ESCORT MEMORY SYSTEMS

A Datalogic Group Company

ems

GWY-01-IND-01

Subnet16 Industrial Gateway Interface Module

Operator's Manual

Document P/N: 17-1306

Escort Memory Systems
170 Technology Circle
Scotts Valley, CA 95066 USA
Telephone: (831) 438-7000
FAX: (831) 438-5768
Website: www.ems-rfid.com
Email: info@ems-rfid.com

RFID AT WORK™

Escort Memory Systems reserves the right to make modifications and improvements to its products and/or documentation without prior notification. Escort Memory Systems shall not be liable for technical or editorial errors or omissions contained herein, nor for incidental or consequential damages resulting from the use of this material.

The text and graphic content of this publication may be used, printed and distributed only when all of the following conditions are met:

- Permission is first obtained from Escort Memory Systems.
- The content is used for non-commercial purposes only.
- Copyright information is clearly displayed (Copyright © 2007, Escort Memory Systems, All Rights Reserved).
- The content is not modified.

The following are trademarks and/or registered trademarks of Escort Memory Systems, a Datalogic Group Company: Escort Memory Systems®, and the Escort Memory Systems logo, Subnet16™, RFID AT WORK™.

Third party product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies: Philips, Rockwell Automation (ControlLogix, RSLogix), Texas Instruments, Infineon, Modicon, Belden, Microsoft and the Open DeviceNet Vendor Association (ODVA).

ESCORT MEMORY SYSTEMS

GWY-01-IND-01

Subnet16 Industrial Gateway - Operator's Manual

For the GWY-01-IND-01 TCP/IP Gateway Interface Module

Publication P/N: 17-1306 REV 06 (06/07)

ESCORT MEMORY SYSTEMS

GWY-01-IND-01

SUBNET16™ INDUSTRIAL GATEWAY INTERFACE MODULE

High frequency, Multi-protocol, Industrial Ethernet RFID Interface Module

OPERATOR'S MANUAL

How to Install, Configure and Operate
Escort Memory Systems'
Subnet16 Industrial Gateway

REGULATORY COMPLIANCE

FCC PART 15.105

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment uses, generates, and can radiate radio frequency energy and, if not installed and used in accordance with these instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

Reorient or relocate the receiving antenna.

Increase the separation between the equipment and receiver.

Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

Consult the dealer or an experienced radio/TV technician for help.

FCC PART 15.21

Users are cautioned that changes or modifications to the unit not expressly approved by Escort Memory Systems may void the user's authority to operate the equipment.

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

CE

This product complies with the following regulatory specifications: EN 60950, EN-300-330, EN-300-683, IEC 68-2-1, IEC 68-2-6, IEC 68-2-27 and IEC 68-2-28.

TELEC

This product complies with TELEC Regulations for Enforcement of the Radio Law Article 6, section 1, No. 1.

Certification #: *PENDING*

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: GETTING STARTED

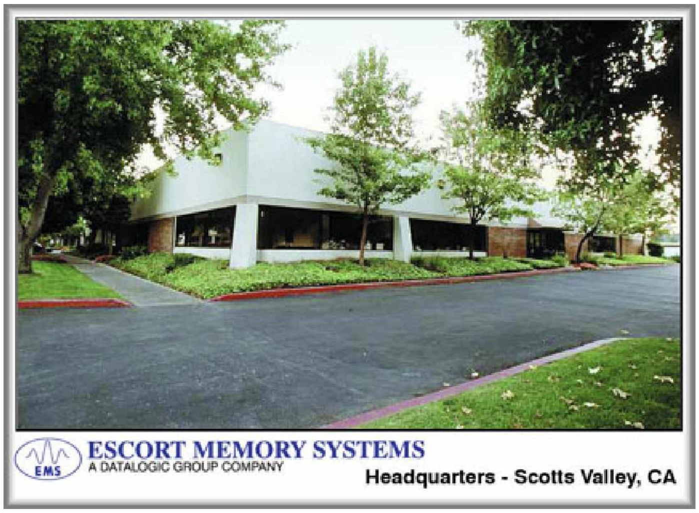
1.1 INTRODUCTION

Welcome to the ***GWY-01-IND-01 - Operator's Manual***. This manual will assist you in the installation, configuration and operation of Escort Memory Systems' Subnet16 Industrial Gateway Interface Module.

The Industrial Gateway can control up to 16 passive high frequency read/write Radio-Frequency Identification (*RFID*) controllers. In order to meet and exceed the requirements of the industrial automation industry, the Industrial Gateway and EMS RFID controllers are designed to be compact, reliable and rugged.

1.1.1 Company Background

Escort Memory Systems has long been an industry leader in providing Radio Frequency Identification devices, building a solid reputation by consistently delivering an extended selection of quality, durable industrial RFID systems.



1.1.2 The Subnet16™ Gateway

Subnet16™ is a 16-Node Multidrop bus architecture and protocol that provides Ethernet connectivity for up to 16 RFID controllers through a single Gateway device.

The Industrial Gateway supports industrial Ethernet standards including *EtherNet IP* and *Modbus-TCP* and is compatible with EMS' Cobalt HF-Series RFID controllers, LRP, HMS and T-Series RFID tags. Many of EMS' legacy RFID controllers with MUX32™ capability are also compatible with either Gateway model.

1.2 SUBNET16 GATEWAY FEATURES

- 10/100 Mbps
- Controls up to 16 RFID controllers
- Supports controller macro functionality
- Flash memory for software updates
- Real-time Calendar/Date functions
- Supports Rockwell Automation's Industrial Ethernet/IP™ (*GWY-01-IND model only*)
- Supports Telemecanique's Modicon Ethernet Modbus TCP™ (*GWY-01-IND model only*)
- Supports standard TCP/IP protocol
- *OnDemand Utilities* for legacy support of PLC5E and RA SCL5/05 PLCs
- Downward compatible with most EMS Mux32™ compliant products
- Auto configuration of RFID controllers
- Automatic Node ID Number Assignment
- Node Fault Detection
- Sources up to 2.5A current over Subnet16™ bus (for test purposes)
- Isolated power and bus interfaces
- ARM7 processing power
- Six LED status indicator lights
- FCC/CE Agency compliance

1.3 About this Manual

This manual provides guidelines and instructions for installing, configuring and operating Escort Memory Systems' Subnet16 Industrial Gateway Interface Module (GWY-01-IND-01).

This document does NOT include explicit details regarding each of the Gateway's RFID commands. Specific RFID command related information is available in the ***CBx Command Protocol – Reference Manual***, which is available at www.ems-rfid.com.

However, this manual does explain the process of issuing commands from a host PC or Programmable Logic Controller (PLC) to a Subnet16 Gateway, Subnet network and attached RFID controllers.

NOTE:

-Throughout this manual, the GWY-01-IND-01 is referred to as the "*Subnet16 Gateway*" or simply "*the Gateway"*.

-HF-Series RFID Controllers and Cobalt RFID Controllers are referred to as *HF-Series Controllers, Cobalt Controllers,* or just "*the Controller."*

-In addition, the terms "*Subnet Node Number*", "*Node ID*" and "*Controller ID*" are used interchangeably.

1.3.1 Who Should Read this Manual?

This manual should be read by those who will be installing, configuring and operating the Gateway. This may include the following people:

§ Hardware Installers

§ System Integrators

§ Project Managers

§ IT Personnel

§ System and Database Administrators

§ Software Application Engineers

§ Service and Maintenance Engineers

1.3.2 HEX Notation

Throughout this manual, numbers expressed in Hexadecimal notation are prefaced with "**0x**"**.** For example, the number "10" in decimal is expressed as "0x0A" in hexadecimal. See Appendix D for a chart containing Hex values, ASCII characters and their corresponding decimal integers.

1.4 GATEWAY DIMENSIONS

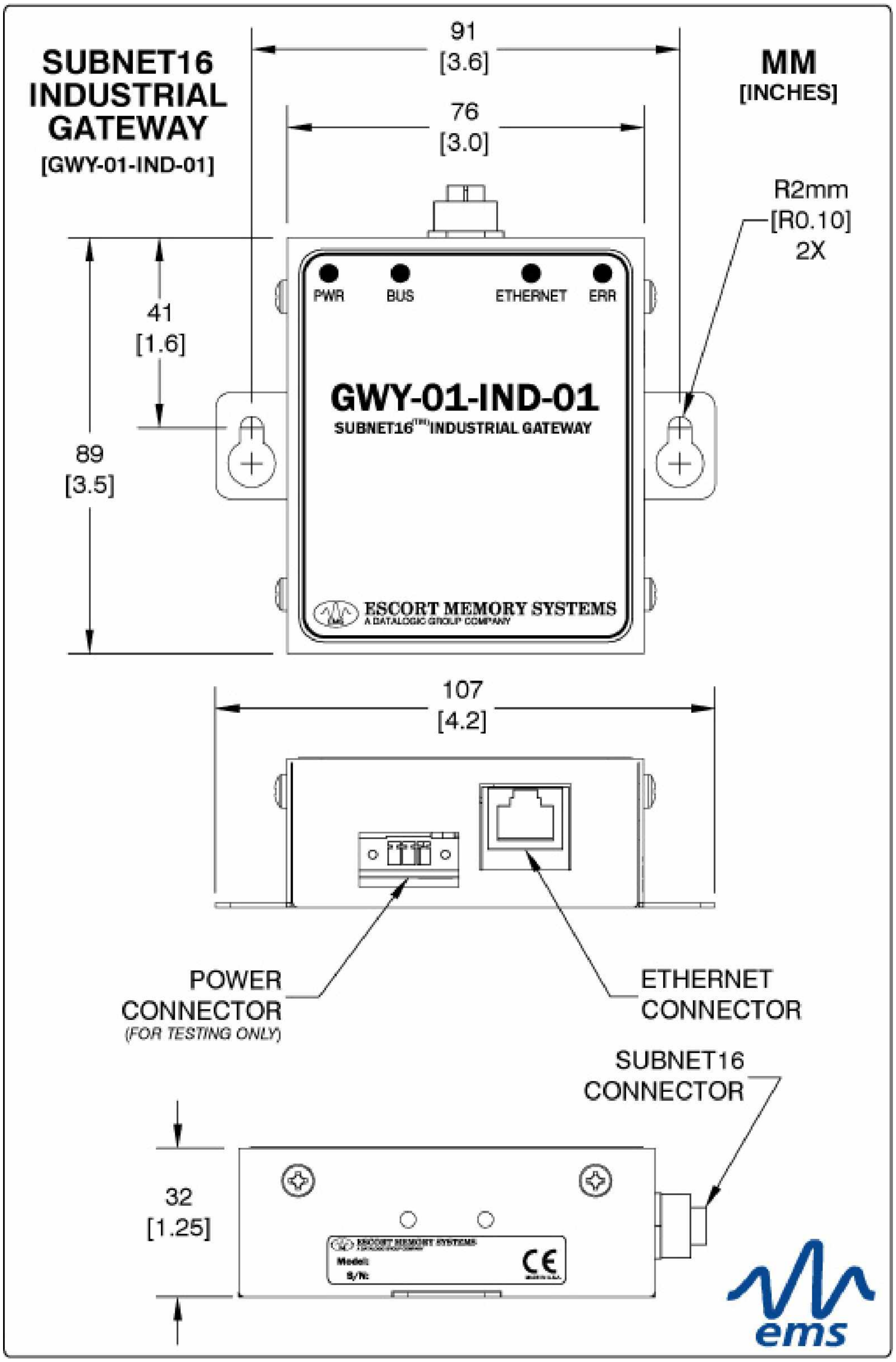


Figure 1-1: Gateway Dimensions

1.5 INSTALLATION GUIDELINES

1.5.1 Hardware Requirements

The following components are required for a complete Subnet16 RFID system:

- One Subnet16 Industrial Gateway Interface Module
- One host PC with an Ethernet network connection
- One to 16 RFID controllers (Cobalt or HF-0405-Series)
- Adequate length cabling, connectors and terminators
- A suitable power supply capable of providing sufficient power to the Gateway and its RFID controllers
- EMS' HMS-Series or LRP-Series or T-Series RFID tags
- For Ethernet/IP users: One ControlLogix PLC with a 1756-ENBT ControlLogix Ethernet/IP module installed

1.5.2 Installation Precautions

The Gateway is designed to withstand 8kV of direct electro-static discharge (ESD) and 15kV of air gap discharge. However, it is not uncommon for some applications to generate considerably higher ESD levels.

- Avoid mounting the Gateway or its RFID controllers near sources of EMI (electro-magnetic interference) or near devices that generate high ESD levels.
- Use adequate ESD prevention measures to dissipate potentially high voltages.
- Do not route cables near unshielded cables or near wiring carrying high voltage or high current.
- Avoid routing cables near motors and solenoids.
- Cables should only cross at perpendicular intersections.

1.5.3 Network & Power Considerations

- Refer to the network diagrams in *Appendix C*. Choose the network architecture (ThickNet vs. ThinNet) that best suits your RFID requirements.
- Construct your chosen network using only EMS approved Subnet16 cables, Drop-T connectors, Terminating Resistors and accessories.
- Review the power requirements of your RFID network and provide a suitable power supply. (See *Appendix B* for power supplies offered by EMS).

1.6 INSTALLING THE GATEWAY

Follow the Steps Below to Install the Gateway

1 Note the *Installation Guidelines* in *Chapter 1, Section 1.5*.

2 Securely mount the Gateway to your chosen location using two M5 (#10) screws, lock washers and nuts. The Gateway may be mounted in any orientation, but should be aligned in such a manner that the LED indicators can be seen during operation.

3 Attach one end of a 5-pin, male-to-male, M12, ThinNet drop cable to the 5-pin, female, M12 connector on the Gateway. Connect the other end of this 5-pin, male-to-male, M12, ThinNet drop cable to the 5-pin, female, M12 connector on EITHER a ***ThickNet to ThinNet Drop-T Connector*** OR a ***ThinNet to ThinNet Drop-T Connector*** (as per your network and RFID application requirements).

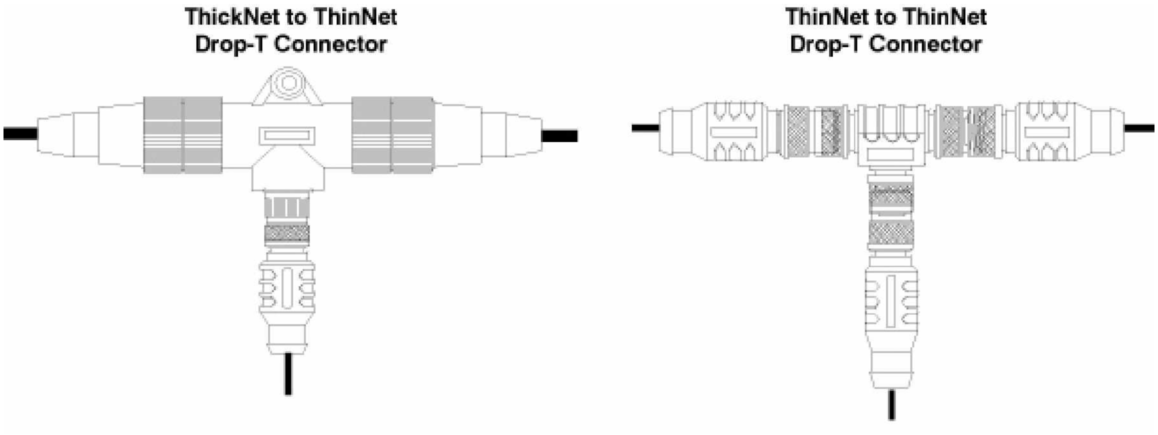


Figure 1-2: Drop-T Connectors

4 Attach one end of a male-to-female trunk cable to each mating connector on the Drop-T Connector. Continue connecting trunk cables and Drop-T connectors as needed. *Note: trunk length should not exceed 300m for ThickNet and 20m for ThinNet.*

5 Connect the male end of a 5-pin, male-to-female ThinNet drop cable to the female end on your Drop-T connector(s). Attach the remaining female end of the ThinNet drop cable to the 5-pin, male, M12 connector on a RS485-based RFID controller (*HF-0405-485, HF-CNTL-485, C1007-485* or *C0405-485)*.

6 Repeat ***Step 5*** for each RFID controller you plan to install. *Note: maximum drop cable length is 2m.*

Figure 1-3: Subnet16 ThinNet Cable

7 ***For ThickNet Networks:*** Using a 5-pin, female, 7/8 - 16, ThickNet to Bare Wire Leads cable (***EMS P/N: CBL-1495-XX***), connect the bare wires to your power supply. Attach the female, ThickNet end to the 5-pin, male, ThickNet end on a Drop-T connector.

For ThinNet Networks: Using a 5-pin, female, M12, ThinNet to Bare Wire Leads cable (***EMS P/N: CBL-1494-XX***), connect the bare wires to your power supply. Attach the female, ThinNet end to the 5-pin, male, ThinNet end on a Drop-T connector *(XX = length in metes).*

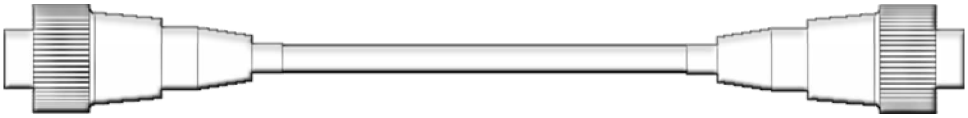


Figure 1-4: Subnet16 ThickNet Cable

8 Connect the Gateway to your host computer via Category 5e Ethernet cabling. A crossover cable may be required if you are connecting the Gateway directly to a computer (rather than to a switch, network hub or router).

9 Turn the power supply ON. The PWR LED on the Gateway will remain lit while power is applied to the unit.

CHAPTER 2: GATEWAY OVERVIEW

2.1 OPERATING MODES

Subnet16 Gateways have two basic modes of operation: **Subnet16™** and **Mux32™**. Operating mode is set via DIP-switch #2 on the main board. (*Refer to Section 2.3 for DIP-switch access and settings*).

DIP-switch 2 ON = Subnet16 (Default Setting)

DIP-switch 2 OFF = Mux32

2.1.1 Subnet16™

Subnet16™ is an advanced feature-rich network protocol that supports a subset of the EMS Mux32 legacy protocol. The advanced features implemented in the Subnet16 protocol allow the Gateway to assign individual Node ID values automatically to each RFID controller connected on the Subnet bus. Subnet16 also allows the Gateway to detect when a new controller is connected to the Subnet or when a controller "falls off the bus" (stops responding).

Through the Subnet16 protocol, the Gateway is able to store a backup copy of each RFID controller's custom configuration settings. In the event that an RFID controller fails, the stored configuration settings can be automatically reassigned to a replacement RFID controller.

Real-time clock functionality is supported in Subnet16 mode. Host-bound data packets are automatically Time/Date stamped as they pass through the Gateway and on to the host.

Many of the RFID commands supported by the Gateway and RFID controllers will only function when the Gateway is in Subnet16 mode.

2.1.2 Mux32™

Mux32 is a well-established, multi-drop protocol incorporated into many of EMS' prior products including HMS-Series and LRP-Series RFID Controllers. Most, but not all, HMS and LRP commands supported by MUX32 are also supported by the Gateway (when in Mux32 mode). Legacy Mux32 controllers must support the "*ABx Fast*" command protocol to work with the Gateway in Mux32 Mode. For more information regarding ABx Fast, please see the *ABx Fast Protocol – Reference Manual* available online at www.ems-rfid.com.

Many advanced Subnet16 features are not available when the Gateway is running in Mux32 mode. RFID controllers must be assigned a unique Node ID number (via *Configuration Tag*) prior to being attached to the Gateway's Subnet.

Note that the Gateway does not support Node IDs 17-31. Node IDs 0 and 32 are reserved for special cases described later in this manual.

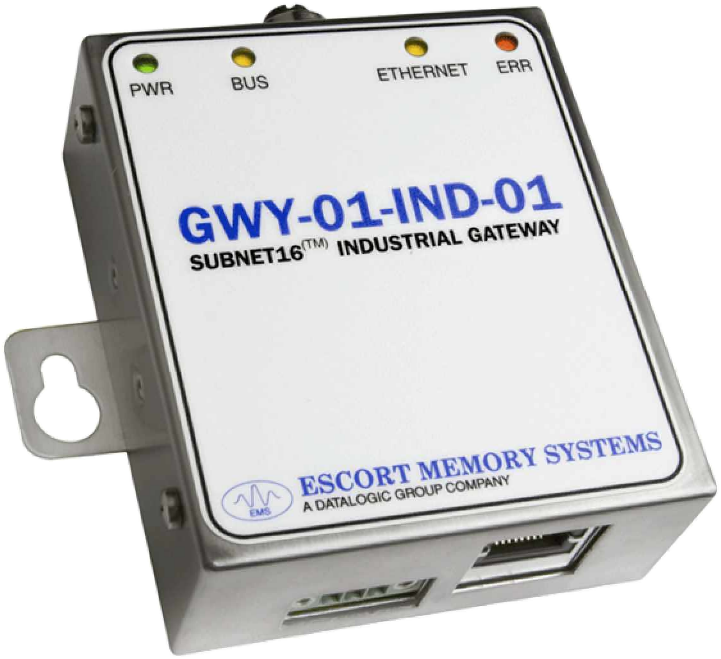
2.2 LED INDICATORS

The Gateway has six LED indicators. Four LEDs are located on the front panel of the unit and two LEDs are located within the Ethernet port just above the RJ45 connector. The LEDs display everything from bus and Ethernet activity, to diagnostic information and power status.

2.2.1 Front Panel LEDs

LED NAME	LED COLOR	DESCRIPTION
PWR	🟩 Green	The ***PWR*** (power) LED will light and remain ON while power is applied to the Gateway.
BUS	🟨 Amber	The ***BUS*** LED will flash ON and OFF to indicate that data is being transmitted between the Gateway and one or more RFID controllers.
ETHERNET	🟨 Amber	The ***ETHERNET*** LED will flash ON and OFF to indicate that data is being transmitted between the host and the Gateway.
ERR	🟥 Red	The ***ERR*** (error) LED will turn ON when the Gateway experiences an error condition. This LED will be cleared when the next valid command is received by the Gateway.

Table 2-1: Front Panel LEDs



2.2.2 Ethernet Connector Port LEDs

There are two LEDs located within the Ethernet connector port just above the RJ45 connector that are used to display Ethernet connection status and data traffic activity.

LED COLOR	DESCRIPTION
🟨 Yellow	The ***AMBER*** LED on the left is the *10/100 Indicator LED,* which will turn ON whenever an Ethernet link is established and will remain ON for the duration of the connection.
🟩 Green	The ***GREEN*** LED on the right is the *Ethernet Data LED,* which will flash ON and OFF when Ethernet traffic is detected by the Gateway (regardless of origin and destination).

Table 2-2: Ethernet Connector Port LEDs

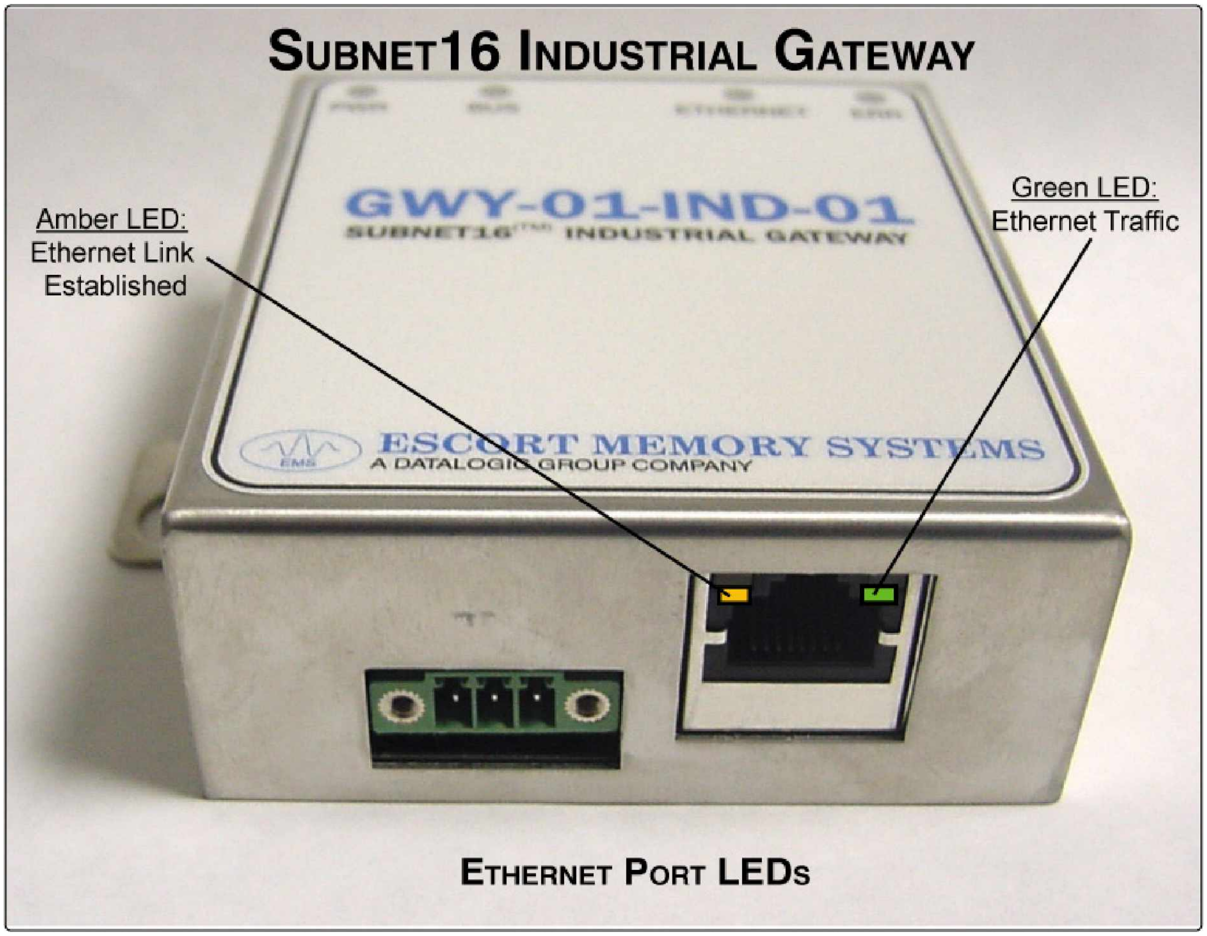


Figure 2-1: Ethernet Connector Port LEDs

2.3 DIP-SWITCH SETTINGS

Certain parameters and features are configured by setting DIP-switches on the unit's circuit board assembly. To access these DIP-switches, remove the four screws securing the lid to the base of the enclosure. After removing the lid, locate the DIP-switch block containing eight small DIP-switches.

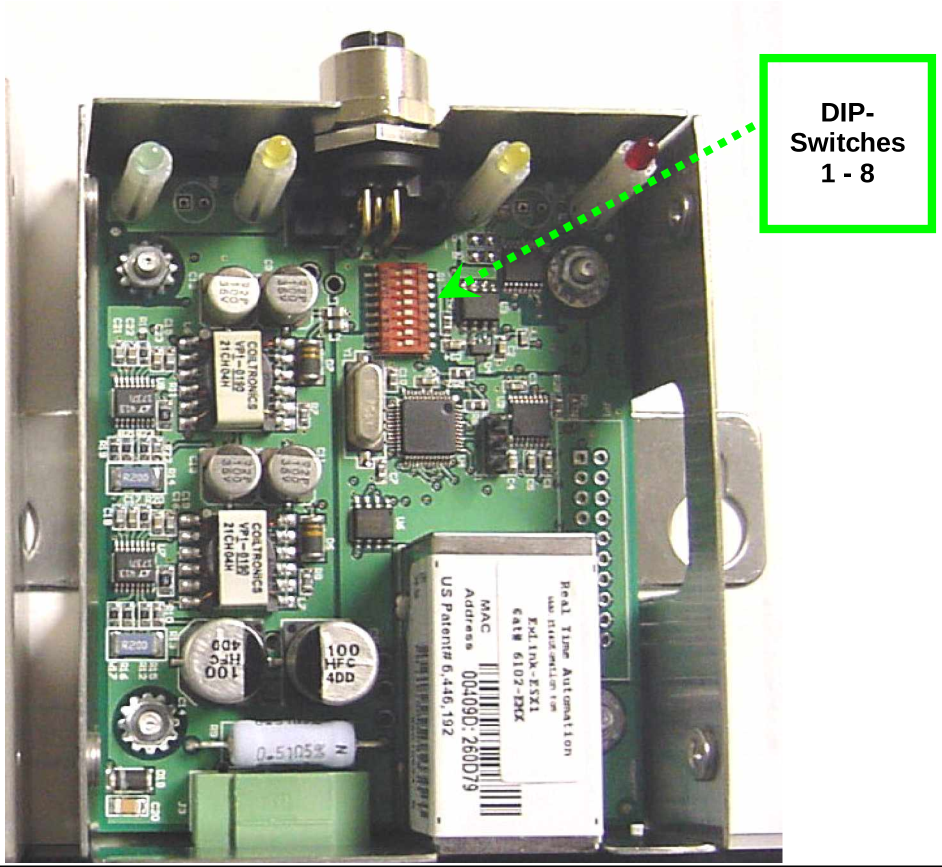
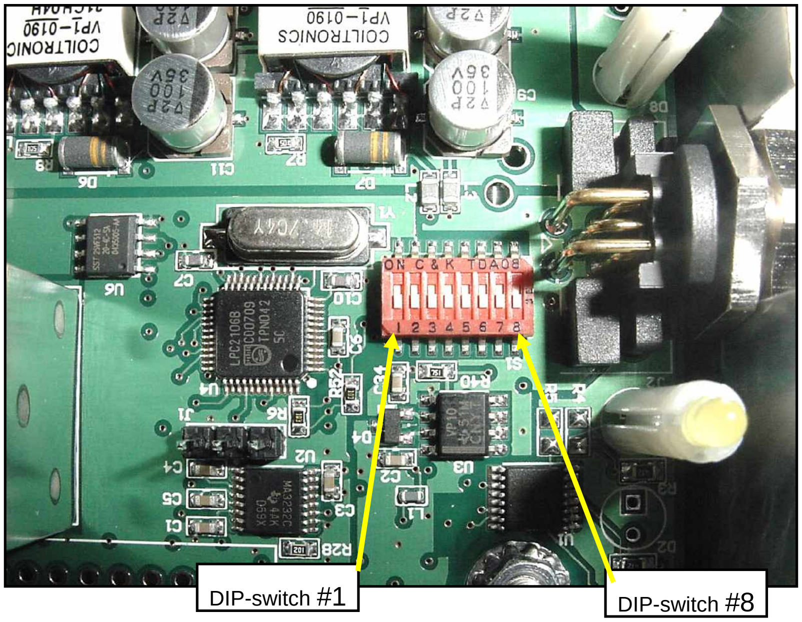
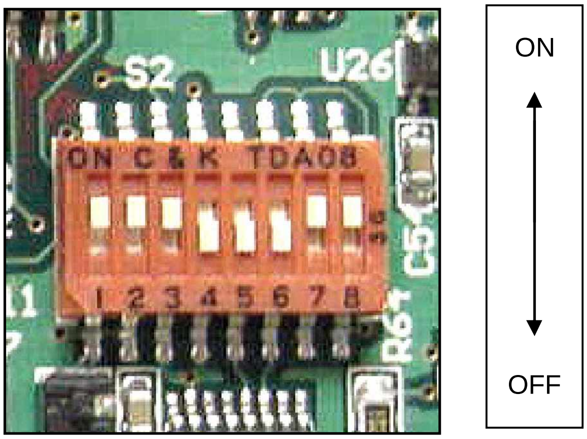


Figure 2-2: Gateway DIP-Switches



2.3.1 DIP-Switch 1: "Run Mode / Idle Mode"

The first DIP-switch selects whether the Gateway powers-up in "*Run Mode*" or "*Idle Mode*."



- **ON (Default Setting):** when this DIP-switch is ON, the Gateway will power-up in *Run Mode,* in which case it will immediately begin "*polling*" the Subnet network in an attempt to ready the connected RFID controllers to begin receiving commands.
- **OFF:** when this DIP-switch is OFF, the Gateway will power-up in *Idle Mode.* This option forces the Gateway to "wait" until the host establishes a connection, after which users must issue ***Command 0x70 - "Start Subnet"*** before the Gateway can process commands to the Subnet.

2.3.2 DIP-Switch 2: "Operating Mode"

The second DIP-switch selects the Gateway's ***Operating Mode***, thereby determining whether the device powers-up in ***Subnet16*** mode or in ***Mux32*** mode.

- **ON (Default Setting):** when this DIP-switch is ON, upon power-up, the Gateway will enter Subnet16 mode, enabling the Subnet16 protocol and expanded feature set.
- **OFF:** when this DIP-switch is OFF, upon power-up, the Gateway will enter Mux32 mode.

2.3.3 DIP-Switch 3: "Reset Default IP Address"

DIP-switch 3 on the *Industrial Gateway* is used to reset the unit's IP address to factory default settings.

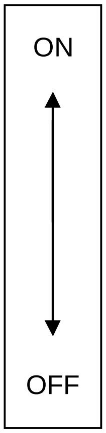
- **ON:** When this DIP-switch is ON, upon power-up, the Gateway will verify that the currently configured IP address matches that of the factory default value. If the IP addresses do not match, the current IP address will be cleared, the factory default IP address will be restored and the Gateway will automatically reboot and begin using the factory default IP address.
- **OFF (Default Setting):** when this DIP-switch is in the OFF position, the Gateway will not attempt to verify IP address settings upon power-up and will continue to use the currently configured IP address.

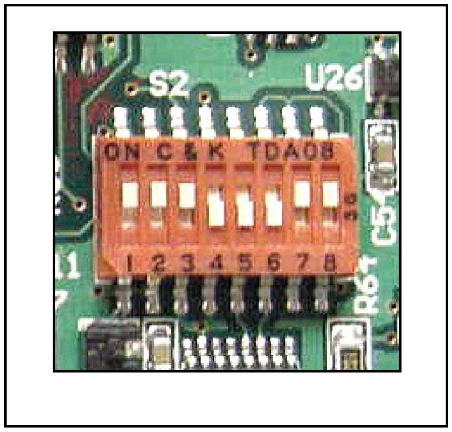
» Factory Default IP Address: 192.168.253.110 «

2.3.4 DIP-Switches 4-8

DIP-switches 4-8 enable/disable a 3-pin main board, serial connection terminal that is used by EMS' Manufacturing during the initial firmware installation process. Users should NOT modify these DIP-switches.

DIP-SWITCH	DEFAULT POSITION
4	OFF
5	OFF
6	OFF
7	ON
8	ON





2.3.5 DIP-Switch Position Status Table

DIP-SWITCH	POSITION	FUNCTION
1	ON	Enable RUN mode @ Power-up (*default*)
	OFF	Enable IDLE mode @ Power-up
2	ON	Enable SUBNET16 mode @ POWER-UP (*default*)
	OFF	Enable MUX32 mode @ Power-up
3	ON	Reset IP Address @ Power-up
	OFF	DO NOT Reset IP Address @ Power-up (*default*)
4	ON	Reserved
	OFF	*Default*
5	ON	Reserved
	OFF	*Default*
6	ON	Reserved
	OFF	*Default*
7	ON	*Default*
	OFF	Reserved
8	ON	*Default*
	OFF	Reserved

Table 2-3: DIP-Switch Position Status Table

NOTE: Use only the first three DIP-switches (switches 1, 2 and 3) to select the options listed above.

2.4 EXTERNAL CONNECTORS

The Gateway has the following external connectors:

§ **Ethernet Connector** (standard 8-pin, RJ45 socket)

§ **Power Connector** (3-pin, screw-down terminal block)

§ **Subnet16™ Connector** (5-pin, female, M12)

NOTE: The external power connector on the Gateway (*Figure 2-3*) is for testing purposes only.

2.4.1 Power and Ethernet Connectors

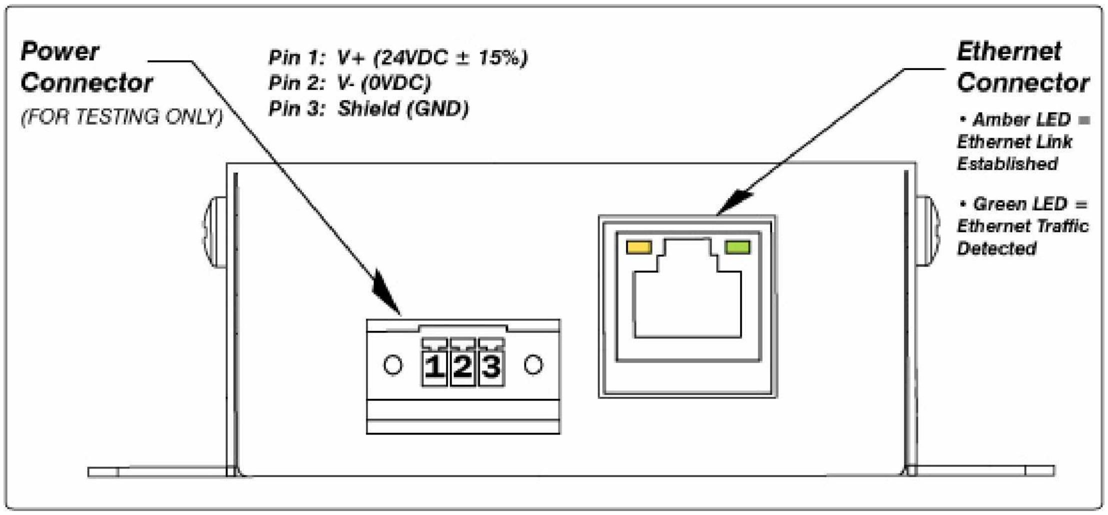


Figure 2-3: Power and Ethernet Connectors

2.4.2 Subnet16 Connector Pin Descriptions

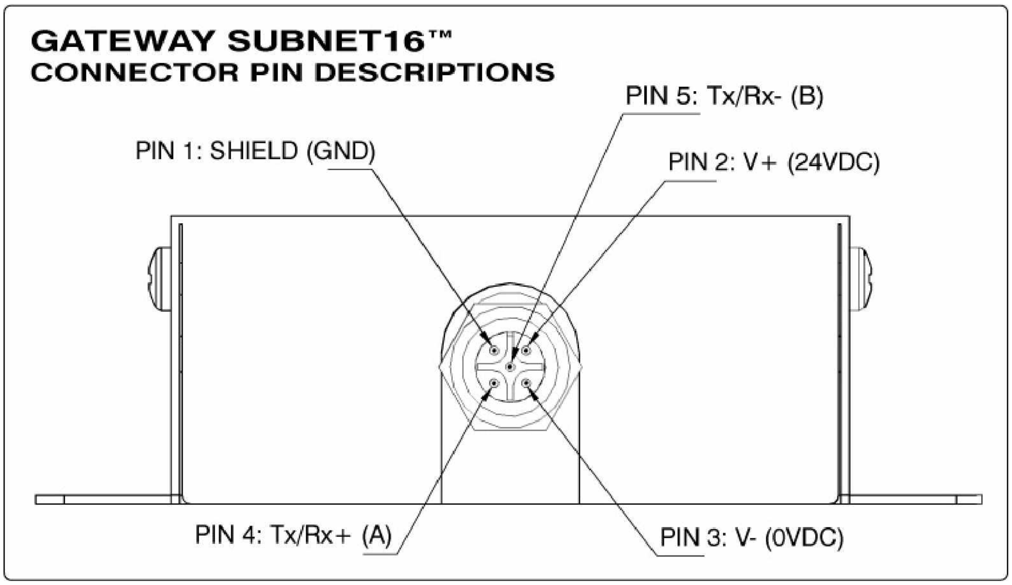


Figure 2-4: Subnet16 Connector Pin Descriptions

2.5 POWER & WIRING

The information presented below is provided to assist the installer in determining the amount of power that will be required by the Gateway and its Subnet network.

2.5.1 Power Requirements

The Gateway requires a power supply capable of providing 120mA @ 24VDC ± 15% (2.88W). However, each RFID controller connected on the Subnet will also require power. Power is applied directly to the trunk and is distributed through drop cables to the Gateway and RFID controllers (*see Appendix C for network diagrams*).

The drop in cable voltage must also be considered when determining the power supply to use (*see Section 2.5.3 below)*.

2.5.2 Total System Current Consumption

Maximum Gateway Current = 120mA @ 24VDC (2.88W)

Maximum RFID Controller Current = (refer to controller's specifications for more information)

§ HF-0405-485 controllers = 150mA

§ C0405-485 controllers = 80mA

§ C1007-485 controllers = 150mA

§ Cobalt HF-CNTL-485 controllers = 400mA

Maximum Current Rating for Gateway Power Feed = 2.5A

CALCULATING TOTAL SYSTEM CURRENT CONSUMPTION

To calculate the total amount of current required to operate the Gateway and any number of attached RFID controllers, follow the formula below.

Total Current Consumption = [*Maximum Gateway Current* + (*Maximum Controller Current* x *Number of Controllers*)] x 1.1 (to add a 10% safety margin)

TOTAL SYSTEM CURRENT CONSUMPTION EXAMPLE

For a Subnet16 Gateway network with eight HF-0405-485 RFID Controllers:

Total Current Required = [0.120A + (0.150A X 8)] x 1.1 = **1.452A**

2.5.3 Cable Voltage Drop

In addition, each RFID controller on the Subnet will experience a certain amount of voltage drop depending on the length of the cable.

CABLE RESISTANCE PER METER

§ **ThinNet** = .05413 ohms per meter

§ **ThickNet** = .0105 ohms per meter

CALCULATING VOLTAGE DROP

Voltage Drop = [(*Max Controller Current* **X** *Number of Controllers)* **X** (*Cable Resistance per Meter* **X** *Cable Length in Meters*)]

Voltage Drop Calculation Example

For a ThinNet network with eight HF-0405-485 RFID Controllers at a cable length of 20 meters:

Voltage Drop = (0.150A x 8) X (.05413 X 20) = **1.299VDC**

It is recommended that the voltage drop calculation be conducted on the RFID controller that is farthest from the Gateway, as it will experience the greatest voltage drop.

Current Rating for Cables

- ThinNet Cable Current Rating = 6.4A
- ThickNet Cable Current Rating = 17.6A (power and ground); 13.6A (data)
- 12mm Connector Current Rating = 3A

2.5.4 Subnet16 Gateway Hardware Details

The Gateway incorporates an ARM7 micro-controller, a power supply circuit (with protected input and output circuits), a real- time clock and an opto-isolated Subnet16 interface (with diagnostic functionality). Subnet16 serial communication is protected by EMI filters and diodes for fault tolerance.

The Gateway power supply circuit accepts 24VDC +/- 15% input and has protection for over voltage and miss wiring. The input voltage is isolated and regulated as required by the circuit. It should be used only for testing purposes.

The outer housing of the Gateway is a fabricated stainless steel enclosure that is rated NEMA 1 and IP30. A mounting bracket is incorporated into the enclosure to provide ease of installation. Four Philips head screws secure the cover to the base. Removing these four screws and the cover is required to access the DIP-switch block.

NOTE:

IP30 – Ingress Protection against solid objects over 2.5mm (tools and wires)

-Ingress Protection (IP) ratings are developed by the European Committee for Electro Technical Standardization (CENELEC) - IEC/EN Publication 60529

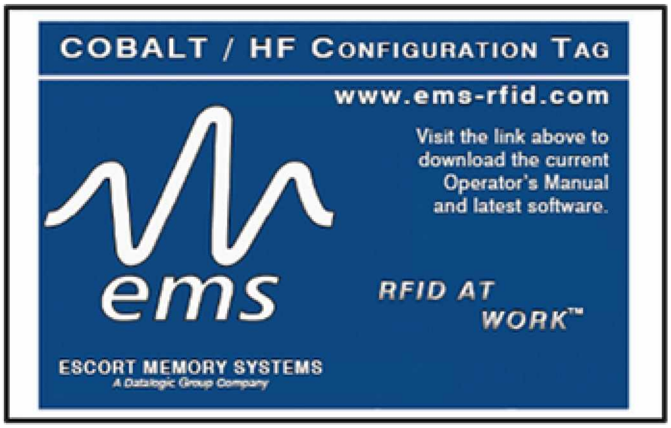
NEMA Type 1 - Enclosures are constructed for indoor use, provide a degree of protection to personnel against incidental contact with the enclosed equipment and provide a degree of protection against falling dirt.

- NEMA Standards Publication 250-2003

2.6 Node ID Configuration & Management

Only RS485-based RFID controllers can be connected to a Gateway's Subnet network and each must be assigned a unique Node ID value between 1 and 16.

When an RFID controller is connected to the Gateway's Subnet network, the Gateway will query the new controller to obtain certain configuration values (specifically the Node ID number). If the Gateway does not detect a Node ID conflict, it will "allow" the RFID controller onto the Subnet network.



COBALT / HF CONFIGURATION TAG

www.ems-rfid.com

Visit the link above to download the current Operator's Manual and latest software.

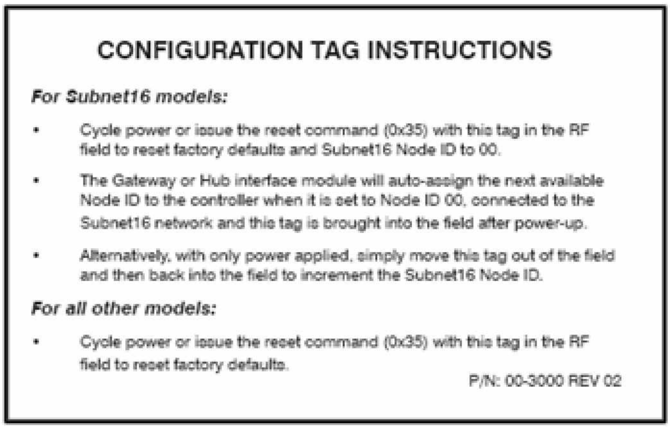
ems

RFID AT WORK™

ESCORT MEMORY SYSTEMS
A Datalogic Group Company

By using the ***Cobalt HF Configuration Tag*** that is included with each RS485-based Cobalt and HF-Series RFID Controller, the Node ID value can be dynamically assigned by the Gateway or can be manually assigned by the user.

For the Gateway to dynamically assign a Node ID value to a controller, the controller must first be initialized with the Node ID value of zero. This is the equivalent of having no Node ID assigned (note: all EMS RS485-based controllers ship with their Node ID value set to 0).



CONFIGURATION TAG INSTRUCTIONS

For Subnet16 models:

- Cycle power or issue the reset command (0x35) with this tag in the RF field to reset factory defaults and Subnet16 Node ID to 00.
- The Gateway or Hub interface module will auto-assign the next available Node ID to the controller when it is set to Node ID 00, connected to the Subnet16 network and this tag is brought into the field after power-up.
- Alternatively, with only power applied, simply move this tag out of the field and then back into the field to increment the Subnet16 Node ID.

For all other models:

- Cycle power or issue the reset command (0x35) with this tag in the RF field to reset factory defaults.

P/N: 00-3000 REV 02

When a controller (that is set to Node ID 0) is connected to the Subnet, it will not initially be recognized by the Gateway until the Configuration Tag is placed in the antenna's RF field and power to the controller is cycled. A few seconds after power is cycled to the controller while the Configuration Tag is in RF range, the controller will display its new assigned Node ID value in binary code from right to left using the five amber LEDs on the controller.

When dynamically assigning a Node ID value for a new controller, the Gateway will either assign the next available Node ID value or the value that the Gateway recognizes as offline or "*missing*" – that is, a Node ID value that previously existed, but has since disappeared from the network.

Because the Gateway stores a backup of each Subnet Node's configuration, should an RFID controller ever fail, a replacement controller can be installed quickly and easily. The new controller will be automatically assigned the same Node ID value and configuration as the replaced controller.

2.7 GATEWAY AND SUBNET NODE NAMING

The Gateway can store a 64-byte ASCII string for each of the 16 Subnet Nodes and one 64-byte ASCII string for the Gateway itself. These text strings can be used to assign logical or *"user friendly"* names to the Gateway and its Subnet Nodes.

For example, you could assign the Gateway a logical name such as "*PRODUCTION LINE 1*" and then name the controller connected to Subnet Node 01 "*PRODUCTION STATION 1.*" The controller at Subnet Node 02 could then be named "*PRODUCTION STATION 2*" (and so forth).

Gateway and Node names can be retrieved and edited by issuing specific commands to the Gateway (which are covered later in this manual). See the table below for specific CBx protocol command ID numbers.

Gateway and Node Naming – CBx Command IDs

	GATEWAY	NODE
GET NAME	Command 0x11	Command 0x30
SET NAME	Command 0x21	Command 0x40

Table 2-4: Gateway and Node Naming – CBx Command IDs

Gateway and Node naming can also be accomplished through the *Cobalt HF TCP/IP Dashboard* software utility (*see* Chapter 2, Section 2.8.1, "Cobalt HF TCP/IP Dashboard" for information).

2.8 CONFIGURATION TOOLS

Escort Memory Systems offers the following powerful RFID configuration utilities for Microsoft Windows XP and Windows 2000 based systems:

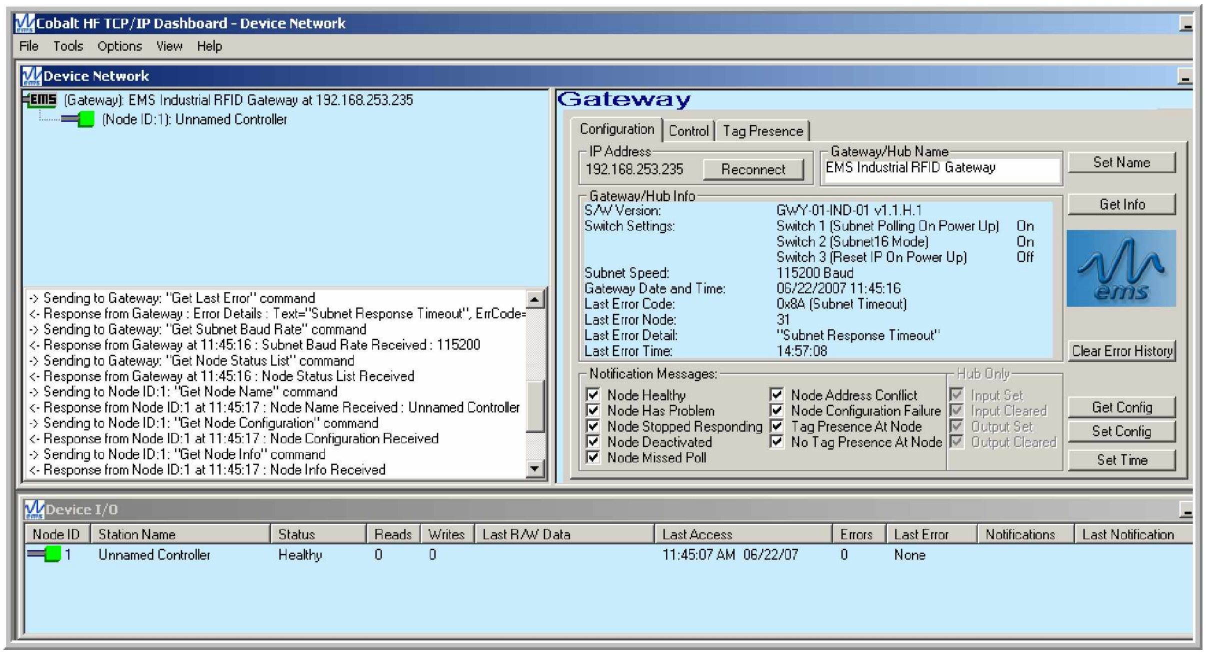
- **Cobalt HF TCP/IP Dashboard**
- **C-Macro Builder**

These configuration tools can be downloaded from the Escort Memory Systems website:

http://www.ems-rfid.com/

2.8.1 Cobalt HF TCP/IP Dashboard

The **Cobalt HF TCP/IP Dashboard™** is a Windows-based software application that provides users with complete control over their EMS RFID Solution. Users can monitor their entire RFID system, from the tag level, to the RFID controller, to the Gateway, and to the host.



(Gateway): EMS Industrial RFID Gateway at 192.168.253.235
(Node ID:1): Unnamed Controller

```
-> Sending to Gateway: "Get Last Error" command
<- Response from Gateway : Error Details : Text="Subnet Response Timeout", ErrCode=
-> Sending to Gateway: "Get Subnet Baud Rate" command
<- Response from Gateway at 11:45:16 : Subnet Baud Rate Received : 115200
-> Sending to Gateway: "Get Node Status List" command
<- Response from Gateway at 11:45:16 : Node Status List Received
-> Sending to Node ID:1: "Get Node Name" command
<- Response from Node ID:1 at 11:45:17 : Node Name Received : Unnamed Controller
-> Sending to Node ID:1: "Get Node Configuration" command
<- Response from Node ID:1 at 11:45:17 : Node Configuration Received
-> Sending to Node ID:1: "Get Node Info" command
<- Response from Node ID:1 at 11:45:17 : Node Info Received
```

IP Address: 192.168.253.235
Gateway/Hub Name: EMS Industrial RFID Gateway

Gateway/Hub Info

S/W Version:	GWY-01-IND-01 v1.1.H.1	
Switch Settings:	Switch 1 (Subnet Polling On Power Up)	On
	Switch 2 (Subnet16 Mode)	On
	Switch 3 (Reset IP On Power Up)	Off
Subnet Speed:	115200 Baud	
Gateway Date and Time:	06/22/2007 11:45:16	
Last Error Code:	0x8A (Subnet Timeout)	
Last Error Node:	31	
Last Error Detail:	"Subnet Response Timeout"	
Last Error Time:	14:57:08	

Notification Messages:

☑ Node Healthy
☑ Node Has Problem
☑ Node Stopped Responding
☑ Node Deactivated
☑ Node Missed Poll
☑ Node Address Conflict
☑ Node Configuration Failure
☑ Tag Presence At Node
☑ No Tag Presence At Node

Hub Only

☑ Input Set
☑ Input Cleared
☑ Output Set
☑ Output Cleared

Node ID	Station Name	Status	Reads	Writes	Last R/W Data	Last Access	Errors	Last Error	Notifications	Last Notification
🟩 1	Unnamed Controller	Healthy	0	0		11:45:07 AM 06/22/07	0	None		

Figure 2-3: Cobalt HF TCP/IP Dashboard

NOTE: Users should close the Dashboard utility before attempting EtherNet/IP communications between the Industrial Gateway and a host PLC processor.

Cobalt HF TCP/IP Dashboard Features:

§ Complete Subnet Node configuration

§ Data packet inspection and Subnet network health monitoring

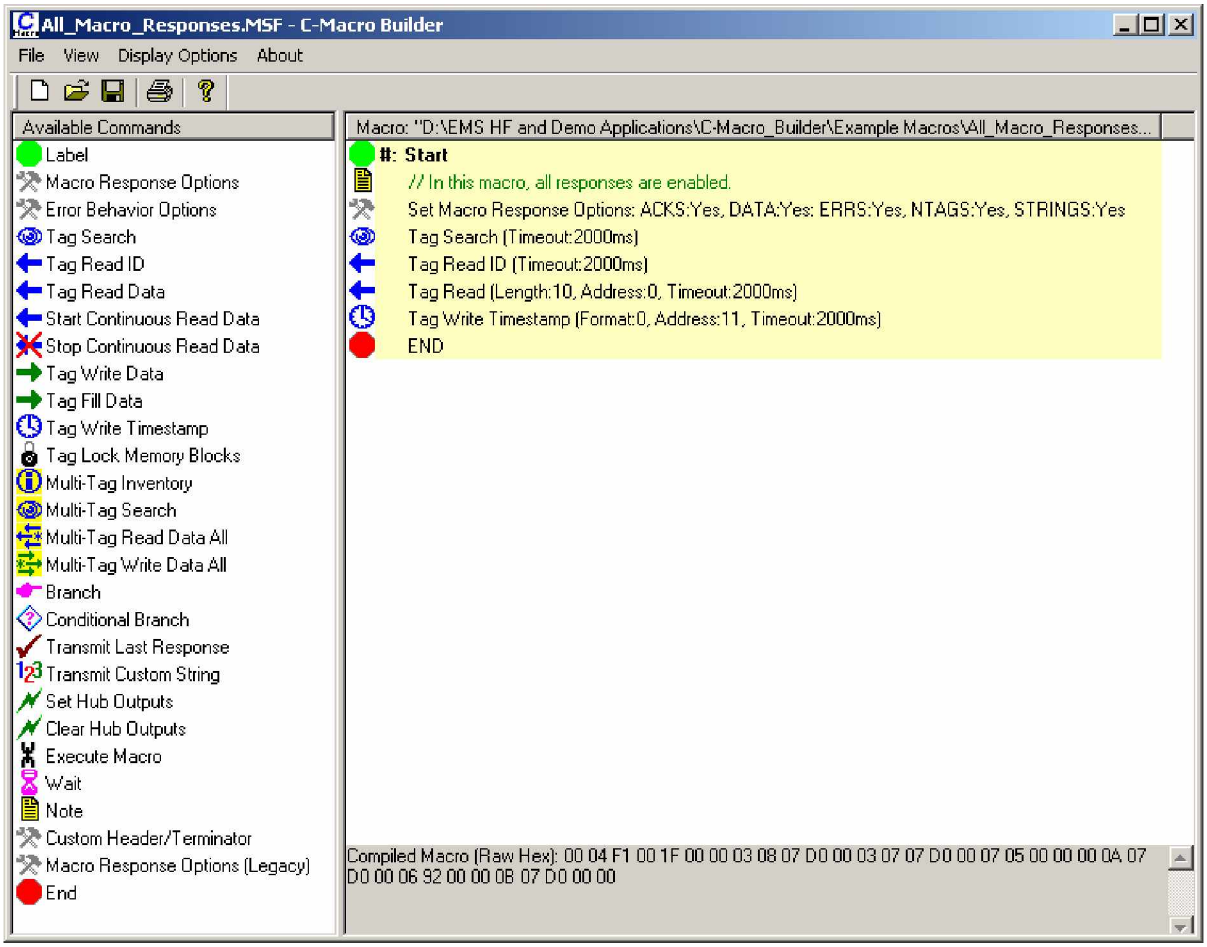
§ Software downloading and firmware upgrade installation routines

§ Gateway and Subnet Node "Friendly" Name Assignment (users can quickly and easily assign logical "friendly" names to the Gateway and its Subnet Nodes).

§ Supports Ethernet, USB, RS485, RS232 and RS422 interfaces

2.8.2 C-Macro Builder

C-Macro Builder™ is an easy to use GUI-driven utility for Windows that allows users to create powerful RFID command macro programs.



```
Macro: "D:\EMS HF and Demo Applications\C-Macro_Builder\Example Macros\All_Macro_Responses...
#: Start
    // In this macro, all responses are enabled.
    Set Macro Response Options: ACKS:Yes, DATA:Yes: ERRS:Yes, NTAGS:Yes, STRINGS:Yes
    Tag Search (Timeout:2000ms)
    Tag Read ID (Timeout:2000ms)
    Tag Read (Length:10, Address:0, Timeout:2000ms)
    Tag Write Timestamp (Format:0, Address:11, Timeout:2000ms)
    END

Compiled Macro (Raw Hex): 00 04 F1 00 1F 00 00 03 08 07 D0 00 03 07 07 D0 00 07 05 00 00 00 0A 07
D0 00 06 92 00 00 0B 07 D0 00 00
```

Figure 2-4: C-Macro Builder

When used in conjunction with the Cobalt HF Dashboard, users can easily create, download, erase, backup and manage multiple RFID command macros and macro triggers for each Subnet Node.

See *Chapter 3, "RFID Command Macros,"* for more on macros.

NOTE:

For specific information regarding the configuration and use of either of these utilities, please see the accompanying documentation included when downloading each software application.

CHAPTER 3: RFID COMMAND MACROS

What are RFID Command Macros?

RFID Command Macros are a powerful feature of EMS' Cobalt and HF-0405-Series Controllers. Macros are simple programs that direct a controller to execute multiple pre-programmed instructions.

Because macros reside within the Cobalt or HF-0405-Series Controller's internal memory, they can be programmed to instruct the controller to automatically read and/or write a specified set of data to an RFID tag without the controller ever having to receive a command from the host. In fact, the controllers do not even require a connection to a host in order to execute macros.

Each macro can contain up to 255 bytes of data and each supported controller can store up to eight macros at a time. Though they are stored locally on the controller, macros are also backed up in the Gateway's flash memory as well.

Why use macros?

The power of macros is in distributed intelligence, the reduction in network bus traffic and the ability to accelerate routine decision making at the point of data collection.

What can macros do?

In addition to the automated reading and writing of data, macro capabilities include:

- The ability to write time stamps to RFID tags
- The ability to filter command responses to only those of interest to the host (such as when an error occurs or when a tag has arrived in the RF field)
- The ability to harness powerful logic and triggering capabilities such as; read, write, start/stop continuous read, data compare, branch, transmit custom string, and set outputs.

What is a macro trigger?

Macros are initiated by "triggers." Triggers can be configured in numerous ways. A simple command from the host, such as "*execute macro number three*" can be considered a trigger.

Triggers can be configured, for example, to activate a macro when a tag enters or leaves a controller's RF field.

EMS RFID controllers can store up to eight separate triggers in addition to the eight macros they can also house. Any trigger can activate any of the eight stored macros.

How are macros created?

Macros are created using the powerful, yet simple, C-MacroBuilder™ utility from EMS. The tool's easy to use GUI allows the user to create powerful RFID macro programs quickly and easily.

When used with EMS' Cobalt HF Dashboard™ utility, users can effortlessly download, erase, and manage their macros and triggers, as well as set the operational configurations of their RFID controllers and Subnet16™ Gateways.

Which communication interfaces support the use of macros?

Macros (and the Dashboard and C-Macro Builder utilities) support communications with Cobalt and HF-0405-Series Controllers across Ethernet, USB, RS232, RS422 and RS485 interfaces.

What happens to existing Macros if a controller must be replaced?

When using a Subnet16 Gateway, users do not need to worry. Macros and triggers normally residing in an RFID controller's *flash* memory are always backed up in the Gateway's *flash* memory as well. Therefore, if a controller should ever require replacement, all existing macro and trigger settings are automatically exported from the Gateway to the new RFID controller.

In short, when an RFID controller is initially connected to the Gateway, macro and trigger data from the controller's flash memory is compared to the macro and trigger data backed up in the Gateway from the previous RFID controller. If the data does not match that which is stored on the Gateway, the controller's flash memory will be overwritten with the backed up data stored in the Gateway's flash memory.

How can I learn more about the Dashboard and C-Macro Builder?

More information regarding macros, triggers, uploading, downloading, configuring and monitoring EMS RFID equipment is available in the respective User's Manuals for these products, which are available on the EMS website at: *www.ems-rfid.com*.

CHAPTER 4: COMMAND MAPPING

A command is initiated by a host PC or *Programmable Logic Controller* (PLC) and is distributed to the Gateway over an Ethernet connection. Once issued, the command is then executed directly by the Gateway or is otherwise routed to the appropriate RFID controller (specified by its numerical "*Node ID*" value, for which there are 16).

In general, there are two types of commands that can be issued:

§ **Controller Commands -** commands intended for one of the attached RFID controllers. "*Read Data*" and "*Write Data*" are two common controller commands.

§ **Gateway Commands** - commands intended for the Gateway itself. Gateway commands are those commands that query the Gateway for information or instruct the Gateway to perform a task. The commands "*Get Node Status List*" and "*Set Notification Mask*" are examples of Gateway commands.

4.1 MAPPING COMMAND & RESPONSE DATA

In order to properly distribute RFID commands to the intended RFID controller (and to correctly return the controller's response to the host), the Gateway separates input data (*Commands*) from output data (*Responses*) by assigning each controller a *Node Input Page* and a *Node Output Page*. Each "*page*" is actually a small block of memory used to temporarily hold command or response data.

4.1.1 Node Input Pages

The Gateway assigns each controller a ***Node Input Page*** that corresponds numerically to its assigned Node ID value. Thus, commands directed to the controller at Node 01, for example, will be written to Node Input Page 01. Node Input Pages hold Gateway and controller-bound command data generated by the host.

NODE INPUT PAGE VALUES

Nodes 01 thru 16 are assigned Node Input Pages 01 thru 16 respectively. The Gateway, itself, is assigned Node Input Page 32.

4.1.2 Node Output Pages

The Gateway also assigns each controller a ***Node Output Page*** used to hold host-bound response data generated by the Gateway or one of the RFID controllers after executing a command.

A controller's numeric Node Output Page value is always ***32*** page values greater than its corresponding Node Input Page value. Therefore, response data from the controller at Node 01, for example, will be written to Node Output Page 33 (as Node Output Page 33 is 32 page values greater than its corresponding Node Input Page of 01).

NODE OUTPUT PAGE VALUES

Nodes 01 thru 16 are assigned Node Output Pages 33 thru 48 respectively. The Gateway is assigned Node Output Page 64.

4.1.3 Node Input/Output Page - Allocation Formula

In the example above, a command was directed to the RFID controller at Node 01. Command packet data was written to Node Input Page 01 and the controller's response packet data was written to Node Output Page 33.

Therefore, command packet data intended for **Node ID [N]** will be written to **Node Input Page [N].** Corresponding response packet data will be written to **Node Output Page [N + 32]**.

4.1.4 Node Input/Output Page Table

NODE ID	NODE INPUT PAGE	NODE OUTPUT PAGE
Node 01	Node Input Page 01	Node Output Page 33
Node 02	Node Input Page 02	Node Output Page 34
Node 03	Node Input Page 03	Node Output Page 35
Node 04	Node Input Page 04	Node Output Page 36
Node 05	Node Input Page 05	Node Output Page 37
Node 06	Node Input Page 06	Node Output Page 38
Node 07	Node Input Page 07	Node Output Page 39
Node 08	Node Input Page 08	Node Output Page 40
Node 09	Node Input Page 09	Node Output Page 41
Node 10	Node Input Page 10	Node Output Page 42
Node 11	Node Input Page 11	Node Output Page 43
Node 12	Node Input Page 12	Node Output Page 44
Node 13	Node Input Page 13	Node Output Page 45
Node 14	Node Input Page 14	Node Output Page 46
Node 15	Node Input Page 15	Node Output Page 47
Node 16	Node Input Page 16	Node Output Page 48

Table 4-1: Node Input/Output Page Table

4.1.5 Gateway Node Input/Output Page Table

Commands intended for the Gateway are written to Node Input Page 32 and Gateway responses are written to Node Output Page 64.

NODE ID	NODE INPUT PAGE	NODE OUTPUT PAGE
Node 32	Node Input Page 32	Node Output Page 64

Table 4-2: Gateway Node Input/Output Page Table

4.1.6 Data Producer vs. Data Consumer

Each time a command is issued or a response is generated, data is being produced and consumed. For every data packet transmitted across the network, there is a *Data Producer* and *Data Consumer*.

§ ***Data Producer*** refers to the physical piece of hardware on the network that initiated or generated the data packet.

§ ***Data Consumer*** refers to the physical piece of hardware for which a generated data packet is intended (the data recipient).

For example, when the host issues an RFID command, it is said to be acting as a Data Producer. However, when the host retrieves a response packet, the host is said to be acting as a Data Consumer.

When an RFID controller executes an assigned command, it is said to be acting as a Data Consumer. Yet when the same controller generates its response packet, it is acting as a Data Producer.

4.2 CBx Command Protocol

The ***CBx Command Protocol*** is an advanced protocol that supports Multi-drop Subnet16 networking with TCP/IP and Industrial Ethernet applications. It is based on a double-byte oriented packet structure where commands always contain a minimum of six data "words" (12 bytes) - even when one or more packet elements are not applicable to the command.

The CBx packet structures described herein are protocol independent and can be implemented the same for all protocols (Ethernet/IP, Modbus TCP, etc.). Note that the first word in the example below (CBx Header and Node ID) is not required for PLC based RFID applications (Modbus TCP, Ethernet/IP, etc.)

4.2.1 CBx - Command Packet Structure

Below is the packet structure of a standard CBx command.

WORD #	COMMAND PACKET ELEMENT	MSB	LSB
00	**CBx Header** in MSB: 0xFF **Node ID** in LSB	0xFF	<Node ID>
01	**Overall Length**: 2-byte value indicating the number of "***words***" in the command packet. This value will always be at least ***6 words.***	0x00	0x06 + (number of any additional words)
02	**Command ID**: 0xAA + 1-byte value indicating command to perform.	0xAA	<Command ID>
03	**0x00** in MSB, **Node ID** in LSB	0x00	<Node ID>
04	**Timeout Value**: 2-byte integer for the length of time allowed for the completion of the command, measured in 1 millisecond units, where *0x07D0 = 2000 x .001 = 2 seconds.*	0x07	0xD0
05	**Start Address**: 2-byte integer indicating the location of tag memory where a Read/Write operation will begin (when applicable).	<Start MSB>	<Start LSB>
06	**Block Size**: 2-byte integer indicating the number of bytes that are to be read from or written to a tag beginning at the specified Start Address (when applicable).	<Length MSB>	<Length LSB>
07	**Additional Data Byte Values 1 & 2**: holds 2 bytes of data used for fills, writes, etc. (when applicable)	<D1>	<D2>
08	**Additional Data Byte Values 3 & 4**: (when applicable)	<D3>	<D4>

Table 4-3: CBx Command Packet Structure

4.2.2 CBx - Response Packet Structure

After performing a command, the Gateway or RFID controller will issue a host-bound response packet. Below is the packet structure of a standard CBx response message.

WORD #	RESPONSE PACKET ELEMENT	MSB	LSB
00	**CBx Header** in MSB: 0xFF **Node ID Echo** in LSB *NOTE: These first two bytes will not be returned in the response packet when a command is performed by Node 01*	0xFF	<Node ID Echo>
01	**Overall Length**: 2-byte integer indicating the number of "***words***" in the response packet. This value will always be at least ***6*** (+ number of any data words retrieved).	0x00	0x06 + (number of any retrieved words)
02	**0xAA** in MSB **Command Echo** in LSB	0xAA	<Command Echo>
03	**Instance Counter**: 1-byte value indicating the number of responses generated by the Node ID identified in the LSB (*see details below*). **Node ID Echo**: 1-byte value indicating the Node ID of the controller that performed the command.	<IC>	<Node ID Echo>
04	**Month and Day Timestamp**	<Month>	<Day>
05	**Hour and Minute Timestamp**	<Hour>	<Minute>
06	**Second Timestamp** in MSB **Additional Data Length** in LSB: 1-byte value indicates number of additional bytes retrieved.	<Second>	<N-bytes>
07	**Retrieved Data Bytes 1 and 2:** holds 2 bytes of retrieved data from tag reads, serial numbers, etc. (when applicable)	<D1>	<D2>

Table 4-4: CBx - Response Packet Structure

INSTANCE COUNTER

The ***Instance Counter*** is a one-byte value used by the Subnet16 Gateway to track the number of responses generated by the controller at a given Node ID. The Gateway tallies, in its internal RAM, separate Instance Counter values for each Node ID.

A Node ID's Instance Counter will be incremented by *one* following each response. If, for example, the controller at Node 01 has generated 10 responses, its Instance Counter value will read 0x0A. However, when the Gateway is rebooted or power-cycled, the Instance Counter values for all Node IDs will be reset to zero (0x00).

4.2.3 CBx - Controller Command Example

In the example below, *Command 0x05 (Read Data*) is issued to the RFID controller at Node 01. The controller is instructed to read four bytes of data from a tag beginning at tag address 0x0020. The Timeout Value, measured in milliseconds, is set for two seconds for the completion of this command (*0x07D0 = 2000 x .001 = 2 seconds*).

WORD	PACKET ELEMENET	MSB	LSB
00	**CBx Header** in MSB: *0xFF* **Node ID** in LSB	0xFF	0x01
01	**Overall Length of Command** (*in "words*")	0x00	0x06
02	**0xAA** in MSB **Command ID** in LSB: (*0x05 - Read Data)*	0xAA	0x05
03	**0x00** in MSB **Node ID** in LSB: (*0x01*)	0x00	0x01
04	**Timeout Value**: (*2 seconds*)	0x07	0xD0
05	**Start Address**: (*0x0020*)	0x00	0x20
06	**Read Length**: (*4 bytes*)	0x00	0x04

4.2.4 CBx - Controller Response Example

Below is a controller's response after successfully completing the *Read Data* command (as issued in the previous example).

Because this example command was performed by the controller at Node 01, the ensuing response will NOT contain the two-byte CBx Header of 0xFF and the Node ID in the first word, as would be the case for all other Node ID responses.

WORD	PACKET ELEMENT	MSB	LSB
01	**Overall Length of Response** (*in "words*")	0x00	0x08
02	**0xAA** in MSB **Command Echo** in LSB: (*0x05 - Read Data)*	0xAA	0x05
03	**Instance Counter** in MSB **Node ID Echo** in LSB	<IC>	0x01
04	**Month and Day Timestamp** (*March 19th*)	0x03	0x13
05	**Hour and Minute Timestamp** (*10:11: AM*)	0x0A	0x0B
06	**Seconds Timestamp** in MSB: (*36 seconds*) **Additional Data Length** in LSB (*4 bytes*)	0x24	0x04
07	**Retrieved Data** (*bytes 1 and 2*)	0x01	0x02
08	**Retrieved Data** (*bytes 3 and 4*)	0x03	0x04

4.2.5 CBx - Gateway Command Example

In this example, the host issues Command 0x13 (*Get Node Status List),* which retrieves from the Gateway, a list that indicates the operating status of the 16 Nodes.

WORD	DESCRIPTION	MSB	LSB
00	**CBx Header** in MSB: *0xFF* **Node ID** in LSB	0xFF	0x20
01	**Overall Length of Command** (*in "words*")	0x00	0x06
02	**0xAA** in MSB **Command ID** in LSB: (*0x13 - Get Node Status List*)	0xAA	0x13
03	**0x00** in MSB **Node ID** in LSB: (*0x20 = Gateway Node 32*)	0x00	0x20
04	**Not Used**: 0x00, 0x00 (*default*)	0x00	0x00
05	**Not Used**: 0x00, 0x00 (*default*)	0x00	0x00
06	**Not Used**: 0x00, 0x00 (*default*)	0x00	0x00

Note that even though the last three words (6 bytes) of this command are not used, these parameters still require zero's (*0x00, 0x00*) and are to be included when calculating *Overall Length*.

4.2.6 CBx - Gateway Response Example

Below is the Gateway response to the command "*Get Node Status List"* (as issued in the previous example).

WORD	DESCRIPTION	MSB	LSB
00	**CBx Header** in MSB: *0xFF* **Node ID Echo** in LSB	0xFF	0x20
01	**Overall Length of Response** (*in "words," not including the previous 2-bytes – CBx Header and Node ID Echo)*	0x00	0x0E
02	**0xAA** in MSB **Command Echo** in LSB: (*0x13*)	0xAA	0x13
03	**Instance Counter** in MSB **Node ID Echo** in LSB (*0x20 = Gateway Node 32*)	<IC>	0x20
04	**Month and Day Timestamp** (*March 19th*)	0x03	0x13

05	**Hour and Minute Timestamp** (*10:11: AM*)	0x0A	0x0B
06	**Seconds Timestamp** in MSB: (*36 seconds*) **Additional Data Length** in LSB: (*16 bytes*)	0x24	0x10
07	**Status of Node ID 1 and 2**	0x00	0x00
08	**Status of Node ID 3 and 4**	0x04	0x00
09	**Status of Node ID 5 and 6**	0x00	0x04
0A	**Status of Node ID 7 and 8**	0x00	0x00
0B	**Status of Node ID 9 and 10**	0x00	0x00
0C	**Status of Node ID 11 and 12**	0x00	0x00
0D	**Status of Node ID 13 and 14**	0x00	0x00
0E	**Status of Node ID 15 and 16**	0x00	0x00

In the above example, the *Node Status Byte* "0x04" (meaning "*Controller Healthy*") was reported for Nodes 03 and 06, indicating that the Gateway recognizes functioning RFID controllers at Node 03 and Node 06. (See the *Node Status Byte Definition Table* below for more information).

Node Status Byte Definition Table

NODE STATUS BYTE	NODE STATUS	STATUS DESCRIPTION
0	CONTROLLER INACTIVE	The controller at this node has not responded to a poll for at least 40 seconds. If a controller does eventually respond at this Node ID, its status will be changed to "0x04 - CONTROLLER HEALTHY."
1	CONTROLLER STOPPED RESPONDING	The controller at this node has not responded to a poll in over 10 seconds. If the controller does not respond to a poll within another 30 seconds, its status will be changed to "0x00 - CONTROLLER INACTIVE." If the controller does eventually respond to a poll, its status will be changed back to "0x04 - CONTROLLER HEALTHY"

2	CONTROLLER HAS PROBLEM	The controller at this node has missed at least 3 consecutive polls. If the controller does not respond to a poll within another 10 seconds, its status will be changed to "0x01 - CONTROLLER STOPPED RESPONDING." If the controller does eventually respond to a poll, its status will be changed back to "0x04 - CONTROLLER HEALTHY."
3	CONTROLLER EXPECTED SOON	This Node Status indicates that a controller is temporarily disconnected or that it is being moved to another Node ID. Because a controller is "expected" to appear soon, the Gateway will poll this node more frequently than other 'inactive' nodes.
4	CONTROLLER HEALTHY	The controller at this node is healthy and responding to polls. However, if the controller misses 3 consecutive polls, its status will be changed to "0x02 - CONTROLLER HAS PROBLEM."
5	CONTROLLER DOWNLOADING	This status is only applied to a controller that is currently downloading and installing new firmware to its flash memory. To optimize polling and allow for the fastest possible firmware installation, the Gateway will temporarily halt the polling of this node until the controller has finished its installation.

Table 4-5: Node Status Byte Definition Table

4.3 Notification Messages

Notification Messages are small host-bound informational packets of data that are issued by the Gateway when a specified ***Notification Event*** (or series of events) occurs within the Gateway or on the Subnet network. For example, the Gateway can be configured to send the host a Notification Message when a controller is attached, or removed, or experiences a problem.

The Gateway stores nine different Notification Messages internally (all of which are disabled by default). A 16-bit value, called the ***Notification Mask***, controls which Notification Events trigger Notification Messages to the host. Bits 01 through 09 in the 16-bit Notification Mask correspond to the nine possible Notification Messages. The remaining 7 bits (bits 10-16) are not implemented at this time (default value is zero for each bit).

Notification Messages are enabled by changing the associated bit from zero to one within the Notification Mask. A bit is either set to "0" (OFF – disabled) or "1" (ON – enabled). When a bit is turned ON, the related Notification Message will be enabled. The next time the enabled Notification Event occurs, the corresponding Notification Message will be generated and immediately delivered to the host.

When a Notification Message is generated, it is written to the ***Node Output Page*** of the controller that triggered the Notification Event. Notification Messages include a one-byte value indicating which of the nine possible Notification Events occurred. Notification Messages also contain a one-byte value that identifies the affected Node ID.

For Notification Messages, a handshaking scheme of enabling and clearing a specific bit in the Output Data Ready Mask is implemented (as previously explained).

To enable all nine Notification Messages, the 2-byte Notification Mask would read: **0x01FF.**

16-bit Notification Mask - Binary Representation
-when enabling all nine Notification Messages:

(0 0 0 0 0 0 0 1) (1 1 1 1 1 1 1 1) = **0x01FF**
[Bit 16 -   - Bit 09]   [Bit 08 -   - Bit 01]

4.3.1 Notification Message Table

The following table contains a listing of the nine possible *Notification Messages.*

BIT	NOTIFICATION MESSAGE	EVENT DESCRIPTION
1	CONTROLLER IS HEALTHY	Sent whenever the status of a controller changes to '*Healthy*'
2	CONTROLLER HAS PROBLEM	Sent whenever a controller is marked '*Has Problem*'
3	CONTROLLER STOPPED RESPONDING	Sent whenever a controller is marked '*Stopped Responding*'
4	CONTROLLER DEACTIVATED	Sent whenever a controller is deactivated (is marked '*Inactive*')
5	CONTROLLER MISSED POLL	Sent whenever a controller misses a poll
6	CONTROLLER ADDRESS CONFLICT	Sent whenever the Gateway detects a Node ID conflict
7	CONTROLLER CONFIGURATION FAILURE	Sent whenever the Gateway fails to configure a controller
8	TAG PRESENT AT NODE*	Sent whenever a tag is first recognized in the RF field of the specified node
9	TAG NOT PRESENT AT NODE*	Sent when no tag is recognized or when a previously recognized tag is no longer acknowledged in the specified node's RF field

Table 4-6: Notification Message Table

* *Tag Presence* must be enabled on the RFID controller.

4.3.2 Notification Mask Example

In the following example, an RFID controller is attached to the Gateway's Subnet network. After power is applied to the controller, the Gateway immediately attempts to determine its Node ID (*Node 04 in this example*). After recognizing a stored Node ID configuration, the Gateway allows the device onto the Subnet network.

Now if bit 01 in the Notification Mask was enabled (set to one = ON), Notification Event 01 would be triggered and the Gateway would immediately write Notification Message 01 to Node Output Page 36 (the Node Output Page number for Node ID 04). The Notification Message would indicate that a new controller was recognized at Node 04 and is functioning properly (i.e. the controller is healthy).

If, on the other hand, the recently connected controller does not power-up, or fails to initialize properly, and bit 02 in the Notification Mask is enabled, Notification Event 02 will be triggered, in which case the Gateway will write Notification Message 02 to Node Output Page 36. This message informs the host that the controller at Node 04 is experiencing a problem.

4.3.3 Notification Message Packet Structure

DESCRIPTION	MSB	LSB
Overall Length of Notification Message (in *words*)	0x00	0x06
0xFE in MSB = Notification Message Flag **Notification Event** in LSB	0xFE	<Notification Event>
Instance Counter in MSB (*a Notification Message is considered a response; therefore the Instance Counter will be incremented by one*) **Node ID** in LSB (*04 for the above example*)	<IC>	0x04
Month and Day Timestamp	<Month>	<Day>
Hour and Minute Timestamp	<Hour>	<Minute>
Seconds Timestamp in MSB **0x00** in LSB	<Second>	0x00

Table 4-7: Notification Message - Packet Structure

CHAPTER 5: ETHERNET/IP INTERFACE

For Use with the GWY-01-IND-01 Industrial Gateway

NOTES:

- Users of the *Cobalt HF Dashboard* utility should exit the application before attempting communications between the Industrial Gateway and an EtherNet/IP host Programmable Logic Controller (PLC).
- When installing the HF-CNTL-IND-01 for communication over EtherNet/IP, the ODVA Guidelines for EtherNet/IP Media System installation should be followed (refer to www.odva.org, ODVA **PUB00148R0** (Pub 148), EtherNet/IP Media Planning and Installation Manual, 2006 ODVA).
- Follow ODVA recommendations for switching and wiring Ethernet/IP.
- If the Ethernet/IP network enables I/O Messaging for remote I/O, etc., or if other UDP traffic is present, then the Gateway must be protected by a switch that incorporates IGMP Snooping or a VLAN.

The GWY-01-IND-01 is designed to support many common Industrial Ethernet and Communications Protocols and can be implemented in a wide variety of existing host / PLC applications. One such popular Industrial Ethernet protocol is **Ethernet/IP** (EIP).

This chapter focuses on the process of setting up and configuring the Subnet16 Industrial Gateway to communicate (via Ethernet/IP) with a ControlLogix Programmable Logic Controller (PLC).

Also in this chapter are descriptions of EMS' *HTML Server* and *OnDemand Utilities*, as well as systematic instructions to help configure the Industrial Gateway for Ethernet/IP.

5.1 WHAT IS ETHERNET/IP?

Built on the standard TCP/IP protocol suite, EtherNet/IP is a high-level application layer protocol for industrial automation applications that uses traditional Ethernet hardware and software to define an application layer protocol that structures the task of configuring, accessing and controlling industrial automation devices.

Ethernet/IP classifies Ethernet nodes as predefined device types with specific behaviors. The set of device types and the EIP application layer protocol is based on the Common Industrial Protocol (CIP) layer used in both DeviceNet and ControlNet. Building on these two widely used protocol suites, Ethernet/IP provides a seamlessly integrated system from the RFID Subnet network to the host and enterprise networks.

Under most EtherNet/IP environments, the Gateway is configured as an EtherNet/IP client device, which will receive and distribute RFID commands issued by the PLC (acting as EtherNet/IP Server).

Plan to perform a test phase where you will construct a small scale, independent network that includes only the essential devices required to test your RFID application. To avoid possible interference with other devices, at first, do not connect your RFID testing environment to an existing office network.

5.1.1 HTML Server & OnDemand PLC Support

Below is a partial list of the programmable logic controllers that are supported by EMS' *HTML Server* and *OnDemand Utilities*.

- ControlLogix – OnDemand supports all current versions
- RA's PLC5E releases:

  Series C, Revision N.1

  Series D, Revision E.1

  Series E, Revision D.1
- PLC5 "Sidecar" Module Series B, Revision A with EIP support
- SLC5/05 releases:

  Series A with firmware revision OS501, FRN5

  All Series B and Series C PLC Controllers

5.1.2 Steps to Configure the Industrial Gateway

This chapter contains instructions regarding:

- Setting Gateway IP address via *HTML Server*
- Configuring Subnet Nodes via *OnDemand Utilities*
- Creating "*Controller Tags*" in PLC
- Verifying PLC and Subnet Node connectivity

5.2 IP CONFIGURATION VIA *HTML* SERVER

One of the first steps in configuring the Gateway is setting its IP address.

Embedded in the Gateway is an *HTML Server* that provides users with a Website-like interface containing tools used to configure the unit. The *HTML Server* is also used to access the Gateway's *OnDemand Utilities*, a suite of tools used to configure the internal parameters of the Gateway and its Subnet Nodes. The *OnDemand Utilities* will be used later in this manual.

NOTE:

All Subnet16 Industrial Gateways ship with the following factory default IP address configured:

192.168.253.110

5.2.1 HTML Server Overview

This section of the manual describes the IP configuration procedure via the Gateway's internal *HTML Server*. Using a standard Web browser, users can access the built-in *HTML Server* where they can modify and save changes to the Gateway's internal configuration settings (including its IP address).

OnDemand Utilities is also Escort Memory Systems' approach to adding *Change of State* messaging to Rockwell Automation's (RA) ControlLogix PLC and adding legacy support for the RA PLC5E and RA SCL5/05 PLCs.

ADDITIONAL INFORMATION:

The ControlLogix PLC refers to a "**tag**" as a small block of internal memory that is used to store outgoing (command) and incoming (response) data. Within each tag, information is stored in two-byte segments, known as registers or "words."

5.2.2 Setting the IP Address of the Gateway

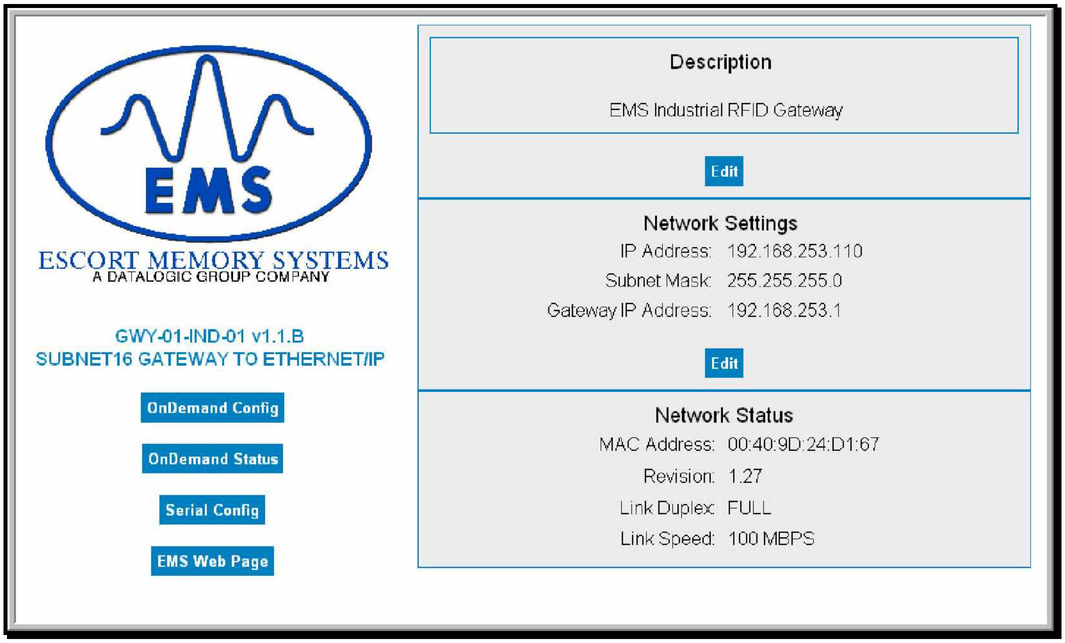
To set the Gateway's IP address using the *HTML Server,* follow the steps below:

1 Open a Web browser on the host PC.

2 In the URL address field, enter the Gateway's IP address (**192.168.253.110 = factory default).**

3 Press ENTER.

The *HTML Server - Main Page* will be displayed.



Description

EMS Industrial RFID Gateway

Network Settings

IP Address: 192.168.253.110
Subnet Mask: 255.255.255.0
Gateway IP Address: 192.168.253.1

Network Status

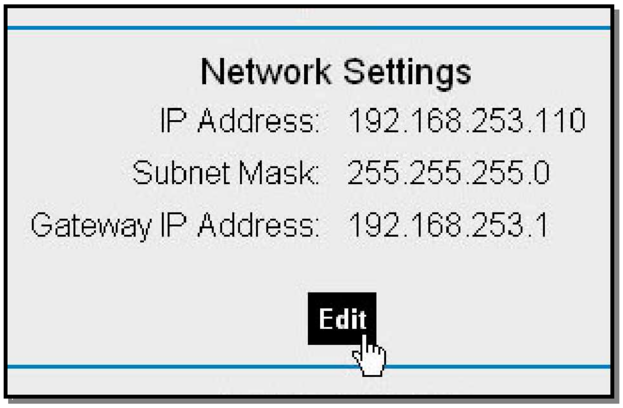
MAC Address: 00:40:9D:24:D1:67
Revision: 1.27
Link Duplex: FULL
Link Speed: 100 MBPS

Figure 5-1: The HTML Server - Main Page

The ***HTML Server - Main Page*** lists the IP address (as well as some other network parameters) as currently stored on the Gateway.

4 Click the button labeled "***EDIT***", located below "***Network Settings***".

The *IP Configuration Page* will be displayed.

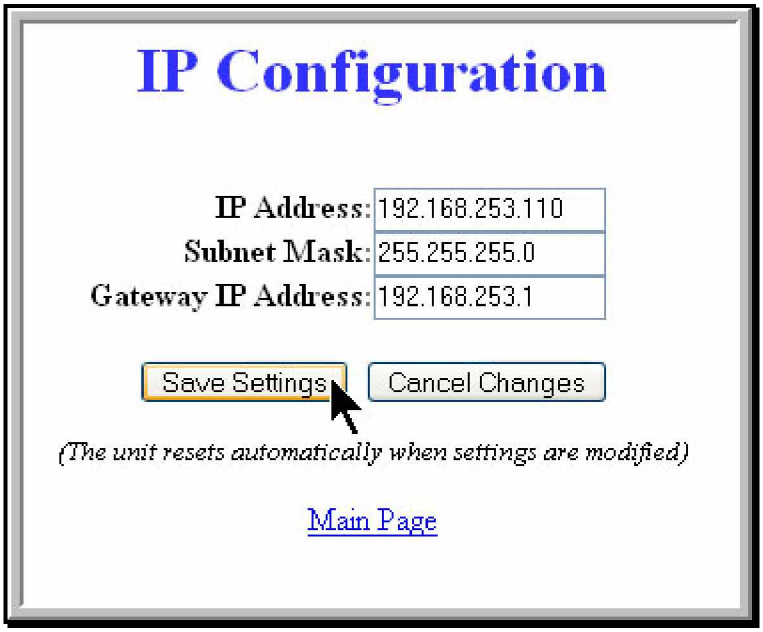


Network Settings

IP Address: 192.168.253.110
Subnet Mask: 255.255.255.0
Gateway IP Address: 192.168.253.1

IP Configuration Page

The *IP Configuration Page* is used to modify and save changes to the IP Address, Subnet Mask and (Network) Gateway IP Address.



IP Configuration

IP Address: 192.168.253.110
Subnet Mask: 255.255.255.0
Gateway IP Address: 192.168.253.1

(The unit resets automatically when settings are modified)

Figure 5-2: The IP Configuration Page

5 In the fields provided, enter the new IP configuration values for the Gateway.

6 Click the *'Save Settings'* button to store the new IP configuration. The Gateway will completely reset and your IP changes will be implemented.

7 After the Gateway has restarted, verify the new IP configuration by opening a Web browser and manually entering the Gateway's new IP address in the URL field. If successful, you should arrive back at the *HTML Server – Main Page*.

ATTENTION:

Disable any firewall services running on the PC. Firewalls can potentially block communications between the PC, the PLC and/or the Gateway.

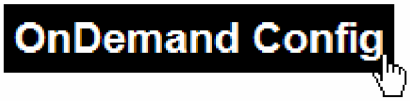
5.3 CONFIGURING SUBNET NODES

Now that you have configured the Gateway's IP address, use the HTML Server to access the Gateway's *OnDemand Utilities*, which are used to configure each Subnet Node that will be used. The Gateway supports the connection of up to 16 individual RFID controllers. Through the use of the *OnDemand Utilities,* the Gateway's Subnet Nodes will be linked to specific "Controller Tags" as defined in the ControlLogix PLC.

To configure the Gateway's Subnet Nodes, follow the steps below:

1 Open a Web browser and enter the Gateway's *new IP address* in the URL field. The *HTML Server – Main Page* will be displayed.

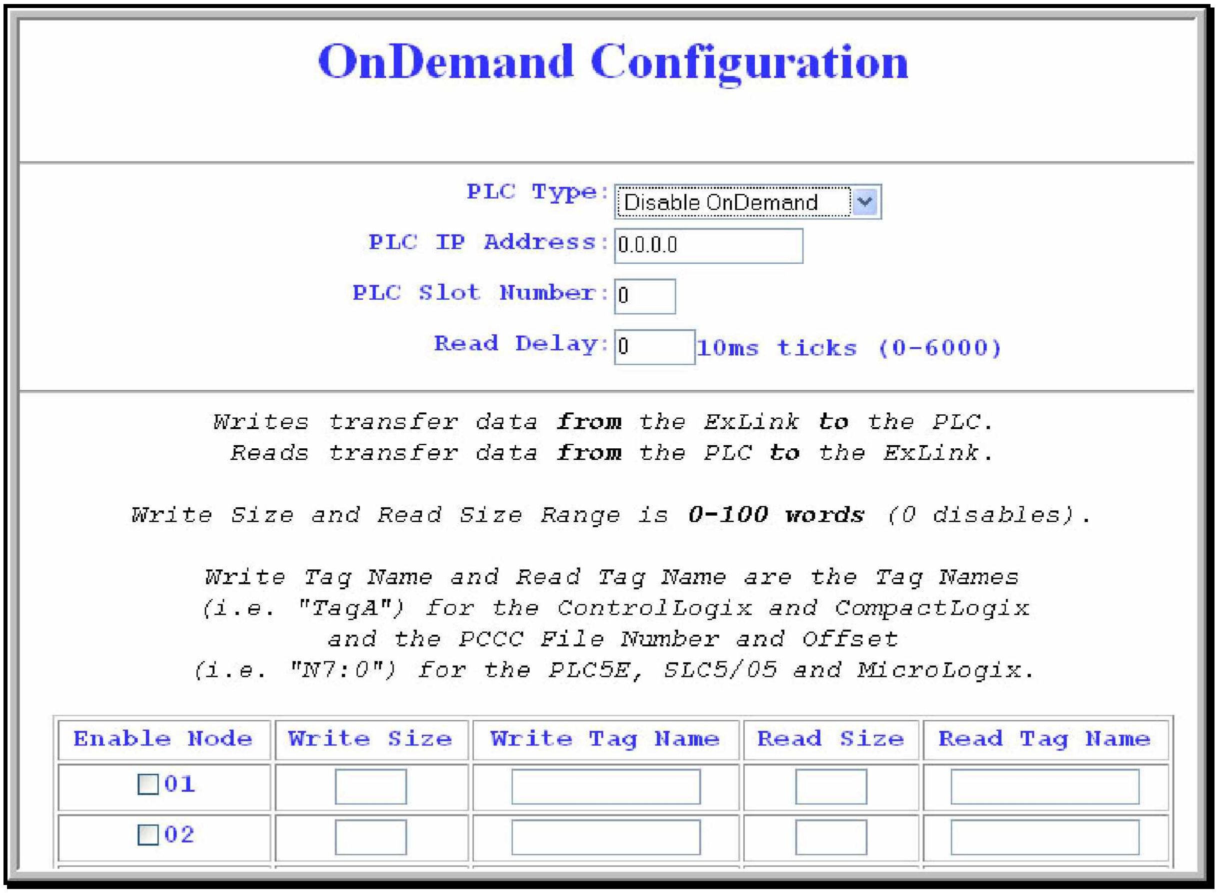
2 At the HTML Server – Main Page, click the button labeled "**OnDemand Config.**"



The *OnDemand Configuration Page* will be displayed.

OnDemand Configuration Page

The ***OnDemand Configuration*** page allows you to select and modify the settings for each Subnet Node.



OnDemand Configuration

PLC Type: Disable OnDemand

PLC IP Address: 0.0.0.0

PLC Slot Number: 0

Read Delay: 0 10ms ticks (0-6000)

Writes transfer data ***from*** *the ExLink* ***to*** *the PLC.*
Reads transfer data ***from*** *the PLC* ***to*** *the ExLink.*

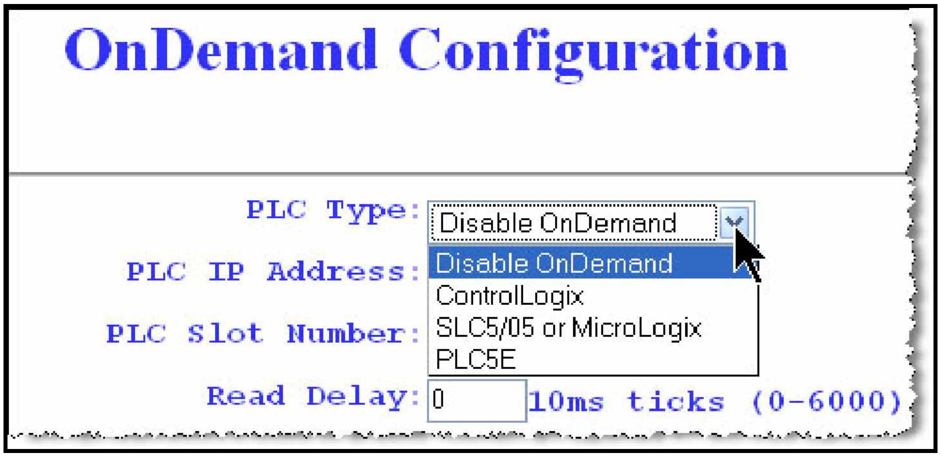
Write Size and Read Size Range is ***0-100 words*** *(0 disables).*

Write Tag Name and Read Tag Name are the Tag Names (i.e. "TagA") for the ControlLogix and CompactLogix and the PCCC File Number and Offset (i.e. "N7:0") for the PLC5E, SLC5/05 and MicroLogix.

Enable Node	Write Size	Write Tag Name	Read Size	Read Tag Name
□01				
□02				

Figure 5-3: The OnDemand Configuration Page

3 In the upper portion of the *OnDemand Configuration Page*, select a ***PLC Type*** from the drop-down menu.



OnDemand Configuration

PLC Type: Disable OnDemand

PLC IP Address:

PLC Slot Number:

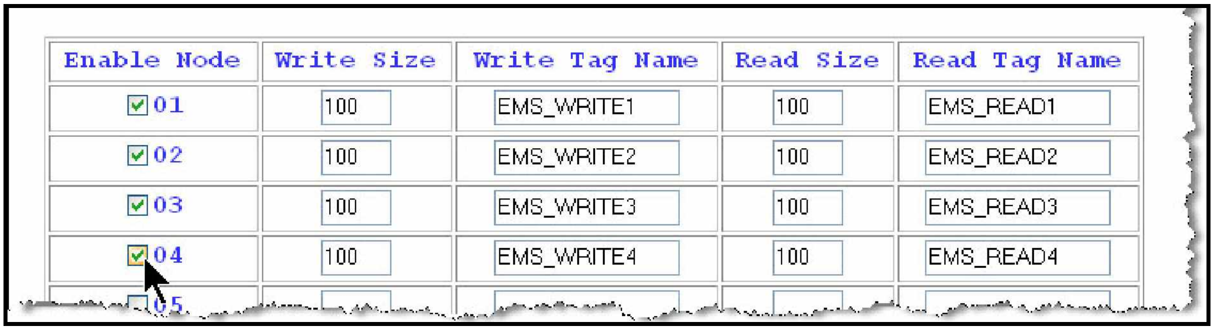
Read Delay: 0 10ms ticks (0-6000)

4 Enter the ***PLC's IP Address***.

5 For the ***PLC Slot Number,*** enter a value between 0 and 255. The PLC Slot Number indicates where in the PLC rack the controller module is installed (normally slot 0 for ControlLogix).

6 In the ***Read Delay*** field, enter a value between 0 and 6000. This number represents (in 10ms "ticks") how frequently the Gateway will query the PLC for the presence of new data. (*Note: a value of 6000 = 60 seconds; zero = disable*).

7 In the column labeled "***Enable Node,***" place a check in the box for **Node 01.**



Enable Node	Write Size	Write Tag Name	Read Size	Read Tag Name
☑01	100	EMS_WRITE1	100	EMS_READ1
☑02	100	EMS_WRITE2	100	EMS_READ2
☑03	100	EMS_WRITE3	100	EMS_READ3
☑04	100	EMS_WRITE4	100	EMS_READ4
☐05				

8 **Write Size:** Enter a value between 1 and 100 (or 0 to disable) for the ***Write Size.*** The Write Size represents the maximum number of 2-byte "words" that the Gateway will attempt to write to PLC memory during a single write cycle. *(Note: due to host/client handshaking overhead, the actual data size required on the PLC is three words larger than the value specified in this field).*

9 **Write Tag Name:** For *ControlLogix* systems, specify a ***Write Tag Name*** that is 40 characters or less (for example *EMS_WRITE1,* for Node 01). The Write Tag Name is a user defined description or title for the area of memory in the PLC where host-bound data will be written for a particular Subnet Node. (*Note: the Write Tag Name is not to be confused with writing to an RFID transponder, which is often referred to as "writing to a tag").*

OR

For *PLC5E, SLC5/05 and MicroLogix* systems, enter the ***PCCC File Number and Offset*** (for example *N7:0*) in the *Write Tag Name* field. Together these values identify the location in the PLC's Status File where host-bound data will be written for the Subnet Node.

10 **Read Size:** Enter a value between 1 and 100 (or 0 to disable) for the ***Read Size.*** The Read Size represents the maximum number of 2-byte "words" that the Gateway will attempt to retrieve from PLC memory during a single read cycle. *Note: the actual data size required on the PLC is three words larger than the value specified in this field.*

11 **Read Tag Name:** For *ControlLogix* systems, specify a ***Read Tag Name*** that is 40 characters or less (for example *EMS_READ1,* for Node 01). The Read Tag Name is a user defined description or title for the area of memory in the PLC from which the Gateway will retrieve data bound for a particular Subnet Node.

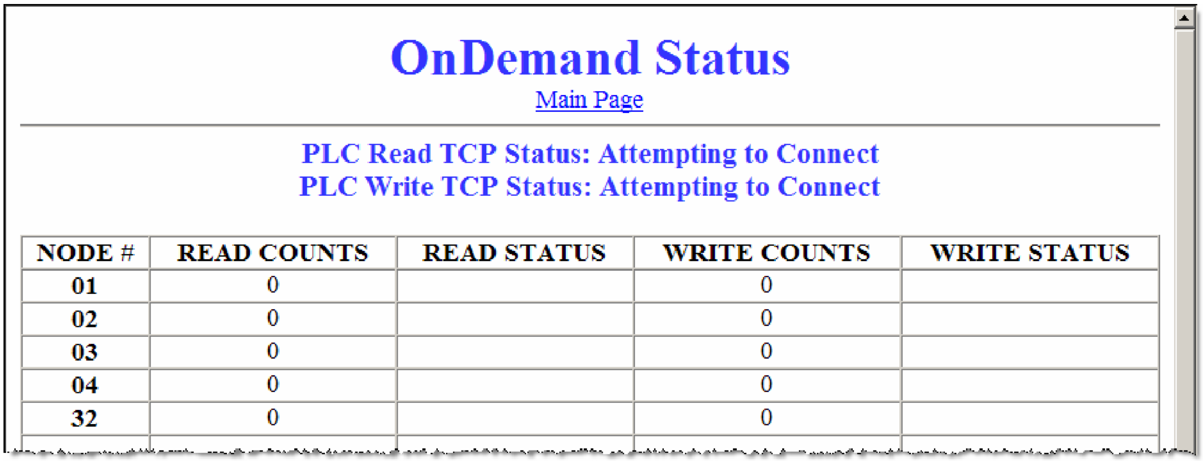
OR

For *PLC5E, SLC5/05 and MicroLogix* systems enter the ***PCCC File Number and Offset*** in the *Read Tag Name* field. Together these values indicate the location in the PLC's Status File where the Gateway will retrieve data destined for the Subnet Node.

12 After entering the proper information for Node 01, repeat steps 7 through 11 and configure Nodes 02 through 16 and 32. Node 32 (the "*Gateway Node*") is used to hold data specifically intended for the Gateway.

13 After entering your information in each field for Nodes 01 - 16 and 32, click the ***Save Settings*** button located at the bottom of the page.

The *OnDemand Status Page* will be displayed.



OnDemand Status

Main Page

PLC Read TCP Status: Attempting to Connect
PLC Write TCP Status: Attempting to Connect

NODE #	READ COUNTS	READ STATUS	WRITE COUNTS	WRITE STATUS
01	0		0	
02	0		0	
03	0		0	
04	0		0	
32	0		0	

5.3.1 Configuring PLC Controller Tags

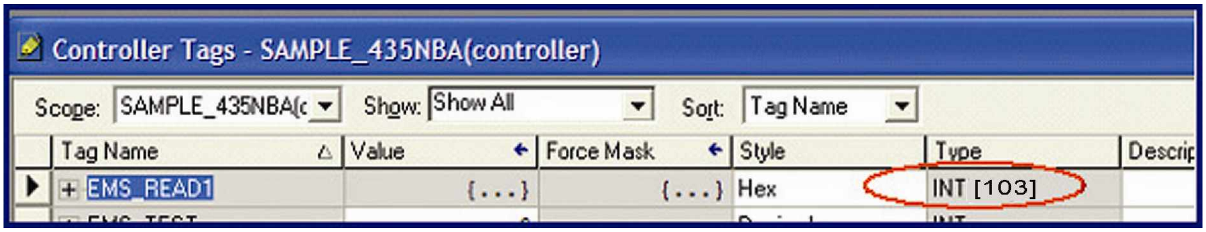
After you have configured each Subnet Node via *OnDemand Node Configuration*, open your PLC program (i.e. RSLogix 5000) and define two **Controller Tags** (a ***Write Tag*** and a ***Read Tag***) for each Subnet Node that has been configured.

Controller Tags – Names and Sizes

All Controller Tags need to be assigned a name and size. Be sure to use the same ***Write Tag Name*** and ***Read Tag Name*** that you specified in the *OnDemand Node Configuration* (i.e., EMS_WRITE1 and EMS_READ1).

These tags must also have the size capacity to store an integer array equal to your previously specified ***Write/Read Size*** **+ three words**.

So for example, if the *Read Size* you specified earlier was 100 words, the corresponding Read Tag in the PLC must be able to store an array of 103 integers.



Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style	Type	Descrip
+ EMS_READ1	{...}	{...}	Hex	INT [103]	

Controller Tags Summary

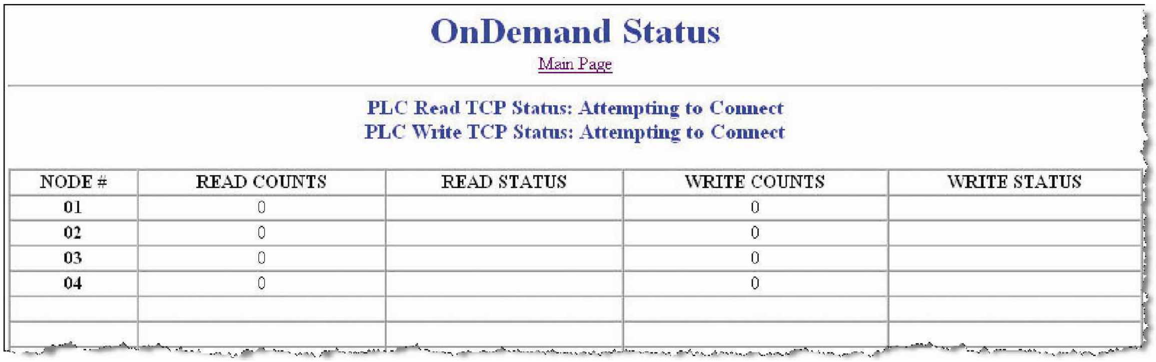
- **Write Tags** hold messages and response data bound for the PLC. This data is generated either by the Gateway or is passed through the Gateway from an RFID controller. (Note: the RFID controller is linked to the proper Write Tag via *OnDemand Node Configuration)*.
- **Read Tags** hold RFID command data intended for the Gateway or a specified Subnet Node. Instructions will be routed to the Node for which the Read Tag has been linked via *OnDemand Node Configuration*.

After creating and defining Write and Read Tags for each Subnet Node, return to the Gateway's *HTML Server – Main Page* to continue.

5.4 CHECKING ONDEMAND STATUS

Now that you have configured the Gateway's Subnet Nodes and defined corresponding Write and Read Tags for each in the PLC, the last step is to check the communication status between the Gateway's' Subnet Nodes and the PLC.

- On the Gateway's *HTML Server - Main Page*, click the link labeled "***OnDemand Status***." The *OnDemand Status Page* will be displayed.



OnDemand Status

Main Page

PLC Read TCP Status: Attempting to Connect
PLC Write TCP Status: Attempting to Connect

NODE #	READ COUNTS	READ STATUS	WRITE COUNTS	WRITE STATUS
01	0		0	
02	0		0	
03	0		0	
04	0		0	

Figure 5-4: OnDemand Status Page

The *OnDemand Status Page* provides information regarding the connection status between the PLC and each configured Subnet Node. This information can be used to verify that read and write connections between each Subnet Node and the PLC have been established successfully.

ATTENTION:

If the Gateway and PLC do not successfully establish a connection, cycle power to the Gateway and verify that Ethernet/IP services are running properly on the PLC. If that does not resolve the issue, restart Ethernet/IP services on the PLC and the 1756-ENBT module.

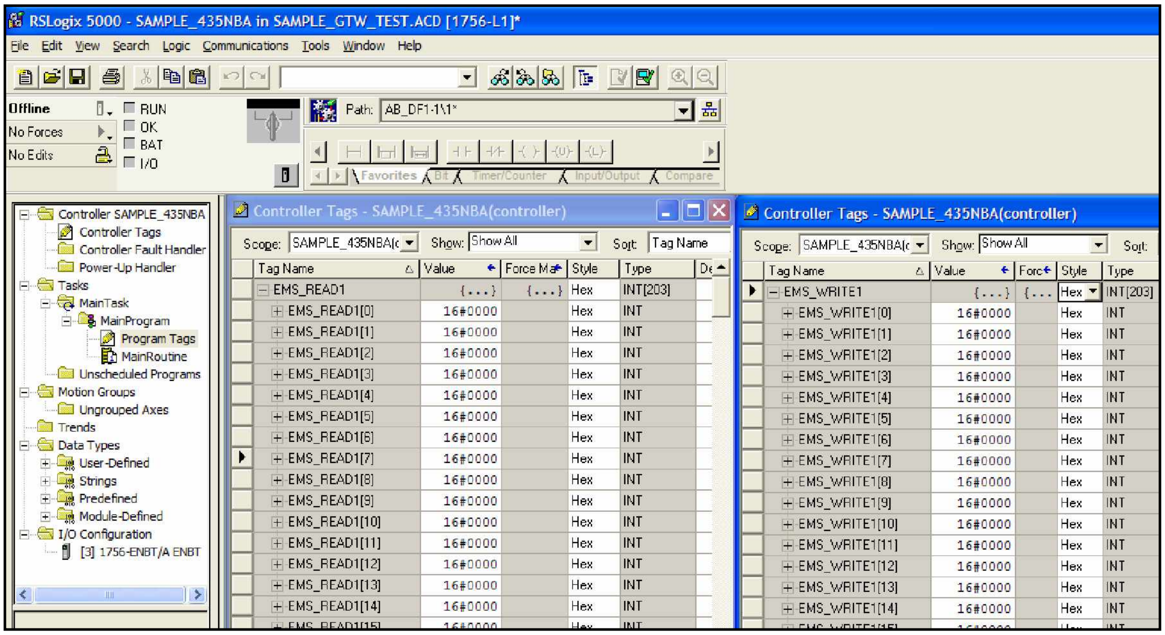
5.5 VERIFYING DATA EXCHANGE WITH RSLOGIX 5000

At this point, communication between the PLC and the Gateway should be properly configured and a connection established. You can verify the exchange of information between devices using RSLogix 5000.

Note that under the Ethernet/IP protocol, the Gateway acts as the client and the PLC acts as the Ethernet/IP server.

§ Additional messaging instructions are not required on the part of the PLC because the Gateway automatically polls each *Read Controller Tag* in PLC memory at the interval specified by the ***Read Delay*** value set via the *OnDemand Configuration Utility*.

§ There is no delay parameter when writing data to the PLC, as the Gateway writes all PLC-bound data to the appropriate *Write Controller Tag* immediately after it is generated.



Controller Tags - SAMPLE_435NBA(controller)

Tag Name	Value	Force Ma	Style	Type
EMS_READ1	{...}	{...}	Hex	INT[203]
EMS_READ1[0]	16#0000		Hex	INT
EMS_READ1[1]	16#0000		Hex	INT
EMS_READ1[2]	16#0000		Hex	INT
EMS_READ1[3]	16#0000		Hex	INT
EMS_READ1[4]	16#0000		Hex	INT
EMS_READ1[5]	16#0000		Hex	INT
EMS_READ1[6]	16#0000		Hex	INT
EMS_READ1[7]	16#0000		Hex	INT
EMS_READ1[8]	16#0000		Hex	INT
EMS_READ1[9]	16#0000		Hex	INT
EMS_READ1[10]	16#0000		Hex	INT
EMS_READ1[11]	16#0000		Hex	INT
EMS_READ1[12]	16#0000		Hex	INT
EMS_READ1[13]	16#0000		Hex	INT
EMS_READ1[14]	16#0000		Hex	INT
EMS_READ1[15]	16#0000		Hex	INT

Controller Tags - SAMPLE_435NBA(controller)

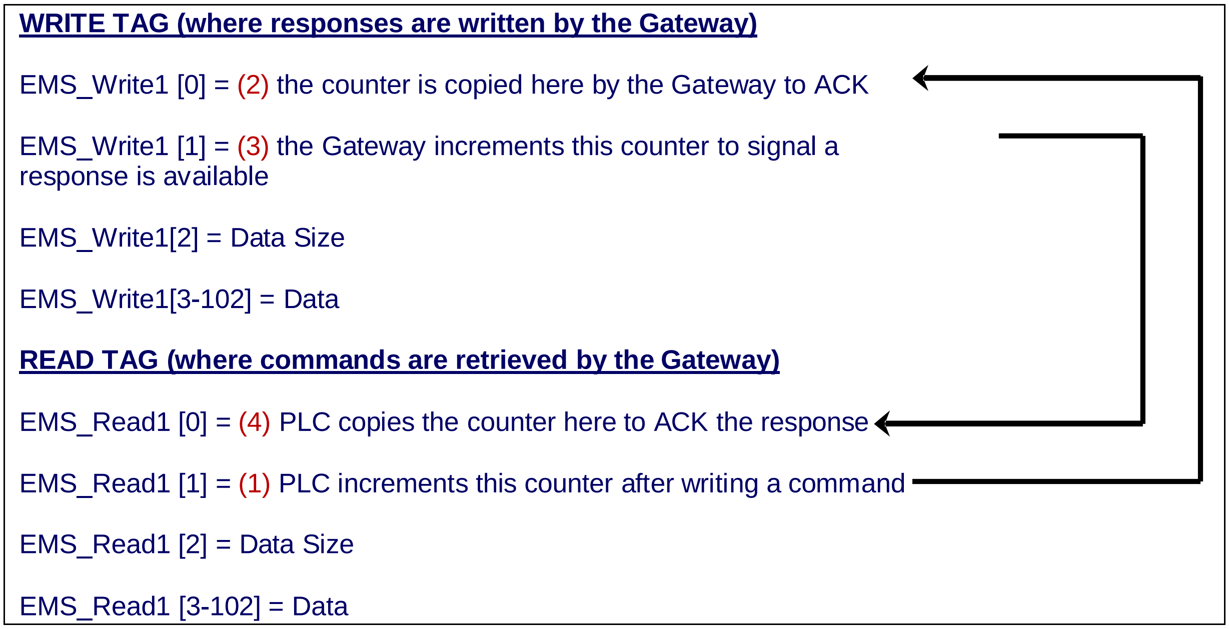
Tag Name	Value	Force	Style	Type
EMS_WRITE1	{...}	{...	Hex	INT[203]
EMS_WRITE1[0]	16#0000		Hex	INT
EMS_WRITE1[1]	16#0000		Hex	INT
EMS_WRITE1[2]	16#0000		Hex	INT
EMS_WRITE1[3]	16#0000		Hex	INT
EMS_WRITE1[4]	16#0000		Hex	INT
EMS_WRITE1[5]	16#0000		Hex	INT
EMS_WRITE1[6]	16#0000		Hex	INT
EMS_WRITE1[7]	16#0000		Hex	INT
EMS_WRITE1[8]	16#0000		Hex	INT
EMS_WRITE1[9]	16#0000		Hex	INT
EMS_WRITE1[10]	16#0000		Hex	INT
EMS_WRITE1[11]	16#0000		Hex	INT
EMS_WRITE1[12]	16#0000		Hex	INT
EMS_WRITE1[13]	16#0000		Hex	INT
EMS_WRITE1[14]	16#0000		Hex	INT
EMS_WRITE1[15]	16#0000		Hex	INT

Figure 5-5: RSLogix5000 (Screen Shot)

5.5.1 Ethernet/IP Handshaking

To ensure that messages to and from the Gateway and HF-Series Controllers are properly delivered and received, a handshaking mechanism has been implemented that uses a pair of dedicated words in the exchange. The first two words in each Controller Tag are dedicated to handshaking.

When new information is generated, the producing device (data producer) increments a counter, and the consuming device (data consumer) copies that same counter value to another memory location to signal that the information has been processed.



WRITE TAG (where responses are written by the Gateway)

EMS_Write1 [0] = (2) the counter is copied here by the Gateway to ACK

EMS_Write1 [1] = (3) the Gateway increments this counter to signal a response is available

EMS_Write1[2] = Data Size

EMS_Write1[3-102] = Data

READ TAG (where commands are retrieved by the Gateway)

EMS_Read1 [0] = (4) PLC copies the counter here to ACK the response

EMS_Read1 [1] = (1) PLC increments this counter after writing a command

EMS_Read1 [2] = Data Size

EMS_Read1 [3-102] = Data

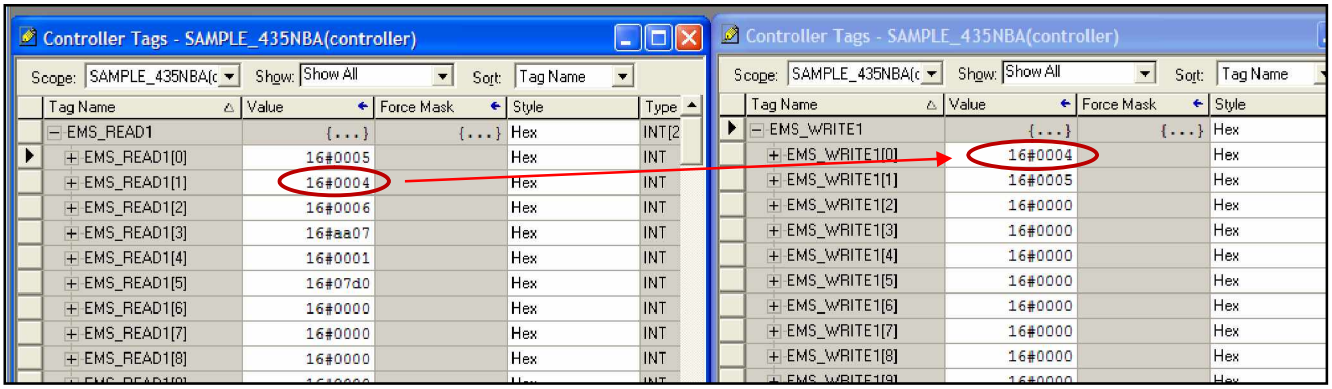
5.5.2 Ethernet/IP Handshaking Example

In the example below, EMS_READ1 is the name of the Read Tag for Node 1 and EMS_WRITE1 is the name of the Write Tag for Node 1.

NOTE: **[0]** indicates the first word, **[1]** indicates the second word in a Controller Tag.

1 The PLC writes the command to the Read Tag (EMS_READ1) and then increments the counter in EMS_READ1 [1]

2 The counter in EMS_READ1 [1] is copied by the Gateway to EMS_WRITE1 [0] which acknowledges that the command has been received.



Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

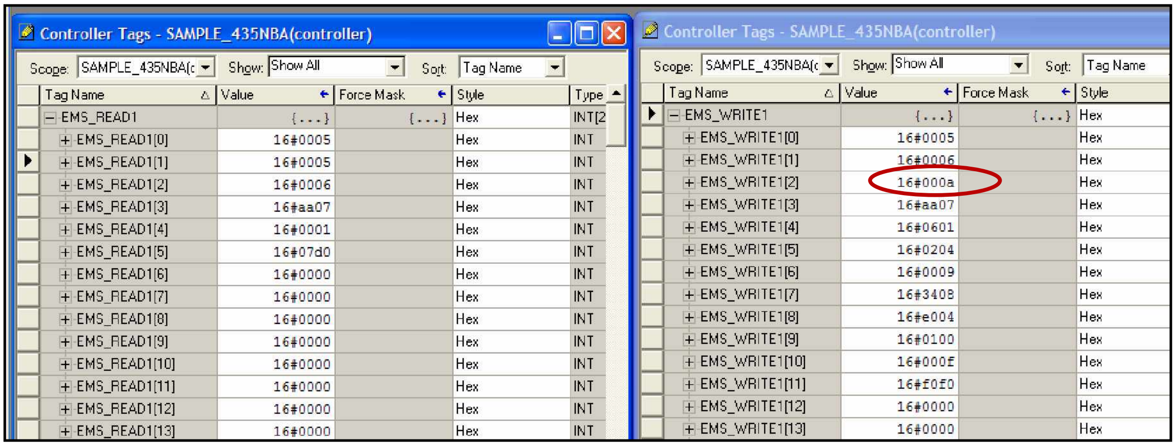
Tag Name	Value	Force Mask	Style	Type
− EMS_READ1	{...}	{...}	Hex	INT[2
+ EMS_READ1[0]	16#0005		Hex	INT
+ EMS_READ1[1]	16#0004		Hex	INT
+ EMS_READ1[2]	16#0006		Hex	INT
+ EMS_READ1[3]	16#aa07		Hex	INT
+ EMS_READ1[4]	16#0001		Hex	INT
+ EMS_READ1[5]	16#07d0		Hex	INT
+ EMS_READ1[6]	16#0000		Hex	INT
+ EMS_READ1[7]	16#0000		Hex	INT
+ EMS_READ1[8]	16#0000		Hex	INT
+ EMS_READ1[9]	16#0000		Hex	INT

Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style
− EMS_WRITE1	{...}	{...}	Hex
+ EMS_WRITE1[0]	16#0004		Hex
+ EMS_WRITE1[1]	16#0005		Hex
+ EMS_WRITE1[2]	16#0000		Hex
+ EMS_WRITE1[3]	16#0000		Hex
+ EMS_WRITE1[4]	16#0000		Hex
+ EMS_WRITE1[5]	16#0000		Hex
+ EMS_WRITE1[6]	16#0000		Hex
+ EMS_WRITE1[7]	16#0000		Hex
+ EMS_WRITE1[8]	16#0000		Hex
+ EMS_WRITE1[9]	16#0000		Hex

3 Following execution of the command, the Gateway copies the response from Node 01 to EMS_WRITE1 (the Write Tag for Node 01) and increments the counter in EMS_WRITE1 [1]. This signals that there is new data for the PLC (i.e. the RFID controller generated response).



Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

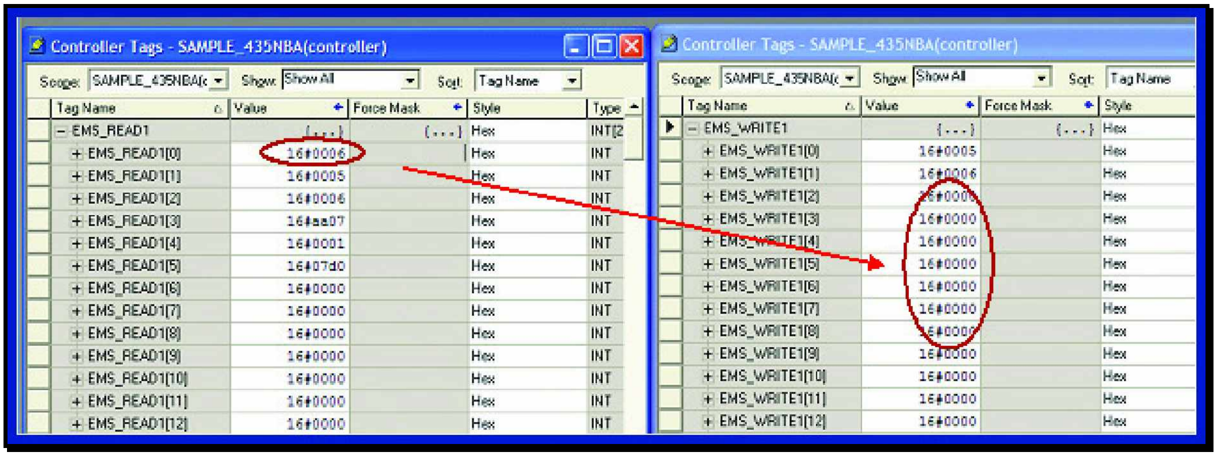
Tag Name	Value	Force Mask	Style	Type
− EMS_READ1	{...}	{...}	Hex	INT[2
+ EMS_READ1[0]	16#0005		Hex	INT
+ EMS_READ1[1]	16#0005		Hex	INT
+ EMS_READ1[2]	16#0006		Hex	INT
+ EMS_READ1[3]	16#aa07		Hex	INT
+ EMS_READ1[4]	16#0001		Hex	INT
+ EMS_READ1[5]	16#07d0		Hex	INT
+ EMS_READ1[6]	16#0000		Hex	INT
+ EMS_READ1[7]	16#0000		Hex	INT
+ EMS_READ1[8]	16#0000		Hex	INT
+ EMS_READ1[9]	16#0000		Hex	INT
+ EMS_READ1[10]	16#0000		Hex	INT
+ EMS_READ1[11]	16#0000		Hex	INT
+ EMS_READ1[12]	16#0000		Hex	INT
+ EMS_READ1[13]	16#0000		Hex	INT

Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style
− EMS_WRITE1	{...}	{...}	Hex
+ EMS_WRITE1[0]	16#0005		Hex
+ EMS_WRITE1[1]	16#0006		Hex
+ EMS_WRITE1[2]	16#000a		Hex
+ EMS_WRITE1[3]	16#aa07		Hex
+ EMS_WRITE1[4]	16#0601		Hex
+ EMS_WRITE1[5]	16#0204		Hex
+ EMS_WRITE1[6]	16#0009		Hex
+ EMS_WRITE1[7]	16#3408		Hex
+ EMS_WRITE1[8]	16#e004		Hex
+ EMS_WRITE1[9]	16#0100		Hex
+ EMS_WRITE1[10]	16#000f		Hex
+ EMS_WRITE1[11]	16#f0f0		Hex
+ EMS_WRITE1[12]	16#0000		Hex
+ EMS_WRITE1[13]	16#0000		Hex

4 After the PLC has processed the response information, it copies the counter from EMS_WRITE1 [1] to EMS_READ1 [0] which signals to the Gateway that the PLC has retrieved the response data.



Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style	Type
− EMS_READ1	{...}	{...}	Hex	INT[2
+ EMS_READ1[0]	16#0006		Hex	INT
+ EMS_READ1[1]	16#0005		Hex	INT
+ EMS_READ1[2]	16#0006		Hex	INT
+ EMS_READ1[3]	16#aa07		Hex	INT
+ EMS_READ1[4]	16#0001		Hex	INT
+ EMS_READ1[5]	16#07d0		Hex	INT
+ EMS_READ1[6]	16#0000		Hex	INT
+ EMS_READ1[7]	16#0000		Hex	INT
+ EMS_READ1[8]	16#0000		Hex	INT
+ EMS_READ1[9]	16#0000		Hex	INT
+ EMS_READ1[10]	16#0000		Hex	INT
+ EMS_READ1[11]	16#0000		Hex	INT
+ EMS_READ1[12]	16#0000		Hex	INT

Controller Tags - SAMPLE_435NBA(controller)

Scope: SAMPLE_435NBA(c Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style
− EMS_WRITE1	{...}	{...}	Hex
+ EMS_WRITE1[0]	16#0005		Hex
+ EMS_WRITE1[1]	16#0006		Hex
+ EMS_WRITE1[2]	16#0000		Hex
+ EMS_WRITE1[3]	16#0000		Hex
+ EMS_WRITE1[4]	16#0000		Hex
+ EMS_WRITE1[5]	16#0000		Hex
+ EMS_WRITE1[6]	16#0000		Hex
+ EMS_WRITE1[7]	16#0000		Hex
+ EMS_WRITE1[8]	16#0000		Hex
+ EMS_WRITE1[9]	16#0000		Hex
+ EMS_WRITE1[10]	16#0000		Hex
+ EMS_WRITE1[11]	16#0000		Hex
+ EMS_WRITE1[12]	16#0000		Hex

5 The Gateway will then clear (set to 0) holding register 40001 of the Write Tag. After which it will be ready to receive another command.

5.6 ETHERNET/IP: OBJECT MODEL

The ***Object Model*** is the logical organization of attributes (parameters) within classes (objects) and services supported by each device.
Objects are broken down into three categories: ***Required Objects***, ***Vendor Specific Objects*** and ***Application Objects***.

§ **Required Objects** are classes that must be supported by all devices on EtherNet/IP. The Gateway has six Required Objects.

§ **Vendor Specific Objects** are classes that add attributes and services that do not fit into the Required Objects or Application Objects categories. The Gateway has two Vendor Specific Objects.

§ **Application Objects** are classes that must be supported by all devices using the same profile. An example of a profile is a Discrete I/O device or an AC Drive. This ensures that all devices with the same profile have a common look on the network.

Data Type Definition Table

EtherNet/IP was designed by the *Open Device Vendors Association* (**ODVA**) as an open protocol. The following table contains a description of the data types used by ODVA that are also found in this chapter.

DATA TYPE	DESCRIPTION
USINT	Unsigned Short Integer (8-bit)
UINT	Unsigned Integer (16-bit)
UDINT	Unsigned Double Integer (32-bit)
STRING	Character String (1 byte per character)
BYTE	Bit String (8-bits)
WORD	Bit String (16-bits)
DWORD	Bit String (32-bits)

Table 5-1: Data Type Definitions

5.6.1 Ethernet/IP Required Objects

Under Ethernet/IP, there are **six** *Required Objects*:

REQUIRED OBJECTS:

§ Identity Object (0x01)

§ Message Router Object (0x02)

§ Assembly Object (0x04)

§ Connection Manager Object (0x06)

§ TCP Object (0xF5)

§ Ethernet Link Object (0xF6)

Identity Object (0x01 - 1 Instance)

Class Attributes

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Vendor Number	UINT	50 DEC	Get
2	Device Type	UINT	0x0C	Get
3	Product Code Number	UINT	6102 DEC	Get
4	Product Major Revision Product Minor Revision	USINT USINT	01 25	Get
5	Status Word (see below for definition)	WORD	See Below	Get
6	Serial Number	UDINT	Unique 32 Bit Value	Get
7	Product Name: Product Name Size Product Name String	USINT USINT[26]	GWY-01-IND-01 07 "Gateway"	Get

Status Word

Bit	Bit = 0	Bit = 1
0	No I/O Connection	I/O Connection Allocated
1 – 15	Unused	Unused

Common Services

Service Code	Implementation		Service Name
	Class Level	Instance Level	
0x0E	Yes	Yes	Get Attribute Single
0x05	No	Yes	Reset

Message Router Object (0x02)

This object has no supported attributes.

Assembly Object (0x04 - 3 Instances)

Class Attributes

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Revision	UINT	1	Get
2	Max Instance	UINT	81	Get

Instance 0x64 Attributes (Input Instance)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
3	Status Information:			Get
	Bitmap of Consume Instances with Data	DINT	0	
	Bitmap of Produce Instances with Data	DINT	0	

User Datagram Protocol (UDP) I/O Sequence Number Handshaking

The data producing device increments the data sequence number by one with the transmission of each new serial data packet. Valid sequence numbers are 1-65535. After the consuming device has processed the data, it must echo the sequence number in the handshake to allow the producing device to remove the data from the queue. This is required for I/O communications because UDP is not guaranteed to arrive in order.

If the Node ID number is passed as part of the I/O message, the message is stored to the appropriate location in the Modbus RTU table. Because communications are asynchronous, the Node ID number is also stored as part of the output data. It is the responsibility of the PLC programmer to make sure the proper request lines up with the proper response if the Gateway is used as a request/response device.

Instance 0x65 Attributes (Input Instance 2)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
3	Serial Produce Data:			Get
	Consume Data Seq. Number Handshake	UINT	0	
	Produce Data Sequence Number	UINT	0	
	Node 1 Serial Produce Data Size	UINT	0	
	Node 1 Serial Produce Data	WORD[100]	All 0's	

Instance 0x66 Attributes (Input Instance 3)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
3	Serial Produce Data:			Get
	Consume Data Seq. Number Handshake	UINT	0	
	Produce Data Sequence Number	UINT	0	
	Node ID (1-32)	UINT	1	
	Node Serial Produce Data Size	UINT	0	
	Node Serial Produce Data	WORD[100]	All 0's	

Instance 0x70 Attributes (Output Instance 1)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
3	Serial Consume Data:			Get / Set
	Produce Data Seq. Number Handshake	UINT	0	
	Consume Data Sequence Number	UINT	0	
	Node 1 Serial Consume Data Size	UINT	0	
	Node 1 Serial Consume Data	WORD[100]	All 0's	

Instance 0x71 Attributes (Output Instance 2)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
3	Serial Consume Data:			Get / Set
	Produce Data Seq. Number Handshake	UINT	0	
	Consume Data Sequence Number	UINT	0	
	Node ID (1-32)	UINT	1	
	Node Serial Consume Data Size	UINT	0	
	Node Serial Consume Data	WORD[100]	All 0's	

Instance 0x80 Attributes (Configuration Instance)

Most I/O clients include a configuration path when opening an I/O connection to a server. There is no configuration data needed.

Instance 0x81 Attributes (Heartbeat Instance – Input Only)

This instance allows clients to monitor input data without providing output data.

Common Services

Service Code	Implementation		Service Name
	Class Level	Instance Level	
0x0E	Yes	Yes	Get Attribute Single
0x10	No	Yes	Set Attribute Single

Connection Manager Object (0x06)

This object has no attributes.

TCP Object (0xF5 - 1 Instance)

Class Attributes

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Status*	DWORD	1	Get
2	Configuration Capability*	DWORD	0	Get
3	Configuration Control*	DWORD	0	Get
4	Physical Link Object* Structure of: Path Size Path	 UINT Array Of WORD	 2 0x20F6 0x2401	Get
5	Interface Configuration* Structure of: IP Address Network Mask Gateway Address Name Server Name Server 2 Domain Name Size Domain Name	 UDINT UDINT UDINT UDINT UDINT UINT STRING	 0 0 0 0 0 0 0	Get

6	Host Name*			Get
	Structure of:			
	Host Name Size	UINT	0	
	Host Name	STRING	0	

*See section 5-3.2.2.1 – 5-3.2.2.6 of "Volume 2: EtherNet/IP Adaptation of CIP" from ODVA for more information regarding these attributes.

Common Services

Service Code	**Implementation**		**Service Name**
	Class Level	**Instance Level**	
0x0E	Yes	Yes	Get Attribute Single

Ethernet Link Object (0xF6 - 1 Instance)

Class Attributes

Attribute ID	**Name / Description**	**Data Type**	**Default Data Value**	**Access Rule**
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	**Name / Description**	**Data Type**	**Default Data Value**	**Access Rule**
1	Interface Speed*	UDINT	100	Get
2	Interface Flags*	DWORD	3	Get
3	Physical Address*	USINT Array[6]	0	Get

*See section 5-4.2.2.1 – 5-4.2.2.3 of "Volume 2: EtherNet/IP Adaptation of CIP" from ODVA for more details on this attribute.

Common Services

Service Code	**Implementation**		**Service Name**
	Class Level	**Instance Level**	
0x0E	Yes	Yes	Get Attribute Single

5.6.2 EtherNet/IP: Vendor Specific Objects

The Gateway has two Vendor Specific Objects:

VENDOR SPECIFIC OBJECTS:

- Gateway Consume Data Object (0x64)
- Gateway Produce Data Object (0x65)

Gateway Consume Data Object (0x64 - 32 Instances)

Class Attributes (Instance 0)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Revision	UINT	1	Get
2	Maximum Consume Data Buffer Size (in words)	UINT	32768	Get
3	Bitmap of Consume Instances with Data Bit 0: Instance 1 … Bit 31: Instance 32	DINT	0	Get

Instance Attributes (Instances 1-32)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Consume Data Size (in words)	UINT	0	Get / Set
2	Consume Data [0-249]	UINT	0	Get / Set
3	Consume Data [250-499]	UINT	0	Get / Set
4	Consume Data [500-749]	UINT	0	Get / Set
5	Consume Data [750-999]	UINT	0	Get / Set
6	Consume Data [1,000-1,249]	UINT	0	Get / Set
…	…	…	…	…
10	Consume Data [2,000-2,249]	UINT	0	Get / Set
…	…	…	…	…
34	Consume Data [8,000-8,249]	UINT	0	Get / Set
…	…	…	…	…
38	Consume Data [9,000-9,249]	UINT	0	Get / Set

…	…	…	…	…
42	Consume Data [10,000-10,249]	UINT	0	Get / Set
…	…	…	…	…
82	Consume Data [20,000-20,249]	UINT	0	Get / Set
…	…	…	…	…
122	Consume Data [30,000-30,249]	UINT	0	Get / Set
…	…	…	…	…
126	Consume Data [31,000-31,249]	UINT	0	Get / Set
…	…	…	…	…
130	Consume Data [32,000-32,249]	UINT	0	Get / Set
131	Consume Data [32,250-32,249]	UINT	0	Get / Set
132	Consume Data [32,500-32,249]	UINT	0	Get / Set
133	Consume Data [32,750-32,767]	UINT	0	Get / Set

Common Services

Service Code	**Implementation**		**Service Name**
	Class Level	**Instance Level**	
0x05	No	Yes	Reset*
0x0E	Yes	Yes	Get Attribute Single
0x10	No	Yes	Set Attribute Single

*This Service Code is used to flush all attributes to zero.

Gateway Produce Data Object (0x65 - 32 Instances)

Class Attributes (Instance 0)

Attribute ID	**Name / Description**	**Data Type**	**Default Data Value**	**Access Rule**
1	Revision	UINT	1	Get
2	Maximum Produce Data Buffer Size (in words)	UINT	32768	Get

3	Bitmap of Produce Instances with Data Bit 0: Instance 1 … Bit 31: Instance 32	DINT	0	Get

Instance Attributes (Instances 1-32)

Attribute ID	Name / Description	Data Type	Default Data Value	Access Rule
1	Produce Data Size (in words)	UINT	0	Get / Set
2	Produce Data [0-249]	UINT	0	Get
3	Produce Data [250-499]	UINT	0	Get
4	Produce Data [500-749]	UINT	0	Get
5	Produce Data [750-999]	UINT	0	Get
6	Produce Data [1,000-1,249]	UINT	0	Get
…	…	…	…	…
10	Produce Data [2,000-2,249]	UINT	0	Get
…	…	…	…	…
34	Produce Data [8,000-8,249]	UINT	0	Get
…	…	…	…	…
38	Produce Data [9,000-9,249]	UINT	0	Get
…	…	…	…	…
42	Produce Data [10,000-10,249]	UINT	0	Get
…	…	…	…	…
82	Produce Data [20,000-20,249]	UINT	0	Get
…	…	…	…	…
122	Produce Data [30,000-30,249]	UINT	0	Get
…	…	…	…	…
126	Produce Data [31,000-31,249]	UINT	0	Get
…	…	…	…	…
130	Produce Data [32,000-32,249]	UINT	0	Get
131	Produce Data [32,250-32,249]	UINT	0	Get

132	Produce Data [32,500-32,249]	UINT	0	Get
133	Produce Data [32,750-32,767]	UINT	0	Get

Common Services

Service Code	**Implementation**		**Service Name**
	Class Level	**Instance Level**	
0x05	No	Yes	Reset*
0x0E	Yes	Yes	Get Attribute Single
0x10	No	Yes	Set Attribute Single

*This Service Code is used to flush all attributes to zero.

5.6.3 Application Object (0x67 _ 10 Instances)

Class Attributes (Instance 0)

Attribute ID	**Name / Description**	**Data Type**	**Default Data Value**	**Access Rule**
1	Revision	UINT	1	Get

Instance Attributes (Instances 1-32)

Attribute ID	**Name / Description**	**Data Type**	**Default Data Value**	**Access Rule**
1	**Instance Type** (0-3): 0 - Disable 1 – ControlLogix 2 – SLC 5/05 3 – PLC5E	USINT	0	Get
2	**PLC IP Address**	UDINT	0	Get
3	**PLC Slot Location** (0-255)	USINT	0	Get
11	**Max Write Size** in Words: 0 – Disabled 1 – 100 Words	UINT	0	Get
12	**Write Tag Name** (ControlLogix Only)	SHORT STRING	0	Get
13	**Write File Number** (SLC/PLC Only) **NX:0** - where **"X"** is the File Number	UINT	7	Get

14	**Write File Offset** (SLC/PLC Only) **N7:Y** - where **"Y"** is the File Offset	UINT	0	Get
15	**Write "Heartbeat" Timeout** • Measured in 10ms "ticks" • 0 = disabled • Max value: 6000 ticks	UINT	100	Get
21	**Max Read Size** in Words 0 – Disable Max Value: 100	UINT	0	Get
22	**Read Tag Name** (ControlLogix Only)	SHORT STRING	0	Get
23	**Read File Number** (SLC/PLC Only) **NX:0 -** Where **"X"** is the File Number	UINT	7	Get
24	**Read File Offset** (SLC/PLC Only) **N7:Y** - Where **"Y"** is the File Offset	UINT	0	Get
25	**Read Poll Rate** • Measured in 10ms "ticks" • 0 = disabled • 6000 ticks max	UINT	100	Get

Common Services

Service Code	**Implementation**		**Service Name**
	Class Level	**Instance Level**	
0x0E	Yes	Yes	Get Attribute Single

CHAPTER 6: MODBUS TCP INTERFACE

One of the most popular and well-proven industrial automation protocols in use today is Modbus. Modbus TCP allows the Modbus protocol to be carried over standard Ethernet networks.

6.1 MODBUS TCP OVERVIEW

Under the Modbus TCP protocol, the Gateway acts as a Modbus Server and the PLC acts as a Modbus Client. By utilizing **Produce** and **Consume** registers for mapping commands and responses, data produced by the Gateway is consumed by the Modbus Client and data produced by the Modbus Client is consumed by the Gateway.

ATTENTION:

The Modbus Client (host or PLC) must connect to the Modbus Server (Gateway) on port **502**.

Maximum number of words transferred to/from an RFID tag per read/write cycle: **100 Words / 200 Bytes**

6.1.1 Modbus TCP Command Packet Structure

Mapping Nodes 1-16 & 32 (Consume Registers)

Consume Registers hold data that is destined for the Gateway or RFID controllers. Modbus TCP commands must be placed in the holding registers, starting at address 40001, of the Device ID (Node Input Page) for which they are intended. Commands utilize at least six registers (double-byte values or words).

MODBUS ADDRESS (4XXXX / 3XXXX)	READ / WRITE PRIVILEGE	REGISTER DESCRIPTION
(40001) 1	R/W	2-byte Gateway Consume Data Overall Length (> 0 indicates data is available; Gateway clears to 0 after data is processed)

2	R/W	MSB = Reader Type;
		LSB = Command ID

3	R/W	MSB = 0x00,
		LSB = Node ID (1-16 or 32)

4	R/W	2-byte Timeout Value (0-65535) measured in milliseconds
5	R/W	2-byte Read/Write Start Address (0-65535)
6	R/W	2-byte Read/Write Length (0-65535 bytes)

7 – 32774	R/W	Gateway Consume Data (when applicable)
32775 – 65536	R/W	Reserved

Table 6-1: Modbus TCP Command Packet Structure

Mapping for Nodes 1-16 & 32 (Consume Registers)

6.1.2 Modbus TCP Response Packet Structure

Mapping Nodes 33-48 & 64 (Produce Registers)

Produce Registers hold data that is destined for the host or PLC.

MODBUS ADDRESS (4XXXX / 3XXXX)	READ / WRITE PRIVILEGE	REGISTER DESCRIPTION
(40001) 1	R/W	Gateway Produce Data Overall Length (> 0 indicates data is available; Modbus Client clears to 0 after data is processed)
2	RO	MSB = Reader Type; LSB = Command ID
3	RO	Node ID Number (33-48, 64)
4	RO	Timeout Value (0-65535)
5	RO	Read/Write Start Address (0-65535)
6	RO	Read/Write Length (0-65535 bytes)
7 – 32774	RO	Gateway Produce Data (when applicable)
32775 – 65536	RO	Reserved

Table 6-2: Modbus TCP Response Packet Structure

Mapping for Nodes 33-48 & 64 (Produce Registers)

6.1.3 Modbus TCP Mapping for Node 65 (Gateway Configuration)

MODBUS ADDRESS (4XXXX)	READ / WRITE PRIVILEGE	REGISTER DESCRIPTION
1	R/W	IP Address 1 (MSB) Example: 192
2	R/W	IP Address 2 Example: 168
3	R/W	IP Address 3 Example: 000
4	R/W	IP Address 4 (LSB) Example: 100
5	R/W	Subnet Mask 1 (MSB) Example: 255
6	R/W	Subnet Mask 2 Example: 255
7	R/W	Subnet Mask 3 Example: 255
8	R/W	Subnet Mask 4 (LSB) Example: 000
9	R/W	Gateway Address 1 (MSB) Example: 192

10	R/W	Gateway Address 2 Example: 168
11	R/W	Gateway Address 3 Example: 000
12	R/W	Gateway Address 4 (LSB) Example: 001
13	RO	MAC Address 1 (MSB) Example: 0x00
14	RO	MAC Address 2 Example: 0x40
15	RO	MAC Address 3 Example: 0x9D
16	RO	MAC Address 4 Example: 0x12
17	RO	MAC Address 5 Example: 0x34
18	RO	MAC Address 6 (LSB) Example: 0x56
19	RO	Link Status: 0 = No Link 1 = Link is OK
20	RO	Ethernet Speed (10M or 100M bits)
21	RO	Link Duplex: 0 = Half Duplex 1 = Full Duplex
22	RO	Revision (Major/Minor)
23 – 1000	R/W	Reserved
1001	RO	(Input) Data Ready Mask - Nodes 1 - 16
1002	RO	(input) Data Ready Mask - Nodes 17 - 32
1003	RO	(Output) Data Ready Mask - Nodes 33 - 48
1004	RO	(Output) Data Ready Mask - Nodes 49 - 64
1005-10099	R/W	Reserved
10100 – 10199	R/W	OnDemand Node 1 Configuration (See *OnDemand Settings Table* below)
10200 – 10299	R/W	OnDemand Node 2 Configuration
…	…	…
13100 – 13199	R/W	OnDemand Node 31 Configuration
13200 – 13299	R/W	OnDemand Node 32 Configuration
13300 – 65536	R/W	Reserved

Table 6-3: Modbus TCP Mapping for Node 65

6.1.4 OnDemand Settings Table – Modbus TCP Example

MODBUS ADDRESS (4XXXX)	READ / WRITE PRIVILEGE	REGISTER DESCRIPTION
10100	R/W	Instance Type: 0 = Disable 1 = ControlLogix 2 = SLC 5/05 3 = PLC 5E
10101	R/W	PLC IP Address 1 MSB
10102	R/W	PLC IP Address 2
10103	R/W	PLC IP Address 3
10104	R/W	PLC IP Address 4 LSB
10105	R/W	PLC Slot Location (0-255)

Table 6-4: OnDemand Settings Table – Modbus TCP Example

6.2 MODBUS TCP HANDSHAKING

Due to the process with which commands and responses are passed between the Gateway and the host, a handshaking procedure is used to notify the host or PLC that returning data is available for retrieval.

OVERALL LENGTH

The handshaking process is governed by the changing of the "***Overall Length***" value within a data packet. The Overall Length value is typically the first word (2-bytes) of a command or response and indicates the total number of data words in the packet.

NODE INPUT AND NODE OUTPUT PAGES

Under the Modbus TCP protocol, host-generated data is written to a pre-defined region of the Gateway's own memory known as a ***Node Input Page***. Host-bound data generated by the Gateway or an RFID controller, is written to a separate region of the Gateway's memory known as a ***Node Output Page*** (in Modbus these regions of memory are called **Device IDs**). Each Subnet Node has its own Node Input and Node Output Page (Device ID) which is used to temporarily hold incoming (controller-bound) and outgoing (host-bound) data.

OUTPUT DATA READY MASK

To notify the host that new data is waiting to be retrieved from one of the Node Output Pages, the Gateway utilizes a separate 32-bit block of internal memory, called the ***Output Data Ready Mask***. The lowest 16 bits of the 32-bit Output Data Ready Mask each represent the status of one Node Output Page (one for each of the 16 Subnet Nodes). For example, the first or lowest bit (*bit 01*) represents Node Output Page 33 - which holds output data from Subnet Node 01. Bit 02 represents Node Output Page 34 - which holds output data from Subnet Node 02, and so forth.

The Gateway, itself, is assigned Subnet Node 32 and thus, its corresponding Node Output Page is 64. Node Output Page 64 is represented by the final bit (*bit 32*) in the Output Data Ready Mask. *See section 4.1 for more information on Node Input and Node Output Pages.*

HOLDING REGISTERS

When writing host-bound data to one of the Node Output Page, the Gateway actually places each byte of the data packet into pre-defined "***holding registers***" within the Node Output Page of the Subnet Node that generated the data. Note that a single holding register stores 2-bytes or one word of data. The 2-byte Overall Length value, for example, is written to the first holding register (which is location **40001)** of the data producer's Node Output Page.

Then, as the Gateway finishes writing host-bound data (such as a controller's response) to the Node Output Page of the data producing Subnet Node, the Overall Length value (stored at holding register 40001) will change from its default value of 0x00 to reflect the number of data words within the newly written host-bound data packet. This change to the Overall Length value (i.e. register 40001) within a Node Output Page, triggers the Gateway to enable (change from zero to one) the corresponding bit in the Output Data Ready Mask. It is when a bit in the Output Data Ready Mask has become enabled, that the host will recognize the pending data.

Finally, after the host has retrieved its pending data, the enabled bit in the Output Data Ready Mask and the Overall Length value at holding register 40001 of the Node Output Page will be reset to zero (0x00), indicating that the host has received and processed its pending data.

6.2.1 Host/Gateway Modbus TCP Handshaking

One implication of this process is that when the host issues a command, it must first write the entire command to the given Node Input Page, leaving the Overall Length value to be written last.

For example, for the host to issue the 6-word command "*Read Data from Node 3*," it must first write the last five words of the command to Node Input Page 03, beginning at register 40002. After which, the host will fill in the first word (at holding register 40001) with the Overall Length of the command packet.

Last Five Words of a Read Data Command

WORD	MSB	LSB	DESCRIPTION
02	0xAA	0x05	Command ID: Read Data
03	0x00	0x03	Node ID
04	0x03	0xE8	Timeout of 1 second
05	0x00	0x20	Read Start Address: 0x20
06	0x00	0x04	Read 4 Bytes

After writing the last five words of the command, the host will write the Overall Length value to holding register 40001 of Node Input Page 03.

First Word of a Read Data Command

WORD	MSB	LSB	DESCRIPTION
01	0x00	0x06	Overall Length (in words)

The moment the Overall Length value (at holding register 40001) of Node Input Page 03 changes from 0x00 to a "non-zero" value, the Gateway will recognize the waiting data and will distribute the command to the RFID controller at Subnet Node 03.

6.2.2 Modbus TCP Command, Response & Handshaking Example

1 The host or PLC issues an RFID command to Subnet Node 04, writing the command string to the holding registers for Device ID 04 (Node Input Page 04). An Overall Length value of 0x06 is written last to holding register 40001.

2 The Gateway recognizes that the Overall Length value at holding register 40001 has changed for Device ID 04 (Node Input Page 04), indicating that a command is waiting to be directed to the RFID controller at Subnet Node 04. The Gateway distributes the command accordingly.

3 The Gateway, now having passed the command to the specified RFID controller, clears the Overall Length holding register of Device ID 04 (Node Input Page 04), setting it back to its default value of zero (0x00).

Note: when a Node Input Page's value at register 40001 is returned to 0x00, the host can assume that the command was at least received and execution was attempted. The host will also assume that it is OK to clear the remaining holding registers and write another command to this Device ID (Node Input Page).

4 Meanwhile, the RFID controller at Subnet Node 04 executes its given command instructions and generates a controller command response.

5 The Gateway retrieves the command response data from the RFID controller at Subnet Node 04 and writes the host-bound content to the holding registers for Device ID 36 (Node Output Page 36). Again, the Overall Length value is written last to holding register 40001.

Note: Host-bound data is always written to a Device ID (Node Output Page) that is 32 greater in number than the Node ID of the data producing device.

6 Because holding register 40001 of Device ID 36 (Node Output Page 36) now contains a non-zero length value, the Gateway enables (change from zero to 1) the forth bit in the *Output Data Ready Mask*. (The fourth bit is allocated to Node Output Page 36, just as the fifth byte is allocated to Node Output Page 37, and so on).

7 Once bit 04 in the *Output Data Ready Mask* becomes enabled, the host retrieves the data string stored in the holding register area for Device ID 36 (Node Output Page 36).

8 After importing the data from Device ID 36 (Node Output Page 36), the host clears (sets back to 0x00) the Overall Length value at holding register 40001 of Device ID 36 (Node Output Page 36). In doing so, bit 4 in the Output Data Ready Mask is also cleared.

Note: the clearing of bit 4 in the Output Data Ready Mask indicates to the Gateway that the host has indeed received the response and that it is now OK to write another response to Node Output Page 36.

This completes the Modbus TCP handshaking cycle.

CHAPTER 7: STANDARD TCP/IP INTERFACE

7.1 STANDARD TCP/IP OVERVIEW

Another means of communicating with the Gateway is through the typical TCP/IP protocol. For this manual, the protocol is referred to as ***Standard TCP/IP*** to distinguish it from other industrial protocols.

In this environment, the Gateway acts as the server and the host or PLC acts as client.

ATTENTION:

The TCP/IP Client (host or PLC) must connect to the TCP/IP Server (Gateway) on port **50200**.

Maximum number of words transferred to/from an RFID tag per read/write cycle: **100 Words / 200 Bytes**

Standard TCP/IP sessions are established between a host PC and the Gateway via TCP/IP client software. A TCP/IP session generally consists of three stages: *connection setup, data transactions* and *connection termination*.

All connections to the Gateway are initiated by client side software only. If, for example, an existing connection terminates unexpectedly, the Gateway will not attempt to contact the client software or re-establish a connection. The client is responsible for opening, maintaining, and closing all TCP/IP sessions.

After establishing a successful connection, communications between the client software and the Gateway can proceed. When communication is no longer necessary, it is the responsibility of the client side application to terminate the connection.

7.2 STANDARD TCP/IP: COMMAND & RESPONSE EXAMPLES

In standard TCP/IP, RFID commands issued by the host resemble Modbus TCP commands; however, an additional **two-byte** header is required (**0xFF** in the MSB, and the target **Node ID** number in the LSB). These two bytes are inserted in front of the standard Modbus TCP command string. The Gateway handles all handshaking tasks.

NOTE: Standard TCP/IP response packets contain the two-byte header (**0xFF** in the MSB and the **Node ID Echo** in the LSB), UNLESS the command was executed by Node 01, in which case there will be no additional 2-byte header in the response.

7.2.1 Standard TCP/IP Controller Command Structure & Example

In the following example, a 14-byte command has been issued to Subnet Node 02, instructing the controller to read six bytes from a tag within RF range. A Timeout Value of two seconds has been set for the completion of the command.

WORD	MSB	LSB	DESCRIPTION

00	0xFF	0x02	"Standard TCP/IP" 2-byte **Command Header:**
			MSB = **0xFF**
			LSB = **Node ID:** *0x02*

01	0x00	0x06	**Overall Length:** 2-byte value indicating number of "words" in the command packet, not including the preceding 2-byte header

02	0xAA	0x05	MSB = **0xAA**
			LSB = **Command ID:** *0x05 - Read Data Command*

03	0x00	0x02	MSB = **0x00** LSB = **Node ID** (must be same as specified in header): *0x02*
04	0x07	0xD0	2-byte **Timeout Value** measured in millisecond increments. (*0x07D0 = 2000 x .001 = 2 seconds*)
05	0x00	0x01	2-byte **Start Address** for the Read: *0x0001*
06	0x00	0x06	2-byte **Read Length**: *6 bytes*

Table 7-1: Standard TCP/IP Controller Command Structure & Example

7.2.2 Standard TCP/IP Controller Response Structure & Example

The following resembles a typical response to the controller command issued in the previous example:

WORD	MSB	LSB	DESCRIPTION

00	0xFF	0x02	"Standard TCP/IP" 2-byte **Response Header:**
			MSB = **0xFF**
			LSB = **Node ID Echo**: *0x02*
			(NOTE: There is no 2-byte response header when a command is executed by Node 01)

01	0x00	0x09	**Overall Length:** 2-byte value indicating number of "words" in the command packet, not including the preceding 2-byte header

02	0xAA	0x05	MSB = **0xAA**
			LSB = **Command Echo**: 0x05 *Read Data Command* Echo

03	<IC>	0x02	MSB = **Instance Counter Value**
			LSB = **Node ID:** *0x02*

04	0x05	0x1B	**Time Stamp:** Month / Day
05	0x01	0x0C	**Time Stamp:** Hour / Minute

06	0x21	0x06	MSB = **Time Stamp:** Seconds
			LSB = Number of **Additional Data Bytes Returned:**

07	<D1>	<D2>	**Returned Data Bytes** 1 & 2
08	<D3>	<D4>	**Returned Data Bytes** 3 & 4
09	<D5>	<D6>	**Returned Data Bytes** 5 & 6

Table 7-2: Standard TCP/IP Controller Response Structure & Example

7.2.3 Standard TCP/IP Gateway Command Structure & Example

This example retrieves the "Gateway Name" from the Gateway. Note that the Gateway is always assigned Node ID 32 (in Hex, Node 32 = 0x20).

WORD	MSB	LSB	DESCRIPTION

00	0xFF	0x20	"Standard TCP/IP" 2-byte **Command Header:**
			MSB = **0xFF**
			LSB = **Node ID**: (*0x20 = Gateway Node 32*).

01	0x00	0x06	**Overall Length:** 2-byte value indicating packet size in number of "words" (not including the 2-byte header).

02	0xAA	0x11	MSB = **0xAA**
			LSB = **Command ID:** *0x11- Get Gateway Name*

03	0x00	0x20	MSB = **0x00** LSB = **Node ID:** (Must be same as specified in header: *0x20)*
04	0x00	0x00	**N/A for this command**
05	0x00	0x00	**N/A for this command**
06	0x00	0x00	**N/A for this command**

Table 7-3: Standard TCP/IP Gateway Command Structure & Example

7.2.4 Standard TCP/IP Gateway Response Structure & Example

Below is the response to the Gateway command example above.

WORD	MSB	LSB	DESCRIPTION
01	0xFF	0x20	"Standard TCP/IP" 2-byte **Response Header:** MSB = **0xFF** LSB = **Node ID Echo:** (*0x20 = Gateway Node 32)*.
02	0x00	0x0E	**Overall Length:** 2-byte value indicating packet size in number of "words" (not including the 2-byte header)

03	0xAA	0x11	MSB = **0xAA**
			LSB = **Command Echo:** *0x11 - Get Gateway Name*

04	<IC>	0x20	MSB = **Instance Counter Value**
			LSB = **Node ID Echo**

05	0x03	0x13	**Time Stamp**: Month / Day
06	0x0D	0x03	**Time Stamp**: Hour / Minute
07	0x18	0x10	MSB = **Time Stamp**: Seconds LSB = Number of **Additional Data Bytes Returned** (*16 bytes of additional data*)
08	0x45	0x4D	**Returned Data Bytes** 1 and 2
09	0x53	0x20	**Returned Data Bytes** 3 and 4
10	0x52	0x46	**Returned Data Bytes** 5 and 6
11	0x49	0x44	**Returned Data Bytes** 7 and 8
12	0x20	0x47	**Returned Data Bytes** 9 and 10
13	0x61	0x74	**Returned Data Bytes** 11 and 12
14	0x65	0x77	**Returned Data Bytes** 13 and 14
15	0x61	0x79	**Returned Data Bytes** 15 and 16

Table 7-4: Standard TCP/IP Gateway Response Structure & Example

The 16-bytes of returned data spell (in ASCII) "EMS RFID Gateway."

CHAPTER 8:
RFID COMMANDS AND ERROR CODES

RFID commands are initiated by the host or PLC and are directed to the Gateway or to one of the Gateway's Subnet Nodes, which, in turn, instructs the corresponding RFID controller to perform or execute an operation.

ATTENTION:

For complete command and response packet structures and examples of each RFID command, please refer to the ***CBx Command Protocol – Reference Manual*** available at www.ems-rfid.com.

8.1 RFID COMMANDS TABLE

The table below lists the CBx protocol RFID commands supported by the Gateway and EMS' RFID Controllers.

COMMAND ID	COMMAND NAME	DESCRIPTION
	RFID Tag Commands	
0x02	Lock Memory Block	Write protects a block of tag memory
0x04	Fill Tag	Writes a specified data byte value to all defined tag addresses
0x05	Read Data	Reads a specified length of data from a contiguous (sequential) area of tag memory
0x06	Write Data	Writes a specified number of bytes to a contiguous area of tag memory
0x07	Read Tag ID	Reads a tag's unique tag ID number
0x08	Tag Search	Instructs the controller to search for a tag in its RF field
0x0C	Execute Macro	Instructs the controller to execute one of its eight possible macros
0x0D	Start Continuous Read	Instructs the controller to start or stop Continuous Read mode.
0x0E	Read Tag ID and Data	Reads a tag's ID and the requested number of bytes from tag memory

COMMAND ID	COMMAND NAME	DESCRIPTION
0x0F	**Start Continuous Read Tag ID and Data**	Places the controller into (or out of) Continuous Read mode and (when evoked) will retrieve a tag's ID.
Gateway Information Commands		
0x10	**Get Gateway Software Version**	Retrieves the version number of the firmware code installed on the Gateway
0x11	**Get Gateway Name**	Retrieves the Gateway's user-defined ASCII name
0x12	**Get Dipswitch Settings**	Retrieves the status of the Gateway configuration dipswitches
0x13	**Get Node Status List**	Retrieves the operational status of the Gateway Subnet Nodes
0x14	**Get Notification Mask**	Retrieves the user-defined 16-bit "*Notification Mask*" that determines for which events the Gateway notifies the host PC
0x15	**Get Last Gateway Error**	Retrieves information from the Gateway regarding the last or most recent error that was experienced
0x16	**Get Gateway Time**	Retrieves the current date and time as set internally on the Gateway
0x1C	**Get Subnet Baud Rate**	Retrieves the baud rate of the Subnet network
0x21	**Set Gateway Name**	Writes to flash memory, a user-defined "friendly" name for the Gateway
0x24	**Set Notification Mask**	Used to customize or modify the Gateway's 16-bit Notification Mask
0x26	**Set Gateway Time**	Used to set the Gateway's internal clock and calendar
0x2C	**Set Subnet Baud Rate**	Used to modify and store changes to the Subnet network baud rate

COMMAND ID	COMMAND NAME	DESCRIPTION
RFID Controller Commands		
0x30	**Get Controller Name**	Retrieves the controller's user-defined name
0x33	**Get Controller Configuration**	Retrieves the controller's configuration settings
0x38	**Get Controller Info**	Retrieves hardware, firmware and serial number information from the controller
0x40	**Set Controller Name**	Used to set (create or modify) the user-defined name for the controller
0x43	**Set Controller Configuration**	Used to set (configure or modify) the controller's configuration parameters and settings
0x4E	**Set Controller Time**	Used to set the time for the controller
0x53	**Initialize Controller**	Removes all configuration settings stored for the controller
0x54	**Reset Controller**	Resets power to the controller
Gateway Subnet Commands		
0x60	**Initialize Gateway**	Clears all Subnet Node configuration information stored in the Gateway's flash memory
0x61	**Reset Gateway**	Performs an electrical reset of the Gateway
0x62	**Initialize All Nodes**	Removes all stored configuration information for all nodes and reconfigures them to factory defaults
0x63	**Initialize All Node Macros**	Removes all stored macros from all nodes
0x70	**Start Subnet**	Instructs the Gateway to begin "polling" the Subnet network
0x71	**Move Controller**	Used to move all stored configuration data for a particular Node ID to another specified Node ID

COMMAND ID	COMMAND NAME	DESCRIPTION
0x79	**Clear Pending Response**	Deletes all pending or buffered responses in the Gateway and resets all Instance Counters to zero
Multi-Tag RFID Commands		
0x92	**Multi-Tag Read ID and Data All**	Retrieves the tag ID number and a contiguous segment of data from all RFID tags in range
0x95	**Multi-Tag Block Read All**	Retrieves a contiguous segment of data from all RFID tags in range
0x96	**Multi-Tag Block Write All**	Writes a contiguous segment of data to all RFID tags in range
0x97	**Multi-Tag Get Inventory**	Retrieves the tag ID number from all RFID tags found in range
0x98	**Multi-Tag Search All**	Checks for the presence of RFID tags in RF range and returns only the number of tags found
0xA5	**Multi-Tag Block Read by ID**	Reads a contiguous segment of data from a specific RFID tag identified by its tag ID
0xA6	**Multi-Tag Block Write by ID**	Writes a contiguous segment of data to a specific RFID tag identified by its tag ID

Table 8-1: RFID Commands Table

8.2 ERROR CODE TABLE

ERROR CODE	ERROR	DESCRIPTION
0x04	FILL TAG FAILED	Fill Tag Command Failed
0x05	READ DATA FAILED	Read Data Command Failed
0x06	WRITE DATA FAILED	Write Data Command Failed
0x07	READ TAG ID FAILED / TAG SEARCH FAILED	Read Tag ID Command Failed, Tag Search Command Failed and/or No Tag Found
0x21	INVALID SYNTAX	Command Contained a Syntax Error
0x23	INVALID TAG TYPE	Invalid Tag Type Specified
0x30	INTERNAL CONTROLLER ERROR	Generic Internal Controller Error
0x31	INVALID CONTROLLER TYPE	Invalid Controller Type (when Setting Configuration)
0x34	INVALID VERSION	Invalid Software Version Specified (when Setting Configuration)
0x35	INVALID RESET	Invalid Hardware Reset
0x36	WRITE CONFIGURATION FAILED	Set Configuration Command Failed
0x37	READ CONFIGURATION FAILED	Get Configuration Command Failed
0x80	UNKNOWN GATEWAY ERROR	Generic Gateway Error – an undetermined error occurred.
0x81	COMMAND MALFORMED	Generic Command Syntax Error
0x82	COMMAND PROTOCOL MISMATCH	An invalid protocol value was specified in the command
0x83	COMMAND INVALID OPCODE	An invalid Opcode (Command ID number) was specified in the command
0x84	COMMAND INVALID PARAMETER	A parameter specified in the command was invalid
0x85	COMMAND INVALID CONTROLLER ID	A Controller ID (Node ID) specified in the command was invalid, or no controller detected/present at the specified node
0x86	COMMAND INACTIVE CONTROLLER ID	A Controller ID (Node ID) specified in the command is currently inactive.
0x87	SUBNET DEVICE SELECT FAILED	Internal Subnet Error – the specified Subnet device failed.

0x88	SUBNET DEVICE FAILED TO ACKNOWLEDGE	Internal Subnet Error - the specified Subnet device failed to respond to the Gateway's polling.
0x89	SUBNET RESPONSE MALFORMED	Internal Subnet Error – a controller returned a malformed response.
0x8A	SUBNET RESPONSE TIMEOUT	Internal Subnet Error – a controller was unable to generate a response before timeout was reached.
0x8B	SUBNET RESPONSE INVALID CHECKSUM	Internal Subnet Error – a controller generated a response that has an invalid checksum.
0x8C	SUBNET DEVICE CONFLICT DETECTED	Internal Subnet Error – a Node ID conflict has been detected
0x8D	BUFFER OVERFLOW	Internal Gateway Error – Gateway buffer limit was exceeded
0x8E	FLASH FAILURE	Internal Gateway Error – Gateway flash memory failure
0x92	SUBNET16 ONLY COMMAND	A Subnet16-only command was issued when in MUX32 mode.
0x93	NODE MISMATCH ERROR	The Node ID specified in the command did not match the Node to which the command was sent.
0x94	CRC ERROR	Cyclic Redundancy Check Error
0x95	PROTOCOL ERROR	Internal Communications Error

Table 8-2: Error Code Table

8.3 ERROR RESPONSE PACKET STRUCTURE

Below is the packet structure of a CBx error response. Note that the one-byte **Error Code** value is returned in the MSB of the seventh data word.

ERROR RESPONSE ELEMENT	MSB	LSB
Overall Length: 2-byte value indicating the number of "*words*" in the Response Packet. This value will always be at least *7* words (6 + 1 for the error code).	0x00	0x07
Error Flag: 0xFF in the MSB indicates that an error occurred. **Error Information Byte**: 0xFF in the LSB indicates that a controller-based error occurred. Any value other than 0xFF indicates that a Gateway-based error occurred (and indicates the command that was attempted when the error occurred).	0xFF	0xFF
Instance Counter: This 1-byte value tallies the number of responses from a given Node ID. **Node ID Echo**: The 1-byte LSB value indicates the Node ID of the controller for which the command was intended.	<IC>	0x01
Month and Day Timestamp	<Month>	<Day>
Hour and Minute Timestamp	<Hour>	<Minute>
Seconds Timestamp in MSB **Number of Additional Bytes Retrieved** in LSB (*0x01 for error responses*)	<Seconds>	0x01
Error Code: 1-byte Error Code in MSB **0x00** in LSB	<Error Code>	0x00

Table 8-3: CBx Error Response Packet Structure

8.4 ERROR RESPONSE EXAMPLE

Below is an example of a typical controller generated error response following a failed *Read Data Command*. For this example, a "*tag not found*" error was generated.

ERROR RESPONSE PARAMETER	MSB	LSB

0x00 in MSB **Overall Length of Response** in LSB (*in words*)	0x00	0x07
Error Flag in MSB **Error Information Byte** in LSB	0xFF	0xFF
0x00 in MSB **Node ID Echo** in LSB	0x00	0x01
Month and **Day Timestamp**: (*March 19th*)	0x03	0x13
Hour and **Minute Timestamp** (*9:30: AM*)	0x09	0x1E
Seconds Timestamp in MSB (*:03 seconds*) **Number of Additional Bytes Retrieved** in LSB (*0x01 for error responses*)	0x03	0x01
Error Code in MSB: (*0x07 = "Tag not Found"*) **0x00 in LSB**	0x07	0x00

APPENDIX A: TECHNICAL SPECIFICATIONS

ELECTRICAL

Supply Voltage	10~30VDC
Power Consumption	120mA @ 24VDC (2.88W)

COMMUNICATION

Communication Interfaces	Point-to-Point: *RS485* Multi-drop: *Subnet16, MUX32, RS485* Ethernet: *Ethernet/IP, Modbus TCP, TCP/IP*
RFID Interface:	Gateway HF-Series RFID System
RF Output Power:	1W
Air Protocols	I-CODE 1, ISO 15693, ISO 14443 A
Air Protocol Speed:	26.5kBaud / 106kBaud with CRC error detection
RS485 Baud Rates	9600 (default), 19.2k, 38.4k, 57.6k, 115.2k

MECHANICAL

Dimensions	76mm x 89mm x 33mm
Weight	.24 KG (.53 lbs)
Enclosure:	Stainless Steel 304 (18-8)

ENVIRONMENTAL

Operating Temperature	-20° to 50°C (-4° to 122°F),
Storage Temperature	-40° to 85°C (-40° to 185°)
Humidity	90% Non-Condensing
Protection Class	IP31
Shock Resistance	IEC 68-2-27 Test EA 30g, 11 milliseconds, 3 shocks each axis
Vibration Resistance	IEC 68-2-6 Test FC 1.5mm; 10 to 55Hz; 2 hours each axis

NOTE: Specifications are subject to change without notice.

APPENDIX B: MODELS AND ACCESSORIES

Escort Memory Systems designs, manufactures and distributes a wide range of high frequency (HF) RFID equipment including RFID controllers, network interface modules (Gateways and Hubs), RFID tags and the cables needed to make it all work.

This portion of the manual lists the products and accessories available for the Gateway and HF-Series RFID product family. To purchase any of the items listed below contact your EMS distributor, call us directly at **(800) 626-3993** or visit our Web site: http://www.ems-rfid.com. Please let us know if you have any questions.

SUBNET16™ GATEWAY INTERFACE MODULES

GWY-01-IND-01

Subnet16™ Industrial Ethernet Gateway

GWY-01-TCP-01

Subnet16™ TCP/IP Gateway

RFID CONTROLLERS

HF-0405-Series RFID Controllers

There are **three** models of the **HF-0405 RFID Controller**:

HF-0405-232-01	for RS232 interface connections
HF-0405-422-01	for RS422 interface connections
HF-0405-485-01	for RS485 interface connections

Cobalt HF-Series RFID Controllers

There are **six** models of the **Cobalt HF RFID Controller**:

HF-CNTL-232-01	for RS232 interface connections
HF-CNTL-422-01	for RS422 interface connections
HF-CNTL-485-01	for RS485 interface connections
HF-CNTL-USB-01	for USB interface connections
HF-CNTL-IND-01	for Industrial Ethernet and standard TCP/IP connections
HF-CNTL-DNT-01	for DeviceNet connections

Cobalt HF-Series Antennas

There are **four** models of the **Cobalt HF Antenna**:

HF-ANT-1010-01	10cm x 10cm
HF-ANT-2020-01	20cm x 20cm
HF-ANT-3030-01	30cm x 30cm
HF-ANT-0750-01	7cm x 50cm (for conveyor applications)

Cobalt C0405-Series RFID Controllers

There are **four** models of the **Cobalt C0405 RFID Controller**:

C0405-232-01	for RS232 interface connections
C0405-422-01	for RS422 interface connections
C0405-485-01	for RS485 interface connections
C0405-USB-01	for USB 2.0 interface connections

Cobalt C1007-Series RFID Controllers

There are **four** models of the **Cobalt C1007 RFID Controller**:

C1007-232-01	for RS232 interface connections
C1007-422-01	for RS422 interface connections
C1007-485-01	for RS485 interface connections
C1007-USB-01	for USB 2.0 interface connections

SOFTWARE & DEMONSTRATION KITS

Software Applications

Cobalt HF TCP/IP Dashboard

Communicate in real time with one or more readers directly or via Multi-drop network. Allows users to configure, monitor and control their RFID devices from anywhere on their network.

C-Macro Builder

An easy to use GUI-driven utility that provides rapid development and implementation of custom RFID macros.

NOTE:

Software utilities and User's Manuals are available at www.ems-rfid.com

Demonstration Kits

00-1163

RS232 Demo Display Kit (includes one HF-0405-232 controller, power supply and display board).

00-1164

RS485 TCP/IP Gateway Demo Display Kit (includes three HF-0405-485 controllers, one GWY-01-TCP-01 Gateway, display board, cables, power supply and carrying case).

POWER SUPPLIES

00-1166

1.88A max @ 24VDC (45W), Universal Input (90-264VAC, 47-63Hz), 5.5x2.5mm plug, positive tip; Note: Requires country specific power cord to mate to IEC 320 power cord receptacle.

00-1167

4.17A max @ 24VDC (100W), Universal Input (90-264VAC, 47-63Hz), 5.5x2.5mm plug, positive tip; Note: Requires country specific power cord to mate to IEC 320 power cord receptacle.

00-1168

5.0A max @ 24VDC (120W), Universal Input (88-132VAC/176-264VAC switch selectable, 47-63Hz) DIN Rail Mount; Note: AC wire receptacles are spring clamp for direct wire connection.

EMS RFID TAGS

Escort Memory Systems designs and manufactures several lines of RFID tags. LRP, HMS and T-Series passive read/write RFID tags are specially suited for the Gateway and EMS RFID Controllers.

SUBNET16 CABLES & ACCESSORIES

EMS P/N	DESCRIPTION
CBL-1478	Cable Assembly: RS232/Power (2.5mm DC Jack), 2m
CBL-1480-XX	Cable: M12, 5-pin, Male/Female, ThinNet
CBL-1481-XX	Cable: M12, 5-pin, Male/Male, ThinNet
CBL-1481-02	Cable: M12, 5-pin, Male/Male, ThinNet, 2m (Gateway to Drop-T)
CBL-1482-XX	Cable: M12, 5-pin, Male/Right-Angle Female, ThinNet
CBL-1483-XX	Cable: 7/8–16, 5-pin, Male/Female, ThickNet
CBL-1484-XX	Cable: 7/8-16, 5-pin, Right-Angle Male/Bare Wire, ThickNet
CBL-1485	Drop-T Connector: 5-pin, 7/8-16 F / M12 F / 7/8-16 M (ThickNet to ThinNet)
CBL-1486	Drop-T Connector: 5-pin, M12, F/F/M (ThinNet to ThinNet)
CBL-1487	Field Mountable Connector: 5-pos, Straight Female M12,
CBL-1488-XX	Cable: 8-pin, Female M12 / Bare Wires
CBL-1489	Termination Resistor Plug: 7/8-16, Male, 5-pin, (ThickNet)
CBL-1490	Termination Resistor Plug: M12, Male, 5-pin, (ThinNet)
CBL-1491	Connector: 5-pos, Right-Angle Female M12, Field Mountable
CBL-1492-XX	Cable: 8-pin, Right-Angle Female M12 / Bare Wires
CBL-1493	Connector: 8-pos, Straight Female M12, Field Mountable
CBL-1494-01	Cable: M12, 5P, F/Bare Wire Leads, ThinNet, 1M
CBL-1495-XX	Cable: 7/8-16, 5P F/Bare Wire Leads
CBL-1496	Plug: Termination Resistor, M12, 5P, F
CBL-1497	Plug: Termination Resistor, 7/8-16, 5P, F
CBL-1498-02	Cable: M12, 5P, M/Bare Wire Leads, THINNET, 2M
CBL-1513	Cable Assembly: M12, 5-Pin, Male, Reverse Keyed to Type A, USB, 3M
CBL-1514	Connector: M12, Male, 5-Pin, Straight, Reverse Keyed (for USB)
CBL-1515-05	Cable: Category 5E Shielded Ethernet/M12, 5-Pin, Male, D-Code, 5M

XX = Length in Meters

APPENDIX C: NETWORK DIAGRAMS

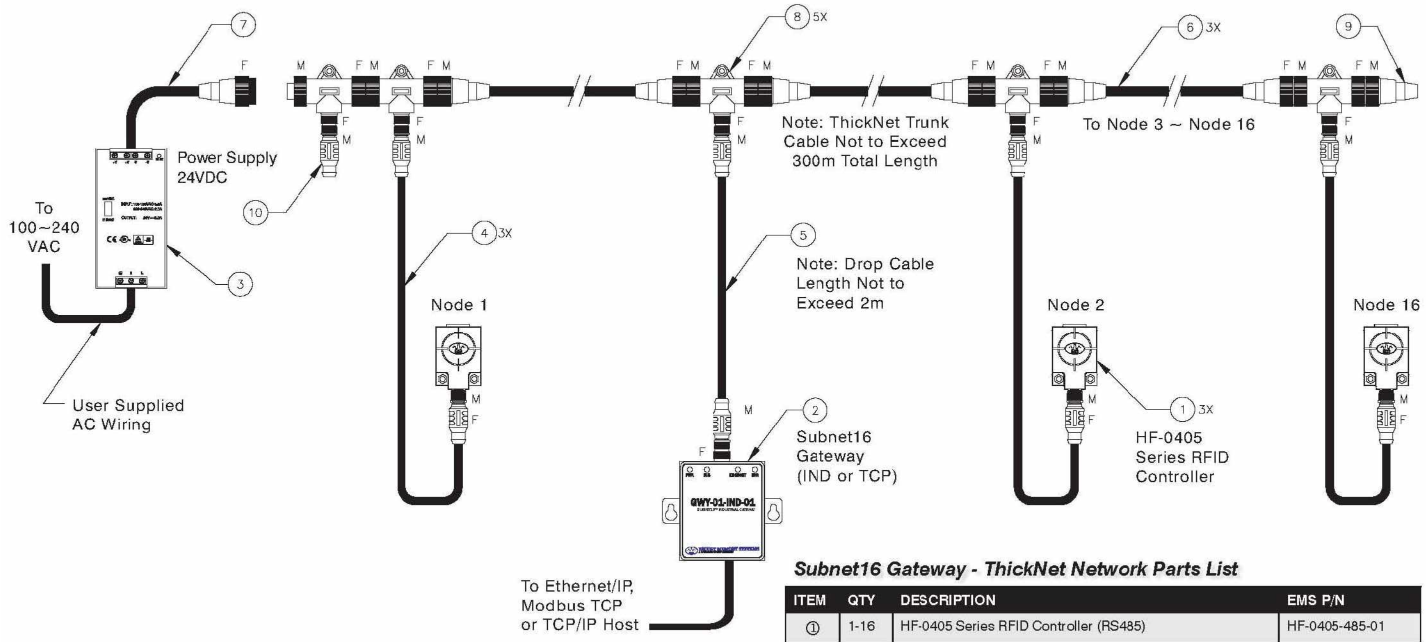
Subnet16 Gateway: HF-0405-Series ThickNet Network Diagram

Subnet16 Gateway: HF-0405-Series ThinNet Network Diagram

Subnet16 Gateway: Cobalt HF-Series ThickNet Network Diagram

Subnet16 Gateway – HF-0405-Series ThickNet Network

SUBNET16 GATEWAY - THICKNET NETWORK DIAGRAM



Subnet16 Gateway - ThickNet Network Parts List

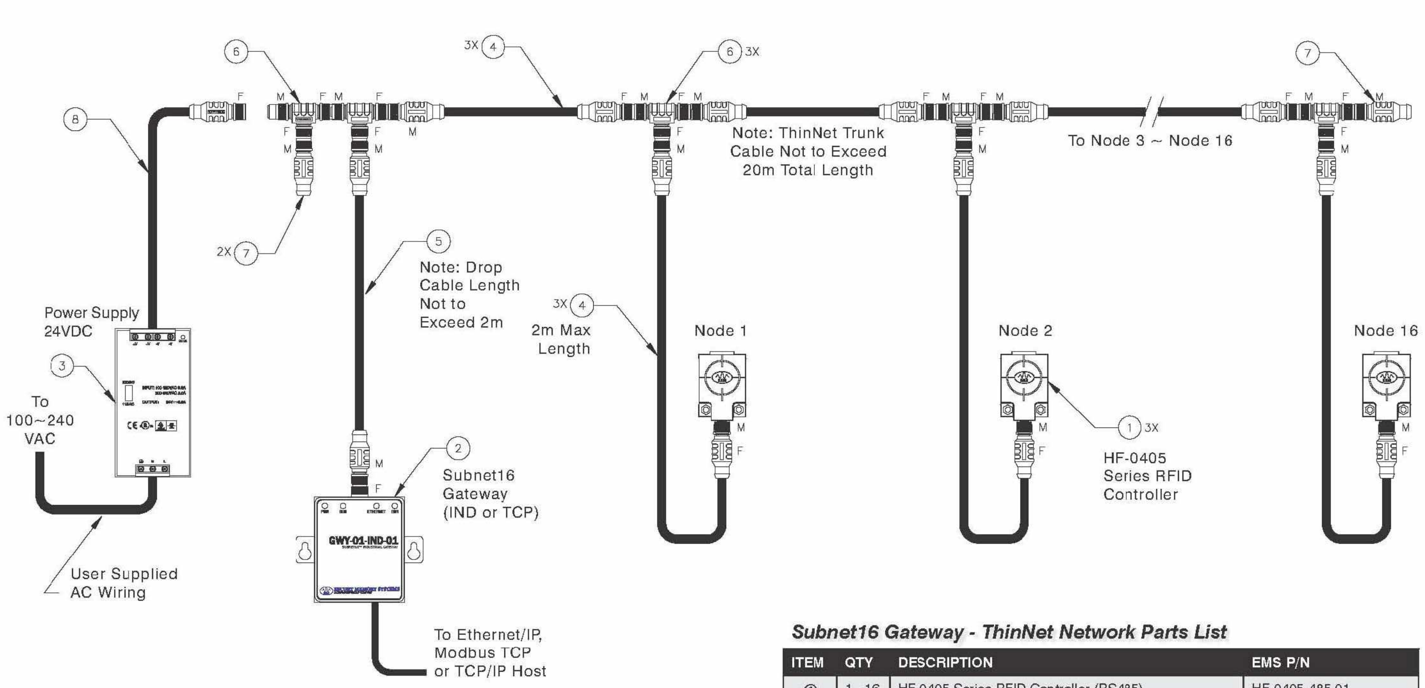
ITEM	QTY	DESCRIPTION	EMS P/N
①	1-16	HF-0405 Series RFID Controller (RS485)	HF-0405-485-01
②	1	Subnet16 Gateway (Industrial or TCP/IP)	GWY-01-IND/TCP-01
③	1	Power Supply (120W, 24VDC, 5.0A Max)	00-1168
④	1~16	Drop Cable (Male/Female, 5-pin, M12, ThinNet)	CBL-1480-02
⑤	1	Drop Cable (Male/Male, 5-pin, M12, ThinNet)	CBL-1481-02
⑥	1~15	Trunk Cable (Male/Female, 5-pin, 7/8 - 16, ThickNet)	CBL-1483-XX
⑦	1	Cable (Female, 5-pin, 7/8 - 16, ThickNet/Bare Wire Leads)	CBL-1495-XX
⑧	1~18	Drop-T Connector (ThickNet/ThinNet)	CBL-1485
⑨	1	Termination Resistor (Male, 7/8 - 16, ThickNet)	CBL-1489
⑩	1	Termination Resistor (Male, M12, ThinNet)	CBL-1490

XX = Length in Meters

ESCORT MEMORY SYSTEMS
A Datalogic Group Company
ems

Subnet16 Gateway – HF-0405-Series ThinNet Network

SUBNET16 GATEWAY - THINNET NETWORK DIAGRAM



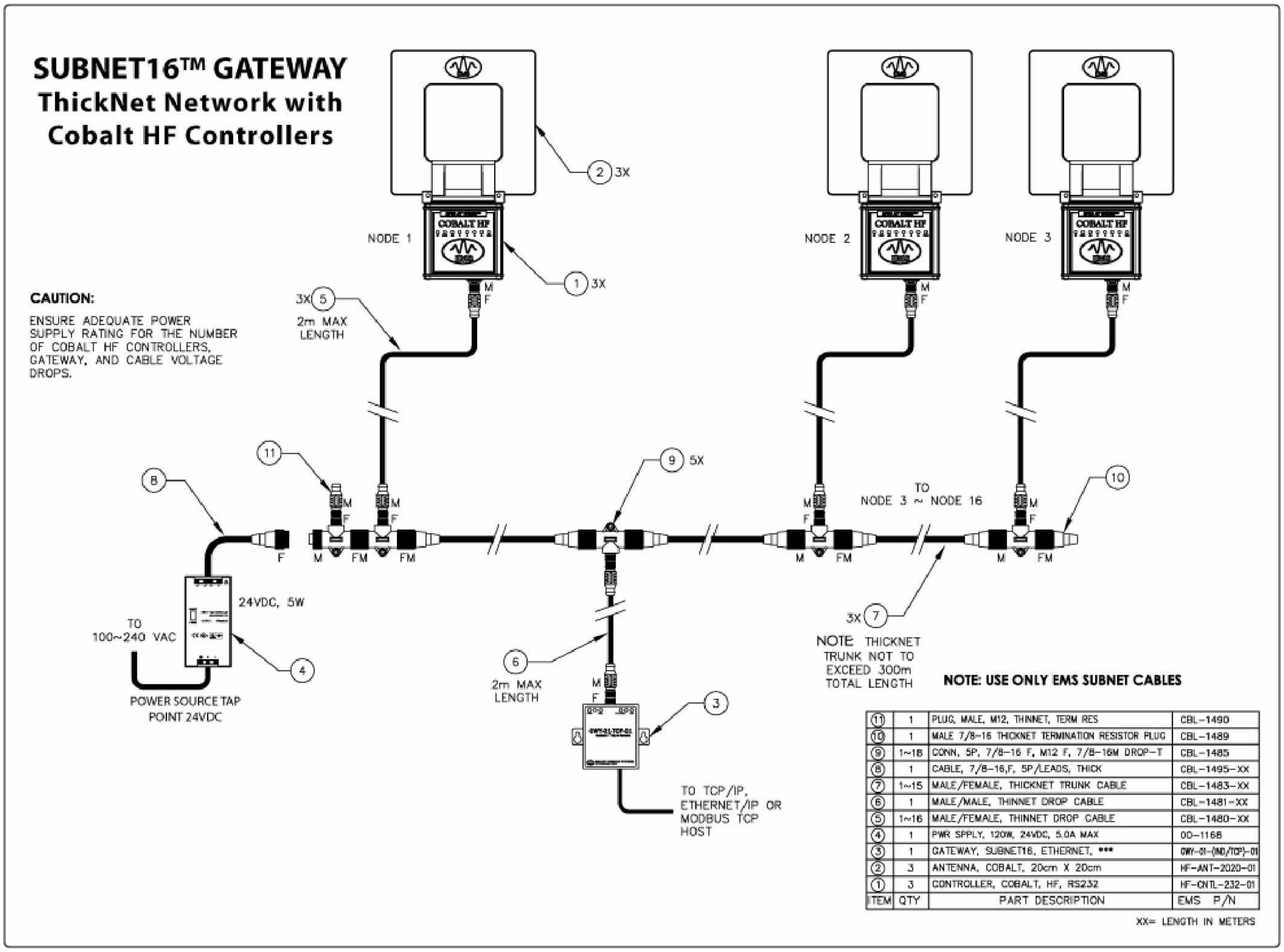
Subnet16 Gateway - ThinNet Network Parts List

ITEM	QTY	DESCRIPTION	EMS P/N
①	1~16	HF-0405 Series RFID Controller (RS485)	HF-0405-485-01
②	1	Subnet16 Gateway (Industrial or TCP/IP)	GWY-01-IND/TCP-01
③	1	Power Supply (120W, 24VDC, 5.0A Max)	00-1168
④	1~16	Drcp Cable (Male/Female, 5-pin, M12, ThinNet, 2m)	CBL-1480-02
④	1~16	Trunk Cable (Male/Female, 5-pin, M12, ThinNet)	CBL-1480-XX
⑤	1	Drcp Cable (Male/Male, 5-pin, M12, ThinNet)	CBL-1481-02
⑥	1~17	Drcp-T Connector (ThinNet/ThinNet)	CBL-1486
⑦	2	Termination Resistor (Male, M12, ThinNet)	CBL-1490
⑧	1	Cable (Female, 5-pin, M12, ThinNet/Bare Wire Leads)	CBL-1494-XX

XX = Length in Meters

ESCORT MEMORY SYSTEMS
A Datalogic Group Company

Subnet16 Gateway – Cobalt HF-Series ThickNet Network

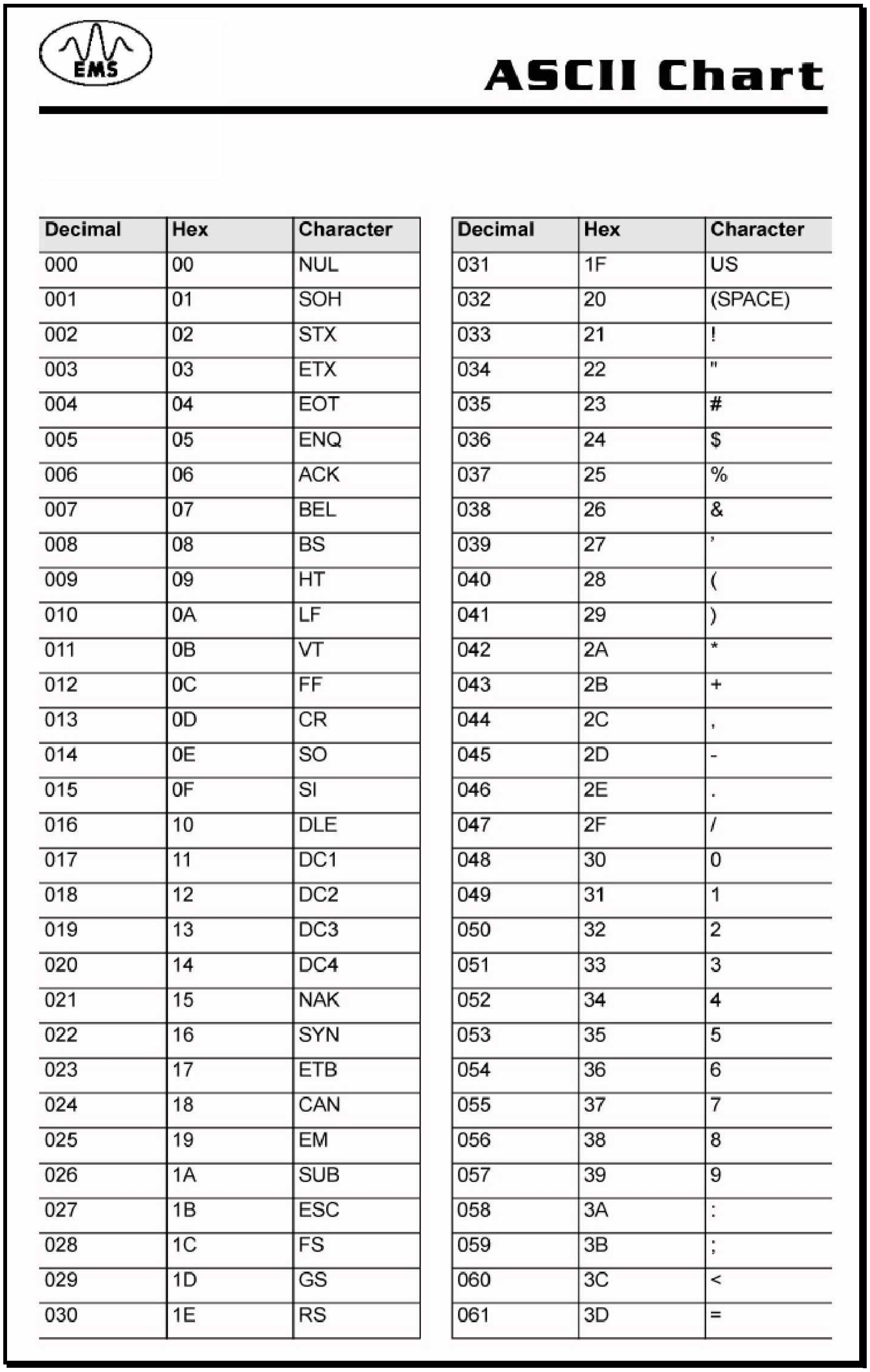

SUBNET16™ GATEWAY
ThickNet Network with
Cobalt HF Controllers
CAUTION:
ENSURE ADEQUATE POWER SUPPLY RATING FOR THE NUMBER OF COBALT HF CONTROLLERS, GATEWAY, AND CABLE VOLTAGE DROPS.
NODE 1
NODE 2
NODE 3
COBALT HF
2 3X
1 3X
3X 5
2m MAX LENGTH
M
F
11
8
9 5X
10
TO NODE 3 ~ NODE 16
FM
24VDC, 5W
TO 100~240 VAC
4
POWER SOURCE TAP POINT 24VDC
6
2m MAX LENGTH
3
TO TCP/IP, ETHERNET/IP OR MODBUS TCP HOST
3X 7
NOTE: THICKNET TRUNK NOT TO EXCEED 300m TOTAL LENGTH

NOTE: USE ONLY EMS SUBNET CABLES

⑪	1	PLUG, MALE, M12, THINNET, TERM RES	CBL-1490
⑩	1	MALE 7/8-16 THICKNET TERMINATION RESISTOR PLUG	CBL-1489
⑨	1~18	CONN, 5P, 7/8-16 F, M12 F, 7/8-16M DROP-T	CBL-1485
⑧	1	CABLE, 7/8-16,F, 5P/LEADS, THICK	CBL-1495-XX
⑦	1~15	MALE/FEMALE, THICKNET TRUNK CABLE	CBL-1483-XX
⑥	1	MALE/MALE, THINNET DROP CABLE	CBL-1481-XX
⑤	1~16	MALE/FEMALE, THINNET DROP CABLE	CBL-1480-XX
④	1	PWR SPPLY, 120W, 24VDC, 5.0A MAX	00-1168
③	1	GATEWAY, SUBNET16, ETHERNET, ***	GWY-01-(IND/TCP)-01
②	3	ANTENNA, COBALT, 20cm X 20cm	HF-ANT-2020-01
①	3	CONTROLLER, COBALT, HF, RS232	HF-CNTL-232-01
ITEM	QTY	PART DESCRIPTION	EMS P/N

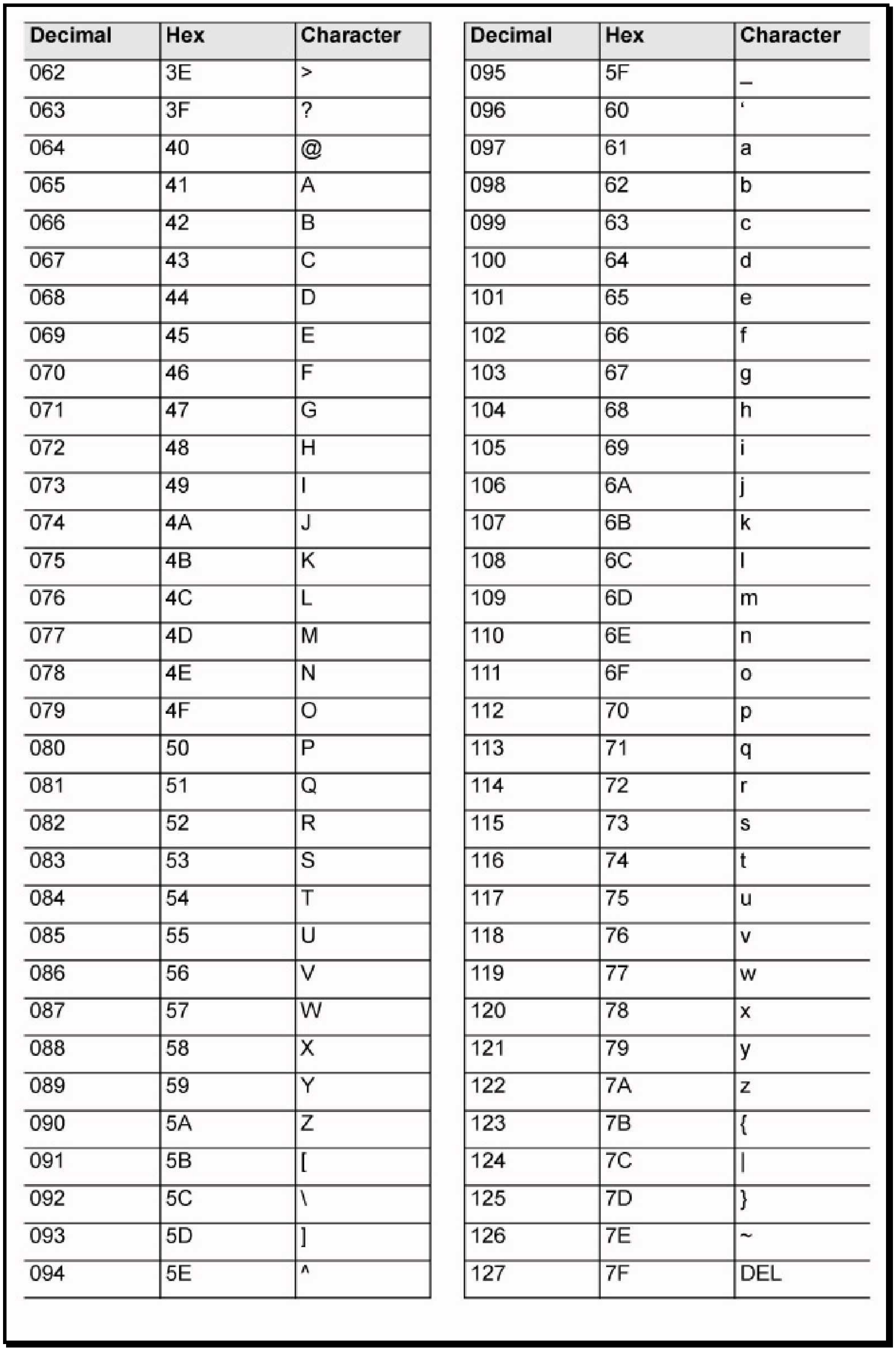
XX= LENGTH IN METERS

APPENDIX D: ASCII CHART



ASCII Chart

Decimal	Hex	Character	Decimal	Hex	Character
000	00	NUL	031	1F	US
001	01	SOH	032	20	(SPACE)
002	02	STX	033	21	!
003	03	ETX	034	22	"
004	04	EOT	035	23	#
005	05	ENQ	036	24	$
006	06	ACK	037	25	%
007	07	BEL	038	26	&
008	08	BS	039	27	'
009	09	HT	040	28	(
010	0A	LF	041	29	)
011	0B	VT	042	2A	*
012	0C	FF	043	2B	+
013	0D	CR	044	2C	,
014	0E	SO	045	2D	-
015	0F	SI	046	2E	.
016	10	DLE	047	2F	/
017	11	DC1	048	30	0
018	12	DC2	049	31	1
019	13	DC3	050	32	2
020	14	DC4	051	33	3
021	15	NAK	052	34	4
022	16	SYN	053	35	5
023	17	ETB	054	36	6
024	18	CAN	055	37	7
025	19	EM	056	38	8
026	1A	SUB	057	39	9
027	1B	ESC	058	3A	:
028	1C	FS	059	3B	;
029	1D	GS	060	3C	<
030	1E	RS	061	3D	=



Decimal	Hex	Character	Decimal	Hex	Character
062	3E	>	095	5F	_
063	3F	?	096	60	‘
064	40	@	097	61	a
065	41	A	098	62	b
066	42	B	099	63	c
067	43	C	100	64	d
068	44	D	101	65	e
069	45	E	102	66	f
070	46	F	103	67	g
071	47	G	104	68	h
072	48	H	105	69	i
073	49	I	106	6A	j
074	4A	J	107	6B	k
075	4B	K	108	6C	l
076	4C	L	109	6D	m
077	4D	M	110	6E	n
078	4E	N	111	6F	o
079	4F	O	112	70	p
080	50	P	113	71	q
081	51	Q	114	72	r
082	52	R	115	73	s
083	53	S	116	74	t
084	54	T	117	75	u
085	55	U	118	76	v
086	56	V	119	77	w
087	57	W	120	78	x
088	58	X	121	79	y
089	59	Y	122	7A	z
090	5A	Z	123	7B	{
091	5B	[	124	7C	|
092	5C	\	125	7D	}
093	5D	]	126	7E	~
094	5E	^	127	7F	DEL

EMS WARRANTY

Escort Memory Systems warrants that all products of its own manufacturing conform to Escort Memory Systems' specifications and are free from defects in material and workmanship when used under normal operating conditions and within the service conditions for which they were furnished. The obligation of Escort Memory Systems hereunder shall expire one (1) year after delivery, unless otherwise specified, and is limited to repairing, or at its option, replacing without charge, any such product that in Escort Memory Systems' sole opinion proves to be defective within the scope of this Warranty.

In the event Escort Memory Systems is not able to repair or replace defective products or components within a reasonable time after receipt thereof, Buyers shall be credited for their value at the original purchase price. Escort Memory Systems must be notified in writing of the defect or nonconformity within the warranty period and the affected product returned to Escort Memory Systems factory or to an authorized service center within thirty (30) days after discovery of such defect or nonconformity. Shipment shall not be made without prior authorization by Escort Memory Systems.

This is Escort Memory Systems' sole warranty with respect to the products delivered hereunder. No statement, representation, agreement or understanding oral or written, made by an agent, distributor, representative, or employee of Escort Memory Systems which is not contained in this warranty, will be binding upon Escort Memory Systems, unless made in writing and executed by an authorized Escort Memory Systems employee.

Escort Memory Systems makes no other warranty of any kind what so ever, expressed or implied, and all implied warranties of merchantability and fitness for a particular use which exceed the aforementioned obligation are here by disclaimed by Escort Memory Systems and excluded from this agreement.

Under no circumstances shall Escort Memory Systems be liable to Buyer, in contract or in tort, for any special, indirect, incidental, or consequential damages, expenses, losses or delay however caused. Equipment or parts that have been subject to abuse, misuse, accident, alteration, neglect, unauthorized repair or installation are not covered by warranty. Escort Memory Systems shall make the final determination as to the existence and cause of any alleged defect. No liability is assumed for expendable items such as lamps and fuses.

No warranty is made with respect to equipment or products produced to Buyer's specification except as specifically stated in writing by Escort Memory Systems in the contract for such custom equipment. This warranty is the only warranty made by Escort Memory Systems with respect to the goods delivered hereunder, and may be modified or amended only by a written instrument signed by a duly authorized officer of Escort Memory Systems and accepted by the Buyer.

Extended warranties of up to four years are available for purchase for most Escort Memory Systems products. Contact Escort Memory Systems or your distributor for more information.

Escort Memory Systems™ and the Escort Memory Systems logo are registered trademarks of Escort Memory Systems, a Datalogic Group Company.

ESCORT MEMORY SYSTEMS
A Datalogic Group Company
ems

170 Technology Circle
Scotts Valley, CA 95066 USA
Telephone: (831) 438-7000
FAX: (831) 438-5768
Website: www.ems-rfid.com
Email: info@ems-rfid.com