AMCI
ADVANCED MICRO CONTROLS INC.

NR25
Networked Resolver
for DeviceNet™

User Manual

AMCI Resolver Based Encoders
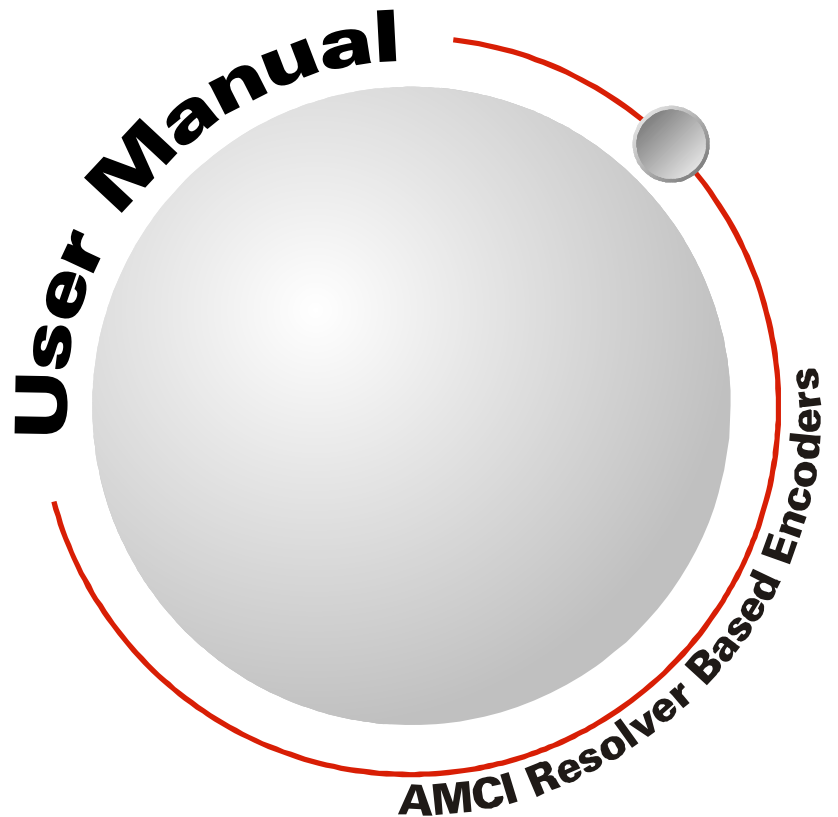
GENERAL INFORMATION

Important User Information

The products and application data described in this manual are useful in a wide variety of different applications. Therefore, the user and others responsible for applying these products described herein are responsible for determining the acceptability for each application. While efforts have been made to provide accurate information within this manual, AMCI assumes no responsibility for the application or the completeness of the information contained herein.

UNDER NO CIRCUMSTANCES WILL ADVANCED MICRO CONTROLS, INC. BE RESPONSIBLE OR LIABLE FOR ANY DAMAGES OR LOSSES, INCLUDING INDIRECT OR CONSEQUENTIAL DAMAGES OR LOSSES, ARISING FROM THE USE OF ANY INFORMATION CONTAINED WITHIN THIS MANUAL, OR THE USE OF ANY PRODUCTS OR SERVICES REFERENCED HEREIN.

No patent liability is assumed by AMCI, with respect to use of information, circuits, equipment, or software described in this manual.

The information contained within this manual is subject to change without notice.

This manual is copyright 2014 by Advanced Micro Controls Inc. You may reproduce this manual, in whole or in part, for your personal use, provided that this copyright notice is included. You may distribute copies of this complete manual in electronic format provided that they are unaltered from the version posted by Advanced Micro Controls Inc. on our official website: *www.amci.com*. You may incorporate portions of this documents in other literature for your own personal use provided that you include the notice "Portions of this document copyright 2014 by Advanced Micro Controls Inc." You may not alter the contents of this document or charge a fee for reproducing or distributing it.

Standard Warranty

ADVANCED MICRO CONTROLS, INC. warrants that all equipment manufactured by it will be free from defects, under normal use, in materials and workmanship for a period of [18] months. Within this warranty period, AMCI shall, at its option, repair or replace, free of charge, any equipment covered by this warranty which is returned, shipping charges prepaid, within eighteen months from date of invoice, and which upon examination proves to be defective in material or workmanship and not caused by accident, misuse, neglect, alteration, improper installation or improper testing.

The provisions of the "STANDARD WARRANTY" are the sole obligations of AMCI and excludes all other warranties expressed or implied. In no event shall AMCI be liable for incidental or consequential damages or for delay in performance of this warranty.

Returns Policy

All equipment being returned to AMCI for repair or replacement, regardless of warranty status, must have a Return Merchandise Authorization number issued by AMCI. Call (860) 585-1254 with the model number and serial number (if applicable) along with a description of the problem during regular business hours, Monday through Friday, 8AM - 5PM Eastern. An "RMA" number will be issued. Equipment must be shipped to AMCI with transportation charges prepaid. Title and risk of loss or damage remains with the customer until shipment is received by AMCI.

24 Hour Technical Support Number

24 Hour technical support is available on this product. If you have internet access, start at www.amci.com. Product documentation and FAQ's are available on the site that answer most common questions.

If you require additional technical support, call (860) 583-7271. Your call will be answered by the factory during regular business hours, Monday through Friday, 8AM - 5PM Eastern. During non-business hours an automated system will ask you to enter the telephone number you can be reached at. Please remember to include your area code. The system will page an engineer on call. Please have your product model number and a description of the problem ready before you call.

We Want Your Feedback

Manuals at AMCI are constantly evolving entities. Your questions and comments on this manual are both welcomed and necessary if this manual is to be improved. Please direct all comments to: Technical Documentation, AMCI, 20 Gear Drive, Terryville CT 06786, or fax us at (860) 584-1973. You can also e-mail your questions and comments to *techsupport@amci.com*

TABLE OF CONTENTS

ADVANCED MICRO CONTROLS INC.

ABOUT THIS MANUAL

> **Read this chapter to learn how to navigate through this manual and familiarize yourself with the conventions used in it. The last section of this chapter highlights the manual's remaining chapters and their target audience.**

Audience

This manual explains the installation and operation of AMCI's NR25 Networked Resolver encoders for DeviceNet™. It is written for the engineer responsible for incorporating the NR25 into a design as well as the engineer or technician responsible for its actual installation. If there are any unanswered questions after reading this manual, call the factory. An applications engineer will be available to assist you.

Navigating this Manual

This manual is designed to be used in both printed and on-line forms. Its on-line form is a PDF document, which requires Adobe Acrobat Reader version 7.0+ to open it. If you decide to print out this manual, all chapters contain an even number of pages which allows you to easily print out a single chapter on a duplex (two-sided) printer.

Bookmarks of all the chapter names, section headings, and sub-headings are in the PDF file to help you navigate through it. The bookmarks should have appeared when you opened the file. If they didn't, press the F5 key on Windows platforms to bring them up. Throughout this manual you will also find blue text that functions as a hyperlink in HTML documents. Clicking on the text will immediately jump you to the referenced section of the manual. If you are reading a printed manual, most links include page numbers.

The PDF file is password protected to prevent changes to the document. You are allowed to select and copy sections for use in other documents and, if you own Adobe Acrobat version 7.0 or later, you are allowed to add notes and annotations.

Manual Conventions

Three icons are used to highlight important information in the manual:

NOTE ▶   **NOTES** highlight important concepts, decisions you must make, or the implications of those decisions.

⚠ CAUTION   **CAUTIONS** tell you when equipment may be damaged if the procedure is not followed properly.

⚠ WARNING   **WARNINGS** tell you when people may be hurt or equipment may be damaged if the procedure is not followed properly.

The following table shows the text formatting conventions:

Format	Description
Normal Font	Font used throughout this manual.
Emphasis Font	Font used for parameter names and the first time a new term is introduced.
Cross Reference	When viewing the PDF version of the manual, clicking on the cross reference text jumps you to referenced section.
HTML Reference	When viewing the PDF version of the manual, clicking on the HTML reference text jumps you to the referenced section of the AMCI website if you have Internet access. You may be asked to confirm access before the program brings you to the AMCI website.

Trademarks and Other Legal Stuff

The AMCI logo is a trademark of Advanced Micro Controls Inc. "DeviceNet" and "CIP" are trademarks of Open DeviceNet Vendor Association, Inc. "EtherNet/IP" is a trademark of ControlNet International, Ltd. under license by Open DeviceNet Vendor Association, Inc. "Adobe" and "Acrobat" are registered trademarks of Adobe Systems Incorporated.

All other trademarks contained herein are the property of their respective holders.

Revision Record

This manual, 940-0D113 is the fourth release of this manual. It changes terminology used in the manual. It was first released on February 17th, 2014.

Revision History

940-0D112: 01/20/2014   Figure 2.15: BUS IN pinout error

940-0D111: 12/12/2013   Explicit message data format clarifications and corrections

940-0D110: 10/04/2013   Initial Release

Where to Go From Here

This manual contains information that is of interest to everyone from engineers to operators. The table below gives a brief description of the content of each chapter to help you find the information you need to assist you in your job.

CHP NUM.	Chapter Title	Chapter Description
1	Introduction to the NR25	Intended for anyone new to the NR25 Networked Resolver encoder, this chapter gives a basic overview of the unit, including an explanation of its programmable features. The chapter also explains the NR25 part numbering system.
2	Installation	This chapter is intended for the engineer or technician responsible for installing and wiring the NR25 Networked Resolver encoder. Information in this chapter includes mechanical drawings, installation guidelines and connector pinout.
3	RSNetWorx for DeviceNet	This chapter is intended for anyone responsible for configuring a DeviceNet network that includes an NR25 either before or after actual installation. This chapter includes information on configuring the NR25 from the RSNetWorx software.
4	I/O Messaging Commands	This chapter covers how to communicate with the NR25 using data that is mapped through the DeviceNet scanner's scan list.
5	Explicit Messaging Commands	This chapter covers how to communicate with the NR25 using explicit messaging. This method is best for "one-shot" commands and does not use data that is mapped through the DeviceNet scanner's scan list.
A	CIP Position Sensor Object	When using the DeviceNet protocol, the NR25 conforms to the Encoder Device Profile as defined by the Common Industrial Protocol (CIP) specification. This chapter explains the Position Sensor Object that is implemented by the NR25 as part of this profile.
B	Additional CIP Objects	This appendix lists additional CIP objects whose use may be required in large, complex, or specialized applications

Overview

The NR25 is a new line of heavy-duty resolver based encoder products from AMCI. This manual is for the NR25 encoders with a DeviceNet interface. AMCI offers NR25 encoders for other network protocols such as EtherNet/IP, Modbus TCP, Profibus, and CANOpen. These products are covered in separate manuals. Go to www.amci.com for a full listing of available networks and protocols.

The NR25 series is composed of absolute single- and multi-turn sensors in an IP67 rated, 2.5 inch diameter package. All NR25 Networked Resolvers offer a maximum single turn position resolution of 16 bits, which is programmable from 1 to 65,536 counts per turn. Two multi-turn units are available in addition to the single turn NR25. One encodes 4,096 turns (12 bit + 16 bit = 28 bit encoder) and the other encodes 16,384 turns (14 bit + 16 bit = 30 bit encoder).

Figure 1.1 NR25 Resolver Based Encoder

A flange mount unit with end connectors is shown in figure 1.1. The following mounting styles are available:

> ➤ Size 25 standard flange mount with 3/8", 1/4", or 10mm shafts
> ➤ Size 25 standard servo mount with 3/8", 1/4", or 10mm shafts
> ➤ Blind shaft mount for 3/8", 1/2", 10mm, and 12mm shafts, mounting bracket designed for 63mm B.C.
> ➤ AMCI face mount with 5/8" shaft for high radial and axial loads

End connect versions of all of these mounting styles are available. Body material is either aluminum with a powder coat finish or 316 stainless steel for use in caustic environments. Outline drawings of all of the packing options are available in the Outline Drawings section of the *Installation* chapter, starting on page 17.

The NR25 resolver based encoders for DeviceNet are designed for use with the DeviceNet thin cable media and mate with the DeviceNet micro (M12) connectors. DeviceNet thin cable media is commonly used for drop lines off of a trunk line on large installations or as the only media type on smaller installations. Thin cable media supports total cable lengths of up to 100 meters (330 feet).

Network address and bus speed are set with rotary and DIP switches that are accessible behind the rear cover of the NR25. Once the network address and speed is set, the remaining parameters of the NR25 are configured over its DeviceNet interface. Parameters allow you to set the count direction, format of the velocity data, number of counts per turn, and the rollover count, which is the number of counts before the position returns to zero.

During normal operation of the network, the NR25 supports Explicit, Bit-Strobe, Poll, and Cyclic connection types. The NR25 can support all of these connection types simultaneously.

All NR25 resolver based encoders have two status LED's to help you determine the state of the device. These LED's are always located on the back cover of the NR25.

> ➤ **Module Status (MS) –** Operating status of the NR25 itself
> ➤ **Network Status (NS) –** Operating state of the DeviceNet protocol

Overview (continued)

Module Status LED

LED State	Definition
Off	No Power
Alternating Red/Green	Self-Test (Run on power up.)
Steady Green	Device Operational
Steady Red	Hardware Fault. (Cycle power. If fault persists, contact AMCI for support.)

Table 1.1 Module Status LED States

Network Status LED

LED State	Definition
Off	No Power
Alternating Red/Green	Power up Self-Test
Flashing Green	No established connections, unit in Idle state
Steady Green	Connection established
Flashing Red	Network Connection Timeout
Steady Red	Duplicate MAC address on network.

Table 1.2 Network Status LED States

Available Data

Position Data

All NR25 encoders offer position data with a maximum resolution of 65,536 counts per turn. Position data can be scaled to engineering units. The position data can also be preset which allows you to align the position data with your machine position without having to physically rotate the shaft.

Velocity Data

Velocity data is also available from the NR25. Velocity data can be scaled to engineering units and is available as pulses/millisecond, pulses/second, pulses/minute, or RPM.

Time Stamp

The time stamp is an unsigned double integer value with an interval of 100 nanoseconds. It will roll over every 429.4967296 seconds. The time stamp can be used to verify active communications between the NR25 and your host controller.

Actual Sensor Reading

This unsigned double integer value is the raw position data from the NR25. Changing the position scaling parameters or presetting the position will have no effect on this value.

Electrical Specifications

Operating Voltage (Supplied over Network)

11 Vdc to 25 Vdc (24 Vdc nominal)

Power Requirements

1.5W max.
50mA @ 24Vdc typical

CIP Implementation

Encoder Device Profile

Supported Network Data Rates

Supports 125, 250, and 500 Kbps

Supported Connection Types

Explicit, Poll, Bit-Strobe, and Cyclic

Single Turn Resolution

Programmable from 1 to 65,536 counts per turn
(16 bit resolution max.)

Multi-turn Resolution

4,096 turns (12 bit) or 16,384 (14 bit)

Direction of Increasing Counts

Default of CW increasing when looking at the shaft.

Programmable to CCW increasing over the
DeviceNet network.

Preset Position

Position can be preset to any value within its range
over the DeviceNet network. Internal Position
Offset can be stored in non-volatile memory and
retrieved on power up.

Positional Accuracy

±10 arc-minutes

Position Response Time

1 millisecond

Velocity Data Format

Programmable to pulses/minute, pulses/second,
pulses/millisecond, or RPM

Velocity Response Time

100 millisecond

Mechanical Specifications

Package Style

2.5 inch housing with flange, servo, or hub shaft
mounting

Connector Location

End

Mechanical Specifications (cont'd)

Housing

Powder coated aluminum or 316 stainless steel

Shaft

0.250", 0.375", 0.625", 6 mm or 10 mm
Blind shaft with 0.375", 0.500", 10 mm or 12 mm
hole

Max. Starting Torque @ 25°C

2.0 oz-in: 0.250", 0.375", 6 mm, and 10 mm shafts
6.0 oz-in: All blind shafts
6.0 oz-in: 0.625" shaft

Moment of Inertia (oz-in-sec^2)

6.00×10^{-4}: 0.250", 0.375", 6 mm, and 10 mm
shafts
7.00×10^{-4}: All blind shafts
8.50×10^{-4}: 0.625" shaft

Max. Operating Speed

6000 RPM max.

Max. Shaft Loading (0.625" shaft)

Axial: 50lbs. (222N)
Radial: 100lbs. (445N)

At specified max. loads, minimum bearing life is
2×10^9 revolutions.

Max. Shaft Loading (All other shafts)

Axial: 20lbs. (89N)
Radial: 40lbs. (178N)

At specified max. loads, minimum bearing life is
2×10^9 revolutions.

Environmental Specifications

Operating Temperature

–40°F to +185°F (–40°C to +85°C)

Shock

50g, 11 millisecond duration

Vibration

20g, 5 to 2000Hz

Enclosure Rating

IP67

Approximate Weight

2.0lbs. (0.91kg) 0.625" shaft - Aluminum Body
3.8lbs. (1.73kg) 0.625" shaft - 316 Steel Body
1.4lbs. (0.65kg) All other shafts - Aluminum Body
2.9lbs. (1.32kg) All other shafts - 316 Steel Body

Part Number Description
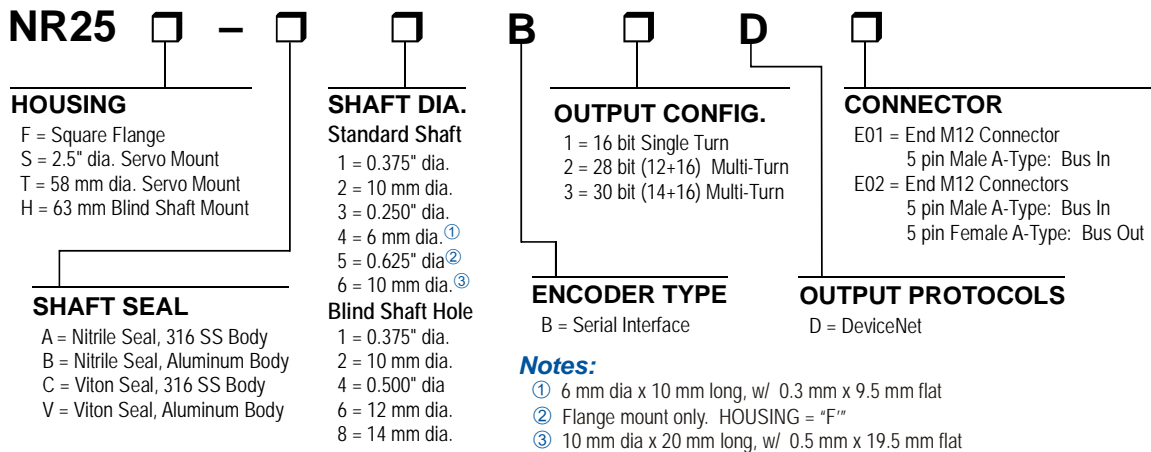
Figure 1.2 Part Number Description

Programmable Parameters

The following parameters are available on all NR25 encoders for DeviceNet. Note that most of these parameter names are pulled from the ODVA (CIP) specification. They are generic, and sometimes confusing, but they are what is defined in the specification. AMCI has decided to adopt these parameter names for all NR25s to avoid additional translations between protocols with one exception. In the ODVA specification, the parameter that sets the number of counts per turn of the shaft is called the 'Measurement Units per Span'. This generic name can be applied to both rotary and linear encoders. Being that the NR25 is a rotary encoder, this manual refers to the parameter as *Counts per Turn*.

Direction Counting Toggle

This parameter allows you to set the direction of shaft rotation needed to produce increasing counts.

➤ A value of "0" sets the direction of increasing counts to clockwise when looking at the shaft. This is the factory default value
➤ A value of "1" sets the direction of increasing counts to counter-clockwise when looking at the shaft.

Effects of Reversing Count Direction

Changing the *Direction Counting Toggle* parameter changes the way the position value is calculated. When you reverse the count direction, the position changes from your current position value to (Maximum number of counts – current position value). For example, assume a 30 bit NR25 with its default of 65,536 counts per turn. If the current position value is 100,000 and you change the Direction Counting Toggle parameter, the current position will change to (2^{30} – 100,000 = 1,073,741,824 – 100,000) = 1,073,641,824. Most applications do not require you to change the count direction after the machine is setup, so the count direction is typically set before the position value is preset.

Changing the count direction on your machine while maintaining the current position value is a three step process. First, read and store the current position value from the NR25. Second, change the *Direction Counting Toggle* value. Third, write the stored position value back to the NR25 as a preset value.

Programmable Parameters (continued)

Scaling Function Control

This parameter determines if the position value reported in the network data is scaled to engineering units or if it is reported at its full 16 bit resolution. Scaling is accomplished with the Counts Per Turn parameter. (see below)

➤ A value of "0" disables scaling and the position data is reported at 16 bit resolution.
➤ A value of "1" enables scaling and the position data is reported in the engineering units set by the *Counts per Turn* parameter.

NOTE ▶ Velocity data can be reported as pulses/second, pulses/millisecond, pulses/minute, or RPM. When reported as pulses/second, pulses/millisecond, or pulses/minute, the velocity data is always scaled by the *Counts per Turn* parameter. This is useful in applications where you require the full position resolution, but want to scale velocity data to an engineering value such as inches/second, or boxes/minute.

Counts per Turn
(ODVA: Measuring Units Per Span)

Used to scale the position and/or velocity data from the NR25.

➤ Values can range from 2 to 65,536 counts per turn for single turn NR25 units.
➤ Values can range from 1 to 65,536 counts per turn for multi-turn NR25 units.

Position data is only scaled if the *Scaling Function Control* parameter equals "1". Velocity data is always scaled.

The number of turns encoded by the NR25 is not controlled with this parameter. The *Total Measurement Range* parameter is used to control when the position data rolls over to zero, which controls the number of turns encoded by the NR25.

Total Measurement Range

⚠ CAUTION When using the Total Measurement Range parameter, it is important to save your parameter values to Flash memory while commissioning your machine. The position value will change when power to the NR25 is cycled if this parameter is not saved to Flash. This step is critically important if implicit messaging is not used to communicate with the NR25.

The *Total Measurement Range* parameter sets the total number of counts before the position value returns to zero. It is always used when determining the position value. If the *Total Measurement Range* parameter is left at its default value of zero, the roll over position is determined by the *Counts per Turn* parameter and the number of turns the NR25 can encode. If the *Total Measurement Range* is non-zero, it places an upper limit on the position value and the *Preset Value* parameter. The *Total Measurement Range* parameter has the following ranges:

➤ **Single Turn NR25**: Range of 0, 2 to 65,536
➤ **28 bit Multi-turn NR25:** Range of 0, 2 to 268,435,455
➤ **30 bit Multi-turn NR25:** Range of 0, 2 to 1,073,741,823

Note that the value of the *Counts per Turn* parameter does not limit the range of values the can be programmed into the *Total Measurement Range* parameter. There is no fixed relationship between the *Total Measurement Range* and *Counts per Turn* parameters, which leads to interesting applications that use the two parameters.

Programmable Parameters (continued)

Total Measurement Range (continued)

Roll Over on Fractional Number of Turns

When the *Total Measurement Range* is less than the total counts available from the NR25, the position will return to zero before the full mechanical travel is completed.

➤ When the *Scaling Function Control* parameter is disabled, the total counts available equals 65,536 multiplied by the number of turns the NR25 can encode.

➤ When the *Scaling Function Control* parameter is enabled, the total counts available equals the value of the *Counts per Turn* parameter multiplied by the number of turns the NR25 can encode.

28 bit NR25: 4096 Turns
Counts per Turn = 1,000 Counts
Total Counts = 4,096,000

Total Measurement Range = 819,200 Counts
NR25 outputs five cycles of counts from 0 to 819,199 over 4,096 turns

Figure 1.3 Fractional Turn Example 1

Figure 1.3 shows what occurs when the *Total Measurement Range* parameter is used to divide the full range of travel of the NR25 into equal parts. In this case, a twenty-eight bit NR25 has its 4,096 turns evenly divided into five cycles of 819.2 turns.

NOTE ⬧ If the value of {Total Counts ÷ *Total Measurement Range*} is an integer, the NR25 remains an absolute rotary sensor. You can remove power from the NR25, rotate it as far as you want, re-apply power, and the NR25 will give you the correct position value. Figure 1.3 is an example of this setup because the division of the two parameters results in a quotient value of five.

Programmable Parameters (continued)
Total Measurement Range (continued)

Roll Over on Fractional Number of Turns (continued)

One Rotation of a Single Turn NR25
Counts per Turn = 36,000 Counts

0   0   0   0

0

Total Measurement Range = 22,500 Counts
NR25 outputs one cycle of counts from 0 to 22,499 for every 225 degrees of rotation.
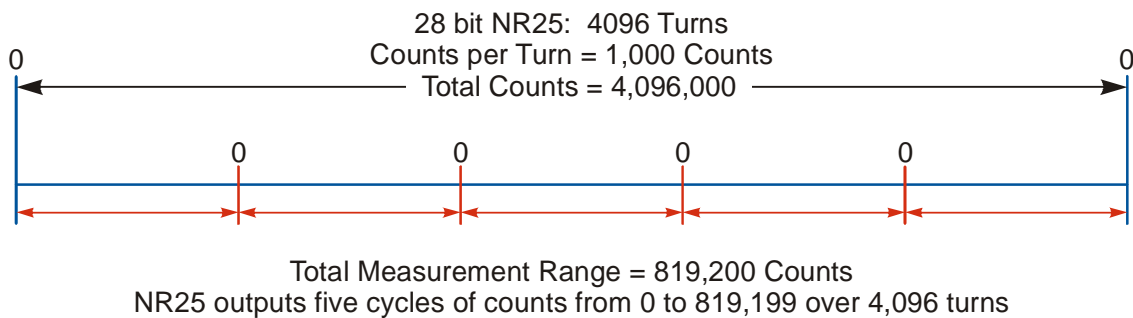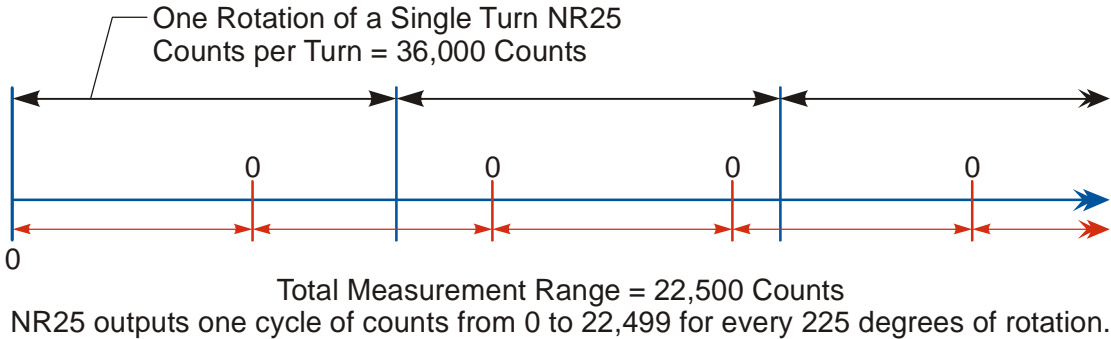
Figure 1.4 Fractional Turn Example 2

Figure 1.4 shows a single turn NR25 where the *Total Measurement Range* parameter is not used to divide the full range of travel into equal parts. In this case, the position value will roll over to zero after 225 degrees of rotation. In this example the value of {Total Counts ÷ *Total Measurement Range*} is a real number, 1.6, instead of an integer.

> **NOTE** ▷   When the {Total Counts ÷ *Total Measurement Range*} quotient is a real number, the NR25 will power up with the correct position value as long as the shaft is rotated less than half of the complete span of the encoder while power was removed. In practical terms:

➤ **For 30-bit NR25 multi-turn encoders:** If you remove power from the sensor and rotate the shaft such that the point that you stop at is less than 8,192 turns from the point where you removed power, the position reading will be correct when you re-apply power.

➤ **For 28-bit NR25 multi-turn encoders:** If you remove power from the sensor and rotate the shaft such that the point that you stop at is less than 2,048 turns from the point where you removed power, the position reading will be correct when you re-apply power.

➤ **For 16-bit NR25 single turn encoders:** If you remove power from the sensor and rotate the shaft such that the point that you stop at is less than 180 degrees from the point where you removed power, the position reading will be correct when you re-apply power.

> **NOTE** ▷   If the point that you re-apply power at is greater than the limits listed above, the position value from the NR25 will be off by at least ±1 turn.

Encoding Additional Turns

When the *Scaling Function Control* parameter is enabled and the *Counts per Turn* parameter is set to a value less than its maximum, the *Total Measurement Range* parameter can be used to require additional rotations of the shaft before the position value reaches the roll over count. For example, assume a single turn NR25 that has its *Counts per Turn* parameter set to 360 and its *Total Measurement Range* parameter set to 64,800. With this setup, the shaft of the NR25 must rotate 180 turns, {64,800 ÷ 360}, before the position returns to zero. In this application, the single turn NR25 acts as a 180 turn encoder with one degree position resolution.

The same trade off between resolution and number of turns encoded can be made with the multi-turn NR25 encoders. For example, if a 30-bit NR25 encoder has its *Counts per Turn* parameter set to 360 and its *Total Measurement Range* parameter set to its maximum of 1,073,741,824, the NR25 will encode 2,982,616.1$\overline{7}$ turns with one degree resolution.

In all of these applications, the NR25 as the same motion restrictions listed in the ***Roll Over on Fractional Number of Turns*** section above. Exceeding these limits will result in a position value error of at least ±1 turn when power is re-applied.

Programmable Parameters (continued)

Velocity Format

The NR25 can transmit velocity data over the network in addition to position data. This parameter sets the units of measure for the velocity data. This integer parameter has four fixed options.

> **16#1F04 (7,940 dec.)** = pulses/second
> **16#1F05 (7,941 dec.)** = pulses/millisecond
> **16#1F07 (7,943 dec.)** = pulses/minute
> **16#1F0F (7,951 dec.)** = RPM

NOTE ▶ Velocity data is always scaled by the *Counts per Turn* parameter. This is useful in applications where you require the full position resolution, but want to scale velocity data to an engineering value such as inches/second, or boxes/minute.

Preset Value

This parameter allows you to preset the position to any value in its single or multi-turn range without rotating the shaft. The range of values depends on the state of the *Scaling Function Control* parameter and the value of the *Total Measurement Range* parameter.

> When the *Scaling Function Control* parameter is disabled:

   If the *Total Measurement Range* parameter equals zero, the Preset Value range is:

   0 to 65,535 for single turn NR25 encoders

   0 to 268,435,455 for 28 bit multi-turn NR25 encoders

   0 to 1,073,741,823 for 30 bit multi-turn NR25 encoders

   If the *Total Measurement Range* parameter is not equal to zero, the Preset Value range is:

   0 to (value of *Total Measurement Range* parameter) - 1

> When the *Scaling Function Control* parameter is enabled:

   If the *Total Measurement Range* parameter equals zero, the Preset Value range is:

   0 to ({value of *Counts per Turn* parameter} - 1) for single turn NR25 encoders

   0 to ({value of *Counts per Turn* parameter * 4,096} - 1) for 28 bit multi-turn NR25 encoders

   0 to ({value of *Counts per Turn* parameter * 16,384} - 1) for 30 bit multi-turn NR25 encoders

   If the *Total Measurement Range* parameter is not equal to zero, the Preset Value range is:

   0 to (value of *Total Measurement Range* parameter) - 1

Device Type

You can program how the NR25 defines itself to the DeviceNet controller. This parameter has a double integer (32 bit) data type and two fixed values.

> **0x22** = Encoder Device (factory default value)
> **0x00** = Generic Device

Non-Volatile Memory

The NR25 has Flash memory available to store parameter values and the position offset that results from a Preset Position operation. These values are not automatically stored to flash when they are changed. You must issue a separate "Save to Flash" command to store these values.

> **NOTE** ▶ The Flash memory in the NR25 is guaranteed for 10,000 write cycles. Exceeding this limit will eventually lead to failure of the memory and the NR25 will no longer function properly.

This limit will not be an issue for most applications. Parameter values are rarely, if ever, changed once the machine is installed and commissioned. For those applications that often change parameter values, these values can be written down to the NR25 as part of the machine's power up sequence and never stored to Flash memory.

Saving the Position Offset Value

When the position is preset with a Preset Position command, the NR25 calculates an internal offset that is used to bring the reported position to the programmed Preset Value. This internal offset is not stored in Flash memory and will be lost when power is removed from the unit.

➤ If part of the machine's power up sequence is homing the position, then the machine's design assumes the position is incorrect on power up. Therefore, there is no reason to store the internal offset to Flash memory.

➤ Absolute position applications will usually preset the position during machine installation and commissioning. The "Save to Flash" command is then issued once to stored the internal offset. As long as there is no mechanical slippage in the machine, the position will always remain correct.

Some applications require the position to be preset multiple times while the machine is running and the last internal offset must be stored to non-volatile memory as part of the machine's power down sequence. If your application falls into this category, you have two options:

1) When you need to preset the position value from the NR25, calculate and store the required offset in your host controller. You will never issue a "Preset Position" command to the NR25. AMCI has sample code available to show you how to accomplish this with ladder logic.

2) Set the *Total Measurement Range* parameter to a non-zero value. When this parameter is non-zero, the NR25 constantly updates and stores the internal offset in a FRAM memory. This includes storing the internal offset when a "Preset Position:" command is issued. FRAM is a new memory technology that is non-volatile, but can be written to like RAM. (According to the manufacturer's specifications, writing to the memory every millisecond will burn out the memory in approximately 300 years.)

   If you decide to use the *Total Measurement Range* parameter only to make sure the internal offset is automatically stored in non-volatile memory, set this parameter to its maximum value so the NR25 remains an absolute sensor.

   ➤When the *Scaling Function Control* parameter is disabled, set the Total Measurement Range parameter to 65,536 multiplied by the number of turns the NR25 can encode. (65,536; 268,435,456; or 1,073,741,824.)
   ➤When the *Scaling Function Control* parameter is enabled, set the Total Measurement Range parameter to the value of the *Counts per Turn* parameter multiplied by the number of turns the NR25 can encode.

If you are using the *Total Measurement Range* parameter to control where the position returns to zero, set it to the appropriate value for your application.

> **NOTE** ▶ The *Total Measurement Range* parameter only affects how the internal position offset is stored. You must still issue a command to save the programmable parameters to non-volatile memory.

Explicit Commands

The NR25 can be completely controlled by explicit instruction commands from the host controller. Most of these commands are not needed in typical applications and the data for these commands is given in the two appendices. The most commonly used commands are explained in chapter 5, *Explicit Messaging Commands*, starting on page 39. These commands are:

- ➤ Program Parameters
- ➤ Preset Position
- ➤ Save to Flash
- ➤ Restore from Flash
- ➤ Return to Factory Defaults
- ➤ Reset Device

The *Return to Factory Defaults* command does not re-save the factory default parameter values to Flash memory. A separate *Save to Flash* command must be issued to return the NR25 to its factory default settings.

Factory Default Settings

The factory default settings for the NR25 are given in the table below.

Parameter	Setting	Description
Bus Address	63	Set by rotary switches on back of NR25.
Bus Speed	125 KHz	Set by DIP switches on back of NR25.
Bus Terminator	OFF	Set by DIP switch on back of NR25.
Count Direction	CW	Affected by "Return to Factory Defaults" command
Scaling Function Control	ON	Affected by "Return to Factory Defaults" command
Counts per Turn	65,536	Affected by "Return to Factory Defaults" command
Total Measurement Range	0	Affected by "Return to Factory Defaults" command
Velocity Format	pulses/second	Affected by "Return to Factory Defaults" command
Internal Position Offset	0	Affected by "Return to Factory Defaults" command
Device Type	0x22	CIP Encoder Profile Device

Table 1.3 Factory Default Settings

This chapter is intended for the engineer or technician responsible for installing and wiring the NR25 networked resolver encoder. Information in this chapter includes mechanical drawings, installation guidelines and connector pinout.

2.5" Servo Mount Outline Drawing

Aluminum Body, End Connect

( ) = Dimensions in millimeters
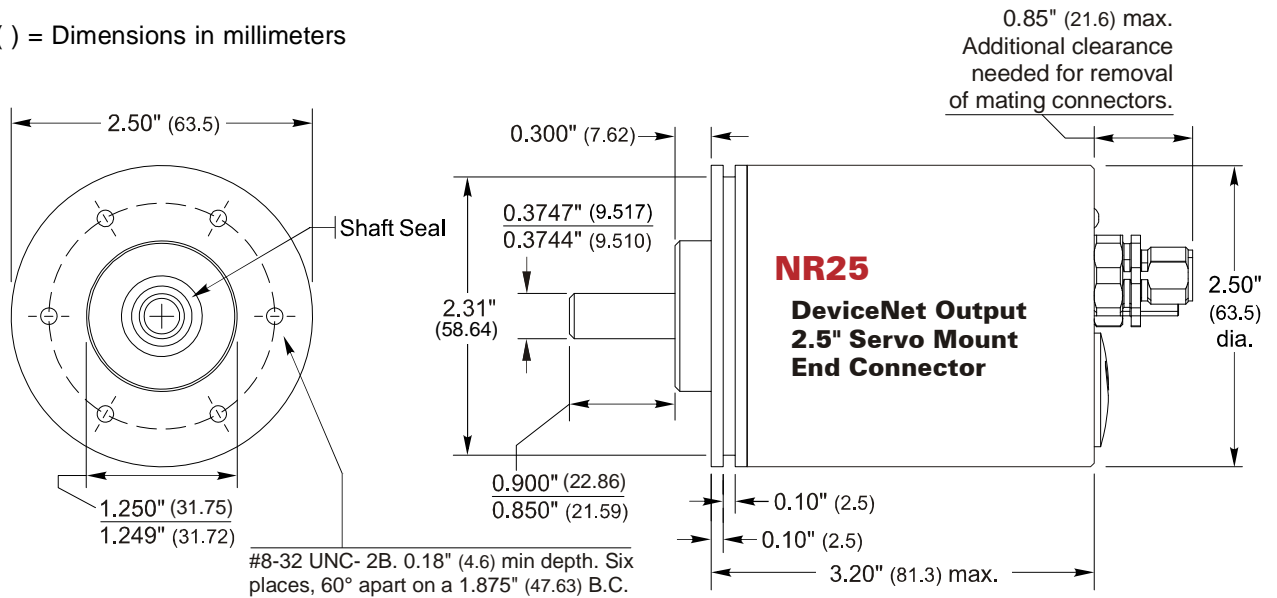
Figure 2.1 2.5" Servo Mount, End Connector Outline Drawing

Stainless Steel Body, End Connect

( ) = Dimensions in millimeters
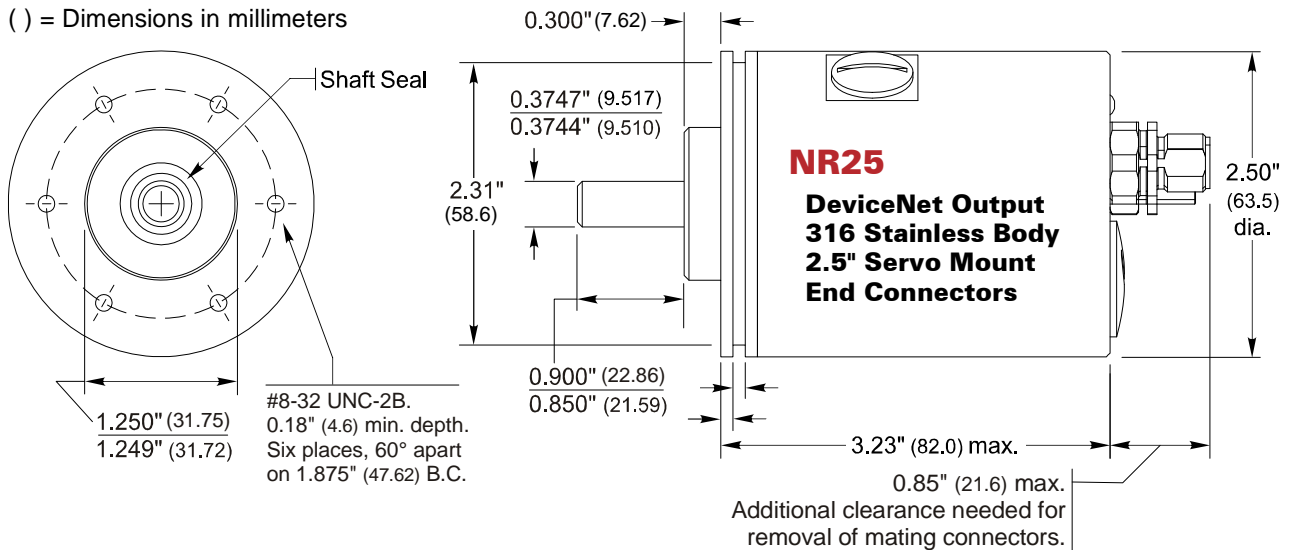
Figure 2.2 2.5" Servo Mount, End Connector Outline Drawing

2.5″ Servo Mount Outline Drawings (continued)

Shaft Details

3/8" Shaft (Shaft Option 1)	10mm Shaft (Shaft Option 2)	1/4" Shaft (Shaft Option 3)

0.3747" (9.517)
0.3744" (9.510)

0.3934" (9.992)
0.3931" (9.985)

0.2497" (6.342)
0.2492" (6.330)

0.900" (22.86)
0.850" (21.59)

0.900" (22.86)
0.850" (21.59)

0.900" (22.86)
0.850" (21.59)
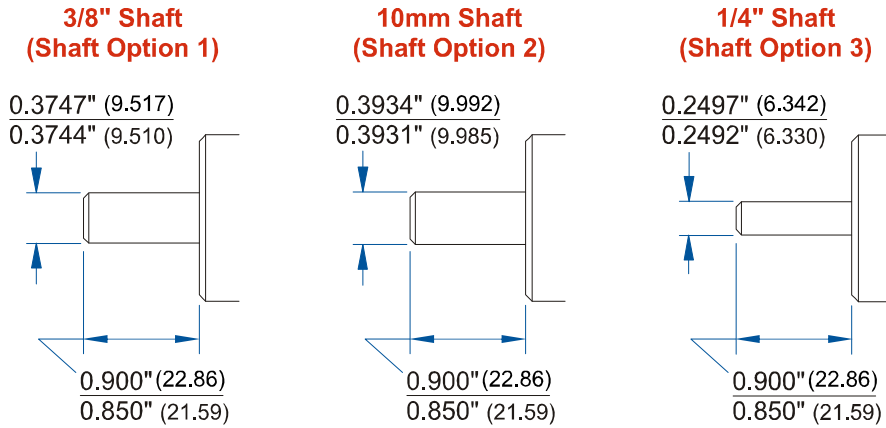
() = Dimensions in mm

Figure 2.3 2.5″ Servo Mount Shaft Details

Shaft Loading

Limit shaft loading to the following values. These values statistically yield an L10 life of $2X10^9$ revolutions. (Statistically, only 10% of the bearings will have failed after $2X10^9$ revolutions.) Shaft loading has an exponential effect on bearing life. The bearings will statistically last longer if you can limit shaft loading below the given values. Consider using the 5/8" shaft NR25 if your shaft loading is expected to be greater than the values given below. Outline drawings for the 5/8" shaft NR25 can be found starting on page 23.

Radial Load	Axial Load
40 lbs. (178 N)	20 lbs. (88 N)

Table 2.1 Servo Mount Shaft Loading

58 mm Servo Mount Outline Drawing

Aluminum Body, End Connect

( ) = Dimensions in millimeters

0.85" (21.6) max.
Additional clearance needed for removal of mating connectors.

2.50" (63.5)

Shaft Seal

0.3934" (9.993)
0.3931" (9.985)

2.283" (58.00)

2.028" (51.50)

5°

NR25
DeviceNet Output
58 mm Servo Mount
End Connector

2.50" (63.5) dia.

1.969" (50.00)
1.968" (49.97)

M4X0.7 0.315" (8.00) min depth. Three places, 120° apart on a 1.654" (42) B.C.

0.807" (20.50)
0.787" (20.00)

0.020" (0.50) deep x 0.77" (19.5) long flat

0.118" (3.00)
0.118" (3.00)
0.157" (4.00)

3.54" (89.9) max.

Figure 2.4 58 mm Servo Mount, End Connector Outline Drawing

Stainless Steel Body, End Connect

( ) = Dimensions in millimeters

0.85" (21.6) max.
Additional clearance needed for removal of mating connectors.

2.50" (63.5)

Shaft Seal

0.3934" (9.993)
0.3931" (9.985)

2.283" (58.00)

2.028" (51.50)

5°

NR25
DeviceNet Output
316 Stainless Body
58 mm Servo Mount
End Connectors

2.50" (63.5) dia.

1.969" (50.00)
1.968" (49.97)

M4X0.7 0.315" (8.00) min depth. Three places, 120° apart on a 1.654" (42) B.C.

0.807" (20.50)
0.787" (20.00)

0.020" (0.50) deep x 0.77" (19.5) long flat

0.118" (3.00)
0.118" (3.00)
0.157" (4.00)

3.54" (89.9) max.
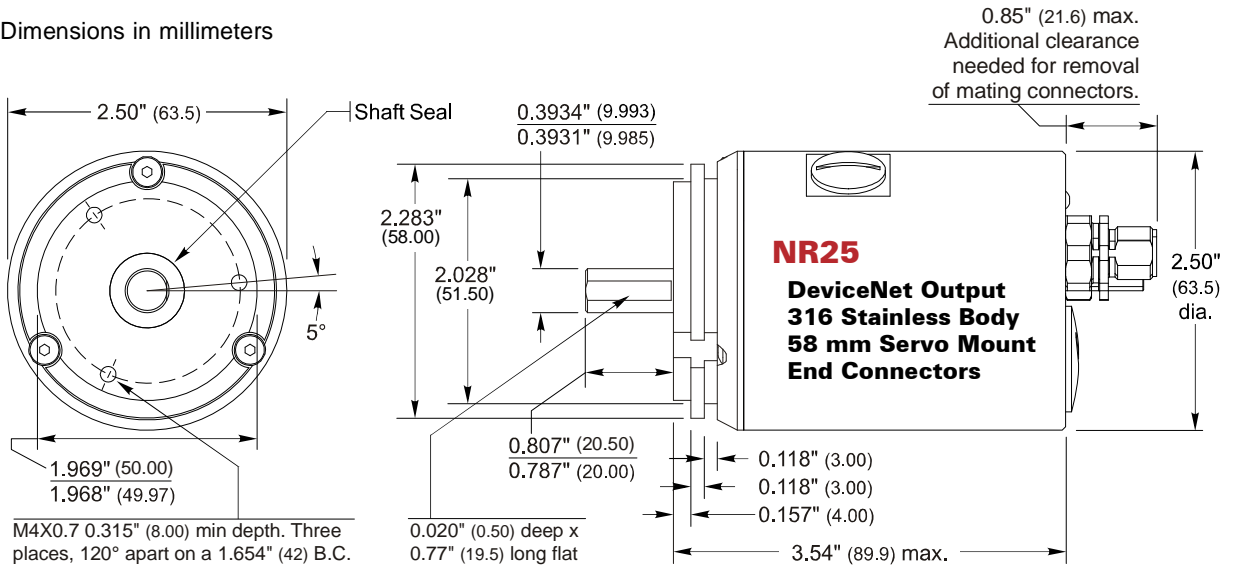
Figure 2.5 58 mm Servo Mount, End Connector Outline Drawing

58 mm Servo Mount Outline Drawings (continued)

Shaft Details

6 mm Shaft (Shaft Option 4)   **10 mm Shaft (Shaft Option 6)**

0.2359" (5.993)
0.2356" (5.985)

0.3934" (9.993)
0.3931" (9.985)

0.01" (0.3)   0.02" (0.5)

0.374" (9.50)   0.413" (10.50)   0.77" (19.5)   0.807" (20.50)
0.394" (10.00)   0.787" (20.00)
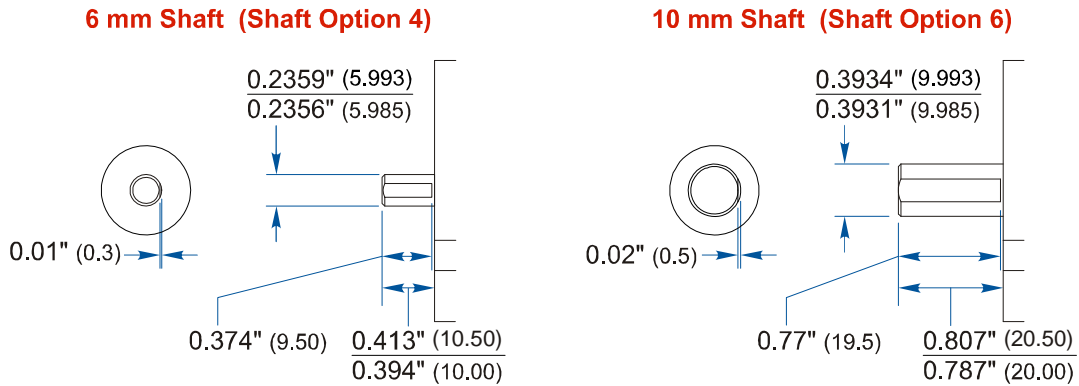
() = Dimensions in mm

Figure 2.6 58 mm Servo Mount Shaft Details

Shaft Loading

Limit shaft loading to the following values. These values statistically yield an L10 life of $2X10^9$ revolutions. (Statistically, only 10% of the bearings will have failed after $2X10^9$ revolutions.) Shaft loading has an exponential effect on bearing life. The bearings will statistically last longer if you can limit shaft loading below the given values. Consider using the 5/8" shaft NR25 if your shaft loading is expected to be greater than the values given below. Outline drawings for the 5/8" shaft NR25 can be found starting on page 23.

Radial Load	Axial Load
40 lbs. (178 N)	20 lbs. (88 N)

Table 2.2 Servo Mount Shaft Loading

Flange Mount Outline Drawings

Aluminum Body, End Connect

( ) = Dimensions in millimeters

0.85" (21.6) max.
Additional clearance
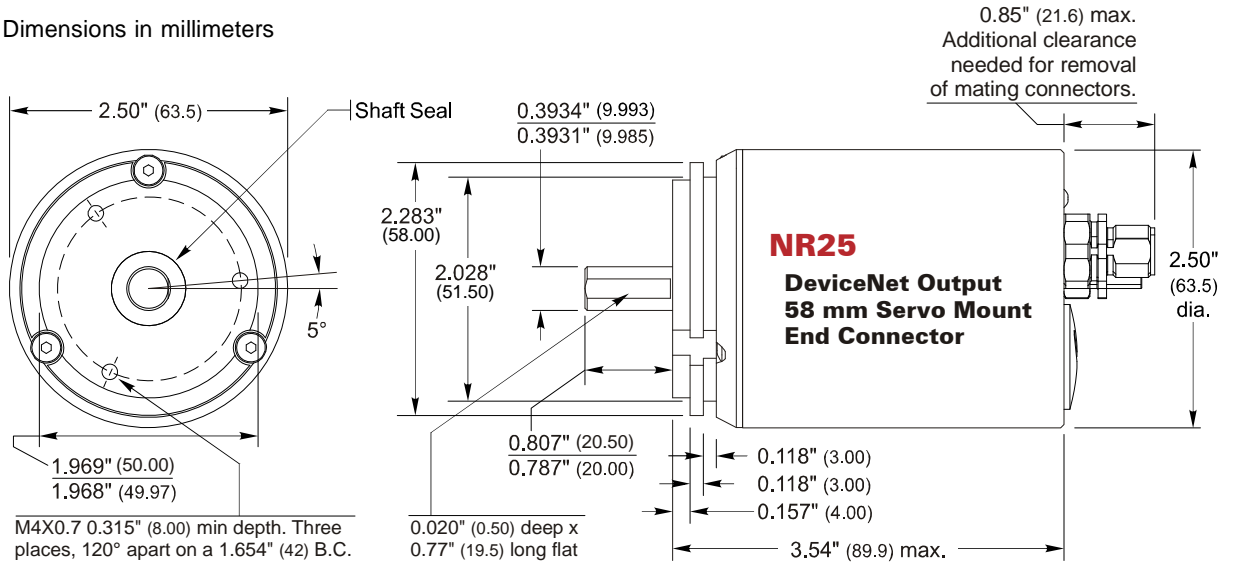needed for removal
of mating connectors.

1.032" (26.21) typ.

0.218" (5.54) dia.
Four places.

1.032" (26.21) typ.

0.300" (7.62)

0.250" (6.35)

0.3747" (9.517)
0.3744" (9.510)

NR25

**DeviceNet Output
Flange Mount
End Connector**

2.50" (63.5) dia.

2.65" (67.3)

Shaft Seal

1.250" (31.75)
1.249" (31.72)

0.900" (22.86)
0.850" (21.59)

3.15" (80) max.

2.65" (67.3)
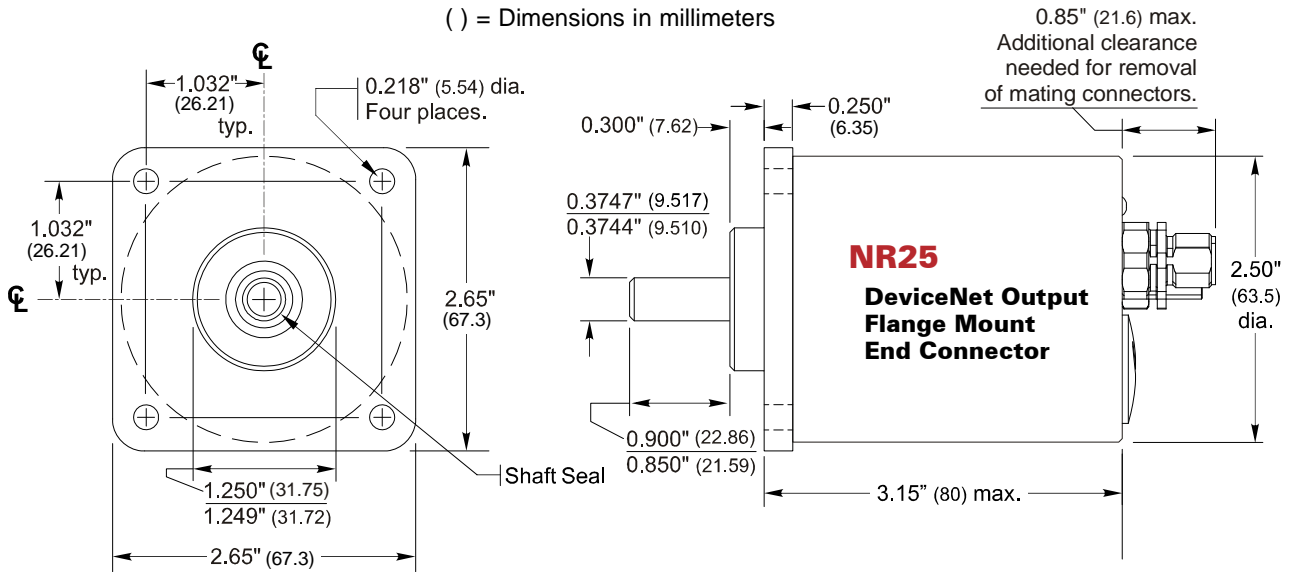
Figure 2.7 Flange Mount, End Connector Outline Drawing

Stainless Steel Body, End Connect

( ) = Dimensions in millimeters

1.032" (26.21) typ.

0.218" (5.54) dia.
Four places.

1.032" (26.21) typ.

0.300" (7.62)

0.3747" (9.517)
0.3744" (9.510)

NR25

**DeviceNet Output
316 Stainless Body
Flange Mount
End Connectors**

2.50" (63.5) dia.

2.65" (67.3)

Shaft Seal

1.250" (31.75)
1.249" (31.72)

0.900" (22.86)
0.850" (21.59)

0.250" (6.35)

3.18" (80.9) max.

2.65" (67.3)

0.85" (21.6) max.
Additional clearance needed for
removal of mating connectors.
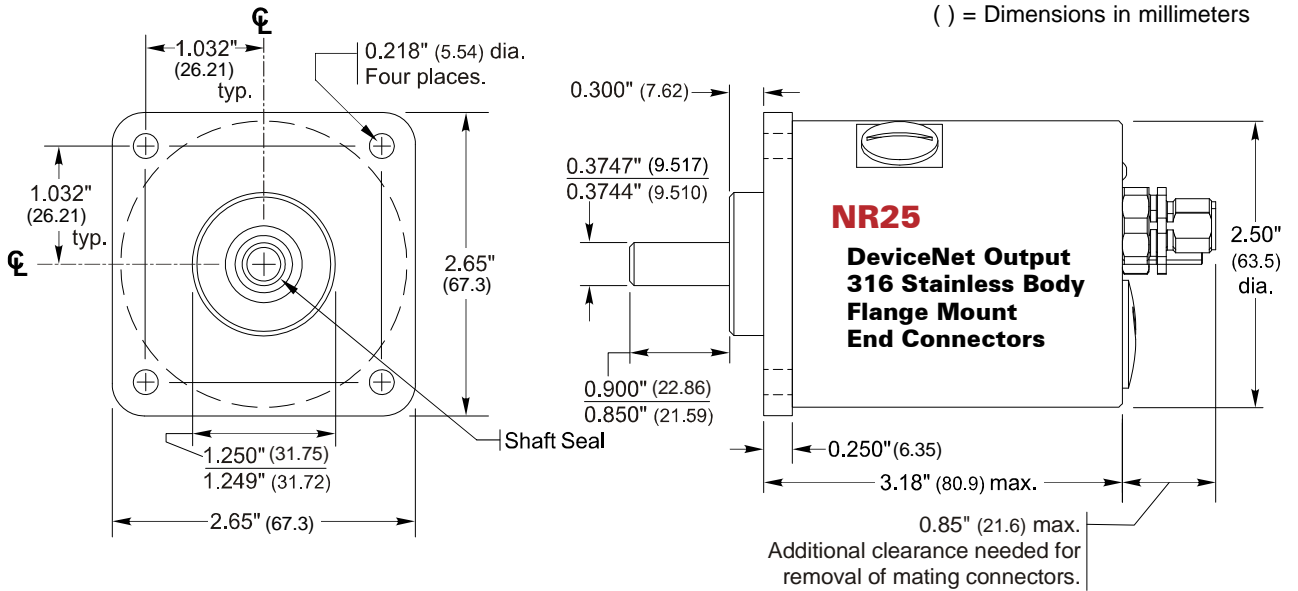
Figure 2.8 Flange Mount, End Connector Outline Drawing

Flange Mount Outline Drawings (continued)

Shaft Details

3/8" Shaft	10mm Shaft	1/4" Shaft
(Shaft Option 1)	(Shaft Option 2)	(Shaft Option 3)

0.3747" (9.517)
0.3744" (9.510)

0.3934" (9.992)
0.3931" (9.985)

0.2497" (6.342)
0.2492" (6.330)

0.900"(22.86)
0.850" (21.59)

0.900"(22.86)
0.850" (21.59)

0.900"(22.86)
0.850" (21.59)
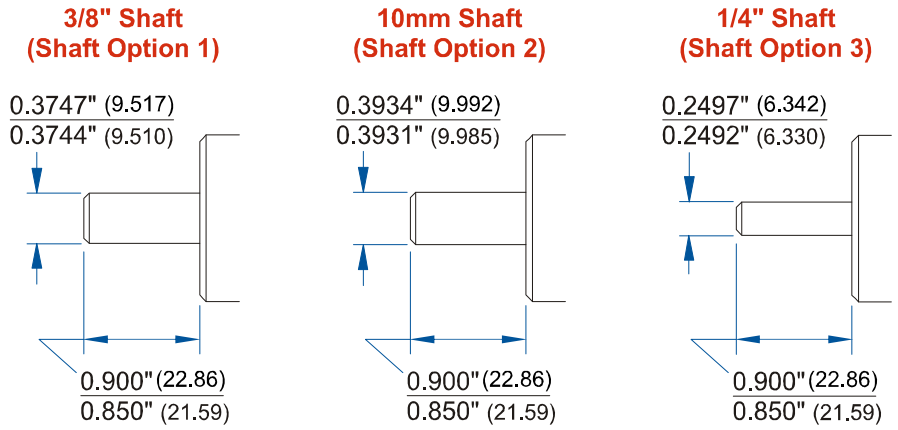
() = Dimensions in mm

Figure 2.9 Flange Mount Shaft Details

Shaft Loading

Limit shaft loading to the following values. These values statistically yield an L10 life of $2X10^9$ revolutions. (Statistically, only 10% of the bearings will have failed after $2X10^9$ revolutions.) Shaft loading has an exponential effect on bearing life. The bearings will statistically last longer if you can limit shaft loading below the given values. Consider using the 5/8" shaft NR25 if your shaft loading is expected to be greater than the values given below. Outline drawings for the 5/8" shaft NR25 can be found starting on page 23.

Radial Load	Axial Load
40 lbs. (178 N)	20 lbs. (88 N)

Table 2.3 Flange Mount Shaft Loading

Flange Mount Outline Drawings (continued)

5/8" Shaft, Aluminum Body, End Connect

2.65" (67.3) **sq.**

0.125" (3.18)

See Keyway Specifications

See Note 1

NR25

DeviceNet Output
5/8" Shaft
Flange Mount
End Connectors

1.032" (26.21) typ.

2.50" (63.5) **dia.**

1.032" (26.21) typ.

See Note 2

$\frac{0.6247" (15.867)}{0.6237" (15.842)}$

$\frac{1.500" (38.10)}{1.499" (38.07)}$ diameter

$\frac{1.45" (36.8)}{1.40" (35.6)}$

3.84" (97.5) **max.**

0.85" (21.6) max.
See Note 3

0.750" (19.05)

1.500" (38.10)

0.470" (11.95)

0.94" (23.9)

1.00" (25.4)

( ) = Dimensions in millimeters

NOTES:

1) **Integral Shaft Seal.**

2) **1/4-20 UNC-2B 0.50" (12.7) minimum depth. Six Places.**

3) **Additional clearance needed for removal of mating connectors.**

KEYWAY SPECIFICATIONS

KEYWAY

$\frac{0.1895" (4.813)}{0.1885" (4.788)}$ X $\frac{0.108" (2.74)}{0.106" (2.69)}$ **Deep**

KEY

$\frac{0.188" (4.78)}{0.187" (4.75)}$ **Sq. X 1.00" (25.4)**
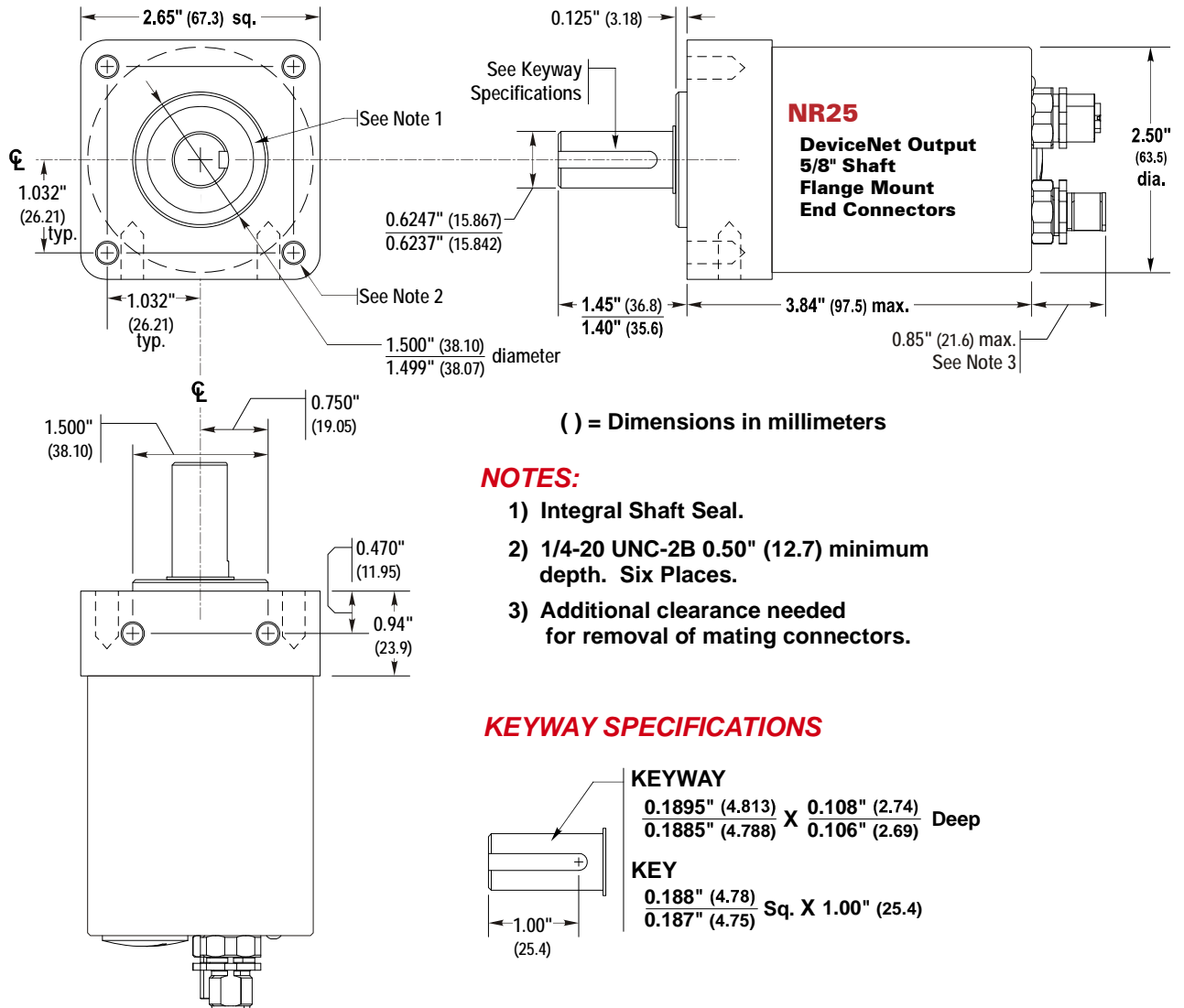
Figure 2.10 5/8" Shaft, Flange Mount, End Connect Outline Drawing

Flange Mount Outline Drawings (continued)

5/8" Shaft, Stainless Steel Body, End Connect

NR25
DeviceNet Output
316 Stainless Body
5/8" Shaft
End Connector

( ) = Dimensions in millimeters

NOTES:

1) **Integral Shaft Seal.**

2) **1/4-20 UNC-2B 0.50" (12.7) minimum depth. Six Places.**

3) **Additional clearance needed for removal of mating connectors.**

KEYWAY SPECIFICATIONS

KEYWAY
$\frac{0.1895" (4.813)}{0.1885" (4.788)}$ X $\frac{0.108" (2.74)}{0.106" (2.69)}$ **Deep**

KEY
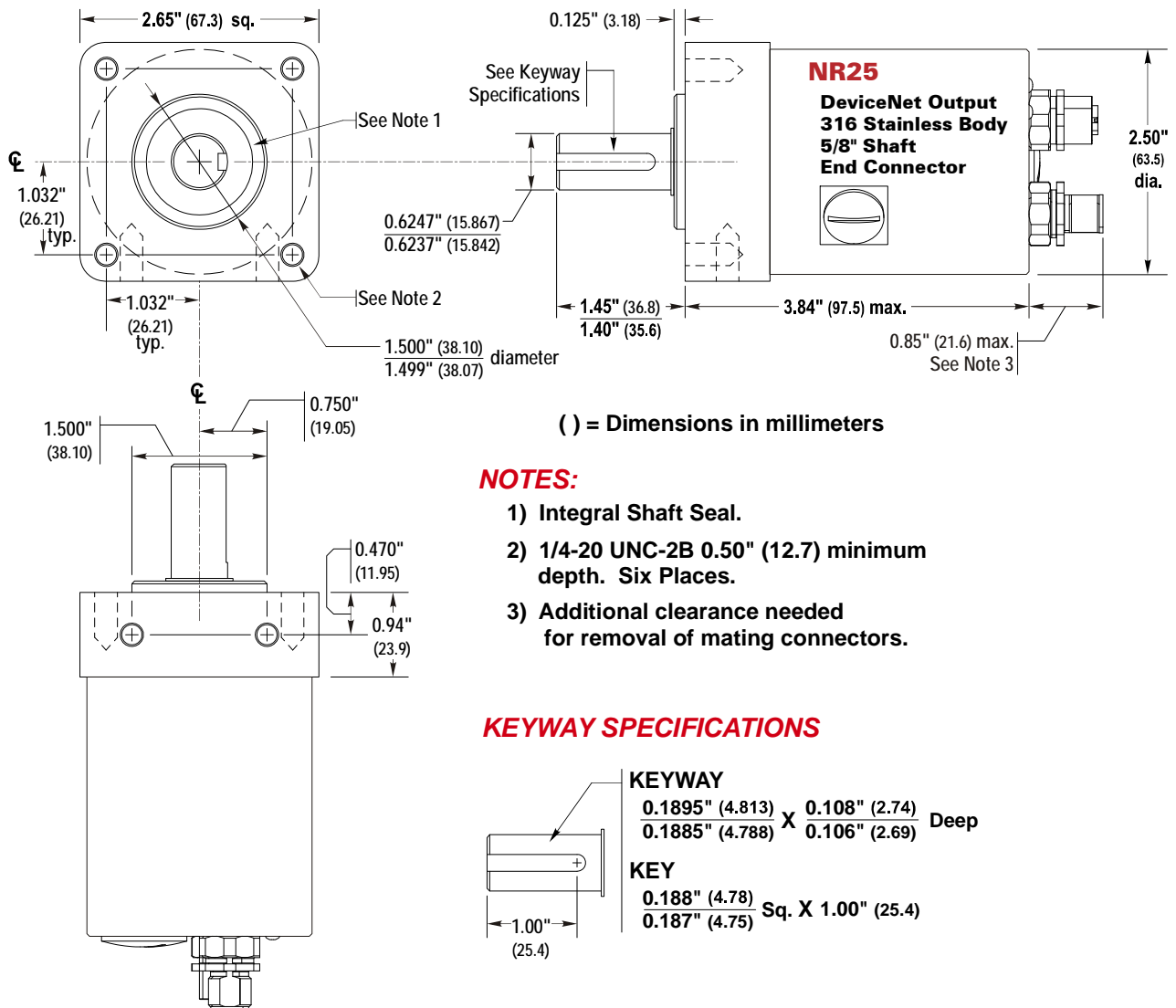$\frac{0.188" (4.78)}{0.187" (4.75)}$ **Sq. X 1.00" (25.4)**

Figure 2.11 5/8" Shaft, Flange Mount, End Connect Outline Drawing

Shaft Loading

Limit shaft loading to the following values. These values statistically yield an L10 life of $2X10^9$ revolutions. (Statistically, only 10% of the bearings will have failed after $2X10^9$ revolutions.) Shaft loading has an exponential effect on bearing life. The bearings will statistically last longer if you can limit shaft loading below the given values.

Radial Load	Axial Load
100 lbs. (445 N)	50 lbs. (222 N)

Table 2.4 Flange Mount, 5/8" Shaft Loading

Hub Shaft Mount Outline Drawings

Aluminum Body, End Connect

( ) = Dimensions in millimeters

VIEW A

0.85" (21.6) max.
Additional clearance needed for
removal of mating connector.

2.50" (63.5) Dia.

Shaft Seal

0.67" (17.0)

0.79" (20.0)

25°

1.125" (28.58)

0.067" (1.70) radius
on 2.48" (63.0) B.C.

2.72" (69.0)

NR25

**DeviceNet Output
Hub Shaft Mount
End Connectors**

2.50" (63.5) dia.

4.29" (109.0) max.

VIEW A

0.65" (16.5) Clearance

**Customer
Side**

SEE
CHART

Shaft Length
0.59" (15) min.
1.10" (27.9) max.

SEE DIA.
CHART

2.48" (63.0) B.C.

0.75" (19.0)

4-40 Thru Tap
3 places,
120° apart

0.14" (3.6)

1.10" (27.9) depth

DIAMETER CHART

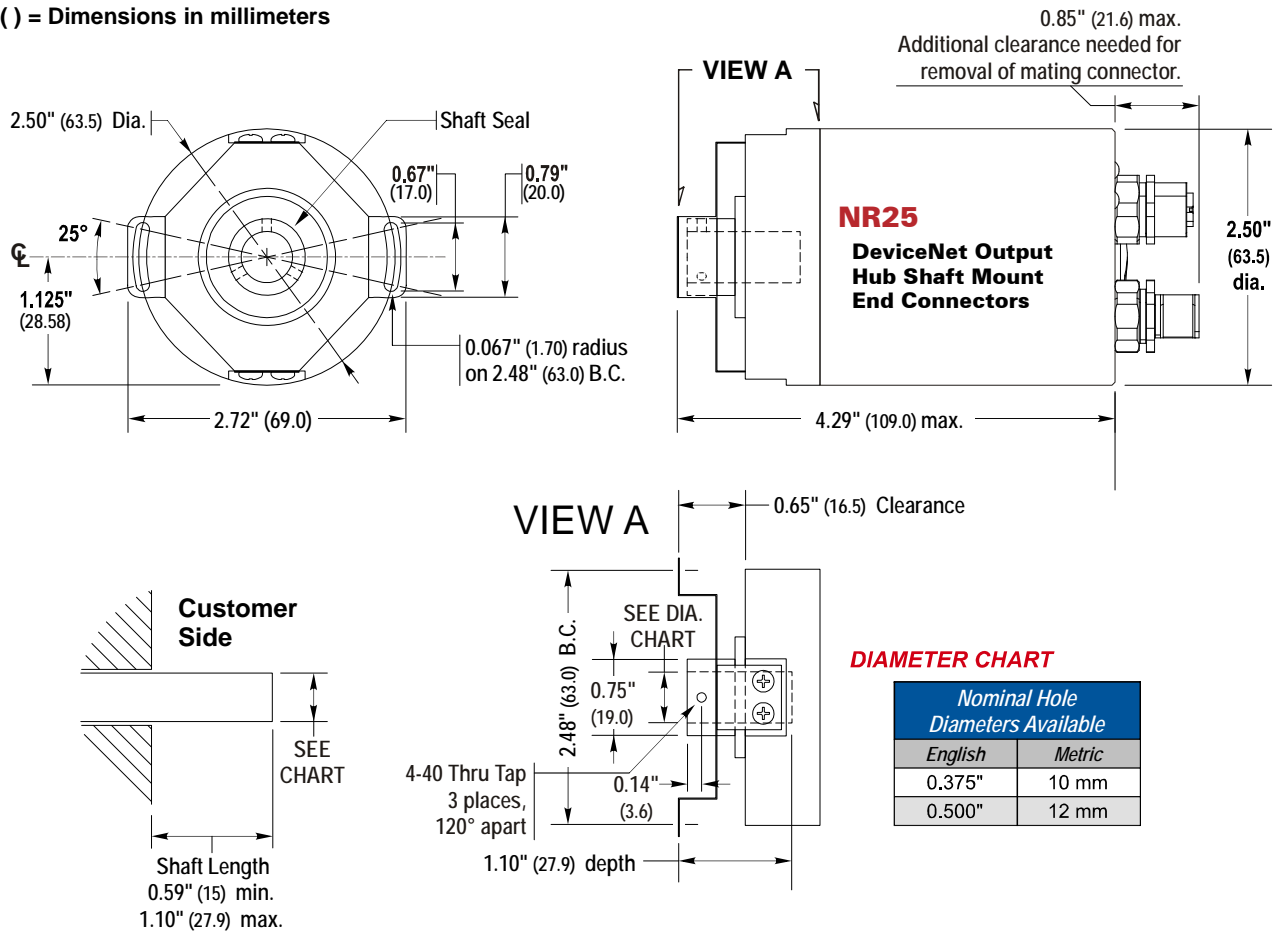| Nominal Hole Diameters Available	
English	*Metric*
0.375"	10 mm
0.500"	12 mm

Figure 2.12 Hub Shaft Mount, End Connect Outline Drawing

Hub Shaft Mount Outline Drawings (Continued)

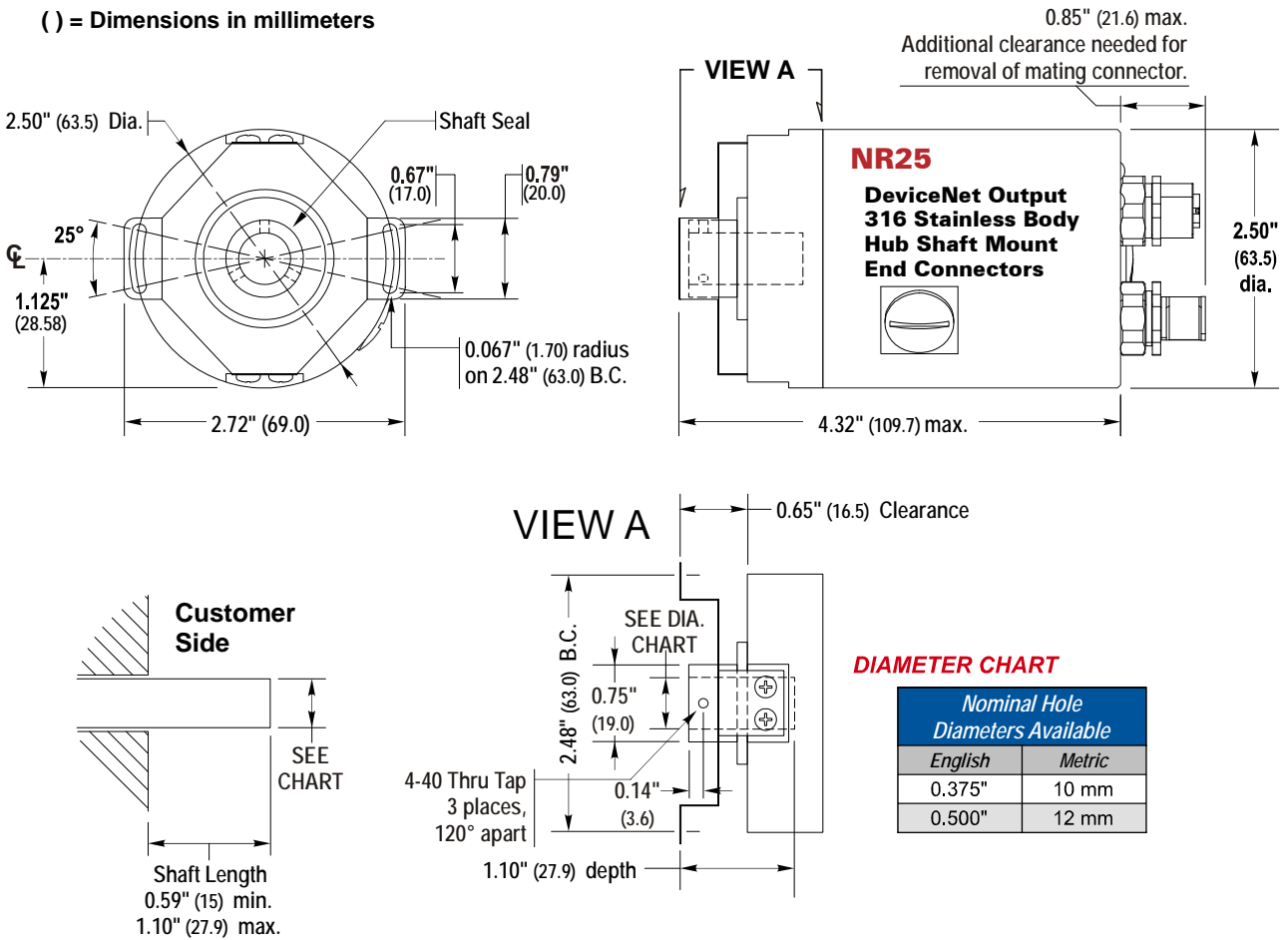### Stainless Steel Body, End Connect

( ) = Dimensions in millimeters

Figure 2.13 Hub Shaft Mount, End Connect Outline Drawing

DIAMETER CHART

Nominal Hole Diameters Available	
English	Metric
0.375"	10 mm
0.500"	12 mm

Available Shaft Diameters

The diameter of the drive shaft must be specified when ordering a blind shaft DuraCoder. Available options are given in the table below. Other diameter options may have become available after the release of this manual. Please check our website, www.amci.com, if you do not see the shaft diameter that fits your application.

Nominal Hole Diameters	
English	Metric
0.375"	10 mm
0.500"	12 mm

Table 2.5 Available Blind Shaft Diameters

Shaft Loading

The load that the NR25 presents *to* your input shaft, which is equal to the load presented to the NR25 *by* your input shaft, is difficult to calculate and depends on the accuracy of the mounting. The flexible metal mounting bracket will be able to absorb most of the radial loading forces, but accurate mounting of the NR25 is important.

Setting the DeviceNet Address and Bus Values

The large plug on the back of the NR25 covers the switches that set the address of the NR25 on the DeviceNet bus. Figure 2.14 below shows the rear cover of the NR25 with the plug removed.

> **NOTE** ▶ The state of the switches is only read on power up. You must cycle power to the NR25 before changes to the switch settings take effect.
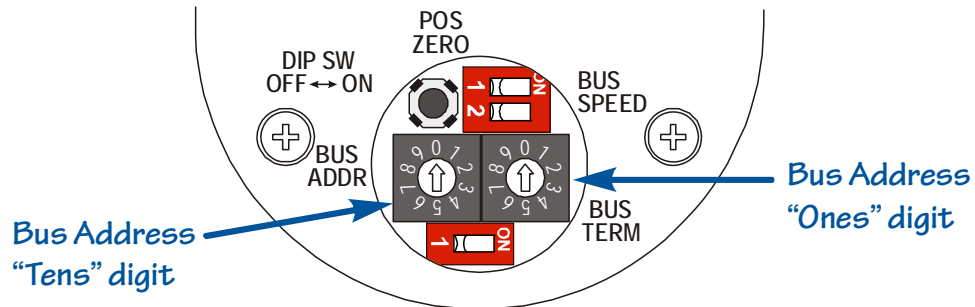

Figure 2.14 Address and Bus Value Switch Locations

DeviceNet Bus Address

The bus address is set with the two rotary switches. When the NR25 is in the orientation shown in figure 2.14, the switch for the "tens" digits on the left and the switch for the "ones" digit is on the right. The valid address range is 00 to 63. Note that no two devices on a DeviceNet network can share the same address.

> ➤ On power up, if the address has been changed to a new valid address, the NR25 will store this address in non-volatile memory. If the bus address switches are set later set to a value above 63, the NR25 will use the stored address as the address of the device.

> ➤ The NR25 for DeviceNet ships with an address of 63.

Bus Speed

Bus speed is set with the bank of two DIP switches. Table 2.6 below shows the settings for the three supported bus speeds. Note that the all devices on the DeviceNet network must have the same bus speed.

Switch Setting	Bus Speed
	125 KHz
	250 KHz
	500 KHz
	Last Stored in Flash

Table 2.6 Bus Speed Settings

> ➤ On power up, if the bus speed has been changed to a new valid value, the NR25 will store the new bus speed in non-volatile memory. If the bus speed switches are set later set to "Last Stored" setting, the NR25 will use the last value stored in Flash memory as the bus speed of the device.

Setting the DeviceNet Address and Bus Values (continued)

DeviceNet Terminators

The DeviceNet *trunk* must be terminated on both ends for proper operation. When using the NR25 you have two options.

➤ The NR25 has an internal terminating resistor that can be turned on or off with the Bus Term switch shown in figure 2.14.

➤ An external bus termination resistor can be used.

If the NR25 is the last device on the DeviceNet network, the bus must be terminated at the NR25.

➤ In large applications, the NR25 will often attach to the DeviceNet trunk through a T-Port tap and a termination resistor can be installed on the un-used connector of the tap. If this is done, the termination resistor must not be turned on inside the NR25. This configuration allows you to physically remove the NR25 from the network without disrupting communications between the other devices

➤ You also have the option to install a terminator on the Bus Out connector of the NR25. As with connectors and cordsets, there are several suppliers that manufacturer DeviceNet Terminators. One such supplier is Rockwell Automation. (RA part # 1485A-T1D5)

Replacing the Plug

Once the bus switches have been set, the plug must be re-installed and securely tighten in order to maintain the IP rating of the NR25. Use care when installing the plug so that the o-ring that seals against the back cover is not damaged.

Connector Pinouts

The diagram below shows the pinouts of the two micro connectors on the NR25 encoders for DeviceNet. These units have a part number that ends in "E02". It is possible to order NR25 units without the Bus Out connector by ordering a part number that ends in "E01". These units have a plug installed in place of the connector. Unused connectors must be capped for the unit to retain its IP67 rating.
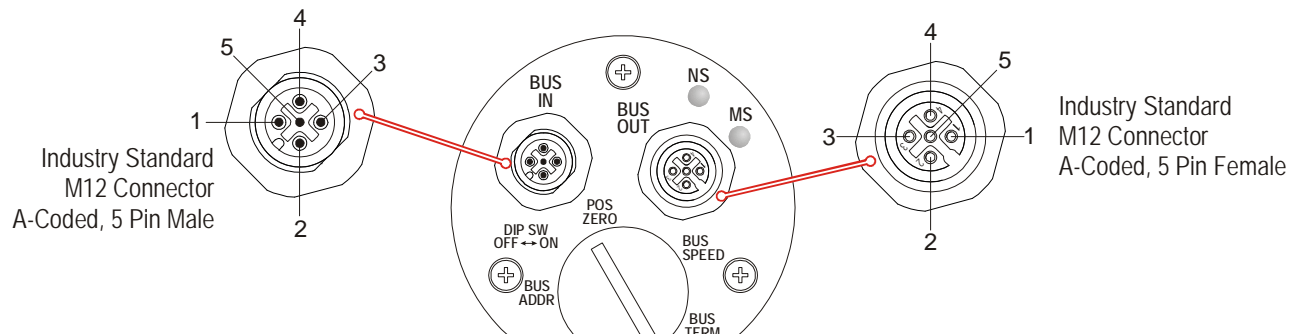

Figure 2.15 Connector Pinout

The following table lists the pinouts of the three connectors used on the NR25 encoders for DeviceNet.

		Pin Number				
		1	**2**	**3**	**4**	**5**
Connector	Bus In	See Note 1 (Shield Drain)	+ Voltage (Red)	– Voltage (Black)	CAN_H (White)	CAN_L (Blue)
	Bus Out	See Note 1 (Shield Drain)	+ Voltage (Red)	– Voltage (Black)	CAN_H (White)	CAN_L (Blue)

1) Pin 1 on the Bus In and Bus Out connectors are internally connected to each other and are connected to the body of the NR25 through a 2 nF capacitor.

Table 2.7   DeviceNet Connector Pinouts

Compatible Wiring

There are many different suppliers of DeviceNet compatible interconnections.   Installing a pre-assembled and tested cordset is the most common way to wire a DeviceNet system. For those companies wishing to make their own cables, cable and connectors are also available from a wide variety of suppliers. A small sampling of companies is given below.

Cordsets

➤ Phoenix Contact: *http://www.phoenixcontact.com*
➤ Turck Inc.: *http://www.turck.com* or *http://www.turck-usa.com*
➤ Lumberg Automation: *http://www.lumberg-automationusa.com*

Cables

➤ Belden Inc. *http://www.belden.com*
➤ Phoenix Contact: *http://www.phoenixcontact.com*

Connectors

AMCI uses Binder connectors on our NR25 encoders for DeviceNet. The table below lists the Binder part number of the straight and right angle mating connectors for the BUS IN and BUS OUT connectors.

NOTE These mating connectors are "same source" suggestions, which means that you are not required to use them. Any industry standard M12 A-coded 5 pin connector will mate with the NR25 BUS IN and BUS OUT connectors.

AMCI #	Binder #	Description
MS-31	99-0436-12-05	BUS IN mating connector, A-coded, female, straight. Screw terminal connections. 0.24" to 0.32" (6 to 8 mm) dia. cable. Plastic body. IP67 rated when properly installed.
	99-0436-52-05	BUS IN mating connector, A-coded, female, right-angle. Screw terminal connections. 0.24" to 0.32" (6 to 8 mm) dia. cable. Plastic body. IP67 rated when properly installed.
MS-32	99-0437-12-05	BUS OUT mating connector, A-coded, male, straight. Screw terminal connections. 0.24" to 0.32" (6 to 8 mm) dia. cable. Plastic body. IP67 rated when properly installed.
	99-0437-52-05	BUS OUT mating connector, A-coded, male, right-angle. Screw terminal connections. 0.24" to 0.32" (6 to 8 mm) dia. cable. Plastic body. IP67 rated when properly installed.

Table 2.8 Suggested Mating Connectors - BUS IN/OUT

Status LED's

As shown in figure 2.15 on page 28, the NR25 has two status LED's on the rear cover. These LED's are present on side connector units as well. The tables below list the various states of the LED's and their meaning.

Module Status LED

LED State	Definition
Off	No Power
Alternating Red/Green	Self-Test (Run on power up.)
Steady Green	Device Operational
Steady Red	Hardware Fault. (Cycle power. If fault persists, contact AMCI for support.)

Table 2.9 Module Status LED States

Network Status LED

LED State	Definition
Off	No Power
Alternating Red/Green	Power up Self-Test
Flashing Green	No established connections, unit in Idle state
Steady Green	Connection established
Flashing Red	Network Connection Timeout
Steady Red	Duplicate MAC address on network.

Table 2.10 Network Status LED States

Position Reset

The large plug on the back of the NR25 covers a momentary push button that can be used to reset the position data to zero. See figure 2.14, *Address and Bus Value Switch Locations* on page 27 for the location of the switch.

➤ The position is reset to zero on every press of the button and the resulting offset is stored in Flash memory.

➤ The position can also be preset over the network, but using the push button may be easier in some applications.

NOTE ➤ Once the position is reset to zero, the plug must be re-installed and securely tighten in order to maintain the IP rating of the NR25. Use care when installing the plug so that the o-ring that seals against the back cover is not damaged.

> **This chapter provides information on adding the NR25 to your DeviceNet network using the Rockwell Automation software RSNetWorx for DeviceNet and RSLinx.**

Communications Driver in RSLinx

If not previously installed, you must add a driver to RSLinx so that your computer can communicate on the DeviceNet network. The instructions to add a driver depends on the version of RSLinx you are running and the revision of the software. Refer to your software's literature for information on installing a DeviceNet driver.

➤ Literature for all Rockwell Automation products can be found at:
   http://literature.rockwellautomation.com

Installation of the EDS File

RSNetWorx require an Electronic Data Sheet, also known as an EDS file, before it can be used to configure the NR25.

➤ The EDS file for the NR25 can be found on the AMCI website at *http://www.amci.com/driverfiles.asp*
   The page shows links to files you can download. The file for the DeviceNet NR25 is a ZIP file that you will have to extract to a local folder. The ZIP file contains both the EDS and icon (ico) files.

Once RSNetWorx for DeviceNet is started, complete the following to install the EDS file:

1) Go to Tools - EDS Wizard and click on the [Next] button to start the process.

2) Follow the prompts until you reach the screen where you can browse for the eds file. Click on the [Browse...] button and navigate to the folder where you extracted the EDS and icon files to.

3) Follow the prompts until you reach the screen where you can select the device icon. Click on the device in the graphic tree and select the [Change Icon] button.

4) In the new screen, click on the [Browse...] button and navigate to the folder where you extracted the EDS and icon files to. Select the icon file.

5) Click through the rest of the screens until you click on the [Finish] button to complete the process.

Adding the NR25 to a Network

Once the EDS file is loaded, the NR25 can now be added to a DeviceNet network by searching the graphical tree. The NR25 will be found under Vendor +> Advanced Micro Controls Inc. (AMCI) +> Encoder. Add the NR25 by a double click on the icon or drag and drop into the network view.

Setting General Properties

Double clicking on the icon in the network view will bring up the property window where you can configure the NR25. Figure 3.1 shows the General Tab where you set the name, description, and address of the NR25. The address that you set must equal the address set by the rotary switches on the NR25.
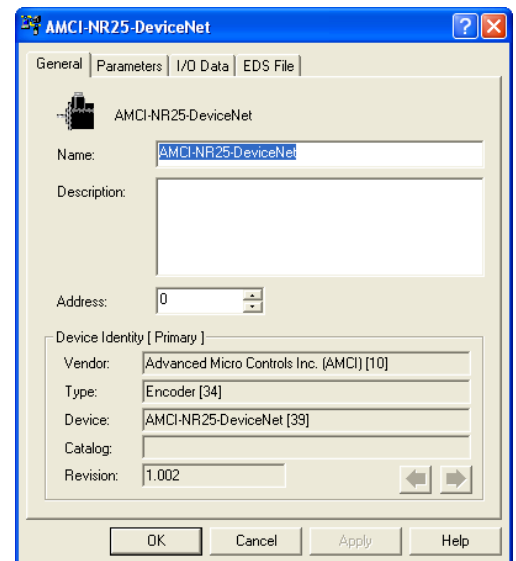

Figure 3.1 Properties Window

Adding the NR25 to a Network (continued)

Setting Parameters

Figure 3.2 shows the Parameter tab were you can set parameter values associated with the NR25. Doing this is optional. You can also accomplish this in ladder logic.

Parameter values are stored in the *.dnt file that you create when you save your project. If you are off-line for the initial network configuration, these parameters can be written to the NR25 when you go on-line.

When on-line, parameter values are written to the NR25 when you click on the *Apply* or *Download to Device* buttons.

NOTE ▶ Parameters that are downloaded to the NR25 are stored in RAM memory and will be lost when you cycle power. Parameter values can be stored to non-volatile Flash memory by using the Class Instance Editor as described below.
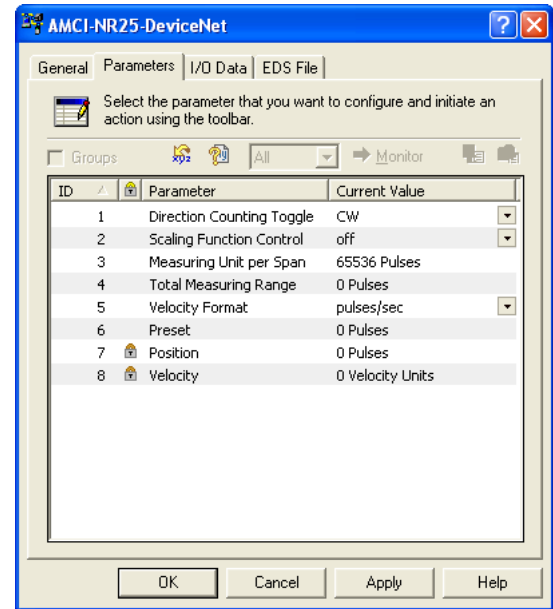
Remaining Tabs

The I/O Data tab displays the data that is available for each different data access methods. The actual data transfer is set when you configure the scanner module.

The EDS File tab shows the revision history of the file and allows you to view it in the Windows Notepad editor.

Figure 3.2 Parameter Tab

Saving Parameter Values

Parameter values can be stored in non-volatile Flash memory from within RSNetWorx by using the Class Instance Editor. You can also accomplish this in ladder logic instead. To open the Class Instance Editor:

➤ Go online with the network if necessary by clicking on *Network* in the menu bar and then selecting *Online*. The software will scan the network and then go online. This process may take several minutes.

➤ Click on the NR25 in the graph window.

➤ Click on "Device" in the menu bar and then select "Class Instance Editor".

The Class Instance Editor opens as is shown in figure 3.3.

1) In the Service Code section of the window, click on the *Description* drop down list, select the "Save" option. The *Value* field will change to "16".

2) In the Object Address section, set *Class:* to a value of "23" and *Instance:* to a value of "0". Also, make sure the *Send the Attribute ID* checkbox is unchecked.

3) Click on the [Execute] button. The NR25 will respond with the message "The execution was completed" in the *Receive Data* text box.
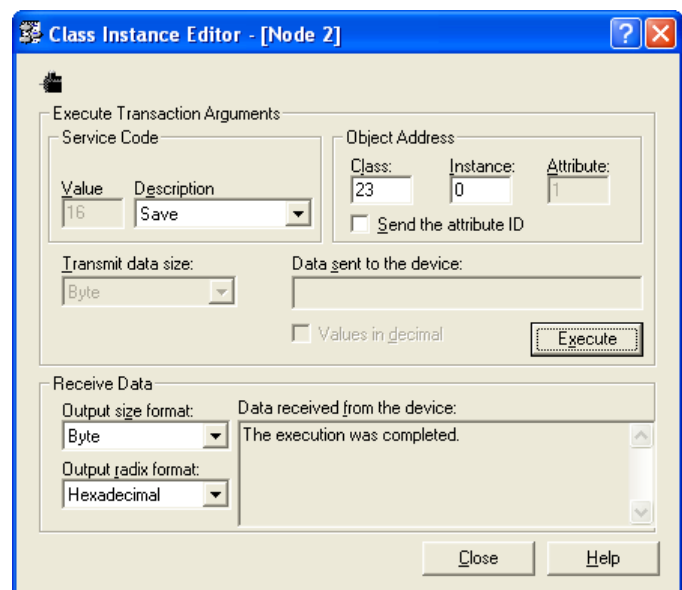

Figure 3.3 Class Instance Editor

Restoring Parameter Values

Changes to parameter values that have not been saved to Flash can be overwritten by restoring them to the values previously saved in Flash. This is done by using the Class Instance Editor. Open the Class Instance Editor as shown in the *Saving Parameter Values* section on the previous page.

With the Class Instance Editor open:

1) In the Service Code section of the window, click on the *Description* drop down list, select the "Restore" option. The *Value* field will change to "15".

1) In the Object Address section, set *Class:* to a value of "23" and *Instance:* to a value of "0". Also, make sure the *Send the Attribute ID* checkbox is unchecked.

2) Click on the [Execute] button. The NR25 will respond with the message "The execution was completed" in the Receive Data text box.

Resetting Parameter Values

Parameter values can be reset to their factory default values by using the Class Instance Editor. Open the Class Instance Editor as shown in the *Saving Parameter Values* section on the previous page.

With the Class Instance Editor open:

1) In the Service Code section of the window, click on the *Description* drop down list, select the "Reset" option. The *Value* field will change to "5".

1) In the Object Address section, set *Class:* to a value of "23" and *Instance:* to a value of "0". Also, make sure the *Send the Attribute ID* checkbox is unchecked.

2) Click on the [Execute] button. The NR25 will respond with the message "The execution was completed" in the Receive Data text box.

NOTE These values are not automatically saved in Flash after they have been reset with this command. If you want the parameters to be at their factory default values after cycling power, you must save these values to Flash as described in the *Saving Parameter Values* section on the previous page.

Reading Parameter Values

The present values of the parameters can be read by using the Class Instance Editor. The most common parameters are available under the Parameters tab of the Properties window as shown in the *Setting Parameters* section on page 32, and that method should be used when setting these values. The Class Instance Editor can be used to read the values that are not available in the Properties tab. To begin, open the Class Instance Editor as shown in the *Saving Parameter Values* section on the previous page.

With the Class Instance Editor open:

1) In the Service Code section of the window, click on the *Description* drop down list, select the "Get Single Attribute" option. The *Value* field will change to "E".

2) In the Object Address section, set *Class:* to a value of "23", the *Instance:* to a value of "1", and *Attribute:* to the hexadecimal value of the attribute you wish to read. A full list of attributes is available in the *Supported Instance Attributes* section of the manual, starting on page 46. For example, to read the Actual Sensor Reading, set the Attribute field to "70".

3) Click on the [Execute] button. The NR25 will respond with the raw, 16 bit position value of the resolver in the Receive Data text box. This value is displayed in hexadecimal, with the least significant byte first.

Adding the NR25 to a Scan List

The NR25 can be read with explicit messaging, but using the I/O messaging available with the DeviceNet scanner is far more efficient. I/O messaging is similar to implicit messaging available with RSLogix 5000 platforms in that the data transfer will occur automatically between the scanner and NR25 once it is configured. Data from the scanner will also be updated automatically in your PLC.

The DeviceNet scanner must be added to the network before it can be configured. Refer to the documentation that came with your scanner to add it to your network. The 1769-SND is used in this example.

Once the scanner is added to the network, double click on it to open its Properties window. The *Scanlist*, *Input*, and *Output* tabs are the most important when configuring the NR25.

Scanlist tab

Figure 3.4 shows the Scanlist tab before any devices are added to the scan list. To add the NR25, simply click on the device to select it and then click on the [>] button.

Once added you can:

➤ Change the *Electronic Key:* parameters to control which devices can be used to replace the NR25 in the network.

➤ Click on the [Edit I/O Parameters] button to change how the NR25 responds to the three different communication methods and what data is transferred.

Figure 3.5 below shows the layout of the Edit I/O Parameters window. Note that the Polled communications method is the most commonly used.

➤ By default, the Strobed communications method is disabled. If enabled, the NR25 will use eight bytes of input data and eight bytes of output data. The eight bytes of output data are used to signal every strobed device on the network simultaneously. A single bit will be assigned to the NR25 in the output data. Set this bit to force the NR25 to update its data.

➤ By default, the Polled communications method is enabled and the NR25 uses eight bytes of input data and five bytes of output data. The five bytes of output data is used the hold the Preset Value and commands.

➤ By default the Cyclic communications method is disabled. When Polled is enabled along with Cyclic communications, the Output size value is disabled and defaults to zero. If you choose to disable the Polled connection, the Output Size field enables and you must set it to a value of "5".

NOTE ⏵ The NR25 does not support the *Change of State* communications method.

Each communication type, Polled, Strobed, and Cyclic, provide their own position and velocity data. For example, of you have all three selected, three sets of position and velocity data will be available in the input registers. See for information on mapping this data to the input registers.
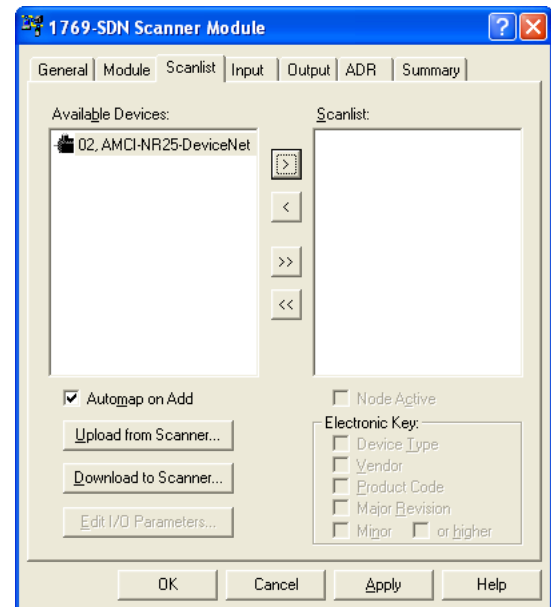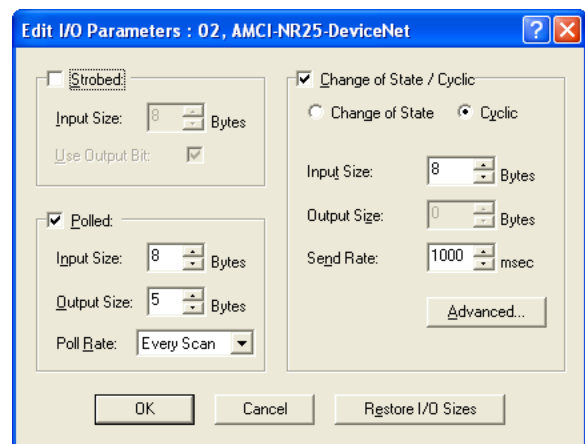

Figure 3.4 Scanlist Tab

Figure 3.5 Edit I/O Parameters

Adding the NR25 to a Scan List (continued)

Input Tab

The Input tab is used to select where the NR25 data appears in the scanner's data table. "AutoMap" is appropriate for most applications, but you also have the option of clicking on the [Advanced...] button and manually mapping the data as you see fit. When using Advanced Mapping, you will need to create a separate mapping for each data word transferred by the NR25. Position data is always thirty two bits long and starts at word 0. The second data word, (velocity data by default) is also thirty two bits long and starts at word 3.

Figure 3.6 shows the NR25 that has been mapped to the scanner using the [AutoMap] button. Position data will appear in the first thirty two bits and the second thirty two bits will contain velocity data by default.
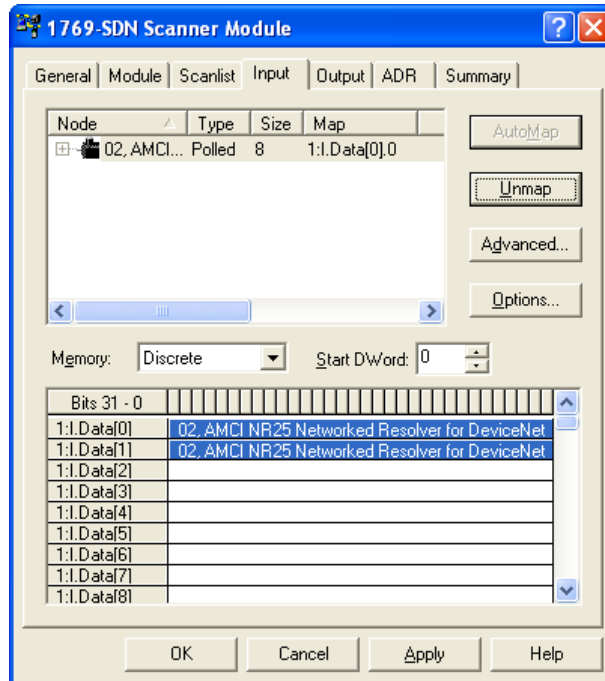
Figure 3.6 NR25 Scan List Automap

Adding the NR25 to a Scan List (continued)

Input Tab (continued)

Figure 3.7 shows an Advanced mapping where the NR25 position data (offset 0:0) is mapped to Input word 0 (offset 0:0) and velocity data (offset (4:0) is mapped to Input word 3 (offset 3:0).

You can also choose not to map data from the NR25. For example, you can choose to map only position data and not velocity data by mapping only the first thirty-two bits of the data transfer from the NR25.

NOTE ▶ Mapping this way will not decrease the amount of data transmitted over the DeviceNet network. It only decreases the amount of data trans-ferred from the scanner to your PLC. The NR25 always transmits eight bytes of input data.

Output Tab

The Output tab is used to map where the output data for the NR25 appears in the scanner's data table.

You can choose not to map data to your NR25. If you do this, you will have to use explicit messaging to send commands to the NR25. Depending on the application, this may actually simplify your ladder logic but at a cost of less efficient data transfer over the DeviceNet network.

Saving the Scan List

Once the NR25 is added to your scanner, the next step is to save the network configuration.

In RSNetWorx for DeviceNet, click on File -> Save.

Navigate to a directory that will be available to your RSLogix software, type in a meaningful name for your network and save the file. A file with a ".dnt" extension will be created.

Downloading the Scan List to the Scanner

Once the scan list is configured correctly, you must download it to the scanner.

1) Go online with the scanner. Click on *Network* in the menu bar and then select *Online*. The software will scan the network and then go online. This process may take several minutes.

2) Once online, click on *Network* in the menu bar and then select *Download to Network*. The software will download the new scan list to your scanner.
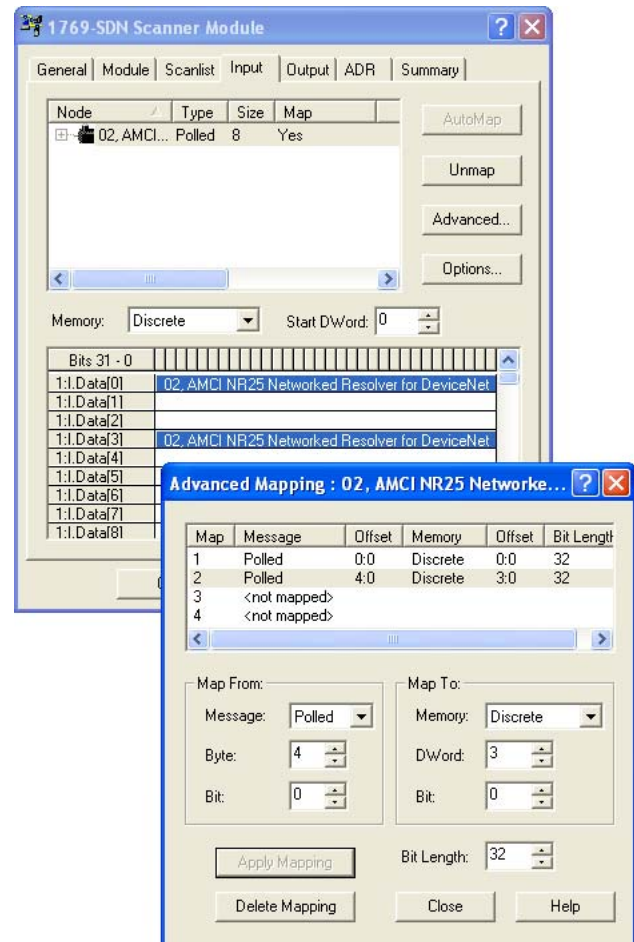
Figure 3.7 Advanced Mapping

I/O MESSAGING COMMANDS

> The five output bytes assigned to the NR25 can be used to preset the position value and save parameters to Flash memory through I/O Messaging. I/O Messaging also allows you to read back the parameter values the NR25 is presently using. I/O Messaging uses DeviceNet bandwidth very efficiently, but at the cost of added complexity to your ladder logic when presetting the position or saving parameters to Flash memory.

I/O Messaging Data Format

NOTE ▷ Input and Output data must be mapped to the scan list of your DeviceNet scanner before you can use I/O Messaging commands to control the NR25. If you are using the Polled communications method, it is strongly advised to poll the NR25 every scan to simplify handshaking. See *Adding the NR25 to a Scan List* starting on page 34 for more information.

Input Data

A total of eight bytes are read from the NR25 when using Polled, Cyclic, or Strobed transfers. The first four bytes are always the position data as a double integer (DINT) value. The remaining four bytes are also a double integer value that defaults to transmitting the velocity data. Parameter values and other data can be transmitted in place of the velocity data by changing the value of the command byte in the output bytes assigned to the NR25.

Output Data

A total of five bytes are written to the NR25 when using Polled or Cyclic transfers. The NR25 considers the first four bytes to be a double integer (DINT) value that stores the Preset Value. The fifth byte is the Command byte. Changing the value of this byte controls the operation of the NR25.

Reading NR25 Parameter Values

Set the command byte to any of the following values to control what data is transferred by the NR25.

Command Byte	Input Word 1 (bytes 0-3)	Input Word 2 (bytes 4-7)
0 $_{hex}$	Position	Velocity data
10 $_{hex}$	Position	Measuring Units per Span parameter value
11 $_{hex}$	Position	Total Measurement Range parameter value (Zero if not used)
13 $_{hex}$	Position	Preset Value parameter value
33 $_{hex}$	Position	Calculated Internal Position Offset
68 $_{hex}$	Position	Time Stamp (100 nanosecond resolution)
69 $_{hex}$	Position	Actual Position (no scaling or offset applied)

Table 4.1 Command Byte Values

NOTE ▷ The NR25 will return velocity data if the Command byte is set to any value not listed above. An error will not be generated.

Presetting the Position Value

Presetting the value involves the following sequence:

1) Write the Preset Value as a double integer to the first four bytes of the output data assigned to the NR25.

2) Set the command byte to a value of "2".

3) Hold this value until you are sure the scanner module has transmitted the command byte to the NR25.

4) Set the command byte to a value of "13" (D_{hex}).

The NR25 will respond by setting the position value to the Preset Value. At this point, the NR25 will transfer position and velocity data over the DeviceNet network.

A sample program is available on the AMCI website for the NR25 for DeviceNet. The sample program contains a routine for presetting the position value using I/O messaging. The sample program can be found on the *http://www.amci.com/sampleprograms.asp* website page.

Saving Offset and Parameters to Flash

The offset that results from a Preset command is stored in RAM memory and is lost when power is cycled to the NR25. The same is true for parameter values. This is acceptable for applications that reset the position value on every power up or decide to configure the NR25 on every power up using explicit messaging.

If your application requires absolute position you can issue a save to Flash memory command using the following sequence with I/O Messaging.

1) Set the command byte to a value of 20_{hex}.

2) Hold this value until you are sure the scanner module has transmitted the command byte to the NR25.

3) Set the command byte to a value of $D0_{hex}$.

The NR25 will respond by saving the internal position offset and all parameter values to Flash memory. At this point, the NR25 will transfer position and velocity data over the DeviceNet network. If you were transferring some data other than velocity, you must reset the command word as shown table 4.1 on page 37

The Flash memory of the NR25 is guaranteed for 10,000 write cycles. A non-recoverable memory error can any time once this value is exceeded. If you need to preset the position often and require it to remain after a power cycle, you have two options:

1) Set the Total Measurement Range parameter to a non-zero value, such as its maximum value for your NR25's configuration. If this is done, the NR25 will automatically store the position offset into FRAM memory instead of Flash. FRAM memory is non-volatile so the position will remain absolute over a power cycle. Additional information, including information on the maximum value of the Total Measurement Range, can be found in the *Non-Volatile Memory* section of the manual starting on page 15.

2) Calculate, store, and apply a position offset that is stored in the PLC.

A sample program is available on the AMCI website for the NR25 for DeviceNet. The sample program contains a routine for saving parameter to Flash using I/O messaging. The sample program can be found on the *http://www.amci.com/sampleprograms.asp* website page.

Explicit Messaging allows you to accomplish everything in ladder logic that you can do in RSNetWorx such as restoring parameter values from Flash. Explicit Messaging also simplifies the tasks of presetting the position value and saving parameter values to Flash memory. Finally, Explicit Messaging allows you to set parameter values from within your RSLogix program by using a custom assembly instance.

Explicit messaging uses message instructions that are available on platforms supported by the RSLogix 500 and RSLogix 5000 software packages. The instructions appear differently on the two platforms, but the data that must be entered to configure them remains the same.

NOTE ▶ Not all processors supported by RSLogix 500 have message instructions that support CIP Messages. Check your processor's specifications to verify their support. A MicroLogix 1500-LRP Rev. C was used when generating the screen captures for the RSLogix 500 sample.

RSLogix 5000 Sample

Figure 5.1 shows the two configuration screens when setting up a message instruction to communicate with a NR25 on a DeviceNet network.

1) **Message Type:** Always "CIP Generic".

2) **Service Type:** Depends on what you are accomplishing with the instruction. Setting the Service Type to *Custom* will allow you to set the **Service Code** field. Otherwise, the Service Code will automatically update when you change the Service Type.

3) **Class:** Always 23_{hex} if you have to set it yourself. Other classes are used, but they will be automatically set when you select the appropriate Service Type.

4) **Instance:** Usually "1". Set to "0" when issuing a Reset, Save to Flash, or Restore from Flash command.

5) **Attribute:** Depends on what you are accomplishing with the instruction.

6) **Source Element:** The data source if you are writing data down to the NR25.

7) **Source Length:** The length of the data you are writing down to the NR25

8) **Path:** Data path to the NR25. The path contains the Scanner Name, Scanner Port number, and the NR25 network address, separated by commas. For example "DNet_Scanner, 2, 2" is a valid path.
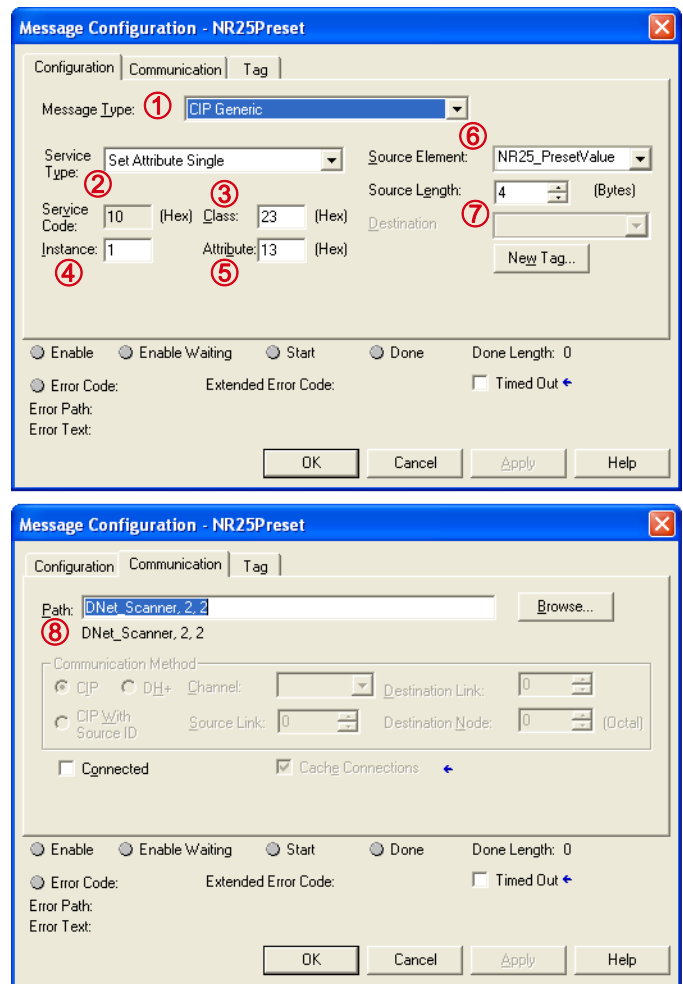

Figure 5.1 RSLogix 5000 MSG Example

The [Tag] tab lists the tag used to control the message instruction. This tag must have a "MESSAGE" data type.

RSLogix 500 Sample

Figure 5.2 shows the configuration screen when setting up a message instruction to communicate with a NR25 on a DeviceNet network. Note that the screen will change as you enter data.
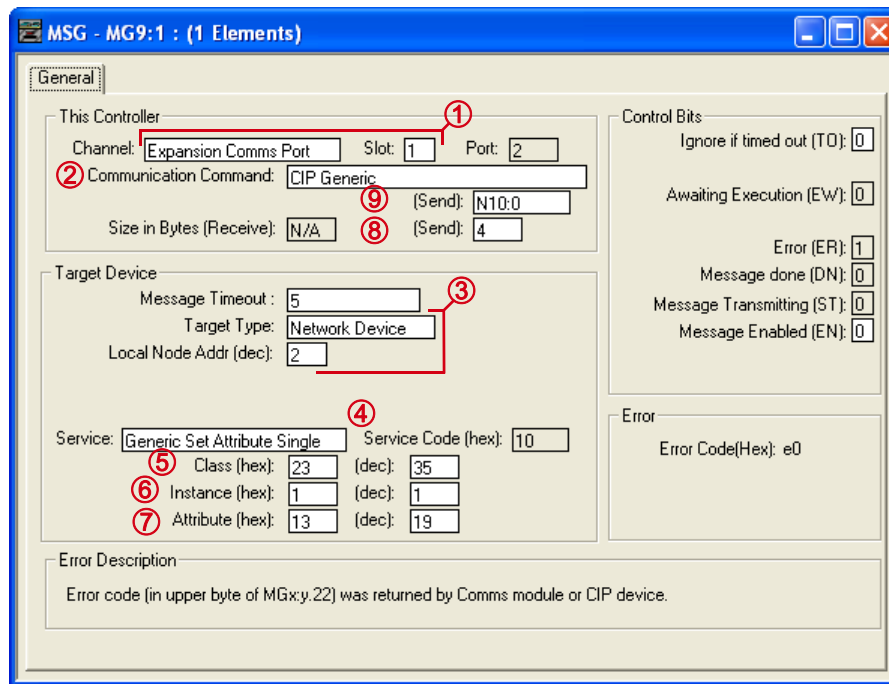
Figure 5.2 RSLogix 500 MSG Example

1) **Communications Channel:** Data path out of the controller. First, set the channel to "Expansion Comms Port" and press [Enter]. Next, set the slot number to the slot of the scanner module.

2) **Communications Command:** Always set to "CIP Generic".

3) **Target Type and Address:** Always set the Target Type to "Network Device" and set the Local Node Address to the MAC address of the NR25.

4) **Service Type:** Depends on what you are accomplishing with the instruction. Setting the Service Type to *Custom* will allow you to set the **Service Code** field. Otherwise, the Service Code will automatically update when you change the Service Type.

5) **Class:** Always 23_{hex} if you have to set it yourself. Other classes are used, but they will be automatically set when you select the appropriate Service Type.

6) **Instance:** Usually "1". Set to "0" when issuing a Reset, Save to Flash, or Restore from Flash command.

7) **Attribute:** Depends on what you are accomplishing with the instruction.

8) **Source Length:** The length of the data you are writing down to the NR25. This length is always in bytes.

9) **Send Address:** The starting address in memory that will be the source of the data you are writing to the NR25.

Program Parameters Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Set_Attribute_Single	10(hex)†	4(hex)	67(hex)	3(hex)	12 bytes

† Fixed when service set to "Set_Attribute_Single".

Table 5.1 Explicit Message: Program Parameters Instruction

NOTE ▶ This command writes parameter values to RAM and will be lost once power is cycled to the NR25. If you wish to use these parameters after cycling power, you must issue a *Save To Flash Instruction* as described on page 42.

Program Parameters Data Format

Twelve bytes of data must be written to the NR25 as part of this instruction. Table 5.2 lists the format of the data words. The location of these words must be entered into the message instruction. For RSLogix 5000 programs, this location is called "Source Element". For RSLogix 500 programs, this location is called "Send Address".

NOTE ▶ All data must be present and valid when programming the NR25.

Byte #	Word #	Parameter	Description	
0	1	Direction Counting Toggle	"0" = Clockwise increasing counts looking at shaft. "1" = Counter-Clockwise increasing counts looking at shaft.	
1		Scaling Function Control	"0" = Disable Scaling Function. The full resolution of 65,536 counts per turn is used for the Measuring Units per Span. "1" = Enable Scaling Function. The number of counts per turn is set by the Measuring Units of Span parameter below.	
2	2	Counts per Turn	Sets the number of counts generated over a single turn if the Scaling Function Control parameter equals "1". Always sets the number of pulses per turn reported in velocity data. This value ranges from 1 to 65,536. A value of 39,370 (16#99CA) is shown to the right.	CA
3				99
4	3			00
5				00
6	4	Total Measurement Range	Sets the number of counts before returning to zero. This value is used regardless of the state of the Scaling Function Control parameter. Parameter ranges: ➤ **Single Turn NR25**: Range of 0, 2 to 65,536 ➤ **28 bit Multi-turn NR25:** Range of 0, 2 to 268,435,456 ➤ **30 bit Multi-turn NR25:** Range of 0, 2 to 1,073,741,824 A value of 648,000 (16#0009 E340) is shown to the right.	40
7				E3
8	5			09
9				00
10	6	Velocity Format	Format of the velocity data. Byte 11 must always equal "1F". Byte 10 = "04" for pulses/second, "05" for pulses/millisecond, "07" for pulses/minute or "0F" for revolutions/minute. A value of "1F04" to the right would set the unit of measure to pulses/second.	04
11				1F

Table 5.2 Program Parameters Data Format

⚠ CAUTION When using the Total Measurement Range parameter, it is important to save your parameter values to Flash memory while commissioning your machine. The position value will change when power to the NR25 is cycled if this parameter is not saved to Flash. This step is critically important if implicit messaging is not used to communicate with the NR25. See the *Save To Flash Instruction* on page 42 for additional information.

Preset Position Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Set_Attribute_Single	10(hex)†	23(hex)	1(hex)	13(hex)	4 bytes

† Fixed when service set to "Set_Attribute_Single".

Table 5.3 Explicit Message: Preset Position Instruction

NOTE ▶ This command alters the internal position offset and stores this value in RAM. This offset will be lost once power is cycled to the NR25. If you wish to save the internal position offset, you must issue a *Save To Flash Instruction* as described below.

Preset Position Data Format

Four bytes of data must be written to the NR25 as part of this instruction. Table 5.4 lists the format of the data words. The location of these words must be entered into the message instruction. For RSLogix 5000 programs, this location is called "Source Element". For RSLogix 500 programs, this location is called "Send Address".

Byte #	Word #	Description	
0	1	**Preset Value.** The value that you want the position to become when you issue this command. The Preset Value can be any number between 0 and the configured full scale count of the encoder. The values on the right show the register values in hexadecimal if the Preset Value is 704,303 (16# 000A BF2F)	2F
1			BF
2	2	See *Preset Value* on page 14 for the acceptable ranges of parameter values for the different NR25 models.	0A
3			00

Table 5.4 Preset Position Data Format

Save To Flash Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Custom	16(hex)	23(hex)	0(hex)	0(hex)	0

Table 5.5 Explicit Message: Save to Flash Instruction

This instruction saves parameter values and the internal position offset to Flash memory. Additional data is not sent with this instruction. Set data length fields to 0 and leave the source field blank.

Restore From Flash Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Custom	15(hex)	23(hex)	0(hex)	0(hex)	0

Table 5.6 Explicit Message: Restore from Flash Instruction

This instruction restores parameter values to the last values saved in Flash memory. Additional data is not sent with this instruction. Set data length fields to 0 and leave the source field blank.

Return to Factory Defaults Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Custom	5(hex)	23(hex)	0(hex)	0	0

Table 5.7 Explicit Message: Restore from Flash Instruction

This instruction restores parameter values to their factory defaults. *Factory Default Settings* are listed on page 16.   The Attribute field is not used and should be set to zero. Additional data is not sent with this instruction. Set data length fields to zero and leave the source field blank.

Reset Device Instruction

Service	Service Code	Class	Instance	Attribute	Source Length
Custom	5(hex)	1(hex)	1(hex)	0	0

Table 5.8 Explicit Message: Restore from Flash Instruction

This instruction forces the NR25 to perform a hardware reset. Network communications will be lost momentarily and all parameter values will be restored from Flash memory. The Attribute field is not used and should be set to zero. Additional data is not sent with this instruction. Set data length fields to zero and leave the source field blank.

Notes:

ADVANCED MICRO CONTROLS INC.

CIP POSITION SENSOR OBJECT

Common Industrial Protocol

DeviceNet is a protocol stack that implements the Common Industrial Protocol (CIP) over CANbus. The CIP is sponsored by the Open DeviceNet Vendors Association (ODVA) and is implemented over a variety of networks. The NR25 follows the Encoder Device Profile that is defined in the CIP specification. The explicit messages described in chapter 5 are custom instances that simplify configuring and programming the encoder when using explicit messaging.

In addition to these custom instances, the NR25 implements the Position Sensor Object, which is a mandatory object for every product that implements the Encoder Device Profile as defined in the specification. The explicit messages that are used to preset the position value and save the programmed parameters are two commands defined in the Position Sensor Object.

The NR25 implements the CIP revision 2 definition of the Position Sensor Object.

NOTE ▶ Using the Position Sensor Object to communicate with the NR25 is *completely optional*. Most applications should communicate with the NR25 using the custom instances as explained in the previous two chapters because it will greatly simplify your PLC programming. The only reasons to use the Position Sensor Object is if you need extremely fine grain control over communications with the NR25 or if you use DeviceNet encoders from multiple vendors and you decide to write code that can be used with any of these sensors.

Supported Services

The following table lists the common services implemented by the NR25 for the Position Sensor Object.

Service Code	Implemented		Service Name	Description of Service
	Class	Instance		
16#05	Yes	No	Reset	Resets all parameter values to the factory default
16#0E	Yes	Yes	Get_Attribute_Single	Returns the contents of the specified attribute
16#10	No	Yes	Set_Attribute_Single	Modifies an attribute value
16#15	Yes	No	Restore	Restores all parameter values from non-volatile storage
16#16	Yes	No	Save	Saves all programmable parameters to the non-volatile storage including the position offset derived from setting the Preset Value, (Attribute 16#13)

The services that are implemented only on the Class level (not on the Instance) should address Instance 0.

Table 5.9 Supported Services

➤ Service Code 16#0E, *Get_Attribute_Single* is used to read data from the Position Sensor Object class.

➤ Service Code 16#10, *Set_Attribute_Single* is used to write data to the Position Sensor Object class.

Supported Class Attributes

The only supported Class attribute is 1 and it returns the revision number of the definition of the object. Because the NR25 implements the CIP revision 2 definition of the Position Sensor Object, this attribute will always return a value of "2".

Supported Instance Attributes

Table 5.11 on this and the following page lists all of instance attributes implemented by the NR25. Table 5.10 below describes the Data Type values used in this table.

Data Type	Length	Description
BOOLEAN	8 bits	Holds single on/off (true/false) value
BYTE	8 bits	Holds up to 8 bits of data which should not be considered to be a scalar value
USINT	8 bits	Unsigned 8 bit value
WORD	16 bits	Holds up to 16 bits of data which should not be considered to be a scalar value
UINT	16 bits	Unsigned 16 bit integer value
DINT	32 bits	Signed 32 bit integer value
UDINT	32 bits	Unsigned 32 bit integer value

Table 5.10 Explanation of Data Types

NOTE ▷ When programming these instances, always use a Class Code of 16#23.

Attrib. ID	Access	Name	Data Type	Description
16#01 - 1	Get	Number of Attributes	USINT	Number of supported Attributes = 21
16#02 - 2	Get	Attribute List	Array of BYTE	List of supported Attributes = 01, 02, 0A, 0B, 0C …71hex
16#0A - 10	Get	Position Value Signed	DINT	Current position value
16#0B - 11	Get	Position Sensor Type	WORD	Specifies the device type 1 = Single turn absolute rotary encoder 2 = Multi-turn absolute rotary encoder
16#0C - 12	Set	Direction Counting Toggle	BOOLEAN	Controls the counting direction: 0 = CW 1 = CCW
16#0E - 14	Set	Scaling Function Control	BOOLEAN	Enables Scaling function 0 = OFF (65,536 counts per turn) 1 = ON (Scaling set by *Measuring Units per Span*, attribute 10hex)
16#10 - 16	Set	Measuring Units per Span (Counts per Turn)	UDINT	Resolution for one revolution: 1 to 65,536 counts per turn
16#11 - 17	Set	Total Measurement Range	UDINT	Counts before roll over to zero. **Single Turn NR25**: Range of 0, 2 to 65,536 **28 bit Multi-turn NR25**: Range of 0, 2 to 268,435,455 **30 bit Multi-turn NR25**: Range of 0, 2 to 1,073,741,823
16#13 - 19	Set	Preset Value	DINT	Sets the position to the specified value. Calculates an internal offset that will be saved to the non-volatile storage if Save service (code 16#16) is issued.
16#18 - 24	Get	Velocity Value	DINT	Current speed. The value is in the format specified by attribute 16#19

Supported Instance Attributes (continued)

Attrib. ID	Access	Name	Data Type	Description
16#19 - 25	Set	Velocity Format	WORD	Format of the velocity attribute: 16#1F04 = pulses/s 16#1F05 = pulses/ms 16#1F07 = steps/min 16#1F0F = RPM
16#29 - 41	Get	Operating Status	BYTE	Encoder diagnostic operating status. Bit 0 = Value of attribute 16#0C (12) Bit 1 = Value of attribute 16#0E (14)
16#2A - 42	Get	Physical Resolution Span	UDINT	Physical resolution of the single-turn resolver sensor
16#2B - 43	Get	Number of Spans	UINT	Maximum number of revolutions that could be measured.
16#2C - 44	Get	Alarms	WORD	Indicates a malfunction has occurred.
16#2D - 45	Get	Supported Alarms	WORD	Information about supported alarms
16#2E - 46	Get	Alarm Flag	BOOLEAN	Indicates that an alarm error occurred: 0 = No errors 1 = Alarm Error
16#33 - 51	Get	Offset Value	DINT	The internal position offset that is calculated after applying the Preset Value through attribute 13hex (19)
16#64 - 100	Set	Device Type	DINT	The way the device identifies itself: 16#22 (default) = Encoder device 16#00 = Generic device
16#70 - 112	Get	Actual Sensor Reading	UDINT	Raw position value read from NR25
16#71 - 113	Get	Time Stamp	UDINT	Value increments every 100 nanoseconds.

Table 5.11 Supported Instance Attributes

NOTE ⟩ For detailed description of the Attributes, see the CIP definition.

Supported Alarms

The NR25 supports the following operational alarm.

➢ Diagnostic Error

This alarm is set when the NR25 fails its power up diagnostics. The Position Error alarm is also set to indicate that the position data may be incorrect.

Attributes 16#2D, *Supported Alarms*, 16#2C, *Alarms*, and 16#2E *Alarm Flag* indicate something about the alarms supported by the NR25.

➢ **16#2D: Supported Alarms –** Reading this attribute returns a value of 3, indicating that the Position Error alarm and Diagnostic Error alarms are both used.

➢ **16#2C: Alarms –** Reading this attribute will return a value of zero if no alarms have occurred and a value of three if an alarm has occurred.

➢ **16#2E: Alarm Flag –** Reading this attribute will return a value of zero if no alarms have occurred and a value of one if an alarm has occurred.

> **The NR25 is a Group 3 device. In addition to the objects listed below, the device also supports the Connection and Acknowledge Handler classes.**

Assembly Object (04hex)

Supported Services

The following table lists the services implemented by the NR25 for the Assembly Object. The services that are implemented only on the Class level (not on the Instance) should address Instance 0.

Service Code	Implemented		Service Name	Description of Service
	Class	**Instance**		
$0E_{hex}$	Yes	Yes	Get_Attribute_Single	Returns the contents of the specified attribute
10_{hex}	No	Yes	Set_Attribute_Single	Modifies an attribute value

Table B.1 Assembly Object Supported Services

Assembly Object Class Attributes

The only supported Class attribute is 1 and it returns the revision number of the definition of the object. Because the NR25 implements the CIP revision 2, this attribute will always return a value of "2".

Assembly Instances

Service	Service Code	Class	Instance (decimal)	Attribute	Length (bytes)	Data
Get	$0E_{hex}$	4	1	3	4	Position
Get	$0E_{hex}$	4	3	3	8	Position and Velocity
Get	$0E_{hex}$	4	104	3	8	Position and Time Stamp
Get	$0E_{hex}$	4	105	3	8	Position and Actual Sensor Reading
Get	$0E_{hex}$	4	101	3	5	32 bit Preset Value and 8 bit command[†]
Set	10_{hex}	4	101	3	5	32 bit Preset Value and 8 bit command[†]
Get	$0E_{hex}$	4	103	3	12	Programmable Parameter Values[‡]
Set	10_{hex}	4	103	3	12	Programmable Parameter Values[‡]

Table B.2 Available Assembly Instances

[†] See the *Presetting the Position Value* section on page 38 for an explanation of how to use the command byte.

[‡] See the *Program Parameters Instruction* section on page 41 for information on the format of the parameter values.

DeviceNet Object (03hex)

Supported Services

The following table lists the services implemented by the NR25 for the DeviceNet Object. The services that are implemented only on the Class level (not on the Instance) should address Instance 0.

Service Code	Implemented		Service Name	Description of Service
	Class	Instance		
0E$_{hex}$	Yes	Yes	Get_Attribute_Single	Returns the contents of the specified attribute
10$_{hex}$	No	Yes	Set_Attribute_Single	Modifies an attribute value

Table B.3 DeviceNet Object Supported Services

DeviceNet Object Class Attributes

The only supported Class attribute is 1 and it returns the revision number of the definition of the object. Because the NR25 implements the CIP revision 2, this attribute will always return a value of "2".

DeviceNet Instance Attributes

Service	Service Code (hex)	Class	Instance (dec)	Attribute	Length (bytes)	Data
Get Set	0E 10	3	1	1	1	MAC ID (USINT data type)
Get Set	0E 10	3	1	2	1	BAUD_Rate (USINT data type)
Get Set	0E 10	3	1	3	1	BOI: Bus-off Interrupt (BOOL data type)
Get Set	0E 10	3	1	4	1	Bus-off Counter (USINT data type)
Get	0E$_{hex}$	3	1	5	2	Byte 1: Allocation Choice Byte Byte 2: MAC ID of Master
Get	0E$_{hex}$	3	1	6	1	MAC ID Switch changed since last power up? (BOOL data type)
Get	0E$_{hex}$	3	1	7	1	BAUD_Rate switch changed since last power up? (BOOL data type)
Get	0E$_{hex}$	3	1	8	1	Actual value of MAC ID switches (USINT data type)
Get	0E$_{hex}$	3	1	9	1	Actual value of BAUD_Rate switches (USINT data type) 0 = 125 kHz 1 = 250 kHz 2 = 500 kHz 3 = Read from Flash Memory

Table B.4 Available DeviceNet Instance Attributes

AMCI

ADVANCED MICRO CONTROLS INC.

20 GEAR DRIVE, TERRYVILLE, CT 06786 T: (860) 585-1254 F: (860) 584-1973

www.amci.com

LEADERS IN ADVANCED CONTROL PRODUCTS