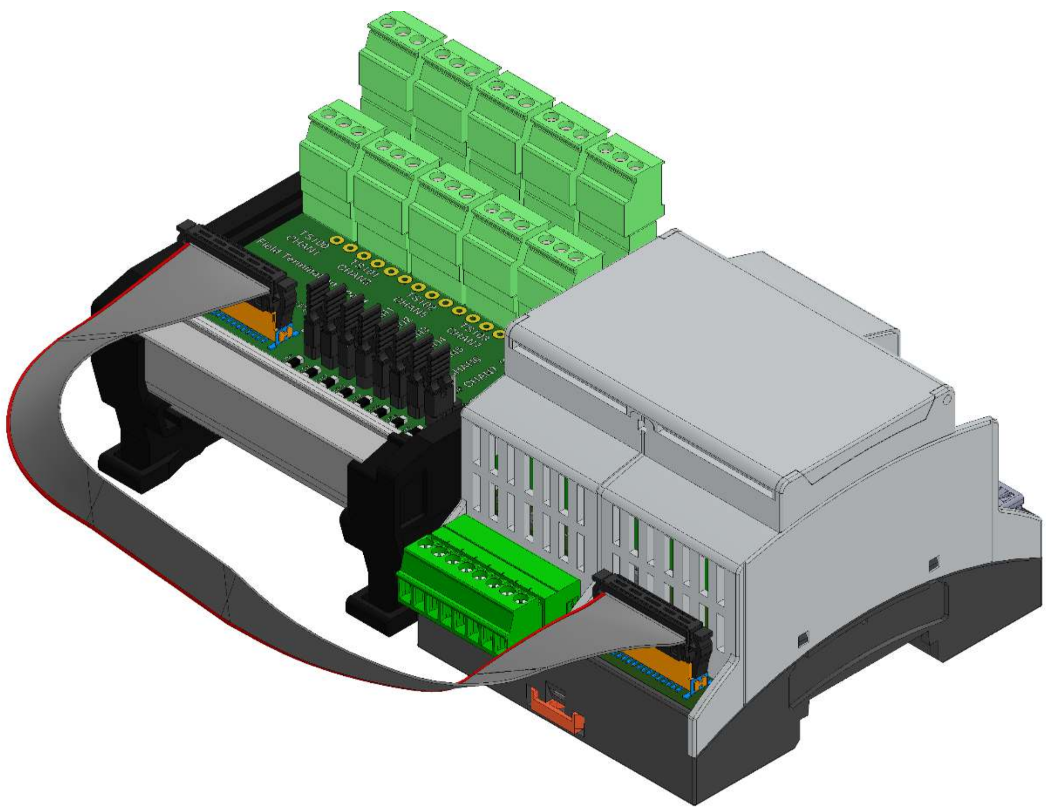
X8II
User Manual



20	12/21/22			SH	General update and proofread of entire manual. Added application tab section to UCT portion.
19	10/17/22			AW	Corrected a Nano holding register address, and added a 3rd Nano solution for Modbus TCP
18	N/A		AW	SH	Updated the UCT Port-Open sequence. Added Modbus support (section 7). Updated UCT section to reflect current release.
17	N/A			AW	Added description of New UCT Port-Open Sequence
16	N/A			SH	Added speed probe wiring diagram.
15	N/A			SH	Updated the UCT section
14	N/A			SH	Modified the IP Settings for X8II Communication on page 24
13	N/A	AA	AW	SH	Reduced capacity NANO's from 64 to 16
12	N/A	SS	AA	SH	UCT Updates, CANOpen added (section 4.4), Removed "Servicing" section 12
11	N/A	AW	AR	SH	Sales edits
10	N/A	AW	AR	SS	Changes by sales and development
9	N/A			SH	Recommended changes by Sandy
8	N/A			SH	Major updates across most sections, sections added. Overall document editing.
7	N/A			SH	Combination of information into one document
6	N/A	AA	PL	SH	Inserted page breaks
5	N/A	AA	PL	SH	Condensed information into one document
4	N/A				
3	N/A			SH	
2	N/A			PL	
1	N/A				
0	N/A				
Rev	Date				Description

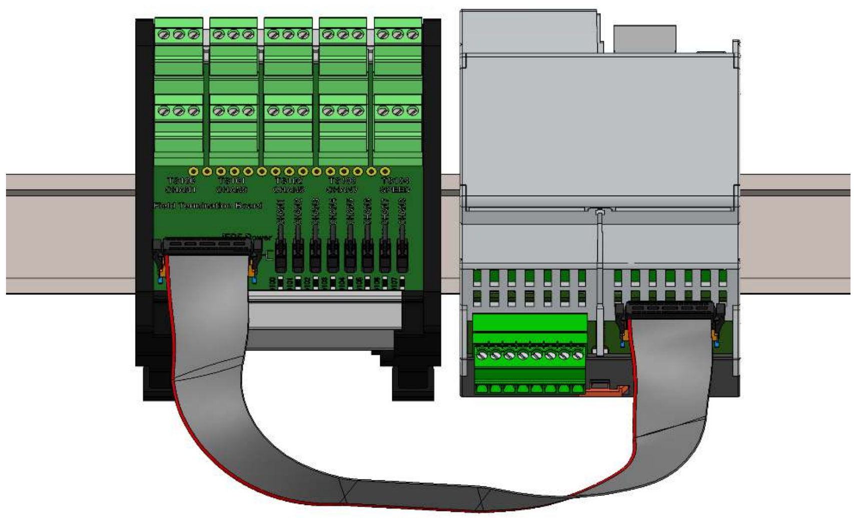
TABLE OF CONTENTS

1 INTRODUCTION

1.1 X8II SYSTEM

The X8II System is designed to be installed and connected to wired or wireless condition monitoring sensors. The system consists of three devices: the X8II, Termination Panel and Ribbon Cable connecting the two. The X8II system is DIN Rail mounted, which allows for easy installation and servicing.

Wired sensors (Accelerometer, Proximity Probe, Temperature and Process Signals) are attached to the Termination Panel (Section 2.5), and the X8II receives the sensor signals and processes them based on a programmed configuration that is individual for each channel. The Termination Panel has plug in screw terminals for field connection to the sensors. The Termination Panel connects the sensor signals to the X8II through a ribbon cable. The X8II is powered from a DIN Rail Power supply (supplied by the customer, more information can be found in section 1.5).

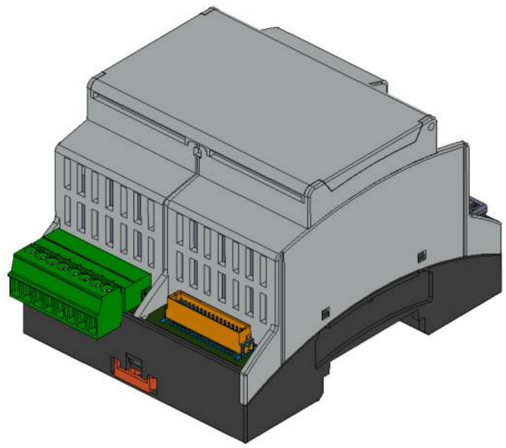


The X8II System

1.2 THE X8II

The X8II is the centerpiece of the system and provides reliable, repeatable measurements for a wide range of monitoring points. It can support both wireless (Wi-Fi 802.11 b/g/n WPA-2) and wired signals (via the Termination Panel, see section 1.3). The X8II comes equipped with encrypted communication for an additional layer of security, an RJ45 access point and an antenna to boost its wireless signal. For more information on how to install the antenna please refer to section 2.3.4 (Antenna Mounting).

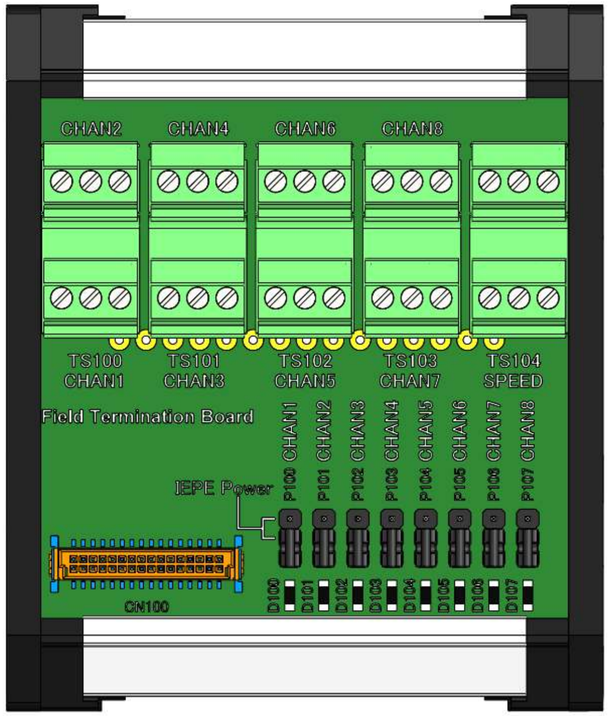
One X8II can communicate with up to 16 NANO devices, and 8 wired sensors. Once measurement data has been received via wireless or wired sensors, the X8II transfers data to the customers preferred destination (MD, EMonitor or infiSYS Lite - see section 4 for more information). See section 10 for full X8II specifications.



The X8II

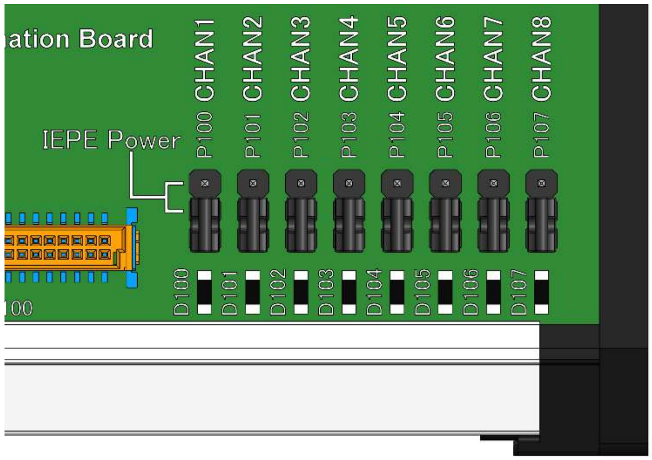
1.3 THE TERMINATION PANEL

The Termination Panel is an optional component designed so the field wiring of accelerometers, proximity probes, temperature, or strain sensors (see section 2.5 for wiring diagrams) can be easily connected to the pluggable screw terminals. The Termination Panel can handle up to 8 single channel wired connections and is connected to the X8II via a short Ribbon cable. It simplifies the field connections and allows easy removal of the field connections when troubleshooting an installation.



Termination Panel Layout

The Termination Panel has a series of 3 Pin Headers. These Headers allow individual configuration of the constant current power source, which is required to power an Accelerometer. When the jumper is in the IEPE Power is ON position a +24V @4mA constant current power source is connected to the positive Accelerometer input terminal for that channel. When the jumper is in the IEPE Power is OFF position no power is applied to the positive Accelerometer input terminal. This allows the end user to configure on a channel-by-channel basis what kind of sensor will be connected. There is a SPARE 3 position terminal that is reserved for future use. It resides adjacent to the SPEED terminal.



3 Pin Configuration Headers

1.4 The NANO

The NANO is a wireless condition monitoring device that is attached directly to the machine being monitored. NANOs come equipped with triaxial vibration and surface temperature sensors. NANOs will communicate, via Bluetooth 5.0, with any X8II within range to upload measurement data. NANOs can be setup to send SMS and email alerts.

For additional information please see the NANO User Manual, which includes all the necessary installation and operation details.

NANO

1.5 System Power Approach

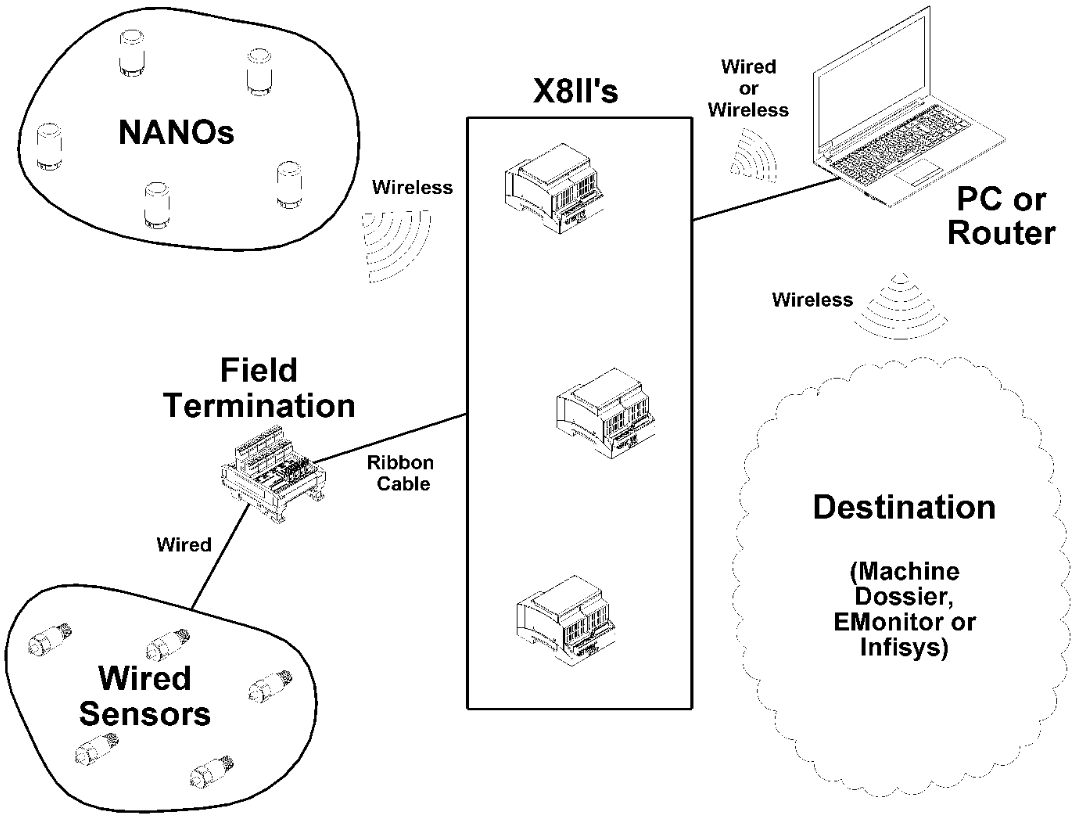
By using the DIN Rail Power Supply method, the customer can choose from the many options that are supplied at a low cost in the DIN rail power supply market. The customer is not limited by being single sourced, but still must comply with the X8II's power requirements of 5.0 Vdc at 6 Amps.

Section 9.3 of this manual lists recommended accessories and suggests using TDK Lambda's DRB50-5-1 Din Rail Mounted Power Supply. This is one of many Din Rail Mounted Power Supplies that will work with the X8II System.

1.6 X8II Example Environment

Below is an example of the X8II system and the interaction between the various components described in sections 1.1-1.5 above.

The NANOs and Wired Sensors will be installed on the machines being monitored and send the desired data wirelessly (in the case of the NANOs) or wired (Wired Sensors and Field Termination) back to the X8II's. The X8II's will then communicate wirelessly or wired (via the RJ45 jack) and send the data back to the customer's choice of on-site hardware (PC or Router). Once the data has been uploaded to a PC or Router, it can then be sent to a pre-defined destination (MD, EMonitor or infiSYS Lite – see section 4).



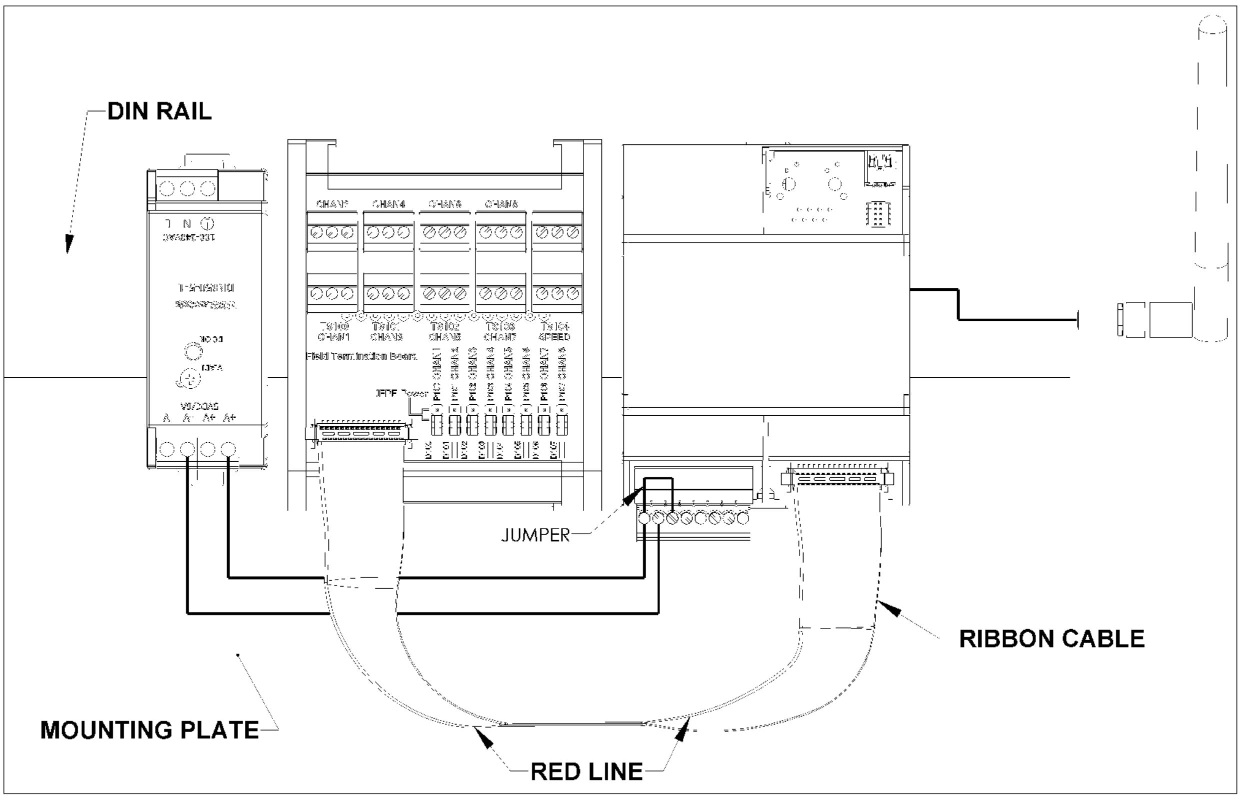
2 PHYSICAL INSTALLATION

2.1 GROUNDING

Proper instrumentation grounding procedures must be followed when installing this product. Each installation will have its own requirements. In all instances ground loops and connection to electrically noisy grounds must be avoided.

2.2 ENCLOSURE LAYOUT

The X8II is designed to be installed so the air vents allow air flow from bottom to top of their enclosure. The Termination Panel is designed to be within a short distance of the X8II. The diagram below shows a typical layout of the system within an enclosure.



Typical Installation

Notes:

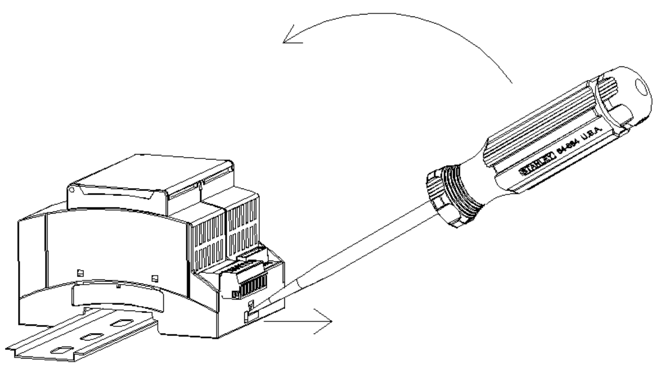
1) The X8II system can be installed in other directions, however this may reduce the allowable ambient temperatures of the installation.
2) A ribbon cable which is longer can be provided, however the use of a longer ribbon cable may allow the coupling of external electrical noise and degrade the overall accuracy of the measurements.

2.3 MECHANICAL INSTALLATION

The X8II, Termination Panel and Ribbon Cable are intended to be installed in a plastic DIN Rail enclosure. This section will detail the necessary steps to install the X8II system in an appropriate DIN Rail enclosure.

2.3.1 X8II

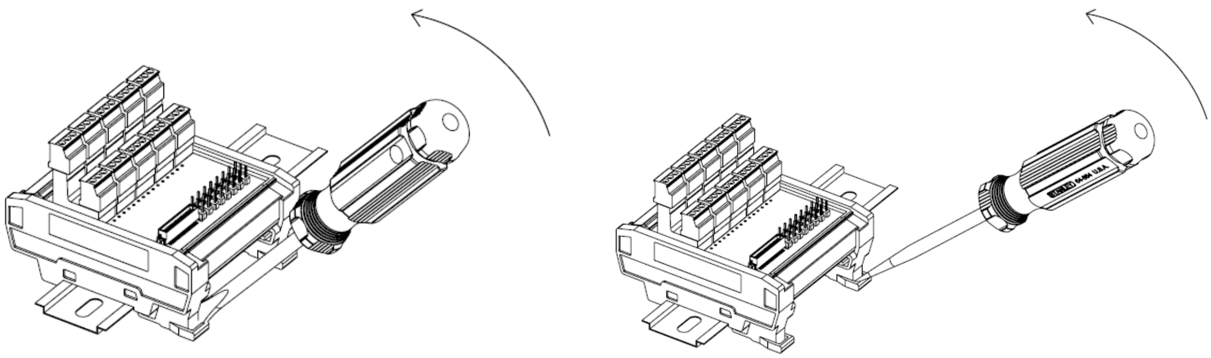
The X8II is a DIN rail mounted device. The typical direction of installation is with the ribbon cable connector and terminal block facing the user when the enclosure is open. This can be seen in section 2.2 Enclosure Layout. To remove the X8II use a screwdriver and pull out the bottom tab as you rotate the screwdriver handle towards the X8II as shown in picture below.



2.3.2 Termination Panel

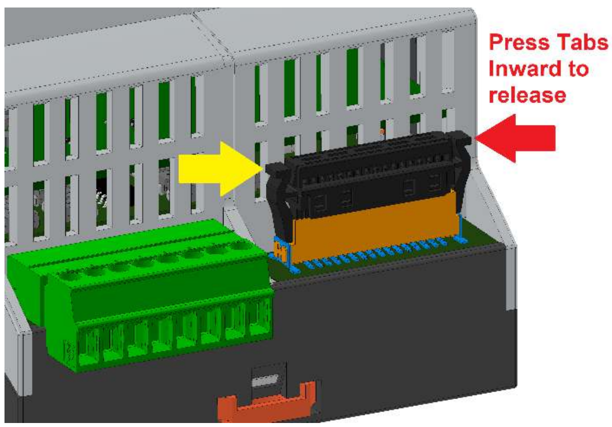
The Termination Panel has two DIN Rail clips, one on each side of the carrier. The Termination Panel should be mounted as shown in the picture under section 2.2 Enclosure Layout section labeled “Typical Installation.” The field wiring terminal blocks should be at the top and the Ribbon Cable connector should be adjacent to the X8II Ribbon Cable connector. To remove the Termination Panel a screwdriver must be inserted into the two black tabs and the screwdriver rotated forward as shown in the pictures below.

Note: The ribbon cable should be kept away from electrical noise sources to keep the vibration measurement signals from being distorted.



2.3.3 Ribbon Cable Installation

A squeezing action must be applied to allow removal of the Ribbon cable connector from the printed circuit board connector. This type of connector displays good vibration resistance yet still allows the end user to easily connect and disconnect when troubleshooting using the plug and swap method.



2.3.4 Antenna Mounting

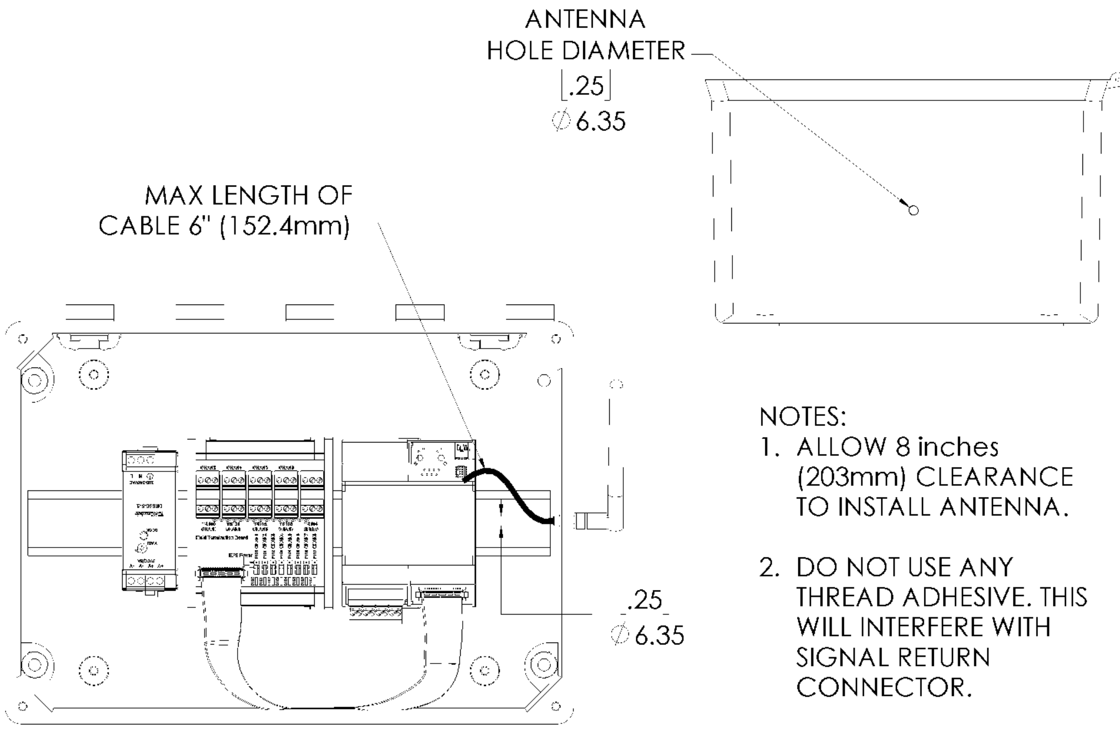
The X8II connects to a cabled antenna. This allows the customer to place the antenna in an optimum location on the outside of the housing. A hole must be drilled in the housing (0.25” or 6.35mm diameter) and the total cable length is 6 inches (150mm). So, the distance from the X8II to the hole in the housing will need to be close enough to allow the cable to reach without stressing/stretching the cable.

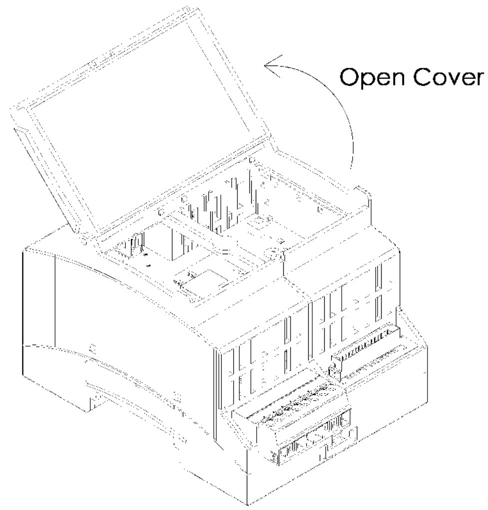
Warnings:

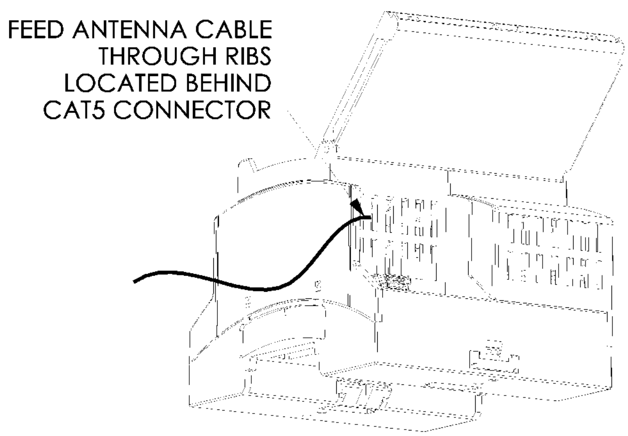
1) DO NOT use any form of sealing compound or thread locker on the antenna threads of the cable connector. These compounds can form an electrical isolation barrier and the wireless communication may fail.
2) Please note the minimum thread height that must protrude through the housing to allow for full contact between the antenna connector and the cable connector

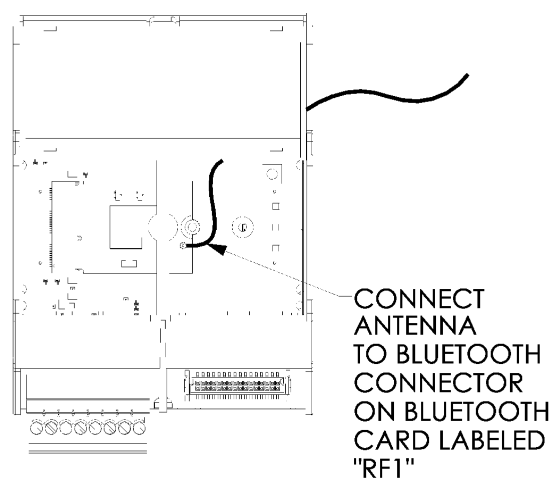
is a minimum of 0.220 inches [5.6mm]. If full threaded contact is not made between the cable connector and the antenna connector intermittent or nonexistent communications may occur.

3) After installation, when rotating the antenna make sure to not rotate it so that the connectors between the cable and the antenna start to unthread. This may cause wireless communications to be intermittent or fail.
4) DO NOT install the X8II in a metal enclosure, as this will interfere with incoming and outgoing wireless signals.




Open Cover

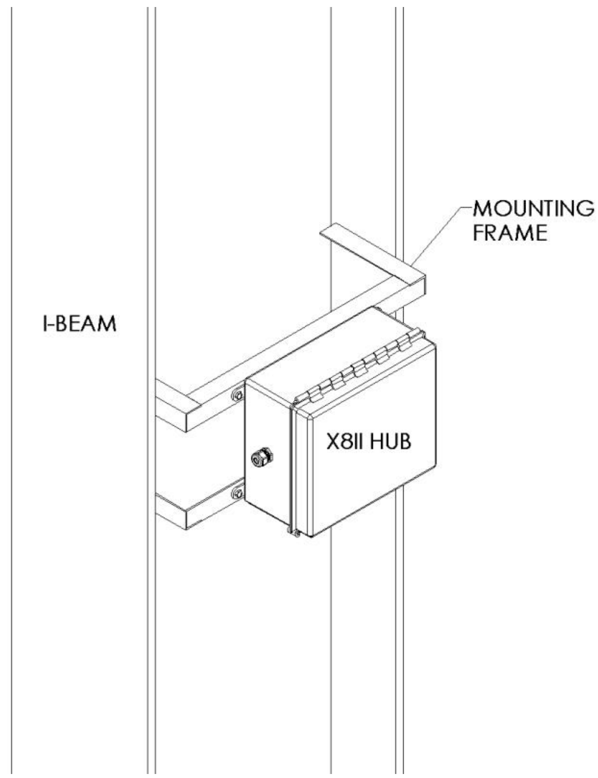

FEED ANTENNA CABLE
THROUGH RIBS
LOCATED BEHIND
CAT5 CONNECTOR


CONNECT
ANTENNA
TO BLUETOOTH
CONNECTOR
ON BLUETOOTH
CARD LABELED
"RF1"

2.4 ENCLOSURE MOUNTING METHODS

The X8II enclosure can be installed in many different environments. This section will describe the most common methods used to mechanically mount the enclosure that contains the X8II system.

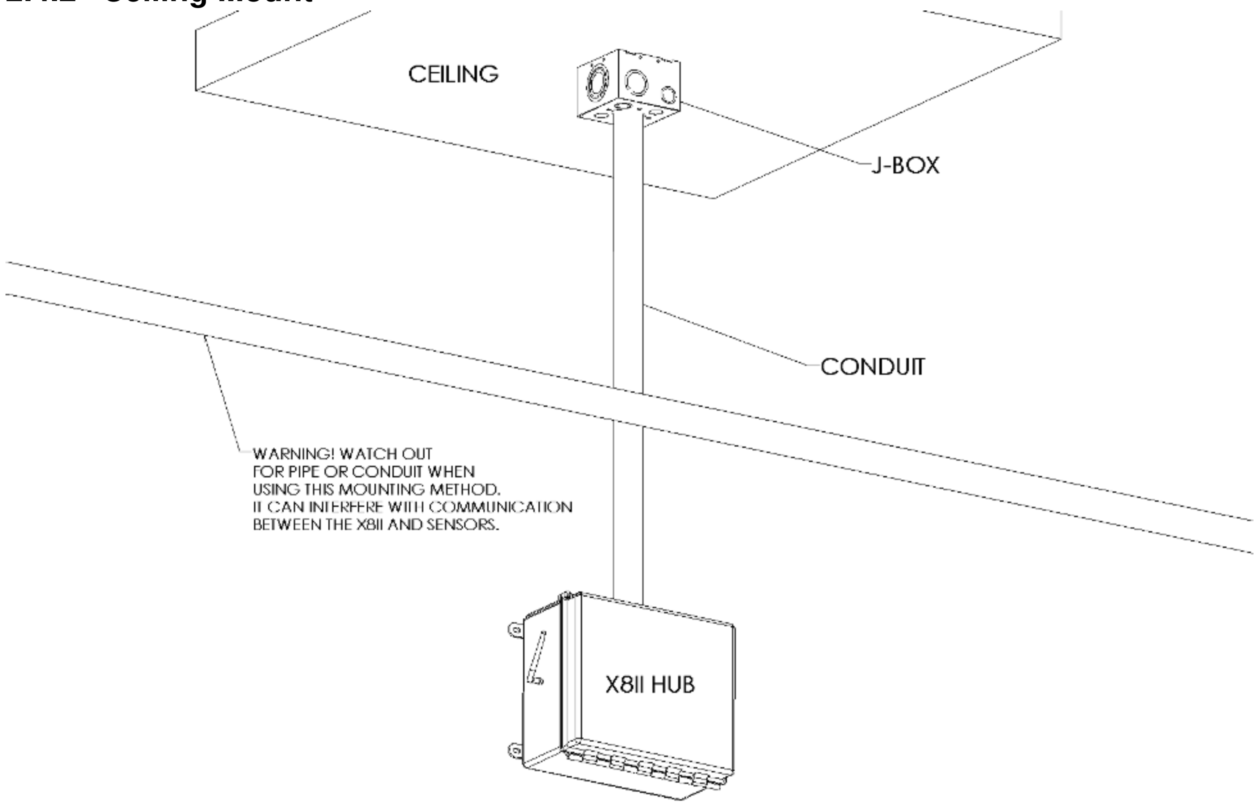
2.4.1 Pole/Beam Mounting



Notes:

1) When mounting from a pole or beam consider local, industry and facility standards/practices for acceptable distance from the mounting surface.
2) X8II should be mounted 2-3 inches beyond any metal surface to ensure proper communication to and from the unit.

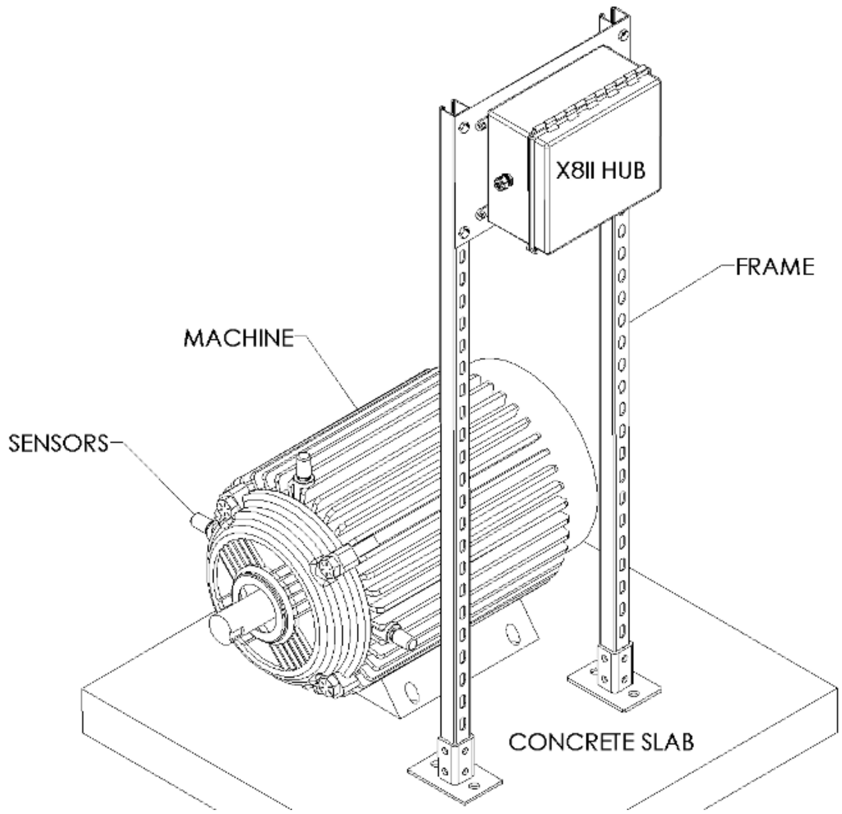
2.4.2 Ceiling Mount



Notes:

1) When suspending the X8II from the ceiling be aware of pipes or conduit sharing the same air space. If they happen to be near line of sight between the X8II and a NANO they can impact communication.
2) Ensure facility and industry standards and practices are followed for acceptable distance from ground to ceiling mounted devices.
3) Installation environment might also limit mounting space for suspended items over specific equipment, ensure all facility requirements are followed.

2.4.3 Machine Mount



Notes:

1) Ensure line of site between NANOs and X8II is not impaired by frame construction.

Warnings:

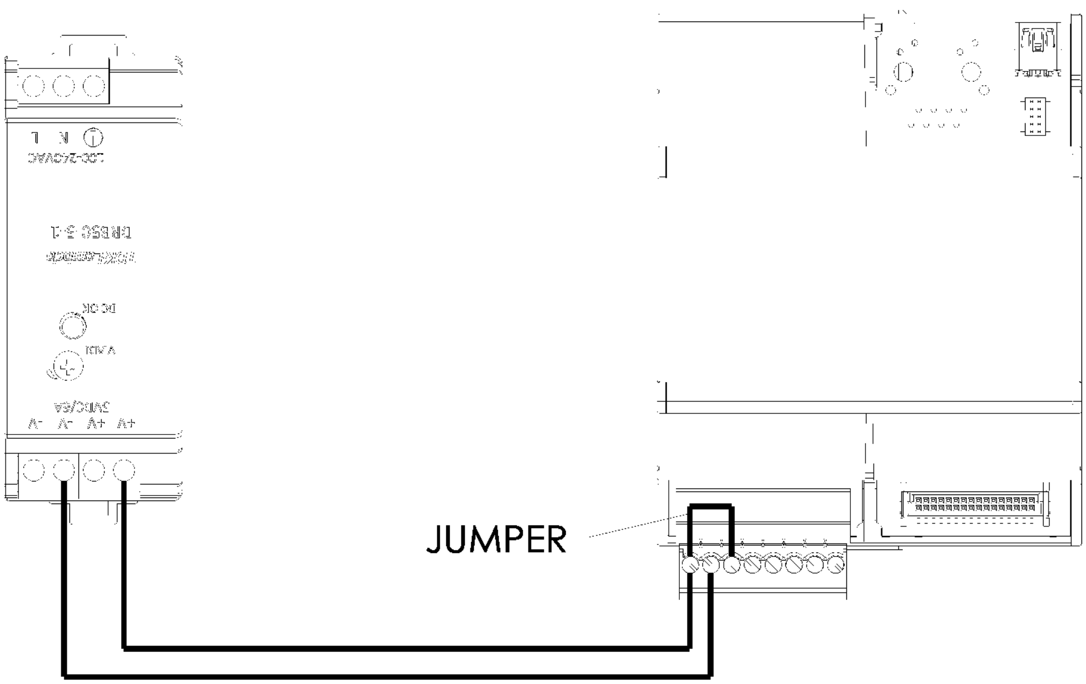
1) Difficult-to-Reach Locations: Narrow corridors, tight spaces, areas crowded with machines and low to the ground applications can hinder wireless connectivity. If the installation environment includes multiple rows of machines, a canyon effect might occur between the rows if wireless devices are installed within a canyon. This effect might hinder communication to these devices. Antenna height can be raised to overcome such an effect, but antenna adjustment is not possible in every case. Another solution is to use the reflectivity of adjacent machines to bounce the signal down into the canyon, taking advantage of the reflective nature of the surrounding equipment.
2) RF Interference: Many common devices produce RF energy: microwave ovens, cell phone hotspots, automobile hotspots, personal Bluetooth devices and many others. RF energy can interfere with factory wireless operations and steps should be taken to ensure the installation environment is protected from RF interference as much as possible. Distance, containing rooms and changing RF band are a few of the ways to mitigate RF interference from outside devices, and should be

implemented if RF interference is prevalent in your factory's installation environment.

3) As stated above in the ceiling mount section, ensure pipes, conduit and any other metal or concrete structures are accounted for during installation. Wireless communication can be affected negatively by any such structures. Steps should be taken to optimize location and line of site between all NANOs and X8IIs. Failure to take these suggestions into account during the initial installation might lead to intermittent signal issues, or worse no signal transmission at all.

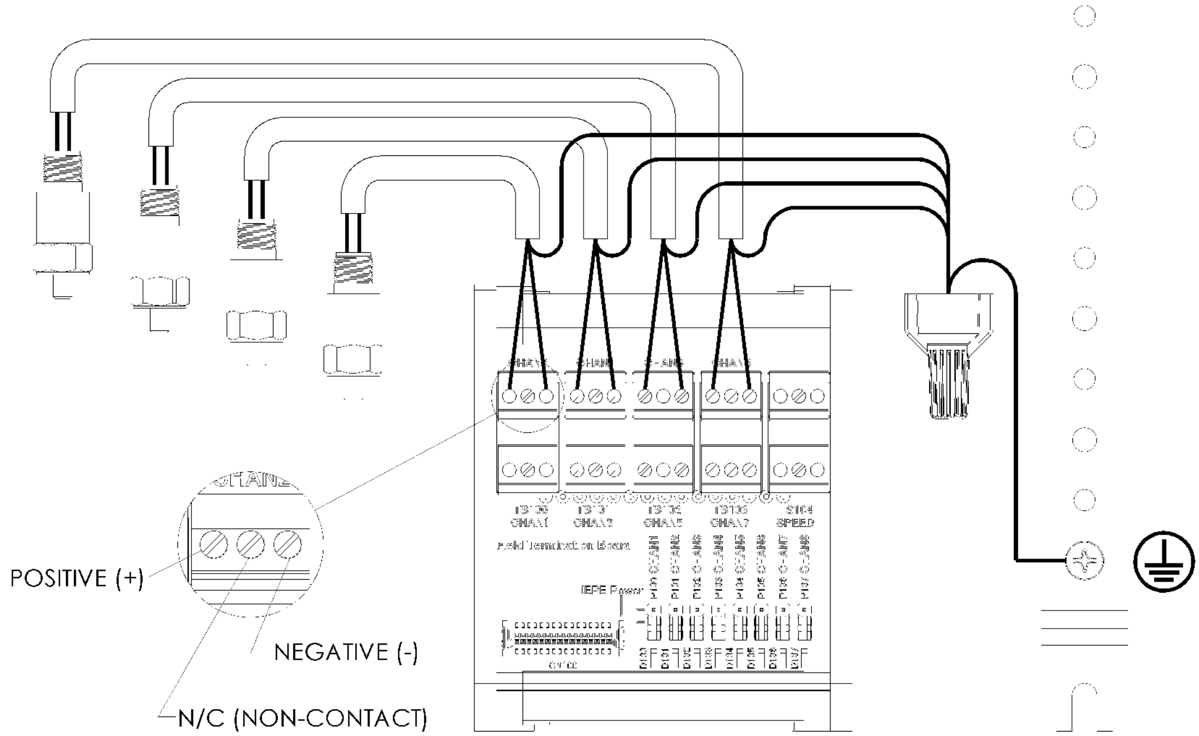
2.5 WIRING DIAGRAMS

2.5.1 DIN RAIL – Stand Alone

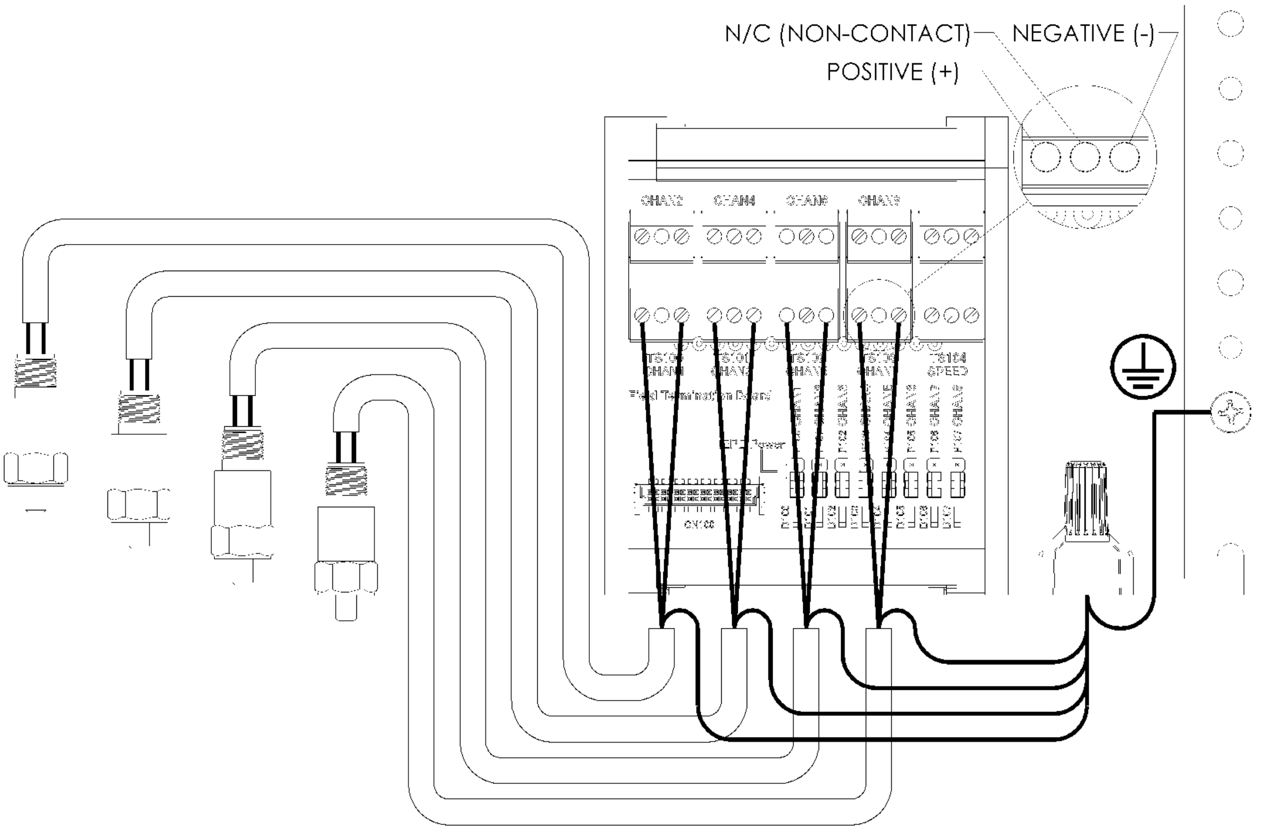


DIN Rail Power Supply

2.5.2 IEPE powered Accelerometers

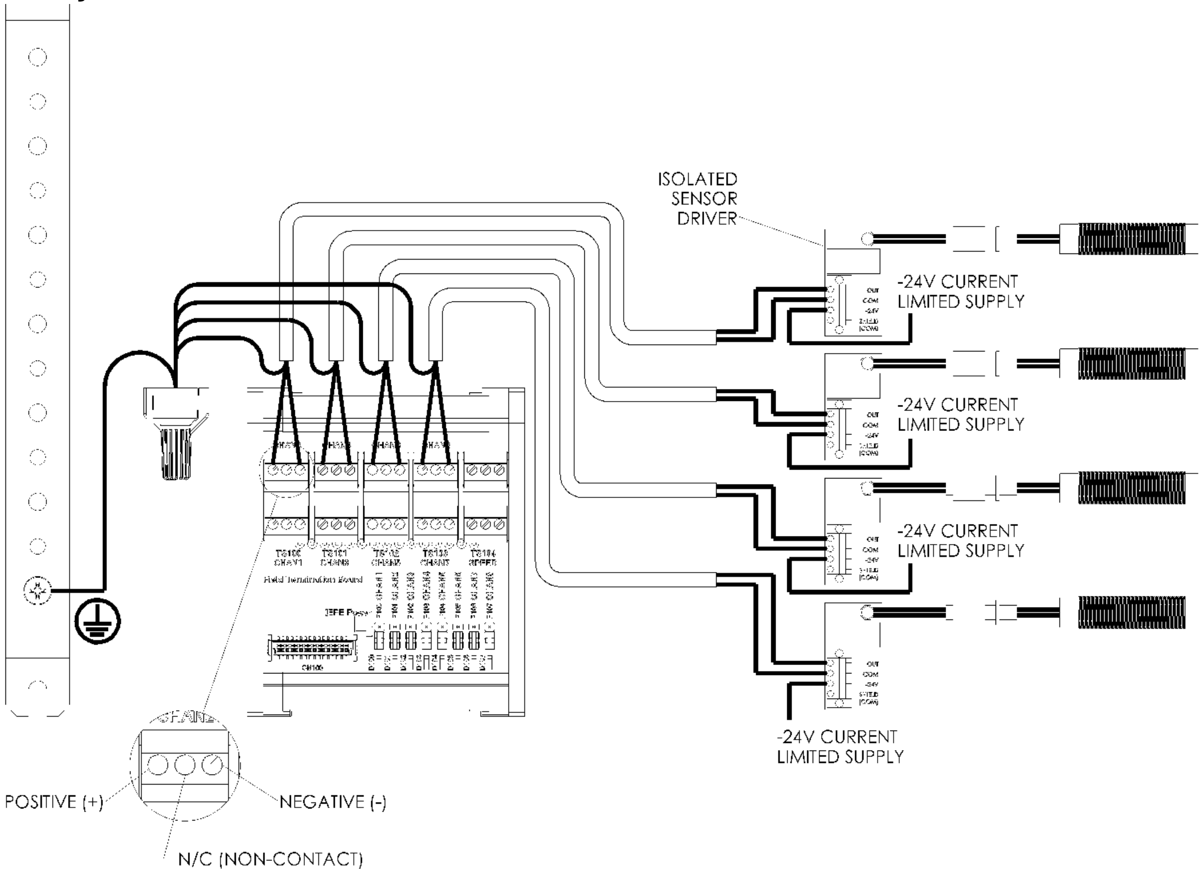


Even Channels

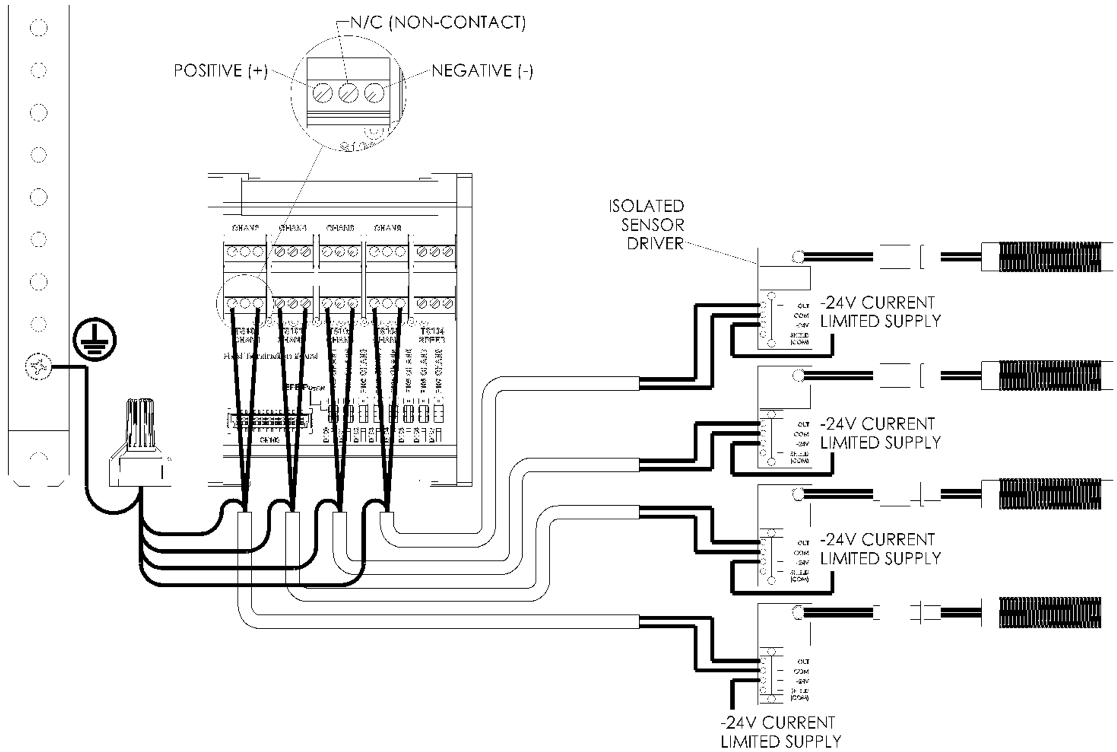


Odd Channels

2.5.3 Eddy Probes

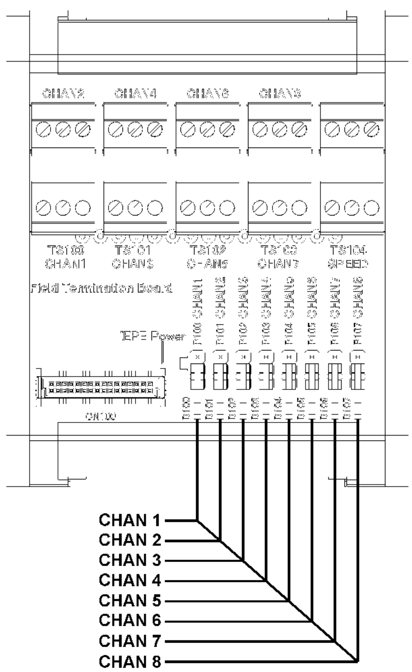


Even Channels

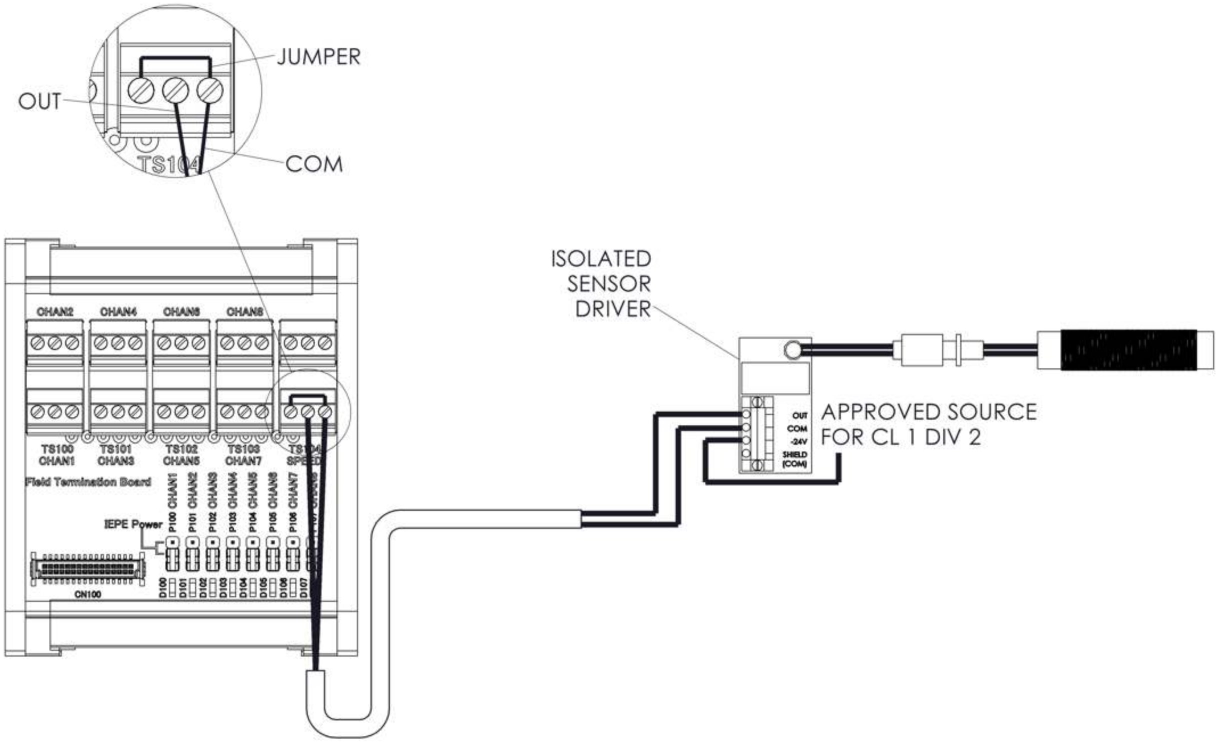


Odd Channels

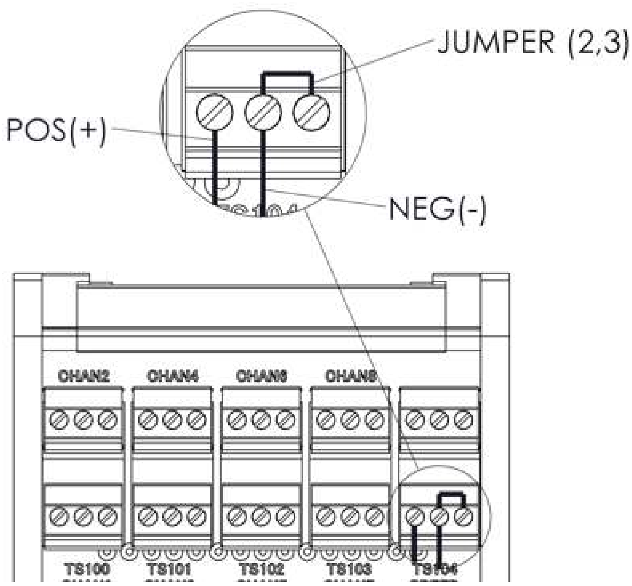
2.5.4 IEPE Power Programming



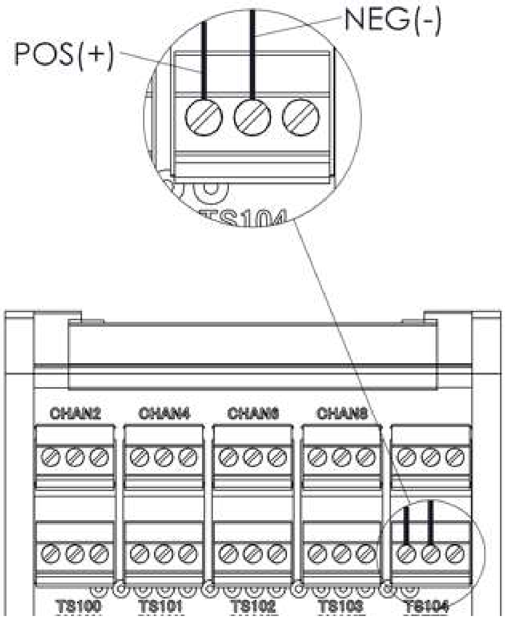
2.5.5 Eddy Probe – Speed Input



2.5.5.1 Powered Hall Effect Wiring



2.5.5.2 Floating Input Signal Wiring



3 WIRELESS NETWORK SETTINGS AND INSTALLATION

This section is designed for IT departments to understand the X8II wireless device and how it connects. Please contact support@sec-america.com or 1-866-602-2245 with questions.

3.1 X8II TRANSMISSION SPECS

Wireless Technology:	Wireless LAN / Wi-Fi
Standard:	IEEE 802.11 b/g/n
Certifications:	FCC / CE / IC / TELEC / SRRC / NCC
Supported Network Security Protocols:	Module is based on Broadcomm BCM43438 / Cypress 43438 • Open (no security) • WPA • WPA-2 Personal (AES) • AES (IEEE 802.11i compatibility) • WPS • TKIP • Cisco Compatible Extensions (CCX, CCX 2.0, CCX 3.0, CCX 4.0, CCX 5.0)
Transmit Power:	Typ. 14 dBm ± 2 at 802.11b CCK Mode 1M Typ. 12 dBm ± 2 at 802.11g OFDM Mode 54M Typ. 12 dBm ± 2 at 802.11n OFDM Mode MCSO
Receiver Sensitivity:	Typ. -95 dBm at 1 Mbps Typ. -75 dBm at 54 Mbps Typ. -89 dBm at MCSO
RF Data Rate:	Up to 65 Mbps
Channels:	13
Application Protocol:	Packet Based Proprietary Protocol : ZEEP (MD) FTP : EMonitor
Encryption:	AES-256
Function:	End Node

3.2 IP SETTINGS FOR X8II COMMUNICATION

Type	Purpose	DNS	IP	Ports
Zeep	Regular encrypted communication between NANOs and cloud servers	zark.machinedossier.net	54.201.129.216	**UDP**: (6000) **TCP**: (4326) **SFTP**: (22)
FTP	X8II remote firmware update	N/A	52.25.143.23	**TCP**: (20-21, 18100-18900)

3.3 TRANSMISSION SIZES

Typical Overall Measurement (Up)	180 Bytes
Typical Spectrum Measurement (Up)	12800 Bytes
Typical Waveform Measurement (Up)	32768 Bytes

Zeep Message (Down)	Typical 300 Bytes
Over-The-Air Firmware Upgrade (Down)	Typical 850KB
EMonitor FTP	Typical 80KB

Measurement messages contain mounting location and X8II and NANO serial numbers for relation in the database, and respective measurement values based on type. The typical measurement taken is an overall value to display vibration trend levels. When vibration levels are above alarm set points, spectrum and waveform data can be collected.

Zeep Messages contain success/failure error messages and can pass new time values if they have been changed.

3.4 X8II Communication & Transmission

3.4.1 Zeep Server Communication

All communications with the server are encrypted with AES-256 with strong 384-bit ECC (Elliptic Curve Cryptography) key generation.

An X8II generates its own private key and public key during the serialization/initial firmware load process. Public keys are loaded into the Zeep Server. For all Zeep pings and data transmissions, the X8II sends its encrypted packets. Only authorized units with proper keys can communicate.

3.4.2 EMonitor and infiSYS Lite Communication

The X8II can be setup to communicate with EMonitor and/or infiSYS Lite FTP servers. Communications with the EMonitor and infiSYS Lite FTP servers are currently not encrypted. For additional information on how to setup and test EMonitor or infiSYS Lite FTP servers see sections 4.2 (EMonitor) or 4.3 (infiSYS Lite).

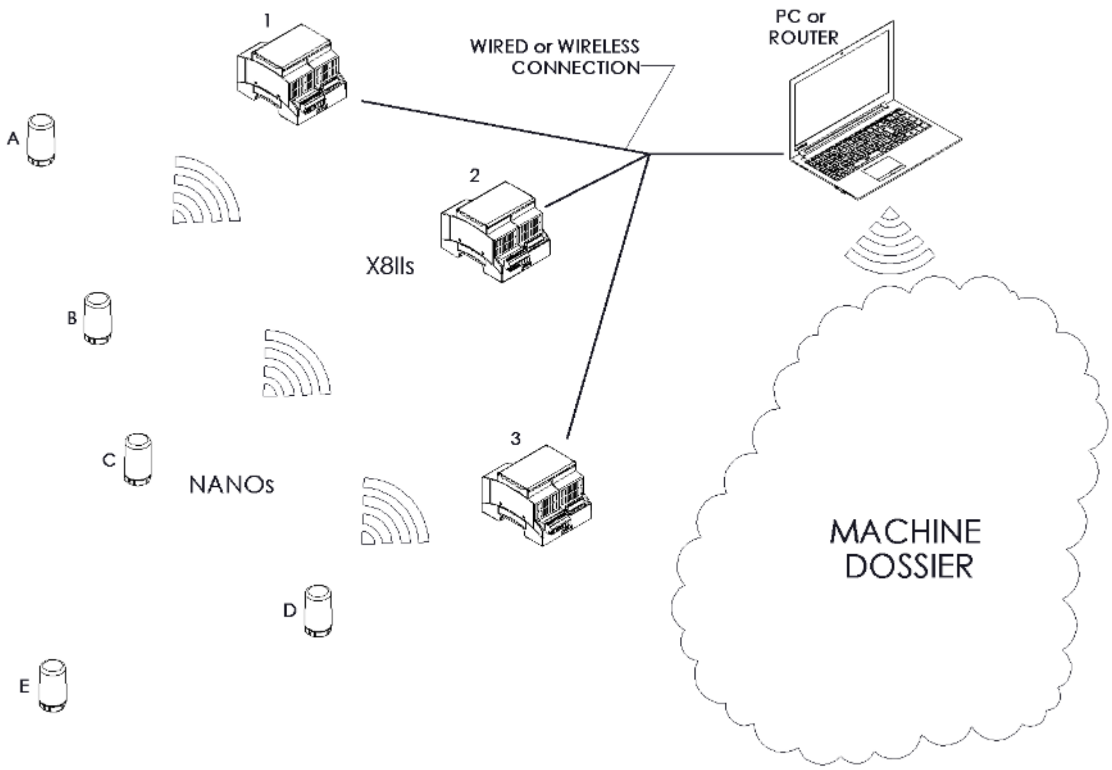
3.5 Communication to MD and EMonitor

The X8II always saves measurements to flash memory when it is determined that the measurement should be sent to one of its destinations (MD or EMonitor). This is done before it tries to send the measurement to its specified destination. In the case of wired channels, it will be determined that it should be saved either because it has reached the scheduled send interval or, if the measurement is between scheduled measurement sends, because the measurement was determined to display a change in the alarm state. In the case of NANO measurements, the X8II will always save measurements to flash memory when received. All measurements saved to flash memory are then placed in a queue to be sent to their respective destinations. These measurements will continually attempt to be sent to their respective destinations until successful.

3.6 OVER-THE-AIR (OTA) FIRMWARE UPDATE

When instructed by the Zeep Server, the X8II will check a list of approved firmware that is hosted on AWS services. If a new version is detected and authorized for the device, it can download the new firmware and install it without requiring any user action or intervention.

3.7 NETWORK TOPOLOGY/COMMUNICATION



Once a NANO has been put into service it will communicate – via Bluetooth transmission – with every X8II within reach of its signal. When a NANO takes measurements, it will send out a signal letting every X8II in range know it is ready to offload measurement data. The first X8II to receive this signal will begin uploading data from the NANO. For example, if NANO C is ready to offload data and sends an advertisement signal out letting X8IIs 1, 2 and 3 know it is ready, the first X8II to receive this signal will upload NANO C's data. Even though this method is like a traditional mesh network system, it is more efficient on NANO battery life since each NANO can power down after it has successfully offloaded measurement data.

Once the data has been received by an X8II, it can be forwarded via a wired or wireless signal to a PC or router which can then transmit the data to a predefined destination (MD, EMonitor, etc.) for further analysis. NANOs cannot forward data directly to a router or PC and cannot forward data to one another, they only forward data to an X8II. By tracking which X8IIs forward which NANOs data, one can see how well the X8IIs are distributed in the installation environment. Efficiency of communication can then be monitored, and X8IIs can be moved into a formation that optimizes throughput without the need to re-connect NANOs since they will automatically connect with any X8II within signal range.

X8IIs can also communicate directly via a hard-wired USB connection. When choosing to communicate with the X8II in this fashion the user can change the configuration, update firmware, or perform direct connection to a NANO for machine vibration diagnostics. For example, let us say X8II #3 above is connected to NANOs C, D and E. If the end user connects to X8II #3 using a mini-USB to USB (male to male) cable and connects the other end to their PC, they can then log into the User Configuration Tool (UCT) software on their PC and select which NANO (C, D or E) they would like to communicate with and configure.
Tips for maximum performance:

1) Locate NANOs as close as possible to at least one X8II unit.
2) Ensure clear line of site between NANOs and X8IIs. This will minimize communication.
3) Minimize RF interference in the installation environment.

One X8II device is capable of connecting, collecting, and relaying data from up to 16 NANO sensors. Multiple factors can impact the total amount of NANOs that an X8II can handle. These factors are:

1) Measurement interval time. Once a day is a typical sampling rate for these sensors, but if this is increased it could burden the system.
2) Network quality of the installation environment.
3) Visibility of the sensors to the X8II unit. Line of sight is a concern when trying to ensure proper communication between the X8II and every connected sensor.
4) Amount of X8II's and NANO's installed in the environment. If an X8II is installed in an environment where there are many other X8II's and NANO's, the X8II can have its capacity reduced to, but no fewer than, 12 NANO's.

4 X8II DESTINATIONS

The purpose of this section is to guide the user on how to properly setup the X8II to communicate with one of the intended destinations: Machine Dossier, EMonitor or infiSYS Lite.

4.1 MACHINE DOSSIER

The purpose of this section is to guide the user on how to properly configure an X8II machine and a NANO Device in the production environment of Machine Dossier.

4.1.1 Setting Up an X8II

Navigate to www.MachineDossier.net and log in with your credentials. Upon log-in, locate the "Add" button on the top left of the browser. Hover over the button and click "Machine". The name of the machine can be set to anything. The type, size, manufacturer, model, etc. are all fields that are used to describe the machine and they do not impact communication. The X8II functions with eight hard wired channels but it can also be tethered to NANO devices via Bluetooth. To properly configure an X8II, follow the snapshots below.

4.1.1.1 Adding a Machine

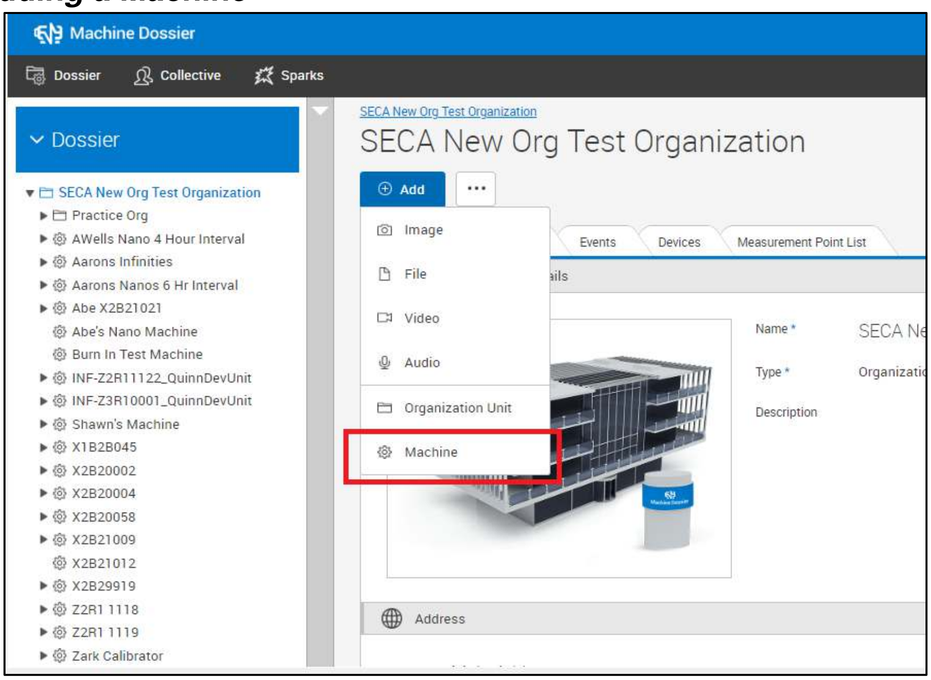


Figure 1: Adding a new machine into MD.

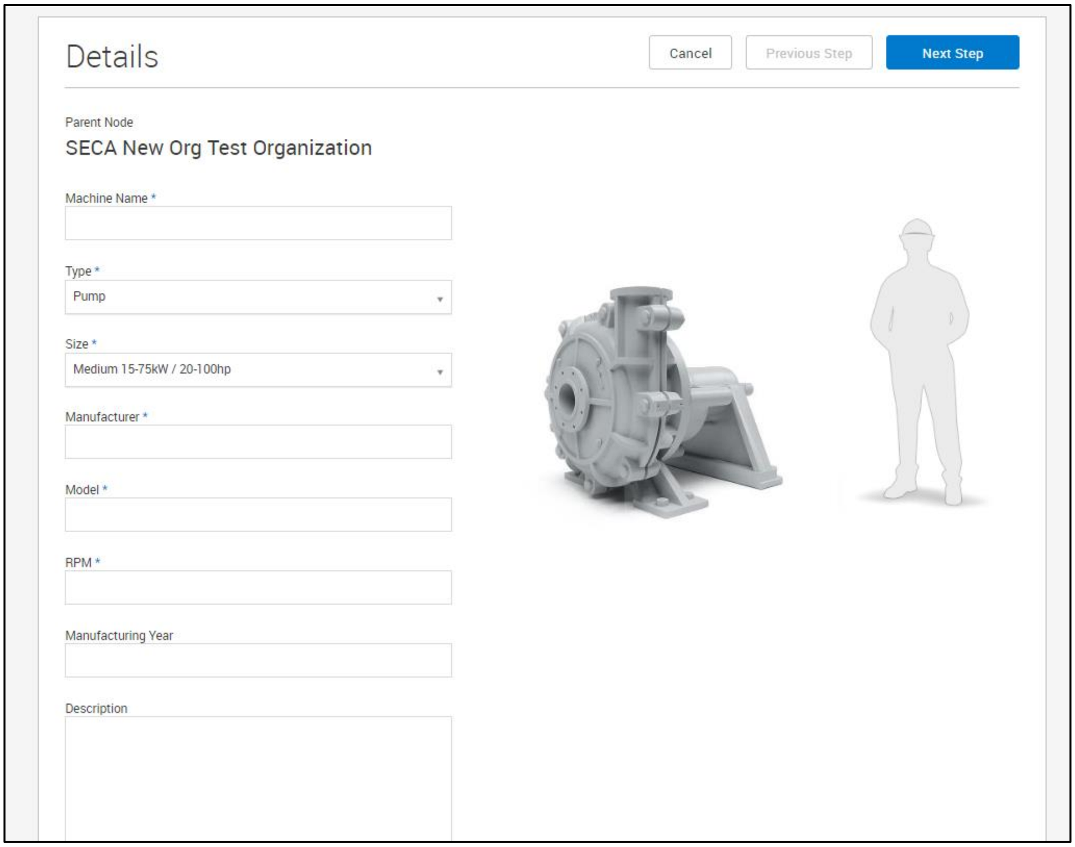


Figure 2: Naming and setting up the X8II machine in MD. Once done click "Next Step" until the device is fully configured in MD.

4.1.1.2 Adding a Measurement Point (Wired)

Measurement points can be added to any existing machine. Follow the images below to add a wired measurement point. **Note**: a machine needs to be selected to add a measurement point. To select the machine, use the pane on the left side of the screen.

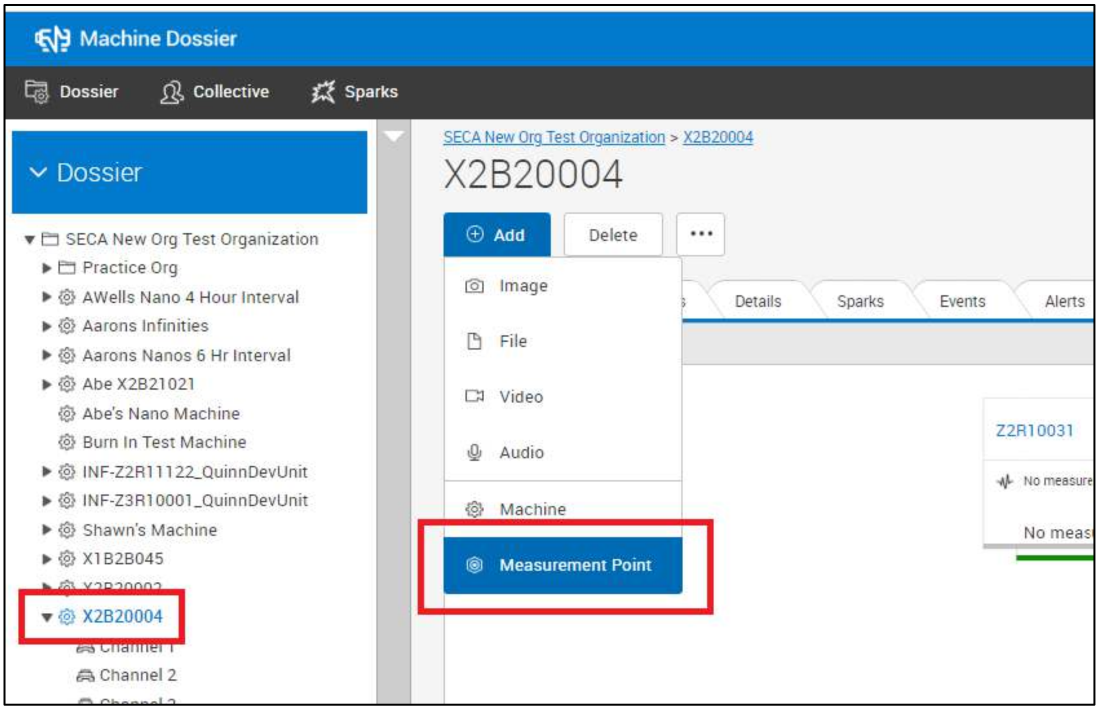


Figure 3: Add a measurement point to the X8II test machine by hovering a mouse over the “Add” button and clicking “Measurement Point.” Notice the machine chosen is X2B20004 and can also be seen in the left pane.

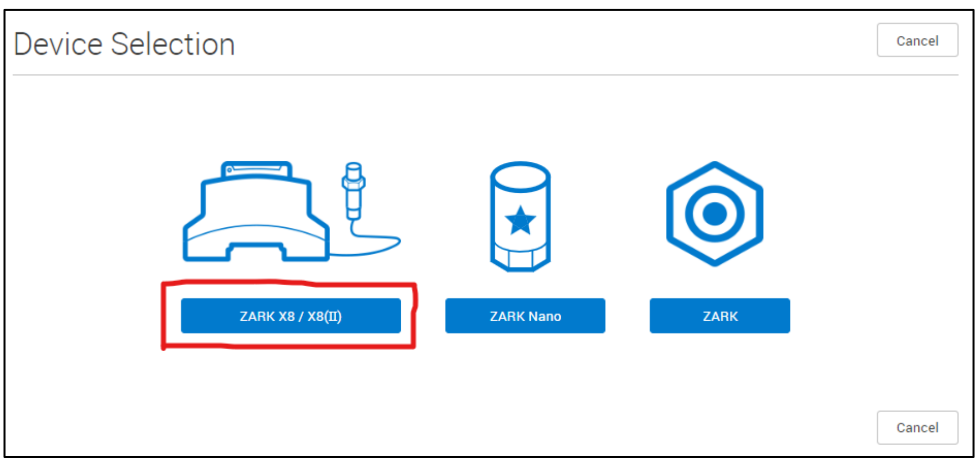


Figure 4: When presented with this screen, make sure to click the button on the left, “Add ZARK X8/X8(II)”.

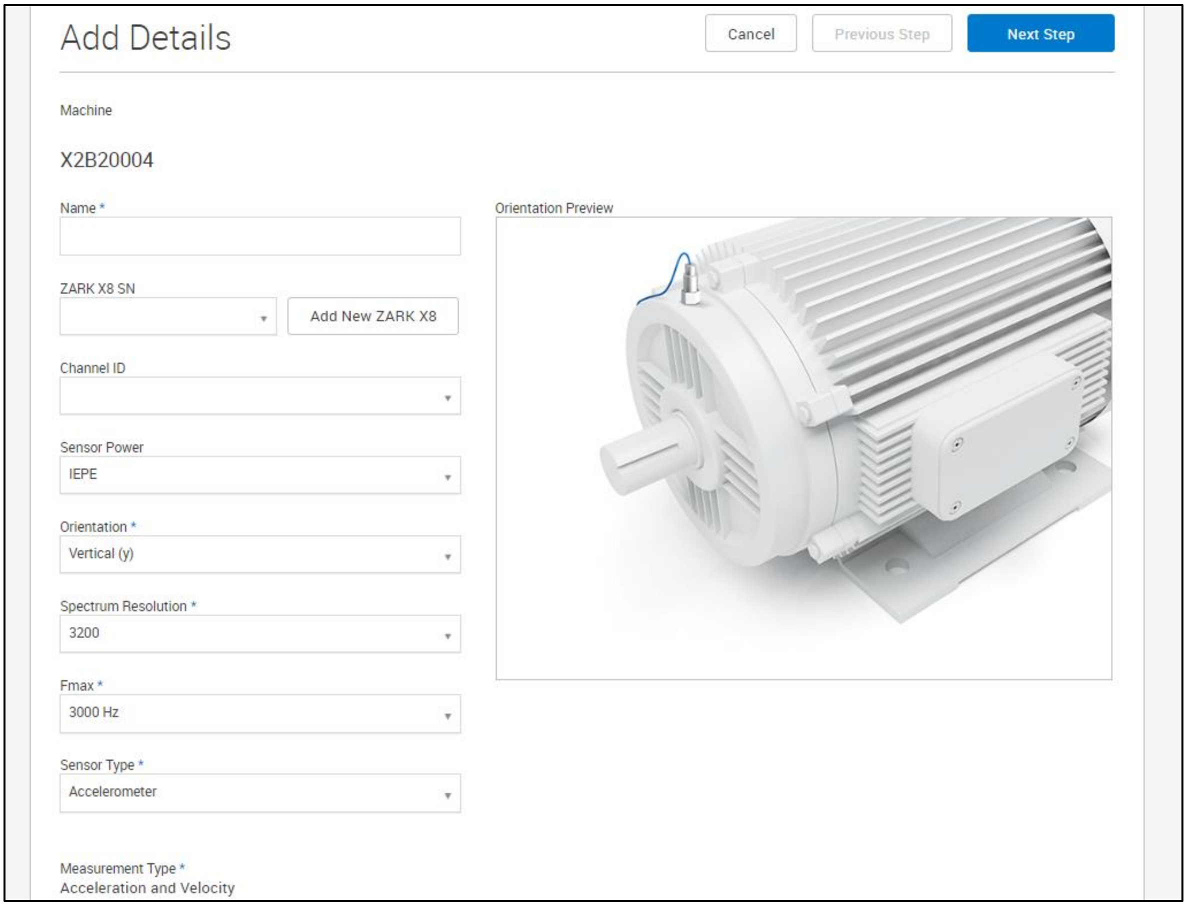


Figure 5: Fill out the details screen with the appropriate information. Defaults can be used for all entry fields except Name, ZARK X8 SN and Channel ID.

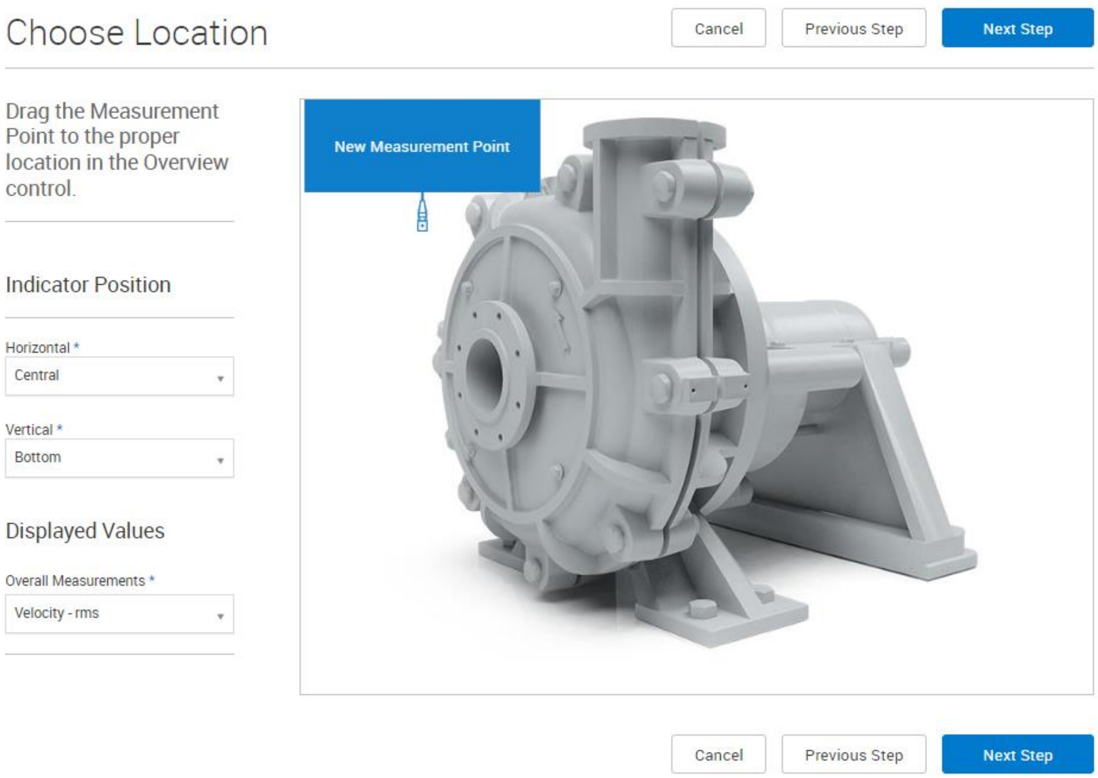


Figure 6: Next choose the sensor position and which measurement type will be displayed.

Once “Next Step” is clicked on the Choose Location screen, a summary screen will appear. Provided the summary is correct, clicking Next Step will create the measurement point.

4.1.2 Adding a NANO

Once the appropriate machine is selected, hover over the “Add” button and click “Measurement Point.” Follow the steps below to add a wireless NANO device.

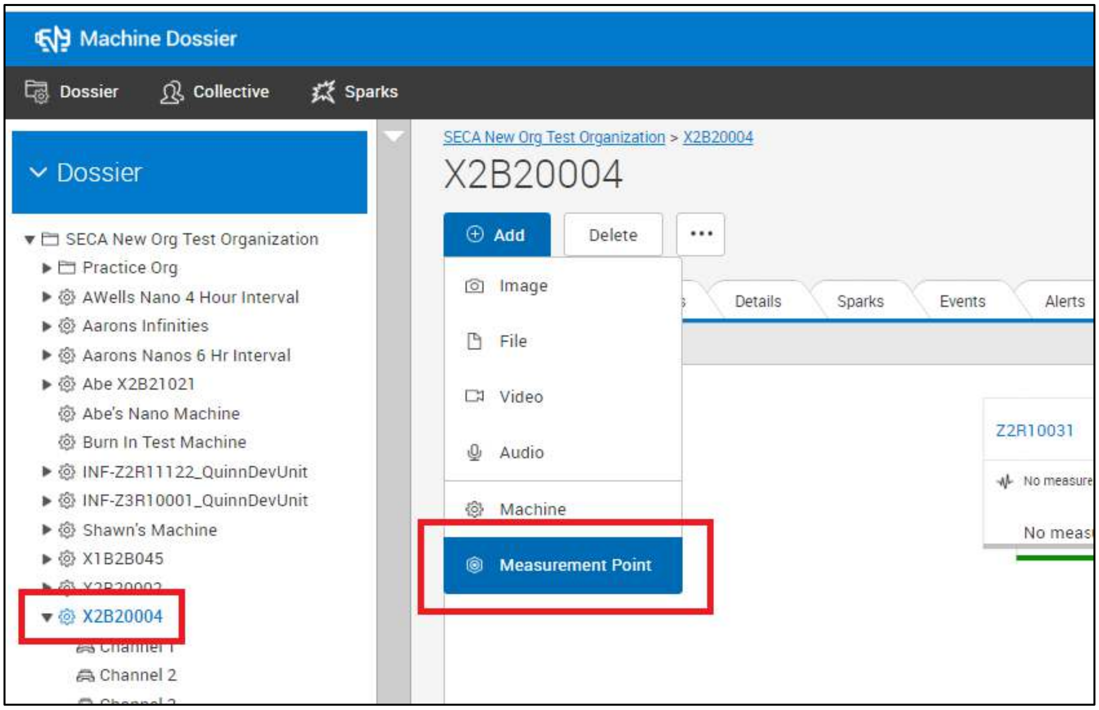


Figure 1: Adding a new measurement point to MD.

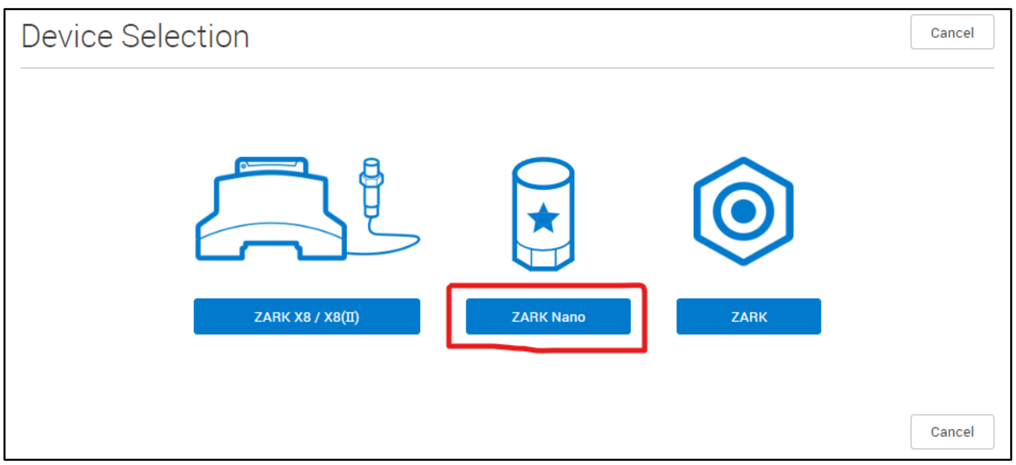


Figure 2: On the “Device Selection” screen click the button in the middle, “ZARK Nano.”

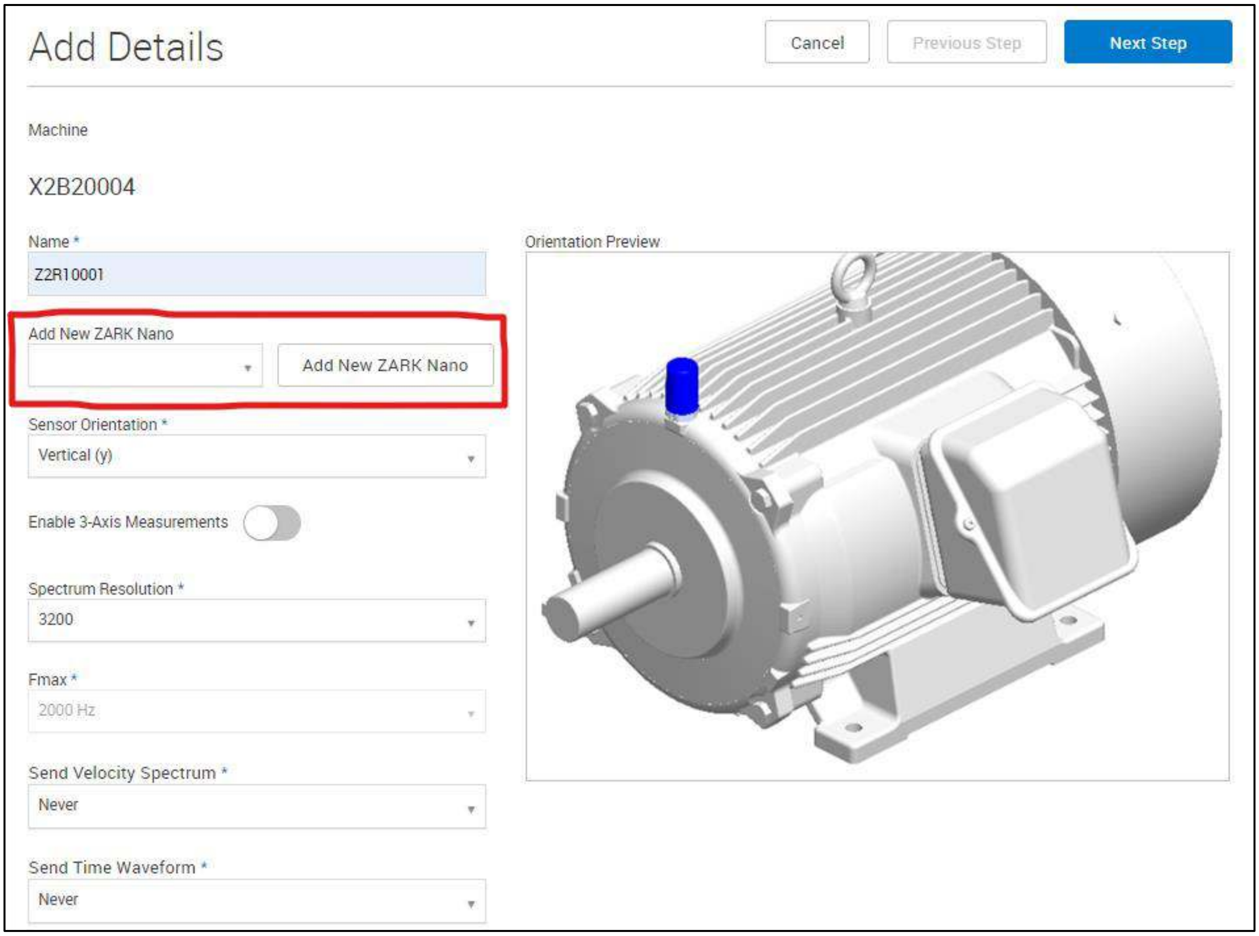


Figure 3: Defaults can be used for most of the information on the “Add Details” form, but Name and Add New ZARK Nano must be filled out to advance. Name can be anything, but typically will be descriptive of location or the device serial number. If adding a new NANO device, the Add New ZARK Nano button must be clicked. Otherwise, an existing NANO can be chosen from the Add New ZARK Nano drop-down menu. If the device you want to add does not show up, the Add New ZARK Nano button must be clicked to set up the device.

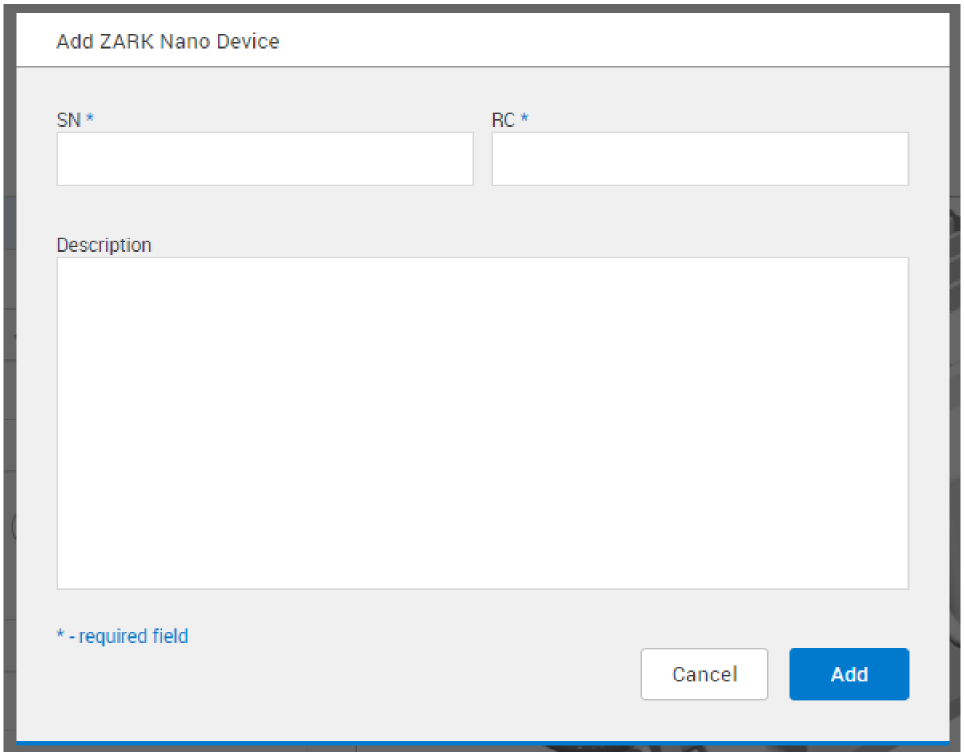


Figure 4: Above is a screenshot of the dialogue box that opens when the Add ZARK Nano Device button is clicked. SN is the device serial number. RC is provided by SECA and specific to each individual unit. Description can be anything.

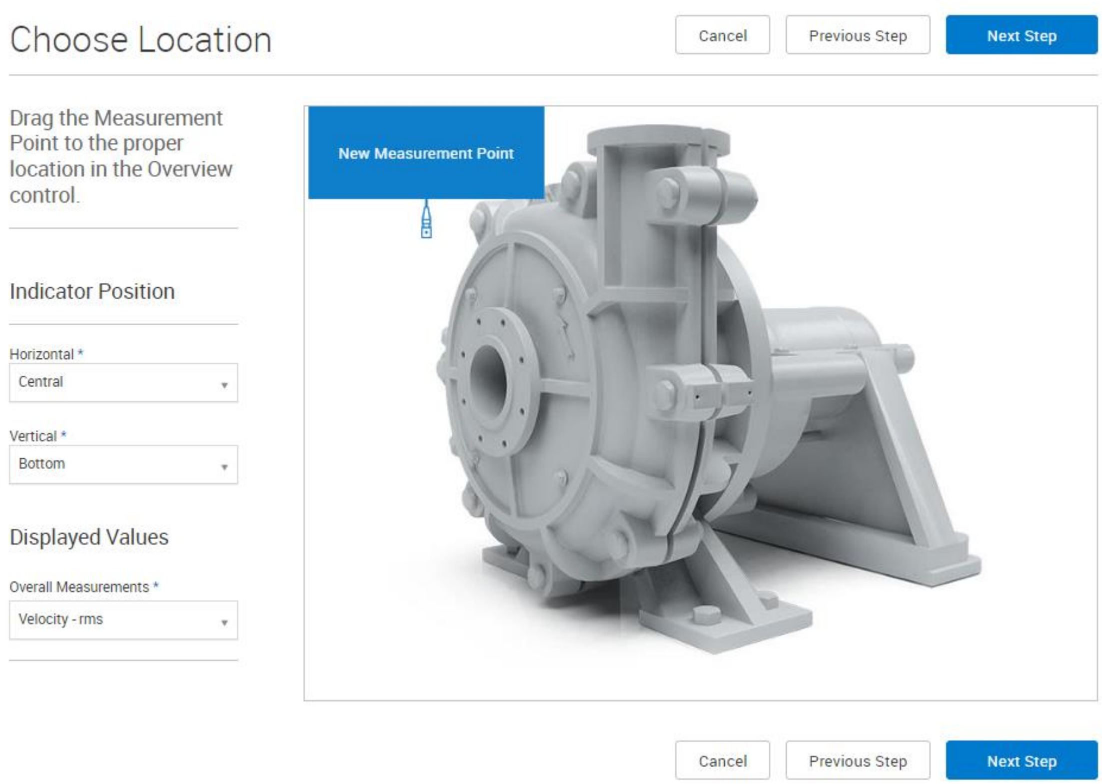


Figure 5: Next choose the sensor position and which measurement type will be displayed.

Once “Next Step” is clicked on the Choose Location screen, a summary screen will appear. Provided the summary is correct, clicking Next Step will create the measurement point.

4.2 EMONITOR

This section will describe how to setup, configure, and test an FTP server for EMonitor communication. EMonitor is designed, manufactured, and sold by Rockwell Automation. Please refer to the manual published by Rockwell Automation for more information on how to work within the EMonitor program once a connection has been established.

4.2.1 Installing an FTP Server for EMonitor

The purpose of this section is to provide a brief description of how to configure a computer to install an FTP Server for the purpose of allowing the X8II to ftp to this server, and for an EMonitor Scheduler to collect a measurement file and import it into EMonitor.

4.2.1.1 Requirements

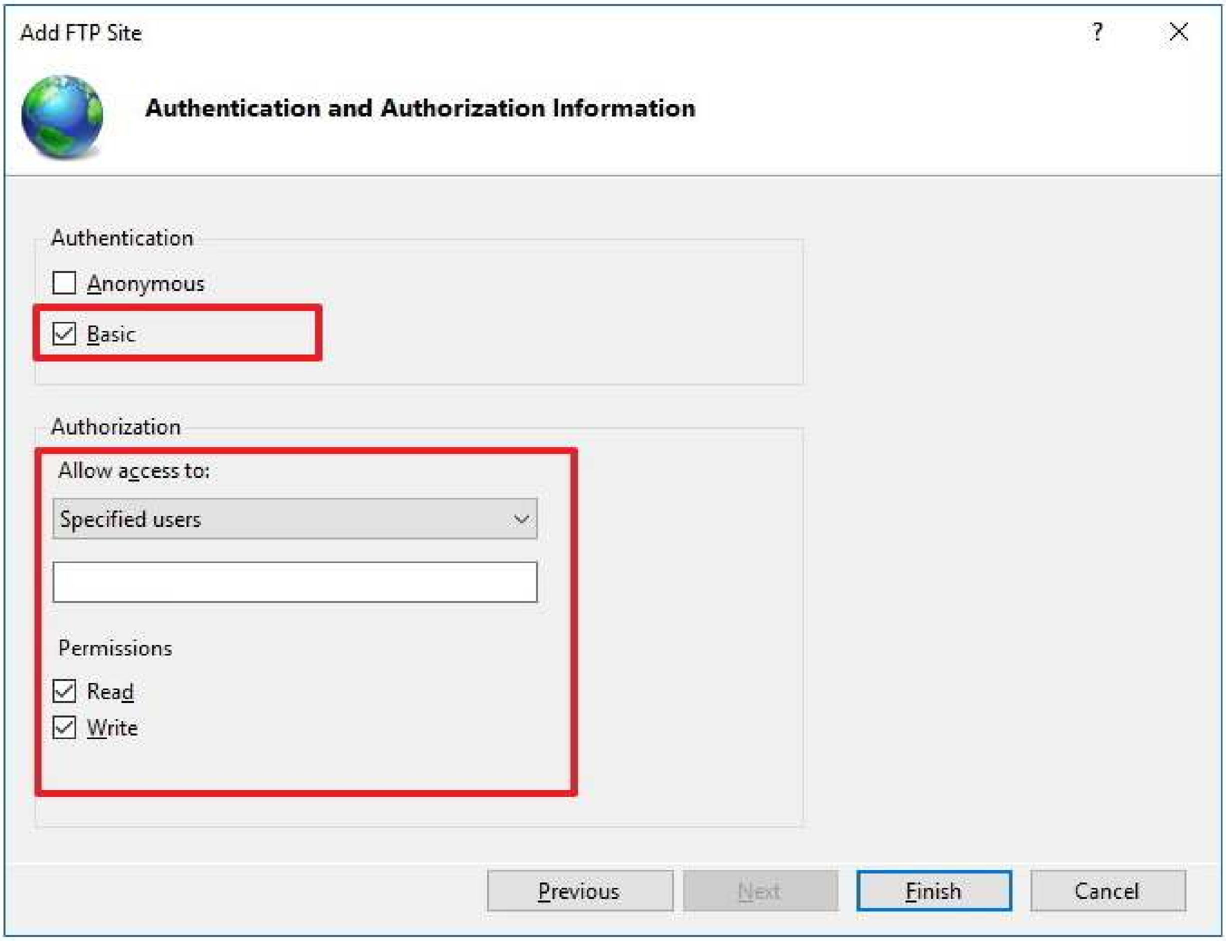
The main requirement is a computer that has EMonitor installed on it, and preferably has Windows 10. If it has an earlier version of Windows, it is up to the reader to research how to install an FTP server on the chosen computer. This guide expects that the computer will be connected to a local network, and the same network as the X8II. **If the computer and the X8II are not on the same local network, this will not work.**

It is important to know the file path of the EMonitor Scheduler folder before installing the FTP server. This is because you will be asked to navigate to a folder on the computer to specify the FTPRoot folder, which is the folder that will receive FTP files. It is important that the X8II sends files to the folder that EMonitor Scheduler expects.

4.2.1.2 Web-based instructions

The easiest way to install an FTP server on a computer that hosts EMonitor, and that is on the same local network as the X8II, is to follow detailed step-by-step instructions available with an internet search.

For instance, for Window 10, you could follow the instructions at this site: https://techgenix.com/ftp-server-windows-10/. When setting up the FTP site, you will be asked to choose “anonymous” or “basic”:



The guide tells you to choose "Basic". This will require a specific username and password from the X8II. EMonitor FTP credentials are dependent on how the host server has been set up. "Anonymous" can also be used to configure the FTP server and might be preferred depending on your network environment.

Follow the instructions at the link provided, and do not bother with "Port Forwarding" because that is only for a scenario where the X8II needs to send the data outside the local network to another network across the internet, which is not supported.

http://techgenix.com/ftp-server-windows-10/

If you have Windows 7 on the computer that you wish to install an FTP server, you can use this Microsoft Link:

https://docs.microsoft.com/en-us/iis/install/installing-publishing-technologies/installing-and-configuring-ftp-7-on-iis-7

4.2.1.3 *After FTP Server Installation*

Once the installation is completed, test the FTP server by "Pinging" it from another computer on the same local network.

To do this, you need to find the local IP address of the computer that has the FTP server on it. On Windows 10 computers, Right-Click on the Windows Start button, and choose "Windows PowerShell" or "Windows PowerShell(Admin)". This will pop up a command line console. Type "ipconfig" in that command line console window. This will provide you with the various IP addresses that the computer is associated with. Look for the Wireless Network section that looks like this:

```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : Home
   Link-local IPv6 Address . . . . . : fe80::a45e:574:69d1:df67%12
   IPv4 Address. . . . . . . . . . . : 192.168.1.116
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1
```

The IP address that you are looking for is the IPv4 Address in the Wireless LAN adapter Wi-Fi section. In this case, it is "192.168.1.116". Write this down because you will need this address to attempt a "ping" from another computer.

Now, go to another computer that is on the same local network. In this case, the FTP server is 92.168.1.119. So, for example, type:

- ping 192.168.1.119 (or whatever your local IP address for the FTP server is).

You should be able to see replies from the FTP server computer that resemble this:

```
Pinging 192.168.1.119 with 32 bytes of data:
Reply from 192.168.1.119: bytes=32 time=48ms TTL=128
Reply from 192.168.1.119: bytes=32 time=3ms TTL=128
Reply from 192.168.1.119: bytes=32 time=4ms TTL=128
Reply from 192.168.1.119: bytes=32 time=3ms TTL=128
```

Next, from the same command line console window, try to connect to the FTP server. Choose some small text file and try to ftp it to the new FTP server:

```
PS C:\users\aaron\documents> ftp 192.168.1.119
Connected to 192.168.1.119.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.1.119:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
ftp>
```

In this case, for user, anonymous was entered, and random letters used for the password, since the anonymous user does not require a password. Using this user and password, the FTP server allowed the login.

Next, type "put" and a space and the name of the test file to ftp to the FTP server:

```
ftp> put test.csv
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
ftp: 256 bytes sent in 0.37Seconds 0.69Kbytes/sec.
ftp>
```

If you succeeded, your FTP server should be ready to accept EMonitor files from the X8II (provided it is configured to be on the same network as the EMonitor server). All you need to do is to update the User Configuration file FTP IP credential to match the IP address from the FTP server. This can be done in the "EMonitor Controls" tab of the User Configuration Tool (UCT). After this is complete, the X8II will begin to collect measurements and send them to the FTP server.

4.2.2 Test for Correct EMonitor FTP Server Setup

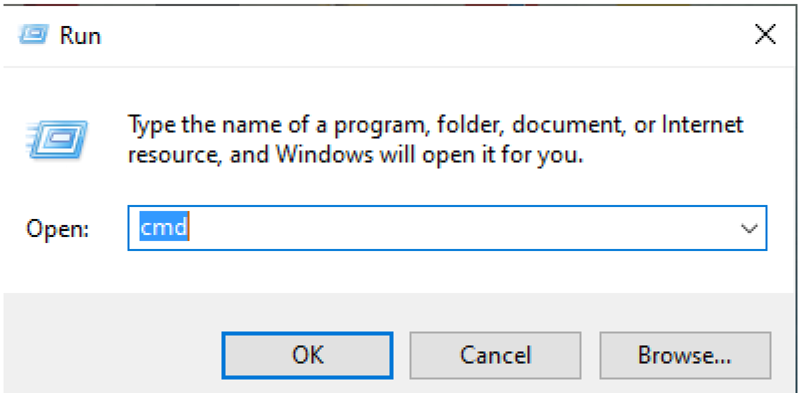
The purpose of this section is to describe a method for confirming that the FTP Server has been set up properly for EMonitor functionality, and that the IP address can be used manually to FTP a file to the UD folder.

4.2.2.1 Requirements

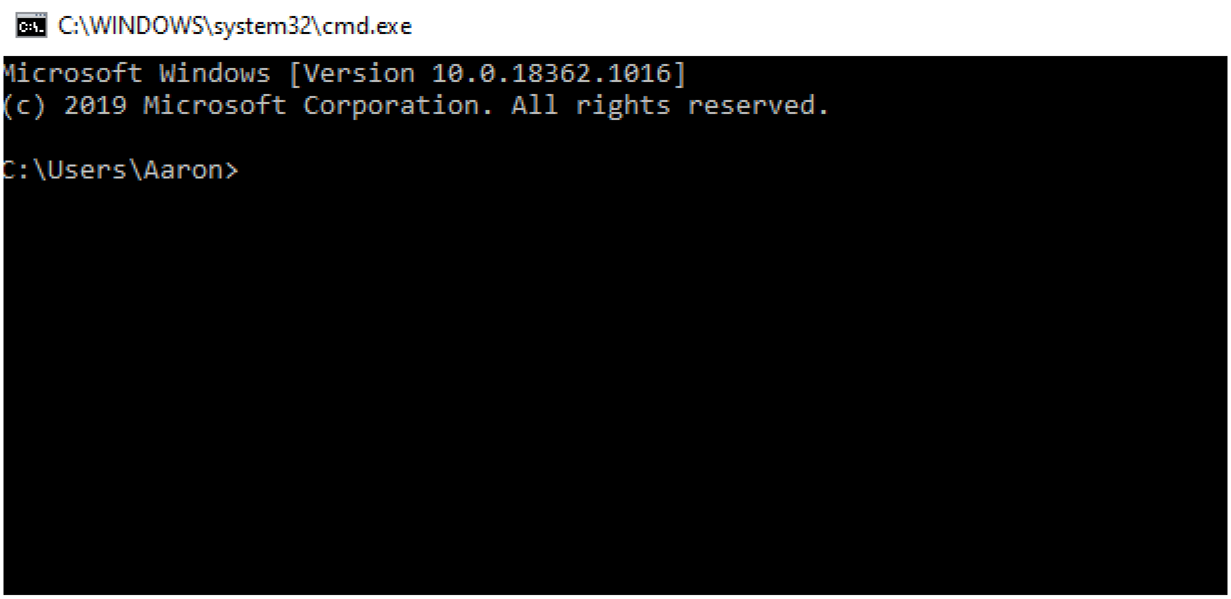
The main requirement is that the device that is trying to FTP to the EMonitor Host FTP Server (whether it be another PC or the X8II unit) is on the same local network as the EMonitor Host FTP server. It is possible that, even though the devices are on the same network, some security settings may prevent them from communicating. Please check with your IT department to see if such security settings exist. If for some reason the instructions below fail, check with your IT department.

4.2.2.2 Get the IP address of the EMonitor host PC

To get the IP address of the EMonitor host PC, in Windows, right-click on the Start menu button in the lower left and Click "Run".

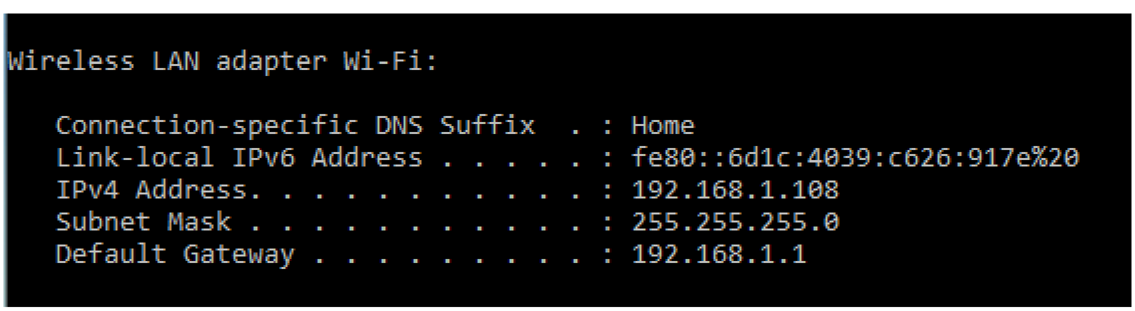


Then type “cmd” into the Open field and click “OK”. This will open a terminal window like below:



In this terminal window, type “ipconfig” and enter.

Search the output for a section that says, “Wireless LAN adapter Wi-Fi”. I should be near the bottom.



Look on the line that says “IPv4 Address…..” and write down the IP address. In the above example it is 192.168.1.108. This will also be the FTP IP address.

4.2.3 Testing the IP Address of the EMonitor Host PC

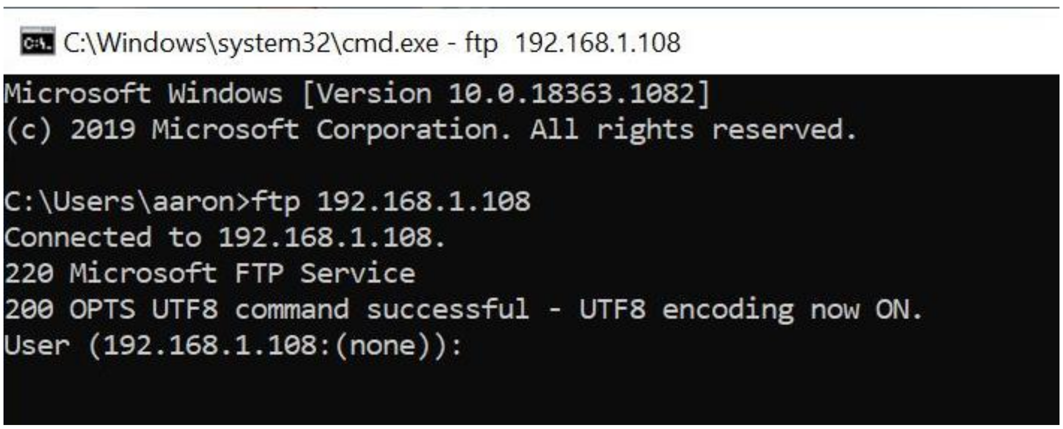
Log onto another PC on the same Network and create a test file. Now that we know the IP address of the EMonitor Host PC, we need to confirm that a file can be FTP’d to that address and that it shows up in the “ud” folder of the EMonitor system. To do this, go to another PC on the same local network as the EMonitor Host PC and create a test text file, such as test.txt and save it to the disk where you can locate it later.

4.2.3.1 Perform a Test FTP

Create a CMD terminal on this other PC the same way that you did above. Use the IP address gathered from above to perform a manual FTP to the EMonitor Host PC. In our example above, the EMonitor Host FTP IP address was “192.168.1.108”.

In a CMD terminal, type:

ftp 192.168.1.108



For User, type “anonymous”. You will then be prompted for a password. Since this is anonymous FTP, a password is not required. So, you could type anything, or just press enter.

After that, could type “dir” to see what is in the root directory of the FTP server:

```
C:\Users\aaron>ftp 192.168.1.108
Connected to 192.168.1.108.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.1.108:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection.
08-06-20  10:20AM                 25884 X1B21958_CHANNEL-1_200806_101947.mde
08-06-20  11:32AM                 25878 X1B21958_CHANNEL-1_200806_112246.mde
08-06-20  11:45AM                 25884 X1B21958_CHANNEL-1_200806_114441.mde
08-06-20  12:08PM                 25890 X1B21958_CHANNEL-1_200806_120627.mde
```

Copy the test.txt file into the folder where your command prompt opened, and then type this in the ftp prompt:

put test.txt

```
C:\Users\aaron>ftp 192.168.1.108
Connected to 192.168.1.108.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.1.108:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
ftp> put test.txt
```

You should see something like this:

```
Connected to 192.168.1.108.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.1.108:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
ftp> put test.txt
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
ftp: 9 bytes sent in 0.00Seconds 9.00Kbytes/sec.
ftp>
```

You can confirm that the file successfully was transferred by typing this:

dir test.txt

```
Connected to 192.168.1.108.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.1.108:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
ftp> put test.txt
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
ftp: 9 bytes sent in 0.00Seconds 9.00Kbytes/sec.
ftp> dir test.txt
200 PORT command successful.
125 Data connection already open; Transfer starting.
09-09-20  12:32PM                    9 test.txt
226 Transfer complete.
ftp: 52 bytes received in 0.00Seconds 26.00Kbytes/sec.
ftp>
```

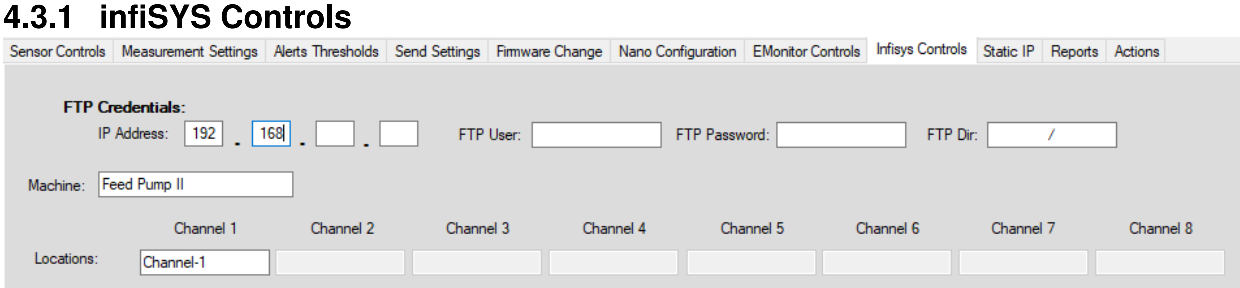
This gives you the information about test.txt on the FTP server site.

It is left to the user to confirm that the location of the FTP folder location is correct for the EMonitor installation.

4.3 INFISYS LITE

This section will describe how infiSYS Lite communicates with the X8II and how to configure this communication. To configure the X8II to communicate with infiSYS Lite, the user will need to physically connect to the X8II and use the User Configuration Tool (UCT). Additional information on the UCT can be found in section 6.

4.3.1 infiSYS Controls



FTP Credentials

To establish an FTP connection with the infiSYS Lite destination, the user will need to enter in the following information:

<u>IP Address</u>: IP address of infiSYS Lite destination.

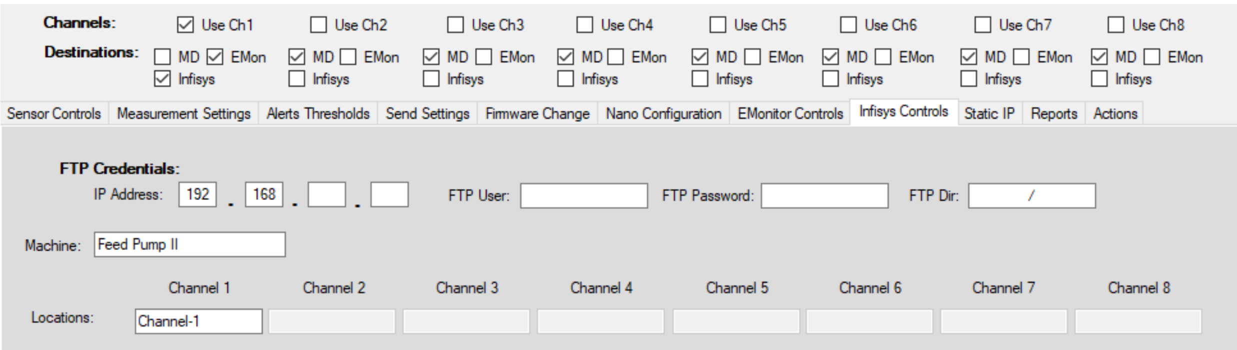
<u>FTP User</u>: Username of FTP server. Customer can choose whatever name they desire.

<u>FTP Password</u>: Password protection for infiSYS server.

FTP Dir: Directory of FTP server.

Machine
User specified name for specific machine being monitored. The Machine name goes into the .csv file that is produced by the X8II and sent to the specified FTP IP Address.

Locations
Wired
Each wired channel is entered in the form seen below and included in the .csv file for wired channels.



In the example above, the Machine has been set to Feed Pump II. Only wired channel 1 is configured. Note that the infiSYS checkbox is checked as a destination for wired channel 1. When wired channel 1 is measured after configuring it as above, the X8II will save a measurement to flash memory for the specific destination of the infiSYS FTP IP address. After the measurement has been saved to flash memory, a real time operating system (RTOS) task, whose sole job is to find saved measurements destined to go to the infiSYS Lite destination, will find that measurement and prepare an FTP file based upon that measurement data and then FTP that file to the specified FTP IP address. This FTP file will include the Machine name and the location specified for wired channel 1.

NANOs (Wireless)
In the NANO Configuration tab (more information can be found in section 6.5.6), infiSYS Lite is included as a destination:

In the above example, the user has included infiSYS Lite as a destination for this NANO and specifies a location. When the NANO sends its measurement to the X8II, the X8II observes the destinations, and if one of the destinations is infiSYS Lite, it saves a copy of the measurement destined to go to the infiSYS FTP IP. Then, the RTOS task specifically searches the saved measurements and finds the new infiSYS-destined measurement, prepares the .csv file, and FTP's it to the infiSYS FTP IP destination. This will use the location specified in the form above, and the machine name will be extracted from the Machine name in the main Infisys Controls tab.

4.4 CANOpen Setup

The purpose of this section is to walk the user through the process of setting up a Controller Area Network (CANOpen) via a Programmable Logic Controller (PLC).

4.4.1 Requirements

- X8II Unit system (running firmware version 2.2.6 or later)
- Modified 8-pin power connector
- CAN-High and CAN-Low wires
- 2 termination resistors (must be 120 Ω, can combine resistors to attain R_{eq} of 120 Ω if necessary)
- PLC Device and a power supply capable of generating 24 V at 0.7 A.
- One micro-USB cable (two is recommended)

4.4.2 Hardware Set-Up

Modify the X8II 8-pin power connector as described below, ensuring that wires across pins are not in contact. (**Figure 1** shows the modification)

1. Jumper between pins 1 and 8
2. Jumper between pins 1 and 3
3. Jumper between pins 2 and 5
4. 120 Ω resistor between pins 6 and 7
5. CAN-Low wire feeding into pin 6
6. CAN-High wire feeding into pin 7

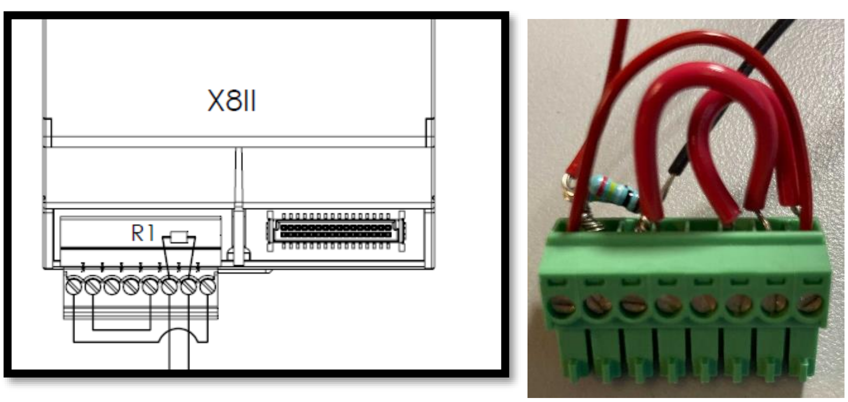


Figure 1: Modification of the 8-pin power connector.

4.4.3 Functionality

As soon as the X8II is powered on, the unit will begin transmitting CAN messages. It does not need to be placed in a special CANOpen mode. A PLC device can be used to read from the unit. At the time of writing this document, the X8II firmware only supports the reading of Velocity RMS values across eight wired channels. As the unit continuously multiplexes through the activated wired channels, the PLC will update the readings. If all eight wired channels are configured, it takes approximately 5 minutes for all eight RMS values to update via CAN. This time interval decreases as fewer channels are activated. However, if the X8II unit is scheduled to send to a destination, whether that be Machine Dossier, Emonitor, and/or InfiSYS, expect longer than 5-minute updates while the unit is sending measurements.

If setup correctly, a PLC is capable of constant SCAN times of 11 seconds, without any signal dropouts or inaccurate readings. **Figure 2** shows float numbers of RMS values, along with the EPOCH timer under device D100 using a test program. The RMS Float values are accurate, and they were tested with various AC input magnitudes.

Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float
D180			14380	1059338284	0.641
D170			29179	1068069371	1.323
D160			3238	1073482918	1.969
D150			15594	1076575466	2.675
D140			-27803	1079415653	3.352
D130			-23692	1082368884	4.113
D120			20156	1083920060	4.853
D100			8426	1623859434	11651918784509666
D110			23866	1085562170	5.636

Figure 2: Device Monitoring table of CANOpen via a PLC.

5 DATA BUFFERING

The NANO device communicates with the X8II via Bluetooth. Upon taking measurements, the NANO waits for communication from the X8II, allowing it to send a measurement. Measurement rates can be configured via the User Configuration Tool (UCT) (Section 6) or Machine Dossier (Section 4.1), with an interval of no less than one hour. Measurement rates are also halved when a NANO is measuring a signal that is specified to be above a certain threshold. However, intervals cannot be halved when resulting in a new interval of less than an hour.

The X8II device can perform multiple tasks at once. Rather than having a linear logic system, the X8II can perform multiple functions simultaneously. Therefore, measurements, sends, and buffers, are more reliable and accurate.

The X8II communicates with the desired destination (MD, EMonitor, etc.) via a wireless (or wired) network connection. The connection is specified via an IP address, allowing the user to collect data and send to virtually any supported platforms that can interpret and display the collected measurements. The sent measurements can either be measurement points, or spectra of frequencies.

6 USER CONFIGURATION TOOL (UCT)

This section will provide a clear step-by-step description of how the X8II can be controlled by a custom Windows Application. The User Configuration Tool is a Windows graphical application that can directly configure an X8II and NANO. Depending on the version, it is also capable of importing, exporting, and retrieving configurations from X8II and NANO units.

6.1 REQUIREMENTS AND RECOMMENDATIONS

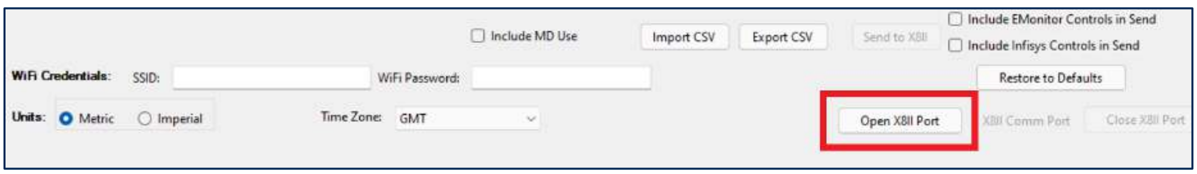
- A micro-USB cable to establish connection with the hardware.
- Access to Microsoft Excel is recommended to access full functionality of the X8II but is not required.
- To have full access to all described features, the X8II must be on firmware version 2.2.1.3 or later.

6.2 DIRECTIONS TO CONNECT TO THE X8II

Prior to launching the UCT program and connecting to the X8II, it is expected that the user has already made a physical connection between the X8II and a PC (with the UCT installed) by using a mini-USB (X8II) to USB (PC) cable (see cable example below).

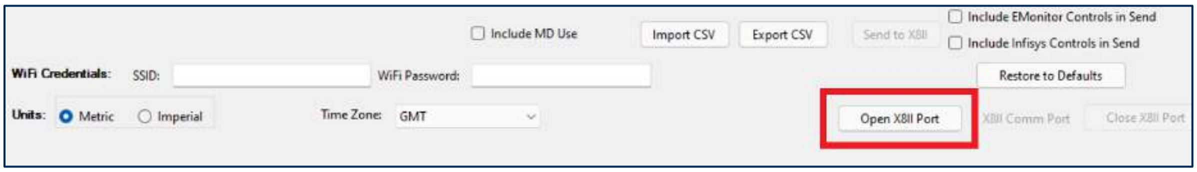
If the X8II is NOT currently powered on, follow this sequence:

1) Power on the X8II and allow it to go through its boot sequence. This shouldn't take any longer than 30 seconds.
2) Power on the PC with the UCT installed. Open the UCT (named X8IIUserConfigTool) and click the "Open X8II Port" button.
3) A dialogue box will appear letting the user know a connection has been established.

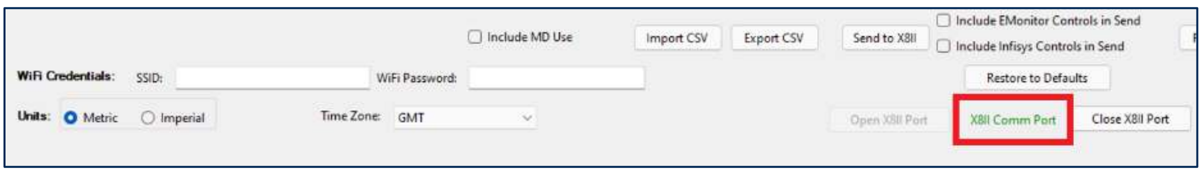


If the UCT is running already and the X8II is also powered up and running, follow this sequence:

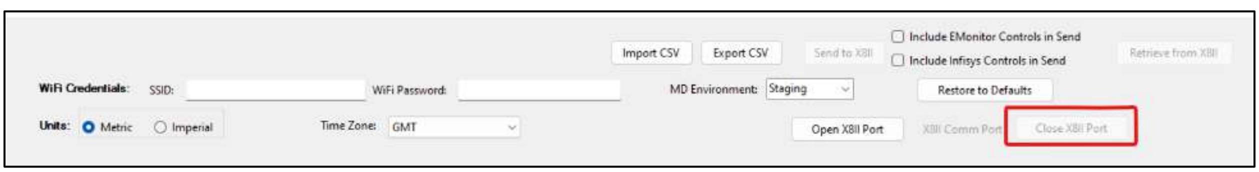
1) Power cycle the X8II
2) In the already opened UCT, click the “Open X8II Port” button.
3) A dialogue box will appear letting the user know a connection has been established.



This will make the X8II available for the UCT to configure. Note the “X8II Comm Port” text will turn green when the port is available.



To disconnect, simply click the “Close X8II Port” button.

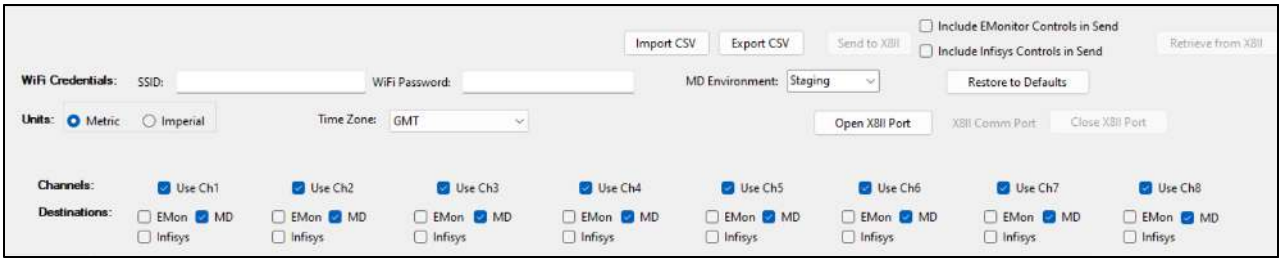


6.3 User Configuration Tool Features

The windows application consists of two main parts, Global Settings and the UCT Ribbon. Contents of each section are described in detail below. Consult the Table of Contents for navigation through the document and the Glossary for explanations of certain abbreviation and terms used throughout this document.

6.4 Global User Configuration Settings

These can be found at the top of the program, above the tabs (UCT Ribbon). The screenshot below shows the Global settings.



Wi-Fi Credentials

- Service Set Identifier (SSID): The name of the network.
- Wi-Fi Password: The password for the specified SSID.

Units

- Imperial vs Metric: Metric is the default, and it will influence the way measurements are displayed in user specified locations.

X8II Connection Settings

- Close X8II Port: This button will close the USB communication port and the X8II will go back to normal operation.
- Retrieve from X8II: This button will retrieve the current settings from the X8II. Upon successful retrieval, all fields and tabs will be filled out with the current configuration settings of the X8II. This should be clicked immediately after connecting to the X8II device.
- Send to X8II: This button will send all current UCT settings to the X8II, overriding any previous configuration. It will test if the WI-FI connection is successful as well. If the Wi-Fi connection test fails, it will pop up a warning, but it will still send all the configurations.
- Restore to Defaults: This button will populate all fields to default settings.
- Export CSV: This button will export and save a .csv file of specified user configurations.
- Import CSV: This button will import a previously exported .csv file and will populate all configurable fields with data from the .csv file. The user can then send this configuration to the X8II.
- Time Zone: Specify which time zone the measurement time stamps will be collected.
- Include Emonitor Controls in Send: If checked, ensures that EMonitor is a destination. This should be checked if EMonitor is chosen as a destination for any wired channels or NANOs and the EMonitor server information should be filled out underneath the EMonitor Controls tab.
- Include Infisys Controls in Send: If checked, ensures that infiSYS is a destination. This should be checked if infiSYS is chosen as a destination for any wired channels or NANOs and the infiSYS server information should be filled out underneath the Infisys Controls tab.

Channels

- The checkboxes control which wired channels will be used while the X8II is operating.

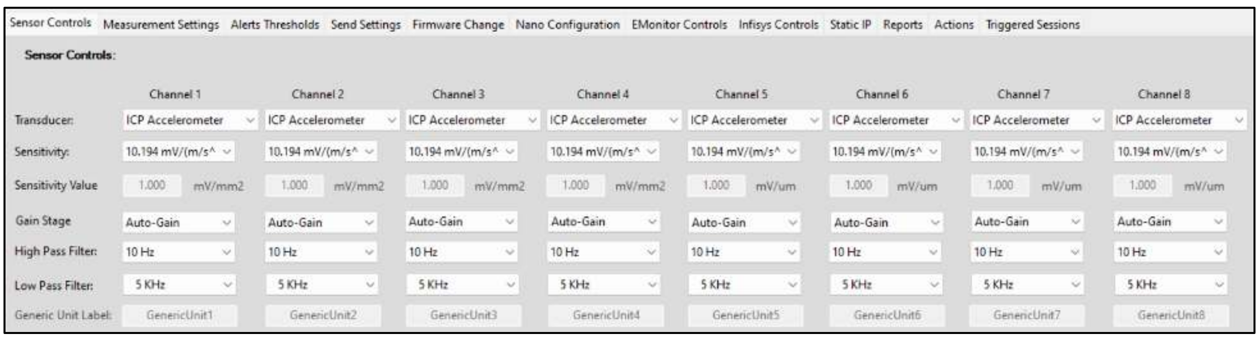
Destinations

- Determine where the Channels will be sending their measurements to; Machine Dossier (MD), EMonitor (Emon), and/or infiSYS Lite.

6.5 THE UCT RIBBON

This section provides detail for the tabs located underneath the Global Settings. “Ribbon” is the name chosen to describe these tabs. Each tab has its own section below.

6.5.1 Sensor Controls



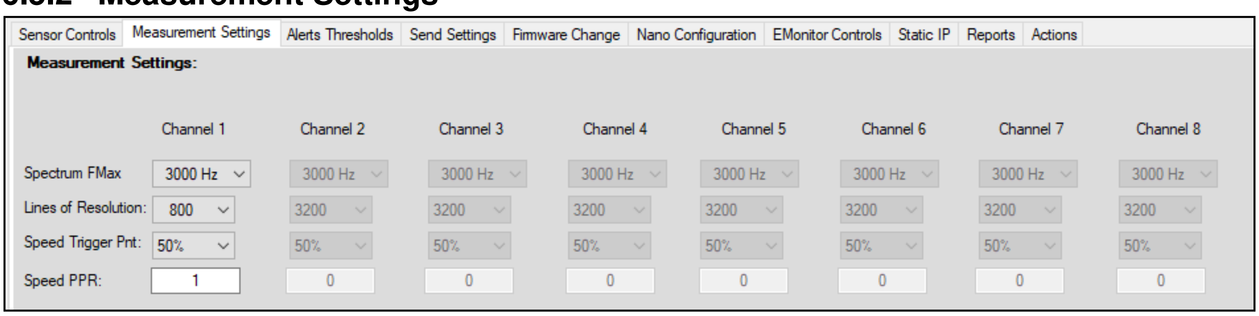
This tab allows the user to make specific changes to sensors.

- **Transducer**: Choose between the following probes.
    - ICP Accelerometer: a sensor that generates an electrical output proportional to applied acceleration. ICP® accelerometers are designed to measure vibration and shock for a wide variety of applications. They are simple to use and accurate over a wide frequency range which makes them the recommended choice for many testing situations.
    - Proximity Probe: convert information on the movement or presence of an object into an electrical signal.
    - Strain Gauge: a sensor whose resistance varies with applied force; It converts force, pressure, tension, weight, etc., into a change in electrical resistance.
    - VAC: A simple AC voltage sensor.
    - VDC: A simple DC voltage sensor.
    - Temperature: Provides temperature data in Celsius or Fahrenheit.
- **Sensitivity**: This drop down can only be configured for the ICP Accelerometer probe. The user may choose between the following the settings.
    - 10 mV/g
    - 50 mV/g

- 100 mV/g
- 500 mV/g

- **Sensitivity Value**: This box can only be filled for the following Transducer Types. The units vary per type of Transducer.
    - Proximity Probe
    - Strain Gauge
    - Temperature
    - VAC
- **Gain Stage**: Allows the user to switch between Auto-Gain and Manual Gain options. If manual is chosen, the drop-down menu allows the following choices:
    - 1x Gain
    - 5x Gain
    - 11x Gain
    - 25x Gain
    - 1/4x Gain
- **High Pass Filter**: Electronic filter that passes signals with a frequency higher than a certain cut off frequency. The following are the supported cut off frequency options for the high pass filter.
    - 2 Hz
    - 5 Hz
    - 10 Hz
    - 25 Hz
    - 1000 Hz
    - OFF
- **Low Pass Filter**: Electronic filter that passes signals with a frequency lower than a certain cut off frequency. The following are the supported cut off frequency options for the low pass filter.
    - 5000 Hz
    - OFF

6.5.2 Measurement Settings



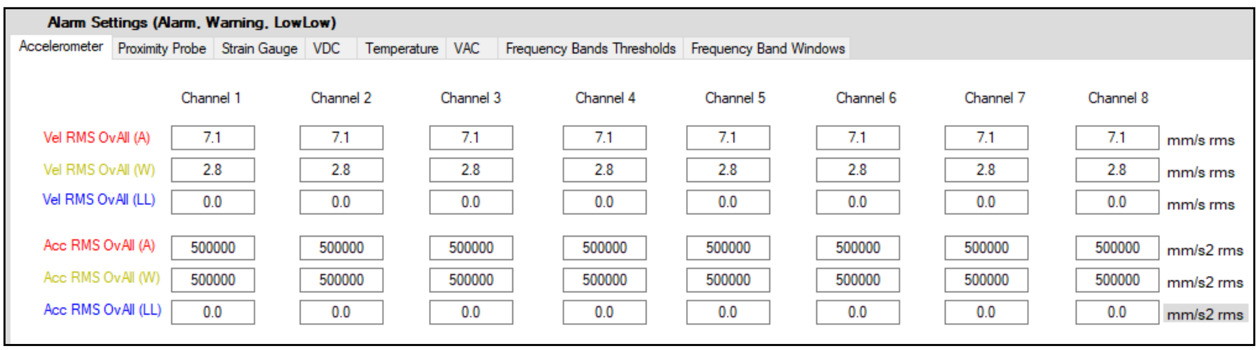
This tab allows the user to make configurations related to the presentation of the recorded measurements.

- **Spectrum Fmax**: Sets the limit to the frequency spectrum.
    - 1500 Hz
    - 3000 Hz
    - 5000 Hz

- 10,000 Hz
- 18,300 Hz

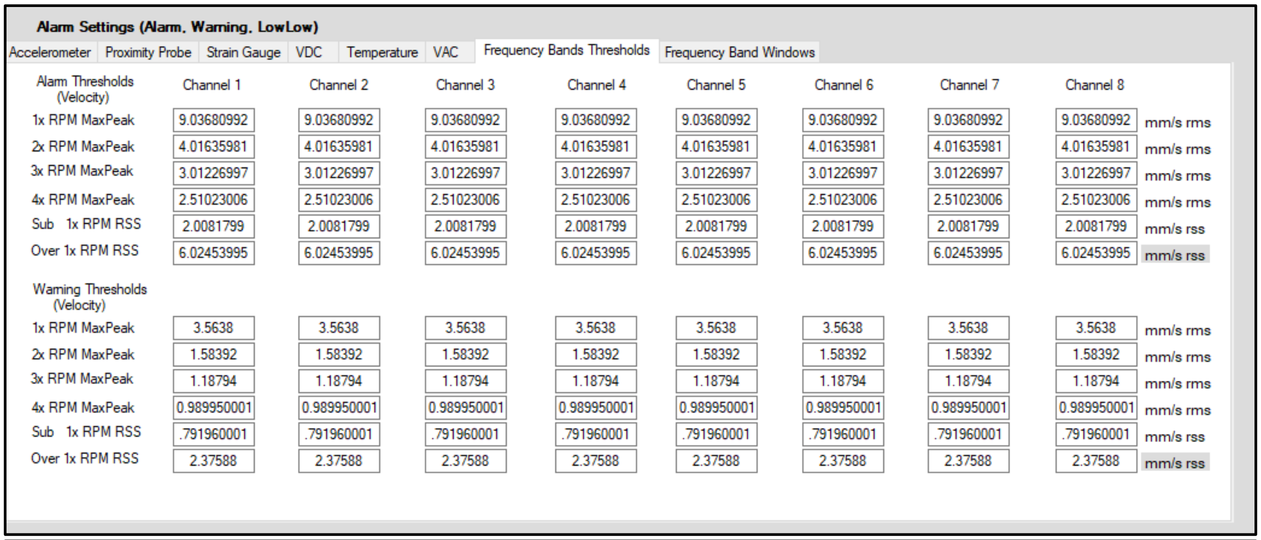
- **Lines of Resolution:** Sets the resolution of the spectrum.
    - 800 LOR
    - 1600 LOR
    - 3200 LOR
- **Speed Trigger Point:** Sets the trigger point for a speed measurement.
    - 10% increments from 10% to 90%.
- **Speed Pulses-Per-Revolution (PPR)**
    - 0 by default unless a Speed Sensor is selected. Only measured when PPR is greater than 0.
    - Must be a positive integer.

6.5.3 Alerts Thresholds



This tab is used for manually setting Alarm, Warning, and LowLow thresholds. The first six subtabs (Accelerometer, Proximity Probe, Strain Gauge, VDC, Temperature and VAC) are different for different transducer types. If no transducers of the selected tab are present, all information will be greyed out. Use the Sensor Controls tab to specify which transducer types are connected to each channel. The LowLow thresholds are for specifying a value below which the X8/X8II concludes that the current channel machine is OFF and will not send a measurement.

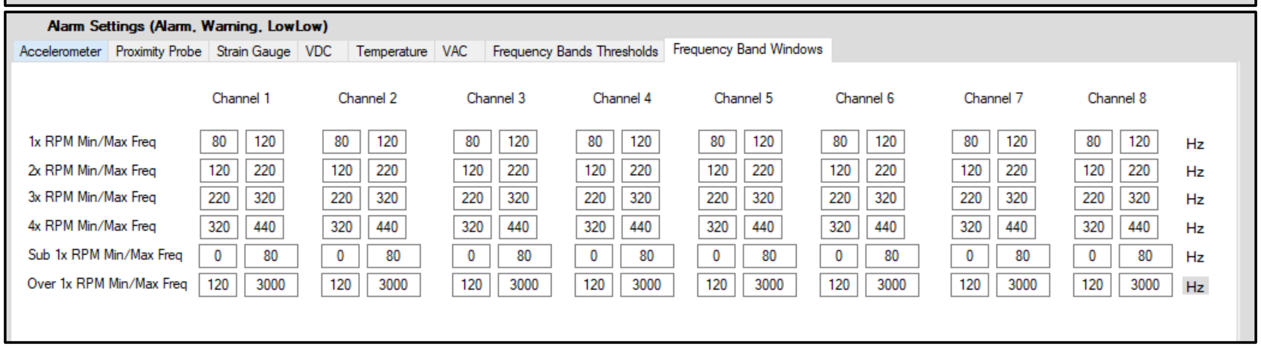
Frequency Bands Thresholds and Frequency Band Windows
These tabs allow the user to specify Frequency Band Alarm and Warning Thresholds and Band Window definitions for the 6 Frequency Bands supported by the X8II.



Alarm Settings (Alarm, Warning, LowLow)

Accelerometer | Proximity Probe | Strain Gauge | VDC | Temperature | VAC | Frequency Bands Thresholds | Frequency Band Windows

Alarm Thresholds (Velocity)	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7	Channel 8	
1x RPM MaxPeak	9.03680992	9.03680992	9.03680992	9.03680992	9.03680992	9.03680992	9.03680992	9.03680992	mm/s rms
2x RPM MaxPeak	4.01635981	4.01635981	4.01635981	4.01635981	4.01635981	4.01635981	4.01635981	4.01635981	mm/s rms
3x RPM MaxPeak	3.01226997	3.01226997	3.01226997	3.01226997	3.01226997	3.01226997	3.01226997	3.01226997	mm/s rms
4x RPM MaxPeak	2.51023006	2.51023006	2.51023006	2.51023006	2.51023006	2.51023006	2.51023006	2.51023006	mm/s rms
Sub 1x RPM RSS	2.0081799	2.0081799	2.0081799	2.0081799	2.0081799	2.0081799	2.0081799	2.0081799	mm/s rss
Over 1x RPM RSS	6.02453995	6.02453995	6.02453995	6.02453995	6.02453995	6.02453995	6.02453995	6.02453995	mm/s rss
Warning Thresholds (Velocity)									
1x RPM MaxPeak	3.5638	3.5638	3.5638	3.5638	3.5638	3.5638	3.5638	3.5638	mm/s rms
2x RPM MaxPeak	1.58392	1.58392	1.58392	1.58392	1.58392	1.58392	1.58392	1.58392	mm/s rms
3x RPM MaxPeak	1.18794	1.18794	1.18794	1.18794	1.18794	1.18794	1.18794	1.18794	mm/s rms
4x RPM MaxPeak	0.989950001	0.989950001	0.989950001	0.989950001	0.989950001	0.989950001	0.989950001	0.989950001	mm/s rms
Sub 1x RPM RSS	.791960001	.791960001	.791960001	.791960001	.791960001	.791960001	.791960001	.791960001	mm/s rss
Over 1x RPM RSS	2.37588	2.37588	2.37588	2.37588	2.37588	2.37588	2.37588	2.37588	mm/s rss

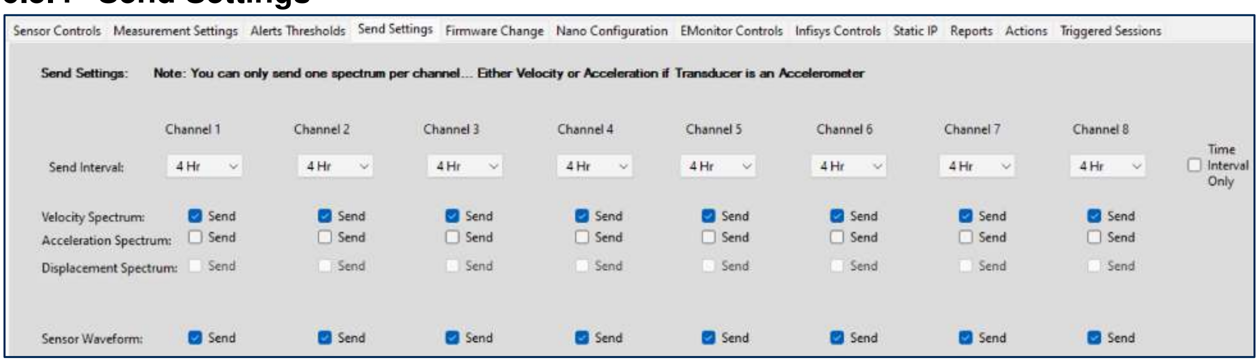


Alarm Settings (Alarm, Warning, LowLow)

Accelerometer | Proximity Probe | Strain Gauge | VDC | Temperature | VAC | Frequency Bands Thresholds | Frequency Band Windows

	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7	Channel 8	
1x RPM Min/Max Freq	80 / 120	80 / 120	80 / 120	80 / 120	80 / 120	80 / 120	80 / 120	80 / 120	Hz
2x RPM Min/Max Freq	120 / 220	120 / 220	120 / 220	120 / 220	120 / 220	120 / 220	120 / 220	120 / 220	Hz
3x RPM Min/Max Freq	220 / 320	220 / 320	220 / 320	220 / 320	220 / 320	220 / 320	220 / 320	220 / 320	Hz
4x RPM Min/Max Freq	320 / 440	320 / 440	320 / 440	320 / 440	320 / 440	320 / 440	320 / 440	320 / 440	Hz
Sub 1x RPM Min/Max Freq	0 / 80	0 / 80	0 / 80	0 / 80	0 / 80	0 / 80	0 / 80	0 / 80	Hz
Over 1x RPM Min/Max Freq	120 / 3000	120 / 3000	120 / 3000	120 / 3000	120 / 3000	120 / 3000	120 / 3000	120 / 3000	Hz

6.5.4 Send Settings



Sensor Controls | Measurement Settings | Alerts Thresholds | Send Settings | Firmware Change | Nano Configuration | EMonitor Controls | Infisys Controls | Static IP | Reports | Actions | Triggered Sessions

Send Settings: Note: You can only send one spectrum per channel... Either Velocity or Acceleration if Transducer is an Accelerometer

	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7	Channel 8	
Send Interval:	4 Hr	4 Hr	4 Hr	4 Hr	4 Hr	4 Hr	4 Hr	4 Hr	Time Interval Only
Velocity Spectrum:	Send	Send	Send	Send	Send	Send	Send	Send	
Acceleration Spectrum:	Send	Send	Send	Send	Send	Send	Send	Send	
Displacement Spectrum:	Send	Send	Send	Send	Send	Send	Send	Send	
Sensor Waveform:	Send	Send	Send	Send	Send	Send	Send	Send	

This tab is used for making channel specific send configurations.

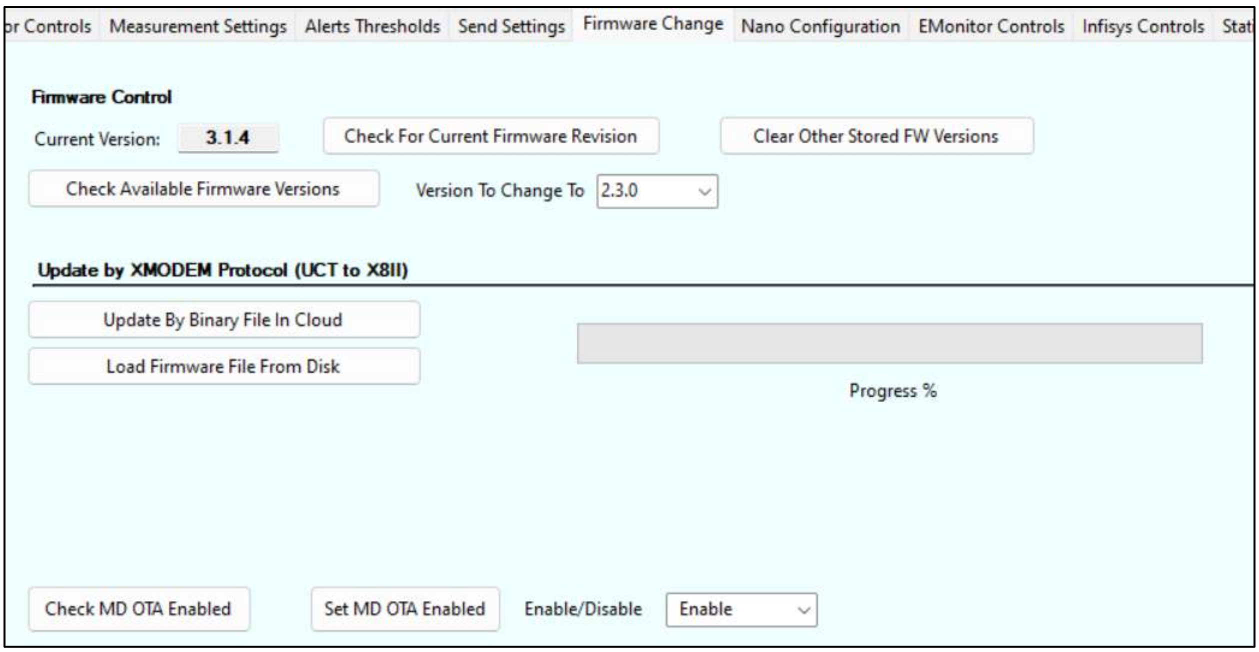
- **Send Interval**: The following intervals are supported.
  - 1 Hr
  - 2 Hr
  - 3 Hr
  - 4 Hr
  - 6 Hr
  - 12 Hr
  - 1 Day
  - 1 Week

- 1 Month

- **Velocity Spectrum**: Sends the entire spectrum associated with the velocity measurement.
- **Acceleration Spectrum**: Sends the entire spectrum associated with the acceleration measurement.
- **Displacement Spectrum**: Sends the entire spectrum associated with the displacement measurement.
- **Sensor Waveform**: Sends the waveform for the associated sensor.
- **Time Interval Only**: This checkbox will prevent the alarm behavior of the X8II and will strictly adhere to the specified interval.

6.5.5 Firmware Change

This tab focuses on manipulating the firmware version of the X8II Hub.



<u>Firmware Control</u>

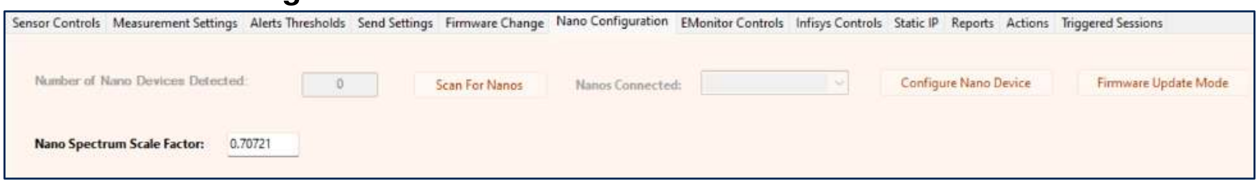
- **Current Version**: Displays the current firmware version of the connected unit.
- **Check For Current Firmware Version**: Retrieves the current firmware version of the connected unit.
- **Clear Other Stored FW Versions**: Deletes all the stored FW version on the unit.
- **Check Available Firmware Versions**: Searches for all available FW versions on the cloud. Version To Change To drop-down menu shows all the available FW versions after the check is complete.

<u>Update By XMODEM Protocol (UCT to X8II)</u>

- **Update By Binary File In Cloud:** After a FW version is selected from the Version To Change To drop-down menu, this button will update the X8II via a download from the cloud.
- **Load Firmware File From Disk**: Loads the selected firmware file to the connected unit. Note: the format of the file must be, “FFXXXXXX.fw” where “XXXXXX” is the version number. For instance, to load firmware version 2.2.3, the user must load the file named “FF020203.fw.”

- **Check MD OTA Enabled**: Checks to see if an OTA upgrade/downgrade by MD is enabled or disabled by the unit.
- **Set MD OTA Enabled**: Sets the OTA firmware change status by MD to the desired choice.
- **Enable/Disable**: Drop-down list allowing the user to either enable or disable the OTA firmware version change by MD.

6.5.6 NANO Configuration



- **Number of Nano Devices Detected**: This field will show how many NANOs that the X8II can see after clicking the “Scan for NANOs” button.
- **Scan for Nanos**: This button will scan for all available NANOs. It will update the **Number of Nanos Detected** box and will show all NANOs in range of the X8II under the **Nanos Connected** drop-down.
- **NANO’s Connected**: This drop-down displays the serial numbers of all NANO units within communication range of the X8II.
- **Configure NANO Device**: Once a NANO is selected from the **Nanos Connected** drop-down, it can be configured via this button which will bring up a new configuration window.
- **Firmware Update Mode**: This button will place the specified NANO from the drop-down list to activate bootloader mode. This is unlikely to be used upon release of the Zark NI Application. NANOs can now be updated via the Zark NI App. Please download the app from the Android or iOS store on your smartphone.
- **Nano Spectrum Scale Factor:** This scale factor allows the user to scale the spectrum values if there is a desire to scale them. It should normally be left at .70721

6.5.6.1 Configure NANO Device

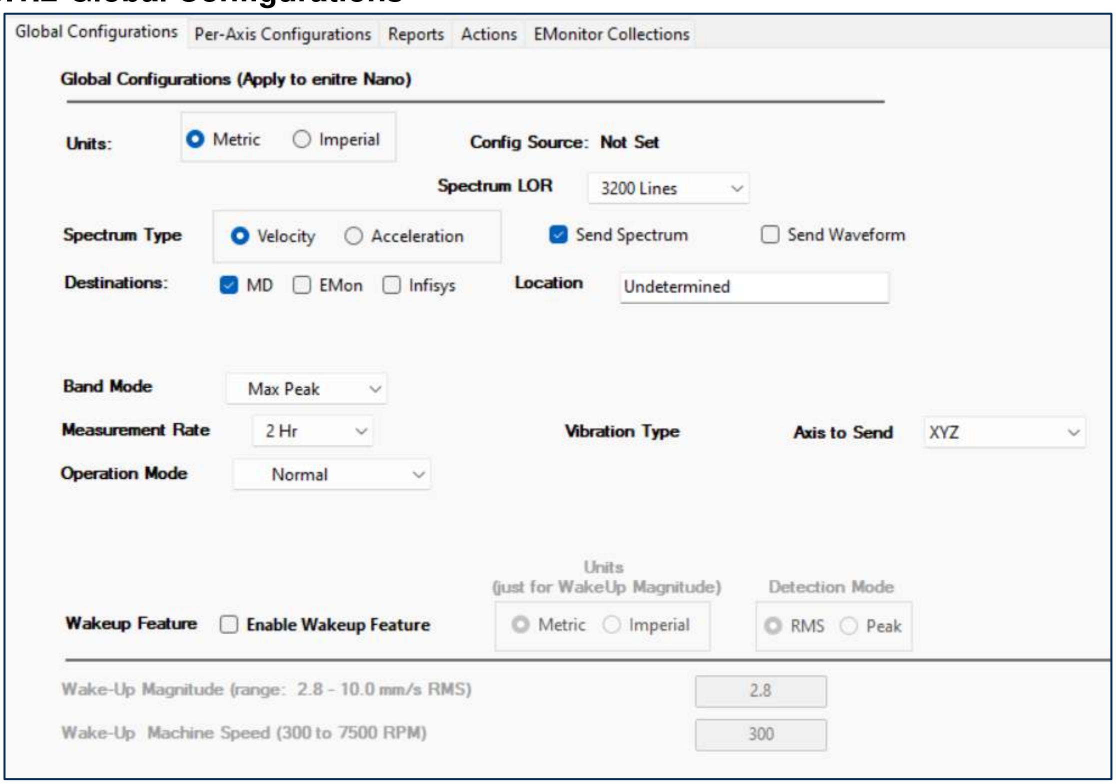
The NANO configuration window has five main sections: *General Use*, *Global Configuration*, *Per-Axis Configurations, Reports* and *Actions* sections.

6.5.6.1.1 General Use

This section reports the serial number of the unit and has three main buttons.

- **Import CSV**: This button is used to import configurations from a CSV file to the Zark Nano device.
- **Export CSV**: This button is used to export configurations to a CSV file from a Zark Nano device.
- **Send Config to NANO**: This button will send the configurations made by the user to the NANO. A confirmation dialogue box will pop up when the send is successful.
- **Retrieve from NANO**: This button will retrieve the configurations from the NANO unit and populate all necessary field.
- **Set to Defaults**: This button will set all configurable fields to default values.

6.5.6.1.2 Global Configurations



- **Units**: Metric vs. Imperial. This will influence the units on how measurements are reported.
- **Config Source**: This will change with configurations either set by the UCT or MD.

- **Spectrum LOR**: Resolution of Spectra of measurements.
  - 800 LOR
  - 1600 LOR
  - 3200 LOR
- **Spectrum Type**: Velocity or Acceleration.
- **Send Spectrum**: Checkbox that will send the spectrum of the measurement.
- **Send Waveform**: Checkbox that will send the waveform of the measurement.
- **Destinations**: the unit can send to Machine Dossier (MD), EMonitor (EMon), and/or infiSYS Lite.
- **Band Mode**: Energy band thresholds of NANOs.
  - Max Peak
  - Root Sum of Squares (RSS)
  - MD Mix: Configurations received by MD. (Called such because it is the non-configurable mix that MD allows)

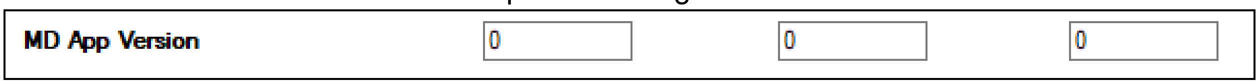
  There are 6 different frequency bands, all of them are velocity thresholds:
  - BandA: 1x RPM
  - BandB: 2x RPM
  - BandC: 3x RPM
  - BandD: 4x RPM
  - BandE: Sub 1x RPM
  - BandF: Over 1x RPM

  Band Mode choices are applied to all 6 for “Max Peak” and “Root Sum of Squares (RSS).”
  If “MD Mix” is specified, the unit measurement type for the 1st four frequency bands (1x, 2x, 3x, 4x RPM) will be Max Peak, but the bottom 2 (Sub 1x RPM and Over 1x RPM) will be RSS.
- **Measurement Rate**: Controls how often the NANO sends a measurement to the specified location.
  - 1 Hour
  - 2 Hour
  - 4 Hour
  - 6 Hour
  - 12 Hour
  - 24 Hour
- **Operation Mode:** Swap between Normal and Triggered Sessions. Triggered Sessions mode allows measurement intervals to be triggered at a defined custom rate.
- **Axis to Send**: Controls which Axis will be sent to the specified location.
  - X, Y, Z and XYZ
- **Wake-Up Feature**: Checkbox that allows the use of the Wake-Up feature if enabled.
- **Units**: Choose between Metric or Imperial units.
- **Detection Mode**: RMS or Peak options are available currently.

- **Wake-Up Axis Magnitude**: Magnitude of wake up required for the feature to trigger. Consult section 6.8 of the NANO User Manual for additional information.
- **Wake-Up Machine Speed**: Machine speed required to for feature to trigger. Consult section 6.8 of the NANO User Manual for additional information.

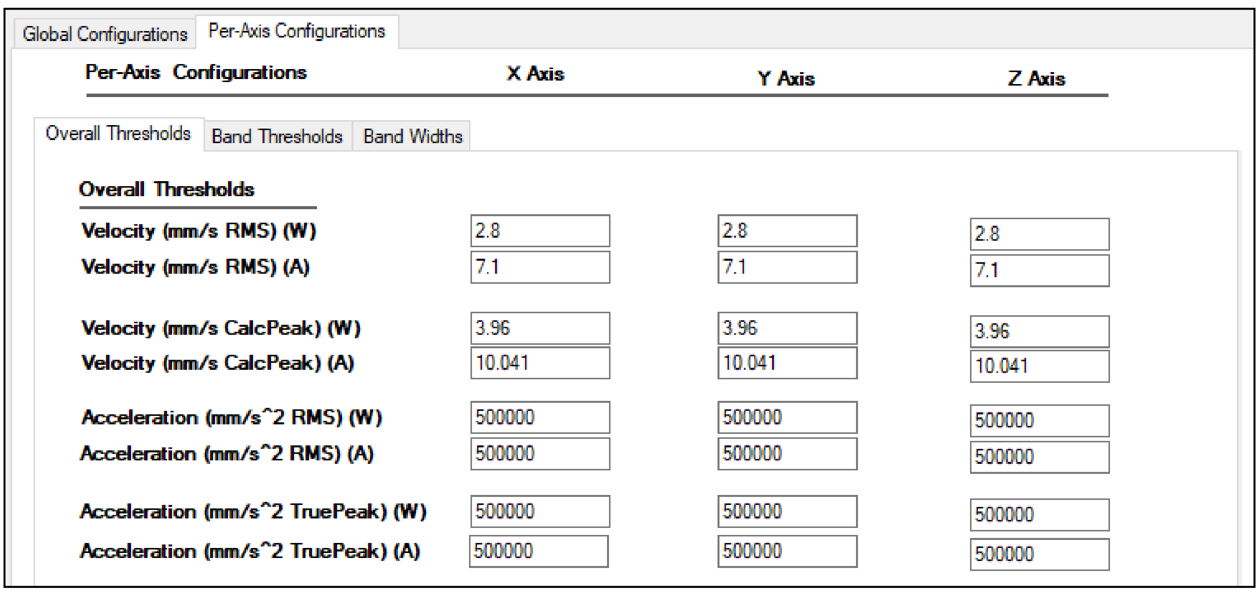
6.5.6.1.3 Per-Axis Configurations

This tab has three main views: *Overall Thresholds*, *Band Thresholds*, and *Band Widths*. These tabs are used to make axis specific configurations to the NANO.



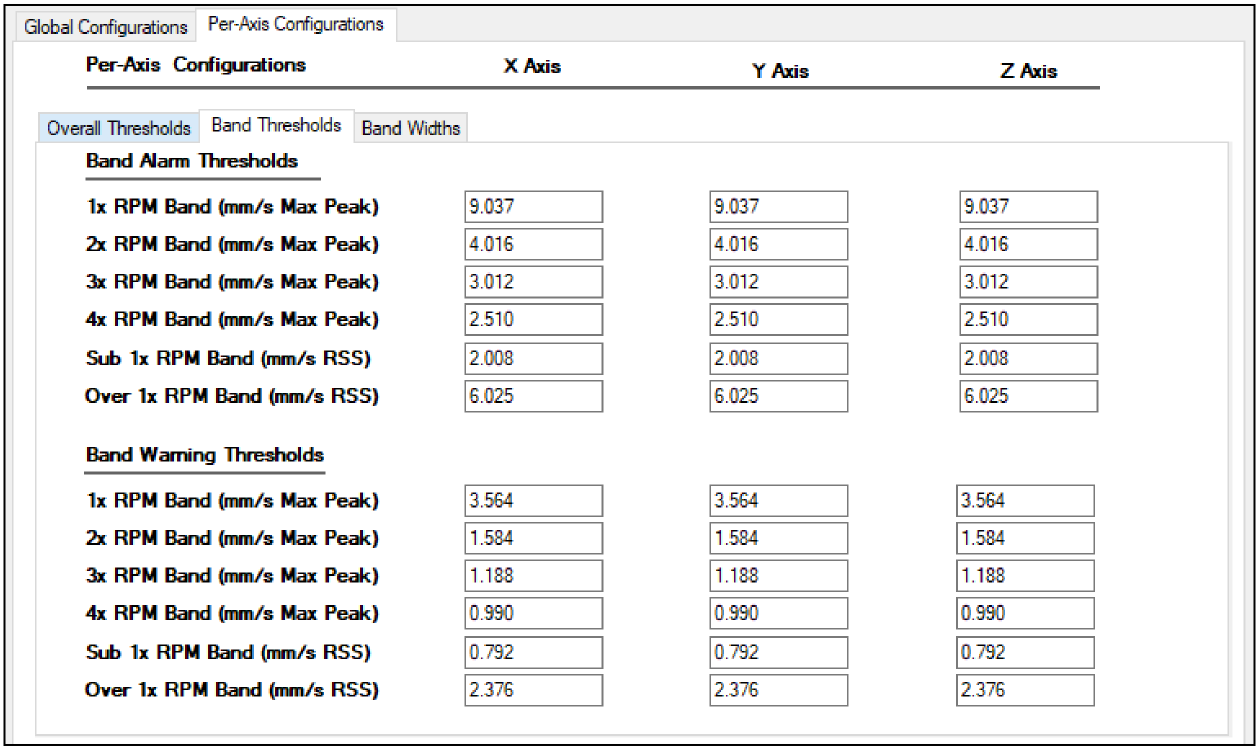
The MD App version is reported below the tab for each axis, but this will only be visible if MD has been selected as one of the destinations in the NANO Configuration form. If EMonitor has been selected, this will not display.

Overall Thresholds



This tab defines warming and alarm values for velocity and acceleration measurements for all three axes.

Band Thresholds



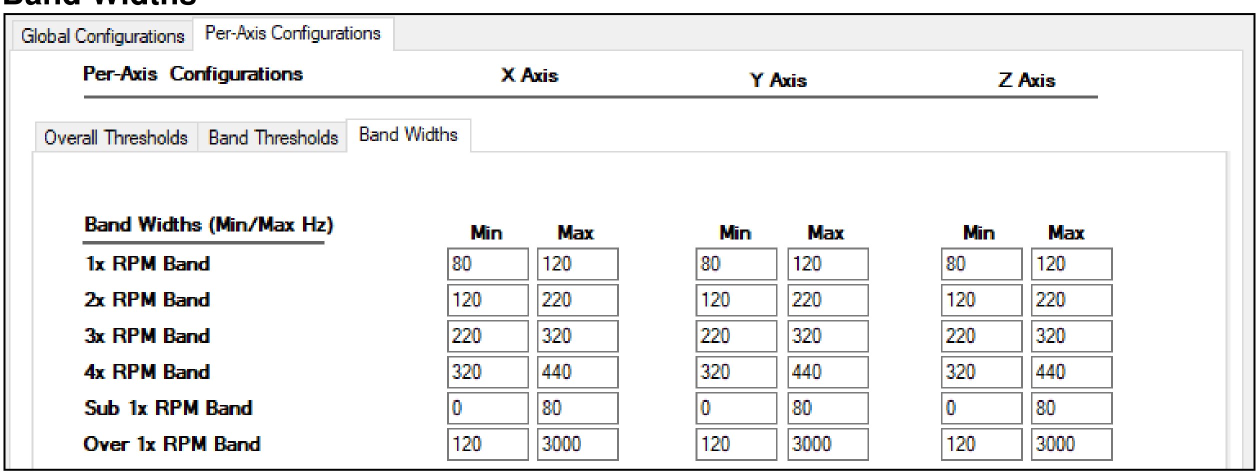
Global Configurations | Per-Axis Configurations

Overall Thresholds | Band Thresholds | Band Widths

Per-Axis Configurations	X Axis	Y Axis	Z Axis
Band Alarm Thresholds			
1x RPM Band (mm/s Max Peak)	9.037	9.037	9.037
2x RPM Band (mm/s Max Peak)	4.016	4.016	4.016
3x RPM Band (mm/s Max Peak)	3.012	3.012	3.012
4x RPM Band (mm/s Max Peak)	2.510	2.510	2.510
Sub 1x RPM Band (mm/s RSS)	2.008	2.008	2.008
Over 1x RPM Band (mm/s RSS)	6.025	6.025	6.025
Band Warning Thresholds			
1x RPM Band (mm/s Max Peak)	3.564	3.564	3.564
2x RPM Band (mm/s Max Peak)	1.584	1.584	1.584
3x RPM Band (mm/s Max Peak)	1.188	1.188	1.188
4x RPM Band (mm/s Max Peak)	0.990	0.990	0.990
Sub 1x RPM Band (mm/s RSS)	0.792	0.792	0.792
Over 1x RPM Band (mm/s RSS)	2.376	2.376	2.376

This tab allows the user configure Band Alarm Thresholds and Band Warning Thresholds for all axes.

Band Widths



Global Configurations | Per-Axis Configurations

Overall Thresholds | Band Thresholds | Band Widths

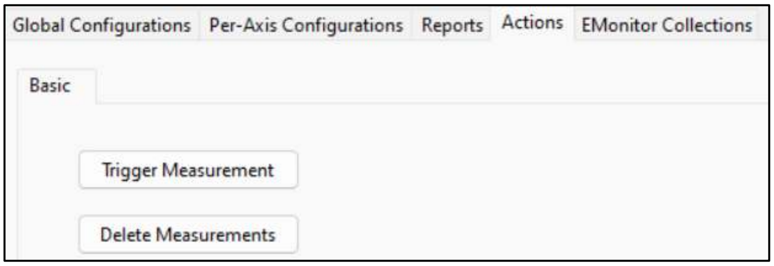
Per-Axis Configurations	X Axis		Y Axis		Z Axis	
Band Widths (Min/Max Hz)	Min	Max	Min	Max	Min	Max
1x RPM Band	80	120	80	120	80	120
2x RPM Band	120	220	120	220	120	220
3x RPM Band	220	320	220	320	220	320
4x RPM Band	320	440	320	440	320	440
Sub 1x RPM Band	0	80	0	80	0	80
Over 1x RPM Band	120	3000	120	3000	120	3000

This tab can be used to specify Min and Max values for the allotted RPM Bands.

6.5.6.1.4 Reports

Get Nano Firmware Version: Retrieves current FW version of selected NANO.

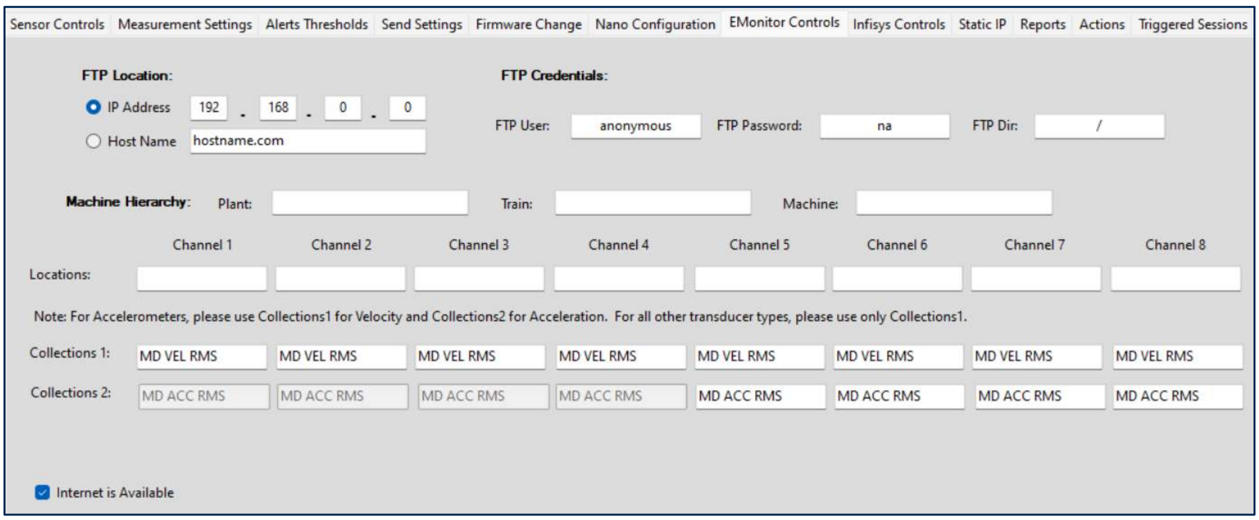
6.5.6.1.5 Actions



<u>Basic</u>

- **Trigger Measurement:** Triggers a measurement from the NANO unit.
- **Delete Measurement:** Deletes all saved measurements on the NANO unit.

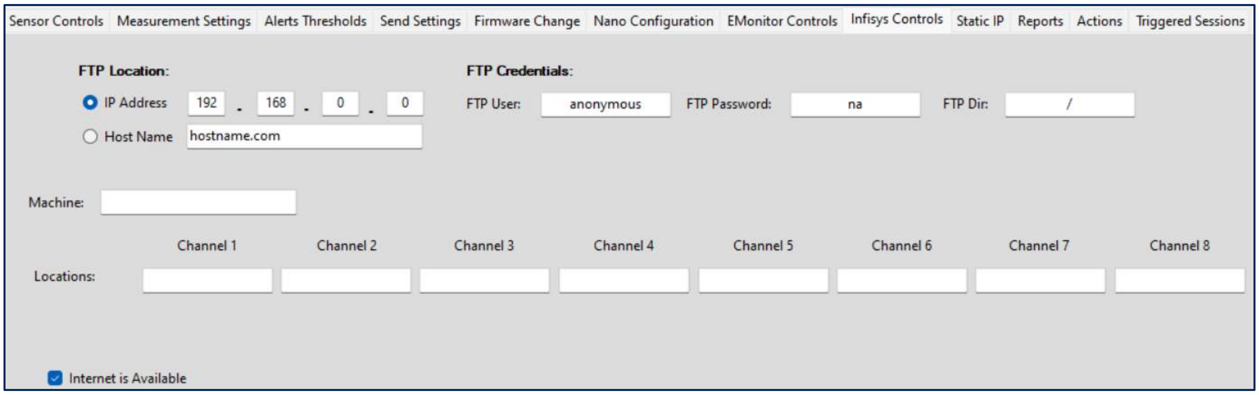
6.5.7 EMonitor Controls



- **FTP Location:**
  - IP Address: Specific EMonitor Host FTP site IP.
  - Host Name: Name of host server.
- **FTP Credentials**: Used specifically for EMonitor
  - FTP User: *anonymous* by default.
  - FTP Password: *N/A* when the username is anonymous.
  - FTP Dir: Default is /. It is expected that the FTP server FTPRoot directory has been setup to be a specific folder expected by the EMonitor Scheduler.
- **Machine Hierarchy**: The specific location for the interface.
  - Plant: Name of overall organization
  - Train: Name of sub-organization
  - Machine: Name of X8II (serial number is most populated)
- **Locations**: This is where the name of each wired channel can be specified.
- **Collections**: Ensure that specified collection in EMonitor matches the UCT field.
  1. Overall velocity collections.
  2. Overall acceleration collections.

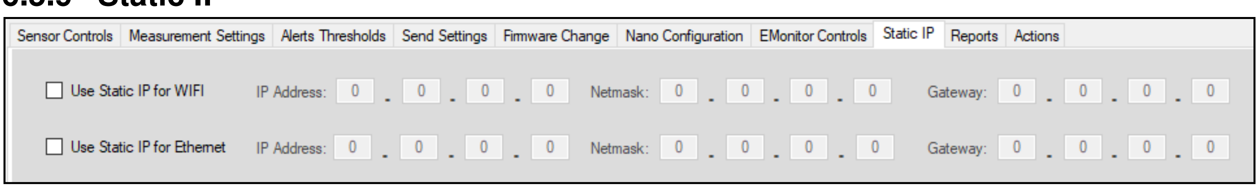
- **Time Zone**: Specify which time zone the measurement time stamps will be collected.
- **Internet is Available:** Checkbox letting the unit know that it can connect to the internet.

6.5.8 infiSYS Controls



- **FTP Location:**
  - IP Address: Specific Emonitor device.
  - Host Name: Name of host server.
- **FTP Credentials**: Used specifically for EMonitor and infiSYS destinations.
  - FTP User: *anonymous* by default.
  - FTP Password: *N/A* when the username is anonymous.
  - FTP Dir: Default is */*. It is expected that the FTP server FTPRoot directory has been setup to be a specific folder.
- **Machine**: Name of X8II (this field is commonly populated with the unit's serial number)
- **Locations**: This is where the name of each wired channel can eb specified.
- **Internet is Available:** Checkbox letting the unit know that it can connect to the internet.

6.5.9 Static IP



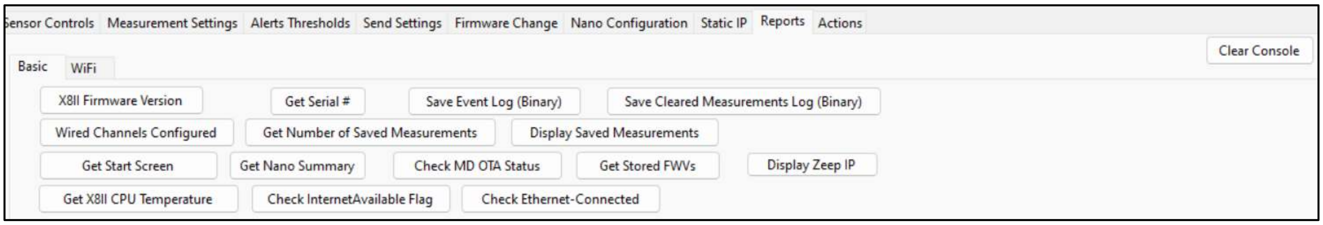
This tab is used to configure static network credentials for the X8II.

- **Use Static IP for WI-FI**: configure static credentials for WI-FI.
- **Use Static IP for Ethernet**: configure static credentials for Ethernet.
  - IP Address: assigns a static IP address to the X8II.
  - Netmask: used to divide the IP address into subnets.
  - Gateway: transforms communication.

6.5.10 Reports

The reports tab is used for debugging and troubleshooting the X8II.

6.5.10.1 Basic

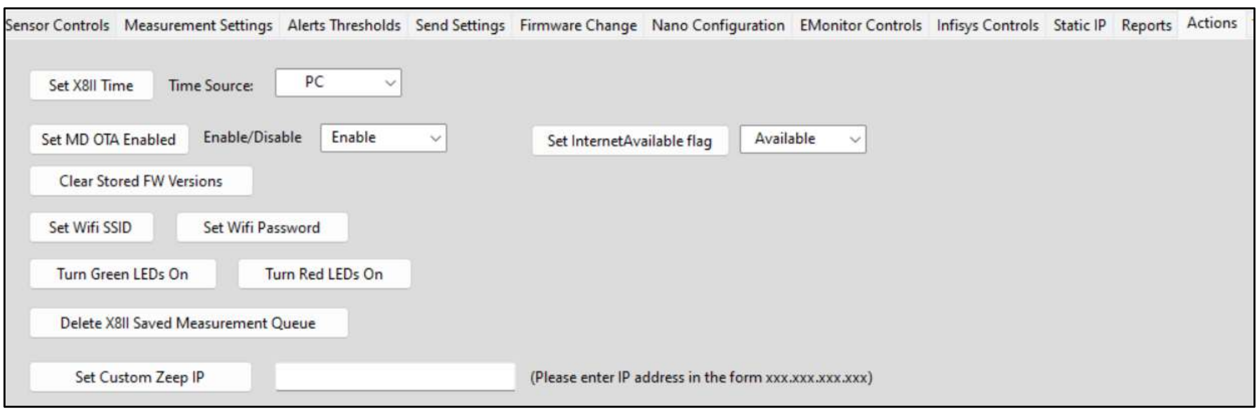


- **Clear Console**: clears the console prompt on the reports tab.
- **X8II Firmware Version**: retrieves the firmware version.
- **Get Serial #**: retrieves the serial number.
- **Save Event Log (Binary)**: retrieves the event log.
- **Save Cleared Measurement Log (Binary)**: saves the history of past measurements.
- **Wired Channels Configured**: retrieves the number of wired channels that are active.
- **Get Number of Saved Measurements**: retrieves the number of saved measurements.
- **Display Saved Measurements**: retrieves the saved measurements and displays them on the console.
- **Get Start Screen**: retrieves the start screen containing general information.
- **Get Nano Summary**: retrieves the number of NANOs with basic summaries.
- **Check MD OTA Status**: retrieves the status of the MD OTA firmware change feature.
- **Get Stored FWVs**: retrieves the list of previously installed firmware versions on the unit.
- **Display Zeep IP:** Used to check the Zeep IP address.
- **Get X8II CPU Temperature**: retrieves the CPU's temperature.
- **Check InternetAvailable Flag:** Will check whether the X8II is set to connect to the internet. This can be toggled off and on at the bottom of the Emonitor Controls tab (6.4.7) or the Infysis Controls (6.4.8) tabs.
- **Check Ethernet-Connected:** Checks to see if the X8II is connected to Ethernet. An active Ethernet cable must be connected to the X8II for this to return a positive result.

6.5.10.2 WI-FI

- **Get Wi-Fi SSID**: retrieves the Wi-Fi network name.
- **Get Wi-Fi Password**: retrieves the Wi-Fi password.
- **Get Wi-Fi MAC Address**: retrieves the network IP address.
- **IPCONFIG:** Reports WiFi, Ethernet and BLE IP information.
- **Test WI-FI**: Tests the network connection with the specified network information. Network information fields must first be populated. Returns a red “X” if the connection fails.

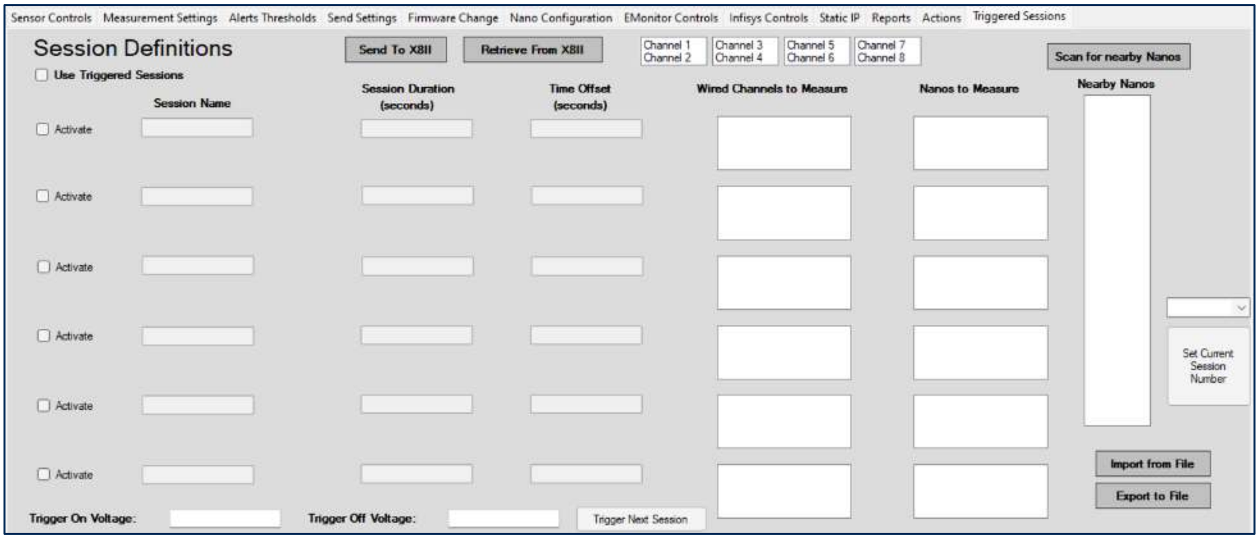
6.5.11 Actions



- **Set X8II Time**: Sends the time to the X8II.
- **Time Source**: drop down of where the time will be obtained.
    - PC: The X8II receives the clock on the PC.
    - Epoch Time: The X8II receives the Epoch Time.
- **Set MD OTA Status**: Sets the OTA firmware change status by MD to the desired choice selected by the **Enable**/**Disable** drop-down list.
- **Set InternetAvailable Flag:** Whichever option is selected from the drop-down menu: Available or Unavailable, this button will send the change up immediately to the X8II without the need to press the “Send to X8II” button.
- **Clear Stored FW Versions**: Clears the library of all previously installe firmware versions.
- **Set Wi-Fi SSID**: Sets the primary network name.
- **Set Wi-Fi Password**: Sets the primary password.
- **Turn Green LEDs On:** Turns all LEDs on the unit green, for testing purposes.
- **Turn Red LEDs On:** Turns all LEDs on the unit red, for testing purposes.

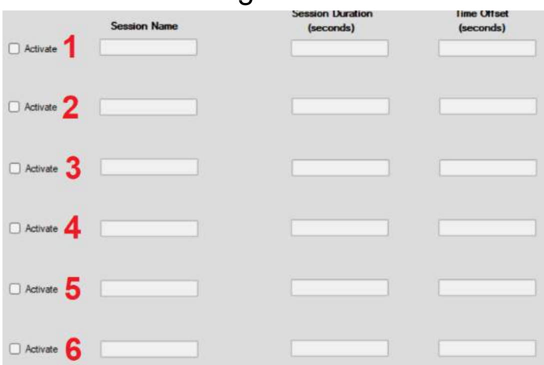
- **Delete X8II Saved Measurement Queue:** Clears X8II saved measurement queue. Note, this means that all saved measurements waiting to be downloaded by a destination will be deleted.
- **Set Custom Zeep IP:** Only used if a custom Zeep IP address is required, otherwise the unit automatically configures the Zeep IP.

6.5.12 Triggered Sessions



- **Use Triggered Sessions:** Checkbox to activate/deactivate triggered sessions.
- **Send to X8II:** Sends triggered sessions configuration changes to the X8II. Please note this button must be clicked to send triggered session changes, the "Send to X8II" button in the Global settings will not send triggered sessions changes. If this button is not clicked after changes are made, the changes will not be saved.
- **Retrieve from X8II:** Retrieves current triggered sessions configurations. This button should be clicked before any changes are made to the triggered sessions tab.
- **Scan for nearby Nanos:** Scans for NANOs within communication range and lists all NANOs within range in the "Nearby Nanos" area.
- **Activate:** Toggle switch to activate/deactivate a session.
- **Session Name:** User specified session name, can be any combination of letters and numbers.
- **Session Duration (seconds):** Time duration (in seconds) of session once triggered. This duration is the timeframe which measurements will be recorded.
- **Time Offset (seconds):** Duration of time the unit waits before the triggered session measurements begin (called "Session Duration").

- **Wired Channels to Measure:** Wired channels selected to measure during the triggered session. The channels (1-8) are located above the “Wired Channels to Measure” boxes and must be dragged and dropped into the desired session.
- **Nanos to Measure:** NANOs that will be measured during the triggered session. NANOs to be measured must be dragged and dropped from the “Nearby Nanos” area once a “Scan for nearby Nanos” is performed.
- **Nearby Nanos:** Once the “Scan for nearby Nanos” button is clicked all units within range will be listed in this area.
- **Set Current Session Number:** The drop-down box above this button is used to identify each active session. How these numbers relate to each session is listed in the image below:



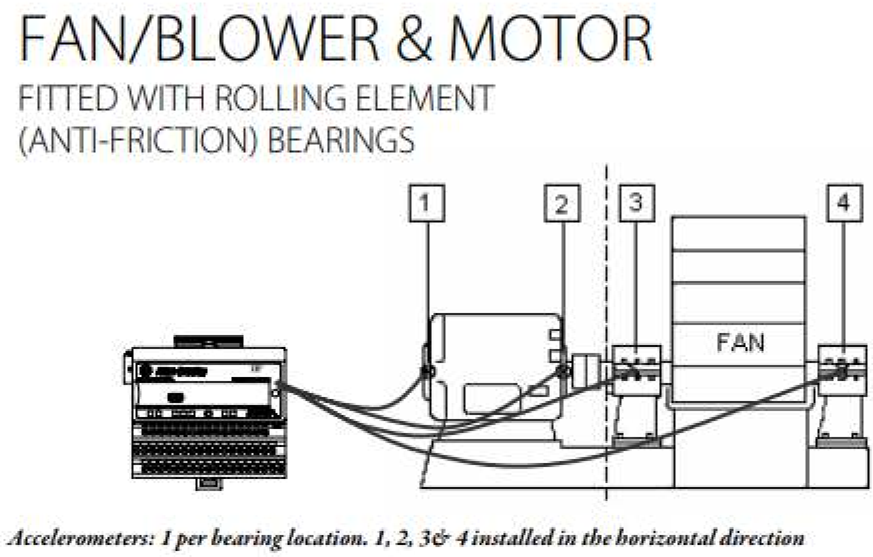
  Selecting a number from the drop-down menu and setting it by using the “Set Current Session Number” button ensures the session number selected will be the next session triggered.
- **Trigger On Voltage:** Voltage threshold for when a triggered session will begin.
- **Trigger Off Voltage:** Voltage threshold for when a triggered session will end.
- **Trigger Next Session:** Once this button is pushed, an active session will be triggered. If a number has been selected via the “Set Current Session Number” button, this will be the next session triggered. If the “Set Current Session Number” is blank, the next active session triggered will follow whatever session was triggered previously. For instance, if the last active triggered session was #1 using the image above, the next triggered session will be #2. If no session has been triggered previously, the unit will trigger the lowest active session number. If only one session is active, it will continue to trigger the same session. Once selected, the X8II will close the port and disconnect from the UCT to trigger the next session. A dialogue box will appear informing the user of this functionality.
- **Import from File:** Import a .CSV file of previously exported Session Definitions.
- **Export to File:** Export a .CSV file of current Session Definitions for later import. This allows the user to save settings for the future.

6.5.13 Applications

This tab contains common application configurations. This consists of Sub-tabs, one for each currently offered application. Current applications include:

- Motor/Blower (Rolling Element Bearing)
- Motor/Blower (Sleeve Bearing)
- Centrifuge w/ GearBox (Rolling Element Bearing)
- Vertical Motor & Pump

An example of one such application is the Fan Blower & Motor (Motor/Blower), which is illustrated in the image below.



In this example, the “Drive” (motor) will be monitored by accelerometers mounted in locations 1 and 2, while the “Driven” (fan) will be monitored by accelerometers in locations 3 and 4.

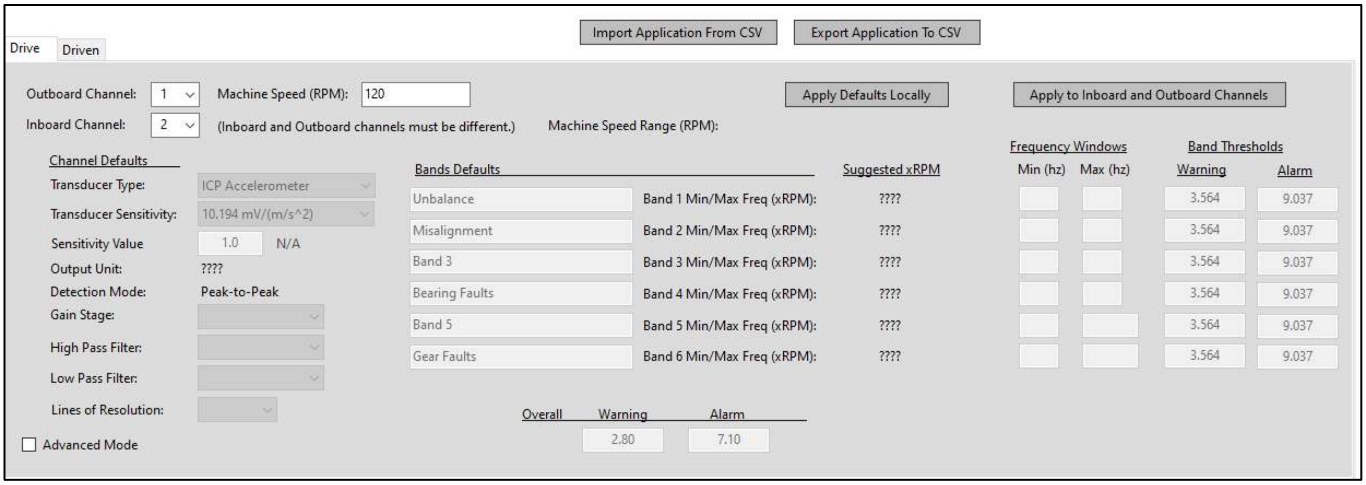
This is just one example, but the general idea of monitoring each bearing location on the “drive” and “driven” piece of equipment is the principle behind all configurations found within the application tab.

6.5.13.1 Common Application Fields

Each application is divided into two sub-tabs, Drive and Driven. These tabs are designed so that the information entered by the user in each tab (Drive and Driven) populates other areas within the UCT.

6.5.13.1.1 Drive

This tab is used to define the sensors responsible for monitoring the piece of machinery that is driving the assembly (motor).



Import Application From CSV: Imports a CSV file that has previously been exported. Once imported, the values from a previously exported CSV will auto populate all fields within the form.

Export Application To CSV: Creates a CSV document saving all entries found in current form. Use the Import Application From CSV button to import these saved settings.

Outboard and Inboard Channel: Drop-down menus that are used to select the channel of each sensor. These channels correlate to channels 1-8 of the UCT, which represent the 8 wired channels connected to the field termination board.

Machine Speed (RPM): Speed of the drive machine in RPMs.

Machine Speed Range (RPM): Once a machine speed is entered and "Apply Defaults Locally" is clicked, the program will choose a range depending on the machine speed entry. The range that is chosen will depend on the type of application that is being used. More information will be given in the application specific section.

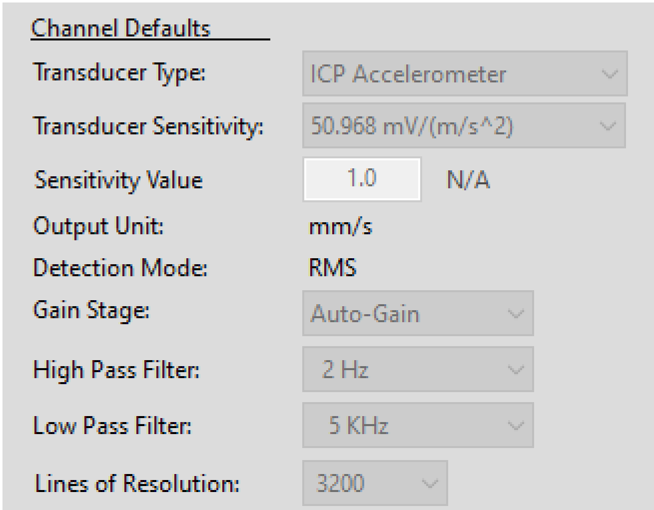
The range will dictate some default values and a change in range will cause these values to change.

Apply Defaults Locally: Once the Outboard and Inboard Channels and Machine Speed are selected, press this button to get the Drive form to update. As mentioned above, some values in the form may change depending on the Machine Speed range.

Apply to Inboard and Outboard Channels: Once this button is clicked, the values found within the form will be updated in other tabs that contain data that correlates. This should be the last step after completing the entry form.

Advanced Mode: Toggling this unlocks all fields found within the tab.

Channel Defaults



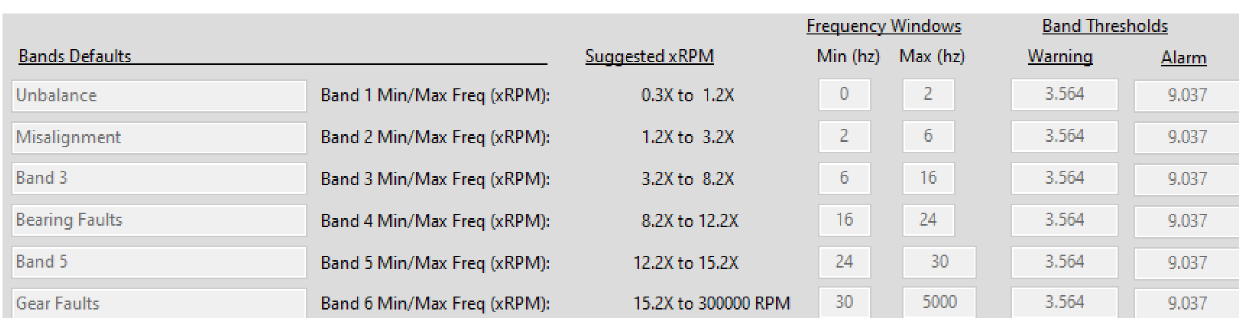
The entry fields in this section are used to populate fields in the Sensor Controls and Measurement Settings tabs. Once the desired settings have been entered and the "Apply to Inboard and Outboard Channels" button is pressed, the fields that correlate in those tabs will be updated.

Refer to section 6.5.1 (Sensor Controls) for more information on the following fields found within this section:

- Transducer Type
- Transducer Sensitivity
- Sensitivity Value
- AC Gain Stage
- High Pass Filter
- Low Pass Filter

More information on Lines of Resolution can be found in section 6.5.2 (Measurement Settings).

Frequency Band Windows and Warning/Alarm Threshold Defaults



Bands Defaults		Suggested xRPM	Frequency Windows Min (hz)	Frequency Windows Max (hz)	Band Thresholds Warning	Band Thresholds Alarm
Unbalance	Band 1 Min/Max Freq (xRPM):	0.3X to 1.2X	0	2	3.564	9.037
Misalignment	Band 2 Min/Max Freq (xRPM):	1.2X to 3.2X	2	6	3.564	9.037
Band 3	Band 3 Min/Max Freq (xRPM):	3.2X to 8.2X	6	16	3.564	9.037
Bearing Faults	Band 4 Min/Max Freq (xRPM):	8.2X to 12.2X	16	24	3.564	9.037
Band 5	Band 5 Min/Max Freq (xRPM):	12.2X to 15.2X	24	30	3.564	9.037
Gear Faults	Band 6 Min/Max Freq (xRPM):	15.2X to 300000 RPM	30	5000	3.564	9.037

This area of the form contains entry fields that allow the user to define certain status conditions that apply to the application. Unless advance mode is selected, these status conditions are already set to include:

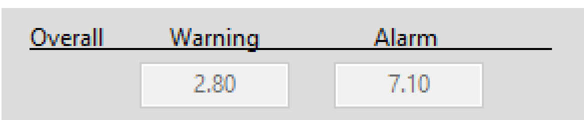
- Unbalance
- Misalignment
- Band 3
- Bearing Faults
- Band 5
- Gear Faults

Each condition is also labeled with a Suggested xRPM range.

Frequency Windows: The entry fields in this section are used to populate fields found in the Frequency Band Windows tab that is found under the Alerts Thresholds tab. Refer to section 6.5.3 (Alerts Thresholds) for more information on Frequency Windows.

Band Thresholds: The entry fields in this section are used to populate fields found in the Frequency Bands Thresholds tab that is found under the Alerts Thresholds tab. Refer to section 6.5.3 (Alerts Thresholds) for more information on Frequency Bands Thresholds.

<u>Overall Warning and Alarm</u>



Warning and Alarm for velocity RMS. These values can also be found in the Accelerometer tab found underneath the Alerts Thresholds tab.

6.5.13.1.2 Driven

This tab is used to define the sensors responsible for monitoring the piece of machinery that is driven (fan) by the drive machine (motor).

Almost all the information found within this tab is the same as the information found within the Drive tab in section 6.5.13.1.1. Refer to the Drive section for more information on each entry field. Key differences are as follows:

- The inboard and outboard channels should not be the same as the Drive tab. For instance, if the drive tab uses channels 1 and 2 for the accelerometers that will be monitoring the inboard and outboard bearings of the motor, the Driven tab should use channels 3 and 4 to monitor the inboard and outboard bearings of the fan.
- **Number of Blades/Teeth/Vanes:** This field is specific depending on which application is chosen. Simply enter the number of blades/teeth/vanes located on the driven machine. This number does impact entry fields once “Apply Defaults Locally” is clicked.

6.5.13.2 Application Specific Fields and Features

Some features and fields are application specific. This section will identify each entry field or feature that differs from application to application.

Motor/Blower (Rolling Element Bearing)

- **Machine Speed Range (RPM):** For this application the machine speed ranges are as follows:
  - 120RPM to 600RPM
  - 600RPM to 1200RPM
  - 1200RPM to 3600RPM
- **Number of Blades:** This is the number of blades of the driven fan. The number will impact default values, so please ensure it is correct.

Motor/Blower (Sleeve Bearing)

- **Machine Speed Range (RPM):** For this application the machine speed ranges are as follows:
  - 120RPM to 600RPM
  - 600RPM to 6000RPM
- **Number of Blades:** This is the number of blades of the driven fan. The number will impact default values, so please ensure it is correct.

Centrifuge w/ GearBox (Rolling Element Bearing)

- **Machine Speed Range (RPM):** For this application the machine speed ranges are as follows:
  - 600RPM and above
- **Number of Teeth:** This is the number of teeth of the gear. The number will impact default values, so please ensure it is correct.

Vertical Motor & Pump

- **Machine Speed Range (RPM):** For this application the machine speed ranges are as follows:
  - 120RPM to 600RPM
  - 600RPM to 1200RPM

- 1200RPM to 3600RPM

- **Number of Vanes:** This is the number of vanes of the driven pump. The number will impact default values, so please ensure it is correct.

7 MODBUS TCP SUPPORT

The X8II supports Modbus TCP for retrieving Channel Overalls and Bias/Gap for wired channels, and Frequency Band Overalls. This support is currently offered for Accelerometers and Proximity Probes. It also supports retrieving NANO measurement data. It requires that the X8II be connected to the local network by a physical ethernet connection.

7.1 WIRED CHANNEL MODBUS TCP

The X8II offers a VM7-B style register address maps for the channel overalls and bias/gap and for frequency bands. It supports the standard Modbus TCP function code 03 for retrieving these values. The register addresses for retrieving the channel overalls and Bias/Gap are shown below:

Overalls and Bias/Gap

Channel		Slot1 Address
1	Measure	42101
1	Gap	42103
2	Measure	42105
2	Gap	42107
3	Measure	42109
3	Gap	42111
4	Measure	42113
4	Gap	42115
5	Measure	42117
5	Gap	42119
6	Measure	42121
6	Gap	42123
7	Measure	42125
7	Gap	42127
8	Measure	42129
8	Gap	42131

These addresses are the same addresses as the VM7-B uses for Slot 1. The “Measure” is a 32-bit floating point variable and is dependent upon the transducer type for the respective channel. If the transducer type is an Accelerometer, then the “Measure” returned for the channel will be Velocity RMS and the “Gap” returned will be the Accelerometer Bias in Volts. If the transducer type is a Proximity Probe, the “Measure” returned will be micrometers Peak-to-Peak, and the “Gap” returned will be Proximity Gap in Volts.

The X8II mimics holding registers stored at the above addresses.

Example 1:
Using the standard Modbus TCP function code 03, the user may specify a start address of 42101 and number of registers equaling 4, and the X8II will return a Modbus TCP byte stream for just Channel 1 for “Measure” and “Gap”. This Modbus TCP function Code 03 byte stream (from a PLC) would be:

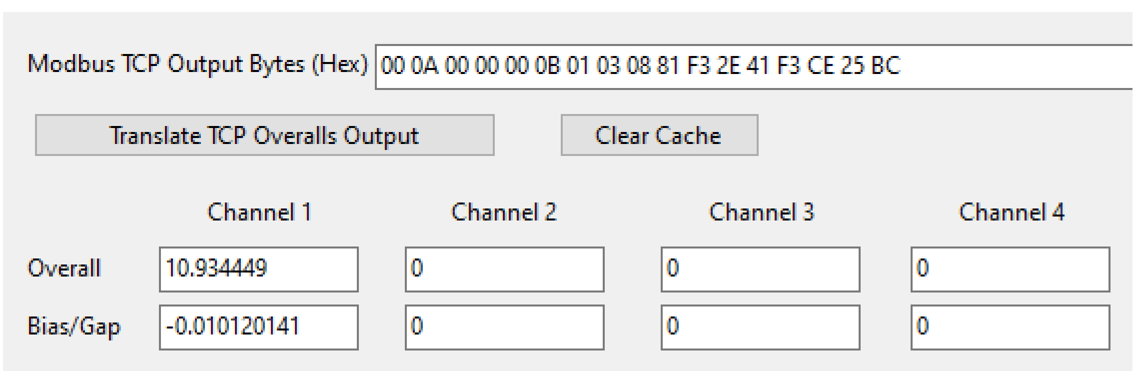
00 0A 00 00 00 06 01 03 A4 75 00 04

The “03” in the byte stream above is the function code in hex. The “A4 75” is the register address (in little-endian) 42101 in hex. The “00 04” is the number of registers requested in hex.

The X8II will parse these bytes and interpret the function code 03, and the arguments passed in the byte stream. It will recognize that the address of 42101 (A4 75) is wired channel 1, and it will return 4 registers of measurement data. Each register represents 16-bits, so a 32-bit float is returned in 2 registers. Requesting 4 registers is equivalent to requesting 2 32-bit floats. For this address, it will be the Overall followed by the Bias/Gap value. In the above example, the X8II will receive a channel 1 overall Modbus TCP byte stream in Velocity RMS and Bias voltage with the following byte stream:

00 0A 00 00 00 0B 01 03 08 81 F3 2E 41 F3 CE 25 BC

The “03” is the function code that is returned in the byte stream, and the “08” following the “03” is the number of bytes of data returned. Two floating point variables are returned in these 8 bytes. The first is “81 F3 2E 41” and is little-endian for a 32-bit float that is “10.934449” and is Velocity RMS, since channel 1 is an Accelerometer. The remaining 2 bytes “F3 CE 25 BC” is the Bias voltage in volts, or “-0.010120141” volts.



The above is from an engineering test program used by Shinkawa Electric Corporation of America (SECA) engineers to test the X8II Modbus TCP functionality.

Example 2:
If the PLC sends a start address of 42101 and requests 32 registers, then all 8 wired channel overalls and Bias/Gap will be returned:

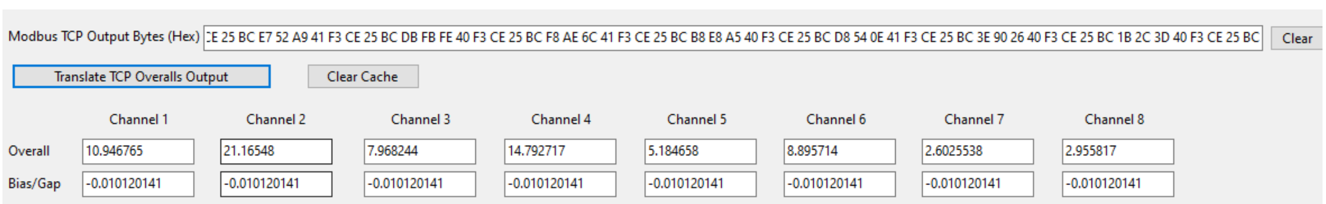
00 0A 00 00 00 06 01 03 A4 75 00 20 (the Modbus TCP function code 03 received from the PLC)

Note this is the same as the Modbus TCP byte stream above, with the exception of the last byte, “20”. This is hex for 32 registers requested.

The X8II receives this over TCP, parses the command, interprets the start address to be 42101 and maps that to channel 1, then returns the next 32 register, which is all 8 wired channels Overalls and Bias/Gap:

00 0A 00 00 00 43 01 03 40 F3 25 2F 41 F3 CE 25 BC E7 52 A9 41 F3 CE 25 BC DB FB FE 40 F3 CE 25 BC F8 AE 6C 41 F3 CE 25 BC B8 E8 A5 40 F3 CE 25 BC D8 54 0E 41 F3 CE 25 BC 3E 90 26 40 F3 CE 25 BC 1B 2C 3D 40 F3 CE 25 BC

The above is the byte stream returned from the X8II to the PLC. The “03” in the 8th byte is the function code, and the “40” is the number of bytes of data returned, in hex. That is 64 in decimal and is the number of bytes required for 8 channels with 2 floats per channel. We can interpret this using an internal test program:



Besides Overalls and Bias/Gap, the X8II offers a 2nd address range (also based upon the VM7-B holding register addresses, specifically for Frequency Bands. The register address map is shown below:

Bands		
1	RPM	43601
1	Band 1	43603
1	Band 2	43605
1	Band 3	43607
1	Band 4	43609
1	Band 5	43611
1	Band 6	43613
1	Unused	43615
1	Unused	43617
1	Unused	43619
1	Unused	43621
1	Unused	43623
2	RPM	43625
2	Band 1	43627

2	Band 2	43629
2	Band 3	43631
2	Band 4	43633
2	Band 5	43635
2	Band 6	43637
2	Unused	43639
2	Unused	43641
2	Unused	43643
2	Unused	43645
2	Unused	43647
3	RPM	43649
3	Band 1	43651
3	Band 2	43653
3	Band 3	43655
3	Band 4	43657
3	Band 5	43659
3	Band 6	43661
3	Unused	43663
3	Unused	43665
3	Unused	43667
3	Unused	43669
3	Unused	43671
4	RPM	43673
4	Band 1	43675
4	Band 2	43677
4	Band 3	43679
4	Band 4	43681
4	Band 5	43683
4	Band 6	43685
4	Unused	43687
4	Unused	43689
4	Unused	43691
4	Unused	43693
4	Unused	43695
5	RPM	43697
5	Band 1	43699
5	Band 2	43701
5	Band 3	43703
5	Band 4	43705
5	Band 5	43707
5	Band 6	43709
5	Unused	43711
5	Unused	43713

5	Unused	43715
5	Unused	43717
5	Unused	43719
6	RPM	43721
6	Band 1	43723
6	Band 2	43725
6	Band 3	43727
6	Band 4	43729
6	Band 5	43731
6	Band 6	43733
6	Unused	43735
6	Unused	43737
6	Unused	43739
6	Unused	43741
6	Unused	43743
7	RPM	43745
7	Band 1	43747
7	Band 2	43749
7	Band 3	43751
7	Band 4	43753
7	Band 5	43755
7	Band 6	43757
7	Unused	43759
7	Unused	43761
7	Unused	43763
7	Unused	43765
7	Unused	43767
8	RPM	43769
8	Band 1	43771
8	Band 2	43773
8	Band 3	43775
8	Band 4	43777
8	Band 5	43779
8	Band 6	43781
8	Unused	43783
8	Unused	43785
8	Unused	43787
8	Unused	43789
8	Unused	43791

For each channel, there is an RPM float variable followed by 6 Band Overall float variables. Note that there are also 5 unused float variables, this is because the VM7-B holding register address map allocated a total of 12 float variables, but the X8II only uses the 1st 7 for each channel.

Like the Overalls and Bias/Gap channel measurement retrieval, the Frequency Band retrieval is also accomplished by using the standard Modbus TCP function code 03, along with a requested number of registers (again where 2 registers are required for each float variable).

Example 3:
The PLC can seek to retrieve Frequency Band measurement data starting at holding register address 43601. These are taken from the VM7-B holding register addresses for Frequency Bands, from Slot 1.

The PLC will send a function code 03 and a starting address for channel 1 of 43601, and 14 registers requested (7 * 2 registers per float). This is for all 7 floats for channel 1 (RPM plus 6 Band overalls). The Modbus TCP bytes stream from the PLC would for this example would be:

00 0A 00 00 00 06 01 03 AA 51 00 0E

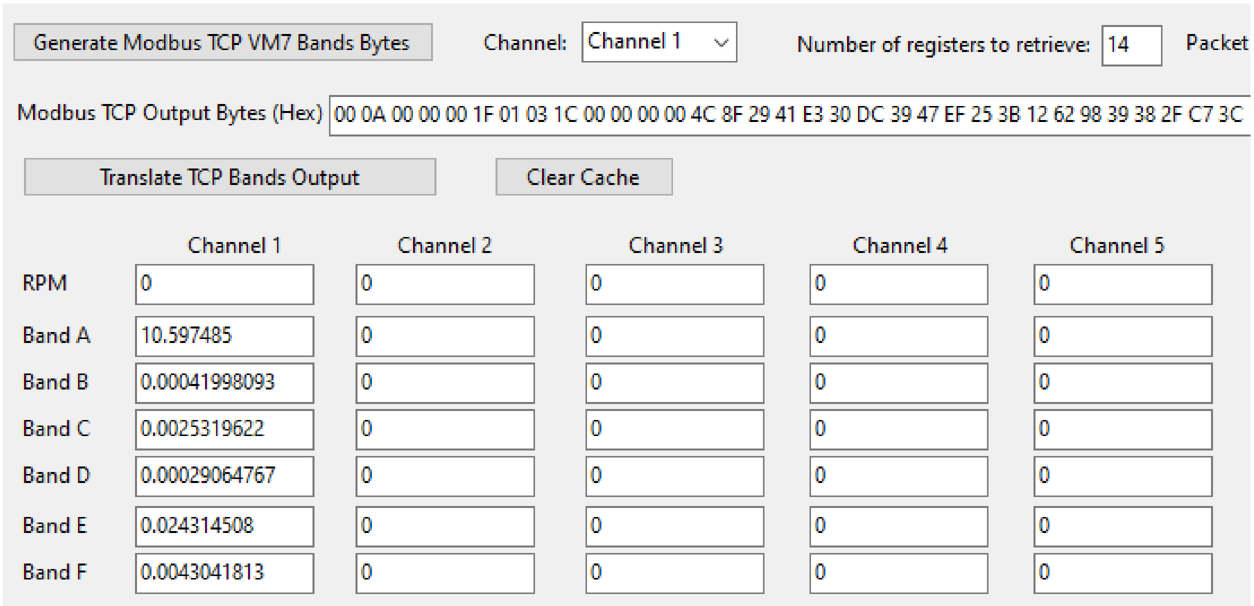
Note the “03” in the 8th byte. Again, that is the function code “03” in hex. The “AA 51” is 43601 in hex, and the “00 0E” is the number of registers requested, or 14 in decimal. This should result in the X8II returning the RPM float and the 6 Band Overalls for Channel 1.

The X8II will receive this command over TCP (ethernet), parse it and conclude it is requesting Band measurements and all 7 floats for Channel 1. The X8II will send back this byte stream over TCP to the PLC:

00 0A 00 00 00 1F 01 03 1C 00 00 00 00 4C 8F 29 41 E3 30 DC 39 47 EF 25 3B 12 62 98 39 38 2F C7 3C 17 0A 8D 3B

Note the “03” in the 8th byte, followed by “1C.” The “1C” in hex is the number of bytes of data returned by the X8II in response to the function code 03 request received. “1C” is hex for 28, which is 7 floats * 4 bytes per float.

The PLC will be able to parse this data, but for purposes of demonstration we can observe using our internal testing program:



Example 4:

The PLC may request measurement data for multiple channels at once, starting with the 1st address for channel 1, or 43601, and request a multiple of 12 floats (recall that each channel has 12 floats total, 5 of them unused). Two registers are required per float variable, so if the PLC requests Band measurements for 4 channels, starting with Channel 1, it would request 4*12*2 registers, or 96 registers. The byte stream that the X8II would receive for this function code 03, starting at 43601 (channel 1) and requesting 96 registers would be:

00 0A 00 00 00 06 01 03 AA 51 00 60

Note again the "03' function code in byte 8, and the last 2 bytes are "00 60" in hex. This is 96 in decimal.

The X8II would receive this byte stream over Modbus TCP, parse the function code 03 command, map the register address to Channel 1, and return the next 96 registers (or 48 floats, where each channel is 12 floats , 5 unused).

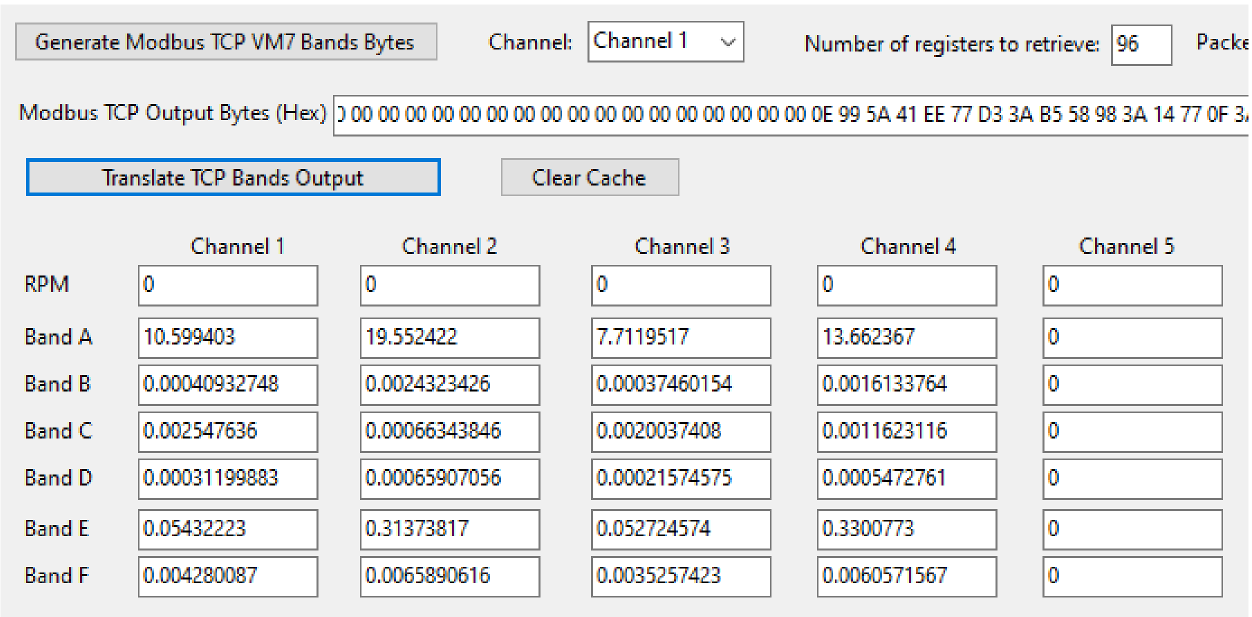
The PLC would receive the following byte stream back from the X8II:

00 0A 00 00 00 C3 01 03 C0 00 00 00 00 28 97 29 41 01 9B D6 39 3D F6 26 3B C6 93 A3 39 FD 80 5E 3D F9 3F 8C 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 5C 6B 9C 41 F0 67 1F 3B 9A EA 2D 3A 7A C5 2C 3A 4A A2 A0 3E 0E E9 D7 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4F C8 F6 40 2B 66 C4 39 31 51 03 3B CF 39 62 39 B9 F5 57 3D 24 10 67 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0E 99 5A 41

EE 77 D3 3A B5 58 98 3A 14 77 0F 3A E4 FF A8 3E 1D 7B C6 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Note, again, the “03” in the 8th byte followed by “C0”, which is the number of bytes of data returned by the X8II. “C0” is 12*16, or 192 bytes, which is 4*12 floats *4 bytes per float.

Again, the PLC would be able to parse this byte stream, but for purposes of demonstration, we can use an internal program to interpret the results:



Note that the speed for each band is 0. That is because the Speed input for the X8II was not used.

7.2 Modbus TCP for NANO Measurements

The X8II also supports retrieval of NANO measurement data from an X8II. This retrieval does not directly connect to a NANO, but instead retrieves the latest stored NANO measurement that the X8II saved to its flash memory.

For Modbus TCP NANO measurement retrieval, the X8II offers 3 different methods. The 1st two require a “custom” Modbus TCP function code, while the 3rd method relies upon standard Modbus TCP function codes (0x03, or Holding Register Reads).

7.2.1 Nano Modbus TCP Method 1

The 1st of the 3 Modbus TCP NANO methods uses 1 Modbus TCP function code, and it is the custom code 0x41. The PLC needs to send a byte stream that includes 0x41 as the function code, followed by the next 8 bytes (the NANO Serial Number). The X8II will receive this, and parse the function code as 0x41, and parse the requested NANO

Serial number. It will then scan its flash memory for saved NANO measurements and if it finds a measurement, it returns a large set of measurement data in the output byte stream

Example 5:
The PLC would send a byte stream using the 0x41 function code, and the NANO serial number, which is 8 bytes of data in the byte stream. For instance, for NANO Serial Number "Z2R10002," the PLC would send the 0x41 function code byte stream:

00 0A 00 00 00 0A 01 41 5A 32 52 31 30 30 30 32

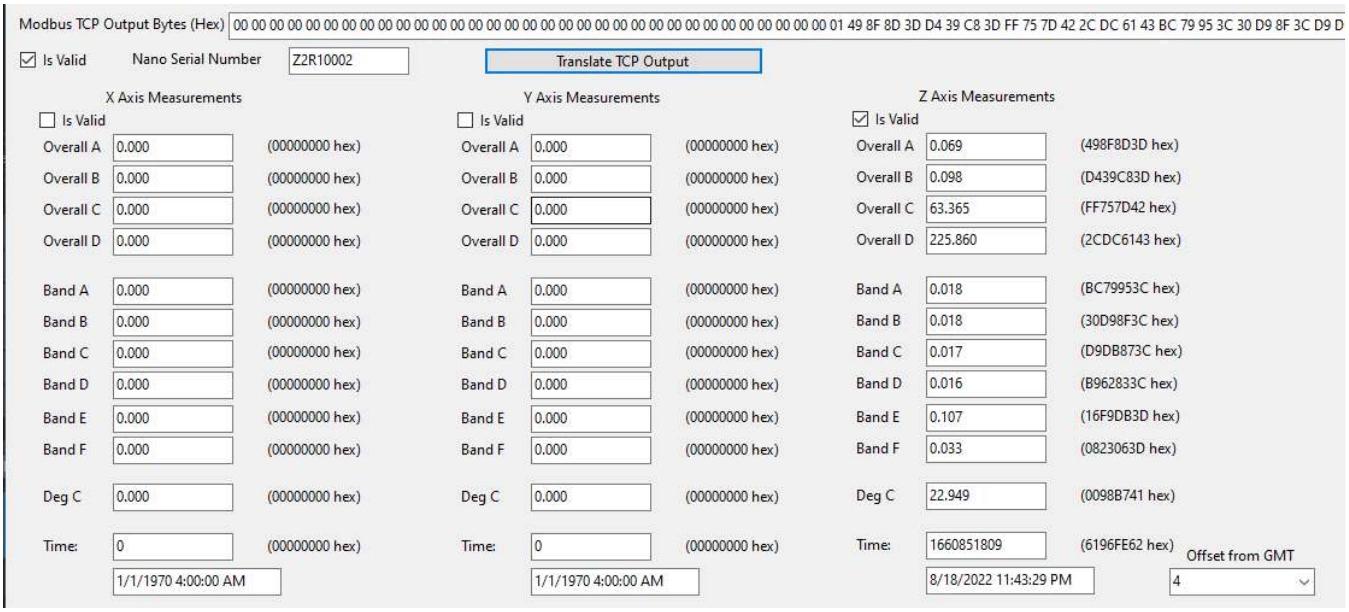
Note the "41" in the 8th byte, followed by 8 bytes of data "5A 32 52 31 30 30 30 32." These 8 bytes represent the "Z2R10002" serial number. The X8II would receive this byte stream, and parse the function code 0x41, and then parse the next 8 bytes as the requested serial number. Then it will search for the most recent measurement stored in this X8II in the flash memory, and if found, it returns the measurement data. The data returned from this custom function code (0x41) includes all 3 axes of the NANO, including a "isValid" flag for each axis and 4 overalls (A, B, C, and D, which are Velocity RMS, Velocity Calc Peak, Acceleration RMS, and Acceleration True Peak, which are the built-in overalls for the NANO), and 6 Frequency Band overalls (A-F) and also the measured temperature in degrees Celsius, and the time stamp of each measurement.

The X8II will return a byte stream like this:

00 0A 00 00 00 9F 01 41 9C 01 5A 32 52 31 30 30 30 32 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 01 49 8F 8D 3D D4 39 C8 3D FF 75 7D 42 2C DC 61 43 BC 79 95 3C
30 D9 8F 3C D9 DB 87 3C B9 62 83 3C 16 F9 DB 3D 08 23 06 3D 00 98 B7 41 61 96
FE 62

Note the "41" in the 8th byte, followed by the "9C", which is the number of bytes of data returned.

In this example, Z2R10002 is configured to not be triaxial, so it only measures the NANO Z-axis. For this reason, the measurement data for the X and Y axis of the NANO will be all 0's. The PLC will be able to interpret this data, but for demonstration purposes, we display the result in an internal program:



Note that only the Z-axis is marked as valid. Also note the temperature and time stamp of the measurement.

7.2.2 Nano Modbus TCP Method 2

The 2nd method that is offered for Modbus TCP NANO measurement retrieval is a 2-step process. This means that the PLC will need to send two Modbus TCP function code commands. The 1st of these two function codes is another "custom" Modbus TCP function code, 0x42. This custom function code is like 0x41, except that instead of immediately returning all the NANO measurement data for 3 axes, the function code prepares the NANO measurement for the 2nd Modbus function code, which is the standard Modbus TCP function code 03, that we have shown for wired channel measurement retrievals. Basically, the 1st function code call (0x42) tells the X8II the NANO serial number to look up, and if found, it stores the measurement data in a fixed holding register address of 44000. The 2nd (standard 03) function code then requests the NANO measurement data stored at holding register 44000, along with the number of registers requested.

Example 6:

Step 1:
The PLC sends the 1st custom Modbus TCP function code, 0x42, along with the NANO Serial number. The byte stream that the PLC would send is this, for NANO Z2R10033:

00 0A 00 00 00 0A 01 42 5A 32 52 31 30 30 33 33

Note the “42” in the 8th byte. This is the custom function code. The next 8 byte represent the “Z2R10033” serial number. The X8II will receive this, and parse the function code and the serial number, and will search its flash memory for the latest saved measurement for Z2R10033. It will return a successful flag to the PLC.

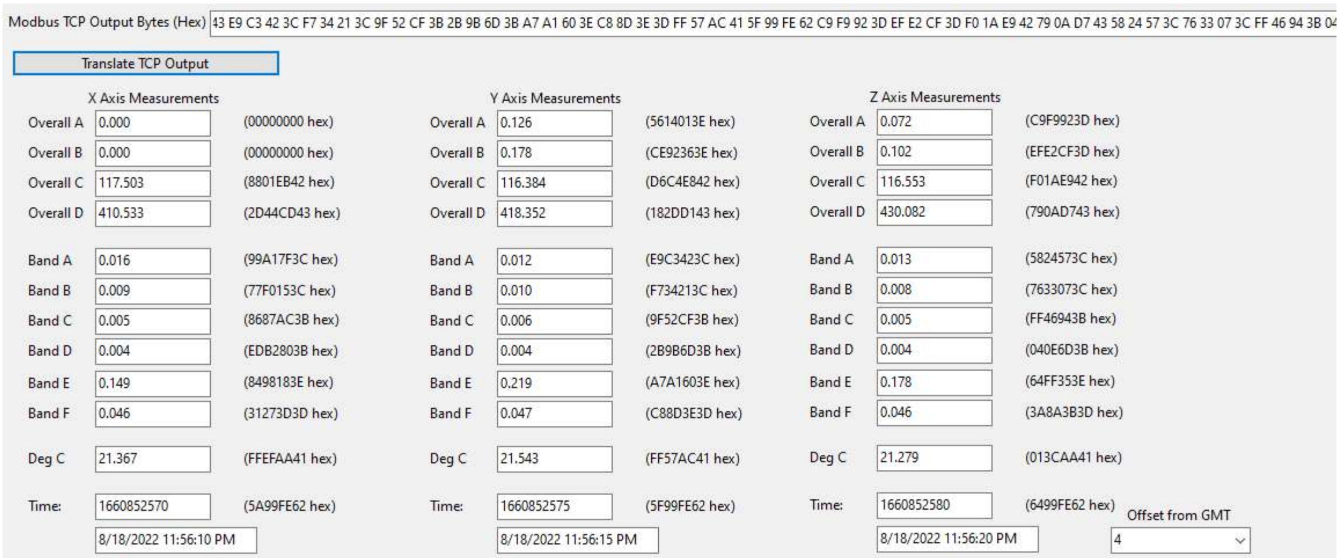
Step 2:
Next, the PLC would send a standard Modbus TCP function code 03, requesting the number of registers for the 3 NANO axes (Overalls A-D, Bands A-F, Temperature and Time Stamp per axis). This byte stream would be:

00 0A 00 00 00 06 01 03 AB E0 00 48

Note the “03” in the 8th byte followed by “AB E0”, which is the “fixed” holding register address dedicated to recently requested NANO measurement storage. Note also the “00 48”, which is the number of registers requested, which is 72 in decimal. This is for 3 axes, where each axis is 12 floats (Overalls A-D, Bands A-F, Temperature and TimeStamp). 12 * 3 = 36 floats, which requires 72 registers (2 registers per float). The X8II will receive this byte stream over TCP (by ethernet), and parse the standard function code 03, and the holding register address, 44000, which is dedicated to storing the recently retrieved NANO measurement (from Step 1), and the number of registers requested. The X8II will return a byte stream like this:

00 0A 00 00 00 93 01 03 90 00 00 00 00 00 00 00 00 88 01 EB 42 2D 44 CD 43 99 A1 7F 3C 77 F0 15 3C 86 87 AC 3B ED B2 80 3B 84 98 18 3E 31 27 3D 3D FF EF AA 41 5A 99 FE 62 56 14 01 3E CE 92 36 3E D6 C4 E8 42 18 2D D1 43 E9 C3 42 3C F7 34 21 3C 9F 52 CF 3B 2B 9B 6D 3B A7 A1 60 3E C8 8D 3E 3D FF 57 AC 41 5F 99 FE 62 C9 F9 92 3D EF E2 CF 3D F0 1A E9 42 79 0A D7 43 58 24 57 3C 76 33 07 3C FF 46 94 3B 04 0E 6D 3B 64 FF 35 3E 3A 8A 3B 3D 01 3C AA 41 64 99 FE 62

Note, again, the “03” in byte 8 followed by “90”, which is the number of bytes that the X8II returned, or 144 bytes. Which is 36 float variables, 12 per axis. The PLC will be able to interpret this output, but for the purpose of demonstration, we show the results from an internal program used for testing this feature:



7.2.3 Nano Modbus TCP Method 3

The 3rd method that is offered for Modbus TCP NANO measurement retrieval is also a 2-step process, but in this case both steps use the standard Modbus TCP function code, 0x03. This method is a generally more compatible version of NANO measurement retrieval since it uses only the standard Modbus TCP function code, 0x03. This means that the PLC will need to send two Modbus TCP function code 0x03 commands.

The 1st of these two-function code 0x03 (Holding Register Read) commands sends a special Holding Register address, 45000, that the X8II recognizes as a request from the PLC to retrieve all Nano serial numbers that have been serviced by the current X8II. Along with these serial numbers, the X8II dynamically assigns special holding register addresses, one for each NANO serial number. The X8II finds all NANO serial numbers that have had NANO measurements stored in its flash memory, up to 16, and then assigns a holding register address sequentially. The 1st NANO is assigned 45002, and each subsequent NANO is assigned an address incremented by 2. So, to send this first Modbus TCP function code 0x03 command, the user specifies the address as 45000, and the number of registers to read is 96 registers. That is because 16 NANO serial number and addresses are requested. This is a fixed size. The NANO serial number is 8 characters, by 10 are allocated due to byte-alignment. So, for the serial number, 10 bytes are required. The associated holding register address is a 16-bit unsigned integer, requiring 2 bytes. So, for each NANO, 12 bytes are returned. For 16 NANO serial numbers, this requires 192 bytes, or 96 registers (0x60) (each register is a 2-byte integer). So, the PLC sends a function code 0x03 command, with holding register address 45000, and requesting 96 registers (or 192 bytes). As mentioned above, the 1st NANO serial number will be assigned 45002, the 2nd will be assigned 45004, etc…

Example 7:
The PLC would send a byte stream using the 0x03 function code, and the special Holding Register address of 45000, or 0xAFC8. For instance, for NANO Serial Number "Z2R10002," the PLC would send the 0x41 function code byte stream:

00 0A 00 00 00 06 01 03 AF C8 00 60

The PLC Ladder Logic will display the serial numbers and associated register addresses, and the user will choose a NANO serial number, and use the associated assigned holding register address to set up the 2nd step in this 2-step process.

For development and testing purposes, we can display the results of this 1st step using an internally-developed display program:

The user can choose whether to use the 1st method (a single custom Modbus TCP function code that retrieves all the data at once), or the 2nd , 2-step method, that first calls a custom Modbus TCP function code that prepares the NANO measurement data, that follow-up standard Modbus TCP function code 03 retrieves from holding register 44000. Or the user may use the more compatible 3rd method (2-step), that uses only the 0x03 function code. Where the 1st step retrieves a list of up to 16 NANO serial numbers and assigned Holding Registers. The 2nd step of this 3rd method specifies a specific assigned Holding Register address to retrieve the measurements for a specific NANO device.

8 Flash Codes and Troubleshooting

The 3 LED lights are labeled on the top PCB board:

1. POWER
2. Wi-Fi
3. BLE

Once power has been established the X8II Hub will run 8 self-tests flashing the POWER LED either **GREEN** or **RED**, depending on whether the specified test condition has **PASSED** or **FAILED**.

Provided power has been established and the LEDs are active, please follow the information below to better understand the LED flash codes and what they mean. For all the tests below, a **GREEN** flash will indicate it passes and a **RED** flash will indicate failure.

1. Flash #1 – Bootloader Present Check
2. Flash #2 – IO Expander Check
3. Flash #3 – RTC Device Check
4. Flash #4 – ADC Device Check
5. Flash #5 – Flash Memory Device Check
6. Flash #6 – External SRAM Memory Device Check
7. Flash #7 – Bluetooth Device Check
8. Flash #8 – Wi-Fi Device Check
9. Flash #9 – Ethernet Self-Test

If all 9 self-tests above pass and flash green, the program will launch, and the device will work normally. If any of the 9 checks flash red, the program will not launch, and the LED will continue to run the self-test flashing the 8 times every 5 seconds.

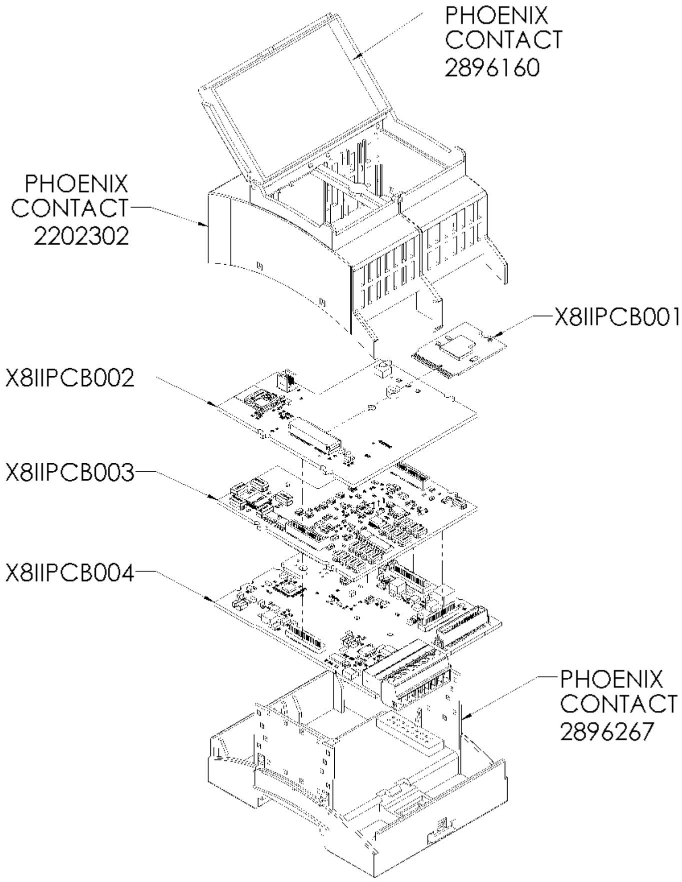
An example of a failed test might look like the following:

Flash 1
Flash 2
Flash 3
Flash 4
Flash 5
Flash 6
Flash 7
Flash 8
Flash 9

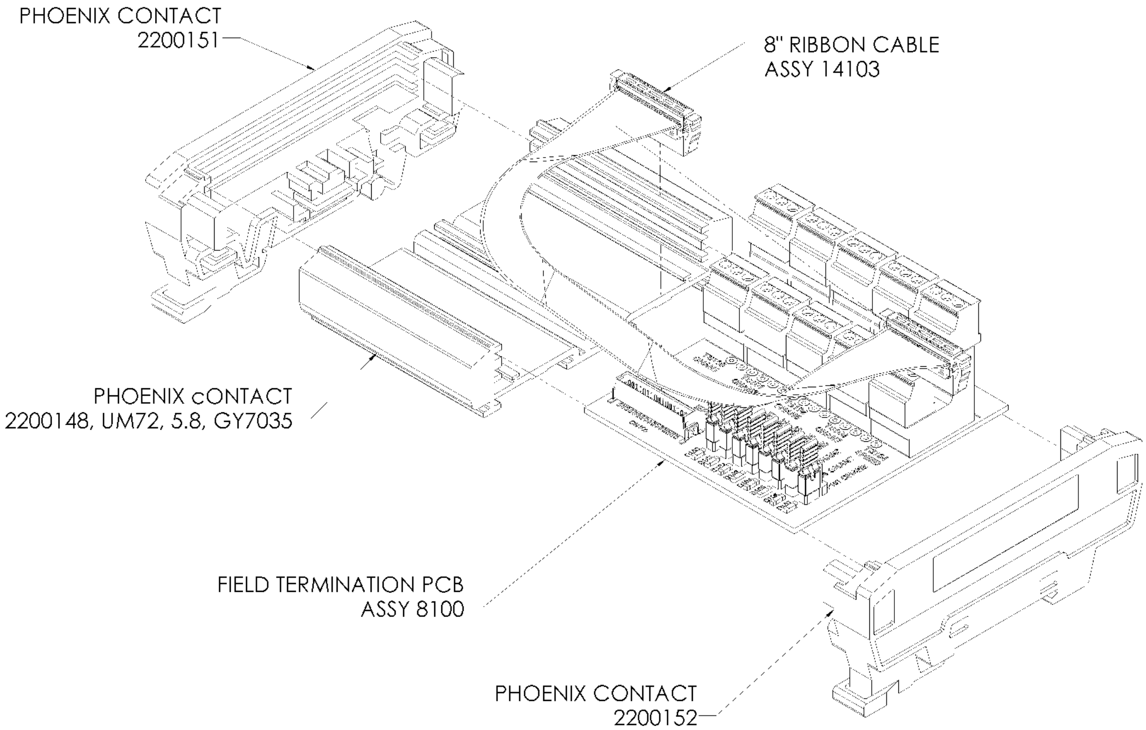
In the above flash code sequence, flashes 2 and 5 fail while all the rest pass. Referring to the flash code definitions, provided earlier in this section, we can see that a failure in flash #2 indicates an IO Expander device failure and a failure in flash #5 indicates a Flash Memory Device failure. Since at least one test failed, the program will not boot, and this flash sequence will continue every 5 seconds. If any of the tests fail, and the program fails to boot, please call customer service. The image above outlines the LED in **yellow**.

9 REPLACEMENT PARTS AND ACCESSORIES

9.1 X8II PARTS LIST



9.2 FIELD TERMINATION PARTS LIST



9.3 RECOMMENDED ACCESSORIES

- DIN Rail Power Supply: TDK-Lambda DRB50-5-1
- Antenna: Linx Technologies CSI-RSFE-300-UFFR
- Circuit Breaker: Phoenix Contact 0916608
- Plastic Enclosure: Hoffman A12106CHSCFG
- DIN Rail Stop: Phoenix Contact 3022276

10 SYSTEM SPECIFICATIONS

X8II Specifications		
Hardware Capacity	Max Sensor Count	Eight (8) Wired Sensors/Sixteen (16) Wireless NANO Sensors, One (1) Speed
	Sensor Types	Accelerometers/Prox Probe/Temperature/Strain Sensor
		Process signals: (0-5VDC/1-5VDC/0-10VDC/1-10VDC and 4-20mA with added 249 ohm terminating resistors per channel on the 4-20mA inputs
	Frequency Range	2Hz (120 cpm) to 20KHz (1,200,000 cpm) ±3dB
	Bias Measurement	0 to +24V or 0 to -24V user selectable
Functions	Vibration Measurement Type	Acceleration: g or mm/s2 ; Velocity: in/s or mm/s
	Vibration Amplitude Type	RMS, True Peak, Calculated Peak
	Waveforms	With wired sensor: 8192 floating point data With wireless NANO sensors: 2048 floating point data
	Number of Resolution Lines (FFT)	800, 1600, 3200 (default -3200)
	Fmax (Sampling rate)	1500, 3000, 5000, 10000, 18300 (default 3000)
	Frequency Bands	Six (6) Frequency Bands with Independent Alarms
	Other Measurement Types	Temp: °F (°C), Frequency: Hz (CPM), Volt: V, Strain µ€, Speed: RPM
	Alarming	Machine OFF, OK, WARNING and ALARM
Activation Method	By Sheduled Time	1,2,3,4,6,12 Hours/1 Day Interval changes based on Alarms to 1 hour interval
	By Detected Event	Immediate activation when set alarm limits are exceeded
Communication	Wi-Fi / Wireless	Standard: IEEE 802.11b/g/n
		Frequency Band:2.4000~2.4840 GHz for Wi-Fi
		Certification: CE/FCC/IC/TELEC/SRRC/NCC
		Wireless Security: WPA/WPA2 support for powerful encryption for and authentication AES in WLAN hardware for faster data encryption and IEEE 802.11i compatibility (For wireless module only)
		Transmit Power: Typ. 14 dBm ± 2 at 802.11b CCK Mode 1M Typ. 12dBm ±2 at 802.11g OFDM Mode 54M Typ. 12 dBm ± 2 at 802.11n OFDM Mode MCS0
		Reception Sensitivity: -65 dBm
		RF Data Rate: Up to 65 Mbps
		Antenna connector: MHF4 connector
	Wired	RJ45 Ethernet
		Note: Cable via Ethernet maximum distance is 100 meters (for greater distance, fiber optics is recommended.)
	Wireless Range	Bluetooth 5.0 communication to the NANO: Maximum of 100 meters with linear vision. (The actual range depends on the obstacles present)

Power	Power Supply Voltage	5.0 Vdc, 6 Amps
Environmental	Operating Temperature	-4°F to +140°F (-20°C to +60°C)
	Operating Humidity	5% ~ 95% Relative Humidity, non-condensing
Physical	Weight	12 oz (340 gr)
	Dimension	X8II Housing (54 x 71 x 90 ; 2 1/8” x 2 13/16” x 3 35/64”)
	Material	X8II Housing: Polycarbonate
	Mounting	DIN Rail
Software	SHINKAWA / Rockwell Automation	Machine Dossier/EMonitor

11 GLOSSARY

Advertisement: Signal sent by NANOs letting all X8IIs in range know it is ready to offload measurement data.

Alarm Thresholds: User specified values that dictate when alert notifications will be sent. These values can be set using the X8UserConfigurationTool.

Buffered Measurements: Queue for measurements that are waiting to be sent by the X8II. This queue begins if a measurement fails to send, which will then cause all the measurements taken during that interval to get in queue behind the original failed measurement.

DAPLINK: Drive that will populate the windows driver list after the USB cable has been connected to your PC while programming the X8IIs WIFI (7.2.4.5).

EMonitor: condition monitoring software used to analyze data received from X8II units.

infiSYS Lite: Another destination where data can be delivered to by the X8II units.

MachineDossier.net (MD): This is our cloud-hosted application in which authorized users can view their organization's X8II collected data, update and change X8II data collection schedules and collaborate with others to diagnose machinery problems.

Measurement Interval Time: Time between sampling rate of sensors. Once a day is a typical sampling rate for these sensors, but if this is increased it could burden the system.

Measurement Point: Physical location of sensors used to sample data that is sent to the X8II. Once physically mounted and connected to the X8II, these points are configurable, and characteristics are defined using MD.

Measurement Start Time: Pre-configured time for sensor to begin the sampling process.

Measurement-For-Send: When the measurement begins, if the Next Send Time for that channel is the same time or earlier, it will be saved to flash memory.

NANO: Sensor that is mounted on the machines being monitored. These sensors are programmed to sample at certain times and communicate with a nearby X8II to offload the data.

Next Send Time: Next scheduled time for measurements to be sent.

NIST Time Server: Server computer that reads the actual time from a reference clock and distributes this information to its clients using a computer network.

Order-Specific Commands: These commands are part of a "flow" or sequence of commands.

Over-The-Air (OTA): Any type of wireless transmission.

Pinging: An attempt to test the reachability of a host on an Internet Protocol (IP) Network. This will be used to test the reachability of the FTP Server after installation has been completed.

PLC (Programmable Logic Controller) - Industrial computer that has been ruggedized and adapted for the control of manufacturing processes.

Real Time Operating System (RTOS): The X8II uses this system in order to perform multiple tasks at one time. Previously, the X8 could only perform one task at a time.

RTC Clock: Real-time-clock, which is internal to the X8II and is what is used to accurately take timestamped measurements. The RTC Clock will sync with your PC to calibrate and can also be manually updated, if necessary.

Self-Test: Consists of 8 different hardware tests. When it completes the last test, the X8II will indicate the result of the self-test with a series of single-LED flashes. It will flash once for each of the 8 tests, in sequence. If the test passes, it will flash green. If the test fails, it will flash red, and have a slightly longer duration. Within about 12 seconds after power-on, this light sequence will begin.

Sending Task: Measurement harvesting task that determines if the send was successful. If the measurement has no more destinations to be sent to, it purges the measurement from the measurement queue.

SQL Database: All data for the MachineDossier.net function is stored in a SQL relational database management system.

Vibration Sensor Mode: If an X8II has been configured already, and the stored network settings allow communication to the Zeep Server and MachineDossier.net, it will boot into this mode.

Zeep Server: A server used to communicate between Machine Dossier and the X8II units.

12 INDEX