PLC Manual

IND500x

Weighing Terminal

METTLER TOLEDO

IND500x Weighing Terminal

METTLER TOLEDO Service

Essential Services for Dependable Performance of Your IND500x Weighing Terminal

Congratulations on choosing the quality and precision of METTLER TOLEDO. Proper use of your new equipment according to this Manual and regular calibration and maintenance by our factory-trained service team ensures dependable and accurate operation, protecting your investment. Contact us about a service agreement tailored to your needs and budget. Further information is available at www.mt.com/service.

There are several important ways to ensure you maximize the performance of your investment:

1. Register your product: We invite you to register your product at www.mt.com/productregistration so we can contact you about enhancements, updates and important notifications concerning your product.
2. Contact METTLER TOLEDO for service: The value of a measurement is proportional to its accuracy – an out of specification scale can diminish quality, reduce profits and increase liability. Timely service from METTLER TOLEDO will ensure accuracy and optimize uptime and equipment life.
    a. Installation, Configuration, Integration and Training: Our service representatives are factory-trained, weighing equipment experts. We make certain that your weighing equipment is ready for production in a cost effective and timely fashion and that personnel are trained for success.
    b. Initial Calibration Documentation: The installation environment and application requirements are unique for every industrial scale so performance must be tested and certified. Our calibration services and certificates document accuracy to ensure production quality and provide a quality system record of performance.
    c. Periodic Calibration Maintenance: A Calibration Service Agreement provides on-going confidence in your weighing process and documentation of compliance with requirements. We offer a variety of service plans that are scheduled to meet your needs and designed to fit your budget.

METTLER TOLEDO reserves the right to make refinements or changes to the product or manual without notice.

COPYRIGHT

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.

⚠ WARNING

FOR CONTINUED PROTECTION AGAINST SHOCK HAZARD CONNECT TO PROPERLY GROUNDED OUTLET ONLY. DO NOT REMOVE THE GROUND PRONG.

⚠ WARNING

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.

⚠ CAUTION

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.

NOTICE

OBSERVE PRECAUTIONS FOR HANDLING ELECTROSTATIC SENSITIVE DEVICES.

Disposal of Electrical and Electronic Equipment

In conformance with the European Directive 2012/19/EC on Waste Electrical and Electronic Equipment (WEEE) this device may not be disposed of in domestic waste. This also applies to countries outside the EU, per their specific requirements.

Please dispose of this product in accordance with local regulations at the collecting point specified for electrical and electronic equipment.

If you have any questions, please contact the responsible authority or the distributor from which you purchased this device.

Should this device be passed on to other parties (for private or professional use), the content of this regulation must also be related.

Thank you for your contribution to environmental protection.

Contents

1 Analog Output Option 1-1

1.1. Specifications 1-1

1.2. Analog Output Operation 1-3

1.3. Setup in the IND500x Terminal 1-6

1.4. Analog Output Kit Spare Parts 1-8

2 EtherNet/IP™ Interface 2-1

2.1. Overview 2-1

2.2. Data Definition 2-4

2.3. Software Setup 2-9

2.4. Troubleshooting 2-10

2.5. EtherNet/IP Option Kit 2-11

2.6. Programming Examples 2-12

3 Modbus TCP Interface 3-1

3.1. Preface 3-1

3.2. Overview 3-1

3.3. Modbus TCP Characteristics 3-2

3.4. Modbus TCP Board 3-2

3.5. Data Definition 3-3

3.6. Software Setup 3-10

3.7. Troubleshooting 3-11

3.8. Modbus TCP Option Kit 3-12

3.9. Modbus TCP Configuration Example 3-13

4 PROFIBUS Option 4-1

4.1. Overview 4-1

4.2. Data Integrity 4-4

4.3. Discrete Data 4-4

4.4. Floating Point 4-6

4.5. Shared Data 4-7

4.6. Discrete Data I/O Space Usage Comparison 4-9

4.7. IND500x PROFIBUS Message Mapping 4-11

4.8. Controlling Discrete I/O Using a PLC Interface 4-16

4.9. Hardware Setup 4-16

4.10. Software Setup 4-17

4.11. PROFIBUS Option Kit Part Numbers 4-18
4.12. Interfacing Examples 4-19

5 PROFINET Option 5-1

5.1. Overview.. 5-1
5.2. PROFINET Interface 5-1
5.3. S2 System Redundancy 5-6
5.4. Software Setup... 5-6
5.5. Troubleshooting 5-8

A Integer and Division Data Formats A-1

B Floating Point Format... B-1

B.1. Operational Overview.. B-1
B.2. Floating Point Command Examples.. B-10

C. PLC Data Sizes.. C-1

C.1. Introduction C-1
C.2. Integer and Divisions Data Formats C-1
C.3. Floating Point Data Format.. C-2

D ASCII Characters D-1

D.1. Standard Characters D-1
D.2. Control Characters D-2

1 Analog Output Option

The IND500x weighing terminal can use the analog output option in either of two ways:

- Indirectly, using the ACM500, when the ACM500 has the analog option board installed in it.
- Directly, by installing an analog option board in the IND500x terminal.

■ Note that only one of these two installations can be used; parallel installations are not supported.

Either solution can be used for displayed weight, ABS-displayed weight (absolute displayed weight), gross weight, rate or ABS-rate. The analog output uses a 16-bit D/A converter for a very precise output.

The Analog Output sub-block permits the selection of a data source for the analog signal and provides a method to calibrate the analog zero and high limit values. The IND500x terminal must be calibrated to the desired scale capacity **before** Analog Output adjustments are performed. If rate is to be used as the source for the analog output signal, it must be enabled in the **Scale > Rate** branch of setup.

The Analog Output card from ACM500 provides one channel which can be either current (4-20 mA) or voltage (0-10 VDC). When the option is installed **in the IND500x, only 4-20 mA output is available**.

1.1. Specifications

1.1.1. Analog Output Option Board from ACM500

Maximum Cable Length	0-10 VDC – 50 ft (15.2 m)
	4-20mA – 1000 ft (300 m)
Min/Max Load Resistance	0-10 VDC – 100k ohms minimum
	4-20 mA – 500 ohms maximum
Outputs	1 channel capable of supplying 4-20 mA or 0-10 VDC
Resolution	16-bit resolution - 65536 levels across entire range

■ Note that if the load resistance ratings are exceeded, the analog output will not operate properly.

■ Figure 1-1 shows an Analog Output Option Board with its connector at bottom center.

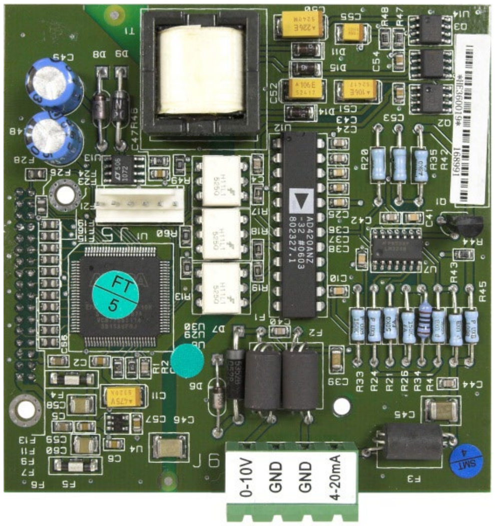


Figure 1-1: Analog Output Option Board for ACM500

1.1.2. Analog Output Option Board from IND500x

Maximum Cable Length	4-20mA – 1000 ft (300 m)
Min/Max Load Resistance	4-20 mA – 500 ohms maximum
Outputs	1 channel capable of supplying 4-20 mA
Resolution	16 bit resolution - 65536 levels across entire range

- Note that if the load resistance ratings are exceeded, the analog output will not operate properly.

Figure 1-2Figure 1-1 shows an Analog Output Option Board for use directly in an IND500x weighing terminal.

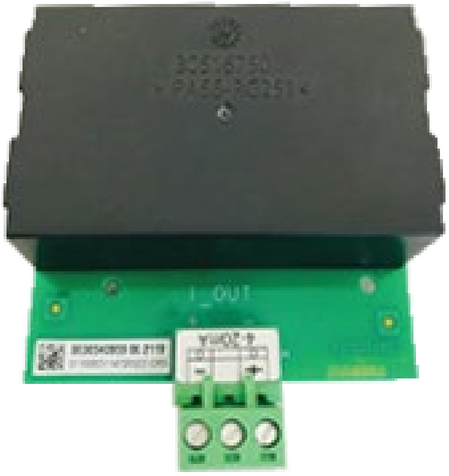


Figure 1-2: Analog Output Option Board for IND500x

1.2. Analog Output Operation

When the source of the analog output is displayed weight, gross weight or rate, the output signals will be at the lower limit (0 VDC or 4 mA) when the value represented is at zero. When the value reaches its maximum limit, the output signal will increase to the high limit (10 VDC or 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 ABS (absolute)-displayed weight and ABS-rate are intended for use when material is being transferred off of a scale in Net mode. In these cases, the displayed weight and rate will show negative values, but the analog output signal will treat them as absolute values (disregarding their negative status). The output signals will increase as the interpreted absolute weight value or absolute rate value increases.

How the analog output functions under zero and over the high limit is determined by the selection for the source field selected – Displayed Weight, ABS – Displayed Weight, Gross Weight, Rate or ABS – Rate and the type of analog signal (4-20 mA or 10 VDC). Table 1-1 details how the analog output reacts under these conditions.

- Note that the analog output option board from IND500x does not provide 0-10 VDC analog signal output.

To use Rate as the source, it must be enabled in setup at **Scale > Rate**. For information on Rate configuration, refer to Chapter 4 of the **IND500x Technical Manual**.

Table 1-1: Analog Output Functions by Source

Source = Displayed Weight, Mode = 4-20 mA	
Under Zero	When the displayed weight (gross or net) drops below zero, the analog signal continues to decrease. When the under zero display blanking point is reached, or the analog signal negative range is exceeded, the analog output immediately switches to approximately 0 mA and remains there until the display is no longer blanked or the analog signal returns to within range.
Over High Limit	When the displayed weight (gross or net) exceeds the high limit, the analog signal continues to increase. When the display blanking point is reached, or the analog signal positive range is exceeded, 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.
Source = ABS – Displayed Weight, Mode = 4-20 mA	
Under Zero	As the ABS-displayed weight increases in value, the analog output signal will climb towards the maximum output (20mA). When the true under zero blanking point is reached, the analog output immediately switches to approximately 0 mA and remains there until the display is no longer blanked.
Over High Limit	Only possible when weighing in gross mode with ABS-displayed weight as the source. Works the same as the standard "Displayed Weight" setting in this case.

Source = Gross Weight, Mode = 4-20 mA	
Under Zero	When the gross weight drops below zero, the analog signal continues to decrease. When the under zero display blanking point is reached, or the analog signal negative range is exceeded, the analog output immediately switches to approximately 0 mA and remains there until the display is no longer blanked or the analog signal returns to within range.
Over High Limit	When the gross weight exceeds the high limit, the analog signal continues to increase. When the display blanking point is reached, or the analog signal positive range is exceeded, 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.
Source = Rate, Mode = 4-20 mA	
Under Zero	When the rate drops below zero, the analog output quickly switches to approximately 0 mA and remains there until the rate returns to within the programmed range. The jump to 0 mA will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal’s setup menu.
Over High Limit	When the rate exceeds the high limit, the analog output quickly switches to approximately 24 mA and remains there until the rate returns to within the programmed range. The jump to 24 mA will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal’s setup menu.
Source = ABS - Rate, Mode = 4-20 mA	
Under Zero	Not applicable. ABS-Rate recognizes negative rate values as valid.
Over High Limit	When the rate exceeds the analog output high limit, the analog output quickly switches to approximately 24 mA and remains there until the rate returns to within the programmed range. The jump to 24mA will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal’s setup menu.
Source = Displayed Weight, Mode = 0-10 VDC *	
Under Zero	When the displayed weight (gross or net) drops below zero, the analog signal continues to decrease. When the under zero display blanking point is reached, or the analog signal negative range is exceeded, the analog output immediately switches to approximately -2.4 VDC and remains there until the display is no longer blanked or the analog signal returns to within range.
Over High Limit	When the displayed weight (gross or net) exceeds the high limit, the analog signal continues to increase. When the display blanking point is reached, or the analog signal positive range is exceeded, the analog output immediately switches to approximately 12.5 VDC and remains there until the weight display is no longer blanked or the analog signal returns to within range.

Source = ABS – Displayed Weight, Mode = 0-10 VDC *	
Under Zero	As the ABS-displayed weight increases in value, the analog output signal will climb towards the maximum output (10VDC). When the true under zero blanking point is reached, the analog output immediately switches to approximately -2.4V and remains there until the display is no longer blanked.
Over High Limit	Only possible when weighing in gross mode with ABS-displayed weight as the source. Works the same as the standard "Displayed Weight" setting in this case.
Source = Gross Weight, Mode = 0-10 VDC *	
Under Zero	When the gross weight drops below zero, the analog signal continues to decrease. When the under zero display blanking point is reached, or the analog signal negative range is exceeded, the analog output immediately switches to approximately -2.4 VDC and remains there until the display is no longer blanked or the analog signal returns to within range.
Over High Limit	When the gross weight exceeds the high limit, the analog signal continues to increase. When the display blanking point is reached, or the analog signal positive range is exceeded, the analog output immediately switches to approximately 12.5 VDC and remains there until the weight display is no longer blanked or the analog signal returns to within range.
Source = Rate, Mode = 0-10 VDC *	
Under Zero	When the rate drops below zero, the analog output quickly switches to approximately -2.4 VDC and remains there until the rate returns to within the programmed range. The jump to -2.4 VDC will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal's setup menu.
Over High Limit	When the rate exceeds the high limit, the analog output quickly switches to approximately 12.5 VDC and remains there until the rate returns to within the programmed range. The jump to 12.5 VDC will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal's setup menu.
Source = ABS - Rate, Mode = 0-10 VDC *	
Under Zero	Not applicable. ABS-Rate recognizes negative rate values as valid.
Over High Limit	When the rate exceeds the analog output high limit, the analog output quickly switches to approximately 12.5 VDC and remains there until the rate returns to within the programmed range. The jump to 12.5 VDC will occur only as quickly as the rate value is updated in the terminal. Rate update is based on Measurement Period and Output Average, selected in the terminal's setup menu.

* Not applicable to the Analog Output option kit provided for the IND500x.

1.3. Setup in the IND500x Terminal

Figure 1-3 illustrates the setup procedures for configuring the Analog Output option for the IND500x terminal.

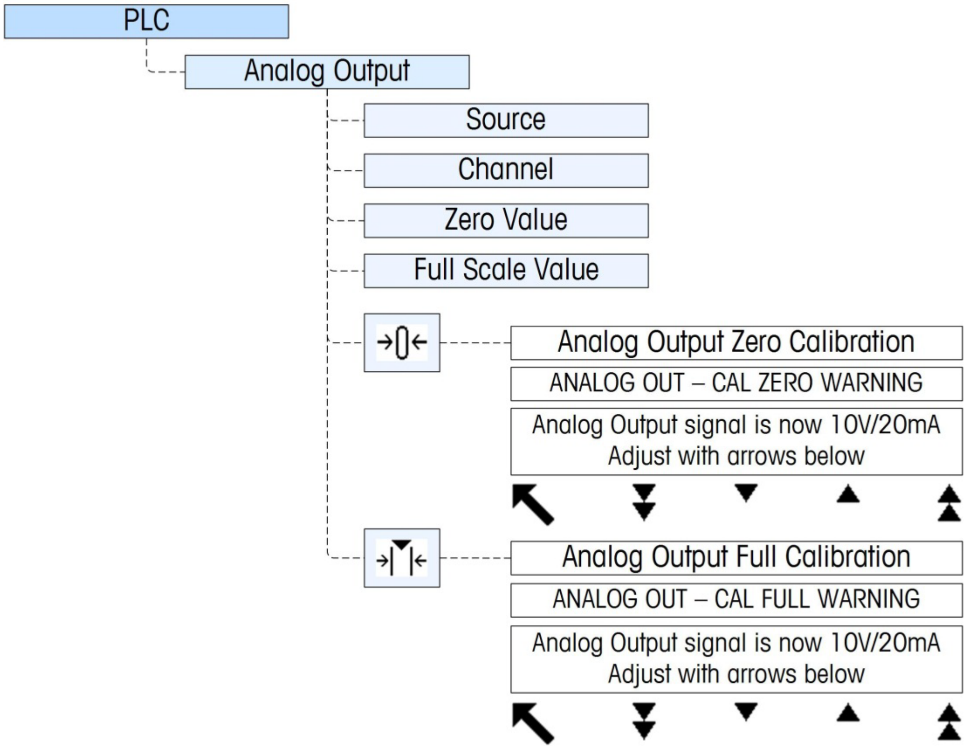

PLC
Analog Output
Source
Channel
Zero Value
Full Scale Value
Analog Output Zero Calibration
ANALOG OUT – CAL ZERO WARNING
Analog Output signal is now 10V/20mA
Adjust with arrows below
Analog Output Full Calibration
ANALOG OUT – CAL FULL WARNING
Analog Output signal is now 10V/20mA
Adjust with arrows below

Figure 1-3: Setup Procedures for Configuring the Analog Output Option Card

- In all cases, bear in mind that the 0-10 VDC output is not available when the analog output option is installed directly in the IND500x.

1.3.1. Analog Output Setup Sub-Block

To configure the Analog Output Kit Option:

1. With power to the IND500x terminal removed, connect a volt or current meter to the appropriate output. If the customer’s device is already connected, the meter is not necessary.
2. Apply power to the terminal and enter Setup. Navigate to PLC sub-block.
3. Select the Analog Output branch, then select the source. Choices are None, Displayed Weight (the default), Gross Weight, Rate. None disables the analog output. Displayed Weight outputs an analog signal based on the displayed net or gross weight. When Gross Weight 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, Rate must be configured at **Scale > Rate**.
4. Next, select the Channel. Options are Scale and None. Scale is the only option available now; None is reserved for future use.

5. At the Zero Value prompt, enter the desired source value for which the analog output should be zero. Typically this would be "0" in most applications; however, any valid value below the high limit can be used.
6. At the Full Scale Value prompt, enter the desired source value at which the analog output should be at its high limit. 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.
7. After all these parameters have been entered, the analog output can be adjusted to meet the customer's requirements using the ZERO softkey →0← and the SPAN softkey →|▼|←. To adjust the zero reference analog signal, press the ZERO softkey →0←.
8. Note that a display message is shown warning that during the adjustment, the analog output will be set to zero and will not monitor changes in 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.
9. At the Analog Output - Cal Zero screen, use the softkeys to adjust the analog output signal to be exactly zero on the customer's device. The available softkeys are described in Table 1-2.

Table 1-2: Softkey Descriptions

▼▼	Coarse Down	This adjusts the analog signal level down in large steps.
▼	Fine Down	This adjusts the analog signal level down in small steps.
▲	Fine Up	This adjusts the analog signal level up in small steps.
▲▲	Coarse Up	This adjusts the analog signal level up in large steps.

10. When the zero adjustment is complete, press the EXIT softkey ↖ to return to the Analog Output screen.
11. Now, the full scale analog output value cab be adjusted by pressing the SPAN softkey →|▼|←. A similar warning message will be shown indicating the analog output will be set to the high value and will not monitor changes in the source. Press the ESCAPE softkey Esc to exit the span adjustment process or press the OK softkey OK✓ to continue the adjustment process.
12. At the Analog Output - Cal Full screen, 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 1-2.

1.3.1.1. Setting a Negative Value

It is sometimes necessary to set a negative value to define the lower end of the span. For example, the 4mA output might be set to correspond to a weight value of -20 kg.

It is not possible to set a negative weight value directly from the IND500x operator interface. However, there are two ways of setting a negative value:

- Use an external QWERTY keyboard connected to the terminal's (optional) USB port to enter the negative value in setup on the terminal, in the Analog Output Zero Value field.

- Enter the negative value directly into Shared Data variable ao0103 (Analog Output Zero Preset).

1.4. Analog Output Kit Spare Parts

There are no associated spare parts with the Analog Output option kit.

1.4.1. Analog Output Kit for ACM500

The ACM500 kit number is 71209099. Table 1-3 shows what the kit contains.

Table 1-3: Analog Output Option Kit, ACM500

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

1.4.2. Analog Output Kit for IND500x

The IND500x kit number is 30674194. Table 1-4 shows what the kit contains.

Table 1-4: Analog Output Option Kit, IND500x

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

2 EtherNet/IP™ Interface

2.1. 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 IND500x 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 IND500x 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 IND500x.

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

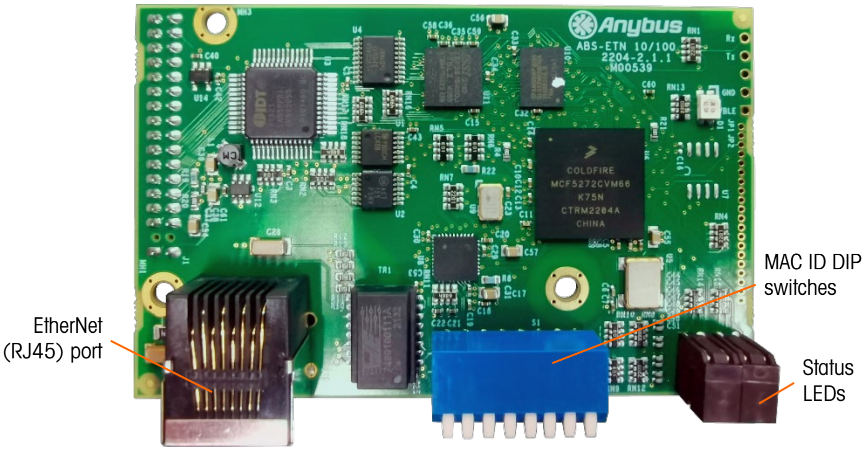


Figure 2-1: EtherNet/IP Module

2.1.1. Definition of Terms

Some terms (such as Target) used by the EtherNet/IP PLC application have a different sense from their use by the IND500x terminal. Table 2-1 provides definitions specific to EtherNet/IP.

Table 2-1: EtherNet/IP Definition of Terms

Term	Definition
Adapter Class	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.
Class 1 Messaging	In EtherNet/IP communication protocol scheduled (cyclic) message transfer between a PLC and CIP Adapter Class device.
Class 3 Messaging	In EtherNet/IP communication protocol unscheduled message transfer between a PLC and CIP Adapter Class device. This is used by the IND500x for explicit messaging.
Connected Messaging	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 Unconnected Messaging.
Connection Originator	Source for I/O connection or message requests. Initiates an I/O connection or explicit message connection.
Explicit Messaging	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.
Implicit Messaging	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 I/O Messaging.
I/O Client	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.
I/O Messaging	Used interchangeably with the term Implicit Messaging.
I/O Server	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
Message Client	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.
Message Server	Function that provides Explicit messaging services to another (Message Client) device. Responds to an Explicit message request from the Message Client.
Scanner Class	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.
Target	Destination for I/O connection or message requests. Can only respond to a request, cannot initiate an I/O connection or message.
Unconnected Messaging	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 Connected Messaging.

2.1.2. Communications

The IND500x terminal utilizes component parts to ensure complete compatibility with the Allen-Bradley EtherNet/IP network. An IND500x 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 2-1).

The wiring between the PLC and the IND500x 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 IND500x only uses Class 1 cyclic data for discrete data and explicit messages for access to the IND500x Shared Data Variables. Explicit message blocks may be connected or unconnected; the PLC programmer must make this choice.

2.1.3. IP Address

Each EtherNet/IP option represents one physical IP Address. This address is chosen by the system designer, and then programmed into the IND500x terminal and PLC. There is no feature for EtherNet/IP to use a host server to assign addresses. The IND500x terminal's address is programmed through **Communication > PLC Interface > EtherNet/IP** in the setup menu. IND500x IP Address entry must be unique for each IND500x.

2.1.4. 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 three data formats: Integer, Divisions and Floating Point.

- **Integer** – reports scale weight as a signed 16 bit integer (± 32767).
- **Divisions** – reports scale weight in display divisions (± 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

Appendix A and B provide detailed information on data formats.

2.2. 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.

2.2.1. Data Integrity

The IND500x has specific bits to allow the PLC to confirm that data was received without interruption and that the IND500x 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 IND500x. Refer to the data charts for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage.

2.2.2. Assembly Instances of Class 1 Cyclic Communications

Class 1 cyclic communications is used for transfer of Discrete Data between the PLC and the IND500x.

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 IND500x 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 IND500x programming controls the Assembly Instance and data size limitations.

2.2.3. Discrete Data

Three formats of discrete data are available with the EtherNet/IP interface option: integer (default), divisions and floating point.

The data format of discrete data will affect the data size required in the configuration of the PLC. The IND500x console PLC message slot setup screen provides data size requirements in bytes.

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 IND500x will clear all Message Slots. Data format is selected in the **Communication > PLC > Data Format** setup block, see Figure 2-2.

2.2.3.1. Examples

250 x .01 scale				
IND500x 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				
IND500x 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				
IND500x 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 or invalid value once the weight exceeded 32.767.

2.2.4. Byte Order

The byte order parameter sets the order in which the data bytes and words will be presented in the PLC data format. Available byte orders are:

- Word Swap - (default) Make the data format compatible with RSLogix 5000 processors.
- Byte Swap - Makes the data format compatible with S7 Profibus.
- Standard – Makes the data format compatible with PLC 5
- Double Word Swap - Makes the data format compatible with the Modicon Quantum PLC for Modbus TCP networks.

Table 2-2 provides examples of the various byte ordering.

Table 2-2: PLC Data Byte Ordering

<table>
<tr><th colspan="2"></th><th colspan="3">Word Swap</th><th colspan="3">Byte Swap</th><th colspan="3">Double Word Swap</th><th colspan="3">Standard</th></tr>
<tr><th colspan="2">Terminal Weight Value</th><td colspan="12">1355</td></tr>
<tr><th colspan="2">PLC</th><th>15</th><th>Bit #</th><th>0</th><th>15</th><th>Bit #</th><th>0</th><th>15</th><th>Bit #</th><th>0</th><th>15</th><th>Bit #</th><th>0</th></tr>
<tr><th>Integer</th><td>Weight value word</td><td colspan="3">0x054B Hex</td><td colspan="3">0x4B05 Hex</td><td colspan="3">0x4B05 Hex</td><td colspan="3">0x054B Hex</td></tr>
<tr><th rowspan="2">Floating Point</th><td>1st Weight value word</td><td colspan="3">0x6000 Hex</td><td colspan="3">0xA944 Hex</td><td colspan="3">0x0060 Hex</td><td colspan="3">0x44A9 Hex</td></tr>
<tr><td>2nd Weight value word</td><td colspan="3">0x44A9 Hex</td><td colspan="3">0x0060 Hex</td><td colspan="3">0xA944 Hex</td><td colspan="3">0x6000 Hex</td></tr>
</table>

2.2.5. Message Slots

There may be up to 4 message slots for discrete data transfer, Class 1 messaging, in Integer, Divisions and Floating Point Data Formats. Each message slot represents the scale but may be controlled by the PLC to present different data in each message slot. The integer and division formats provide two 16-bit words of input and two 16-bit words of output data per Slot. Each Message Slot's first input word provides scale weight data. The type of data displayed, such as Gross, Tare, etc., is selected by the PLC using the Message Slot's second output word bits 0, bit 1 and bit 2. The following two Tables provide input and output usage information.

The data format for the slots are described in Appendix A and B.

Table 2-3: 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
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

Table 2-4: 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

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 2-5 and *Refer to the Message Slot Table in Appendix B for details

Table 2-6 for details.

The number of Message Slots is selected in the **Communication > PLC > Data Format** setup menu, Figure 2-2.

The format of the data is shown in Appendix B

Table 2-5: EtherNet/IP PLC Floating Point Input Words

PLC Input Words	Bits 0 – 7	Bits 8 – 15
0	Message Slot 1: Reserved	Message Slot 1: Command Response
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	Message Slot 1: Scale Status
4	Message Slot 2: Reserved	Message Slot 2: Command Response
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	Message Slot 2: Scale Status
8	Message Slot 3: Reserved	Message Slot 3: Command Response
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	Message Slot 3: Scale Status

PLC Input Words	Bits 0 – 7	Bits 8 – 15
12	Message Slot 4: Reserved	Message Slot 4: Command Response
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	Message Slot 4: Scale Status

*Refer to the Message Slot Table in Appendix B for details

Table 2-6: 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

2.2.5.1. Floating Point Data Format and Compatibility

In Floating Point Data Format, the PLC and IND500x 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 IND500x 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.

2.2.5.2. Shared Data Mode

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

The IND500x Shared Data document 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.

2.2.6. Controlling the Discrete I/O Using a PLC Interface

The IND500x 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 IND500x 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 IND500x Terminal Technical Manual for discrete I/O wiring.

2.3. Software Setup

When the IND500x 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 > EtherNet/IP**. Figure 2-2 shows the EtherNet/IP setup block.

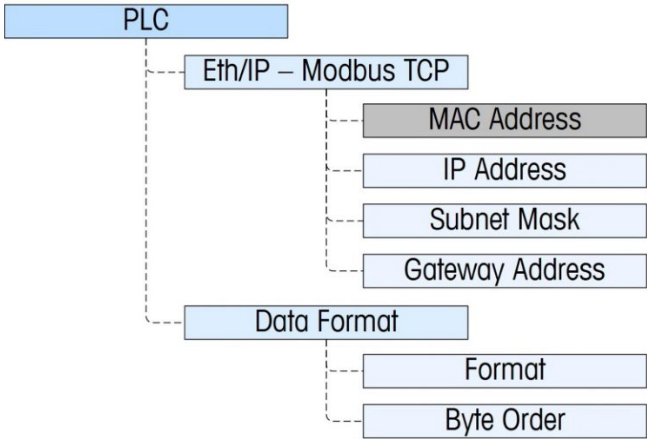


Figure 2-2: EtherNet/IP Setup Block

2.3.1. EtherNet/IP Setup Block

2.3.1.1. EtherNet/IP setup:

The EtherNet/IP Setup block in Setup at **Communication > PLC Interface > EtherNet/IP**, allows the EtherNet/IP interface's IP address, subnet mask and gateway address to be specified. The MAC address is displayed, but cannot be modified.

2.3.1.2. Data Format setup:

2.3.1.2.1. Format

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

2.3.1.2.2. Byte Order

Available selections are Standard, Byte Swap, Word Swap (default), and Double Word Swap.

2.3.1.2.3. Message Slots

Select 1, 2, 3 or 4 slots.

2.4. Troubleshooting

If the IND500x does not communicate with PLC, do the following:

- Check wiring and network termination.
- Confirm that the IND500x settings for data type, size and IP Address assignment match those in the PLC and that each IND500x 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, DeviceNet or Remote I/O, a master reset of the IND500x should be performed.

2.4.1. 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 2-3 shows the array of the LEDs on the card. Table 2-7 explains the meaning if the indicators.

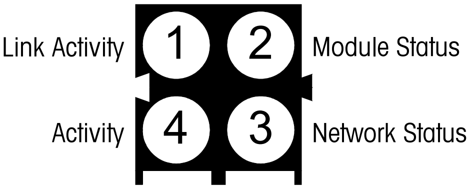


Figure 2-3: EtherNet/IP Status Indicator LEDs

Table 2-7: 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

2.5. EtherNet/IP Option Kit

No spare parts are associated with the EtherNet/IP option kit. The kit CIMF part number is 64058677. Table 2-8 shows what comes in the kit.

Table 2-8: EtherNet/IP Option Kit

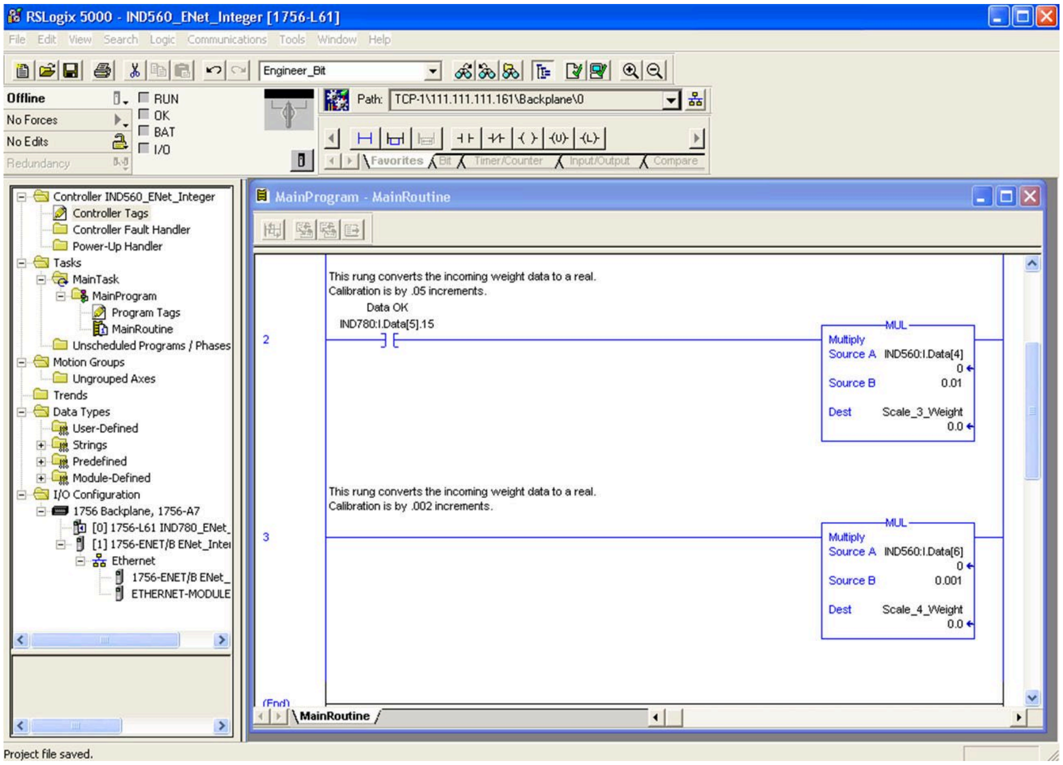
Description	Qty.
Installation Instructions	1
PCB package	1
Clamp-on ferrite	1

Description	Qty.
Medium grommet	1
M3 x 8 screws	3
Small metric grip bushing	1
Back plate cover	1

2.6. Programming Examples

The following Figures show sample screen images of ladder logic programming examples for RSLogix 5000 software (version 16).

- Note: The Utilities folder of the documentation CD (part number 30725989) contains complete versions of the examples. These screen images are provided for illustrative purposes only.



RSLogix 5000 - IND560_ENet_Integer [1756-L61]

Path: TCP-1\111.111.111.161\Backplane\0

MainProgram - MainRoutine

Rung 2: This rung converts the incoming weight data to a real. Calibration is by .05 increments.

Data OK IND780:I.Data[5].15

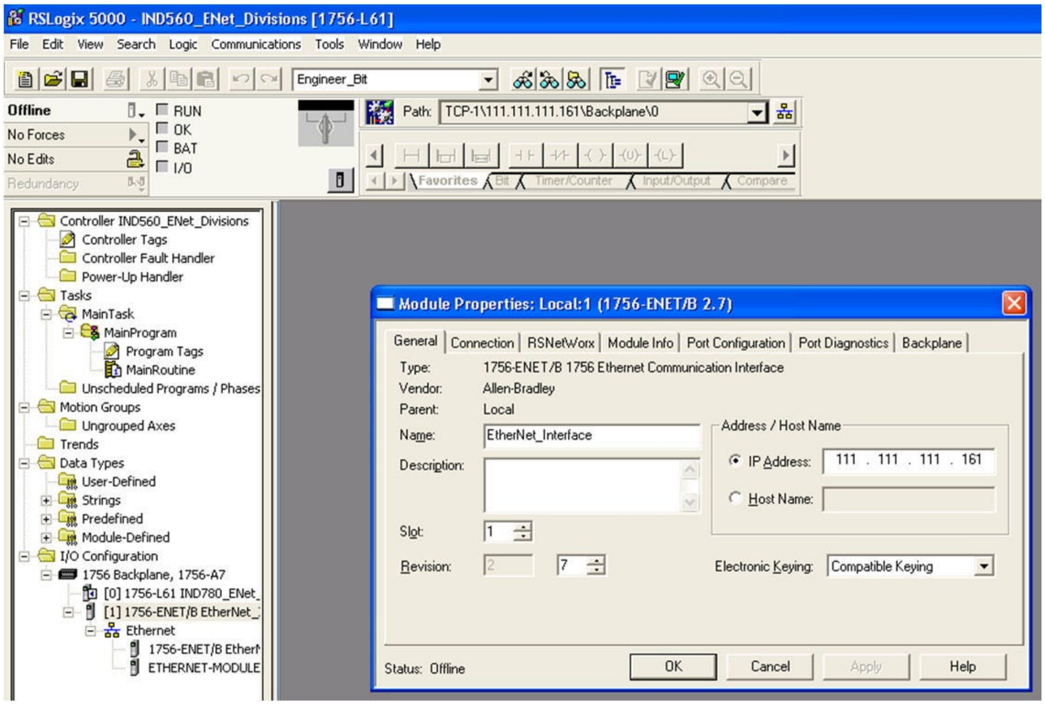
MUL Multiply
Source A IND560:I.Data[4] 0
Source B 0.01
Dest Scale_3_Weight 0.0

Rung 3: This rung converts the incoming weight data to a real. Calibration is by .002 increments.

MUL Multiply
Source A IND560:I.Data[6] 0
Source B 0.001
Dest Scale_4_Weight 0.0

Project file saved.

Figure 2-4: Integer Weight Data to Real



Path: TCP-1\111.111.111.161\Backplane\0

Module Properties: Local:1 (1756-ENET/B 2.7)

Type: 1756-ENET/B 1756 Ethernet Communication Interface

Vendor: Allen-Bradley

Parent: Local

Name: EtherNet_Interface

Description:

Slot: 1

Revision: 2 7

Address / Host Name

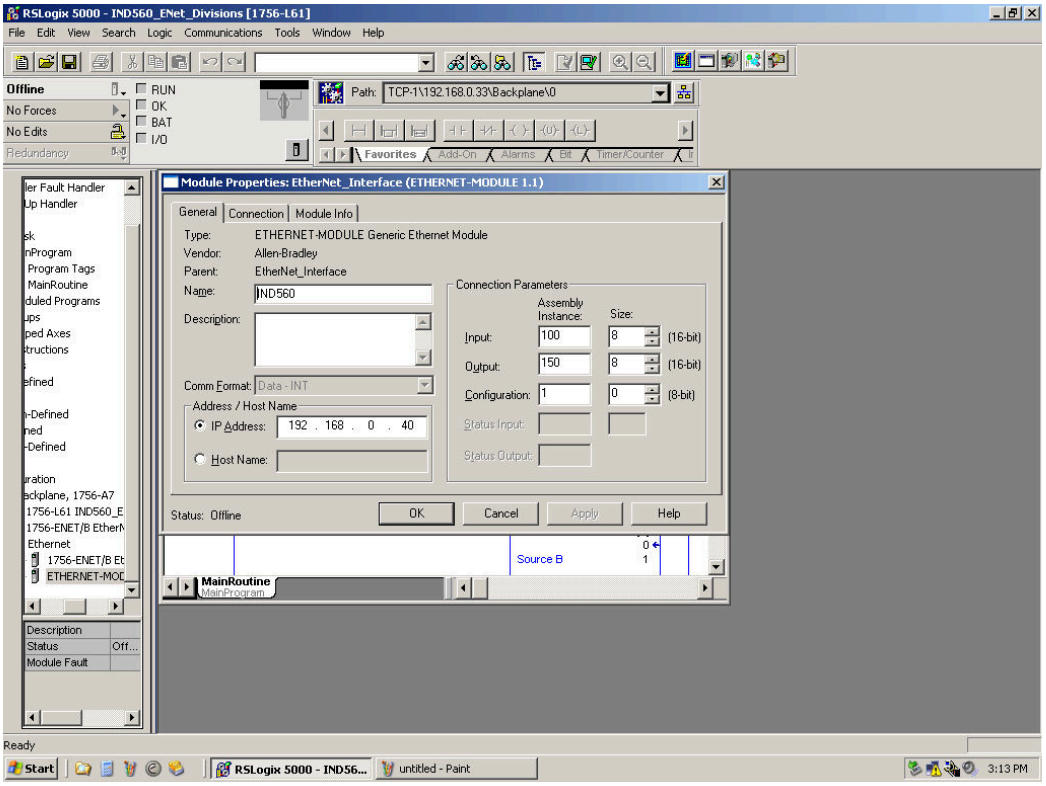
● IP Address: 111 . 111 . 111 . 161

○ Host Name:

Electronic Keying: Compatible Keying

Status: Offline

Figure 2-5: EtherNet Bridge in PLC Setup



Path: TCP-1\192.168.0.33\Backplane\0

Module Properties: EtherNet_Interface (ETHERNET-MODULE 1.1)

Type: ETHERNET-MODULE Generic Ethernet Module

Vendor: Allen-Bradley

Parent: EtherNet_Interface

Name: IND560

Description:

Comm Format: Data - INT

Address / Host Name

● IP Address: 192 . 168 . 0 . 40

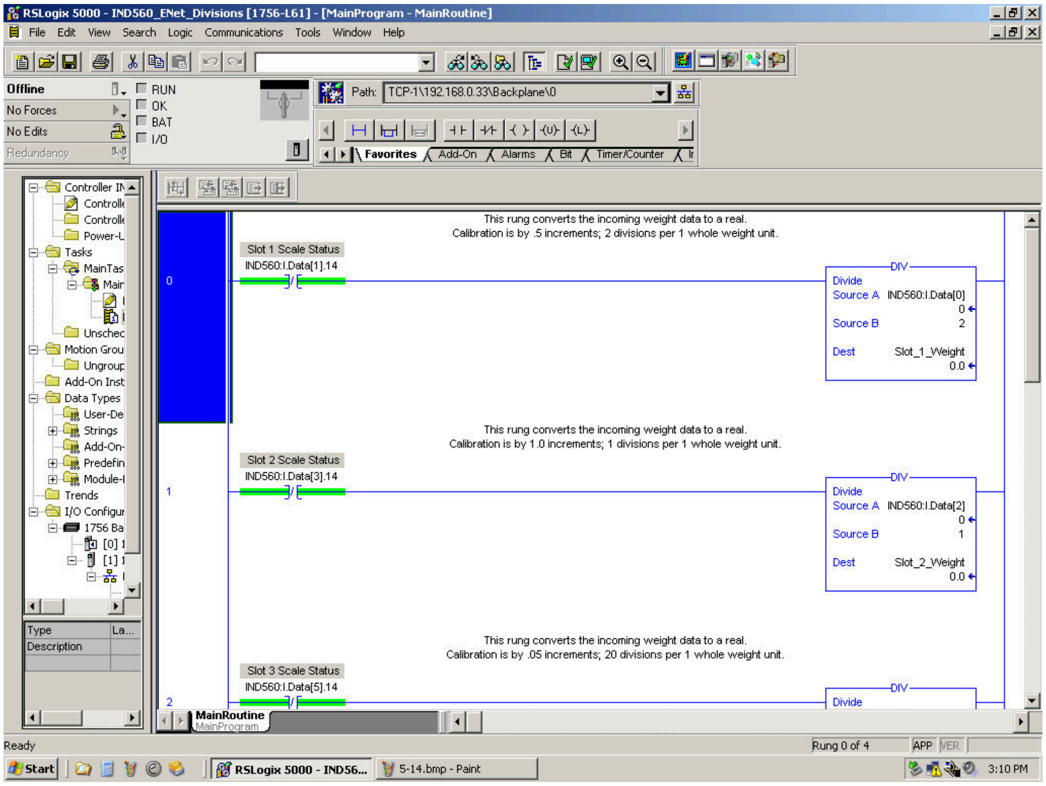
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	100	8	(16-bit)
Output:	150	8	(16-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

Status: Offline

Figure 2-6: EtherNet Module, Divisions



This rung converts the incoming weight data to a real.
Calibration is by .5 increments; 2 divisions per 1 whole weight unit.

0 Slot 1 Scale Status IND560:I.Data[1].14 — DIV Divide Source A IND560:I.Data[0] 0 Source B 2 Dest Slot_1_Weight 0.0

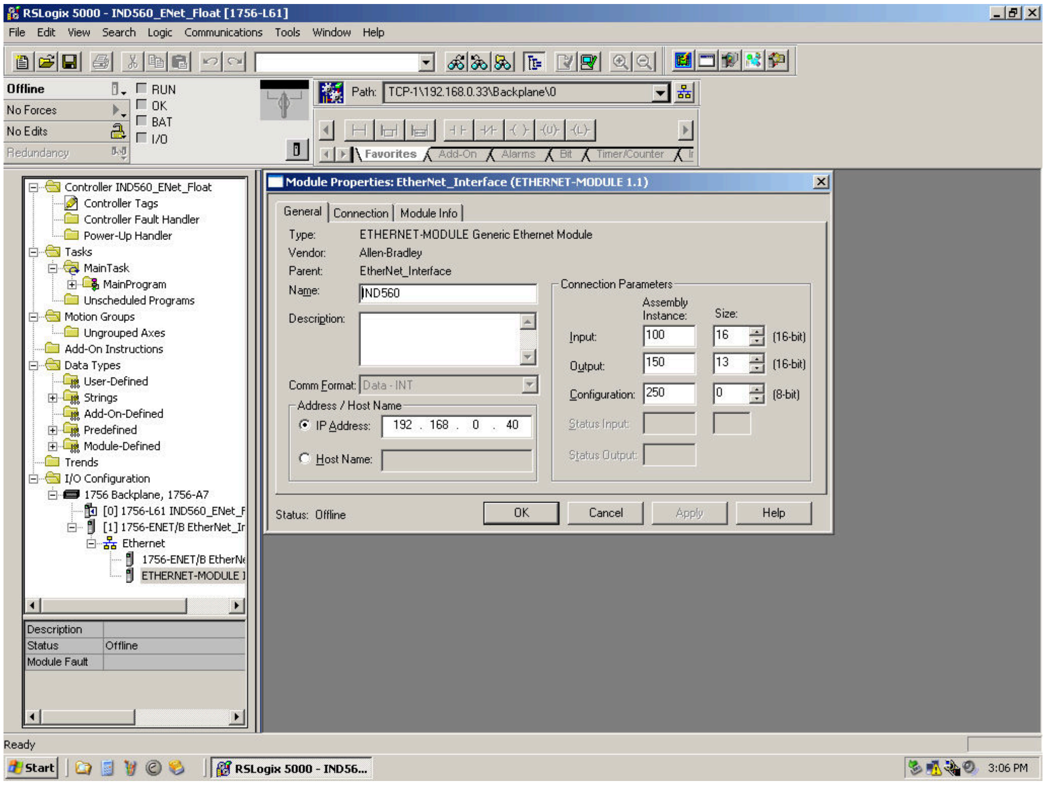
This rung converts the incoming weight data to a real.
Calibration is by 1.0 increments; 1 divisions per 1 whole weight unit.

1 Slot 2 Scale Status IND560:I.Data[3].14 — DIV Divide Source A IND560:I.Data[2] 0 Source B 1 Dest Slot_2_Weight 0.0

This rung converts the incoming weight data to a real.
Calibration is by .05 increments; 20 divisions per 1 whole weight unit.

2 Slot 3 Scale Status IND560:I.Data[5].14 — DIV Divide

Figure 2-7: Integer Weight Data to Real, Division



Description
Status Offline
Module Fault

Module Properties: EtherNet_Interface (ETHERNET-MODULE 1.1)

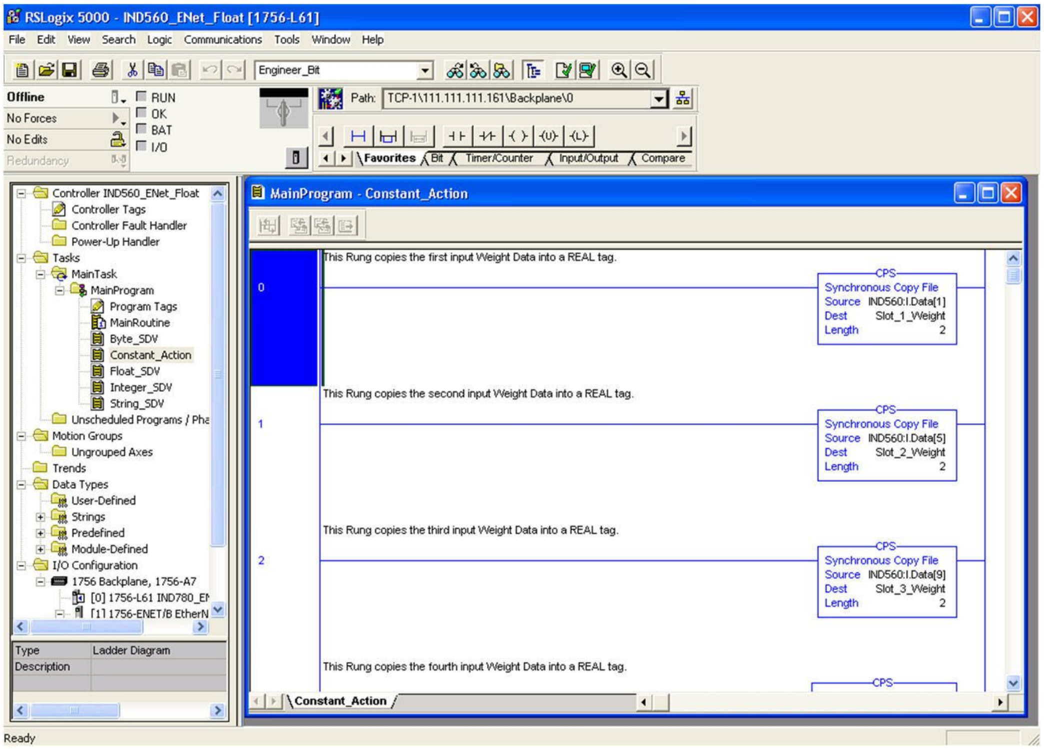
Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: EtherNet_Interface
Name: IND560
Description:
Comm Format: Data - INT
Address / Host Name
● IP Address: 192 . 168 . 0 . 40
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	100	16	(16-bit)
Output:	150	13	(16-bit)
Configuration:	250	0	(8-bit)
Status Input:			
Status Output:			

Status: Offline

Figure 2-8: EtherNet Module, Floating Point



RSLogix 5000 - IND560_ENet_Float [1756-L61]

MainProgram - Constant_Action

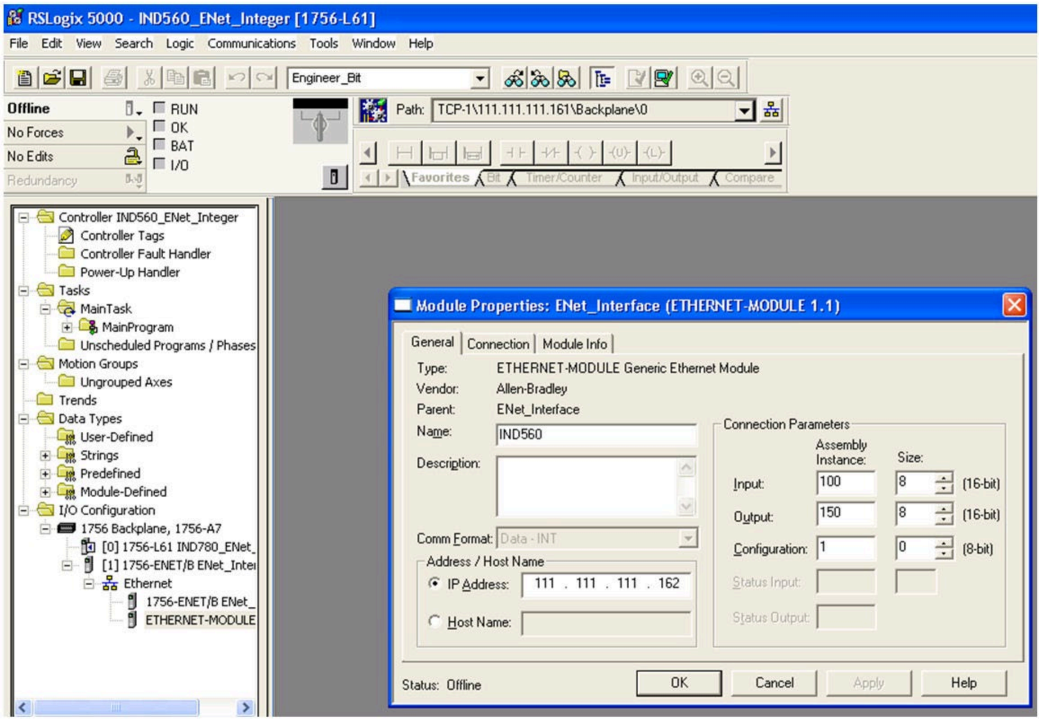
0 This Rung copies the first input Weight Data into a REAL tag.
CPS Synchronous Copy File
Source IND560:I.Data[1]
Dest Slot_1_Weight
Length 2

1 This Rung copies the second input Weight Data into a REAL tag.
CPS Synchronous Copy File
Source IND560:I.Data[5]
Dest Slot_2_Weight
Length 2

2 This Rung copies the third input Weight Data into a REAL tag.
CPS Synchronous Copy File
Source IND560:I.Data[9]
Dest Slot_3_Weight
Length 2

This Rung copies the fourth input Weight Data into a REAL tag.

Figure 2-9: Input Weight Data to Real Tag



RSLogix 5000 - IND560_ENet_Integer [1756-L61]

Module Properties: ENet_Interface (ETHERNET-MODULE 1.1)

General | Connection | Module Info

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: ENet_Interface
Name: IND560
Description:
Comm Format: Data - INT
Address / Host Name
● IP Address: 111 . 111 . 111 . 162
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	100	8	(16-bit)
Output:	150	8	(16-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

Status: Offline

Figure 2-10: EtherNet Module, Integer

3 Modbus TCP Interface

3.1. Preface

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

3.2. Overview

Modbus protocol is a messaging structure developed by Modicon in 1979. It is used to establish device/controller 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 device/controller 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.

3.2.1. Specifications

Network Type	Ethernet-TCP/IP based simple Client/Server network.
Topology	Star, tree or line structures; all topologies that can be implemented with standard Ethernet technology, including switched networks, are applicable.
Installation	Standard 10, 100 Mbit/s Ethernet technology based on copper cables, fiber optic or wireless standards can be used. The IND500x Modbus TCP option provides an RJ-45 Ethernet port connection
Speed	10, 100 Mbit/s.
Max. stations	Nearly unlimited.
Network features	Client/Server network based on standard Ethernet technology and TCP/UDP/IP protocols in Layer 3-4.
User Organization	Modbus-IDA user Group.

3.3. 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 IND500x.

3.4. Modbus TCP Board

Figure 3-1 shows a view of the EtherNet/IP Option Board used for Modbus TCP communication. The board's port, DIP switches and status lights are indicated. Note that the module's address is set in software, and the DIP switches must all be set to OFF.

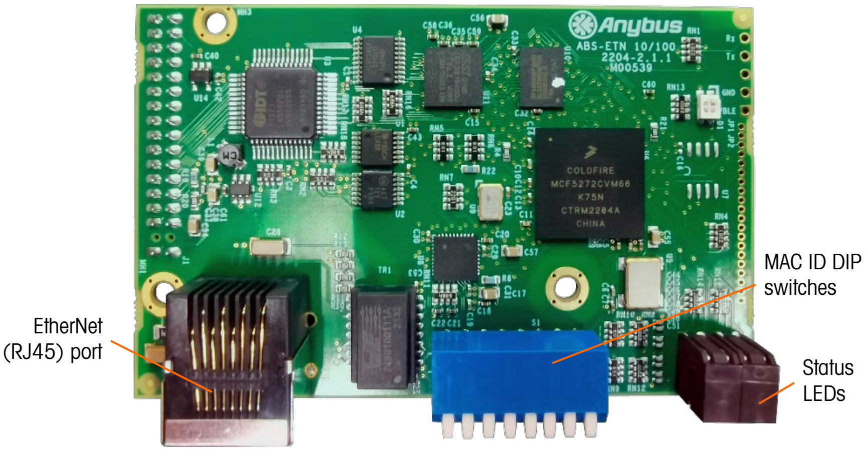


Figure 3-1: EtherNet/IP - Modbus TCP Option Board Components

3.4.1. Communications

The IND500x terminal utilizes component parts to ensure complete compatibility with the Modbus TCP network. An IND500x 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 3-1).

The wiring between the PLC and the IND500x 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.

3.4.2. 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 IND500x terminal and PLC. The IND500x terminal's address is programmed at **Communication > PLC > EtherNet/IP** in the setup menu. The IND500x IP Address entry must be unique for each IND500x.

3.4.3. Data Formats

The Modbus TCP Kit option provides discrete data transfer 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 three data formats: Integer, Divisions, and Floating Point.

- **Integer** – reports scale weight as a signed 16 bit integer (± 32767).
- **Divisions** – reports scale weight in display divisions (± 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

Appendix A and B provide detailed information on data formats.

3.5. Data Definition

3.5.1. Data Integrity

The IND500x has specific bits to allow the PLC to confirm that data was received without interruption and that the IND500x 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 IND500x. Refer to the data charts in Appendix A and B for specific information regarding the Data OK, Update in Progress and Data Integrity bits and their usage.

3.5.2. Discrete Data

Three formats of discrete data are available with the Modbus TCP Kit option: integer, divisions and floating point. Only one type of data format may be selected and used by the IND500x.

The integer and division formats allow bi-directional communication of discrete bit encoded information or 16 bit binary word numerical values. Refer to Appendix A for details.

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.

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 as needed with these formats.

3.5.2.1.1. Examples

250 x .01 scale				
IND500x 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				
IND500x 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 or invalid value when the weight exceeds 32,767.

150 x .001 scale				
IND500x 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 format could not be used because it would send a negative or invalid value when the weight exceeds 32,767.

3.5.3. Byte Order

The byte order parameter sets the order in which the data bytes and words will be presented in the PLC data format. Available byte orders are:

- Word Swap - (default) Makes the data format compatible with RSLogix 5000 processors.
- Byte Swap - Makes the integer mode compatible with Modicon Quantum processors.
- Standard – Makes the data format compatible with PLC 5
- Double Word Swap - Makes the data format compatible with the Modicon Quantum PLC for Modbus TCP networks.

Table 3-1 provides examples of the various byte ordering.

Table 3-1: PLC Data Byte Ordering

		Word Swap	Byte Swap	Double Word Swap	Standard
Terminal Weight Value		1355			
PLC		**15 Bit # 0**	**15 Bit # 0**	**15 Bit # 0**	**15 Bit # 0**
Integer	Weight value word	0x054B Hex	0x4B05 Hex	0x4B05 Hex	0x054B Hex
Floating Point	1st Weight value word	0x6000 Hex	0xA944 Hex	0x0060 Hex	0x44A9 Hex
	2nd Weight value word	0x44A9 Hex	0x0060 Hex	0xA944 Hex	0x6000 Hex

3.5.4. Register Mapping

The memory of the Modbus TCP Kit option board is mapped as shown in Table 3-2. The read and write areas of memory are offset by 1024. In a Quantum PLC, the PLC would read data from the IND500x starting at 400001 and would write data to the IND500x starting at register 401025.

Table 3-2: Modbus TCP-IP PLC Input and Output data map

Register #	Area	Offset In Area
1	Read Data (from IND500x)	0000h…0001h
2		0002h….0003h
3		0004h…0005h
4		0006h…0007h
……	……	………
1024	Write Data (To IND500x)	0000h…0001h
1025		0002h….0003h
1026		0004h…0005h
1027		0006h…0007h
……		………

3.5.5. Message Slots

The IND500x can be configured for up to 4 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. 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 the PLC documentation for I/O mapping.

Table 3-3: 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

Table 3-4: 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
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

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 3-5 and Table 3-6 for details.

The number of Message Slots is set up in **Communication > PLC > Data Format** setup menu.

Table 3-5: 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*
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	Message Slot 1: Scale Status
400005	Message Slot 2: Reserved	Message Slot 2: Command Response
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	Message Slot 2: Scale Status
400009	Message Slot 3: Reserved	Message Slot 3: Command Response
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	Message Slot 3: Scale Status
400013	Message Slot 4: Reserved	Message Slot 4: Command Response
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	Message Slot 4: Scale Status

*See Message Slot Table Appendix B for details

Table 3-6: 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
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

3.5.6. Integer and Division

When one of these formats is selected, the IND500x 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 IND500x 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.

Appendix A provides detailed information on the integer and division data formats.

3.5.7. Floating Point

3.5.7.1. 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 20 Hz or 58 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 Appendix B for additional information. The method of handling string and floating point data may vary between different PLCs. The IND500x provides floating point data in the byte order entered in the Data Format setup.

Appendix B provides detailed information on the floating point data format. The byte order shown follows the convention of the Modicon Quantum platform. Note that this is Double Word Swapped.

3.5.7.2. Floating Point Data Format and Compatibility

In Floating Point Message mode, the PLC and IND500x 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 IND500x 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 IND500x provides floating point data in the word order set up by the user.

3.5.8. Controlling the Discrete I/O Using a PLC Interface

The IND500x 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 IND500x Terminal Technical Manual for discrete I/O wiring. Also note that the outputs must be unassigned in the IND500x terminal setup.

3.6. Software Setup

The IND500x terminal automatically detects the presence of a Modbus TCP Kit option board if one is installed. When the option is detected, the IND500x terminal enables the Modbus TCP parameters in a program block at **Communications > PLC Interface > Ethernet/IP**. Figure 3-2 shows the Modbus TCP setup block.

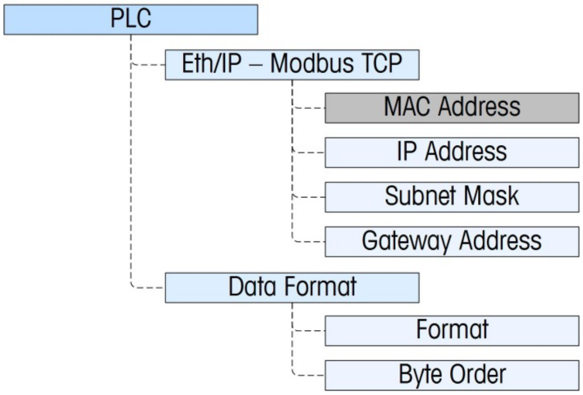


Figure 3-2: The Modbus TCP Setup Block

3.6.1. Modbus TCP Setup Block

3.6.1.1. Modbus TCP setup

Setup for Modbus TCP is the same setup block used for EtherNet/IP setup. The Modbus setup, found at **Communication > PLC > Ethernet/IP**, allows the Modbus TCP interface's IP address, subnet mask and gateway address to be specified. The MAC address is displayed but cannot be modified.

3.6.1.2. Data Format setup

3.6.1.2.1. Format

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

- **Integer** – reports scale weight as a signed 16 bit integer (± 32767).
- **Divisions** – reports scale weight in display divisions (± 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

3.6.1.2.2. Byte Order

Available selections are Standard, Byte Swap, Word Swap (default), and Double Word Swap.

Message Slots

Select 1, 2, 3 or 4 slots.

3.7. Troubleshooting

If the IND500x does not communicate with PLC, do the following:

- Check wiring and network termination.
- Confirm that the IND500x settings for data type and IP Address assignment match those in the PLC and that each IND500x 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 IND500x must be performed.

3.7.1. Status LEDs

The EtherNet/ IP – Modbus TCP interface card has four status LEDs indicators to indicate communication and fault status of the card. Figure 3-1 indicates the location of these LEDs, and Figure 3-3 shows the array of the LEDs on the card. Table 3-7 explains the meaning

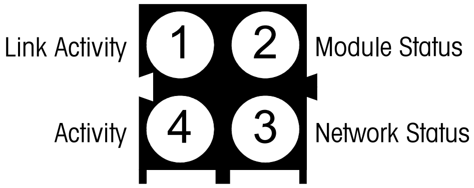


Figure 3-3:– Modbus TCP Board Status LED Array

Table 3-7: EtherNet / IP LEDs Status Indications

LED #	State	Status
1 - Link Activity	Off	No link (or no power)
	Green	Connected to an Ethernet network

LED #	State	Status
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
	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

3.8. Modbus TCP Option Kit

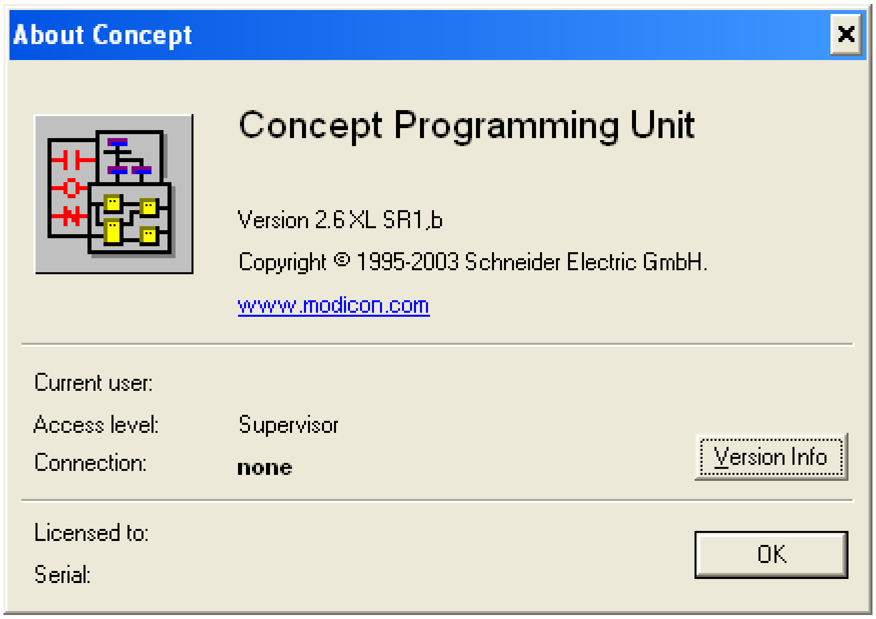
No spare parts are associated with the Modbus TCP option kit. The kit CIMF part number is 64058677. Table 3-8 shows what comes in the kit.

Table 3-8: Modbus TCP Option Kit

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

3.9. Modbus TCP Configuration Example

This demo was set up using Concept Version 2.6 XL, SR1, b (Figure 3-4).



About Concept

Concept Programming Unit

Version 2.6 XL SR1,b

Copyright © 1995-2003 Schneider Electric GmbH.

www.modicon.com

Current user:

Access level: Supervisor

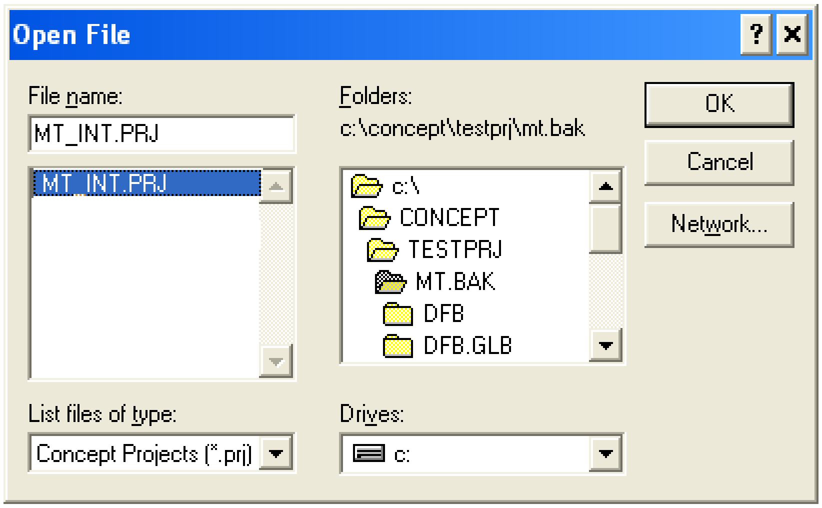
Connection: none

Licensed to:

Serial:

Figure 3-4: 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 3-5).



Open File

File name: MT_INT.PRJ

MT_INT.PRJ

Folders: c:\concept\testprj\mt.bak

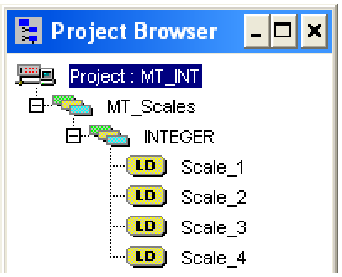
List files of type: Concept Projects (*.prj)

Drives: c:

Figure 3-5: 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 3-6) item to open the PLC Configuration window.

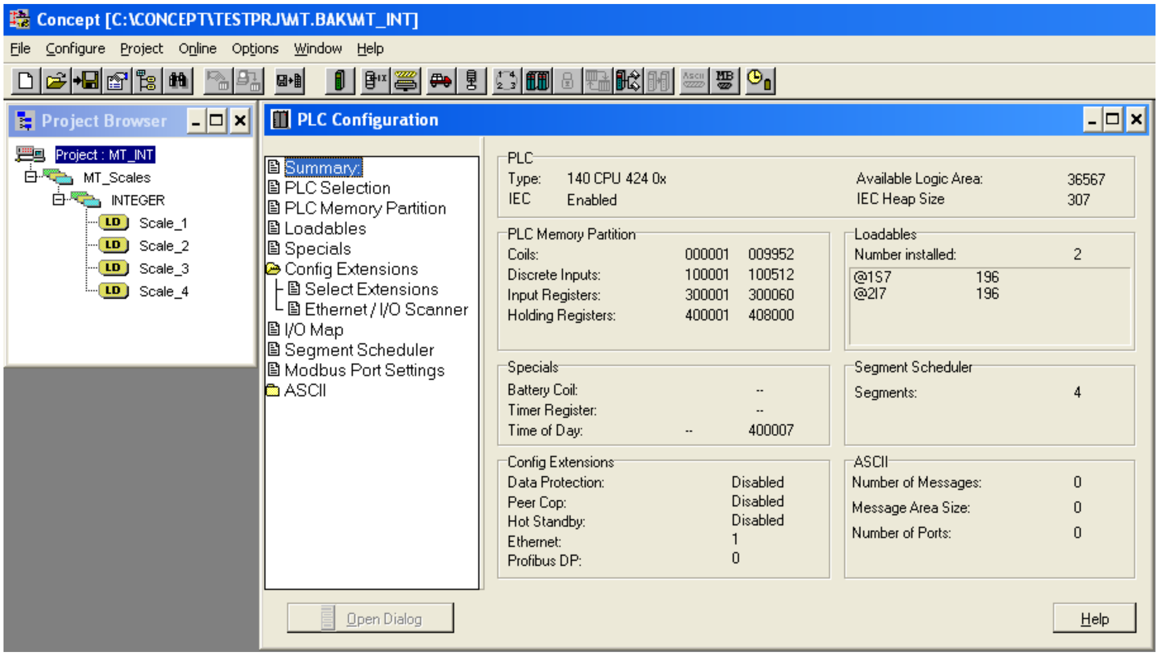


Project Browser

- Project : MT_INT
  - MT_Scales
    - INTEGER
      - LD Scale_1
      - LD Scale_2
      - LD Scale_3
      - LD Scale_4

Figure 3-6: Project Viewed in Project Browser

The PLC Configuration window (Figure 3-7) will open.



Concept [C:\CONCEPT\TESTPRJ\MT.BAK\MT_INT]

Project Browser

- Project : MT_INT
  - MT_Scales
    - INTEGER
      - LD Scale_1
      - LD Scale_2
      - LD Scale_3
      - LD Scale_4

PLC Configuration

- Summary
- PLC Selection
- PLC Memory Partition
- Loadables
- Specials
- Config Extensions
  - Select Extensions
  - Ethernet / I/O Scanner
- I/O Map
- Segment Scheduler
- Modbus Port Settings
- ASCII

PLC

Type:	140 CPU 424 0x	Available Logic Area:	36567
IEC	Enabled	IEC Heap Size	307

PLC Memory Partition

Coils:	000001	009952
Discrete Inputs:	100001	100512
Input Registers:	300001	300060
Holding Registers:	400001	408000

Loadables

Number installed:	2
@1S7	196
@2I7	196

Specials

Battery Coil:		--
Timer Register:		--
Time of Day:	--	400007

Segment Scheduler

Segments:	4

Config Extensions

Data Protection:	Disabled
Peer Cop:	Disabled
Hot Standby:	Disabled
Ethernet:	1
Profibus DP:	0

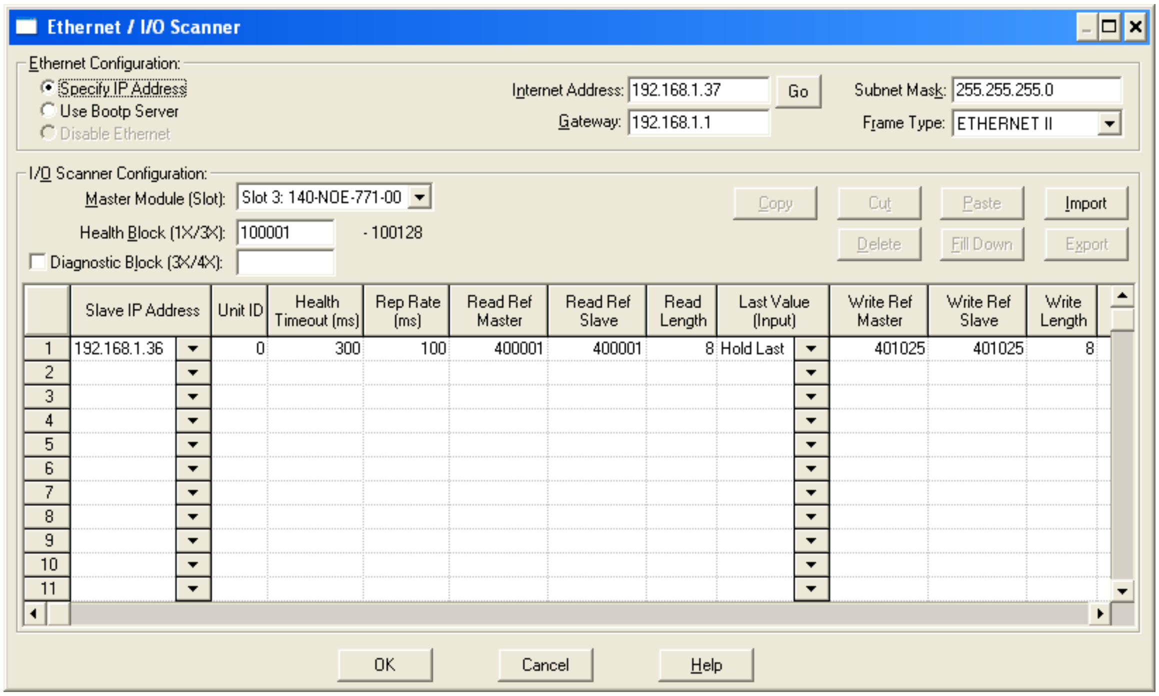
ASCII

Number of Messages:	0
Message Area Size:	0
Number of Ports:	0

Figure 3-7: 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 3-8).

Here, the IP addresses must be configured – the PLC's, that of the IND500x with which it communicates. The data communicated to and from the IND500x is also configured in this window.



Ethernet Configuration:

● Specify IP Address
○ Use Bootp Server
○ Disable Ethernet

Internet Address: 192.168.1.37

Gateway: 192.168.1.1

Subnet Mask: 255.255.255.0

Frame Type: ETHERNET II

I/O Scanner Configuration:

Master Module (Slot): Slot 3: 140-NOE-771-00

Health Block (1X/3X): 100001 - 100128

□ Diagnostic Block (3X/4X):

	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
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											

Figure 3-8: 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 3-8). The following elements must be configured:

Remote device IP Address: IP Address of the IND500x terminal's Modbus TCP interface. This value is configured in the IND500x Setup tree at **Communication > PLC Interface > EtherNet/IP**.

Unit ID: This value is typically 0

Health Timeout:

Rep Rate:

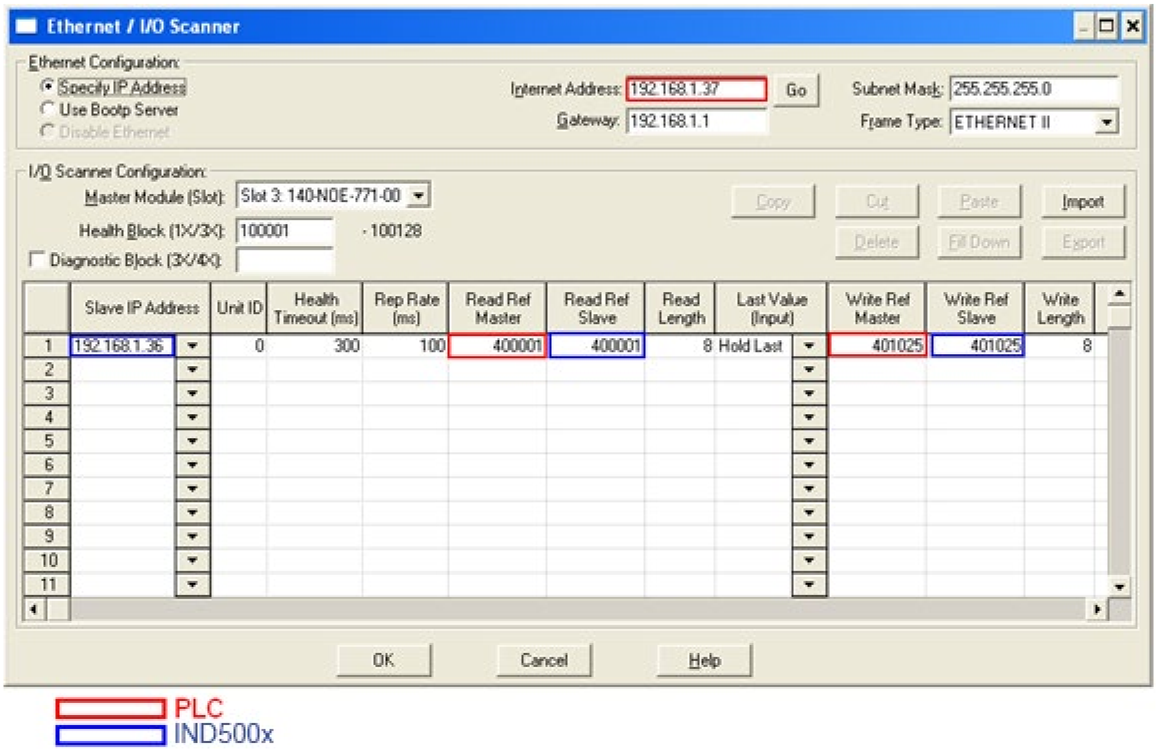
Read Ref Master: The start of PLC registers to which the IND500x's information is written. This address ALWAYS is 400001

Read Ref Remote device: The start of IND500x register where the scale data is stored. This address can be any value 4XXXXX PLC address.

* NOTE, data in the Read Ref Remote device is read and then stored in the Read Ref Master.

Read Length & Write Length: This is determined by the IND500x settings, and is determined by # of scales, Mode of operation etc. In our example we are using 4 slots in INTEGER Mode. In the IND500x 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 IND500x IP address and address settings must be configured – refer to Figure 3-9. The Ethernet card used on the configuration shown is the 140-NOE-771-00



Ethernet / I/O Scanner

Ethernet Configuration:

● Specify IP Address
○ Use Bootp Server
○ Disable Ethernet

Internet Address: 192.168.1.37 Go

Gateway: 192.168.1.1

Subnet Mask: 255.255.255.0

Frame Type: ETHERNET II

I/O Scanner Configuration:

Master Module (Slot): Slot 3: 140-NOE-771-00

Health Block (1X/3X): 100001 - 100128

□ Diagnostic Block (3X/4X):

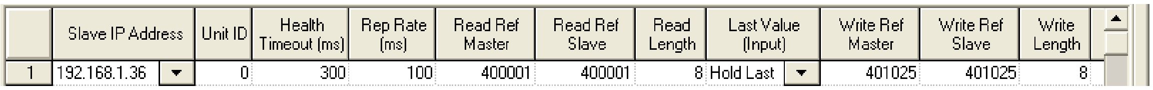
	Slave IP Address	Unit ID	Health Timeout (ms)	Rep Rate (ms)	Read Ref Master	Read Ref Slave	Read Length	Last Value (Input)	Write Ref Master	Write Ref Slave	Write Length
1	192.168.1.36	0	300	100	400001	400001	8	Hold Last	401025	401025	8
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											

Figure 3-9: PLC and IND500x Values for Ethernet / I/O Scanner

Examples of how to configure the Modicon Ethernet I/O scanner for various scale configurations are provided below.

3.9.1. Integer and Division Mode Configuration

The IND500x Configured for 4 slots in either integer or division mode. 8 Words are Read into the PLC and 8 words are written to the IND500x. Table 3-9 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 3-10: Integer or Division Mode Configuration

Table 3-9: Configuration of Integer or Division Mode

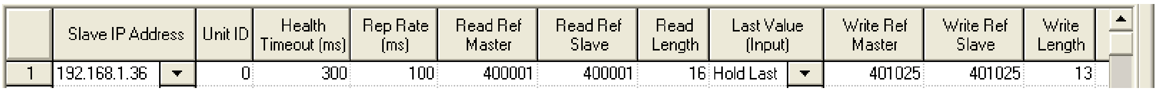
Description	Slot / Scale*	Address in IND500x	Format
Read by PLC from IND500x:			
Weight Data	Slot 1	400001	Int
Status Data	Slot 1	400002	Int
Weight Data	Slot 2	400003	Int
Status Data	Slot 2	400004	Int
Weight Data	Slot 3	400005	Int
Status Data	Slot 3	400006	Int

Description	Slot / Scale*	Address in IND500x	Format
Read by PLC from IND500x:			
Weight Data	Slot 4	400007	Int
Status Data	Slot 4	400008	Int
The PLC will write to:			
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

* 4001, 40001, 400001 are PLC Memory Dependent.

3.9.2. Floating Point Mode Configuration

The IND500x Configured for 4 slots in Floating Point mode FP. 16 Words are Read into the PLC and 13 words are written to the IND500x. Table 3-10 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	16	Hold Last	401025	401025	13

Figure 3-11: FLP Mode Configuration

Table 3-10: Configuration Floating Point Mode

Description	Slot / Scale*	Address in IND500x	Format
Read by PLC from IND500x:			
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
Weight Data	Slot 3	400010-400011	Float
Command Ack Register	Slot 3	400009	Int

Description	Slot / Scale*	Address in IND500x	Format
Read by PLC from IND500x:			
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
The PLC will write to:			
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 data can be configured to correspond with any slot number. 4001, 40001, 400001 PLC Memory Dependent.

3.9.3. Integer Logic Examples

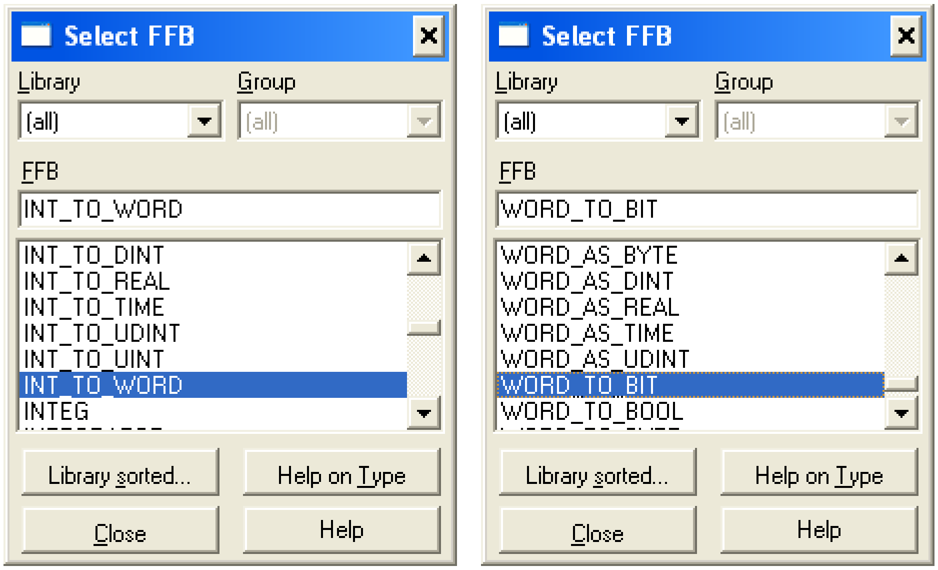
2 Words of Data are associated with a scale when in integer mode.

- Weight Data for scale 1 is stored in the IND500x in register 400001.
- Status Data for this weight and the IND500x is in register 400002.

3.9.3.1. 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 integer value from the IND500x 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 3-12 and Figure 3-13 show an example of logic that can be used to read the status word.



Library: (all)
Group: (all)
FFB: INT_TO_WORD

Library: (all)
Group: (all)
FFB: WORD_TO_BIT

Figure 3-12: Selecting Integer-to-Word (left) and Word-to-Bit (right) Conversions

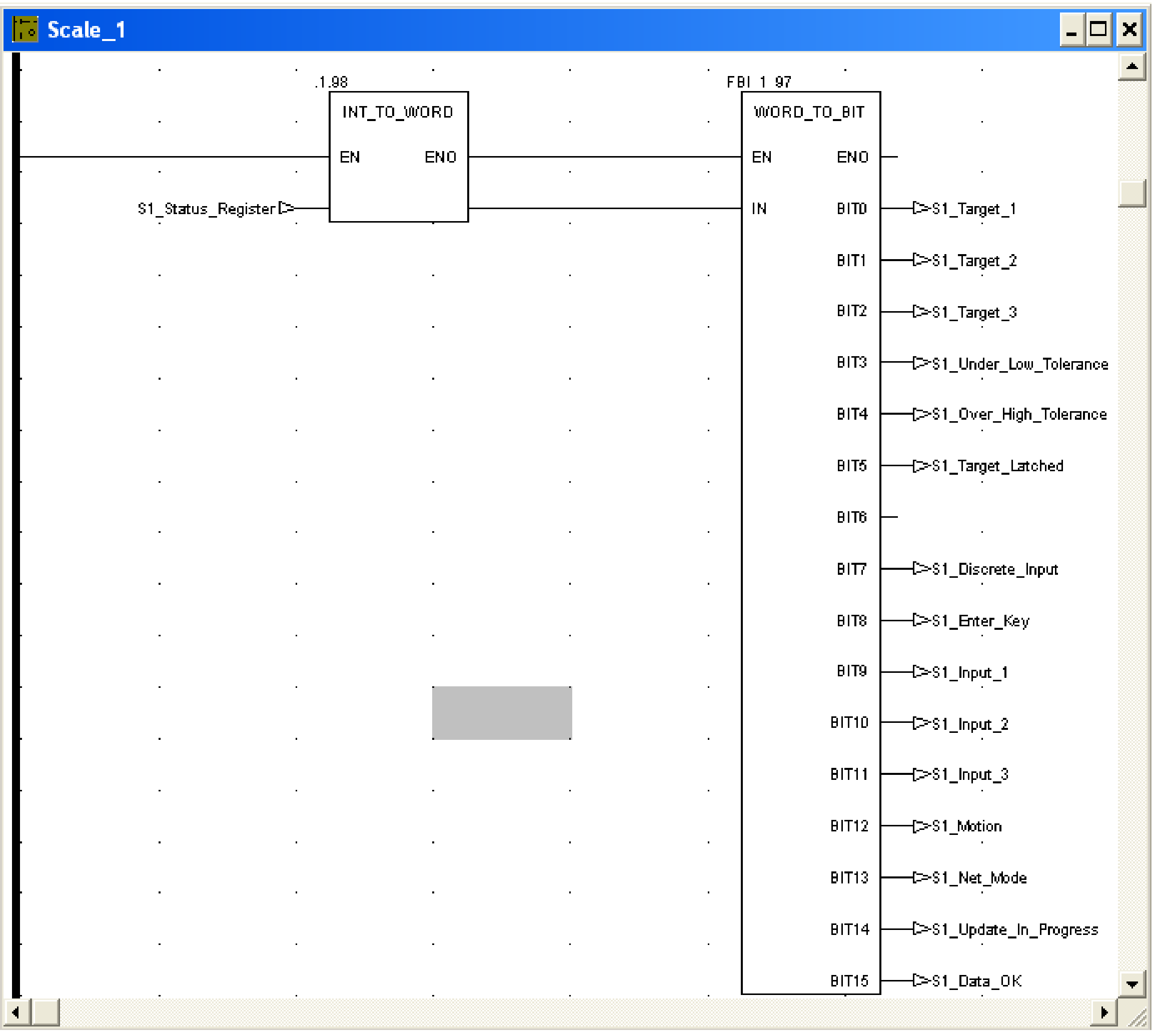
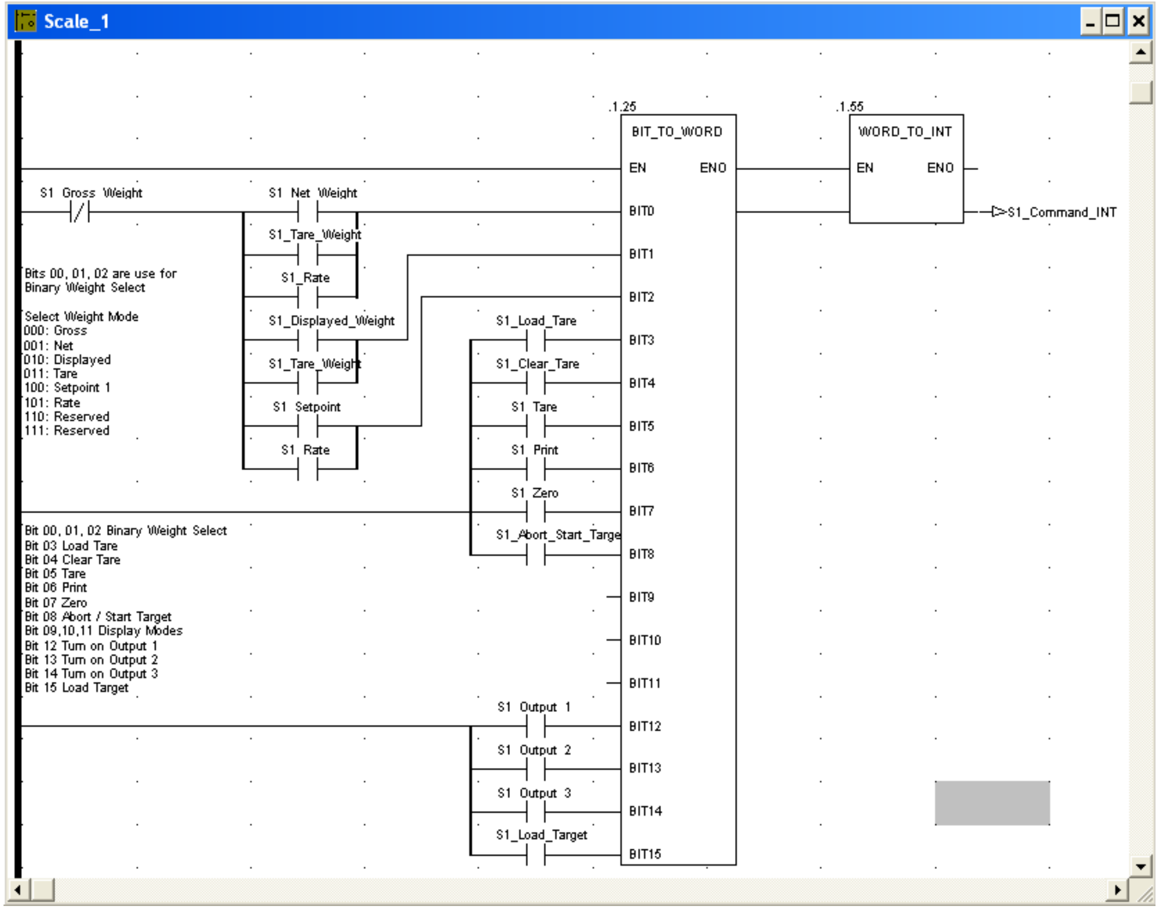


Figure 3-13: Integer-to-Word and Word-to-Bit Logic

3.9.3.2. 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 IND500x. Figure 3-14 shows an example of logic that can be used control the command word.



Bits 00, 01, 02 are use for
Binary Weight Select

Select Weight Mode
000: Gross
001: Net
010: Displayed
011: Tare
100: Setpoint 1
101: Rate
110: Reserved
111: Reserved

Bit 00, 01, 02 Binary Weight Select
Bit 03 Load Tare
Bit 04 Clear Tare
Bit 05 Tare
Bit 06 Print
Bit 07 Zero
Bit 08 Abort / Start Target
Bit 09,10,11 Display Modes
Bit 12 Turn on Output 1
Bit 13 Turn on Output 2
Bit 14 Turn on Output 3
Bit 15 Load Target

Figure 3-14: Bit to Word and Word to Integer Logic

4 PROFIBUS Option

4.1. Overview

- In this manual, the weighing sensor or terminal is referred to as **the device**. **Controller** or **control system** is used to designate an automation system such as a Programmable Logic Controller (PLC). Programmable automation controllers, distributed control systems and industrial PCs are also referred to as **controllers**.

The PROFIBUS option card enables the IND500x terminal to communicate to a PROFIBUS DP master according to DIN 19 245. It consists of an IND500x 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 IND500x.

The data mapped within the I/O block is defined as Discrete or Shared Data Variables. Discrete data can be set as Integer, Division, or Floating Point.

Discrete data is sent in groups defined as message blocks. The number of message blocks (1 to 4) is setup within the IND500x. While the format of each message block is the same, the data received and displayed within a message block is dependant 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 remote devices, all of which are viewed by it as some sort of remote I/O rack. On power up, the FIM queries each PROFIBUS remote node to determine which of the recognized types a device might be and configures itself accordingly. The PROFIBUS option appears to the FIM to be a small ET200U I/O rack.

The Siemens S5-115 series PLC also interfaces to the PROFIBUS using an I/O processor, an IM-308. 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.

Figure 4-1 shows the PROFIBUS option board

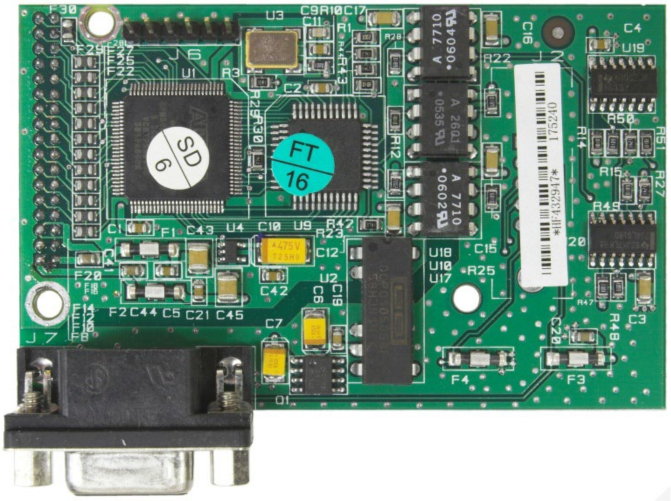


Figure 4-1: PROFIBUS Kit Option Board

4.1.1. 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 IND500x terminal supports the PROFIBUS-DP which is designed for high-speed data transfer at the sensor actuator level. (DP means Distributed Peripherals.) At this level, controllers such as PLCs exchange data via a fast serial link with their distributed peripherals. The data exchange with these distributed devices is mainly cyclic. The central controller (master) reads the input information from the remote devices 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:

4.1.1.1. 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 extendible with repeaters
- 12 megabaud maximum rate

4.1.1.2. Medium Access

Hybrid medium-access protocol according to DIN 19 245 Part 1

- Mono-Master or Multi-Master systems supported
- Device (terminal) and controllers, max 126 stations possible

4.1.1.3. Communications

Peer-to-Peer (user data transfer) or Multicast (synchronization)

- Cyclic Device-Controller user data transfer and acyclic Device-Device data transfer

4.1.1.4. 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

4.1.1.5. Synchronization

Enables synchronization of the inputs and/or outputs of all DP-Remote controllers

- Sync-Mode: Outputs are synchronized
- Freeze-Mode: Inputs are synchronized

4.1.1.6. Functionality

- Cyclic user data transfer between DP-Master(s) and DP-Remote controller(s)
- Activation or deactivation of individual DP-Remote controller(s)
- Checking of the configuration of the DP-Remote controller(s)
- Powerful diagnosis mechanisms, three hierarchical levels of the diagnosis
- Synchronization of inputs and/or outputs
- Address assignment for the DP-Remote device s over the bus
- Configuration of the DP-Master (DPM1) over the bus
- Maximum 246 byte input and output data per DP-Remote controller, typical 32 byte

4.1.1.7. Security and Protection Mechanisms

- All messages are transmitted with Hamming Distance HD=4
- Watch-Dog Timer at the DP-Remote controller
- Access protection for the inputs/outputs at the DP-Remote controllers
- Data transfer monitoring with configurable timer interval at the DP-Master (DPM1)

4.1.1.8. 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-Remote controller for example, Input/Output device with binary or analog inputs/outputs, drives

4.1.1.9. Cabling and Installation

- Coupling or uncoupling of stations without affecting other stations

- Proven and easy to handle two conductor transmission technique

4.1.2. Node/Rack Address

Each IND500x PROFIBUS option card represents one physical node. The node address is chosen by the system designer and then programmed into the IND500x and PLC. The IND500x's node address is programmed in Setup at **Communication > PLC**. 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 IND500x's PROFIBUS GSD-type files.

The IND500x setup allows selection of the logical rack (node) address, data format (Integer/Floating Point/Divisions), the number of message slots assigned to the node, and the option of sending and receiving Shared Data. The number of input and output words required and the mapping of the I/O data is dependent on these selections.

The IND500x PROFIBUS GSD has a block of I/O defined for each of the 16 possible IND500x PROFIBUS combinations. The IND500x terminal will determine the number of input and output words needed for the number of configured message slots and chosen data format. The PLC must be configured for the same amount of space.

4.1.3. Data Formats

The terminal's PROFIBUS option card has two types of data exchanges: discrete data and shared data. The locations for each of these types of data are predefined by the IND500x.

Each message slot selected to pass data through the terminal's PROFIBUS option has its own assigned input and output words for continuous information to and from the PLC. Shared data access is only available when the Setup/ Communications/PLC/PROFIBUS Share Data option is Enabled. This data is used to pass information that cannot be sent in the discrete data because of size or process speed limitations. It uses additional input and output word space. The length of shared data value and data type is dependent on the type of shared data field requested. In no case does it exceed 10 words (20 bytes).

4.2. Data Integrity

The terminal has specific bits to allow the PLC to confirm that the data was received without interrupt, 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 in Appendix A and B for specific information regarding the Data OK, Update in Progress, and Data Integrity bits and their usage.

4.3. Discrete Data

There are three formats of discrete data available with the PROFIBUS option card: integer, division, and floating point.

- **Integer** – (default) Reports scale weight as a signed 16 bit integer.

- **Divisions** – Reports scale weight in display divisions. 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.

Appendix A and B provide detailed information on data formats.

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 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.

4.3.1. Examples

250 x .01 scale					**50,000 x 10 scale**				
IND500x Displays:	0	2.00	51.67	250.00	IND500x Displays:	0	200	5160	50000
Format sent:					Format sent:				
Integer	0	200	5167	25000	Integer	0	200	5160	–(xxxxx)
Division	0	200	5167	25000	Division	0	20	516	5000
Floating Point	0	2.00	51.67	250.00	Floating Point	0	200	5160	50000

Any of the formats could be used in this case.

The integer format could not be used because it would send a negative or invalid value once the weight exceeded 32,760.

150 x .001 scale				
IND500x Displays:	0	2.100	51.607	150.000
Format sent:				
Integer	0	2100	–(xxxxx)	–(xxxxx)

150 x .001 scale				
Division	0	2100	–(xxxxx)	–(xxxxx)
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.

Please refer to Appendix A and Appendix B for each format's detailed description of data available to determine which is most suitable.

4.3.2. Byte Order

The byte order parameter sets the order in which the data bytes and words will be presented in the PLC data format. Available Byte Orders are:

- Word Swap - (default) Makes the data format compatible with RSLogix 5000 processors.
- Byte Swap - Makes the data format compatible with S7 Profibus.
- Standard - Makes the data format compatible with PLC 5.
- Double Word Swap - Makes the data format compatible with the Modicon Quantum PLC.

Table 4-1 provides examples of the various byte ordering.

Table 4-1: PLC Data Byte Ordering

		Word Swap	Byte Swap	Double Word Swap	Standard
Terminal Weight Value		1355			
PLC		**15 Bit # 0**	**15 Bit # 0**	**15 Bit # 0**	**15 Bit # 0**
Integer	Weight value word	0x054B Hex	0x4B05 Hex	0x4B05 Hex	0x054B Hex
Floating Point	1st Weight value word	0x6000 Hex	0xA944 Hex	0x0060 Hex	0x44A9 Hex
	2nd Weight value word	0x44A9 Hex	0x0060 Hex	0xA944 Hex	0x6000 Hex

4.4. Floating Point

4.4.1. Operational Overview

For a general account of Floating Point operation, please refer to Appendix B, Floating Point Format.

4.4.2. 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.

There are two data integrity bits that the IND500x uses 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 IND500x is freely sending weight updates. If the PLC program detects this condition, it should send a new command to the IND500x.

4.4.3. Floating Point Numbers

The Simatic TI505 PLCs support the IEEE Standard floating point numbers. According the Simatic TI505 Programming Reference Manual real numbers are stored in the single-precision 32-bit format, according to ANSI/IEEE Standard 754-1985, in the range 5.42101070 E-20 to 9.22337177 E18.

Siemens S5 PLCs do not support inherently the IEEE-format floating point numbers. S5 PLCs do support floating point numbers in their own unique format. You can implement a software "function block" in the S5 PLC that converts between the S5 floating point numbers and the IEEE Standard floating point numbers.

The Siemens S7 PLCs support the IEEE Standard floating point numbers.

4.5. Shared Data

4.5.1. Operational Overview

PROFIBUS PLCs can access the terminal's Shared Data. Since the PROFIBUS communications 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, as there is for Allen-Bradley and its block transfer. PROFIBUS PLCs can read IND500x Shared Data variables and write new values to IND500x Shared Data variables. 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 IND500x 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.

4.5.2. 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)

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.

4.5.3. 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 will perform 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″.

Refer to the IND500x Shared Data Reference for a complete listing of Shared Data Fields.

4.6. Discrete Data I/O Space Usage Comparison

The following tables show a comparison of the integer, division, floating point, and shared data formats' input and output data usage.

Table 4-2 shows a comparison between the integer data formats and the floating point format of the input data. The input data is from the IND500x terminal to the PLC, with node configured beginning at address "0" and data format configured for four Message Slots.

Table 4-2: Input Data Comparison

Address Word #	Integer, Division	Floating Point
IW:0 or WX:0	1st Slot (weight)	1st Slot command response
IW:1 or WX:1	1st Slot (status)	1st Slot floating point
IW:2 or WX:2	2nd Slot (weight)	Value
IW:3 or WX:3	2nd Slot (status)	1st Slot status
IW:4 or WX:4	3rd Slot (weight)	2nd Slot command response*
IW:5 or WX:5	3rd Slot (status)	2nd Slot floating point*
IW:6 or WX:6	4th Slot (weight)	Value
IW:7 or WX:7	4th Slot (status)	2nd Scale status*
IW:8 or WX:8	Null	3rd Slot command response
IW:9 or WX:9	Shared Data Access Status	3rd Slot floating point
IW:10 or WX:10	Shared Data Read Field Value**	Value
IW:11 or WX:11	Shared Data Read Field Value**	3rd Slot status
IW:12 or WX:12	Shared Data Read Field Value**	4th Slot command response
IW:13 or WX:13	Shared Data Read Field Value**	4th Slot floating point
IW:14 or WX:14	Shared Data Read Field Value**	Value
IW:15 or WX:15	Shared Data Read Field Value**	4th Slot status
IW:16 or WX:16	Shared Data Read Field Value**	Shared Data Access Status
IW:17 or WX:17	Shared Data Read Field Value**	Shared Data Read Field Value**
IW:18 or WX:18	Shared Data Read Field Value**	Shared Data Read Field Value**
IW19 or WX:19	Shared Data Read Field Value**	Shared Data Read Field Value**
IW:20 or WX:20		Shared Data Read Field Value**
~	▨	~
IW:26 or WX:26	▨	Shared Data Read Field Value**

** The length of shared data value is dependent on the type of shared data field requested. In no case does it exceed 10 words (20 bytes).

Table 4-3 shows a comparison between the integer data formats and the floating point format of the output data from the PLC to the IND500x terminal, with node configured beginning at address 0 and data format configured for four Message Slots.

Table 4-3: Output Data Comparison

Address Word #	Integer or Division	Floating Point
QW:0 or WY:0	1st Slot (load value)	Reserved
QW:1 or WY:1	1st Slot (command)	1st Slot command
QW:2 or WY:2	2nd Slot (load value)	1st Slot Floating point
QW:3 or WY:3	2nd Slot (command)	load value
QW:4 or WY:4	3rd Slot (load value)	2nd Slot command*
QW:5 or WY:5	3rd Slot (command)	2nd Slot Floating point
QW:6 or WY:6	4th Slot (load value)	load value*
QW:7 or WY:7	4th Slot (command)	3rd Slot command
QW:8 or WY:8	Shared Data Command ('1' = Read,'2' = Write)	3rd Slot Floating point
QW:9 or WY:9	Null	load value
QW:10 or WY:10	Shared Data Variable Name First two characters of SDV Name ex: 'wt' of 'wt0101'	4th Slot command
QW:11 or WY:11	Shared Data Variable Name Middle two characters of SDV Name ex: '01' of 'wt0101'	4th Slot Floating point
QW:12 or WY:12	Shared Data Variable Name Last two characters of SDV Name ex: '01' of 'wt0101'	load value
QW:13 or WY:13	Shared Data Write Value**	Shared Data Command ('1' = Read,'2' = Write)
QW:14 or WY:14	Shared Data Write Value**	Null
QW:15 or WY:15	Shared Data Write Value**	Shared Data Variable Name First two characters of SDV Name ex: 'wt' of 'wt0101'
QW:16 or WY:16	Shared Data Write Value**	Shared Data Variable Name Middle two characters of SDV Name ex: '01' of 'wt0101'
QW:17 or WY:17	Shared Data Write Value**	Shared Data Variable Name Last two characters of SDV Name ex: '01' of 'wt0101'
QW:18 or WY:18	Shared Data Write Value**	Shared Data Write Value**
QW:19 or WY:19	Shared Data Write Value**	Shared Data Write Value**
QW:20 or WY:20	Shared Data Write Value**	Shared Data Write Value**

Address Word #	Integer or Division	Floating Point
QW:21 or WY:21	Shared Data Write Value**	Shared Data Write Value**
QW:22 or WY:22	Shared Data Write Value**	Shared Data Write Value**
		Shared Data Write Value**
~		~
QW:27 or WY:27		Shared Data Write Value**

** The length of shared data value is dependent on the type of shared data field requested. In no case does it exceed 10 words (20 bytes).

4.7. IND500x PROFIBUS Message Mapping

4.7.1. Division/Integer–Shared Data Disabled

Message Slots = 1	Total Size = 2 Words
Message Slots = 2	Total Size = 4 Words
Message Slots = 3	Total Size = 6 Words
Message Slots = 4	Total Size = 8 Words

4.7.1.1. Request (PLC to IND500x)

Word 0 – Word 1: Slot 1 (1st Message Slot)

Word 2 – Word 3: Slot 2 (2nd Message Slot)

Word 4 – Word 5: Slot 3 (3rd Message Slot)

Word 6 – Word 7: Slot 4 (4th Message Slot)

4.7.1.2. Response (IND500x to PLC)

Word 0 – Word 1: Slot 1 (1st Message Slot)

Word 2 – Word 3: Slot 2 (2ndt Message Slot)

Word 4 – Word 5: Slot 3 (3rdt Message Slot)

Word 6 – Word 7: Slot 4 (4th Message Slot)

4.7.2. Division/Integer–Shared Data Enabled

Message Slot = 1	total Size = 17 Words

4.7.2.1. Request (PLC to IND500x)

Word 0 – Word 1: Message Slot 1

Word 2 Request Command: 1 - Read SDV / 2 - Write SDV

Word 3: NULL

Word 4 – Word 6: SDV name : example wt0101

Word 7 – Word 16: SDV write value

4.7.2.2. Response (IND500x to PLC)

Word 0 – Word 1: Message Slot 1

Word 2: NULL

Word 3: SD access status

Word 4 – Word 13: SD read value

4.7.3. Division/Integer–Shared Data Enabled

Message Slot = 2 Total Size = 19 words

4.7.3.1. Request (PLC to IND500x)

Word 0 – Word 1: Message Slot 1

Word 2 – Word 3: Message Slot 2

Word 4 Request Command: 1 - Read SDV / 2 - Write SDV

Word 5: NULL

Word 6 – Word 8: SDV name : example wt0101

Word 9 – Word 18: SDV write value

4.7.3.2. Response (IND500x to PLC)

Word 0 – Word 1: Slot 1

Word 2 – Word 3: Slot 2

Word 4: NULL

Word 5: SD access status

Word 6 – Word 15: SD read value

4.7.4. Division/Integer–Shared Data Enabled

Message Slot = 3 Total Size = 21 Words

4.7.4.1. Request (PLC to IND500x)

Word 0 – Word 1: Message Slot 1

Word 2 – Word 3: Message Slot 2

Word 4 – Word 5: Message Slot 3

Word 6 Request Command: 1 - Read SDV / 2 - Write SDV

Word 7: NULL

Word 8 – Word 10: SDV name : example wt0101

Word 11 – Word 20: SDV write value

4.7.4.2. Response (IND500x to PLC)

Word 0 – Word 1: Message Slot 1

Word 2 – Word 3: Message Slot 2

Word 4 – Word 5: Message Slot 3

Word 6: NULL

Word 7: SD access status

Word 8 – word 17: SD read value

4.7.5. Division/Integer–Shared Data Enabled

Message Slot = 4	Total Size = 23 Words

4.7.5.1. Request (PLC to IND500x)

Word 0 – Word 1: Message Slot 1

Word 2 – Word 3: Message Slot 2

Word 4 – Word 5: Message Slot 3

Word 6 – Word 7: Message Slot 4

Word 8 Request Command: 1 - Read SDV / 2 - Write SDV

Word 9: NULL

Word 10 – Word 12: SDV name : example wt0101

Word 13 – Word 22: SDV write value

4.7.5.2. Response (IND500x to PLC)

Word 0 – Word 1: Slot 1

Word 2 – Word 3: Slot 2

Word 4 – Word 5: Slot 3

Word 6 – Word 7: Slot 4

Word 8: NULL

Word 9: SD access status

Word 10 – word 19 : SD read value

4.7.6. Floating Point Shared Data Disabled

Message Slots = 1	Total Size = 4
Message Slots = 2	Total Size = 8
Message Slots = 3	Total Size = 12
Message Slots = 4	Total Size = 16

4.7.6.1. Request (PLC to IND500x)

Word 0: Reserved

Word 1 – Word 3: Message Slots 1

Word 4 – Word 6: Message Slots 2

Word 7 – Word 9: Message Slots 3

Word 10 – Word 12: Message Slots 4

4.7.6.2. Response (IND500x to PLC)

Word 0 – Word 3: Slot 1

Word 4 – Word 7: Slot 2

Word 8 – Word 11: Slot 3

Word 12 – Word 15: Slot 4

4.7.7. Floating Point-Share Data Enabled

Message Slots = 1 Total Size = 19

4.7.7.1. Request (PLC to IND500x)

Word 0: Reserved

Word 1 – Word 3: Message Slot 1

Word 4: Request Command : 1 - Read SDV / 2 - Write SDV

Word 5: NULL

Word 6 – Word 8: SDV name : example wt0101

Word 9 – Word 18: SDV write value

4.7.7.2. Response (IND500x to PLC)

Word 0 – Word 3: Message Slot 1

Word 4: SD access status

Word 5 – Word 14: SD read value

4.7.7.3. Floating Point-Share Data Enabled

Message Slots = 2 Total Size = 22

4.7.7.4. Request (PLC to IND500x)

Word 0: Reserved

Word 1 – Word 3: Message Slot 1

Word 4 – Word 6: Message Slot 2

Word 7: Request Command: 1 - Read SDV / 2 - Write SDV

Word 8: NULL

Word 9 – Word 11: SDV name : example wt0101

Word 12 – Word 21: SDV write value

4.7.7.5. Response (IND500x to PLC)

Word 0 – Word 3: Message Slot 1

Word 4 – Word 7: Message Slot 2

Word 8: SD access status

Word 9 – Word 18: SD read value

4.7.8. Floating Point-Share Data Enabled

Message Slots = 3 Total Size = 25

4.7.8.1. Request (PLC to IND500x)

Word 0: Reserved

Word 1 – Word 3: Message Slot 1

Word 4 – Word 6: Message Slot 2

Word 7 – Word 9: Message Slot 3

Word 10: Request Command: 1 - Read SDV / 2 - Write SDV

Word 11: NULL

Word 12 – Word 14: SDV name : example wt0101

Word 15 – Word 24: SDV write value

4.7.8.2. Response (IND500x to PLC)

Word 0 – Word 3: Message Slot 1

Word 4 – Word 7: Message Slot 2

Word 8 – Word 11: Message Slot 3

Word 12: SD access status

Word 13 – Word 22: SD read value

4.7.9. Floating Point-Share Data Enabled

Message Slots= 4 Total Size = 28

4.7.9.1. Request (PLC to IND500x)

Word 0: Reserved

Word 1 – Word 3: Message Slot 1

Word 4 – Word 6: Message Slot 2

Word 7 – Word 9: Message Slot 3

Word 10 – Word 12: Message Slot 4

Word 13 : Request Command: 1 - Read SDV / 2 - Write SDV

Word 14: NULL

Word 15 – Word 17: SDV name: example wt0101

Word 18 – Word 27: SDV write value

4.7.9.2. Response (IND500x to PLC)

Word 0 – Word 3: Message Slot 1

Word 4 – Word 7: Message Slot 2

Word 8 – Word 11: Message Slot 3

Word 12 – Word 15: Message Slot 4

Word 16: SD access status

Word 17 – Word 26: SD read value

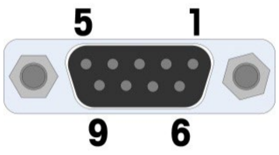
4.8. Controlling Discrete I/O Using a PLC Interface

The IND500x 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 IND500x discrete I/O updates are synchronized with the A/D 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 IND500x terminal setup in order to be controlled by the PLC.

4.9. Hardware Setup

4.9.1. Wiring

The IND500x terminal's PROFIBUS option card has a DB-9 connector to connect to the PROFIBUS network interface (Figure 4-2). Cable distance, type, and termination are specified by PROFIBUS. (Refer to the PLC documentation for cable design guidelines for the various PLCs.)



PIN	SIGNAL
1	Not used
2	Not used
3	RxD/TxD +
4	RTS
5	GND bus
6	+5V bus
7	Not used
8	RxD/TxD -
9	Not used

NOTES:

1. **USE MATING CONNECTORS AND CABLE RECOMMENDED FOR PROFIBUS CONNECTIONS.**
2. **REFER TO PROFIBUS INTERNATIONAL DOCUMENTATION FOR OTHER CONSIDERATIONS.**

Figure 4-2: PROFIBUS Option Card DB-9 Connector

- The IND500x harsh unit requires a right angle connector Siemens part number 6ES7 972-0BA41-0XA0. The panel mount can use the right angle or straight connector METTLER TOLEDO part number 64054361.

4.10. Software Setup

The IND500x terminal automatically detects the presence of a PROFIBUS option card if one is installed, and adds the setup parameters to the options block. To configure the terminal for PROFIBUS, enter Setup and advance to the **Communications > PLC > PROFIBUS** sub-block (Figure 4-3).

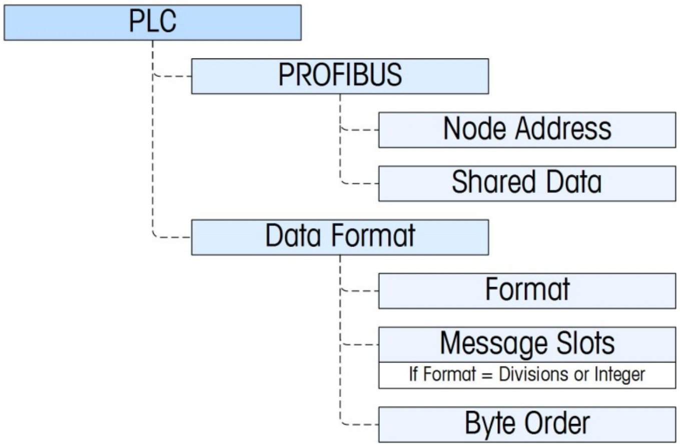


Figure 4-3: PROFIBUS Setup Block

4.10.1. PROFIBUS Setup Sub-Block

4.10.1.1. Profibus setup

The Profibus Setup block at **Communication > PLC > Profibus** lets you specify how the PROFIBUS interface is used. Enter a unique Node Address of 0–125. Set Shared Data as Enabled or Disabled

4.10.1.2. Data Format setup

4.10.1.2.1. Format

At **Communication > PLC > Data Format**, select the Format from a drop-down list. Select Divisions, Integer (default) or Floating Point.

4.10.1.2.2. Byte Order

Available selections are Standard, Byte Swap, Word Swap (default), and Double Word Swap. See Table 4-1 for definitions.

4.10.1.2.3. Message Slots

Select 1, 2, 3 or 4 slots.

4.10.2. PROFIBUS GSD or Type Files

There are thirteen configurations contained in the PROFIBUS GSD or type file for the IND500x terminal's different combinations of data formats. The length of the messages is different for each of the data formats, but the length of the input and output messages are the same within each format. The IND500x supports the message types shown in Table 4-4

- The PROFIBUS GSD file for the IND500x is included on the IND500x documentation CD (P/N 30725989).

Table 4-4: Message Types Supported by the IND500x

Configuration		Functionality
I/O 2 Wrd	(2 words in/ 2 words out)	One message slot in integer/division
I/O 4 Wrd	(4 words in/ 4 words out)	Two message slots in integer/division
I/O 6 Wrd	(6 words in/ 6 words out)	Three message slots in integer/division
I/O 8 Wrd	(8 words in/ 8 words out)	Four message slots in integer/division
I/O 17 Wrd	(17 words in/ 17 words out)	One message slot in integer/division and Shared Data Variable
I/O 19 Wrd	(19 words in/ 19 words out)	Two message slots in integer/division and Shared Data Variable
I/O 21 Wrd	(21 words in/ 21 words out)	Three message slots in integer/division and Shared Data Variable
I/O 23 Wrd	(23 words in/ 23 words out)	Four message slots in integer/division v and Shared Data Variable
I/O 4 Wrd	(4 words in/ 4 words out)	One message slot in float ing point
I/O 8 Wrd	(8 words in/ 8 words out)	Two message slots in float ing point
I/O 12 Wrd	(12 words in/ 12 words out)	Three message slots in float ing point
I/O 16 Wrd	(16 words in/ 16 words out)	Four message slots in float ing point
I/O 19 Wrd	(19 words in/ 19 words out)	One message slot in float ing point and Shared Data Variable
I/O 22 Wrd	(22 words in/ 22 words out)	Two message slots in float ing point and Shared Data Variable
I/O 25 Wrd	(25 words in/ 25 words out)	Three message slots in float ing point and Shared Data Variable
I/O 28 Wrd	(28 words in/ 28 words out)	Four message slots in float ing point and Shared Data Variable

4.11. PROFIBUS Option Kit Part Numbers

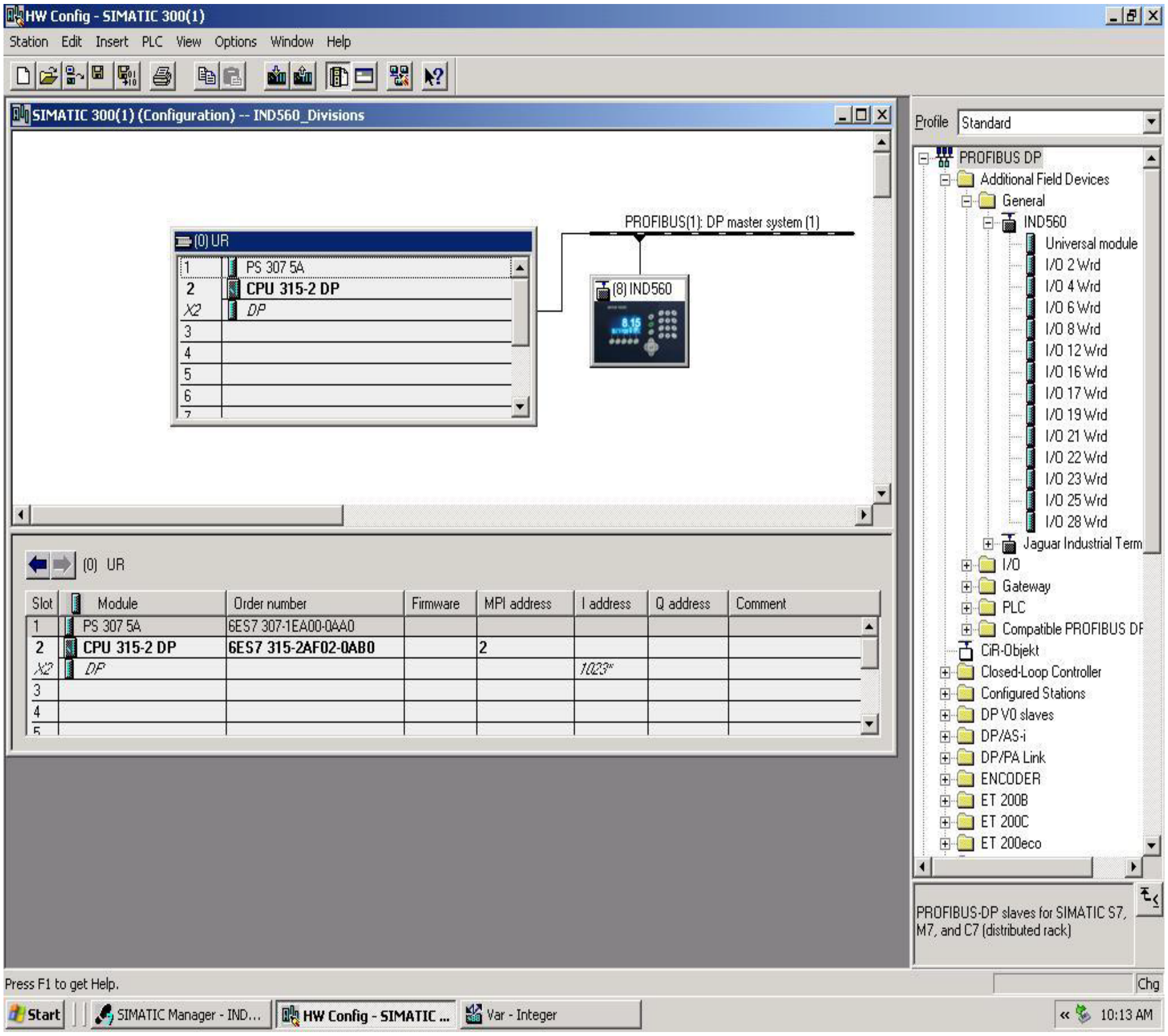
There are two PROFIBUS options. CIMF part number 71209096 is vertically mounted and designed to work with the harsh IND500x and CIMF part number 71209097 is horizontally mounted and designed to work with the panel mount IND500x. Refer to 错误!未找到引用源。 and Figure 4-1 for images of these two different boards. There are no associated spare parts with the PROFIBUS option kit. Table 4-5 shows what each kit contains.

Table 4-5: PROFIBUS Option Kit

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

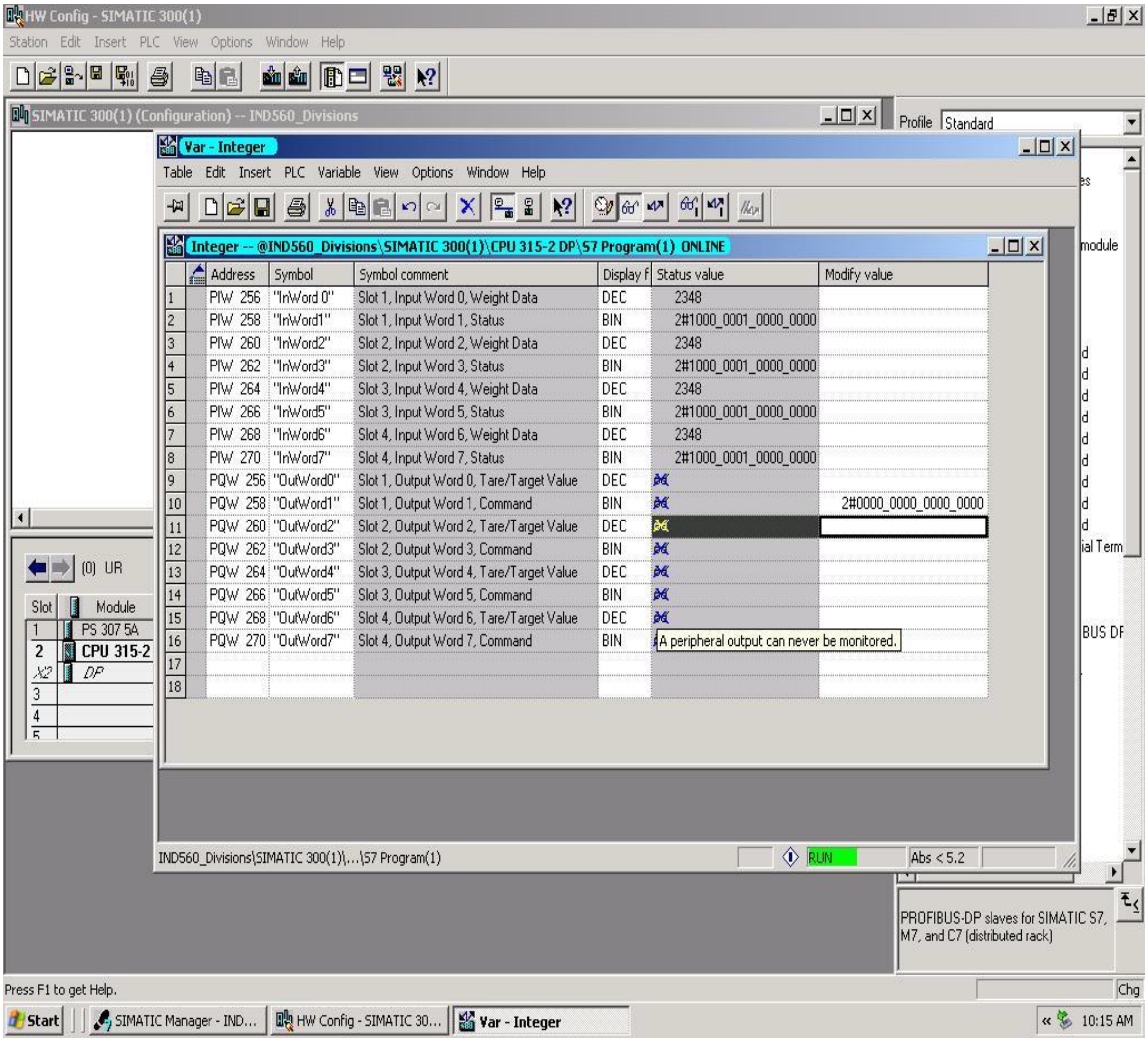
4.12. Interfacing Examples

Figure 4-4 and Figure 4-5 show sample screens of hardware setup and I/O monitoring of the IND500x in the Siemens Step 7 software. The documentation CD, part number 30725989, contains complete versions of these examples.



Slot	Module	Order number	Firmware	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	**CPU 315-2 DP**	**6ES7 315-2AF02-0AB0**		2			
X2	*DP*				*1023**		
3							
4							
5							

Figure 4-4: Hardware Setup



HW Config - SIMATIC 300(1)
Station Edit Insert PLC View Options Window Help
SIMATIC 300(1) (Configuration) -- IND560_Divisions
Profile Standard
Var - Integer
Table Edit Insert PLC Variable View Options Window Help
(0) UR
Start
SIMATIC Manager - IND...
HW Config - SIMATIC 30...
Var - Integer
10:15 AM
Chg

Integer -- @IND560_Divisions\SIMATIC 300(1)\CPU 315-2 DP\S7 Program(1) ONLINE

	Address	Symbol	Symbol comment	Display f	Status value	Modify value
1	PIW 256	"InWord 0"	Slot 1, Input Word 0, Weight Data	DEC	2348	
2	PIW 258	"InWord1"	Slot 1, Input Word 1, Status	BIN	2#1000_0001_0000_0000	
3	PIW 260	"InWord2"	Slot 2, Input Word 2, Weight Data	DEC	2348	
4	PIW 262	"InWord3"	Slot 2, Input Word 3, Status	BIN	2#1000_0001_0000_0000	
5	PIW 264	"InWord4"	Slot 3, Input Word 4, Weight Data	DEC	2348	
6	PIW 266	"InWord5"	Slot 3, Input Word 5, Status	BIN	2#1000_0001_0000_0000	
7	PIW 268	"InWord6"	Slot 4, Input Word 6, Weight Data	DEC	2348	
8	PIW 270	"InWord7"	Slot 4, Input Word 7, Status	BIN	2#1000_0001_0000_0000	
9	PQW 256	"OutWord0"	Slot 1, Output Word 0, Tare/Target Value	DEC		
10	PQW 258	"OutWord1"	Slot 1, Output Word 1, Command	BIN		2#0000_0000_0000_0000
11	PQW 260	"OutWord2"	Slot 2, Output Word 2, Tare/Target Value	DEC		
12	PQW 262	"OutWord3"	Slot 2, Output Word 3, Command	BIN		
13	PQW 264	"OutWord4"	Slot 3, Output Word 4, Tare/Target Value	DEC		
14	PQW 266	"OutWord5"	Slot 3, Output Word 5, Command	BIN		
15	PQW 268	"OutWord6"	Slot 4, Output Word 6, Tare/Target Value	DEC		
16	PQW 270	"OutWord7"	Slot 4, Output Word 7, Command	BIN	A peripheral output can never be monitored.	
17						
18						

Slot	Module
1	PS 307 5A
2	CPU 315-2
X2	DP
3	
4	
5	

IND560_Divisions\SIMATIC 300(1)\...\S7 Program(1) RUN Abs < 5.2

PROFIBUS-DP slaves for SIMATIC S7, M7, and C7 (distributed rack)

Press F1 to get Help.

Figure 4-5: I/O Monitoring

5 PROFINET Option

5.1. 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 IND500x. PROFINET utilizes commercial, off-the-shelf Ethernet hardware (for example, switches and routers) and is fully compatible with the Ethernet TCP/IP protocol suite.

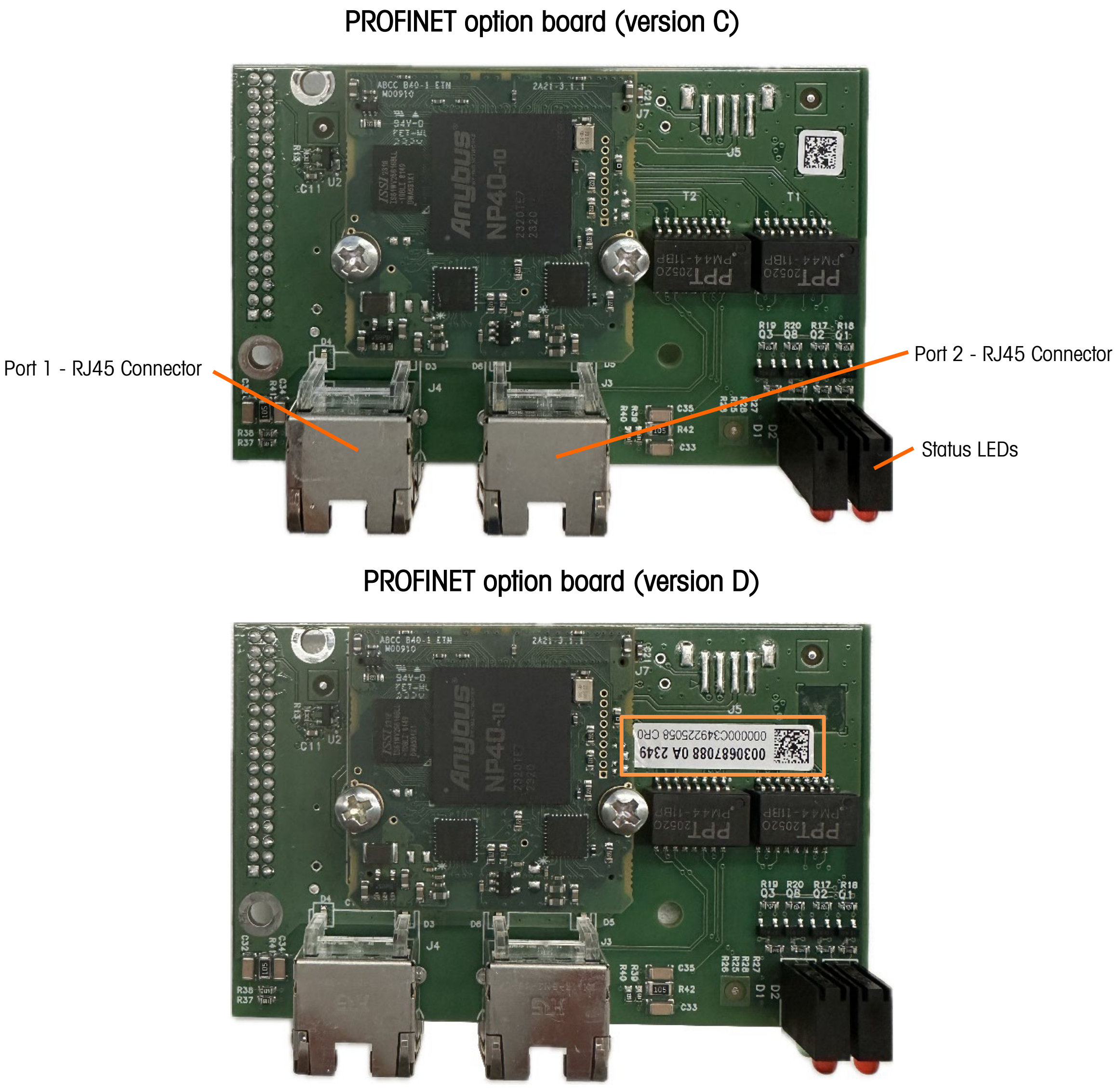
The PROFINET option enables the IND500x terminal to communicate to PROFINET enabled Programmable Logic Controllers (PLCs) through direct connection to the PROFINET network at 100 MBPS speed. This solution consists of an internal module and internal software to implement the data exchange.

5.2. PROFINET Interface

The part # of the IND500x PROFINET option kit is 30708926. Figure 5-1 shows a PROFINET module and its components.

- Note that a module was updated on the PROFINET PCB for the ACM500. No action is required if no further ACM500 with PROFINET units will be added to the DCS/PLC network. If further ACM500 with PROFINET units will be added to the network, the older PROFINET PCBs in the existing ACM500 units (which have no manufacturing date / material number printed on the PCB) will be required to be replaced with new PROFINET PCBs (manufacture date / material number printed on PCB) and GSDML file updated to at least v2.42.

The PROFINET option board (version D) comes with a label to indicate the PCBAC number, version, as well as the production year and week (e.g., 30687088 represents the PCBAC number; A represents the PCBAC version; 2349 means the PCBAC was produced in the 49th week of 2023).


PROFINET option board (version C)

PROFINET option board (version D)

Figure 5-1: PROFINET Module

5.2.1. Definition of Terms

The following terms are used in this chapter.

Table 5-1: 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.

Term	Definition
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 remote device.
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.
MRP	Media Redundancy Protocol. An Ethernet Ring Topology used with PROFINET IO to provide media redundant communications. Messages are sent out one Ethernet port of the PLC and come back in the other. If the PLC detects a media break in the ring then it reconfigures the network within 200 milliseconds so that messages will be sent out both ports of the PLC. Requires PLC's and devices that are MRP enabled. Any switches on the network must also be MRP enabled. Non-MRP enabled devices may be connected to the loop by using MRP enabled switches.
Submodule	Hardware or logical component of a module
PDEV	**P**hysical **DEV**ice. 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 1ms, 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.

5.2.2. Communications

The IND500x terminal uses component parts to ensure complete compatibility with the Siemens PROFINET network. An IND500x terminal is recognized as a generic PROFINET device by the PLC.

5.2.3. IP Address

Each PROFINET option represents one physical IP Address. This address can be chosen by the system designer, and then programmed into the IND500x terminal and PLC, or the address can be automatically assigned by the PLC. Each IND500x within a system must have a unique PROFINET IP Address.

The IND500x terminal’s PROFINET IP address is programmed in the terminal's setup menu at **Communication > PLC > PROFINET**.

5.2.4. Supported Data Transfer

The PROFINET interface provides both cyclic data transfer and acyclic messaging.

5.2.5. Connection Methods

The dual ports on the PROFINET Interface module provide several possible methods for connecting the IND500x to the control Network. Those methods are described in this section. It is important that in both the Daisy Chain and MRP Redundant Loop configurations, the physical network wiring matches the network topology defined on the PLC as it relates to Port 1 and Port 2. If the wiring does not match the defined topology, errors will be reported.

5.2.5.1. Star Network

A star network consists of multiple devices attached to one or more Ethernet switches.

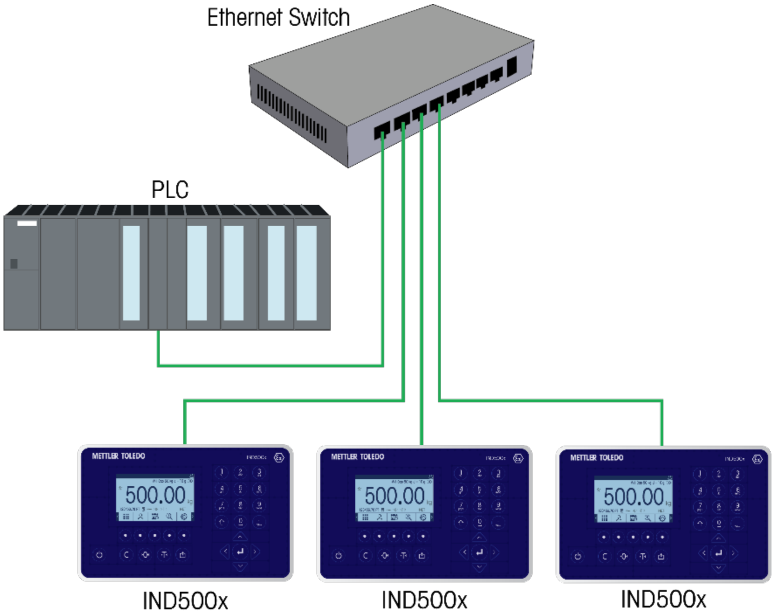


Figure 5-2: Star Network Example

5.2.5.2. Daisy Chain

A Daisy Chain network has the advantage of not requiring switches for multiple devices to be connected to the Controller. This has advantages in a cabinet or tight space where there may not be sufficient area to run individual cables all the way back to a central point such as a switch.

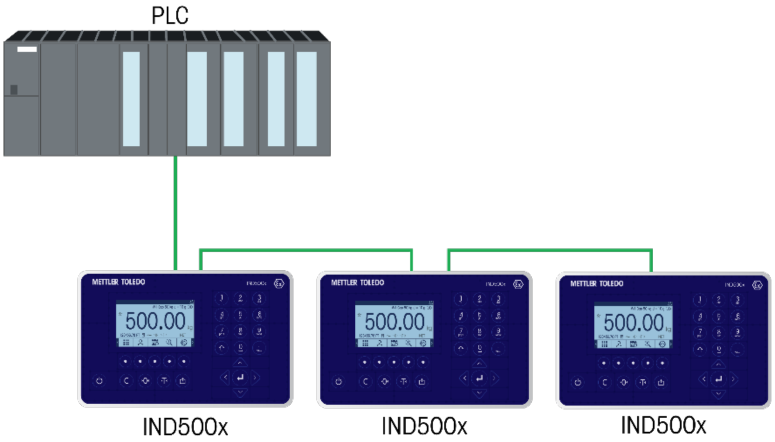


Figure 5-3: Daisy Chain Example

5.2.5.3. MRP Redundant Loop

An MRP redundant loop is very similar to the Daisy Chain topology, where the PLC is connected on one end of the loop, and devices are daisy chained along the loop until the loop is terminated back at the same PLC on a second Ethernet port. This provides a 'Ring' topology where messages can be routed either direction around the ring and has the advantage of not requiring any switches as long as the PLC and the devices are MRP capable. If a break in the Ring occurs, the PLC will quickly detect it by noticing that messages are no longer making it back to the PLC on the opposite end of the ring that is attached to it. Under those conditions the PLC will then start transmitting the messages out both ports so that all devices on the ring can still get the messages. The result is a network of daisy chains out each port that continue to function regardless of the break. PROFINET MRP is designed to make the break detection and switch over in less than 200 milliseconds. NOTE that your process must be able to tolerate a loss of communications for up to 200 ms.

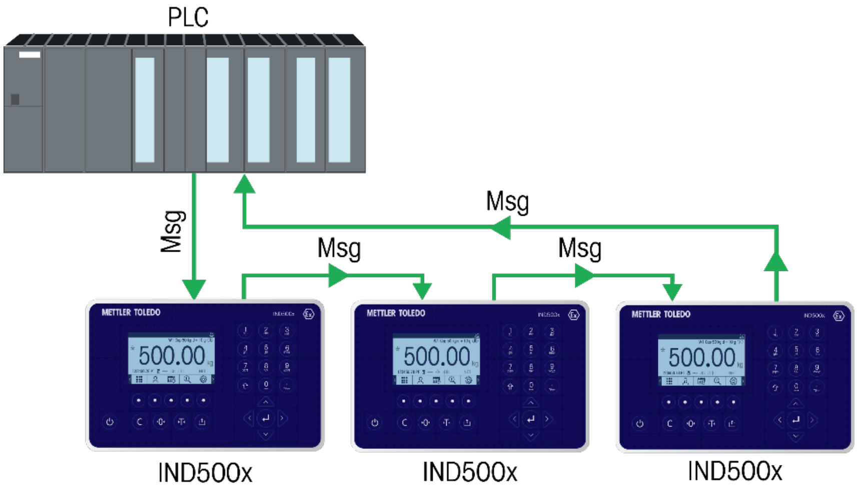


Figure 5-4: Intact MRP Ring

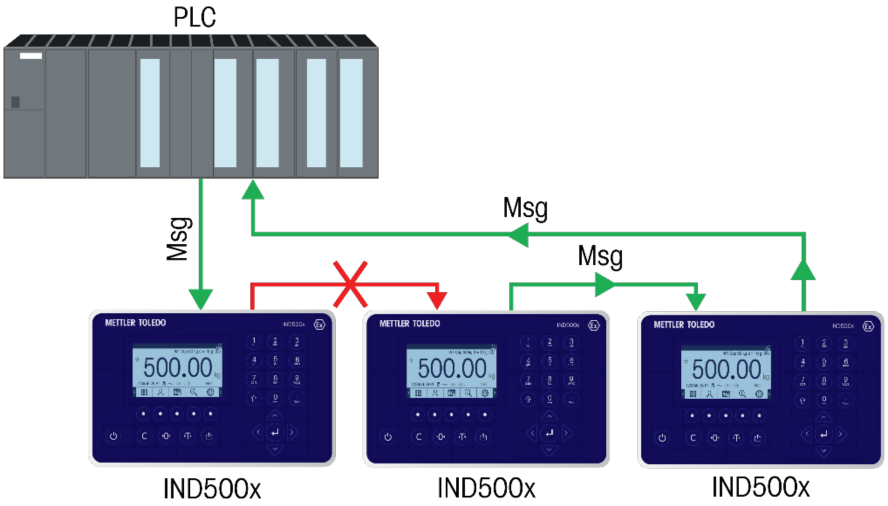


Figure 5-5: MRP Ring with Break

- Note that messages still reach all devices because the network is self-healing.

5.3. S2 System Redundancy

The IND500x PROFINET option supports communication with both primary and backup controllers, using a single connection. If the primary controller fails, communication is seamlessly switched to the backup controller.

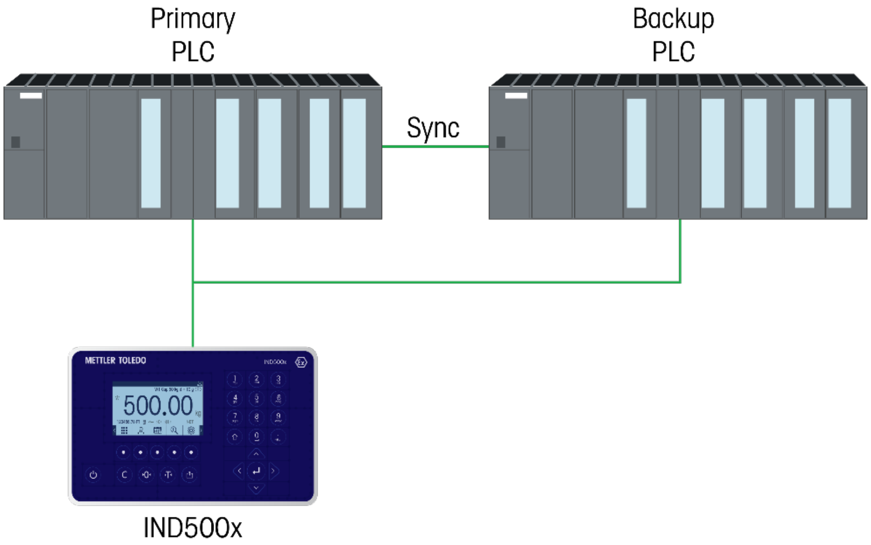


Figure 5-6: S2 System Redundancy

5.4. Software Setup

The IND500x terminal automatically detects the presence of a PROFINET option board when installed and makes the PROFINET configuration parameters available within the setup menu. To configure the terminal for PROFINET communication, enter setup and navigate to the **Communication > PLC > PROFINET** sub-block (Figure 5-7).

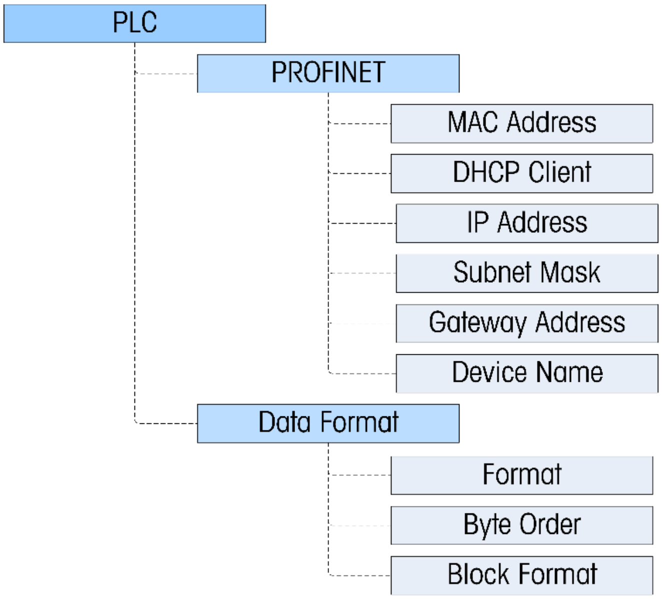


Figure 5-7: PROFINET Setup Block

5.4.1. PROFINET and Data Format Setup Blocks

5.4.1.1. PROFINET setup

The PROFINET setup block at **Communication > PLC > PROFINET** allows the user to specify how the PROFINET interface is used.

5.4.1.1.1. MAC Address

The MAC address is displayed but cannot be modified.

5.4.1.1.2. DHCP Client

The default value is **Disabled**. This should be changed to **Enabled** if the PLC is to be configured as a DHCP server.

5.4.1.1.3. IP Address

By default, the PLC programming software can assign the IP Address, SubNet Mask, Gateway Address or Device Name fields using information received from the network.

If the user needs to set the IP Address, SubNet Mask, Gateway Address and Device Name fields in the IND500x (by disabling the DHCP Client), the PLC program should be configured to match the settings configured in the IND500x.

5.4.1.2. Data Format setup

In Setup, navigate to **Communication > PLC Interface > Data Format**. The following must be configured for PROFINET.

5.4.1.2.1. Format

The data format is SAI.

5.4.1.3. Byte Order

Available selections are Little Endian, Byte Swap, Word Swap, Big Endian, Automatic.

5.4.1.4. Block Format

Available selections are 1, 2 and 4 blocks.

5.5. Troubleshooting

If the IND500x 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 IND500x to ensure that any updated settings take effect.
- Check for IP Address conflicts. Use a Ping command from a PC to verify addresses.
- Check physical wiring and network connections.
- Confirm that the IND500x settings for data type and IP Address assignment match those in the PLC and that each IND500x on the network has a unique address.
- Confirm that the number of message slots assigned within the IND500x's setup menu match both the Input and Output assignment in the Siemen's HW Configuration Tool.
- If the communication interface in the IND500x was changed from another type (e.g. Ethernet/IP or ControlNet), a master reset of the IND500x may need to be performed.
- Replace the PROFINET interface kit if communication problems persist.

5.5.1. Status LEDs

The PROFINET interface card has a link activity LED on each of the two RJ45 connectors and four status LEDs indicators to indicate communication and fault status of the card. Figure 5-8 shows the layout of the LED module, and Figure 5-9 shows the array of the LEDs on the card. Table 5-2 and Table 5-3 explain the meaning of the indicators.

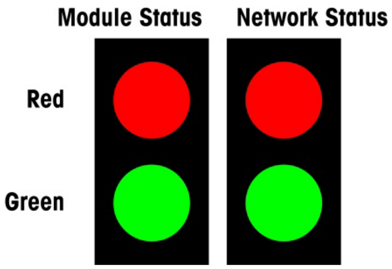


Figure 5-8: PROFINET Status Indicator LEDs

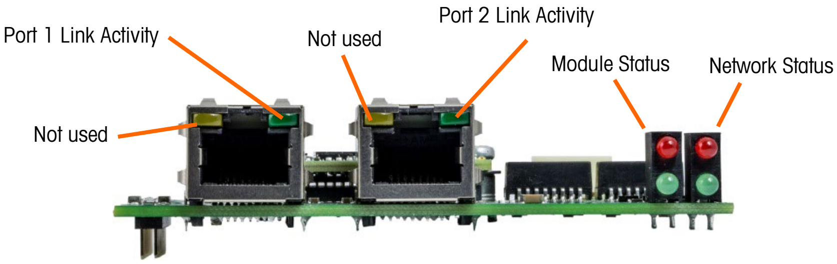


Figure 5-9: PROFINET Status Indicator LEDs

Table 5-2: Module Status LEDs

LED State	Description	Comments
Off	Not Initialized	• No power • Module in 'SETUP' or 'NW_INIT' state
Green	Normal Operation	• Module has shifted from the 'NW_INIT' state
Green, 1 flash	Diagnostic Event(s)	• Diagnostic event(s) present
Red	Exception error	• Device in state EXCEPTION
	Fatal event	• Major internal error (this indication is combined with a red network status LED)
Alternating Red/Green	Firmware update	• Do NOT power off the module. Turning the module off during this phase could cause permanent damage.

Table 5-3: Network Status LEDs

LED State	Description	Comments
Off	Offline	• No Power • No connection with IO Controller
Green	Online (RUN)	• Connection with IO Controller established
Green, 1 flash	Online (STOP)	• IO Controller in STOP state • IO data bad • IRT synchronization not finished
Green, blinking	Blink	• Used by engineering tools to identify the node on the network
Red	Fatal event	• Major internal error (this indication is combined with a red module status LED)
Red, 1 flash	Station Name error	• Station Name not set
Red, 2 flashes	IP address error	• IP address not set
Red, 3 flashes	Configuration error	• Expected Identification differs from Real Identification

A Integer and Division Data Formats

When one of these formats is selected, the IND500x will have two 16-bit words for input data and two 16-bit words for output data in each Message Slot. There can be up to four slots and the number of slots is setup at the IND500x. 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 IND500x will send specific weight data to the PLC input based on the selections the IND500x 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 logic value, and one 16-bit word for bit encoded command information.

If the indicator is setup for more than one scale slot, the "Select 1, 2, or 3" commands in write word 1, select the type of data that will be displayed in its scale slot. Also note that most commands in write word 1 are only active in the first scale slot.

Table A-1 and Table A-2 provide detailed information on the integer and division data formats. Note that the designation of "Read" or "Write" data is based on the PLC's viewpoint-"Read" data refers to the PLC's input data and "Write" data refers to the PLC's output data.

If the indicator is setup for more than one scale slot, the "Select 1, 2, or 3" commands in write word 1, select the type of data that will be displayed in its scale slot. Also note that most commands in write word 1 are only active in the first scale slot.

Table A-1: Discrete Read Integer or Division – IND500x > PLC, per Message Slot

Bit number	First Word	Second Word
0	See Note 1	Target 1[2]
1		Target 2[2]
2		Target 2[2]
3		Comparator 5[3]
4		Comparator 4[3]
5		Comparator 3[3]
6		Comparator 2[3]
7		Comparator 1[3]
8		Enter Key[4]
9		Input 1[5]
10		Input 2[5]

Bit number	First Word	Second Word
11		Input 3[5]
12		Motion[6]
13		Net Mode[7]
14		Update in Process[8]
15		Data OK[9]

A.1.1.1.1. Notes for Table A-1

1 The first word is a 16 bit, signed integer that may represent the indicator's gross weight, net weight, displayed weight, tare weight, or rate. The **bits 0** to **2** in the PLC 2nd output word designate the type of data that is being is being sent by the indicator.

2 The second word **bits 0, 1** and **2** indicate the state of the target comparison logic. When in the material transfer mode; **bit 0** is Feed, **bit 1** is Fast Feed and **bit 2** is Tolerance Ok (within range). When in the over/under mode; **bit 0** is Under, **bit 1** is OK and **bit 2** is Over. An 'ON' condition is indicated by the bit being set to '1'; an 'OFF' condition is indicated by the bit being set to '0'.

3 The second word Comparator bits indicate the state of the associated comparator logic; when the bit is set to '1' the comparator state is 'ON'; when the bit is set to '0' the comparator state is 'OFF'. The setup of each comparator will determine when the state is 'ON' or 'OFF'.

4 The second word **bit 8** is set to '1' when the Enter Key has been pressed on the indicator keypad. The bit can be reset to '0' by changing the state of the second output word **bits 9, 10** and **11**.

5. The second word **bits 9, 10**, and **11** indicate the state of the associated hardware input internal to the indicator; these are 0.1.1, 0.1.2 and 0.1.3. When the input is 'ON" the associated bit is set to '1'.

6 The second word **bit 12**; The motion bit is set to '1' when the scale is in motion (unstable).

7 The second word **bit 13**; The net mode bit is set to '1' when scale is in the net mode (a tare has been taken).

8 The second word **bit 14** (update in process) is set to '1' when the indicator is in process of updating the data to the PLC communications adapter. The PLC should ignore all data while this bit is set to '1'.

9 The second word **bit 15**; The data ok bit is set to '1' when the indicator operating conditions are normal. The bit is set to '0' during power-up, during indicator setup, when the scale is over capacity or under zero, and when in the x10 display mode; additionally, the first word integer value is set to zero (0). This value is also set to zero (0) when in x10 mode. The PLC should continuously monitor the data ok bit in the IND500x communication and also any PLC data connection fault bit that exists in the PLC (refer to the PLC manufacturer's documentation) to determine the validity of the data in the PLC.

IND500xx NOTE: If communication between the IND500xx and the ACM500 is interrupted, the Data OK bit is not affected and will remain at its last transmitted status.

Table A-2: Discrete Write Integer or Division –PLC > IND500x, per Message Slot

Bit number	First Word	Second Word [Scale Command]
0	See Note 1	Select 1[2]
1		Select 2[2]
2		Select 3[2]
3		Load Tare Value 1st message slot only[12]
4		Clear Tare[4] 1st message slot only[12]
5		Tare[5] 1st message slot only[12]
6		Print[6] 1st message slot only[12]
7		Zero[7] 1st message slot only[12]
8		Start/Abort Target[8] 1st message slot only[12]
9		Message Display Mode[9] 1st message slot only
10		Message Display Mode[9] 1st message slot only[12]
11		Message Display Mode[9] 1st message slot only[12]
12		Output 1[10] 1st message slot only[12]
13		Output 2[10] 1st message slot only[12]
14		Output 3[10] 1st message slot only[12]
15		Load Target Value[11] 1st message slot only[12]

A.1.1.1.2. Notes for Table A-2

1 First word is a 16 bit, signed integer that represents a value to be downloaded to the indicator. The value represents a tare or target value. When using the divisions format, the data set must be in the number of divisions, not an integer weight value. A value must be loaded in this word before setting the **bits 3** or **15** in the second word. To load the target value ,first enter the value into the first word and then set bit 15 (Load Target) "On"

2 The select bits change the type of data being sent from the indicator in the first word. Use a decimal value in binary format within **bits 0, 1**, and **2** to change the data reported by the indicator. '0' = gross weight, '1' = net weight, '2' = displayed weight, '3' = tare weight, '4' = target, '5' = rate; any value above 5 will equal gross weight.

3 A transition from '0' to '1' will cause the value in the first word to be loaded into the tare register of the indicator and set the indicator into the net mode. Set this bit to '1' only after the first word has been loaded with the required value.

4 A transition from '0' to '1' will cause the indicator tare register to be set to '0' and the indicator will be set to the gross weight mode.

5 A transition from '0' to '1' will cause the weight on the scale to be used as the tare value and set the indicator to the net mode (equivalent to a tare command). Note that the scale will not tare while the scale is "In Motion". If the indicator has not tared within 3 seconds, the command must be resent. A good practice is to check for no motion –bit 12 of input word 1-"Off"

6 A transition from ′0′ to ′1′ will issue a print command.

7 A transition from ′0′ to ′1′ will cause the scale to re-zero, but only within the ranges established in scale setup.

8 A transition from ′0′ to ′1′ will cause the target logic to start. A transition from ′1′ to ′0′ will cause the target logic to abort. The use of the PLC in conjunction with the indicator console keypad and/or a remote input is not advised, as unexpected results may occur.

9 The message display mode bits will cause messages to be displayed on the indicator display above the soft key prompts; messages are limited to 20 characters. The use of the display mode bits will clear the Enter Key bit in the second word of the indicator output data. The message display mode bits cause a value to be written to shared data pd0119, which is available for use by Task Expert applications. The transition from ′0′ to a decimal value in binary form to the second word **bits 9, 10** and **11** will initiate the message events.

Setting the message display bits to a value of ′1′ will cause the characters in shared data aw0101 to be displayed and pd0119 will be set to ′1′.

Setting to ′2′ = display aw0102 and pd0119 = ′2′.

Setting to ′3′ = display aw0103 and pd0119 = ′3′.

Setting to ′4′ display aw0104 and pd0119 = ′4′.

Setting to ′5′ = display aw0105 and pd0119 = ′5′.

Setting to ′6′ = start Prompt sequence, pd0119 = ′6′ and xc0134 = ′1′.

Setting to ′7′ = display pd0118 and pd0119 = ′7′.

The message display mode bits must return to ′0′ before a new message can be displayed.

10 The output bits will cause the associated hardware output to be turned ′ON′ and ′OFF′. This is the indicator internal outputs only; 0.1.1, 0.1.2 and 0.1.3. The output bits will not override the hardware outputs being used by the indicator logic as setup within the indicator. Setting a bit to ′1′ will cause the output to turn ′ON′; setting the bit to ′0′ will cause the output to turn ′OFF′.

11 A transition from ′0′ to ′1′ will cause the value in the first word to be loaded into the target register of the indicator and will be used the next time the target logic is started. Set this bit to ′1′ only after the first word has been loaded with the required value.

12 These are bit commands to the indicator that function only in the first message slot.

B Floating Point Format

B.1. Operational Overview

The IND500x uses integer commands from the PLC to select the floating point weight input data. The IND500x 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 IND500x 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 IND500x 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 IND500x before sending another command.

The IND500x can report two types of values to the PLC: real-time and static. When the PLC requests a real-time value, the IND500x 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 IND500x acknowledges the command from the PLC once and updates the value once. The IND500x 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 IND500x can send a rotation of up to nine different real-time values. The PLC sends commands to the IND500x to add a value to the rotation. Once the rotation is established, the PLC must instruct the IND500x to begin its rotation automatically, or the PLC may control the pace of rotation by instructing the IND500x to advance to the next value. If the IND500x 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 20 Hz or 58 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 IND500x switches to the next value in the rotation order. The IND500x 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 B-5 through Table B-8 for additional information. The method of handling string and floating point data varies between PLC types. The IND500x provides floating point data in the order entered in Data Format setup.

The tables in this appendix 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 B-1: Discrete Read Floating Point – IND500x > PLC Input, per Message Slot

Bit number	1st Word Command Response, except PROFIBUS	1st Word Command Response, PROFIBUS	2nd Word FP value	3rd Word FP value	4th Word Scale Status
0	RESERVED[1]	FP Input Indicator 1[2]	See Note 4	See Note 4	Target 1[6]
1		FP Input Indicator 2[2]			Comparator 1[7]
2		FP Input Indicator 3[2]			Target 2[6]
3		FP Input Indicator 4[2]			Comparator 2[7]
4		FP Input Indicator 5[2]			Target 3[6]
5		Data integrity1[3]			Always = 1
6		Command Ack 1[4]			TE bit 1[8]
7		Command Ack 2[4]			TE bit 2[8]
8	FP Input Indicator 1[2]	RESERVED[1]			Enter Key[9]
9	FP Input Indicator 2[2]				Input 1[10]
10	FP Input Indicator 3[2]				Input 2[10]
11	FP Input Indicator 4[2]				Input 3[10]
12	FP Input Indicator 5[2]				Motion[11]
13	Data integrity1[3]				Net Mode[12]
14	Command Ack 1[4]				Data Integrity 2[3]
15	Command Ack 2[4]				Data OK[13]

Notes for Table B-1

1 For all options except PROFIBUS, bits **0** to **7** are reserved; for the PROFIBUS option, bits **8** to **15** are reserved.

2 The Floating Point Indicator bits (1st word bits 8-12) are used to determine what type of floating or other data is being sent in the second and third words. See the Floating Point Indicator Table for the information from these bits in decimal format.

3 The Data Integrity bits (1st word **bit 13** and 4th word **bit 14**) should be used to assure that communication is still valid and that data are valid. Both of these bits are set to '1' for one update from the indicator, then are set to '0' for the next update from the indicator and this change of state is on every update and is constant as long as the communications link is not disrupted.

4 The first word Command Acknowledge bits (**bits 14** and **15**) are used by the indicator to inform the PLC that a new command was received. The decimal values of these bits will rotate sequentially from 1 to 3 as long as a command other than '0' is being sent (3rd output word). The decimal value of these bits will be '0' when the 3rd output word (PLC output command word) is decimal '0'.

5 The second and third words are 32 bit, single precision floating point data. The data may represent the various scale weight data or setup configuration data. The PLC output command word determines what data will be sent.

6 The fourth word, **bits 0, 2** and **4** indicate the state of the Target comparison logic. When in the material transfer mode; **bit 0** is Feed, **bit 2** is Fast Feed and **bit 4** is Tolerance Ok (within range). When in the over/under mode; **bit 0** is Under, **bit 2** is OK and **bit** 4 is Over. An 'ON' condition is indicated by the bit being set to '1'; an 'OFF' condition is indicated by the bit being set to '0'.

7 The fourth word; Comparator bits indicate the state of the associated comparator logic; when the bit is set to '1' the comparator state is 'ON'; when it is set to '0' the comparator state is 'OFF'. The setup on each comparator will determine when the state is 'ON' or 'OFF'.

8 The fourth word; TE **bit 1** is the state of shared data variable ac0101. TE **bit 2** is the state of shared data variable ac0102. A Task Expert (TE) application may use these bits to instruct the PLC to perform a procedure or function.

NOTE: Interminals with software revisions below 4.03, TE bit 1 is triggered by either ac0101 or as0101, and TE bit 2 is triggered by either ac0102 or as0102.

9 The fourth word **bit 8** is set to '1' when the Enter Key has been pressed on the keypad of the indicator. The bit can be reset to '0' by sending the command 75 (decimal) in the PLC output command word.

10 The fourth word **bits 9, 10**, and **11** indicate the state of the associated hardware input internal to the indicator; these are 0.1.1, 0.1.2 and 0.1.3. When the input is 'ON' the associated bit is set to '1'.

11 The fourth word **bit 12**; The motion bit is set to '1' when the scale is in motion.

12 The fourth word **bit 13**; The net mode bit is set to '1' when scale is in the net mode (a tare has been taken).

13 The fourth word **bit 15**; The data ok bit is set to '1' when the indicator operating conditions are normal, and when in the x10 display mode. Note that, when in x10 mode, the data sent is the higher resolution. The bit is set to '0' during power-up, during indicator setup, when the scale is over capacity or under zero. The PLC should continuously monitor the data ok bit in the IND500x communication and also any PLC data connection fault bit that exists in the PLC (refer to the PLC manufacturer documentation) to determine the validity of the data in the PLC.

NOTE: If communication between the IND500x and the ACM500 is interrupted, the Data OK bit is not affected and will remain at its last transmitted status.

Table B-2: Floating Point Input Indication

Dec	Data	Dec	Data
0	Gross Weight [1]	11	Low-pass filter frequency
1	Net Weight [1]	12	Notch filter frequency
2	Tare Weight [1]	13	Target value [3]
3	Fine Gross Weight [1]	14	+ Tolerance value [3]
4	Fine Net Weight [1]	15	Fine feed value [3]
5	Fine Tare Weight [1]	16	- Tolerance value [3]
6	Rate[1]	17	Spill value [3]
7	Custom field #1	18	Primary units, low increment size
8	Custom field #2 [2]	30	No data response command successful
9	Custom field #3	31	Invalid Command
10	Custom field #4 [2]	-	-

Notes for Table B-2

1 Data is refreshed on every indicator update

2 Data is ASCII characters and is limited to the first 4 characters

3 Value that is in the Target registers, may not be the active Target value

Table B-3: Discrete Write Floating Point – PLC >> IND500x, per Message Slot

Bit Number	1st Word [Scale command]	2nd Word	3rd Word
0	See Note 1	See Notes 2 and 3	
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			

Notes for Table B-3

1 The first word is a 16 bit integer and is used to send commands to the indicator. The commands are used to:

- instruct the indicator to report a specific type of data in words 2 and 3. Examples are Gross Weight, Net Weight, + Tolerance Value, etc.
- instruct the indicator to load the floating point data in the second and third words for a tare value, target value; or other value
- instruct the indicator to turn on internal outputs or perform a functions, such as Clear Tare, Print,Tare, Start Weigh,etc

2 The second and third words represent a 32 bit single precision floating point value that will be used for downloading a tare, target or other value to the indicator.

3 Not all commands require a floating point value in the second and third words.

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

Dec	Hex	Command
0	0	Report next rotation field @ next A/D update [1]
1	1	Report next rotation field [1,3]
2	2	Report next rotation field [1,3]
3	3	Reset (cancel) rotation
10	A	Report gross weight [2]
11	B	Report net weight [2]
12	C	Report tare weight [2]
13	D	Report fine gross weight [2]
14	E	Report fine net weight [2]
15	F	Report fine tare weight [2]
16	10	Report Rate[2]
17	11	Report custom float value #1 [2,5]
18	12	Report custom string value #2 [2,4,5]
19	13	Report low-pass filter frequency [2,5]
20	14	Report notch filter frequency [2]
21	15	Report target value [2,5]
22	16	Report (+) tolerance value [2,5]
23	17	Report fine feed [2,5]
24	18	Report (-) tolerance value [2,5]
25	19	Report spill value[5]
27	1B	Report custom float value #3 [5]
28	1C	Report custom string value #4 [5]
30	1E	Report primary units [5]
40	28	Add gross weight to rotation [7]
41	29	Add net weight to rotation [7]
42	2A	Add tare weight to rotation [7]
43	2B	Add fine gross weight to rotation [7]
44	2C	Add fine net weight to rotation [7]
45	2D	Add fine tare weight to rotation [7]
46	2E	Add rate to rotation [7]
47	2F	Add custom value #1 to rotation [7]
48	30	Add custom value #2 to rotation [7]
60	3C	Load programmable tare value [6]
61	3D	Pushbutton tare command [7]
62	3E	Clear command [7]
63	3F	Print command [7]
64	40	Zero command [7]
68	44	Trigger 1 command [7]
69	45	Trigger 2 command [7]
70	46	Trigger 3 command [7]
71	47	Trigger 4 command [7]
72	48	Trigger 5 command [7]
73	49	Set low-pass filter frequency [6]
74	4A	Set notch filter frequency [6]
75	4B	Reset (clear) ENTER key [7]
80	50	Clear display message [7,8]
81	51	Display Message 1 [7,8]
82	52	Display Message 2 [7,8]
83	53	Display Message 3 [7,8]
84	54	Display Message 4 [7,8]
85	55	Display Message 5 [7,8]

Dec	Hex	Command
86	56	Display Message 6 [7,8]
87	57	Display Message 7 [7,8]
88	58	Disable weight display [7]
89	59	Enable weight display [7]
90	5A	Set discrete output 0.1.1 "ON" [7]
91	5B	Set discrete output 0.1.2 "ON" [7]
92	5C	Set discrete output 0.1.3 "ON" [7]
93	5D	Set discrete output 0.1.4 "ON" [7]
100	64	Set discrete output 0.1.1 "OFF" [7]
101	65	Set discrete output 0.1.2 "OFF" [7]
102	66	Set discrete output 0.1.3 "OFF" [7]
103	67	Set discrete output 0.1.4 "OFF" [7]
110	6E	Set target value [6]
111	6F	Set target fine feed value [6]
112	70	Set - tolerance value [6]
113	71	Target Apply[12]
116	74	Target use gross weight [7]
117	75	Target use net weight [7]
119	77	Start [7,10]
121	79	Enable target latching [7]
122	7A	Disable target latching [7]
124	7C	Set Spill Value [6]
131	83	Set (+) tolerance value [6]
160	A0	Apply scale setup (reinitialize) [7, 9]
162	A2	Disable indicator tare (IDNet only) [7]
163	A3	Enable indicator tare (IDNet only) [7]
192	C0	Trigger OK key [7]

Dec	Hex	Command
193	C1	Trigger ENTER key [7]
194	C2	Pause [7,10]
195	C3	Resume
196	C4	Abort [7,10]
210	D2	Set Comparator 1 limit
211	D3	Set Comparator 1 high limit
212	D4	Set Comparator 2 limit
213	D5	Set Comparator 2 high limit
214	D6	Set Comparator 3 limit
215	D7	Set Comparator 3 high limit
216	D8	Set Comparator 4 limit
217	D9	Set Comparator 4 high limit
218	DA	Set Comparator 5 limit
219	DB	Set Comparator 5 high limit
220	DC	Disable Keypad
221	DD	Enable Keypad
222	DE	Report Comparator 1 limit
223	DF	Report Comparator 1 high limit
224	E0	Report Comparator 2 limit
225	E1	Report Comparator 2 high limit
226	E2	Report Comparator 3 limit
227	E3	Report Comparator 3 high limit
228	E4	Report Comparator 4 limit
229	E5	Report Comparator 4 high limit
230	E6	Report Comparator 5 limit
231	E7	Report Comparator 5 high limit
232	E8	Apply Comparator Values

Notes for Table B-4

1 Rotation is set up by commands 40 to 48 (dec). On each indicator update the next field of the rotation setup is reported in the second and third words of the floating point output from the indicator. The floating point indication date reports what the field data represents. To keep up with the rotation changes, the PLC program scan time should be 30 milliseconds or less. A command of '0' without rotation setup will report the scale gross weight. The commands acknowledge bits are set to the value of '0'.

2 A command that requests data that is refreshed on every indicator update.

3 Toggling between commands 1 and 2 will allow the PLC to control the rotation field change.

4 Only 4 characters of a string field are reported; the PLC must process the data as a string value.

5 A command that request a specific value; as long as the request is in the command word to the indicator no other data will be reported by the indicator.

6 A command that requires a floating point value be in the second and third word when the command is sent to the indicator. If the command is successful the returned floating point value will equal the value sent to the indicator.

7 A command that will not report back a value; the floating point data from the indicator will be zero.

8 The message display commands will cause messages to be displayed on the indicator display above the soft key prompts; this is limited to 20 characters. The message display commands cause a value to be written to shared data PD0119; PD0119 values can be use by Task Expert applications. The command 81 to 87 (dec) will initiate the message events. Command 81will cause the characters in shared data AW0101 to be displayed and PD0119 will be set to '1'. Command 82 = display AW0102 and PD0119 = '2'. Command 83' = display AW0103 and PD0119 = '3'. Command 84 display AW0104 and PD0119 = '4'.

Command 85 = display AW0105 and PD0119 = '5'. Command 86 = start Prompt sequence, PD0119 = '6' and XC0134 = '1'. Command = display PD0118 and PD0119 = '7'. Command 80 (dec) will remove the message display.

9 If shared data classes pl, ds, ll, nt, ce, zr, ct, cm, xs, cs, dp, wk, ao, rp, or dc are changed by the PLC this command (160 dec) will trigger the changes into effect. Shared data is not available with Modbus TCP.

10 A command that can only be used with the IND500x Fill.

11 If Fill-500x is not installed in the terminal, this command can be used to access the corresponding Shared Data field incorporated into a custom TaskExpert program.

12 Target Apply is used to load Target Values from Shared Data to Active Values. in Active Values, Target Apply can be used to update Active Value display. Apply Target can be used in Manual Filling, OverUnder, Basic Auto Filling when Latching is disabled. Apply Target can also be used in Advanced Auto Filling, Drum Filling and Basic Auto Filling when Latching is enabled if Filling is ready after Aborted.

B.2. Floating Point Command Examples

Table B-5: 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 IND500x terminal to report net weight)	11 (dec) loaded into command word 0	none required		
2 (IND500x 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 IND500x terminal will update the net value every interface update cycle.

Table B-6: 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 (IND500x 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 IND500x terminal to start "using" new target value)	114 (dec) loaded into command word 0			
5 (IND500x 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 IND500x 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 B-7: 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 (IND500x 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 (IND500x 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 (IND500x 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 IND500x 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 (IND500x 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 IND500x terminal sends the rate value at the next interface update cycle)	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 IND500x terminal sends	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
the gross value at next interface update cycle)				
11 (PLC leaves 0 in command word and IND500x 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 IND500x 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 B-8: 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 (IND500x 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 (IND500x 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 (IND500x 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 IND500x 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			

Step #	Scale command (from PLC)	Scale Floating Point Value	Command response from terminal	Floating Point Value
8 (IND500x 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 IND500x terminal does NOT see this as another command to report the next rotation field.	2 (dec) loaded into command word 0			
10 (IND500x 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 (IND500x 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 (IND500x 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 IND500x 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

C. PLC Data Sizes

C.1. Introduction

This appendix provides information about data sizes for Integer, Division, and Floating Point data transfer between the IND500x and a PLC system. RSLogix5000 communication formats are included as an example of how the 8 bit sizes required by the IND500x are translated into 16-bit integer sizes that can be used by the RSLogix500.

C.2. Integer and Divisions Data Formats

In the IND500x, both Integer and Division data formats are 16-bit signed integers that support values of -32,768 to +32,767.

Integer is the weight in the displayed units. Divisions are the internal calibrated scale divisions – displayed weight divided by division size. Divisions are used to accommodate large weight values in integer form. The Divisions value must be multiplied by the division size in the PLC to obtain the weight in the displayed units. The IND500x integer data formats use no overhead bytes, with 4 bytes input and 4 bytes output per message slot.

■ For this appendix, Input and Output references are made with respect to the PLC.

Table C-1 shows data sizes per message slot for data sent to the PLC and received from the PLC by the IND500x. The corresponding RSLogix5000 format is included as an example.

Table C-1: IND500x Integer Data Sizes

	IND500x Floating Point Data Bytes (8 bit)		**Logix5000 Comm Format** INT (16 bit)	
Message Slots	**PLC Input from IND500x**	**PLC Output to IND500x**	**Input Size**	**Output Size**
1	4	4	2	2
2	8	8	4	4
3	12	12	6	6
4	16	16	8	8

C.3. Floating Point Data Format

The IND500x uses IEEE-754 single precision floating point. The IND500x Floating Point Input has a 0 byte overhead and uses 8 bytes per message slot. The Output has a 2 byte overhead and uses 6 bytes per message slot.

- Input and Output references are always with respect to the PLC.

The IND500x Floating Point data format uses 16 bit words. INT is the only communication format recommended for use with the IND500x Floating Point format. Note that DINT (double integer) is not compatible with even quantities of IND500x message slots, because – as seen in Table C-2 – the IND500x input (PLC output) is not in multiples of four 16 bit words for even message slot quantities.

Table C-2: IND500x Floating Point Data Sizes

	IND500x Floating Point Data Bytes (8 bit)		**Logix5000 Comm Format** INT (16 bit)	
Message Slots	**PLC Input from IND500x**	**PLC Output to IND500x**	**Input Size**	**Output Size**
1	8	8	4	4
2	16	14	8	7
3	24	20	12	10
4	32	26	16	13

D ASCII Characters

D.1. Standard Characters

Char.	Dec.	Hex.
NUL	0	00
SOH	1	01
STX	2	02
ETX	3	03
EOT	4	04
ENQ	5	05
ACK	6	06
BEL	7	07
BS	8	08
HT	9	09
LF	10	0A
VT	11	0B
FF	12	0C
CR	13	0D
SO	14	0E
SI	15	0F
DLE	16	10
DC1	17	11
DC2	18	12
DC3	19	13
DC4	20	14
NAK	21	15
SYN	22	16
ETB	23	17
CAN	24	18
EM	25	19
SUB	26	1A
ESC	27	1B
FS	28	1C
GS	29	1D
RS	30	1E
US	31	1F
SP	32	20
!	33	21
″	34	22

Char.	Dec.	Hex.
#	35	23
$	36	24
%	37	25
&	38	26
′	39	27
(	40	28
)	41	29
*	42	2A
+	43	2B
,	44	2C
-	45	2D
.	46	2E
/	47	2F
0	48	30
1	49	31
2	50	32
3	51	33
4	52	34
5	53	35
6	54	36
7	55	37
8	56	38
9	57	39
:	58	3A
;	59	3B
<	60	3C
=	61	3D
>	62	3E
?	63	3F
@	64	40
A	65	41
B	66	42
C	67	43
D	68	44
E	69	45

Char.	Dec.	Hex.
F	70	46
G	71	47
H	72	48
I	73	49
J	74	4A
K	75	4B
L	76	4C
M	77	4D
N	78	4E
O	79	4F
P	80	50
Q	81	51
R	82	52
S	83	53
T	84	54
U	85	55
V	86	56
W	87	57
X	88	58
Y	89	59
Z	90	5A
[	91	5B
\	92	5C
]	93	5D
^	94	5E
_	95	5F
`	96	60
a	97	61
b	98	62
c	99	63
d	100	64
e	101	65
f	102	66
g	103	67
h	104	68

Char.	Dec.	Hex.
i	105	69
j	106	6A
k	107	6B
l	108	6C
m	109	6D
n	110	6E
o	111	6F
p	112	70
q	113	71
r	114	72
s	115	73
t	116	74
u	117	75
v	118	76
w	119	77
x	120	78
y	121	79
z	122	7A
{	123	7B
|	124	7C
}	125	7D
~	126	7E
□	127	7F
Reserved	128	8A
	to	*to*
	159	9F
	160	A0
¡	161	A1
¢	162	A2
£	163	A3
€	164	A4
¥	165	A5
Š	166	A6
§	167	A7
š	168	A8

Char.	Dec.	Hex.
©	169	A9
ª	170	AA
«	171	AB
¬	172	AC
-	173	AD
®	174	AE
¯	175	AF
°	176	B0
±	177	B1
²	178	B2
³	179	B3
Ž	180	B4
µ	181	B5
¶	182	B6
·	183	B7
ž	184	B8
¹	185	B9
º	186	BA
»	187	BB
Œ	188	BC
œ	189	BD
Ÿ	190	BE
¿	191	BF
À	192	C0
Á	193	C1
Â	194	C2
Ã	195	C3
Ä	196	C4
Å	197	C5
Æ	198	C6
Ç	199	C7
È	200	C8
É	201	C9
Ê	202	CA
Ë	203	CB
Ì	204	CC
Í	205	CD
Î	206	CE
Ï	207	CF
Ð	208	D0
Ñ	209	D1
Ò	210	D2
Ó	211	D3
Ô	212	D4
Õ	213	D5
Ö	214	D6
×	215	D7
Ø	216	D8
Ù	217	D9
Ú	218	DA
Û	219	DB
Ü	220	DC
Ý	221	DD
Þ	222	DE
ß	223	DF
à	224	E0
á	225	E1
â	226	E2
ã	227	E3
ä	228	E4
å	229	E5
œ	230	E6
ç	231	E7
è	232	E8
é	233	E9
ê	234	EA
ë	235	EB
ì	236	EC
í	237	ED
î	238	EE
ï	239	EF
ð	240	F0
ñ	241	F1
ò	242	F2
ó	243	F3
ô	244	F4
õ	245	F5
ö	246	F6
÷	247	F7
ø	248	F8
ù	249	F9
ú	250	FA
û	251	FB
ü	252	FC
ý	253	FD
þ	254	FE
ÿ	255	FF

D.2. Control Characters

Char	Definition	Function
SOH	START OF HEADING	A transmission control character used as the first character of a heading of an information message.
STX	START OF TEXT	A transmission control character that precedes a text and that is used to terminate a heading.
ETX	END OF TEXT	A transmission control character that terminates a text.
EOT	END OF TRANSMISSION	A transmission control character used to indicate the conclusion of the transmission of one or more texts.
ENQ	ENQUIRY	A transmission control character used as a request for a response from a remote station; the response may include station identification and/or station status. When a ″Who are you″ function is required on the general switched transmission network, the first use of ENQ after the connection is established will have the meaning ″Who are you″ (station identification). Subsequent use of ENQ may, or may not, include the function ″Who are you″, as determined by agreement.

Char	Definition	Function
ACK	ACKNOWLEDGE	A transmission control character transmitted by a receiver as an affirmative response to the sender.
BEL	BELL	A control character that is used when there is a need to call for attention; it may control alarm or attention devices.
BS	BACKSPACE	A format effector that moves the active position one character position backwards on the same line.
HT	HORIZONTAL TABULATION	A format effector that advances the active position to the next pre-determined character position on the same line.
LF	LINE FEED	A format effector that advances the active position to the same character position of the next line.
VT	VERTICAL TABULATION	A format effector that advances the active position to the same character position on the next pre-determined line.
FF	FORM FEED	A format effector that advances the active position to the same character position on a pre-determined line of the next form or page.
CR	CARRIAGE RETURN	A format effector that moves the active position to the first character position on the same line.
SO	SHIFT OUT	A control character that is used in conjunction with SHIFT IN and ESCAPE to extend the graphic character set of the code.
SI	SHIFT IN	A control character that is used in conjunction with SHIFT OUT and ESCAPE to extend the graphic character set of the code.
DLE	DATA LINK ESCAPE	A transmission control character that will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphic characters and transmission control characters can be used in DLE sequences.
DC1	DEVICE CONTROL ONE	A device control character that is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to restore a device to the basic mode of operation (see also DC2 and DC3), or for any other device control function not provided by other DCs.
DC2	DEVICE CONTROL TWO	A device control character that is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to set a device to a special mode of operation (in which case DC1 is used to restore normal operation), or for any other device control function not provided by other DCs.

Char	Definition	Function
DC3	DEVICE CONTROL THREE	A device control character that is primarily intended for turning off or stopping an ancillary device. This function may be a secondary level stop, for example, wait, pause, stand-by or halt (in which case DC1 is used to restore normal operation). If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.
DC4	DEVICE CONTROL FOUR	A device control character that is primarily intended for turning off, stopping, or interrupting an ancillary device. If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.
NAK	NEGATIVE ACKNOWLEDGE	A transmission control character transmitted by a receiver as a negative response to the sender.
SYN	SYNCHRONOUS IDLE	A transmission control character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between data terminal equipment.
ETB	END OF TRANSMISSION BLOCK	A transmission control character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.
CAN	CANCEL	A character, or the first character of a sequence, indicating that the data preceding it is in error. As a result, this data is to be ignored. The specific meaning of this character must be defined for each application and/or between sender and recipient.
EM	END OF MEDIUM	A control character that may be used to identify the physical end of a medium, or the end of the used portion of a medium, or the end of the wanted portion of data recorded on a medium. The position of this character does not necessarily correspond to the physical end of the medium.
SUB	SUBSTITUTE	A control character used in the place of a character that has been found to be invalid or in error. SUB is intended to be introduced by automatic means.
ESC	ESCAPE	A control character that is used to provide additional control functions. It alters the meaning of a limited number of contiguously following bit combinations.
FS	FILE SEPARATOR	A control character used to separate and qualify data logically; its specific meaning has to be specified for each application. If this character is used in hierarchical order, it delimits a data item called a file.
GS	GROUP SEPARATOR	A control character used to separate and qualify data logically; its specific meaning has to be specified for each application. If this character is used in hierarchical order, it delimits a data item called a group.

Char	Definition	Function
RS	RECORD SEPARATOR	A control character used to separate and qualify data logically; its specific meaning has to be specified for each application. If this character is used in hierarchical order, it delimits a data item called a record.
US	UNIT SEPARATOR	A control character used to separate and qualify data logically; its specific meaning has to be specified for each application. If this character is used in hierarchical order, it delimits a data item called a unit.

To protect your product's future:
METTLER TOLEDO Service assures the quality, measuring accuracy and preservation of value of this product for years to come.

Please request full details about our attractive terms of service.

▶ **www.mt.com/service**

www.mt.com/IND500x
For more information

Mettler-Toledo, LLC
1900 Polaris Parkway
Columbus, OH 43240
Phone 800 438 4511
Fax 614 438 4900

30753830 Rev. C, 12/2023

30753830