	Message Slot 1 Command
3		Message Slot 1 Command
4	2	Message Slot 1 Floating Point data
5		Message Slot 1 Floating Point data
6	3	Message Slot 1 Floating Point data
7		Message Slot 1 Floating Point data
8	4	Message Slot 2 Command
9		Message Slot 2 Command
10	5	Message Slot 2 Floating Point data
11		Message Slot 2 Floating Point data
12	6	Message Slot 2 Floating Point data
13		Message Slot 2 Floating Point data
14	7	Message Slot 3 Command
15		Message Slot 3 Command
16	8	Message Slot 3 Floating Point data
17		Message Slot 3 Floating Point data
18	9	Message Slot 3 Floating Point data
19		Message Slot 3 Floating Point data
20	10	Message Slot 4 Command
21		Message Slot 4 Command
22	11	Message Slot 4 Floating Point data
23		Message Slot 4 Floating Point data
24	12	Message Slot 4 Floating Point data
25		Message Slot 4 Floating Point data
26	13	Message Slot 5 Command
27		Message Slot 5 Command
28	14	Message Slot 5 Floating Point data
29		Message Slot 5 Floating Point data
30	15	Message Slot 5 Floating Point data
31		Message Slot 5 Floating Point data
32	16	Message Slot 6 Command
33		Message Slot 6 Command
34	17	Message Slot 6 Floating Point data
35		Message Slot 6 Floating Point data
36	18	Message Slot 6 Floating Point data
37		Message Slot 6 Floating Point data
38	19	Message Slot 7 Command
39		Message Slot 7 Command

PLC Output Bytes	PLC Output Words	Usage
40	20	Message Slot 7 Floating Point data
41		Message Slot 7 Floating Point data
42	21	Message Slot 7 Floating Point data
43		Message Slot 7 Floating Point data
44	22	Message Slot 8 Command
45		Message Slot 8 Command
46	23	Message Slot 8 Floating Point data
47		Message Slot 8 Floating Point data
48	24	Message Slot 8 Floating Point data
49		Message Slot 8 Floating Point data
50	25	Message Slot 9 Command
51		Message Slot 9 Command
52	26	Message Slot 9 Floating Point data
53		Message Slot 9 Floating Point data
54	27	Message Slot 9 Floating Point data
55		Message Slot 9 Floating Point data
56	28	Message Slot 10 Command
57		Message Slot 10 Command
58	29	Message Slot 10 Floating Point data
59		Message Slot 10 Floating Point data
60	30	Message Slot 10 Floating Point data
61		Message Slot 10 Floating Point data
62	31	Message Slot 11 Command
63		Message Slot 11 Command
64	32	Message Slot 11 Floating Point data
65		Message Slot 11 Floating Point data
66	33	Message Slot 11 Floating Point data
67		Message Slot 11 Floating Point data
68	34	Message Slot 12 Command
69		Message Slot 12 Command
70	35	Message Slot 12 Floating Point data
71		Message Slot 12 Floating Point data
72	36	Message Slot 12 Floating Point data
73		Message Slot 12 Floating Point data

In the template data format a message slot is a shared data variable. The number of message slots is limited to 496 bytes (I/O).

### 5.3.6. Integer and Division

When one of these formats is selected, the IND780 will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC's input data will contain one 16-bit word for the scale's weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC's output data. The PLC's output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 5-6 and Table 5-7 provide detailed information on the integer and division data formats. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 5-6: Discrete Read Integer or Division – IND780 >> PLC, per Message Slot**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

#### Notes for Table 5-6:

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
- 5 **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).

- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 5-7: Discrete Write Integer or Division –PLC >> IND780, per Message Slot**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

**Notes for Table 5-7 :**

- WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 and 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- A transition from "0" to "1" initiates a CLEAR command.
- A transition from "0" to "1" initiates a TARE command.
- A transition from "0" to "1" initiates a PRINT command.
- A transition from "0" to "1" initiates a ZERO command.
- If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is

written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.

- 10 **Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- 11 When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

### 5.3.7. Floating Point

#### 5.3.7.1. Operational Overview

The IND780 uses integer commands from the PLC to select the floating point weight input data. The IND780 recognizes a command when it sees a new value in the Message Slot command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the IND780 recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. The IND780 also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC should wait until it receives the command acknowledgment from the IND780 before sending another command.

The IND780 can report two types of values to the PLC: real-time and static. When the PLC requests a real-time value, the IND780 acknowledges the command from the PLC once but sends and updates the value at every interface update cycle. If the PLC requests a static value, the IND780 acknowledges the command from the PLC once and updates the value once. The IND780 will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, feed, and tolerance values are examples of static data.

The IND780 can send a rotation of up to nine different real-time values. The PLC sends commands to the IND780 to add a value to the rotation. Once the rotation is established, the PLC must instruct the IND780 to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the IND780 to advance to the next value. If the IND780 is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next interface update cycle. (The interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command, the IND780 switches to the next value in the rotation order. The IND780 stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point command examples in Table 5-12 through Table 5-15 for additional information. The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the order entered in Data Format setup.

Table 5-8 through Table 5-11 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 5-8: Discrete Read Floating Point – IND780 >> PLC Input, per Message Slot**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmnd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>
17	Cmnd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

**Notes for Table 5-8:**

- The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- Bit 0** and **Bit 1** are used only in material transfer mode.
- Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- Bit 6** and **Bit 7** are user defined, and corresponds to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- Bit 9, Bit 10, and Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- Bit 12** is set to a "1" when the scale is in motion (unstable).
- The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.

14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 5-9: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency
12	Notch filter frequency
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size
19–28	Not used
29	Last IND780 error code
30	No data response – command successful
31	No data response – command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* SDV means Shared Data Variable.

**Table 5-10: Discrete Write Floating Point – PLC >> IND780, per Message Slot**

Octal Address	WORD 0 <sup>0</sup>	WORD 1	Word 2 <sup>1</sup> FP load value	Word 3 <sup>1</sup> FP load value	Bit Number
0	Reserved	Command Word <sup>2</sup>	X	X	0
1			X	X	1
2			X	X	2
3			X	X	3
4			X	X	4

Octal Address	WORD 0 <sup>0</sup>	WORD 1	Word 2 <sup>1</sup> FP load value	Word 3 <sup>1</sup> FP load value	Bit Number
5			X	X	5
6			X	X	6
7			X	X	7
10			X	X	8
11			X	X	9
12			X	X	10
13			X	X	11
15			X	X	12
16			X	X	13
17			X	X	14

**Notes for Table 5-10:**

- 0 WORD 0 is a reserved 16 bit word, only present in message slot 1.
- 1 The Bits in WORD 2 and WORD 3 are a single-precision floating point value. This value is used with the command in WORD 0 to instruct the terminal to download the floating point value into the field specified in the command.
- 2 The command Word 1 is used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 4-11, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.

Not all commands will require a value in the floating point load value words.

**Table 5-11: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName
0	00	Report next rotation field @ next interface update cycle <sup>1</sup>	
1	01	Report next rotation field <sup>1,2</sup>	
2	02	Report next rotation field <sup>1,2</sup>	
3	03	Reset rotation	
10	0a	Report gross weight <sup>1,3</sup>	
11	0b	Report net weight <sup>1,3</sup>	
12	0c	Report tare weight <sup>1,3</sup>	
13	0d	Report fine gross weight <sup>1,3</sup>	
14	0e	Report fine net weight <sup>1,3</sup>	
15	0f	Report fine tare weight <sup>1,3</sup>	
16	10	Report Rate <sup>1,3</sup>	
17	11	Report SDV ALxx01 <sup>6</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>	
20	14	Report notch filter frequency <sup>3</sup>	
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12
25	19	Report spill value <sup>3,10</sup>	SPxx09
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>
29	1d	Report last error <sup>3</sup>	

Dec	Hex	Command	SDName
76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
78	4e	Disable LCD display <sup>7</sup>	
79	4f	Enable LCD display <sup>7</sup>	
80	50	Set normal display mode <sup>7,9</sup>	PD0119
81	51	Display message 1 <sup>7,9</sup>	PD0119
82	52	Display message 2 <sup>7,9</sup>	PD0119
83	53	Display message 3 <sup>7,9</sup>	PD0119
84	54	Display message 4 <sup>7,9</sup>	PD0119
85	55	Display Message 5 <sup>7,9</sup>	PD0119
86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
88	58	Disable weight display <sup>7</sup>	
89	59	Enable weight display <sup>7</sup>	
90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
104	68	Master control relay OFF <sup>7, 12</sup>	
105	69	Master control relay ON <sup>7, 12</sup>	

Dec	Hex	Command	SDName
30	1e	Report primary increment size <sup>3</sup>	
40	28	Add gross weight to rotation <sup>7</sup>	
41	29	Add net weight to rotation <sup>7</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>	
46	2e	Add rate to rotation <sup>7</sup>	
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>
60	3c	Load numeric tare value <sup>4</sup>	
61	3d	Pushbutton tare command <sup>7</sup>	
62	3e	Clear command <sup>7</sup>	
63	3f	Print command <sup>7</sup>	
64	40	Zero command <sup>7</sup>	
65	41	Select scale 1 <sup>7</sup>	
66	42	Select scale 2 <sup>7</sup>	
67	43	Select next scale <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>
73	49	Set low-pass filter corner frequency <sup>4</sup>	
74	4a	Set notch filter frequency <sup>4</sup>	
75	4b	Reset ENTER key <sup>7</sup>	

Dec	Hex	Command	SDName
110	6e	Set target value <sup>4,10,13</sup>	SPxx05
111	6f	Set target fine feed value <sup>4,10</sup>	
112	70	Set - tolerance value <sup>4,10</sup>	
113	71	Set target value and apply <sup>4,10,14</sup>	
114	72	Start target logic <sup>7,10,15</sup>	
115	73	Abort target logic <sup>7,10,15</sup>	
116	74	Target use gross weight <sup>7,10</sup>	SPxx04
117	75	Target use net weight <sup>7,10</sup>	SPxx04
118	76	Target use rate <sup>7,10</sup>	SPxx04
119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
121	79	Enable target latching <sup>7,10</sup>	SPxx06
122	7a	Disable target latching <sup>7,10</sup>	SPxx06
123	7b	Reset target latch <sup>7</sup>	SPxx07
124	7c	Set target spill value <sup>4,10</sup>	
131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
160	a0	Apply scale setup <sup>7</sup>	QC0149
161	a1	Write Calibration to EEPROM <sup>7</sup>	
162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
163	a3	Enable Tare on IND780 console <sup>7</sup>	
164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
168	a8	Select scale 3 <sup>7</sup>	
169	a9	Select scale 4 <sup>7</sup>	

### Notes for Table 5-11:

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.

13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

### 5.3.7.2. Floating Point Data Format and Compatibility

In Floating Point Data Format, the PLC and IND780 terminal exchange weight, target, and tare data in single-precision floating-point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

There are two data integrity bits that the IND780 uses to maintain data integrity when communicating with the PLC. One bit is in the beginning word of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC, if the PLC program detects this condition, it should send a new command to the terminal.

The Allen-Bradley SLC PLC programs do not have to make special checks to guarantee the validity of the floating-point data.

The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the word order set up by the user.

### 5.3.7.3. Floating Point Command Examples

Table 5-12 through Table 5-15 provide floating point command examples.

**Table 5-12: Data Requirement: Only Net Weight Sent (continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word 0	none required		
2 (IND780 terminal sees new command)			Command ack. =1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word, the IND780 terminal will update the net value every interface update cycle.				

**Table 5-13: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2 (PLC sends command to set target 1 cutoff value)	110 (dec) loaded into command word 0	floating point value = 21.75		
3 (IND780 terminal sees new command, loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word 0			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0

The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.

**Table 5-14: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack.= 1 F.P. ind = 30	0.0
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.				
7 (PLC sends the command to begin the rotation at interface update cycle)	0 (dec) loaded into command word 0			
8 (IND780 terminal sends gross weight at interface update cycle ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value at next interface update cycle)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 5-15: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	0.0
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0	RESERVED for Future Use		

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0
At this point, the rotation has been set up. Now the PLC needs to send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word 0			
8 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net weight
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word 0			
10 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
11 (PLC sends the command to report the next field in the rotation.)	1 (dec) loaded into command word 0			
12 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
13 (PLC sends the command to report the next field.)	2 (dec) loaded into command word 0			

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
14 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value				

#### 5.3.7.4. Shared Data Mode

The Shared Data mode PLC communications is provided using CIP explicit messages.

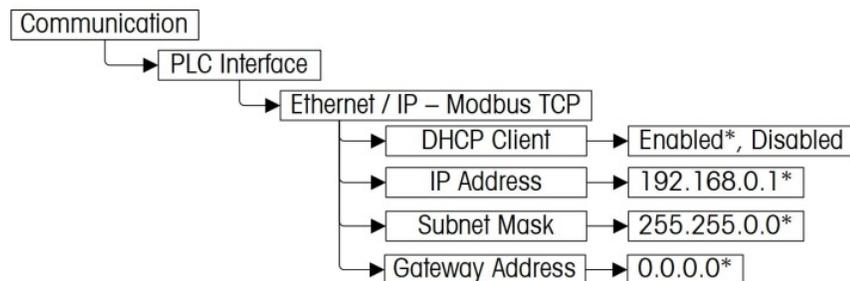
The **IND780 Shared Data Reference** lists the Shared Data Variables available to EtherNet/IP. This document also includes the hex Class Code, Instance and Attribute for the shared data. The PLC must use Get Attribute Single (0e hex) to read a Shared Data Variable and Set Attribute Single (10 hex) to write a Shared Data Variable.

#### 5.3.8. Controlling the Discrete I/O Using a PLC Interface

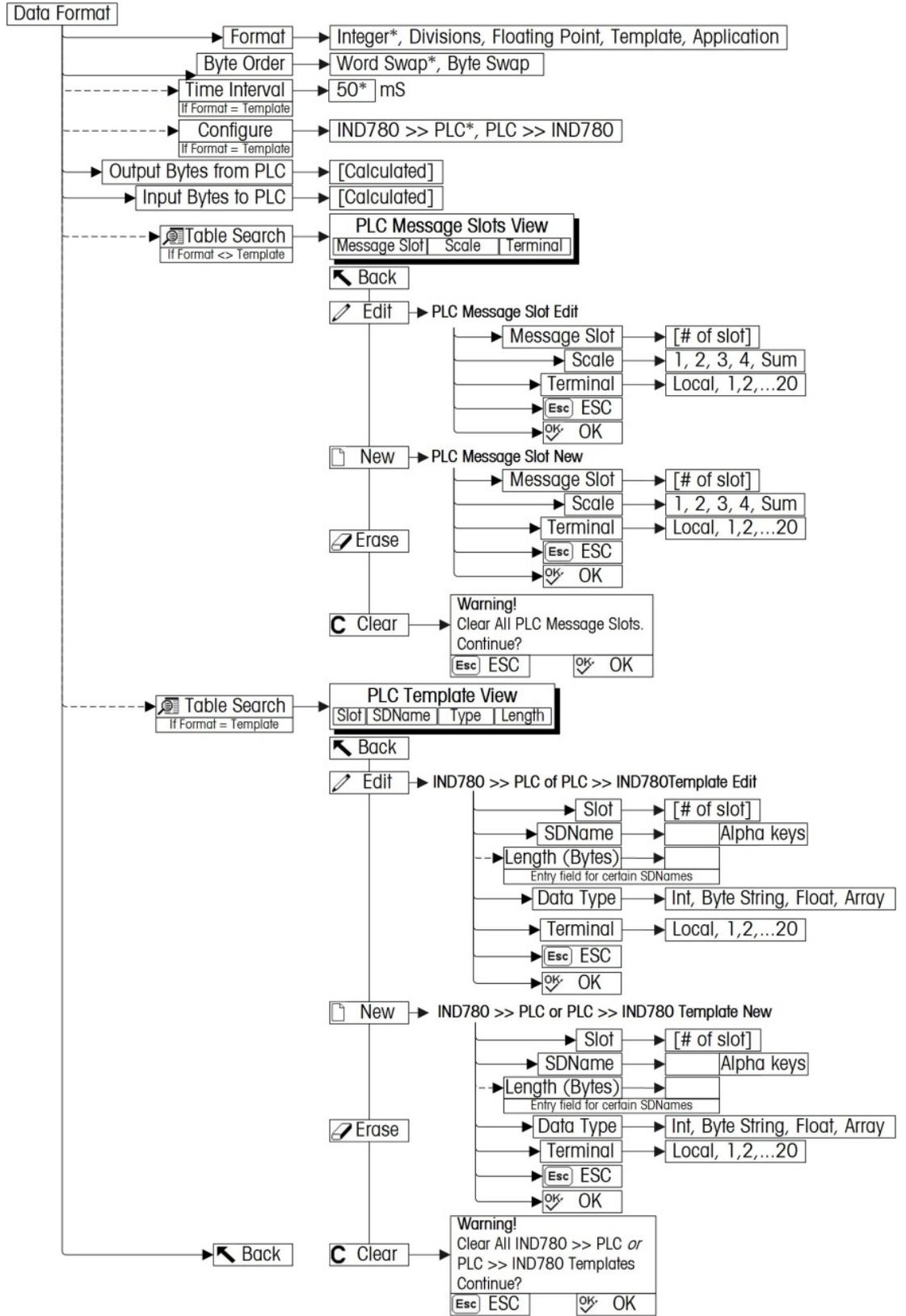
The IND780 terminal provides the ability to directly control its discrete outputs and read its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the IND780 terminal's discrete I/O updates are synchronized with the terminal's interface update rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the **IND780 Terminal Technical Manual** or **Installation Manual** for discrete I/O wiring.

## 5.4. Software Setup

When the IND780 terminal detects the presence of a EtherNet / IP Kit option board, the EtherNet / IP parameters are enabled in a Setup program block at Communication > PLC Interface > EtherNet / IP. Figure 5-6 shows the EtherNet / IP setup block, and Figure 5-7 the **Communication > PLC Interface > Data Format** setup block. Default values are marked with an asterisk.



**Figure 5-6: EtherNet / IP Setup Block**



**Figure 5-7: PLC Data Format Setup Block**

## 5.4.1. EtherNet / IP Setup Block

### 5.4.1.1. Ethernet / IP Setup

The EtherNet / IP Setup block lets you specify IP address of the EtherNet / IP interface. In setup, select **Communication > PLC Interface > EtherNet / IP** and set the IP Address, Subnet Mask and Gateway Address (Figure 5-8). Alternatively, DHCP Client can be enabled if the IP address is to be assigned automatically by the network server.

IP=172.18.54.80		15/Oct/2008 08:27	
<b>EtherNet/IP - Modbus/TCP</b>			
MAC Address	ETHIP 00301102eab7		
DHCP Client	Disabled		
IP Address	192	168	000.001
Subnet Mask	255	255	255.000
Gateway Address	000	000	000.000
←			

Figure 5-8: EtherNet IP Setup Screen

### 5.4.1.2. Data Format Setup

1. In setup, select **Communication > PLC Interface > Data Format**. Figure 5-9 shows the screen with Template selected as the format, revealing the Time Interval, Configure, and directional displays. When Application format is selected, only the Format box appears on this screen.

IP=192.168.0.1		07/Mar/2006 11:07	
<b>PLC Data Format</b>			
Format	Template		
Byte Order	Word Swap		
Time Interval	50	mS	
Configure	IND780 >> PLC		
PLC >> IND780	4	bytes	
IND780 >> PLC	112	bytes	
←			

Figure 5-9: PLC Data Format Screen

2. Select the Format (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the Format will delete any existing Message Slots.

**Integer** Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).

**Divisions** Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display

units.

- Floating Point** Displays weight in floating point data format
- Template** Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed “on the fly” by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the **Shared Data Reference** (document number 64059110) provided on the CD-ROM supplied with your terminal.
- Application** Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the **TaskExpert Reference Manual** (document number 64060431).

3. Select Byte Order: [default Word Swap]

- Word Swap** Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.
- Byte Swap** Makes the floating point format compatible with S7 PROFIBUS.

- 4. Set the **Time Interval** in milliseconds. (Only when the Format is Template)
- 5. Use the **Configure** drop-down list to select the direction of data flow – IND780 >> PLC or PLC >> IND780. (Only when the Format is Template)
- 6. The bottom of this screen (Figure 5-9) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.

7. Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of Message Slots (maximum number is 12; default is 0), or to configure the Template for communications between the IND780 and the PLC. Figure 5-10 shows the Message Slots View, which is viewed whenever the Format is not Template.

IP=172.18.54.111		06/Dec/2007 16:25	
PLC Message Slots View			
Message	Scale	Terminal	
▶ 1	1	Local	
2	- End -		
			 <b>C</b>

**Figure 5-10: PLC Message Slots View Screen**

Figure 5-11 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1		07/Mar/2006 11:08	
PLC >> IND780 Template View			
Slot	SDName	Type	Length
▶ 1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		
			 <b>C</b>

**Figure 5-11: PLC Template View**

Press the EXIT softkey  to exit the view screen.

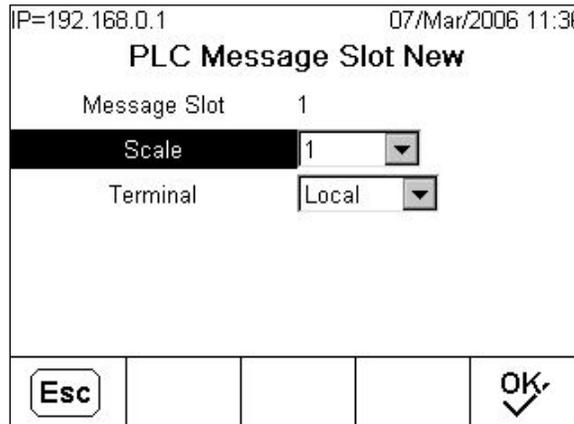
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey **C** to delete all existing slots.

Figure 5-12 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 5-10).

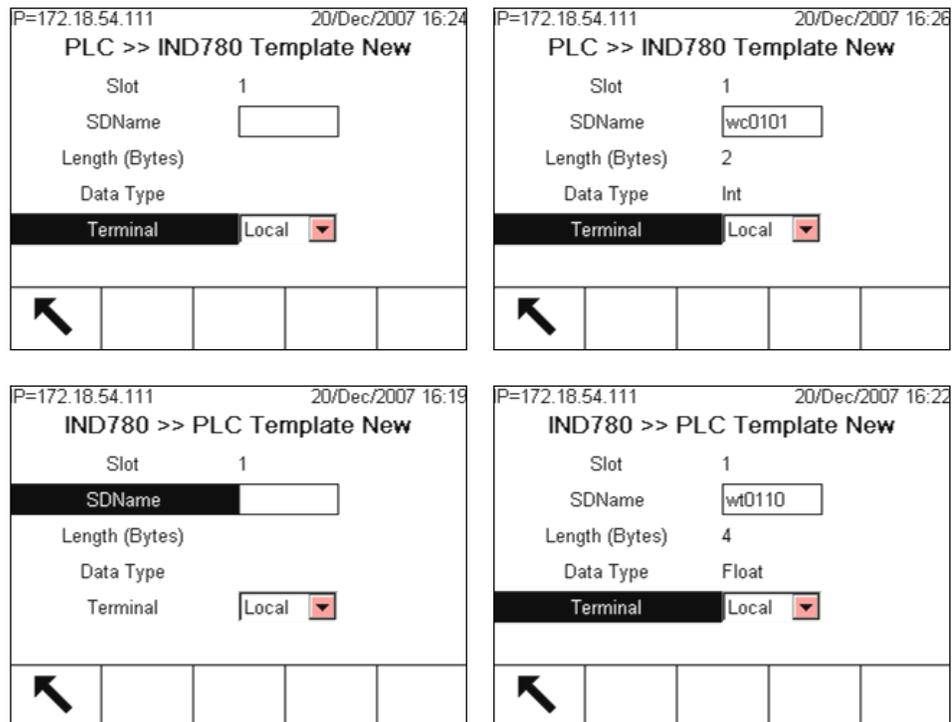


**Figure 5-12: New Message Slot Screen**

**Scale** Enter the scale number to be associated with the new Message Slot.

**Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 5-13 shows a series of Template New screens, as accessed from the Template View screen Figure 5-11.



**Figure 5-13: New Template Screens, PLC to IND780 and IND780 to PLC**

**Slot** Automatically assigned.

**SDName** An alphabetic entry field used to define the Shared Data variable associated

- with the slot.
- Length (Bytes) and Data Type** Automatically displayed once the SDName has been entered.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

For a complete listing of Shared Data Fields, refer to the IND780 Shared Data Reference. Refer to Table 5-6 (Discrete Read) and Table 5-7 (Discrete Write) in this chapter for additional information on mapping of discrete read data to the PLC.

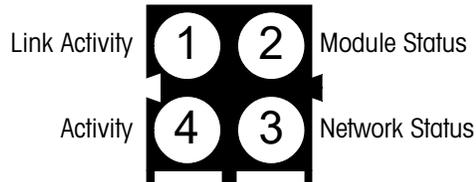
## 5.5. Troubleshooting

If the IND780 does not communicate with PLC, do the following:

- Check wiring and network termination.
- Confirm that the IND780 settings for data type and IP Address assignment match those in the PLC and that each IND780 has a unique address.
- Replace the EtherNet / IP interface kit if the problem persists.
- If the communication kit was changed from another type, like PROFIBUS, ControlNet or Remote I/O, a master reset of the IND780 must be performed.

### 5.5.1. Status LEDs

The EtherNet/ IP interface card has four status LEDs indicators to indicate communication and fault status of the card. Figure 4-2 indicates the location of these LEDs, and Figure 5-14 shows the array of the LEDs on the card. Table 5-16 explains the meaning if the indicators.



**Figure 5-14: Ethernet / IP Status Indicator LEDs**

**Table 5-16: Ethernet/ IP LEDs Status Indications**

LED #	State	Status
1 - Link Activity	Off	No link (or no power)
	Green	Connected to an Ethernet network
2 - Module Status	Off	No power
	Green	Normal operation
	Green, flashing	Stand by, not initialized
	Red	Major fault
	Red, flashing	Minor fault
	Alternating Red/Green	Self test
3 - Network Status	Off	No IP address (or no power)

	Green	EtherNet/ IP connection(s) established
	Green, flashing	No EtherNet/ IP connections established
	Red	Duplicate IP address detected
	Red, flashing	One or several connections timed out
	Alternating Red/Green	Self test
4 - Activity	Off	No Ethernet activity (or no power)
	Green	Receiving or transmitting Ethernet packet

## 5.6. EtherNet / IP Option Kit

No spare parts are associated with the EtherNet / IP option kit. The kit CIMF part number is 64058677. Table 5-17 shows what comes in the kit.

**Table 5-17: EtherNet / IP Option Kit**

Description	Qty.
Installation Instructions	1
PCB package	1
Clamp-on ferrite	1
M3 x 8 screws	3

## 5.7. Programming Examples

The following Figures show sample screen images of ladder logic programming examples for RSLogix 5000 software (version 13).

- **Note:** The Utilities folder of the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.

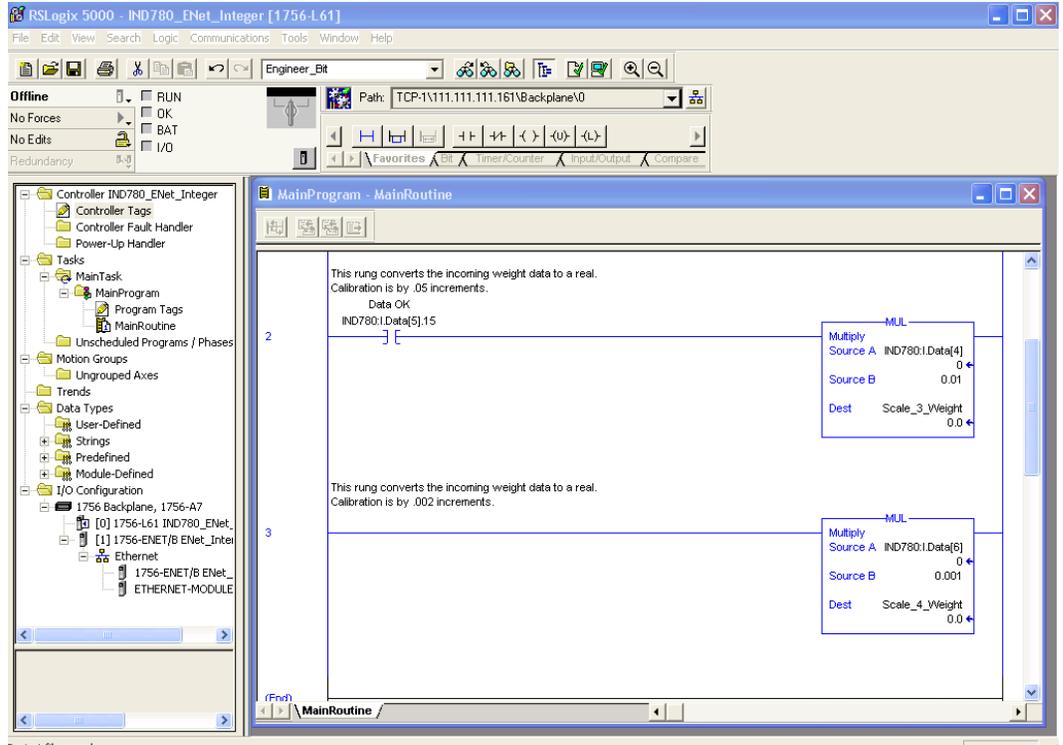
The following RSLogix 5000 screens for Integer, Division and Floating Point data formats only show an example of a particular Input and Output size configuration. The Connection Parameters I/O sizes must be appropriately configured with reference to the number of slots assigned in the IND780 PLC Data Format Message Slots settings. Table 5-18 and Table 5-19 show the relationship between the IND780 message slots and the RSLogix 5000 I/O sizing for Integer, Division and Floating Point data formats.

**Table 5-18: Message Slot and PLC I/O Sizes (Integer/ Division)**

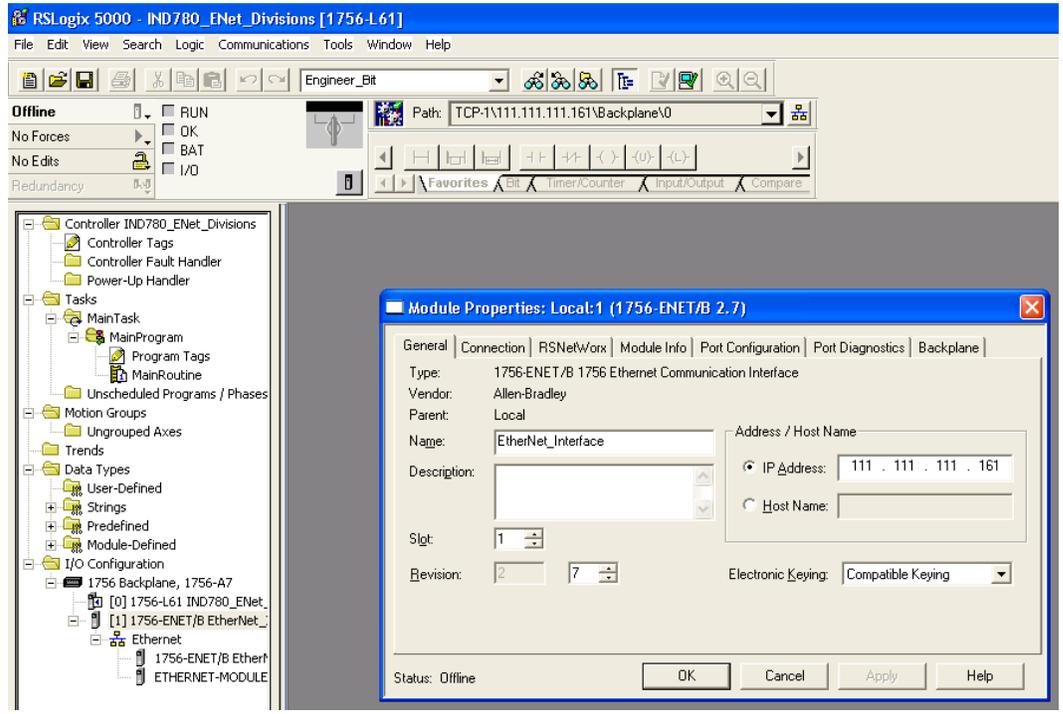
IND780 Integer/ Division Data			RSLogix 5000 Comm Format	
Message Slots	Bytes (8 Bit)		INT (16 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780	Input	Output
1	4	4	2	2
2	8	8	4	4
3	12	12	6	6
4	16	16	8	8
5	20	20	10	10
6	24	24	12	12
7	28	28	14	14
8	32	32	16	16
9	36	36	18	18
10	40	40	20	20
11	44	44	22	22
12	48	48	24	24

**Table 5-19: Message Slot and PLC I/O Sizes (Floating Point)**

IND780 Floating Point Data			RSLogix 5000 Comm Format	
Message Slots	Bytes (8 Bit)		INT (16 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780	Input	Output
1	8	8	4	4
2	16	14	8	7
3	24	20	12	10
4	32	26	16	13
5	40	32	20	16
6	48	38	24	19
7	56	44	28	22
8	64	50	32	25
9	72	56	36	28
10	80	62	40	31
11	88	68	44	34
12	96	74	48	37



**Figure 5-15: Integer Weight Data to Real**



**Figure 5-16: EtherNet Bridge in PLC Setup**

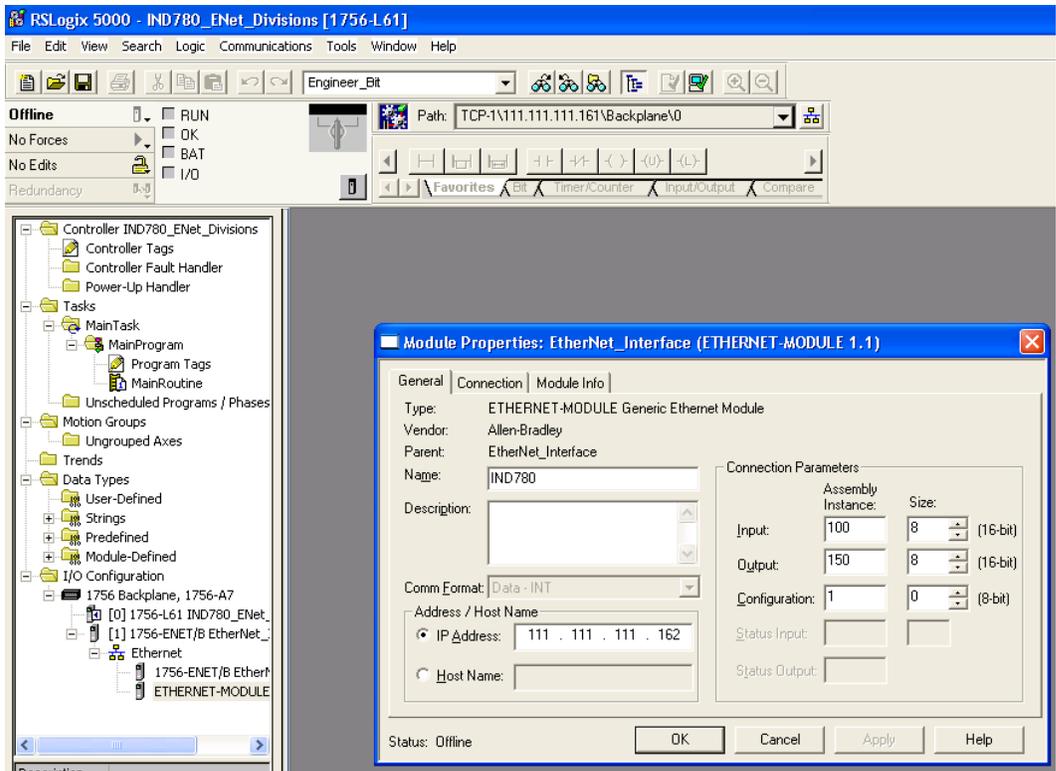


Figure 5-17: EtherNet Module, Divisions

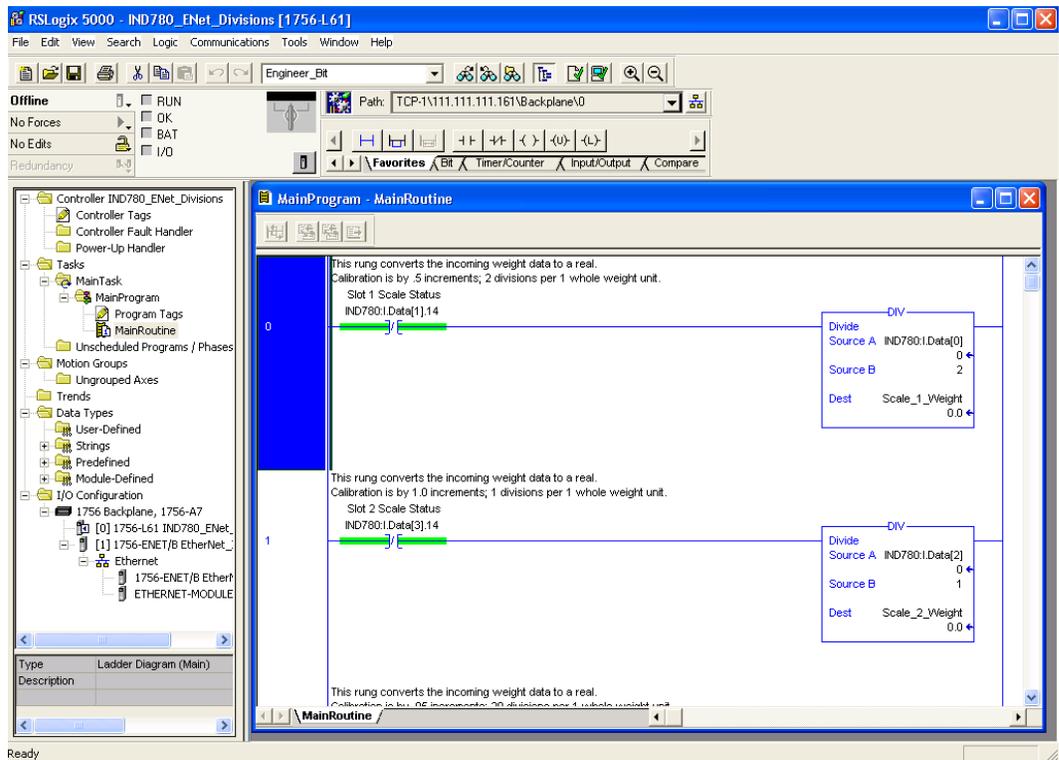
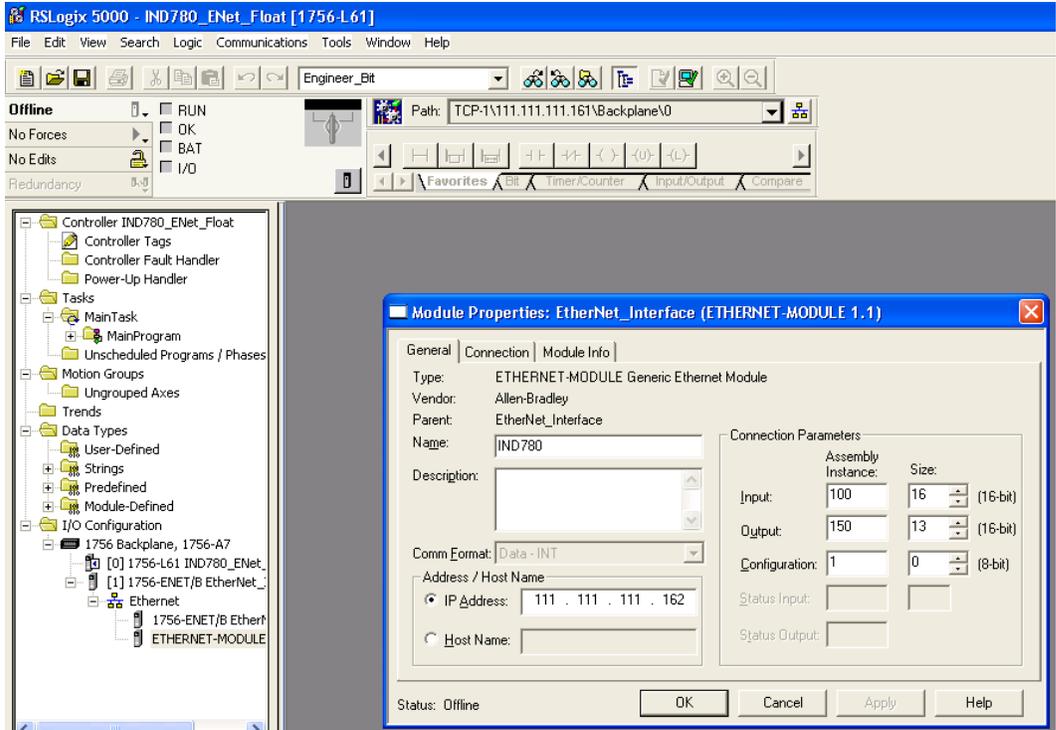
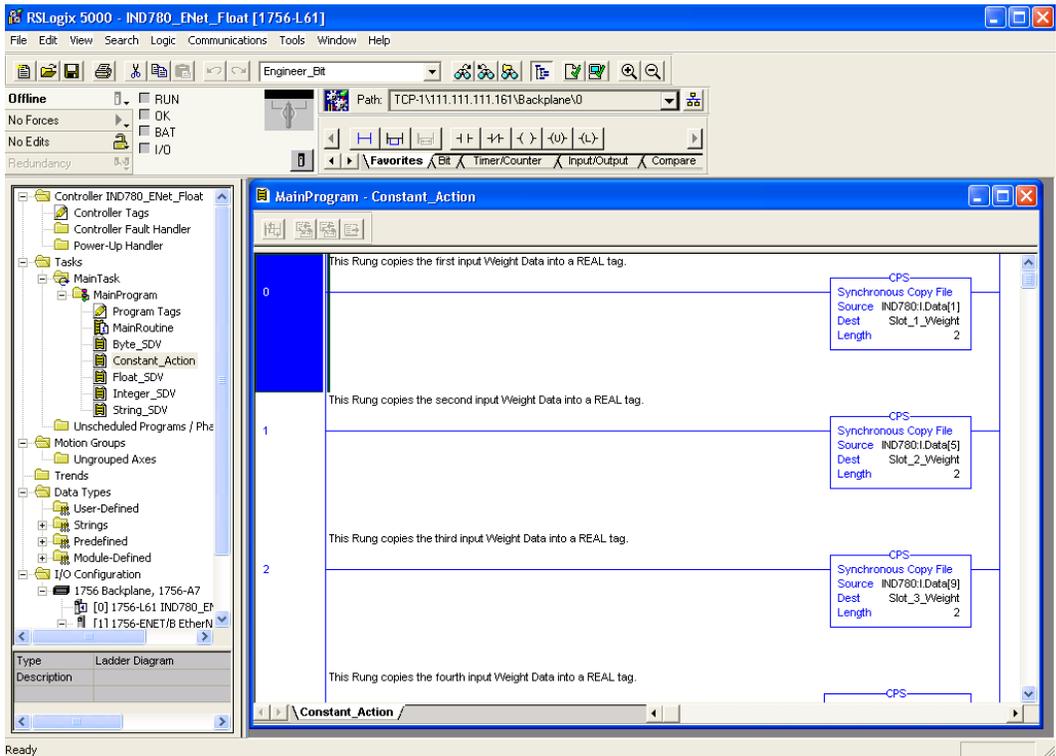


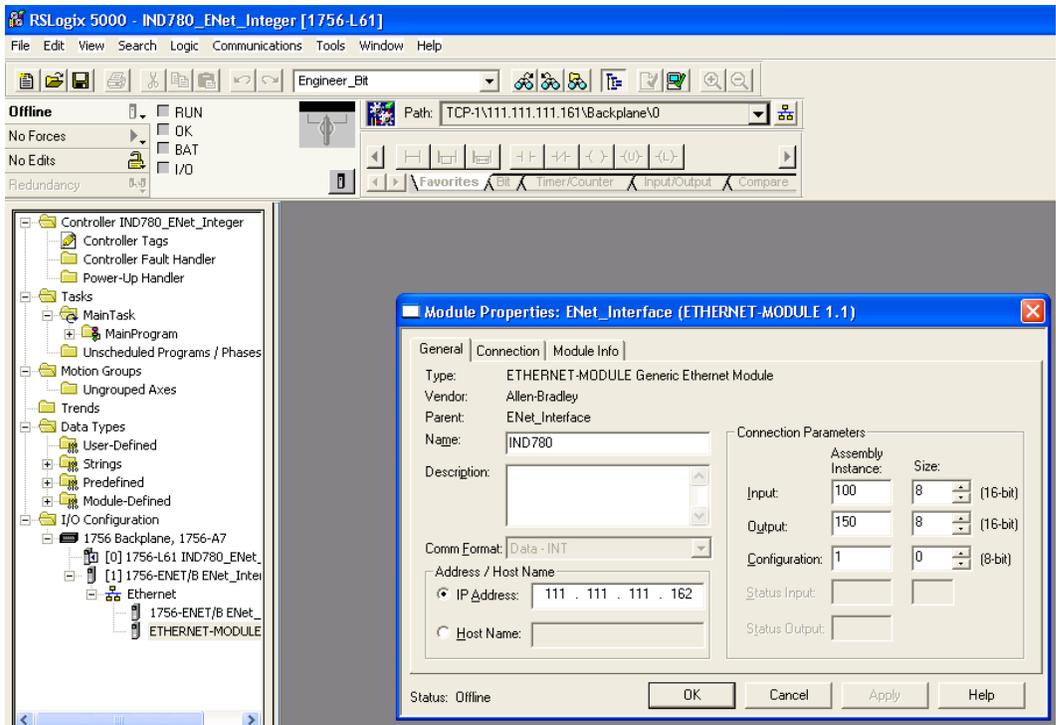
Figure 5-18: Integer Weight Data to Real, Division



**Figure 5-19: EtherNet Module, Floating Point**



**Figure 5-20: Input Weight Data to Real Tag**



**Figure 5-21: EtherNet Module, Integer**

# 6 Modbus TCP

## 6.1. Preface

Users should note that the Modbus TCP option uses the Ethernet / IP board (P/N: 64058677) described in chapter 4 of this manual.

- For use with the Modbus TCP protocol, the Ethernet / IP board must be revision 1.32 or higher.

## 6.2. Overview

Modbus protocol is a messaging structure developed by Modicon in 1979. It is used to establish master-slave/client-server communication between intelligent devices. It is an open standard network protocol, widely used in the industrial manufacturing environment. Modbus can be used in multiple master-slave applications to monitor and program devices; to communicate between intelligent devices and sensors and instruments; and to monitor field devices using PCs and HMIs. Modbus is also an ideal protocol for RTU applications where wireless communication is required.

TCP/IP is an Internet transport protocol of that consists of a set of layered protocols, providing a reliable data transport mechanism between machines.

The open Modbus TCP/IP specification was developed in 1999. The ModbusTCP protocol takes the Modbus instruction set and wraps TCP/IP around it.

### 6.2.1. Specifications

<b>Network Type</b>	Ethernet-TCP/IP based simple Client/Server network.
<b>Topology</b>	Star, tree or line structures; all topologies that can be implemented with standard Ethernet technology, including switched networks, are applicable.
<b>Installation</b>	Standard 10, 100 or 1000 Mbit/s Ethernet technology based on copper cables, fiber optic or wireless standards can be used. The IND780 Modbus TCP option provides an RJ-45 Ethernet port connection
<b>Speed</b>	10, 100, 1000 Mbit/s. The IND780 Modbus TCP option only supports 10 or 100 Mbit/s
<b>Max. stations</b>	Nearly unlimited.
<b>Data</b>	Up to 1.500 Byte per Telegram frame. Total: nearly unlimited.
<b>Network features</b>	Client/Server network based on standard Ethernet technology and TCP/UDP/IP protocols in Layer 3-4.
<b>User Organization</b>	Modbus-IDA user Group.

## 6.3. Modbus TCP Characteristics

- User-programmable IP addressing
- Capability for bi-directional discrete mode communications (Cyclic Messaging) of weight or display increments, status, and control data between the PLC and the IND780.

## 6.4. Modbus TCP Board

Figure 6-1 shows a view of the Ethernet / IP Option Board used for Modbus TCP communication. Figure 6-2 indicates the board's port, DIP switches and status lights. Note that the module's address is set in software, and the DIP switches must all be set to OFF.

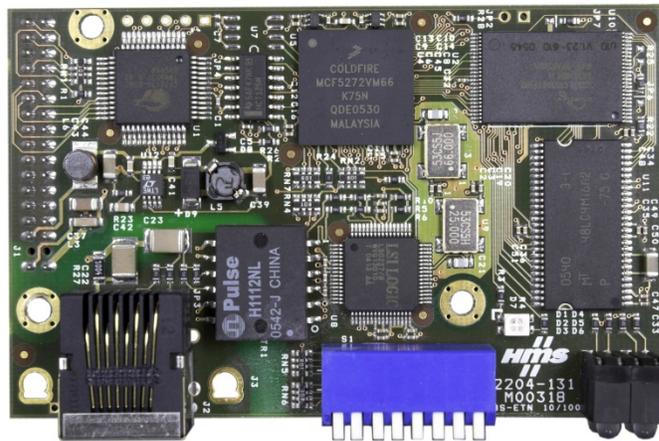


Figure 6-1: Ethernet / IP – Modbus TCP Option Board



Figure 6-2: Ethernet / IP - Modbus TCP Option Board Components

### 6.4.1. Communications

The IND780 terminal utilizes component parts to ensure complete compatibility with the Modbus TCP network. An IND780 terminal is recognized as a generic Modbus TCP device by the PLC.

Each Modbus TCP option connected to the network represents a physical IP Address. The connection is made via a RJ-45 connector on the option card (see Figure 6-2).

The wiring between the PLC and the IND780 Modbus TCP connection uses EtherNet twisted pair cable. The cable installation procedures and specification including distance and termination requirements are the same as recommended by Schneider Electric (Modicon) for the Modbus TCP network.

The IND780 uses cyclic data for discrete data messaging and the template data format for access to the IND780 Shared Data Variables. It does not support explicit or unscheduled messaging with the Modbus TCP interface.

### 6.4.2. IP Address

Each Modbus TCP interface option represents one physical IP Address. This address is chosen by the system designer, and then programmed into the IND780 terminal and PLC. There is also a DHCP client feature for the Modbus TCP interface to use a host server to assign addresses. The IND780 terminal's address is programmed in setup at **Communication > PLC Interface > Modbus TCP**. IND780 IP Address entry must be unique for each IND780.

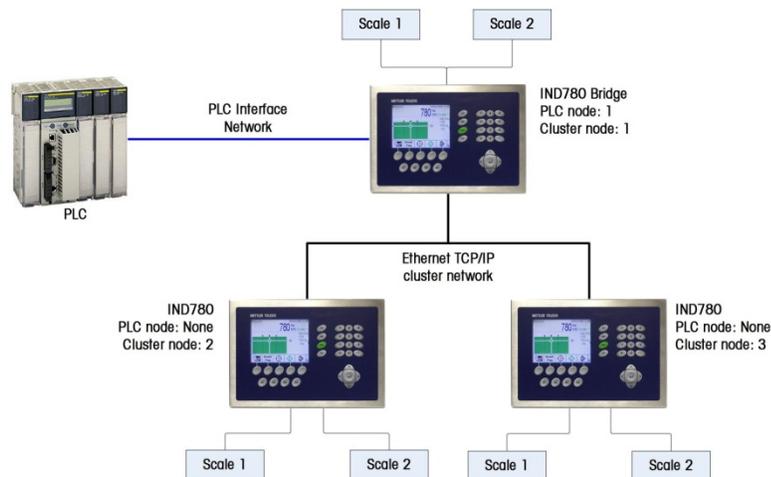
### 6.4.3. Data Formats

The Modbus TCP Kit option provides discrete data transfer, cyclic messaging. Discrete data is continuously available. The option has its own logical IP address to send and receive information to and from the PLC. There are five data formats: Integer, Divisions, Floating Point, Template and Application.

## 6.5. Sharing a PLC Interface

### 6.5.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. The following example (Figure 6-3) shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 6-3: Typical Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate PLC chapters in the IND780 PLC Interface Manual for further details on message slots' limitations.

To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure 6-4 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to the Software Setup section, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

PLC Message Slots View		
Message	Scale	Terminal
1	1	Local
2	1	6
3	- End -	

Figure 6-4: PLC Message Slots View Screen

## 6.6. Data Definition

### 6.6.1. Data Integrity

The IND780 has specific bits to allow the PLC to confirm that data was received without interruption and that the IND780 is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received by the IND780. Refer to the data charts for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage.

### 6.6.2. Discrete Data

Five formats of discrete data are available with the Modbus TCP Kit option: Integer, divisions, floating point, template and application. Only one type of data format may be selected and used by the IND780.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word (see Table 6-5 and Table 6-6 for explanation) numerical values.

The floating-point format allows bi-directional communication of discrete bit encoded information and numeric data encoded in IEEE 754, single precision floating point format.

The data format of discrete data will affect the data size required in the configuration of the PLC. The IND780 console PLC message slot setup screen provides data size requirements in bytes, except for the Application data format.

Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions or increments. The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Floating point is the only data format that includes decimal point information. Integer and division formats ignore decimal points. Accommodation of decimal point location must take place in the PLC logic, when it is needed with these formats.

Changing the Data Format to be used by the IND780 will clear all Message Slots. Data format is set up in the **Communication > PLC Interface > Data Format** screen.

6.6.2.1. Examples

<b>250 x .01 scale</b>				
IND780 Displays:	0	2.00	51.67	250.00
Format sent:				
Int	0	200	5167	25000
Div	0	200	5167	25000
FLT	0	2.00	51.67	250.00

Any of the formats could be used in this case.

<b>50,000 x 10 scale</b>				
IND780 Displays:	0	200	5160	50000
Format sent:				
Int	0	200	5160	-(15536)
Div	0	20	516	5000
FLT	0	200	5160	50000

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

<b>150 x .001 scale</b>				
IND780 Displays:	0	2.100	51.607	150.000
Format sent:				
Int	0	2100	-(13929)	18928
Div	0	2100	-(13929)	18928
FLT	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32,767.

### 6.6.3. Byte Order

**Double Word Swap** takes the IEE 754 single-precision floating point format and swaps the two words plus byte swaps in the 32-bit double word. This format makes floating point data compatible with Modicon Quantum processors.

**Byte Swap** makes the integer mode compatible with Modicon Quantum processors

**Word Swap** takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word.

### 6.6.4. Message Slots

There may be up to 12 message slots for discrete data transfer, in Integer, Divisions and Floating Point Data Formats. Each message slot is assigned to an internal local or remote scale and scales may be repeated in additional message slots. Remote scales must reside in an IND780 Ethernet TCP clustered to the IND780 containing the Modbus TCP interface. The integer and division formats provide (two 16-bit words of input and two 16-bit words of output data) per Message Slot. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. The following two Tables provide input and output usage information.

- 4000, 40001 and 400001 are PLC processor memory-dependent. Refer to PLC documentation for I/O mapping.
- Any scale can be configured as any slot.

**Table 6-1: Modbus TCP PLC Input Data and Data Usage (Integer and Division)**

PLC Input Bytes	Register Address	Usage
0	400001	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	400002	Message Slot 1 Scale Status
3		Message Slot 1 Scale Status
4	400003	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	400004	Message Slot 2 Scale Status
7		Message Slot 2 Scale Status
8	400005	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	400006	Message Slot 3 Scale Status
11		Message Slot 3 Scale Status
12	400007	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	400008	Message Slot 4 Scale Status
15		Message Slot 4 Scale Status
16	400009	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data

PLC Input Bytes	Register Address	Usage
18	400010	Message Slot 5 Scale Status
19		Message Slot 5 Scale Status
20	400011	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	400012	Message Slot 6 Scale Status
23		Message Slot 6 Scale Status
24	400013	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	400014	Message Slot 7 Scale Status
27		Message Slot 7 Scale Status
28	400015	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	400016	Message Slot 8 Scale Status
31		Message Slot 8 Scale Status
32	400017	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	400018	Message Slot 9 Scale Status
35		Message Slot 9 Scale Status
36	400019	Message Slot 10 Weight Data
37		Message Slot 10 Weight Data
38	400020	Message Slot 10 Scale Status
39		Message Slot 10 Scale Status
40	400021	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	400022	Message Slot 11 Scale Status
43		Message Slot 11 Scale Status
44	400023	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	400024	Message Slot 12 Scale Status
47		Message Slot 12 Scale Status

**Table 6-2: Modbus TCP PLC Output Words and Word Usage (Integer and Division)**

PLC Output Bytes	Register Address	Usage
0	401025	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	401026	Message Slot 1 Scale Command
3		Message Slot 1 Scale Command
4	401027	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	401028	Message Slot 2 Scale Command
7		Message Slot 2 Scale Command

PLC Output Bytes	Register Address	Usage
8	401029	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	401030	Message Slot 3 Scale Command
11		Message Slot 3 Scale Command
12	401031	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	401032	Message Slot 4 Scale Command
15		Message Slot 4 Scale Command
16	401033	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data
18	401034	Message Slot 5 Scale Command
19		Message Slot 5 Scale Command
20	401035	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	401036	Message Slot 6 Scale Command
23		Message Slot 6 Scale Command
24	401037	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	401038	Message Slot 7 Scale Command
27		Message Slot 7 Scale Command
28	401039	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	401040	Message Slot 8 Scale Command
31		Message Slot 8 Scale Command
32	401041	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	401042	Message Slot 9 Scale Command
35		Message Slot 9 Scale Command
36	401043	Message Slot 10 Weight Data
37		Message Slot 10 Weight Data
38	401044	Message Slot 10 Scale Command
39		Message Slot 10 Scale Command
40	401045	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	401046	Message Slot 11 Scale Command
43		Message Slot 11 Scale Command
44	401047	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	401048	Message Slot 12 Scale Command
47		Message Slot 12 Scale Command

The floating point format provides four 16-bit words of input data and three 16-bit words of output data per Message Slot. See Table 6-3 and Table 6-4 for details.

The number of Message Slots is configured in setup at **Communication > PLC Interface > Data Format**.

- 4000, 40001 and 400001 are PLC processor memory-dependent. Refer to PLC documentation for I/O mapping
- Any scale can be configured as any slot.

**Table 6-3: Modbus TCP PLC Floating Point Input Words**

Register Address	Bits 0 – 7	Bits 8 – 15
400001	Message Slot 1, Reserved	Message Slot 1, Command Response. Refer to Message Slot Table
400002	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
400003	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
400004	Message Slot 1, Scale Status. Refer to Message Slot Table	Message Slot 1, Scale Status. Refer to Message Slot Table
400005	Message Slot 2, Reserved	Message Slot 2, Command Response. Refer to Message Slot Table
400006	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
400007	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
400008	Message Slot 2, Scale Status. Refer to Message Slot Table	Message Slot 2, Scale Status. Refer to Message Slot Table
400009	Message Slot 3, Reserved	Message Slot 3, Command Response. Refer to Message Slot Table
400010	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
400011	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
400012	Message Slot 3, Scale Status. Refer to Message Slot Table	Message Slot 3, Scale Status. Refer to Message Slot Table
400013	Message Slot 4, Reserved	Message Slot 4, Command Response. Refer to Message Slot Table
400014	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
400015	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
400016	Message Slot 4, Scale Status. Refer to Message Slot Table	Message Slot 4, Scale Status. Refer to Message Slot Table
400017	Message Slot 5, Reserved	Message Slot 5, Command Response. Refer to Message Slot Table
400018	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
400019	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
400020	Message Slot 5, Scale Status. Refer to Message Slot Table	Message Slot 5, Scale Status. Refer to Message Slot Table
400021	Message Slot 6, Reserved	Message Slot 6, Command Response. Refer to Message Slot Table
400022	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
400023	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
400024	Message Slot 6, Scale Status. Refer to Message Slot Table	Message Slot 6, Scale Status. Refer to Message Slot Table
400025	Message Slot 7, Reserved	Message Slot 7, Command Response. Refer to Message Slot Table
400026	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data

Register Address	Bits 0 – 7	Bits 8 – 15
400027	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
400028	Message Slot 7, Scale Status. Refer to Message Slot Table	Message Slot 7, Scale Status. Refer to Message Slot Table
400029	Message Slot 8, Reserved	Message Slot 8, Command Response. Refer to Message Slot Table
400030	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
400031	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
400032	Message Slot 8, Scale Status. Refer to Message Slot Table	Message Slot 8, Scale Status. Refer to Message Slot Table
400033	Message Slot 9, Reserved	Message Slot 9, Command Response. Refer to Message Slot Table
400034	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
400035	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
400036	Message Slot 9, Scale Status. Refer to Message Slot Table	Message Slot 9, Scale Status. Refer to Message Slot Table
400037	Message Slot 10, Reserved	Message Slot 10, Command Response. Refer to Message Slot Table
400038	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
400039	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
400040	Message Slot 10, Scale Status. Refer to Message Slot Table	Message Slot 10, Scale Status. Refer to Message Slot Table
400041	Message Slot 11, Reserved	Message Slot 11, Command Response. Refer to Message Slot Table
400042	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
400043	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
400044	Message Slot 11, Scale Status. Refer to Message Slot Table	Message Slot 11, Scale Status. Refer to Message Slot Table
400045	Message Slot 12, Reserved	Message Slot 12, Command Response. Refer to Message Slot Table
400046	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
400047	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
400048	Message Slot 12, Scale Status. Refer to Message Slot Table	Message Slot 12, Scale Status. Refer to Message Slot Table

**Table 6-4: Modbus TCP PLC Floating Point Output Words**

PLC Output Bytes	Register Address	Usage
0	401025	Reserved
1		Reserved
2	401026	Message Slot 1 Command
3		Message Slot 1 Command
4	401027	Message Slot 1 Floating Point data
5		Message Slot 1 Floating Point data
6	401028	Message Slot 1 Floating Point data
7		Message Slot 1 Floating Point data
8	401029	Message Slot 2 Command

<b>PLC Output Bytes</b>	<b>Register Address</b>	<b>Usage</b>
9		Message Slot 2 Command
10	401030	Message Slot 2 Floating Point data
11		Message Slot 2 Floating Point data
12	401031	Message Slot 2 Floating Point data
13		Message Slot 2 Floating Point data
14	401032	Message Slot 3 Command
15		Message Slot 3 Command
16	401033	Message Slot 3 Floating Point data
17		Message Slot 3 Floating Point data
18	401034	Message Slot 3 Floating Point data
19		Message Slot 3 Floating Point data
20	401035	Message Slot 4 Command
21		Message Slot 4 Command
22	401036	Message Slot 4 Floating Point data
23		Message Slot 4 Floating Point data
24	401037	Message Slot 4 Floating Point data
25		Message Slot 4 Floating Point data
26	401038	Message Slot 5 Command
27		Message Slot 5 Command
28	401039	Message Slot 5 Floating Point data
29		Message Slot 5 Floating Point data
30	401040	Message Slot 5 Floating Point data
31		Message Slot 5 Floating Point data
32	401041	Message Slot 6 Command
33		Message Slot 6 Command
34	401042	Message Slot 6 Floating Point data
35		Message Slot 6 Floating Point data
36	401043	Message Slot 6 Floating Point data
37		Message Slot 6 Floating Point data
38	401044	Message Slot 7 Command
39		Message Slot 7 Command
40	401045	Message Slot 7 Floating Point data
41		Message Slot 7 Floating Point data
42	401046	Message Slot 7 Floating Point data
43		Message Slot 7 Floating Point data
44	401047	Message Slot 8 Command
45		Message Slot 8 Command
46	401048	Message Slot 8 Floating Point data
47		Message Slot 8 Floating Point data
48	401049	Message Slot 8 Floating Point data

PLC Output Bytes	Register Address	Usage
49		Message Slot 8 Floating Point data
50	401050	Message Slot 9 Command
51		Message Slot 9 Command
52	401051	Message Slot 9 Floating Point data
53		Message Slot 9 Floating Point data
54	401052	Message Slot 9 Floating Point data
55		Message Slot 9 Floating Point data
56	401053	Message Slot 10 Command
57		Message Slot 10 Command
58	401054	Message Slot 10 Floating Point data
59		Message Slot 10 Floating Point data
60	401055	Message Slot 10 Floating Point data
61		Message Slot 10 Floating Point data
62	401056	Message Slot 11 Command
63		Message Slot 11 Command
64	401057	Message Slot 11 Floating Point data
65		Message Slot 11 Floating Point data
66	401058	Message Slot 11 Floating Point data
67		Message Slot 11 Floating Point data
68	401059	Message Slot 12 Command
69		Message Slot 12 Command
70	401060	Message Slot 12 Floating Point data
71		Message Slot 12 Floating Point data
72	401061	Message Slot 12 Floating Point data
73		Message Slot 12 Floating Point data

In the template data format a message slot is a shared data variable. The number of message slots is limited to 256 bytes (I/O).

### 6.6.5. Integer and Division

When one of these formats is selected, the IND780 will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC's input data will contain one 16-bit word for the scale's weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC's output data. The PLC's output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 6-5 and Table 6-6 provide detailed information on the integer and division data formats. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 6-5: Discrete Read Integer or Division – IND780 >> PLC, per Message Slot**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

#### Notes for Table 6-5

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
- 5 **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).

- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 6-6: Discrete Write Integer or Division – PLC >> IND780, per Message Slot**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

**Notes for Table 6-6**

- WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 and 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- A transition from "0" to "1" initiates a CLEAR command.
- A transition from "0" to "1" initiates a TARE command.
- A transition from "0" to "1" initiates a PRINT command.
- A transition from "0" to "1" initiates a ZERO command.
- If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is

written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.

10 **Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.

11 When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

## 6.6.6. Floating Point

### 6.6.6.1. Operational Overview

The terminal uses integer commands from the PLC to select the floating point weight output data. The terminal recognizes a command when it sees a new value in the scale's command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the terminal recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. It also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC waits until it receives the command acknowledgment from the terminal before sending another command.

The terminal has two types of values that it can report to the PLC: real-time and static. When the PLC requests a real-time value, the terminal acknowledges the command from the PLC once but sends and updates the value at every interface update cycle. If the PLC requests a static value, the terminal acknowledges the command from the PLC once and updates the value once. The terminal will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, fine feed, and tolerance values are examples of static data.

The terminal can send a rotation of up to nine different real-time values for each scale. The PLC sends commands to the terminal to add a value to the rotation. Once the rotation is established, the PLC must instruct the terminal to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the terminal to advance to the next value. If the terminal is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next interface update cycle. (The interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command, the terminal switches to the next value in the rotation. The terminal stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point rotation examples in Table 6-11 through Table 6-14 for additional information. The method of handling string and floating point data may vary between different PLCs. The IND780 provides floating point data in the byte order entered in the Data Format setup.

Table 6-7 through Table 6-10 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data. The byte order follows the convention of the Modicon Quantum platform. Note that this is Double Word Swapped.

**Table 6-7: Discrete Read Floating Point – IND780 >> PLC Input, per Message Slot**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity 1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmnd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>
17	Cmnd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

#### Notes for Table 6-7

- The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- Bit 0** and **Bit 1** are used only in material transfer mode.
- Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- Bit 6** and **Bit 7** are user defined, and correspond to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- Bit 9**, **Bit 10**, and **Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- Bit 12** is set to a "1" when the scale is in motion (unstable).
- The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.

- 12 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 6-8: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency
12	Notch filter frequency
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size
19–28	Not used
29	Last IND780 error code
30	No data response – command successful
31	No data response – command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* SDV means Shared Data Variable.

**Table 6-9: Discrete Write Floating Point – PLC >> IND780, per Message Slot**

Octal Address	WORD 0 <sup>0</sup>	WORD 1	WORD 2 <sup>1</sup> FP load value	WORD 3 <sup>1</sup> FP load value	Bit number
0	Reserved	Command Word <sup>2</sup>	X	X	<b>0</b>
1			X	X	<b>1</b>
2			X	X	<b>2</b>

3			X	X	<b>3</b>
4			X	X	<b>4</b>
5			X	X	<b>5</b>
6			X	X	<b>6</b>
7			X	X	<b>7</b>
10			X	X	<b>8</b>
11			X	X	<b>9</b>
12			X	X	<b>10</b>
13			X	X	<b>11</b>
14			X	X	<b>12</b>
15			X	X	<b>13</b>
16			X	X	<b>14</b>
17			X	X	<b>15</b>

**Notes for Table 6-9**

- 0 WORD 0 is a reserved 16 bit word, only present in message slot 1.
  - 1 The Bits in WORD 2 and WORD 3 are a single-precision floating point value. This value is used with the command in WORD 1 to instruct the terminal to download the floating point value into the field specified in the command.
  - 2 The command Word 1 is used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 6-10, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.
- Not all commands will require a value in the floating point load value words.

**Table 6-10: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName
0	00	Report next rotation field @ next interface update cycle <sup>1</sup>	
1	01	Report next rotation field <sup>1,2</sup>	
2	02	Report next rotation field <sup>1,2</sup>	
3	03	Reset rotation	
10	0a	Report gross weight <sup>1,3</sup>	
11	0b	Report net weight <sup>1,3</sup>	
12	0c	Report tare weight <sup>1,3</sup>	
13	0d	Report fine gross weight <sup>1,3</sup>	
14	0e	Report fine net weight <sup>1,3</sup>	
15	0f	Report fine tare weight <sup>1,3</sup>	
16	10	Report Rate <sup>1,3</sup>	
17	11	Report SDV ALxx01 <sup>6</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>	
20	14	Report notch filter frequency <sup>3</sup>	
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12
25	19	Report spill value <sup>3,10</sup>	SPxx09
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>
29	1d	Report last error <sup>3</sup>	
30	1e	Report primary increment size <sup>3</sup>	
40	28	Add gross weight to rotation <sup>7</sup>	

Dec	Hex	Command	SDName
76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
78	4e	Disable LCD display <sup>7</sup>	
79	4f	Enable LCD display <sup>7</sup>	
80	50	Set normal display mode <sup>7,9</sup>	PD0119
81	51	Display message 1 <sup>7,9</sup>	PD0119
82	52	Display message 2 <sup>7,9</sup>	PD0119
83	53	Display message 3 <sup>7,9</sup>	PD0119
84	54	Display message 4 <sup>7,9</sup>	PD0119
85	55	Display Message 5 <sup>7,9</sup>	PD0119
86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
88	58	Disable weight display <sup>7</sup>	
89	59	Enable weight display <sup>7</sup>	
90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
104	68	Master control relay OFF <sup>7, 12</sup>	
105	69	Master control relay ON <sup>7, 12</sup>	
110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
111	6f	Set target fine feed value <sup>4,10</sup>	

Dec	Hex	Command	SDName
41	29	Add net weight to rotation <sup>7</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>	
46	2e	Add rate to rotation <sup>7</sup>	
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>
60	3c	Load numeric tare value <sup>4</sup>	
61	3d	Pushbutton tare command <sup>7</sup>	
62	3e	Clear command <sup>7</sup>	
63	3f	Print command <sup>7</sup>	
64	40	Zero command <sup>7</sup>	
65	41	Select scale 1 <sup>7</sup>	
66	42	Select scale 2 <sup>7</sup>	
67	43	Select next scale <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>
73	49	Set low-pass filter corner frequency <sup>4</sup>	
74	4a	Set notch filter frequency <sup>4</sup>	
75	4b	Reset ENTER key <sup>7</sup>	

Dec	Hex	Command	SDName
112	70	Set - tolerance value <sup>4,10</sup>	
113	71	Set target value and apply <sup>4,10,14</sup>	
114	72	Start target logic <sup>7,10,15</sup>	
115	73	Abort target logic <sup>7,10,15</sup>	
116	74	Target use gross weight <sup>7,10</sup>	SPxx04
117	75	Target use net weight <sup>7,10</sup>	SPxx04
118	76	Target use rate <sup>7,10</sup>	SPxx04
119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
121	79	Enable target latching <sup>7,10</sup>	SPxx06
122	7a	Disable target latching <sup>7,10</sup>	SPxx06
123	7b	Reset target latch <sup>7</sup>	SPxx07
124	7c	Set target spill value <sup>4,10</sup>	
131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
160	a0	Apply scale setup <sup>7</sup>	QC0149
161	a1	Write Calibration to EEPROM <sup>7</sup>	
162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
163	a3	Enable Tare on IND780 console <sup>7</sup>	
164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
168	a8	Select scale 3 <sup>7</sup>	
169	a9	Select scale 4 <sup>7</sup>	

### Notes for Table 6-10

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.

15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

6.6.6.2. Floating Point Data Format and Compatibility

In Floating Point Message mode, the PLC and IND780 terminal exchange weight, target, and tare data in single-precision floating point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight and rate data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

Some PLCs require special integrity checking to communicate floating point numbers. There are two data integrity bits that the IND780 uses to maintain data integrity when communicating with the PLC. One bit is in the beginning word of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC, if the PLC program detects this condition, it should send a new command to the terminal.

The method of handling string and floating point data varies between PLCs platforms. The IND780 provides floating point data in the word order set up by the user.

6.6.6.3. Floating Point Command Examples

Table 6-11 through Table 6-14 provide floating point command examples.

**Table 6-11: Data Requirement: Only Net Weight Sent (continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word O:XX1	none required		
2 (IND780 terminal sees new command)			Command ack. =1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word the IND780 terminal will update the net value every interface update cycle.				

**Table 6-12: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2	110 (dec) loaded	floating point value		

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
(PLC sends command to set target 1 cutoff value)	into command word O:XX1	= 21.75		
3 (IND780 terminal sees new command , loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word O:XX1			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	(null value)
The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.				

**Table 6-13: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word O:XX1			
2 (IND780 terminal sees new command)			Command ack.= 1 F.P. ind = 30	
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word O:XX1			
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word O:XX1			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	(null value)
At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.				
7 (PLC sends the command to begin the rotation at interface update cycle)	0 (dec) loaded into command word O:XX1			
8 (IND780 terminal sends gross weight at interface update cycle ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wf.

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 6	Floating point value = rate
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value at next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word O:XX1		Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 6-14: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word O:XX1			
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind = 30	
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word O:XX1			
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word O:XX1			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	(null value)
At this point, the rotation has been set up. Now the PLC needs send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word O:XX1			

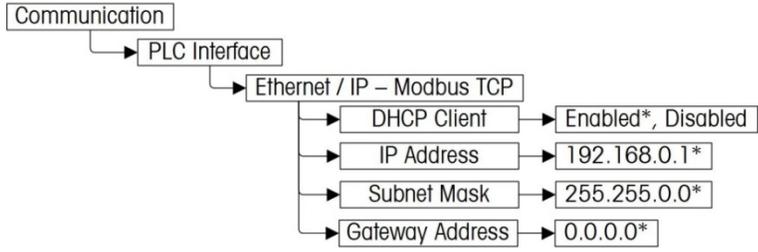
Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
8 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net weight
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word O:XX1			
10 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 2 F.P. ind = 6	Floating point value = rate
At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value				

### 6.6.7. Controlling the Discrete I/O Using a PLC Interface

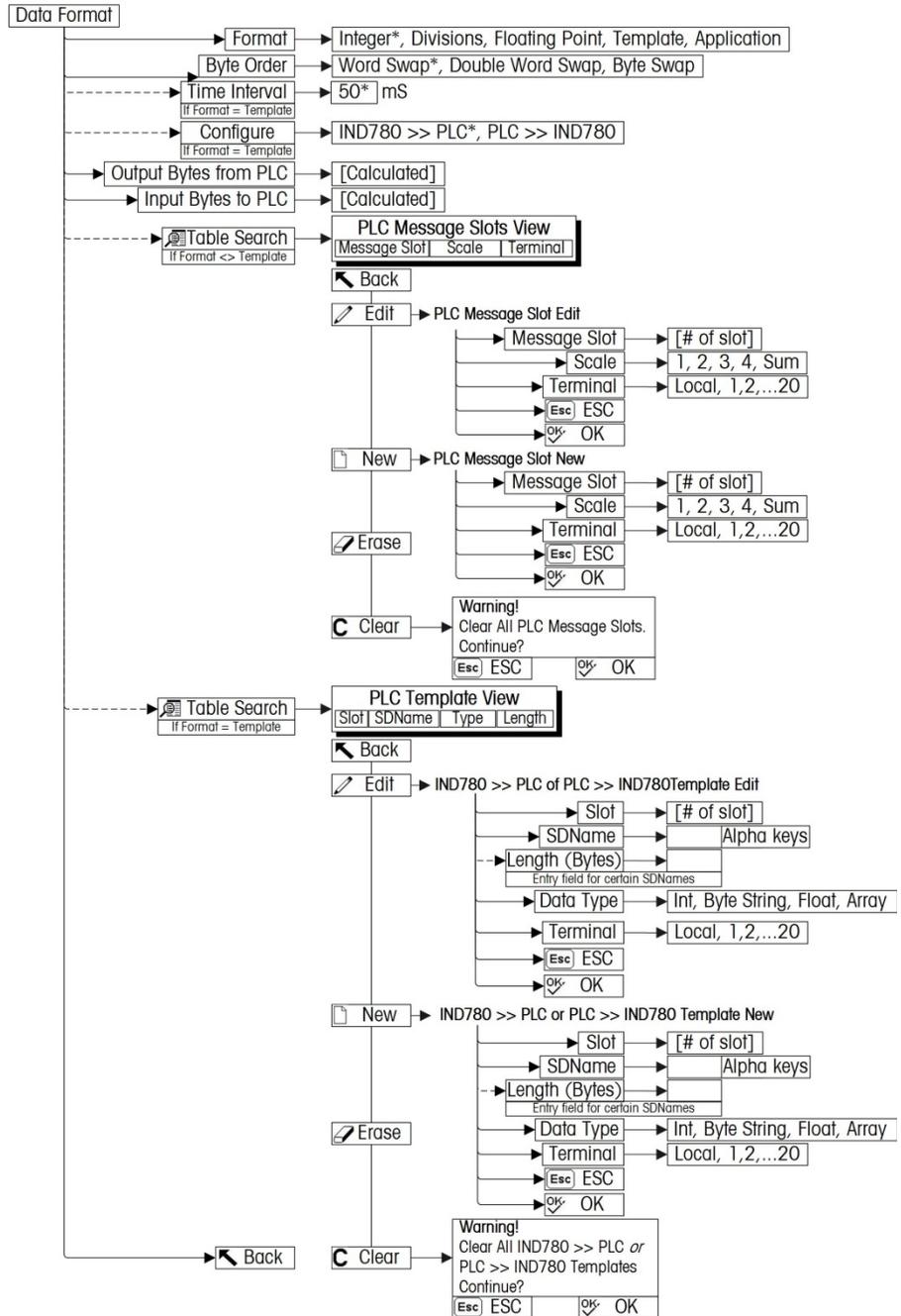
The IND780 terminal provides the ability to directly control some of its discrete outputs and read some of its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the terminal's discrete I/O updates are synchronized with the terminal's interface update cycle rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the IND780 Terminal Technical Manual for discrete I/O wiring. Also note that the outputs must be unassigned in the IND780 terminal setup.

## 6.7. Software Setup

The IND780 terminal automatically detects the presence of a Modbus TCP Kit option board if one is installed. When the option is detected, the IND780 terminal enables the Modbus TCP parameters in setup at **Communications > PLC Interface > Ethernet / IP – Modbus/TCP**. Figure 6-5 shows the Modbus TCP setup block and Figure 6-6 the **Communication > PLC Interface > Data Format** setup block. Default values are indicated with an asterisk (\*).



**Figure 6-5: The Modbus TCP Setup Block**



**Figure 6-6: Modbus TCP Data Format Setup Blocks**

## 6.7.1. Modbus TCP and Data Format Setup Blocks

### 6.7.1.1. Modbus TCP setup

The Modbus TCP setup block lets you specify the IP address of the Modbus TCP Interface. In setup, access **Communication > PLC Interface > Modbus TCP** (Figure 6-7) and set the proper IP address, Subnet Mask, and Gateway Address. Alternatively, DHCP client can be enabled if the IP address is to be assigned automatically by the network server.

IP=172.18.54.80		15/Oct/2008 08:27	
<b>EtherNet/IP - Modbus/TCP</b>			
MAC Address	ETHIP 00301102eab7		
DHCP Client	Disabled		
IP Address	192	168	000.001
Subnet Mask	255	255	255.000
Gateway Address	000	000	000.000
←			

Figure 6-7: Modbus TCP Setup Screen

### 6.7.1.2. Data Format setup

1. In setup, select **Communication > PLC Interface > Data Format**. Figure 6-8 shows the screen with Template selected as the format, revealing the Time Interval, Configure, and directional displays. When Application format is selected, only the Format box appears on this screen.

22/Oct/2007 17:11	
<b>PLC Data Format</b>	
Format	Template
Byte Order	Double Word Swa
Time Interval	25 mS
Configure	IND780 >> PLC
PLC >> IND780	100 bytes
IND780 >> PLC	14 bytes
←	

Figure 6-8: PLC Data Format Screen

1. Select the Format (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the Format will delete any existing Message Slots.

<b>Integer</b>	Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).
<b>Divisions</b>	Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.

<b>Floating Point</b>	Displays weight in floating point data format
<b>Template</b>	Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed "on the fly" by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the Shared Data Reference Manual (document number 64059110) provided on the CD-ROM supplied with your terminal.
<b>Application</b>	Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

2. Select **Byte Order** (see Table 6-15): [default Word Swap]

**Table 6-15: PLC Data Byte-Ordering**

	Byte Swap	Word Swap	Double Word Swap
Integer	1 2	2 1	1 2
Float	1 2 3 4	4 3 2 1	3 4 1 2
String	A B C D	ABCD	ABCD

<b>Word Swap</b>	Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.
<b>Byte Swap</b>	Makes the floating point format compatible with S7 PROFIBUS.
<b>Double Word Swap</b>	Makes the data format compatible with the Modicon Quantum PLC for Modbus TCP networks.

3. Set the **Time Interval** in milliseconds. (Only when the Format is Template)
4. Use the **Configure** drop-down list to select the direction of data flow – IND780 >> PLC or PLC >> IND780. (Only when the Format is Template)
5. The bottom of this screen (Figure 6-8) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.

6. Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of **Message Slots** (maximum number is 12; default is 0), or to configure the Template for communications between the IND780 and the PLC. Figure 6-9 shows the Message Slots View, which is viewed whenever the Format is **not** Template.

IP=172.18.54.111 06/Dec/2007 16:25

**PLC Message Slots View**

Message	Scale	Terminal
1	1	Local
2	- End -	







**Figure 6-9: PLC Message Slots View Screen**

Figure 6-10 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1 07/Mar/2006 11:08

**PLC >> IND780 Template View**

Slot	SDName	Type	Length
1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		







**Figure 6-10: PLC Template View**

Press the EXIT softkey  to exit the view screen.

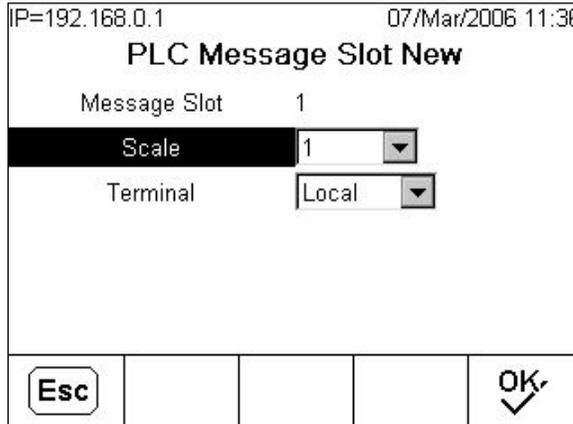
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey  to delete all existing slots.

Figure 6-11 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 6-9).

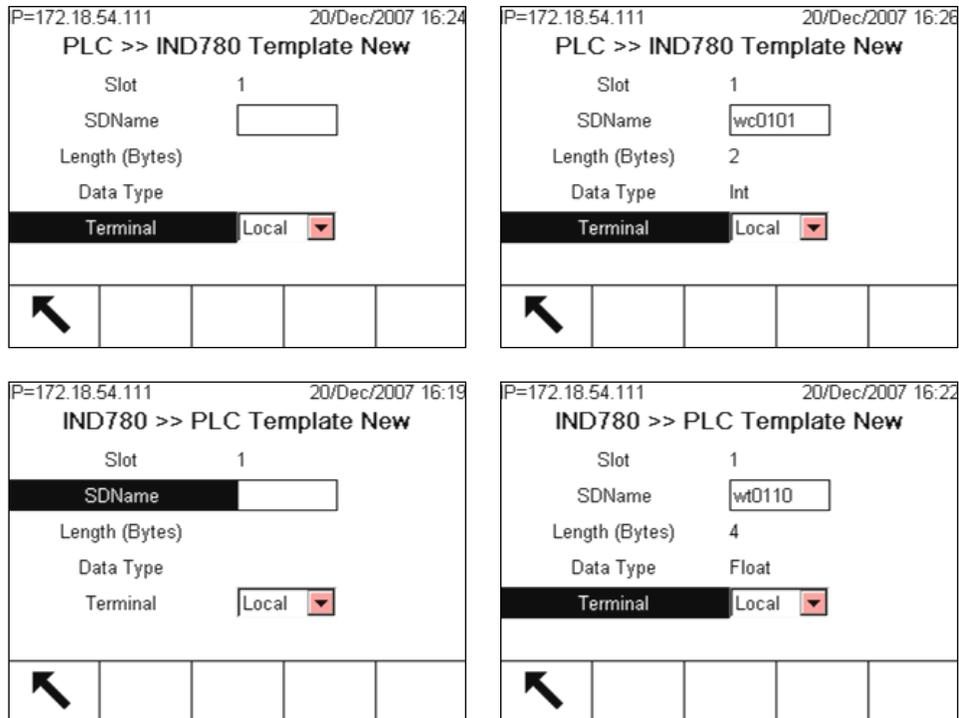


**Figure 6-11: New Message Slot Screen**

**Scale** Enter the scale number to be associated with the new Message Slot.

**Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 6-12 shows a series of Template New screens, as accessed from the Template View screen (Figure 6-10).



**Figure 6-12: New Template Screens, PLC to IND780 and IND780 to PLC**

**Slot** Automatically assigned.

<b>SDName</b>	An alphabetic entry field used to define the Shared Data variable associated with the slot.
<b>Length (Bytes) and Data Type</b>	Automatically displayed once the SDName has been entered.
<b>Terminal</b>	Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

- For a complete listing of Shared Data Fields, refer to the IND780 Shared Data Reference Manual. Refer to the Table 6-5 (Discrete Read) and Table 6-6 (Discrete Write) in this chapter for additional information on mapping of discrete read and write data to the PLC.

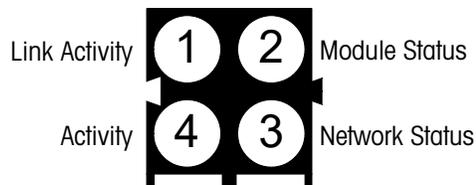
## 6.8. Troubleshooting

If the IND780 does not communicate with PLC, do the following:

- Check wiring and network termination.
- Confirm that the IND780 settings for data type and IP Address assignment match those in the PLC and that each IND780 has a unique address.
- Replace the EtherNet / IP – Modbus TCP interface kit if the problem persists.
- If the communication kit was changed from another type, like PROFIBUS, ControlNet or Remote I/O, a master reset of the IND780 must be performed.

### 6.8.1. Status LEDs

The EtherNet/ IP – Modbus TCP interface card has four status LEDs indicators to indicate communication and fault status of the card. Figure 6-2 indicates the location of these LEDs, and Figure 6-13 shows the array of the LEDs on the card. Table 6-16 explains the meaning of each LED.



**Figure 6-13: EtherNet / IP – Modbus TCP Board Status LED Array**

**Table 6-16: EtherNet / IP LEDs Status Indications**

LED #	State	Status
1 - Link Activity	Off	No link (or no power)
	Green	Connected to an Ethernet network
2 - Module Status	Off	No power
	Green (1 Hz flash)	IP address not set via switch
	Red (1 Hz flash)	Invalid MAC address (internal error)
	Red (2 Hz flash)	Failed to load Ethernet configuration
	Red (4 Hz flash)	Fatal internal error

LED #	State	Status
	Red	Duplicate IP address detected
3 - Network Status	This LED indicates the number of established Modbus/TCP connections to the terminal. The number of established connections is equal to the number of "blinks" or "flashes" on this LED.	
4 - Activity	Off	No Ethernet activity (or no power)
	Green, flashing	Flashes green each time a packet is received or transmitted

## 6.9. Modbus TCP Option Kit

No spare parts are associated with the Modbus TCP option kit. The kit CIMF part number is 64058677. Table 6-17 shows what comes in the kit.

**Table 6-17: Modbus TCP Option Kit**

Description	Qty.
Installation Instructions	1
PCB Package	1
Clamp-on ferrite	1
M3 x 8 screws	3

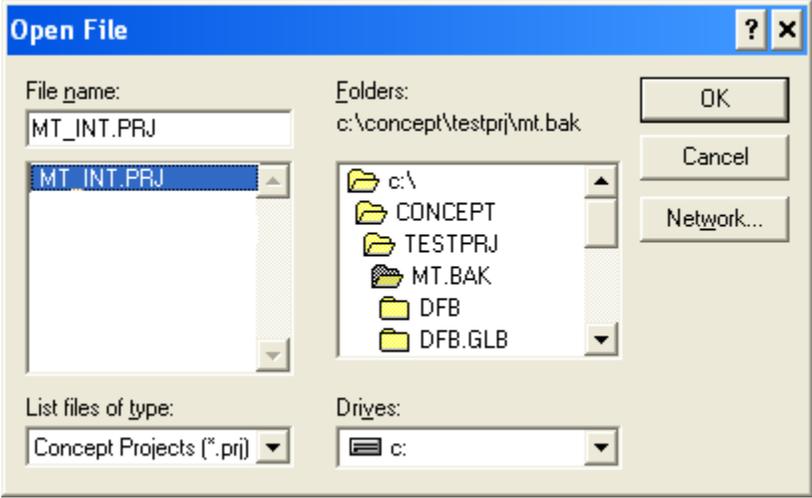
## 6.10. Modbus TCP Configuration Example

This demo was set up using Concept Version 2.6 XL, SR1, b (Figure 6-14).



**Figure 6-14: Concept Programming Unit Welcome Screen**

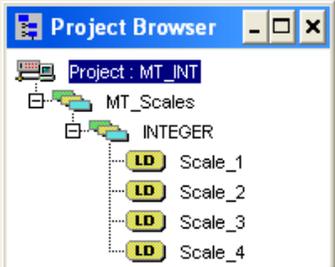
Open a project by accessing the file menu and selecting OPEN, then selecting the project. In this example, the project is named MT\_INT.PRJ (Figure 6-15).



**Figure 6-15: Project Selection Dialog**

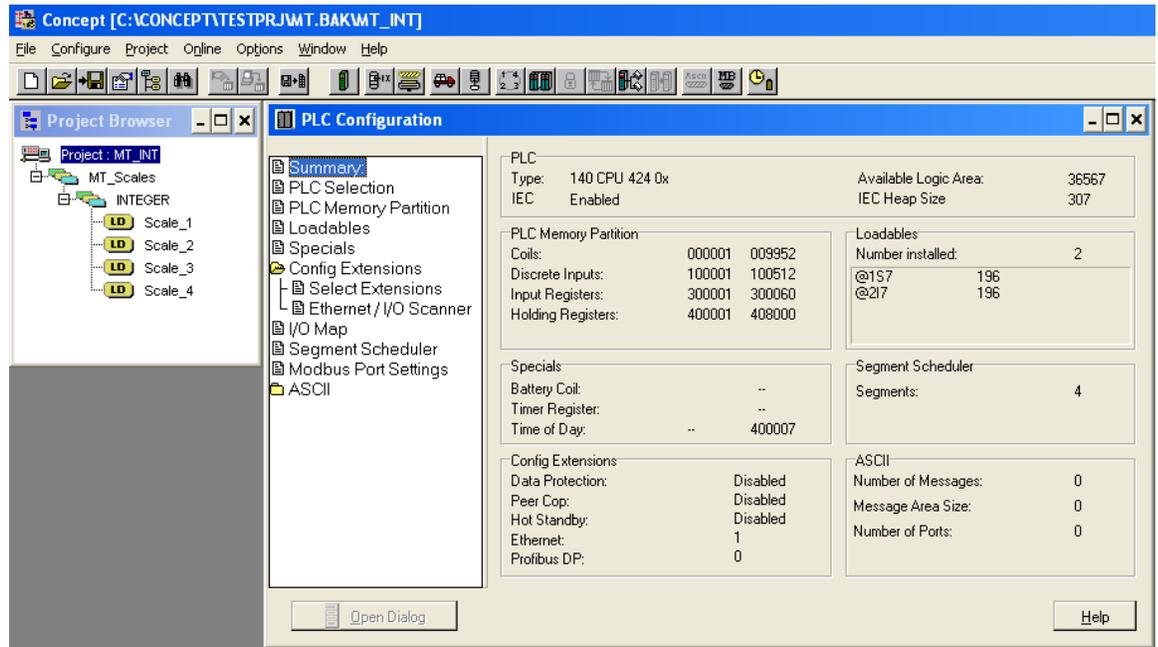
Once the project is open, the project browser should appear; if it does not appear, click on  to display it.

Next the Network card must be configured. Double click on your project in the project browser. In this example, click on the blue highlighted (Figure 6-16) item to open the PLC Configuration window.



**Figure 6-16: Project Viewed in Project Browser**

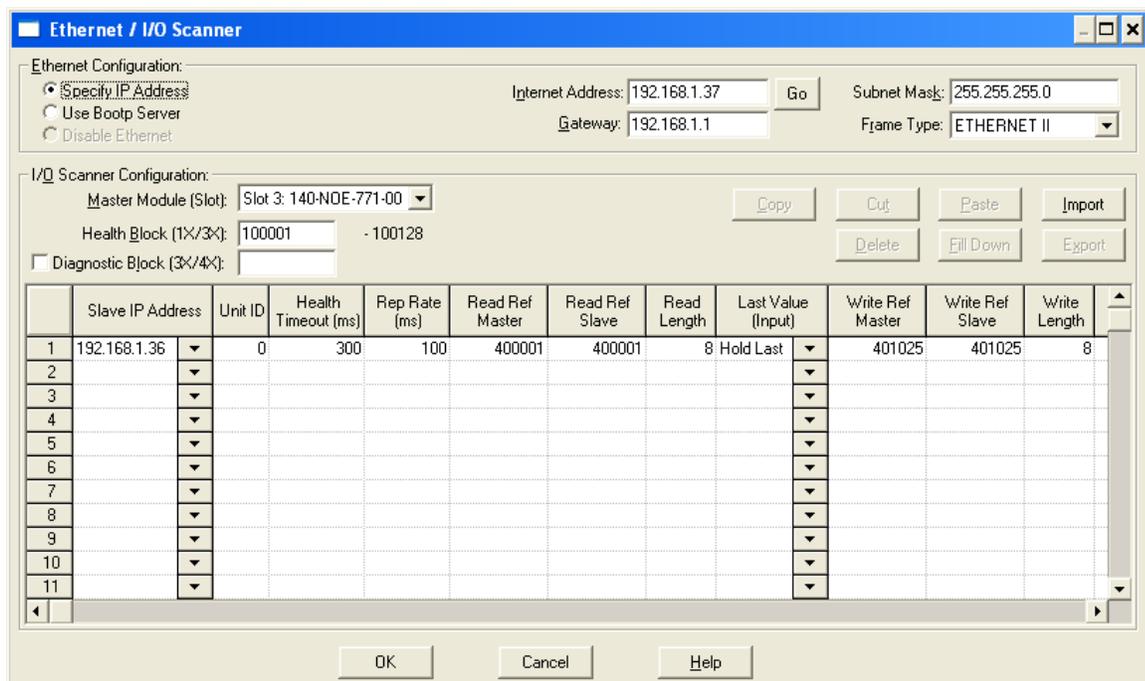
The PLC Configuration window (Figure 6-17) will open.



**Figure 6-17: PLC Configuration Window**

Click on the Config Extensions Folder in the center pane, above. The branch will expand to show Ethernet / I/O Scanner. Double click on the Ethernet / I/O Scanner to bring up the details of the Ethernet card (Figure 6-18).

Here, the IP addresses must be configured – the PLC’s, that of the IND780 with which it communicates. The data communicated to and from the IND780 is also configured in this window.

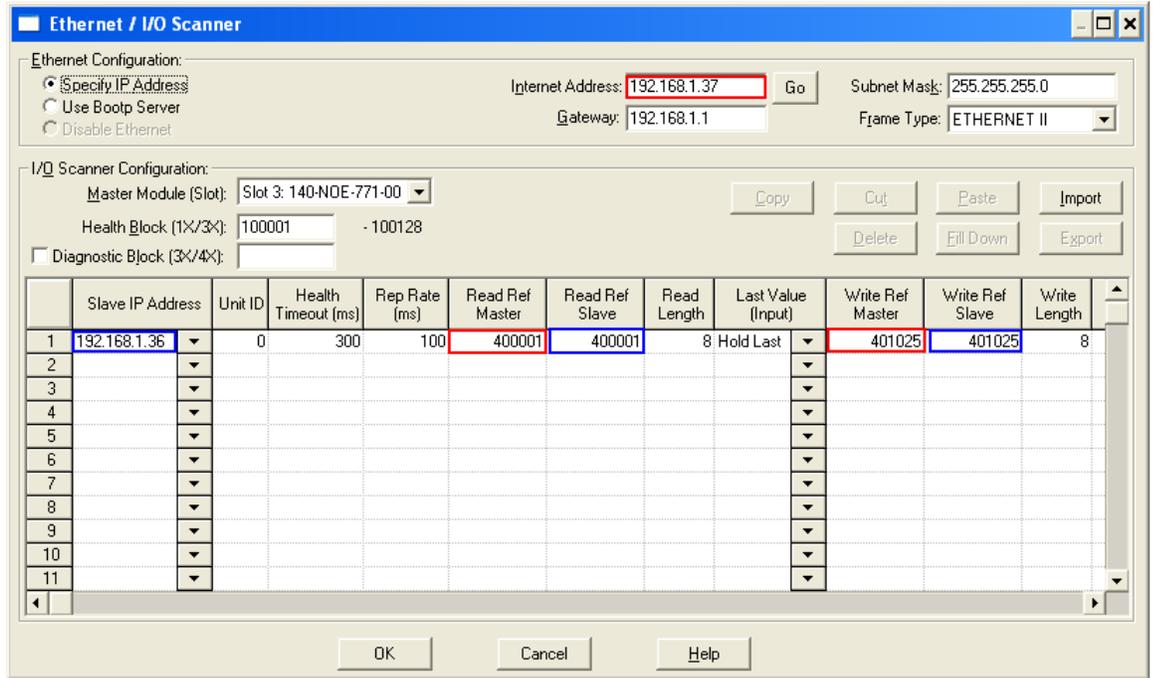


**Figure 6-18: Ethernet / I/O Scanner Window**

For a more detailed description of each column in the configuration window, click on the Help button (at lower right in Figure 6-18). The following elements must be configured:

- Slave IP Address** IP Address of the IND780 terminal’s Modbus TCP interface. This value is configured in the IND780 Setup tree at **Communication > PLC Interface > Ethernet/IP - Modbus/TCP**.
- Unit ID** This value is typically 0
- Health Timeout**
- Rep Rate**
- Read Ref Master** The start of PLC registers to which the IND780’s information is written. This address ALWAYS is 400001
- Read Ref Slave** The start of IND780 register where the scale data is stored. This address can be any value 4XXXXX PLC address.  
\* NOTE, data in the Read Ref Slave is read and then stored in the Read Ref Master.
- Read Length & Write Length** This is determined by the IND780 settings, and is determined by # of scales, Mode of operation etc. In our example we are using 4 scales in INTEGER Mode. In the IND780 we are reading 16 bytes and writing 16 bytes. When configuring the PLC each 4XXXX register address word consists of 2 bytes of information. This gives a total of 16 bytes / 2 bytes per word, or 8 for Read Length and 8 for Write Length.

Both the PLC and the IND780 IP address and address settings must be configured – refer to Figure 6-19. The Ethernet card used on the configuration shown is the 140-NOE-771-00



  PLC  
  IND780

**Figure 6-19: PLC and IND780 Values for Ethernet / I/O Scanner**

Examples of how to configure the Modicon Ethernet I/O scanner for various scale configurations are provided below.

6.10.1.1. Integer and Division Mode Configuration

The IND780 Configured for 4 Scales in either INT or DIV mode. 8 Words are Read into the PLC and 8 words are written to the IND780. Table 6-18 indicates the values for each scale.

	Slave IP Address	Unit ID	Health Timeout (ms)	Rep Rate (ms)	Read Ref Master	Read Ref Slave	Read Length	Last Value (Input)	Write Ref Master	Write Ref Slave	Write Length
1	192.168.1.36	0	300	100	400001	400001	8	Hold Last	401025	401025	8

**Figure 6-20: INT or DIV Mode Configuration**

**Table 6-18: Configuration for Four Scales, INT or DIV Mode**

Description	Slot / Scale*	Address in IND780	Format
<b>Read by PLC from 780:</b>			
Weight Data	Slot 1	400001	Int
Status Data	Slot 1	400002	Int
Weight Data	Slot 2	400003	Int
Status Data	Slot 2	400004	Int

Description	Slot / Scale*	Address in IND780	Format
Weight Data	Slot 3	400005	Int
Status Data	Slot 3	400006	Int
Weight Data	Slot 4	400007	Int
Status Data	Slot 4	400008	Int
<b>The PLC will write to:</b>			
Data Value to be written	Slot 1	401025	Int
Command Word	Slot 1	401026	Int
Data Value to be written	Slot 2	401027	Int
Command Word	Slot 2	401028	Int
Data Value to be written	Slot 3	401029	Int
Command Word	Slot 3	401030	Int
Data Value to be written	Slot 4	401031	Int
Command Word	Slot 4	401032	Int

\* Note that any scale can be configured to correspond with any slot number.

4001, 40001, 400001 PLC Memory Dependent.

#### 6.10.1.2. Floating Point Mode Configuration

The IND780 Configured for 4 Scales in Floating Point mode FP. 16 Words are Read into the PLC and 13 words are written to the IND780. Table 6-19 indicates the values for each scale.

1	Slave IP Address	Unit ID	Health Timeout (ms)	Rep Rate (ms)	Read Ref Master	Read Ref Slave	Read Length	Last Value (Input)	Write Ref Master	Write Ref Slave	Write Length
	192.168.1.36	0	300	100	400001	400001	16	Hold Last	401025	401025	13

**Figure 6-21: FLP Mode Configuration**

**Table 6-19: Configuration for Four Scales, FP Mode**

Description	Slot / Scale*	Address in IND780	Format
<b>Read by PLC from 780:</b>			
Weight Data	Slot 1	400002-400003	Float
Command Ack Register	Slot 1	400001	Int
Status Register	Slot 1	400004	Int
Weight Data	Slot 2	400006-400007	Float
Command Ack Register	Slot 2	400005	Int
Status Register	Slot 2	400008	Int

Description	Slot / Scale*	Address in IND780	Format
Weight Data	Slot 3	400010-400011	Float
Command Ack Register	Slot 3	400009	Int
Status Register	Slot 3	400012	Int
Weight Data	Slot 4	400014-400015	Float
Command Ack Register	Slot 4	400013	Int
Status Data	Slot 4	400016	Int
<b>The PLC will write to:</b>			
Reserved	Slot 1	401025	Int
Command Word	Slot 1	401026	Int
Data Value to be Written	Slot 1	401027-401028	Float
Command Word	Slot 2	401029	Int
Data Value to be Written	Slot 2	401030-401031	Float
Command Word	Slot 3	401032	Int
Data Value to be Written	Slot 3	401033-401034	Float
Command Word	Slot 4	401035	Int
Data Value to be Written	Slot 4	401036-401037	Float

\* Note that any scale can be configured to correspond with any slot number.

4001, 40001, 400001 PLC Memory Dependent.

## 6.10.2. Integer Logic Examples

2 Words of Data are associated with a scale when in INT mode.

- Weight Data for scale 1 is stored in the IND780 in register 400001.
- Status Data for this weight and the IND780 is in register 400002.

### 6.10.2.1. Read Logic

The 400001 weight data can be read directly by the PLC. However, to understand the 400002 Status data fully some basic logic is needed to break the data Word into Bits.

In concept the use of an INT\_TO\_WORD instruction will first read the INT value from the IND780 in a form that can be broken into bits. Then once the data is in a word format, a WORD\_TO\_BIT instruction will complete the process of extracting the individual bits. Figure 6-22 and Figure 6-23 show an example of logic that can be used to read the status word.

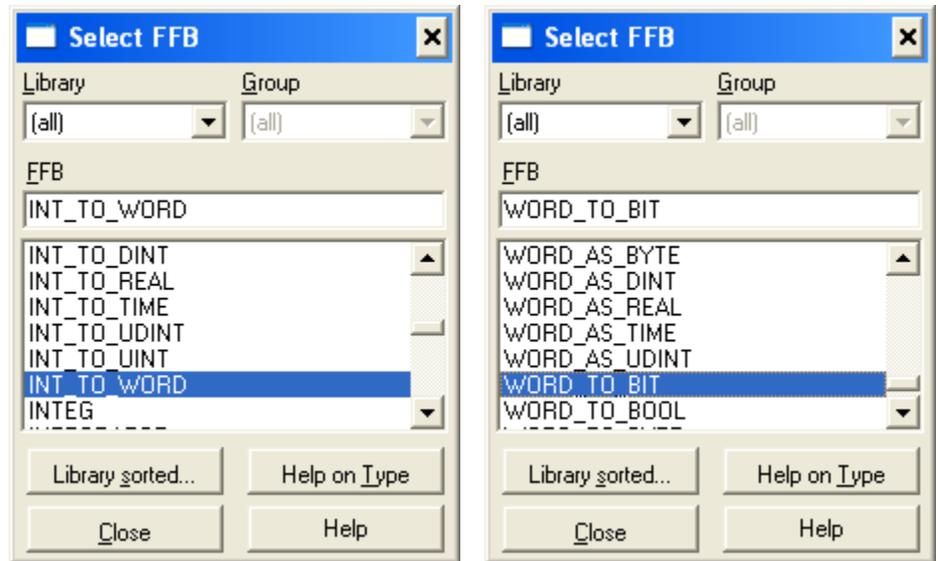


Figure 6-22: Selecting Integer-to-Word (left) and Word-to-Bit (right) Conversions

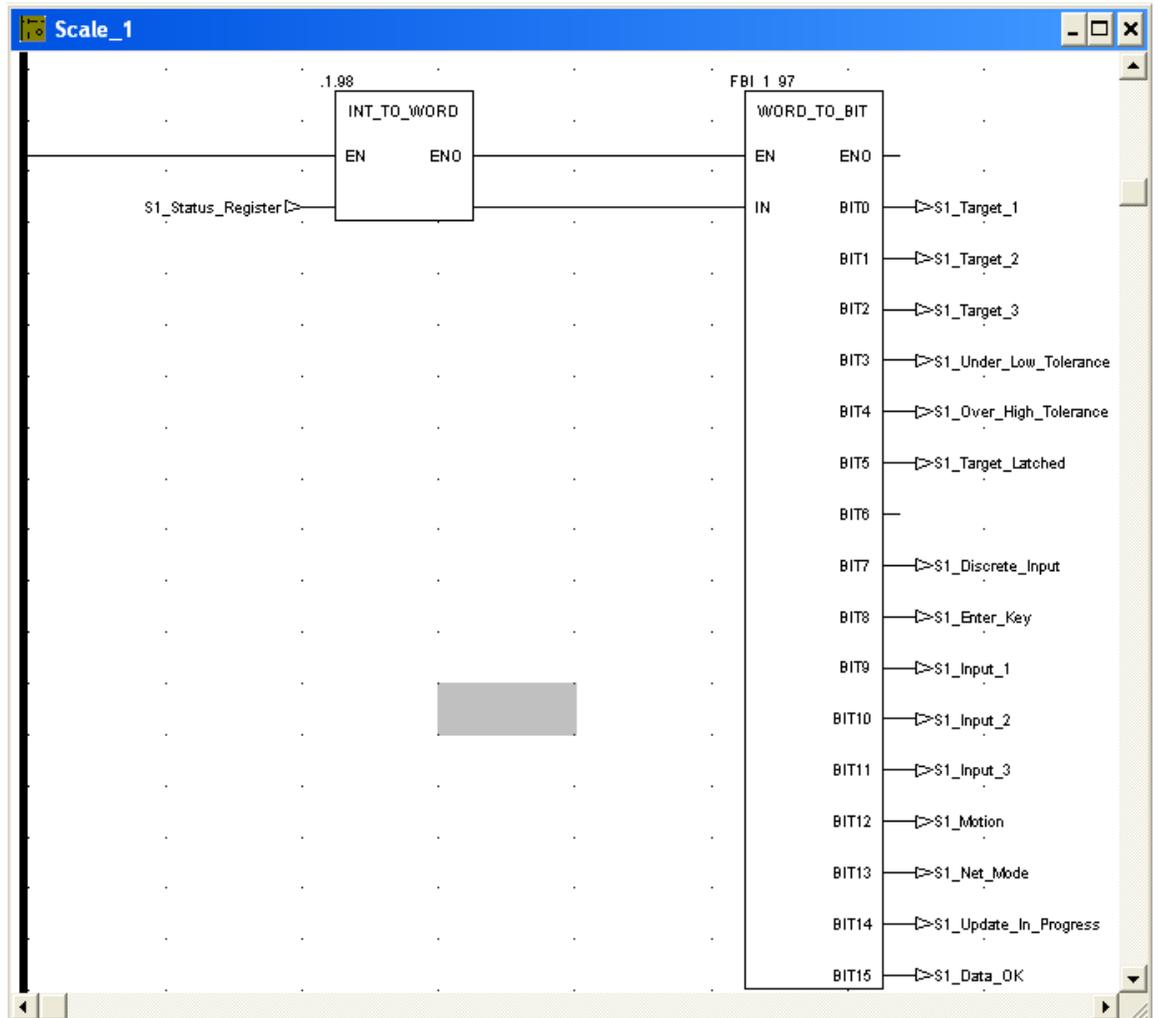


Figure 6-23: Integer-to-Word and Word-to-Bit Logic

6.10.2.2. Write Logic

The 401025 Data Value can be written directly by the PLC. However, to utilize the 401026 command Word fully some basic logic is needed to convert the command bits into a data Word.

In concept, the use of a BIT\_TO\_WORD instruction will first get the command bits into a WORD value. Next the use of a WORD\_TO\_INT instruction will complete the process of packing the individual command bits into an INTEGER format that can be written to the IND780. Figure 6-24 shows an example of logic that can be used control the command word.

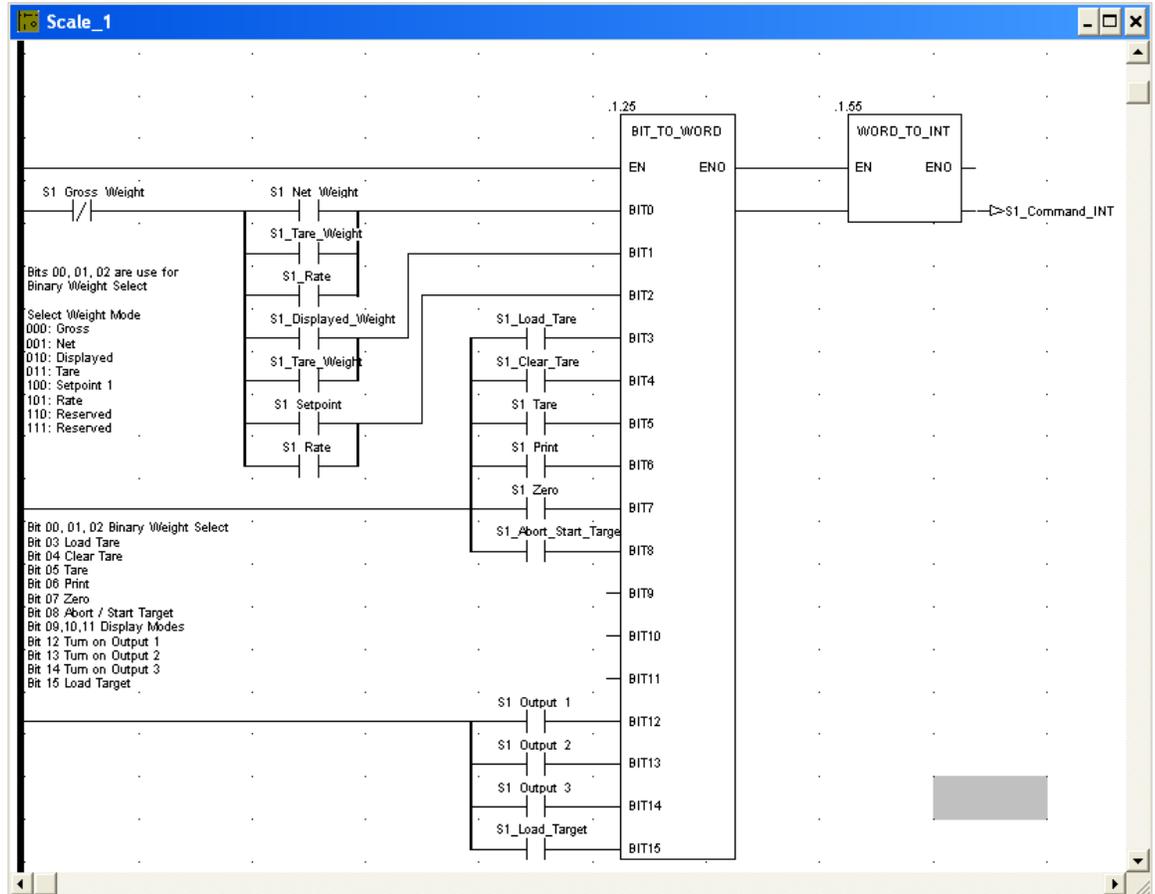


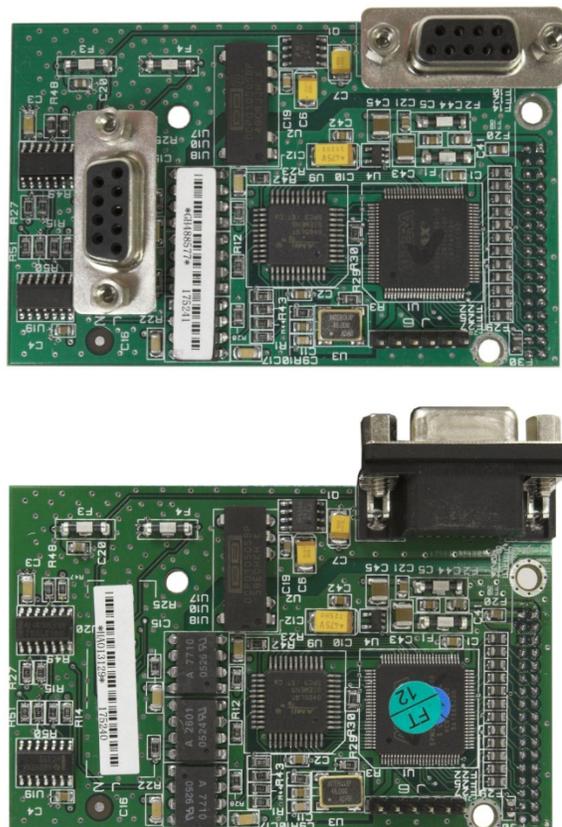
Figure 6-24: Bit to Word and Word to Integer Logic

# 7 PROFIBUS

## 7.1. Overview

The PROFIBUS option card enables the IND780 terminal to communicate to a PROFIBUS DP master according to DIN 19 245. It consists of an IND780 terminal backplane-compatible module and software that resides in the terminal, which implements the data exchange.

The PROFIBUS option card interfaces to programmable logic controllers (PLCs) such as Texas Instruments 505 series, Siemens S5 series and Siemens S7 series PLCs. The PROFIBUS appears as a block of I/O on the PROFIBUS network. The size and mapping of the I/O depends on the setup of the PROFIBUS card at the IND780. Figure 7-1 shows examples of PROFIBUS interface modules used in the IND780 Panel Mount and Harsh Enclosures.



**Figure 7-1: PROFIBUS Interface Module: Panel Mount (top) and Harsh (bottom)**

The data mapped within the I/O block is defined as Discrete or Shared Data Variables. Based upon the IND780 setup, discrete data is either Integer, Division or Floating Point.

Discrete data is sent in groups defined as message blocks. The number of message blocks (1 to 12) is set up within the IND780. While the format of each message block is the same, the data received and displayed within a message block depends on the commands within the block.

The Texas Instruments (TI) 505 PLCs interface to the PROFIBUS via an I/O processor called a Field Interface Module (FIM). The FIM bus master recognizes a fixed set of PROFIBUS slave devices, all of which are viewed by it as some sort of remote I/O rack. On power up, the FIM queries each PROFIBUS slave node to determine which of the recognized types a device might be. The FIM then configures itself accordingly. The PROFIBUS option appears to the FIM as a small ET200U I/O rack.

The Siemens S5-115 series PLC also interfaces to the PROFIBUS using an IM-308 I/O processor. This device must be locally programmed with the terminal interface type files. Newer Siemens S7 PLCs have the PROFIBUS option on their main controller card.

**7.1.1. Communications**

PROFIBUS is based on a variety of existing national and international standards. The protocol architecture is based on the Open Systems Interconnection (OSI) reference model in accordance with the international standard ISO 7498.

The IND780 terminal supports the PROFIBUS-DP. DP stands for Distributed Peripherals, and is designed for high-speed data transfer at the sensor actuator level. At this level, controllers such as PLCs use a fast serial link to exchange data with their distributed peripherals. Data exchange with these distributed devices is mainly cyclic. The central controller (master) reads the input information from the slaves and sends the output information back to them. It is important that the bus cycle time is shorter than the program cycle time of the controller, which is approximately 10 ms in most applications. The following is a summary of the technical features of the PROFIBUS-DP communications protocol:

7.1.1.1. Transmission Technique

PROFIBUS DIN 19 245 Part 1

- EIA RS 485 twisted pair cable or fiber optic
- 9.6 kbit/s up to 12 Mbit/s, max distance 200 m at 1.5 Mbit/s extendable with repeaters
- 12 megabaud maximum rate

7.1.1.2. Medium Access

Hybrid medium-access protocol according to DIN 19 245 Part 1

- Mono-Master or Multi-Master systems supported
- Master and Slave Devices, max 126 stations possible

7.1.1.3. Communications

Peer-to-Peer (user data transfer) or Multicast (synchronization)

- Cyclic Master-Slave user data transfer and acyclic Master-Master data transfer

7.1.1.4. Operation Modes

- Operate: Cyclic transfer of input and output data

- Clear: Inputs are read and outputs are cleared
- Stop: Only master-master functions are possible

#### 7.1.1.5. Synchronization

Enables synchronization of the inputs and/or outputs of all DP-Slaves

- Sync-Mode: Outputs are synchronized
- Freeze-Mode: Inputs are synchronized

#### 7.1.1.6. Functionality

- Cyclic user data transfer between DP-Master(s) and DP-Slave(s)
- Activation or deactivation of individual DP-Slaves
- Checking of the configuration of the DP-Slaves
- Powerful diagnosis mechanisms, three hierarchical levels of the diagnosis
- Synchronization of inputs and/or outputs
- Address assignment for the DP-Slaves over the bus
- Configuration of the DP-Master (DPM1) over the bus
- Maximum 246 byte input and output data per DP-Slave, typical 32 byte
- Maximum total bytes (input bytes + output bytes) 358

#### 7.1.1.7. Security and Protection Mechanisms

- All messages are transmitted with Hamming Distance HD=4
- Watch-Dog Timer at the DP-Slaves
- Access protection for the inputs/outputs at the DP-Slaves
- Data transfer monitoring with configurable timer interval at the DP-Master (DPM1)

#### 7.1.1.8. Device-Types

- DP-Master Class 2 (DPM2) for example, programming/configuration device
- DP-Master Class 1 (DPM1) for example, central controller like PLC, CNC, or RC
- DP-Slave for example, Input/Output device with binary or analog inputs/outputs, drives

#### 7.1.1.9. Cabling and Installation

- Coupling or uncoupling of stations without affecting other stations
- Proven and easy to handle two conductor transmission technique

### 7.1.2. Node/Rack Address

Each IND780 PROFIBUS option card represents one physical node. The node address is chosen by the system designer and then programmed into the IND780 and PLC using the Setup/ Communications/PLC Interface/PROFIBUS tree, accessible through the IND780 front panel user interface. The node address and number of input and output words used to communicate between

the terminal and the PLC are programmed into the PLC by using its PROFIBUS network configuration software and the IND780's PROFIBUS GSD-type files.

The IND780 setup allows selection of the logical rack (node) address, the option of sending and receiving shared data (Explicit Messages Enabled/Disabled), data format (Integer/Divisions/Floating Point/Template/Application), Byte Order (Byte Swap/Word Swap), and the number of message slots or data assigned to the node. The number of input and output words required and the mapping of I/O data is dependent on these selections.

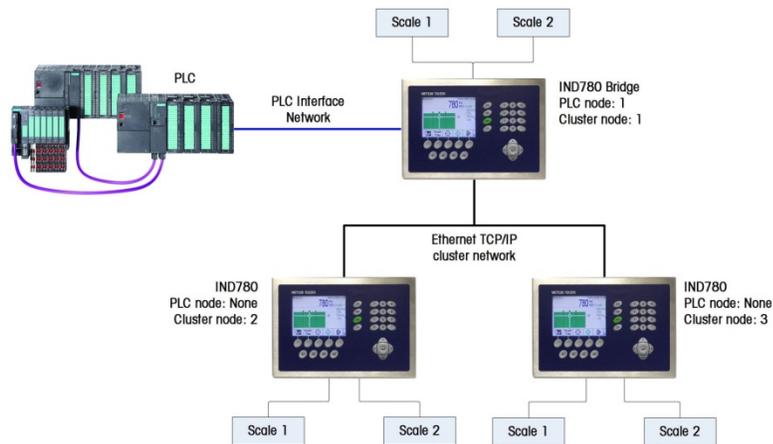
The IND780 appears as a block of I/O data on the PROFIBUS bus. The IND780 terminal will determine the number of input and output words needed for the number of configured message slots and/or the chosen data format. The number of words required is displayed on the PLC Data Format screen of the IND780.

The IND780 PROFIBUS GSD has 25 different blocks of I/O defined. Blocks are defined for the various Integer/Division/Floating Point modes. Each block is identified as the number of input and output words configured within the block.

## 7.2. Sharing a PLC Interface

### 7.2.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. The following example (Figure 7-2) shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 7-2: Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate PLC chapters in this manual for further details on message slots' limitations.

To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure

7-3 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to section 7.5, **Software Setup**, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

PLC Message Slots View		
Message	Scale	Terminal
1	1	Local
2	1	6
3	- End -	

**Figure 7-3: PLC Message Slots View Screen**

## 7.3. Data Definition

### 7.3.1. Data Formats

The terminal's PROFIBUS option card has five types of data exchanges that may be selected. The data types are: Integer, Divisions, Floating Point, Template and Application.

In Integer/Divisions/Floating Point mode, each message slot selected to pass data through the terminal's PROFIBUS option has its own assigned input and output words for continuous communication to and from the PLC. In these modes, Shared data access is only available when the Setup/ Communications/PLC Interface/PROFIBUS Explicit Messages option is "Enabled". This data is used to pass information that cannot be sent in the discrete data because of its size or due to process speed limitations. It uses additional input and output word space. The length of shared data value and data type depends on the type of shared data field requested. In no case does it exceed 10 words (20 bytes).

In Template mode, the designer selects the types of data – and, in case of string data, the length of the data – contained within each data slot.

In Application mode, a Task Expert program is written to determine the information contained within the input and output messages to the PLC. The formatting of the data is set up within the application program.

### 7.3.2. Data Integrity

The terminal has specific bits to allow the PLC to confirm that the data was received without interruption, and the scale is not in an error condition. It is important to monitor these bits. The PLC code must use them to confirm the integrity of the data received for the scale. Refer to the detailed data charts for specific information regarding the Data OK, update in progress, and data integrity bits and their usage.

### 7.3.3. Discrete Data

Three formats of discrete data are available with the PROFIBUS option card: Integer, division, and floating point.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16-bit binary word (signed integer) numerical values.

The floating-point format allows bi-directional communication of discrete bit-encoded information or numeric data encoded in IEEE 754, single-precision floating-point format.

The discrete data format affects the input/output word space required per message slot and the amount of input/output words used by the PROFIBUS option card.

Integer and division formats require two 16-bit words of input and two 16-bit words of output data per message slot. One slot uses two 16-bit words of input and two 16-bit words of output; two slots use four 16-bit words of input and four 16-bit words of output; three slots use six 16-bit words of input and six 16-bit words of output; and four slots use eight 16-bit words of input and eight 16-bit words of output.

The floating-point format provides four 16-bit words of input data and three 16-bit words of output data per Message Slot. Refer to Table 7-10.

requires more space per messages slot because floating point data uses two 16-bit words of data to represent the numeric data alone. The floating-point format requires four 16-bit words of input and four 16-bit words of output data per slot. Four scales using the floating-point format would use 16 words of input and 16 words of output data.

Selection of the appropriate format depends on different issues. The range or capacity of the scale used in the application should be considered. The integer format can represent a numerical value of up to 32,767; the division format can represent a numerical value of up to 32,767 divisions (or increments); and, the floating-point format can represent a numerical value encoded in IEEE 754, single precision floating-point format.

Floating point is the only format that includes decimal point information as a part of its data. All other formats ignore decimal points in their data. Accommodation of decimal point location must take place in the PLC logic, when it is needed with these formats.

7.3.3.1.

Examples

250 x .01 scale				
Scale Reads:	0	2.00	51.67	250.00
Format sent:				
Int	0	200	5167	25000
Div	0	200	5167	25000
FLT	0	2.00	51.67	250.00

Any of the formats could be used in this case.

50,000 x 10 scale				
Scale Reads:	0	200	5160	50000
Format sent:				
Int	0	200	5160	-(xxxxx)
Div	0	20	516	5000
FLT	0	200	5160	50000

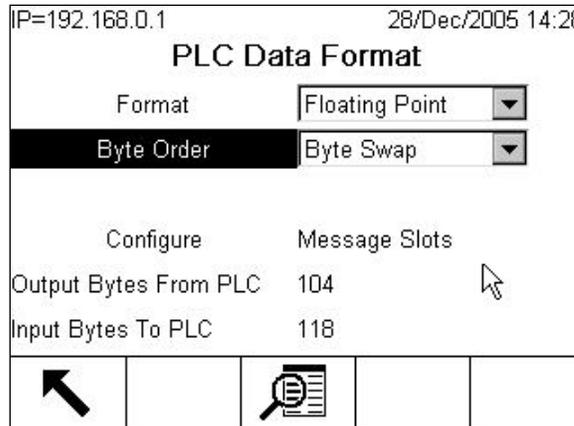
The integer format could not be used because it would send a negative value once the weight exceeded 32,760.

150 x .001 scale				
Scale Reads:	0	2.100	51.607	150.000
Format sent:				
Int	0	2100	-(xxxxx)	-(xxxxx)
Div	0	2100	-(xxxxx)	-(xxxxx)
FLT	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32,767. Please see the detailed description of data available for each format to determine which is most suitable.

**7.3.4. PLC Data Byte-Ordering**

Data Byte Ordering is selected from the Setup/Communication/Plc Interface/Data Format screen. Set Byte Order to **Byte Swap**. The example in Figure 7-4 shows the "PLC Data Format" screen, with the Format set to Floating Point and Byte Order set to Byte Swap.



**Figure 7-4: PLC Data Format**

7.3.4.1. Setup

The sequence for setting up the IND780 to communicate on a PROFIBUS Network is detailed in the PROFIBUS Setup section, below.

**7.3.5. Integer/Divisions Format**

The IND780 supports up to 12 scale slots that it places into one larger Integer Assembly. Scales may be assigned to Scale Slot Assemblies in any combination using the PLC Data Format menus. A scale can be assigned to multiple slots to get different data from the scale, such as weight and rate, reported concurrently to the PLC. The commands are typically single-bit.

When the Integer or Division format is selected, each message slot will have two 16-bit words for Read data and two 16-bit words for Write data. Read data refers to the data sent from the IND780 to the PLC (PLC Read). Write Data refers to data sent from the PLC to the IND780 (PLC Write).

The Read data consists of one 16-bit word for the IND780’s weight information (Word 0 IN) and one 16-bit word for bit-encoded status information (Word 1 IN). The IND780 selects the type of data that is placed in Word 0 IN based on the selection contained in Word 1 OUT, Bits 0, 1 and 2.

The PLC’s output words – Word 0 OUT and Word 1 OUT – consist of one 16-bit integer value, which may be used to download a tare or target (Word 0 Out) and one 16-bit word for bit-encoded command information (Word 1 OUT).

Table 7-1 and Table 7-2 provide detailed information on the integer and division data formats. Note that the functions of Word 1 IN, Bits 0, 1 and 2 depend on the application setup of OVER/UNDER or Material Transfer.

Table 7-4 and Table 7-5 are memory maps of the Input and Output Assemblies.

The IND780 writes the PLC Display command to pd0119 and the Display buffer to pd0118. In order to display this data, the system integrator must write a Task Expert application.

7.3.5.1. Discrete Read INTEGER or Division Word 0 IN

WORD 0 is a 16-bit, signed integer that may represent the scale's gross, net, tare, rate, target, or displayed weight. Three bits, set by the PLC in the Discrete Write Word 1, Bits 0, 1, and 2, designate what data is represented in Discrete Word 0.

7.3.5.2. Discrete Read INTEGER or Division Word 1 IN

**Table 7-1: Discrete Read Integer (weight) or Division (div) – IND780 >> PLC**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

**Notes for Table 7-1:**

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
- 5 **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit

and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.

- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

7.3.5.3. Discrete Write INTEGER or Division Word 1 OUT

**Table 7-2: Discrete Write Integer (weight) or Division (div)—PLC >> IND780**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

**Notes for Table 7-2:**

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- 2 A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 or 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- 3 A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- 4 A transition from "0" to "1" initiates a CLEAR command.
- 5 A transition from "0" to "1" initiates a TARE command.
- 6 A transition from "0" to "1" initiates a PRINT command.
- 7 A transition from "0" to "1" initiates a ZERO command.
- 8 If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- 9 **Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.
- 10 **Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- 11 When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

**Table 7-3: Discrete Write INTEGER or Division Word 1 OUT**

Descriptor Label	Short Name	Data Type
Bits 0-2 Select Weight Mode 0=Gross Weight 1=Net Weight 2=Displayed Weight 3=Tare Weight 4=Setpoint 1 5=Rate 6,7=Reserved	SelectMode	16 Binary
Bit 3 Load Preset Tare on 0 to 1 transition	LoadTare	
Bit 4 Clear Tare on 0 to 1 transition	ClearTare	
Bit 5 Tare Scale on 0 to 1 transition	TareScale	
Bit 6 Print Scale Weight on 0 to 1 transition	PrintWeight	
Bit 7 Zero Scale on 0 to 1 transition	ZeroScale	
Bit 8 Enable Setpoints, 1=Restart, 0=Abort *	EnblSetpoint	
Bits 9-11 Display Mode, command written to pd0119 0=Normal Display Mode 1=Display Literal 1 2=Display Literal 2 3=Display Literal 3 4=Display Literal 4 5=Display Literal 5 6=Start ID1 Sequence 7=Display Literal from Block Transfer display input, written to pd0118 in IND780	DisplayMode	
Bit 12 Discrete Output 1, Option Board Slot 5	DiscreteOut1	
Bit 13 Discrete Output 2, Option Board Slot 5	DiscreteOut2	
Bit 14 Discrete Output 3, Option Board Slot 5	DiscreteOut3	
Bit 15 Load Setpoint 1 Value	LoadSetpoint	

\* A transition from 0 to 1 of the "enable setpoint" bit causes the PLC Messaging to issue a "restart setpoint" command to the setpoint. Then, the setpoint logic temporarily disables its discrete outputs while it reloads the setpoint values from Shared Data and restarts the setpoint. If the setpoint was running at the time of the restart command, this disabling and restarting causes the setpoint's discrete outputs to temporarily shutoff and turn back on. A transition from 1 to 0 of the "enable setpoint" bit causes the PLC Messaging to issue an "abort setpoint" command to the setpoint.

**Table 7-4: Integer Input Assembly Map, IND780 >> PLC**

Offset #	Integer or Division Integer
0	1st Scale Slot (weight)*
2	1st Scale Slot (status)
4	2nd Scale Slot (weight)
6	2nd Scale Slot (status)
8	3rd Scale Slot (weight)
10	3rd Scale Slot (status)
12	4th Scale Slot (weight)
14	4th Scale Slot (status)
16	5th Scale Slot (weight)
18	5th Scale Slot (status)
20	6th Scale Slot (weight)
22	6th Scale Slot (status)

Offset #	Integer or Division Integer
24	7th Scale Slot (weight)
26	7th Scale Slot (status)
28	8th Scale Slot (weight)
30	8th Scale Slot (status)
32	9th Scale Slot (weight)
34	9th Scale Slot (status)
36	10th Scale Slot (weight)
38	10th Scale Slot (status)
40	11th Scale Slot (weight)
42	11th Scale Slot (status)
44	12th Scale Slot (weight)
46	12th Scale Slot (status)
48	Shared Data Access Status
50	Shared Data Read Field Value**
52	Shared Data Read Field Value**
54	Shared Data Read Field Value**
56	Shared Data Read Field Value**
58	Shared Data Read Field Value**
60	Shared Data Read Field Value**
62	Shared Data Read Field Value**
64	Shared Data Read Field Value**
66	Shared Data Read Field Value**
68	Shared Data Read Field Value**

\* Scale slots can be assigned to any combination of scales, including replicating the same scale in multiple slots, in order to get different data such as weight and rate from the same scale concurrently.

\*\* The length of shared data value depends on the type of shared data field requested. However, it cannot exceed 10 words (20 bytes).

**Table 7-5: Integer Output Assembly Map, PLC >> IND780**

Offset#	Integer or Division Integer Format
0	1 <sup>st</sup> Scale Slot Integer Value
2	1 <sup>st</sup> Scale Slot Command
4	2 <sup>nd</sup> Scale Slot Integer Value
6	2 <sup>nd</sup> Scale Slot Command
8	3 <sup>rd</sup> Scale Slot Integer Value
10	3 <sup>rd</sup> Scale Slot Command
12	4 <sup>th</sup> Scale Slot Integer Value
14	4 <sup>th</sup> Scale Slot Command
16	5 <sup>th</sup> Scale Slot Integer Value
18	5 <sup>th</sup> Scale Slot Command
20	6 <sup>th</sup> Scale Slot Integer Value
22	6 <sup>th</sup> Scale Slot Command
24	7 <sup>th</sup> Scale Slot Integer Value
26	7 <sup>th</sup> Scale Slot Command
28	8 <sup>th</sup> Scale Slot Integer Value
30	8 <sup>th</sup> Scale Slot Command

Offset#	Integer or Division Integer Format
32	9 <sup>th</sup> Scale Slot Integer Value
34	9 <sup>th</sup> Scale Slot Command
36	10 <sup>th</sup> Scale Slot Integer Value
38	10 <sup>th</sup> Scale Slot Command
40	11 <sup>th</sup> Scale Slot Integer Value
42	11 <sup>th</sup> Scale Slot Command
44	12 <sup>th</sup> Scale Slot Integer Value
46	12 <sup>th</sup> Scale Slot Command
48	Shared Data Command
50	Shared Data Field Name –terminal name
52	Shared Data Field Name – variable name
54	Shared Data Field Name – variable name
56	Shared Data Field Name – variable name
58	Shared Data Write Value**
60	Shared Data Write Value**
62	Shared Data Write Value**
64	Shared Data Write Value**
66	Shared Data Write Value**
68	Shared Data Write Value**
70	Shared Data Write Value**
72	Shared Data Write Value**
74	Shared Data Write Value**
76	Shared Data Write Value**

\* Scale slots can be assigned to any combination of scales, including replicating the same scale in multiple slots in order to get different data such as weight and rate from the same scale concurrently.

\*\* The length of shared data value is dependent on the type of shared data field. However, it cannot exceed 10 words (20 bytes).

## 7.3.6. Floating Point

### 7.3.6.1. Operational Overview

When the Floating Point format is selected at the IND780, each message slot configured will have four 16-bit words for Read data and three 16-bit words for Write data. Read data refers to the data sent from the IND780 to the PLC (PLC Read). Write Data refers to data sent from the PLC to the IND780 (PLC Write). The first word of the Write data memory map is reserved.

The terminal uses integer commands from the PLC to select the floating point weight output data. The terminal recognizes a command when it sees a new value in the scale slot command word. If the command has an associated floating point value (for example, loading a target value), it must be loaded into the floating point value words before the command is issued. Once the terminal recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. It also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC should wait until it receives the command acknowledgment from the terminal before sending another command.

The terminal has two types of values that it can report to the PLC: real-time and static. When the PLC requests a real-time value, the terminal acknowledges the command from the PLC once but

sends and updates the value at every interface update cycle. If the PLC requests a static value, the terminal acknowledges the command from the PLC once and updates the value once. The terminal will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, feed, and tolerance values are examples of static data.

The terminal can send a rotation of up to nine different real-time values for each message slot. The PLC sends commands to the terminal to add a value to the rotation. Once the rotation is established, the PLC must instruct the terminal to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the terminal to advance to the next value. If the terminal is asked to alternate its output data automatically, it will switch to the next value in its rotation at the next interface update cycle. (The interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate “report next field” commands (1 and 2). When the PLC changes to the next command, the terminal switches to the next value in the rotation. The terminal stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point rotation examples for additional information.

Table 7-5 through Table 7-8 provide detailed information on the floating-point data format. Read data refers to the PLC’s input data and write data refers to the PLC’s output data.

The IND780 supports up to 12 scale slots that it places into one large Floating Point Assembly. Scales can be assigned to Scale Slot Assemblies in any combination, using the IND780 Setup menus. One scale can be assigned to multiple scale slots to get different data for a scale, such as weight and rate, reported concurrently to the PLC. Commands are byte-oriented.

**Table 7-6: Discrete Read Floating Point (float) – IND780 >> PLC**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity 1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
17	Cmd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

**Notes for Table 7-6:**

- 1 The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale's gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale's output word determines what data will be sent.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- 4 **Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- 5 **Bit 5** will be set to a "1" when the associated scale is selected on the IND780 console panel.
- 6 **Bit 6** and **Bit 7** are user defined, and corresponds to user's Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- 7 The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- 8 **Bit 8** is set to a "1" when the ENTER key is pressed on the terminal's keypad. The bit clears to "0" when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- 9 **Bit 9**, **Bit 10**, and **Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 10 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 11 The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- 12 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).
- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 7-7: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale**
8	ALxx02 – SDV, instance by scale**
9	ALxx01 – SDV, instance by scale**

Dec	Data
10	ALxx02 – SDV, instance by scale**
11	Low-pass filter corner frequency, instance by scale
12	Notch filter frequency, instance by scale
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target fine feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size
19–28	Reserved
29	Last terminal error code
30	No data response-command successful
31	No data response-command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* Replace “xx” with the scale instance – e.g. “01” for scale 1.

**Table 7-8: Discrete Write Floating Point (float)—PLC >> IND780**

WORD 0 <sup>1</sup> OUT	WORD 1 <sup>2</sup> OUT	WORD 2 OUT <sup>3</sup> FP load value	WORD 3 OUT <sup>3</sup> FP load value
Reserved	Scale command	X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X
		X	X

**Notes for Table 2-8:**

1. WORD 0 OUT is a reserved 16-bit word, only present in message slot 1.

2. WORD 1 OUT Scale Command: The scale command is used to instruct the terminal what data to send in the discrete read data, to load the floating point data in the write command, and to control the terminal's discrete outputs or display. See the PLC Output Command Table for a list of the available commands and their respective decimal or hex value. Not all commands will require a value in the floating point load value words.
3. WORD 2 & WORD 3 OUT- A 32 bit, single precision floating point value. This value is used with the command in WORD 0 to instruct the terminal to download the floating point value into the field specified by the command.

**Table 7-9: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName	Dec	Hex	Command	SDName
0	00	Report next rotation field @ next interface update cycle <sup>1</sup>		76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
1	01	Report next rotation field <sup>1,2</sup>		78	4e	Disable LCD display <sup>7</sup>	
2	02	Report next rotation field <sup>1,2</sup>		79	4f	Enable LCD display <sup>7</sup>	
3	03	Reset rotation		80	50	Set normal display mode <sup>7,9</sup>	PD0119
10	0a	Report gross weight <sup>1,3</sup>		81	51	Display message 1 <sup>7,9</sup>	PD0119
11	0b	Report net weight <sup>1,3</sup>		82	52	Display message 2 <sup>7,9</sup>	PD0119
12	0c	Report tare weight <sup>1,3</sup>		83	53	Display message 3 <sup>7,9</sup>	PD0119
13	0d	Report fine gross weight <sup>1,3</sup>		84	54	Display message 4 <sup>7,9</sup>	PD0119
14	0e	Report fine net weight <sup>1,3</sup>		85	55	Display Message 5 <sup>7,9</sup>	PD0119
15	0f	Report fine tare weight <sup>1,3</sup>		86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
16	10	Report Rate <sup>1,3</sup>		87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
17	11	Report SDV ALxx01 <sup>6</sup>		88	58	Disable weight display <sup>7</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>		89	59	Enable weight display <sup>7</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>		90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
20	14	Report notch filter frequency <sup>3</sup>		91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>	92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>	93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10	100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12	101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
25	19	Report spill value <sup>3,10</sup>	SPxx09	102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>	103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>	104	68	Master control relay OFF <sup>7, 12</sup>	
29	1d	Report last error <sup>3</sup>		105	69	Master control relay ON <sup>7, 12</sup>	
30	1e	Report primary increment size <sup>3</sup>		110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
40	28	Add gross weight to rotation <sup>7</sup>		111	6f	Set target fine feed value <sup>4,10</sup>	
41	29	Add net weight to rotation <sup>7</sup>		112	70	Set - tolerance value <sup>4,10</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>		113	71	Set target value and apply <sup>4, 10, 14</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>		114	72	Start target logic <sup>7,10,15</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>		115	73	Abort target logic <sup>7,10,15</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>		116	74	Target use gross weight <sup>7,10</sup>	SPxx04
46	2e	Add rate to rotation <sup>7</sup>		117	75	Target use net weight <sup>7,10</sup>	SPxx04
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>	118	76	Target use rate <sup>7,10</sup>	SPxx04
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>	119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
60	3c	Load numeric tare value <sup>4</sup>		120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
61	3d	Pushbutton tare command <sup>7</sup>		121	79	Enable target latching <sup>7,10</sup>	SPxx06
62	3e	Clear command <sup>7</sup>		122	7a	Disable target latching <sup>7,10</sup>	SPxx06
63	3f	Print command <sup>7</sup>		123	7b	Reset target latch <sup>7</sup>	SPxx07
64	40	Zero command <sup>7</sup>		124	7c	Set target spill value <sup>4, 10</sup>	
65	41	Select scale 1 <sup>7</sup>		131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
66	42	Select scale 2 <sup>7</sup>		160	a0	Apply scale setup <sup>7</sup>	QC0149
67	43	Select next scale <sup>7</sup>		161	a1	Write Calibration to EEPROM <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>	162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>	163	a3	Enable Tare on IND780 console <sup>7</sup>	
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>	164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	

Dec	Hex	Command	SDName	Dec	Hex	Command	SDName
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>	165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>	166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
73	49	Set low-pass filter corner frequency <sup>4</sup>		167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
74	4a	Set notch filter frequency <sup>4</sup>		168	a8	Select scale 3 <sup>7</sup>	
75	4b	Reset ENTER key <sup>7</sup>		169	a9	Select scale 4 <sup>7</sup>	

**Notes for Table 7-9:**

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.
10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

**7.3.7. Floating Point Output Assembly Map, PLC to IND780**

**Table 7-10: Floating Point Output Assembly Map**

Offset #	Floating Point Format
0	Reserved
2	1st Scale Slot Command
4	1st Scale Slot Floating Point Value
6	1st Scale Slot Floating Point Value
8	2nd Scale Slot Command
10	2nd Scale Slot Floating Point Value
12	2nd Scale Slot Floating Point Value
14	3rd Scale Slot Command
16	3rd Scale Slot Floating Point Value

Offset #	Floating Point Format
18	3 <sup>rd</sup> Scale Slot Floating Point Value
20	4 <sup>th</sup> Scale Slot Command
22	4 <sup>th</sup> Scale Slot Floating Point Value
24	4 <sup>th</sup> Scale Slot Floating Point Value
26	5 <sup>th</sup> Scale Slot Command
28	5 <sup>th</sup> Scale Slot Floating Point Value
30	5 <sup>th</sup> Scale Slot Floating Point Value
32	6 <sup>th</sup> Scale Slot Command
34	6 <sup>th</sup> Scale Slot Floating Point Value
36	6 <sup>th</sup> Scale Slot Floating Point Value
38	7 <sup>th</sup> Scale Slot Command
40	7 <sup>th</sup> Scale Slot Floating Point Value
42	7 <sup>th</sup> Scale Slot Floating Point Value
44	8 <sup>th</sup> Scale Slot Command
46	8 <sup>th</sup> Scale Slot Floating Point Value
48	8 <sup>th</sup> Scale Slot Floating Point Value
50	9 <sup>th</sup> Scale Slot Command
52	9 <sup>th</sup> Scale Slot Floating Point Value
54	9 <sup>th</sup> Scale Floating Point Value
56	10 <sup>th</sup> Scale Slot Command
58	10 <sup>th</sup> Scale Slot Floating Point Value
60	10 <sup>th</sup> Scale Floating Point Value
62	11 <sup>th</sup> Scale Slot Command
64	11 <sup>th</sup> Scale Slot Floating Point Value
66	11 <sup>th</sup> Scale Floating Point Value
68	12 <sup>th</sup> Scale Slot Command
70	12 <sup>th</sup> Scale Slot Floating Point Value
72	12 <sup>th</sup> Scale Floating Point Value
74	Shared Data Command
76	Shared Data Field Name –terminal name
78	Shared Data Field Name – variable name
80	Shared Data Field Name – variable name
82	Shared Data Field Name – variable name
84	Shared Data Write Value**
86	Shared Data Write Value**
88	Shared Data Write Value**
90	Shared Data Write Value**
92	Shared Data Write Value**
~	~
102	Shared Data Write Value**

- \* You can assign scale slots to any combination of scales, including replicating the same scale in multiple slots, in order to get different data, such as weight and rate, from the same scale concurrently.
- \*\* The length of shared data value is dependent on the type of shared data field. However, it cannot exceed 10 words (20 bytes).

### 7.3.8. Floating Point Input Assembly Map, IND780 to PLC

**Table 7-11: Floating Point Input Assembly Map**

Offset #	Floating Point Field	Offset #	Floating Point Field
0	1st Scale Slot Command response *	54	7 <sup>th</sup> Scale Slot status
2	1st Scale Slot floating point #	56	8 <sup>th</sup> Scale Slot Command response
4	1st Scale Slot floating point #	58	8 <sup>th</sup> Scale Slot floating point #
6	1st Scale Slot status	60	8 <sup>th</sup> Scale Slot floating point #
8	2 <sup>nd</sup> Scale Slot Command response	62	8 <sup>th</sup> Scale Slot status
10	2 <sup>nd</sup> Scale Slot floating point #	64	9 <sup>th</sup> Scale Slot Command response
12	2 <sup>nd</sup> Scale Slot floating point #	66	9 <sup>th</sup> Scale Slot floating point #
14	2 <sup>nd</sup> Scale Slot status	68	9 <sup>th</sup> Scale Slot floating point #
16	3 <sup>rd</sup> Scale Slot Command response	70	9 <sup>th</sup> Scale Slot status
18	3 <sup>rd</sup> Scale Slot floating point #	72	10 <sup>th</sup> Scale Slot Command response
20	3 <sup>rd</sup> Scale Slot floating point #	74	10 <sup>th</sup> Scale Slot floating point #
22	3 <sup>rd</sup> Scale Slot status	76	10 <sup>th</sup> Scale Slot floating point #
24	4 <sup>th</sup> Scale Slot Command response	78	10 <sup>th</sup> Scale Slot status
26	4 <sup>th</sup> Scale Slot floating point #	80	11 <sup>th</sup> Scale Slot Command response
28	4 <sup>th</sup> Scale Slot floating point #	82	11 <sup>th</sup> Scale Slot floating point #
30	4 <sup>th</sup> Scale Slot status	84	11 <sup>th</sup> Scale Slot floating point #
32	5 <sup>th</sup> Scale Slot Command response	86	11 <sup>th</sup> Scale Slot status
34	5 <sup>th</sup> Scale Slot floating point #	88	12 <sup>th</sup> Scale Slot Command response
36	5 <sup>th</sup> Scale Slot floating point #	90	12 <sup>th</sup> Scale Slot floating point #
38	5 <sup>th</sup> Scale Slot status	92	12 <sup>th</sup> Scale Slot floating point #
40	6 <sup>th</sup> Scale Slot Command response	94	12 <sup>th</sup> Scale Slot status
42	6 <sup>th</sup> Scale Slot floating point #	96	Shared Data Access Status
44	6 <sup>th</sup> Scale Slot floating point #	98	Shared Data Read Field Value**
46	6 <sup>th</sup> Scale Slot status	100	Shared Data Read Field Value**
48	7 <sup>th</sup> Scale Slot Command response	102	Shared Data Read Field Value**
50	7 <sup>th</sup> Scale Slot floating point #	~	~
52	7 <sup>th</sup> Scale Slot floating point #	116	Shared Data Read Field Value**

- \* You can assign scale slots to any combination of scales, including replicating the same scale in multiple slots, in order to get different data, such as weight and rate, from the same scale concurrently.
- \*\* The length of shared data value is dependent on the type of shared data field requested. However, it cannot exceed 10 words (20 bytes).

7.3.8.1. Floating Point Data Format and Compatibility

In Floating Point Message mode, the PLC and terminal exchange weight, target, and tare data in single-precision floating point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

The IND780 uses two bits to maintain data integrity when communicating to the PLC. One bit is in the beginning byte of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the IND780 is freely sending weight updates. If the PLC program detects this condition, it should send a new command to the IND780.

7.3.8.1.1. Floating Point Command Examples

Table 7-12 through Table 7-15 provide floating point command examples where the IND780 terminal is configured as node 3, using input and output words starting at address 10.

**Table 7-12: Data Requirement: Only Net Weight Sent (continuously), Message Slot 1**

Step #	Action	Scale Command (from PLC)	Scale Floating Point Value	Command Response from IND780	Floating Point Value
1	PLC sends command to IND780 to report net weight	11 (dec) loaded into command word QW OR WY:11	none required		
2	IND780 sees new command			Command ack. =1 F.P. ind. = 1 (net)	Net weight in floating point

As long as the PLC leaves the 11 (dec) in the command word the IND780 will update the net value every interface update cycle.

**Table 7-13: Data Requirement: Load Target 1 Cutoff Value = 21.75 for Scale 1**

Step #	Action	Scale command (from PLC)	Scale Floating Point Value	Command response from IND780	Floating Point Value
1	PLC loads floating point value first		floating point value = 21.75		
2	PLC sends command to set target 1 cutoff value	110 (dec) loaded into command word QW OR WY:11	floating point value = 21.75		
3	IND780 sees new command, loads			Command ack. = 1	Floating point value = 21.75

Step #	Action	Scale command (from PLC)	Scale Floating Point Value	Command response from IND780	Floating Point Value
	the value into the target and sends a return message to indicate the new target value			F.P. ind = 13	
4	PLC instructs IND780 terminal to start "using" new target value	114 (dec) loaded into command word QW OR WY:11			
5	IND780 sees new command			Command ack. = 2 F.P. ind = 30	(null value)

The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780. After the PLC finishes loading its target value, it can resume monitoring the weight information it requires by sending a command to report some type of weight or set up a rotation of reported data.

**Table 7-14: Data Requirement: Rotation of Gross Weight and Rate Updated on interface Update Cycle**

Step #	Action	Scale Command (from PLC)	Scale Floating Point Value	Command Response from IND780	Floating Point Value
1	PLC clears out any previous rotation with reset	3 (dec) loaded into command word QW OR WY:11			
2	IND780 sees new command			Command ack.= 1 F.P. ind = 30	
3	PLC adds gross weight to rotation	40 (dec) loaded into command word QW OR WY:11	(null value)		
4	IND780 sees new command			Command ack. = 2 F.P. ind = 30	
5	PLC adds rate to the rotation	46 (dec) loaded into command word QW OR WY:11	RESERVED for Future Use		
6	IND780 sees new command			Command ack. = 3 F.P. ind = 30	(null value)
At this point, the rotation has been set up. Now the PLC needs to command the IND780 to begin the rotation.					
7	PLC sends the command to begin the rotation at interface update	0 (dec) loaded into command word QW OR WY:11			
8	IND780 sends gross weight at Interface update cycle ~ 60 msec			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.

Step #	Action	Scale Command (from PLC)	Scale Floating Point Value	Command Response from IND780	Floating Point Value
9	PLC leaves 0 in command word and IND780 sends the rate value at next interface update	0 (dec) loaded into command word QW OR WY:11	RESERVED For Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
10	PLC leaves 0 in command word and IND780 sends the gross value at next interface update	0 (dec) loaded into command word QW OR WY:11		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11	PLC leaves 0 in command word and IND780 sends the rate value at the next interface update	0 (dec) loaded into command word QW OR WY:11	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
<p>This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.</p>					

- Refer to the IND780 Terminal Shared Data Reference (document 64059110), provided on the IND780 documentation CD (part number 64057241).

**Table 7-15: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Action	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1	PLC clears out any previous rotation with reset	3 (dec) loaded into command word			
2	IND780 sees new command			Command ack. = 1 F.P. ind = 30	
3	PLC adds net weight to rotation	41 (dec) loaded into command word	(null value)		
4	IND780 sees new command			Command ack. = 2 F.P. ind = 30	
5	PLC adds rate to the rotation	46 (dec) loaded into command word QW OR WY:11	RESERVED for Future Use		
6	IND780 sees new command			Command ack. = 3 F.P. ind = 30	(null value)
7	PLC sends the command to report the first field in the rotation	1 (dec) loaded into command word			

Step #	Action	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
--------	--------	--------------------------	----------------------------	--------------------------------	----------------------

At this point, the rotation has been set up. Now the PLC needs to send commands to the IND780 to begin the rotation and advance to the next value when required.

8	IND780 acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
9	PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 does NOT see this as another command to report the next rotation field	2 (dec) loaded into command word			
10	IND780 acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
11	PLC sends the command to report the next field in the rotation	1 (dec) loaded into command word			
12	IND780 acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field			Command ack. = 3 F.P. ind = 1	Floating point value = net wt.
13	PLC sends the command to report the next field	2 (dec) loaded into command word			

Step #	Action	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
14	IND780 acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate

At approximately every 60 msec the IND780 updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value.

### 7.3.9. Floating Point Numbers

The Simatic TI505 PLCs support the IEEE Standard floating point numbers. According to the Simatic TI505 Programming Reference Manual, real numbers are stored in the single-precision 32-bit format, per ANSI/IEEE Standard 754-1985, in the range 5.42101070 E-20 to 9.22337177 E18.

Siemens S5 PLCs do not inherently support the IEEE-format floating point numbers. S5 PLCs do support floating point numbers in their own unique format. A software “function block” can be implemented in the S5 PLC to convert between the S5 floating point numbers and the IEEE Standard floating point numbers.

Siemens S7 PLCs support the IEEE Standard floating point numbers.

### 7.3.10. Shared Data

#### 7.3.10.1. Operational Overview

PROFIBUS PLCs can access the terminal’s Shared Data. Since PROFIBUS communication supports up to 244-byte messages at speeds typically in the range of 1.5 to 12 megahertz, there is no need for two separate modes of communication, unlike Allen-Bradley and its block transfer. PROFIBUS PLCs can read IND780 Shared Data variables, write new values to IND780 Shared Data variables, and write operator messages on the terminal’s lower display.

For PROFIBUS, the PLC output data has additional fields for accessing Shared Data. The PLC must specify the Shared Data command and variable name in the PLC output message. If the command is a write command, then the PLC output message must also contain the write field value. The maximum length of the value is 20 bytes. When the Shared Data command is a read command, the PLC input message will have a read field containing the data from the Shared Data variable specified in the output message. The maximum length of the data reported in the read field is 20 bytes. The Shared Data variables are self-typing. The IND780 terminal determines the type of any valid data field in the message from the variable’s name and definition in Shared Data. The terminal will not allow string data to be written in a floating point variable or vice versa.

#### 7.3.10.2. Shared Data Input

The input information for the shared data consists of two sections: The shared data status and the shared data read field value (if requested by the shared data output command). The shared data

status information is a word that contains an integer value. This integer value represents one of the following status values:

0	Null status
1	Command completed successfully
2	Invalid shared data name
3	Invalid shared data command
4	Cannot write because field is write-protected (legal for trade)
5	Invalid Data Format
6	Cannot Access Remote Terminal
7	Invalid Service Request
8	Messaging not Enabled
9	Access Denied

The shared data read field value contains the value of the shared data variable specified in the shared data output (from the PLC to the terminal). It is only present when the command from the shared data output requests read shared data. This value is self-typing; for example, it could be a floating point number or a string variable. The length is determined by the variable selected but will not exceed 20 bytes. See the tables following the Shared Data Output for a list of possible variables and their contents.

#### 7.3.10.3. Shared Data Output

The output information for the shared data consists of four sections: the shared data command, the shared data name, the shared data variable name, and the shared data write value (if required by the shared data output command). The shared data command information is a word that contains an integer value. This integer value represents one of the following status values:

0	Null command
1	Read shared data
2	Write shared data

The terminal processes a shared data command "on demand" by the PLC. When a new value is placed in the shared data command word, the terminal performs the command issued. The terminal does not provide real time information to the PLC; it supplies a snapshot of the data, not an automatic update of new values of the same shared data command. Instead, the PLC must request the information again by setting a new value in the shared data command word.

To do successive reads, for example, the PLC must alternate between a "null" command and a "read" command in the shared data command word. For the most efficient processing, the PLC should set up the terminal name, the variable name, and the write value (if any) while it is setting the "null" command. Once that is completed, the PLC can then set the shared data command to "read" or "write".

For a complete listing of Shared Data Fields, refer to the IND780 Shared Data Reference provided on the IND780 documentation CD, part number 64057241

### 7.3.11. Controlling Discrete I/O Using a PLC Interface

The IND780 terminal provides the ability to directly control its discrete outputs and read its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the IND780 discrete I/O updates are synchronized with the terminal's interface update rate, not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Note that the outputs must be unassigned in the IND780 terminal setup in order to be controlled by the PLC

## 7.4. Hardware Setup

### 7.4.1. Wiring

The IND780 terminal's PROFIBUS option card has a DB-9 connector to connect to the PROFIBUS network interface (Figure 7-5). Cable distance, type, and termination are specified by PROFIBUS. (See the PLC documentation for cable design guidelines for the various PLCs.)

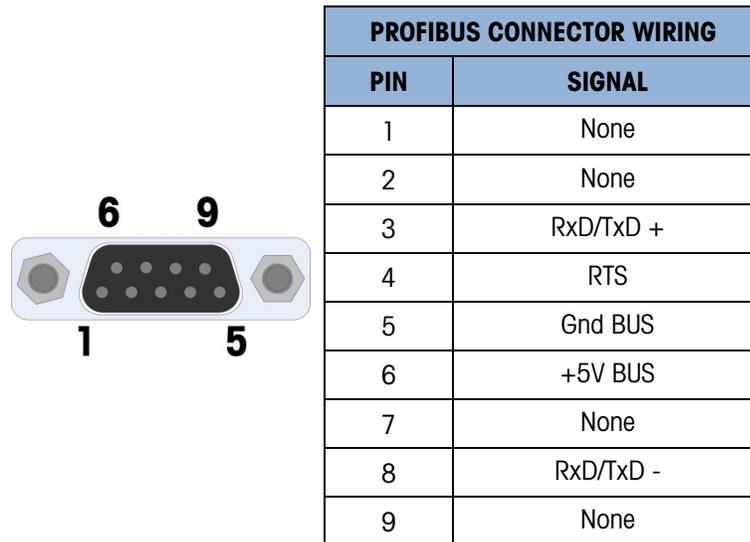


Figure 7-5: PROFIBUS Option Card DB-9 Connector Termination

The IND780 harsh unit requires a right angle connector, Siemens part number 6ES7 972-0BA12-OXA0. The panel mount can use the right angle connector, or a straight connector, METTLER TOLEDO part number 64054361.

## 7.5. Software Setup

When the IND780 terminal detects the presence of a PROFIBUS option card, the PROFIBUS screen is enabled in setup at **Communication > PLC Interface > PROFIBUS** (Figure 7-6). To format PLC data, access **Communication > PLC Interface > Data Format** (Figure 7-7). Default settings are indicated with an asterisk. You must enter setup and configure each scale that is interfaced with the PROFIBUS network.



**Figure 7-6: PROFIBUS Setup Block**

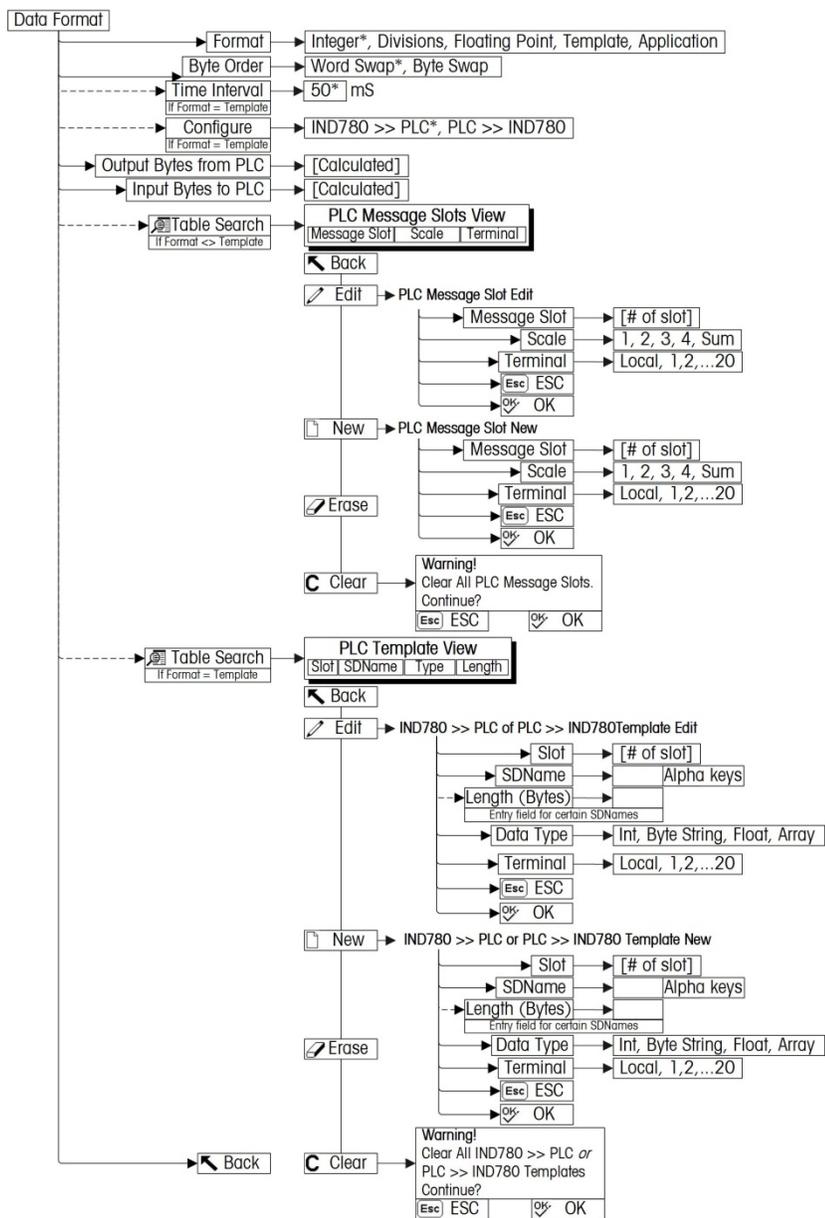


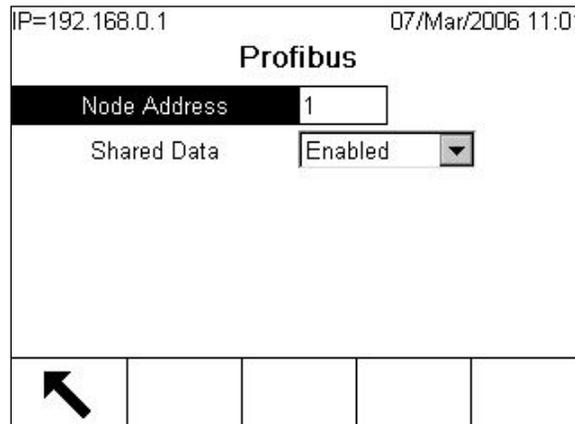
Figure 7-7: PLC Data Format Setup Block

## 7.5.1. PROFIBUS Setup

### 7.5.1.1. PROFIBUS Setup

The PLC Setup block lets you specify how the PROFIBUS interface is used. Several options are available to correspond with your system setup.

1. Select **Communication > PLC Interface > PROFIBUS** (Figure 7-8).

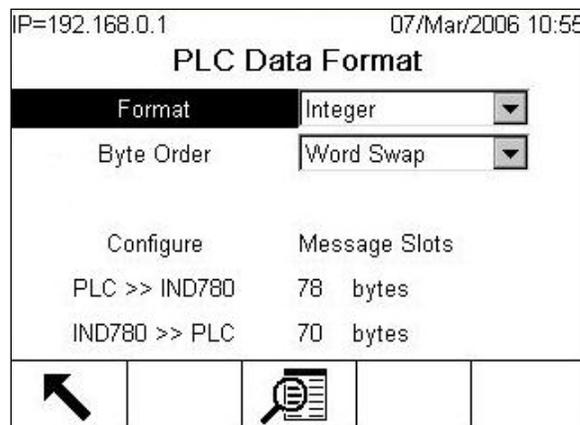


**Figure 7-8: PROFIBUS Setup Screen**

2. Enter the Node Address (0–125).
  3. Select Shared Data Enabled or Disabled.
- Refer to the Discrete Read and Discrete Write tables in this manual for additional information on mapping of Discrete read data to the PLC.

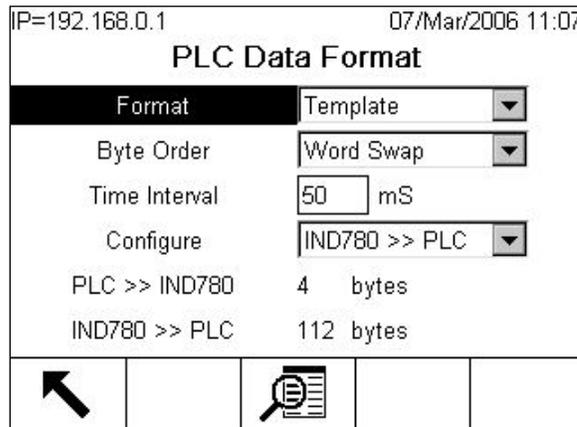
### 7.5.1.2. Data Format Setup

1. Select **Communication > PLC Interface > Data Format** (Figure 7-9).



**Figure 7-9: PROFIBUS PLC Data Format Screen**

- Figure 7-10 shows the screen with Template selected as the format, revealing the Time Interval, Configure, and directional displays. When Application format is selected, only the Format box appears on this screen.



**Figure 7-10: PLC Data Format Screen**

- Select the Format (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the Format will delete any existing Message Slots.

**Integer** Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).

**Divisions** Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.

**Floating Point** Displays weight in floating point data format

**Template** Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed "on the fly" by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the Shared Data Reference Manual (document number 64059110) provided on the CD-ROM supplied with your terminal.

**Application** Allows a TaskExpert program to interact with the PLC input and output data. The TaskExpert program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

- Select Byte Order: [default Word Swap]

**Word Swap** Takes the IEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with RSLogix 5000 processors.

**Byte Swap** Makes the floating point format compatible with S7 PROFIBUS.

- Set the **Time Interval** in milliseconds. (Only when the Format is Template)
- The bottom of this screen (Figure 7-9) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.
- Use the **Configure** drop-down list to select the direction of data flow – IND780 >> PLC or PLC >> IND780. (Only when the Format is Template)

- Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of Message Slots (1, 2, 3, 4,... 12) [default is 0], or to configure the Template for communications between the IND780 and the PLC. Figure 7-11 shows the Message Slots View, which is viewed whenever the Format is **not** Template.

IP=172.18.54.111 06/Dec/2007 16:25

**PLC Message Slots View**

Message	Scale	Terminal
1	1	Local
2	- End -	







**Figure 7-11: PLC Message Slots View Screen**

Figure 7-12 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1 07/Mar/2006 11:08

**PLC >> IND780 Template View**

Slot	SDName	Type	Length
1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		







**Figure 7-12: PLC Template View**

- Press the EXIT softkey  to exit the view screen.
- Press the EDIT softkey  to edit an existing slot.
- Press the NEW softkey  to enter a new slot.
- Press the DELETE softkey  to delete an existing slot.
- Press the CLEAR softkey  to delete all existing slots.

Figure 7-13 shows the screen that opens when NEW is selected from the Message Slots View screen (Figure 7-11).

IP=192.168.0.1 07/Mar/2006 11:36

**PLC Message Slot New**

Message Slot 1

Scale 1

Terminal Local

Esc [Left Arrow] [Right Arrow] OK

**Figure 7-13: New Message Slot Screen**

- Scale** Enter the scale number to be associated with the new Message Slot.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 7-14 shows a series of Template New screens, as accessed from the Template View screen (Figure 7-12).

P=172.18.54.111 20/Dec/2007 16:24

**PLC >> IND780 Template New**

Slot 1

SDName

Length (Bytes) 2

Data Type Int

Terminal Local

P=172.18.54.111 20/Dec/2007 16:26

**PLC >> IND780 Template New**

Slot 1

SDName wc0101

Length (Bytes) 2

Data Type Int

Terminal Local

P=172.18.54.111 20/Dec/2007 16:19

**IND780 >> PLC Template New**

Slot 1

SDName

Length (Bytes) 4

Data Type Float

Terminal Local

P=172.18.54.111 20/Dec/2007 16:22

**IND780 >> PLC Template New**

Slot 1

SDName wt0110

Length (Bytes) 4

Data Type Float

Terminal Local

**Figure 7-14: New Template Screens, PLC to IND780 and IND780 to PLC**

- Slot** Automatically assigned.

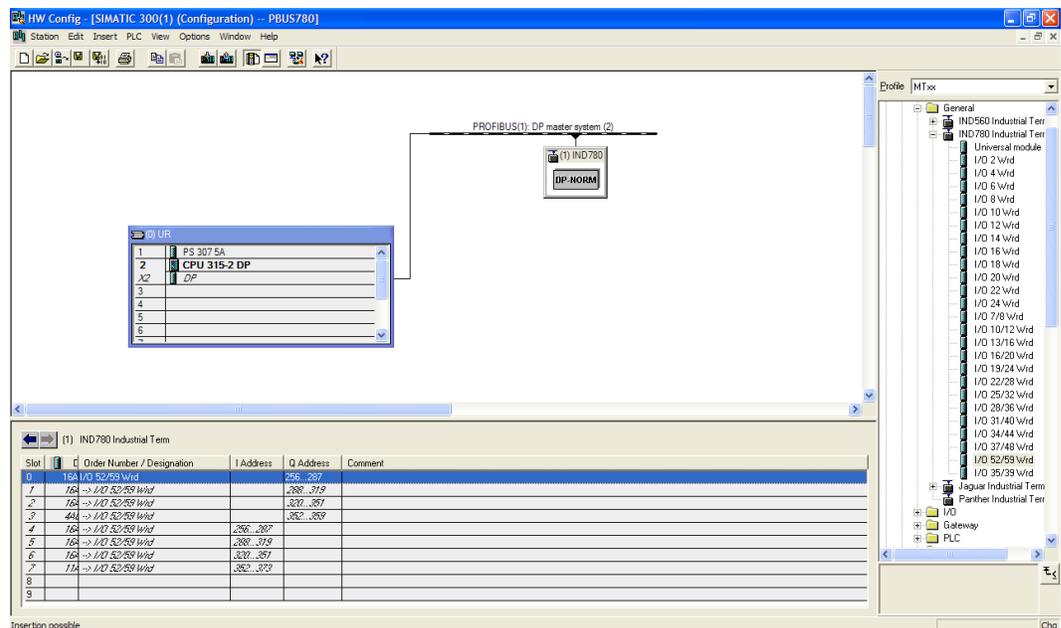
- SDName** An alphabetic entry field used to define the Shared Data variable associated with the slot.
- Length (Bytes) and Data Type** Automatically displayed once the SDName has been entered. The maximum length of a Shared Data variable cannot exceed 16 bytes when Block Transfer is disabled, or 14 bytes when Block transfer is enabled.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

For a complete listing of Shared Data Fields, refer to the **IND780 Shared Data Reference**. Refer to Table 7-6 (Discrete Read) and Table 7-8 (Discrete Write) in this manual for additional information on mapping of discrete read data to the PLC.

### 7.5.2. PROFIBUS GSD or Type Files

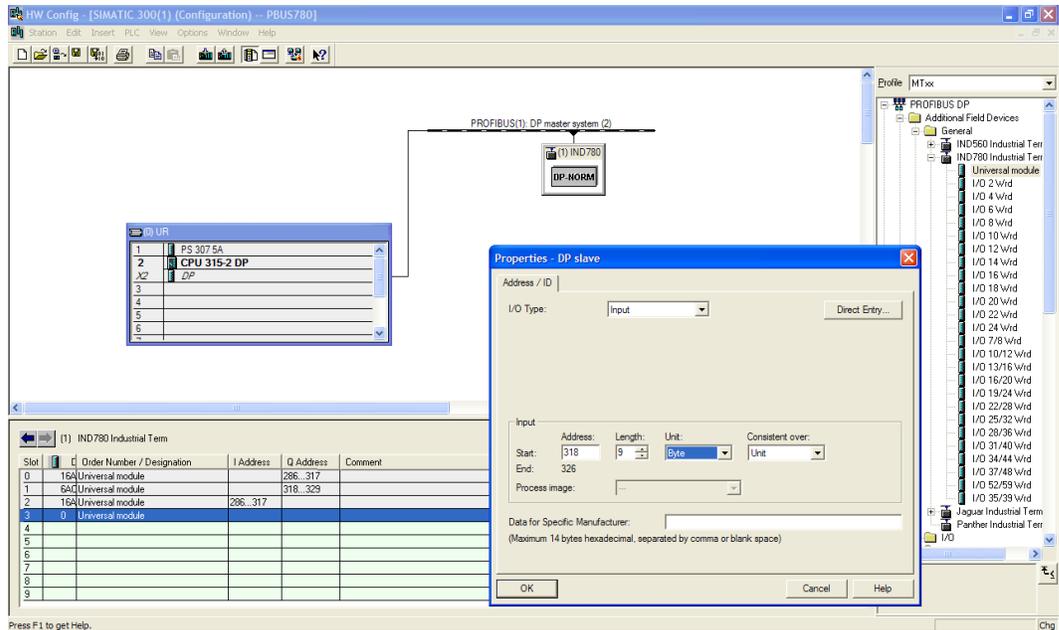
There are twenty six standard configurations of the PROFIBUS GSD or type files for the IND780 terminal's different combinations of data formats for Integer, Division and Floating Point. The length of the messages is different for each of these data formats. The type file selected for these configurations should match the message lengths shown on the PLC Data Format Screen.

- **Note:** The Utilities folder of the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.



**Figure 7-15: PROFIBUS DP Network Configuration**

If the IND780 is set up for the Template or Application Mode, the programmer may have to build a custom PLC Block using the GSD module type "Universal Module". When building this Block the Input/Output "Unit" Parameter must be set for "Word". A "slot" within the block should not be greater than 16 words. If more than 16 words are required, add another "slot".



**Figure 7-16: PROFIBUS DP Slave Properties**

The PROFIBUS GSD files for the IND780 are available free of charge. They are included on the documentation CD (part number 64057241).

## 7.6. Troubleshooting

If the IND780 does not communicate with PLC do the following:

- Check wiring and network termination.
- Confirm that the IND780 settings for data type and rack assignment match those in the PLC.
- Replace the PROFIBUS interface kit if the problem persists.
- If the communication kit was changed from another type, like A-B RIO, ControlNet or EtherNet I/P, a master reset of the IND780 must be performed.

## 7.7. PROFIBUS Option Kit

There are two PROFIBUS options. CIMF part number 71209096 has a vertical connector and is designed to work with a panel mounted IND780. CIMF part number 71209097 has a horizontally mounted connector and is designed to work with a harsh enclosure IND780. No spare parts are associated with the PROFIBUS option kit. Table 7-16 shows what each kit contains.

**Table 7-16: PROFIBUS Option Kit**

Description	Qty.
Installation Instructions	1

Description	Qty.
PCB Package	1
Cylindrical ferrites	3
M3 x 8 screws	3

# 8 PROFINET

## 8.1. Overview

PROFINET is an open industrial networking standard that was developed by Siemens as an Ethernet replacement for its widely popular PROFIBUS Network. The network supports Cyclic and Acyclic messaging, both of which have been implemented in the IND780. The PROFINET Standard is supported and Maintained by the PROFIBUS and PROFINET International (PI) Organization.

PROFINET utilizes commercial, off-the-shelf Ethernet hardware (for example, switches and routers) and is fully compatible with the Ethernet TCP/IP protocol suite.

The IND780 PROFINET option implements PROFINET IO for cyclic Data exchange with the PLC, and uses Acyclic messages for Shared Data Access by the PLC.

The PROFINET option enables the IND780 terminal to communicate to PROFINET enabled Programmable Logic Controllers (PLCs) through direct connection to the PROFINET network at 100 MBPS speed. The option consists of an internal module and internal software to implement the data exchange.

The IND780 appears as a block of I/O data on the PROFINET bus. The IND780 terminal will determine the number of input and output words needed for the number of configured message slots and/or the chosen data format. The number of words required is displayed on the PLC Data Format screen of the IND780.

The IND780 PROFINET GSDML file has 23 different blocks of I/O defined. Blocks are defined for the various Integer/Division/Floating Point modes. Each block is identified as the number of input and output words configured within the block.

Figure 8-1 and Figure 8-2 show a PROFINET module and its components. Note that the module's address is set in software.

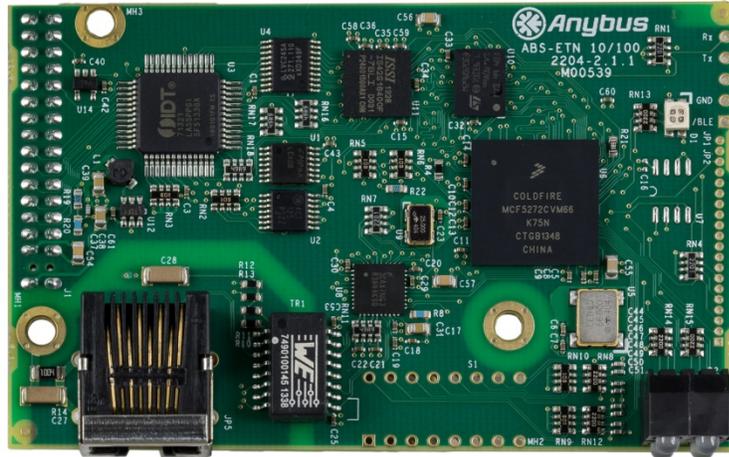


Figure 8-1: PROFINET Module

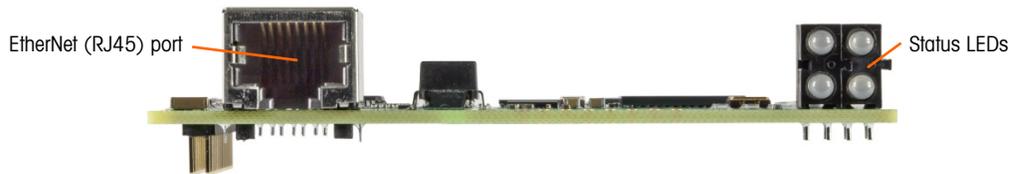


Figure 8-2: PROFINET Module Components

### 8.1.1. Definition of Terms

The following terms are used in this document.

Table 8-1: PROFINET Definition of Terms

Term	Definition
DAP	Device Access Point
DCP	Discovery and basic Configuration Protocol. Used for IP configuration over PROFINET.
DHCP	De-facto standard for dynamic IP address management
GSDML	XML-based descriptive language for GSD-files
Initial Record Data	Record Data write-requests destined for a sub-module. Comparable to PROFIBUS-DP User Parameter Data.
IOCS	IO Consumer Status
IOPS	IO Provider Status
IO Controller	Controlling device which acts as a client for several IO devices. Usually a PLC. Comparable to a PROFIBUS-DP Class 1 master.
IO Device	Field device assigned to an IO Controller. Comparable to a PROFIBUS DPV1 slave.
IO Supervisor	Programming device with commissioning and diagnostic functions. Comparable to a PROFIBUS-DP Class 2 master.
Module	Hardware or logical component of a network device.
Submodule	Hardware or logical component of a module

Term	Definition
PDEV	Physical DEvice. From specification version 2.0 it is possible to describe the physical Ethernet interface and its ports (PDEV, or Physical Device) with a special mechanism. This is done with special sub-modules at slot 0 (the module at slot 0 is the access point for the device).
PNIO	Short for PROFINET IO.
PROFINET IO	PROFINET IO is a communication concept for the implementation of modular, decentralized applications. Comparable to PROFIBUS-DP, where I/O data of field devices are cyclically transmitted to the process image of a PLC. The real time capabilities of PROFINET IO are further divided into RT and IRT (see below).
PROFINET IO RT	PROFINET IO with Real Time capabilities. Optimized real time communication channel for time critical I/O data and Alarms. Implemented in software.
PROFINET IRT	PROFINET IO with Isochronous Real Time capabilities. Necessary for motion control application which require an update rate of 1 ms, or less, with no jitter. Implemented in hardware.
PROFINET CBA	PROFINET Component Based Automation. Comparable to PROFIBUS FMS.
Record Data	Comparable to PROFIBUS DPV1 acyclic Read/Write.

### 8.1.2. Communications

The IND780 terminal uses component parts to ensure complete compatibility with the Siemens PROFINET network. An IND780 terminal is recognized as a generic PROFINET device by the PLC.

### 8.1.3. IP Address

Each PROFINET option represents one physical IP Address. This address is chosen by the system designer, and then programmed into the IND780 terminal and PLC, or the address can be assigned by the PLC using DCP or DHCP and MAC Address association. The IND780 terminal's address is programmed at **Communication > PLC Interface > PROFINET** in the terminal's setup menu. IND780 IP Address entry must be unique for each IND780.

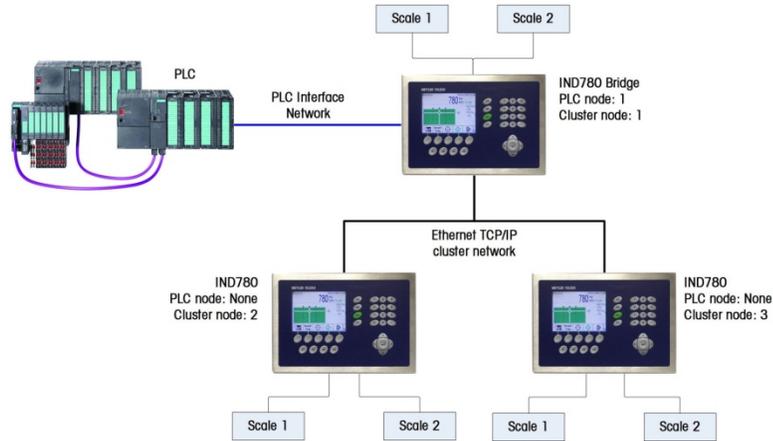
### 8.1.4. Acyclic Messaging

Unlike the PROFIBUS implementation, PROFINET has an acyclic messaging capability that is used for Shared Data Access. Access to Shared Data is done in a manner that is very similar to the method used by ControlNet and Ethernet/IP.

## 8.2. Sharing a PLC Interface

### 8.2.1. Network Topology

It is possible for a PLC interface to be concurrently shared between multiple clustered terminals. Figure 8-3 shows a network topology where a PLC interface located in one IND780 terminal, known as the bridge, is shared across the terminal's Ethernet TCP/IP cluster.



**Figure 8-3: Terminal Cluster Network Topology**

Up to 20 terminals can be connected in the cluster. However, the number of message slots available for communications depends on the type of PLC interface. All available PLC message slots can be configured to send or receive data to or from the remote terminals via the cluster. Refer to the appropriate PLC chapters in the IND780 PLC Interface Manual for further details on message slots' limitations.

To allow bridging of the PLC interface, first a cluster network must be set up among the IND780 terminals. Once the cluster is established, configure the PLC interface on the bridge terminal, set up its data format and assign the required local or remote scale's data to the message slots. Figure 8-4 shows an example of two message slots, the first being allocated to scale 1 of the local bridge terminal and a second slot to scale 1 of the remote terminal node 6. For details on configuring the PLC interface in a cluster terminal, refer to the Software Setup section, below.

The use of PLC interface sharing is not recommended when real time weight or rate information is required by the PLC for fast process control. The determinism of the terminal's PLC interface data update rate is influenced and limited by the characteristics of Ethernet TCP/IP communications used in the cluster network.

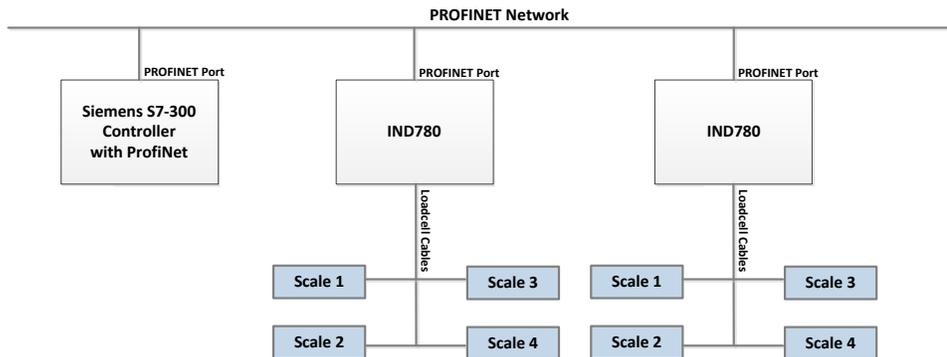
P=172.18.54.122 09/Oct/2007 17:20

PLC Message Slots View		
Message	Scale	Terminal
1	1	Local
2	1	6
▶ 3	- End -	

Navigation icons: back, edit, save, delete, and clear (C).

**Figure 8-4: PLC Message Slots View Screen**

Figure 8-5 shows a typical PROFINET network arrangement, with IND780 terminals connected directly to a Siemens S7 controller **without** the use of the terminal's PLC interface sharing.



**Figure 8-5: Typical Network Topology**

## 8.3. Data Definition

### 8.3.1. Data Formats

The terminal's PROFINET option card has five types of data exchanges that may be selected. The data types are: Integer, Divisions, Floating Point, Template and Application.

In Integer/Divisions/Floating Point mode, each message slot selected to pass data through the terminal's PROFINET option has its own assigned input and output words for continuous communication to and from the PLC.

In Template mode, the designer selects the types of data – and, in the case of string data, the length of the data – contained within each data slot.

In Application mode, a Task Expert program is written to determine the information contained within the input and output messages to the PLC. The formatting of the data is set up within the application program.

### 8.3.2. Data Integrity

The IND780 has specific bits to allow the PLC to confirm that data was received without interruption and that the IND780 is not in an error condition. It is important to monitor these bits. Any PLC code should use them to confirm the integrity of the data received by the IND780. Refer to the data charts for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC, if the PLC program detects this condition, it should send a new command to the terminal.

### 8.3.3. Discrete Data

Five formats of discrete data are available with the PROFINET Kit option: Integer, divisions, floating point, template and application. Only one type of data format may be selected and used by the IND780 at one time.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word (refer to Table 8-6 and Table 8-7 for an explanation) numerical values.

The floating-point format allows bi-directional communication of discrete bit encoded information and numeric data encoded in IEEE 754, single precision floating point format.

The data format of discrete data will affect the data size required in the configuration of the PLC. The IND780 console PLC message slot setup screen provides data size requirements in bytes, except for the Application data format.

Selection of the appropriate format depends on issues such as the range or capacity of the scale used in the application. The integer format can represent a numerical value up to 32,767. The division format can represent a value up to 32,767 scale divisions or increments. The floating-point format can represent a value encoded in IEEE 754, single precision floating point format.

Floating point is the only data format that includes decimal point information. Integer and division formats ignore decimal points. Accommodation of decimal point location must take place in the PLC logic, when it is needed with these formats.

Changing the Data Format to be used by the IND780 will clear all Message Slots. Data format is set up in the **Communication > PLC Interface > Data Format** screen.

8.3.3.1.1. Examples

<b>250 x .01 scale</b>				
IND780 Displays:	0	2.00	51.67	250.00
Format sent:				
Integer	0	200	5167	25000
Division	0	200	5167	25000
Floating Point	0	2.00	51.67	250.00

Any of the formats could be used in this case.

<b>50,000 x 10 scale</b>				
IND780 Displays:	0	200	5160	50000
Format sent:				
Integer	0	200	5160	-(15536)
Division	0	20	516	5000
Floating Point	0	200	5160	50000

The integer format could not be used because it would send a negative value once the weight exceeded 32,767.

150 x .001 scale				
IND780 Displays:	0	2.100	51.607	150.000
Format sent:				
Integer	0	2100	-(13929)	18928
Division	0	2100	-(13929)	18928
Floating Point	0	2.100	51.607	150.000

The integer and division formats could not be used because they would send a negative value once the weight exceeded 32.767.

### 8.3.4. Byte Order

**Word Swap** takes the IEEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with Rockwell ControlLogix processors.

**Byte Swap** makes the floating point format compatible with Siemens S7 processors.

### 8.3.5. Message Slots

There may be up to 12 message slots for discrete data transfer. cyclic messaging, in Integer, Divisions and Floating Point Data Formats. Each message slot is assigned to a local or remote scale and scales may be repeated in additional message slots. Remote scales must reside in an IND780 Ethernet TCP clustered to the IND780 containing the Ethernet I/P interface. The integer and division formats provide two 16-bit words of input and two 16-bit words of output data per Message Slot. Each Message Slot's first input word provides scale weight data and the input weight data may be selected by the PLC using the Message Slot's second output word bit 0, bit 1 and bit 2. The following two Tables provide input and output usage information.

**Table 8-2: PROFINET PLC Input Data and Data Usage (Integer and Division)**

PLC Input Bytes	PLC Input Words	Usage
0	0	Message Slot 1 Weight Data
1		Message Slot 1 Weight Data
2	1	Message Slot 1 Scale Status
3		Message Slot 1 Scale Status
4	2	Message Slot 2 Weight Data
5		Message Slot 2 Weight Data
6	3	Message Slot 2 Scale Status
7		Message Slot 2 Scale Status
8	4	Message Slot 3 Weight Data
9		Message Slot 3 Weight Data
10	5	Message Slot 3 Scale Status
11		Message Slot 3 Scale Status
12	6	Message Slot 4 Weight Data
13		Message Slot 4 Weight Data
14	7	Message Slot 4 Scale Status
15		Message Slot 4 Scale Status

PLC Input Bytes	PLC Input Words	Usage
16	8	Message Slot 5 Weight Data
17		Message Slot 5 Weight Data
18	9	Message Slot 5 Scale Status
19		Message Slot 5 Scale Status
20	10	Message Slot 6 Weight Data
21		Message Slot 6 Weight Data
22	11	Message Slot 6 Scale Status
23		Message Slot 6 Scale Status
24	12	Message Slot 7 Weight Data
25		Message Slot 7 Weight Data
26	13	Message Slot 7 Scale Status
27		Message Slot 7 Scale Status
28	14	Message Slot 8 Weight Data
29		Message Slot 8 Weight Data
30	15	Message Slot 8 Scale Status
31		Message Slot 8 Scale Status
32	16	Message Slot 9 Weight Data
33		Message Slot 9 Weight Data
34	17	Message Slot 9 Scale Status
35		Message Slot 9 Scale Status
36	18	Message Slot 10 Weight Data
37		Message Slot 10 Weight Data
38	19	Message Slot 10 Scale Status
39		Message Slot 10 Scale Status
40	20	Message Slot 11 Weight Data
41		Message Slot 11 Weight Data
42	21	Message Slot 11 Scale Status
43		Message Slot 11 Scale Status
44	22	Message Slot 12 Weight Data
45		Message Slot 12 Weight Data
46	23	Message Slot 12 Scale Status
47		Message Slot 12 Scale Status

**Table 8-3: PROFINET PLC Output Words and Word Usage (Integer and Division)**

PLC Output Bytes	PLC Output Words	Usage
0	0	Message Slot 1 Command Data
1		Message Slot 1 Command Data
2	1	Message Slot 1 Scale Command
3		Message Slot 1 Scale Command
4	2	Message Slot 2 Command Data
5		Message Slot 2 Command Data

PLC Output Bytes	PLC Output Words	Usage
6	3	Message Slot 2 Scale Command
7		Message Slot 2 Scale Command
8	4	Message Slot 3 Command Data
9		Message Slot 3 Command Data
10	5	Message Slot 3 Scale Command
11		Message Slot 3 Scale Command
12	6	Message Slot 4 Command Data
13		Message Slot 4 Command Data
14	7	Message Slot 4 Scale Command
15		Message Slot 4 Scale Command
16	8	Message Slot 5 Command Data
17		Message Slot 5 Command Data
18	9	Message Slot 5 Scale Command
19		Message Slot 5 Scale Command
20	10	Message Slot 6 Command Data
21		Message Slot 6 Command Data
22	11	Message Slot 6 Scale Command
23		Message Slot 6 Scale Command
24	12	Message Slot 7 Command Data
25		Message Slot 7 Command Data
26	13	Message Slot 7 Scale Command
27		Message Slot 7 Scale Command
28	14	Message Slot 8 Command Data
29		Message Slot 8 Command Data
30	15	Message Slot 8 Scale Command
31		Message Slot 8 Scale Command
32	16	Message Slot 9 Command Data
33		Message Slot 9 Command Data
34	17	Message Slot 9 Scale Command
35		Message Slot 9 Scale Command
36	18	Message Slot 10 Command Data
37		Message Slot 10 Command Data
38	19	Message Slot 10 Scale Command
39		Message Slot 10 Scale Command
40	20	Message Slot 11 Command Data
41		Message Slot 11 Command Data
42	21	Message Slot 11 Scale Command
43		Message Slot 11 Scale Command
44	22	Message Slot 12 Command Data
45		Message Slot 12 Command Data
46	23	Message Slot 12 Scale Command

PLC Output Bytes	PLC Output Words	Usage
47		Message Slot 12 Scale Command

The floating point format provides four 16-bit words of input data and three 16-bit words of output data) per Message Slot. Refer to Table 8-4 and Table 8-5 for details.

The number of Message Slots is set up in **Communication > PLC Interface > Data Format** setup menu (Figure 8-10).

**Table 8-4: PROFINET PLC Floating Point Input Words**

PLC Output Words	Bits 0 – 7	Bits 8 – 15
0	Message Slot 1, Reserved	Message Slot 1, Command Response; refer to Message Slot Table
1	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
2	Message Slot 1 Floating Point data	Message Slot 1 Floating Point data
3	Message Slot 1, Scale Status; refer to Message Slot Table	Message Slot 1, Scale Status; refer to Message Slot Table
4	Message Slot 2, Reserved	Message Slot 2, Command Response; refer to Message Slot Table
5	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
6	Message Slot 2 Floating Point data	Message Slot 2 Floating Point data
7	Message Slot 2, Scale Status; refer to Message Slot Table	Message Slot 2, Scale Status; refer to Message Slot Table
8	Message Slot 3, Reserved	Message Slot 3, Command Response; refer to Message Slot Table
9	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
10	Message Slot 3 Floating Point data	Message Slot 3 Floating Point data
11	Message Slot 3, Scale Status; refer to Message Slot Table	Message Slot 3, Scale Status; refer to Message Slot Table
12	Message Slot 4, Reserved	Message Slot 4, Command Response; refer to Message Slot Table
13	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
14	Message Slot 4 Floating Point data	Message Slot 4 Floating Point data
15	Message Slot 4, Scale Status; refer to Message Slot Table	Message Slot 4, Scale Status; refer to Message Slot Table
16	Message Slot 5, Reserved	Message Slot 5, Command Response; refer to Message Slot Table
17	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
18	Message Slot 5 Floating Point data	Message Slot 5 Floating Point data
19	Message Slot 5, Scale Status; refer to Message Slot Table	Message Slot 5, Scale Status; refer to Message Slot Table
20	Message Slot 6, Reserved	Message Slot 6, Command Response; refer to Message Slot Table
21	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data
22	Message Slot 6 Floating Point data	Message Slot 6 Floating Point data

<b>PLC Output Words</b>	<b>Bits 0 – 7</b>	<b>Bits 8 – 15</b>
23	Message Slot 6, Scale Status; refer to Message Slot Table	Message Slot 6, Scale Status; refer to Message Slot Table
24	Message Slot 7, Reserved	Message Slot 7, Command Response; refer to Message Slot Table
25	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
26	Message Slot 7 Floating Point data	Message Slot 7 Floating Point data
27	Message Slot 7, Scale Status; refer to Message Slot Table	Message Slot 7, Scale Status; refer to Message Slot Table
28	Message Slot 8, Reserved	Message Slot 8, Command Response; refer to Message Slot Table
29	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
30	Message Slot 8 Floating Point data	Message Slot 8 Floating Point data
31	Message Slot 8, Scale Status; refer to Message Slot Table	Message Slot 8, Scale Status; refer to Message Slot Table
32	Message Slot 9, Reserved	Message Slot 9, Command Response; refer to Message Slot Table
33	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
34	Message Slot 9 Floating Point data	Message Slot 9 Floating Point data
35	Message Slot 9, Scale Status; refer to Message Slot Table	Message Slot 9, Scale Status; refer to Message Slot Table
36	Message Slot 10, Reserved	Message Slot 10, Command Response; refer to Message Slot Table
37	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
38	Message Slot 10 Floating Point data	Message Slot 10 Floating Point data
39	Message Slot 10, Scale Status; refer to Message Slot Table	Message Slot 10, Scale Status; refer to Message Slot Table
40	Message Slot 11, Reserved	Message Slot 11, Command Response; refer to Message Slot Table
41	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
42	Message Slot 11 Floating Point data	Message Slot 11 Floating Point data
43	Message Slot 11, Scale Status; refer to Message Slot Table	Message Slot 11, Scale Status; refer to Message Slot Table
44	Message Slot 12, Reserved	Message Slot 12, Command Response; refer to Message Slot Table
45	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
46	Message Slot 12 Floating Point data	Message Slot 12 Floating Point data
47	Message Slot 12, Scale Status; refer to Message Slot Table	Message Slot 12, Scale Status; refer to Message Slot Table

**Table 8-5: PROFINET PLC Floating Point Output Words**

PLC Output Bytes	PLC Output Words	Usage
0	0	Reserved
1		Reserved
2	1	Message Slot 1 Command
3		Message Slot 1 Command
4	2	Message Slot 1 Floating Point data
5		Message Slot 1 Floating Point data
6	3	Message Slot 1 Floating Point data
7		Message Slot 1 Floating Point data
8	4	Message Slot 2 Command
9		Message Slot 2 Command
10	5	Message Slot 2 Floating Point data
11		Message Slot 2 Floating Point data
12	6	Message Slot 2 Floating Point data
13		Message Slot 2 Floating Point data
14	7	Message Slot 3 Command
15		Message Slot 3 Command
16	8	Message Slot 3 Floating Point data
17		Message Slot 3 Floating Point data
18	9	Message Slot 3 Floating Point data
19		Message Slot 3 Floating Point data
20	10	Message Slot 4 Command
21		Message Slot 4 Command
22	11	Message Slot 4 Floating Point data
23		Message Slot 4 Floating Point data
24	12	Message Slot 4 Floating Point data
25		Message Slot 4 Floating Point data
26	13	Message Slot 5 Command
27		Message Slot 5 Command
28	14	Message Slot 5 Floating Point data
29		Message Slot 5 Floating Point data
30	15	Message Slot 5 Floating Point data
31		Message Slot 5 Floating Point data
32	16	Message Slot 6 Command
33		Message Slot 6 Command
34	17	Message Slot 6 Floating Point data
35		Message Slot 6 Floating Point data
36	18	Message Slot 6 Floating Point data
37		Message Slot 6 Floating Point data
38	19	Message Slot 7 Command
39		Message Slot 7 Command

PLC Output Bytes	PLC Output Words	Usage
40	20	Message Slot 7 Floating Point data
41		Message Slot 7 Floating Point data
42	21	Message Slot 7 Floating Point data
43		Message Slot 7 Floating Point data
44	22	Message Slot 8 Command
45		Message Slot 8 Command
46	23	Message Slot 8 Floating Point data
47		Message Slot 8 Floating Point data
48	24	Message Slot 8 Floating Point data
49		Message Slot 8 Floating Point data
50	25	Message Slot 9 Command
51		Message Slot 9 Command
52	26	Message Slot 9 Floating Point data
53		Message Slot 9 Floating Point data
54	27	Message Slot 9 Floating Point data
55		Message Slot 9 Floating Point data
56	28	Message Slot 10 Command
57		Message Slot 10 Command
58	29	Message Slot 10 Floating Point data
59		Message Slot 10 Floating Point data
60	30	Message Slot 10 Floating Point data
61		Message Slot 10 Floating Point data
62	31	Message Slot 11 Command
63		Message Slot 11 Command
64	32	Message Slot 11 Floating Point data
65		Message Slot 11 Floating Point data
66	33	Message Slot 11 Floating Point data
67		Message Slot 11 Floating Point data
68	34	Message Slot 12 Command
69		Message Slot 12 Command
70	35	Message Slot 12 Floating Point data
71		Message Slot 12 Floating Point data
72	36	Message Slot 12 Floating Point data
73		Message Slot 12 Floating Point data

In the template data format a message slot is a shared data variable. The number of message slots is limited to 496 bytes (I/O).

### 8.3.6. Integer and Division

When one of these formats is selected, the IND780 will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. The PLC's input data will contain one 16-bit word for the scale's weight information and one 16-bit word for bit encoded status information for each Message Slot. The IND780 will send specific weight data to the PLC input data based on the data it receives from the PLC's output data. The PLC's output words consist of one 16-bit integer value, which may be used to download a tare or target, and one 16-bit word for bit encoded command information.

Table 8-6 and Table 8-7 provide detailed information on the integer and division data formats. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 8-6: Discrete Read Integer or Division – IND780 >> PLC, per Message Slot**

Octal Address	WORD 0 IN <sup>1</sup>	WORD 1 IN	Bit number
0	X	Feed <sup>2</sup>	<b>0</b>
1	X	Fast Feed <sup>2</sup>	<b>1</b>
2	X	Tolerance OK <sup>3</sup>	<b>2</b>
3	X	Under low tolerance <sup>4</sup>	<b>3</b>
4	X	Over high tolerance <sup>4</sup>	<b>4</b>
5	X	Comparator 3 <sup>5</sup>	<b>5</b>
6	X	Comparator 2 <sup>5</sup>	<b>6</b>
7	X	Comparator 1 <sup>5</sup>	<b>7</b>
10	X	ENTER key <sup>6</sup>	<b>8</b>
11	X	Input 1 <sup>7</sup>	<b>9</b>
12	X	Input 2 <sup>7</sup>	<b>10</b>
13	X	Input 3 <sup>7</sup>	<b>11</b>
14	X	Motion <sup>8</sup>	<b>12</b>
15	X	Net mode <sup>9</sup>	<b>13</b>
16	X	Update in progress <sup>10</sup>	<b>14</b>
17	X <sup>12</sup>	Data OK <sup>11</sup>	<b>15</b>

#### Notes for Table 8-6

- 1 WORD 0 is a 16-bit, signed integer value that may represent the scale's gross, net, tare, target, rate or displayed weight. Three bits, set by the PLC in the output word, designate what data is sent by the terminal in this word.
- 2 **Bit 0** and **Bit 1** are used only in in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer mode and over/under mode.
- 4 When equal to 1, **Bit 3** and **Bit 4** indicate that target is under negative tolerance or over positive tolerance, respectively.
- 5 **Bits 5, 6** and **7** provide Comparators 1, 2 and 3 statuses. If the comparator is active, the respective bit is set to 1.
- 6 **Bit 8** is set to a "1" when the ENTER key is pressed on the keypad of the terminal. The bit remains on for 30 seconds then resets to 0.
- 7 **Bit 9, bit 10, and Bit 11** mirror the state of the first three discrete inputs on the internal I/O board slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is "ON" then the bit is set to a "1".
- 8 **Bit 12** is set to a "1" when the scale is in motion (unstable).
- 9 **Bit 13** is set to a "1" when the scale is in net mode (a tare has been taken).

- 10 **Bit 14** is set to a "1" when the terminal is in the process of updating its data for the PLC scanner. The PLC should ignore ALL of the data in this case and simply re-scan it.
- 11 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, in expanded mode, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent remains in the standard resolution, and will not match the terminal's display. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode word 0 data may be present; do not use the data under this condition as the communication connection may stop at any time.
- 12 When number is negative, word 0 **Bit 15** is high and is the least significant bit. Otherwise, when the number is positive, **Bit 15** is the most significant bit.

**Table 8-7: Discrete Write Integer or Division –PLC >> IND780, per Message Slot**

Octal Address	WORD 0 OUT <sup>1</sup>	WORD 1 OUT	Bit number
0	X	Select 1 <sup>2</sup>	<b>0</b>
1	X	Select 2 <sup>2</sup>	<b>1</b>
2	X	Select 3 <sup>2</sup>	<b>2</b>
3	X	Load Tare <sup>3</sup>	<b>3</b>
4	X	Clear <sup>4</sup>	<b>4</b>
5	X	Tare <sup>5</sup>	<b>5</b>
6	X	Print <sup>6</sup>	<b>6</b>
7	X	Zero <sup>7</sup>	<b>7</b>
10	X	Abort/Start Target <sup>8</sup>	<b>8</b>
11	X	Display mode <sup>9</sup>	<b>9</b>
12	X	Display mode <sup>9</sup>	<b>10</b>
13	X	Display mode <sup>9</sup>	<b>11</b>
14	X	Output 1 <sup>10</sup>	<b>12</b>
15	X	Output 2 <sup>10</sup>	<b>13</b>
16	X	Output 3 <sup>10</sup>	<b>14</b>
17	X	Load Target <sup>11</sup>	<b>15</b>

**Notes for Table 8-7**

- WORD 0 is a 16-bit, signed integer value that may represent the scale's tare or target value to be downloaded. **Bit 3** or **bit 15** is then triggered to instruct the terminal to load the value into either the tare or target register.
- A binary value in **bit 0, bit 1, and bit 2** select the data that will be sent by the terminal in Discrete Read WORD 0. 0 = gross weight, 1 = net weight, 2 = displayed weight, 3 = tare weight, 4 = target, 5 = rate, 6 and 7 = reserved. Any value greater than 7 will cause gross weight to be sent.
- A transition from "0" to "1" loads the value from WORD 0 into the tare register of the IND780; the IND780 will use the loaded value as the tare.
- A transition from "0" to "1" initiates a CLEAR command.
- A transition from "0" to "1" initiates a TARE command.
- A transition from "0" to "1" initiates a PRINT command.
- A transition from "0" to "1" initiates a ZERO command.
- If **bit 8** is set to "0", all of the scale's target logic is aborted. Setting **bit 8** to "1" again restarts the terminal's target logic.
- Bits 9-11** control Display Mode. A command is written to pd0119. Note that pd0119=0 for normal display mode, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 = display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is

written into pd0118. Note: Until these bits are used pd0119 will be set at 0. Change of state of any of these bits resets PLC input word, **Bit 8**, Enter Key.

- 10 **Bit 12, bit 13, and bit 14** can be used to control the state of the first three discrete outputs on the terminal's internal I/O board in slot 5. These are addressed as 0.5.1, 0.5.2, and 0.5.3. Setting the bit to a "1" state causes the output to be turned ON. This action will occur regardless of the discrete output assignment within the IND780.
- 11 When **bit 15** is changed from "0" to "1" the value in WORD 0 is loaded into the target register in the terminal and loaded into the target logic.

### 8.3.7. Floating Point

#### 8.3.7.1. Operational Overview

The IND780 uses integer commands from the PLC to select the floating point weight input data. The IND780 recognizes a command when it sees a new value in the Message Slot command word. If the command has an associated floating point value (for example: loading a target value), it must be loaded into the floating point value words before the command is issued. Once the IND780 recognizes a command, it acknowledges the command by setting a new value in the command acknowledge bits of the scale's command response word. The IND780 also tells the PLC what floating point value is being sent (via the floating point input indicator bits of the command response word). The PLC should wait until it receives the command acknowledgment from the IND780 before sending another command.

The IND780 can report two types of values to the PLC: real-time and static. When the PLC requests a real-time value, the IND780 acknowledges the command from the PLC once but sends and updates the value at every interface update cycle. If the PLC requests a static value, the IND780 acknowledges the command from the PLC once and updates the value once. The IND780 will continue to send this value until it receives a new command from the PLC. Gross weight and net weight are examples of real-time data. Tare weight, target, feed, and tolerance values are examples of static data.

The IND780 can send a rotation of up to nine different real-time values. The PLC sends commands to the IND780 to add a value to the rotation. Once the rotation is established, the PLC must instruct the IND780 to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the IND780 to advance to the next value. If the IND780 is asked to automatically alternate its output data, it will switch to the next value in its rotation at the next interface update cycle. (The interface update cycle has an update rate of up to 17 Hz or 60 milliseconds.)

The PLC may control the rotation by sending alternate report next field commands (1 and 2). When the PLC changes to the next command, the IND780 switches to the next value in the rotation order. The IND780 stores the rotation in its shared data so the rotation does not have to be re-initialized after each power cycle. When the PLC does not set up an input rotation, the default input rotation consists of gross weight only. See the floating-point command examples in Table 8-12 through Table 8-15 for additional information. The method of handling string and floating point data varies between Allen-Bradley PLC generations. The IND780 provides floating point data in the order entered in Data Format setup.

Table 8-8 through Table 8-11 provide detailed information on the floating-point data format. Read data refers to the PLC's input data and write data refers to the PLC's output data.

**Table 8-8: Discrete Read Floating Point – IND780 >> PLC Input, per Message Slot**

Octal Address	WORD 0 Command Response	WORD 1 <sup>1</sup> FP value	WORD 2 <sup>1</sup> FP value	WORD 3 Status	Bit number
0	RESERVED	X	X	Feed <sup>2</sup>	<b>0</b>
1		X	X	Fast Feed <sup>2</sup>	<b>1</b>
2		X	X	Tolerance OK <sup>3</sup>	<b>2</b>
3		X	X	Under Negative Tolerance <sup>4</sup>	<b>3</b>
4		X	X	Over Positive Tolerance <sup>4</sup>	<b>4</b>
5		X	X	Selected scale <sup>5</sup>	<b>5</b>
6		X	X	Custom Bit <sup>6</sup>	<b>6</b>
7		X	X	Custom Bit <sup>6</sup>	<b>7</b>
10	FP Input Ind 1 <sup>7</sup>	X	X	ENTER key <sup>8</sup>	<b>8</b>
11	FP Input Ind 2 <sup>7</sup>	X	X	Input 1 <sup>9</sup>	<b>9</b>
12	FP Input Ind 3 <sup>7</sup>	X	X	Input 2 <sup>9</sup>	<b>10</b>
13	FP Input Ind 4 <sup>7</sup>	X	X	Input 3 <sup>9</sup>	<b>11</b>
14	FP Input Ind 5 <sup>7</sup>	X	X	Motion <sup>10</sup>	<b>12</b>
15	Data integrity1 <sup>11</sup>	X	X	Net mode <sup>12</sup>	<b>13</b>
16	Cmnd Ack 1 <sup>13</sup>	X	X	Data integrity 2 <sup>11</sup>	<b>14</b>
17	Cmnd Ack 2 <sup>13</sup>	X	X	Data OK <sup>14</sup>	<b>15</b>

**Notes for Table 8-8**

- 1 The bits in WORD 1 and WORD 2 are a single-precision floating point value that may represent the scale’s gross, tare, net, target, fine gross, fine tare, fine net, or filter setting data. The PLC command in the respective scale’s output word determines what data will be sent.
- 2 **Bit 0** and **Bit 1** are used only in material transfer mode.
- 3 **Bit 2** indicates that tolerance is OK in both material transfer and over/ under modes.
- 4 **Bit 3** and **Bit 4** are active in both material transfer and over/under modes.
- 5 **Bit 5** will be set to a “1” when the associated scale is selected on the IND780 console panel.
- 6 **Bit 6** and **Bit 7** are user defined, and corresponds to user’s Task Expert Application usage. **Bit 6** is associated to Shared Data Variable ASxx01, instance by scale. **Bit 7** is associated with Shared Data Variable ASxx02, instance by scale.
- 7 The Floating Point Input Indication bits (WORD 0, **Bits 8-12**) are used to determine what type of data is being sent in the floating point value (WORD 1 and WORD 2). These bits correspond to a decimal value of 0-31 that represents a particular type of data. See the Floating Point Input Indication Table to determine what type of data.
- 8 **Bit 8** is set to a “1” when the ENTER key is pressed on the terminal’s keypad. The bit clears to “0” when the PLC sends floating point command 75 to the IND780 terminal or after 30 seconds of no ENTER key activity.
- 9 **Bit 9**, **Bit 10**, and **Bit 11** mirror the state of the first three discrete inputs of the internal I/O board in slot 5 (0.5.1, 0.5.2 and 0.5.3). If the input is “ON” then the bit is set to a “1”.
- 10 **Bit 12** is set to a “1” when the scale is in motion (unstable).
- 11 The Data Integrity bit in WORD 0 - **bit 13** is used in conjunction with the bit in WORD 3 - **bit 14** to insure that the floating point data is valid. For the data to be valid both bits must have the same polarity. These bits will change to the opposite state every interface update cycle. If they do not have the same value the data is invalid and the PLC should ignore ALL of the data in this case and re-scan it.
- 12 **Bit 13** is set to a “1” when the scale is in net mode (a tare has been taken).

- 13 **Bit 14** and **Bit 15** (Command Acknowledge bits) are used by the terminal to inform the PLC that it has received a new, valid command. The terminal rotates sequentially among values 1, 2, 3, 1, 2, 3, 1, 2, ... to acknowledge it has processed a new command.
- 14 **Bit 15** is set to a "1" when the scale is operating properly (**NOT** over capacity, under capacity, in power-up, or in setup mode) or in x10 mode. Note that when in x10 mode, the data sent is in the higher resolution. The PLC program should continuously monitor this bit and the PLC processor rack fault bit (see A-B RIO PLC documentation) to determine the validity of the discrete and/or explicit data transfer. While in the setup mode the input floating data will be set to 0; additionally the communication connection may stop at any time.

**Table 8-9: Floating Point Input Indication**

Dec	Data
0	Gross Weight*
1	Net Weight*
2	Tare Weight*
3	Fine Gross Weight*
4	Fine Net Weight*
5	Fine Tare Weight*
6	Rate
7	ALxx01 – SDV, instance by scale **
8	ALxx02 – SDV, instance by scale **
9	AJxx01 – SDV, instance by scale **
10	AJxx02 – SDV, instance by scale **
11	Low-pass filter frequency
12	Notch filter frequency
13	Target value, instance by scale
14	Target positive tolerance value, instance by scale
15	Target Fine Feed value, instance by scale
16	Target negative tolerance value, instance by scale
17	Target spill value, instance by scale
18	Primary units, low increment size
19–28	Not used
29	Last IND780 error code
30	No data response – command successful
31	No data response – command failed

\* These are real-time fields that the PLC may request either through an input rotation or a report command. All other fields may only be requested through a report command.

\*\* SDV means Shared Data Variable.

**Table 8-10: Discrete Write Floating Point – PLC >> IND780, per Message Slot**

Octal Address	WORD 0 <sup>0</sup>	WORD 1	Word 2 <sup>1</sup> FP load value	Word 3 <sup>1</sup> FP load value	Bit Number
0	Reserved	Command Word <sup>2</sup>	X	X	0
1			X	X	1
2			X	X	2
3			X	X	3

Octal Address	WORD 0 <sup>0</sup>	WORD 1	Word 2 <sup>1</sup> FP load value	Word 3 <sup>1</sup> FP load value	Bit Number
4			X	X	4
5			X	X	5
6			X	X	6
7			X	X	7
10			X	X	8
11			X	X	9
12			X	X	10
13			X	X	11
15			X	X	12
16			X	X	13
17			X	X	14

### Notes for Table 8-10

- 0 WORD 0 is a reserved 16 bit word, only present in message slot 1.
- 1 The Bits in WORD 2 and WORD 3 are a single-precision floating point value. This value is used with the command in WORD 0 to instruct the terminal to download the floating point value into the field specified in the command.
- 2 The command Word 1 is used to instruct the IND780 what data to send in the discrete read data, to load the floating point data in the write command, and to control the IND780 discrete outputs or display. See Table 8-11, the PLC Output Command Table, for a list of the available commands and their respective decimal or hex value.

Not all commands will require a value in the floating point load value words.

**Table 8-11: PLC Output Command Table (Floating Point Only)**

Dec	Hex	Command	SDName	Dec	Hex	Command	SDName
0	00	Report next rotation field @ next interface update cycle <sup>1</sup>		76	4c	Start ID2 sequence <sup>7,9</sup>	PD0119
1	01	Report next rotation field <sup>1,2</sup>		78	4e	Disable LCD display <sup>7</sup>	
2	02	Report next rotation field <sup>1,2</sup>		79	4f	Enable LCD display <sup>7</sup>	
3	03	Reset rotation		80	50	Set normal display mode <sup>7,9</sup>	PD0119
10	0a	Report gross weight <sup>1,3</sup>		81	51	Display message 1 <sup>7,9</sup>	PD0119
11	0b	Report net weight <sup>1,3</sup>		82	52	Display message 2 <sup>7,9</sup>	PD0119
12	0c	Report tare weight <sup>1,3</sup>		83	53	Display message 3 <sup>7,9</sup>	PD0119
13	0d	Report fine gross weight <sup>1,3</sup>		84	54	Display message 4 <sup>7,9</sup>	PD0119
14	0e	Report fine net weight <sup>1,3</sup>		85	55	Display Message 5 <sup>7,9</sup>	PD0119
15	0f	Report fine tare weight <sup>1,3</sup>		86	56	Start ID1 sequence <sup>7,9</sup>	PD0119
16	10	Report Rate <sup>1,3</sup>		87	57	Display SDV PD0118 <sup>7,9</sup>	PD0119
17	11	Report SDV ALxx01 <sup>6</sup>	SPxx05 <sup>6</sup>	88	58	Disable weight display <sup>7</sup>	
18	12	Report SDV ALxx02 <sup>6</sup>	SPxx11 <sup>6</sup>	89	59	Enable weight display <sup>7</sup>	
19	13	Report low-pass filter frequency <sup>3</sup>		90	5a	Set discrete output 0.5.1 "ON" <sup>7</sup>	DI0505
20	14	Report notch filter frequency <sup>3</sup>		91	5b	Set discrete output 0.5.2 "ON" <sup>7</sup>	DI0506
21	15	Report Target value <sup>3,10</sup>	SPxx05 <sup>6</sup>	92	5c	Set discrete output 0.5.3 "ON" <sup>7</sup>	DI0507
22	16	Report Positive Tolerance <sup>3,10</sup>	SPxx11 <sup>6</sup>	93	5d	Set discrete output 0.5.4 "ON" <sup>7</sup>	DI0508
23	17	Report Fine Feed <sup>3,10</sup>	SPxx10	100	64	Set discrete output 0.5.1 "OFF" <sup>7</sup>	DI0505
24	18	Report (-) Tolerance value <sup>3,10</sup>	SPxx12	101	65	Set discrete output 0.5.2 "OFF" <sup>7</sup>	DI0506
25	19	Report spill value <sup>3,10</sup>	SPxx09	102	66	Set discrete output 0.5.3 "OFF" <sup>7</sup>	DI0507

Dec	Hex	Command	SDName
27	1b	Report AJxx01 <sup>3,5,6</sup>	AJxx01 <sup>6</sup>
28	1c	Report AJxx02 <sup>3,5,6</sup>	AJxx02 <sup>6</sup>
29	1d	Report last error <sup>3</sup>	
30	1e	Report primary increment size <sup>3</sup>	
40	28	Add gross weight to rotation <sup>7</sup>	
41	29	Add net weight to rotation <sup>7</sup>	
42	2a	Add tare weight to rotation <sup>7</sup>	
43	2b	Add fine gross weight to rotation <sup>7</sup>	
44	2c	Add fine net weight to rotation <sup>7</sup>	
45	2d	Add fine tare weight to rotation <sup>7</sup>	
46	2e	Add rate to rotation <sup>7</sup>	
47	2f	Add ALxx01 to rotation <sup>5,6,7</sup>	ALxx01 <sup>6</sup>
48	30	Add ALxx02 to rotation <sup>5,6,7</sup>	ALxx02 <sup>6</sup>
60	3c	Load numeric tare value <sup>4</sup>	
61	3d	Pushbutton tare command <sup>7</sup>	
62	3e	Clear command <sup>7</sup>	
63	3f	Print command <sup>7</sup>	
64	40	Zero command <sup>7</sup>	
65	41	Select scale 1 <sup>7</sup>	
66	42	Select scale 2 <sup>7</sup>	
67	43	Select next scale <sup>7</sup>	
68	44	Custom print 1 <sup>7</sup>	CPxx01 <sup>7</sup>
69	45	Custom print 2 <sup>7</sup>	CPxx02 <sup>7</sup>
70	46	Custom print 3 <sup>7</sup>	CPxx03 <sup>7</sup>
71	47	Custom print 4 <sup>7</sup>	CPxx04 <sup>7</sup>
72	48	Custom print 5 <sup>7</sup>	CPxx05 <sup>7</sup>
73	49	Set low-pass filter corner frequency <sup>4</sup>	
74	4a	Set notch filter frequency <sup>4</sup>	
75	4b	Reset ENTER key <sup>7</sup>	

Dec	Hex	Command	SDName
103	67	Set discrete output 0.5.4 "OFF" <sup>7</sup>	DI0508
104	68	Master control relay OFF <sup>7, 12</sup>	
105	69	Master control relay ON <sup>7, 12</sup>	
110	6e	Set target value <sup>4,10, 13</sup>	SPxx05
111	6f	Set target fine feed value <sup>4,10</sup>	
112	70	Set - tolerance value <sup>4,10</sup>	
113	71	Set target value and apply <sup>4, 10, 14</sup>	
114	72	Start target logic <sup>7,10,15</sup>	
115	73	Abort target logic <sup>7,10,15</sup>	
116	74	Target use gross weight <sup>7,10</sup>	SPxx04
117	75	Target use net weight <sup>7,10</sup>	SPxx04
118	76	Target use rate <sup>7,10</sup>	SPxx04
119	77	Target, absolute weight 1 speed <sup>7,10</sup>	SPxx08
120	78	Target, absolute weight 2 speed <sup>7,10</sup>	SPxx08
121	79	Enable target latching <sup>7,10</sup>	SPxx06
122	7a	Disable target latching <sup>7,10</sup>	SPxx06
123	7b	Reset target latch <sup>7</sup>	SPxx07
124	7c	Set target spill value <sup>4, 10</sup>	
131	83	Set + tolerance value <sup>4,10</sup>	SPxx11
160	a0	Apply scale setup <sup>7</sup>	QC0149
161	a1	Write Calibration to EEPROM <sup>7</sup>	
162	a2	Disable Tare on IND780 console <sup>7,11</sup>	
163	a3	Enable Tare on IND780 console <sup>7</sup>	
164	a4	Disable push button Tare on IND780 console <sup>7,11</sup>	
165	A5	Enable push button Tare on IND780 console <sup>7</sup>	
166	a6	Disable numeric Tare on IND780 console <sup>7,11</sup>	
167	a7	Enable numeric Tare on IND780 console <sup>7</sup>	
168	a8	Select scale 3 <sup>7</sup>	
169	a9	Select scale 4 <sup>7</sup>	

### Notes for Table 8-11

1. A command that requests real-time fields from the terminal. The terminal updates this input data to the PLC at the cycle update rate of the PLC interface.
2. A command used by the PLC to select the next field from the input rotation. The PLC must alternate between these two commands to tell the terminal when to switch to the next field of the input rotation.
3. A command requiring the terminal to report a specific value in the PLC input message. As long as one of these commands is sent in the Scale Command, the terminal will respond with the requested data and not data from an input rotation. The data reported in the PLC input message is the data when the command was issued.
4. A command that requires a floating point value output from the PLC to the terminal. The terminal reflects back this value in the floating point data of the input message to the PLC.
5. A command used between the PLC and a Task Expert application. This data has a four-byte length and is defined by the application.
6. Instance is by scale.
7. PLC input Message Slot floating point value will be 0 as long as command is present.
8. IND780 discrete output will be turned on or off regardless of assignment within the IND780.
9. A command is written to pd0119. Note that pd0119==0 for normal display, 1 = display message 1 (aw0101), 2 = display message 2 (aw0102), 3 = display message 3 (aw0103), 4 = display message 4 (aw0104), 5 =

display message 5 (aw0105), 6 = Start ID1 sequence, 7 = display message that is written into pd0118, 8 = Start ID2 sequence. Note: Until these commands are used pd0119 will be set at 0.

10. Target, Fine Feed, Tolerances, Restart, Abort and other Target variables are relative to each scale. Note: Restart of an active Target will cause the associated outputs to cycle off then back on.
11. Does not disable the PLC tare functions.
12. Setting the master control relay OFF turns off all outputs and stops target logic. The target logic must be re-started after the master control relay is turned ON.
13. Target logic must be restarted to apply the new target value.
14. New target value is applied when command is sent.
15. Target control can be paused and resumed using the Abort and Start commands. Note that any changes made to the target values since the original START command was given will be loaded before target control is resumed.

### 8.3.7.2. Floating Point Data Format and Compatibility

In Floating Point Data Format, the PLC and IND780 terminal exchange weight, target, and tare data in single-precision floating-point format. The IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985, specifies the format for single-precision floating point numbers. It is a 32-bit number that has a 1-bit sign, an 8-bit signed exponent, and a 23-bit mantissa. The 8-bit signed exponent provides scaling of weight data. The 23-bit mantissa allows representation of 8 million unique counts.

Although the single-precision floating point number provides greater numerical precision and flexibility than integer weight representations, it has limitations. The weight representation may not be exact, particularly for the extended-resolution weight fields for high-precision bases.

There are two data integrity bits that the IND780 uses to maintain data integrity when communicating with the PLC. One bit is in the beginning word of the data; the second is in the ending byte of the data for a scale slot. The PLC program must verify that both data integrity bits have the same polarity for the data in the scale slot to be valid. There is a possibility that the PLC program will see several consecutive invalid reads when the terminal is freely sending weigh updates to the PLC, if the PLC program detects this condition, it should send a new command to the terminal.

The IND780 provides floating point data in the word order set up by the user.

### 8.3.7.3. Floating Point Command Examples

Table 8-12 through Table 8-15 provide floating point command examples.

**Table 8-12: Data Requirement: Only Net Weight Sent (continuously) for Scale 1**

Step #	Scale Command (From PLC)	Scale Floating Point Value	Command Response From Terminal	Floating Point Value
1 (PLC sends command to IND780 terminal to report net weight)	11 (dec) loaded into command word 0	none required		
2 (IND780 terminal sees new command)			Command ack. = 1 F.P. ind. = 1 (net)	Net weight in floating point
As long as the PLC leaves the 11 (dec) in the command word, the IND780 terminal will update the net value every interface update cycle.				

**Table 8-13: Data Requirement: Load Target Value = 21.75 for Scale 1**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC loads floating point value first)		floating point value = 21.75		
2 (PLC sends command to set target 1 cutoff value)	110 (dec) loaded into command word 0	floating point value = 21.75		
3 (IND780 terminal sees new command , loads the value into the target and ends a return message to indicate the new target value)			Command ack. = 1 F.P. ind = 13	Floating point value = 21.75
4 (PLC instructs IND780 terminal to start "using" new target value)	114 (dec) loaded into command word 0			
5 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0

The PLC should always wait to receive a command acknowledgment before sending the next command to the IND780 terminal. After the PLC finishes loading its target value, it can resume monitoring the weight information required by sending a command to report some type of weight or set up a rotation of reported data.

**Table 8-14: Data Requirement: Rotation of Gross Weight and Rate Updated on Interface Update Cycle**

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack.= 1 F.P. ind = 30	0.0
3 (PLC adds gross weight to rotation)	40 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0			
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0

At this point, the rotation has been set up. Now the PLC needs to command the IND780 terminal to begin the rotation.

Step #	Scale Command (from PLC)	Scale Floating Point Value	Command Response from Terminal	Floating Point Value
7 (PLC sends the command to begin the rotation at interface update cycle)	0 (dec) loaded into command word 0			
8 (IND780 terminal sends gross weight at interface update cycle ~ 60 msec)			Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
9 (PLC leaves 0 in its command word and the IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
10 (PLC leaves 0 in its command word and IND780 terminal sends the gross value at next interface update cycle)	0 (dec) loaded into command word 0		Command ack. = 0 F.P. ind = 0	Floating point value = gross wt.
11 (PLC leaves 0 in command word and IND780 terminal sends the rate value at the next interface update cycle)	0 (dec) loaded into command word 0	RESERVED for Future Use	Command ack. = 0 F.P. ind = 6	Floating point value = rate
This rotation continues until the PLC sends a different command. At approximately every 60 msec the IND780 terminal updates its data with the next field in its rotation. The PLC must check the floating point indication bits to determine which data is in the floating point value.				

**Table 8-15: Data Requirement: Rotation of Net Weight and Rate Updated on PLC Command**

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
1 (PLC clears out any previous rotation with reset)	3 (dec) loaded into command word 0			
2 (IND780 terminal sees new command)			Command ack.= 1 F.P. ind = 30	0.0
3 (PLC adds net weight to rotation)	41 (dec) loaded into command word 0	(null value)		
4 (IND780 terminal sees new command)			Command ack. = 2 F.P. ind = 30	0.0
5 (PLC adds rate to the rotation)	46 (dec) loaded into command word 0	RESERVED for Future Use		
6 (IND780 terminal sees new command)			Command ack. = 3 F.P. ind = 30	0.0

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
At this point, the rotation has been set up. Now the PLC needs to send commands to the IND780 terminal to begin the rotation and advance to the next value when required.				
7 (PLC sends the command to report the first field in the rotation.)	1 (dec) loaded into command word 0			
8 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net weight
9 (PLC sends the command to report the next field.) Note: if the PLC leaves the 1 (dec) in the command, the IND780 terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word 0			
10 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
11 (PLC sends the command to report the next field in the rotation.)	1 (dec) loaded into command word 0			
12 (IND780 terminal acknowledges the command and sends net weight at every interface update cycle until the PLC gives the command to report the next rotation field.)			Command ack. = 1 F.P. ind = 1	Floating point value = net wt.
13 (PLC sends the command to report the next field.)	2 (dec) loaded into command word 0			

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
14 (IND780 terminal acknowledges the command and sends rate at every interface update cycle until the PLC gives the command to report the next rotation field.)		RESERVED for Future Use	Command ack. = 2 F.P. ind = 6	Floating point value = rate
At approximately every 60 msec the IND780 terminal updates its data with new data, but it does not advance to the next field in the rotation until the PLC sends it the command to report the next field. The PLC should check the floating point indication bits to determine which data is in the floating point value				

### 8.3.8. Controlling the Discrete I/O Using a PLC Interface

The IND780 terminal provides the ability to directly control its discrete outputs and read its discrete inputs via the (digital) PLC interface options. System integrators should be aware that the IND780 terminal's discrete I/O updates are synchronized with the terminal's interface update rate and not with the PLC I/O scan rate. This may cause a noticeable delay in reading inputs or updating outputs as observed from the PLC to real world signals. Consult the **IND780 Terminal Technical Manual** for discrete I/O wiring.

## 8.4. Shared Data Access

The Shared Data mode PLC communications is provided using Acyclic messaging to the IND780 terminal.

The IND780 Shared Data document lists the Shared Data Variables available to EtherNet/IP, ControlNet, and PROFINET. This document also includes the hex Class Code, Instance and Attribute for the shared data. The PLC must use a combination of RDREC (SFB52) and WRREC (SFB53) to read a Shared Data Variable and WRREC (SFB53) to write a Shared Data Variable.

## 8.5. Software Setup

When the IND780 terminal detects the presence of a PROFINET Kit option board, the PROFINET parameters are enabled in a Setup program block at **Communication > PLC Interface > PROFINET**. Figure 8-6 shows the PROFINET setup block, and Figure 8-7 graphs the **Communication > PLC Interface > Data Format** setup block. Default values are indicated by an asterisk.

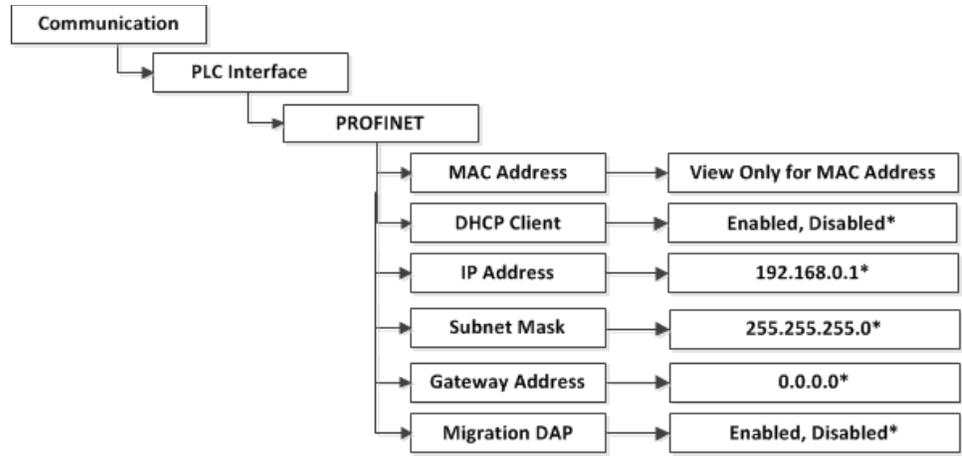


Figure 8-6: PROFINET Setup Block

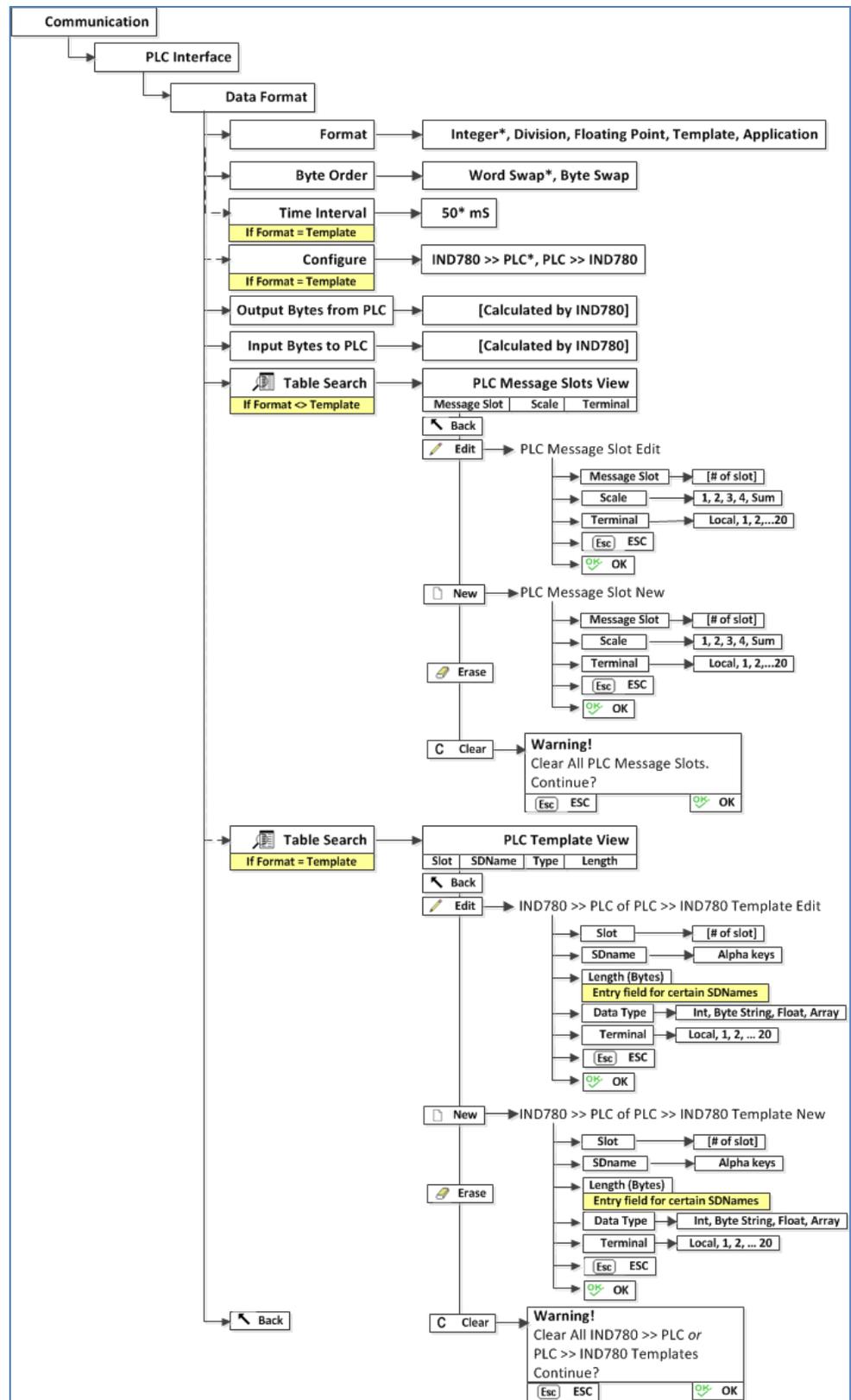
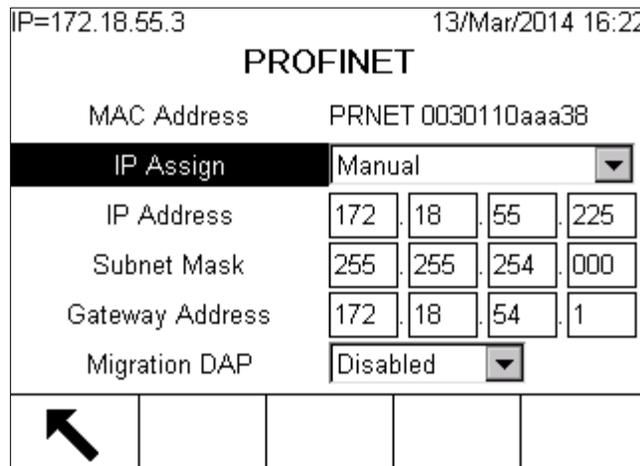


Figure 8-7: PLC Data Format Setup Block

## 8.5.1. PROFINET Setup Block

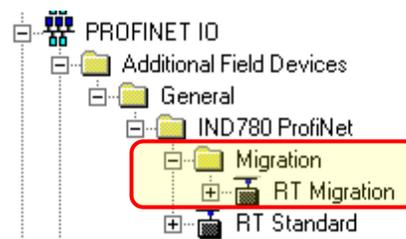
### 8.5.1.1. PROFINET setup

The PROFINET Setup block lets you specify IP address of the PROFINET interface. From Setup, select **Communication > PLC Interface > PROFINET**. Set the IP Assign to "Manual" and then set the IP Address, Subnet Mask and Gateway Address (Figure 8-8). Alternatively, DHCP Client can be enabled if the IP address is to be assigned automatically by the network server, by setting the IP Assign value to "DHCP". By default, the "IP Assign" value is set to "DCP" so that the PLC Programming software can assign the IP Address. Note that in all cases, the Device Name must be set from the PLC programming software before communications to the PLC will be established.



**Figure 8-8: PROFINET Setup Screen**

The Migration DAP option allows the IND780 to communicate with older PROFINET PLC Controllers that do not support PROFINET IO Specification 2.0. or later and only support DAP's with no Physical Device (PDEV). If this option is Enabled, then the user must select the modules from the Migration DAP in the Hardware Configuration

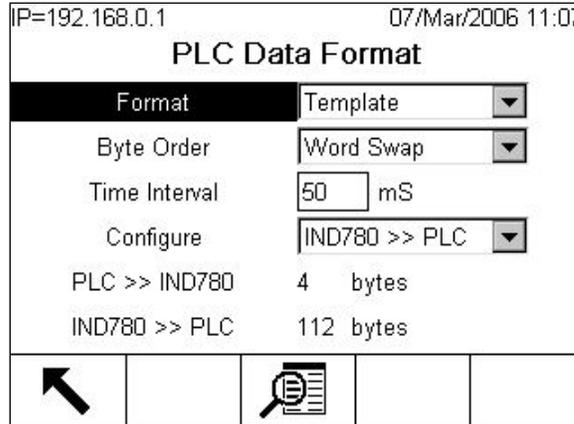


**Figure 8-9: Migration DAP for PLCs that do not support PROFINET IO 2.0 or higher.**

8.5.1.2.

Data Format setup

1. In Setup, select **Communication > PLC Interface > Data Format**. Figure 8-10 shows the screen with Template selected as the format, revealing the Time Interval, Configure, and directional displays. When Application format is selected, only the Format box appears on this screen.



**Figure 8-10: PLC Data Format Screen**

2. Select the Format (Integer, Divisions, Floating Point, Template or Application): [default Integer] Changing the Format will delete any existing Message Slots.

<b>Integer</b>	Reports scale weight as a signed 16 bit integer ( $\pm 32767$ ).
<b>Divisions</b>	Reports scale weight in display divisions ( $\pm 32767$ ). The PLC multiplies the reported divisions by the increment size to calculate the weight in display units.
<b>Floating Point</b>	Displays weight in floating point data format
<b>Template</b>	Allows the end-user to customize PLC communications input and output data by allowing them to set up input and output templates with shared data variables. This format is fixed and cannot be changed "on the fly" by the PLC program. The template editor also calculates the input and output sizes upon exit of the editor. In order to populate the data fields, consult the Shared Data Reference Manual (document number 64059110) provided on the CD-ROM supplied with your terminal.
<b>Application</b>	Allows a special application such as TaskExpert or Q.i to interact with the PLC input and output data. The application program is responsible for setting up all input and output data from the IND780 and to control the data length of the input and output assemblies. For application commands used in TaskExpert, refer to the TaskExpert Reference Manual (document number 64060431).

3. Select Byte Order: [default Word Swap]

<b>Word Swap</b>	Takes the IEEE 754 single-precision floating point format and swaps the two words in the 32-bit double word. This format is compatible with SIMATIC Step 7 processors.
<b>Byte Swap</b>	Takes the IEEE 754 single-precision floating point format and swaps the bytes in the two words of the 32-bit double word. This format is compatible with Siemens S7 processors.

4. Set the **Time Interval** in milliseconds. (Only when the Format is Template)

5. Use the **Configure** drop-down list to select the direction of data flow – IND780 >> PLC or PLC >> IND780. (Only when the Format is Template)
6. The bottom of this screen (Figure 8-10) displays the number of bytes that will be sent by the IND780 and the number of bytes expected from the PLC.
7. Depending on which format is selected, press the VIEW TABLE softkey  to set up the number of Message Slots (maximum number is 12; default is 0), or to configure the Template for communications between the IND780 and the PLC. Figure 8-11 shows the Message Slots View, which is viewed whenever the Format is not Template.

IP=172.18.54.111		06/Dec/2007 16:25	
PLC Message Slots View			
Message	Scale	Terminal	
▶ 1	1	Local	
2	- End -		
			 C

**Figure 8-11: PLC Message Slots View Screen**

Figure 8-12 shows the Template View, accessed when Format is set to Template. Each message slot will be a Shared Data Variable, entered by the user. These entered message slots are associated only to the Shared Data Variable entered and not to a scale.

IP=192.168.0.1		07/Mar/2006 11:08	
PLC >> IND780 Template View			
Slot	SDName	Type	Length
▶ 1	pb0202	Int	2
2	xt0103	Int	2
3	- End -		
			 C

**Figure 8-12: PLC Template View**

Press the EXIT softkey  to exit the view screen.

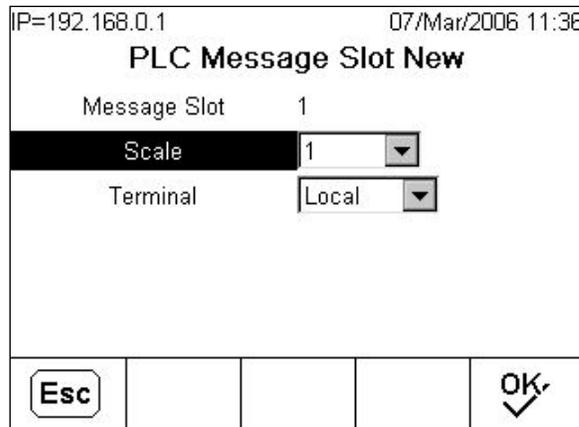
Press the EDIT softkey  to edit an existing slot.

Press the NEW softkey  to enter a new slot.

Press the DELETE softkey  to delete an existing slot.

Press the CLEAR softkey **C** to delete all existing slots.

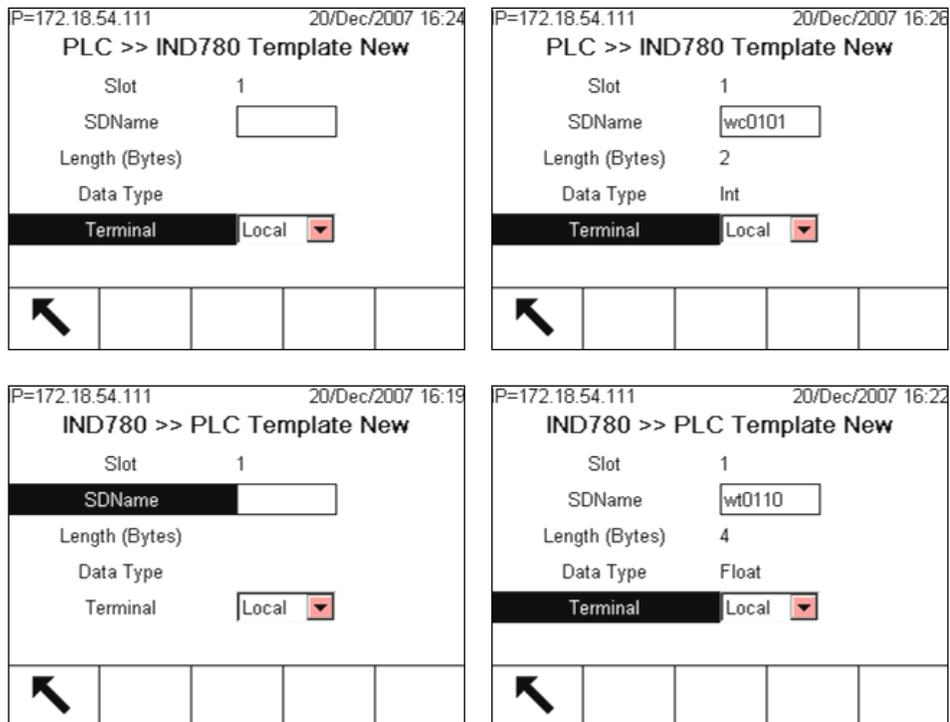
Figure 8-13 shows the screen that opens when NEW is selected from the Message Slots View screen.



**Figure 8-13: New Message Slot Screen**

- Scale** Enter the scale number to be associated with the new Message Slot.
- Terminal** Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

Figure 8-14 shows a series of Template New screens, as accessed from the Template View screen.



**Figure 8-14: New Template Screens, PLC to IND780 and IND780 to PLC**

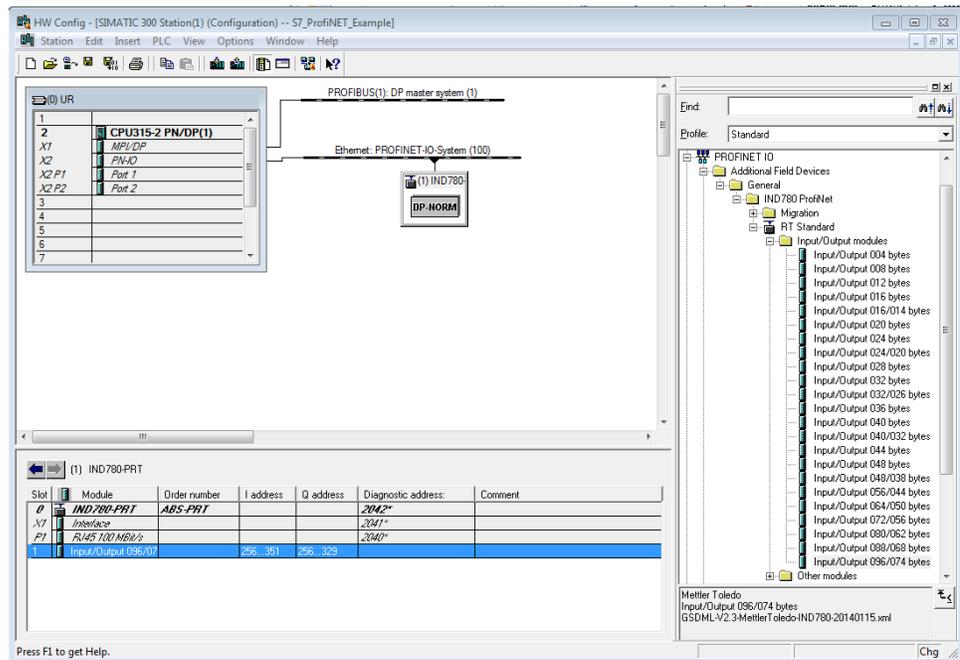
<b>Slot</b>	Automatically assigned.
<b>SDName</b>	An alphabetic entry field used to define the Shared Data variable associated with the slot.
<b>Length (Bytes) and Data Type</b>	Automatically displayed once the SDName has been entered.
<b>Terminal</b>	Default is Local; remote terminals 1 through 20 may be selected if remote PLC interface sharing is used.

For a complete listing of Shared Data Fields, refer to the IND780 Shared Data Reference. Refer to Table 8-6 (Discrete Read) and Table 8-7 (Discrete Write) in this chapter for additional information on mapping of discrete read data to the PLC.

## 8.6. PROFINET GSDML File

There are twenty three standard configurations of the PROFINET GSDML file for the IND780 terminal's different combinations of data formats for Integer, Division and Floating Point. The length of the messages is different for each of these data formats. The type file selected for these configurations should match the message lengths shown on the PLC Data Format Screen.

- **Note:** The Utilities folder of the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.



**Figure 8-15: Hardware Network Setup**

Figure 8-15 shows an IND780 placed on the PROFINET I/O Network as node 1, with the Input/Output module selected as 096/074 bytes, which would be used in Floating Point Mode with 12 Message Slots assigned

The PROFINET GSDML file for the IND780 is available free of charge. It is included on the documentation CD (part number 64057241) and on the Compact Flash drive in the unit itself (see

directory “/PLC Config Files/PROFINET”), which is available via FTP over the IND780’s TCP/IP network connection.

## 8.7. Assigning the IP Address and Device Name

By default, assigning the IND780’s IP address and Device Name takes place via the DCP (Discovery and basic Configuration Protocol). This function is accessed via the Engineering Software as shown below.

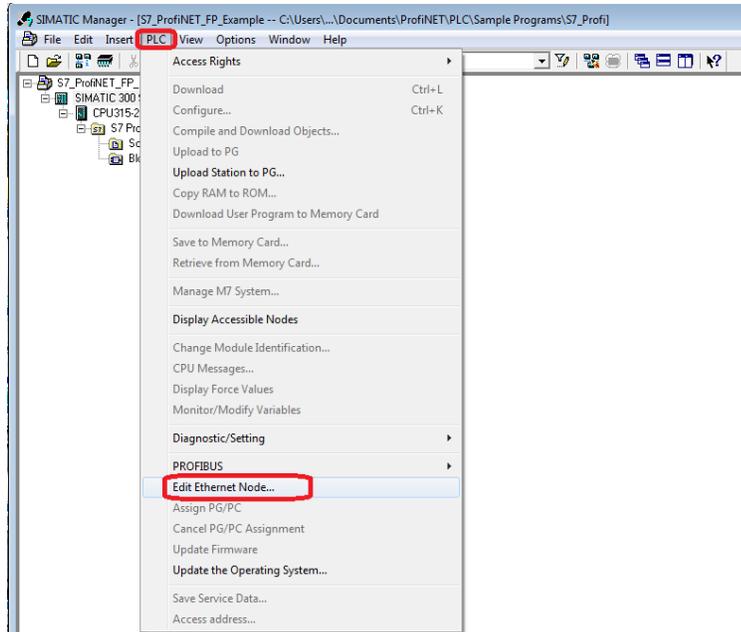


Figure 8-16: Configuration via DCP

Using the Browse function as shown below initiates the discovery of MAC addresses on the network. Select the MAC address you wish to work with by clicking on it, then clicking the “OK” button to continue.

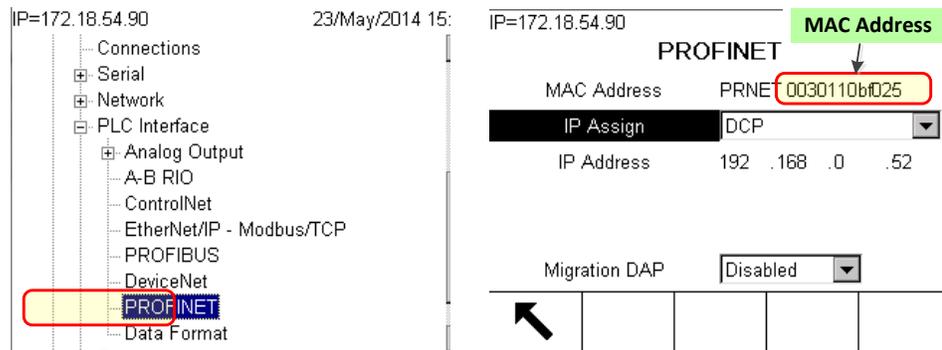


Figure 8-17: Where to find the IND780’s PROFINET MAC address

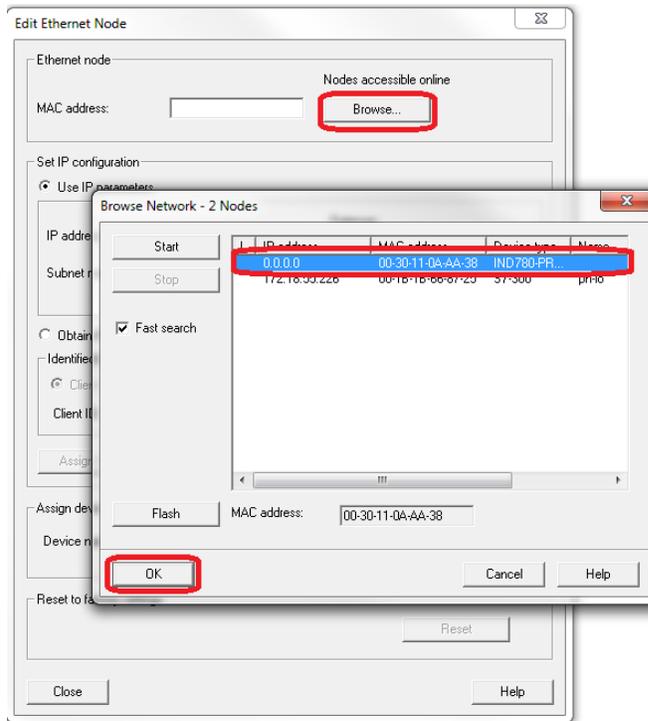


Figure 8-18: Select the IND780's MAC address

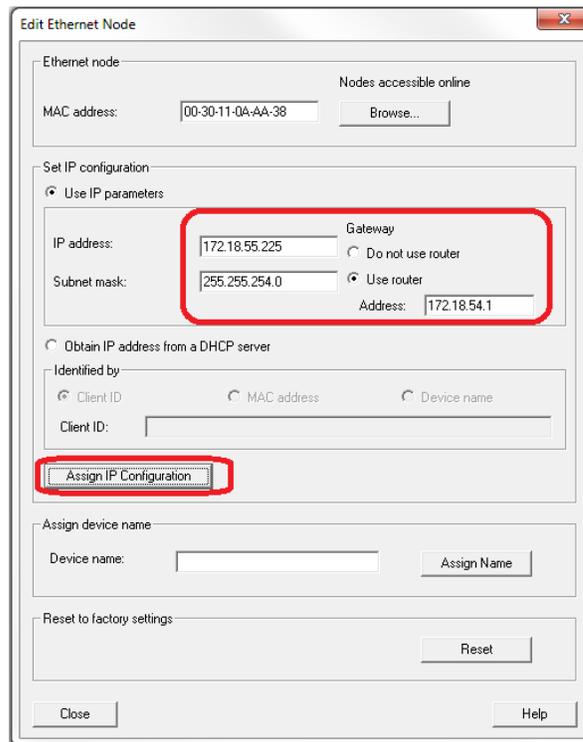


Figure 8-19: Assign the IND780's IP address via DCP

**Figure 8-20: Assign the IND780's Device Name via DCP**

After the message is received that the parameters have been successfully transferred, you may close the form. If the Data Format in the IND780 has been properly configured and the Network properly configured in the Siemens "HW Config." window, the module should start communicating with the PLC.

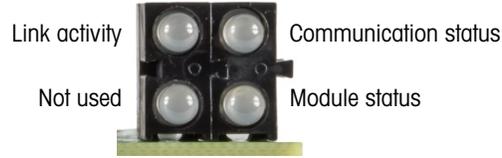
## 8.8. Troubleshooting

If the IND780 does not communicate with the PLC, do the following:

- Confirm that both the IP Address Configuration, and the Device name Configuration have been assigned in the PLC (note that the Device Name must always be assigned using DCP). Cycle power on the IND780 to ensure that any updated settings take effect.
- Check wiring and network connections.
- Confirm that the IND780 settings for data type and IP Address assignment match those in the PLC and that each IND780 has a unique address.
- Replace the PROFINET interface kit if the problem persists.
- If the communication kit was changed from another type, like Ethernet/IP, ControlNet or Remote I/O, a master reset of the IND780 may need to be performed.

### 8.8.1. Status LEDs

The PROFINET interface card has four status LEDs indicators to indicate communication and fault status of the card. Figure 8-2 indicates the location of these LEDs, and Figure 8-21 shows the array of the LEDs on the card. Table 8-16 explains the meaning of the indicators.



**Figure 8-21: PROFINET Status Indicator LEDs**

**Table 8-16: PROFINET LEDs Status Indications**

LED #	State	Status
1 - Link Activity	Off	No link (or no power)
	Green	Connected to an Ethernet network
	Green, flashing	Receiving/Transmitting data
2 - Communication Status	Off	Off Line – No connection with IO Controller
	Green	On line, RUN - Connection with IO Controller established - IO Controller is in RUN State
	Green, 1 flash	On line, STOP - Connection with IO Controller established - IO Controller is in STOP State
3 - Module Status	Off	No power or not initialized
	Green	Initialized, no error
	Green, 1 flash	Diagnostic data available
	Green, 2 flashes	Blink. Used by an engineering tool to identify the module.
	Red, 1 flash	Configuration Error - Too many modules/submodules - I/O size derived from IO Controller configuration is too large - Configuration mismatch (no module, wrong module)
	Red, 3 flashes	No Station Name or no IP address assigned
	Red, 4 flashes	Internal error
	Off	No power or not initialized
4 – Not Used	-	-

## 8.9. PROFINET Option Kit

No spare parts are associated with the PROFINET option kit. The kit part number is 30130721. Table 8-17 shows what the kit includes.

**Table 8-17: PROFINET Option Kit**

Description	Qty.
Installation Instructions	1
PCB package	1
Clamp-on ferrite	1
M3 x 8 screws	3

## 8.10. Siemens S7-300 Programming Examples

The following Figures show sample screen images of ladder logic programming examples for SIMATIC Step 7 software (version V5.5 + SP3).

- **Note:** The Utilities folder on the documentation CD (part number 64057241) contains complete versions of the examples. These screen images are provided for illustrative purposes only.

The following SIMATIC Step 7 screens for Integer, Division and Floating Point data formats only show an example of a particular Input and Output size configuration. The Connection Parameters I/O sizes must be appropriately configured with reference to the number of slots assigned in the IND780 PLC Data Format Message Slots settings. Table 8-18 and Table 8-19 show the relationship between the IND780 message slots and the SIMATIC Step 7 I/O sizing for Integer, Division and Floating Point data formats.

**Table 8-18: Message Slot and PLC I/O Sizes (Integer/ Division)**

IND780 Integer/ Division Data		
Message Slots	Bytes (8 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780
1	4	4
2	8	8
3	12	12
4	16	16
5	20	20
6	24	24
7	28	28
8	32	32
9	36	36

IND780 Integer/ Division Data		
Message Slots	Bytes (8 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780
10	40	40
11	44	44
12	48	48

**Table 8-19: Message Slot and PLC I/O Sizes (Floating Point)**

IND780 Floating Point Data		
Message Slots	Bytes (8 Bit)	
	IND780 >> PLC Input	PLC Output >> IND780
1	8	8
2	16	14
3	24	20
4	32	26
5	40	32
6	48	38
7	56	44
8	64	50
9	72	56
10	80	62
11	88	68
12	96	74

## 8.10.1. Floating Point Mode Program Example

UDT1 -- "IND\_Float\_Data" -- S7\_ProfiNET\_Example\SIMATIC 300 Station(1)\CPU315-2 PN/DP(1)\...\UDT1

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	CMD_Response	"Command_Respons		Command Response word from Terminal
+2.0	Data	REAL	0.000000e+000	Floating Point Data from Terminal
+6.0	Scale_Status	"IND_FP_Status"		Scale Status from Terminal
=8.0		END_STRUCT		

UDT2 -- "Command\_Response" -- S7\_ProfiNET\_Example\SIMATIC 300 Station(1)\CPU315-2 PN/DP(1)\...\UDT2

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Reserved0	BOOL	FALSE	Reserved
+0.1	Reserved1	BOOL	FALSE	Reserved
+0.2	Reserved2	BOOL	FALSE	Reserved
+0.3	Reserved3	BOOL	FALSE	Reserved
+0.4	Reserved4	BOOL	FALSE	Reserved
+0.5	Reserved5	BOOL	FALSE	Reserved
+0.6	Reserved6	BOOL	FALSE	Reserved
+0.7	Reserved7	BOOL	FALSE	Reserved
+1.0	FP_Input_Ind1	BOOL	FALSE	Floating Point Indicator Bit 1
+1.1	FP_Input_Ind2	BOOL	FALSE	Floating Point Indicator Bit 2
+1.2	FP_Input_Ind3	BOOL	FALSE	Floating Point Indicator Bit 3
+1.3	FP_Input_Ind4	BOOL	FALSE	Floating Point Indicator Bit 4
+1.4	FP_Input_Ind5	BOOL	FALSE	Floating Point Indicator Bit 5
+1.5	Data_Integrity_1	BOOL	FALSE	Floating Point Data integrity bit #1
+1.6	Cmd_Ack_1	BOOL	FALSE	Command Acknowledgement bit 1
+1.7	Cmd_Ack_2	BOOL	FALSE	Command Acknowledgement bit 2
=2.0		END_STRUCT		

UDT3 -- "IND\_FP\_Status" -- S7\_ProfiNET\_Example\SIMATIC 300 Station(1)\CPU315-2 PN/DP(1)\...\UDT3

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Feed	BOOL	FALSE	Feed Running
+0.1	Fast_Feed	BOOL	FALSE	Fast Feed Running
+0.2	Tolerance_OK	BOOL	FALSE	Feed within Tolerance
+0.3	Under_Neg_Tol	BOOL	FALSE	Feed under Negative Tolerance
+0.4	Over_Pos_Tol	BOOL	FALSE	Feed over Positive Tolerance
+0.5	Selected_Scale	BOOL	FALSE	Scale currently selected on display.
+0.6	Custom_Bit1	BOOL	FALSE	Bit 1 reserved for Task Expert Application Usage
+0.7	Custom_Bit2	BOOL	FALSE	Bit 2 reserved for Task Expert Application Usage
+1.0	Enter_Key	BOOL	FALSE	Enter Key press on terminal detected.
+1.1	Input_1	BOOL	FALSE	Digital I/O Card Input #1 current state
+1.2	Input_2	BOOL	FALSE	Digital I/O Card Input #2 current state
+1.3	Input_3	BOOL	FALSE	Digital I/O Card Input #3 current state
+1.4	Motion	BOOL	FALSE	Scale in Motion
+1.5	Net_Mode	BOOL	FALSE	Scale currently in NET mode
+1.6	Data_Integrity_2	BOOL	FALSE	Floating Point Data Integrity bit #2
+1.7	Data_OK	BOOL	FALSE	Scale okay for normal operation.
=2.0		END_STRUCT		

**Figure 8-22: UDT Definitions used in the Floating Point sample program**

The UDT's (User Defined Types) used in the sample program are:

UDT1 = IND780 Float Data. Format of the Floating Point data that comes back from the terminal, including the status registers with their supporting UDT's.

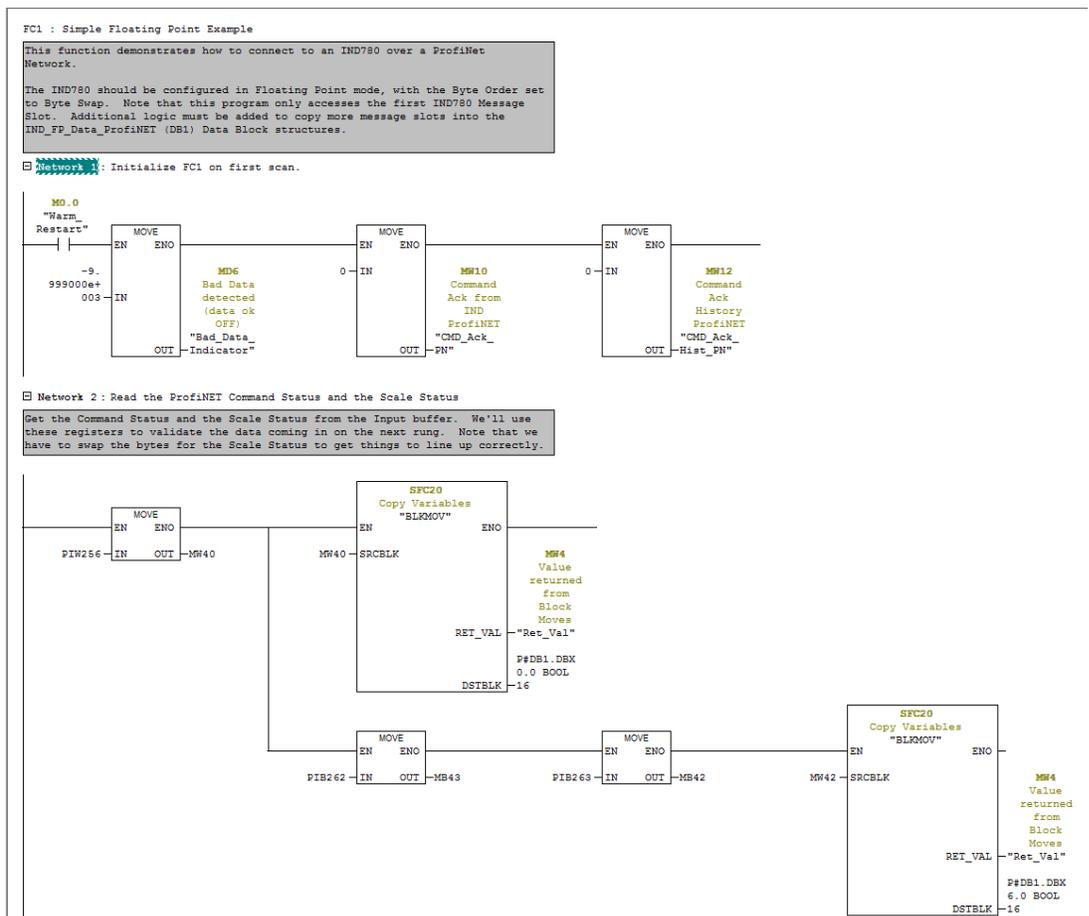
UDT2 = Command Response. Status register indicating the response of the IND780 Terminal to a command sent to it over the Field Bus.

UDT3 = Indicator Floating Point Status: Indicates the state of the measuring device (Scale or Flowmeter).

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Msg_Slot1	"IND_Float_Data"		IND780 Message Slot 1 (see PLC Data Format in IND780's Setup)
+8.0	Msg_Slot2	"IND_Float_Data"		IND780 Message Slot 2 (see PLC Data Format in IND780's Setup)
+16.0	Msg_Slot3	"IND_Float_Data"		IND780 Message Slot 3 (see PLC Data Format in IND780's Setup)
+24.0	Msg_Slot4	"IND_Float_Data"		IND780 Message Slot 4 (see PLC Data Format in IND780's Setup)
+32.0	Msg_Slot5	"IND_Float_Data"		IND780 Message Slot 5 (see PLC Data Format in IND780's Setup)
+40.0	Msg_Slot6	"IND_Float_Data"		IND780 Message Slot 6 (see PLC Data Format in IND780's Setup)
+48.0	Msg_Slot7	"IND_Float_Data"		IND780 Message Slot 7 (see PLC Data Format in IND780's Setup)
+56.0	Msg_Slot8	"IND_Float_Data"		IND780 Message Slot 8 (see PLC Data Format in IND780's Setup)
+64.0	Msg_Slot9	"IND_Float_Data"		IND780 Message Slot 9 (see PLC Data Format in IND780's Setup)
+72.0	Msg_Slot10	"IND_Float_Data"		IND780 Message Slot 10 (see PLC Data Format in IND780's Setup)
+80.0	Msg_Slot11	"IND_Float_Data"		IND780 Message Slot 11 (see PLC Data Format in IND780's Setup)
+88.0	Msg_Slot12	"IND_Float_Data"		IND780 Message Slot 12 (see PLC Data Format in IND780's Setup)
=96.0		END_STRUCT		

**Figure 8-23: Data Block 1 (DB1) used in the Floating Point sample program**

Data Block 1 (DB1) has defined all twelve (12) possible message slots for the IND780. This is NOT necessary for proper operation of the program, but does allow data to be stored in the data block if the program is designed to collect data from all 12 slots.

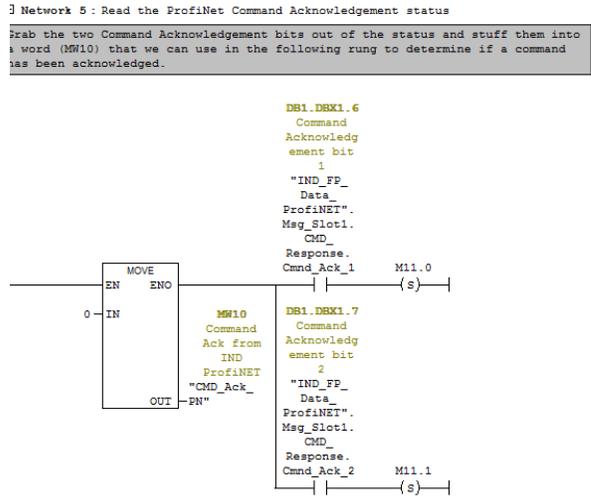


**Figure 8-24: Networks 1 & 2 of FC1**

When the program first starts up, Network 1 initializes it with its starting values.

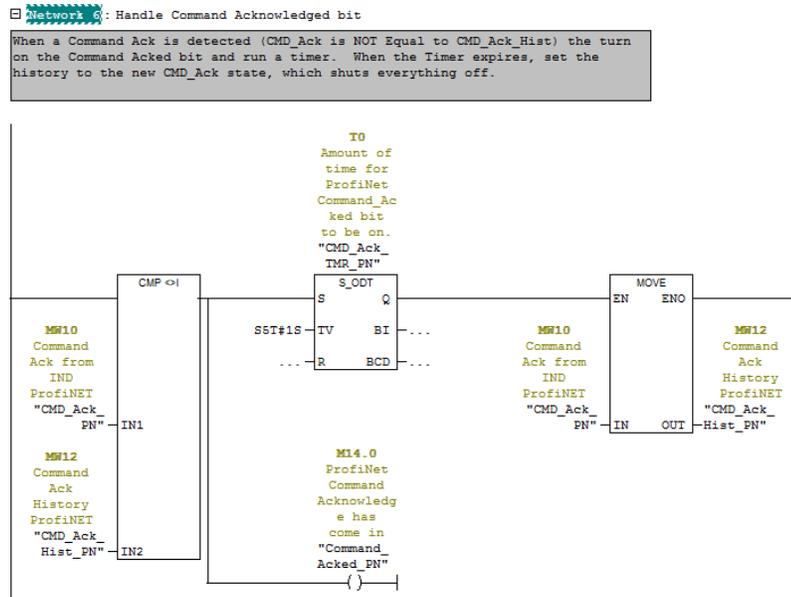


Network 4 acts on the Data OK bit if it is off, indicating that the instrument is not in a valid mode to send data. In that case, set the output to -9999.0 to flag the operator that something is wrong. Set the Data Displayed indicator to -1 to flag that something is wrong with the data coming back.



**Figure 8-27: Network 5 of FC1**

Network 5 reads the Command Acknowledge bits from the IND780 and turns the result into an Integer word from the returned bits. The integer will be used in the following networks to act on any commands that are sent.



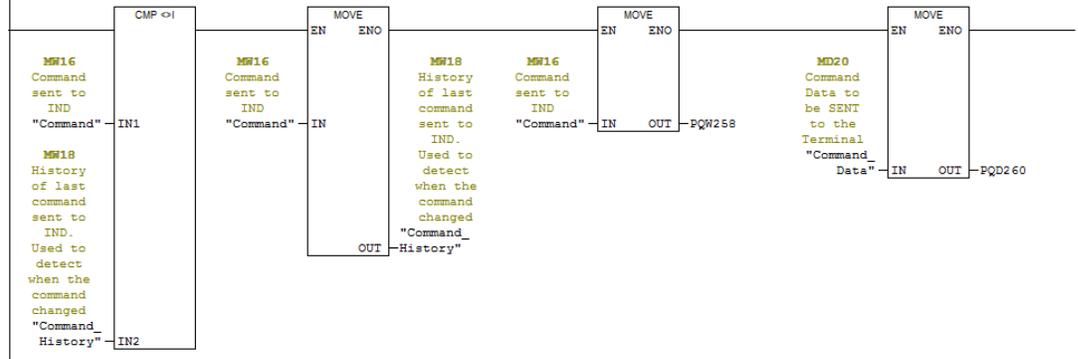
**Figure 8-28: Network 6 of FC1**

Network 6 looks for a change in the Command Acknowledge state. If one is found, it turns on the Command\_Acked\_PN bit and runs a timer. When the timer expires, the history is updated, which in turn causes the Command\_Acked\_PN bit to turn off. The Command\_Acked\_PN bit is used later when processing commands to the IND780.

□ Network 7: Update the Command to the IND

The IND only acts on a command if it sees a change from the previous command. That means if you need to send another Tare after the first Tare, there must be another command between them - such as Display Net Weight.

This rung waits until the comand changes, then it writes the Command, and the Command Data to the PLC's output buffer to the IND. It then updates the history.



**Figure 8-29: Network 7 of FC1**

Network 7 looks for a change in the Command being sent to the IND780. If a new command has been issued the move it to the output buffer and update the history. Programming the command in this manner allows the freedom of manually updating the command output buffer if desired for troubleshooting purposes.

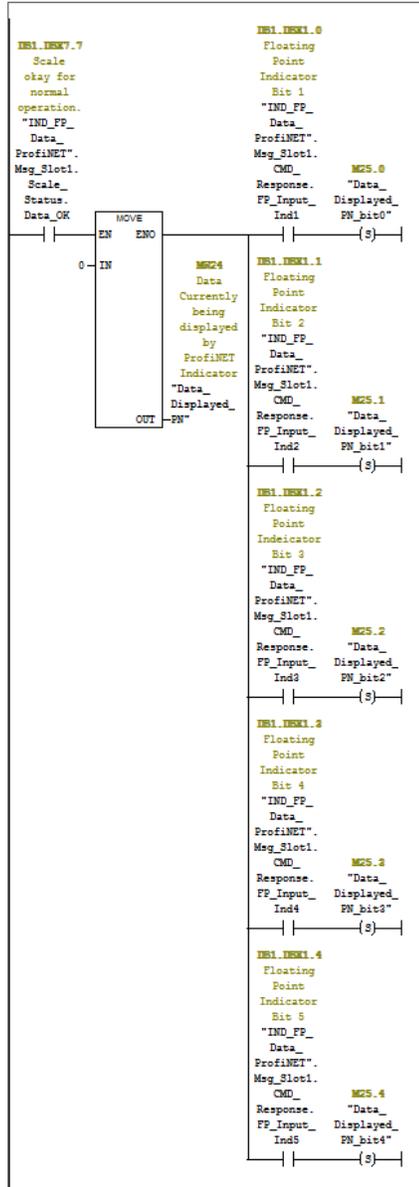


Figure 8-30: Network 8 of FC1

Network 8 reads the Floating Point Indicator bits in the Command Status Word and creates an Integer from the result that is used later to determine what kind of data is being returned from the IND780.

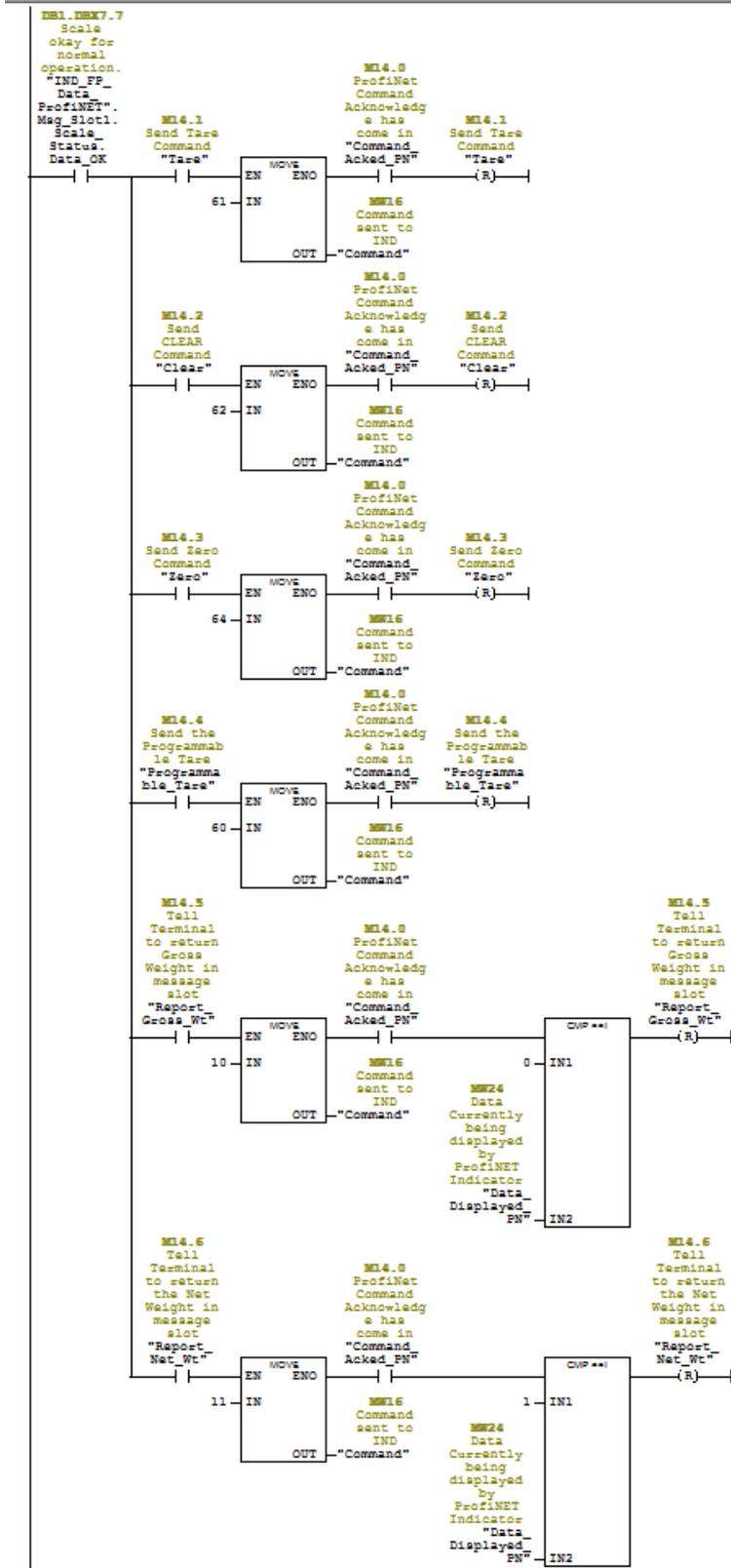


Figure 8-31: Network 9 of FC1

Network 9 issues commands to the IND780 as the user sets the desired command bits. It then waits for the commands to be acknowledged before clearing the bits that were set. If a different type of data for display is being requested, then wait until the IND780 indicates that the correct data is being displayed before clearing the command bit.

8.10.1.1. Running the Sample Program

The sample program can be run from the Variable Access Table as shown in Figure 8-32 and Figure 8-33. below.

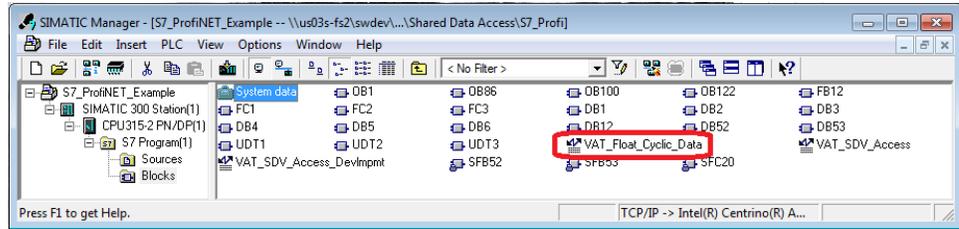


Figure 8-32: VAT\_Float\_Cyclic\_Data

VAT\_Float\_Cyclic\_Data is a Variable Access Table that allows the user to monitor the processed cyclic data and issue commands to the IND780 as shown below:

Address	Symbol	Display format	Status value	Modify value
// Filtered Data as read from unit.				
2	DB1.DBX 2	"IND_FP_Data_ProfiNET".Msg_Slot1.Data	FLOATING_POINT	220.0
// Incoming Data Handling				
5	DB1.DBX 1.5	"IND_FP_Data_ProfiNET".Msg_Slot1.CMD_Response.Data_Integrity_1	BOOL	true
6	DB1.DBX 7.6	"IND_FP_Data_ProfiNET".Msg_Slot1.Scale_Status.Data_Integrity_2	BOOL	true
7	DB1.DBX 7.7	"IND_FP_Data_ProfiNET".Msg_Slot1.Scale_Status.Data_OK	BOOL	true
// Data code for data currently being displayed.				
// 0=Gross, 1=Net, 2=Tare, 3=Fine Gross, 4=Fine Tare, 5=Rate, 6=Rate, 30=CMD Success, 31=CMD Failed				
11	MW 24	"Data_Displayed_PN"	DEC	0
// Command Ack Handling				
14	DB1.DBX 1.6	"IND_FP_Data_ProfiNET".Msg_Slot1.CMD_Response.Cmd_Ack_1	BOOL	false
15	DB1.DBX 1.7	"IND_FP_Data_ProfiNET".Msg_Slot1.CMD_Response.Cmd_Ack_2	BOOL	true
16	MW 10	"CMD_Ack_PN"	DEC	2
17	M 14.0	"Command_Acked_PN"	BOOL	false
// Commands being sent - for monitoring only. Use the Command Trigger bits below to send commands.				
20	MW 16	"Command"	DEC	10
21	MD 20	"Command_Data"	FLOATING_POINT	120.0 120.0
// Command Trigger Bits				
24	M 14.1	"Tare"	BOOL	false
25	M 14.2	"Clear"	BOOL	false
26	M 14.3	"Zero"	BOOL	false
27	M 14.4	"Programmable_Tare"	BOOL	false
28	M 14.6	"Report_Net_Wt"	BOOL	false
29	M 14.5	"Report_Gross_Wt"	BOOL	false

Filtered Floating Point Data from IND780

Floating Point Data Filter bits: Data\_OK bit must be on and Data Integrity bits 1&2 must be the same polarity for the current Floating Point data sample to be valid.

Command Data. This value is updated when new data needs to be sent to the IND780 with a command.

Command Bits. Set these bits to trigger their associated commands to be sent. The bits will be cleared when the IND780 acknowledges the command.

Figure 8-33: VAT\_Float\_Cyclic\_Data Description

## 8.10.2. Integer Mode Program Example

UDT1 -- "IND\_Integer\_Data" -- S7\_ProfiNet\_Int\_Example\SIMATIC 300 Station\CPU315-2 PN/DP(1)\...\UDT1

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Data	REAL	0.000000e+000	Integer Data from Terminal Multiplied by a Scaling Value
+4.0	Scale_Status	"IND_Int_Status"		Scale Status from Terminal
=6.0		END_STRUCT		

UDT2 -- "IND\_Int\_Status" -- S7\_ProfiNet\_Int\_Example\SIMATIC 300 Station\CPU315-2 PN/DP(1)\...\UDT2

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Feed	BOOL	FALSE	Feed Running
+0.1	Fast_Feed	BOOL	FALSE	Fast Feed Running
+0.2	Tolerance_OK	BOOL	FALSE	Feed within Tolerance
+0.3	Under_Low_Tol	BOOL	FALSE	Feed under Low Tolerance
+0.4	Over_High_Tol	BOOL	FALSE	Feed over High Tolerance
+0.5	Comparator_3	BOOL	FALSE	Comparator 3 Active State
+0.6	Comparator_2	BOOL	FALSE	Comparator 2 Active State
+0.7	Comparator_1	BOOL	FALSE	Comparator 1 Active State
+1.0	Enter_Key	BOOL	FALSE	Enter Key press on terminal detected.
+1.1	Input_1	BOOL	FALSE	Digital I/O Card Input #1 current state
+1.2	Input_2	BOOL	FALSE	Digital I/O Card Input #2 current state
+1.3	Input_3	BOOL	FALSE	Digital I/O Card Input #3 current state
+1.4	Motion	BOOL	FALSE	Scale in Motion
+1.5	Net_Mode	BOOL	FALSE	Scale currently in NET mode
+1.6	Update_In_Progress	BOOL	FALSE	Terminal Updating PLC Data - Do NOT use data!
+1.7	Data_OK	BOOL	FALSE	Scale okay for normal operation.
=2.0		END_STRUCT		

**Figure 8-34: UDT Definitions used in the Integer sample program**

The UDT's (User Defined Types) used in the sample program are:

UDT1 = IND780 Integer Data. Format of the Floating Point data that comes back from the terminal, including the status registers with their supporting UDT's.

UDT2 = Indicator Integer Status: Indicates the state of the measuring device (Scale or Flowmeter).

DB1 -- "IND\_Int\_Data\_ProfiNET" -- S7\_ProfiNet\_Int\_Example\SIMATIC 300 Station\CPU315-2 PN/DP(1)\...\DB1

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Msg_Slot1	"IND_Integer_Data"		Message Slot 1
+6.0	Msg_Slot2	"IND_Integer_Data"		Message Slot 3
+12.0	Msg_Slot3	"IND_Integer_Data"		Message Slot 3
+18.0	Msg_Slot4	"IND_Integer_Data"		Message Slot 4
+24.0	Msg_Slot5	"IND_Integer_Data"		Message Slot 5
+30.0	Msg_Slot6	"IND_Integer_Data"		Message Slot 6
+36.0	Msg_Slot7	"IND_Integer_Data"		Message Slot 7
+42.0	Msg_Slot8	"IND_Integer_Data"		Message Slot 8
+48.0	Msg_Slot9	"IND_Integer_Data"		Message Slot 9
+54.0	Msg_Slot10	"IND_Integer_Data"		Message Slot 10
+60.0	Msg_Slot11	"IND_Integer_Data"		Message Slot 11
+66.0	Msg_Slot12	"IND_Integer_Data"		Message Slot 12
=72.0		END_STRUCT		

**Figure 8-35: Data Block 1 (DB1) used in the Integer sample program**

Data Block 1 (DB1) has defined all twelve (12) possible message slots for the IND780. This is NOT necessary for proper operation of the program, but does allow data to be stored in the data block if the program is designed to collect data from all 12 slots.

FC1 : Simple Integer Example

This function demonstrates how to connect to an IND780 over a Profinet Network.  
 The IND780 should be configured in Integer mode, with the Byte Order set to Byte Swap. Note that this program only accesses the first IND780 Message Slot. Additional logic must be added to copy more message slots into the IND\_Int\_Data\_ProfiNET (DB1) Data Block structures.

Network 1: Initialize FC1 on first scan.

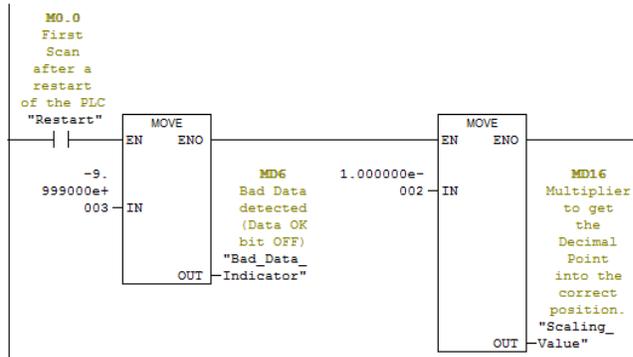


Figure 8-36: Network 1 of FC1

Network 1 initializes the program when it first starts up with its starting values.

Network 2: Copy the Input Status to the Scale Status Bits in DB1

First, swap the bytes of the status word to get them to line up properly. The BLKMOV function can't copy directly from an Input Buffer, so we have to first move it to MW26. Then we can use the BLKMOV, which will map the bits to their proper locations in the DB1 Scale Status area.

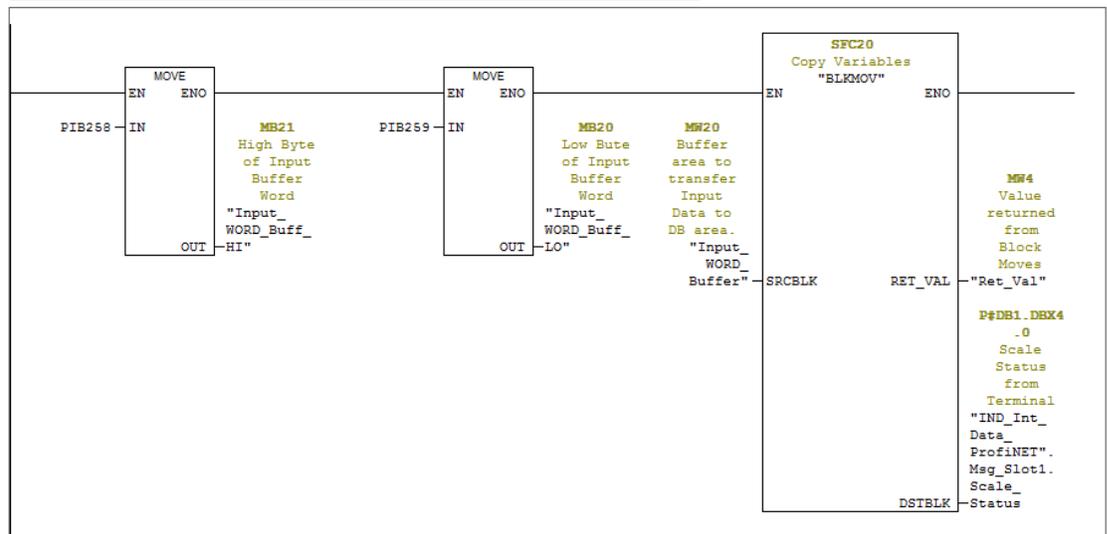


Figure 8-37: Network 2 of FC1

Network 2 Swaps the bytes of the Scale Status Word, then uses a BLKMOV to map the Status Bits to their proper locations for later use.

Network 3: Get the data from the Terminal and Normalize it to the Increment

We need to 'Normalize' the incoming Integer Data by multiplying it by the multiplier that is defined for the scale (that value is Hard Coded in Network 1 above, and is based on the resolution defined by the terminal's increment setting contained in the scale setup that is in the IND terminal itself). But first, we have to convert the Integer data coming back to a Double Integer, and then convert the Double Integer into a Real data type. After all of that, we can finally multiply the data by the Increment and store the result in the Message Slot data area.

Note that the Data OK bit MUST be ON, and the Update In Progress bit MUST be OFF, or the data coming back should be ignored.

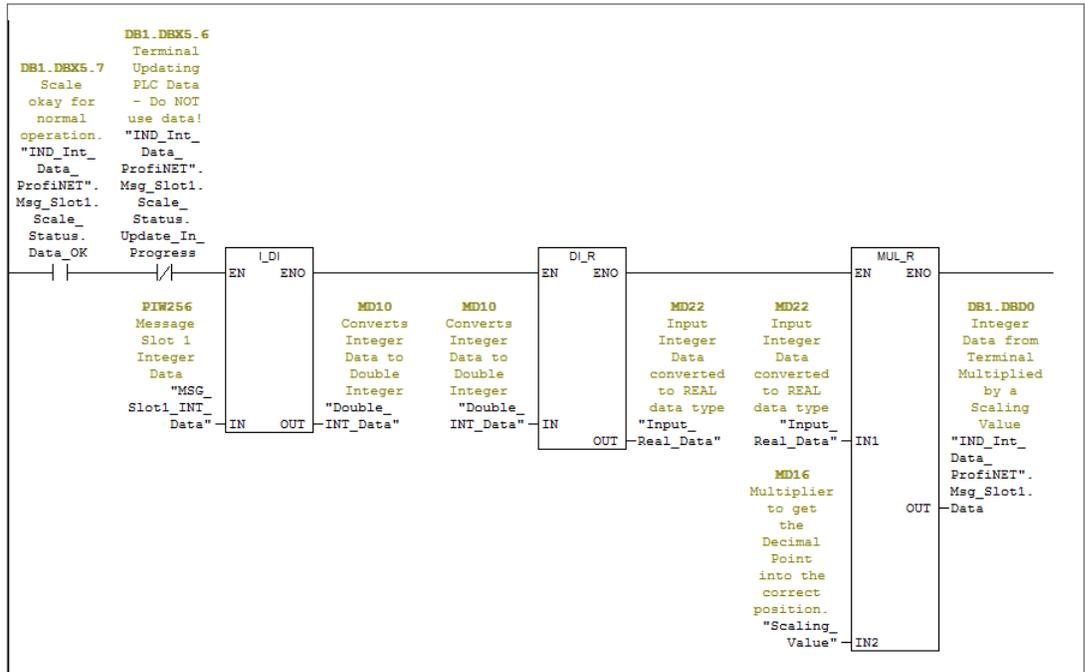
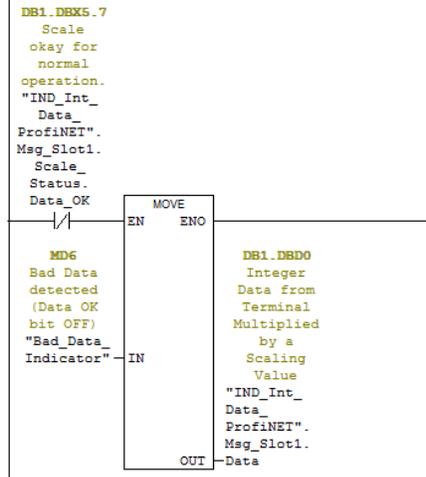


Figure 8-38: Network 3 of FC1

Network 3 filters the Data Input with the Data OK and the Update In Progress bits from the Status word. If the Data OK bit is off, or the Update in Progress bit is set, then the data may be invalid and should be discarded. If the filter bits are okay, then the Integer value needs to be converted to a Real data type for later use. To do that, the program must first convert the Integer to a Double Integer, and then convert the Double Integer to Real value. Finally, the Real value is multiplied by a Scaling value that puts the decimal point into the same place that the Terminal Display uses. The value is then transferred to DB1.

□ Network 4 : Profinet Data OK bit is OFF

If the Data OK bit is off, signal the process by setting the data to a known bad value that flags everyone that something is wrong at the scale. Also, set the Data Displayed variable to -1 to flag all other routines that there is a problem.



**Figure 8-39: Network 4 of FC1**

Network 4 looks at the Data OK bit. If the Data OK bit is off, then the terminal is not in a proper state to deliver valid data. In that case, the "Bad Data" indicator value is written to DB1 instead of the data.

□ Network 5 : Sample Commands to the Terminal.

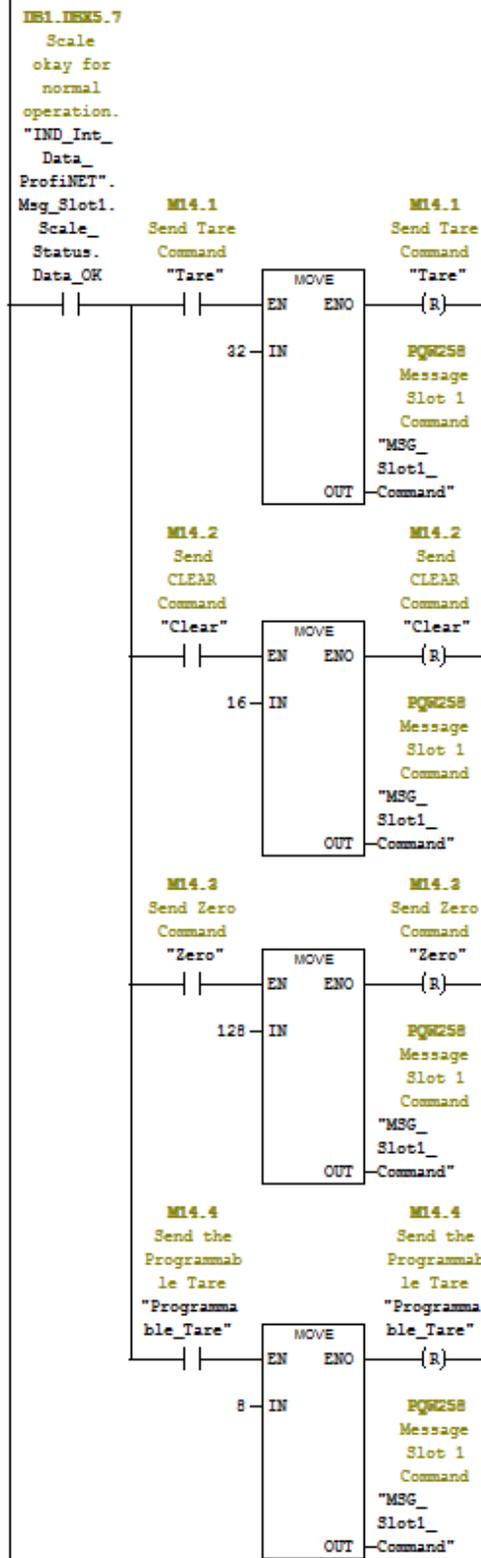


Figure 8-40: Network 5 of FC1

If the terminal is in a state to receive commands (the Data OK bit is on) then Network 5 looks at the command trigger bits which could be set either by other networks in the program, or directly by a user from a VAT Table. If a command trigger bit is found to be on, write the command value to the Command Output and clear the trigger bit. This network is intended as a sample for how commands may be sent to the Terminal.

Network 6 : More sample commands to the terminal

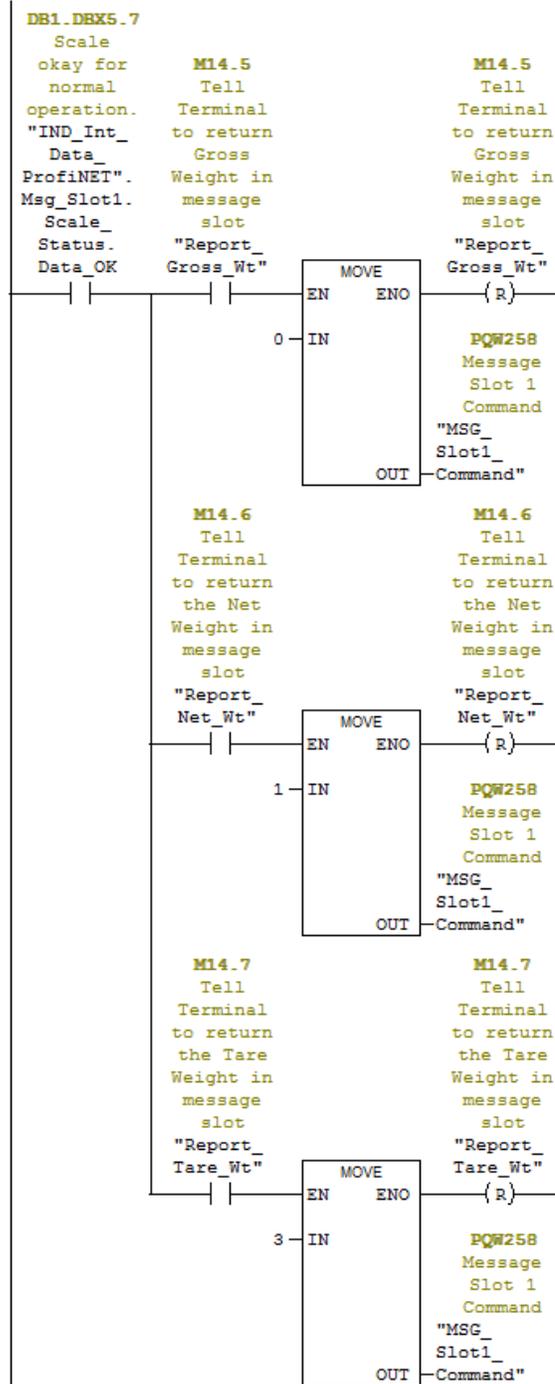


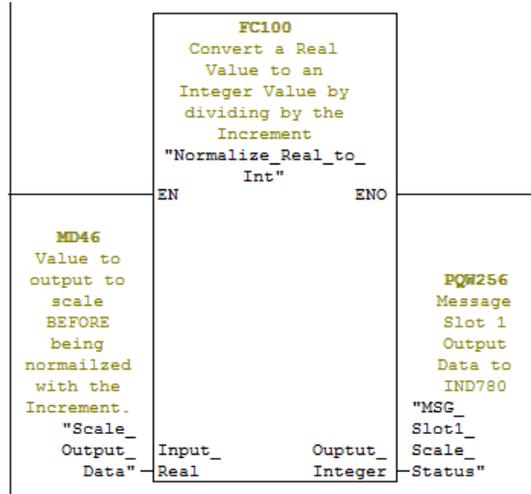
Figure 8-41: Network 6 of FC1

Network 6 shows some additional commands that could be sent to the terminal. Note that these commands are also filtered with the Data OK bit.

```

Network 7 : Output the Normalized Data to the Terminal.

Take the Floating Point Data that was written to the Normalized_Data_Out
variable and divide it by the Increment value to generate an Integer Value that
can be output to the IND780 Terminal.
    
```

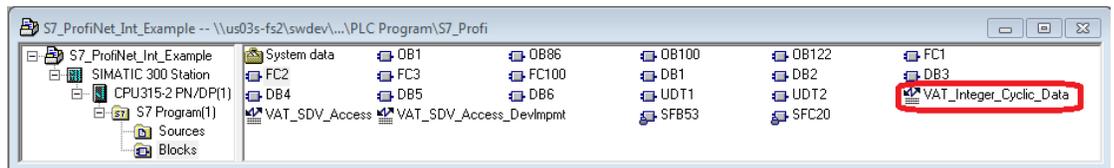


**Figure 8-42: Network 7 of FC1**

Network 7 shows how Real Typed data can be converted and sent as an Integer value to the terminal to supplement commands. This data can be used to pre-load a Tare value, set a Target, set up Tolerances, etc. Note that FC100 is provided with the sample program, but is not covered here.

8.10.2.1. Running the Sample Program

The sample program can be run from the Variable Access Table as shown in Figure 8-43 and Figure 8-44 below.



**Figure 8-43: VAT\_Integer\_Cyclic\_Data**

VAT\_Integer\_Cyclic\_Data is a **Variable Access Table** that allows the user to monitor the processed cyclic data and issue commands to the IND780 as shown below:

Address	Symbol	Display format	Status value	Modify value
1	// Filtered Data as read from unit.			
2	DB1.DBX 0 "IND_Int_Data_ProfiNET".Msg_Slot1.Data	FLOATING_POINT	220.1	
3				
4	// Incoming Data Handling			
5	DB1.DBX 5.7 "IND_Int_Data_ProfiNET".Msg_Slot1.Scale_Status.Data_OK	BOOL	true	
6	DB1.DBX 5.6 "IND_Int_Data_ProfiNET".Msg_Slot1.Scale_Status.Update_In_Progress	BOOL	false	
7				
8	// Command Data to Output to Terminal			
9	MD 46 "Scale_Output_Data"	FLOATING_POINT	20.1	20.1
10				
11	// Command Trigger Bits			
12	M 14.1 "Tare"	BOOL	false	
13	M 14.2 "Clear"	BOOL	false	
14	M 14.3 "Zero"	BOOL	false	
15	M 14.4 "Programmable_Tare"	BOOL	false	
16	M 14.5 "Report_Gross_Wt"	BOOL	false	
17	M 14.6 "Report_Net_Wt"	BOOL	false	
18	M 14.7 "Report_Tare_Wt"	BOOL	false	
19				
20	// Raw Data Inputs			
21	PW 256 "MSG_Slot1_INT_Data"	DEC	2201	
22	PW 258 "MSG_Slot1_INT_Scale_Stat"	HEX	W#16#8100	
23	MD 16 "Scaling_Value"	FLOATING_POINT	0.1	0.1
24				

**Callouts:**

- Filtered Integer Data from IND780, converted to Floating Point.
- Integer Data Filter bits: Data\_OK bit must be on AND Update\_In\_Progress must be OFF for data to be valid.
- Command data associated with the commands below to send TO the IND780 (must first be converted to Integer).
- Command Bits. Set these bits to trigger their associated commands to be sent to the IND780. The bits will be cleared after the commands are sent. **NOTE:** Be sure to assign the data in "Scale\_Output\_Data" BEFORE setting these bits.
- Actual Integer data as returned from the IND780.
- Scale Status as returned from the IND780
- Hard-coded Scaling value used to convert Integer data to and from the IND780 to Floating Point values. Based on the resulting values from the Scale Increment Setting in the Scale Setup.

**Figure 8-44: VAT\_Integer\_Cyclic\_Data Description**

### 8.10.3. Shared Data Access Overview

In order to access Shared Data, a program must provide the following information to the Read and Write routines:

- Class Code
- Instance Number
- Attribute Number
- Length

This information can be found in the Shared Data Reference Manual (part number 6405910) for each Shared Data variable. For example, here is how you would find that information for a 'WT' type Shared Data variable:

## I. Scale Data

### Scale Functionality

#### Dynamic Scale Weight (WT)

Access: "Read Only." Access level is not customizable.	
Class Code: wt	Data Type: D
ControlNet Class Code: 68 hex	
Instances: 5	Instance 1 - 4 = Scale platforms 1 - 4 Instance 5 = Sum scale.

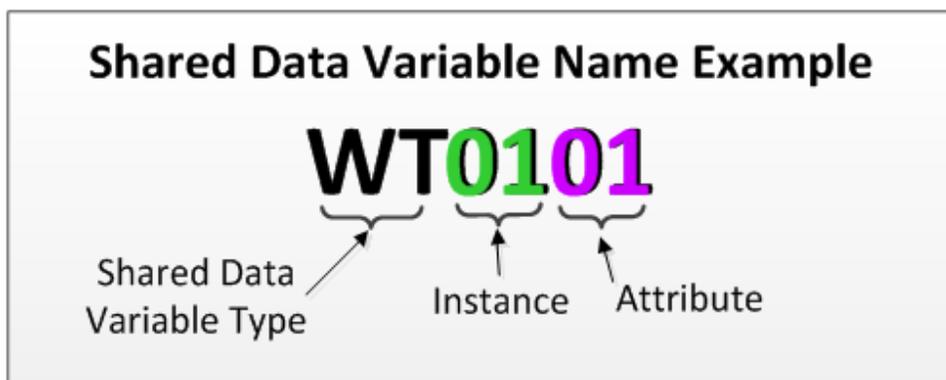
**Attributes:**

Note: The last two digits of each shared data variable is its attribute.

wt-00	Composite wt block	Struct	no	Composite of entire block
wt-01	Displayed Gross Weight	S13	rt	
wt-02	Displayed Net Weight	S13	rt	When user has enabled MinWeigh, the first character contains an "*" when the MinWeigh conditions are not met.
wt-03	Weight Units	S4	rt	lb pounds, kg kilograms, grams, oz ounces, oz Troy, dwt pennyweights, metric tons, ton, or custom units name
wt-04	Displayed Aux Gross Weight	S13	rt	
wt-05	Displayed Aux Net Weight	S13	rt	
wt-06	Aux Weight Units	S7	rt	lb pounds, kg kilograms, grams, oz ounces, lb-oz pounds & ounces, oz Troy, ounces, dwt pennyweights, metric tons, ton, or custom units name
wt-07	Rate Period	S2	rt	No, Sec, Min, Hour
wt-08	Displayed Rate	S13	rt	

**Figure 8-45: How to find the Shared Data Class, Instance, Attribute, and Length**

If you have used Shared Data Variable names before, then you are already familiar with using the Instance and Attribute in the name definition as shown in Figure 8-46:



**Figure 8-46: How a Shared Data Name is Constructed**

This information can help you in setting up your program to read or write the Shared Data variables that you need to access.

The method to access Shared Data in the IND780 Terminal is identical for both the Floating Point and Integer Modes using PROFINET. Both sample programs use identical routines and variables, and are covered here as a single process.

The Shared Data Access over PROFINET shown here makes use of the system SFB blocks SFB52 (RDREC DP) and SFB53 (WRREC DP) to read and write information to the IND780 using Acyclic messages over the PROFINET link.

Any Shared Data Variable access (both read and write) requires that the Class Code, Instance, Attribute, and Length all be embedded in a message written to the IND780. Note that this information is included in both message blocks below, and must be populated before the operation is started.

The figure shows two screenshots of the SIMATIC Manager interface. The top screenshot is titled 'DB5 -- "SDV\_Read\_Buffer" -- S7\_ProfiNet\_Int\_Example\SIMATIC 300 Station\CPU315-2 PN/DP(1)\...DB5'. It displays a table with the following data:

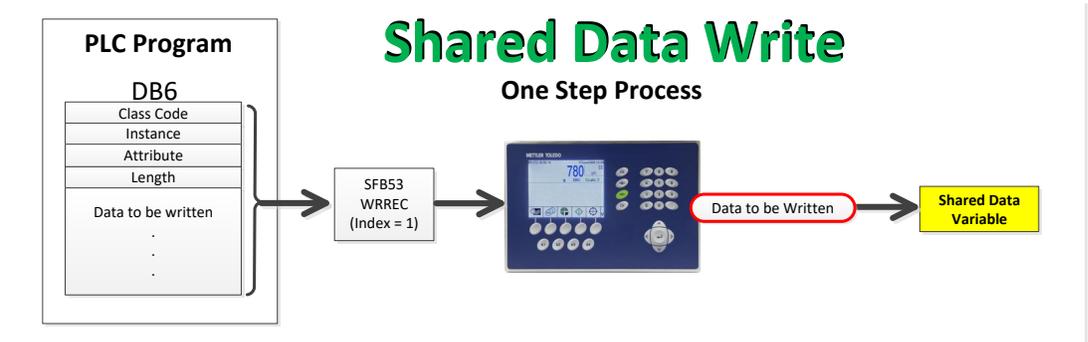
Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Class_Code	INT	0	Shared Data Variable Class Code
+2.0	Instance	INT	0	Instance (usually a scale or flowmeter number)
+4.0	Attribute	INT	0	Attribute pointer into the Class Code - selects specific Shared Data Variable
+6.0	Length	INT	0	Length of message read back (Read Command)
+8.0	Data_Buffer	ARRAY[0..25]		Data sent back from Terminal
+4.0		DWORD		
=112.0		END_STRUCT		

The bottom screenshot is titled 'DB6 -- "SDV\_Write\_Buffer" -- S7\_ProfiNet\_Int\_Example\SIMATIC 300 Station\CPU315-2 PN/DP(1)\...DB6'. It displays a table with the following data:

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Class_Code	INT	0	Shared Data Variable Class Code
+2.0	Instance	INT	0	Instance (usually a scale or flowmeter number)
+4.0	Attribute	INT	0	Attribute pointer into the Class Code - selects specific Shared Data Variable
+6.0	Length	INT	0	Length of message to send to Terminal (Write Command)
+8.0	Data_Buffer	ARRAY[0..25]		Data to send to Terminal
+4.0		DWORD		

**Figure 8-47: Read Buffer (DB5) and Write Buffer (DB6)**

A Shared Data Write is the simplest function since it requires only a call to SFB53 (WRREC) with the data to be written to the specified Shared Data variable, to complete the action.

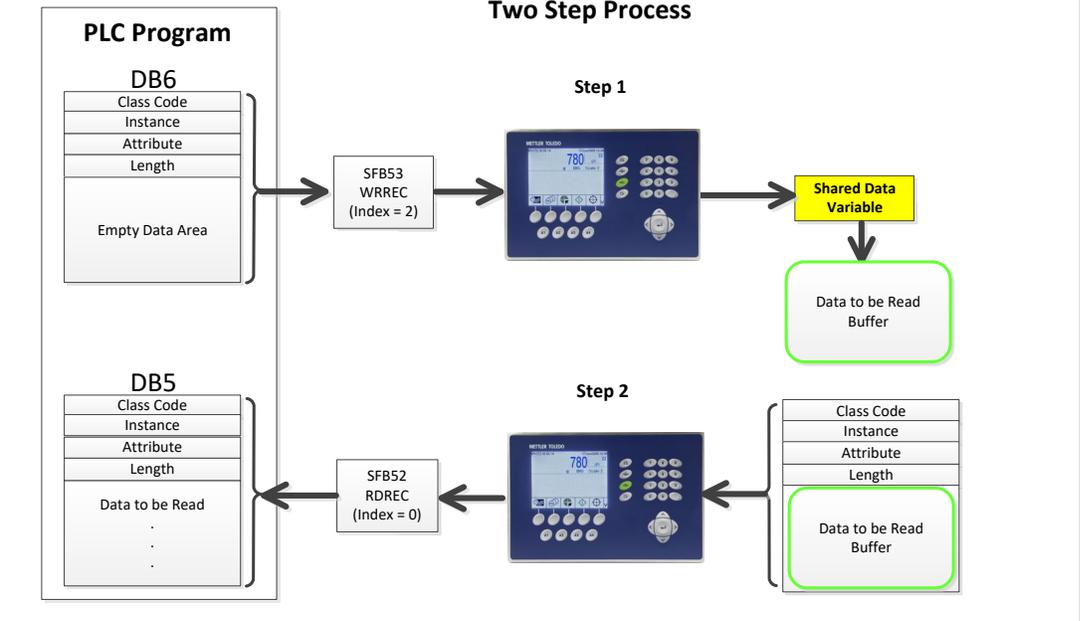


**Figure 8-48: Shared Data Write – One Step Process**

A Shared Data Read is a two-step process where a call to SFB53 (WRREC) must be done first to tell the IND780 what shared data variable to read. The write is then followed by a call to SFB52 (RDREC) to read the result back from the IND780.

# Shared Data Read

## Two Step Process



**Figure 8-49: Shared Data Read – Two Step Process**

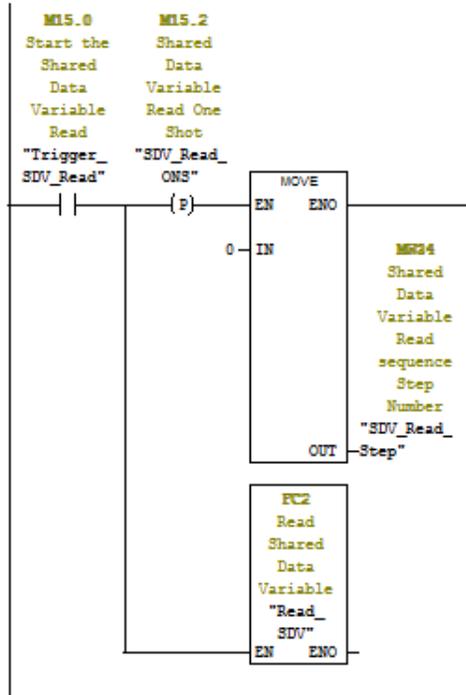
When the Shared Data Read process is complete, the Class Code, Instance, and Attribute returned from the IND780 are compared to the requested values to make sure that the proper request was fulfilled.

### 8.10.4. Shared Data Access Program Details

#### 8.10.4.1. OB1 Program details

In addition to the call to either the Floating Point or Integer cyclic data function (FC1), OB1 has the following logic added:

Network 2: If a Shared Data Access Read has been triggered, run it!



Network 3: If a Shared Data Access Write has been triggered, run it!

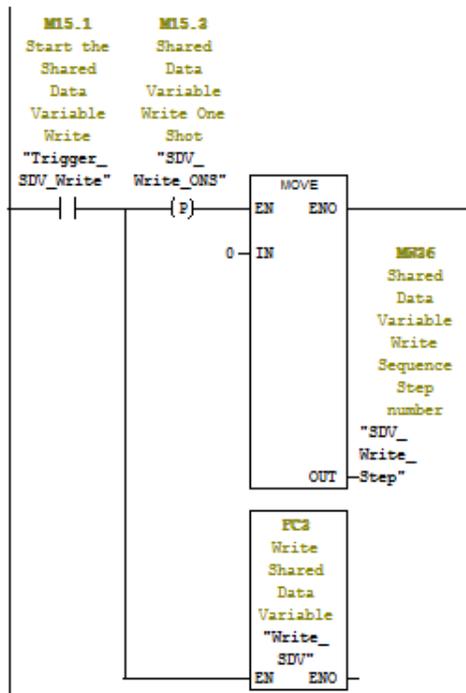


Figure 8-50: Networks 2 & 3 of OB1

Networks 2 & 3 of OB1 show how the functions that handle both the Read and Write to Shared Data in the IND780 can be called. In both cases, a bit can be set either by the program or by a user via a Variable Access Table (VAT) that triggers the function to run.

8.10.4.2. FC3 (Write Shared Data) Program details

Since FC3 is the simpler routine, and contains a subset of the instructions included in FC2 (Read Shared Data), we will look at it first.

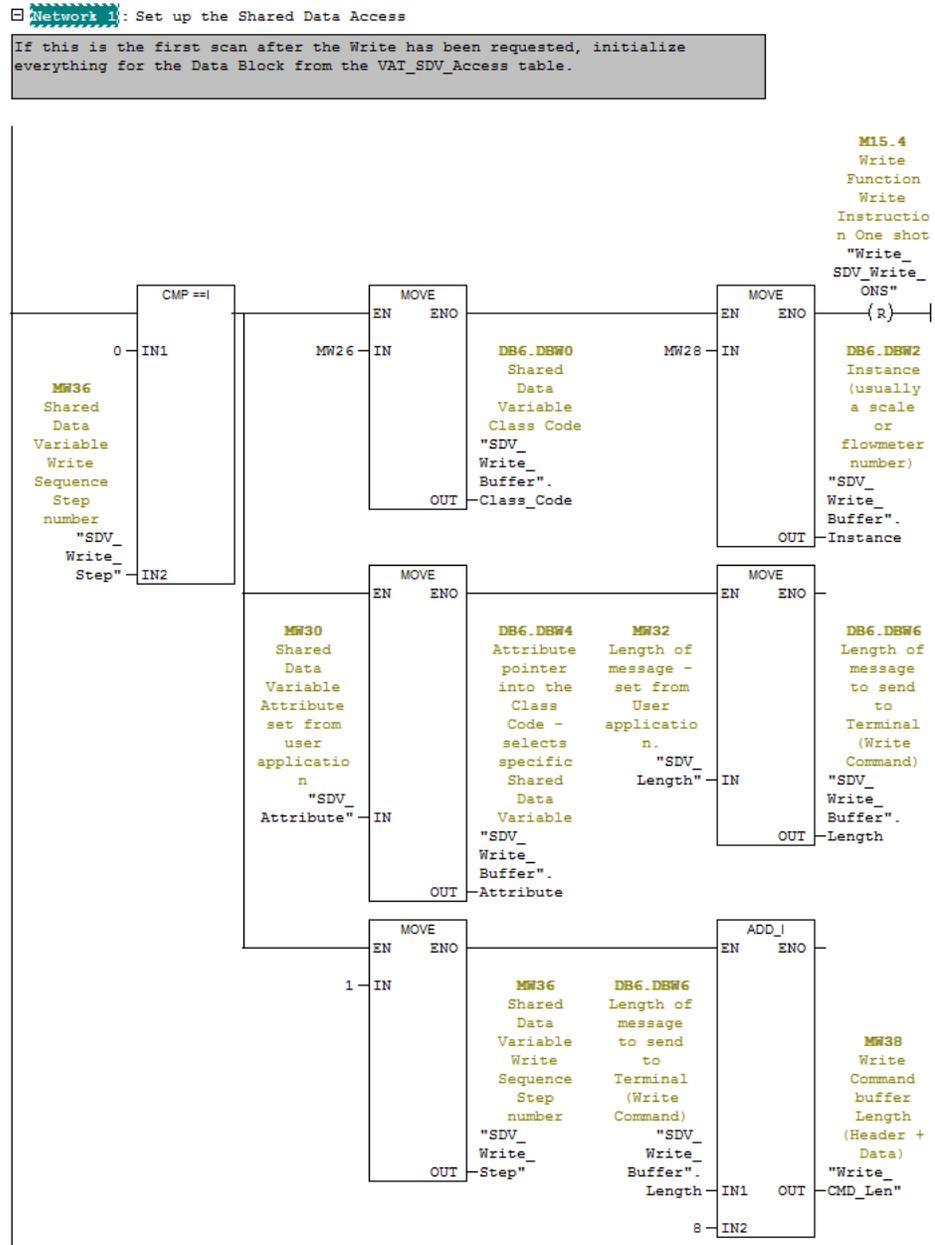


Figure 8-51: Network 1 of FC3

Network 1 Checks to see if this is the first scan since the Write flag was set. If it is, then get the Shared Data Class, Instance, Attribute, and data length from the requestor (in this case, the VAT table). **Note that in this case, the data to be written has already been populated into the Data Buffer of DB6.**

Kick the step variable to the next step in the sequence, and calculate how long the entire Write buffer will be by adding the header length to the data length provided by the caller.

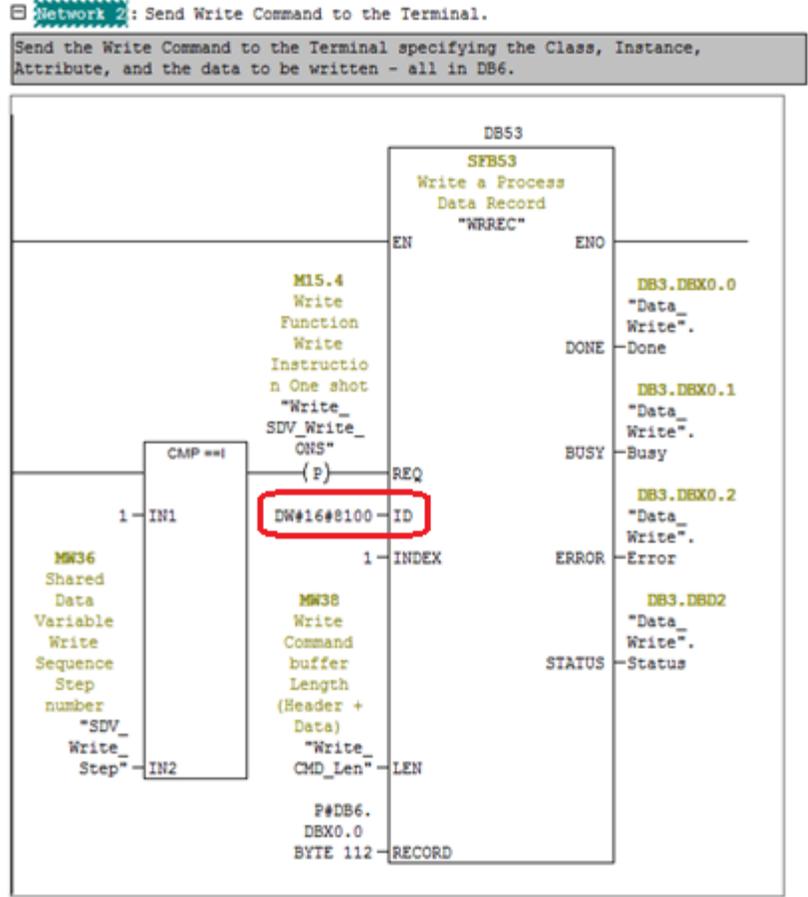
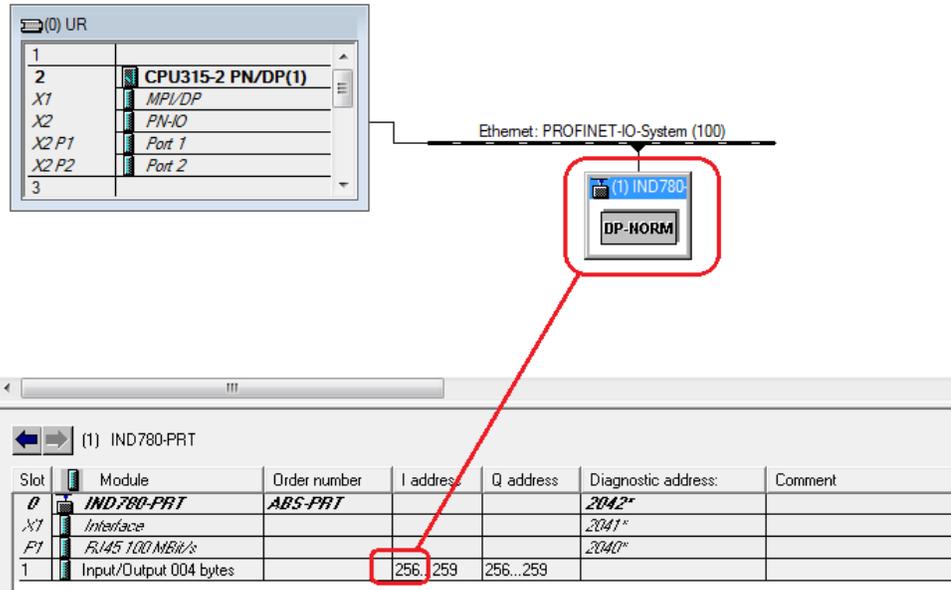


Figure 8-52: Network 2 of FC3

Network 2 calls SFB53 (WRREC) to send the data in DB6 to the IND780 specified by the ID code. Note that DB3 is used to store the Handshaking status bits of SFB53.



**Figure 8-53: Hardware Configuration and the WRREC ID Code specification.**

The ID code uses the I Address (256 Decimal in this case) converted to Hexadecimal, which is 0x0100, and adds 0x8000 to it, to get 0x8100 as the ID code. This is how the WRREC routine knows where to send the message.

□ Network 3: Wait for the Write command to complete.

While the Busy Bit is set, wait for completion. If the Error bit is set then return a -98 in the step to indicate that the Write command failed that the user should check the WRREC returned status for more information. If the Done bit is set then move on to the next step in the Write sequence.

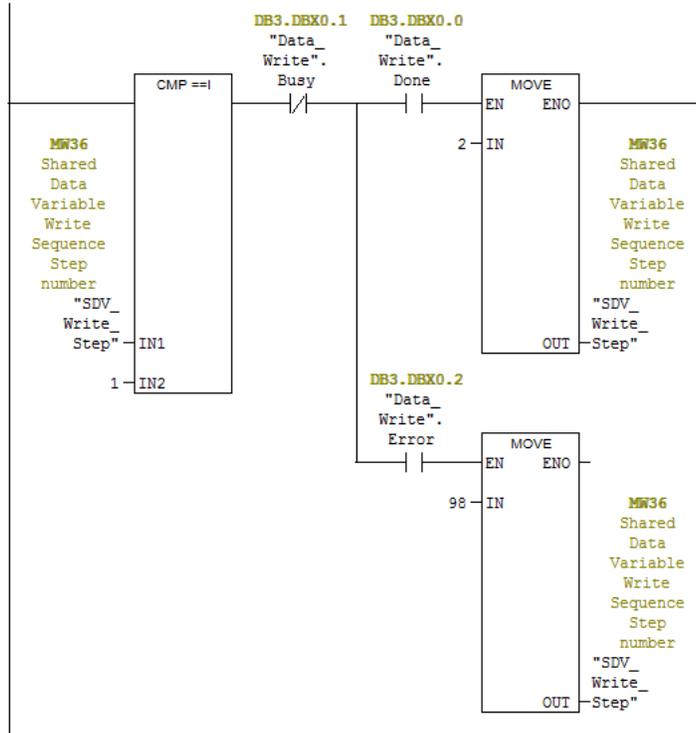
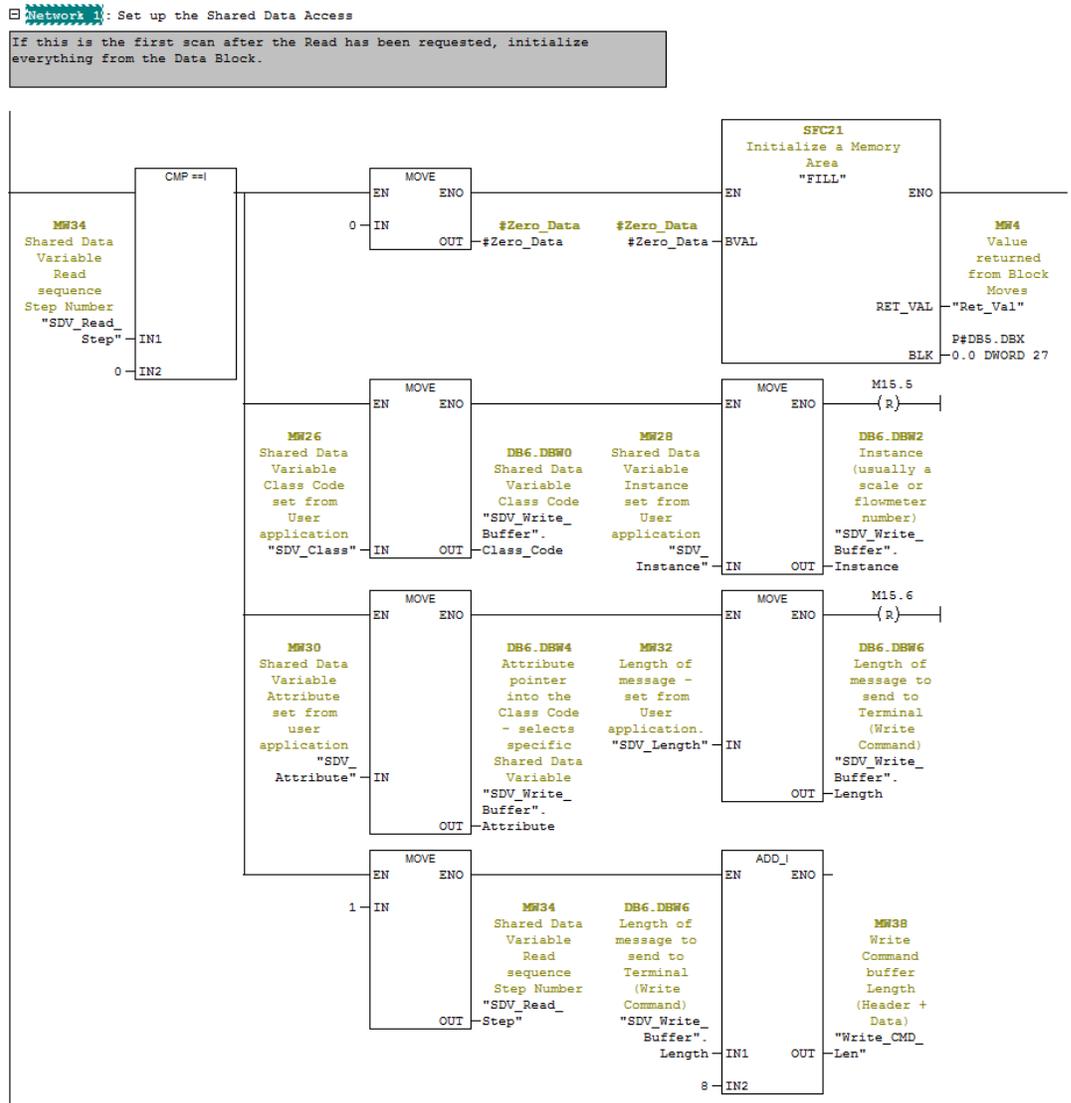


Figure 8-54: Network 3 of FC3

Network 3 confirms that the program is executing step 1 of the Shared Data Write routine, and then waits for the Data\_Write Busy signal to go low. At that point, if the Data\_Write Done flag is set then move on to the next step in the sequence. If instead the Data\_Write Error flag is set, move the value of 98 into the sequence step to flag that an error has occurred.



### 8.10.4.3. FC2 (Read Shared Data) Program details

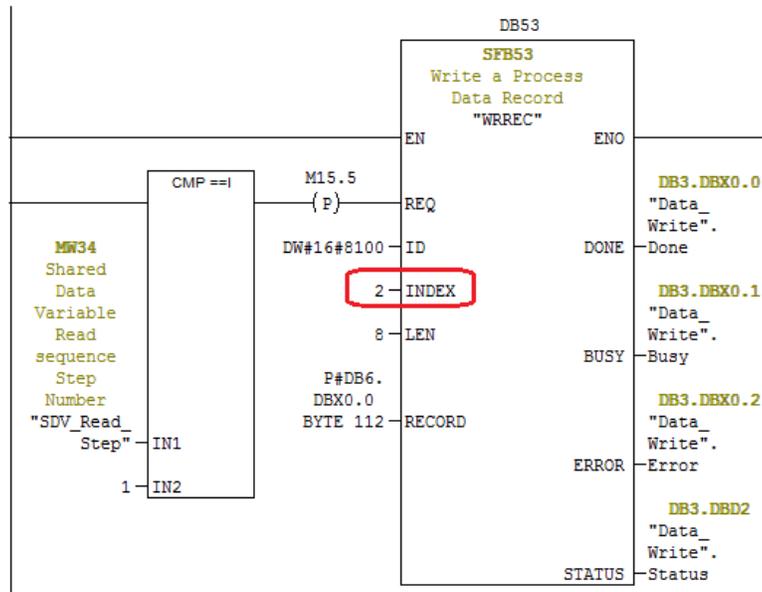


**Figure 8-56: Network 1 of FC2**

On the first scan of FC2 after the Read flag has been set, Network 1 initializes DB5 to all zeroes. Then the Class code, Instance, Attribute and data length are copied into DB5. The command length is calculated by adding the header length (8) to the data length. The step sequence is then incremented to the next step.

□ Network 2 : Send Read Command to the Terminal.

Send the Read Command to the Terminal that we want to read the specified Shared Data Variable on the next Read Data Command.



**Figure 8-57: Network 2 of FC2**

When the sequence step is 1, network 2 write the contents of DB6 to the IND780 specified in the ID code (see Figure 8-53 for how the ID code is generated). Note that the Index is set to 2. This tells the IND780 that the WRREC routine is requesting that the data contained in the Shared Data variable specified by the Class, Instance, and Attribute, be returned to the PLC. The IND780 will buffer up the requested data and wait for the corresponding REREC routine call to be issued.

Note that DB3 is again used for the handshake data.

□ Network 3: Wait for the command for the Read to complete

After the Write to the Terminal (which contains the request to read the specified Shared Data value) has executed, wait for either the Done or Error flags to be set. If the Error flag gets set, cancel the request and return a -98 to indicate that the Read Command failed and that the user should check the returned Write Status for more information.

If the done flag is set, then move on to the next step in the Shared Data Variable Read sequence.

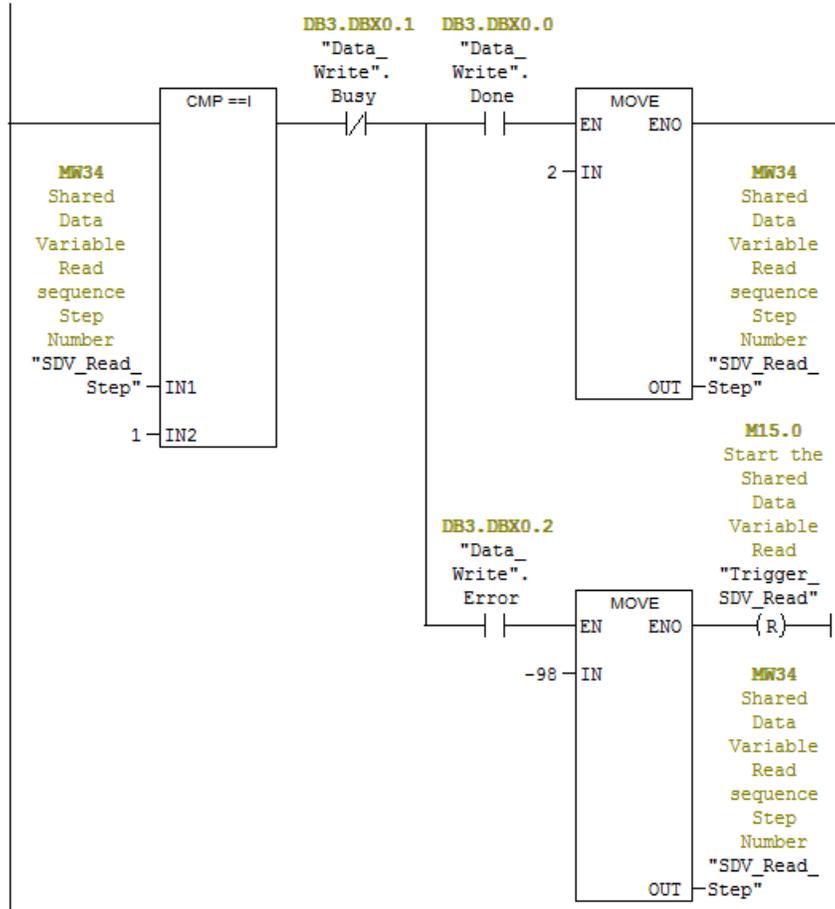


Figure 8-58: Network 3 of FC2

If the sequence step is still 1, then wait for the Data\_Write busy flag to turn off. After that, if the Data\_Write Done flag is set then increment the step counter to move on to the next step. If the Data\_Write Error flag is set, then put a -98 into the step counter and clear the Read Request flag. The -98 will tell the caller that the read failed on the initial contact with the IND780.

□ Network 4 : Read the Result

If the command write above was successful, send a command to read the requested data from the IND Instrument. Return the read data into DB5.

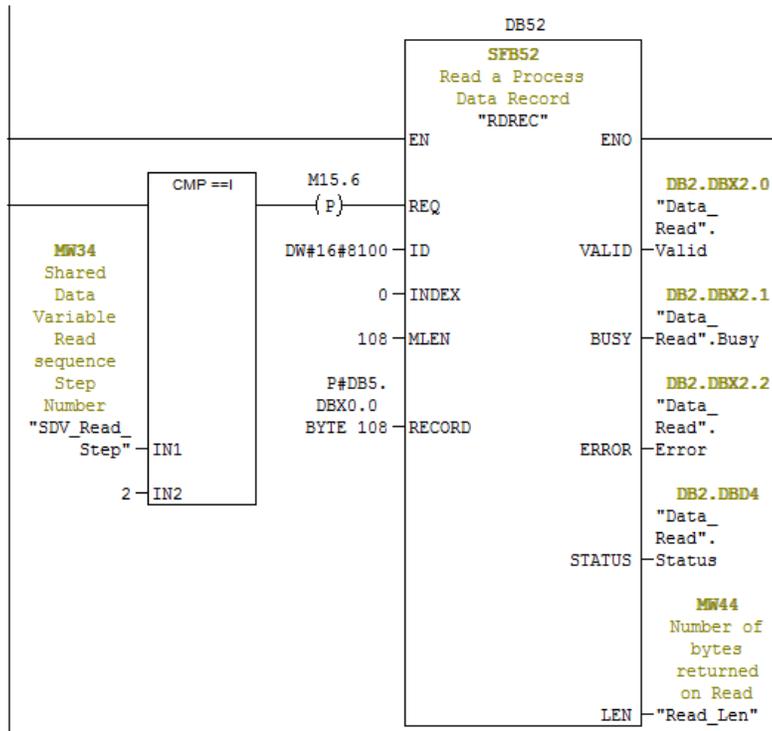


Figure 8-59: Network 4 of FC2

If the sequence step is 2, then trigger the Read of the requested data from the IND780. Note that the ID is the same as in the WRREC, and that DB2 is used for the Handshake data.

Network 5: Wait for the Read to Complete  
 Wait for the Busy Flag to go away. Once it does, check the Valid Read flag and the Error flag. If the Error flag is set, shut down the routine and return a -99 to indicate that the Read failed (the user should check the returned status data from the above read command for more information).

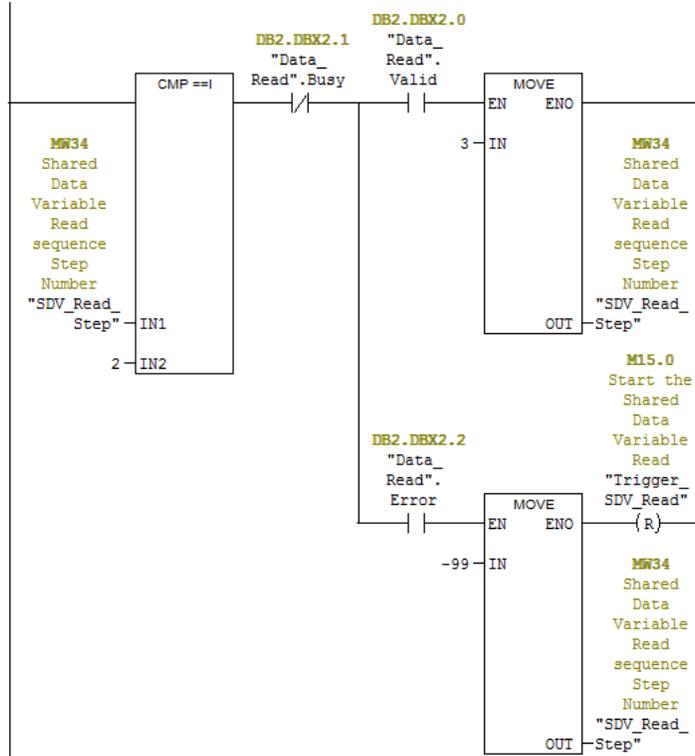


Figure 8-60: Network 5 of FC2

If the sequence step is 2, then wait for the Data\_Read busy flag to turn off. Once it does, if the Data\_Read Valid flag is set then increment the routine’s step counter. If the Data\_Read Error flag is set, the move a -99 into the step counter to flag that an error occurred in the second half of the Read routine, and clear the Read Request flag.

Network 6: Confirm that we got what we wanted.

First, assume that the Shared Data Read returned the WRONG Shared Data variable (because the Siemens compares seem to insist that we do things this way). Then, check to see if the Class, Instance, and Attribute in the returned Read data match what was sent in the Write command. If they all match, then make our step number a positive final value. Otherwise, return a -100 to flag that the READ failed.

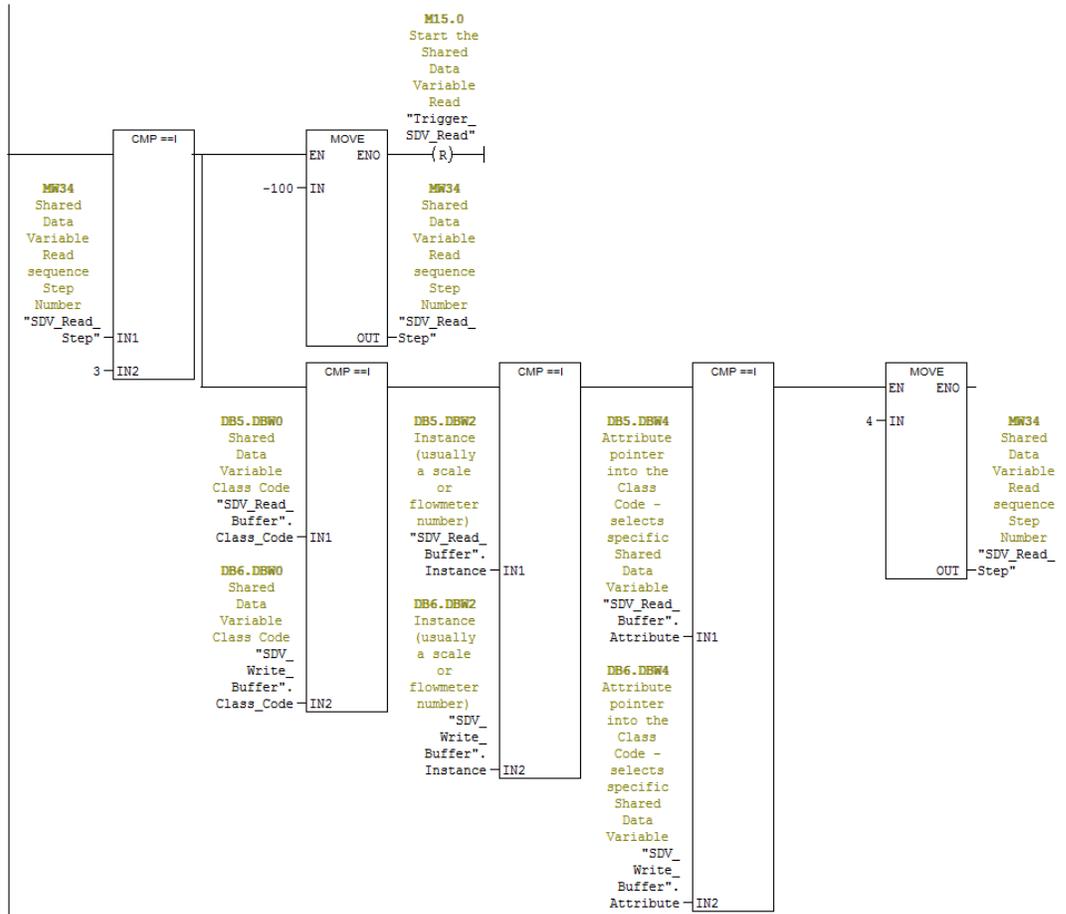
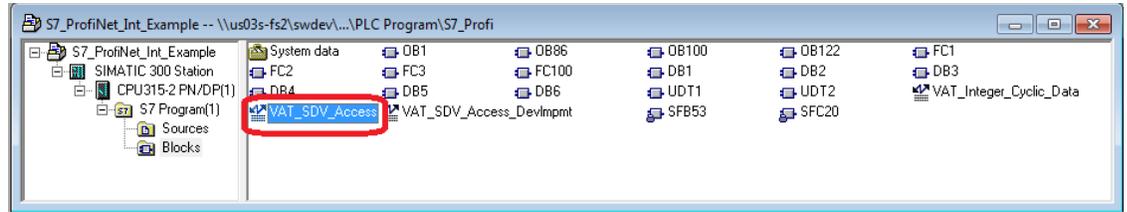


Figure 8-61: Network 6 of FC2

If the sequence step is 3, then process the final result by first assuming that the sequence failed and making sure that the Read Request bit is cleared to indicate that we're done. Next, compare the returned Class Code, Instance, and Attribute numbers with the ones that were originally sent. If they all match, the increment the Step number and allow the routine to exit normally. If they do NOT match, then the routine will automatically exit with the error codes set.

### 8.10.4.4. Running the Sample Program

The sample program can be run from the Variable Access Table as shown in the Figure 8-62 and Figure 8-63.



**Figure 8-62: VAT\_SDV\_Access**

VAT\_SDV\_Access is a Variable Access Table that allows the user to monitor the processed cyclic data and issue commands to the IND780 as shown below:

Address	Symbol	Display format	Status value	Modify value
// Shared Data Access, specify Class, Instance, Attribute and length.				
MW 26	"SDV_Class"	HEX	W#16#006B	W#16#006B
MW 28	"SDV_Instance"	DEC	1	1
MW 30	"SDV_Attribute"	DEC	1	1
MW 32	"SDV_Length"	DEC	4	4
// Shared Data Read Trigger and Status				
M 15.0	"Trigger_SDV_Read"	BOOL	false	
DB2.DBX 2.0	"Data_Read".Valid	BOOL	true	
DB2.DBX 2.2	"Data_Read".Error	BOOL	false	
// First 20 bytes of the Read Buffer (from IND780) appears here. More is available if needed.				
DB5.DBX 8	"SDV_Read_Buffer".Data_Buffer[0]	CHARACTER	'1234'	
DB5.DBX 12	"SDV_Read_Buffer".Data_Buffer[1]	CHARACTER	DW#16#00000000	
DB5.DBX 16	"SDV_Read_Buffer".Data_Buffer[2]	CHARACTER	DW#16#00000000	
DB5.DBX 20	"SDV_Read_Buffer".Data_Buffer[3]	CHARACTER	DW#16#00000000	
DB5.DBX 24	"SDV_Read_Buffer".Data_Buffer[4]	CHARACTER	DW#16#00000000	
// Shared Data Write Trigger and Status				
M 15.1	"Trigger_SDV_Write"	BOOL	false	
DB3.DBX 0.0	"Data_Write".Done	BOOL	false	
DB3.DBX 0.2	"Data_Write".Error	BOOL	false	
// first 20 bytes of the Write Buffer (to IND780) appears here. More is available if needed.				
DB6.DBX 8	"SDV_Write_Buffer".Data_Buffer[0]	CHARACTER	'5678'	'5678'
DB6.DBX 12	"SDV_Write_Buffer".Data_Buffer[1]	CHARACTER	DW#16#00000000	DW#16#00000000
DB6.DBX 16	"SDV_Write_Buffer".Data_Buffer[2]	CHARACTER	DW#16#00000000	DW#16#00000000
DB6.DBX 20	"SDV_Write_Buffer".Data_Buffer[3]	CHARACTER	DW#16#00000000	
DB6.DBX 24	"SDV_Write_Buffer".Data_Buffer[4]	CHARACTER	DW#16#00000000	

**Callouts:**

- Shared Data Class, Instance, Attribute, and Length, for the requested operation (read or write). See Shared Data Reference Manual (P/N 6405910) for this information.
- Set this bit to trigger a Read of the Shared Data Variable specified above.
- Indicates the results of the requested Read.
- The first 20 bytes (4 bytes for each line) of the Read buffer. The read results will show up here. Lines can be added to show more of the buffer if desired.
- Set this bit to trigger a Write to the Shared Data Variable specified a the top of this form.
- Indicates the results of the requested Write.
- The first 20 bytes (4 bytes for each line) of the Write buffer. This is the Data that will be written TO the specified Shared Data Variable. Lines can be added to show more of the buffer if desired.

**Figure 8-63: VAT\_SDV\_Access - Description**

# 9 Indirect Shared Data Access

## 9.1. Overview

Indirect Data Access allows a PLC to access any Shared Data Variable by name, by routing the request through a small set of Indexes (PROFINET) or through a single Class on the CIP (EtherNet/IP and ControlNet). The data payload will contain the Shared Data Variable to be accessed, together with data to be written to that variable.

## 9.2. Indirect Access Data Structure

The following Data Structure applies **both** to writing data to the terminal **and** to reading data from the terminal.

### 9.2.1. Data Writes

In Data Writes, the header information must be supplied by the Controller (PLC) to instruct the terminal where to write the data. The trailing data is then applied to the specified variable.

### 9.2.2. Data Reads

In Data Reads, only the header information is sent to the terminal, and a follow-up read instruction performs the actual read. The terminal populates the returned structure with the variable's header information so that the Controller can verify that it is receiving the correct data. The value of the variable will be contained in the Data section after the header, as shown in Figure 9-1.

Word 0	Variable Group		Header
Word 1	Variable Subgroup		
Word 2	Variable Item		
Word 3	Data Length		
Word 4	Byte 2	Byte 1	Data
Word 5	Byte 4	Byte 3	
Word 6	Byte 6	Byte 5	
Word 7	Byte 8	Byte 7	
...			
Word n	Byte (2n+2)	Byte (2n+1)	

Figure 9-1: Indirect Access Data Structure

The components of this structure are detailed in the following sections.

### 9.2.3. Variable Group

- The Variable Group is similar to the 'Class' in a CIP variable specification.

For both Read and Writes, this value contains the first two characters of the Shared Data name. For example, when reading wt0101, this value would be the characters **wt**, or **0x5754** (Hex), or **22356** (decimal).

#### 9.2.4. Variable Subgroup

- The Variable Subgroup is the same as the 'Instance' in a CIP variable specification.

This field contains an actual number representing the instance of the variable. In the case of wt0110, the instance is shown by the middle two characters '01' of the name, Therefore, the number contained in this field would be a 0x0001 (Hex) or 1 (decimal).

#### 9.2.5. Variable Item

- The Variable Item is the same as the 'Attribute' in a CIP variable specification.

This field contains an actual number representing the attribute of the variable. In the case of wt0110, the attribute is shown by the last two characters '10' of the name, Therefore, the number contained in this field would be a 0x000A (Hex) or 10 (decimal).

#### 9.2.6. Length

For a Read, the length is the number of bytes returned by the IND780 in the data section.

For a Write, the length is the number of bytes sent from the PLC in the data section.

#### 9.2.7. Byte Array

For a Read the byte array contains the data returned from the terminal that represents the value in the Shared Data variable requested.

For a Write the byte array contains the data to be written to the Shared Data Variable specified by the header values.

#### 9.2.8. PROFINET Acyclic Messages

For most control systems using PROFINET networks, there are two types of commands that can be used to execute acyclic messages: RDREC (SFB52) and WRREC (SFB53). The index and length parameters within these blocks are used to specify what command is required. To read a variable, a combination of RDREC and WRREC commands are used. To write a value to a variable, the WRREC command is used. Index numbers used are interface-dependent due to the restrictions of the particular network type.

#### 9.2.9. EtherNet/IP and ControlNet Acyclic Messages

For control systems using EtherNet/IP or ControlNet networks, there are two types of commands that can be used to execute asynchronous messages: CIP Generic Message Instruction for Get Attribute Single (service code 0x0e) or CIP Generic Message Instruction for Set Attribute Single (service code 0x10). These commands use parameters called class code, instance number, attribute number and length to be configured in the message block in order to specify what command is required.

#### 9.2.10. Acyclic Writes

A single acyclic message will contain both the Shared Data Variable indication (in the header) as well as the data that needs to be written (in the Data section). No other acyclic messages are required.

### 9.2.11. Acyclic Reads

A read requires the use of a two-step process.

1. An Acyclic message to request the Shared Data Variable to be read. This message sends the request (using a Write or Set message instruction) from the PLC, and only contains the header information required to specify the Shared Data variable to be read.
2. An Acyclic message to read the data. The Controller (PLC) issues a Get or Read Acyclic message to the terminal to return the requested data. The terminal will return the full structure containing the header, followed by the Data being returned. The Controller will use the Header information to verify that the data returned matches the variable that was requested.

The terminal should continue to return the value for the same Shared Data Variable requested in number 1 above for any following Get or Read Acyclic messages until a new Read Request is received. This feature will enable higher frequency data to be returned to the controller.

## 9.3. Command Codes

The following is a table of the Command Codes that will be used by the PLC to trigger the indicated functions.

**Table 9-1: Command Codes**

Command	Description	PROFINET slot, index (hex)	CIP class, instance, attribute (hex)	Data Type
Read Variable	Read a variable specified by prior Request Read	1, 0 and 1,3*	310, 01, 01	Varies by type of variable
Write Variable	Write value into variable specified	1,1	310, 01, 02	
(Write) Request Read Variable	Specify variable that will be used in the Read command	1,2	310, 01, 03	

\* IND780 supports **both** slot and Index 1, 0 and 1, 3.

## 9.4. EtherNet/IP Example

This example first shows how to write a two byte integer value to ai0102 (an Application Dynamic Integer Field in the AI block), and then how to perform a read.

- Note that this example is applicable to the ControlNet implementation as well.
- Any required Byte Order manipulations are not shown here.

### 9.4.1. Write

A Write is the simplest example because it is a one-step operation. A single message instruction specifies everything.

In this example, the integer value **10** will be written to **ai0102**.

First, populate the data structure to be sent to the terminal. The structure is shown in Figure 9-2.

Write\_Buffer

Word 0	"AI"	← Variable Group (Class)
Word 1	1	← Variable Subgroup (Instance)
Word 2	2	← Variable Item (Attribute)
Word 3	2	← Data Length
Word 4	10	← Value to Write (two byte Integer)

**Figure 9-2: Data Structure Sent to Terminal**

Next, the CIP Generic message instruction is set up to use the "Set Attribute Single" service to write the data to ai0101.

Command	Description	PROFINET slot, index (hex)	CIP class, instance, attribute (hex)	Data Type
Read Variable	Read a variable specified by prior Request Read	1, 0 and 1,3*	310, 01, 01	Varies by type of variable
Write Variable	Write value into variable specified	1,1	310, 01, 02	
(Write) Request Read Variable	Specify variable that will be used in the Read command	1,2	310, 01, 03	

\* IND780 supports both slot and Index 1, 0 and 1, 3.

**Figure 9-3: CIP Generic Message Instruction**

### 9.4.2. Read

A Read requires that the PLC first write a request to the terminal for the variable that it wants to read so that the terminal can queue up the data.

First, populate the Data Structure to be sent to the terminal. In this case, only the header information is needed, to tell the terminal which variable to read.

Word 0	"AI"	← Variable Group (Class)
Word 1	1	← Variable Subgroup (Instance)
Word 2	2	← Variable Item (Attribute)
Word 3	2	← Data Length

**Figure 9-4: Read Variable Instruction to Terminal**

Next, the CIP Generic message instruction is set up to use the "Set Attribute Single" service to write the Read Request information to the terminal:

Command	Description	PROFINET slot, index (hex)	CIP class, instance, attribute (hex)	Data Type
Read Variable	Read a variable specified by prior Request Read	1, 0 and 1,3*	310, 01, 01	Varies by type of variable
Write Variable	Write value into variable specified	1,1	310, 01, 02	
(Write) Request Read Variable	Specify variable that will be used in the Read command	1,2	310, 01, 03	

\* IND780 supports both slot and Index 1, 0 and 1, 3.

Figure 9-5: Writing the Read Request to Terminal

The PLC will then issue a Read instruction to get the data that the terminal has queued up.

Command	Description	PROFINET slot, index (hex)	CIP class, instance, attribute (hex)	Data Type
Read Variable	Read a variable specified by prior Request Read	1, 0 and 1,3*	310, 01, 01	Varies by type of variable
Write Variable	Write value into variable specified	1,1	310, 01, 02	
(Write) Request Read Variable	Specify variable that will be used in the Read command	1,2	310, 01, 03	

\* IND780 supports both slot and Index 1, 0 and 1, 3.

Figure 9-6: Read Instruction to Acquire Data from Terminal

Before accepting the data for use in the program, the PLC should check that the Variable Group, Subgroup, and Item Number returned by the terminal match the ones sent in the request.

- Note: The PLC can continue to issue Read instructions to the terminal without first writing a request as long as it wants to continue to receive data from the previously requested Shared Data variable.

## 9.5. PROFINET Example

This example first shows how to write a two byte integer value to ai0102 (an Application Dynamic Integer Field in the AI block), and then how to perform a read.

- Note: Any required Byte Order manipulations are not shown here.

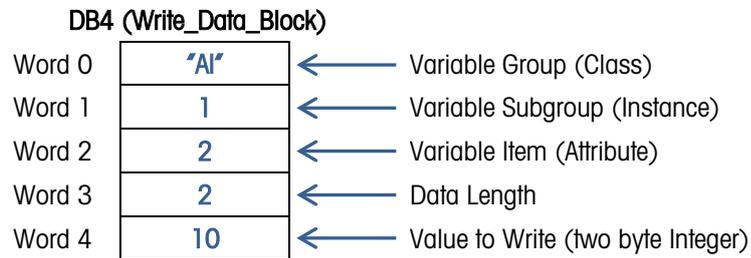
The PROFINET example works almost exactly the same way as the EtherNet/IP example, but – since it uses different instructions for the messages – the messages will be configured differently.

### 9.5.1. Write

A Write is the simplest example because it is a one-step operation. A single message instruction specifies everything.

In this example, the integer value 10 will be written to ai0102.

First, populate the data structure that will be sent to the terminal. The structure is shown in Figure 9-7.



**Figure 9-7: Data Structure Sent to Terminal**

Next, the WRREC message instruction is set up to write the data to ai0101.

Command	Description	PROFINET slot, index (hex)	CIP class, instance, attribute (hex)	Data Type
Read Variable	Read a variable specified by prior Request Read	1, 0 and 1, 3*	310, 01, 01	Varies by type of variable
Write Variable	Write value into variable specified	1, 1	310, 01, 02	
(Write) Request Read Variable	Specify variable that will be used in the Read command	1, 2	310, 01, 03	

\* IND780 supports both slot and Index 1, 0 and 1, 3.

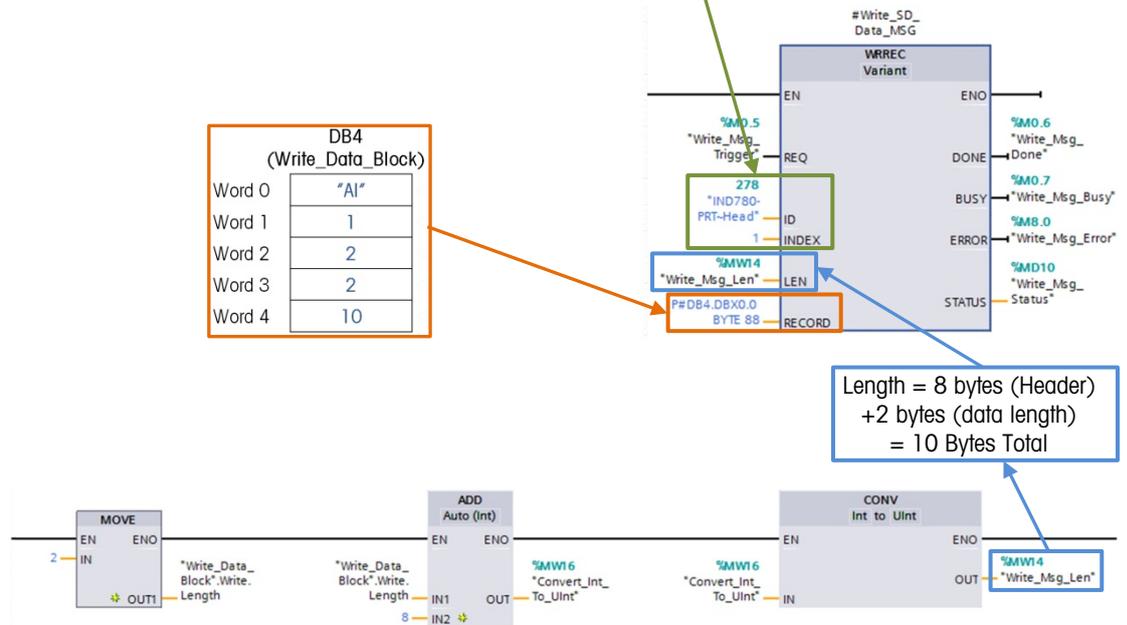


Figure 9-8: Writing Data to ai0101

### 9.5.2. Read

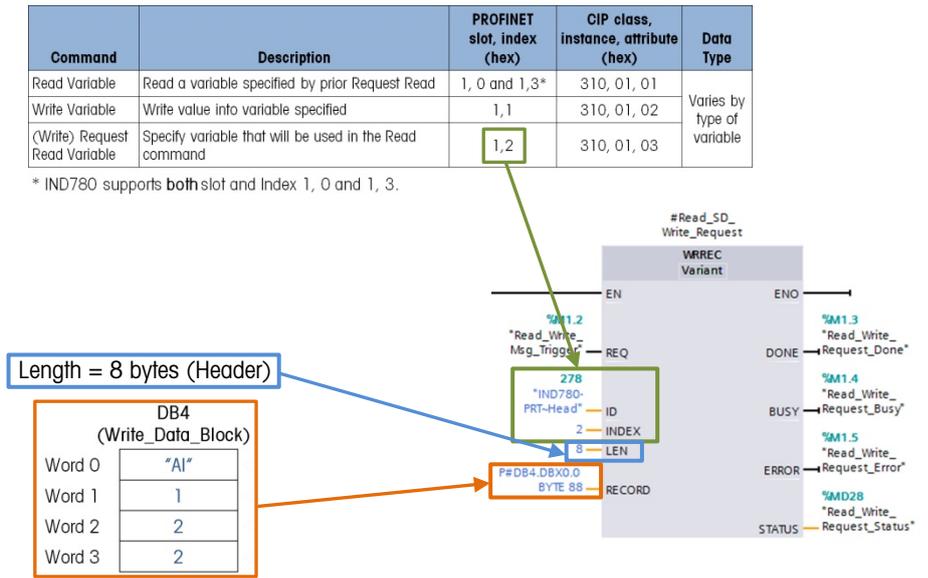
A Read requires that the PLC first write a request to the terminal for the variable that it wants to read so that the terminal can queue up the data.

First, populate the Data Structure to be sent to the terminal. In this case, only the header information is needed to tell the terminal which variable to read.

DB4 (Write_Data_Block)		
Word 0	"AI"	← Variable Group (Class)
Word 1	1	← Variable Subgroup (Instance)
Word 2	2	← Variable Item (Attribute)
Word 3	2	← Data Length

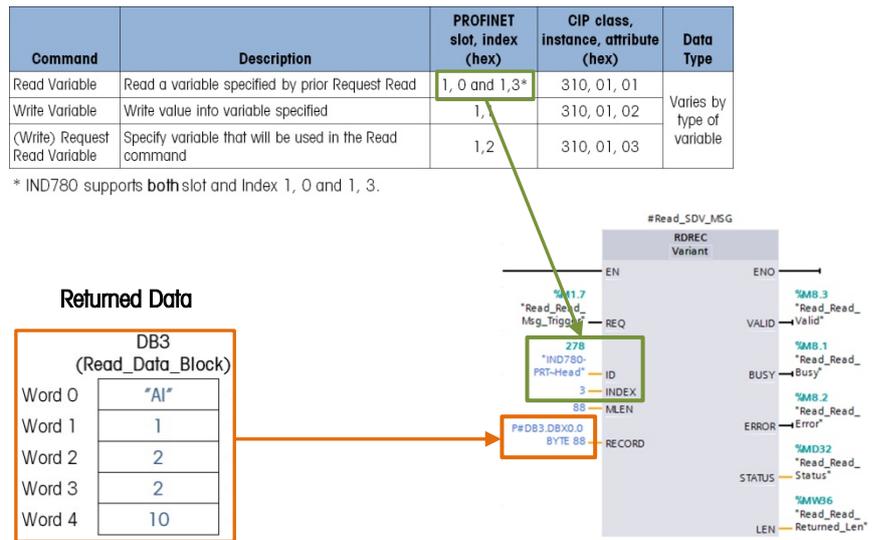
Figure 9-9: Data Structure Instruction Sent to Terminal

Next, set up the WRREC message instruction to write the Read Request information to the terminal.



**Figure 9-10: Sending Read Request Information to Terminal**

The PLC will then issue a RDREC instruction to get the data that the terminal has queued up.



**Figure 9-11: Instructing the Terminal to Send Data**

Before accepting the data for use in the program, the PLC should check that the Variable Group, Subgroup, and Item Number returned by the Terminal match the ones sent in the request.

- The PLC can continue to issue Read instructions to the Terminal without first writing a request as long as it wants to continue to receive data from the previously requested Shared Data variable.

## METTLER TOLEDO Service

### To protect your METTLER TOLEDO 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 METTLER TOLEDO service agreement tailored to your needs and budget.

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

[www.mt.com/IND780](http://www.mt.com/IND780)

For more information

**Mettler-Toledo, LLC**

1900 Polaris Parkway  
Columbus, OH 43240  
Phone 800 438 4511  
Fax 614 438 4900

© 2019 Mettler-Toledo, LLC  
64057518 Rev. 11, 06/2021



64057518