NORDBO ROBOTICS

NRS-6 Force & Torque Sensor

User Manual

Version 1.9

Copyright

Disclaimer

The information provided in this documentation contains general descriptions of use, of one or more of Nordbo Robotics' products. Following this documents instruction for installation and use must not be considered a substitute for making a risk assessment and testing the specific application and setup. Neither Nordbo Robotics nor any of our affiliates or subsidiaries shall be held responsible for the consequences that arise from misuse of – or misunderstanding of – any information stated in this document.

If any instructions are considered unclear it is the user's responsibility to contact Nordbo Robotics for clarification of the issue(s) before making use of said unclear instructions.

If the user has any suggestions for improvements, amendments or if errors have been found in this release, please notify Nordbo Robotics.

Nordbo Robotics reserves the right to make refinements or changes without notice.

Warranty

Warranty void if products are opened. The warranty further excludes failure resulting from improper use or installation, normal wear and tear, abuse, accidents, neglect, fire, water, lightning or other acts of nature, causes external to the product or other factors beyond Nordbo's control.

Contact

Nordbo Robotics A/S
Noatunvej 2
5000 Odense, Denmark

Phone: +45 81 81 98 81
Mail: info@nordbo-robotics.com

Glossary

The following table explains the terms used throughout the manual.

TERMS	EXPLANATION
NRS-6	General product name of Nordbo Robotics' range of force and torque sensors measuring force and torque in 6-axis.
FT sensor / sensor	Used when referring to the NRS-6 sensor.
NRC-DAC	Digital to Analog Conversion, expansion module for NRC-ETH – enables the NRC-ETH to transmit analog values of force and torque.
NRC-ETH	CAN to Ethernet converter module – reads data from NRS-6 and outputs it through analogue outputs or Ethernet. Serves as a user interface of the NRS-6. Sometimes referred to as signal converter or converter.
Application	A "System" designed for solving the problem of automating an action or process.
FB	Short for "Function block", an instantiated program type in PLCs.
FBD	Short for "Function block diagram", a programming language in IEC 61131-3 standard.

Table of Contents

Copyright 1
Disclaimer 1
Warranty 1
Contact 1
TERMS 2
EXPLANATION 2
1. Introduction 3
1.1. Intended use 3
1.2. Prerequisites 3
1.3. Safety Warnings 3
2. Product information 5
2.1. NRS-6 Standalone 5
2.2. (Optional) NRC-ETH 5
2.3. NRS-6 Technical specifications 5
2.4. Mechanical specifications: NRS-6XXX-D80 7
2.5. Mechanical specifications: NRS-6050-D50 7
2.6. (Optional) NRC-ETH/DAC Converter 7
Technical Specifications 8
3. (Optional) Configuring the NRC-ETH 11
Step 1 11
Step 2 11
Step 3 11
Step 4 11
Step 3 13
3.1. Configuring the DAC setup 15
Step 1 15
Step 2 15
Step 3 15
4. NRS-6 using Ethernet 15
The following section describes how to use the NRS-6 with an ethernet connection. The user is required to have purchased the optional NRC signal converter to follow this guide. 15
4.1. Connecting using ethernet 16
Step 1 16
Step 2 16
4.2. Communicating using ethernet 18
Data types 18
Communication protocol 18
If an error occurs when handling a request 19

4.3. Ethernet Commands 19
SENSOR_TRANSMIT 0x07 20
SENSOR_TRANSMIT_DATA 0x06 20
SENSOR_SET_CURRENT_TARE 0x15 20
SENSOR_SET_DATA_RATE 0x08 21
SENSOR_SET_DAC_SPAN 0x21 21
ERROR 0x45 22
5. NRS-6 using DAC 23
5.1. Connecting using DAC 23
5.2. Reading from the NRS-6 using DAC 24
6. NRS-6 using CAN 26
6.1. Connect using CAN 26
6.2. Communicate using CAN 26
Polling messages 27
Set CAN address 27
Set transmission mode 28
Set CAN data rate 28
Set current tare 29
Periodic messages 29
CMD_BROADCAST 29
7. Using the NRS-6 with Codesys 30
For the Ethernet functionality 30
For DAC functionality 30
Step 1 31
Step 2 31
Step 3 32
Step 4 32
Step 5 32
Step 6 32
7.1. How to configure the function blocks 32
Schneider Electric Modicon and PacDrive controllers 32
Step 1 32
Step 2 33
Step 3 33
Step 4 33
Step 5 33
Step 6 35
Step 7 35
Step 8 35
Ethernet function block inputs 35

DAC function block inputs 38
7.2. Description of the function blocks' output 40
Ethernet function block outputs 40
DAC function block outputs 41
8. Troubleshooting 41
8.1. Resetting the NRC 41
Soft reset 42
Hard reset 42
8.2. If the NRC does not start when connected to power: 42
8.3. If the NRC is powered on, but there is no data: 43
8.4. If NRC is powered on, sensor is connected, but there is no data: 44
8.5. If the NRC is on, but the user or application cannot connect: 45
8.6. If your problem not listed: 47
8.7. Maintenance 47
8.8. Repairs 47
9. Declarations and Certificates 48

1. Introduction

1.1. Intended use

This document serves as a guide for integrating Nordbo Robotics' NRS-6 force and torque sensor in an application using any controller, with supported analog or ethernet interface.

1.2. Prerequisites

It is recommended that the person executing the instructions of this installation guide has basic knowledge and skill in using analog and/or ethernet interfaces. It is also recommended that the reader should have basic knowledge in using a computer networking. It is required that the person executing the instructions within this guide has made a risk assessment of the application in which the product is integrated before using this manual.

1.3. Safety Warnings

Following instructions must be read thoroughly by anyone intending to use this product.

Warning! This symbol indicates that potentially hazardous, dangerous, or unwanted situations can arise from not following the instructions correctly. If safety instructions are not followed properly it may result in death, personnel injury, or equipment damage.

Any controller must always be **powered off** before attempting to connect wires to it, or when connecting cables between any of Nordbo Robotics' products.

The product may be damaged if dropped. Be aware that M4/M8 connectors the NRS-6 and the NRC can break if overtightened.

The NRS-6 is water and dust resistant according to IP67 test standards.

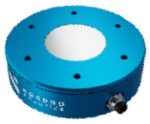
The sensor can be used within the specified measurement range. Using the sensor outside of its range is considered misuse. Nordbo Robotics is not liable for any damage or injury resulting from misuse.

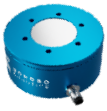
2. Product information

2.1. NRS-6 Standalone

The product consists of the items listed below.

1x NRS-6050-D80 / 1x NRS-6200-D80 / 1x NRS-6050-D50





1x NRC-ETH 3m sensor cable

1x USB key with relevant software

2.2. (Optional) NRC-ETH

1x NRC-ETH - CAN to Ethernet converter

1x NRC-ETH 2m power cable

2.3. NRS-6 Technical specifications

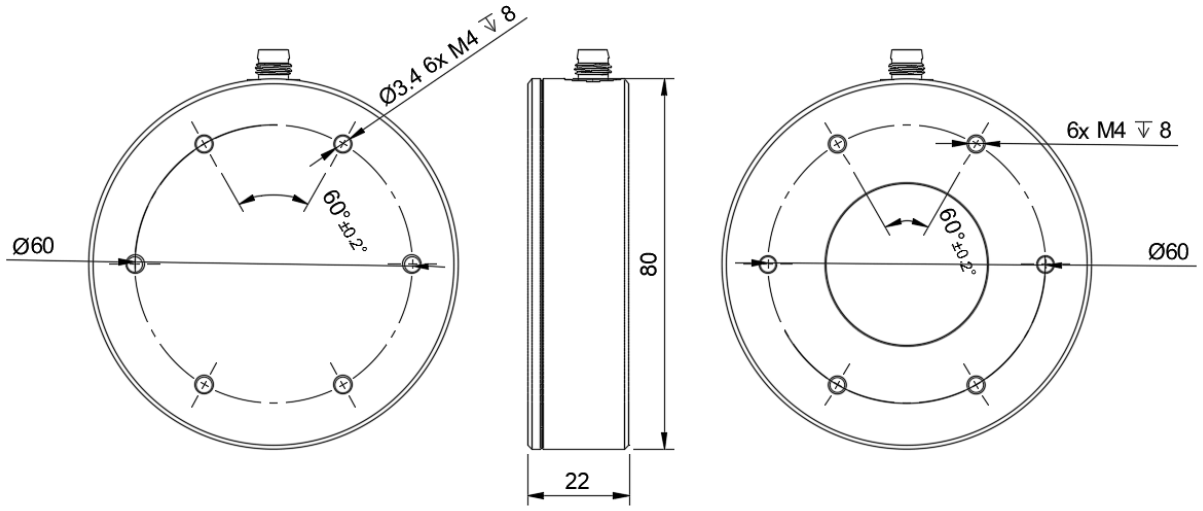
	NRS-6050-D50	NRS-6050-D80	NRS-6200-D80
Dimensions (Diameter x Height)	50 x 22 mm	80 x 22 mm	80 x 22 mm
Weight	100 g	300 g	300 g
Software	UR, AUBO & Codesys		
Hysteresis	< 0,2 %		
Crosstalk	< 5 %		
Maximum sampling frequency	1000 Hz		
Operating temperature	0° to 50°		
Operating humidity	< 85%		
Power requirement (CAN)	5 VDC @ 250 mA		
Power requirement (Ethernet)	6-40 VDC @ 1000 mA		
	Fx, Fy	**Fx, Fy**	**Fx, Fy**
Max force	± 500 N	± 500 N	± 2000 N
Resolution*	0,015 N	0,015 N	0,005 N
Overload**	± 700 N	± 700 N	± 2700 N
Signal noise**	0,025 N	0,025 N	0,056 N
Noise-free resolution	0,1 N	0,1 N	0,25 N
Full scale non-linearity	< 4%	< 4%	< 4%
Axis deformation	30 µm	30 µm	30 µm
	Fz	**Fz**	**Fz**
Max force	± 500 N	± 500 N	± 2000 N
Resolution*	0,015 N	0,015 N	0,005 N
Overload**	± 1200 N	± 1200 N	± 4700 N
Signal noise**	0,032 N	0,032 N	0,085 N
Noise-free resolution	0,15 N	0,15 N	0,35 N

Full scale non-linearity	< 4%	< 4%	< 4%
Axis deformation	27 µm	27 µm	27 µm
	Tx, Ty	**Tx, Ty**	**Tx, Ty**
Max torque	± 10 Nm	± 10 Nm	± 40 Nm
Resolution*	$0{,}32*10^{-3}$ Nm	$0{,}32*10^{-3}$ Nm	$1{,}2*10^{-3}$ Nm
Overload**	± 15 Nm	± 15 Nm	± 55 Nm
Signal noise**	$0{,}35*10^{-3}$ Nm	$0{,}35*10^{-3}$ Nm	$0{,}8*10^{-3}$ Nm
Noise-free resolution	$1{,}5*10^{-3}$ Nm	$1{,}5*10^{-3}$ Nm	$3{,}0*10^{-3}$ Nm
Full scale non-linearity	< 4%	< 4%	< 4%
Axis deformation	0,2°	0,2°	0,2°
	Tz	**Tz**	**Tz**
Max torque	± 5 Nm	± 5 Nm	± 30 Nm
Resolution*	$0{,}25*10^{-3}$ Nm	$0{,}25*10^{-3}$ Nm	$0{,}95*10^{-3}$ Nm
Overload**	± 15 Nm	± 15 Nm	± 55 Nm
Signal noise**	$0{,}4*10^{-3}$ Nm	$0{,}4*10^{-3}$ Nm	$0{,}95*10^{-3}$ Nm
Noise-free resolution	0,002 Nm	0,002 Nm	$3{,}5*10^{-3}$Nm
Full scale non-linearity	< 4%	< 4%	< 4%
Axis deformation	0,05°	0,05°	0,05°

** Signal noise is defined as the standard deviation of a typical one-second no-load signal*
*** The sensor is not overload protected and may break if forces are not kept below the overload values for each axis.*

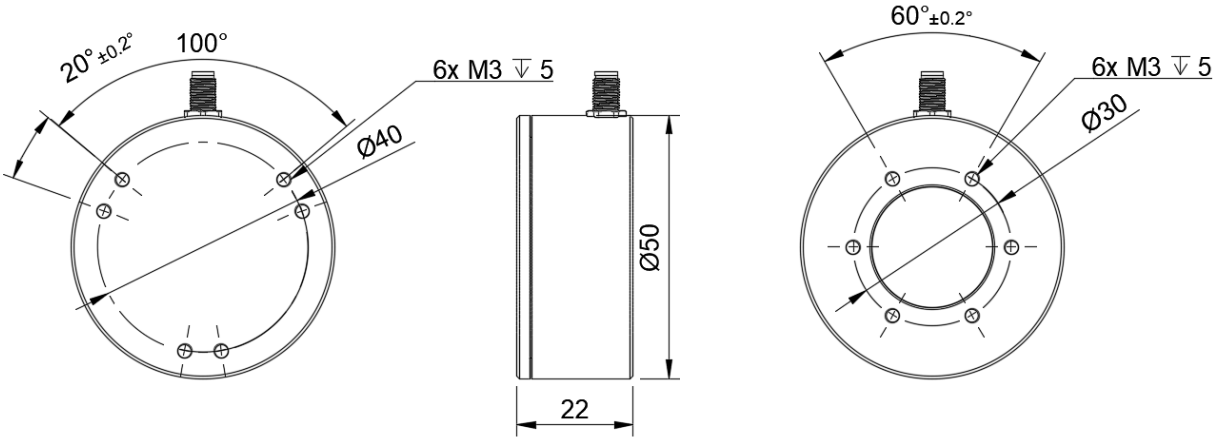
2.4. Mechanical specifications: NRS-6XXX-D80

Bottom view **Top view**



2.5. Mechanical specifications: NRS-6050-D50

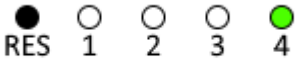
Bottom view **Top view**



2.6. (Optional) NRC-ETH/DAC Converter

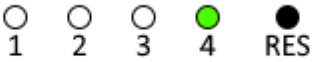
The NRC-ETH/DAC signal converter has 4 LEDs displaying its operating status and a button for resetting the firmware without connecting the device. The LEDs are numbered 1-4, and the reset button marked “RES”. On the left side, a 3-pin connection is used to power the converter. If the converter is the DAC version the right side is further supplied with an 8-pin connector for a DAC connection. On the backside, an RJ45 connection is available for ethernet.

For version: NRC-ETH



ID	Description	Functionality
RES	Reset Button	Restores device to factory settings. Press and hold the reset button until all LEDs have turned on and off again. Doing so will delete all user-defined settings on the device permanently and reset the IP address to default!
1	White LED	Blinks when the firmware of the converter is running correctly
2	White LED	Turned on when the NRS-6 is connected to the converter and is running without errors
3	White LED	Blinks when an Ethernet data stream has been started externally and is running
4	Green LED	Turned on when power is on

For version: NRC-ETH-2

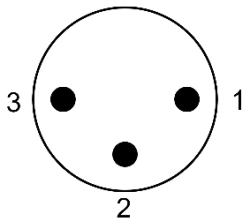


ID	Description	Functionality
RES	Reset Button	Restores device to factory settings. Press and hold the reset button until all LEDs have turned on and off again. Doing so will delete all user-defined settings on the device permanently and reset the IP address to default!
1	White LED	Turned on when the NRS-6 is connected to the converter and is running without errors
2	White LED	Blinks when an Ethernet data stream has been started externally and is running
3	White LED	Blinks when the firmware of the converter is running correctly
4	Green LED	Turned on when power is on

Technical Specifications

Model	NRS-ETH
Compatibility	Windows, Linux, UR & Codesys
Width, W	55 mm
Length, L	91 mm (incl. connectors)
Height, H	39 mm
Weight	115 g
Operating temperature	0° to 50°
Operating humidity	< 85 %
Power requirement (CAN)	5 VDC @ 100 mA
Power requirement (Ethernet)	6-40 VDC @ 500 mA

The power connector of the NRC is a 3-pin connector with 2 wires attached – one red and one black. Pin number 2 is not connected.



Warning! A wrong connection of the power supply may destroy the NRC permanently. Always make sure that the power is properly connected.

Wire color	Pin number	Power outlet connector
Black	1	Ground
Not connected	2	-
Red	3	Between +6 to +40 volts DC

3. (Optional) Configuring the NRC-ETH

The following section describes how to configure the NRC-ETH/DAC using a web interface. Some configuration can be necessary before using the NRC-ETH/DAC in an application. This is done through the convert's web interface. The interface can be accessed by connecting to the IP address of the NRC using an internet browser. The following section elaborates on how this is done.

OBS.: The user is required to have purchased the optional NRC-ETH or NRC-DAC signal converter to follow this guide.

Step 1

Plug the converter into a pc using an Ethernet cable. If no switch or router is available, connect the converter and computer directly using an Ethernet cable.

Step 2

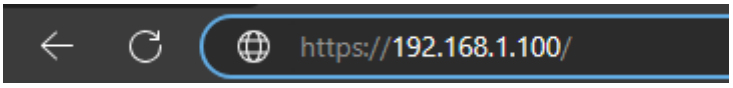
Plug a computer into the same switch or router using another Ethernet cable.

Step 3

Configure the computer with static IP-address in the range of the converter.

Step 4

Access the converters web interface through any browser by typing in the signal converters IP address in the browser's address bar (By default, the IP address is "192.168.1.100").

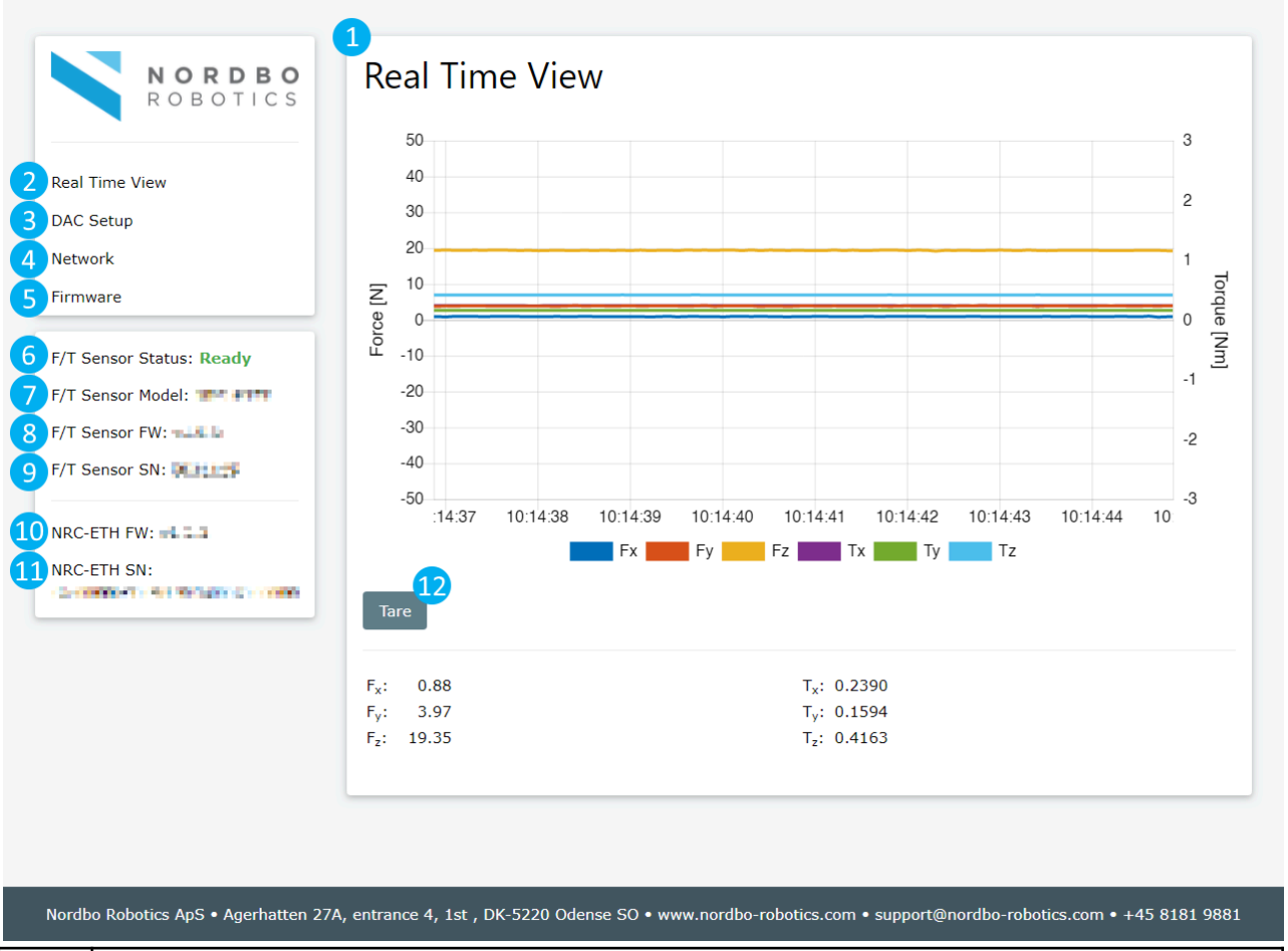


When entering the web interface of the converter, the layout will be displayed as seen on the figure below.

Note:

The latest converter box has default IP 192.168.**1**.100

The older converter box has default IP 192.168.**0**.100

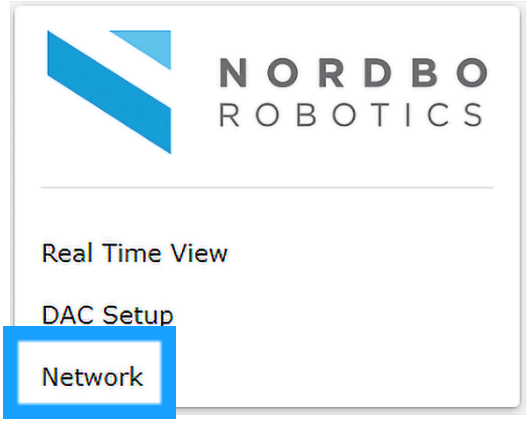


No.	Description
1	Real-Time View screen showing the output from the NRS-6 in real-time
2	The button navigates to the Real-Time View menu
3	The button navigates to the DAC Setup menu
4	The button navigates to the Network menu
5	The button navigates to the Firmware menu
6	Indicates if NRS-6 is connected and is ready to operate
7	Indicates which sensor model is currently connected to the converter
8	Displays which firmware version is currently active on the connected NRS-6
9	Displays the serial number of the NRS-6 currently connected

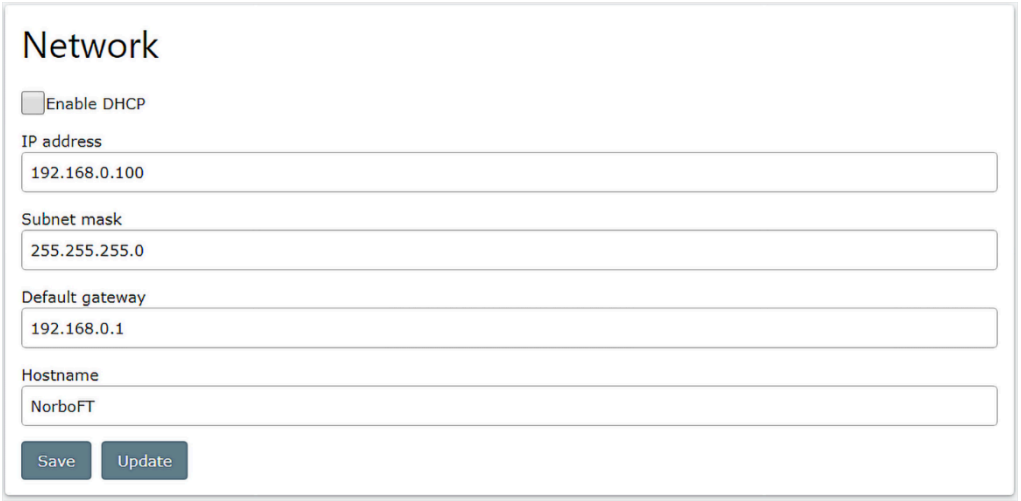
10	Displays the firmware version of the converter NRC currently logged into
11	Displays serial number of the NRC-ETH currently logged into
12	Clicking “Tare” will offset all measurements by the forces currently measured by the NRS-6, effectively setting force and torque measurements to 0.0 N/Nm

Step 3

Using the web interface, select “Network” in the navigation menu to the left.



Inside the network menu you have the option to change several settings. The network menu is displayed below



Enable DHCP: Select if application should use a DHCP-server and the signal converter’s hostname for the connection. Select or deselect feature by checking the box
IP Address: Change the signal converter’s IP address to an IP address in the controller’s local network (E.g. Controller: “192.168.1.10”, NRC-ETH: “192.168.1.11”)
Subnet mask: Change the NRC-ETH’s subnet mask to match the subnet mask of the controller’s network (E.g. Both: IP address “192.168.1.X”, subnet mask: “255.255.255.0”)
Default gateway: Change the default gateway e.g. “192.168.1.1”
Hostname: Define the hostname e.g. “NordboFT”

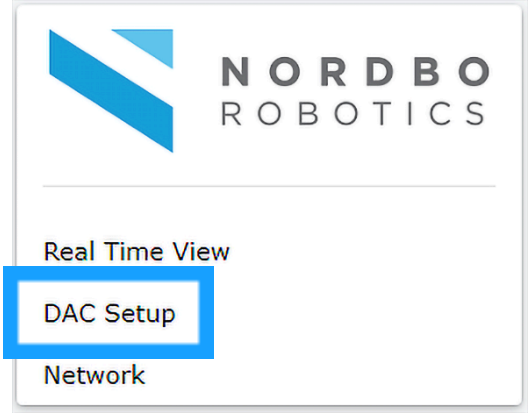
When the desired changes have been made, press “Save” to update the NRC-ETH according to the new settings.

OBS.: If the NRC-ETH’s address was changed the user will have to re-access the NRC-ETH’s web interface by entering the new IP address to make any further settings.
OBS.: Pressing “Update” will replace any assigned values with the values that are currently stored and active on the NRC-ETH.

3.1. Configuring the DAC setup

Step 1

When inside the web interface, select “DAC Setup” in the interface’s navigation menu to the left.



Selecting “DAC Setup” in the navigation menu will bring the user to the following page. Here the user can configure the DAC module.
Note: Changes made in this menu will only affect the NRC-DAC version of the signal converter.

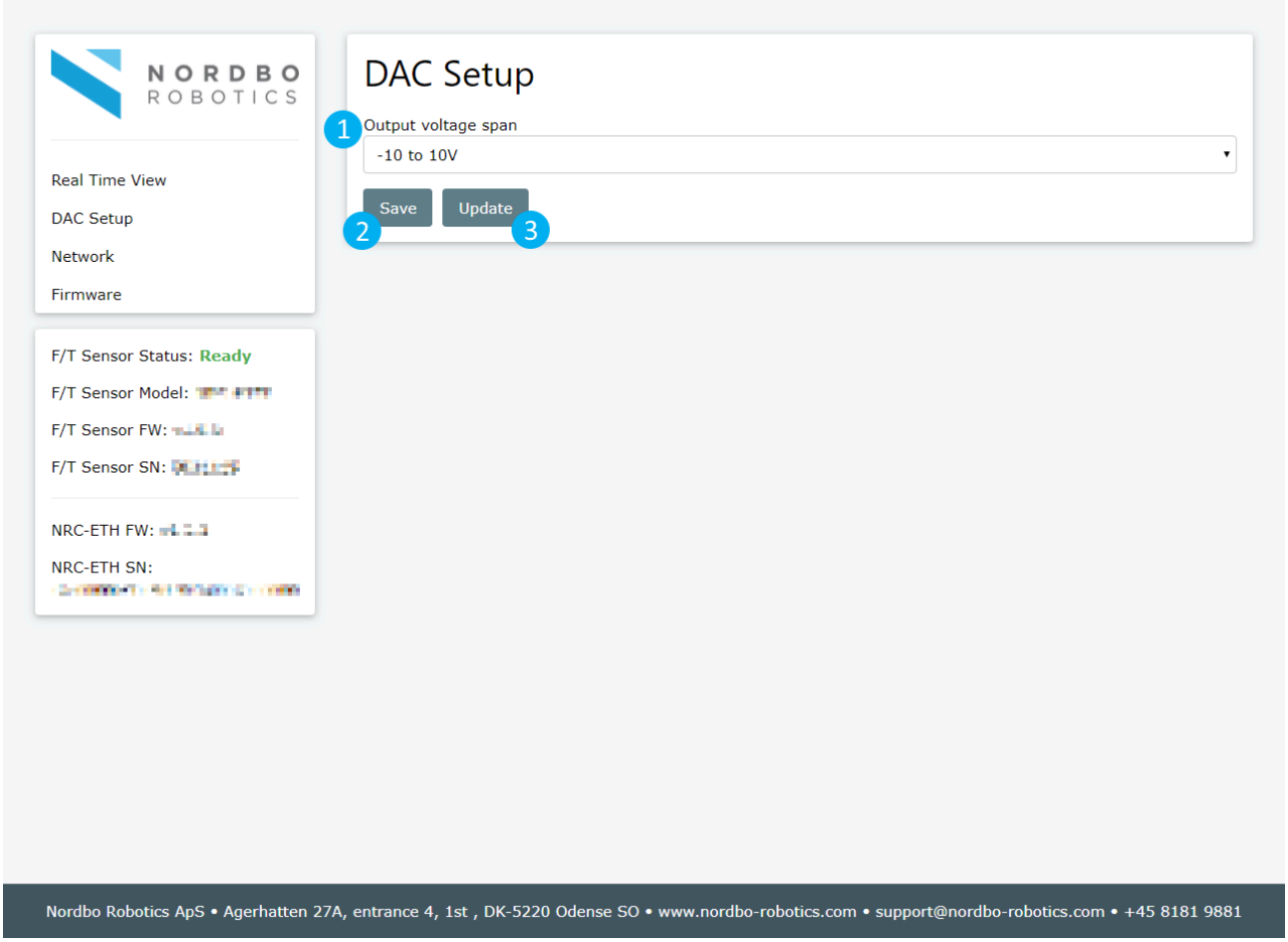
Step 2

Click the drop-down menu (1) to see a list of applicable output voltage span. Select a mode supported by the receiving device.

Step 3

Click “Save" (2) to save the selected output voltage span.

Note: Clicking “Update” (3) will load the currently active output voltage span from the NRC and replace any unsaved changes in the drop-down menu



4. NRS-6 using Ethernet

The following section describes how to use the NRS-6 with an ethernet connection. The user is required to have purchased the optional NRC signal converter to follow this guide.

4.1. Connecting using ethernet

The following section describes how to connect the force-torque sensor to your application with ethernet using the NRC signal converter.
If the application is designed to communicate with the NRS-6 force torque sensor using an ethernet connection, the reading device must support TCP/IP communication. The ethernet cable must be connected to the NRC using a RJ45 cable plug.

Step 1

Connect the NRC-ETH to the controller using an ethernet switch and an RJ45 cable.
OBS.: It is recommended to connect the controller and NRC-ETH through an Ethernet switch, but the controller and NRC-ETH can be connected directly.

Step 2

To set up the controller for communicating with the NRC-ETH, the user must change the controller's IP address and subnet mask to be in the application's local network. Assigning the default gateway is optional but recommended.

Example: If using the NRC-ETH with default IP address and subnet mask, of "192.168.1.100" and "255.255.255.0" respectively, you can set the controllers IP address and subnet mask to "192.168.1.101" and "255.255.255.0". Instructions on changing the IP and other parameters of the NRC can be found in section 3. (Optional) Configuring the NRC-ETH/DAC.
OBS.: See section 2.6 (Optional) NRC-ETH/DAC Converter for information about the signal converter.

4.2. Communicating using ethernet

It is possible to communicate with the NRC-ETH, and thereby with the NRS-6, using TCP/IP. Nordbo Robotics offers a variety of program examples for communicating with the NRC-ETH through Ethernet. These can be changed to suit the user's needs. The code examples can be found on the supplied USB flash drive or www.gitlab.com/nordbo-robotics-public/nrs-6

Data types

The following table describes the data types used when communicating with the NRC-ETH.

name	data type	size in bits
BOOL	0 = False, >0 = True	8 bits
UINT8 (Byte)	Unsigned Integer	8 bits
DOUBLE	IEEE 754 floating point	64 bits
VECTOR6D	6 x DOUBLE	6 x 64 bits
STRING	ASCII char array	8 bits per char

Communication protocol

When communicating with the NRC-ETH, each package sent from the NRC-ETH consists of a header and a body – Each package sent to the NRC-ETH must follow this protocol.
The header consists of two bytes, and the body consists of up to 254 bytes.

Header	Header	Body	body	body
Byte 1	Byte 2	Byte 3	...	Byte 256
Package length	Command type	Parameter 1	Parameter ...	Parameter 254

The first byte of the header describes the length of the current package, INCLUDING the header. The minimum size of a package is 2 bytes.
The second byte of the header describes the command type. A list of available commands can be found in the table below.
The body allows the package to contain additional parameters. Additional parameters could be: start (0x01)/stop (0x00) or "200" milliseconds sampling rate.

command name	Command type (hex)
SENSOR_TRANSMIT	0x07
SENSOR_TRANSMIT_DATA	0x06
SENSOR_SET_CURRENT_TARE	0x15
SENSOR_SET_DATA_RATE	0x08
SENSOR_DAC_SET_SPAN	0x21
ERROR	0x45

An acknowledge is given to every request made to the NRC-ETH. The request will contain a header with the package length, the command type of the request and a body containing the status of the request (Accepted/Rejected).

E.g.: User **sends** following request:

package length	command type	body
Byte 1	Byte 2	Byte 3
3	0x07	0x01

In case the request is **accepted**, this is the acknowledge package:

Package length	command type	body
Byte 1	Byte 2	Byte 3
3	0x07	0x01

In case the request is **rejected**, this is the acknowledge package:

package length	command type	body
Byte 1	Byte 2	Byte 3
3	0x07	0x00

If an error occurs when handling a request

If an error occurs when handling a request, the NRC-ETH will respond with an error package. The header has the length and the command type "ERROR" (0x45). The first byte of the body is the command type of the request that generated the error, the second byte is an Error ID and all subsequent bytes contain an error message using ASCII. The length of the response may vary depending on the error message.

4.3. Ethernet Commands

In this section the user can find a list of supported commands. For each command, the Request section describes the format expected from the user. The Response section describes the format returned by the NRC-ETH. If a command does not provide a Request section the user cannot instantiate this command.

SENSOR_TRANSMIT 0x07
Start or stop a continuous force/torque data transmission.
Request:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Body: Mode	UINT8

Supported Modes:

NO	command name	COMMAND TYPE (HEX)
1	SENSOR_TRANSMIT_START	0x01
2	SENSOR_TRANSMIT_STOP	0x00

NO 1: Requests NRC_ETH to begin data transmission.
NO 2: Requests NRC-ETH to end data transmission.

Response:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Accepted/Rejected	BOOL

SENSOR_TRANSMIT_DATA 0x06
The command type that the NRC-ETH uses when sending data. The command type "SENSOR_TRANSMIT_DATA" cannot be sent to the NRC-ETH

Response:

byte content	data type

Header: Package length	UINT8
Header: Command type	UINT8
Body: Force and torque data (Fx, Fy, Fz, Tx, Ty, Tz)	VECTOR6D

SENSOR_SET_CURRENT_TARE 0x15
Triggers sensor tare.
Request:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Body: Mode	UINT8

Supported Modes:

NO	command name	COMMAND TYPE (HEX)
1	SENSOR_TARE_TYPE_ZERO	0x00
2	SENSOR_TARE_TYPE_NEGATIVE	0x01

NO 1: Remove current tare values.
NO 2: Subtracts current load from the measured values, setting the output to 0N/Nm.

Response:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Accepted/Rejected	BOOL

SENSOR_SET_DATA_RATE 0x08
Set the sample data rate . The body contains the number of milliseconds between data samples. Allowed data rate is 1 millisecond >= and <= 210 milliseconds.
Request:

byte content	data type
Header: Package length	UINT8

Header: Command type	UINT8
Body: Desired data rate in milliseconds (1-210 ms)	UINT8

Response:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Accepted(True)/Rejected(False)	BOOL

SENSOR_SET_DAC_SPAN 0x21
Set the voltage span to transfer analog force and torque values using the DAC module.

Note: Has no effect with if the NRC does not support DAC.

Request:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Body: DAC Span mode	UINT8

Supported Modes:

NO	command name	COMMAND TYPE (HEX)
1	DAC_SPAN_0V_TO_5V	0x00
2	DAC_SPAN_0V_TO_10V	0x01
3	DAC_SPAN_PM_5V	0x02
4	DAC_SPAN_PM_10V	0x03
5	DAC_SPAN_PM_2_5V	0x04

NO 1: Force and torque values will be scaled to a voltage between 0 and +5 volts.
NO 2: Force and torque values will be scaled to a voltage between 0 and +10 volts.
NO 3: Force and torque values will be scaled to a voltage between -5 volts and +5 volts.
NO 4: Force and torque values will be scaled to a voltage between -10 and +10 volts.
NO 5: Force and torque values will be scaled to a voltage between -2.5 volts and +2.5 volts.

Response:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Accepted(True)/Rejected(False)	BOOL

ERROR 0x45

The command type “Error” is sent by the NRC-ETH if an error occurs during a request. The command type “Error” cannot be sent to the NRC-ETH.

Response:

byte content	data type
Header: Package length	UINT8
Header: Command type	UINT8
Body: Last request command type	UINT8
Body: Error code	UINT8
Body: Error message	STRING

Error codes:

Error name	error id
ERROR_CODE_PACKAGE_SIZE	0x01
ERROR_NOT_IMPLEMENTED	0x02
ERROR_DEVICE_LOST	0x03
ERROR_WRONG_ARGUMENTS	0x04

Below is an example error package:

package length	command type	body	Body	Body
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5-9
14	0x45	0x07	0x04	“SOME ERROR”
4 bytes + 10 x 1-byte ASCII chars	Command type	Last command type requested	Error ID	Error message

5. NRS-6 using DAC

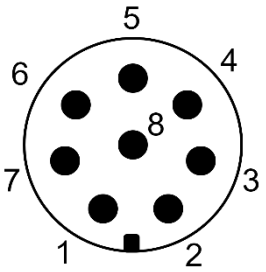
The following section describes how to use the NRS-6 with a DAC connection. This section applies only the NRS-DAC.

5.1. Connecting using DAC

Wire output(s) from the NRC-DAC connector to analog inputs on the controller.

The controller must have at least one analog input. Up to 6 analog inputs can be used based on the application's needs. Using the pin numbers of the 8-pin DAC connector illustration, the following table describes what the different pins of the connector transmits, along with their wire colors.

pin	Wire color	Value
1	White	Y-axis torque
2	Brown	X-axis torque
3	Green	Z-axis force
4	Yellow	Y-axis force
5	Gray	Ground
6	Pink	Ground
7	Blue	Z-axis torque
8	Red	X-axis force



5.2. Reading from the NRS-6 using DAC

The voltage input must be calculated into force and torque before it can be used. Following formula can calculate any given voltage span into force and torque values. To get a result, the user must have the following constant values: The voltage span's minimum and maximum values and the force/torque sensor's minimum and maximum values of a given axis.
Example: An analog voltage V_{cur} spans from V_{min} 0 millivolts to V_{max} 5000 millivolts (5 volts), and the NRS-6050-D80's Y-axis' force spans from FT_{min} -500 Newton to FT_{max} +500 Newton.

$$Result = \frac{(V_{cur} - V_{min}) * FT_{max} * 2}{V_{max} - V_{min}} + FT_{min}$$

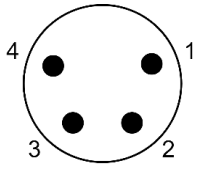
Using this formula, the analog input value will result in the desired force or torque value. The formula must be applied to every axis required by the application.

6. NRS-6 using CAN

The following section describes how to use the NRS-6 with a CAN connection.

6.1. Connect using CAN

A connection to the sensor is established using the 4-pin connector, which is wired as depicted below:

Pin no.	Description	illustration
1	CAN high	
2	5V DC	
3	Ground	
4	CAN low	

The NRS-6 requires a power supply of 5V/250mA.

6.2. Communicate using CAN

The CAN protocol describes 3 classes of messages:

- POLLING, for configuration
- PERIODIC, for automatic transmission of forces and torques
- CMD_BROADCAST, for finding connected sensors with unknown addresses

CLASS	ID
POLLING	2
PERIODIC	3
CMD_BROADCAST	7

Polling messages

POLLING messages are used to configure the sensor and needs to provide the class ID, source address and destination address, along with a payload, following the structure below:

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
Class	Source	Destination								

Class = 2
Source = [0 .. 15]
Destination = [0 .. 15]

All POLLING messages are acknowledged by a message with an empty payload.

Set CAN address

Sets the CAN address of the sensor. The default can address is 5.

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
0x2	Source	Destination	0x32	Address						

Source = [0 .. 15]
Destination = [0 .. 15]
Address = [1 .. 15]

Set transmission mode

Sets the transmission mode of the sensor.

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
0x2	Source	Destination	0x7	Mode						

Source = [0 .. 15]
Destination = [0 .. 15]
Mode = 0 or 1, where 0 enables automatic transmission of calibrated data continuously and 1 disables automatic transmission

Set CAN data rate

Sets the data transmission rate in milliseconds.

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
0x2	Source	Destination	0x8	Rate						

Source = [0 .. 15]
Destination = [0 .. 15]
Rate = [1 .. 210]

Set current tare

Sets the current tare on the sensor.

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
0x2	Source	Destination	0x8	Tare						

Source = [0 .. 15]
Destination = [0 .. 15]
Tare = 0 or 1, where 0 removes the current tare values and 1 sets the current tare values by subtracting the current load from the measured values, setting the output to 0N/Nm.

Periodic messages

PERIODIC messages can be set up to automatically transmit forces and torques. They provide the class ID, source address and message ID, along with a payload, following the structure below:

Header			Payload							
3 bits	4 bits	4 bits	B0 LSB	B1 MSB	B2 LSB	B3 MSB	B4 LSB	B5 MSB	B6	B7
Class	Source	Message ID	F/T 1		F/T 2		F/T 3			

Class = 3
Source = [0 .. 15]
Message ID = A or B, where A is force and B is torque
LSB = Least Significant Byte
MSB = Most Significant Byte

CMD_BROADCAST

Broadcast to all connected sensors. This will generate a response from every device connected, revealing their address.

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
Class	0x0	0x0F	0xFF							

Class = 7
Destination = [0 .. 15]

Response:

Header			Payload							
3 bits	4 bits	4 bits	B0	B1	B2	B3	B4	B5	B6	B7
0x7	Source	0x0	0xFF							

Source = [0 .. 15]

7. Using the NRS-6 with Codesys

Different controllers have different ways to set up the hardware for communicating with the NRC-ETH. Refer to the manufacturer of the target PLC's support services for programming- and/or hardware manuals. Following settings must be set up:

For the Ethernet functionality

- The PLC's IP address must be within the same subnet as the NRC-ETH, e.g. 192.168.1.X.. (See Section 3. Configuring the NRC-ETH/DAC).
- The PLC's subnet mask must be the same as the subnet mask of the NRC-ETH (See Section 3....)
- in Section 3. (Optional) Configuring the NRC-ETH/DAC (e.g. 255.255.255.0)
- If default gateway is used, it must be assigned to the same as the NRC-ETH e.g. 192.168.1.1 (See Section 3. (Optional) Configuring the NRC-ETH/DAC)
- Refer to PLC manufacturer's manuals for guidance on making above changes

For DAC functionality

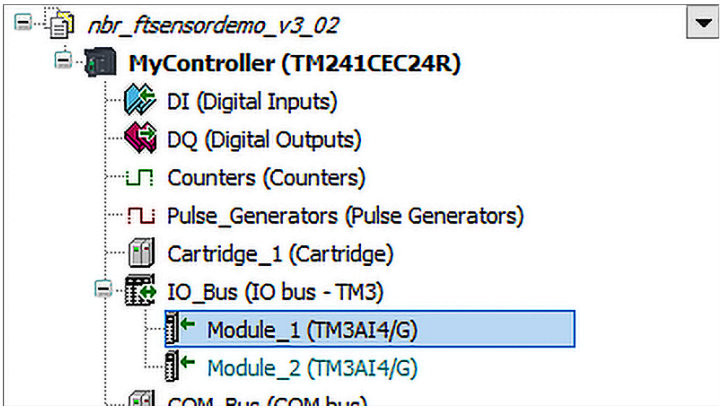
First setup the PLC's analog inputs to the voltage range selected on the NRC-ETH (See Section 3. (Optional) Configuring the NRC-ETH/DAC). After this, insert variables to store the analog values. Following setup was performed in Schneider Electric SoMachine.

Step 1

Identify if the application will be using on-board analog inputs on the PLC or external analog input modules

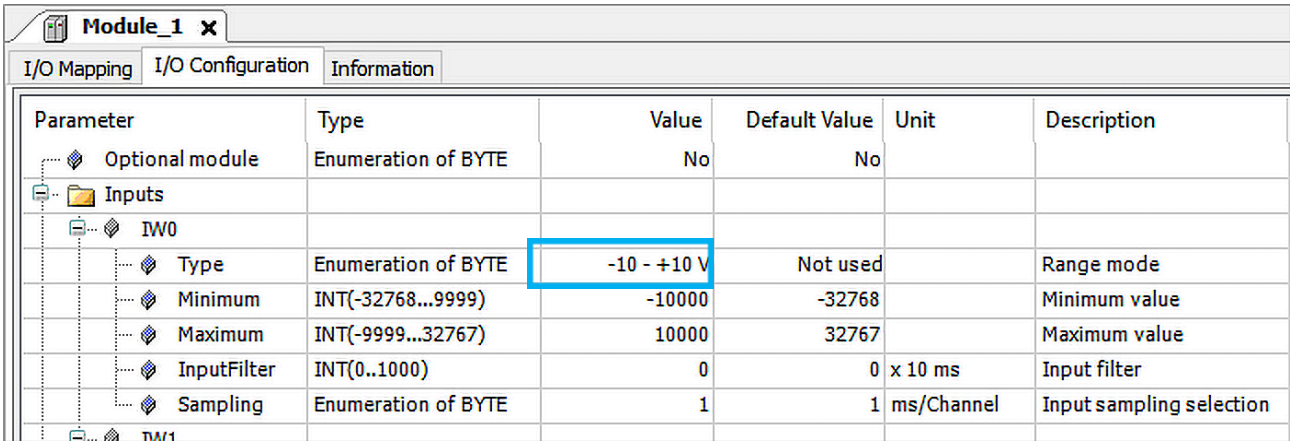
Step 2

In the tab Devices tree in the project browser on the left, go to the module identified in the step above. Double click the module



Step 3

Go to the tab I/O Configuration. In the marked tile assign the voltage range configured on the NRC. Click the tile and select desired voltage range from dropdown menu.



Step 4

Repeat for all inputs needed in the application (E.g. IW0, IW1, IW2)

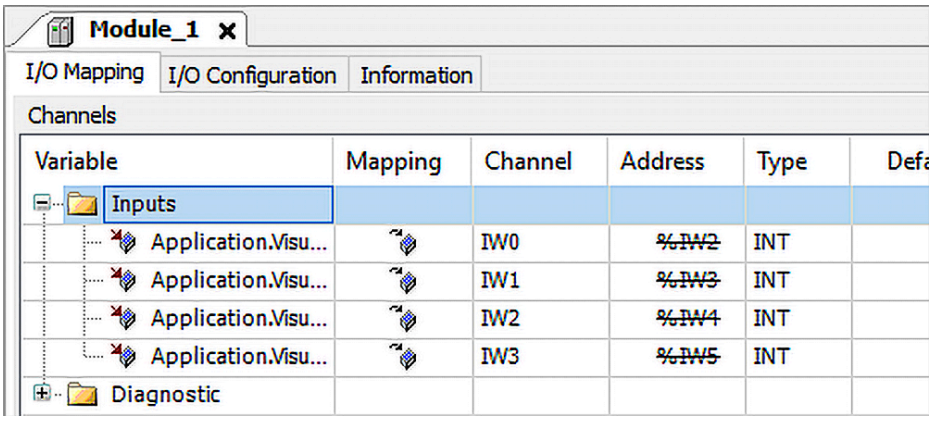
Step 5

Select the tab “I/O Mapping” and go to the folder “Inputs”

Step 6

Do one of the two following:

- Assign a variable with its full namespace in the tiles of the column “Variable”
  - E.g. “Application.GVL.iAnalogIn1”
- Use the direct address
  - E.g. “%IW2”



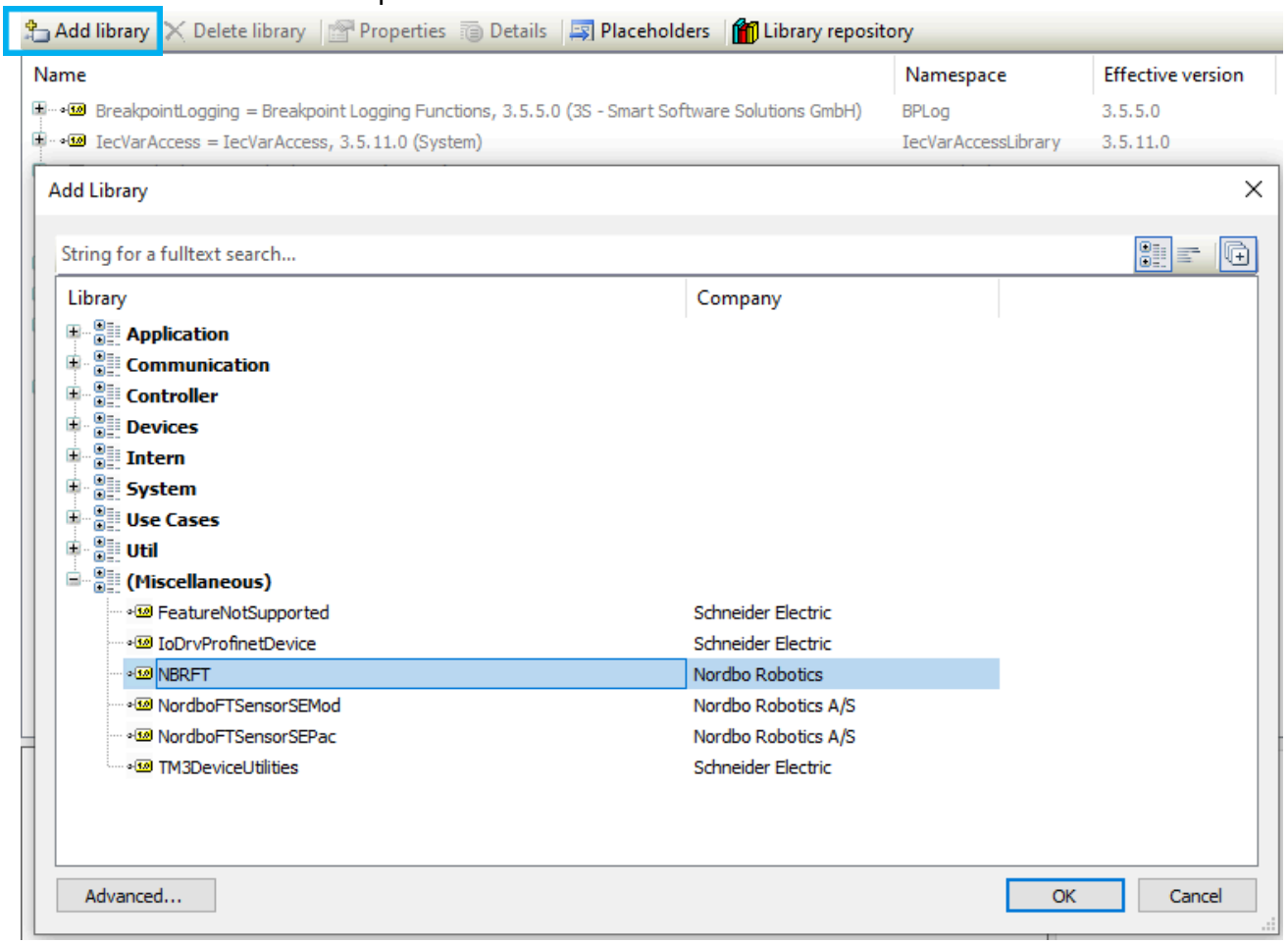
7.1. How to configure the function blocks

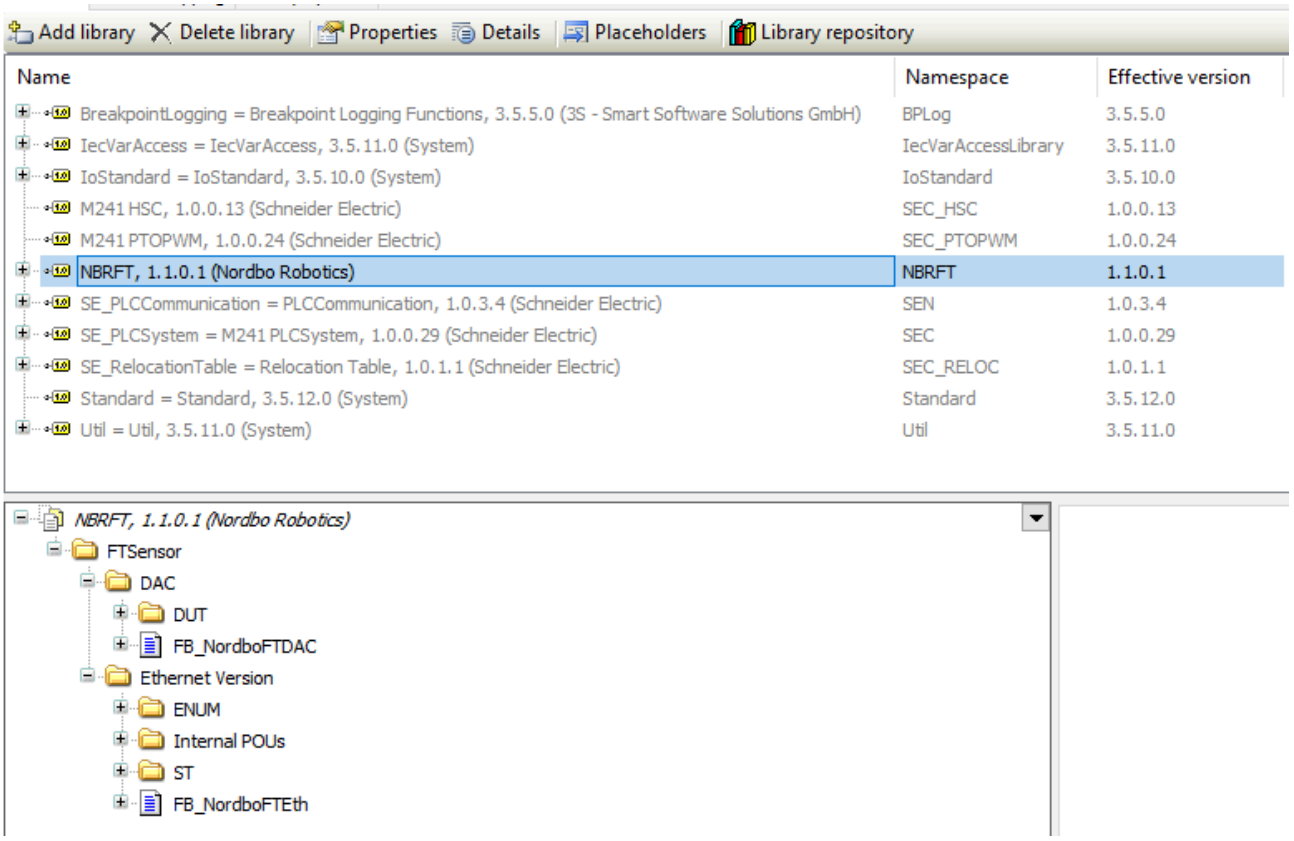
The following step describes how to set up the function blocks of the PLC libraries. It has been split into subsections that describe the procedure for specific controller types.

Schneider Electric Modicon and PacDrive controllers

For setting up the Ethernet function block, the following procedure must be followed.

In order to install the library, please go to ‘Library Manager’ and click ”Add library’. Search ‘NBRFT’ in the search field. And then press ‘OK’.





When the library is properly installed in SoMachine or SoMachine Motion, the user can access it using the namespace “NBRFT” (Nordbo Robotics Force Torque). Define and call the function block to communicate with the NRC-ETH using Ethernet.

Step 1

Identify whether the DAC or Ethernet function block is needed

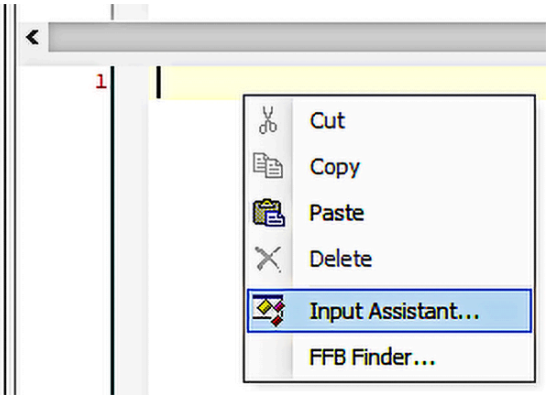
- Define a variable of the type “NBRFT.FB_NordboFTEth” for Ethernet
  - E.g. “myVariableName : NBRFT.FB_NordboFTEth;” in a VAR scope or Global Variable List
- Define a variable of the type “NBRFT.FB_NordboFTDAC” for DAC
  - E.g. “myVariableName : NBRFT.FB_NordboFTDAC;” in a VAR scope or Global Variable List

Step 2

Call function block in any POU

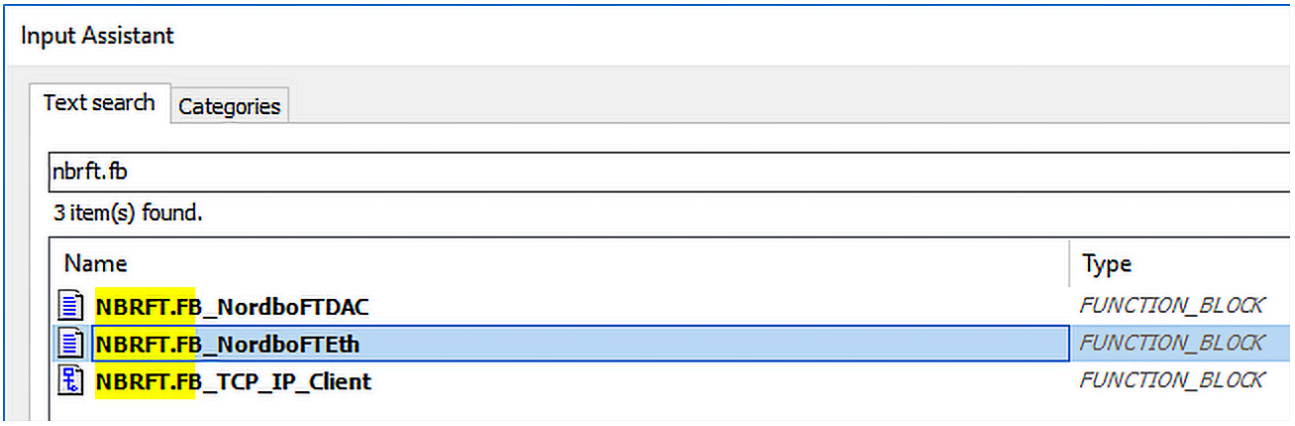
Step 3

For Structured Text POU the user can use Input Assistant to add the function block. Right click in the program editor and select “Input Assistant…”



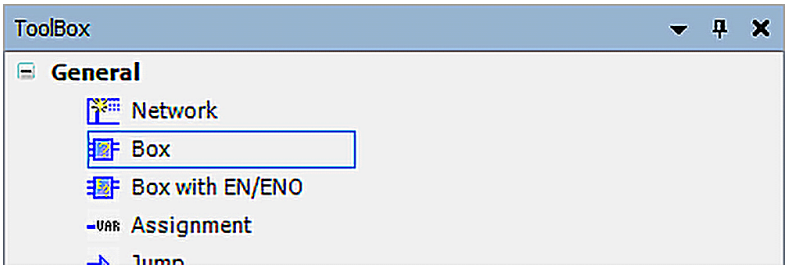
Step 4

Go to the tab “Text search” and type in part of the function block name (including namespace) E.g. “NBRFT.FB”. Double click the desired function block to insert it.

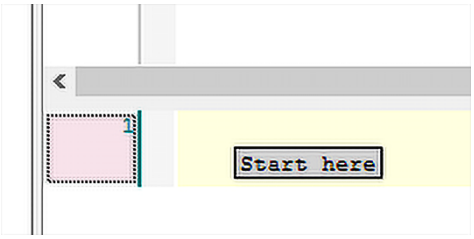


Step 5

For Function Block Diagram add a Box to a Network by clicking “Box” in the Toolbox panel (on the right side of SoMachine by default), drag and drop it to the area of the next figure



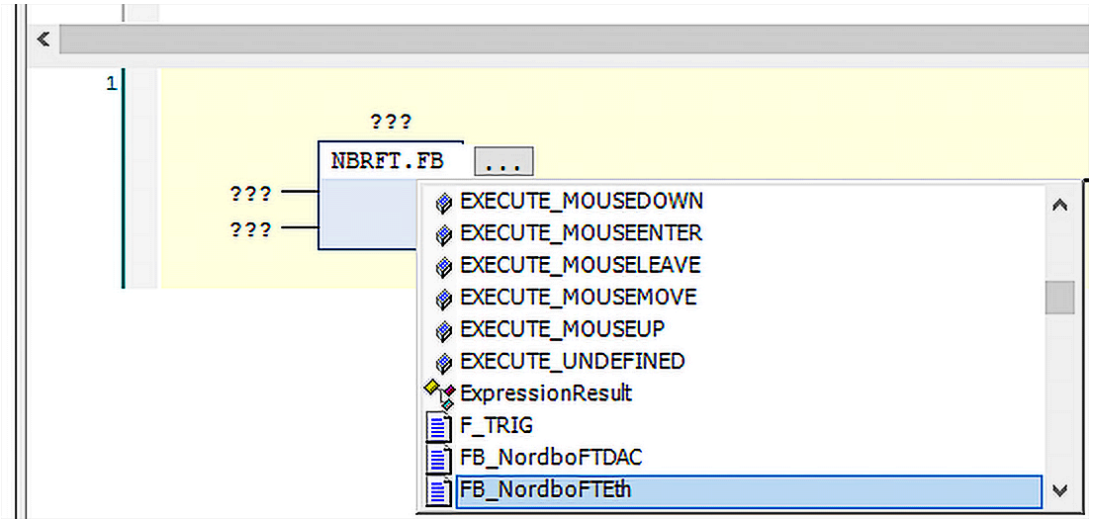
- Drop the Box object on the area with the text “Start here” – when the area turns green it means that the action will be completed



Step 6

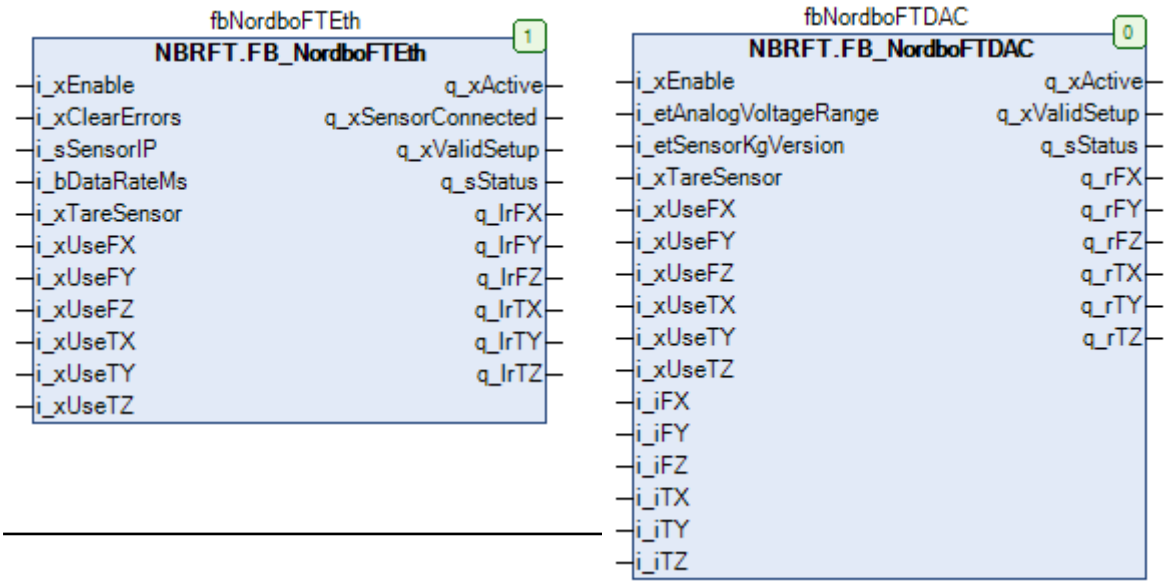
For the **Ethernet function block**, click the "???" inside the box and type in "NBRFT.FB_NordboFTEth" in the box

For the **DAC function block**, click the "???" inside the box and type in "NBRFT.FB_NordboFTDAC" in the box



Step 7

Press Enter to insert the selected function block

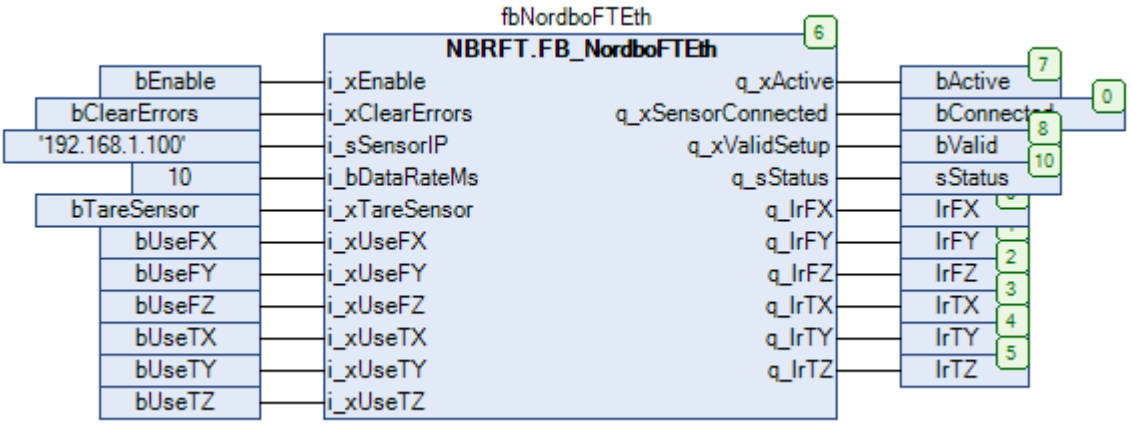


Step 8

The function block must now be set up according to the NRC-ETH setup. This section has been split into two, one describing the inputs of the Ethernet function block, and another describing the inputs of the DAC function block.

Ethernet function block inputs

Input name	Data type	Description
i_xEnable	BOOL	Enables function block and connects/disconnects TCP/IP client
i_ClearErrors	BOOL	Clear errors internally in function block
i_sSensorIP	STRING(15)	NRC-ETH IP address
i_bDataRateMs	BYTE	Number of milliseconds between each data package from the NRC-ETH
i_xTareSensor	BOOL	Set all force and torque values to 0.0 N(m) with current physical load as reference for the values (E.g. FX = Force X, TY = Torque Y)
i_xUseFX, -FY, -FZ, -TX, -TY, -TZ	BOOL	Enable calculation of the respective force or torque value (E.g. FX = Force X, TY = Torque Y)



- Enable should be a variable as it allows to reconnect if the connection is lost
- Clear errors should be a variable as it allows the user to reset errors in the function block
- It is recommended to input the sensor's IP address and the desired data rate as constants, even though they can be changed

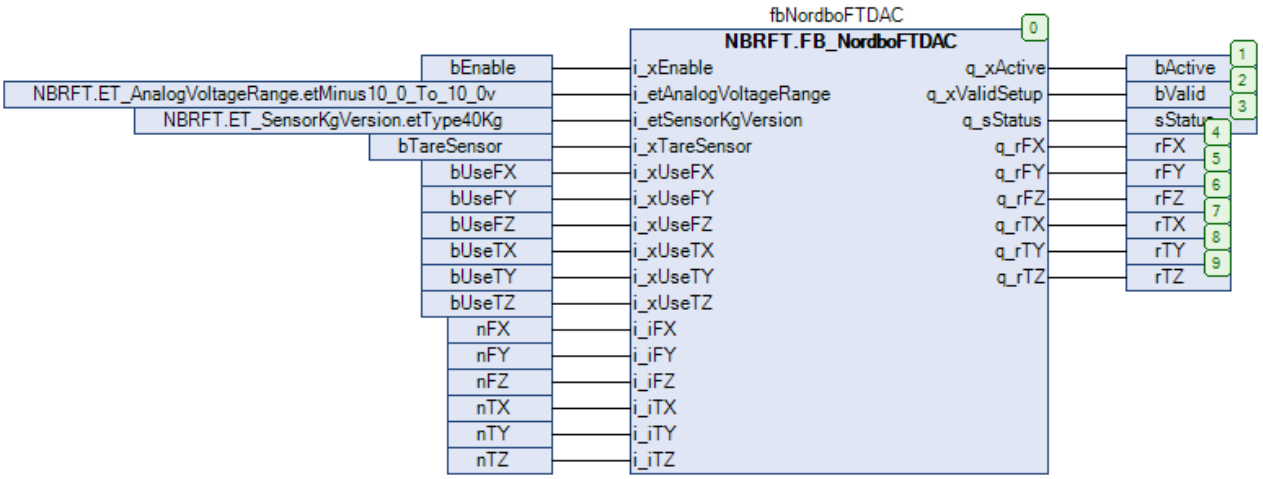
OBS.: Changing the data rate or using tare sensor while the data stream is active will stop the data stream until NRC-ETH has processed the requests – only do this if there it is certain that this cannot harm the process or things around it

    - Changing data rate to an unsupported value while data stream is active will generate an error and continue data stream with the same data rate as before the change was attempted
    - To change the IP-address the function block must be disabled before changing the IP address, and enabled no earlier than 1 cycle after the IP address has been changed
- Tare Sensor should be made variable, as resetting sensor values between cycles of the application is a commonly used feature
    - Tare sensor reacts on rising edge, and will not tare again before input has been False for at least 1 cycle
- The last six inputs (i_xUse*, e.g. i_xUseFX) enable the calculation of output values for their respective axes. The primary purpose of this is to allow the user to minimize CPU usage for smaller controllers by disabling calculation of unused values
- Unused axes will return 0.0 Newtons/Newton meters.

DAC function block inputs

Input name	Data type	Description
i_xEnable	BOOL	Enables function block
i_etAnalogVoltageRange	NBRFT.ET_AnalogVoltageRange	Inputs the voltage range of the analog input module using a custom data type from the library
i_etSensorKgVersion	NBRFT.ET_SensorKgVersion	Inputs the sensor's measuring range using a custom data type from the library
i_xTareSensor	BOOL	Set all force and torque values to 0.0 N(m) with current physical load as reference for the values
i_xUseFX, -FY, -FZ, -TX, -TY, -TZ	BOOL	Enable calculation of the respective force or torque value (E.g. FX = Force X, TY = Torque Y)

i_iFX, -FY, -FZ, -TX, -TY, -TZ	INT	Analog value inputs for the respective axes (E.g. FX = Force X, TY = Torque Y)



- Enable can be both constant and variable – use variable to be able to disable the function block to minimize CPU usage when FB is not used
- It is recommended to input a constant of the type NBRFT.ET_AnalogVoltageRange
- Entering the type's namespace and inserting a dot (".") should bring up a list of supported values – select desired number on the list
- NBRFT.ET_AnalogVoltageRange.etNoneSelected (Will not output anything)
- NBRFT.ET_AnalogVoltageRange.etMinus2_5_To_2_5v (-2.5 to 2.5 volt)
- NBRFT.ET_AnalogVoltageRange.etMinus5_0_To_5_0v (-5.0 to 5.0 volt)
- NBRFT.ET_AnalogVoltageRange.etMinus10_0_To_10_0v (-10.0 to 10.0 volt)
- NBRFT.ET_AnalogVoltageRange.et0_To_5v (0.0 to 5.0 volt)
- NBRFT.ET_AnalogVoltageRange.et0_To_10v (0.0 to 10.0 volt)
- It Is recommended to input a constant of the type NBRFT.ET_SensorKgVersion
- Entering the type's namespace and inserting a dot (".") should bring up a list of supported values – select desired number on the list
- NBRFT.ET_SensorKgVersion.etNoneSelected (Will not output anything)
- NBRFT.ET_SensorKgVersion.etType50Kg
- NBRFT.ET_SensorKgVersion.etType200Kg
- Tare Sensor should be made variable, as resetting sensor values between cycles of the application is an often-used feature
- The six inputs (i_xUse*, e.g. i_xUseFX) enable the calculation of output values for their respective axes. The primary purpose of this is to allow the user to minimize CPU usage for smaller controllers by disabling calculation of unused values

- Unused axes will return 0.0 Newtons/Newton meters.
- The six inputs (i_i*, e.g. i_iFX) must receive the values from the analog inputs for the function block to work properly.

7.2. Description of the function blocks' output

In this section the function blocks' outputs are described. It has been split into two sections, one describing outputs of the Ethernet function block, and one describing the outputs of the DAC function block.

Ethernet function block outputs

The following data is returned from the function block as a result of the setup and use and may be used in the user's application.

Output name	Data type	Description
q_xActive	BOOL	Displays that function block is currently active
q_xSensorConnected	BOOL	Displays that connection to sensor is OK while function block is enabled
q_xValidSetup	BOOL	Displays that input configuration (data rate and IP address) do not differ from allowed input values. Note that IP address is not necessarily the correct address just because Valid Setup is true.
q_sStatus	STRING(80)	Displays the function block's current status
q_lrFX, -FY, -FZ, -TX, -TY, -TZ	LREAL	Data read from the respective axis of the NRS-6, received from NRC-ETH (E.g. FX = Force X, TY = Torque Y)

- Active is only true if function block is enabled, the connection to the sensor has been established correctly and there are no errors active in the TCP/IP client
- Sensor Connected displays that NRC-ETH is connected through Ethernet and that the connection is active
- Valid Setup is True if data rate is within the supported range and the assigned IP-address is longer or equals to the length of the address '0.0.0.0'
    - Changing IP address while data stream is active will generate an error
- The status output displays the current state of the function block setup and functionality
- The axes' output data is to be used in the process of the user's application for whatever they are needed for. Values may be used without defining variables for them
    - Outputs that are not enabled will return the value 0.0

DAC function block outputs

Output name	Data type	Description
q_xActive	BOOL	Displays that function block is currently active
q_xValidSetup	BOOL	Displays if input configuration (voltage range and sensor weight version) differ from allowed input values, or if the combination of input values is invalid.
q_sStatus	STRING(80)	Displays the function block's current status
q_lrFX, -FY, -FZ, -TX, -TY, -TZ	LREAL	Data read from the respective axis of the NRS-6, received from NRC-ETH (E.g. FX = Force X, TY = Torque Y)

- Active output is true if the function block's setup is valid, one or more output calculations is active and the enable input is true
- Valid Setup is True if voltage- or current range and sensor weight version have allowed values
- The status output displays the current state of the function block setup and functionality
- The axes' output data is to be used in the process of the user's application for whatever they are needed for. Values may be used without defining variables for them
    - Outputs that are not enabled will return the value 0.0

8. Troubleshooting

8.1. Resetting the NRC

If required (i.e. an invalid IP making communication impossible, etc.), the device can be restored to factory settings. On the front of the NRC a reset button can be found next to the four LEDs.

Soft reset

A brief press of this button will soft reset the device, equivalent to power cycling. All settings will remain through a soft reset.

Hard reset

To perform a hard reset, press and hold the reset button for 6 seconds. The button must be hold down until all 4 LEDs have turned on and off again. This indicates that a hard reset has been performed and the settings have been reset to factory defaults. After performing a hard reset the device can be found on the default IP network address.

Note:

The latest converter box has default IP 192.168.**1**.100

The older converter box has default IP 192.168.**0**.100

8.2. If the NRC does not start when connected to power:

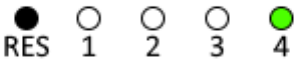
If the NRCs green LED is off when the power is connected.
The user cannot access the web interface.

Solution

The power supply may either be broken or supplying insufficient voltage- or current levels.
Make sure that the power connector is an original cable supplied by Nordbo Robotics. Check if the power cable is damaged. Make sure that the power connector of the NRC is not damaged.
The power supply must deliver between +6-volts and +40-volts DC, and a minimum of 500 mA regardless of voltage. Try a different power supply and see if resolves the problem.
If the NRC is wired different than described the NRC may be destroyed permanently. Make sure that the red wire of the power connector is connected to +VDC, and the black wire is connected to ground.

8.3. If the NRC is powered on, but there is no data:

Solutions:
Make sure that the sensor is connected properly to the NRC. Check web interface to see if the sensor is connected.
Restart the firmware by pressing and holding the RES button for 1 second.
If using PLC libraries, URCaps or program examples supplied by Nordbo Robotics, refer to the related subject's documentation. These can be found using Nordbo Robotics' support page.
If the user has made their own program for communicating with the NRC, Nordbo Robotics cannot offer support for fixing it. Creating and/or using custom programs will always be the user's responsibility.



Please see Section (Optional) NRC-ETH/DAC Converter for details.

8.4. If NRC is powered on, sensor is connected, but there is no data:

Green LED is on.
LED 1 is blinking showing that the firmware is running.
LED 2 is on showing that sensor is connected.
Solutions: Check the NRC's web interface to identify if the data never reaches the NRC. If the data is visible in the NRC's web interface, the problem occurs in the user program.
If the user is using a program example, URCap or PLC library issued by Nordbo Robotics, they can contact our support regarding the issue. Most issues can be fixed by rebooting the equipment running the program.
If the user is making a custom program for communicating with the NRC, Nordbo Robotics cannot offer support for this matter.
If the data does not show in the NRC's web interface, the NRS-6 may have malfunctioned. If this has happened within the time of warranty, the user must contact Nordbo Robotics immediately regarding this matter, so that the user may get a replacement product. If the user has any questions regarding this matter after the period of warranty, they may also contact Nordbo Robotics about this, but the product cannot be replaced.
If the user opens the product's casings the warranty will no longer be valid as this will be considered intentionally damaging the equipment.

8.5. If the NRC is on, but the user or application cannot connect:

Green LED is on
LED 1 is blinking, but the user cannot access the NRC's web interface, or their application cannot communicate with the NRC.
Solutions: If LED 1 is not blinking, the application has likely crashed. Attempt to restart it by pressing and holding the "Reset" button for 1 second. Wait at least 15 seconds for the firmware to reboot before attempting to reconnect.
Check if the user's application and the NRC are still physically connected, either with the DAC cable or with an ethernet cable. Make sure that no cables or connectors are damaged.
Check if the device that is used for the attempting to connect is in the same local network as the NRC, and ensure that the user does not have the same IP address as the NRC.
E.g.: NRC: IP address: 192.168.1.100, Subnet mask: 255.255.255.0, User: 192.168.1.101, Subnet mask: 255.255.255.0.
Confirm that the ethernet port of the user's computer or application is working properly, and that the ethernet port (or DAC connector if this is used) of the NRC has not been damaged.
If the user has recently changed the Network settings using the web interface, make sure that the setup was not invalid. If the network settings have been corrupted, and it is impossible to find the device on the local network, the user must perform a hard reset of the device. This is done by powering off the NRC, holding the Reset-button while powering it back on, and holding it until all LEDs have glowed and turned off again. When this is done, reset the power of the NRC and it should be available on the default IP address.

Note:

The latest converter box has default IP 192.168.**1**.100

The older converter box has default IP 192.168.**0**.100

8.6. If your problem not listed:

If none of the above solutions have solved the problem, the user can contact Nordbo Robotics support using the contact information provided in section 8.8 Repairs. When contacting Nordbo support, you must inform the *product*, the *issue*, *how* and *when* the error occurred, and if it is a *recurring* problem.

8.7. Maintenance

The product does not require any specific maintenance. Updating the firmware of the NRC and the sensor when new updates are published may help improving user experience and performance of the system.

8.8. Repairs

It is important that the user does not attempt to repair the product themselves. Any attempt of opening the product's casing or of repairing the product will violate the warranty of the product. If the product is damage as a result of irresponsible or aggressive product handling the warranty is invalid. If a product is damaged or malfunctioning upon receival, the user must contact Nordbo Robotics regarding the issue. Shipping addresses for returning products for repair, or shipping addresses for replacing malfunctioning products and necessary details will be provided during the dialogue with Nordbo Robotics. Nordbo Robotics cannot be held responsible for products lost during shipping by companies of the customer's choice.

Contact us:

Nordbo Robotics Support
+45 81 81 98 81
support@nordbo-robotics.com

9. Declarations and Certificates

CE/EU Declaration of Incorporation, original

According to the European Machinery Directive 2006/42/EC Annex II, part 1.

We, the manufacturer:

Nordbo Robotics A/S

Agerhatten 27A, Entrance 4

DK-5220 Odense, Denmark

Hereby declare that the following product:

Type: Industrial 6-axis Force & Torque sensor

Model: NRS-6050-D80 and NRS-6200-D80

Serial number form: YYMM001 (Ex. 1908001)

Meets the requirements of the Machinery Directive 2006/42/EC as partly completed machinery.

The product must not be put to use before the complete machine in which the product is installed is in full compliance with all requirements of the European Machinery Directive 2006/42/EC.

A comprehensive risk assessment must be performed for each application to ensure that all essential requirements are being met.

Technical documentation specified by Annex VII part B is available to national authorities upon request.

The product declares in conformity with the following directives, according to which the product is CE marked.

2014/30/EU Electromagnetic Compatibility Directive (EMC)

2011/65/EU Restriction of the use of certain hazardous substances (RoHS)

Person responsible for documentation: Mr. Jimmy Alison Jørgensen

Jimmy Alison Jørgensen

Chief Technical Officer

Nordbo Robotics A/S

Nordbo Robotics A/S

Noatunvej 2

5000 Odense Denmark

info@nordbo-robotics.com

+45 81 81 98 81