AMCI ADVANCED MICRO CONTROLS INC.

Manual #: 940-0N120

NXFE2

mini-NEXUS

1 Axis Servo/Stepper Controller

with Integral 2-Port Ethernet Switch
Device Level Ring functionality for EtherNet/IP
Media Redundancy Protocol for PROFINET

E2 Technology

User Manual

AMCI mini-NEXUS Networked Specialty I/O

EtherNet/IP™

PROFI® INDUSTRIAL ETHERNET NET

Modbus

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 2021 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 2021 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) 583-1254. 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.

Waste Electrical and Electronic Equipment (WEEE)

At the end of life, this equipment should be collected separately from any unsorted municipal waste.

Table of Contents

GENERAL INFORMATION

Reference: About this Manual

Reference: Introduction to the NXFE2

Reference: Introduction to the NXFE2 (continued)

Reference: Move Profiles

Reference: Calculating Move Profiles

Reference: Homing the NXFE2

Reference: Configuration Mode Data Format

Reference: Command Mode Data Format

Reference: Command Mode Data Format (continued)

Reference: Configuring Network Interfaces

Task: Installing the NXFE2

Task: Set the IP Address and Protocol

Task: Implicit Communications with an EDS

Task: Implicit Communications Without an EDS

Task: EtherNet/IP Explicit Messaging

Task: Modbus TCP Configuration

Task: PROFINET Network Configuration

Task: PROFINET Network Configuration (continued)

About this Manual

Read this chapter to learn how to navigate through this manual and familiarize yourself with the conventions used in it.

Audience

This manual explains the set-up, installation, and operation of the NXFE2 Servo/Stepper Controller Module from AMCI. It is written for the engineer responsible for incorporating these modules into a design, as well as the engineer or technician responsible for its actual installation.

Applicable Units

This manual applies to all NXFE2 units.

Navigating this Manual

This manual is designed to be used in both printed and on-line formats. Its on-line form is a PDF document, which requires Adobe Acrobat Reader version 7.0+ to open it. The manual is laid out with an even number of pages in each chapter. This makes it easier to print a chapter to a duplex (double sided) printer.

Bookmarks of all the chapter names, section headings, and sub-headings were created in the PDF file to help navigate 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.

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 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 will open your default web browser to the referenced web page.

Trademark Notices

The AMCI logo is a trademark of Advanced Micro Controls Inc. “CIP” is a trademark of Open DeviceNet Vendor Association, Inc. “EtherNet/IP” is a trademark of ControlNet International, Ltd. under license by Open DeviceNet Vendor Association, Inc. “PROFINET” is a registered trademark of PROFIBUS & PROFINET International (PI). “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, 940-0N120, is the first revision of this manual. It was first released June 30th, 2021.

Revision History

940-0N120 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 each chapter’s contents to help you find the information you need to do your job.

Section Title	Intended Audience
Introduction to the NXFE2	Anyone new to the NXFE2. This chapter gives a basic overview of the features available on the unit, typical applications, and electrical specifications.
Move Profiles	Anyone that needs detailed information on how the NXFE2 can be used to control motion in your application.
Calculating Move Profiles	Anyone that need information on calculating detailed move profiles.
Homing the NXFE2	Anyone that needs detailed information on the available means to define a known position on the machine, including information on homing profiles available on the NXFE2.
Configuration Mode Data Format	Anyone that needs detailed information on the format of the network data to and from the NXFE2 used to configure the unit.
Command Mode Data Format	Anyone that needs detailed information on the format of the network data to and from the NXFE2 used to command the unit.
Configuring Network Interfaces	Instructions for the optional task of configuring network interfaces on your computer or laptop.
Installing the NXFE2	Anyone that must install an NXFE2 on a machine. Includes information on mounting, grounding, and wiring specific to the units.
Set the IP Address and Protocol	Task instructions that covers the options for setting the IP address on an NXFE2.
Implicit Communications with an EDS	Task instructions that cover how to add an NXFE2 to an EtherNet/IP host that supports the use of EDS files.
Implicit Communications Without an EDS	Task instructions for adding an NXFE2 to a project as a generic device. This configuration is for EtherNet/IP hosts that do not support EDS files while supporting implicit communications.
EtherNet/IP Explicit Messaging	Task instructions for adding message instructions to you host controller program that read and write data to the NXFE2 through message instructions.
Modbus TCP Configuration	Task instructions for communicating with an NXFE2 using the Modbus TCP protocol.
PROFINET Network Configuration	Task instructions that covers how to add an NXFE2 to a PROFINET network.

REFERENCE 1

INTRODUCTION TO THE NXFE2

This reference section contains the information you need to decide if the NXFE2 Servo/Stepper Controller Module is the right product for your application.

The NXFE2

The AMCI NXFE2 is a DC powered module that accepts command data from a host and generates step and direction pulses for controlling a servo or stepper system. The NXFE2 driver can be configured to use Ethernet/IP, PROFINET, or Modbus/TCP protocols to communicate with the host controller.

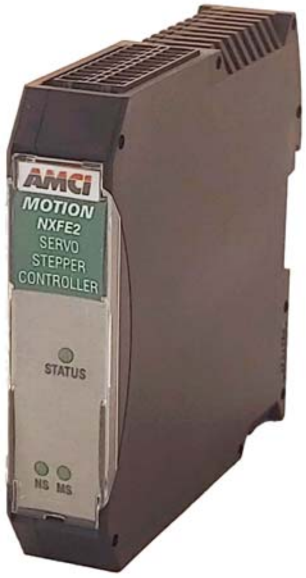


Figure R1.1 NXFE2 Module

The NXFE2 is a member of the growing line of products from AMCI that incorporate our *E2 Technology*. E2 Technology by AMCI is an innovative new multi-protocol approach to Ethernet distributed I/O.

E2 Technology products are simple and intuitive, allowing easy transition between multiple common industrial Ethernet protocols without the need to physically switch parts. An advanced web server integrated into all AMCI E2 Technology devices facilitates simple device configuration and troubleshooting via web-browser. Furthermore, an impressive array of advanced features for each supported protocol has been incorporated into the devices to meet many unique application requirements.

Each unit has two Ethernet ports which are internally connected through an onboard, two port, 10/100 Mbps ethernet switch. These ports allow you to wire your network in a “daisy-chain” fashion, which may lower network wiring costs and complexities.

The two ports also allow the units to function as members of a redundant Device Level Ring (DLR) network when using the EtherNet/IP protocol or as clients in a Media Redundancy Protocol (MRP) network when using PROFINET.

In DLR environments, the units act as Beacon-Based Ring Nodes. All units can process beacon packets at the default rate of every 400 microseconds. Beacon-based nodes can respond faster to network changes than nodes that only process Announce packets.

General Functionality

The NXFE2 module offers a motion control axis that function in an open-loop configuration. The axis can be used to drive a stepper or a servo drive with pulse inputs. Step/Direction and CW/CCW pulse formats are supported.

The NXFE2 offers a full 32 bit (±2 billion+) motor position register, move lengths of up to 2^{31} (±1 billion+) counts and programmable acceleration types. Differential and single-ended position pulse outputs are available, along with three discrete status outputs, five discrete inputs for various control functions, and three differential inputs for an incremental encoder.

The combination of host and driver gives you several advantages:

- Sophisticated I/O processing can be performed in the host (PLC or other controller) before sending commands to the Networked Indexer/Driver.
- All motion logic is programmed in the host, eliminating the need to learn a motion control language
- The integral two port Ethernet switch simplifies network cabling
- The DLR functionality eliminates single point failures in EtherNet/IP environments
- The MRP functionality eliminates single point failures in PROFINET environments

The NXFE2 (continued)

I/O connections

- Differential outputs to your servo or stepper driver. Maximum output frequency is 1 MHz.
- Single ended outputs to your servo or stepper driver. Use these outputs if your driver does not have differential inputs. Maximum output frequency is 100 kHz. You cannot use the differential and single-ended output at the same time.
- Differential quadrature encoder inputs. An encoder is typically used by the NXFE2 for open loop position feedback. See ***Encoder Functionality*** below for information on other uses of the encoder.
- Home Input. This single ended DC input is typically used when defining the home position on the machine.
- CW Limit and CCW Limit Inputs. These single ended inputs are used to define the maximum clockwise and counter-clockwise positions on the machine. If one of these inputs becomes active while traveling in that direction, the NXFE2 will immediately stop the move.
- Emergency Stop Inputs. If one or both of the CW and CCW Limits are not required, then the input can be configured as an Emergency Stop input. The NXFE2 will immediately stop the move if an Emergency Stop input becomes active.
- Capture Input. This single ended input can be used to capture the motor position or encoder position during a move. This is useful in applications where you must capture the position value and the event is too short to be captured by the host controller.
- External Input. A single ended input that can be used to bring moves to a controlled stop.
- Three Digital Outputs per axis. (Move_Complete, Online, and No_Fault.) These single ended outputs allow you externally monitor the status of the module. These outputs trigger on certain events, and must be set to their proper states by the host controller before and after they are triggered.

Encoder Functionality

In addition to the discrete I/O points, the NXFE2 has three inputs for a 5Vdc differential quadrature encoder. The inputs will also accept 12 to 24Vdc single ended encoder inputs with current limiting resistors.

Using the encoder inputs gives you the ability to:

- Home the machine to the encoder marker pulse
- Verify final move position

The encoder input also allows you to drive the motor through a feature called *Encoder Follower Mode*. In this mode, the servo or stepper motor follows the rotation of an external encoder. This encoder is typically attached to another motor. The ratio of encoder pulses to stepper pulses is programmable over a wide range. This mode electronically couples the two motors together through a programmable gear ratio.

Modes of Operation

The NXFE2 has two modes of operation, *Configuration Mode* and *Command Mode*.

Configuration Mode

This mode give you that ability to configure the axis for your application without having to set any programming switches. It also allows you to change the axis configuration at any time. The axis must be configured after every power up. The following parameters are set in Configuration Mode:

- The enabled or disabled state of each of the five inputs
- The configuration of the CW Limit and CCW Limit as overtravel limits or Emergency Stop inputs
- The enabled or disabled state of the Backplane Home Proximity bit
- If a quadrature encoder will be used or not
- If Diagnostic Feedback will be used to test the axis
- The Output Pulse type (CW/CCW or Step & Direction)
- If a homing operation will be to the Home Input or the encoder marker pulse
- The Starting Speed of most moves available with the NXFE2. This parameter allows you to start a move above the motor’s low frequency resonance point and is also useful in micro-stepping applications.
- The Homing Timeout value. This parameter allows you to automatically stop a homing operation if it takes longer to occur than expected.

Command Mode

This mode give you the ability to set the home position of the axis, program, carry out, and control, servo or stepper motor moves, detect and reset errors as they occur, and run diagnostics on the I/O if needed.

- Allows you to set the home position in the machine
- Allows you to preset the motor position register
- Allows you to preset the encoder position register
- Allows you to run six different move types
  - Relative Move
  - Absolute Move
  - Jog
  - Registration Move
  - Blend Move
  - Encoder Follower

  See chapter 2, ***Move Profiles***, starting on page 19, for a complete description of these moves.
- Allows you to hold and resume relative and absolute moves
- Allows you to bring a move to an immediate stop via a backplane bit or hard-wired input
- Allows you to reset command errors so a new command can be issued
- Allows you to feed the driver outputs into the encoder inputs and perform diagnostics.

Module Functionality

The table below lists the functionality offered by the NXFE2.

Feature	Description
Digital Inputs	Home Limit, CW Overtravel Limit, CCW Overtravel Limit, Capture Input, and External Input. Either or both Overtravel Limits can be configured as E-Stop Inputs.
Digital Outputs	No Fault, Move Complete, and Online DC outputs.
Encoder Inputs	Allows the NXFE2 to used a quadrature encoder for position verification or motion control through a mode called Encoder Follower.
Programmable Parameters	Starting Speed, Running Speed, Acceleration, Deceleration, Distance to Move, and Accel/Decel Types are fully programmable.
Homing	Allows you to set the machine to a known position. The NXFE2 can home to a discrete input or to an encoder marker pulse.
Jog Move	Allows you to jog the motor in either direction based on an input bit from your host controller.
Registration Move	Allows you to jog the motor in either direction based on an input bit from your host controller. When a controlled stop is received, the move will output a programmable number of steps before coming to a stop.
Relative Move	Allows you to drive the motor a specific number of steps in either direction from the current location.
Absolute Move	Allows you to drive the motor from one known location to another known location.
Blend Move	Allows you to perform a sequence of relative moves without stopping between them.
Hold Move	Allows you to suspend a move and restart it without losing your position value.
Resume Move	Allows you to restart a previously held move operation.
Immediate Stop	Allows you to immediately stop all motion if an error condition is detected by your host controller.
Encoder Follower	The NXFE2 can be configured to control the position of a motor based on feedback from an external encoder. The ratio of encoder pulses to motor pulses is fully programmable and can be changed on-the-fly.

Table R1.1 Indexer Functionality

Specifications

Motor Control Outputs

Differential and single-ended output available.

Programmable CW/CCW or Step & Direction formats

Discrete Inputs

Five
Home, Capture, CW Limit, CCW Limit, and External

CCW Limit and CCW Limit inputs can be programmed to act as Emergency Stop inputs.

Discrete Outputs

Three
Output state controlled by host controller. Outputs will also transition on move complete, input error, or on-line fault events.

Encoder Inputs

+5Vdc differential ABZ inputs
X4 decoding always used
Maximum pulse train frequency: 250KHz
(1MHz maximum counting frequency)

Encoder can be used for position feedback, homing to Z pulse, and with Encoder Follower Mode.

Number of I/O Words Required

10 Input Words, 10 Output Words
Words are divided into a mixture of 16 and 32 bit integer values.

Differential Stepper Outputs

+5Vdc differential (3.5Vdc to 4Vdc peak typical)
Maximum Output Current: 20mA
Maximum Output Frequency: 1MHz

Single-Ended Stepper Outputs

Open Collector Sinking
Voltage Range: 12Vdc to 24Vdc
Maximum Output Current: 30mA
Maximum Output Frequency: 100KHz
An external supply is required for operation.

Discrete Input Electrical Specs

Open Collector Sinking
All inputs share a common return
ON Voltage Range: 8Vdc to 24Vdc
OFF Voltage Range: 0Vdc to 5Vdc
Maximum Input Current: 15mA @ 24Vdc
An external supply is required for operation.

Discrete Output Electrical Specs

Same electrical specifications as the Single-Ended Stepper Outputs

Throughput Time

100 microseconds

Inputs must remain in a steady state for 100 microseconds after a transition for the transition to be recognized.

Programming Storage

RAM Memory.
The NXFE2 must be configured on power up.

Power Supply

10 to 28 Vdc, surge to 30Vdc without damage to module. 112 mA @ 24Vdc (2.7W)

Environmental Conditions

Operating Temperature:	-4 to 122°F (-20 to 50°C)
Relative Humidity:	5 to 95% (w/o condensation)
Storage Temperature:	-40 to 185°F (-40 to 85°C)

Connectors

Mating connectors are included with the NXFE2 and are available separately under the following AMCI part numbers.

Connector	AMCI Part #	Wire	Strip Length	Min. Tightening Torque
I/O	MS-2X11	28 - 16 AWG	0.275 inches	Spring Cage Connector
Power	MS-4M	28 - 12 AWG	0.394 inches	4.43lb-in (0.5 Nm)

I/O Connector

As shown in figure R1.2, the I/O connector is located on the top of the module. All digital I/O connections are made at this connector. The mating connector is supplied with the NXFE2 and is also available from AMCI under the part number MS-2X11. It can be ordered from Phoenix Contact under their part number 173 88 98.

Motor Control Outputs

The NXFE2 can be programmed to output pulses in either Step & Direction format or CW/CCW format. The module offers both differential and single ended motor control outputs. The 5 Vdc differential outputs offer a maximum output frequency of 1 MHz. The single ended outputs are open collector sinking. They work with a supply voltage or 12 to 24 Vdc and can sink a maximum current of 30 mA. The maximum output frequency of the single ended outputs is 100 kHz.

NOTE ➤ The two output types cannot be used together.

Regardless of the output type used, the NXFE2 can still be programmed to output pulses at 1MHz. The single ended outputs are not guaranteed to work correctly above 100 kHz.

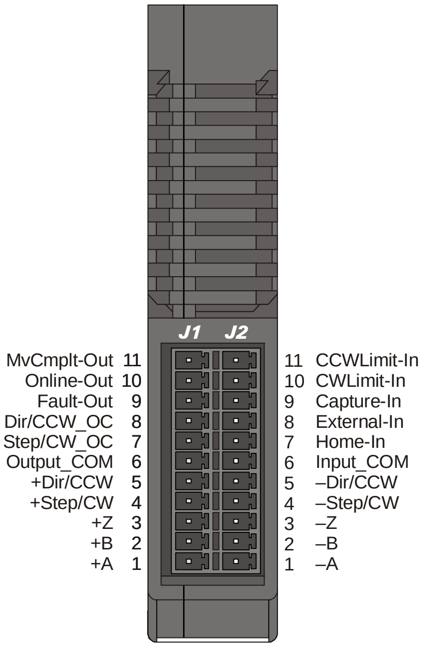


Figure R1.2 I/O Connector

Quadrature Encoder Inputs

The NXFE2 has three differential inputs available for a quadrature encoder. The use of an encoder is optional. X4 decoding is used when determining encoder position. The encoder is typically used for position verification. Note that the NXFE2 is an open loop position controller. An encoder attached to the NXFE2 cannot be used to “close the loop” in a servo style control. When the NXFE2 is used to supply pulses to a servo controller, position feedback must be brought into the servo controller, not the NXFE2.

When the encoder inputs are enabled, the NXFE2 can be programmed to home the motor position to the Z input. The encoder position can also be captured on the transition of the discrete DC Capture Input.

It is also possible to use an encoder that is not mechanically coupled to the motor controlled by the NXFE2. This configuration allows you to monitor the encoder data, or use a feature called *Encoder Follower*. When this feature is active, the NXFE2 will change the motor control outputs in response to a change in encoder position. The ratio of encoder turns to motor turns is fully programmable.

The encoder inputs are designed to interface with a 5 Vdc differential encoder. They can be used with encoders with single ended outputs through proper wiring and the use of an external current limiting resistor.

Discrete Inputs

The NXFE2 offers a total of five DC digital inputs. These inputs are single ended and have a single common pin. The common pin must be tied to the power supply common for proper operation. The inputs accept 8 to 27 Vdc (12 to 24 Vdc nominal) without the need of a external current limiting resistor.

The inputs can be individually programmed to act as normally open or normally closed contacts. The inputs have the following functions:

CW & CCW Limit

These inputs define the maximum limits of travel. CW travel is defined by an increasing motor position value. CCW travel is defined by a decreasing motor position value. In general, activating a limit switch during a move will bring the move to an immediate stop. When a limit is active, the only motion allowed by the NXFE2 is a jog or registration move in the opposite direction. For example, if the CW Limit is active, the NXFE2 will only allow a CCW jog or registration move. This allows you to move off of an activated limit switch.

Discrete Inputs (continued)

Emergency Stop Input

Either, or both, of the CW and CCW Limit inputs can be programmed as an Emergency Stop Input. When programmed as an Emergency Stop Input, the motor control outputs are brought to an immediate stop when the input becomes active, regardless of the direction of motion. A move cannot be started while an Emergency Stop Input is active.

Home Input

Many applications require that the machine be brought to a known position before normal operation can begin. This is commonly called "homing" the machine or bringing the machine to its "home" position. The NXFE2 allows you to define this starting position in three ways. The first is with a Position Preset Command. The second is with a sensor mounted on the machine. When you define one of the inputs as the Home Input, you can issue commands to the NXFE2 that will cause the unit to seek this sensor. The third option is homing to the Z pulse of a quadrature encoder. When using the Z pulse, you can use the Home Input as a home proximity sensor. How the NXFE2 actually finds the Home sensor is described in chapter 4, ***Homing the NXFE2***, starting on page 39.

External Input

The External Input can be used to bring some moves to a Controlled Stop. (The motion will decelerate before it comes to a stop.) Not all moves are affected by this input. The section ***Basic Move Types***, starting on page 23, describes each move type in detail, including if the move is affected by this input.

Capture Input

The Capture Input can be used to capture a position value during a move. This is useful in applications where you must capture the position value and the event is too short to be captured by the host controller. If the NXFE2 is configured to use an external quadrature encoder, the encoder position will be captured. If the module is not configured to use an external quadrature encoder, the motor position will be captured.

Discrete Outputs

There are three DC digital outputs on the NXFE2. The state of the outputs can be monitored and controlled by the host controller. These outputs are Move Complete, Online, and No_Fault.

All outputs are open collector sinking. An output is connected to the Output_COM pin when the output is active. The Output_COM pin must be connected to the output power supply common for operation. The output power supply can be within the range of 12 to 24 Vdc nominal. (8 Vdc to 27 Vdc max.)

An output is "active" when it is conducting current. The output state is reported to the host controller as a "1" when the output is active. When measuring the output voltage on the pin, it will be near zero volts when the output is active.

Move Complete

Set to its active state when an Absolute, Relative, or Blend move completes without error. If this output is used, the host controller must reset the output between moves.

Online

The host controller must set this bit to its active state. The NXFE2 will reset this bit if there is a module reset or a loss of network communications.

No_Fault

The host controller must set this bit to its active state. The NXFE2 will reset this bit if there an Input Error during the move, a new configuration is written to the module, or there is a loss of network communication.

Discrete Outputs (continued)

Set_State Bit

Some host controllers will continuously write new data to the NXFE2. Because of this, the NXFE2 implements a “Set_State” bit. Output states can only be changed by the host controller when this bit equals “1”.

NOTE

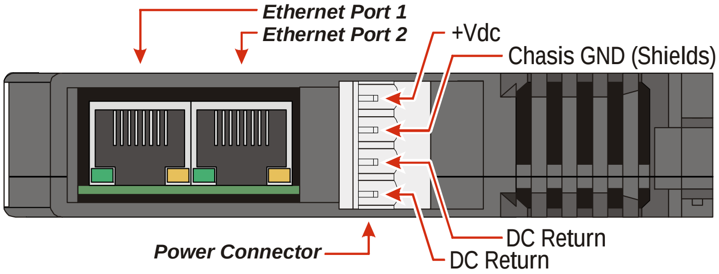
1) All three outputs are updated when the Set_State bit equals “1”. Your host controller logic must take this into consideration when updating the state of the outputs.

2) With the exception of the General Purpose output, the Set_State bit cannot be used to force the outputs on or off. When a No_Fault, Move_Complete, of On_Line event occurs, the outputs will change to their designated states for a maximum of five milliseconds before returning to the states specified by the host controller.

3) The Set_State bits should be reset during a move operation. This will prevent the host controller from overriding the No_Fault and Move_Complete outputs while a move is occurring.

Ethernet Ports

The two Ethernet ports are located on the bottom of the NXFE2. When using the drive in non-redundant networks, either port can be used to attach to your core network. The second port can be used to daisy-chain an additional device onto the network. When installed in a DLR or MRP network, both ports are used.



Bottom View

Figure R1.3 Location of Ethernet Ports and Motor Connector

Power Connector

Figure R1.3 also shows the location of the Power Connector. The mate to this connector is included with the NXFE2 and is also available from AMCI under the part number MS-4M. It can be ordered from Phoenix Contact under their part number 187 80 37.

Front Panel

The front panel of the NXFE2 contains the drive status LED as well as the two network status LED's.

Status LED

The Status LED is a bi-color red/green LED shows the general status of the module.

- **Steady Green:** Module OK, no motion is in progress.
- **Blinking Green:** Module OK, motion is occurring.
- **Blinking Red:** Configuration Error, Command Error, Input Error, Home Invalid Error, Invalid Profile Error.
- **Steady Red:** No physical network connection.

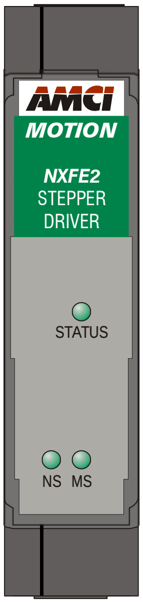


Figure R1.4 NXFE2 Front Panel

Module Status (MS) LED

The Module LED is a bi-color red/green LED that indicates the health of the Ethernet co-processor on the module.

LED State	EtherNet/IP Definition	Modbus TCP Definition	PROFINET Definition
Off	No Power		
Alternating Red/Green	Power up Self-Test (Occurs very quickly on power up.)		
Flashing Green	Waiting for valid physical connection to the network.		Not Implemented
Steady Green	Drive and Network are operational.		Device Name or IP Address are set.
Flashing Red	Initializing: IP Address Conflict		Initializing: Device Name or IP Address are not set.
	The IP Address or Network Protocol has been changed. (The Network Status LED will also be flashing red.) Cycle power to the unit to continue.		
Steady Red	Major Fault. Cycle power to the unit to attempt to clear the fault.		

Table R1.2 Module Status LED States

Front Panel (continued)

Network Status (NS) LED

The Network Status LED is a bi-color red/green LED. The state of the LED depends on the protocol the NXFE2 is configured to for.

LED State	EtherNet/IP Definition	Modbus TCP Definition	PROFINET Definition
Off	No Power OR No IP Address		No power, duplicate IP address on the network, mis-match in Device Name, or no connection to IO Controller.
Alternating Red/Green	Power up Self-Test (Occurs very quickly on power up.)		
Flashing Green	Ethernet connection, but no CIP connections	Indicates number of connections with 2 second delay between groups. The unit supports up to 5 concurrent connections.	On-line, Stop state. A connection with the IO Controller is established and it is in its STOP state.
Steady Green	Valid Ethernet network and CIP connections	Not Implemented	On-line, Run state. A connection with the IO Controller is established and it is in its RUN state.
Flashing Red	If the MS LED is steady green: Network Connection Timeout	Not Implemented	Not Implemented
	If the MS LED is blinking green: IP Address or Network Protocol changed: Cycle power		
Steady Red	Duplicate IP address on network.		Not Implemented

Table R1.3 Network Status LED States

REFERENCE 2

MOVE PROFILES

When a move command is sent to the NXFE2, the module calculates the entire profile before starting the move or issuing an error message. This chapter explains how the profiles are calculated and the different available moves.

Definitions

Units of Measure

Distance: Every distance is measured in steps. Your driver and motor combination determines the specific number of steps needed to complete one rotation of the motor's shaft. It is up to you to determine how many steps are required to travel the appropriate distance in your application.

Speed: All speeds are measured in steps/second. Since the number of steps needed to complete one shaft rotation is determined by setup, it is up to you to determine how many steps per second is required to rotate the motor shaft at your desired speed.

Acceleration: The typical unit of measure for acceleration and deceleration is steps/second/second, or steps/second^2. However, when programming an NXFE2, all acceleration and deceleration values must be programmed in the unit of measure of steps/second/millisecond.

- To convert from steps/second^2 to steps/second/millisecond, divide the value by 1000. This must be done when converting from a value used in the equations to a value programmed into the unit.
- To convert from steps/second/millisecond to steps/second^2, multiply the value by 1000. This must be done when converting from the value programmed into a unit to the value used in the equations.

Motor Position

Motor Position is defined in counts and the range is -2,147,483,648 to +2,147,483,647.

Home Position

The Home Position is any position on your machine that you can sense and stop at. There are two ways to defining the Home Position. The first is using the Preset Position command to set the Motor Position register to a known value. The second method is using one of the *Find Home* commands. If you use the unit's *Find Home* commands, the motor position and encoder position registers will automatically be set to zero once the home position is reached. Defining a Home Position is completely optional. Some applications, such as those that use the NXFE2 for speed control, don't require position data at all.

Count Direction

Relative moves with a positive offset will increase the motor position register reported back to the host. A negative offset will decrease the motor position register. Some of the moves, such as a Jog Move, have a positive and negative command. A positive command, such as the CW Jog Move command, will result in an increase in the motor position value.

Starting Speed

The Starting Speed is the speed that most moves will begin and end at. This value is set while configuring the unit and it has a valid range of 1 to 1,000,000 steps/second. This value is typically used to start the move above the motor's low frequency resonances and, in micro-stepping applications, to limit the amount of time needed for acceleration and deceleration. AMCI does not specify a default value in this manual because it is very dependent on motor size and attached load. With that said, a starting speed between 0.25 and 0.5 RPS is usually a good starting point.

Definitions (continued)

Target Position

The Target Position is the position that you want the move to end at. There are two ways to define the Target Position, with relative coordinates or absolute coordinates.

Relative Coordinates

Relative coordinates define the Target Position as an offset from the present position of the motor. Most NXFE2 moves use relative coordinates.

- The range of values for the Target Position when it is treated as an offset is ±1,073,741,823 counts.
- The Motor Position value reported back to the host can exceed ±1,073,741,823 counts. The only way to move beyond ±1,073,741,823 counts is with multiple relative moves or jog commands.

Absolute Coordinates

Absolute coordinates treat the Target Position as an actual position on the machine. Note that you must set the Home Position on the machine before you can run an Absolute Move. (See ***Home Position*** on the previous page.)

- The range of values for the Target Position when it is treated as an actual position on the machine is ±1,073,741,823 counts. The move will be positive if the Target Position is greater than the Current Position and negative if the Target Position is less than the Current Position.
- The Motor Position value reported back to the host can exceed ±1,073,741,823 counts. However, you cannot move beyond ±1,073,741,823 counts with an Absolute Move. The only way to move beyond ±1,073,741,823 counts is with multiple relative moves or jog commands.

Definition of Acceleration Types

Most of the move commands allow you to define the acceleration type used during the move. The NXFE2 supports three types of accelerations and decelerations.

Linear Acceleration

The axis accelerates (or decelerates) at a constant rate until the programmed speed is reached. This offers the fastest acceleration, but consideration must be given to insure the smoothest transition from rest to the acceleration phase of the move. The smoothest transition occurs when the configured Starting Speed is equal to the square root of the programmed Acceleration value. Note that other values will work correctly, but you may notice a quick change in velocity at the start or end of the acceleration phase.

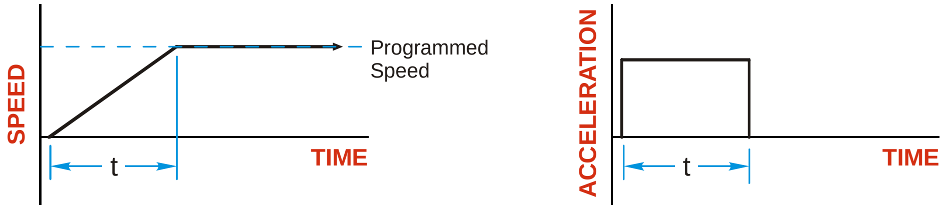


Figure R2.1 Linear Acceleration

Additional information, including example move calculations, can be found in section 3, ***Calculating Move Profiles*** starting on page 33.

Definition of Acceleration Types (continued)

Triangular S-Curve Acceleration

The axis accelerates (or decelerates) at a constantly changing rate that is slowest at the start and end of the acceleration phase of the move. The Triangular S-Curve type offers the smoothest acceleration, but it takes twice as long as a Linear Acceleration to achieve the same velocity.

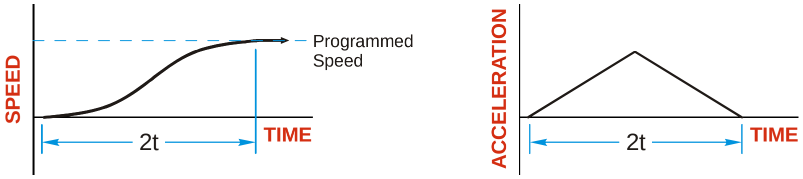


Figure R2.2 Triangular S-Curve Acceleration

Additional information, including example move calculations, can be found in section 3, ***Calculating Move Profiles*** starting on page 33.

Trapezoidal S-Curve Acceleration

The Trapezoidal S-Curve acceleration is a good compromise between the speed of Linear acceleration and the smoothness of Triangular S-Curve acceleration. Like the Triangular S-Curve, this acceleration type begins and ends the acceleration phase smoothly, but the middle half of the acceleration phase is linear. Because of this, the Trapezoidal S-Curve acceleration only requires 33% more time to achieve the same velocity as a Linear Acceleration, compared to the 100% more time of a Triangular S-Curve acceleration.

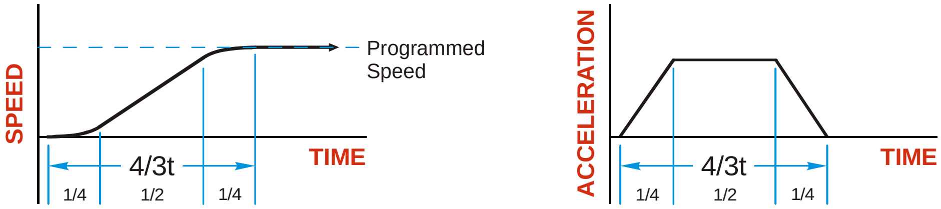


Figure R2.3 Trapezoidal S-Curve Acceleration

A Simple Move

As shown in the figure below, a move from A (Current Position) to B (Target Position) consists of several parts.

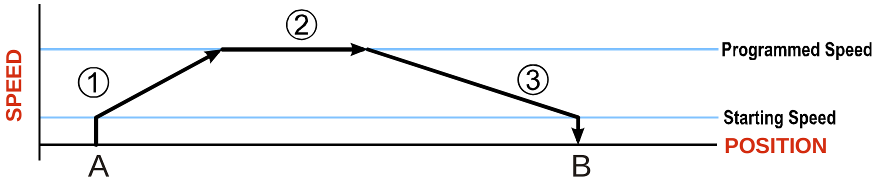


Figure R2.4 A Trapezoidal Profile

1) The move begins at point A, where the motor jumps from rest to the configured *Starting Speed*. The motor then accelerates at the programmed *Acceleration Value* until the speed of the motor reaches the *Programmed Speed*. Both the Acceleration Value and the Programmed Speed are programmed when the move command is sent to the NXFE2.
2) The motor continues to run at the Programmed Speed until it reaches the point where it must decelerate before reaching point B.
3) The motor decelerates at the *Deceleration Value*, which is also programmed by the move command, until the speed reaches the Starting Speed, which occurs at the Target Position (B). The motor stops at this point. Note that the acceleration and deceleration values can be different in the move.

A Simple Move (continued)

Figure R2.4 on the previous page shows a Trapezoidal Profile. A Trapezoidal Profile occurs when the Programmed Speed is reached during the move. This occurs when the number of steps needed to accelerate and decelerate are less than the total number of steps in the move. Figure R2.5 below shows a Triangular Profile. A Triangular Profile occurs when the number of steps needed to accelerate to the Programmed Speed and decelerate from the Programmed Speed are greater than the total number of steps in the move. In this case, the profile will accelerate as far as it can before decelerating and the Programmed Speed is never reached.

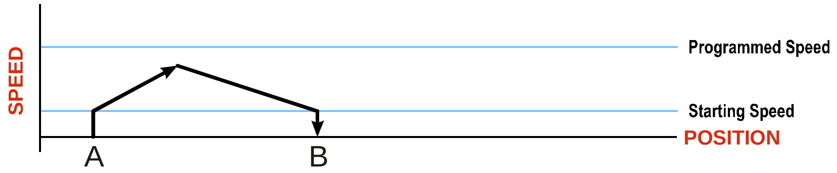


Figure R2.5 A Triangular Profile

Controlled and Immediate Stops

Once a move is started, there are several ways to stop the move before it comes to an end. These stops are broken down into two types:

- **Controlled Stop:** The NXFE2 immediately begins decelerating at the move's programmed deceleration value until it reaches the configured Starting Speed. Motion stops at this point. The motor position value is still considered valid after a Controlled Stop and the machine does not need to be homed again before Absolute Moves can be run.
- **Immediate Stop:** The NXFE2 immediately stops outputting pulses regardless of the speed the motor is running at. Because it is possible for the inertia of the load attached to the motor to pull the motor beyond the stopping point, the motor position value is considered invalid after an Immediate Stop and the machine must be homed again before Absolute Moves can be run.

Host Control

Hold Move Command: This command can be used with some moves to bring the axis to a Controlled Stop. The move can be resumed and finished, or it can be aborted. Not all moves are affected by this command. The section ***Basic Move Types***, starting on page 23, describes each move type in detail, including if the move is affected by this command.

Immediate Stop Command: When this command is issued from the host, the axis will come to an Immediate Stop. The move cannot be restarted and the machine must be homed again before Absolute Moves can be run.

Hardware Control

External Input: Triggering this input will bring the move to a controlled stop. The controlled stop is triggered on an inactive-to-active state change on the input. Only Relative, Absolute, Jog, and Registration Moves can be stopped this way, all other moves ignore this input. Relative and Absolute moves are put in a hold state, and have the option of being continued with a Resume Command from the host controller.

CW Limit and CCW Limit Inputs: In most cases, activating these inputs during a move will bring the axis to an Immediate Stop. The exceptions are the *CW/CCW Find Home* commands, the *CW/CCW Jog Move* commands, and the *CW/CCW Registration Move* commands. The *CW/CCW Find Home* commands are explained in chapter 3, ***Homing the NXFE2***, which starts on page 39. The ***CW/CCW Jog Move*** commands are fully explained on page 25, and the ***CW/CCW Registration Move*** commands are fully explained on page 27.

Emergency Stop Input: It is possible to configure either or both of the CW and CCW inputs as Emergency Stop Inputs. When an Emergency Stop Input is activated, the axis will come to an Immediate Stop, regardless of the direction of travel.

Controlling Moves In Progress

The NXFE2 has the ability to place a running move on hold and later resume the move if an error did not occur while the move was in its Hold state. One potential application for this feature is bringing a move to a controlled stop when your controller senses an end-of-stock condition. The move can be put in its Hold state until the stock is replenished and then the move can be resumed.

Note that you do not have to resume a move once it has been placed in its Hold state. You can place a move in its Hold state to prematurely end the move with a controlled stop and issue any type of new move from the stopped position.

The figure below shows a profile of a move that is placed in its Hold state and later resumed.

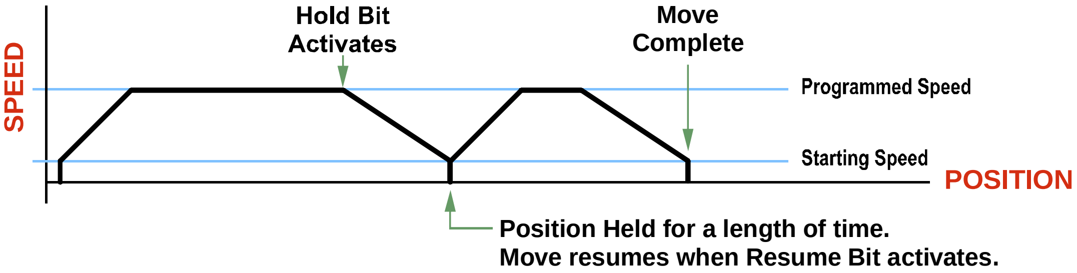


Figure R2.6 Hold/Resume a Move Profile

The move types are explained in the following sections. Refer to the appropriate section to learn how the Hold and Resume Bits are used for that move type.

Basic Move Types

Relative Move

Relative Moves move an offset number of steps (n) from the Current Position (A). A trapezoidal profile is shown to the right, but Relative Moves can also generate triangular profiles. The command's Target Position is the move's offset. The offset can be in the range of ±1,073,741,823 counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.

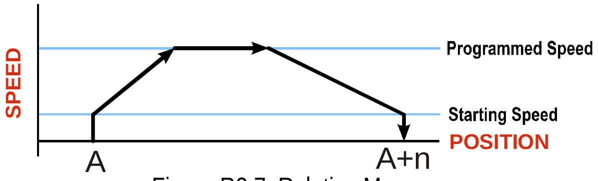


Figure R2.7 Relative Move

NOTE

1) You do not have to preset the position or home the machine before you can use a Relative Moves. That is, the Position Invalid status bit can be set.

2) Relative Moves allow you to move your machine without having to calculate absolute positions. If you are indexing a rotary table, you can preform a relative move of 30° multiple times without recalculating new positions in your controller. If you perform the same action with Absolute Moves, you would have to calculate your 30° position followed by your 60° position, followed by your 90° position, etc.

Relative Moves can be brought to a Controlled Stop by using the Hold Move Command from the network data or by triggering the External Input. When either of these two inputs are used, the axis will immediately decelerate at the programmed rate and stop. When stopped successfully, the NXFE2 will set a *Hold State* bit in the input data table. The Relative Move can be restarted with the Resume Move command from the network data or the move can be aborted. The Resume Move command allows you to change the move's Programmed Speed, Acceleration Value and Type, and the Deceleration Value and Type. The Target Position cannot be changed with the Resume Move Command.

Basic Move Types (continued)

Relative Move (continued)

Controlled Stops

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data or trigger the External Input. Note that your holding position will most likely not be the final position you commanded. You can resume a held Relative Move by using the Resume Move command or abandon the held move by starting a new one. The use of the Hold Move and Resume Move bits is further explained in the ***Controlling Moves In Progress*** section starting on page 23.
- If the External Input is active when a Relative Move is started, the NXFE2 will output one step in the specified direction before entering the Hold State.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- The E-Stop Input is activated. The E-Stop Input must be deactivated before a new move command will be accepted.
- A CW/CWW Limit Switch is activated. Only Jog or Registration moves will be accepted while the Limit Switch remains active. The Jog or Registration move must be in the opposite direction of the activated switch. This allows you to move off of the activated Limit Switch.

Absolute Move

Absolute Moves move from the Current Position (A) to a given position (B). (The NXFE2 calculates the number of steps needed to move to the given position and moves that number of steps.) A trapezoidal profile is shown to the right, but Absolute Moves can also generate triangular profiles. The command's Target Position can be in the range of ±1,073,741,823 counts. The move will be clockwise if the Target Position is greater than the Current Position and counter-clockwise if the Target Position is less than the Current Position.

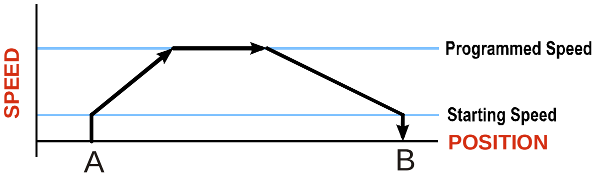


Figure R2.8 Absolute Move

NOTE

1) The *Home Position* of the machine must be set before running an Absolute Move. See chapter 4, ***Homing the NXFE2***, which starts on page 39, for information on homing the machine.
2) The Motor Position must be valid before you can use an Absolute Move. The Motor Position becomes valid when you preset the position or home the machine.
3) Absolute Moves allow you to move your machine without having to calculate relative positions. If you are controlling a rotary table, you can drive the table to any angle without having to calculate the distance to travel. For example an Absolute Move to 180° will move the table to the correct position regardless of where the move starts from.

If the External Input is active when an Absolute Move is started, the NXFE2 will output one step in the specified direction before entering the Hold State.

Basic Move Types (continued)

Absolute Move (continued)

Controlled Stops

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data or trigger the External Input. Note that your holding position will most likely not be the final position you commanded. You can resume a held Absolute Move by using the Resume Move bit or abandon the held move by starting a new one. The use of the Hold Move and Resume Move bits is explained in the ***Controlling Moves In Progress*** section starting on page 23.
- If the External Input is active when a Absolute Move is started, the NXFE2 will output one step in the specified direction before entering the Hold State.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- The E-Stop Input is activated. The E-Stop Input must be deactivated before a new move command will be accepted.
- A CW/CWW Limit Switch is activated. Only Jog or Registration moves will be accepted while the Limit Switch remains active. The Jog or Registration move must be in the opposite direction of the activated switch. This allows you to move off of the activated Limit Switch.

CW/CCW Jog Move

Jog Moves move in the programmed direction as long as the command is active. Two commands are available, the CW Jog Move will cause clockwise steps while the CCW Jog Move will cause counter-clockwise steps. These commands are often used to give the operator manual control over the axis.

NOTE ➤ The Target Position value must be zero during a Jog Move. If it is not zero, the move will actually be a ***CW/CCW Registration Move***, and the final stopping position may be unexpected.

Jog Moves are typically used to drive the machine under direct operator control, but they can also be used when you are interested in controlling the speed of the shaft instead of its position. One such application is driving a conveyor belt. To accommodate these applications, the running speed, acceleration, and deceleration of the Jog Move can be changed *while the move is in progress*.

The CW Limit and CCW Limit inputs behave differently for CW/CCW Jog Moves and CW/CCW Registration Moves than all other move types. Like all moves, activating a limit in the same direction of travel will bring the move to an Immediate Stop. Unlike other moves, a Jog or Registration move can be started when an end limit switch is active provided that the commanded direction is opposite that of the activated switch. For example, a CW Jog Move can be issued while the CCW limit switch is active. This allows you to move off of an activated end limit switch.

If the initial Programmed Speed is greater than the configured Starting Speed when the command is issued, then the move's Programmed Speed, Acceleration Value and Type, and Deceleration Value and Type can be changed while the move is running. The axis will accelerate or decelerate to the new Programmed Speed when it is changed. Note that the acceleration and deceleration parameters are not changed unless the Programmed Speed is also changed.

If the initial Programmed Speed is set to a value less the configured Starting Speed when the command is issued, the NXFE2 will perform a *Constant Speed Jog*. The move will run at the Programmed Speed only. The move's Programmed Speed, Acceleration Value and Type, and Deceleration Value and Type cannot be changed while the move is running. Attempting to do so will cause an error.

Basic Move Types (continued)

CW/CCW Jog Move (continued)

If the Programmed Speed is set to zero when the command is issued, the NXFE2 will perform a *One Shot Jog* and output one pulse in the specified direction. The pulse is output at the configured Starting Speed.

The figure below shows a Jog Move where the Programmed Speed is greater than the starting speed. The Jog Move begins at the programmed Starting Speed, accelerates at the programmed rate to the Programmed Speed. When the Programmed Speed is changed, the motor will accelerate or decelerate to the new Programmed Speed using the new accelerate/decelerate parameter values. If you write a Programmed Speed to the NXFE2 that is less than the configured Starting Speed, the Jog Move will continue at the present speed and the *Invalid_Jog_Change* status bit will be set to “1”.

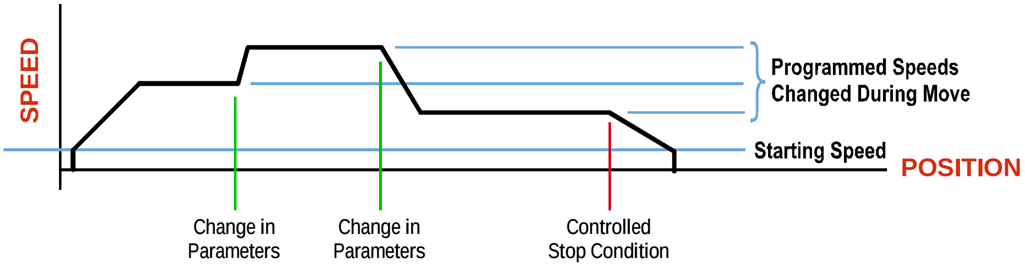


Figure R2.9 Jog Move

Controlled Stops

- The Jog Move Command bit is reset to “0”.
- An inactive-to-active transition on the External Input.
- If the External Input is active when a Absolute Move is started, the NXFE2 will output one step in the specified direction before entering the Hold State.

NOTE Issuing a Hold Move command during a Jog operation will result in a Command Error response from the NXFE2. The Jog move will continue.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- The E-Stop Input is activated. The E-Stop Input must be deactivated before a new move command will be accepted.
- A Limit Switch is activated in the same direction of the Jog move. (For example, the CW Limit Switch activates during a CW Jog move.) A Limit Switch is activated in the opposite direction of the Jog move is ignored.

NOTE Note that it is possible to *start* a move while a CW or CCW Limit Switch is active as long as the direction of travel is *opposite* that of the activated Limit Switch. For example, it is possible to start a clockwise Jog Move while the CCW Limit Switch is active. This allows you to move off of an activated Limit Switch.

Basic Move Types (continued)

CW/CCW Registration Move

Similar to a Jog Move, a Registration Move will travel in the programmed direction as long as the command is active. CW Registration Moves will cause clockwise motion when the motor is wired as shown in this manual, while CCW Registration Moves will cause counter-clockwise motion. When the command terminates under Controlled Stop conditions, the NXFE2 will output a programmed number of steps as part of bringing the move to a stop. Note that all position values programmed with a Registration Move are relative values, not absolute machine positions.

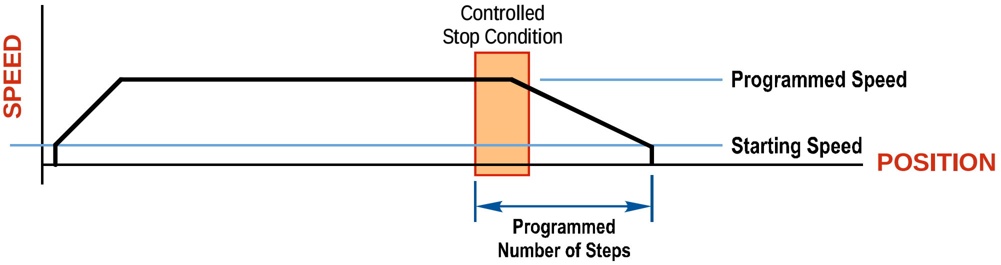


Figure R2.10 Registration Move

If the Programmed Number of Steps are less than the number of steps needed to bring the axis to a stop based on the Programmed Speed and Deceleration values set with the command, the NXFE2 will issue an error message instead of starting the move.

Like the CW/CCW Jog Moves, activating a limit that is opposite to the direction of travel, for example activating the CCW Limit during a CW Registration Move, has no effect.

An additional feature of the NXFE2 is the ability to program the driver to ignore the Controlled Stop conditions until a minimum number of steps have occurred. This value is programmed through the Minimum Registration Move Distance parameter, which is programmed with a separate programming block. The Minimum Registration Move Distance parameter is stored in memory and use for all registration moves once set. (Its value can be changed at any time.) The figure below shows how the Minimum Registration Move Distance parameter affects when the Stop Condition is applied to the move. As shown in the second diagram, Controlled Stop conditions are level triggered, not edge triggered. If a Controlled Stop Condition occurs before the Minimum Registration Move Distance is reached and stays active, the move will begin its controlled stop once the Minimum Registration Move Distance is reached.

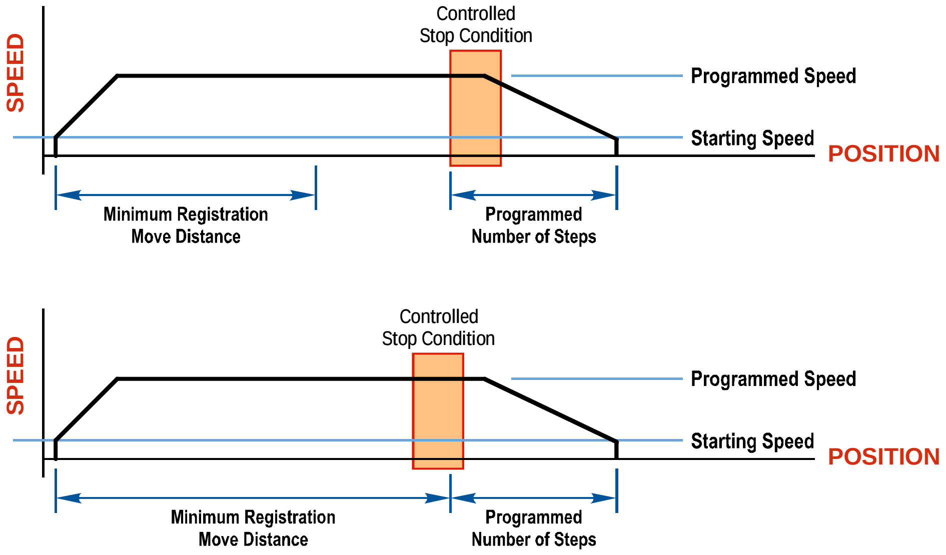


Figure R2.11 Min. Registration Move Distance

Basic Move Types (continued)

CW/CCW Registration Move (continued)

An error message will be issued if the External Input is active when a Registration Move is initiated and the minimum registration distance is set to zero.

Controlled Stops

- The Registration Move Command bit is reset to “0”.
- An inactive-to-active transition on the External Input.

NOTE Issuing a Hold Move command during a Registration move will result in a Command Error response from the NXFE2. The Registration move will continue.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- The E-Stop Input is activated. The E-Stop Input must be deactivated before a new move command will be accepted.
- A Limit Switch is activated in the same direction of the Registration move. (For example, the CW Limit Switch activates during a CW Registration move.) A Limit Switch is activated in the opposite direction of the Registration move is ignored.

NOTE Note that it is possible to *start* a move while a CW or CCW Limit Switch is active as long as the direction of travel is *opposite* that of the activated Limit Switch. For example, it is possible to start a clockwise Registration Move while the CCW Limit Switch is active.

Blend Moves

All of the moves explained so far must be run individually to their completion or be stopped before another move can begin. The NXFE2 gives you the ability to assemble more complex profiles from a series of relative moves that are stored in memory and then run with a single command. Each Blend Move consists of two to sixteen segments.

Each Relative Move defines a *segment* of the Blend Move. The following restrictions apply when programming Blend Moves.

1)Each segment of the Blend Move must be written to the NXFE2 before the move can be initiated.

- The NXFE2 supports Blend Moves with up to sixteen segments.

2)Each segment is programmed as a relative move. Blend Moves cannot be programmed with absolute coordinates.

3)All segments run in the same direction. The sign of the target position is ignored and only the magnitude of the target position is used. The move’s direction is controlled by the bit pattern used to start the move.

4)The Programmed Speed of each segment must be greater than or equal to the Starting Speed.

5)The Programmed Speed cannot be the same between segments. All other parameters can remain the same between segments.

6)For all segments except for the last one, the programmed position defines the end of the segment. For the last segment, the programmed position defines the end of the move.

7)Once you enter a segment, that segment’s programmed acceleration and deceleration values are used to change the speed of the motor.

8)The blend segment must be long enough to achieve the Programmed Speed for the segment. During programming, the NXFE2 will respond with an Invalid Profile error if the segment that is being programmed cannot reach the Programmed Speed. In this case, the segment must be longer, or the acceleration and/or deceleration must be increased.

Blend Moves (continued)

The figure below shows a three segment Blend Move that is run twice. It is first run in the clockwise direction (CW Blend Move), and then in the counter-clockwise direction (CCW Blend Move).

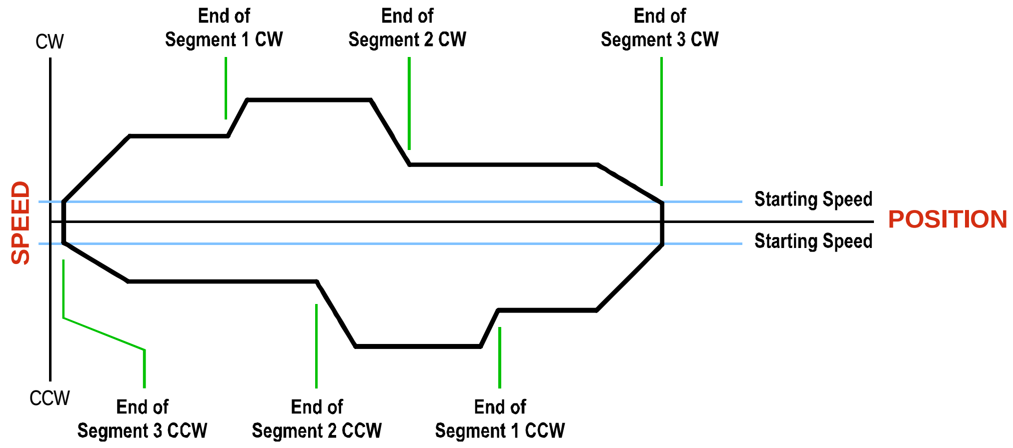


Figure R2.12 Blend Move

NOTE

1) You do not have to preset the position or home the machine before you can use a Blend Move. Because the Blend Move is based on Relative Moves, it can be run from any location.
2) The Blend Move is stored in the internal memory of the NXFE2 and can be run multiple times once it is written to the unit. The Blend Move data stays in memory until power is removed, the unit is sent new Configuration Data, or a new Blend Move is written to the NXFE2.
3) There are two control bits used to specify which direction the Blend Move is run in. This gives you the ability to run the Blend Move in either direction.

Controlled Stops

- The move completes without error.

NOTE The External Input is ignored when running a Blend Move. Issuing a Hold Move command with result in a Command Error and the move will continue.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- The E-Stop Input is activated. The E-Stop Input must be deactivated before a new move command will be accepted.
- A CW/CWW Limit Switch is activated. Only Jog or Registration moves will be accepted while the Limit Switch remains active. The Jog or Registration move must be in the opposite direction of the activated switch. This allows you to move off of the activated Limit Switch.

Encoder Follower

The final form of motion control available with the NXFE2 is Encoder Follower. A quadrature encoder is required but it is not mounted on the motor controlled by the NXFE2. The encoder is typically mounted on a second motor, but it can be mounted anywhere, including on something as simple as a hand crank.

In this mode, the NXFE2 will output motor control pulses in response to changes in the position of the encoder. The NXFE2 does not accept move commands over the backplane while in this mode.

Two commands are available. The *+Encoder Follower Move* will output clockwise pulses when the encoder count is increasing and the *–Encoder Follower Move* will output counter-clockwise pulses when the encoder count is increasing.

Increasing encoder counts is defined as when the encoder’s A signal leads the B signal.

The External Input is ignored when the axis is in Encoder Follower mode.

Multiplier and Divisor Parameters

The NXFE2 has Multiplier and Divisor parameters that allow you to adjust the ratio between the encoder pulses and the motor pulses. The Multiplier and Divisor parameter each have a range of 1 to 32,768. It is possible to enter a ratio greater than one when programming these values. If the ratio is greater than one, the NXFE2 will output multiple motor steps per encoder count.

As with all encoder functions on the NXFE2, X4 decoding is used when determining the encoder count value. This fact must be taken into account when calculating the appropriate Multiplier and Divisor values.

Note that the maximum encoder input frequency is 250KHz, and the maximum motor output frequency is 1 MHz.

It is possible for your application to mathematically exceed this 1 MHz maximum output frequency when the Multiplier/Divisor ratio is greater than one. For example, assume your Multiplier/Divisor ratio is 3, and your encoder input frequency is 125 kHz. Ideally, your motor output frequency would be:

125 kHz X 4{decoding} X 3{ratio} = 1.5 MHz

As this exceeds the 1 MHz limit, the NXFE2 will output pulses at its 1 MHz rate until motor position catches up with the encoder position.

Acceleration Parameters

The Linear Acceleration type is the only one available when using Encoder Follower Moves. Unlike other moves, the Acceleration and Deceleration values can be set to zero with this move type. If they are zero, the axis will immediately match the encoder speed when a move begins. If this stalls the motor, set the Acceleration and Deceleration parameters to their maximum value of 2000 steps/millisecond/second. If a move still causes the motor to stall, then decrease the Acceleration and Deceleration values until the motor runs correctly.

NOTE

1) The NXFE2 will always output the correct number of steps when operating in this mode. However, the use of the Acceleration and Deceleration parameters, especially low values, can cause the motor’s motion to lag behind the encoder’s motion.

2) If the Acceleration and Deceleration parameters are set to zero, electrical noise on the encoder inputs may cause the motor to oscillate back and forth by ±1 step when there is no encoder motion.

Encoder Follower (continued)

Controlled Stops

- The encoder stops moving.
- Both of the Jog Move command bits equal zero.
- Encoder Follower moves cannot be brought to a controlled stop by using the Hold Move control bit in the Network Output Registers or the External Input.

Immediate Stops

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Registers.
- A positive transition on an input configured as an E-Stop Input.
- A CW or CWW Limit Switch is reached.

Notes

REFERENCE 3

CALCULATING MOVE PROFILES

This reference was added for customers that must program very precise profiles. Understanding this section is not necessary before programming the NXFE2 and therefore can be considered optional.

The equations in this reference use a unit of measure of steps/second/second (steps/second^2) for acceleration and deceleration. However, when programming the NXFE2, all acceleration and deceleration values must be programmed in the unit of measure of steps/second/millisecond.

- To convert from steps/second^2 to steps/second/millisecond, divide the value by 1000. This must be done when converting from a value used in the equations to a value programmed into the NXFE2.
- To convert from steps/second/millisecond to steps/second^2, multiply the value by 1000. This must be done when converting from the value programmed into the NXFE2 to the value used in the equations.

Linear Acceleration Equations

When you choose to use constant accelerations, the speed of the move will increase linearly towards the Programmed Speed. This is the fastest form of acceleration, resulting in the fastest move between two points at its programmed speed. For the smoothest transition from the starting speed, the starting speed should be equal to the square root of the acceleration in steps/sec^2. For example, if the choose acceleration is 20,000 steps/sec^2, the smoothest transition occurs when the starting speed is 141. ($141^2 \approx 20{,}000$)

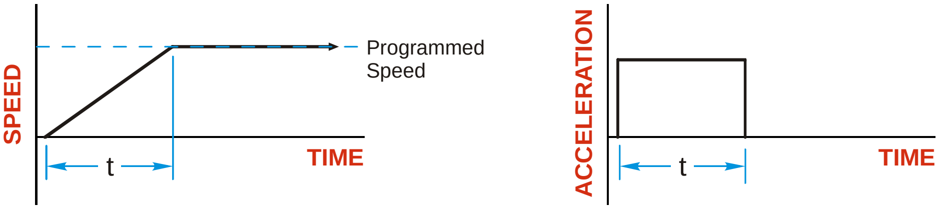
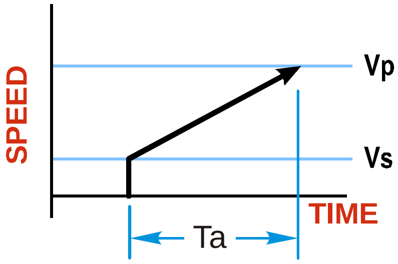


Figure R3.1 Constant Acceleration Curves

Variable Definitions

The following variables are used in these equations:



- V_S = Configured Starting Speed of the move
- V_P = Programmed Speed of the move
- **a** = Acceleration value. Must be in the units of steps/second^2
- **d** = Deceleration value. Must be in the units of steps/second^2
- T_A **or** T_D = Time needed to complete the acceleration or deceleration phase of the move
- D_A **or** D_D = Number of Steps needed to complete the acceleration or deceleration phase of the move

Linear Acceleration Equations (continued)

Table R3.1 gives the equations to calculate acceleration time and distance for a linear acceleration move.

Acceleration Type	T_A or T_D (Time to Accelerate or Decelerate)	D_A or D_D (Distance to Accelerate or Decelerate)	a (Average Acceleration)
Linear	$T_A = (V_P – V_S)/a$ $T_D = (V_P – V_S)/d$	$D_A = T_A*(V_P + V_S)/2$ $D_D = T_D*(V_P + V_S)/2$	$a = (V_P^2 – V_S^2)/2D_A$

Table R3.1 Acceleration Equations

If the sum of the D_A and D_D values of the move is *less than* the total number of steps in the move, your move will have a Trapezoidal profile.

If the sum of the D_A and D_D values of the move is *equal to* the total number of steps in the move, your move will have a Triangular profile and your move will reach the Programmed Speed before it begins to decelerate.

If the sum of the D_A and D_D values of the move is *greater than* the total number of steps in the move, your move will have a Triangular profile and it *will not* reach the Programmed Speed before it begins to decelerate.

As an example, lets assume the values in table R3.2 for a move profile.

Name	Value	NXFE2 Parameter Values
Acceleration (a)	20,000 steps/sec^2	20
Deceleration (d)	25,000 steps/sec^2	25
Starting Speed (V_S)	141 steps/sec	141
Programmed Speed (V_P)	100,000 steps/sec	100,000

Table R3.2 Sample Values

From figure R3.1:

Time to accelerate: $T_A = (V_P – V_S)/a = (100,000 - 141)/20,000 = 4.993$ seconds
Time to decelerate: $T_D = (V_P – V_S)/d = (100,000 - 141)/25,000 = 3.994$ seconds
Distance to Accelerate: $D_A = T_A*(V_P + V_S)/2 = 4.993 * (100,000 + 141)/2 = 250,002$ steps
Distance to Decelerate: $D_D = T_D*(V_P + V_S)/2 = 3.994 * (100,000 + 141)/2 = 199,982$ steps

Total Distance needed to accelerate and decelerate: 250,002 + 199,982 = 449,984 steps

If a move with the above acceleration, deceleration, starting speed, and programmed speed has a length greater than 449,984 steps, the NXFE2 will generate a Trapezoidal profile. If the move is equal to 449,984 steps, the unit will generate a Triangular profile and the it will output one pulse at the programmed speed. If the move is less than 449,984 steps, the unit will generate a Triangular profile and the programmed speed will not be reached.

In the case of a Triangular profile where the programmed speed is not reached, it is fairly easy to calculate the maximum speed (V_M) attained during the move. Because the move is always accelerating or decelerating, the total distance traveled is equal to the sum of D_A and D_D.

$D_A = T_A*(V_M + V_S)/2$ and $T_A = (V_M – V_S)/a$. By substitution:
$D_A = (V_M – V_S)/a * (V_M + V_S)/2 = (V_M^2 – V_S^2)/2a$. By the same method,
$D_D = (V_M^2 – V_S^2)/2d$.

Therefore, total distance traveled equals:

$$D_A + D_D = (V_M^2 – V_S^2)/2a + (V_M^2 – V_S^2)/2d.$$

In the case where the acceleration and deceleration values are equal, this formula reduces to:

$$D_A + D_D = (V_M^2 – V_S^2)/a$$

Linear Acceleration Equations (continued)

Continuing the example from table R3.2, assume a total travel distance of 300,000 steps.

$$
\begin{aligned}
D_A + D_D &= \frac{V_M^2 - V_S^2}{2a} + \frac{V_M^2 - V_S^2}{2d} \\
300{,}000 \text{ steps} &= \frac{V_M^2 - 141^2}{2(20{,}000)} + \frac{V_M^2 - 141^2}{2(25{,}000)} \\
300{,}000 \text{ steps} &= \frac{V_M^2 - 20{,}000}{40{,}000} + \frac{V_M^2 - 20{,}000}{50{,}000} \\
300{,}000 \text{ steps} &= \frac{5}{5}\left(\frac{V_M^2 - 20{,}000}{40{,}000}\right) + \frac{4}{4}\left(\frac{V_M^2 - 20{,}000}{50{,}000}\right) \\
300{,}000 \text{ steps} &= \frac{5V_M^2 - 100{,}000}{200{,}000} + \frac{4V_M^2 - 80{,}000}{200{,}000} \\
300{,}000\ (200{,}000) &= 9V_M^2 - 180{,}000 \\
\frac{60{,}000.18 \times 10^6}{9} &= V_M^2 \\
V_M &= 81{,}650 \text{ steps/sec}
\end{aligned}
$$

Once you have calculated the maximum speed, you can substitute this value into the time and distance formulas in table R3.1 to calculate time spent and distance traveled while accelerating and decelerating.

Total Time Equations

For Trapezoidal Profiles you must first determine the number of counts that you are running at the Programmed Speed. This value, (D_P below), is equal to your D_A and D_D values subtracted from your total travel. You can then calculate your total profile time, (T_P below), from the second equation.

D_P = (Total Number of Steps) – (D_A + D_D)

$T_P = T_A + T_D + D_P/V_P$

For Triangular Profiles, the total time of travel is simply:

$T_P = T_A + T_D$

S-Curve Acceleration Equations

S-Curve accelerations smoothly change the acceleration value applied during the move. This limits mechanical shocks to the system as the load accelerates. Triangular S-Curve accelerations vary the applied acceleration throughout the acceleration phase. Trapezoidal S-Curve accelerations have a period of linear acceleration in the middle of the acceleration phase.

When using variable accelerations, the starting speed does not have to be equal to the square root of the programmed acceleration value for the smoothest transition from stopped to accelerating. S-Curve accelerations provides smooth transitions at the beginning and end of the acceleration phase.

Triangular S-Curve Acceleration

The left half of figure R3.2 shows the speed profile of a move during its acceleration phase. The figure shows the triangular S-curve acceleration in red along with the equivalent constant acceleration in black. The equivalent constant acceleration is equal to the change in speed divided by the time it takes to achieve this change in speed.

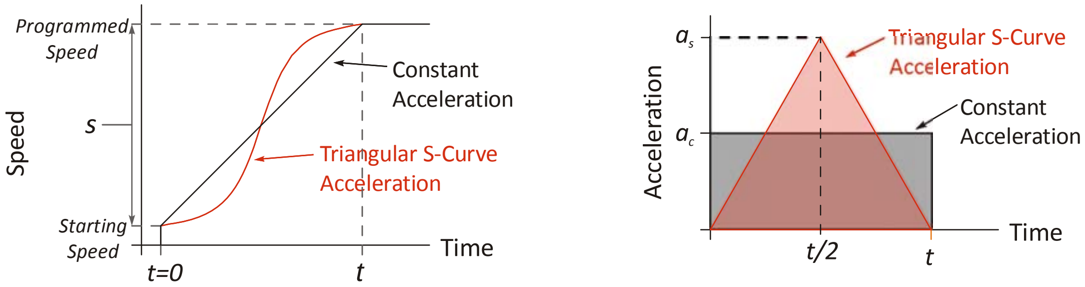


Figure R3.2 Triangular S-Curve Example

Speed is equal to acceleration multiplied by the time it is applied. This is shown graphically in figure R3.2 as the area of the rectangle. In order for the Triangular S-curve acceleration to reach the same speed in the same amount of time, the area of the triangle must equal the area of the square. Area of a triangle is one half of the base length multiplied by the height. Therefore:

$$a_c t = \frac{a_s t}{2} \quad \text{Area of rectangle = Area of triangle}$$

$$a_s = 2a_c$$

This means that a triangular S-curve acceleration profile requires twice the programmed maximum acceleration as a constant acceleration profile to achieve the same speed in the same amount of time.

The above examples assume that you can increase the programmed acceleration value to keep the acceleration time the same. If your constant acceleration value is the maximum your system will allow, then using S-curve accelerations will double the time needed to accelerate to your desired speed.

S-Curve Acceleration Equations (continued)

Trapezoidal S-Curve Acceleration

The left half of figure R3.3 shows the speed profile of a move during its acceleration phase. The figure shows the trapezoidal S-curve acceleration in red. In the NXFE2, the period of constant acceleration is 50% of the acceleration phase. The equivalent constant acceleration is in black. The equivalent constant acceleration is equal to the change in speed divided by the time it takes to achieve the change in speed.

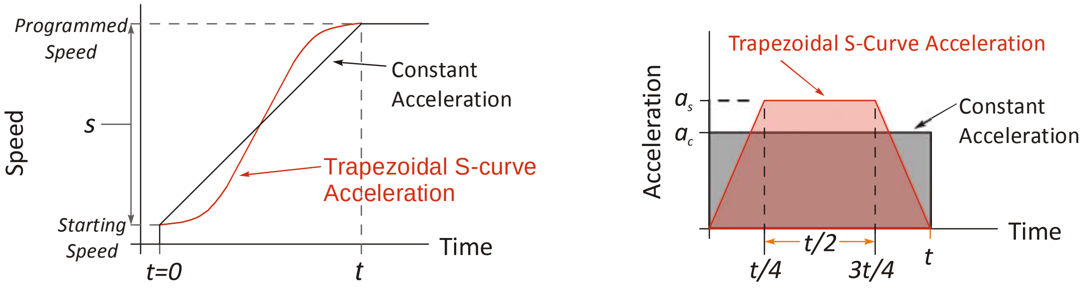


Figure R3.3 Move Profile Example

Speed is equal to acceleration multiplied by the time it is applied. The right side of figure R3.3 shows this graphically as the area of the rectangle. In order for the Trapezoidal S-curve acceleration to reach the same speed in the same amount of time, the area of the polygon must equal the area of the rectangle.

$$\frac{a_s t}{2} + \frac{a_s t}{4} = a_c t \quad \text{Area of polygon = Area of rectangle}$$

$$\frac{2a_s t}{4} + \frac{a_s t}{4} = a_c t$$

$$\frac{3a_s t}{4} = a_c t$$

$$a_s = \frac{4}{3}a_c$$

This means a trapezoidal S-curve acceleration profile requires its programmed acceleration value to be 4/3 that of the constant acceleration value used to achieve the same speed in the same amount of time.

The above examples assume that you can increase the programmed acceleration value to keep the time of the acceleration phase the same. If your constant acceleration value is the maximum your system will allow, then using S-curve accelerations will lengthen the time needed to accelerate to your desired speed by 133.3% (4/3).

Notes

REFERENCE 4

HOMING THE NXFE2

This chapter explains the various ways of homing an axis on the NXFE2. Inputs used to home the module are introduced and diagrams that show how the module responds to a homing command are given.

Definition of Home Position

The Home Position is any position on your machine that you can sense and stop at. Once at the Home Position, the motor position register of the NXFE2 must be set to an appropriate value. If you use the module's *CW/CCW Find Home* commands, the motor position register will be set to zero once the home position is reached. The Encoder Position register will also be reset to zero if the quadrature encoder is enabled.

NOTE ➤ Defining a Home Position is completely optional. Some applications, such as those that use a servo or stepper for speed control, don't require position data at all.

With the exception of Absolute Moves, the NXFE2 can still perform all of its move commands if the Home Position is not defined.

Position Preset

One of the ways to define the Home Position is to issue the Preset Position command to the NXFE2. Before doing this, you will need a way of sensing position outside the NXFE2 module. The machine position data must be brought into the host, the correct preset value calculated, and this value written to the NXFE2 axis with the Position Preset command.

CW/CCW Find Home Commands

The other choice is to use the module's Find Home commands to order the NXFE2 to find the Home Position based on sensors brought into the unit. The CW Find Home command begins searching by outputting CW pulses, and ends when the home sensor triggers while the NXFE2 is outputting CW pulses at the configured Starting Speed. The CCW Find Home command operates in the same way but starts and ends with CCW pulses.

Homing Inputs

Five inputs can be used when homing the module. These inputs are either physical inputs attached to the module or bits in the PLC output data words.

Physical Inputs

- **Home Input:** This input is used in one of two ways: 1) This input is used to define the actual home position of the machine. 2) The input is used as a home proximity input when using the encoder marker pulse to home the machine.
- **Encoder Marker (Z) Pulse:** If you configure the NXFE2 to use an encoder, you have the option of using the encoder's marker pulse to home the machine.
- **CW Limit Switch Input:** This input is used to prevent overtravel in the clockwise direction.
- **CCW Limit Switch Input:** This input is used to prevent overtravel in the counter-clockwise direction.

Backplane Inputs

- **Network Home Proximity Bit:** The NXFE2 can be configured to ignore changes on the physical homing input until the Network Home Proximity Bit makes a 0→1 transition. The NXFE2 will home on the next inactive-to-active change on the physical input once this transition occurs. You must program your host to control the state of this bit. Do not use the Network Home Proximity bit if you only want to home to the Home Limit Switch.

Homing Configurations

The NXFE2 axis must be correctly configured before one of the homing commands will be accepted. One of the following must be part of the module configuration before you can run the homing commands.

1) Enable the Home Input
2) Enable the encoder and connect to the encoder Z-pulse input

NOTE 1) You do not have to configure and use CW or CCW Limits. If you choose to configure the module this way, then the NXFE2 has no way to automatically prevent overtravel during a homing operation. You must prevent overtravel by some external means, or ensure that the homing command is issued in the direction that will result in reaching the homing input directly.

2) When using one of the DC inputs as a Home Input, you can use a bit in the network data as a home proximity input. Using this bit is completely optional.

3) When using an encoder's Z-pulse as the homing sensor, the Home Input will function as a hardware home proximity sensor if it is enabled. Using this feature is completely optional.

Homing Profiles

NOTE The CW Find Home command is used in all of these examples. The CCW Find Home command will generate the same profiles in the opposite direction.

Home Input Only Profile

Figure R4.1 below shows the move profile generated by a CW Find Home command when you use the Home Input without the Network Home Proximity bit.

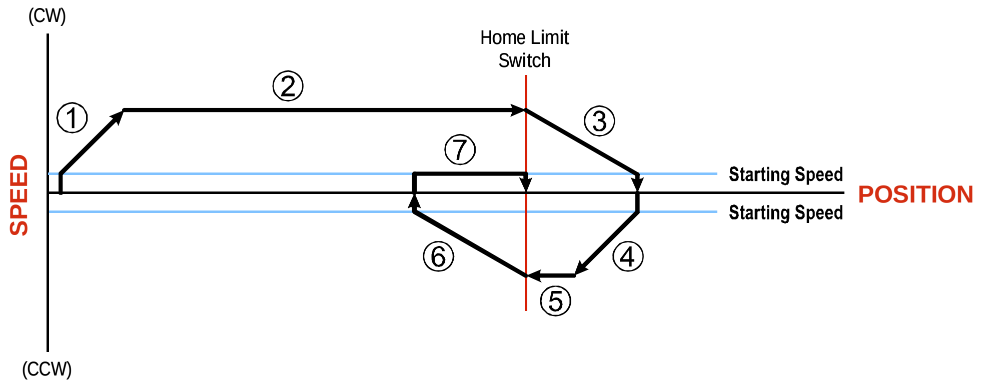


Figure R4.1 Home Input Profile

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed until the Home Input activates
3) Deceleration to the Starting Speed and stop, followed by a two second delay.
4) Acceleration to the Programmed Speed opposite to the requested direction.
5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
6) Deceleration to the Starting Speed and stop, followed by a two second delay.
7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from inactive to active.

NOTE If the Home Input is active when the command is issued, the move profile begins at step 5 above.

Homing Profiles (continued)

Profile with Proximity Input

Figure R4.2 below shows the move profile generated by a CW Find Home command when you use:

- Home Input with Network Home Proximity bit
- Marker Pulse home with Home Input as proximity sensor
- Marker Pulse home with Network Home Proximity bit

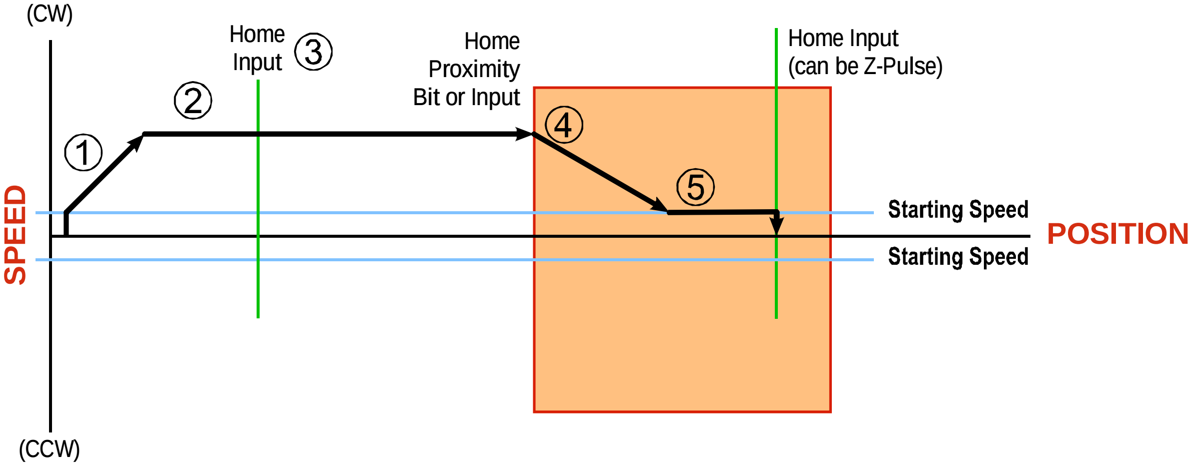


Figure R4.2 Homing with Proximity

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed
3) Ignores homing input, (Home Input or Marker Pulse) because proximity input has not made a 0→1 transition.
4) Deceleration towards the Starting Speed when the proximity input (Backplane bit or Home Input) transitions from its inactive to active state. The axis will stop as soon as the Home Input becomes active.
5) The Starting Speed is the minimum speed the profile will run at. If the axis decelerates to the Starting Speed before reaching the Home Input, it will continue at this speed.

NOTE Figure R4.2 shows the Proximity Input, which is either the Network Home Proximity bit or the Home Input, staying active until the NXFE2 reaches its home position. This is valid, but does not have to occur. As stated in step 4, the NXFE2 starts to hunt for the home position as soon and the Proximity Input makes a 0→1 transition.

Homing Profiles (continued)

Profile with Overtravel Limit

Figure R4.3 below shows the move profile generated by a CW Find Home command when you use:

- CW Overtravel Limit
- Home Input without Network Home Proximity Bit

The profile is generated when you encounter an overtravel limit in the direction of travel. (In this example, hitting the CW limit while traveling in the CW direction.) Hitting the overtravel limit associated with travel in the opposite direction is an Immediate Stop condition. The axis will stop all motion and issue a *Home Invalid* error to your host.

The NXFE2 will stop the axis with an error if both overtravel limits are activated while the unit is trying to find the home position.

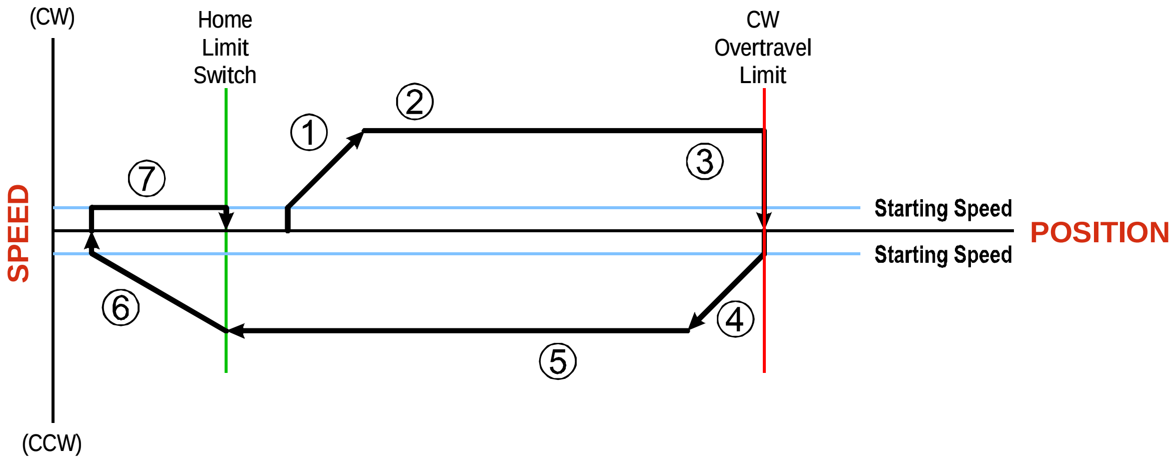


Figure R4.3 Profile with Overtravel Limit

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed
3) Hit CW Limit and immediately stop, followed by a two second delay.
4) Acceleration to the Programmed Speed opposite to the requested direction.
5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
6) Deceleration to the Starting Speed and stop, followed by a two second delay.
7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from inactive to active.

NOTE ➤ If the overtravel limit is active when the Find Home Command is active, the profile will begin at step 4.

REFERENCE 5

CONFIGURATION MODE DATA FORMAT

This chapter covers the formats of the Network Output Data used to configure the NXFE2 as well as the formats of the Network Input Data that contains the responses from the module. Each NXFE2 requires ten 16-bit words (20 bytes) for Output Data as well as ten 16-bit words for Input Data.

Modes of Operation

The NXFE2 has two operating modes, Configuration Mode and Command Mode. You switch between these modes by changing the state of a single bit in the Network Output Data.

Configuration Mode

Configuration Mode gives you the ability to select the proper configuration for your application without having to set any switches. The NXFE2 must be configured on every power up.

Command Mode

This mode gives you the ability to program and execute moves, and reset errors when they occur. The command data formats are described in the following chapter.

Power Up Behavior

The NXFE2 must be configured before move commands can be issued to it. For EtherNet/IP systems that use an EDS file, and for PROFINET systems, configuration data is written to the NXFE2 when it connects to the network. For EtherNet/IP systems that do not use and EDS file, and Modbus TCP systems, the host controller must write configuration data to the NXFE2 on power up. In these systems, the NXFE2 will report a configuration error whenever the module is powered up or it resets.

Data Format

The NXFE2 has multiple parameters that are thirty-two bits in length. These parameters are:

- Starting Speed
- Target Position
- Programmed Speed
- Stopping Distance
- Minimum Registration Move Distance
- Position Preset Value
- Encoder Preset Value

The unit will select the big or small endian format for thirty-two bit values based on the protocol the NXFE2 is configured for. For EtherNet/IP, the default order is *little endian*, (least significant bits first). For PROFINET and Modbus TCP, the default format is big endian, (most significant bits first). In order to support the host controllers that do not use the protocol's default order, the NXFE2 has a parameter named Swap_32_bit_word_order. To use the default format of the protocol, reset the parameter's configuration bit to "0". If your host controller does not use the protocol's default order, set this bit to "1" to reverse the word order.

This bit also controls the format of the thirty-two bit values that can be read from the NXFE2. These values are:

- Motor Position
- Encoder Position
- Captured Encoder Position

Output Data Format

The correct format for the Network Output Data when the NXFE2 is in Configuration Mode is shown below.

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Configuration Data	Range
0	1024	Configuration Word 0	See below
1	1025	Configuration Word 1	See below
2	1026	32 bit Starting Speed	Combined value between 1 and 1,000,000 steps/sec.
3	1027		
4	1028	Homing Timeout	0 to 300 seconds
5	1029	Reserved	0
6	1030	Reserved	0
7	1031	Reserved	0
8	1032	Reserved	0
9	1033	Reserved	0

Table R5.1 Network Output Data Format: Configuration Mode

Starting Speed

The Starting Speed parameter sets the step frequency that all moves begin and end at. Portions of a homing operation are also performed at this speed. This parameter is used to avoid any low frequency resonance points when starting a move. This parameter is also useful in micro-stepping applications when it is not necessary to accelerate from a full stop. The range of the Starting Speed parameter is 1 to 1,000,000 steps per second.

NOTE

1)As a suggestion, a Starting Speed that is equal to the square root of the acceleration rate will create a profile without any jumps in velocity at the beginning of the move profile when using constant linear acceleration. The same holds true for the end of the move profile if the Starting Speed is equal to the square root of the deceleration rate.

2) The Starting Speed can always be programmed in the range of 1 to 1,000,000 steps per second. This 1 MHz output frequency can only be reached by the differential outputs of the NXFE2. The module's 24Vdc open collector outputs are limited to a maximum output frequency of 100,000 steps per second.

Homing Timeout

The Homing Timeout parameter defines how long a homing operation will be allowed to continue before motion stops and the NXFE2 sets its *Home Invalid* status bit. (The *Home Invalid* status bit is available when the axis is in Command Mode.) The Homing Timeout parameter has a range of 0 to 300 seconds. Setting this parameter to a value of 0 disables this feature and the NXFE2 will search for the home marker until it finds it or the command is aborted by the host controller.

Output Data Format (continued)

Configuration Word 0 Format

To Make These Settings...		...Set These Bits To															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Capture Input	Disabled			Bit 13 is Reserved. It must always be set to zero.													0
	Enabled																1
External Input	Disabled															0	
	Enabled															1	
Home Input	Disabled														0		
	Enabled														1		
CW Input	Not Used												0	0			
	CW Limit Switch												0	1			
	E-Stop Input												1	0			
	Do Not Use												1	1			
CCW Input	Not Used										0	0					
	CCW Limit Switch										0	1					
	E-Stop Input										1	0					
	Do Not Use										1	1					
Backplane Home Proximity Operation	Disabled									0							
	Enabled									1							
Quadrature Encoder	Will Not Be Used								0								
	Will Be Used								1								
Diagnostic Feedback	Will Not Be Used							0									
	Will Be Used							1									
Output Pulse Type	CW/CCW						0										
	Step & Direction						1										
Home Operation	To Limit Switch					0											
	To Marker Pulse					1											
Swap_32_bit_ word_order	Use protocol default				0												
	Swap 32 bit words				1												
Card Axis	To Enable		0														
	To Disable		1														
Operation Mode	Command Mode	0															
	Configuration Mode	1															
Write in Your Bit Pattern Here:		**1**		**0**													

Table R5.2 Configuration Bits MSW

NOTE Do not set the Backplane Home Proximity Operation bit if you intend to only home to a Home Limit Switch. This bit should only be set if you need to use the Backplane Home Proximity bit to force the NXFE2 to ignore multiple triggers on the home input.

Bit 12, Swap_32_bit_word_order - The unit will select the big or small endian format for 32 bit values based on the protocol the NXFE2 is configured for. For EtherNet/IP, the default order is *little endian*, (least significant bits first). For PROFINET and Modbus-TCP, the default format is big endian, (most significant bits first). To use the default format of the protocol, reset this bit to "0". If your host controller does not use the protocol's default order, set this bit to "1" to reverse the word order.

Output Data Format (continued)

Configuration Word 1 Format

To Make These Settings...		...Set These Bits To															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Capture Input Active State	Active Low[a] (NC)																0
	Active High[b] (NO)																1
External Input Active State	Active Low (NC)															0	
	Active High (NO)															1	
Home Input Active State	Active Low (NC)														0		
	Active High (NO)														1		
CW (or E-Stop) Input Active State	Active Low (NC)													0			
	Active High (NO)													1			
CCW (or E-Stop) Input Active State	Active Low (NC)												0				
	Active High (NO)												1				
Write in Your Bit Pattern Here:		0	0	0	0	0	0	0	0	0	0	0					

Bits 5 through 15 are Reserved.
They must always be set to zero.

[a] Active Low inputs are considered active when they are not conducting current. (Voltage on the input is between 0Vdc and 2Vdc.)

[b] Active High inputs are considered active when they are conducting current. (Voltage on the input is between 8Vdc and 24Vdc.)

Table R5.3 Configuration Bits LSW

Input Data Format

While in Configuration Mode, the input registers assigned to the NXFE2 will echo back the last configuration data written to the axis. The only exceptions are bits 13 through 15 in Configuration Word 0.

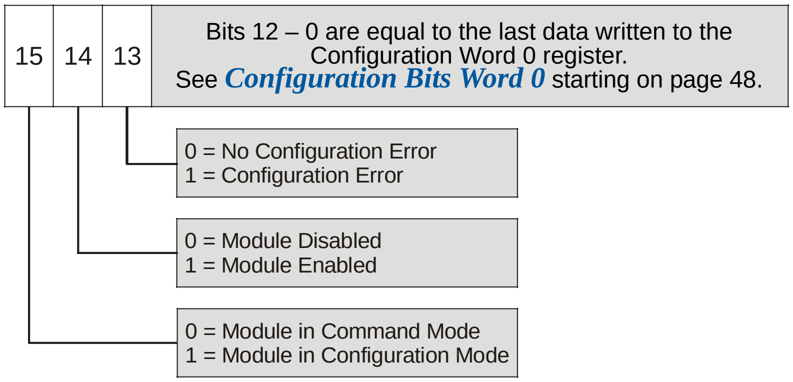
EtherNet/IP or PROFINET Word	Modbus/TCP Register	Description
0	0	Mirror of Configuration Word 0, bits 0 - 12*
1	1	Mirror of Configuration Word 1
2	2	Mirror of 32 bit Starting Speed
3	3	
4	4	Mirror of Homing Timeout
5	5	16#0000
6	6	16#0000
7	7	16#0000
8	8	16#0000
9	9	16#0000

* Bits 13 - 15 are global status bits and are described in the following section.

Table R5.4 Network Input Data Format: Configuration Mode

Global Status Bits

The upper three bits of *Configuration Word 0* are used to report the status of the module as a whole. These bits have same meaning regardless of the mode the module is in. While in Configuration mode, they do not echo back the last values of these bits written to the NXFE2. The figure below shows meaning of each of these status bits.



Bits 12 – 0 are equal to the last data written to the Configuration Word 0 register.
See ***Configuration Bits Word 0*** starting on page 48.

0 = No Configuration Error
1 = Configuration Error

0 = Module Disabled
1 = Module Enabled

0 = Module in Command Mode
1 = Module in Configuration Mode

Figure R5.1 Global Status Bits

Invalid Configurations

The following section lists the configuration settings that will cause a configuration error. Bit 13 of input word 0 is the Configuration Error bit.

In the following tables, all bit places that do not show a value are “don’t care” terms for that error.

Configuration Bits Word 0

If these bits are set as shown....																
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**...a Configuration Error will result because:**
NOTE: This bit is always set when in Configuration Mode		1	1													*ANY* of the reserved bits are set
											1	1				Configuration tried to use an invalid setting for the CW/E-Stop Input.
									1	1						Configuration tried to use an invalid setting for the CCW/E-Stop Input.
					0	1										Configuration tried to enable Diagnostic Feedback while the Output Pulse Type was not set to Step/Dir.
						1	1									Configuration tried to simultaneously enable the Quadrature Encoder and Diagnostic Feedback
				1			0									Configuration tried to request a *Home to Marker Pulse* setting without enabling the Quadrature Encoder.
				1			1	0					0			Configuration tried to request a *Home to Marker Pulse* setting without enabling the Home Input or Backplane Proximity bit. (Marker pulse homing requires one of the two for home proximity sensing.)
				1			1	1					1			Configuration tried to request a *Home to Marker Pulse* setting while enabling both the Home Input and the Backplane Proximity bit. (Marker pulse homing requires one and only one of the two for home proximity sensing.)

Table R5.5 Configuration Bits MSW Errors

Configuration Bits Word 1

If these bits are set as shown....																
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**...a Configuration Error will result because:**
1	1	1	1	1	1	1	1		1	1						*ANY* of the reserved bits are set

Table R5.6 Configuration Bits LSW Errors

Starting Speed Parameter

The Starting Speed parameter must not be outside its range of 1 to 1,000,000. This means that a value of zero *will* cause an error. This parameter is a thirty-two bit value. The format of this value is affected by the *Swap_32_bit_word_order* parameter setting.

Homing Timeout Parameter

This parameter must be in the range of 0 to 300 decimal, 16#0000 to 16#012C in hexadecimal.

Reserved Words

The five reserved words in the configuration programming block must always be zero.

REFERENCE 6

COMMAND MODE DATA FORMAT

This chapter covers the formats of the Network Output Data used to command the NXFE2 as well as the formats of the Network Input Data that contains the responses from the indexer. The NXFE2 requires ten 16-bit words (20 bytes) for Output Data as well as ten 16-bit words for Input Data.

Data Format

The NXFE2 has multiple parameters that are communicated as thirty-two bit integers. These parameters are:

- Starting Speed
- Target Position
- Programmed Speed
- Stopping Distance
- Minimum Registration Move Distance
- Position Preset Value
- Encoder Preset Value

The order of the sixteen bit words in the thirty-two bit values is automatically determined by the communications protocol used by the NXFE2. For EtherNet/IP, the default order is little endian, (least significant word first). For PROFINET and Modbus-TCP, the default format is big endian, (most significant word first). AMCI is aware of some controllers that do not use the protocol's default word format. To accommodate these cases, AMCI has the *Swap_32_bit_word_order* parameter in the configuration data. (See ***Configuration Word 0 Format*** on page 45.) When this bit parameter is reset to zero, the default format for the protocol is used. When this bit parameter is set to one, the opposite format is used.

This parameter also affects the thirty-two bit values that can be read from the NXFE2. These values are:

- Motor Position
- Encoder Position
- Captured Encoder Position

Command Bits Must Transition

Commands are only accepted when the command bit makes a 0→1 transition. The easiest way to do this is to write a value of zero into the Command Word 0 before writing the next command.

This condition also applies when switching from Configuration Mode to Command Mode.

NOTE ➤ If your absolute or relative move only occurs once, it is probably because your code does not reset the command bit to zero before issuing another command. For absolute and relative moves, the command bits can be reset before the move completes without affecting the move. In most cases, the command bits can be reset once an absolute or relative move begins and the NXFE2 will be ready to accept the next move command once the present one completes. The NXFE2 has three status bits, *Moving_CW*, *Moving_CCW*, and *Stopped*, that can be used to determine when a move has begun, and when it has completed. These bits are described later in this chapter, in the ***Input Data Format*** section starting on page 67.

Output Data Format

The following table shows the format of the output network data words when writing command data to the NXFE2.

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function
0	1024	Command Word 0
1	1025	Command Word 1
2	1026	Command Parameters *Word meaning depends on the command set to the NXFE2*
3	1027	
4	1028	
5	1029	
6	1030	
7	1031	
8	1032	
9	1033	

Table R6.1 Command Mode Data Format

Command Word 0

The Command Word 0 is used to control the motion of the axis. Consequently, only one bit in the word can be on at a time. AMCI strongly suggests that you use word length move or copy instructions when setting this word instead of bit instructions to set and reset the individual bits.

Command Word Transition (hex)	# of bit set	Description
16#0000	None	This word must equal zero before commanding any move.
16#0000 → 16#0001	0	Perform an ABSOLUTE MOVE
16#0000 → 16#0002	1	Perform an RELATIVE MOVE
16#0000 → 16#0004	2	Perform a HOLD MOVE operation
16#0000 → 16#0008	3	Perform a RESUME MOVE operation
16#0000 → 16#0010	4	Perform an IMMEDIATE STOP operation
16#0000 → 16#0020	5	Perform a CW FIND HOME operation
16#0000 → 16#0040	6	Perform a CCW FIND HOME operation
16#0000 → 16#0080	7	Perform one of the following based on Command Parameter data: 1) CW JOG MOVE 2) CW REGISTRATION MOVE 3) CW ENCODER FOLLOWER MOVE
16#0000 → 16#0100	8	Perform one of the following based on Command Parameter data: 1) CCW JOG MOVE 2) CCW REGISTRATION MOVE 3) CCW ENCODER FOLLOWER MOVE

(continued on the next page)

Output Data Format (continued)

Command Word 0 (continued)

Command Word Transition (hex)	# of bit set	Description
16#0000 → 16#0200	9	PRESET the current position. This command can also be used to reset the *Move Complete* status bit.
16#0000 → 16#0400	10	PRESET ENCODER current position
16#0000 → 16#0800	11	RESET COMMAND ERRORS
16#0000 → 16#1000	12	Run CW BLEND MOVE PROFILE
16#0000 → 16#2000	13	Run CCW BLEND MOVE PROFILE
16#0000 → 16#4000	14	Set Minimum Registration Move Distance

Table R6.2 Command Word 0

The remaining bit of this word is bit 15.

- **Bit 15: MODE CONTROL Bit:** A ‘0’ places the NXFE2 in Command Mode.
  A ‘1’ places the NXFE2 in Configuration Mode.

Command Word 1

To Make These Settings...		...Set These Bits To															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acceleration Type	Constant															0	0
	Constant															1	0
	Triangular S															0	1
	Trapezoidal S															1	1
No_Fault Output when Set_State = “1”	Off														0		
	On														1		
On_Line Output when Set_State = “1”	Off													0			
	On													1			
Move_Complete when Set_State = “1”	Off												0				
	On												1				
Set_State	Off											0					
	On											1					
Backplane Home Proximity	Not Near Home										0						
	Near Home										1						
Blend Move Transmit	Accept New Block									0→1							
	Acknowledge OK’d									1→0							
Encoder Follower Mode	Disabled								0								
	Enabled								1								
Reserved Bits			0	0	0	0	0	0									
Program Blend Move	Disabled	0															
	Enabled	1															

Table R6.3 Command Bits LSW

Command Blocks

The following section lists the output data format for the fourteen different commands.

Absolute Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0001
1	1025	***Command Word 1***		See pg. 51
2	1026	32 bit Abs. Target Position	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	1027			
4	1028	32 bit Programmed Speed	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.4 Absolute Move Command Block

Relative Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0002
1	1025	***Command Word 1***		See pg. 51
2	1026	32 bit Rel. Target Position	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	1027			
4	1028	32 bit Programmed Speed	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.5 Relative Move Command Block

Command Blocks (continued)

Hold Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0004
1	1025	***Command Word 1***		See pg. 51
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table R6.6 Hold Move Command Block

Unused words are ignored by the NXFE2 and can be any value, including parameter values from the previous command.

Resume Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0008
1	1025	***Command Word 1***		See pg. 51
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	32 bit Programmed Speed	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table R6.7 Resume Move Command Block

Unused words are ignored by the NXFE2 and can be any value, including parameter values from the previous command. This is typically the case when resuming a move, the words are listed as “Unused” to highlight that the target position of a held move cannot be changed when the move is resumed.

Command Blocks (continued)

Immediate Stop

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0010
1	1025	***Command Word 1***		See pg. 51
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table R6.8 Immediate Stop Command Block

Unused words are ignored by the NXFE2 and can be any value, including parameter values from the previous command.

CW Find Home

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0020
1	1025	***Command Word 1***		See pg. 51
2	1026	*Reserved*		*Must be 16#0000*
3	1027	*Reserved*		*Must be 16#0000*
4	1028	32 bit Programmed Speed	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.9 Find Home CW Command Block

Only Constant Acceleration is available when homing an NXFE2 axis. Bit 0 of the Command Word 0 must equal zero.

Command Blocks (continued)

CCW Find Home

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0040
1	1025	***Command Word 1***		See pg. 51
2	1026	*Reserved*		*Must be 16#0000*
3	1027	*Reserved*		*Must be 16#0000*
4	1028	32 bit Programmed Speed	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.10 Find Home CCW Command Block

Only Constant Acceleration is available when homing an NXFE2 axis. Bit 0 of the Command Word 0 must equal zero.

CW Jog

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0080
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal “0”
2	1026	*Reserved*		*Must be 16#0000*
3	1027	*Reserved*		*Must be 16#0000*
4	1028	32 bit Programmed Speed	Steps/Second	1 to 1,000,000 (1 to 16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.11 Jog CW Command Block

Command Blocks (continued)

CW Registration Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0080
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal “0”
2	1026	32 bit Stopping Distance	Steps	1 to +1,073,741,823 (1 to 16#3FFF:FFFF)
3	1027			
4	1028	32 bit Programmed Speed	Steps/second	1 to 1,000,000 (1 to 16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.12 Registration Move CW Command Block

+Encoder Follower Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0080
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal “1”
2	1026	Encoder Count Multiplier		1 to 32,767
3	1027	Encoder Count Divisor		1 to 32,767
4	1028	*Reserved*		*Must be 16#0000*
5	1029	*Reserved*		*Must be 16#0000*
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.13 +Encoder Follower Command Block

Command Blocks (continued)

CCW Jog

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0100
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal "0"
2	1026	*Reserved*		*Must be 16#0000*
3	1027	*Reserved*		*Must be 16#0000*
4	1028	32 bit Programmed Speed	Steps/Second	1 to 1,000,000 (1 to 16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.14 Jog CCW Command Block

CCW Registration Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0100
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal "0"
2	1026	32 bit Stopping Distance	Steps	1 to +1,073,741,823 (1 to 16#3FFF:FFFF)
3	1027			
4	1028	32 bit Programmed Speed	Steps/second	1 to 1,000,000 (1 to 16#000F:4240)
5	1029			
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.15 Registration Move CCW Command Block

Command Blocks (continued)

–Encoder Follower Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0080 or 16#0100
1	1025	***Command Word 1***		See pg. 51 Bit 8 must equal "1"
2	1026	Encoder Count Multiplier		1 to 32,767
3	1027	Encoder Count Divisor		1 to 32,767
4	1028	*Reserved*		*Must be 16#0000*
5	1029	*Reserved*		*Must be 16#0000*
6	1030	Acceleration	Steps/sec/ms	1 to 2000
7	1031	Deceleration	Steps/sec/ms	1 to 2000
8	1032	*Reserved*		*Must be 16#0000*
9	1033	*Reserved*		*Must be 16#0000*

Table R6.16 –Encoder Follower Command Block

Preset Motor Position

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0200
1	1025	***Command Word 1***		See pg. 51
2	1026	32 bit Position Preset Value	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	1027			
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.17 Preset Position Command Block

➤ Presetting the position will also reset the *Move Complete* status bit.

Command Blocks (continued)

Preset Encoder Position

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	*Command Word 0*		16#0400
1	1025	*Command Word 1*		See pg. 51
2	1026	32 bit Encoder Preset Value	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	1027			
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.18 Preset Position Command Block

Reset Errors

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	*Command Word 0*		16#0800
1	1025	*Command Word 1*		16#0000
2	1026	*Reserved*		*16#0000*
3	1027	*Reserved*		*16#0000*
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.19 Reset Errors Command Block

Resetting errors will not reset the *Move_Complete*, *Position_Invalid*, or *Configuration_Error* status bits.

Command Blocks (continued)

Run CW Blend Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#1000
1	1025	***Command Word 1***		16#0000
2	1026	*Reserved*		*16#0000*
3	1027	*Reserved*		*16#0000*
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.20 Run CW Blend Move Command Block

Run CCW Blend Move

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#2000
1	1025	***Command Word 1***		16#0000
2	1026	*Reserved*		*16#0000*
3	1027	*Reserved*		*16#0000*
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.21 Run CCW Blend Move Command Block

Command Blocks (continued)

Set Min. Registration Move Distance

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#4000
1	1025	***Command Word 1***		16#0000
2	1026	32 bit Encoder Preset Value	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	1027			
4	1028	*Reserved*		*16#0000*
5	1029	*Reserved*		*16#0000*
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.22 Set Minimum Registration Move Distance Command Block

Programming Blend Move Profiles

The previous ***Run CW Blend Move*** and ***Run CCW Blend Move*** sections show the word values needed to perform a Blend Move. These sections are on page 60. This section covers how to program a Blend Move Profile. A Blend Move Profile must be programmed into the NXFE2 before a Blend Move can be executed.

Multiple writes to the NXFE2 are required to program a single Blend Move Profile. The NXFE2 uses a simple handshaking protocol to control this transfer of data. Two bits in the Output Network Data and two bits in the Input Network Data are used in the transfer of data blocks.

The two bits in the Output Network Data were introduced in the ***Command Word 1*** section on page 51.

- **Program_Blend_Profile** (Bit 15 of Command Word 1) Must make a 0→1 transition to start the Blend Move Profile programming cycle and must remain a “1” while programming the Blend Move Profile.
- **Program_Blend_Transmit** (Bit 7 of Command Word 1) Make a 0→1 transition on this bit to tell the NXFE2 that new Blend Move Profile data is available. Make a 1→0 transition on this bit after the NXFE2 sets its *Program Blend Profile Acknowledge* bit in its Input Network Data data to complete the transfer of the block of data.

The two bits in the Input Network Data are introduced here. The remaining values in the Input Network Data are introduced in the ***Input Data Format*** section starting on page 67.

- **Blend_Move_ACK** (Bit 9 of Status Word 1) This bit is used by the NXFE2 to tell the host controller when it has accepted a new data block, and when it is ready to receive the next data block.
- **Profile_Invalid** (Bit 9 of Status Word 0) This bit is used by the NXFE2 to tell the host controller when there was an error in the last Program Blend Profile data block sent to the module.

The following logical steps are used to program a Blend Move Profile.

1) Your host controller writes the *First Block* to the NXFE2 with the *Program Blend Profile* and *Program Blend Profile Transmit* bits both set to 1. The format of the First Block is determined by the network protocol and the value of the Swap_32_bit_word_order parameter. The format of the First Block is described in the following section.
2) The NXFE2 responds by storing the information in the data block, setting the *Invalid Blend Profile* bit if there is an error in the data, and setting the *Program Blend Profile Acknowledge* bit.
3) Your host controller must detect when the *Program Blend Profile Acknowledge* bit is set. When this bit transitions, check the value of the *Invalid Blend Profile* bit and act appropriately.
4) Your host controller then must reset the *Program Blend Profile Transmit* bit.
5) The NXFE2 responds by resetting its *Program Blend Profile Acknowledge* bit.
6) a: If you have additional data blocks that must be sent to the NXFE2, repeat these instructions from step 1, sending the next Segment Programming Block.
   b: If you have sent the last data block, reset the *Program Blend Profile* bit to end the programming cycle.

NOTE Two sample programs for the NXFE2 are available on the ***http://www.amci.com*** website. One of them includes the sample code needed to program a Blend Move Profile.

Two different types of data blocks must be used when programming a Blend Move Profile. The first data block type programs the Number of Segments and the acceleration/deceleration types used between the segments. (Constant, Triangular S-Curve, Trapezoidal S-Curve) This must be the first data block written to the NXFE2 when programming a Blend Move Profile. The second data block type is used to program the Segment Length, the Target Speed, and the actual acceleration/deceleration values. The second data block can be used multiple times in the programming cycle, once for each segment in the profile.

Blend Move Programming Blocks

The following blocks are used to program a Blend Move.

EtherNet/IP Protocol

Default First Block

This is the block format when the Swap_32_bit_word_order bit in the configuration data equals zero. (Its default value.)

EtherNet/IP Word	Function	Units	Range
0	***Command Word 0***		16#0000
1	***Command Word 1***		16#8080
2	Acceleration Type MSW	Programs Accel. Type for Segments 9 - 16	
3	Number of Segments		2 to 16
4	Final Deceleration Value and Type	Programs final Deceleration Value and Type used at the end of the profile.	
5	Acceleration Type LSW	Programs Accel. Type for Segments 1 - 8	
6	*Reserved*		*16#0000*
7	*Reserved*		*16#0000*
8	*Reserved*		*16#0000*
9	*Reserved*		*16#0000*

Table R6.23 Default Blend Move First Programming Block - EtherNet/IP

Swapped First Block

This is the block format when the Swap_32_bit_word_order bit in the configuration data equals one.

EtherNet/IP Word	Function	Units	Range
0	***Command Word 0***		16#0000
1	***Command Word 1***		16#8080
2	Number of Segments		2 to 16
3	Acceleration Type MSW	Programs Accel. Type for Segments 9 - 16	
4	Acceleration Type LSW	Programs Accel. Type for Segments 1 - 8	
5	Final Deceleration Value and Type	Programs final Deceleration Value and Type used at the end of the profile.	
6	*Reserved*		*16#0000*
7	*Reserved*		*16#0000*
8	*Reserved*		*16#0000*
9	*Reserved*		*16#0000*

Table R6.24 Swapped Blend Move First Programming Block - EtherNet/IP

Blend Move Programming Blocks (continued)

EtherNet/IP Protocol (continued)

Segment Programming Block

EtherNet/IP Word	Function	Units	Range
0	*Command Word 0*		16#0000
1	*Command Word 1*		16#8080
2	32-bit Segment Length	Counts	1 to (1,073,741,823 - 1)[a b] 1 to 16#3FFF:FFFE
3			
4	32-bit Target Speed	Counts/sec	1 to 1,000,000 1 to 16#000F:4240
5			
6	Accel/Decel Value	Counts/sec/ms	1 to 2000
7	Reserved		Must be 0
8	Reserved		Must be 0
9	Reserved		Must be 0

a. Combined length of all segments cannot exceed 1,073,741,823 counts

b. The Segment Length must be long enough to achieve the Programmed Speed at the programmed Accel/Decel Value.

Table R6.25 Blend Move Segment Programming Block - EtherNet/IP

PROFINET and Modbus TCP Protocols

Default First Block

This is the block format when the Swap_32_bit_word_order bit in the configuration data equals zero. (Its default value.)

PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	*Command Word 0*		16#0000
1	1025	*Command Word 1*		16#8080
2	1026	Number of Segments		2 to 16
3	1027	Acceleration Type MSW	Programs Accel. Type for Segments 9 - 16	
4	1028	Acceleration Type LSW	Programs Accel. Type for Segments 1 - 8	
5	1029	Final Deceleration Value and Type	Programs final Deceleration Value and Type used at the end of the profile.	
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.26 Default Blend Move First Programming Block - PROFINET and Modbus TCP

Blend Move Programming Blocks (continued)

PROFINET and Modbus TCP Protocols (continued)

Swapped First Block

This is the block format when the Swap_32_bit_word_order bit in the configuration data equals one.

PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0000
1	1025	***Command Word 1***		16#8080
2	1026	Acceleration Type MSW	Programs Accel. Type for Segments 9 - 16	
3	1027	Number of Segments		2 to 16
4	1028	Final Deceleration Value and Type	Programs final Deceleration Value and Type used at the end of the profile.	
5	1029	Acceleration Type LSW	Programs Accel. Type for Segments 1 - 8	
6	1030	*Reserved*		*16#0000*
7	1031	*Reserved*		*16#0000*
8	1032	*Reserved*		*16#0000*
9	1033	*Reserved*		*16#0000*

Table R6.27 Swapped Blend Move First Programming Block - PROFINET and Modbus TCP

Segment Programming Block

EtherNet/IP Word	Modbus/TCP Register	Function	Units	Range
0	1024	***Command Word 0***		16#0000
1	1025	***Command Word 1***		16#8080
2	1026	32-bit Segment Length	Counts	1 to (1,073,741,823 - 1)[a] [b] 1 to 16#3FFF:FFFE
3	1027			
4	1028	32-bit Target Speed	Counts/sec	1 to 1,000,000 1 to 16#000F:4240
5	1029			
6	1030	Accel/Decel Value	Counts/ms/sec	1 to 2000
7	1031	Reserved		Must be 0
8	1032	Reserved		Must be 0
9	1033	Reserved		Must be 0

[a] Combined length of all segments cannot exceed 1,073,741,823 counts

[b] The Segment Length must be long enough to achieve the Programmed Speed at the programmed Accel/Decel Value.

Table R6.28 Blend Move Segment Programming Block - PROFINET and Modbus TCP

Blend Move Programming Blocks (continued)

Programming Block Bit Values

To Make These Settings...		...Set These Bits To															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acceleration Type Segment 1 or 9	Constant															0	0
	Triangular S															0	1
	Trapezoidal S															1	1
Segment 2 or 10. Same values as segment 1														▨	▨		
Segment 3 or 11. Same values as segment 1												▨	▨				
Segment 4 or 12. Same values as segment 1										▨	▨						
Segment 5 or 13. Same values as segment 1								▨	▨								
Segment 6 or 14. Same values as segment 1						▨	▨										
Segment 7 or 15. Same values as segment 1				▨	▨												
Segment 8 or 16. Same values as segment 1		▨	▨														

Table R6.29 Acceleration Type Settings

To Make These Settings...		...Set These Bits To															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Deceleration Value and Type: Final Segment	Constant	0	0	0	0	0	Deceleration Value 1 to 2000										
	Triangular S	0	1	0	0	0											
	Trapezoidal S	1	1	0	0	0											

Table R6.30 Final Segment Deceleration Settings

Input Data Format

The correct format for the Network Input Data when the NXFE2 is in Command Mode is shown below. The EtherNet/IP and Modbus/TCP addresses are shown.

EtherNet/IP or PROFINET Word	Modbus/TCP Register	Function	Units	Range
0	0	Status Bits Word 0		See Description below
1	1	Status Bits Word 1		See Description Below
2	2	Current Position	Steps	–2,147,483,648 to 2,147,483,647 (32 bit signed value)
3	3			
4	4	Current Encoder Position / Diagnostic Feedback Value	Counts	–2,147,483,648 to 2,147,483,647 (32 bit signed value)
5	5			
6	6	Capture Data	Steps or Counts	–2,147,483,648 to 2,147,483,647 (32 bit signed value)
7	7			
8	8	Reserved		16#0000
9	9	Network Connectivity		See Description below

Table R6.31 Network Input Data Format: Command Mode

Status Bits Word 0

Status Bits Word 0

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MODE	ModOK	ConErr	CmdErr	InErr	PosInvld	ProInvld	HmInvld	MvCmp	Decel	Accel	Homed	Stopped	Hold	MvCCW	MvCW

Figure R6.1 Status Bits MSW

BIT 0: **CW Motion:** Set when the NXFE2 axis is outputting pulses for clockwise motion.

BIT 1: **CCW Motion:** Set when the NXFE2 axis is outputting pulses for counter-clockwise motion.

BIT 2: **Hold State:** Set when the NXFE2 axis has stopped motion as a result of a Hold Move Command. See page 53 for information on the ***Hold Move*** Command.

BIT 3: **Axis Stopped:** Set when the NXFE2 axis is not in motion for any reason.

BIT 4: **At Home:** This bit is only set after the successful completion of a homing command. The next command that causes motion will reset this bit. Any move off of the home position followed by any move back on to the home position will not result in this bit being set again. The Home Input State bit, found in the Status Bits LSW will always turn on when the Home Input is active.

BIT 5: **Move Accelerating:** Set when the NXFE2 axis is accelerating during any move.

BIT 6: **Move Decelerating:** Set when the NXFE2 axis is decelerating during any move.

BIT 7: **Move Complete:** Set when the NXFE2 axis has successfully completed an Absolute, Relative, or Blend Move. This bit is reset by the next move command or by a Preset Position command.

BIT 8: **Home Invalid Error:** Set when the NXFE2 could not home the axis because of an error in a ±Find Home command data, the homing operation exceeded its programmed timeout value, an E-Stop input activates, or the limit switch that is opposite to the current direction of travel activates.

Input Data Format (continued)

Status Bits Word 0 (continued)

BIT 9: Profile Invalid: Set when there was an error in the last Program Blend Profile data block written to the NXFE2. See the ***Blend Move Programming Blocks*** section starting on page 63 for information on the format of the data blocks.

BIT 10: Position Invalid: Set under the following conditions:

1) Module switched from Configuration Mode to Command Mode
2) An Immediate Stop command was issued while motion was occurring
3) An Emergency Stop input was activated while motion was occurring.
4) If either of the CW or CCW Limit inputs become active during a Relative, Absolute, or Blend Move.

BIT 11: Input Error:
±Find Home Commands: Set when an E-Stop input activates during homing operation or when the limit switch that is opposite to the direction of travel activates, for example when the CCW Limit activates during a +Find Home command.
±Jog Move Commands: Set when an E-Stop input activates during the move or when the limit switch that is in the direction of travel activates, for example when the CW Limit activates during a +Jog Move command.
±Registration Move Commands: Set when an E-Stop input activates during the move or when the limit switch that is in the direction of travel activates, for example when the CW Limit activates during a +Registration Move command.
All Other Move Commands: Set when an E-Stop input activates during the move or when either of the end limit switches activates.
Note that the Input Error bit is latched but you do not have to issue a Clear Errors command block to clear the bit. The next move command will clear this bit automatically if none of the inputs are active when the command is issued.

BIT 12: Command Error: Set when the last command issued to the NXFE2 forced an error. This bit must be reset by issuing a ***Reset Errors*** command. Other commands, including the ***Immediate Stop*** command, will not be accepted until this error is reset.

BIT 13: Configuration Error: Set to “1” when the axis has a configuration error.

BIT 14: Module OK: Set to “1” when the axis is enabled. Reset to “0” when the axis is disabled. The axis is automatically enabled on power up and can only be disabled by writing the correct configuration data to the unit.

BIT 15: Operation Mode: Set to “1” when the module is in Configuration Mode. Reset to “0” when the module is in Command Mode.

Input Data Format (continued)

Status Bits Word 1

Status Bits Word 1

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
0	0	EFMode	HmProx	0	Non0RD	BMvAck	BlndMv	MvCmpOut	OnLineOut	NoFltOut	CCWIn	CWIn	HomeIn	ExtIn	CapIn

Figure R6.2 Status Bits Word 1

BIT 0: Capture Input State: Set when the Capture Input is active. Its active state is set in the ***Configuration Word 1 Format*** when the module is configured. See page 46 for additional information.

BIT 1: External Input State: Set when the External Input is active. Its active state is set in the ***Configuration Word 1 Format*** when the module is configured. See page 46 for additional information.

BIT 2: Home Input State: Set when the Home Input is active. Its active state is set in the ***Configuration Word 1 Format*** when the module is configured. See page 46 for additional information.

BIT 3: CW Limit/Emergency Stop Input State: Set when the CW Input/E-Stop Input is active. The function of this input is set in the ***Configuration Word 0 Format*** when the module is configured and its active state is set in the ***Configuration Word 1 Format***. See pages 45 and 46 for additional information.

BIT 4: CCW Limit/Emergency Stop Input State: Set when the CCW Input/E-Stop Input is active. The function of this input is set in the ***Configuration Word 0 Format*** when the module is configured and its active state is set in the ***Configuration Word 1 Format***. See pages 45 and 46 for more information.

BIT 5: No_Fault Output State: This bit shows the state of the No_Fault output for the module. This bit must be set to its active state ("1") by the host controller. Once set, this bit will be reset by the following conditions:

- Input Error during a move. (E-Stop, CW Limit, or CCW Limit Input activated during move.)
- Axis switched from Command Mode to Configuration Mode
- Loss of network communications
- Reset by host controller.

BIT 6: On_Line Output State: This bit shows the state of the On_Line output. The On_Line bit must be set to its active state ("1") by the host controller. Once set, this bit will be reset by the following conditions:

- Loss of network communications
- Reset by host controller.

BIT 7: Move_Complete Output State: This bit shows the state of the Move Complete output. This bit will be set by the following conditions:

- Absolute, Relative, or Blend Move completes without error.
- Set by host controller.

Once set, this bit must be reset to its inactive state ("0") by the host controller.

BIT 8: Running Blend Move: Set when the axis is presently running a Blend Move Profile. This bit is reset when the move completes.

BIT 9: Blend Move Ack Bit: Set when the NXFE2 has accepted a Blend Move Profile programming block. See ***Programming Blend Move Profiles***, which starts on page 62, for additional information.

BIT 10: Non-Zero Minimum Registration Distance: Set when the Minimum Registration Move Distance parameter is programmed to a non-zero value.

Input Data Format (continued)

Status Bits LSW (continued)

BIT 11: Reserved: This bit will always equal zero.

BIT 12: Backplane Home Proximity Bit State: This bit always equals the state of the Backplane Home Proximity bit, which is bit 6 in the ***Command Word 1***.

BIT 13: Encoder Follower Mode: Set when the module is in Encoder Follower Mode.

BITS 14 & 15: Reserved: These bits will always equal zero.

Network Connectivity, Word 9

Network Connectivity Bits

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
0	HeartBeat	NetErr	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure R6.3 Network Connectivity Bits

BITS 0 - 12: Reserved: These bits will always equal zero.

BIT 13: Unacknowledged Network Disconnect: This bit is set to "1" if the module loses its network connection. On systems that use the EtherNet/IP protocol, downloading a new program to the PLC may also cause this bit to be set. This bit is set until a Reset Errors command is sent to the module.

BIT 14: Heartbeat Bit: This bit changes state every 500 milliseconds and can be used to verify active network communications.

BIT 15: Reserved: This bit will always equal zero.

Current Position

These two words form a thirty-two bit value. The format of the thirty-two bit value is controlled by the protocol that the NXFE2 is configured for and the value of the Swap_32_bit_word_order parameter. This value can be thought of as the motor position and it is affected by move commands to the axis. This value can be preset using the Preset Position command.

Encoder Position/Diagnostic Feedback Value

These two words form a thirty-two bit value. The format of the thirty-two bit value is controlled by the protocol that the NXFE2 is configured for and the value of the Swap_32_bit_word_order parameter. This value can be thought of as the encoder position and it changes in response to pulses on the encoder inputs. This value can be preset using the Preset Encoder Position command.

When running in Diagnostic Feedback Mode, a Preset Position command will preset the motor's Current Position register pair as well as this register pair. (Both register pairs will have the same value after a Position Preset command.) The Preset Encoder Position command will cause a Command Error while in this Diagnostic Feedback Mode.

Capture Data

These two words form a thirty-two bit value. The format of the thirty-two bit value is controlled by the protocol that the NXFE2 is configured for and the value of the Swap_32_bit_word_order parameter. This value is updated when the Capture Input makes an inactive-to-active transition. (The active state of the Capture Input is configurable in the ***Configuration Bits Word 1***, as described on page 46.) If the module has not been configured to use an encoder, then the Capture Data value will be taken from the Current Position value. If the module has been configured to use an encoder, then the Capture Data value will be taken from the Encoder Position value. The NXFE2 is configured to use, or not use, an encoder by the state of bit 8 as described in the ***Configuration Word 0 Format*** section on page 45.

REFERENCE 7

CONFIGURING NETWORK INTERFACES

This section lists suggestions for configuring the network interfaces on your computer or laptop before attaching to the NXFE2.

Firewall Settings

Firewalls are hardware devices or software that prevent unwanted network connections from occurring. Firewall software has been present on Windows based computers since XP, and it may prevent your computer from communicating with the NXFE2. The internal webserver uses port 80, which is the default http port, and should work without changing any firewall settings. Configuring your firewall to allow communication with the NXFE2 is beyond the scope of this manual.

Disable All Unused Network Interfaces

Routing and default gateway setting on your computer might prevent connection to the NXFE2. When using the Net Configurator utility, broadcast packets that are used to find the NXFE2 often go out the wrong port. The easiest way to avoid this problem is to temporarily disable all network interfaces that are not attached to the NXFE2.

This includes all wireless interfaces as well as all Bluetooth interfaces.

Configure Your Network Interface

Before you can communicate with the NXFE2, your network interface must be on the same subnet as the module.

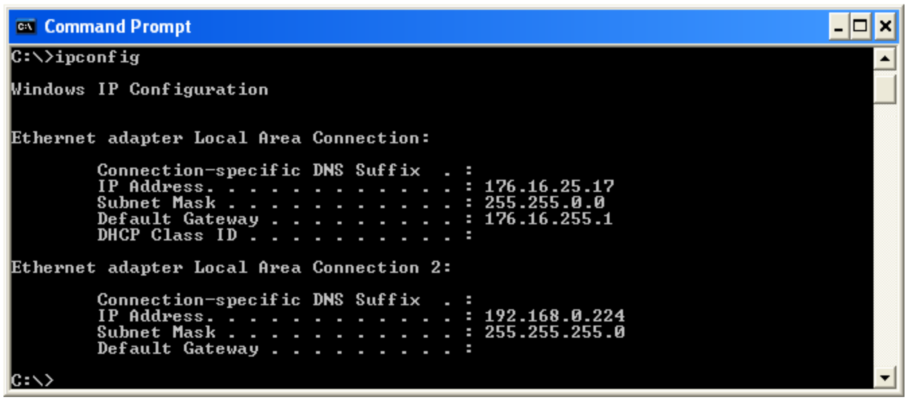
NOTE ➔ The rest of this procedure assumes you are using the 192.168.0.xxx subnet. If you are not, you will have to adjust the given network addresses accordingly.

The easiest way to check the current settings for your NIC is with the ‘ipconfig’ command.

- For Windows 7, click on the [Start] button, and type “cmd” in the “*Search programs and files*” text box. Press [Enter] on the keyboard.
- For Windows 10, press the [Win+X] keys and select “Command Prompt” from the resulting popup. There is no need to run the command prompt as the administrator, so do not select “Command Prompt (Admin)”.

Configure Your Network Interface (continued)

A DOS like terminal will open. Type in ‘ipconfig’, press [Enter] on the keyboard and the computer will return the present Address, Subnet Mask, and Default Gateway for all of your network interfaces. If your present address is 192.168.0.xxx, where ‘xxx’ does not equal 50, and your subnet mask is 255.255.255.0, then you are ready to configure your NXFE2 module. Figure R7.1 shows the output of an ipconfig command that shows the “Local Area Connection 2” interface on the 192.168.0.xxx subnet.



```
C:\>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 176.16.25.17
        Subnet Mask . . . . . . . . . . . : 255.255.0.0
        Default Gateway . . . . . . . . . : 176.16.255.1
        DHCP Class ID . . . . . . . . . . :

Ethernet adapter Local Area Connection 2:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.0.224
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . :

C:\>
```

Figure R7.1 ipconfig Command

If your present address is not in the 192.168.0.xxx range, type in ‘ncpa.cpl’ at the command prompt and hit [Enter] on the keyboard.

- For Windows 7, this open the *Network Connections* window. Double click on the appropriate interface. In the window that opens, select “Internet Protocol Version 4 (TCP/IP v4)” from the list and then click on the [Properties] button.
- For Windows 10, this open the *Network Connections* window. Double click on the appropriate interface. In the window that opens, select “Internet Protocol Version 4 (TCP/IP v4)” from the list and then click on the [Properties] button.

Set the address and subnet mask to appropriate values. (192.168.0.1 and 255.255.255.0 will work for an NXFE2 that has factory default settings.) The default gateway and DNS server settings can be ignored.

Test Your Network Interface

Going back to the terminal you opened in the last step, type in ‘ping aaa.bbb.ccc.ddd’ where ‘aaa.bbb.ccc.ddd’ in the IP address of the NXFE2. The computer will ping the unit and the message “Reply from aaa.bbb.ccc.ddd: bytes=32 time<10ms TTL=128” should appear four times.

If the message “Request timed out.” or “Destination host unreachable” appears, then one of four things has occurred:

- You set a new IP address, but have not yet cycled power to the NXFE2
- You did not enter the correct address in the ping command.
- The IP address of the NXFE2 is not set correctly.
- The NXFE2 and the computer are not on the same subnet.

TASK 1

INSTALLING THE NXFE2

This chapter contains information on properly handling the NXFE2 as well as mounting the module in a safe environment. This chapter also covers how to wire your field I/O to the NXFE2.

1.1 Module Location

NXFE2 modules are suitable for use in industrial environments that meet the following criteria:

- Only non-conductive pollutants normally exist in the environment, but an occasional temporary conductivity caused by condensation is expected.
- Transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

Note that these criteria apply to the system as a whole. These criteria are equivalent to the *Pollution Degree 2* and *Over Voltage Category II* designations of the International Electrotechnical Commission (IEC)

1.2 Safe Handling Guidelines

1.2.1 Prevent Electrostatic Damage

CAUTION

Electrostatic discharge can damage the NXFE2 if you touch connector pins. Follow these guidelines when handling the module.

1) Touch a grounded object to discharge static potential before handling the module.
2) Work in a static-safe environment whenever possible.
3) Wear an approved wrist-strap grounding device.
4) Do not touch the pins of the I/O or Ethernet connectors.
5) Do not disassemble the module
6) Store the module in its anti-static bag and shipping box when it is not in use.

1.2.2 Prevent Debris From Entering the Module

WARNING

During DIN rail mounting of all devices, be sure that all debris (metal chips, wire strands, tapping liquids, etc.) is prevented from falling into the module. Debris may cause damage to the module or unintended machine operation with possible personal injury. The DIN rail for the modules should be securely installed and grounded before the modules are mounted on it.

1.2.3 Remove Power Before Servicing

WARNING

Remove power before removing or installing the NXFE2 module.

1.3 Mounting

1.3.1 DIN Rail Installation

NXFE2 modules can be mounted on following DIN Rail

- EN 05 022 - 35 x 7.5 (35 x 7.5 mm)
- EN 05 022 - 35 x 15 (35 x 15 mm)

The DIN Rail must be securely mounted to a panel and solidly grounded before the modules are installed. Grounding is usually accomplished through the mounting hardware, by first removing any paint or other material from all surfaces that may interfere with proper grounding. Another option is to install a heavy gauge wire from the DIN rail to your system's ground bus.

When installing the 7.5mm deep DIN rail there is a 0.40 inch clearance for bolt heads between the rail and the NXFE2 module. When installing a module on 15mm deep rail, the clearance is 0.70 inches.

1.3 Mounting (continued)

1.3.2 Dimensions

Figure T1.1 shows the dimensions of the NXFE2 module. The NXFE2 module is a low power module that does not require any additional spacing when mounting the unit.

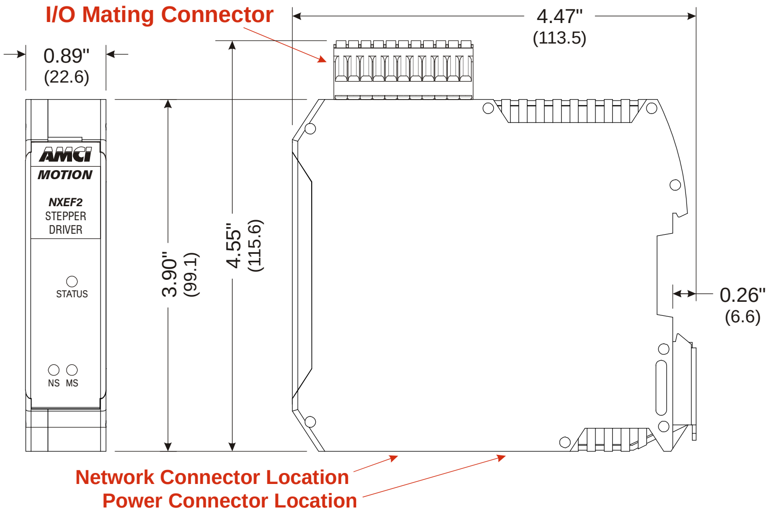


Figure T1.1 NXFE2 Outline

1.3.3 Mounting the NXFE2 Module

Mounting the NXFE2 on a DIN rail is a very simple process thanks to the design of the enclosure.

1) Partially engage the connector into the enclosure.
2) Engage the top clip in the enclosure with the top of the DIN rail and rotate the module down until the metal bracket snaps on to the DIN Rail.

Once all of your modules are installed, it is strongly suggested to use the end caps from Phoenix Contact with the part number of 271 37 80 to secure the modules on the DIN Rail. These end caps prevent the module from sliding along the DIN rail if it is subjected to shock or vibration during machine operation.

1.4 Powering the Module

1.4.1 Required Power

The NXFE2 requires 12 to 24 Vdc ± 20% supply to power the module. Surges to 30 Vdc will not damage it. Current draw is 112 mA @ 24 Vdc (2.7W).

This current draw does not include current through the inputs or outputs. If using the same supply to power the I/O, their load must be considered when sizing the supply.

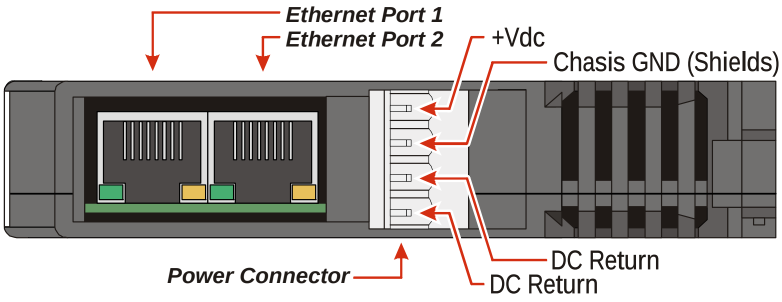
- Inputs require 10 mA each.
- Differential outputs can supply up to 20 mAdc each.
- Single ended outputs can sink up to 30 mAdc each.

1.4 Powering the Module (continued)

1.4.2 Wiring

As shown in the figure below, the power connector is located on the bottom of the module. The mating connector is included with the NXFE2. Spares are available from AMCI under the part number MS-4M. They are also available from Phoenix Contact under their part number 187 80 37.

WARNING Remove power from the NXFE2 before servicing. Do not connect or disconnect the MS-4M connector while power is applied.



Bottom View

Figure T1.2 Power Connector Location

Power connections should be tight, as loose connections may lead to arcing which will heat the connector. Phoenix Contact specifies a tightening torque of 4.4 to 5.4 lb-in (0.5 to 0.6 Nm).

The power supply is connected to the pins marked “+Vdc In” and “DC Return”. The “Chassis GNDIN” pin can be used to attach the NXFE to earth ground.

NOTE Each NXFE2 module must have its own power connection.

1.5 I/O Connector Pin Out

Figure T1.3 below shows the pin out of the I/O connector on the NXFE2 module. All mating connectors are included with the module. Additional connectors can be order for AMCI under the part number MS-2x11.

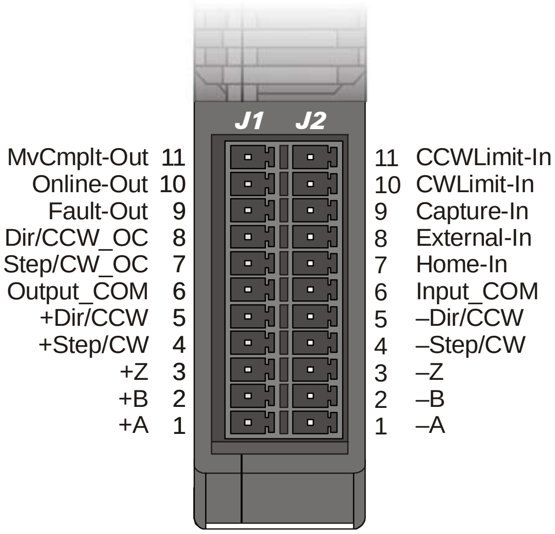


Figure T1.3 I/O Connector Pin Out

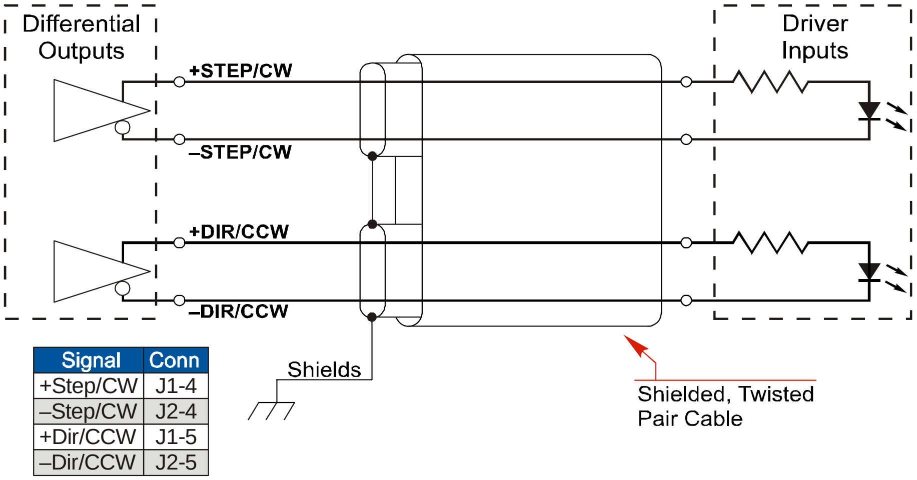
1.6 Driver Wiring

Connections to your driver can be made through the module's differential outputs or the single-ended open collector sinking outputs. You cannot use both output types at the same time.

1.6.1 Differential Wiring

Differential outputs are low voltage, low current outputs that require shielded, twisted pair cable to carry the signals. A suggested cable is the Belden 9729, which supports a 1 MHz pulse frequency to over 1000 feet of cable length; but any quality, shielded, twisted pair cable should suffice. A simple wiring diagram is shown below. Note that the cable shields should be grounded at the module end only.

Differential Output From NXFE2



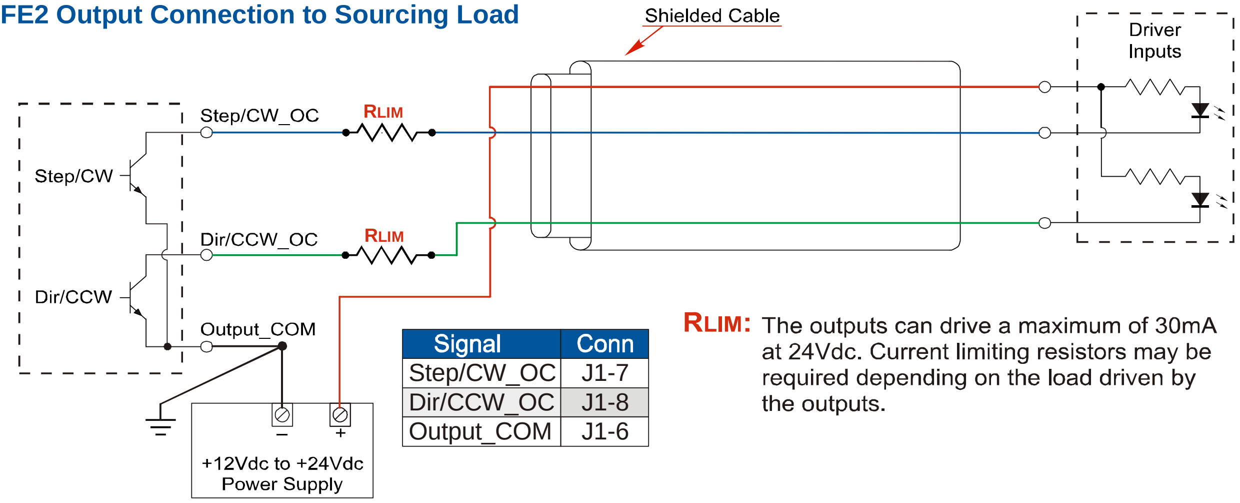
Signal	Conn
+Step/CW	J1-4
–Step/CW	J2-4
+Dir/CCW	J1-5
–Dir/CCW	J2-5

Figure T1.4 Differential Driver Wiring

1.6.2 Open Collector Wiring

All singled ended outputs in the NXFE2 are open collector sinking. They share a single common pin that must be tied to the ground connection of the power supply powering the outputs. Figure T1.5 below shows how to wire the outputs to open collector sourcing inputs on your driver.

NXFE2 Output Connection to Sourcing Load



Signal	Conn
Step/CW_OC	J1-7
Dir/CCW_OC	J1-8
Output_COM	J1-6

RLIM: The outputs can drive a maximum of 30mA at 24Vdc. Current limiting resistors may be required depending on the load driven by the outputs.

Figure T1.5 Open Collector Driver Wiring (Sourcing Inputs)

NOTE RLIM may be needed to limit current flow. The NXFE2 drivers can conduct a maximum of 30 milliamperes. The exact value of RLIM is based on the power supply voltage and the current requirements of the Driver Inputs.

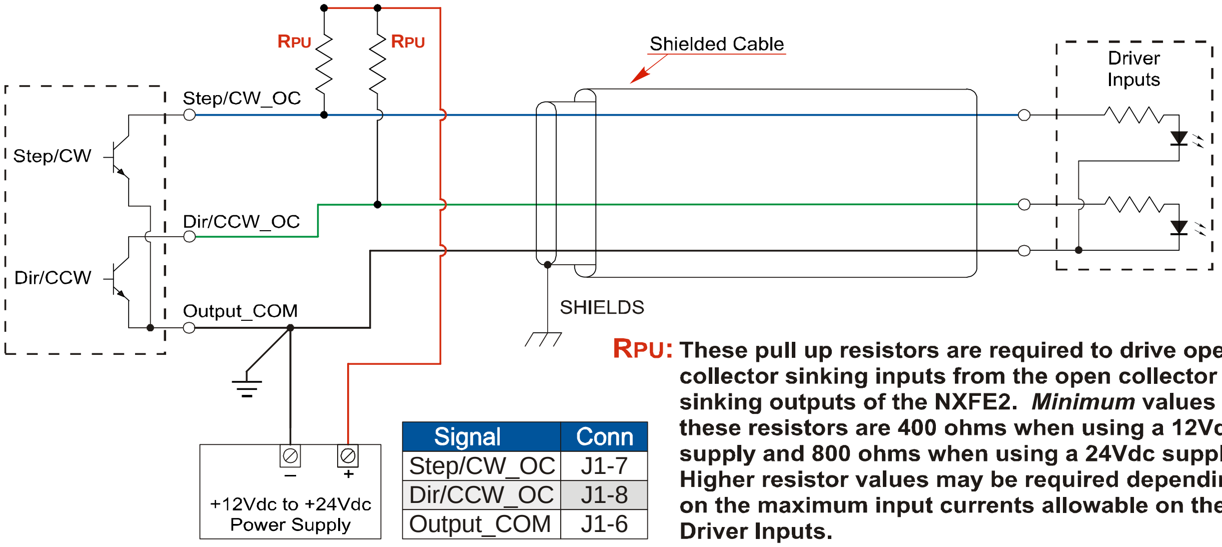
1.6 Driver Wiring (continued)

1.6.2 Open Collector Wiring (continued)

It is possible to wire the NXFE2 outputs to open collector sinking inputs on a driver by using pull up resistors.

NOTE This configuration will effectively reverse the logic states of the outputs. A Driver Input will be normally on and turn off when the NXFE2 output is in a logic “1” state.

NXFE2 Output Connection to Sinking Load



Signal	Conn
Step/CW_OC	J1-7
Dir/CCW_OC	J1-8
Output_COM	J1-6

RPU: These pull up resistors are required to drive open collector sinking inputs from the open collector sinking outputs of the NXFE2. *Minimum* values for these resistors are 400 ohms when using a 12Vdc supply and 800 ohms when using a 24Vdc supply. Higher resistor values may be required depending on the maximum input currents allowable on the Driver Inputs.

Figure T1.6 Open Collector Driver Wiring (Sinking Inputs)

CAUTION Carefully select the value of the pull up resistors. (RPU in the figure.) The NXFE2 outputs can conduct a maximum of 30 milliamperes without damage, so the minimum resistor value is 400 ohms / 0.5 watt when using a 12 Vdc supply and 800 ohms / 1.0 watt when using a 24 Vdc supply. *You must increase the size of these resistors if the inputs to your driver cannot conduct 30 milliamps without damage.*

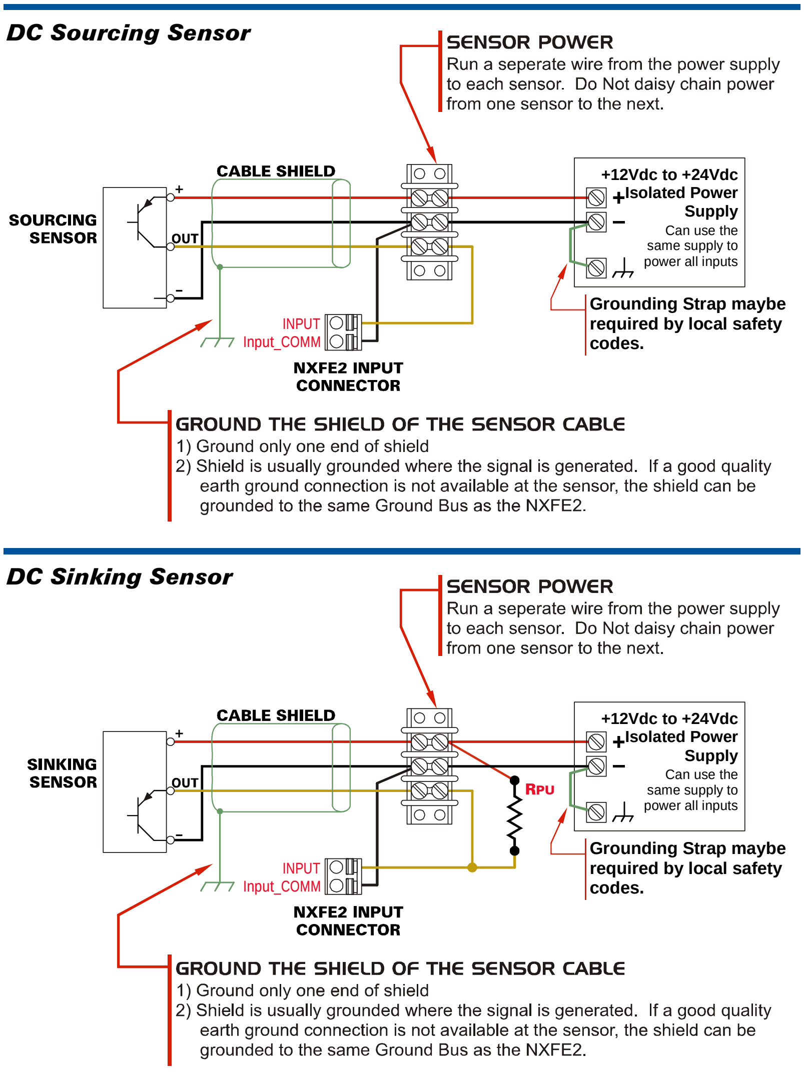
1.7 Digital Output Wiring

The module has three digital outputs named Move_Complete, Axis Fault, and Online. These outputs are electrically identical to the open collector Step/CW_OC and Dir/CCW_OC outputs. These digital outputs share a common ground pin with the driver outputs. Wiring is identical to the diagrams given in figures T1.5 and T1.6.

NOTE The digital outputs can, but do not have to, use the power supply that powers the open collector driver outputs. If you use two different supplies, the low side of the supplies must be tied together at pin J1-6. For this reason, the two supplies should be mounted close together and share the same Earth Ground bus to avoid ground loop issues.

1.8 Discrete Input Wiring

Figure T1.7 below shows how to wire discrete DC sourcing and sinking sensors to the open collector sinking inputs of the NXFE2.


DC Sourcing Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SOURCING SENSOR
+
OUT
–
+12Vdc to +24Vdc
+ Isolated Power Supply
–
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
INPUT
Input_COMM
NXFE2 INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the NXFE2.
DC Sinking Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SINKING SENSOR
+
OUT
–
RPU
+12Vdc to +24Vdc
+ Isolated Power Supply
–
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
INPUT
Input_COMM
NXFE2 INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the NXFE2.

Input Pins

NXFE2

Signal	Pin
Input_COM	J2-6
Home-In	J2-7
External-In	J2-8
Capture-In	J2-9
CWLimit-In	J2-10
CCWLimit-In	J2-11

Figure T1.7 Input Wiring

Input Specifications:

Single Ended. Accepts 8 to 27Vdc without the need for an external current limiting resistor.

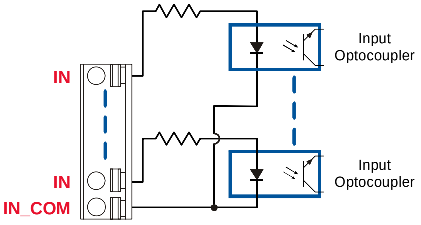


Figure T1.8 DC Input Schematic

Because they are low power signals, cabling from the sensor to the NXFE2 should be done using a twisted pair cable with an overall shield. The shield should be grounded at the end when the signal is generated, which is the sensor end. If this is not practical, the shield should be grounded to the same ground bus as the NXFE2

Sinking Output Sensors Require a Pull Up Resistor

Sinking output sensors require an external pull up resistor because the input to the NXFE2 also sinks current. Table T1.1 below shows the values of pull up resistors that will allow the NXFE2 input to activate along with the current that the sensor must be able to sink when it is active.

Input Voltage	Pull Up Resistor	Sensor Current When Active
12	1.4K	8.6mA
24	3.8K	6.3mA

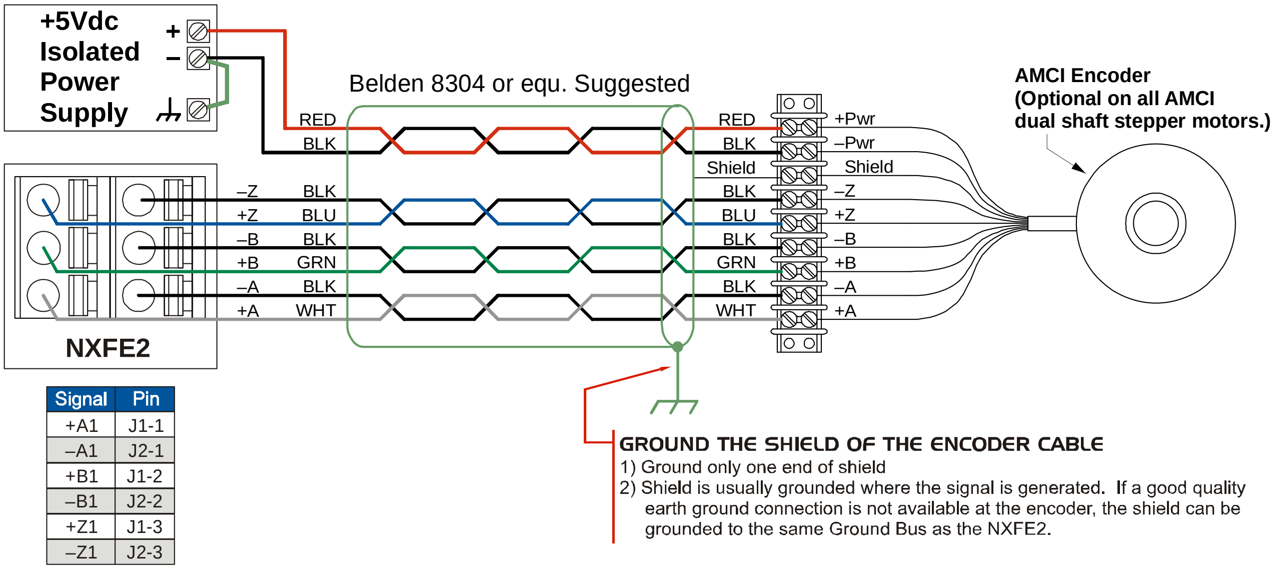
Table T1.1 Pull Up Resistor

The logical states of the sensor and NXFE2 input will be reversed. The NXFE2 input is off when the sensor is active. You can set the logic state of the NXFE2 input when you configure the unit.

1.9 Encoder Wiring

1.9.1 Differential Encoder Wiring

The encoder inputs on the NXFE2 accept 5Vdc differential signals from your incremental encoder. AMCI offers optical encoders as an option when ordering any of our dual-shaft stepper motors. Figure T1.9 below shows a typical encoder installation using an AMCI optical encoder.



Signal	Pin
+A1	J1-1
–A1	J2-1
+B1	J1-2
–B1	J2-2
+Z1	J1-3
–Z1	J2-3

GROUND THE SHIELD OF THE ENCODER CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the encoder, the shield can be grounded to the same Ground Bus as the NXFE2.

Figure T1.9 Differential Encoder Wiring

The table below lists the wire color code for the AMCI encoder used at the time this manual was released. These encoders have an eighteen inch integral cable. Note that there is no single color coding standard in the encoder market so you must refer to your encoder's documentation if you are not using an AMCI optical encoder.

Function	Wire Color
+Pwr (+5 Vdc)	Red
–PWR (Gnd)	Black
+A	Brown
–A	White
+B	Blue
–B	Green
+Z	Orange
–Z	Yellow

Table T1.2 AMCI Encoder Wire Color Chart

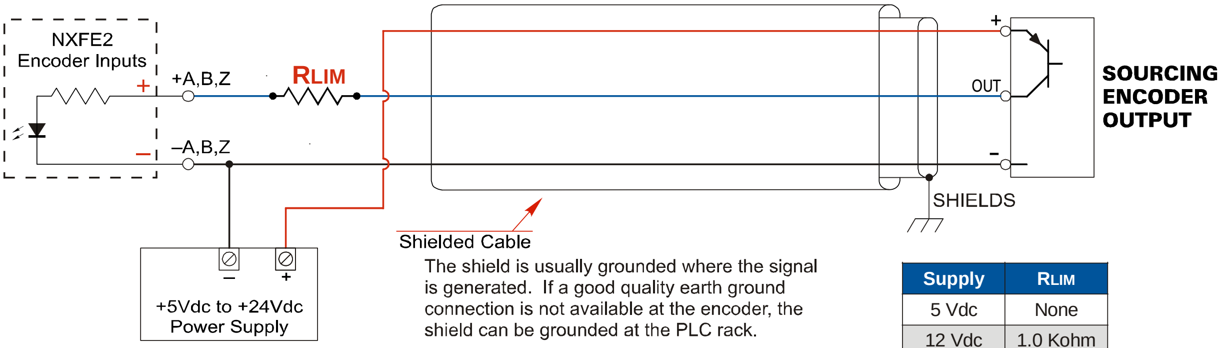
1.9 Encoder Wiring (continued)

1.9.2 Single Ended Wiring

Figure T1.10 below shows how to wire the encoder inputs to both a single ended sourcing and single ended sinking encoder outputs.

CAUTION The encoder inputs on the NXFE2 are rated for 5Vdc only. You must use a current limiting resistor on each input if the outputs of your encoder are greater than 5Vdc. Appropriate current limiting resistors are shown in the figure below.

NXFE2 Encoder Input Connection to Sourcing Encoder Output

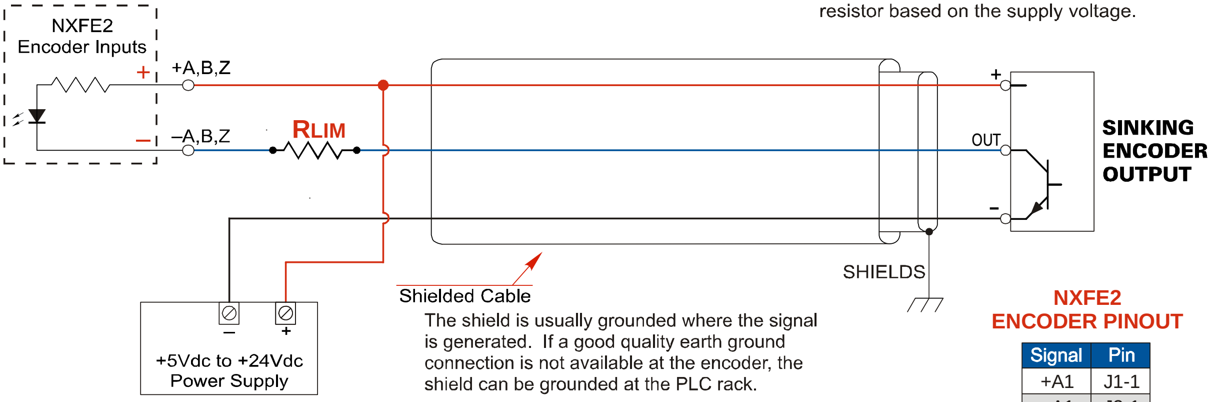


The shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the encoder, the shield can be grounded at the PLC rack.

Supply	RLIM
5 Vdc	None
12 Vdc	1.0 Kohm
15 Vdc	1.2 Kohm
24 Vdc	2.7 Kohm

RLIM: The inputs are designed to accept 5Vdc but can use any voltage up to 24Vdc with the appropriate current limiting resistor. See the table above for the required resistor based on the supply voltage.

NXFE2 Encoder Input Connection to Sinking Encoder Output



The shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the encoder, the shield can be grounded at the PLC rack.

NXFE2 ENCODER PINOUT

Signal	Pin
+A1	J1-1
–A1	J2-1
+B1	J1-2
–B1	J2-2
+Z1	J1-3
–Z1	J2-3

Figure T1.10 Single Ended Encoder Wiring

1.10 Diagnostic Feedback Wiring

The NXFE2 can be placed in a Diagnostic Feedback Mode that allows you to check the functionality of the differential motor control outputs and the encoder A and B inputs. You must wire the outputs to the inputs as shown in the table below before you can use this mode.

Wire This Pin...		...To This Pin	
Function	Pin #	Pin #	Function
+Step1	J1-4	J1-1	+A1
–Step1	J2-4	J2-1	–A1
+Direction1	J1-5	J1-2	+B1
–Direction1	J2-5	J2-2	–B1

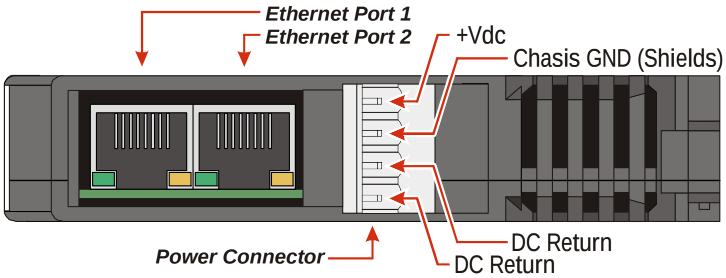
Table T1.3 Diagnostic Feedback Wiring

The axis you are testing must be configured to output step and direction pulses The NXFE2 will issue a configuration error if the axis is configured to output CW/CCW pulses.

1.11 Ethernet Connections

The Ethernet connectors are located on the bottom of the unit. The connectors are standard RJ-45 jacks that will accept any standard 100baseT cable. Because the port can run at 100 Mbit speeds, Category 5, 5e, or 6 cable should be used.

The Ethernet ports on the units are “auto-sense” ports that will automatically switch between 10baseT and 100baseT depending on the network equipment they are attached to. The ports also have “auto switch” capability. This means that a standard cable can be used to connect the NXFE2 to any device, including a personal computer.



Bottom View

Figure R7.2 Ethernet Port Locations

The Network Status LED’s are fully described on page 17 in the ***Status LED*** section of the Specifications reference.

1.12 EtherNet/IP Connections

1.12.1 Non-DLR Applications

The Networked Driver has two Ethernet ports with a built-in Ethernet switch connecting the two. In non-DLR applications, either port can be used to attach the unit to the network. The remaining port can be used to extend the network to another device if this would reduce wiring costs.

1.12.2 DLR Applications

In Device Level Ring applications, the Networked Driver functions as Beacon-Based Ring Nodes. In these applications, both ports are used when wiring the ring, daisy chaining from one unit in the ring to the next.

1.13 PROFINET Connections

1.13.1 Non-MRP Applications

The Networked Driver has two Ethernet ports with a built-in Ethernet switch connecting the two. In non-MRP applications, either port can be used to attach the unit to the network. Also, both ports can be used in non-MRP applications. For example, if two units are located some distance from your controller, then you need only run one cable from your controller to the first unit. The second unit can then be attached to the first with a short cable. There is no need to run two cables from your controller.

1.13.2 MRP Applications

In Media Redundancy Protocol applications, the Networked Driver functions as a Media Redundancy Client (MRC). In these applications, both ports are used when wiring the ring, daisy chaining from one unit in the ring to the next.

1.14 Modbus TCP Connections

The Networked Driver has two Ethernet ports with a built-in Ethernet switch connecting the two. Either port can be used to attach the unit to the network. The remaining port can be used to extend the network to another device if this would reduce wiring costs.

Set the IP Address and Protocol

This section is intended for the engineer or technician responsible for setting the IP address of an AMCI NXFE2.

NOTE ➔ If you are using an NXFE2 in a PROFINET environment, you can skip this task. The IP Address is written to the unit as part of the standard PROFINET configuration. You can proceed to ***PROFINET Network Configuration***, which starts on page 115

2.1 Determine the Best Method for Setting the IP Address

There are three methods for setting the IP address on an NXFE2. Table T2.1 below outlines the available methods and when you can use them.

Method	Restrictions	Section
Use Factory Default Settings	1) The machine must use 192.168.0.xxx subnet. 2) The 192.168.0.50 address must be available.	2.2a
Use the Embedded Web Server	No restrictions on use. This is the preferred method. The internal webserver can be used to set the NXFE2 to any IPv4 address. The IP address and protocol will be stored in nonvolatile memory and used on subsequent power-ups.	2.2b
Use the AMCI NET Configurator Utility	No restrictions on use. The software can be used to set the NXFE2 to any IPv4 address and protocol. The IP address and protocol choice will be stored in nonvolatile memory and used on subsequent power-ups.	2.2c

Table T2.1 Methods for Setting the IP Address

NOTE ➔ There is a MAC address label on each NXFE2 which has a writable surface. There is room on the label for writing the programmed IP address of the unit. It is a best practice to use this label to document the IP address of the unit in case it is ever repurposed.

NOTE ➔ When using EtherNet/IP, the NXFE2 units also supports the DHCP protocol. You will need an EtherNet/IP DHCP server, such as the one available from Rockwell Automation, in order to use this protocol. The AMCI Net Configurator utility offers the same functionality and should be used unless your company policy prevents you from installing third party utilities.

2.2a Use Factory Default Settings

The factory default address for the NXFE2 units is 192.168.0.50 with a subnet mask of 255.255.255.0. The easiest way to verify this address is with the ping command as described in section ***Configure Your Network Interface*** starting on page 71.

If the driver does not respond to this address, then it may take some effort to determine the correct address. There is a label on the driver that lists the MAC address of the device. There is space on the label for noting the IP address of the device if it is changed. If the address was not documented, a program called Wireshark (***https://www.wireshark.org/***) can be used to determine the address of the driver.

Task Complete

2.2b Use the Embedded Web Server

PREREQUISITE: You must know the present IP address of the NXFE2. The factory default address is 192.168.0.50.

PREREQUISITE: Task 1.4: ***Powering the Module***, starting on page 74. You must be able to power the NXFE2.

PREREQUISITE: Task 1.11 ***Ethernet Connections***, found on page 81. You must attach your NXFE2 to your computer.

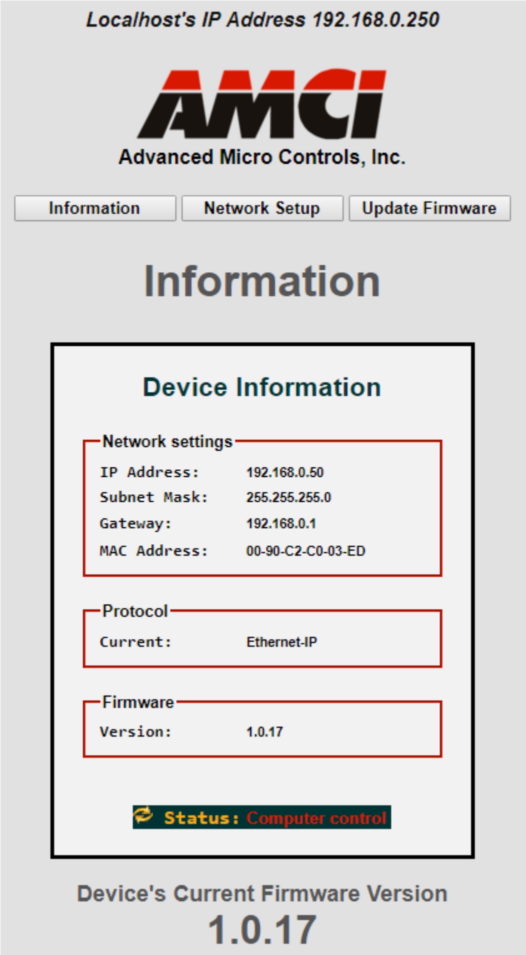
PREREQUISITE: Reference 7: ***Configuring Network Interfaces***, which starts on page 71. The network interfaces on your computer must be on the same subnet before you can communicate with an NXFE2.

2.2b.1 Disconnect the NXFE2 from the host controller and cycle power to the NXFE2.

This ensures that the unit does not have any open connections to the host controller.

2.2b.2 Start your web browser and connect to the NXFE2

The internal HTML pages should work with any browser. Once your web browser is running, enter the present IP address of the NXFE2 into the address bar. The default address is 192.168.0.50. The unit will respond with the following page. Note that the Firmware Version number may be different.



Localhost's IP Address 192.168.0.250

Advanced Micro Controls, Inc.

Information

Device Information

Network settings

IP Address: 192.168.0.50

Subnet Mask: 255.255.255.0

Gateway: 192.168.0.1

MAC Address: 00-90-C2-C0-03-ED

Protocol

Current: Ethernet-IP

Firmware

Version: 1.0.17

Status: Computer control

Device's Current Firmware Version

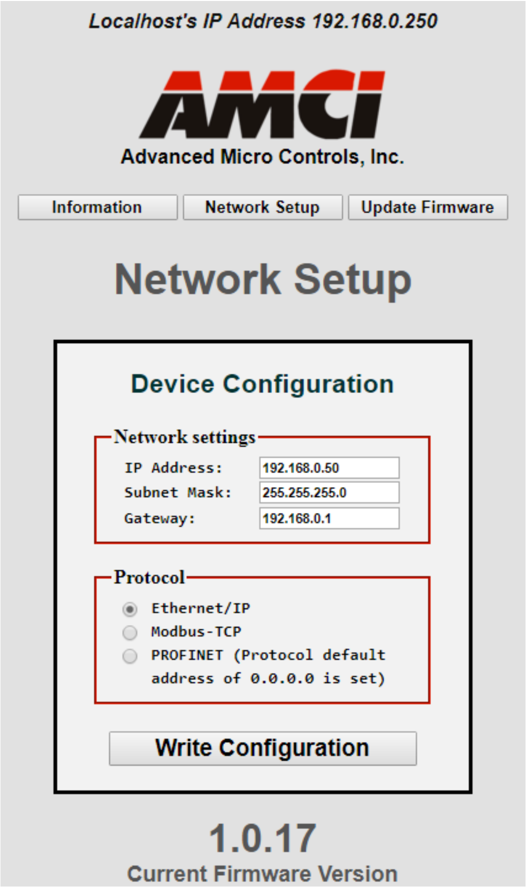
1.0.17

Figure T2.1 NXFE2 Information Webpage

2.2b Use the Embedded Web Server (continued)

2.2b.3 Network Setup Page

1) Click on the [Network Setup] button to switch to the Network Setup page shown below. This page shows the current IP address settings, as well as the configured protocol.



Localhost's IP Address 192.168.0.250

AMCI

Advanced Micro Controls, Inc.

Network Setup

Device Configuration

Network settings

IP Address: 192.168.0.50

Subnet Mask: 255.255.255.0

Gateway: 192.168.0.1

Protocol

● Ethernet/IP

○ Modbus-TCP

○ PROFINET (Protocol default address of 0.0.0.0 is set)

1.0.17

Current Firmware Version

Figure T2.2 NXFE2 Network Setup Web Page

2) Enter your desired values into the IP Address, Subnet Mask, and Default Gateway fields.

NOTE ➤ The Default Gateway setting is not optional! It must be set to a valid address on the chosen subnet. Because the Default Gateway is often not used in device level networks, if you do not have a required value for it, AMCI suggests setting the Default Gateway to the IP address of your host controller.

3) If need be, click on the proper radio button to select the required protocol.

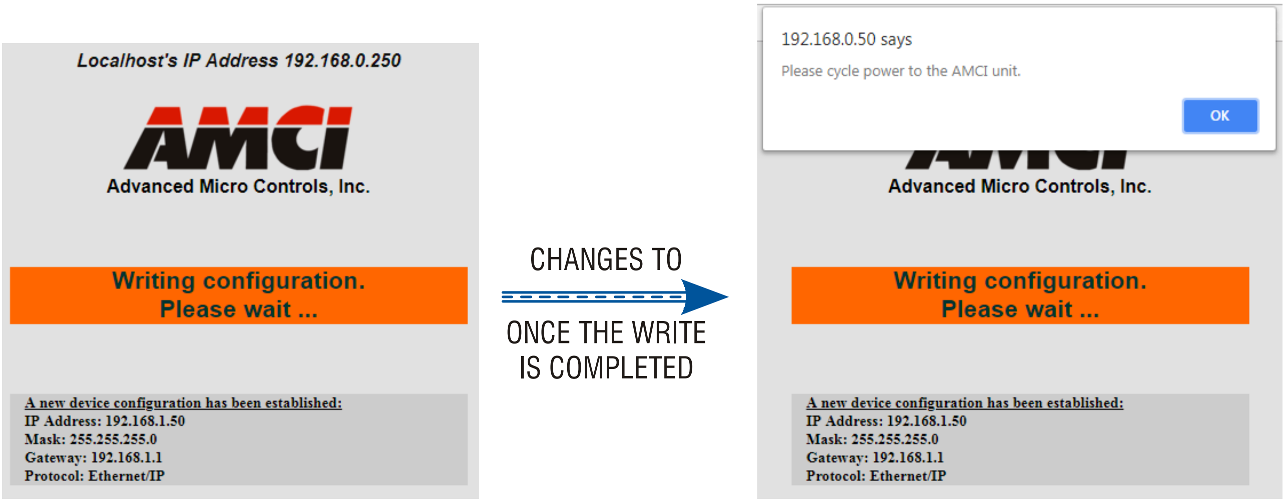
4) Click on the [Write Configuration] button to write the new configuration to the unit. If there are any errors with the data, the unit will display a warning message instead of accepting the new values.

2.2b Use the Embedded Web Server (continued)

2.2b.3 Network Setup Page (continued)

5) If the values are accepted, the following pages will be displayed while the data is being written to the unit.

NOTE ➨ Wait for the pop up window to appear before cycling power to the NXFE2. Cycling power before this window appears may corrupt the non-volatilize memory of the NXFE2. The NXFE2 will also flash the Network Status LED red to indicate that power must be cycled.



Localhost's IP Address 192.168.0.250

Advanced Micro Controls, Inc.

Writing configuration.
Please wait ...

A new device configuration has been established:
IP Address: 192.168.1.50
Mask: 255.255.255.0
Gateway: 192.168.1.1
Protocol: Ethernet/IP

192.168.0.50 says

Please cycle power to the AMCI unit.

Advanced Micro Controls, Inc.

Writing configuration.
Please wait ...

A new device configuration has been established:
IP Address: 192.168.1.50
Mask: 255.255.255.0
Gateway: 192.168.1.1
Protocol: Ethernet/IP

Figure T2.3 Write Configuration to Flash Memory Pages

6) Once instructed to, cycle power to the unit. You can now enter the new IP address into the address bar of your web browser to reconnect with the NXFE2.

Task Complete

2.2c Use the AMCI NET Configurator Utility

PREREQUISITE: You must know the present IP address of the NXFE2. The factory default address is 192.168.0.50.

PREREQUISITE: Task 1.4: ***Powering the Module***, starting on page 74. You must be able to power the NXFE2.

PREREQUISITE: Task 1.11 ***Ethernet Connections***, found on page 81. You must attach your NXFE2 to your computer.

PREREQUISITE: Reference 7: ***Configuring Network Interfaces***, which starts on page 71. The network interfaces on your computer must be on the same subnet before you can communicate with an NXFE2.

2.2c.1 Download the AMCI Net Configurator Utility

The AMCI Net Configurator utility is available on our website, ***www.amci.com***. The latest version available should be used. It can be found in the *Support* section under *Software*. The program exists as a ZIP file, and at the time of this writing, the link was “AMCI NET Configurator Utility for all networked products...”.

2.2c.2 Install the AMCI Net Configurator Utility

Once downloaded, simply extract the program from the ZIP file and run the program to install the AMCI Net Configurator utility on your computer. The software installs as most products do, giving you the option to change the file locations before installing the utility. Once the install is complete, a link to the utility is available on the Start Menu.

The install process only copies the utility to the designated location and creates links to the Start Menu. No changes are made to your registry settings.

2.2c Use the AMCI Net Configurator Utility (continued)

2.2c.3 Verify that Your Host Controller is Disconnected from the NXFE2

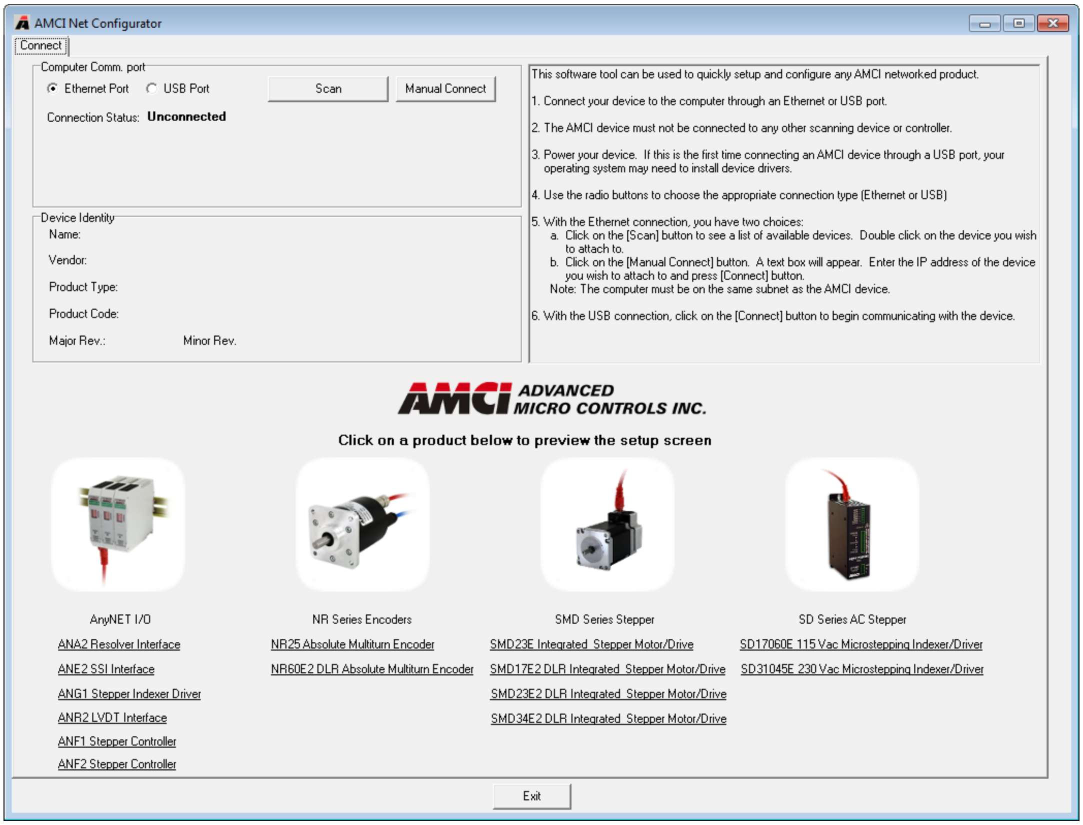
In order to program the NXFE2, the AMCI Net Configurator utility must act as a bus master. Therefore, physically disconnect your host controller from the NXFE2 before starting the Net Configurator utility. Conflicting commands from the Net Configurator utility and the host controller will prevent the NXFE2 from communicating properly with either bus master.

2.2c.4 Apply or Cycle Power to the NXFE2

Cycling power to the NXFE2 will reset any connections it may have with the host controller.

2.2c.5 Start the AMCI Net Configurator Utility

Double click on the utility's icon. A welcome screen similar to the one in figure T2.4 below will appear.



Connection Status: **Unconnected**

Device Identity
Name:
Vendor:
Product Type:
Product Code:
Major Rev.: Minor Rev.

This software tool can be used to quickly setup and configure any AMCI networked product.

1. Connect your device to the computer through an Ethernet or USB port.

2. The AMCI device must not be connected to any other scanning device or controller.

3. Power your device. If this is the first time connecting an AMCI device through a USB port, your operating system may need to install device drivers.

4. Use the radio buttons to choose the appropriate connection type (Ethernet or USB)

5. With the Ethernet connection, you have two choices:
   a. Click on the [Scan] button to see a list of available devices. Double click on the device you wish to attach to.
   b. Click on the [Manual Connect] button. A text box will appear. Enter the IP address of the device you wish to attach to and press [Connect] button.
   Note: The computer must be on the same subnet as the AMCI device.

6. With the USB connection, click on the [Connect] button to begin communicating with the device.

Click on a product below to preview the setup screen

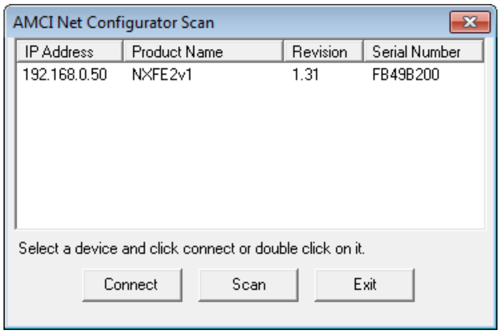
AnyNET I/O	NR Series Encoders	SMD Series Stepper	SD Series AC Stepper
ANA2 Resolver Interface	NR25 Absolute Multiturn Encoder	SMD23E Integrated Stepper Motor/Drive	SD17060E 115 Vac Microstepping Indexer/Driver
ANE2 SSI Interface	NR60E2 DLR Absolute Multiturn Encoder	SMD17E2 DLR Integrated Stepper Motor/Drive	SD31045E 230 Vac Microstepping Indexer/Driver
ANG1 Stepper Indexer Driver		SMD23E2 DLR Integrated Stepper Motor/Drive	
ANR2 LVDT Interface		SMD34E2 DLR Integrated Stepper Motor/Drive	
ANF1 Stepper Controller			
ANF2 Stepper Controller			

Figure T2.4 Net Configurator Welcome Screen

2.2c Use the AMCI Net Configurator Utility (continued)

2.2c.6 Press the [SCAN] button and Connect to the NXFE2

Pressing the [Scan] button will open the window shown in figure T2.5. The NXFE2 will appear in the scan list only if the unit and your network interface are on the same subnet. Optionally, you can press the [Manual Connect] button and enter the IP address of the NXFE2.



AMCI Net Configurator Scan

IP Address	Product Name	Revision	Serial Number
192.168.0.50	NXFE2v1	1.31	FB49B200

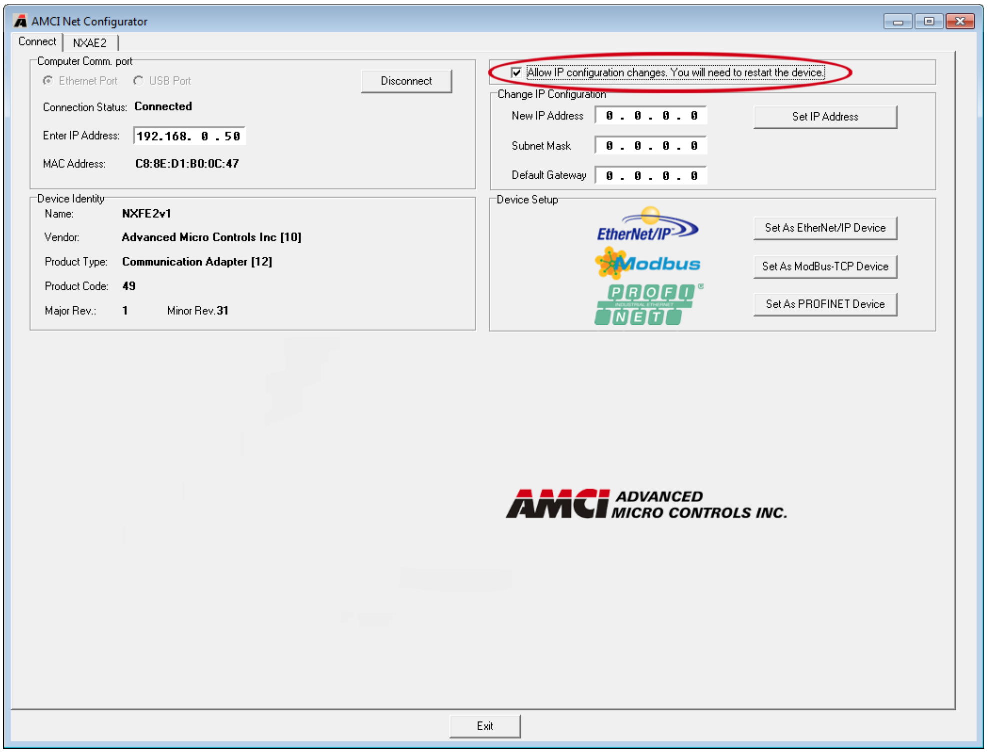
Select a device and click connect or double click on it.

Figure T2.5 Scan for NXFE2

If scanning for the NXFE2, click on the IP Address of the module and click on the [Connect] button. The Net Configurator utility will connect to the unit.

2.2c.7 Click on the "Allow IP..." Checkbox to Access the IP Settings

Figure T2.6 below shows the screen that results when you are connected to the NXFE2. In order to change the IP Address of the unit, you must first click on the checkbox next to the text “Allow IP configuration changes. You will need to restart the device.” Once the checkbox is selected, the [Set IP Address] button will be enabled.



AMCI Net Configurator

Connect | NXAE2

Computer Comm. port

● Ethernet Port ○ USB Port

Connection Status: Connected

Enter IP Address: 192.168. 0 . 50

MAC Address: C8:8E:D1:B0:0C:47

☑ Allow IP configuration changes. You will need to restart the device.

Change IP Configuration

New IP Address 0 . 0 . 0 . 0

Subnet Mask 0 . 0 . 0 . 0

Default Gateway 0 . 0 . 0 . 0

Device Identity

Name: NXFE2v1

Vendor: Advanced Micro Controls Inc [10]

Product Type: Communication Adapter [12]

Product Code: 49

Major Rev.: 1 Minor Rev.31

Figure T2.6 Enable IP Address Changes

2.2c Use the AMCI Net Configurator Utility (continued)

2.2c.8 Set the IP Address, Subnet Mask, and Default Gateway

Enter your desired values into the IP Address, Subnet Mask, and Default Gateway fields.

NOTE ➤ The Default Gateway setting is not optional! In order to comply with the ODVA specification, it must be set to a valid address on the chosen subnet. Because the Default Gateway is often not used in device level networks, if you do not have a required value for it, AMCI suggests setting the Default Gateway to the IP address of your host controller.

2.2c.9 Set the Communications Protocol

The factory default protocol for the NXFE2 is EtherNet/IP. In order to use the Modbus TCP or PROFINET protocols, simply click on the appropriate button.

2.2c.10 Write the New IP Address to the NXFE2

Click on the [Set IP Address] button. If there is an error in the settings, the utility will tell you what is wrong. Once they are all correct, the utility will write the new IP address settings to the unit. These settings are automatically saved to nonvolatile memory.

2.2c.11 Remove Power from the NXFE2

The new IP address will not be used until power to the NXFE2 has been cycled.

Task Complete

Notes

Protocol Specific Information

Manual Sections

The remainder of this manual is divided into three sections, one for each supported protocol. Each section has the protocol name in the page header.

Starting Pages

EtherNet/IP protocol: Page 93

Modbus TCP protocol: Page 111

PROFINET protocol: Page 115

Notes

Implicit Communications with an EDS

Many EtherNet/IP platforms support the use of EDS files to simplify the addition and configuration of devices. This chapter covers the installation and use of the EDS file for systems that are programmed with Rockwell Automation Studio 5000 version 20 and above. Other systems will follow a similar pattern. Consult your controller's documentation if you need additional information.

Note: Use of an EDS file is completely optional. The NXFE2 can always be added to a system as a generic module. If you are using RSLogix 5000 version 19 and below, or RSLogix 500, adding the unit as a generic device is the only option available.

Using the EDS file simplifies configuration by sizing the configuration values appropriately.

3.1 Obtain the EDS file

All AMCI EDS files are located on our website at the following address:

- ***http://www.amci.com/industrial-automation-support/configuration-files/***

Products are listed in alphabetical order. Simply download the ZIP file and extract it to its own directory. The ZIP file contains the EDS text file and a custom icon file for the device.

3.2 Install the EDS file

3.2.1 Start the EDS Hardware Installation Tool

1) Once Studio 5000 is running, in the menu bar select Tools → EDS Hardware Installation Tool. This will open the EDS Wizard.

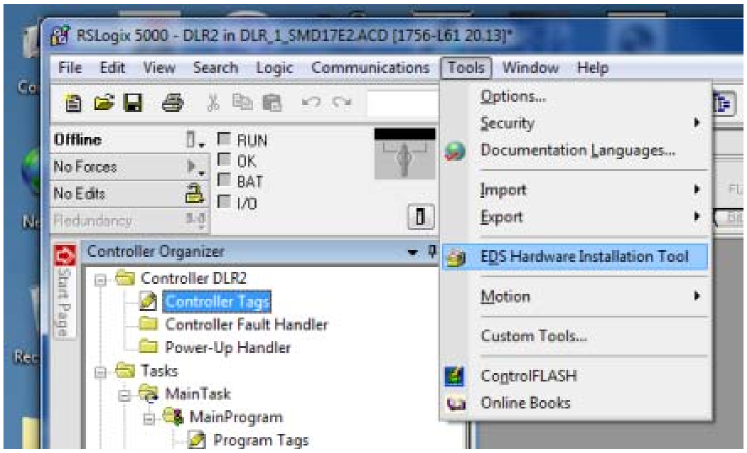


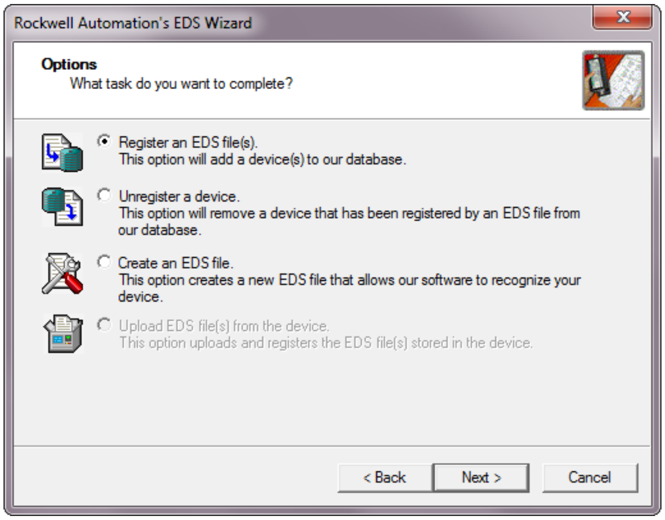
Figure T3.1 Opening the EDS Wizard

2) Click on [Next >] to advance to the Options screen.

3.2 Install the EDS file (continued)

3.2.2 Install the EDS File

1) On the Options screen, select the Register an EDS file(s) radio button and press [Next >].



Rockwell Automation's EDS Wizard

Options
What task do you want to complete?

● Register an EDS file(s).
This option will add a device(s) to our database.

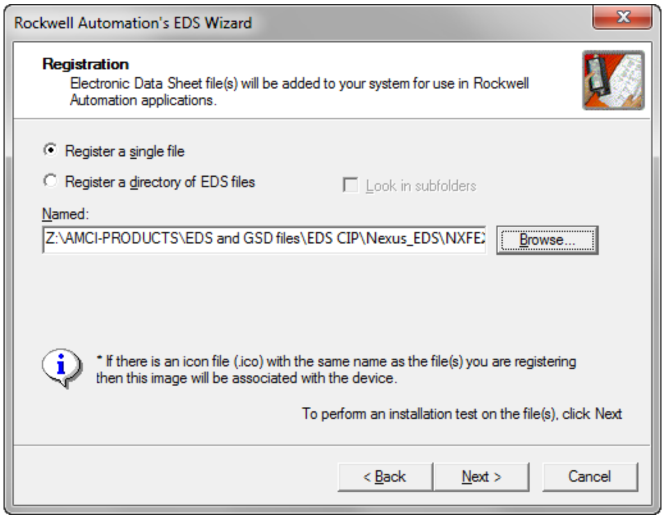
○ Unregister a device.
This option will remove a device that has been registered by an EDS file from our database.

○ Create an EDS file.
This option creates a new EDS file that allows our software to recognize your device.

○ Upload EDS file(s) from the device.
This option uploads and registers the EDS file(s) stored in the device.

Figure T3.2 EDS Options Screen

2) The registration screen will open. Select the Register a single file radio button.



Rockwell Automation's EDS Wizard

Registration
Electronic Data Sheet file(s) will be added to your system for use in Rockwell Automation applications.

● Register a single file

○ Register a directory of EDS files □ Look in subfolders

Named:
Z:\AMCI-PRODUCTS\EDS and GSD files\EDS CIP\Nexus_EDS\NXFE

* If there is an icon file (.ico) with the same name as the file(s) you are registering then this image will be associated with the device.

To perform an installation test on the file(s), click Next

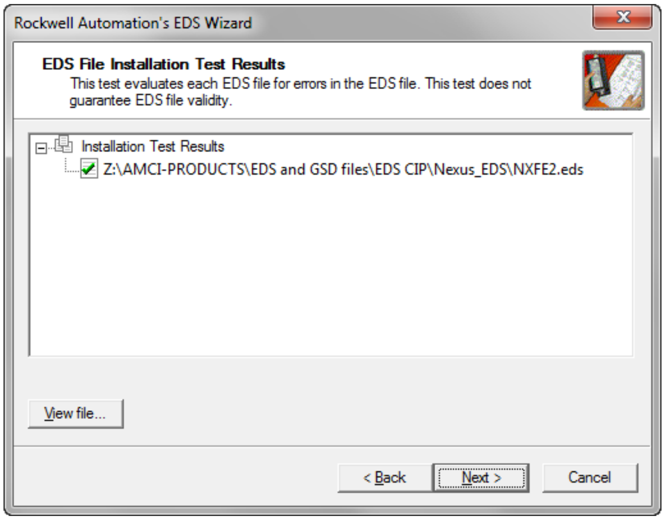
Figure T3.3 EDS Registration Screen

3) Click on the [Browse...] button and browse to the folder that contains the extracted EDS file you downloaded from the AMCI website. Select the EDS file and click on the [Open] button to return to the registration screen. Click on the [Next >] button to advance to the EDS file test screen.

3.2 Install the EDS file (continued)

3.2.2 Install the EDS File (continued)

4) Once at the EDS File Installation Test Results screen, expand the tree as needed to view the results of the installation test for the EDS file. You should see a green check mark next to the file name indicating that the EDS file is correct.



Rockwell Automation's EDS Wizard

EDS File Installation Test Results

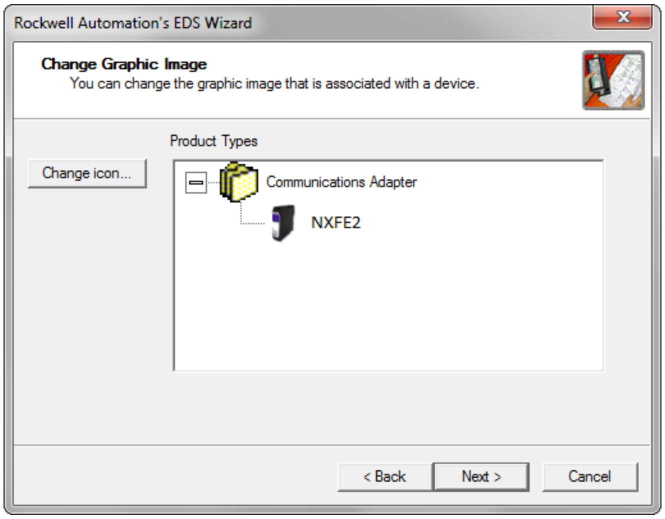
This test evaluates each EDS file for errors in the EDS file. This test does not guarantee EDS file validity.

Installation Test Results

☑ Z:\AMCI-PRODUCTS\EDS and GSD files\EDS CIP\Nexus_EDS\NXFE2.eds

Figure T3.4 EDS Test Screen

5) Press on the [Next >] button to advance to the Change Graphic Image screen. This screen gives you the ability to change the icon associated with the device.



Rockwell Automation's EDS Wizard

Change Graphic Image

You can change the graphic image that is associated with a device.

Figure T3.5 Change ECS Icon Screen

6) Click on the [Change icon...] button. In the window that opens, click on [Browse...] and browse to the folder that contains the extracted EDS and icon files you downloaded from the AMCI website.

7) Select the icon file (*.ico) associated with the device. Click on the [Open] button and then on [OK] to return to the Change Graphic Image screen.

8) Click on the [Next...] button to advance to the completion screen. The Completion screen tells you that you have successfully completed the wizard.

9) Click on the [Finish] button to exit the EDS wizard.

3.3 Host System Configuration

Studio 5000 is used to configure both the ControlLogix and CompactLogix platforms. When using these platforms, you have the option of using a separate Ethernet Bridge module or an Ethernet port built into the processor.

If the Ethernet port is built into processor, the only step you have to take before adding an AMCI NXFE2 is to create a new project with the correct processor or modify an existing project. Once this is done, the Ethernet port will automatically appear in the Project Tree. If you are using an Ethernet bridge module, you will have to add it to the I/O Configuration tree before adding the unit to your project.

Refer to your Rockwell Automation documentation if you need instructions for configuring the ethernet port.

3.4 Add the NXFE2 to Your Project

You can add an NXFE2 to the project once the Ethernet port (built-in or bridge module) is configured. As shown in figure T3.6 below, the Ethernet port will be listed under the I/O Configuration tree.

1) Right click on the Ethernet port and then click on "New Module..." in the pop-up menu.

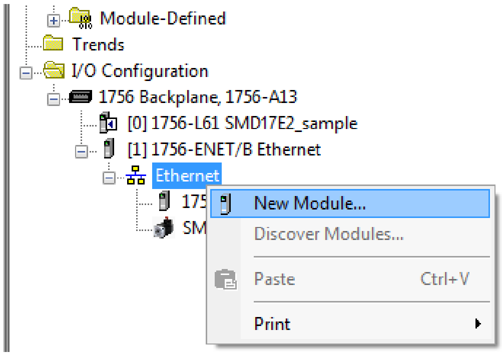


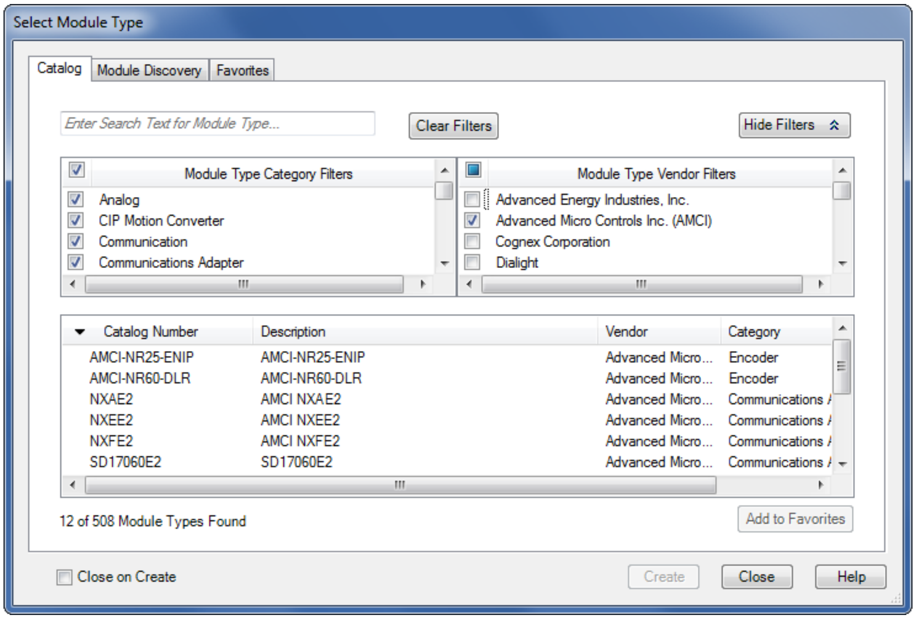
Figure T3.6 Adding an AMCI Ethernet Unit

2) In the resulting Select Module Type screen, select "Advanced Micro Controls Inc. (AMCI)" in the Vendor Filters. This will limit the results to catalog numbers from AMCI.

3) Select "NXFE2" in the resulting list.

4) Click on the [Create] button to create the module.

5) Click on [Close] if necessary to close the Select Module Type screen.



☑ Module Type Category Filters
☑ Analog
☑ CIP Motion Converter
☑ Communication
☑ Communications Adapter

■ Module Type Vendor Filters
□ Advanced Energy Industries, Inc.
☑ Advanced Micro Controls Inc. (AMCI)
□ Cognex Corporation
□ Dialight

Catalog Number	Description	Vendor	Category
AMCI-NR25-ENIP	AMCI-NR25-ENIP	Advanced Micro...	Encoder
AMCI-NR60-DLR	AMCI-NR60-DLR	Advanced Micro...	Encoder
NXAE2	AMCI NXAE2	Advanced Micro...	Communications /
NXEE2	AMCI NXEE2	Advanced Micro...	Communications /
NXFE2	AMCI NXFE2	Advanced Micro...	Communications /
SD17060E2	SD17060E2	Advanced Micro...	Communications /

12 of 508 Module Types Found

Figure T3.7 Selecting the NXFE2

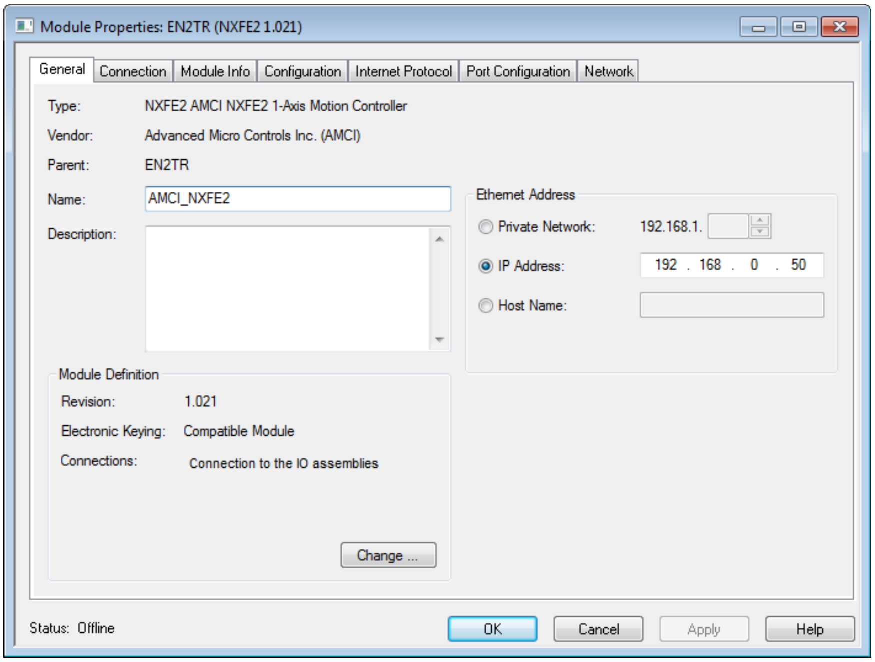
3.5 NXFE2 Properties

If you are continuing from step 3.4, the resulting New Module screen is used to configure the network connection between the NXFE2 and your controller. If you need to open the screen to perform this task at a later time, right click on the unit in the project tree and then select “Properties” from the drop-down menu

NOTE Tabs that are not listed in the steps below are filled with reasonable defaults by the EDS file.

3.5.1 General Tab

The Name, Description, and IP address of the device must be specified here. The [Change...] button allows you to change the Module Definition if needed.



Module Properties: EN2TR (NXFE2 1.021)

Type: NXFE2 AMCI NXFE2 1-Axis Motion Controller

Vendor: Advanced Micro Controls Inc. (AMCI)

Parent: EN2TR

Name: AMCI_NXFE2

Description:

Ethernet Address

○ Private Network: 192.168.1.

● IP Address: 192 . 168 . 0 . 50

○ Host Name:

Module Definition

Revision: 1.021

Electronic Keying: Compatible Module

Connections: Connection to the IO assemblies

Status: Offline

Figure T3.8 NXFE2 Properties Screen - EDS

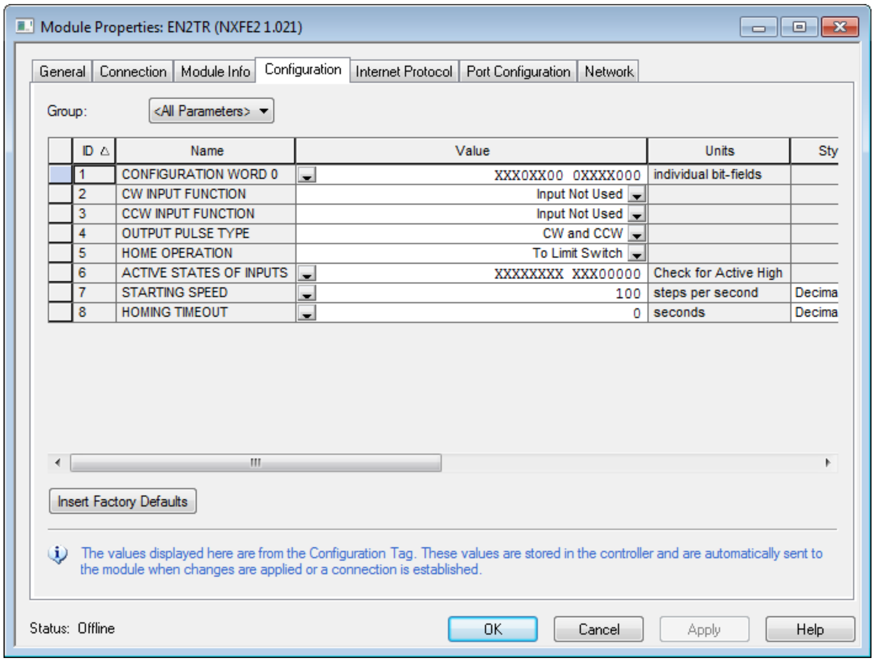
3.5.2 Connection Tab

The default RPI time is eight milliseconds. The minimum allowable time is two milliseconds. This value can be changed in this tab.

3.5 NXFE2 Properties (continued)

3.5.3 Configuration Tab

Use the Configuration tab to eliminate the need to configure the NXFE2 through the Output Words assigned to the module. This tab is used to define configuration data that is written down to the NXFE2 whenever it connects to the EtherNet/IP network. You can also click on the [Apply] button at any time to write the configuration data to the NXFE2.



Group: <All Parameters>

ID △	Name	Value	Units	Sty
1	CONFIGURATION WORD 0	XXX0XX00 0XXXX000	individual bit-fields	
2	CW INPUT FUNCTION	Input Not Used		
3	CCW INPUT FUNCTION	Input Not Used		
4	OUTPUT PULSE TYPE	CW and CCW		
5	HOME OPERATION	To Limit Switch		
6	ACTIVE STATES OF INPUTS	XXXXXXXX XXX00000	Check for Active High	
7	STARTING SPEED	100	steps per second	Decima
8	HOMING TIMEOUT	0	seconds	Decima

The values displayed here are from the Configuration Tag. These values are stored in the controller and are automatically sent to the module when changes are applied or a connection is established.

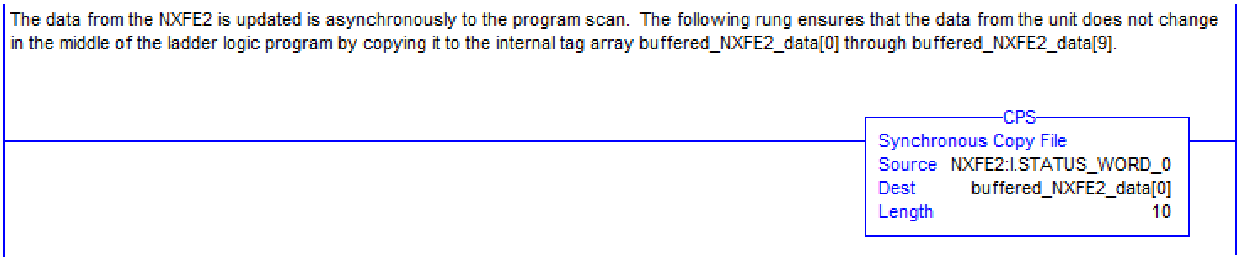
Status: Offline

Figure T3.9 EDS Configuration Screen Example

The EDS file defines tags that follow the format of the Configuration Data given in the ***Output Data Format*** section of reference chapter 5 starting on page 44. Refer to that section to determine the proper format of the parameter values.

3.6 Buffering the Input Data

Input data is updated asynchronously to the program scan. The input data should be buffered with Synchronous Copy File instructions to guarantee stable data during the program scan.


The data from the NXFE2 is updated is asynchronously to the program scan. The following rung ensures that the data from the unit does not change in the middle of the ladder logic program by copying it to the internal tag array buffered_NXFE2_data[0] through buffered_NXFE2_data[9].

CPS
Synchronous Copy File
Source NXFE2:I.STATUS_WORD_0
Dest buffered_NXFE2_data[0]
Length 10

Figure T3.10 Buffer Input Data

Implicit Communications Without an EDS

This chapter tells you how to configure implicit connections in EtherNet/IP systems that do not use EDS files. Refer to the next chapter for information on using message instructions for explicit messaging.

Rockwell Automation's RSLogix 5000 version 20+ software is used for the example installation in this chapter.

4.1 Host System Configuration

RSLogix 5000 is used to configure both the ControlLogix and CompactLogix platforms. When using these platforms, you will use a separate Ethernet Bridge module or an Ethernet port built into the processor.

If the Ethernet port is built into processor, the only step you have to take before adding the NXFE2 is to create a new project with the correct processor or modify an existing project. Once this is done, the Ethernet port will automatically appear in the Project Tree. If you are using an Ethernet bridge module, you will have to add it to the I/O Configuration tree before adding the unit to your project.

Refer to your Rockwell Automation documentation if you need instructions for configuring the ethernet port.

4.2 Add the NXFE2

You can add the NXFE2 to the project once the Ethernet port (built-in or bridge module) is configured.

1) Right click on the Ethernet port and then click on "New Module..." in the pop-up menu.

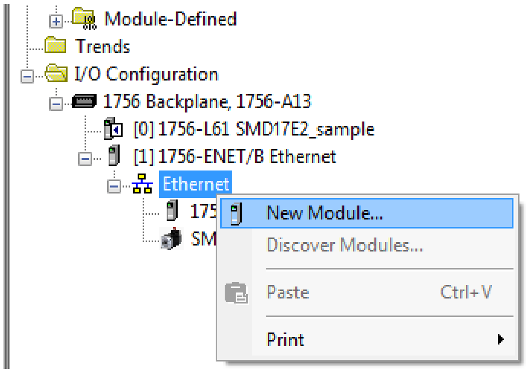
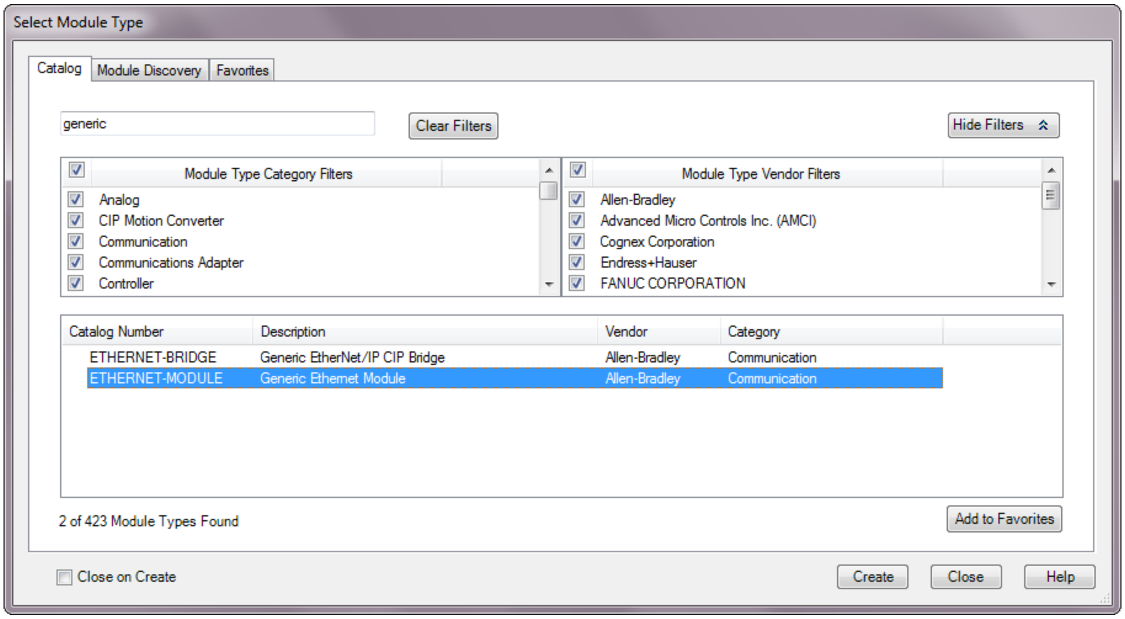


Figure T4.1 Adding an AMCI NXFE2

4.2 Add the NXFE2 (continued)

2) In the resulting Select Module Type screen, type “generic” into the filter as shown in figure T4.2. This will limit the results in the Catalog Number list.
3) Select the Catalog Number “ETHERNET-MODULE” in the list.
4) Click on the [Create] button to create the module.
5) Click on [Close] if necessary to close the Select Module Type screen.



generic

☑ Module Type Category Filters
☑ Analog
☑ CIP Motion Converter
☑ Communication
☑ Communications Adapter
☑ Controller

☑ Module Type Vendor Filters
☑ Allen-Bradley
☑ Advanced Micro Controls Inc. (AMCI)
☑ Cognex Corporation
☑ Endress+Hauser
☑ FANUC CORPORATION

Catalog Number	Description	Vendor	Category
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Allen-Bradley	Communication
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley	Communication

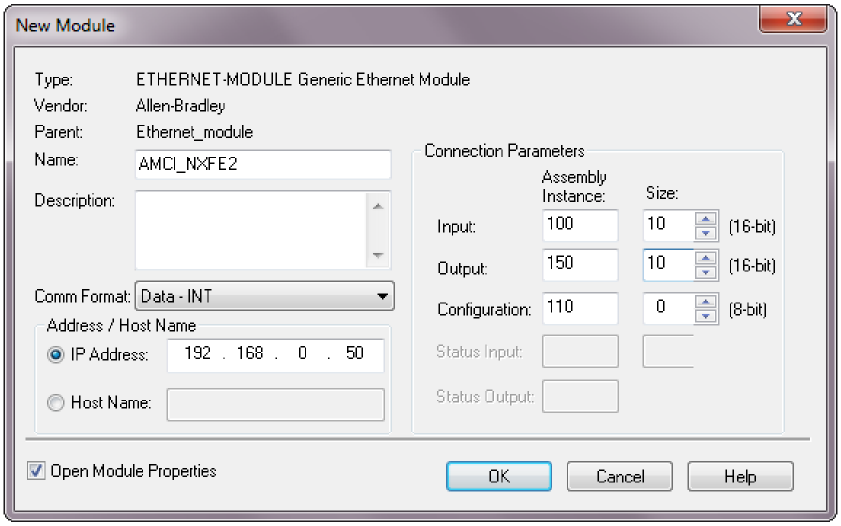
2 of 423 Module Types Found

☐ Close on Create

Figure T4.2 Selecting a Generic Device

4.2 Add the NXFE2 (continued)

6) Set the following parameters in the Module Properties window. All parameters not listed here are optional. Figure T4.3 shows a completed screen.



Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: Ethernet_module
Name: AMCI_NXFE2
Description:
Comm Format: Data - INT

Address / Host Name
● IP Address: 192 . 168 . 0 . 50
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	100	10	(16-bit)
Output:	150	10	(16-bit)
Configuration:	110	0	(8-bit)
Status Input:			
Status Output:			

☑ Open Module Properties

Figure T4.3 Configuration Screen - Generic Device

- **Name:** A descriptive name for the NXFE2
- **Comm Format:** Data - INT

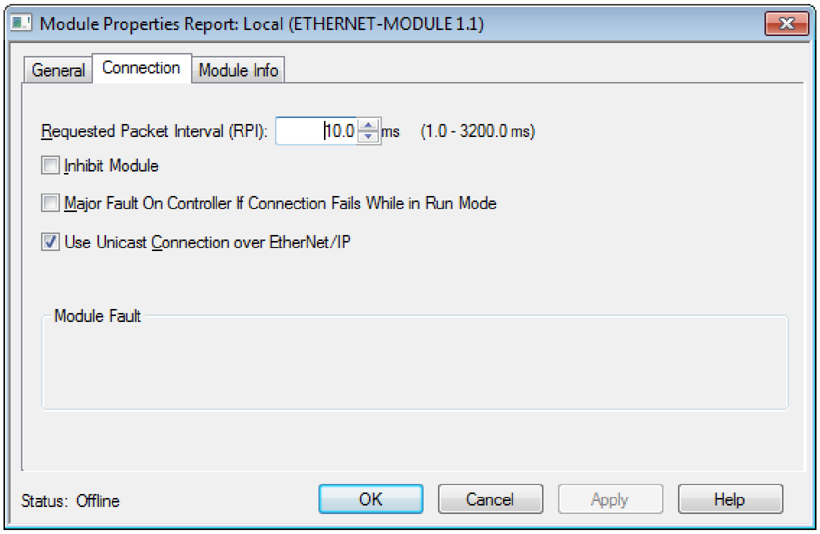
CAUTION The Comm Format defaults to Data - DINT. The NXFE2 will not be able to communicate with the host controller if this format is not changed when the device is added to the system. Once added, the Comm Format cannot be changed. The device must be deleted and added to the project again if the Comm Format is incorrect.

- **IP Address:** Must be the address you set for the NXFE2. Refer to the ***Set the IP Address and Protocol*** task chapter starting on page 83 for information on setting the IP Address of the unit.
- **Input:** Assembly Instance = 100, Size = 10 words.
- **Output:** Assembly Instance = 150, Size = 10 words.
- **Configuration:** Assembly Instance = 110, Size = 0 bytes.

7) Verify that the “Open Module Properties” check box is selected and click on [OK]. The Module Properties window will open.

4.2 Add the NXFE2 (continued)

8) Click on the "Connections" tab and set the RPI time that is required for your system. The suggested RPI time for an NXFE2 is eight milliseconds. The minimum RPI time is two milliseconds. The number of nodes on the network has an effect on the minimum RPI time. You may have to increase this RPI time if your network is heavily loaded. The remaining checkboxes can be left at their default settings. *The Unicast Connection field that is shown as checked in the figure is an optional setting.* When done, click on [OK] to complete the setup.


Module Properties Report: Local (ETHERNET-MODULE 1.1)

Requested Packet Interval (RPI): 10.0 ms (1.0 - 3200.0 ms)

□ Inhibit Module

□ Major Fault On Controller If Connection Fails While in Run Mode

☑ Use Unicast Connection over EtherNet/IP

Status: Offline

Figure T4.4 Setting the RPI Time

4.3 Check for Communication Errors

You can immediately check for errors if you are on-line with the processor while setting up the NXFE2. These error codes will appear in the Module Fault section of the Connection tab of the Module Properties window.

- **Error Code 16#0315 Connection Request Error: Invalid Segment Type.** You have specified an invalid assembly instance in the connection parameters.
- **No communications, but no error code.** Most commonly caused when the Comm Format not set to "Input Data - INT" or when the number of inputs words is incorrect.

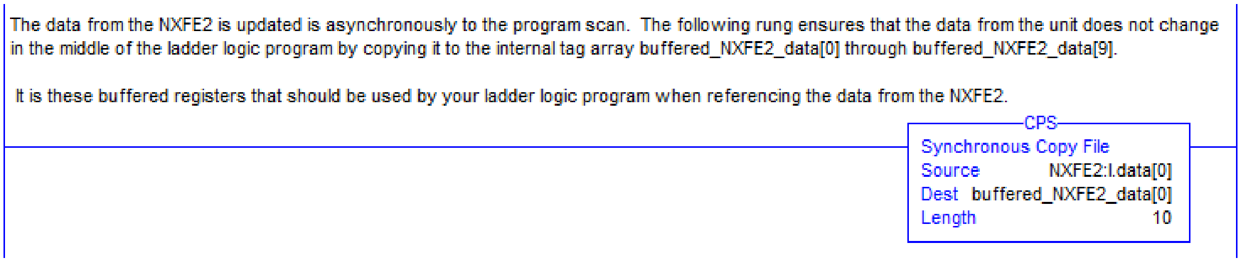
4.4 Configure the NXFE2

The device will join the EtherNet/IP network as soon as the request is made to it. The NXFE2 will report a Configuration Error until configuration data is written to the module. Configuration is accomplished by writing a block of data to the device that is formatted according to the specifications in the ***Output Data Format*** section of reference chapter 5 starting on page 44.

4.5 Buffer I/O Data

Data to and from the NXFE2 should be buffered once per scan using Synchronous Copy instructions. This is to insure stable input data during the program scan and guarantee that complete command data is delivered to the device. Ten word integer arrays can be used for this purpose.

These data tags should be buffered with Synchronous Copy File instructions to guarantee stable input data during the program scan and guarantee that complete command data is delivered to the device.



The data from the NXFE2 is updated is asynchronously to the program scan. The following rung ensures that the data from the unit does not change in the middle of the ladder logic program by copying it to the internal tag array buffered_NXFE2_data[0] through buffered_NXFE2_data[9].

It is these buffered registers that should be used by your ladder logic program when referencing the data from the NXFE2.

CPS
Synchronous Copy File
Source NXFE2:I.data[0]
Dest buffered_NXFE2_data[0]
Length 10

Figure T4.5 Buffer I/O Data

Notes

ETHERNET/IP EXPLICIT MESSAGING

All controllers that support EtherNet/IP support explicit messaging. When using explicit messaging, Message Instructions must be added to your program to communicate with the NXFE2. Explicit messaging can be use on platforms that also support implicit messaging.

Rockwell Automation controllers which are programmed with the RSLogix 500 software only support explicit messaging. A MicroLogix 1100 will be used as an example in this chapter.

5.1 Required Message Instructions

Only two instructions are required to transfer data between the PLC and the NXFE2. One instruction reads data from the unit and the other writes data to it. The sample programs available from AMCI use this style of programming. The two instructions are alternately triggered using the instruction's ENABLE bits. The remainder of the program controls when data in the source tags of the write instruction changes. The following table gives the required attributes for the instructions.

	Read Instruction	Write Instruction
Service Type	Read Assembly	Write Assembly
Service Code	E (hex)	10 (hex)
Class	4 (hex)	4 (hex)
Instance	100 (decimal)	150 (decimal)
Attribute	3 (hex)	3 (hex)
Length	20 bytes	20 bytes

Table T5.1 Message Instruction Attributes

NOTE Only RSLogix 500 version 8.0 or above can be used to configure Message Instructions to communicate with an EtherNet/IP device. Message Instructions do not work correctly in version 10 of RSLogix 500.

5.2 Create Four New Data Files.

- An Integer file to contain the data read from the NXFE2. This file must be at least 10 words in length.
- An Integer file to contain the data written to the NXFE2. This file must be at least 10 words in length.
- A Message (MG) data file. This file must have at least two elements, one to control the Read Operation and one to control the Write Operation.
- An Extended Routing Information (RIX) data file. This file is used to store information used by the Message Instructions. This file must have at least two elements, one for the Read Operation and one for the Write Operation.

5.3 Add the Message Instructions to your Ladder Logic

The following rungs show how you can alternately read data from and write data to the unit.

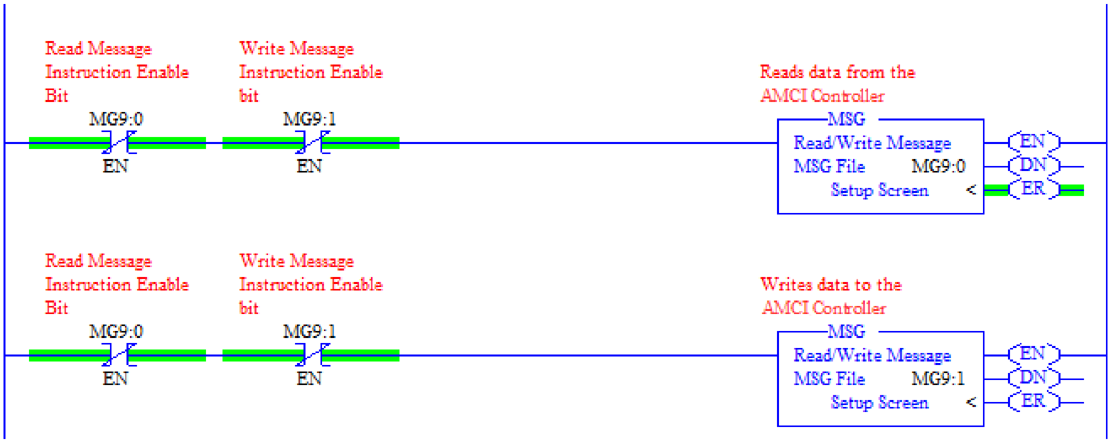
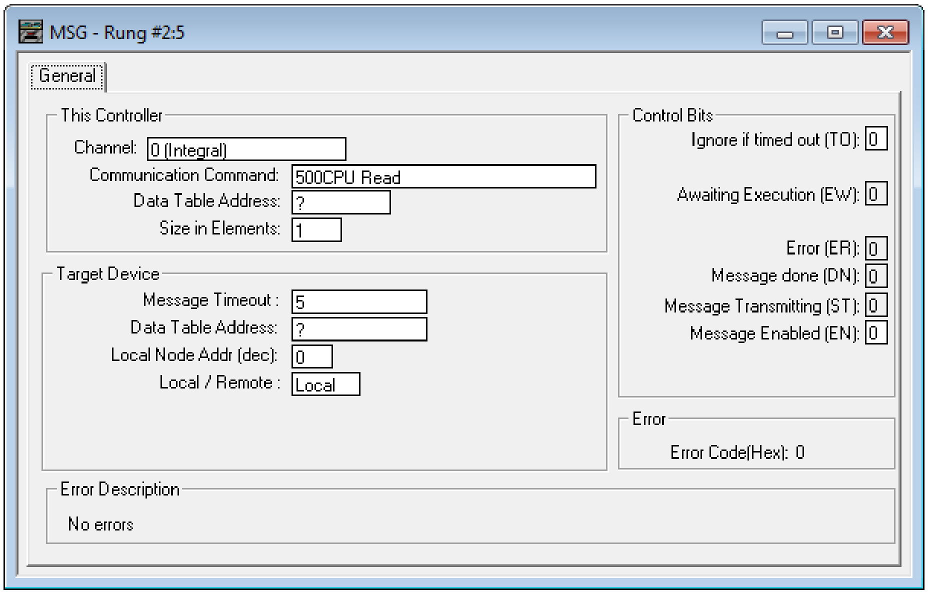

Read Message Instruction Enable Bit
MG9:0
EN
Write Message Instruction Enable bit
MG9:1
EN
Reads data from the AMCI Controller
MSG
Read/Write Message
MSG File MG9:0
Setup Screen <
EN
DN
ER
Read Message Instruction Enable Bit
MG9:0
EN
Write Message Instruction Enable bit
MG9:1
EN
Writes data to the AMCI Controller
MSG
Read/Write Message
MSG File MG9:1
Setup Screen <
EN
DN
ER

Figure T5.1 Message Instruction Example

1) Double click on *Setup Screen* text inside the Message Instruction. The following window will open. Note that this is the default window and its appearance will change considerably as you progress through these steps.



MSG - Rung #2:5

General

This Controller

Channel: 0 (Integral)

Communication Command: 500CPU Read

Data Table Address: ?

Size in Elements: 1

Target Device

Message Timeout : 5

Data Table Address: ?

Local Node Addr (dec): 0

Local / Remote : Local

Control Bits

Ignore if timed out (TO): 0

Awaiting Execution (EW): 0

Error (ER): 0

Message done (DN): 0

Message Transmitting (ST): 0

Message Enabled (EN): 0

Error

Error Code(Hex): 0

Error Description

No errors

Figure T5.2 Message Instruction Setup Screen

2) Double click in the *Channel* field, click on the ▼, select “1 (Integral)”, and press Enter.

3) Double click in the *Communication Command* field, click on the ▼, select “CIP Generic” and press Enter.

4) If the Message Instruction is being used to read data from the NXFE2, enter the integer file where the data will be placed in the *Data Table Address (Received)* field and press enter.

5) If the Message Instruction is being used to write data to the NXFE2, enter the integer file where the source data will be located in the *Data Table Address (Send)* field and press Enter.

6) Enter “20” as the number of bytes needed in either the *Size In Bytes (Receive)* or *Size In Bytes (Send)* fields. The NXFE2 requires 20 bytes for both Receive and Send.

7) Enter a RIX address in the *Extended Routing* Info field. Please note that each Message Instruction must have its own RIX address.

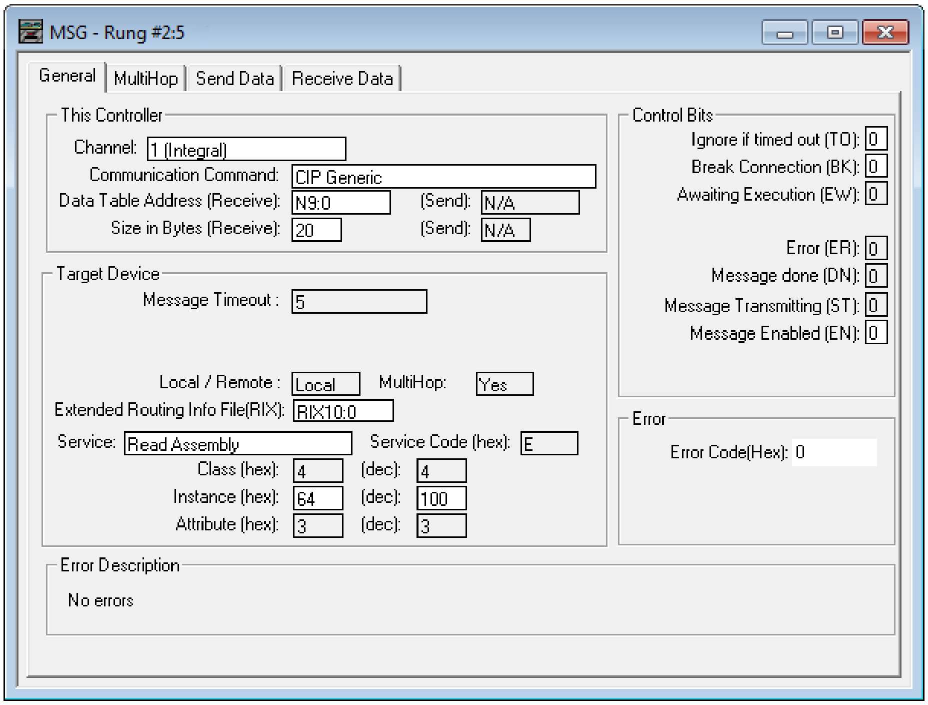
5.3 Add the Message Instructions to your Ladder Logic (continued)

8) Double click in the *Service* field and select "Read Assembly" for a Message Instruction that is being used to read data from the NXFE2, or "Write Assemble" for a Message Instruction that is being used to send data to the NXFE2, and press Enter.

9) For *Read* operations, the *Service Code* field will change to "E" (hex). For *Write* operations, the *Service Code* field will change to "10" (hex). For both read and write operations, the *Class* field will change to "4" (hex), and the *Attribute* field will change to "3" (hex).

10)For Read operations, enter a value of 100 decimal (64 hex) in the *Instance* field.
For Write operations, enter a value of 150 decimal (96 hex) in the *Instance* field.

The figure below show a typical configuration for Message Instructions being used to read data from the NXFE2. Please note that the Data Table Address (Receive) field may be different in your application.



This Controller

Channel: 1 (Integral)

Communication Command: CIP Generic

Data Table Address (Receive): N9:0 (Send): N/A

Size in Bytes (Receive): 20 (Send): N/A

Target Device

Message Timeout : 5

Local / Remote : Local MultiHop: Yes

Extended Routing Info File(RIX): RIX10:0

Service: Read Assembly Service Code (hex): E

Class (hex): 4 (dec): 4

Instance (hex): 64 (dec): 100

Attribute (hex): 3 (dec): 3

Control Bits

Ignore if timed out (TO): 0

Break Connection (BK): 0

Awaiting Execution (EW): 0

Error (ER): 0

Message done (DN): 0

Message Transmitting (ST): 0

Message Enabled (EN): 0

Error

Error Code(Hex): 0

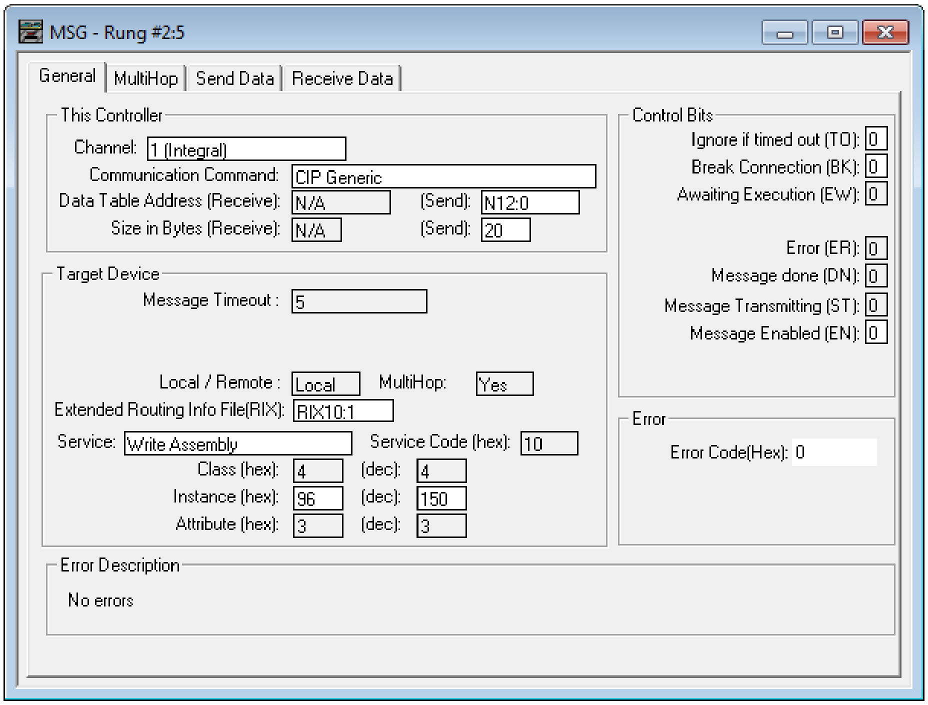
Error Description

No errors

Figure T5.3 Read Message Instruction Setup Screen

5.3 Add the Message Instructions to your Ladder Logic (continued)

The figure below show a typical configuration for Message Instructions being used to write data to the NXFE2. Please note that the Data Table Address (Send) field may be different in your application.



This Controller

Channel: 1 (Integral)

Communication Command: CIP Generic

Data Table Address (Receive): N/A (Send): N12:0

Size in Bytes (Receive): N/A (Send): 20

Target Device

Message Timeout : 5

Local / Remote : Local MultiHop: Yes

Extended Routing Info File(RIX): RIX10:1

Service: Write Assembly Service Code (hex): 10

Class (hex): 4 (dec): 4

Instance (hex): 96 (dec): 150

Attribute (hex): 3 (dec): 3

Control Bits

Ignore if timed out (TO): 0

Break Connection (BK): 0

Awaiting Execution (EW): 0

Error (ER): 0

Message done (DN): 0

Message Transmitting (ST): 0

Message Enabled (EN): 0

Error

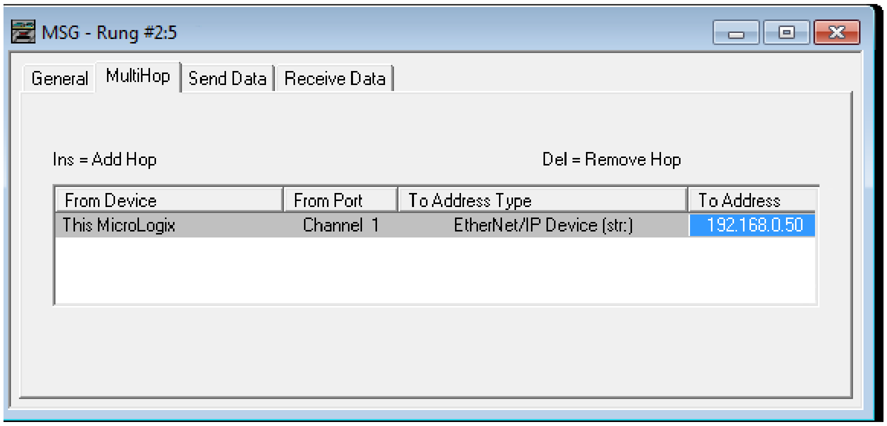
Error Code(Hex): 0

Error Description

No errors

Figure T5.4 Write Message Instruction Setup Screen

Click on the MultiHop tab on the top of the window. As shown in figure T5.5, enter the IP address of the NXFE2 and press Enter.



Ins = Add Hop Del = Remove Hop

From Device	From Port	To Address Type	To Address
This MicroLogix	Channel 1	EtherNet/IP Device (str.)	192.168.0.50

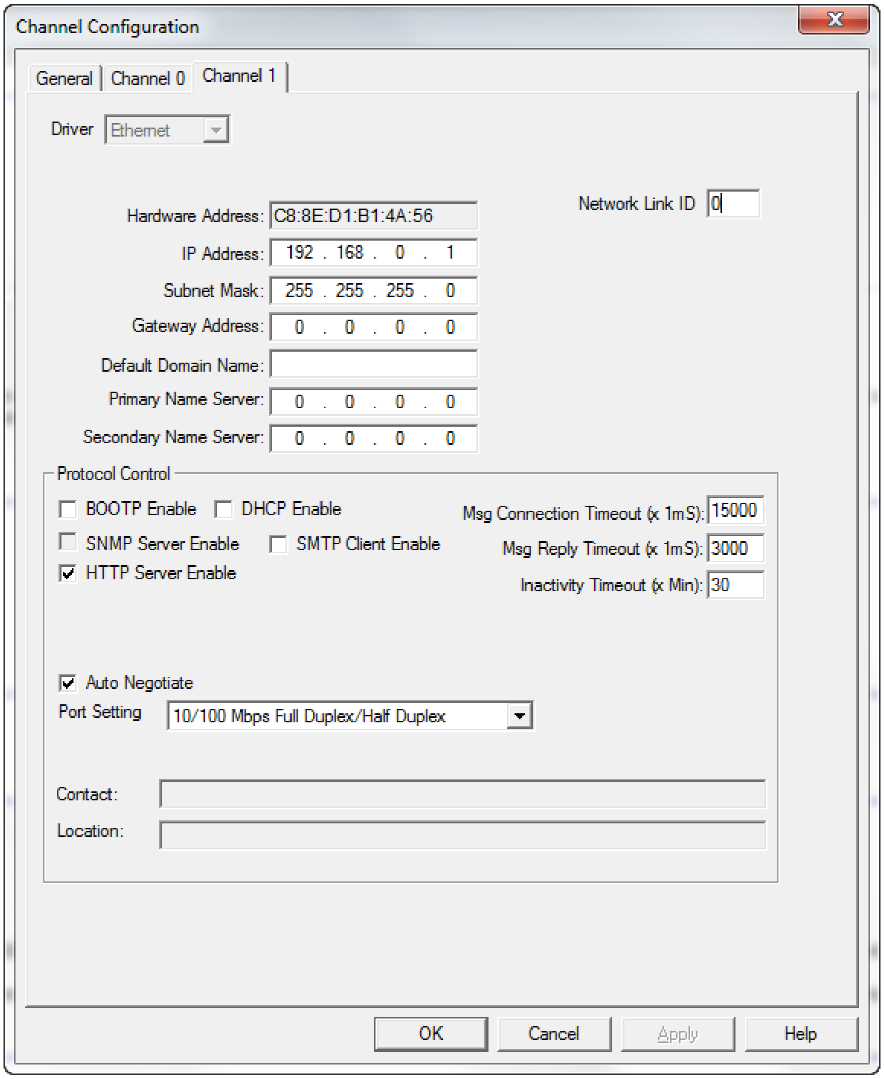
Figure T5.5 Message Instruction MultiHop Settings

After you are finished adding both the read and write message instructions to your program, save and download the program to the PLC.

5.4 Troubleshooting

If you are unable to communicate with the NXFE2, the problem may be that the Ethernet port of your MicroLogix 1100 has not been configured. To check this:

1) Double click on Channel Configuration in the Project Tree and then select the Channel 1 tab. The following window will open.



Driver: Ethernet

Hardware Address: C8:8E:D1:B1:4A:56

Network Link ID: 0

IP Address: 192 . 168 . 0 . 1

Subnet Mask: 255 . 255 . 255 . 0

Gateway Address: 0 . 0 . 0 . 0

Default Domain Name:

Primary Name Server: 0 . 0 . 0 . 0

Secondary Name Server: 0 . 0 . 0 . 0

Protocol Control

□ BOOTP Enable □ DHCP Enable

□ SNMP Server Enable □ SMTP Client Enable

☑ HTTP Server Enable

Msg Connection Timeout (x 1mS): 15000

Msg Reply Timeout (x 1mS): 3000

Inactivity Timeout (x Min): 30

☑ Auto Negotiate

Port Setting: 10/100 Mbps Full Duplex/Half Duplex

Contact:

Location:

Figure T5.6 MicroLogix Ethernet Configuration Screen

2) Enter the IP address and Subnet Mask of your MicroLogix 1100, (not the address of the NXFE2) and click on [Apply]. The Ethernet Port should now be working.

NOTE AMCI is aware of an issue with the RIX data type in version 10 of RSLogix 500. If you are experiencing communications errors and are running version 10, please contact Rockwell Automation for support.

Notes

MODBUS TCP CONFIGURATION

An AMCI NXFE2 that has been configured for the Modbus TCP protocol requires a host controller to issue configuration data and read data from the unit. This chapter tell you how the I/O words used by an AMCI NXFE2 are mapped to the Modbus I/O registers.

Enable Modbus TCP Protocol

The internal web page utility can be used to change the communications protocol used by the NXFE2. This is typically done while setting the IP address. Specifically, follow the steps in section 2.2b, ***Use the Embedded Web Server*** which starts on page 84.

Modbus Addressing

The register addresses used in this manual are the *Modbus logical reference numbers*†, which are unsigned integers starting at zero. This is often called *zero based* addressing. In this scheme, the first register is given an address of zero. This is the actual addressing scheme used in the Modbus packets.

Another common addressing scheme is *one based* or *data model* addressing. In this scheme, the register's number is used as its address, so the first register, Register 1 in the data model, has an address of 1.

Modbus Table Mapping

The Discrete Input and Input Register tables in the Modbus data model map to the same physical memory locations in the NXFE2 units.

- These registers hold data that is reported from the NXFE2 to the host controller.
- Addresses for these registers and inputs start at 0 in zero based addressing.

As examples:

- Discrete Input 0 is the same memory location as bit 0 of the first Input Register.
- Register address 3, the fourth register, contains Discrete Inputs 48 through 63.

The Coil and Holding Register tables in the Modbus data model map to the same physical memory locations in the NXFE2 units.

- These registers hold data that is from the host controller to the unit. This data is typically commands.
- Addresses for these registers start at 1024 in zero based addressing. Coil addresses start at 16,384 in zero based addressing (1024*16).

As examples:

- Coil 16384 is the same memory location as bit 0 of the first Holding Register.
- Register address 1025, the address of the second Holding Register, contains Discrete Inputs 16,400 through 16,415 in zero based addressing.

Host Addressing

Your host controller may not use these basic addressing schemes for communicating over a Modbus connection. For example, Modicon controllers use addresses starting at 30000 for Input Registers and addresses starting at 40000 for Holding Registers. GE hosts use their %R memory for Holding Registers and %AI memory for Input Registers.

In these cases, when you add the NXFE2 to your host controller you must define a mapping between your host controller's addressing scheme and the zero based Modbus TCP addresses. Refer to your host controller's documentation for information on how to accomplish this.

† MODBUS Application Protocol Specification V1.1b3, section 4.3: MODBUS Data model. ***www.modbus.org***

AMCI Modbus TCP Memory Layout

The NXFE2 has a starting Input Register address of 0 and a starting Output Register address of 1024. Input Registers hold the data from the unit while Output Registers hold the data to be written to the unit. Figure T6.1 shows how an NXFE2 is mapped to the Modbus data reference.

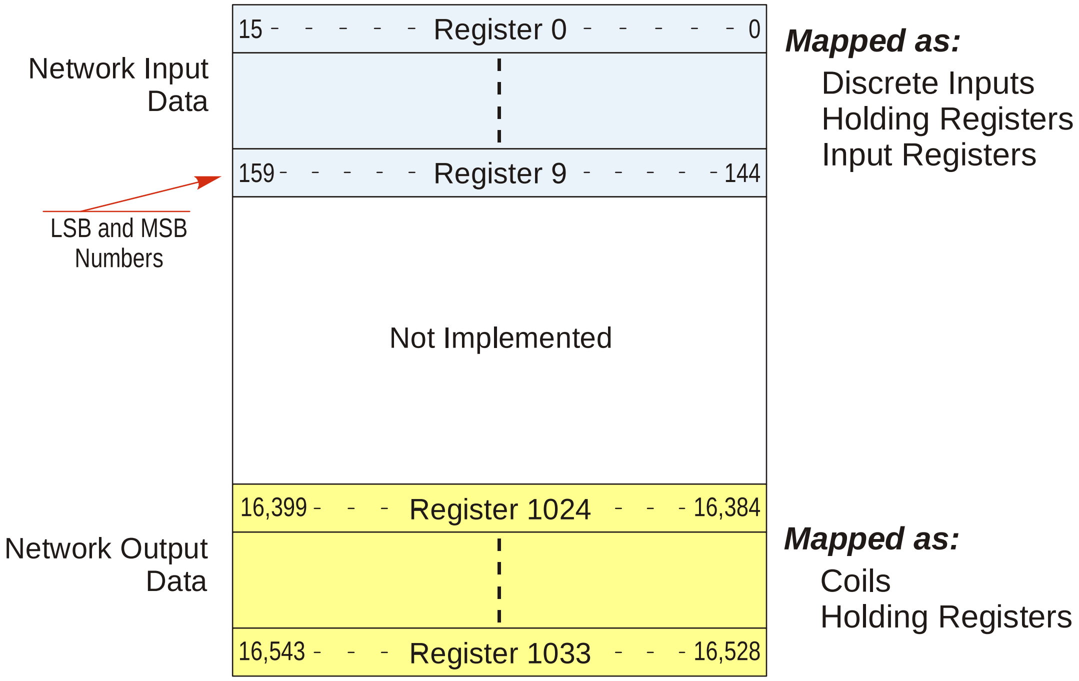


Figure T6.1 Modbus Data Reference Map

The complete specification for the Modbus protocol can be downloaded at http://www.modbus.org/specs.php.

Supported Number of Connections

All NXFE2 units support five concurrent connections. When connections exist, the Network Status (NS) LED on the front of the unit will flash green. The number of blinks indicate the number of active connections. There is a two second break between groups of flashes.

Supported Modbus Functions

Function Code	Function Name	NXFE2 Register	Addressing method
1	Read Coils	OUTPUT	Bit:Addresses starting at 16,384
2	Read Discrete Inputs	INPUT	Bit:Addresses starting at 0
3	Read Holding Registers	OUTPUT & INPUT	Word: Out Regs.Starting at 1024 In Regs.Starting at 0
4	Read Input Registers	INPUT	Word:Addresses starting at 0.
5	Write Single Coil	OUTPUT	Bit:Addresses starting at 16,384
6	Write Single Register	OUTPUT	Word:Addresses starting at 1024
15	Write Multiple Coils	OUTPUT	Bit:Addresses starting at 16,384
16	Write Multiple Registers	OUTPUT	Word:Addresses starting at 1024
22	Mask Write Register	OUTPUT	Word:Addresses starting at 1024
23	Read/Write Registers	INPUT/OUTPUT	Word: Out Regs.Starting at 1024 In Regs.Starting at 0

Table T6.1 Supported Modbus Functions

Table T6.1 above lists all of the Modbus functions supported by an NXFE2. AMCI supports all of these functions so that you can control the unit as you see fit. However, if you are looking for the easiest way to interface with your unit, then you only need to use the *Read/Write Registers* function, which is function code 23.

NOTE ➔ Each NXFE2 buffers the data that is sent to it over the network. If you use the *Read/Write Registers* function to write data to the unit, the data read with that command will not contain the response to the new write data. The response to the new data will be sent with the next data read.

Supported Modbus Exceptions

Code	Name	Description
01	Illegal function	The NXFE2 does not support the function code in the query.
02	Illegal data address	The data address received in the query is outside the initialized memory area.
03	Illegal data value	The data in the request is illegal.

Table T6.2 Supported Modbus Exceptions

Notes

PROFINET Network Configuration

This chapter outlines the steps commonly needed to get an NXFE2 communicating with the PROFINET master. A Siemens SIMATIC S7-1212C controller is used in this example.

Basic Steps

Configuring a PROFINET host requires a few basic steps.

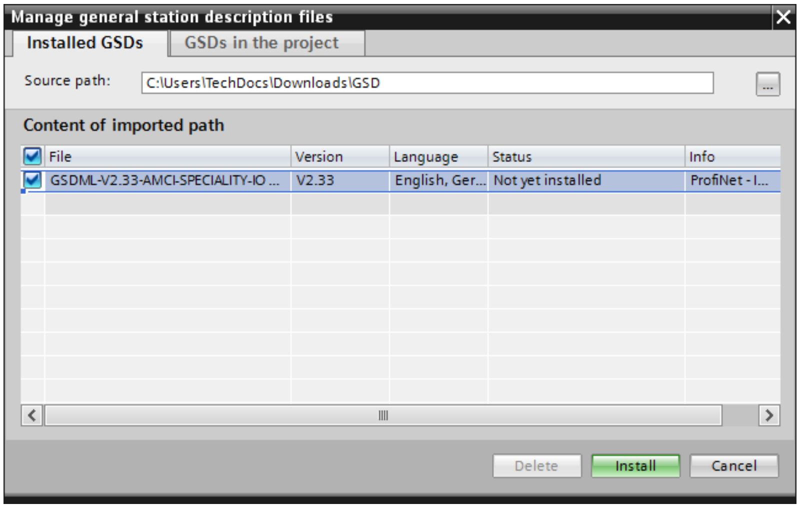
1) Download the ZIP archive that contains the GSDML files for the NXFE2 from the www.amci.com website.
2) Install the GSDML file into the configuration software for your host controller.
3) Add the NXFE2 to the PROFINET Network.
4) Set the I/O word addresses used to communicate with the unit.

7.1 Download the GSDML files

The GSDML files are available on the AMCI website on the ***http://www.amci.com/industrial-automation-support/configuration-files/*** web page. The file is a ZIP archive that has to be extracted to a folder on your computer. Extracting the ZIP file will leave you with multiple files. One is the GSDML file and the others are icon files for the various devices.

7.2 GSDML File Installation

1) Open or create a new project that will include the NXFE2 and open the Project View of the project.
2) In the menu, select *Options -> Manage general station description files (GSD).*
3) In the window that opens, click on the [...] button and navigate to the folder that contains the extracted GSDML file you downloaded from the AMCI website. Once at the folder, click on the [OK] button.
4) Click on the check box next to the name of the GSD file and click on the [Install] button. The system will install the GSD file.
5) Click the [Close] button and wait for the software to finish installing the file and updating the Hardware Catalog.



Source path: C:\Users\TechDocs\Downloads\GSD

Content of imported path

☑ File	Version	Language	Status	Info
☑ GSDML-V2.33-AMCI-SPECIALITY-IO ...	V2.33	English, Ger...	Not yet installed	ProfiNet - I...

Figure T7.1 GSD File Installation

7.3 Configure the PROFINET Network

A CPU must be added to the project and the PROFINET network must be configured before an NXFE2 can be added to the system.

Refer to Siemens documentation for information on configuring the PROFINET network to suit your application.

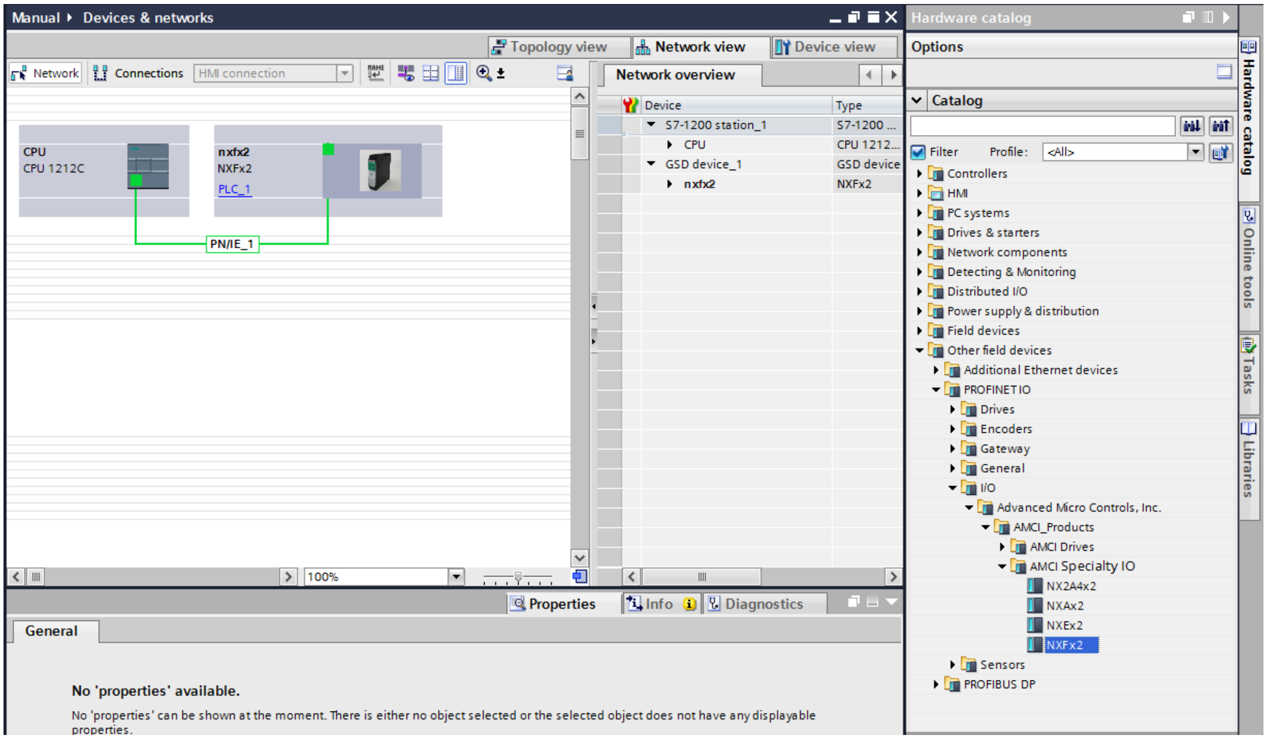
7.4 Add the NXFE2 to the PROFINET Network

1) With the project open in Project View, double click on “Device & Networks” in the project tree.

2) If need be, click on the “Hardware Catalog” vertical tab to open the Hardware Catalog.

3) You can search for “NXF”, or browse to the NXFx2 icon by clicking through *Other field devices +> PROFINET IO +> IO +> Advanced Micro Controls Inc. +> AMCI_Products +> AMCI_Speciality_IO*. Drag and drop the appropriate icon onto the PROFINET network.

4) Drag the green square on the NXFx2 icon onto the PROFINET network line to connect the device to the network.



Device	Type
S7-1200 station_1	S7-1200 ...
CPU	CPU 1212...
GSD device_1	GSD device
nxfx2	NXFx2

No 'properties' available.

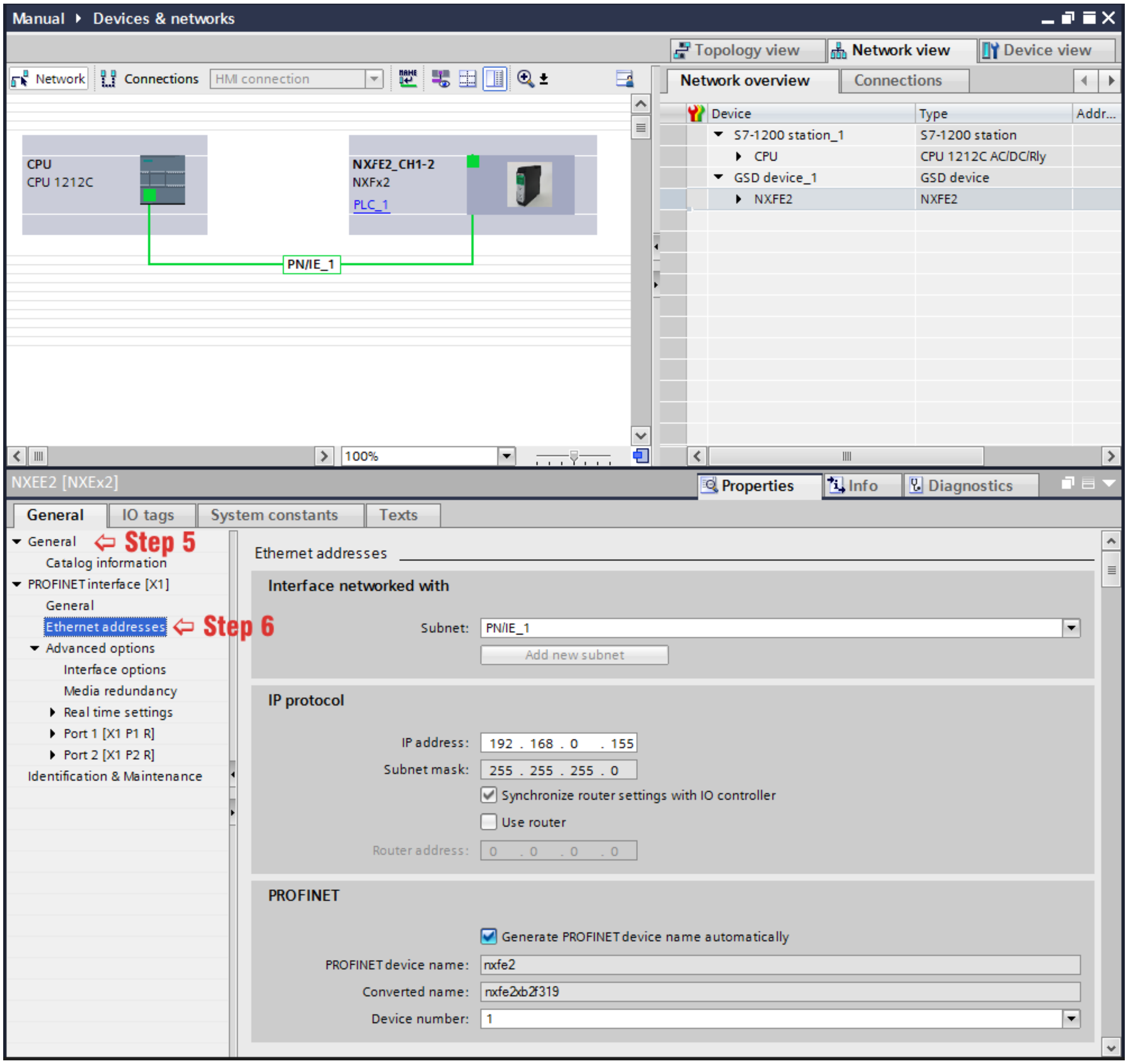
No 'properties' can be shown at the moment. There is either no object selected or the selected object does not have any displayable properties.

Figure T7.2 NXFE2 Added to PROFINET Network

7.4 Add the NXFE2 to the PROFINET Network (continued)

5) Right click on the NXFx2 icon and select “Properties” from the pop up menu. The Inspector window will open at the bottom of the screen. Under the “General” tab, select the “►General” heading. You can rename the NXFE2 by changing the Name: field.

6) Under the “►PROFINET interface [x1]” heading, select “Ethernet addresses”. Under the IP protocol section, set the desired IP address and subnet mask for the NXFE2.



Network overview | Connections

Device	Type	Addr...
▼ S7-1200 station_1	S7-1200 station	
▸ CPU	CPU 1212C AC/DC/Rly	
▼ GSD device_1	GSD device	
▸ NXFE2	NXFE2	

Ethernet addresses

Interface networked with

Subnet: PN/IE_1

IP protocol

IP address: 192 . 168 . 0 . 155

Subnet mask: 255 . 255 . 255 . 0

☑ Synchronize router settings with IO controller

☐ Use router

Router address: 0 . 0 . 0 . 0

PROFINET

☑ Generate PROFINET device name automatically

PROFINET device name: nxfe2

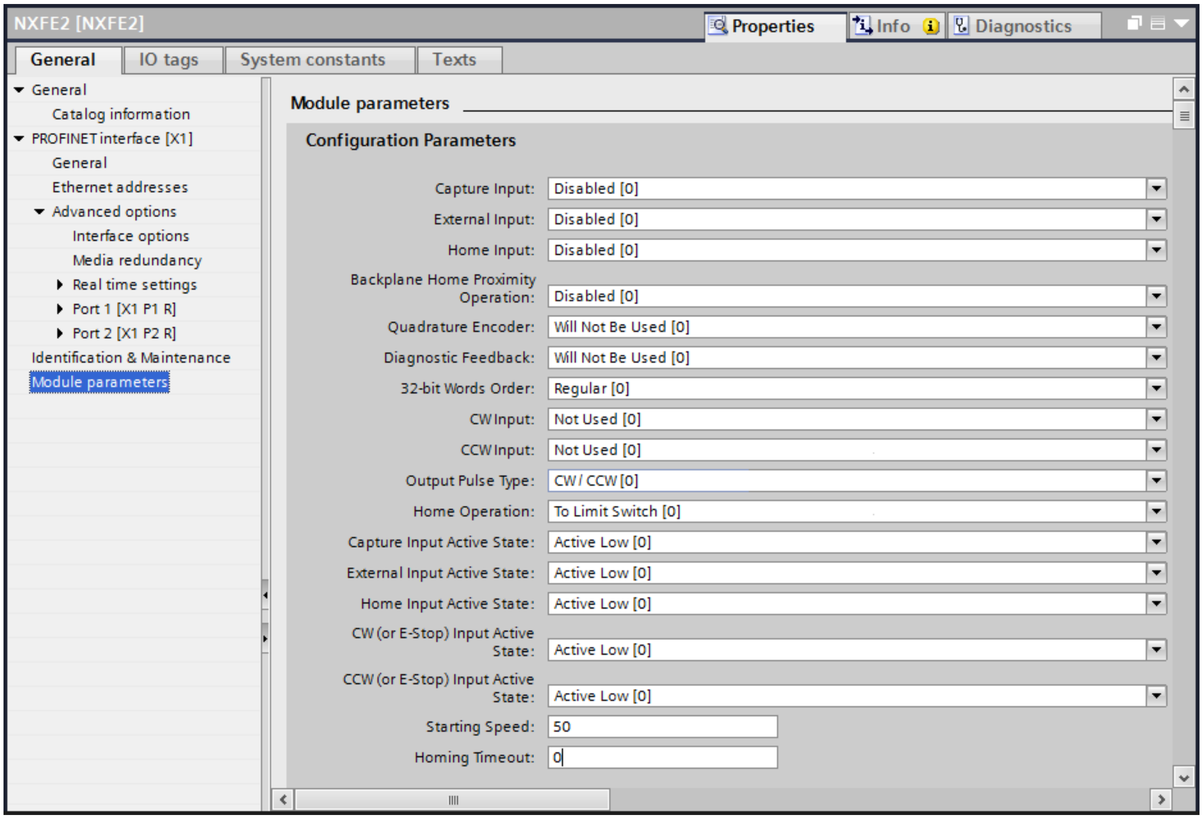
Converted name: nxfe2xb2f319

Device number: 1

Figure T7.3 NXFE2 IP Addressing

7.5 Configure the NXFE2

1) Continuing with the Inspector Window, click on the "Module parameters" heading. The display will change to show all of the parameters of the NXFE2. These configuration parameters are written down to the NXFE2 whenever it connects to the network.



Module parameters

Configuration Parameters

Capture Input: Disabled [0]
External Input: Disabled [0]
Home Input: Disabled [0]
Backplane Home Proximity Operation: Disabled [0]
Quadrature Encoder: Will Not Be Used [0]
Diagnostic Feedback: Will Not Be Used [0]
32-bit Words Order: Regular [0]
CW Input: Not Used [0]
CCW Input: Not Used [0]
Output Pulse Type: CW / CCW [0]
Home Operation: To Limit Switch [0]
Capture Input Active State: Active Low [0]
External Input Active State: Active Low [0]
Home Input Active State: Active Low [0]
CW (or E-Stop) Input Active State: Active Low [0]
CCW (or E-Stop) Input Active State: Active Low [0]
Starting Speed: 50
Homing Timeout: 0

Figure T7.4 Configuration Parameters

2) Configuration is done with a combination of drop down menus and value fields. Drop down menus limit you to a set of valid selections. When entering numbers into value fields, a pop up will appear that shows you the valid range of values.

NOTE Even though your selections are limited, it is still possible to misconfigure the NXFE2 by using this screen. If an error occurs, the NXFE2 will show that a configuration error exists after it connects to the network. The ***Invalid Configurations*** section on page 48 shows the configuration bit patterns and parameter values that will generate an error.

NOTE This data is only written to the unit when it connects to the network, which is typically on power up. The unit can still be reconfigured at any time by writing the appropriate data to the output registers of the NXFE2.

NOTE These configuration parameters cannot be changed while on line. To make any changes, you must go off line, make the changes, save the project, download it, and then go back on line.

7.6 Set the I/O Configuration

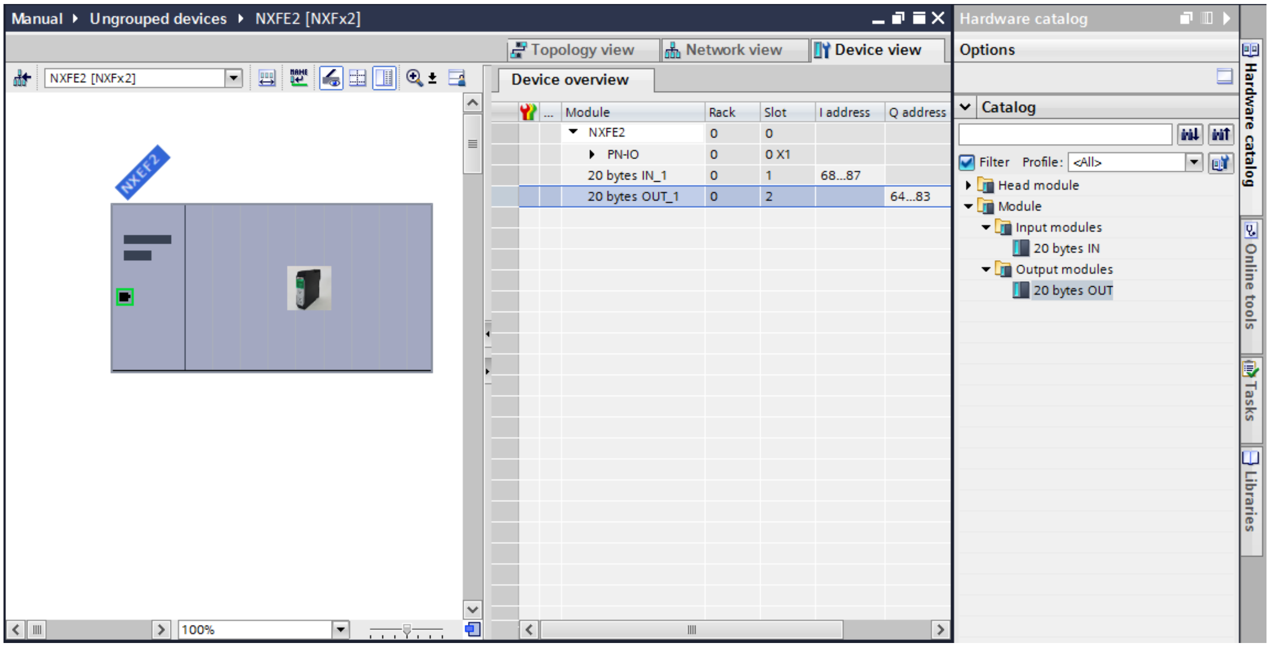
The NXFE2 units require 10 Input Words (20 Input Bytes) and 10 Output Words (20 Output Bytes). All required Input and Output Bytes are defined by the GSDML file and divided into suitable modules. These settings are shown in the Table T7.1.

Input / Output Bytes of an NXFE2	Input / Output Modules of an NXFE2
20 Input Bytes	Input Module - Slot 1: 20 bytes
20 Output Bytes	Output Module - Slot 2: 20 bytes

Table T7.1 PROFINET I/O Configuration

1) With the NXFx2 icon selected on the PROFINET bus, click on the “Device view” tab. The view in the Hardware Catalog will change. Expand the Module tree to show both the Input and Output modules.

2) To map the I/O bytes to the CPU, double click on the “20 bytes IN” and “20 bytes OUT” icons in the Hardware Catalog. The system will automatically assign the next I and Q addresses to the data table.



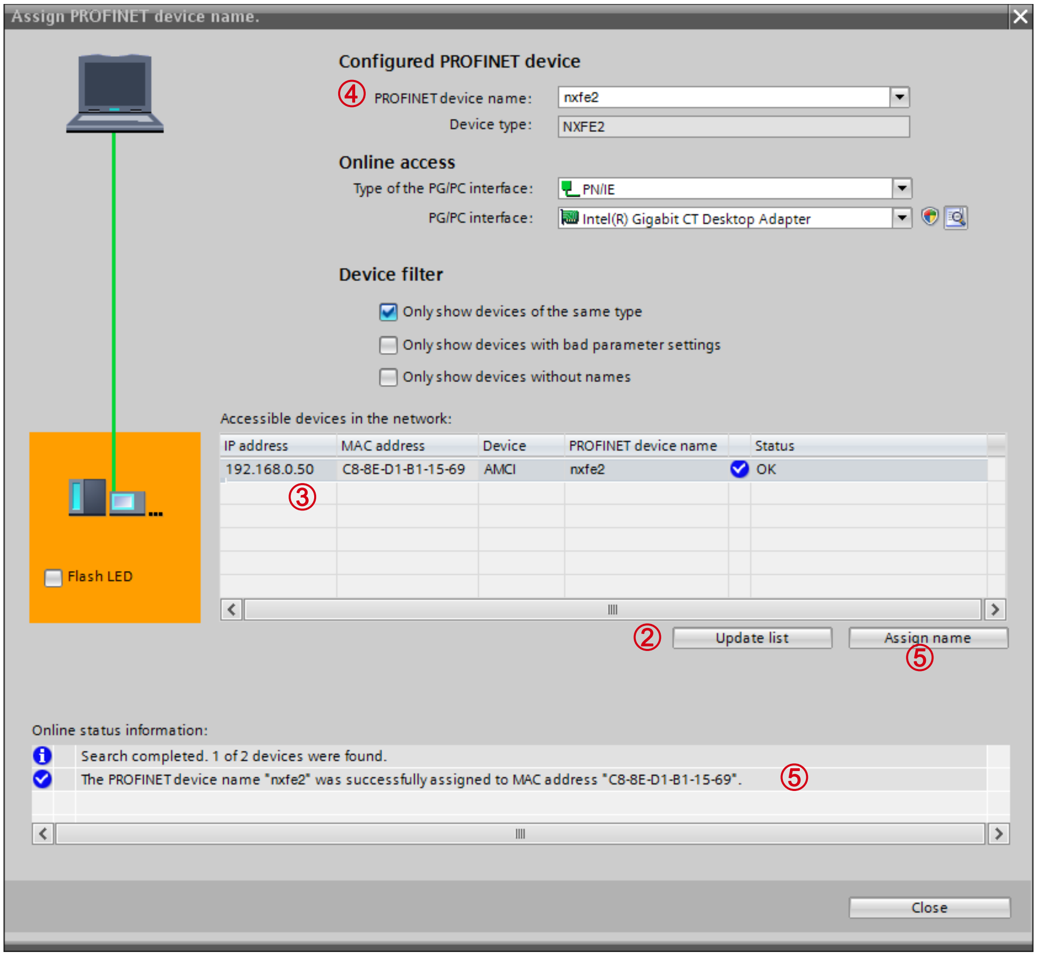
Device overview

Module	Rack	Slot	I address	Q address
NXFE2	0	0		
PN-IO	0	0 X1		
20 bytes IN_1	0	1	68...87	
20 bytes OUT_1	0	2		64...83

Figure T7.5 I/O Byte Mapping

7.7 Set the NXFE2 Device Name

1) Right click on the NXFE2 icon and select “Assign device name” from the resulting popup menu.
2) In the “Assign PROFINET device name.” windows that appears, click on the [Update list] button.
3) Once the NXFE2 appears in the table, select the NXFE2 in the table.
4) Confirm that the “PROFINET device name:” at the top of the screen is correct.
5) Click on the [Assign name] button to write the device name to the NXFE2. The “Online status information:” table at the bottom of the screen will show that the name was successfully assigned to the MAC address of the NXFE2.



Assign PROFINET device name.

Configured PROFINET device

④ PROFINET device name: nxfe2

Device type: NXFE2

Online access

Type of the PG/PC interface: PN/IE

PG/PC interface: Intel(R) Gigabit CT Desktop Adapter

Device filter

☑ Only show devices of the same type

☐ Only show devices with bad parameter settings

☐ Only show devices without names

Accessible devices in the network:

IP address	MAC address	Device	PROFINET device name	Status
192.168.0.50	C8-8E-D1-B1-15-69	AMCI	nxfe2	✓ OK

Online status information:

- Search completed. 1 of 2 devices were found.
- ✓ The PROFINET device name "nxfe2" was successfully assigned to MAC address "C8-8E-D1-B1-15-69". ⑤

Figure T7.6 Setting NXFE2 Device Name

7.8 Verify and Download the New Configuration

1) Continue by adding any remaining devices to your PROFINET network.
2) Compile and download the project to the CPU.

MRP Installations

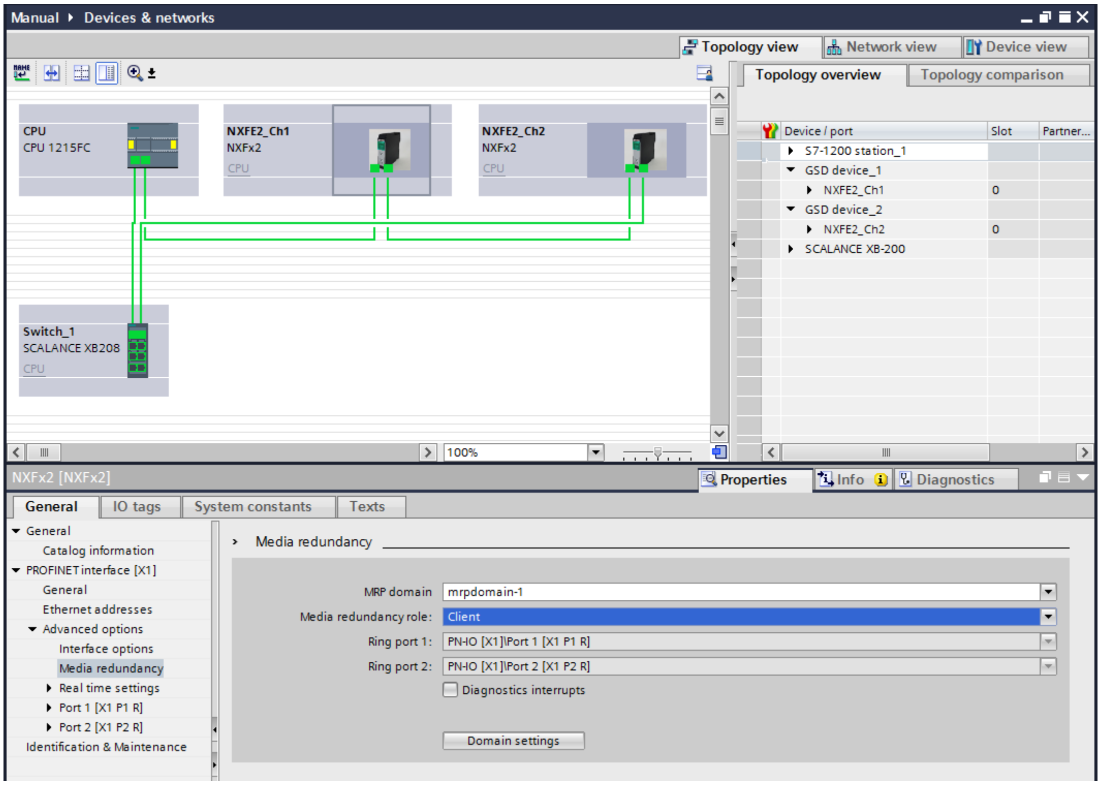
At this point, the NXFE2 is configured and ready to use. If you are using the unit in a redundant, ring based, network that uses the Media Redundancy Protocol (MRP), continue with the following instructions.

> **Media Redundancy Protocol (MRP) installations require that the NXFE2 be installed in a ring topology. In these applications, both Ethernet ports are used when wiring the ring, daisy chaining from one unit in the ring to the next. The steps below covers typical software configuration that must also be completed.**

7.9 Configure the NXFE2 as an MRC

The NXFE2 functions as a Media Redundancy Client (MRC) in an MRP network.

1) Switch to Topology view and drag the additional connections between the appropriate ports.
2) Click on the NXFx2 icon to select it. In the Inspector window, select *Advanced options +> Media redundancy*. Use the “MRP domain:” drop down menu to select the appropriate domain. Use the “Media redundancy role:” drop down menu to select “Client”.
3) Continuing in the Inspector window, select *Advanced options +> Port 1 +> Port interconnection*. Under “Partner port:”, the partner port you assigned to the port when you drew the topology is shown. If you do not know which port will be the partner port in the actual installation, you can use the drop down menu to select “Any partner”.
4) If need be, repeat step 3 for Port 2 of the NXFE2.



Device / port	Slot	Partner...
S7-1200 station_1		
GSD device_1		
NXFE2_Ch1	0	
GSD device_2		
NXFE2_Ch2	0	
SCALANCE XB-200		

Media redundancy

MRP domain: mrpdomain-1
Media redundancy role: Client
Ring port 1: PN-IO [X1]\Port 1 [X1 P1 R]
Ring port 2: PN-IO [X1]\Port 2 [X1 P2 R]
□ Diagnostics interrupts

Figure T7.7 MRP Topology and Client settings

5) Continue configuring the rest of the devices on the network before compiling the project and downloading it to the CPU.

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