Product Manual - SparkPNT SXM-E



Table of contents:

Welcome

- Get Started
- Quick Start
- Product Overview
- Hardware Overview
- Hardware Assembly
- Software Overview
- Basic Configuration
- Advanced Configurations
- Support

Product Overview

- SparkPNT SXM-E
  - SparkPNT SXM-E
- Parts List
- Specifications
  - Features
    - Connections
  - Antennas
  - Schematic
  - Dimensions
- How It Works
- Suggested Reading
    - GPS Basics
    - What is GPS RTK?
    - Setting up a Rover Base RTK System
    - How to Build a DIY GNSS Reference Station
    - mosaic-X5 GNSS Breakout Board Hookup Guide
    - Qwiic OLED 1.3" Hookup Guide
    - How to Install CH340 Drivers
    - Serial Terminal Basics
    - Logic Levels
    - I2C
    - Serial Communication
    - How to Work with Jumper Pads and PCB Traces
    - GPS Basics
    - GPS vs GNSS
    - What is Correction Data?
    - Real-Time Kinematics Explained
    - DIY RTK Surveying

    - New Video: Unlocking High-Precision RTK Positioning

Hardware Overview

- Power
- Septentrio logo mosaic-X5
- Espressif logo ESP32-WROVER
- Ethernet PHY Interfaces
- USB-C Connectors
- µSD Socket
- SMA Connectors
- I/O Terminals
- Status LEDs
- OLED Display
- Buttons
- Jumpers

Hardware Assembly

- What is in the Box?
- USB-C Ports
- Antennas
    - GNSS Antenna
    - WiFi Antenna
- Ethernet Jacks
- SD Card Slot
- IO Terminals
    - Radio Transceivers
- Outdoor Enclosure
        - How to Build a DIY GNSS Reference Station
- Enclosure Disassembly

Software Overview

- Septentrio logo mosaic-X5
    - RxTools Software Suite
        - Installation Instructions
    - USB Driver
    - Web Interface
        - Firmware Update
        - GPS Data
        - More Videos
- Espressif logo ESP32
    - USB Driver
    - Terminal Emulator
    - Serial Terminal Basics
    - WiFi Credentials for the Network Bridge

    - ESP32 Firmware - Update
    - ESP32 Firmware - Compiling from source
    - How SparkFun does it
    - Using Docker
        - Clone, fork or download the repo
        - Install Docker Desktop
        - Using Docker to create the firmware binary
- Footnotes

Basic Configuration

- Base Station - Setup
- Base Station - Output Corrections
- NTRIP Client - PointOneNav
- NTRIP Client - Test Corrections

Base Station - Setup

- Temporary Base
- Fixed Base

Base Station - Output Corrections

- Create a New Base Station Reservation
- Configure the NTRIP Server

NTRIP Client - PointOneNav

NTRIP Client - Test Corrections

Advanced Configurations

- Enabling GNSS Signals
- Mining with Onocoy

Enabling GNSS Signals

- Supported GNSS Signals
- Enabling Signal Reception
    - Web Interface
    - Command Interface
- Output L5 Corrections
    - Web Interface
    - Command Interface

Mining with Onocoy

- Onocoy Documentation
- Device Configuration
    - Enable L5 Band Reception
    - Connect to the Onocoy Network
    - Enable L5 Corrections
- Receive Rewards

Support

- Technical Support

- 🗃️ Resources

Technical Support

- Order and Shipping Issues
  - Returns Process
- Troubleshooting Tips
  - What should be in the Box?
  - WiFi Connectivity
  - WiFi Provisioning
  - Data Logging
  - OLED Dead Pixels
  - Vehicle Power - Dead Battery
  - L5 Band
  - Expected Accuracy
  - Reprogramming the ESP32
  - mosaic-X5 Security Credentials
  - Enclosure Disassembly

Resources

- Product Resources
- Product Specifications
- Product Comparison

Product Resources

- Product Resources
  - Additional Resources
- 🏭 Manufacturer's Resources
  - Septentrio: mosaic-X5
  - Espressif: ESP32-WROVER

Product Specifications

- Features
  - Connections
- Antennas

Product Comparison

Welcome



This is the online product manual for the SparkPNT SXM-E; a high-performance, precision, all-band GNSS receiver. If this is your first time using the SXM-E, check out our Quick Start Guide for initial instructions on setting up the SXM-E as an RTK base station or NTRIP server. Again, thank you for purchasing our SparkPNT SXM-E!

Designed and manufactured in Boulder, Colorado, USA, the SparkPNT SXM-E is the perfect solution for your high-precision positioning and navigation needs. Based around the multi-constellation, multi-frequency mosaic-X5 module from Septentrio, this is our most advanced GNSS receiver to date. It supports signals from the GPS (USA), GLONASS (Russia), Beidou (China), Galileo (Europe), Navic (India), and QZSS (Japan) constellations with on-module support for future L-band signals. The mosaic-X5 module is a *very* sophisticated chip running a full internal web page server; where its position can be monitored and fully configured through that web page using a standard browser. It can be configured as a Real-Time Kinematic (RTK) Base Station to provide corrections to other RTK Rovers, or as an RTK Rover, where it can achieve a horizontal positioning accuracy of 6 millimeters (0.6cm + 0.5ppm). For research applications, the mosaic-X5 can deliver position updates at 100Hz (100 times per second), log raw GNSS data, and monitor/mitigate signal jamming or spoofing.

Under the hood, the SXM-E is powered by the mosaic-X5 GNSS module from Septentrio; and the Espressif ESP32-WROVER processor (16MB flash, 8MB PSRAM). The mosaic-X5 has USB-C connectivity (with Ethernet-over-USB), multiple UARTs, and supports full Ethernet connectivity. The only interface it doesn't offer is WiFi and that's why we've included the ESP32 with its own Ethernet connection. You can connect the mosaic-X5 directly to your Ethernet network - our product supports Power-over-Ethernet too. Or you can link the mosaic-X5 Ethernet port to the ESP32 Ethernet port and have the ESP32 provide WiFi connectivity. In that mode, the ESP32 becomes an Ethernet to WiFi Bridge, seamlessly passing WiFi traffic to and from the mosaic-X5 via Ethernet.

Get Started

Quick Start

A beginner's guide to get your device up and running

Product Overview

A high-level overview of the kit, design specifications, and its functionality

Hardware Overview

A detailed overview of the device's interfaces and components

Hardware Assembly

A setup guide and instructions for connecting to the device interfaces

Software Overview

A guide on the software used to configure and operate the device

Basic Configuration

Instructions to configure the device as a base station

Advanced Configurations

Support

Resources for technical support and customer service inquiries

Product Overview

Designed and manufactured in Boulder, Colorado, USA, the SparkPNT SXM-E is the perfect solution for your high-precision positioning and navigation needs. Based around the multi-constellation, multi-frequency mosaic-X5 module from Septentrio, this is our most advanced GNSS receiver to date. It supports signals from the GPS (USA), GLONASS (Russia), Beidou (China), Galileo (Europe), Navic (India), and QZSS (Japan) constellations with on-module support for future L-band signals. The mosaic-X5 module is a *very* sophisticated chip running a full internal web page server; where its position can be monitored and fully configured through that web page using a standard browser. It can be configured as a Real-Time Kinematic (RTK) Base Station to provide corrections to other RTK Rovers, or as an RTK Rover, where it can achieve a horizontal positioning accuracy of 6 millimeters (0.6cm + 0.5ppm). For research applications, the mosaic-X5 can deliver position updates at 100Hz (100 times per second), log raw GNSS data, and monitor/mitigate signal jamming or spoofing.

Under the hood, the SXM-E is powered by the mosaic-X5 GNSS module from Septentrio; and the Espressif ESP32-WROVER processor (16MB flash, 8MB PSRAM). The mosaic-X5 has USB-C connectivity (with Ethernet-over-USB), multiple UARTs, and supports full Ethernet connectivity. The only interface it doesn't offer is WiFi and that's why we've included the ESP32 with its own Ethernet connection. You can connect the mosaic-X5 directly to your Ethernet network - our product supports Power-over-Ethernet too. Or you can link the mosaic-X5 Ethernet port to the ESP32 Ethernet port and have the ESP32 provide WiFi connectivity. In that mode, the ESP32 becomes an Ethernet to WiFi Bridge, seamlessly passing WiFi traffic to and from the mosaic-X5 via Ethernet.

SparkPNT SXM-E

SparkPNT SXM-E



Housed in an extruded aluminum enclosure, this is the perfect solution for creating your own continuously operating reference station (CORS).

Purchase from SparkPNT

PRODUCT COMPARISON

Check out our product comparison page for a table with the basic features of the SparkPNT SXM-E compared against other mosaic-X5 products.

Parts List

The SXM-E comes with everything you need to get up and running.



Everything that is included in the kit.

- Quick Start Guide
- Cased GNSS Receiver
  - Aluminum Enclosure
  - 10-Way Pluggable Screw Terminal Blocks
  - Qwiic 1.3" OLED Display
- L1/L2/L5 GNSS Surveying Antenna
- Reinforced RG58 TNC-SMA Cable (10m)
- WiFi + Bluetooth RP-SMA Antenna
- 32GB microSD Card (Class 10)
- USB-C Power Supply (5W)
- Reversible USB-C Cable (A to C, 2m)
- Ethernet Cable (CAT-6, 1m)
- Silicone Bumpers - 5x11mm (4 Pack)

Specifications

Below are the full specifications for this device:

Features

- Septentrio mosaic-X5 multi-constellation, multi-frequency GNSS Receiver
  - Highly secure against jamming and spoofing with AIM+ unique interference mitigation technology
  - Update rate up to 100 Hz
  - 448 hardware channels for simultaneous tracking of all visible supported satellite signals:
    - GPS: L1C/A, L1PY, L2C, L2P, L5
    - GLONASS: L1CA, L2CA, L2P, L3 CDMA
    - Galileo: E1, E5a, E5b, E5 AltBoc, E6
    - Beidou: B1I, B1C, B2a, B2b, B2I, B3
    - QZSS: L1C/A, L1 C/B, L2C, L5
    - NavIC: L5
    - SBAS: Egnos, WAAS, GAGAN, MSAS, SDCM (L1, L5)
    - On-module L-band
  - Accuracy:
    - Standalone:

        - Horizontal: 1.2m
        - Vertical: 1.9m
    - RTK:
        - Horizontal: 6mm + 0.5ppm
        - Vertical: 10mm + 1ppm
  - Time to Fix: <45s
  - Operating temperature: -40 to 85 °C
  - USB-C interface (UART and Ethernet-over-USB)
- ESP32-WROVER processor
  - Flash: 16MB
  - PSRAM: 8MB
  - USB-C interface (UART via CH340)
- Enclosure
  - Materials: Extruded Aluminum
  - Three Push Buttons
  - Six LED Indicators
  - Two USB-C Ports
  - microSD Slot for Data Logging
    - Connected directly to the mosaic-X5 for fast data logging
  - OLED display
    - 128x64 pixels
- Dimensions: 180.6 x 101.8 x 41mm (7.1 x 4.0 x 1.6in)
- Weight: 415.15g (0.92lbs)

Connections

- SMA Connections:
  - GNSS Antenna (L1/L2/L5/L-Band) - provides 3.3V for an active antenna
  - WiFi + Bluetooth Antenna (2.4GHz)
- 3.5mm Screw Cage Connections:
  - 9V-36V DC input (isolated)
  - GND
  - mosaic-X5 - 3.3V/5V switchable
    - UART (TX/RX/CTS/RTS): `COM1`
    - Event A/B Inputs
    - PPSO
    - LOG
  - ESP32 - 3.3V/5V switchable

        - UART (TX/RX/CTS/RTS)
            - GPIO: (RX: `32`, TX: `34`, CTS: `35`, RTS: `33`)
        - I^2C (SCL/SDA)
- Ethernet:
    - KSZ8041NLI Ethernet PHY interface
    - 10Base-T / 100Base-TX with auto-negotiate and Auto MDI/MDI-X
- Power Options:
    - USB-C Ports
    - Power-over-Ethernet (PoE)
    - 9V-36V DC input (fully isolated)

Antennas

- GNSS Survey Antenna
    - Frequency Bands: L1, L2, L5, L6
    - Gain: ≥2.5dBi
    - APC (NGS Calibrated: ANTEX, ANTINFO):
        - L1: 48.0mm
        - L2/L5: 35.9mm
        - Average: 42.0mm
    - Dimensions: 150 x 58.4mm (5.9 x 2.3in)
    - Weight: 473g (1.05lbs)
- Whip Antenna
    - WiFi/Bluetooth (SPP and BLE)
    - 2.4GHz

Capabilities

Feature	Enabled	Notes
SBAS	✓	Default
DGNSS Rover	✓	Permitted
DGNSS Base	✓	Permitted
RTK Rover	✓	Permitted

Feature	Enabled	Notes
RTK Base	✓	Permitted
RTCM v2.3	✓	Depends on RTK/DGNSS permissions
RTCM v3.x	✓	Depends on RTK permission
CMR v2.0	✓	Depends on RTK/DGNSS permissions
Time Sync	✗	Not permitted
xPPS Output	✓	Default
Event tracking	✓	Permitted
Logging on disk(s)	✓	Permitted with advanced options
APME	✓	Permitted
RAIM	✓	Permitted
Moving Base	✓	Permitted
Measurement Availability	✓	Available
Interference Mitigation	✓	Permitted
DSK1	✓	
Frequency Sync	✗	Not permitted
PPP	✗	Not permitted
- PPPAtomiChron	✗	
Augmentation Data Service	✗	None
- LBAS2	✗	

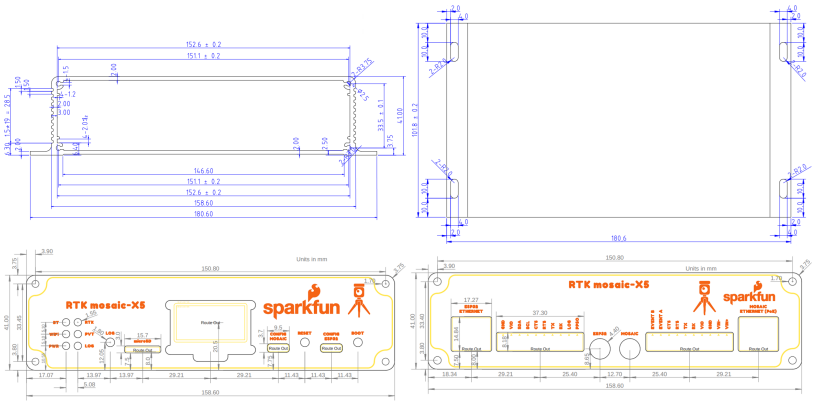
Schematic

Users can download the full schematic for the SXM-E in `*.pdf` format.

Dimensions

Metal Enclosure | **Printed Circuit Board** | **Antenna**

Details about the aluminum enclosure can be found on the Metal Enclosure - Custom Aluminum Extrusion (6in. x 4in. PCB) product page.



Dimensions of the SXM-E aluminum enclosure and the front/rear panels, in millimeters.

How It Works

The Septentrio mosaic-X5 GNSS receiver provides all-in-view satellite tracking; multi-constellation and multi-frequency. When you turn the SXM-E on, it needs to do two things: acquire a GNSS signal and access the internet.

If you connect to the SXM-E's `CONFIG ESP32` USB port and open a terminal emulator / serial console, you will see diagnostic data which tells you the status of the SXM-E. You can find more information on how to do that in the quick start guide and in the software overview.



OLED showing Error Measurements.

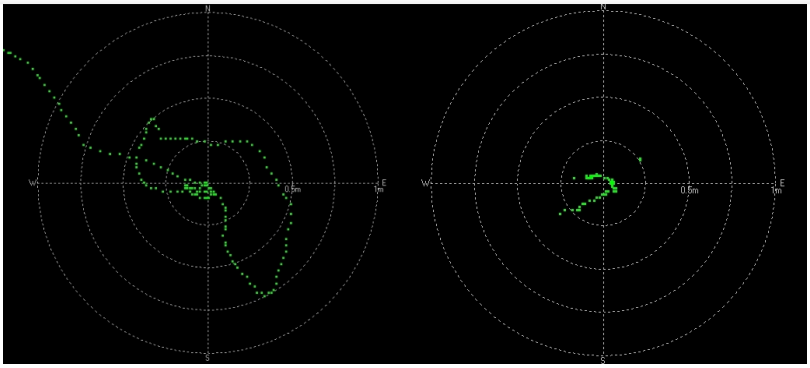
It also features Septentrio's unique AIM+ (Advanced Interference Mitigation and Monitoring) technology for continuous detection and mitigation of GNSS jamming and spoofing.

Suggested Reading

As a more sophisticated product, we will skip over the more fundamental tutorials (i.e. **Ohm's Law** and **What is Electricity?**). However, below are a few tutorials that may help users familiarize themselves with various aspects of the board.



GPS Basics



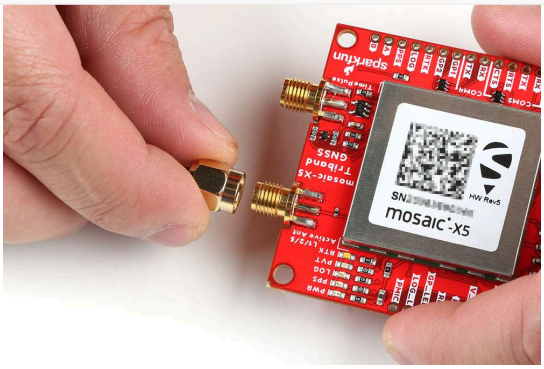
What is GPS RTK?



Setting up a Rover Base RTK System



How to Build a DIY GNSS Reference Station



mosaic-X5 GNSS Breakout Board Hookup Guide

Qwiic OLED 1.3" Hookup Guide

How to Install CH340 Drivers



Serial Terminal Basics



Logic Levels



I²C



Serial Communication



How to Work with Jumper Pads and PCB Traces

RELATED BLOG POSTS

Additionally, users may be interested in these blog post articles on GNSS technologies:



GPS Basics



GPS vs GNSS



What is Correction Data?



Real-Time Kinematics Explained



DIY RTK Surveying



New Video: Unlocking High-Precision RTK Positioning

READ BEFORE HANDLING PCB!

ESD SENSITIVITY

The mosaic-X5 module is sensitive to ESD. Use a proper grounding system to make sure that the working surface and the components are at the same electric potential.

ESD PRECAUTION

As recommended by the manufacturer, we highly encourage users to take the necessary precautions to avoid damaging their module.

- The PCB is not grounded through the enclosure.
- The PCB features ESD protection on the USB-C connectors and ethernet jacks.
- The mosaic-X5 module also has ESD protection from the `ANT_1` antenna input.

iFixit Anti-Static Wrist Strap



Purchase from SparkFun

ESP32 FIRMWARE

We have intentionally kept the ESP32 firmware as simple as possible - supporting only two modes: Ethernet *(**Mode:** `1`)* and WiFi *(**Mode:** `2`)*. The intention is that you can easily develop your own firmware for the SXM-E using the Espressif ESP IDF if the SparkFun firmware does not meet your needs.

You can of course modify the hardware too, should you want to. The design is completely open-source.

PROVISIONING LIMITATIONS

- The ESP32 firmware we provide is only compatible with basic `SSID` and `Password` WiFi authentication.
- The firmware is not compatible with networks that implement other provisioning methods such as a captive portal, a QR code, or Wi-Fi protected setup.

Hardware Overview

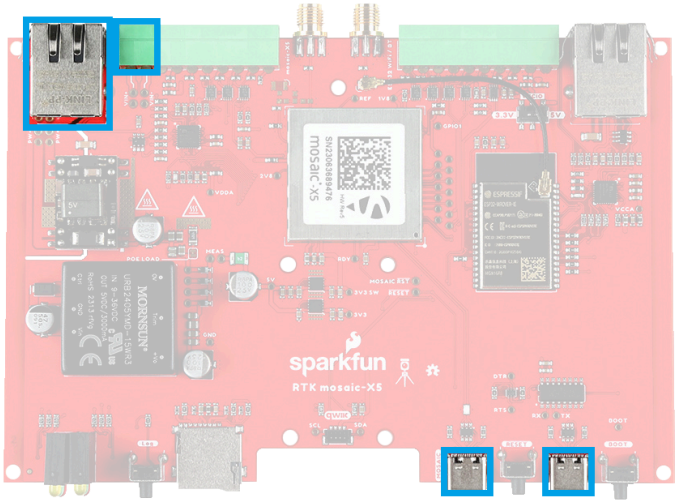
In this section, we walk you through the hardware design, interfaces, I/O connections, power options and more.

INFO

The SXM-E is inspired by the Septentrio mowi (mosaic wireless) open-source design. The SparkFun design builds upon mowi, adding: separate Ethernet ports with mag jacks and Power-over-Ethernet; robust I/O headers with screw cage connections and configurable I/O voltage (3.3V or 5V); µSD data storage; and an OLED display.

Power

The mosaic-X5 and the ESP32 both required 3.3V power. To simplify the power circuitry, the four power sources are combined into a common 5V rail which then feeds individual 3.3V 1A regulators for the mosaic-X5 and the ESP32.



Power connections on the SXM-E PCB.

The SXM-E can be powered individually or in combination, with any of the following:

- **USB Ports** - **5V**; delivered via the `MOSAIC CONFIG` and/or `ESP32 CONFIG` USB-C connectors.
- **Power-over-Ethernet (PoE)** - **Range: 36 to 57V**; delivered via the `MOSAIC ETHERNET` RJ45 MagJack connector.
- **External DC Power** - **Range: 9 to 36V**; delivered via the `VIN+` and `VIN-` screw cage terminals.

MEASURE CURRENT DRAW

If you want to measure the board's current draw, you can open the `MEAS` jumper and measure the current via a pair of breakout pads *(see the **Jumpers** section)*.

PROTECTION COMPONENTS

Diodes are used to combine and protect the power sources from each other. Also, a 2A resettable fuse (green) provides additional protection.

INFO

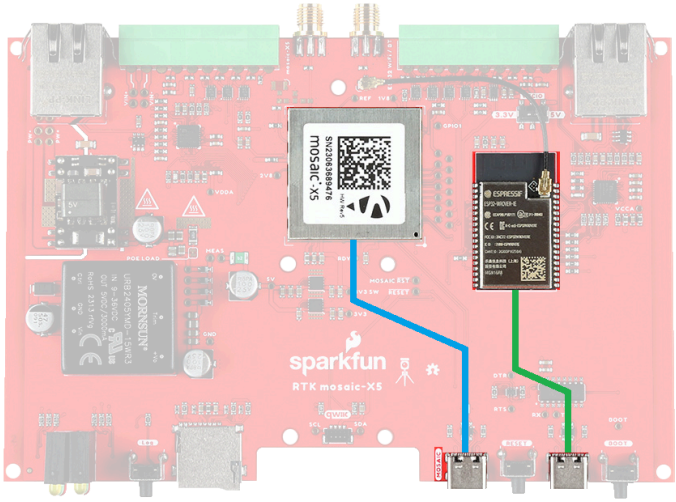
For more details, users can reference the schematic and the datasheets of the individual components on the board.

USB-C Connectors **Power-over-Ethernet (PoE)** **External DC Power**

The mosaic-X5 and ESP32 both have USB-C connections. These USB ports can be used to power the SXM-E during the initial configuration when the mosaic-X5 or ESP32 are connected to a computer.



USB-C connectors on the SXM-E.



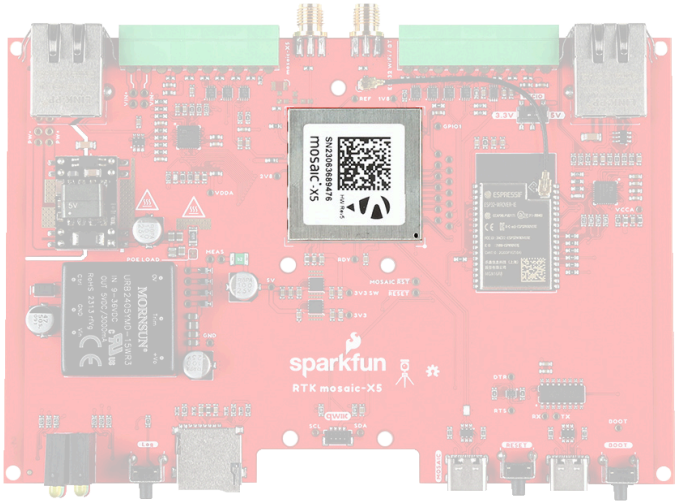
The USB-C device connections on the SXM-E PCB.

CH340 DRIVER

The CH340 allows the ESP32-WROVER to communicate with a computer/host device through the USB-C connection. This allows the ESP32 to show up as a device on the serial (or COM) port of the computer. Users will need to install the latest drivers for the computer to recognize the CH340 *(see **USB Driver** section)*.

mosaic-X5

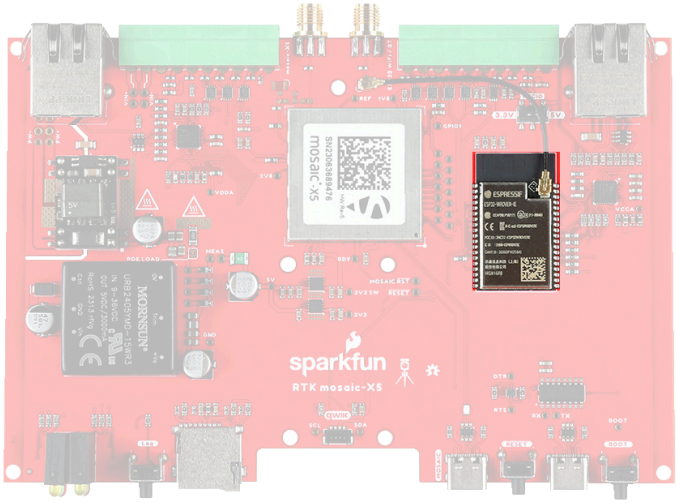
The heart of our product is of course the mosaic-X5 GNSS module from Septentrio. It is a *very* sophisticated chip with multiple interfaces: UARTS, USB and Ethernet. The `COM2` and `COM3` UART pins, plus `GPIO1` and `GPIO2`, are available as 0.1" test points should you need access to them.



The Septentrio mosaic-X5 GNSS module.

ESP32-WROVER

The only interface the mosaic-X5 doesn't offer is WiFi and that's why we've included an Espressif ESP32-WROVER processor (16MB flash, 8MB PSRAM) with its own Ethernet connection. You can connect the mosaic-X5 directly to your Ethernet network - our product supports Power-over-Ethernet too. Or you can link the mosaic-X5 Ethernet port to the ESP32 Ethernet port and have the ESP32 provide WiFi connectivity. In that mode, the ESP32 becomes an Ethernet to WiFi Bridge, seamlessly passing WiFi traffic to and from the mosaic-X5 via Ethernet.



The Espressif ESP32-WROVER processor.

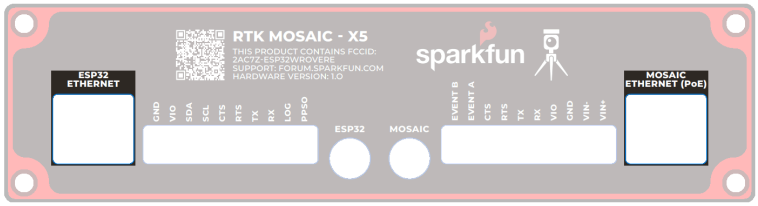
Think of the ESP32 as a co-processor, or riding shotgun... The mosaic-X5 `COM4` UART is linked to the ESP32, allowing the two to communicate directly without needing the Ethernet link. In our firmware, the ESP32 requests NMEA GGA data over this link and then displays it on the I^2C OLED display.

> **⚠ WIFI NETWORK COMPATIBILITY**
>
> The ESP32 is only compatible with 2.4GHz bands and cannot access the 5GHz band.

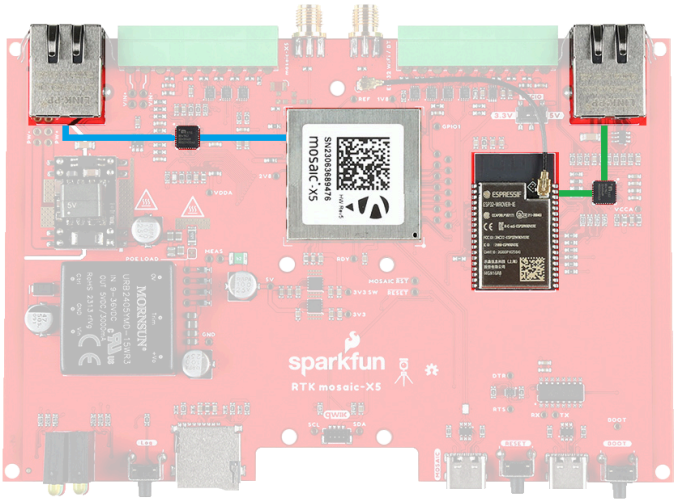
> **ⓘ ESP32 FIRMWARE**
>
> We have intentionally kept the ESP32 firmware as simple as possible - supporting only two modes: Ethernet *(**Mode:** `1`)* and WiFi *(**Mode:** `2`)*. The intention is that you can easily develop your own firmware for the SXM-E using the Espressif ESP IDF if the SparkFun firmware does not meet your needs.
>
> You can of course modify the hardware too, should you want to. The design is completely open-source.
>
> > **ⓘ PROVISIONING LIMITATIONS**
> >
> > - The ESP32 firmware we provide is only compatible with basic `SSID` and `Password` WiFi authentication.
> > - The firmware is not compatible with networks that implement other provisioning methods such as a captive portal, a QR code, or Wi-Fi protected setup.

Ethernet PHY Interfaces

The mosaic-X5 and ESP32 have identical KSZ8041NLI Ethernet PHY interfaces, both connected using Reduced Media-Independent Interfaces (RMII). These allow the mosaic-X5 and ESP32 to be linked directly to your Ethernet network or router, or to each other when the ESP32 is acting as an Ethernet-to-WiFi Bridge.



Ethernet connections: ESP32 (left) and mosaic-X5 (right).



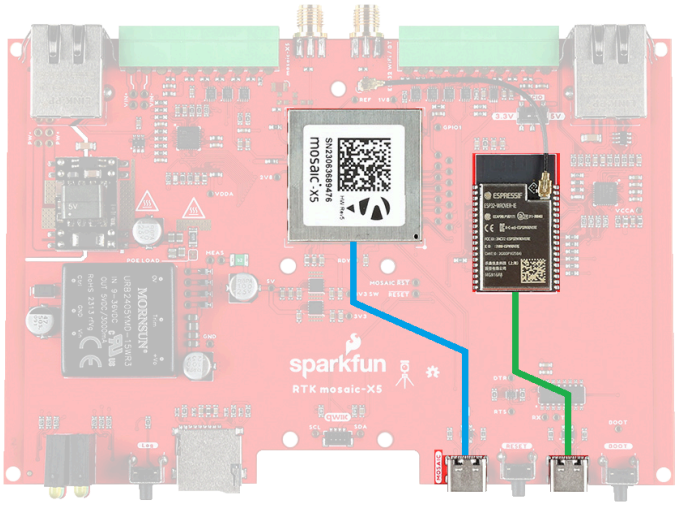
The two Ethernet physical layer interfaces and connections.

USB-C Connectors

The mosaic-X5 and ESP32 both have USB-C connections. The MOSAIC USB port is high-speed and connected to the X5 through a balancing transformer. The ESP32 USB port is connected through a CH340 USB-UART IC.



USB-C connections: mosaic-X5 (left) and ESP32 (right).



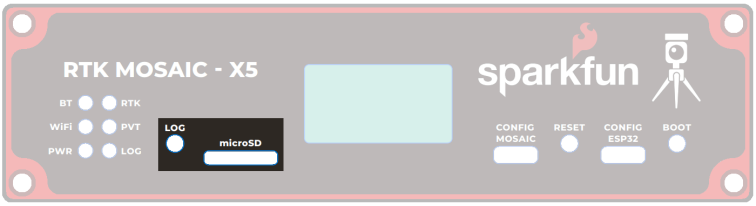
The USB-C data connections on the SXM-E PCB.

> **INFO**
>
> The SXM-E can draw power from either or both USB ports, in addition to Power-over-Ethernet and the DC-DC external input described above.

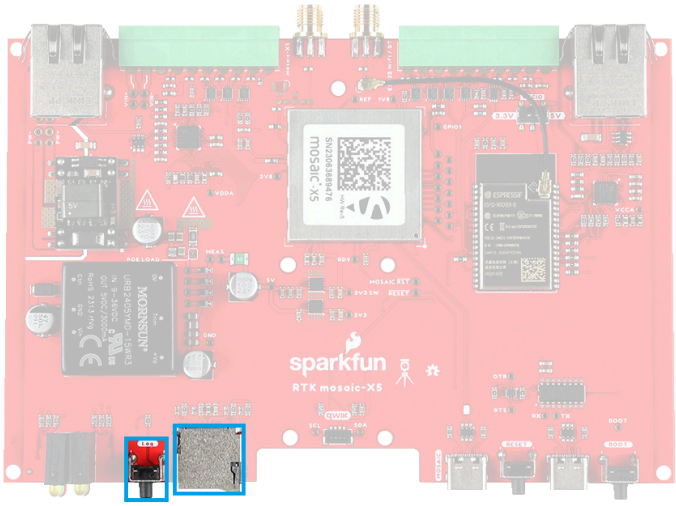
> **CH340 DRIVER**
>
> The CH340 allows the ESP32-WROVER to communicate with a computer/host device through the USB-C connection. This allows the ESP32 to show up as a device on the serial (or COM) port of the computer. Users will need to install the latest drivers for the computer to recognize the CH340 *(see **USB Driver** section)*.

µSD Socket

The µSD socket is connected directly to the mosaic-X5 via a one-bit SDIO interface for fast data logging. The mosaic-X5 supports µSD cards with a **FAT32** file system *(i.e. only cards **up to 32GB** in size)*.



µSD slot and `LOG` button.

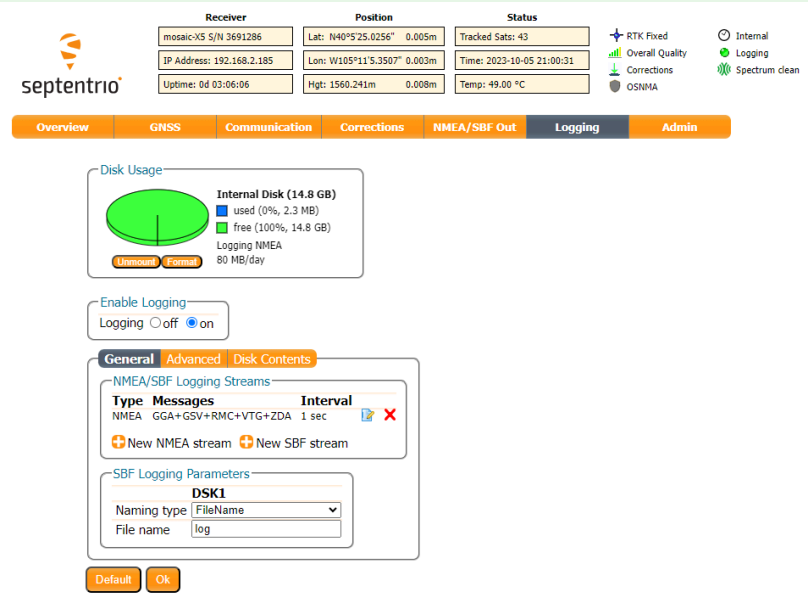


µSD socket and `Log` button on the SXM-E PCB.

OPERATION INSTRUCTIONS

INITIAL CONFIGURATION

Before logging can take place, it is necessary to define a "logging stream" using the **Logging** page or **RxTools**. Streams can contain NMEA or SBF (Septentrio Binary Format) data; SBF can contain RTCM and/or RINEX.



µSD logging stream configuration.

INSTRUCTIONAL VIDEO

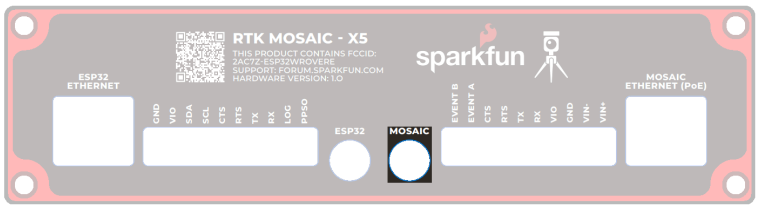
How to log data to the SD card of the Septentrio mosaic receiver module

Once the stream is defined, users can control the data logging operation through the `LOG` button.

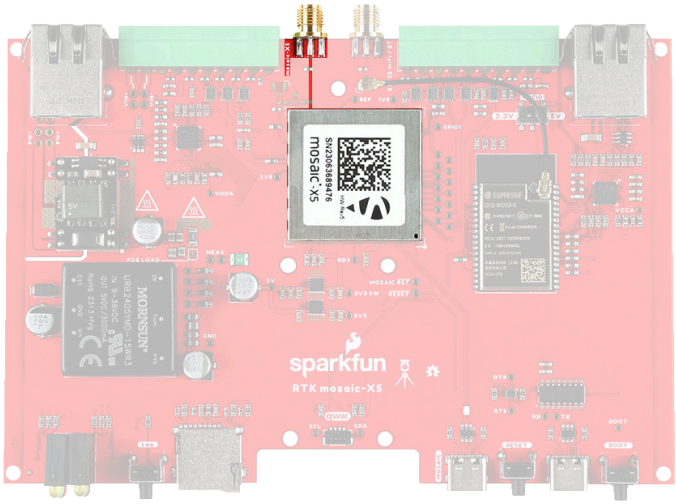
- A short press of the `LOG` button *(< 5s)* toggles data logging to the SD card on and off.
- The red `LOG` LED will flash while logging is taking place.
- A long press, holding the `LOG` button for more than 5 seconds *(> 5s)* and then releasing it, will force the board to:
- Unmount the SD card if it was mounted
- Mount the SD card if it was unmounted

SMA Connectors

The SXM-E has robust SMA connectors for the mosaic-X5 GNSS antenna and the ESP32 WiFi / BT antenna.

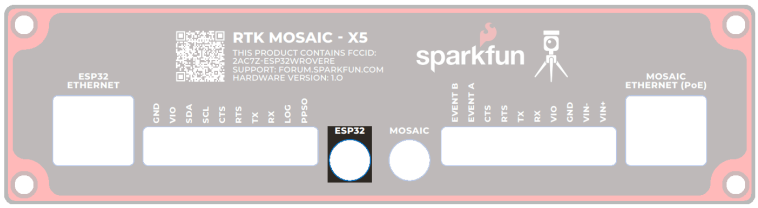


The SMA connector for the GNSS antenna.

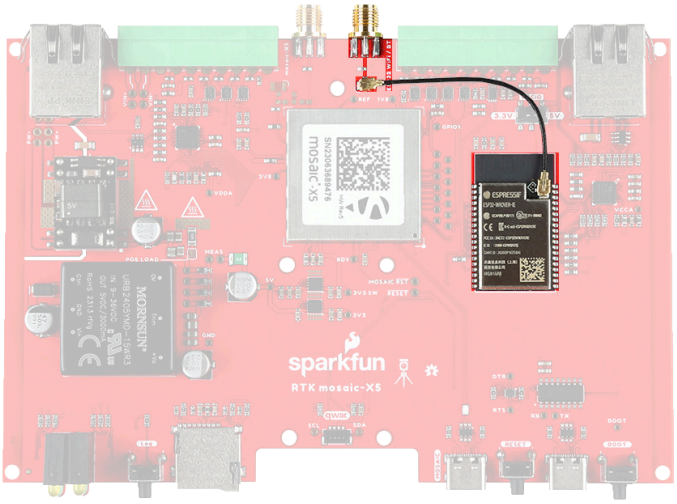


The connection for the GNSS antenna to the mosaic-X5.

The mosaic-X5 SMA connector is standard polarity and provides 5V power for an active antenna.



The RP-SMA connector for the WiFi/BLE antenna.



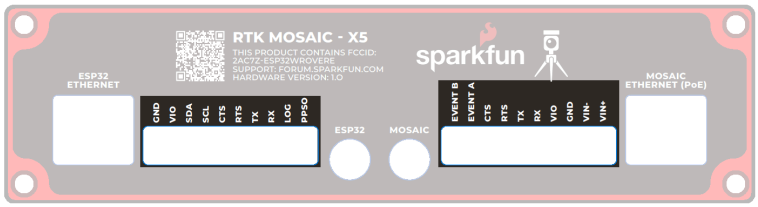
The connection for the WiFi / BLE antenna to the ESP32.

The ESP32 WiFi / BT SMA connector is reverse-polarity (RP). A short u.FL cable connects the SMA connector to the ESP32-WROVER itself.

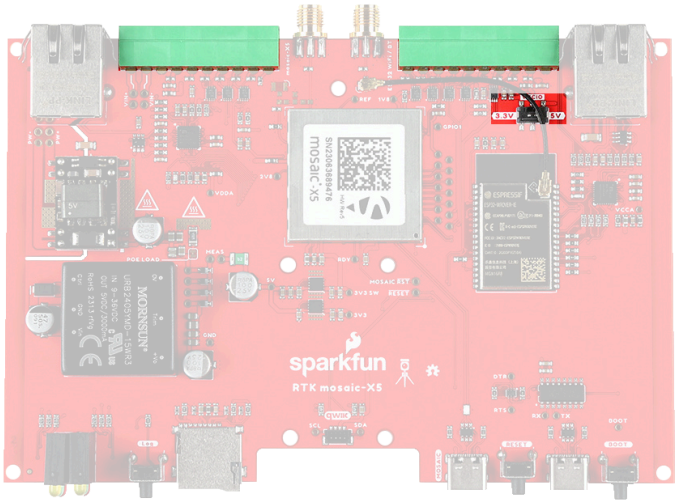
> **CONNECTOR POLARITY**
>
> When selecting antennas and/or cables for the SXM-E, double-check the polarity for the connections.

I/O Terminals

The RTK moasic-X5 is equipped with two 10-way 3.5mm screw cage terminal connectors.



I/O Screw Terminal Connections.



I/O Screw Terminal Connections on the SXM-E PCB.

These terminals are described in the tabs below. For more information on the I/O terminals, you can refer to the schematic.

External Power | **VIO & GND** | **UART - mosaic-X5** | **EVENT A & B** | **PPSO** | **LOG** | **UART - ESP32** | **SDA & SCL**

The `VIN+` and `VIN-` terminals allow the SXM-E to be powered by an external DC power source - typically a 12V / 24V vehicle battery.

Terminal	Function
VIN+	External voltage: **Min: 9V**; **Max: 36V**
VIN-	Ground / Chassis / 0V

> **INFO**
> The DC-DC converter in the SXM-E provides 1.5kV isolation between `VIN+`/`VIN-` and **5V/GND**. There is no direct electrical connection between `VIN-` and `GND`.

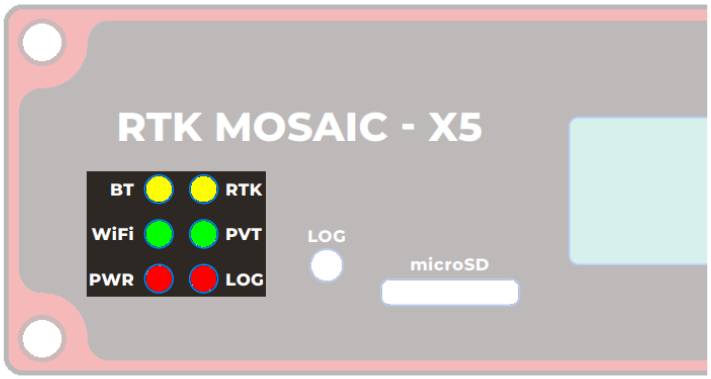
> **GROUND LOOP**
> If desired, users can link `VIN-` to the adjacent `GND` screw cage terminal. However, this will bypass the voltage isolation and could introduce an unwanted ground loop, particularly if the GNSS antenna ground (shield, 0V) is also connected to the chassis.

Status LEDs

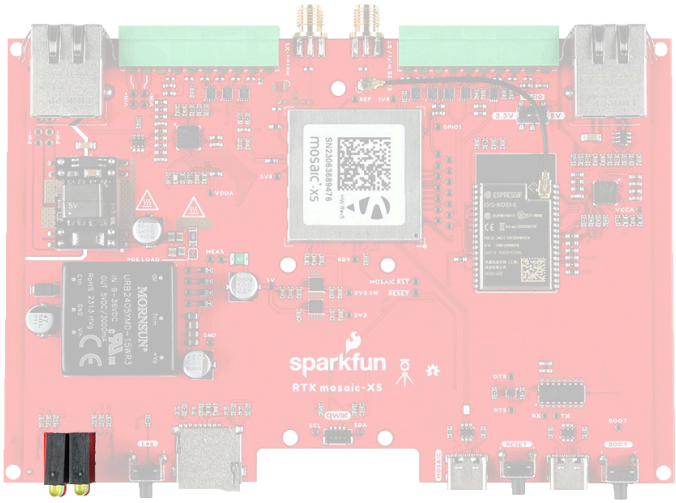
There are six status LEDs on the SXM-E:

- `PWR` - Power *(Red)*
  - Illuminates when power is applied
- `LOG` - µSD Logging *(Red)*
  - Solid Red - µSD card is mounted
  - Blinking Red - Data is being logged
  - Off - µSD is dismounted or not present
- `WiFi` - WiFi *(Green)*
  - The ESP32 firmware is using WiFi

    - Connected to ESP32 GPIO pin 12
- `PVT` - Position Velocity Time *(Green)*
    - Solid Green - The mosaic-X5 has valid Position, Velocity and Time
    - Off - Satellite signal not present or acquired
- `BT` - Bluetooth *(Yellow)*
    - The ESP32 firmware is using BT
    - Connected to ESP32 GPIO pin 13
    - *Not used by firmware v1.0.0*
- `RTK` - Real-Time Kinematic *(Yellow)*
    - Solid Yellow - The mosaic-X5 has an RTK Fixed solution
    - Blinking Yellow - The mosaic-X5 has an RTK Float solution
    - Off - No RTK solution



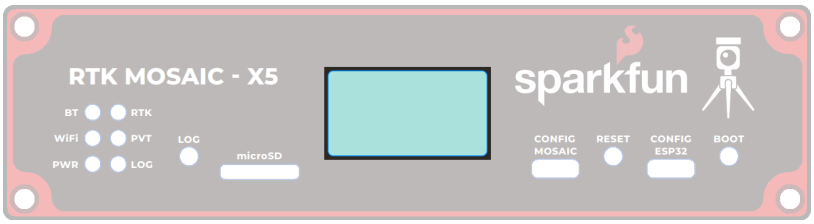
The status indicator LEDs on the SXM-E.



The status indicator LEDs on the SXM-E PCB.

OLED Display

The SXM-E has a 128x64 pixel OLED display, controlled by the ESP32 via I^{2}C. After some initial diagnostic messages, the display will show position, time and other data from the mosaic-X5 NMEA **GGA** message, plus the Ethernet / WiFi IP address.

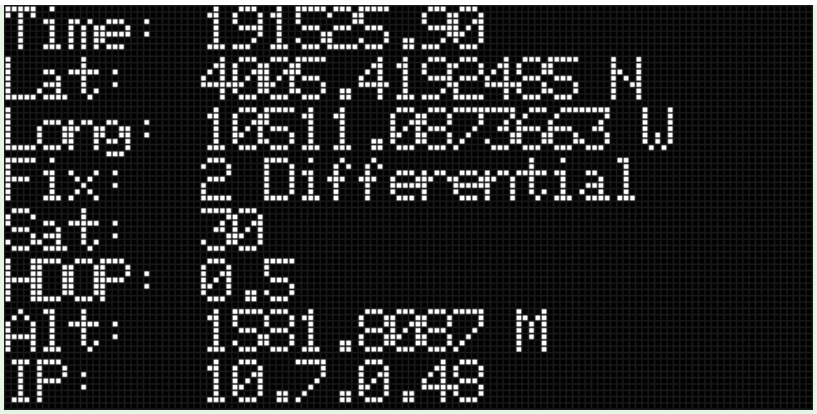


The OLED display on the SXM-E.

- **Time:** - Universal Time Coordinate (UTC) in `HHMMSS.SS` format
  - Note: the display is updated approximately every 2 seconds. The time shown may not be precise.
- **Lat:** - Latitude in `DDMM.MMMMMMM` `N`/`S` format (this is the format used by the GGA message)
- **Long:** - Longitude in `DDDMM.MMMMMMM` `E`/`W` format (this is the format used by the GGA message)
- **Fix:** - the GNSS fix type
- **Sat:** - the number of satellites used in the navigation solution
  - Note: this is not the same as "Satellites In View"
- **HDOP:** - Horizontal Dilution Of Precision
  - Note: this is a dimensionless number indicating the horizontal position accuracy
- **Alt:** - the Altitude in meters
  - Note: this is the altitude above Mean Sea Level, not Geoid
- **IP:** - the mosaic-X5 IP address, obtained via Ethernet or WiFi
  - The X5's internal configuration page can be viewed at this address

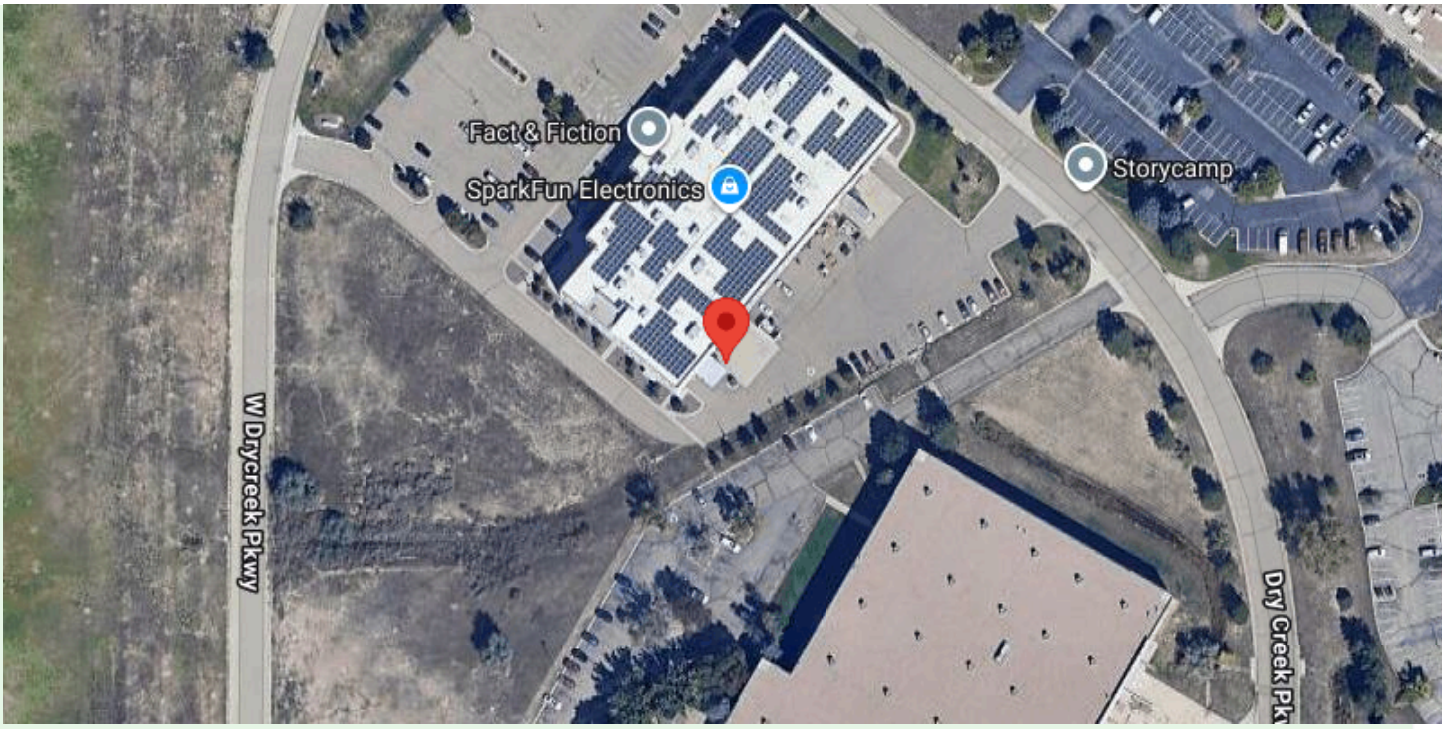
EXAMPLE DATA

Below, is an example data set recorded on the SXM-E at SparkFun HQ; where the antenna was located on the roof of the building. The antenna was above the loading dock, just about where it is displayed on the map. The table, below, is an example of how to convert the information on the display into the time and coordinate values.



Example information on the OLED display of the SXM-E.

	Data	**Format**	**Value**
Time	`191525.90`	`HHMMSS.SS`	7:15:25.90 PM (UTC) 12:15:25.90 PM (MST)
Latitude	`4005.4192485 N`	`DDMM.MMMMMMM` `N`/`S`	40° 5.4162485' N 40° 5' 24.97491" N 40.090270808° N
Longitude	`10511.0873663 W`	`DDDMM.MMMMMMM` `E`/`W`	105° 11.0873663' W 105° 11' 5.241978" W 105.184789438° W



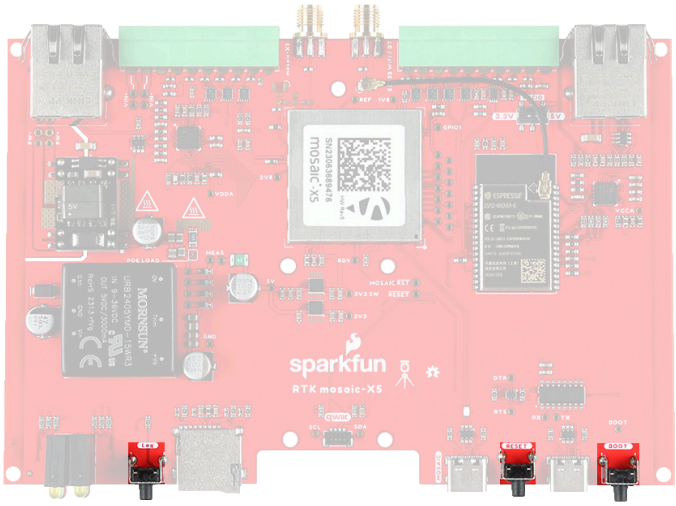
The location of the antenna on Google Maps, as determined from the coordinates on the OLED display. Source: Google Maps

Buttons

There are three buttons on the SXM-E: `RESET`, `BOOT`, and `LOG`.



Buttons on the SXM-E.



Buttons on the SXM-E PCB.

mosaic-X5: Data Logging | **ESP32: Reset** | **ESP32: Boot Control**

Once a logging stream is defined, users can control the data logging operation through the `LOG` button.

- A short press of the `LOG` button *(< 5s)* toggles data logging to the SD card on and off.
  - The red `LOG` LED will flash while logging is taking place.
- Holding the `LOG` button for more than 5 seconds *(> 5s)* and then releasing it, will force the board to:
  - Unmount the SD card if it was mounted
  - Mount the SD card if it was unmounted

INSTRUCTIONAL VIDEO

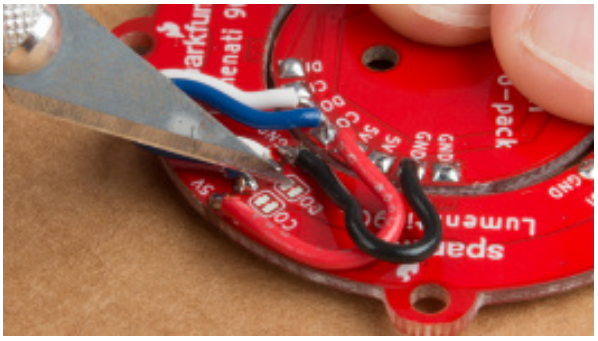
How to log data to the SD card of the Septentrio mosaic receiver module

Jumpers

NEVER MODIFIED A JUMPER BEFORE?

Check out our Jumper Pads and PCB Traces tutorial for a quick introduction!

How to Work with Jumper Pads and PCB Traces

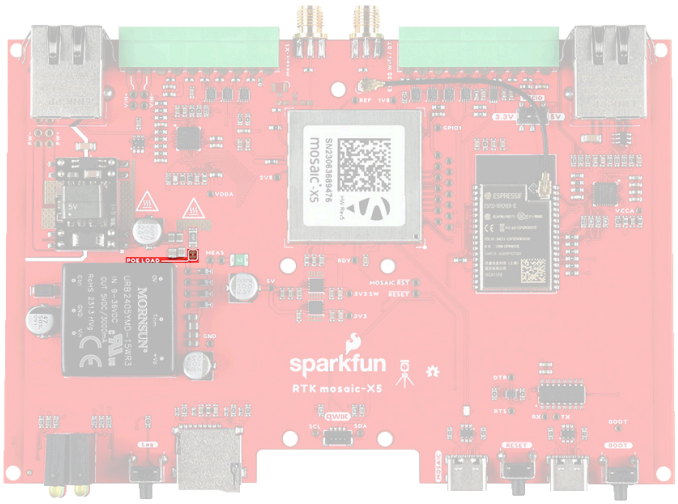


Check out our Jumper Pads and PCB Traces tutorial for a quick introduction!

Go to Tutorial

There are several jumpers on the RTK moasic-X5 PCB which can be used to (e.g.) disable the LEDs or allow measurement of the board's current draw.

Top **Bottom**



The jumpers on the top of the SXM-E PCB.

- **POE LOAD** - This jumper can be used to disconnect the Power-over-Ethernet (PoE) module 50Ω load.
  - The PoE module has a minimum load of 100mA. We included the 50Ω load to ensure this is met. If you can ensure this by other means, open this jumper to disconnect the load.

Hardware Assembly

> **WARNING**
>
> When assembling the SXM-E, users should attach any power connections last. While there shouldn't be any issues with hot-swapping peripherals, it is common practice to power electronics as the last step of the assembly process *(and the power should be disconnected before removing components)*.

What is in the Box?

The SparkPNT SXM-E comes packaged as a complete kit, with all the accessories you'd need to set up an RTK base station or NTRIP server. Inside the kit, users will find the SXM-E, GNSS antenna, and another box containing additional accessories. Inside, the accessory box, users will find the CAT-6 Ethernet cable, USB cable, SMA to TNC cable, USB power supply, WiFi antenna, a pack of silicone bumpers, and 32GB SD card.



Everything that is included with the SXM-E.

USB-C Ports

The USB ports are utilized to configure the mosaic-X5 module and ESP32 WiFi settings. Additionally, the USB ports can also be used as a power source for the SXM-E.

The USB port to the mosaic-X5 can be used to configure the module through an IP port, for serial communication to stream the GNSS data, and access the SD card as a mass storage device. To connect to the mosaic-X5, users only need to plug a USB-C cable into the `CONFIG MOSAIC` USB port and their computer.



The SXM-E with USB-C cable being attached.

With the default firmware, the USB port for the ESP32 is used for serial communication to configure the network settings of the Ethernet-to-WiFi bridge. To configure the network settings for the ESP32, users only need to plug a USB-C cable into the `CONFIG ESP32` USB port and their computer.



The SXM-E with USB-C cable being attached.

SOFTWARE REQUIREMENTS

Depending on their computer's operating system, users may need to install USB drivers to interface with the mosaic-X5 and/or the ESP32. Users may also need to install a terminal emulator for serial communication with the mosaic-X5 and the ESP32.

Antennas

This kit includes two separate antennas, a GNSS antenna to receive the GNSS signals and WiFi antenna for remote access.

CONNECTOR POLARITY

When selecting antennas and/or cables for the SXM-E, double-check the polarity of the connection.

EXTENSION CABLES

Your SXM-E is equally at home on your desk, lab bench or in a server rack. But you're still going to want to put the GNSS antenna outdoors, so it will have the best view of the sky. This kit includes a 10m (~33') cable for the GNSS antenna. For longer distances, we recommend using good quality low-loss RG58 SMA extension cables. The GNSS SMA antenna connection is standard polarity. If you want to extend the ESP32 WiFi / BT antenna connection too, you'll need a Reverse Polarity (RP) cable for that.

SMA Female to SMA Male Cable (10m, RG58)



TIP

Use this extension cable for the GNSS antenna. This cable will not work with the WiFi/BLE antenna due to the polarity of the connectors.

Purchase from SparkFun

RP-SMA Male to RP-SMA Female Cable (10M, RG58)



TIP

Use this extension cable for the WiFi/BLE antenna. This cable will not work with the GNSS antenna due to the polarity of the connectors.

Purchase from SparkFun

GNSS Antenna

In order to receive GNSS signals, users will need a compatible antenna. With the parts included in this kit, connect the L1/L2/L5 (tri-band) GNSS antenna to the SXM-E using the TNC-to-SMA cable.



Attaching a tri-band GPS antenna to the TNC-SMA cable.



Attaching a TNC-SMA cable to the SMA connector on the SXM-E.

MOUNTING LOCATION

Users should mount their GNSS antenna outside, where it will have a clear, unobstructed view of the sky. Avoid areas with nearby buildings, EMF structures (i.e. radio towers or power lines), and vegetation (i.e. trees). These objects can increase errors due to signal muti-path, interference, and elevated noise plane.



The tri-band GPS antenna, mounted outside with an unobstructed view of the sky.

MOUNTING HARDWARE

This kit does not include any mounting hardware for the antenna. To permanently mount the antenna outside, we recommend the following products:

GNSS Antenna Mounting Hardware Kit



Purchase from SparkFun

GNSS Magnetic Antenna Mount



Purchase from SparkPNT

WiFi Antenna

For the WiFi connection, users will need a compatible antenna. Connect the WiFi antenna, included in this kit, to the SXM-E.



Attaching a 2.4GHz WiFi antenna to the RP-SMA connector on the SXM-E.

⚠ WIFI NETWORK COMPATIBILITY

The ESP32 is only compatible with 2.4GHz bands and cannot access the 5GHz band. The ESP32 firmware we provide is only compatible with basic `SSID` and `Password` WiFi authentication.

- The firmware is not compatible with networks that implement other provisioning methods such as a captive portal, a QR code, or Wi-Fi protected setup.

Ethernet Jacks

There are two ethernet jacks on the SXM-E, which can be used to provide network access to the mosaic-X5 module. In addition, one of the ethernet jacks supports power over ethernet (PoE) to power the device.

LAN Cable | **Wireless LAN**

The jack to the mosaic-X5 allows users to provide internet access and power; it supports PoE. To provide network access, users should connect the SXM-E from the `MOSAIC ETHERNET (PoE)` jack to their local area network with the (CAT-6) ethernet cable provided in the kit.

- To power the device, a PoE network switch or PoE injector should be installed in between the network connection to the SXM-E.



The SXM-E with ethernet cable being attached to the `MOSAIC ETHERNET (PoE)` jack.

CONFIGURATION: MOSAIC-X5 SETTINGS

Users can configure the mosaic-X5 module through the network connection (i.e. ethernet or WiFi).

SD Card Slot

An µSD card slot is available for users to log and store data, locally on the board. Users will need to insert a compatible SD card and configure the mosaic-X5 module for data logging.



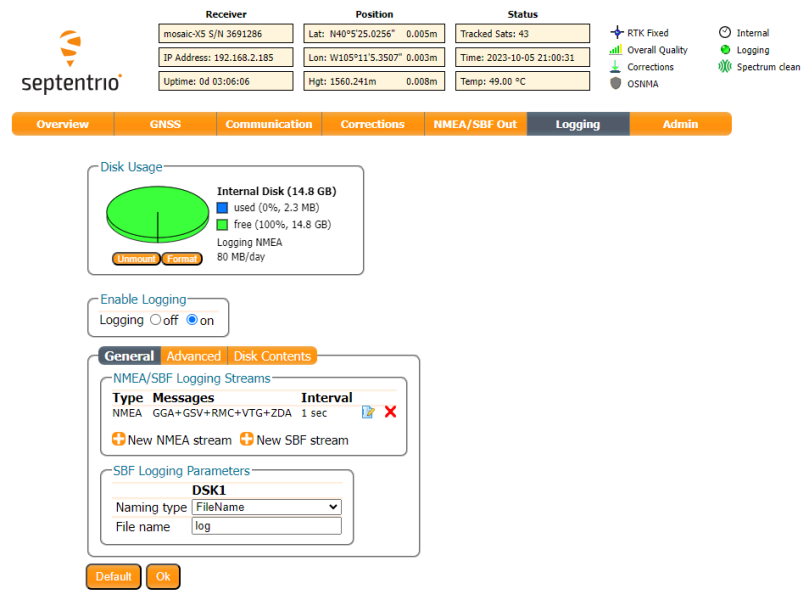
Inserting an SD card into the SXM-E.

SD CARD COMPATIBILITY

The mosaic-X5 supports µSD cards with a **FAT32** file system *(i.e. only cards **up to 32GB** in size)*.

INITIAL CONFIGURATION

Before logging can take place, it is necessary to define a "logging stream" using the **Logging** page or **RxTools**. Streams can contain NMEA or SBF (Septentrio Binary Format) data; SBF can contain RTCM and/or RINEX.



microSD logging stream configuration.

INSTRUCTIONAL VIDEO

How to log data to the SD card of the Septentrio mosaic receiver module

BUTTON OPERATION

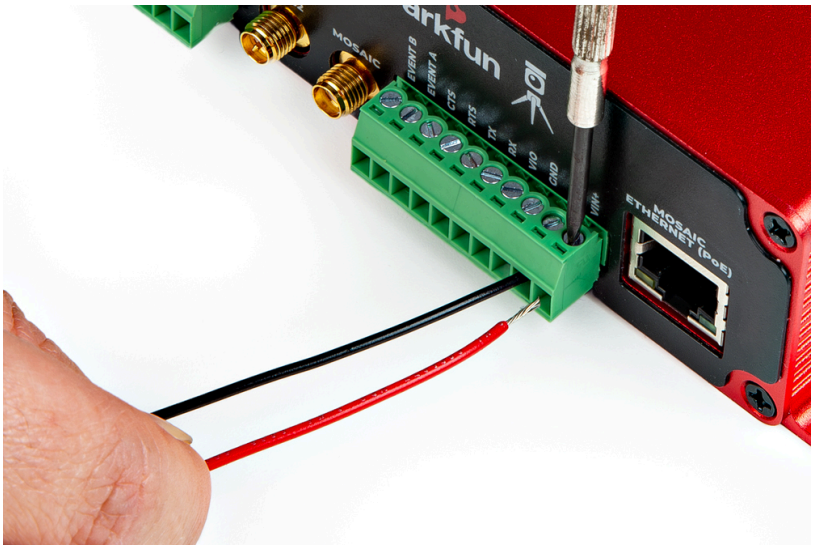
There are multiple ways to configure and enable data logging to an SD card. However, the simplest method is with the `LOG` button. Once the stream is defined,

- Pressing the `LOG` button *(< 5s)* toggles data logging to the SD card on and off.
- Holding the `LOG` button for more than 5 seconds *(> 5s)* and then releasing it, will force the board to:
  - Unmount the SD card if it was mounted
  - Mount the SD card if it was unmounted

For more information, please reference the SD Card Slot section.

IO Terminals

Users can easily attach accessories to the SXM-E by wiring them into the pluggable screw terminal blocks on the back of the enclosure.



Connecting a wire to the screw terminal block.

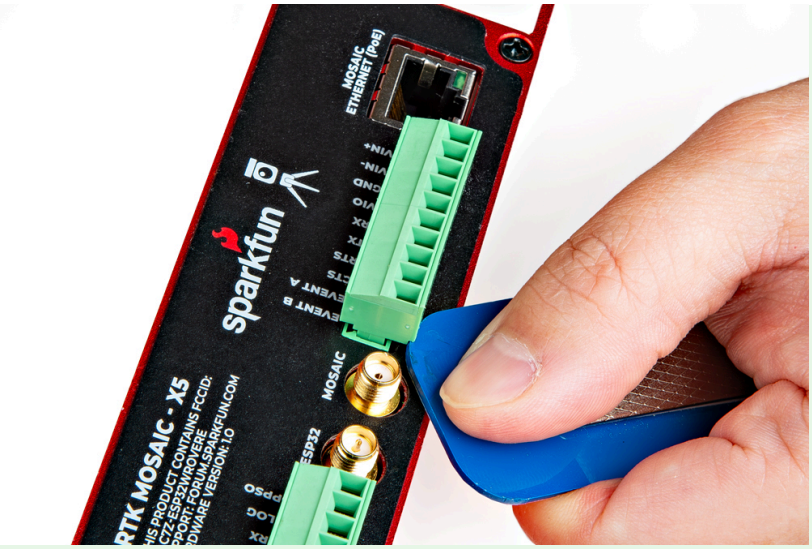
MULTIPLE CONNECTIONS

For multiple connections or wiring harnesses, users can disconnect the pluggable terminal blocks from their sockets on the SXM-E.



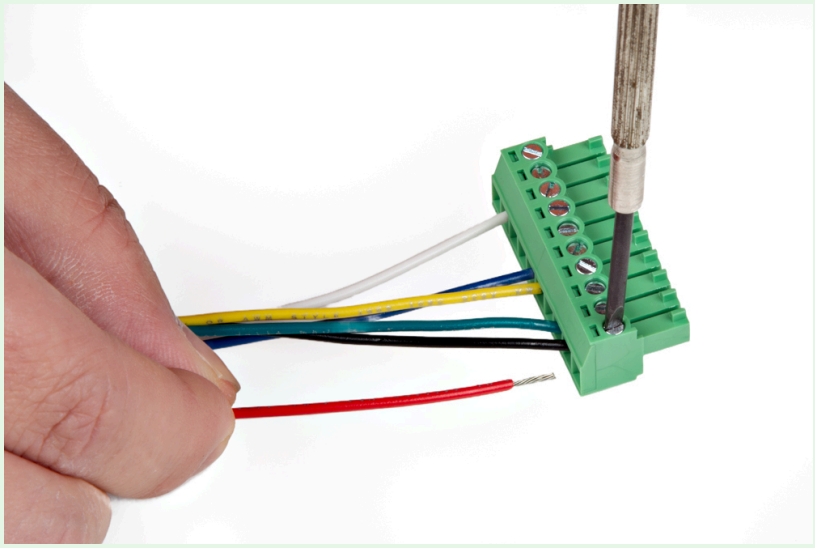
Components of the pluggable screw terminal block.

Users can wiggle or use a soft/rigid object to carefully pry the pluggable terminal block off from its connector. In the picture below, a plastic name tag (~1.5mm thick) is used to carefully pry the terminal block up. We have also found the edge of a PCB ruler works great too.



Using a soft/rigid object to carefully pry the pluggable terminal block free from its socket.

Once wired up, users can simply push the pluggable terminal block back into its socket.



Connecting a wire to the screw terminal block.



Attaching the pluggable terminal block to its socket on the SXM-E.

WARNING

To avoid shorts or damaging the SXM-E, verify the wiring with the labels on the back of the enclosure.

CONNECTING A RADIO

Radio Transceivers

Users can also utilize the terminal blocks to interface with one of our radio transceivers for RTK correction data. We recommend utilizing our breadboard cable to connect those radios to the SXM-E.

SiK Telemetry Radio V3 - 915MHz, 100mW

WRL-19032



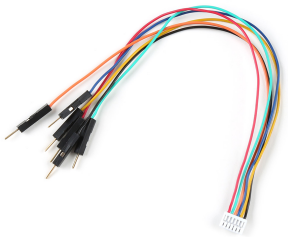
Purchase from SparkFun

SparkFun LoRaSerial Kit - 915MHz (Enclosed)
WRL-20029



Purchase from SparkFun

Breadboard to JST-GHR-06V Cable - 6-Pin x 1.25mm Pitch (For LoRaSerial)
CAB-23353



Purchase from SparkFun

WIRING THE CONNECTIONS

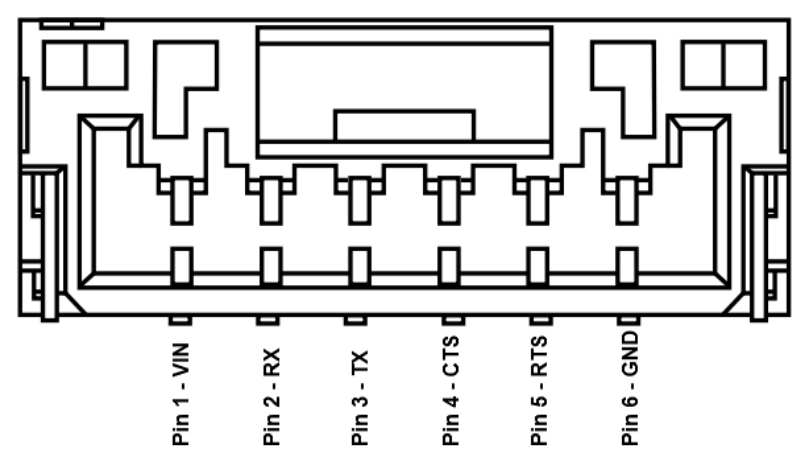
When connecting the SXM-E to one of our radio transceivers, users need to be aware of the pin connections between the products. Although the labels on each device may vary, their pins will function the same *(except for the power input/output pins)*.

WARNING

Please remember that the power output (`VIO`) is preset to **3.3V** by default. When utilizing the SiK telemetry radios, users will need to open the enclosure and move the `VIO` switch.

Radio Pins **Terminal Block**

Below is a diagram of the pin connections for the 6-pin JST GH connector on the radios, which should be also labeled on their enclosure.

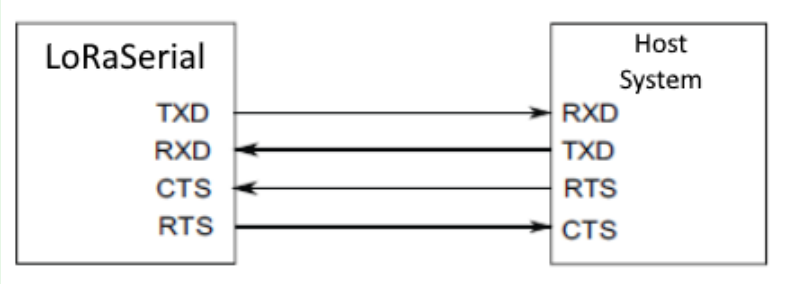


The pin connections for the JST connector on the radios.

Label	`5V`	`RX/RXI`	`TX/TXO`	`CTS`	`RTS`	`GND`
Function	**Voltage Input** • SiK: 5V • LoRaSerial: 3.3 to 5V	UART - Receive	UART - Transmit	Flow Control *Clear-to-Send*	Flow Control *Ready-to-Send*	Ground

When connecting the SXM-E to either of the radios, the wiring connections should follow the table below. If the flow control is not enabled, then only the `RX`, `TX`, and `GND` pins are utilized. As an example, the wiring between a host system *(i.e. SXM-E)* and the LoRaSerial Kit radio is shown in the image below; as documented in the LoRaSerial product manual.

SXM-E	RX	TX	RTS	CTS	GND
Radio	TX	RX	CTS	RTS	GND



Wiring instructions for the LoRaSerial radio.
Source: LoRaSerial product manual

PAIRING RADIOS

By default, the radios in the LoRaSerial Kit - 915MHz are pre-configured for point-to-point communication and a paired with each other. *For instructions on other configurations, please reference the product manual for the LoRaSerial Kit.*

Outdoor Enclosure

The SXM-E comes in a beautiful custom extruded aluminum enclosure, with machined end panels and matching stickers. The slotted flanges make it easy to install and secure the enclosure in many locations. But the enclosure only provides limited protection against the ingress of dust and water; it is IP42.

RECOMMENDED HARDWARE

If you want to permanently install it outside, you'll need a suitable weatherproof box. We recommend the Orbit 57095 - also available from Amazon - back when we put together our very first DIY GNSS Reference Station.

How to Build a DIY GNSS Reference Station



Check out our guide on how to create your own GNSS reference station.

Go to Tutorial

AC NOT REQUIRED!

The Orbit enclosure comes with a built-in power outlet, but you don't need it! The SXM-E can be powered by Power-over-Ethernet (PoE), meaning all you really need to run up to the roof is a standard 8-core CAT-6 Ethernet cable. Choose a PoE Ethernet Switch that meets your needs. We have had good experiences with the TP-Link TL-SG1005P - available from many retailers including Amazon.

Enclosure Disassembly

Due to the ESD sensitivity of the mosaic-X5 module, we don't recommend disassembling the SXM-E. However, if users must access the PCB to troubleshoot an issue, make a modification, or repair a component, we highly recommend that they take the necessary ESD precautions to avoid damaging the mosaic-X5 module.

READ BEFORE HANDLING PCB!

ESD SENSITIVITY

The mosaic-X5 module is sensitive to ESD. Use a proper grounding system to make sure that the working surface and the components are at the same electric potential.

ESD PRECAUTION

As recommended by the manufacturer, we highly encourage users to take the necessary precautions to avoid damaging their module.

- The PCB is not grounded through the enclosure.
- The PCB features ESD protection on the USB-C connectors and ethernet jacks.
- The mosaic-X5 module also has ESD protection from the `ANT_1` antenna input.

iFixit Anti-Static Wrist Strap

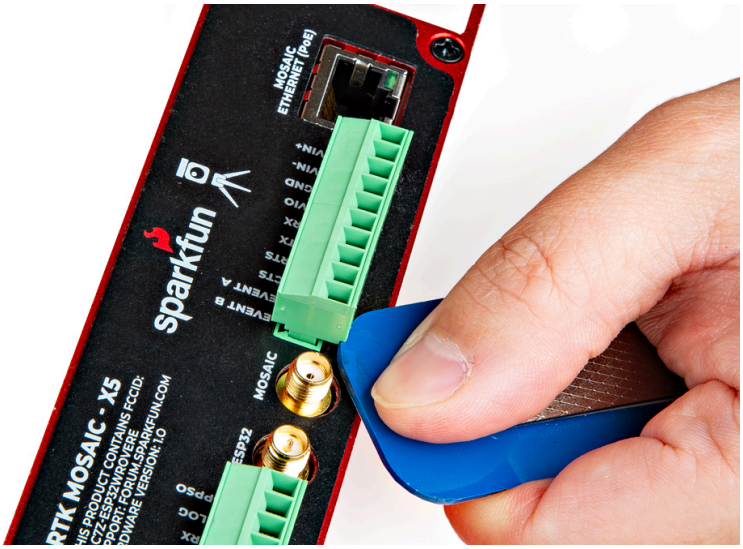


Purchase from SparkFun

Before disassembling the enclosure, users should disconnect the power and all cables from the SXM-E. This will prevent users from creating any shorts and facilitate the teardown process further on.

The internal PCB is only held in place by the front and rear panels of the enclosure. While users can remove the panels first, we recommend that users disconnect the pluggable screw terminal blocks first. Users may find it more difficult to separate the pluggable terminal block from the PCB later in the disassembly process.

1. Users may find it easiest to disconnect the pluggable terminal blocks first; as opposed to later in the teardown process. Users can wiggle or use a soft/rigid object to carefully pry the terminal block off.



Carefully pry the pluggable terminal blocks off.

> **TIP**
>
> In the picture above, a plastic name tag (~1.5mm thick) is used to carefully pry the terminal block up. We have also found the edge of a PCB ruler works great too.

2. Once the pluggable terminal blocks have been removed, users can remove the front and rear panels of the enclosure. They are held in place with four, M3, Phillips head screws on the corners of each panel.

> **TIP**
>
> We recommend removing the front panel first to prevent the Qwiic cable from being yanked off the OLED display or main PCB. Once the front panel is free, carefully lift the panel from the enclosure and disconnect the Qwiic cable from the top connector on the OLED display.

3. At this point, if users have previously disconnected all the cables and the pluggable terminal blocks from the back, the SXM-E PCB should slide out of the enclosure. Users can then, remove the rear panel from the enclosure to complete the teardown process.

To re-assemble the SXM-E, simply reverse the process above.

Software Overview

ESP32 FIRMWARE

We have intentionally kept the ESP32 firmware as simple as possible - supporting only two modes: Ethernet *(**Mode:** `1`)* and WiFi *(**Mode:** `2`)*. The intention is that you can easily develop your own firmware for the SXM-E using the Espressif ESP IDF if the SparkFun firmware does not meet your needs.

You can of course modify the hardware too, should you want to. The design is completely open-source.

PROVISIONING LIMITATIONS

- The ESP32 firmware we provide is only compatible with basic `SSID` and `Password` WiFi authentication.
- The firmware is not compatible with networks that implement other provisioning methods such as a captive portal, a QR code, or Wi-Fi protected setup.

mosaic-X5

WARNING

Beginning with firmware `v4.15.0+`, Septentrio has introduced a mandatory user login requirement to comply with the *EU Radio Equipment Directive (RED) 2014/53/EU, specifically Article 3(3) (d), (e) and (f)* regulations. This update has been gradually applied to all receivers leaving production (beginning August 1, 2025). For more details, please refer to the following articles:

- Cybersecurity guidelines: general
- Cybersecurity guidelines: Log-in procedure

RxTools Software Suite

TIP

Even if you aren't necessarily interested in it, we highly recommend that users install the RXTools software suite before plugging in their board. For Windows PCs, it also includes the

USB driver for the module that enables the Ethernet-over-USB support and virtual `COM` ports.

Users should install the RXTools software suite on their computer to interact with the mosaic-X5 module through the USB interface. The software package includes the USB-IP driver[1] necessary to recognize the board as an ethernet device on Windows PCs. For Linux systems, the standard Linux CDC-ACM driver is suitable.

Download the RxTools Software from Septentrio

NOTE

The following system requirements and installation instructions are from the RxTools *v22.1.0* user manual. This information may change in later iterations of the software suite. Please refer to the user manual *(of the version you are utilizing)* for the most accurate information.

System Requirements

Operating Systems

- Windows 7
- Windows 8
- Windows 10
- Fedora 23 *(or later)* using Qt technology.
- The standalone tools (except `bin2asc`) will run on older distributions.

Hardware

- CPU: 1 GHz processor
- RAM: 1 GB RAM
- Screen Resolution: 1024×768 or higher resolution

INFO

The minimal hardware requirements for a *1Hz update*[2]

Installation Instructions

Windows **Linux**

Users can install RxTools software suite by running the installation executable[3], located in the `RxTools\windows` directory of the downloaded `*.zip` file[4]. During the installation process, users will be notified if a previous version of RxTools is already installed then the previous version will be uninstalled. Next, users will need to provide an installation directory for the RxTools software suite. Users will then select which of the following applications[5] are installed:

> **TIP**
>
> For RxTools v22.1.0, the filename of the installation executable is `RxTools_22_1_0_Installer.exe`.

Once installed, the following applications will be available:

- RxControl
- SBF Converter
- SBF Analyzer
- RxLogger
- RxUpgrade
- RxDownload
- RxPlanner
- Data Link
- RxAssistant
- RxLauncher

USB Driver

Windows | **Linux**

If users haven't already installed the RxTools software suite on their Windows PC, they will need to install the USB driver[1] necessary to recognize and interact with the mosaic-X5 module through the USB interface.

A Windows USB driver for the mosaic-X5 can be installed through two methods:

1. **RxTools Software Suite**
   - The driver is installed during the installation process.

2. **mosaic-X5 GNSS Receiver Module**
    - The installation file for the Windows USB driver will be available from the mass-storage device when the board is initially connected to the computer.

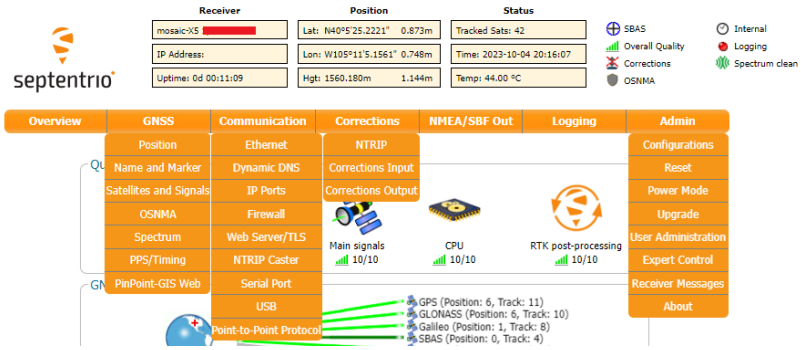
Once installed, the driver emulates two virtual serial ports, which can be accessed as standard `COM` ports to the receiver.

> **HAVING TROUBLE?**
>
> For users who are having trouble locating the USB driver, we have archived a copy of the installation file. Users can download the archived copy[10], by clicking on the button below.
>
> **Download USB Driver *(v3.0.2)***

> **TERMINAL EMULATORS**
>
> Most terminal emulation programs will not make a distinction between virtual or native COM ports. However, for virtual serial ports, the port settings *(i.e. baud rate, etc.)* are not relevant and the default configuration is used in the terminal emulation program. However, the physical/native `COM` ports will have the following default setting:
>
> - Baudrate: 115200bps
> - Data Bits: 8
> - Parity: No
> - Stop Bits: 1
> - Flow Control: None

Web Interface

With the USB driver installed, the mosaic-X5 module supports Ethernet-over-USB. The default IP address allocated for the Ethernet-over-USB interface is `192.168.3.1`. This IP can be entered in any browser to open a connection to the receiver's Web Interface as shown below.



All the drop-down navigation tabs in the web interface.

> **INFO**
>
> The default IP address cannot be changed; this feature is only to be used when a single receiver is connected to your computer.

> **INVALID IP ADDRESS *(WIFI ONLY)***
>
> One of the documentation pages on Septentrio's website, specifies a default IP address of `192.168.20.1` for the web interface. However, that address is for a WiFi enabled product and cannot be used with this product.

Firmware Update

For the latest firmware, please check Septentrio's product page for the mosaic-X5 module. The button below, will redirect users to the resources on the mosaic-X5 product page.

Find the Latest Firmware

GPS Data

Besides logging data with just the `LOG` button, users can configure/enable the output of data streams through the `COM` ports and data logging to the SD card through the web interface or RxTools software suite. The videos below illustrate how these features can be operated through the web interface.

Data Logging

This video illustrates how users can configure the settings for data logging to the SD card.

Stream Data

This video illustrates how users can configure and enable a data stream through a TCP/IP connection.

Related Video

While this instructional video is for a different product line, the information, for the most part, is still relevant.

Sharing Internet Access

By default, the mosaic-X5 GNSS receiver is not configured to access the internet through the USB interface. In order to receive or cast RTK corrections to/from a RTK network, such as NTRIP, users will need to enable capability.

- Users will need to use the web interface or RxTool software suite to enable the `Outgoing Internet Access Over USB` from the **Communication** > **USB** drop-down menu of the navigation tabs.
- This also requires users to allowing Internet sharing through on their computer as well. The procedure to do so depends on your operating system.
  - On a Windows PC, users must enable `Allow other network users to connect through this computer's Internet connection`, through the properties option of the network adapter with

internet access.

- On a Linux computer, users will likely need to configure a network bridge.

> **NEW IP ADDRESS**
>
> Once a network bridge is enabled, the receiver will receive its IP address from the computer's DHCP server. Depending on the routing table, the module may no longer be reachable at its default IP address *(`192.168.3.1`)*.

More Videos

Users can find other instructional videos on Septentrio's YouTube Channel. Feel free to check out their playlists as well:

- Getting Started
- How to *(Videos)*

<details>
<summary>Other Videos</summary>



</details>

How to copy the configurati
Septentrio



How to monitor the CPU of a
Septentrio



How to generate and save a
Septentrio



How to log SBF data for sup
Septentrio

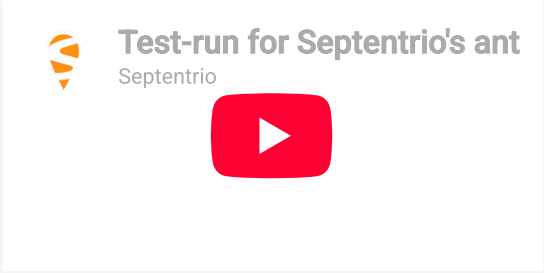


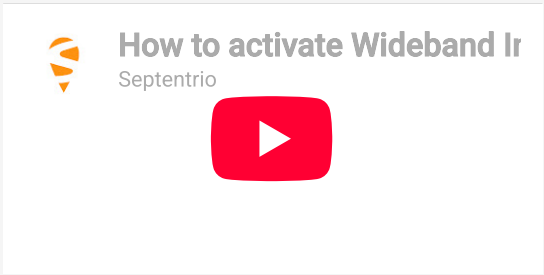
How to set up the PX4 Pixha
Septentrio

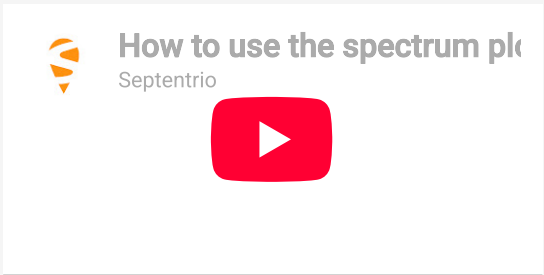


How to set up ArduPilot with
Septentrio









ESP32

USB Driver

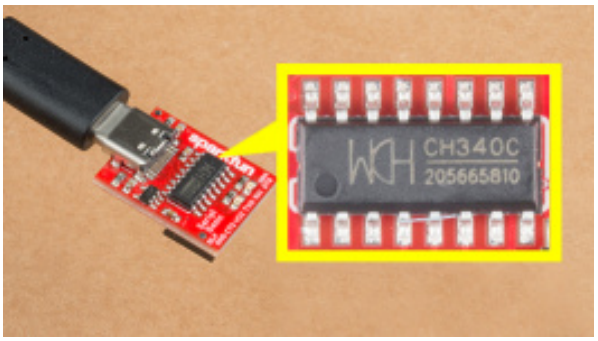
Users will need to install a USB driver for the CH340 serial-to-USB chip, in order to communicate with the ESP32 module. The latest USB drivers for the CH340 are available from the manufacturer, on the WCH website:

Download the latest CH340 USB driver from WCH

NEED DIRECTIONS?

For users having trouble installing the CH340 USB driver, check out our video and hookup guide:

How to Install CH340 Drivers



Go to Tutorial

Terminal Emulator

In order to configure the WiFi settings on the ESP32, users will need to install a serial terminal emulator on their computer.

Windows | **Linux**

For Windows computers, we highly recommend TeraTerm.

NEED DIRECTIONS?

Serial Terminal Basics

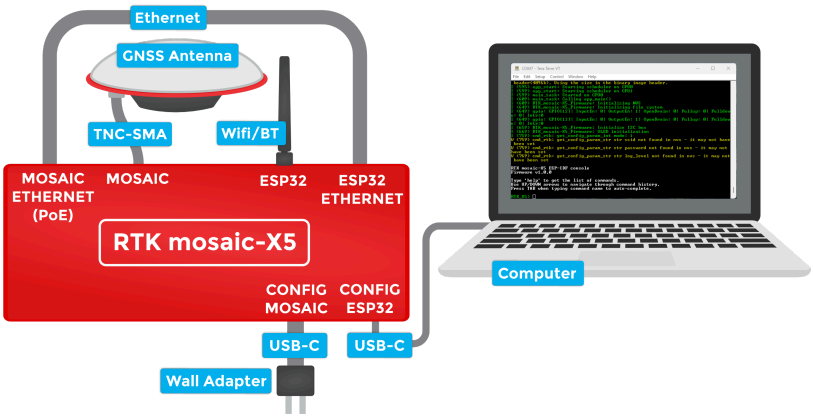


Check out our hookup guide on how to install your favorite terminal emulator.

Go to Tutorial

WiFi Credentials for the Network Bridge

With the CH340 USB driver and a terminal emulator installed, users will now be able to configure the WiFi credentials on the ESP32. In order for the firmware to operate properly, users should have their SXM-E assembled with the network bridge in *WiFi Mode*.



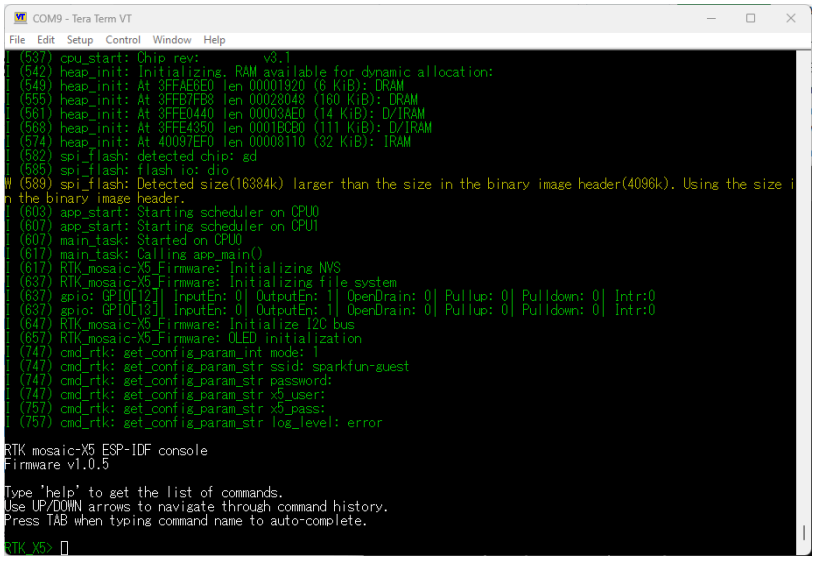
WiFi Mode for the SXM-E.

1 - **Open a Serial Terminal**

- Open the connection to the CH340 using a baud rate of **115200bps**

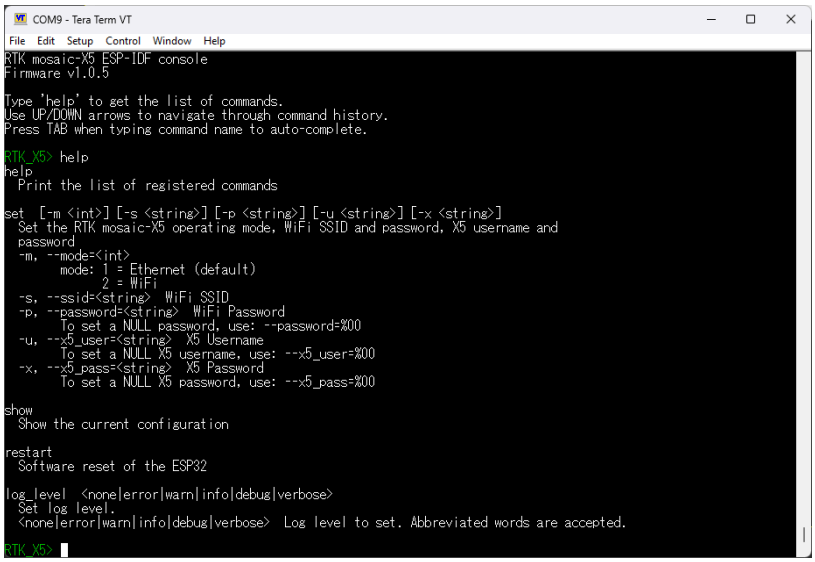
2 - **Put the ESP32 firmware into WiFi mode**

- When you have the Serial Terminal open, you should see the `RTK_X5>` console prompt. If you do not, hit Enter on your keyboard. If needed, click the `RESET` button on the front of the SXM-E to restart the ESP32 firmware.



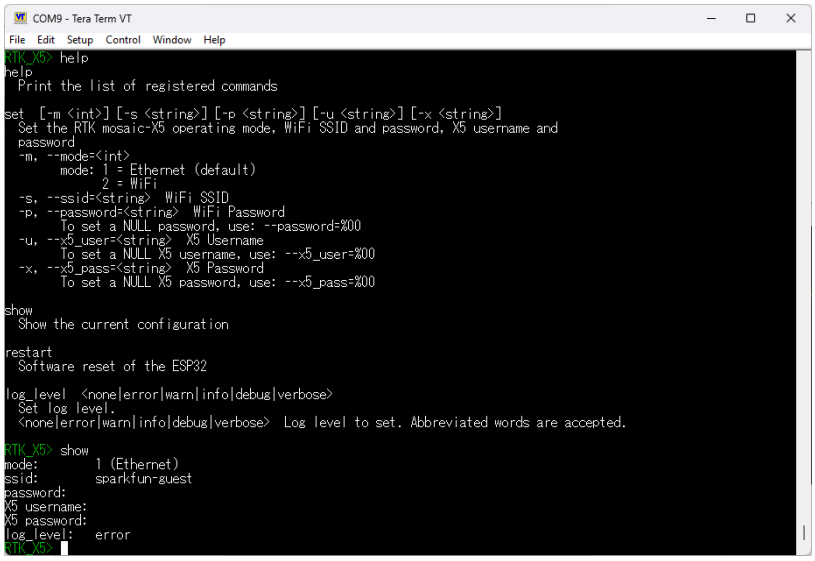
`RTK_X5>` in the console prompt

- Type `help` and hit enter to see the help.



Output from `help` command

- Type `show` to see the current configuration.



Output from `show` command

- By default the firmware will be in **Mode:** `1` (Ethernet). To change the mode to **Mode:** `2` (WiFi), we type:

```
set -m 2
```

> **LONG ARGUMENTS**
>
> The console supports both short and long args, so we could type:
>
> ```
> set --mode=2
> ```

- To set the WiFi SSID, type one of:

```
set -s YOUR_SSID
set --ssid=YOUR_SSID
```

  - Likewise, to set the WiFi password, type one of:

    ```
    set -p YOUR_PASSWORD
    set --password=YOUR_PASSWORD
    ```

  - To save time, you can set all three together with one of:

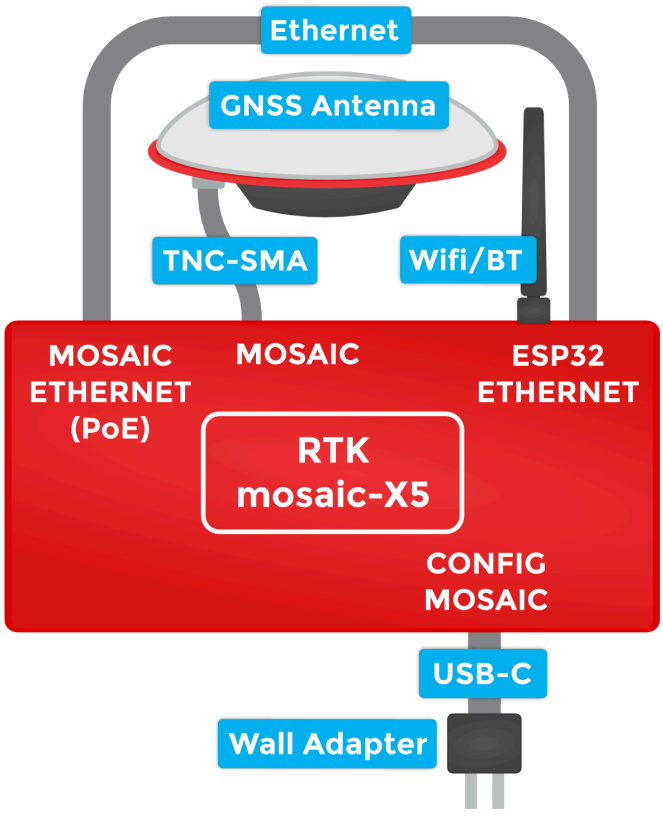
    ```
    set -m 2 -s YOUR_SSID -p YOUR_PASSWORD
    set --mode=2 --ssid=YOUR_SSID --password=YOUR_PASSWORD
    ```

- Finally, type `restart` to restart the firmware with the new settings:

```
restart
```

> **NULL (EMPTY) PASSWORD**
>
> To clear the password - making it NULL / empty - type one of:
>
> ```
> set -p %00
> set --password=%00
> ```

Once the mosaic-X5 has acquired a satellite signal and is connected to the WiFi network, the OLED will display the antenna's position as Latitude (Lat), Longitude (Long) and Altitude (Alt); the WiFi IP (Internet Protocol) network address. The firmware mode, SSID and password are stored in flash (non-volatile) memory. After changing them, you can disconnect the computer and power the SXM-E using the supplied wall adapter.



WiFi connections for the SXM-E.

> **INFO**

> When powering the SXM-E on for the first time, you may see the firmware restart (reboot) several times while it waits for the mosaic-X5 to initialize. This is not an error or anything to be concerned about.

With the SXM-E operating with the configured WiFi network bridge, users should be able to open a web browser on any connected device and navigate to the IP address shown on the OLED display. The browser should be able to access the mosaic-X5's internal web page, where users can configure the mosaic-X5.

ESP32 Firmware - Update

The firmware running on the ESP32 is based heavily on the firmware for the Septentrio mowi. SparkFun added some bells and whistles, primarily to support the OLED display. The full firmware source code is available in the GitHub repo. It was developed and compiled with the Espressif ESP-IDF version 5.1.5.

The most recent version of the firmware is v1.0.5, released on April 22nd 2026. If you purchased your SXM-E before this date, you may enjoy the improvements in v1.0.5:

Changes added at v1.0.5:

- The firmware now supports a username and password for the X5 itself
  - This is to support mosaic-X5 firmware versions >= 4.15.1 where a username and password are mandatory *on IP interfaces* (webUI, Ethernet-over-USB, CLI over TCP/IP)
  - The username and password are *not* mandatory on COM and USB (virtual COM) ports, *unless* you set the `Default Access Level Per Interface` (`setDefaultAccessLevel` / `sdal`) to *none* for COM and/or USB ports
  - The ESP32 firmware only needs to know the username and password if the default access levels have been changed
  - For more information about the X5 Log-in procedure, please read this Septentrio documentation
  - If you upgrade the X5 firmware to 4.15.1:
    - Enter your user-defined username and password, using the Factory credentials *RxAdmin* and *S3pt3ntr10*, as described in the Log-in procedure
    - You will need to enter the user-defined username and password whenever you connect over an IP interface
    - Logging in via the ESP32 on a COM port does not remove the need to log in on IP interfaces. You still need to log in on each IP interface individually and separately
    - Upgrade the ESP32 Firmware to v1.0.5, following the procedure described below
    - If the ESP32 needs to know the username and password: in the ESP32 Serial Terminal / console, use option *set -u* to set the X5 username and option *set -x* to set the X5

password
    - Turn the SXM-E off and back on again

Changes added at v1.0.4:

- The firmware now includes an efficient SBF and NMEA parser
    - Previously the NMEA GPGGA and SBF IPStatus messages were polled, this resulted in the OLED being updated once every ~2 seconds
    - In v1.0.4: NMEA Stream10 carries the GPGGA message at 1Hz; SBF Stream10 carries the IPStatus message OnChange
    - The OLED is now updated on the arrival of the GPGGA message, at exactly 1Hz
- The firmware now ensures that the SBF and NMEA protocols are enabled for output on X5 COM4
    - Previously, the firmware would stall if either SBF or NMEA were disabled
- The firmware now supports the NMEA *GN* Talker ID, in addition to *GP*
    - Previously setting the Talker ID to *GN* would cause the firmware to stall
- The OLED now displays the Latitude and Longitude in *DD MM SS.SSSS* format, to match the format of the X5's internal web page
    - Previously the format was *DDMM.MMMMMM*, copied directly from the GPGGA message
- The firmware will perform a soft reset of the GNSS during startup - if needed
    - The X5 can go into a Ready for SUF download state if it is reset four times with no antenna connected
    - This can be cleared with a soft reset

To upgrade the ESP32 firmware, using a Windows PC:

- Download the complete SXM-E repo as a Zip file:
    - https://github.com/sparkfun/SparkFun_RTK_mosaic-X5/archive/refs/heads/main.zip
- Alternatively, download the latest or a previous release (*Source code zip*) from the repo releases page
- Unzip the Zip file (Extract All)
- Open a *Command Prompt* and navigate into the *Firmware* sub-folder
- Connect the SXM-E `CONFIG ESP32` port to your PC using a USB-C cable
    - It should appear in Device Manager as a CH340 COM port
- Run *ESP32_FLASH_ERASE.bat* to completely clear the ESP32 memory
    - The .bat file should find the COM port number for you
    - If you see the error `'wmic' is not recognized as an internal or external command, operable program or batch file`, you will need to install WMIC or provide the COM port manually:

```
ESP32_FLASH_ERASE.bat COM1
```

- If you need to run `esptool` manually, use the following replacing COM1 with your COM port:

```
esptool.exe --chip esp32 -p COM1 -b 460800 erase_flash
```

- Then run *ESP32_FLASHER.bat* to upload the firmware
  - The .bat file should find the COM port number for you
  - If you see the error `'wmic' is not recognized as an internal or external command, operable program or batch file`, you will need to install WMIC or provide the COM port manually:

```
ESP32_FLASHER.bat COM1
```

- If you need to run `esptool` manually, use:

```
esptool.exe --chip esp32 -p COM1 -b 460800 --before=default_reset --after=hard_reset
write_flash --flash_mode dio --flash_freq 40m --flash_size 4MB 0x1000 RTK_mosaic-
X5_Firmware\build\bootloader\bootloader.bin 0x10000 RTK_mosaic-
X5_Firmware\build\RTK_mosaic-X5_Firmware.bin 0x8000 RTK_mosaic-
X5_Firmware\build\partition_table\partition-table.bin
```

On Linux / MacOS, you can `pip install esptool` and then run the above commnds, replacing `esptool.exe` with `esptool`

ESP32 Firmware - Compiling from source

This is information about how to compile the SXM-E firmware from source. This is for advanced users who would like to modify the functionality of this RTK product.

- How SparkFun does it
- Using Docker

How SparkFun does it

At SparkFun, we use GitHub Actions and a Workflow to compile each release of the SXM-E firmware. We run the build-for-release workflow directly on GitHub. A virtual ubuntu machine installs the Espressif IDF and generates the firmware binary for the ESP32. The updated firmware, bootloader and partition-table .bin files are pushed to the repo. We then create the new release manually.

You are welcome to clone or fork this repo and do the exact same thing yourself. But you may need a paid GitHub Pro account to run the GitHub Actions, especially if you keep your clone / fork private.

The non-release-build workflow builds the firmware binaries and attaches them as an Artifact to the workflow run. Navigate to Actions \ Non-Release Build, select the latest run of Non-Release Build, the binary files are in the Artifact. This may be useful if you want to build the firmware on GitHub but don't want the changes pushed to your repo.

You can then use (e.g.) ESP32_FLASHER.bat to upload the binary onto the ESP32.

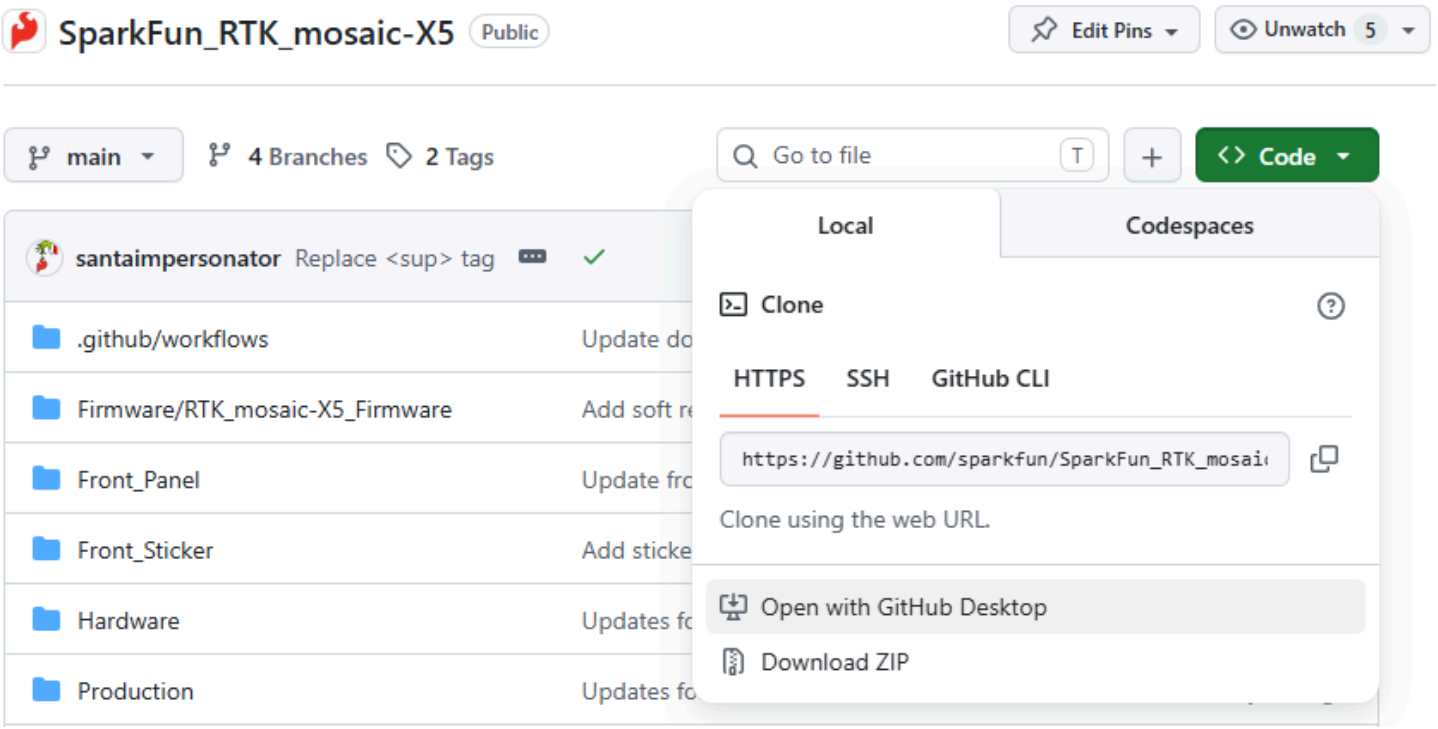
Using Docker

Here is a step-by-step guide for how to install Docker and compile the firmware locally from scratch:

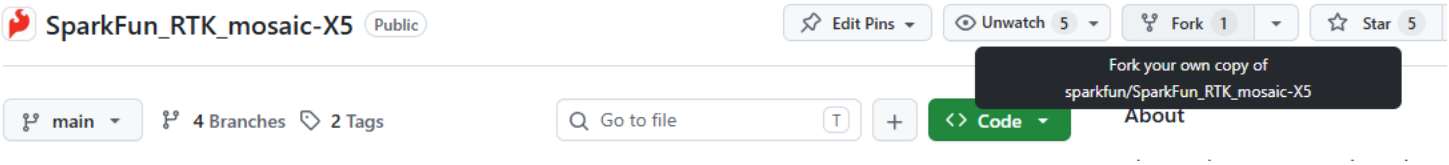
Clone, fork or download the repo

To build the SXM-E firmware, you obviously need a copy of the source code.

If you are familiar with Git and GitHub Desktop, you can clone the SXM-E repo directly into GitHub Desktop:



If you want to *contribute* to RTK Everywhere, and already have a GitHub account, you can Fork the repo:



Clone your fork to your local machine, make changes, and send us a Pull Request. This is exactly what the SparkFun Team do when developing the code. Please use the `release_candidate` branch for any such changes. We are very unlikely to merge anything directly into `main`, unless it is (e.g.) docs corrections or improvements.

If you don't want to do either of those, you can simply Download a Zip copy of the repo instead. You will receive a complete copy as a Zip file. You can do this from the green **Code** button, or click on the icon below to download a copy of the main (released) branch:

Download ZIP

For the real Wild West experience, you can also download a copy of the `release_candidate` code branch. This is where the team is actively changing and testing the code, before it becomes a full release. The code there will *usually* compile and will *usually* work, but we don't guarantee it! We may be part way through implementing some breaking changes at the time of your download...

Download ZIP - release candidate

Install Docker Desktop

- **(Optional)** Head to Docker and create an account. A free "Personal" account will cover occasional compilations of the firmware
- Download and install Docker Desktop - there are versions for Mac, Windows and Linux. You may need to restart to complete the installation.
- Run the Desktop
  - You don't *need* to have an account and you don't *need* to be signed in
  - You only need to be signed in if you want to store or share your Container on Docker Hub
  - If you don't sign in, Docker Desktop will run in Personal mode - which will cover local compilations of the firmware
- On Windows, you may see an error saying "**WSL needs updating** Your version of Windows Subsystem for Linux (WSL) is too old". If you do:
  - Open a command prompt
  - Type `wsl --update` to update WSL. At the time of writing, this installs Windows Subsystem for Linux 2.6.1
  - Restart the Docker Desktop

- If you are using Docker for the first time, the "What is a container?" and "How do I run a container?" demos are useful - *but not essential*
  - On Windows, you may want to give Docker Desktop permission to access to your Network, so it can access (e.g.) HTML ports
  - You can Stop the container and Delete it when you are done
- You may want to prevent Docker from running when your machine starts up
  - Uncheck "Start Docker Desktop when you sign in to your computer" in the Desktop settings

Using Docker to create the firmware binary

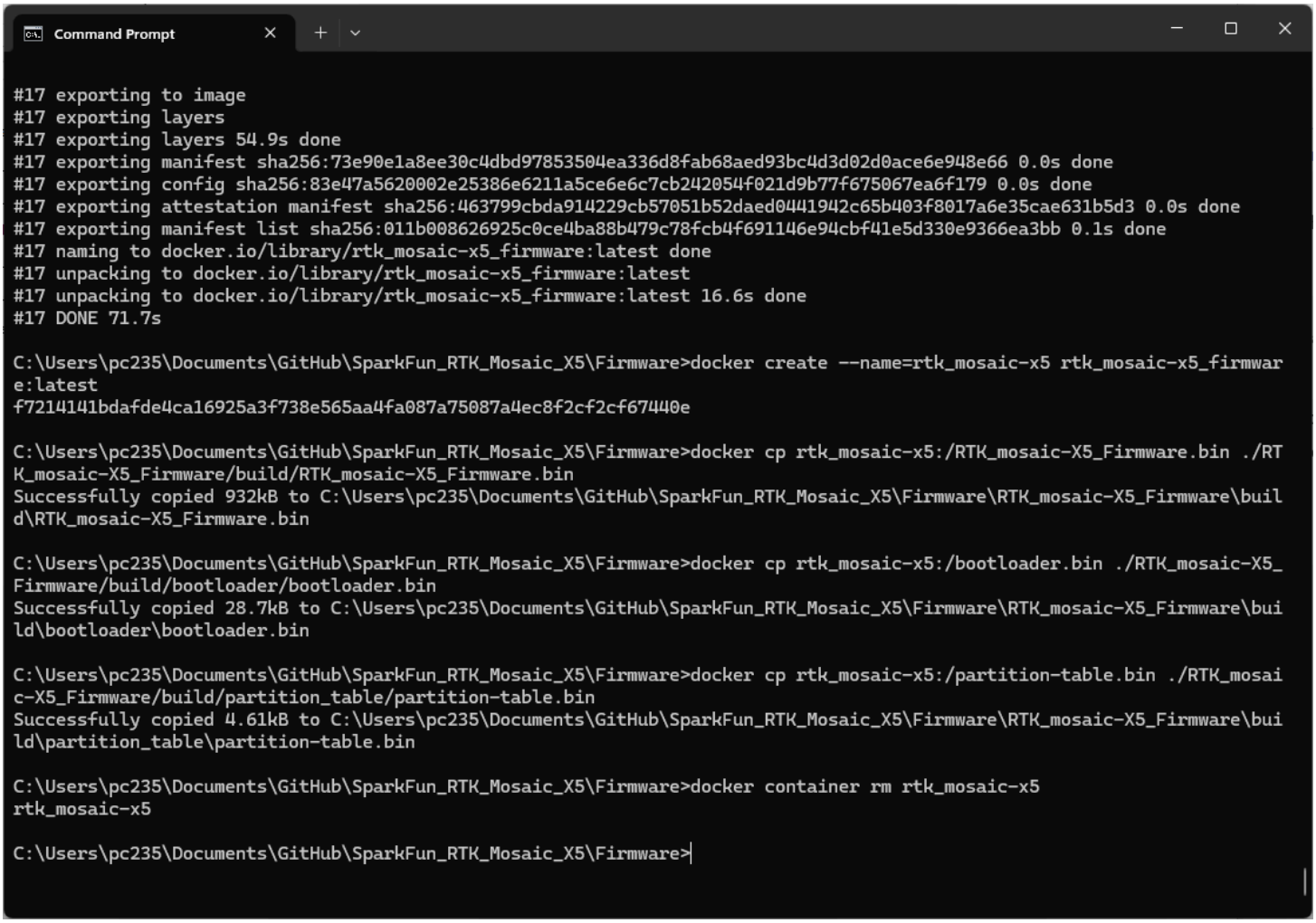
- **Make sure you have Docker Desktop running.** You don't need to be signed in, but it needs to be running.
- Open a Command Prompt and `cd` into the SparkFun_RTK_mosaic-X5 folder
- Check you are in the right place. Type `dir` and hit enter. You should see the following files and folders:

```
.gitattributes
.gitignore
CONTRIBUTING.md
ISSUE_TEMPLATE.md
LICENSE.md
mkdocs.yml
README.md
```

- `cd Firmware` and then `dir` again. You should see:

```
RTK_mosaic-X5_Firmware
compile_run_terminal.bat
compile_with_docker.bat
Dockerfile
ESP32_FLASH_ERASE.bat
ESP32_FLASHER.bat
esptool.exe
```

- The file that does most of the work is the `Dockerfile`
- Run `compile_with_docker.bat`. It does everything for you:



- Hey presto! You have your newly compiled firmware binary!
- The three new files are:
  - **RTK_mosaic-X5_Firmware\build\RTK_mosaic-X5_Firmware.bin**
  - **RTK_mosaic-X5_Firmware\build\bootloader\bootloader.bin**
  - **RTK_mosaic-X5_Firmware\build\partition_table\partition-table.bin**

You can then use (e.g.) ESP32_FLASHER.bat to upload the binary onto the ESP32.

Footnotes

1. On Linux, the standard Linux CDC-ACM driver is suitable. ↩ ↩[2]

2. Higher data rates will require higher CPU speed and more memory capacity. ↩

3. Users will need administrative privileges to install the RxTools software. ↩ ↩[2]

4. Users may need to extract the RxTools installation files from the downloaded, compressed file. ↩ ↩[2]

5. Please see the release notes for the issues and limitations of the RxTools applications. ↩

6. On most Linux operating systems, the `/dev/ttyS*` devices are owned by `root` and belong to the `uucp` group with read/write (`rw`) access. Additionally, the devices are normally locked by writing a file in the `/var/lock/` directory, with the same permissions. ↩

7. Requires c privileges. ↩

8. By default, users will normally have read/write (`rw`) access to the `/dev/ttyS*` serial ports. ↩

9. Changing these permissions also requires `root` privileges. ↩

10. This driver version was archived at the time of initial publication and is not updated. For the latest USB driver from Septentrio, please install their driver through the RxTools software suite. ↩

Basic Configuration

Base Station - Setup

Instructions on configuring the SXM-E as a base station and establishing its position with the different reference poistion types

Base Station - Output Corrections

Instructions on setting up the SXM-E as an NTRIP server to provide RTK correction to the RTK2go NTRIP caster

NTRIP Client - PointOneNav

Instructions on setting up the SXM-E as an NTRIP client with PointOneNav

NTRIP Client - Test Corrections

Instructions to test the corrections from the SXM-E, by setting up an NTRIP client using the RTK2go network

Base Station - Setup

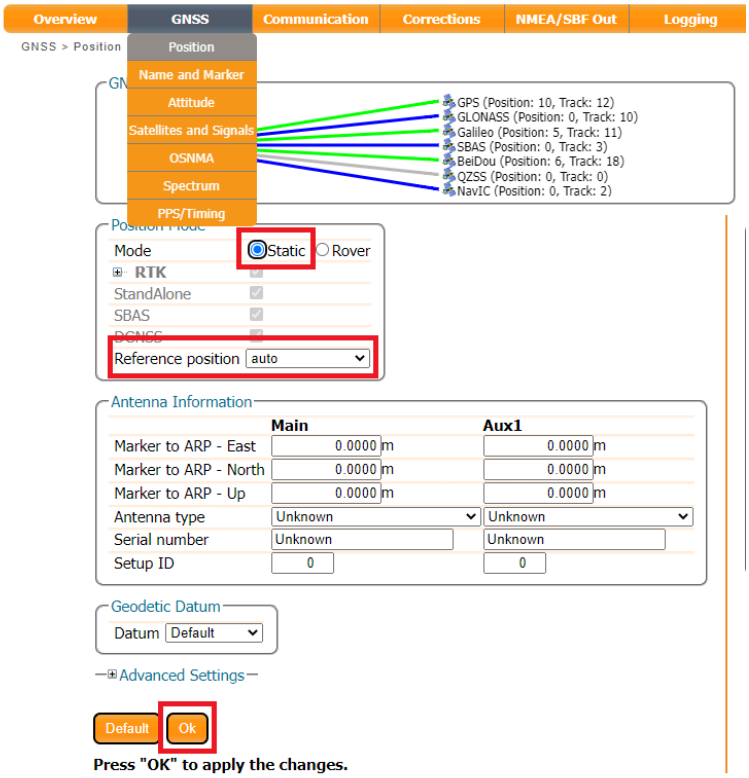
Commercial RTK correction services are great. But there will be times when you want to generate your own correction data so you can share it with your Rover(s) with the shortest baselines. Put simply, an RTK Base knows where it is and - based on its location - can share satellite signal correction data with local Rovers, to allow them to achieve centimeter-level positioning.

For RTK to work, the Base station antenna location is needed to calculate an accurate and reliable position. How do we do that?

In this section, we discuss two different ways of establishing the Base station antenna position: temporary and fixed.

Temporary Base

The mosaic-X5 can be set to "Static" mode and can determine the antenna location (reference position) *automatically*. This is equivalent to the "Survey-In" mode on the u-blox and Quectel GNSS modules. The module will refine its estimate of the antenna position and use that to generate correction data for Rovers.



Selecting the auto reference position (JPG).

> **INFO**
>
> In this mode the module refines its estimate of the antenna position; a new antenna position will be calculated each time the mosaic-X5 is restarted. For this reason, the auto function should only be used to generate a temporary base antenna location.

Fixed Base

The best way to determine the base antenna location is to: log raw GNSS signal data for typically 24 hours, convert it to RINEX format and then submit it to a Precise Point Positioning (PPP) post-processing service such as:

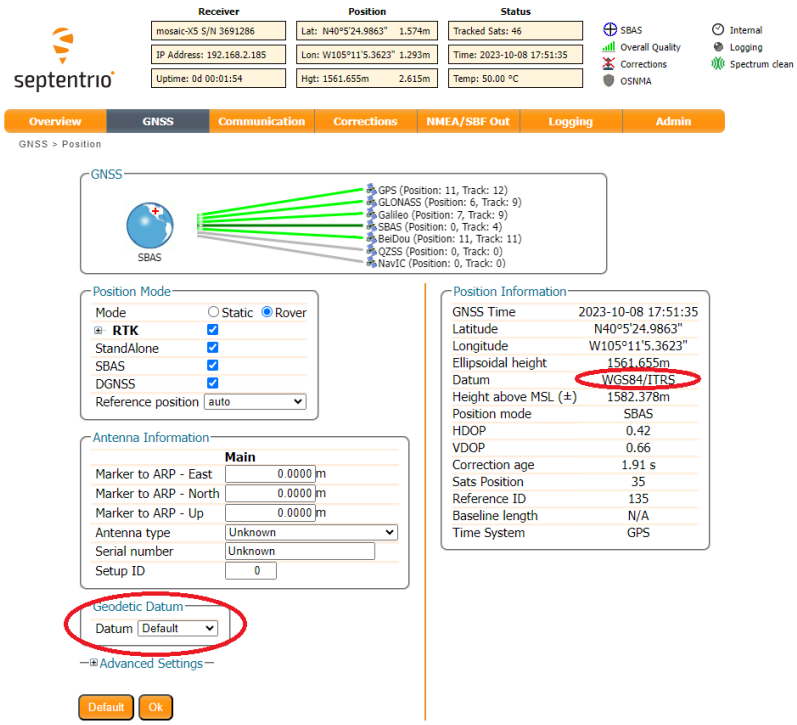
- NRCAN
- OPUS
- APPS

There are some great articles written about PPP. Here we are just covering the essentials. For more information check out:

- Gary Miller’s great PPP HOWTO
- Suelynn Choy, GNSS PPP

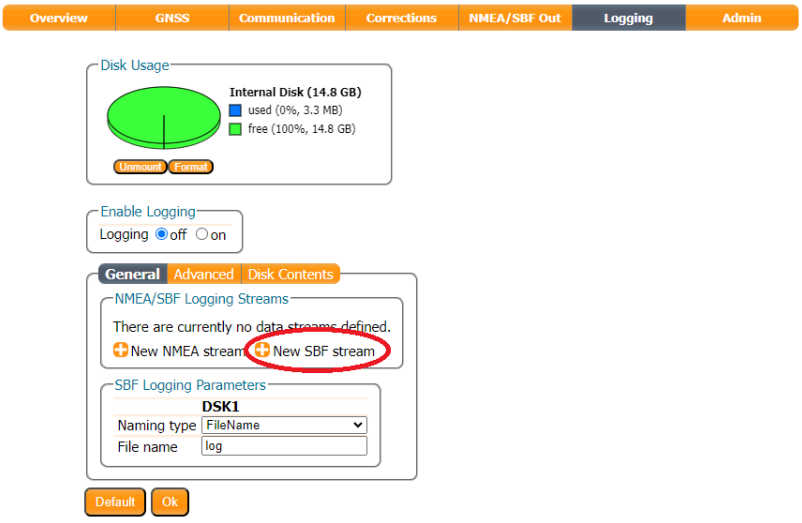
Once the precise antenna position is known, it can be programmed into the module. The corrections the module generates will then be based on that precise, fixed antenna position.

First, let's check what Datum the module is using. It defaults to **WGS84/ITRS**. In North America, it might be better to select **NAD83** but here we'll go with the default.

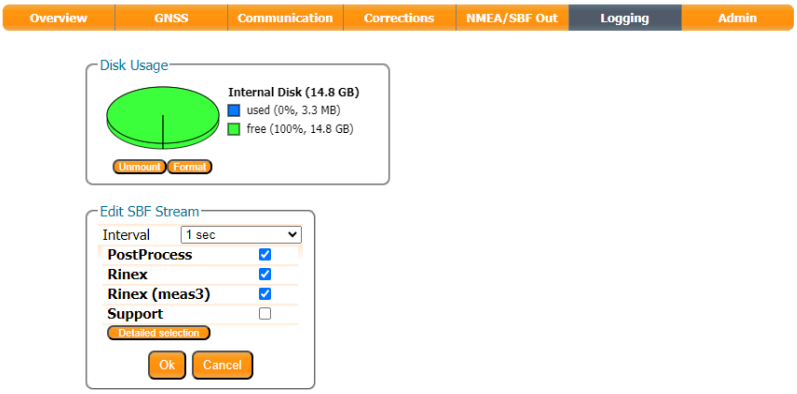


Check the datum

Set up an SBF logging stream to log **PostProcess**, **Rinex**, **Rinex (meas3)** with an interval of 1 sec:

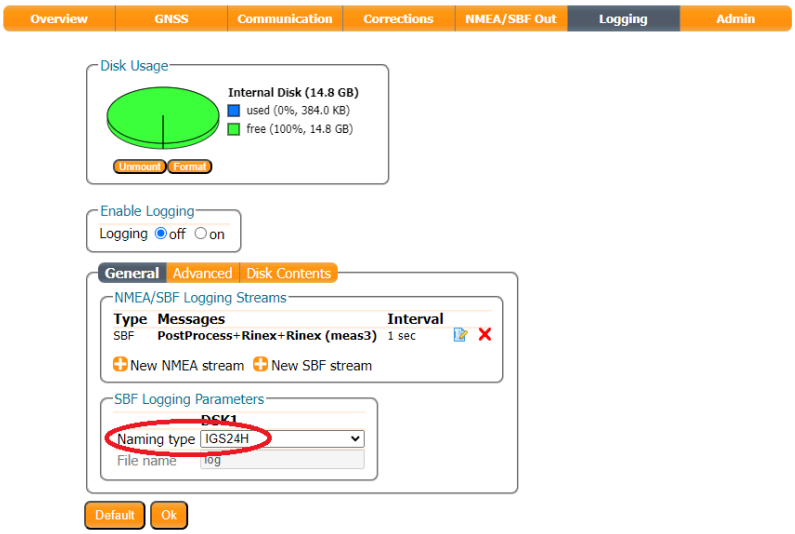


Selecting the logging stream



Selecting the logging stream

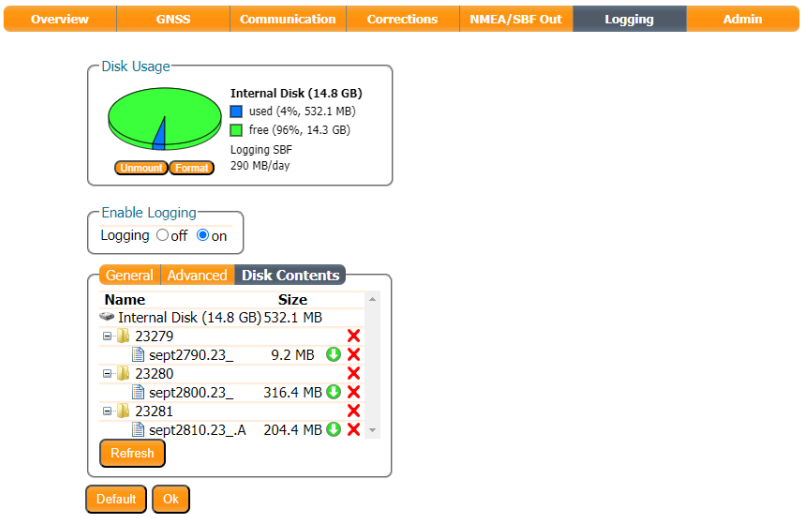
The **IGS24H** Naming Type is useful. When selected, the mosaic-X5 will log data in intervals of 24 hours, opening a new file at UTC midnight.



Selecting the logging stream

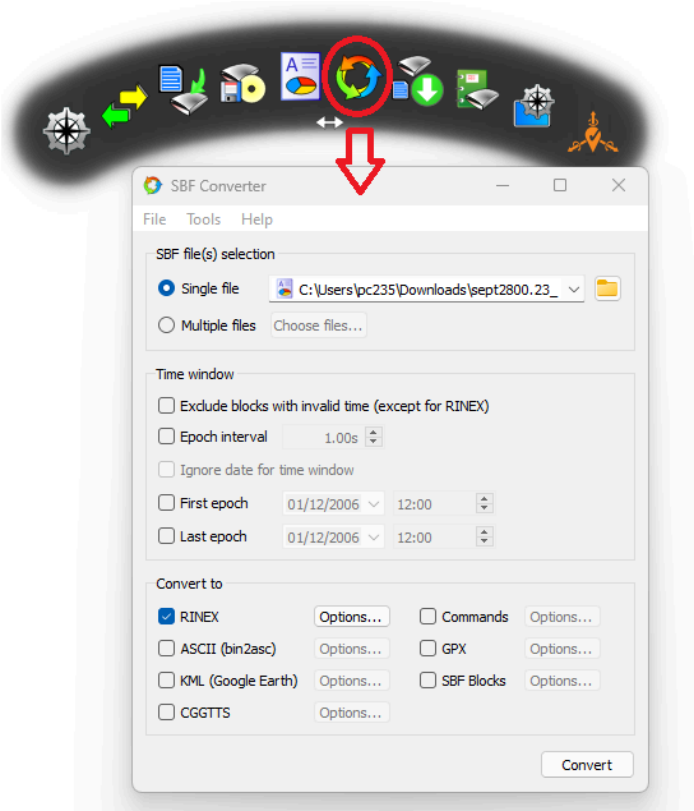
Use the **Enable Logging** radio button to start logging data, or press the **LOG** pushbutton. The red LOG LED will blink while data is being logged.

Use the **Disk Contents** tab to download the SBF data to your computer. Click the green arrow to download an individual file. Or - if the file is large - dismount the disk, eject it and use your computer to copy the files from microSD manually.



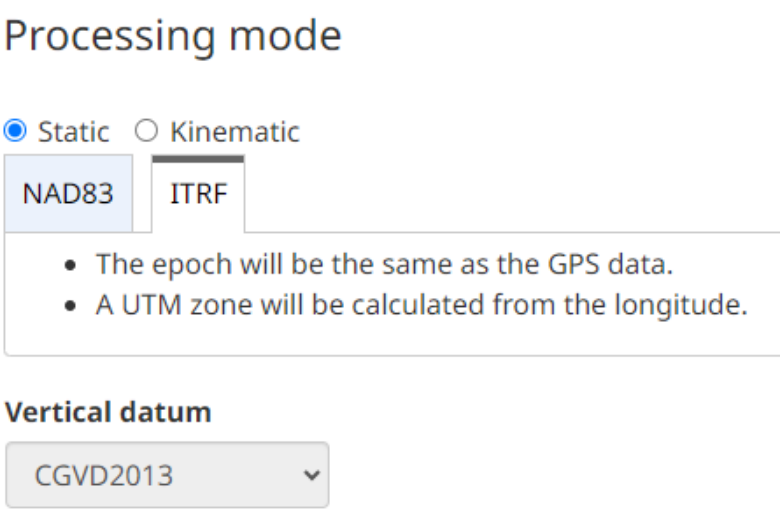
Downloading the logging data

Use the **RxTools \ SBF Converter** utility to convert the data to RINEX format.



Convert to RINEX

Upload the RINEX data to your chosen PPP post-process service. We have found NRCAN is very easy to use and produces excellent results. We select the **ITRF** tab because we are using the WGS84/ITRS datum. (If you are using NAD83, select that tab instead.)



Select the datum for post-processing

After you have uploaded your RINEX data, it only takes a few minutes to receive your antenna position:

The estimated coordinates **ITRF20 2023-10-07** for the **SEPT2800.23O** RINEX file are as follows:

Latitude	N40° 05' 25.0255" ± 0.002 m (95%)
Longitude	W105° 11' 05.3509" ± 0.002 m (95%)
Ellipsoidal Height	1560.187 m ± 0.008 m (95%)

Orthometric Height CGVD2013 CGG2013a 1576.7929 m
[40.09028486,-105.18481970,1560.187]

UTM Zone 13 (North)

Northing	4437794.398 m
Easting	484244.638 m
Scale factor (point)	0.99960306
Scale factor (combined)	0.99935846

[4437794.398,484244.638,1560.187]

Cartesian coordinates

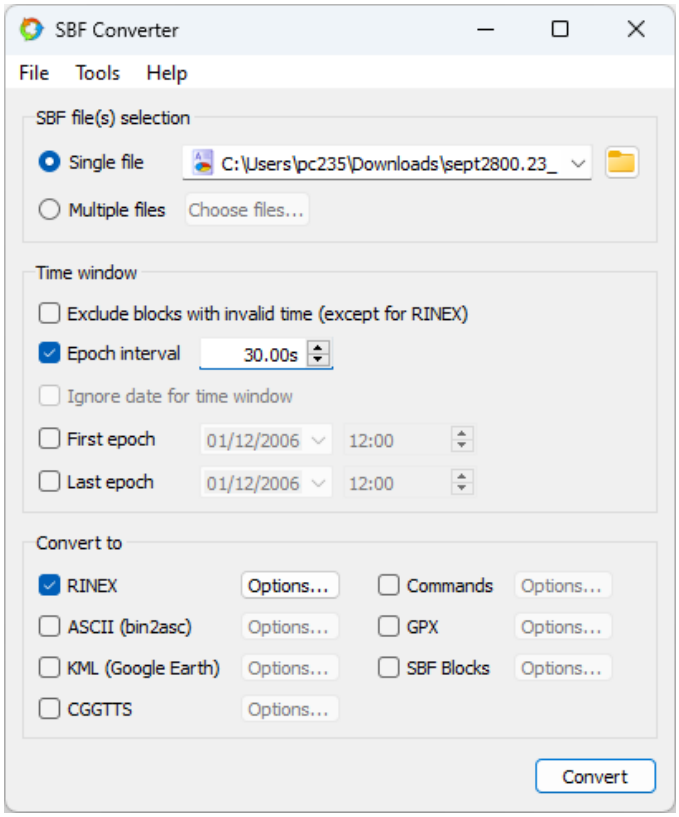
X	-1280187.189 ± 0.003 m (95%)
Y	-4716810.476 ± 0.006 m (95%)
Z	4086664.704 ± 0.005 m (95%)

[-1280187.189,-4716810.476,4086664.704]

Orbits and Clocks Used: **NRCan Ultra-rapid**
GNSS Data: **GPS & GLONASS**
GRS80 ellipsoid used for (x,y,z) to (lat,lon,h) transformation

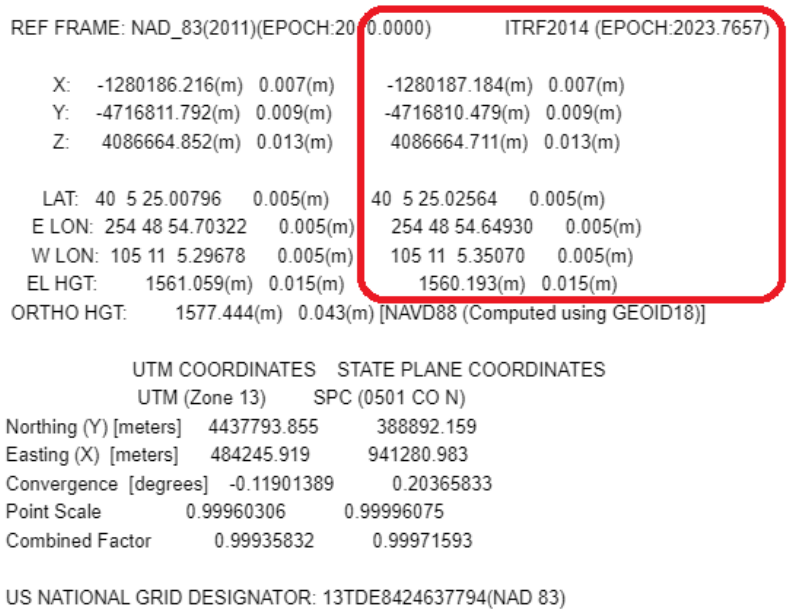
Precise antenna position

Both OPUS and APPS have file size limits. You can shrink the size of the RINEX file by selecting (e.g.) a 30-second Epoch Interval in **SBF Converter**:



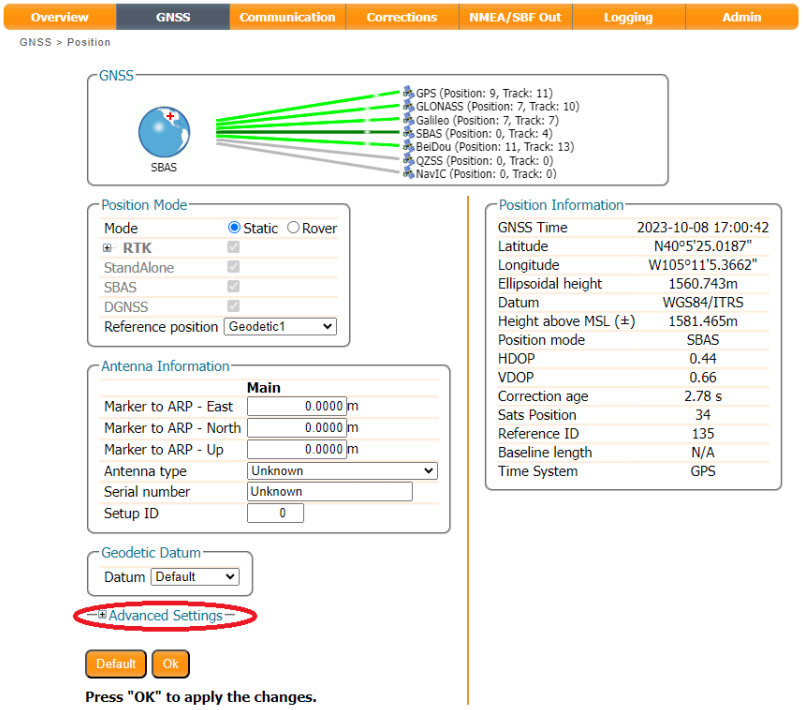
Setting the Epoch interval

Using a smaller, 30-second file OPUS' ITRF results match NRCAN to within a cm:

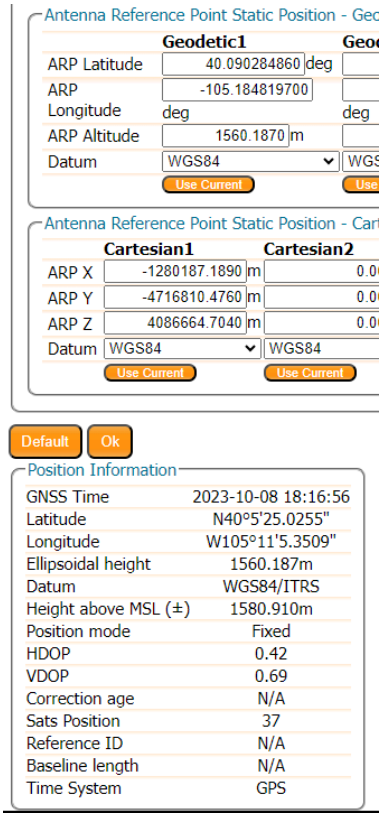


Precise antenna position

We can now store those coordinates in the mosaic-X5 module memory, either as **Geodetic** (Latitude, Longitude, Altitude) or **Cartesian** (ECEF X/Y/Z) coordinates. The mosaic-X5 allows you to store 5 of each.

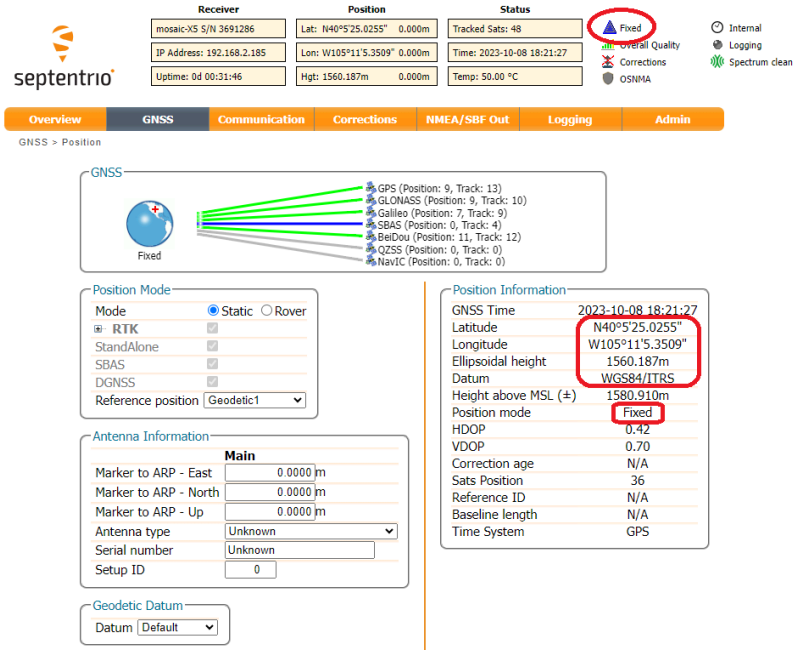


Setting the fixed position



Setting the fixed position

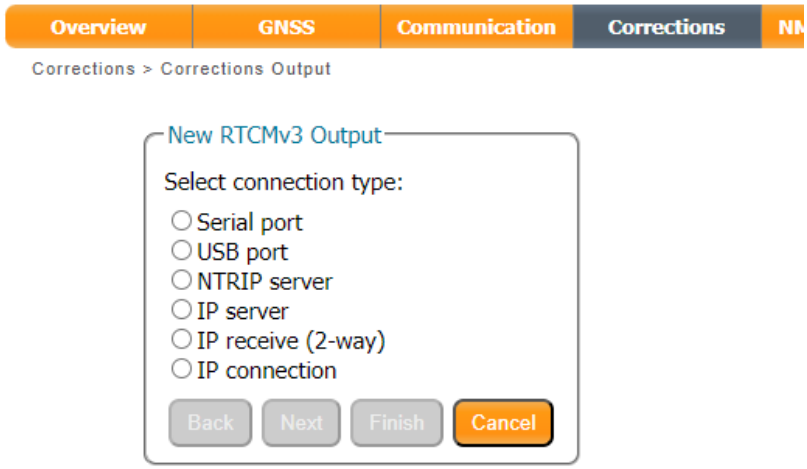
We can now generate corrections using that static / fixed antenna position.



Setting the fixed position

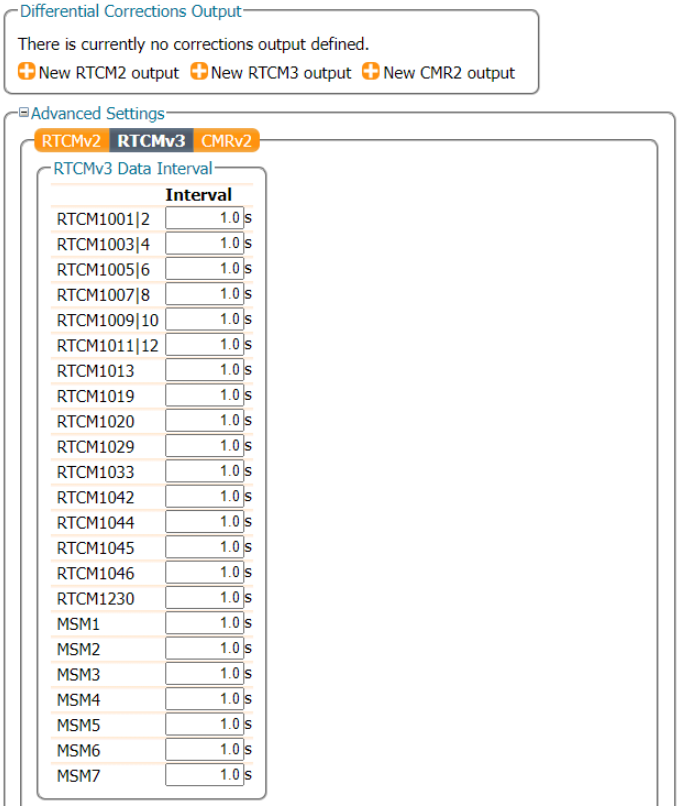
Base Station - Output Corrections

Now we know the Base's antenna position accurately, we can generate and share RTK corrections in a number of ways. The mosaic-X5 is a very clever module and is able to share corrections on a local network as an NTRIP server, through an NTRIP Caster, or directly using Serial, USB, or an IP connection. For example, here are the output options available for RTCMv3:



Selecting the RTCMv3 Output Type

In the Advanced Settings, you can also define which messages to output and at what internal:



Advanced settings for RTCMv3 Output

In this section, we are going to concentrate on sharing the corrections using an NTRIP Caster. Think of an NTRIP Caster as an intermediary, a way of sharing RTCM corrections between Bases and Rovers. There are several good free-to-use NTRIP Caster services available, including:

- RTK2go
- Emlid Caster

Here we are going to use RTK2go.

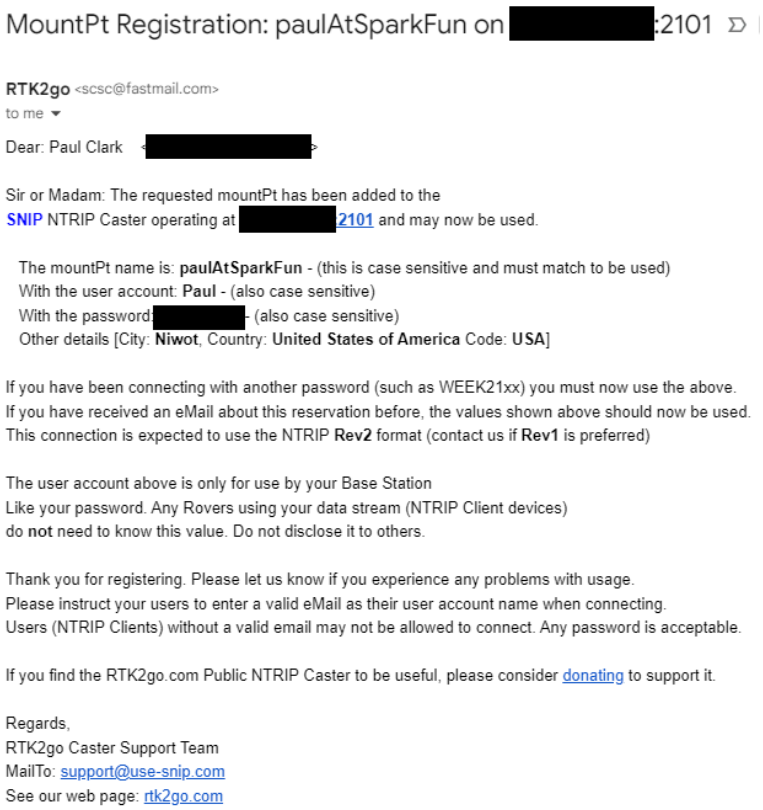
Create a New Base Station Reservation

Creating a New Reservation for your base station is really easy. Simply follow the instructions...

At SparkFun, we use **MountPt** names like `bldr_dwntwn2` , `bldr_dwntwn3` , `bldr_SparkFun1` - bldr is short for Boulder. Choose a name that is short, easy to remember and - we suggest - includes clues about who you are or where your server is. You can see the full list of all active bases at rtk2go.com:2101.

The mosaic-X5 is a high-end receiver so be sure to select **NTRIP Rev2** as the protocol and provide a login user name. IBAN - the banking network - provides a handy list of three-letter country codes.

The registration process typically takes under 8 hours to complete, somewhat longer on weekends. Once your mount point has been set up, RTK2go will send you a nice email confirming the mount point (**mountPt**) name, password and other details (Country, Country Code etc..). Make a note of those details, as we need to enter them into the mosaic-X5 NTRIP Server configuration.



MountPt Registration: paulAtSparkFun on [redacted]:2101

RTK2go <scsc@fastmail.com>
to me

Dear: Paul Clark <[redacted]>

Sir or Madam: The requested mountPt has been added to the
SNIP NTRIP Caster operating at [redacted]2101 and may now be used.

The mountPt name is: **paulAtSparkFun** - (this is case sensitive and must match to be used)
With the user account: **Paul** - (also case sensitive)
With the password [redacted] - (also case sensitive)
Other details [City: **Niwot**, Country: **United States of America** Code: **USA**]

If you have been connecting with another password (such as WEEK21xx) you must now use the above.
If you have received an eMail about this reservation before, the values shown above should now be used.
This connection is expected to use the NTRIP **Rev2** format (contact us if **Rev1** is preferred)

The user account above is only for use by your Base Station
Like your password. Any Rovers using your data stream (NTRIP Client devices)
do **not** need to know this value. Do not disclose it to others.

Thank you for registering. Please let us know if you experience any problems with usage.
Please instruct your users to enter a valid eMail as their user account name when connecting.
Users (NTRIP Clients) without a valid email may not be allowed to connect. Any password is acceptable.

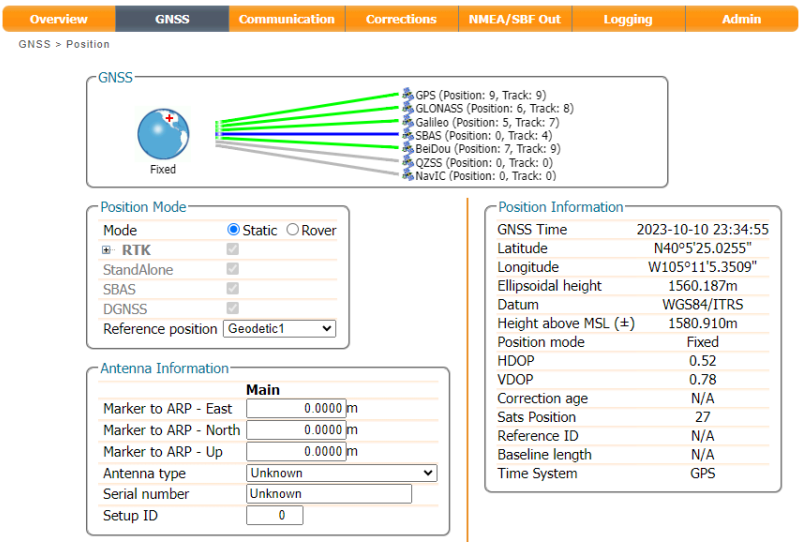
If you find the RTK2go.com Public NTRIP Caster to be useful, please consider donating to support it.

Regards,
RTK2go Caster Support Team
MailTo: support@use-snip.com
See our web page: rtk2go.com

RTK 2 Go Confirmation Email

Configure the NTRIP Server

Check that the **Position Mode** is set to **Static**, using either the precise position from your RINEX post-process data or an *automatic* **Reference Position**.



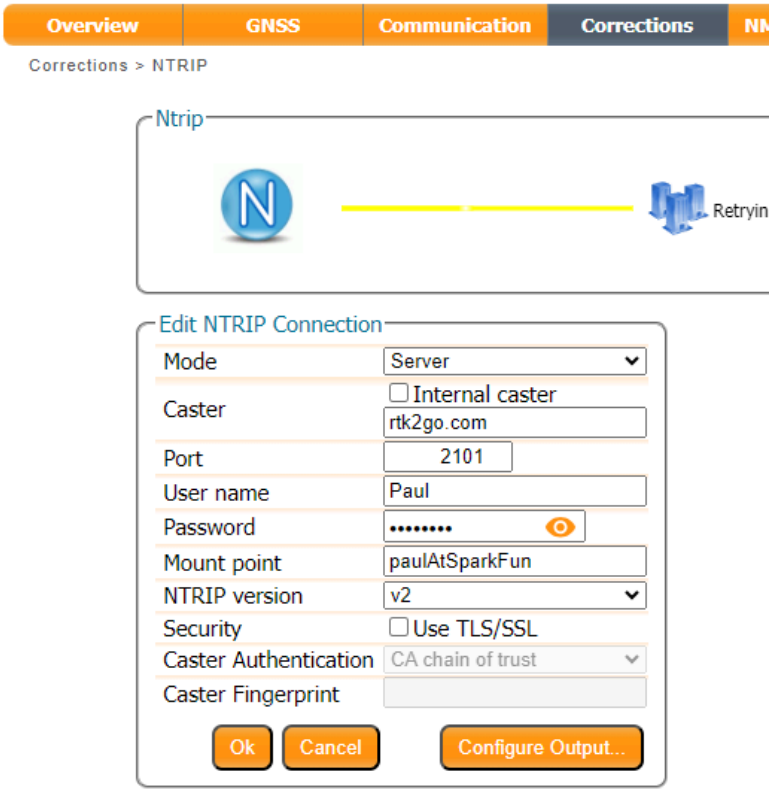
Configuring the Reference Position

In the **Corrections** tab, select **New NTRIP server**:



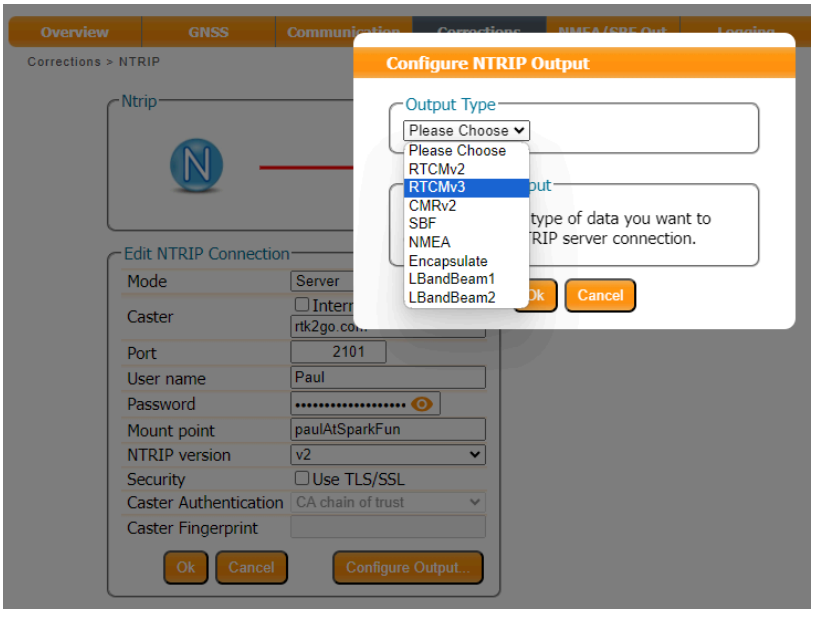
Configuring the NTRIP server

Enter the details from the confirmation email:



Configuring the NTRIP server

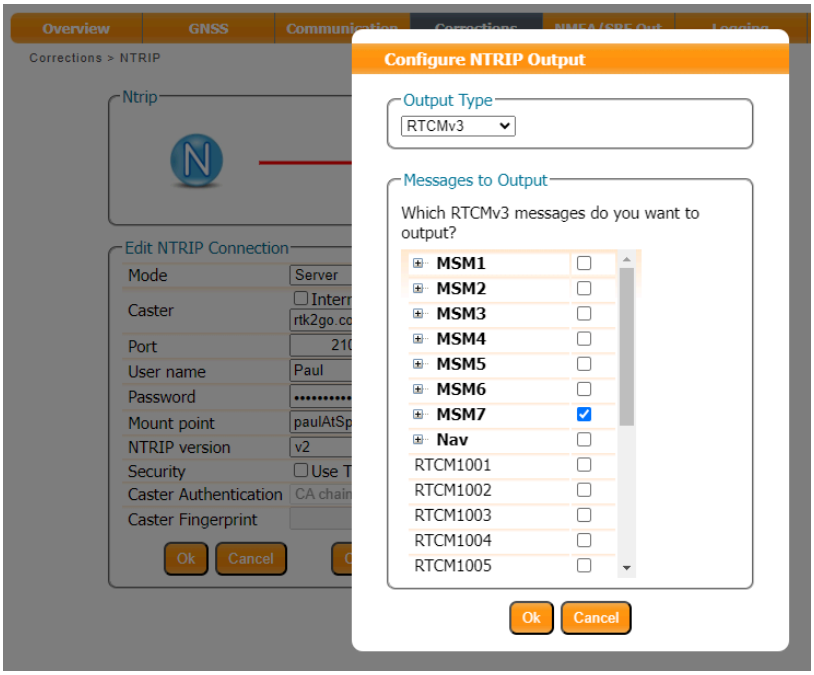
Then use the **Configure Output** button to select the RTCM Output Type and which RTCM messages to send. Here we select **RTCMv3**. The mosaic-X5 supports both **RTCMv2** and **RTCMv3**, but **v3** has better data compression and message integrity so we will use that.



Configure Output

We also need to select which type of RTCM message to send. Here we select **MSM7** since these are the longest highest precision messages. The message length does not matter since we are using

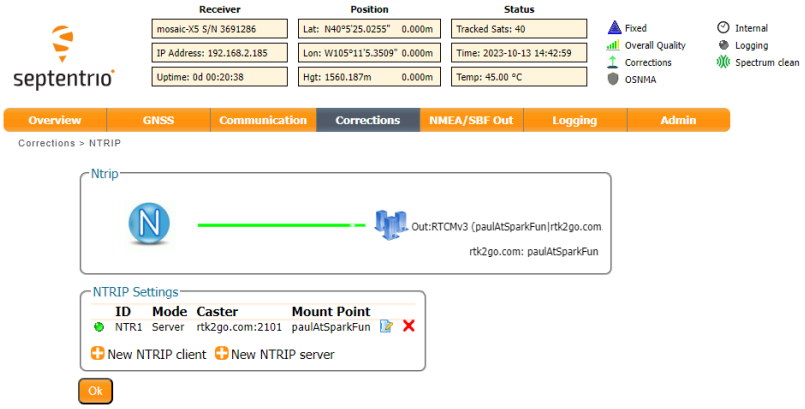
Ethernet and WiFi, but it could be important if we were using Serial (UART) or LoRa to carry the correction messages.



Configure Output

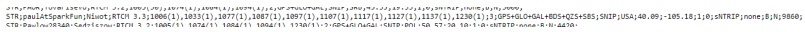
> **INFO**
>
> Remember to click the `OK` button at the bottom of the page, after making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

All being well, you should see a green line for NTRIP showing that your corrections are being forwarded to the caster:



NTRIP connection

We can check RTK2go.com:2101 to see if the corrections are being received:



Caster Status

We can see more details by opening the Caster Status Report. Here I am using the Mount Point **paulAtSparkFun**, so the full report is available at rtk2go.com:2101/SNIP::MOUNTPT?baseName=paulAtSparkFun:

Caster Status
Report

We've mentioned before that the mosaic-X5 is a very clever GNSS module. We can configure it to send corrections to *multiple* NTRIP Casters if desired, simply by creating additional connections.

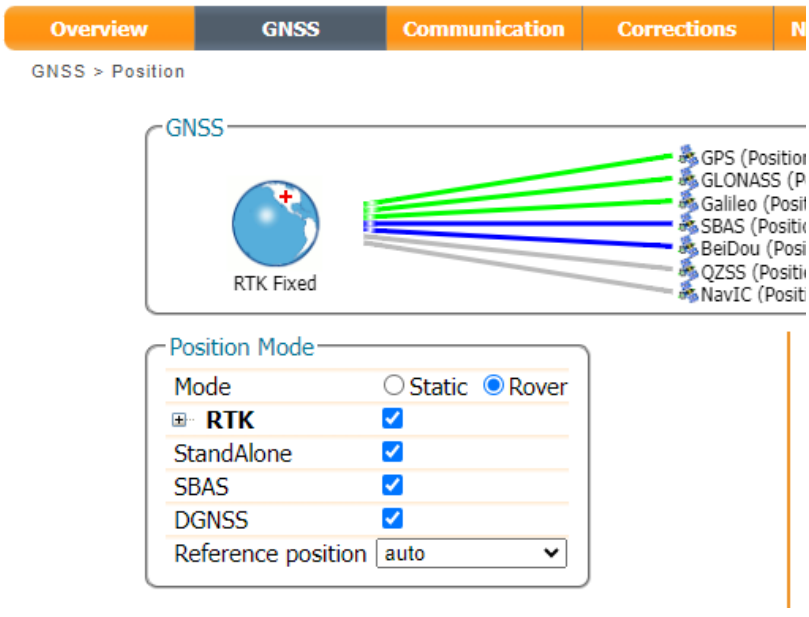
NTRIP Client - PointOneNav

In Real Time Kinematic (RTK) positioning, a Rover uses corrections from a local Base station to accurately augment its position to the centimeter level. This is done by using carrier-phase measurements of the received GNSS signals and differencing techniques. Put simply, an RTK Base knows where it is and - based on its location - can share satellite signal correction data with local Rovers, to allow them to achieve centimeter-level positioning.

NTRIP (Networked Transport of RTCM via Internet Protocol) is a standardized way of transferring RTCM correction data between RTK Bases and Rovers. It is the protocol used by NTRIP Casters for inbound and outbound corrections.

In this section, we are going to configure the mosaic-X5 as a Rover and have it use data from a commercial Correction Service which pools data from many base stations and provides you with corrections for your location.

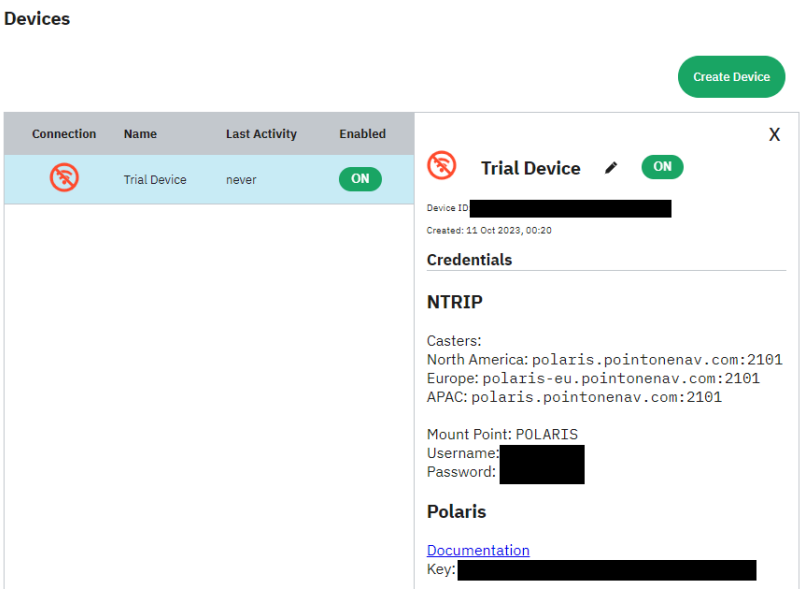
To begin, check that the mosaic-X5 is configured as a **Rover**:



Rover Mode

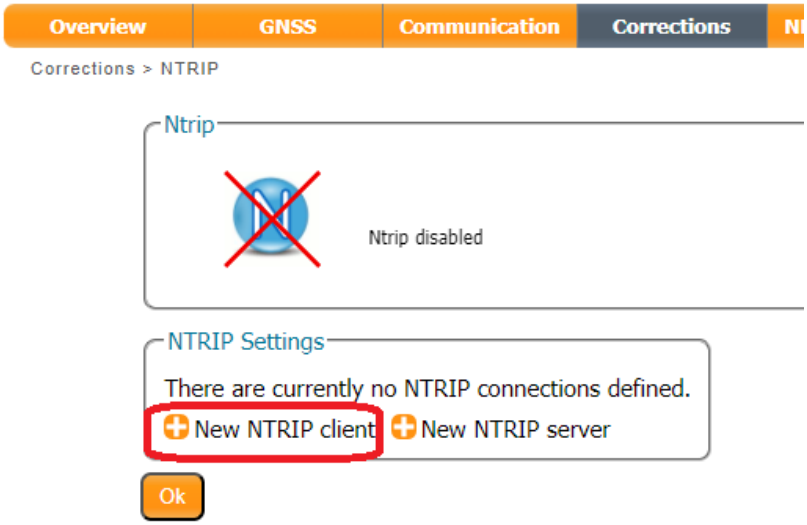
We've had good experiences with PointOneNav where you get a 14-day free trial without needing to give them your credit card details. Enter your details in the **Create your account** window, create an account and verify your email address. A Trial Device is created automatically. Make a note of the Caster address, Mount Point, Username and Password. You will need to enter those into the mosaic-X5 NTRIP Client configuration. We suggest bookmarking the link to the login page. If you are in Europe, the NTRIP Caster server is different. Please check the Polaris NTRIP API documentation

for details. PointOneNav supports TLS encryption too. Again, please check the documentation for details.



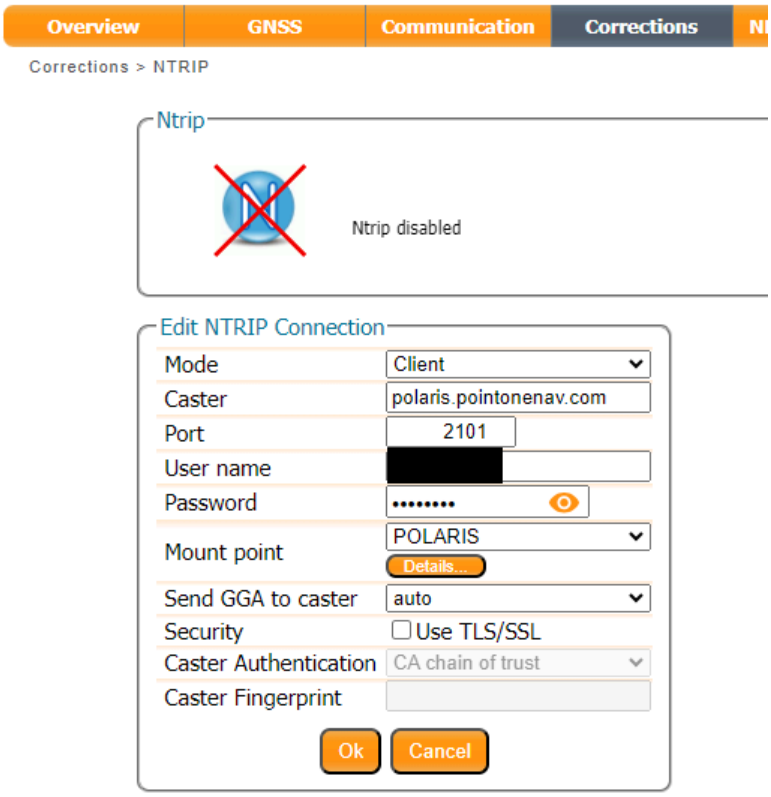
Point One Trial Device

In the **Corrections** tab, select **New NTRIP client**:



Configuring the NTRIP client

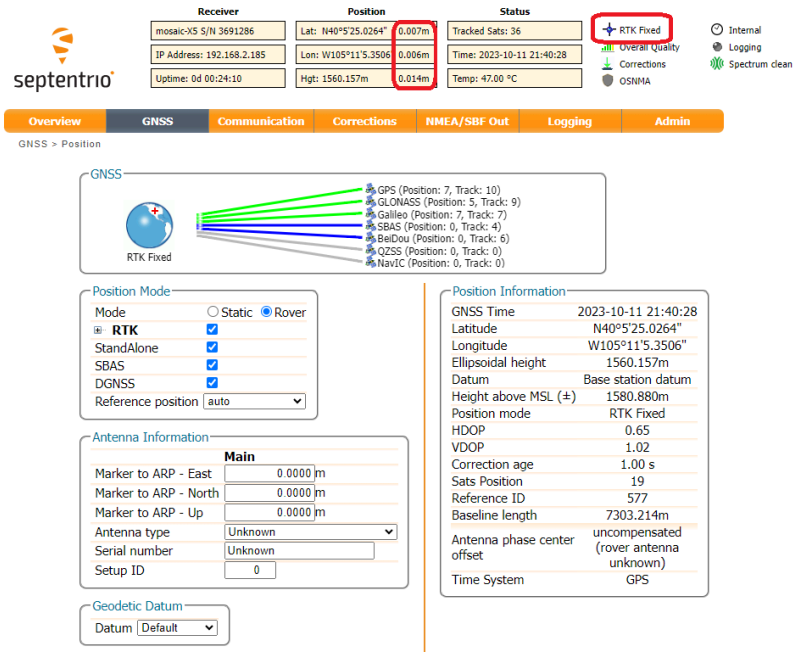
Enter the details of your PointOneNav Trial Device. The **Mount point** pull-down is populated after you enter the **Caster** and **Port**.



Configuring the NTRIP client

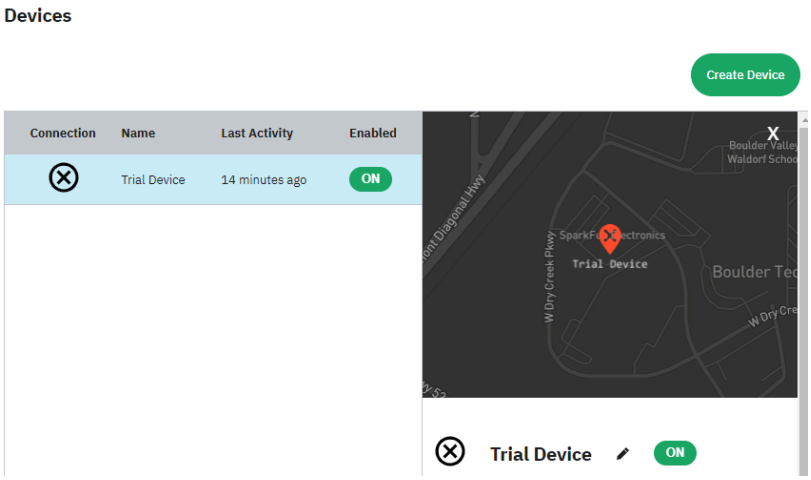
Remember to press **OK** after each change - and save the configuration.

Once the mosaic-X5 is connected to the correction service, you should get **RTK Fixed** almost instantly. It really is as simple as that. Here at SparkPNT, we are able to get ~7mm accuracy using PointOneNav.



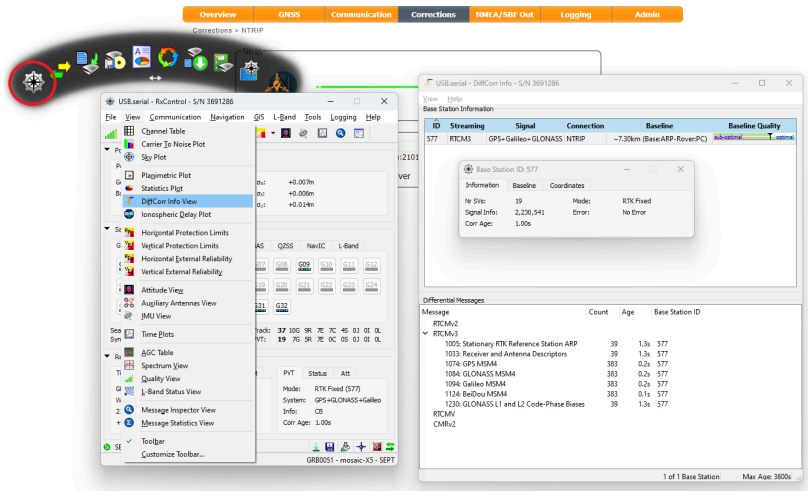
Position Accuracy

The NTRIP connection is two-way. The mosaic-X5 is sending its approximate position to the Correction Service so that the Correction Service can send us the best data for our location. Another thing we like about PointOneNav is that they display your location in the web dashboard. This is a very handy, simple way of seeing where your Rover devices are.



Point One Nav Device Location

If we want to, we can find out where our correction data is coming from. 7mm is pretty darn good, so the Base must be quite close. It turns out it is just 7.3km north of our location. Open **RxTools** \ **RxControl** and connect to the mosaic-X5; USB is ideal or you can connect over your network using IP. Open the **View** \ **DiffCorr Info View** (Differential Correction Information View) and a window is displayed showing which RTCM stations we are connected to and the baseline distance(s). Double-click on the Station ID and another window pops up where the information and coordinates of the station can be seen.



RTCM Station Information

NTRIP Client - Test Corrections

Now that the Base mosaic-X5 is sending RTCM corrections to RTK2go via NTRIP, we can configure a second mosaic-X5 GNSS receiver to use those corrections. We've mentioned before that the mosaic-X5 is a very clever GNSS module. We can configure it to use more than one NTRIP Caster or Correction Service. It will use the best data available to give an RTK fix. This is very handy if one of your correction sources goes offline, the mosaic-X5 will fall back to another.

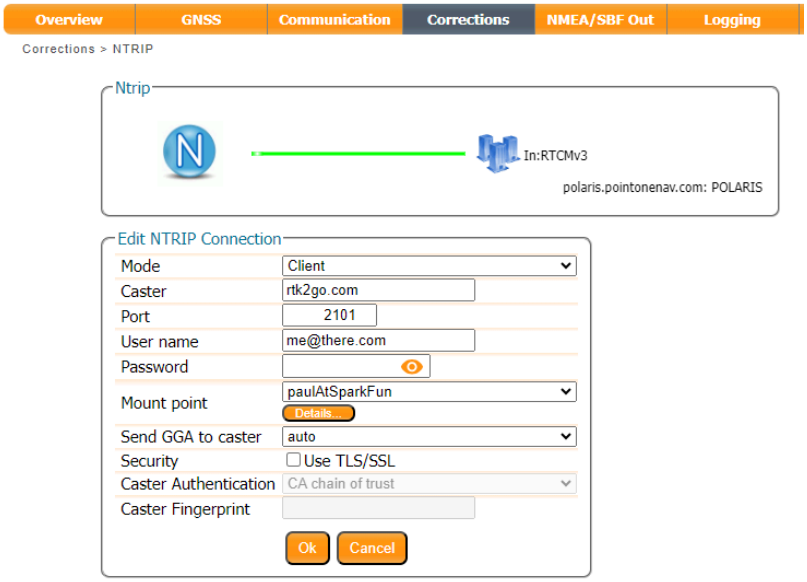
The configuration process is exactly the same as we used for the PointOneNav NTRIP client. We simply adding another NTRIP client.

In the **Corrections** tab, select **New NTRIP client**:



Configuring the NTRIP client

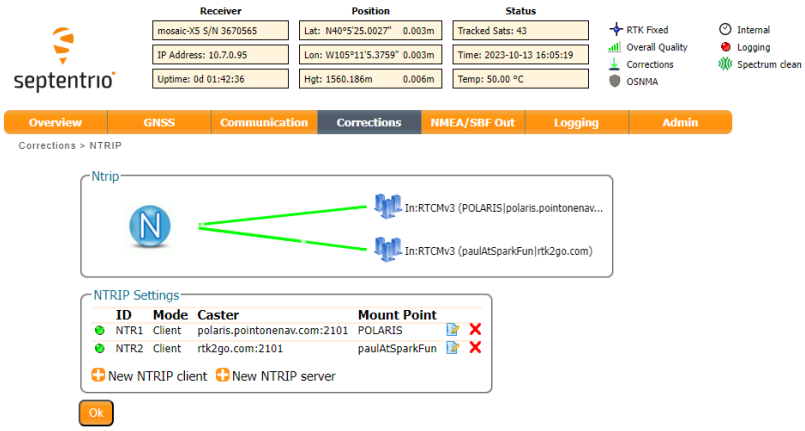
Enter the details of your RTK2go mount point. You don't need a password, but you do need to enter a username (email address) - a fake one is OK. The **Mount point** pull-down is populated after you enter the **Caster** and **Port**.



Configuring the NTRIP client

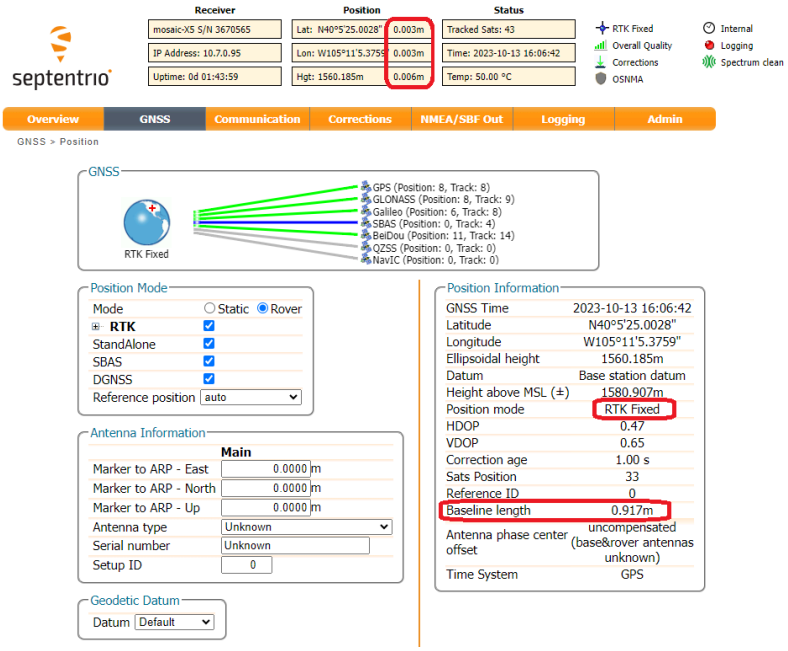
Remember to press **OK** after each change - and save the configuration.

All being well, you should now see two green lines for your two NTRIP correction sources:



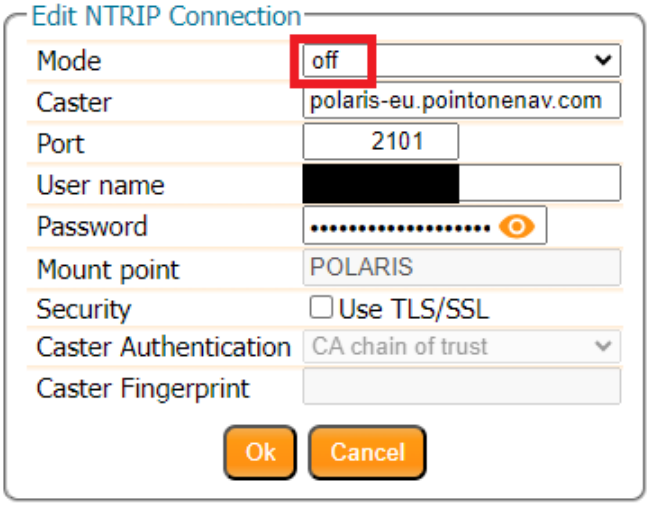
Configuring the NTRIP client

Just for giggles, I placed the Rover and Base antennas 3' (3 feet = 0.914m) apart on the SparkFun building parapet. The Rover mosaic-X5 is showing 3mm horizontal accuracy and a Base-Rover baseline length of 0.917m. I am very happy with that!



RTK Baseline Length

If you want to temporarily disable an NTRIP client, click the **Edit NTRIP settings** icon and then set the **Mode** to **off**. If you have two clients configured, this allows you to disable one and use the corrections from the other.



Disable NTRIP client

Advanced Configurations

Enabling GNSS Signals

Instructions to enable the reception of specific GNSS signals and include its use for RTK corrections

Mining with Onocoy

Instructions for setting up the SXM-E as an RTK server on the Onocoy network

Enabling GNSS Signals

Supported GNSS Signals

The mosaic-X5 module is capable of receiving most of the GNSS signals from the various frequency bands of each constellation. By default, the module is only configured to track and receive signals (marked in green, in the table below) from specific satellites and frequency bands. Whereas, the signals marked in grey are also supported by the module, but each signal needs to be enabled before they can be utilized for *(differential)* positioning or output RTK correction data as well. Meanwhile, any ~~signals colored in red and striked out~~ are not supported by the mosaic-X5 module; likely due to their proprietary nature, existence outside the module's frequency range, or are under development/recently implemented.

		Frequency Band																			
		L5/E5			L2					L6/E6				L1/E1							
GNSS Constellation	GPS	L5			L2PY		L2C		~~L2M~~					L1CA			L1PY			~~L1M~~	
	GLONASS				L3		L2P		L2CA					L1CA						~~L1P~~	
	Galileo	E5a			E5b					E6BC			~~E6PRS~~	E1BC						~~E1PRS~~	
	Beidou	B2a			B2b			B2I		B3I				B1I					B1C		
	SBAS	L5												L1							
	QZSS	L5			L2C					~~L6E/LEX~~			~~L6D~~	L1CA			L1C		~~L1S/SAIF~~		
	Navic	L5																			

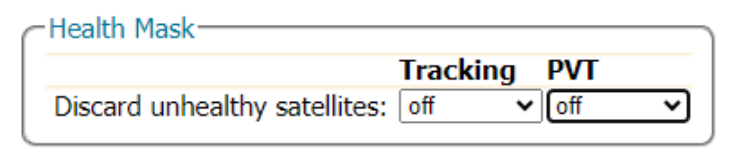
LEGEND FOR GNSS SIGNALS

- Supported; Enabled by default
- Supported; Not enabled by default
- ~~**Not supported**~~

Enabling Signal Reception

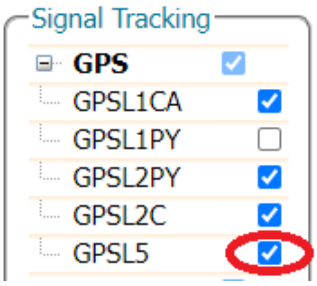
Below, are instructions to configure the mosaic-X5 module to receive any of the supported GNSS signals that aren't enabled by default. Here, we will focus specifically on enabling the GPS L5 band, but others signals can also be enabled in these steps.

Web Interface

> **⚠ ENABLING THE GPS-L5 SIGNAL**
>
> Currently, the GPS-L5 service is pre-operational and marked as "*unhealthy*"; therefore, extra configuration steps are required to enable this frequency band.
>
> From the **Admin** tab, click on **Expert Control** from the drop-down menu. Then, navigate through the **Control Panel** > **Navigation** > **Receiver Operation** > **Masks** menus to find the `Discard unhealthy satellites` setting. Set both the **Tracking** and **PVT** options to `off`, from their drop-down menus.
>
> 
>
>
> *Enabling the tracking and PVT use of signals from "unhealthy" satellites.*

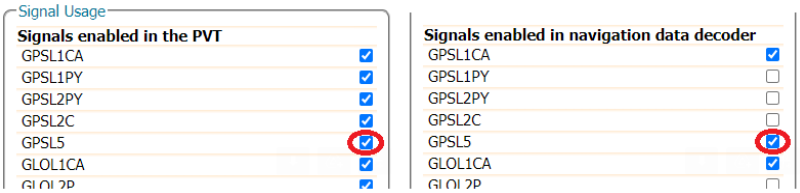
Through the web interface, users will need to access the advanced settings of the configuration menu by clicking on the **Expert Control** option from the **Admin** tab's drop-down menu. Then, navigate through the **Control Panel** > **Navigation** > **Advance User Settings** menus:

> **Tracking** > **Signal Tracking** : Enable GPSL5 signal tracking



Enable the tracking of GPS L5 signals

> **PVT** > **Signal Usage** - Enable GPSL5 in both the *PVT* and *navigation data decoder* sections



Enable the use of GPS L5 signals for data decoding and the PVT solution

> **INFO**
>
> Make sure to click the `OK` button at the bottom of each page, where you are making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

Command Interface

> **⚠ `GPS-L5` SIGNAL**
>
> Since the GPS-L5 service is currently pre-operational and marked as "*unhealthy*", it takes some extra configuration steps to enable the GPS-L5 frequency band and corrections.
>
> - `setHealthMask, Tracking, off`
> - `setHealthMask, PVT, off`

- `setSignalTracking, +GPSL5`
- `setSignalUsage, +GPSL5, +GPSL5`

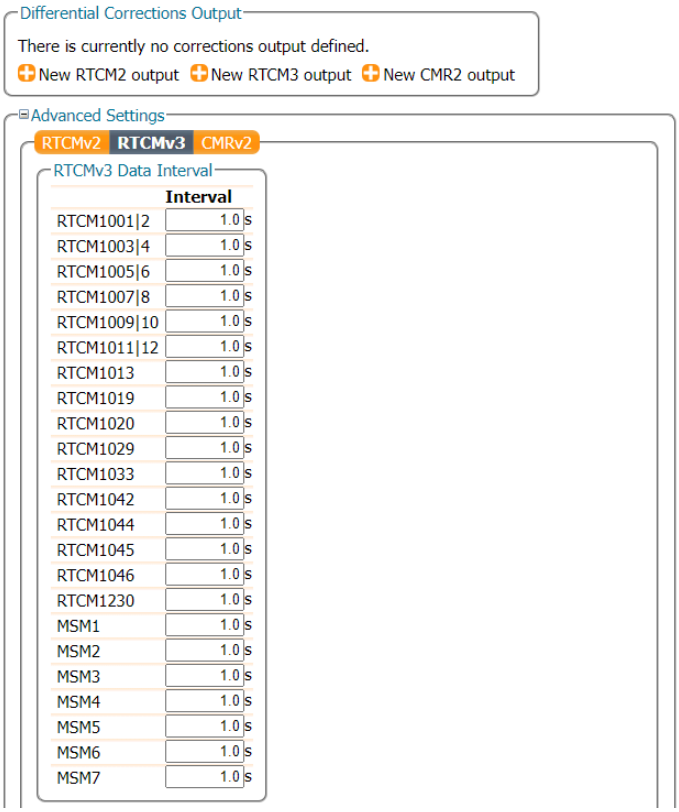
Output L5 Corrections

Below, are instructions for outputting RTK corrections for the GPS L5 frequency band, from the SXM-E.

> **⚠ WARNING**
>
> The SXM-E must be configured to receive those GNSS signals *(see instructions above)*; otherwise, it won't have any data to provide the corrections with.

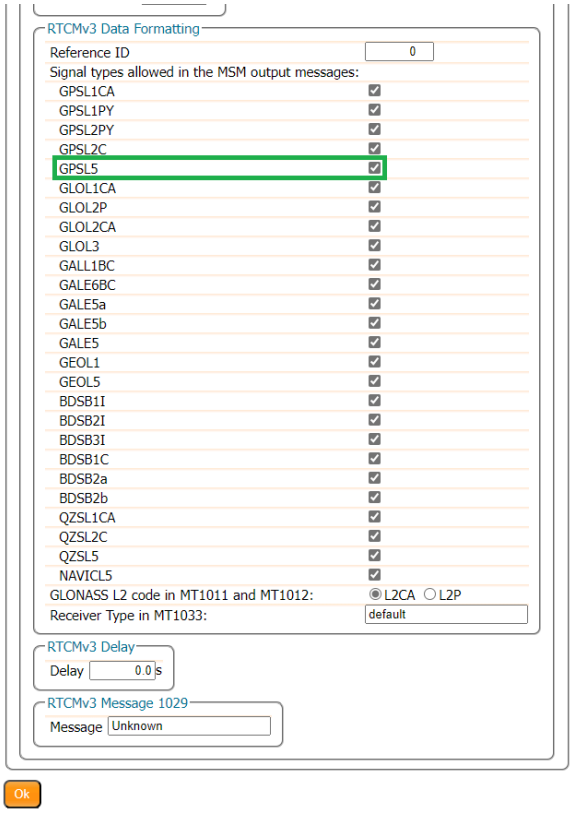
Web Interface

From the drop-down menu of the **Corrections** tab of the web interface, select **Corrections** > **Corrections Output** to bring up the corrections output settings. On this page, locate the **Advanced Settings** menu, where users can define the data intervals and formatting. Clicking on this will prompt a security dialog box to pop up, click the `Proceed` button to make changes to these settings. RTK corrections for the GPS L5 frequency band are only available in the RTCMv3 messages; therefore, users will need to select the **RTCMv3** tab.



Open the Advanced Settings for RTCMv3 Output

Scroll down to the **RTCMv3 Data Formatting** section, select the box next to the `GPSL5` to enable the RTK corrections for the GPS L5 frequency band. In the example below, we have enable all the available signals to be included in the MSM output messages.



ENable the corrections for the GPS L5 signal in the MSM output

> **TIP**
>
> Make sure to click the `OK` button at the bottom of each page, where you are making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

Command Interface

- `setRTCMv3Formatting, 0, +GPSL5`

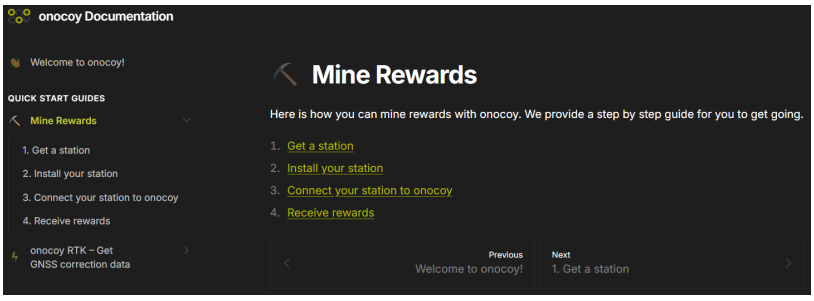
Mining with Onocoy

Some RTK networks, such as Onocoy, offer users incentives for operating a reference station and contributing RTK corrections to their network. These incentives are often tied to a form of cryptocurrency, which is rewarded based upon a performance scale after the data is validated.

> **WARNING**
>
> This section is for educational purposes only!
>
> This section doesn't contain nor should it be represented as an endorsement of any of the projects mentioned nor as financial advice. Don’t invest anything you aren’t willing to lose.

Onocoy Documentation

Onocoy provides users with documentation for mining rewards with an RTK reference station. To get started, users will need to have installed and mounted the hardware of their SXM-E and created a user account with Onocoy. Then, users can obtain NTRIP server credentials from their account, to connect their reference station to the Onocoy network.



Onocoy documentaion for mining rewards.

Device Configuration

In order to connect their SXM-E to the Onocoy network, users will need to adjust the configurations of the mosaic-X5 and setup an NTRIP server to the Onocoy network.

Enable L5 Band Reception

Before setting up the SXM-E as an NTRIP server, users will need to enable the GPS L5 frequency band in order to provide corrections for it. These steps, are documented in the **Enabling GNSS Signals** section of this tutorial:

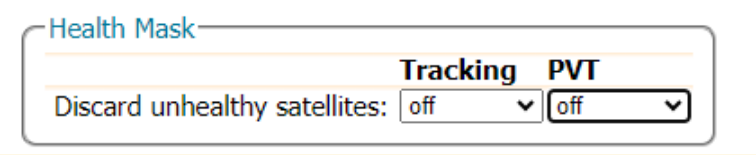
L5 SIGNAL RECEPTION

Below, are instructions to enable the SXM-E to receive GNSS signals from the GPS L5 frequency band.

ENABLING THE GPS-L5 SIGNAL

Currently, the GPS-L5 service is pre-operational and marked as "*unhealthy*"; therefore, extra configuration steps are required to enable this frequency band.

From the **Admin** tab, click on **Expert Control** from the drop-down menu. Then, navigate through the **Control Panel** > **Navigation** > **Receiver Operation** > **Masks** menus to find the `Discard unhealthy satellites` setting. Set both the **Tracking** and **PVT** options to `off`, from their drop-down menus.

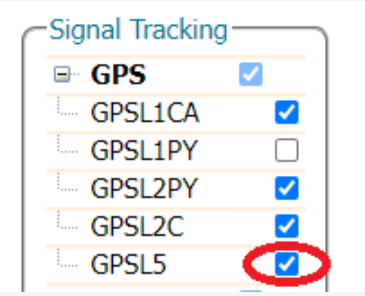


Enabling the tracking and PVT use of signals from "unhealthy" satellites.

Through the web interface, users will need to access the advanced settings of the configuration menu by clicking on the **Expert Control** option from the **Admin** tab's drop-down

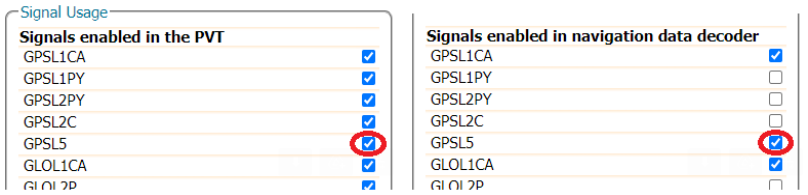
menu. Then, navigate through the **Control Panel** > **Navigation** > **Advance User Settings** menus:

> **Tracking** > **Signal Tracking** : Enable GPSL5 signal tracking



Enable the tracking of GPS L5 signals

> **PVT** > **Signal Usage** - Enable GPSL5 in both the *PVT* and *navigation data decoder* sections



Enable the use of GPS L5 signals for data decoding and the PVT solution

> **INFO**
>
> Make sure to click the `OK` button at the bottom of each page, where you are making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

Connect to the Onocoy Network

In order to connect the SXM-E to the Onocoy network, users will need to set up an NTRIP caster in the mosaic-X5 configuration settings. In order to receive the rewards, Onocoy expects a specific set of configurations as listed in their online documentation.

Enable NTRIP server functionality for NTRIP 2.0 devices *(see instructions **NTRIP 1.0** below)*

- Configure the NTRIP caster to point to servers.onocoy.com, port 2101.
- For the mountpoint name, you can choose any alphanumeric text you like. This is only informative and will show up in your dashboard, not publicly. It will also help you identify which server device is which.
- For username and password, enter the credentials for your device as created in Step 2.

Enable the right NTRIP messages

- Enable RTCM3, MSM Messages 1005, 1077, 1087, 1097, 1117, 1127, 1137 and 1230 in the reference station receiver.
  - onocoy supports MSM4/5/6/7 messages
- Activate them to be sent out at 1Hz rate (except for message 1005, which can be configured for a much lower rate, for example once every 30 seconds).
- Enable all GNSS constellations and bands the receiver supports. Typically, receivers are preconfigured to this already.

NTRIP 1.0 devices

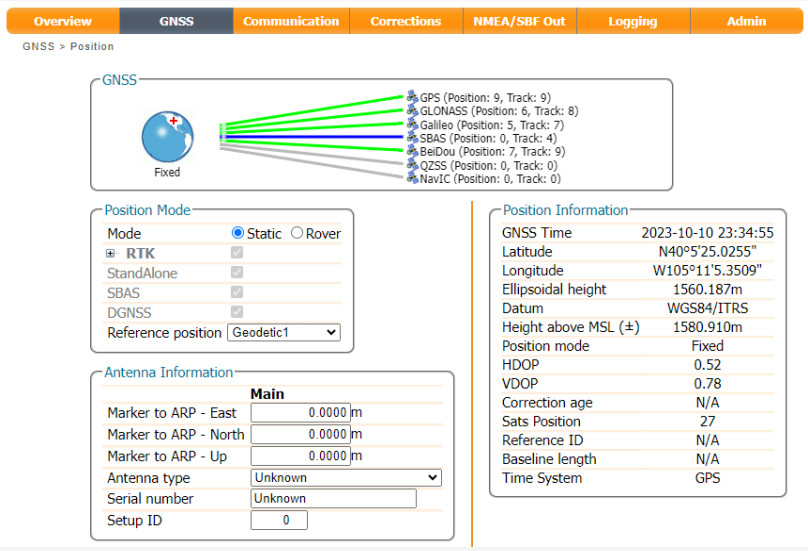
- For NTRIP 1.0-only devices (for example ESP32 XBee based units such as Ardusimple NTRIP servers), please enter the NTRIP username as *MOUNTPOINT* and password into the password field. Leave the device's username field, if any, empty.

The specific parameters for connecting an NTRIP server to the Onocoy network, as listed in their documentation.

Most of steps for these requirements, are documented in the **Output RTK Corrections** section of this tutorial:

CONFIGURE THE NTRIP SERVER

Check that the **Position Mode** is set to **Static**, using either the precise position from your RINEX post-process data or an *automatic* **Reference Position**.



Configuring the Reference Position

In the **Corrections** tab, select **New NTRIP server**:



Configuring the NTRIP server

Ignore - Step from Original Instructions

Ignore this step, documented in the **Output RTK Corrections** section:

~~Enter the details from the confirmation email:~~

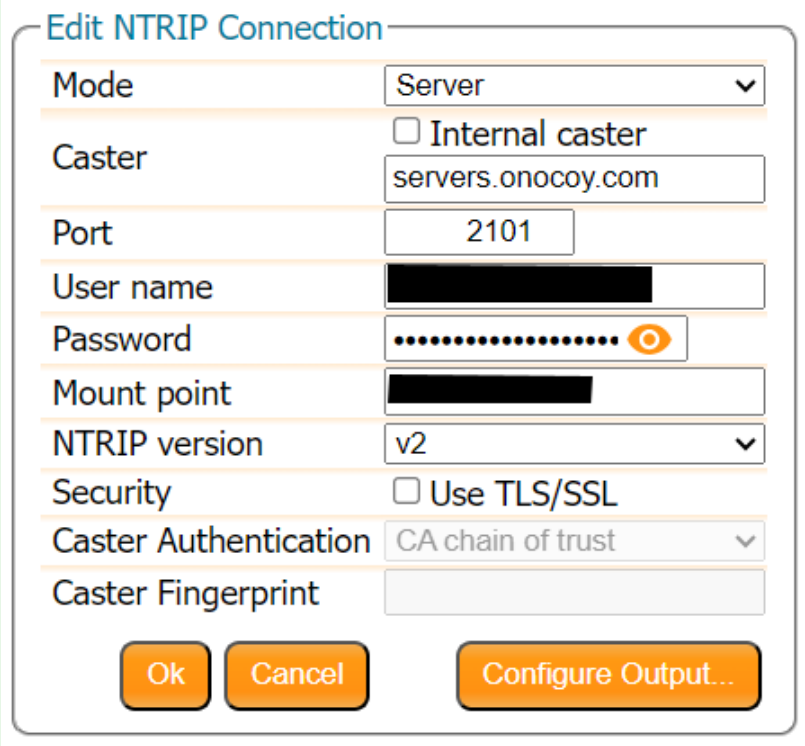


~~*Configuring the NTRIP server*~~

SUCCESS

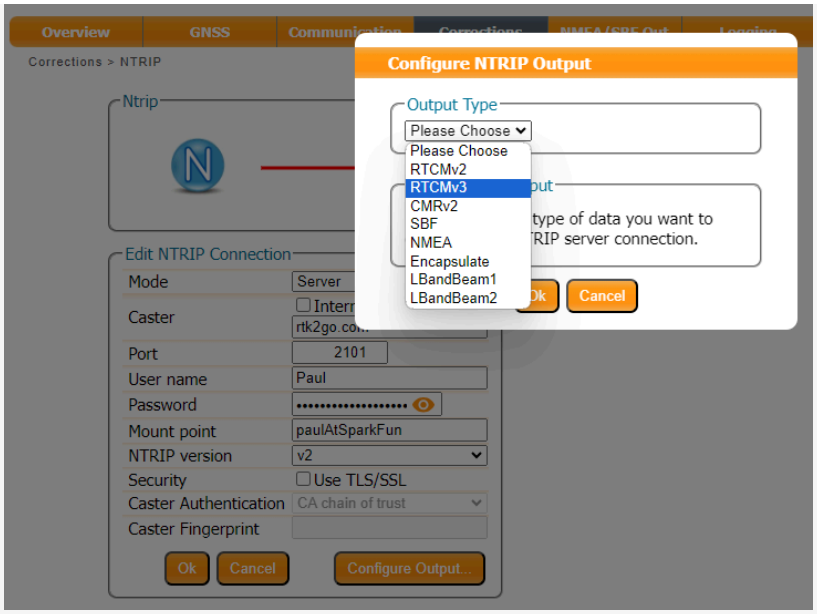
Instead of the *confirmation email*, use the information provided in the Onocoy documentation:

> - Configure the NTRIP caster to point to `servers.onocoy.com`, port `2101`.
> - For the `mountpoint` name, you can choose any alphanumeric text you like. This is only informative and will show up in your dashboard, not publicly. It will also help you identify which server device is which.
> - For `username` and `password`, enter the credentials for your device as created in Step 2.



Configuring the connection to Onocoy

Then use the **Configure Output** button to select the RTCM Output Type and which RTCM messages to send. Here we select **RTCMv3**. The mosaic-X5 supports both **RTCMv2** and **RTCMv3**, but **v3** has better data compression and message integrity so we will use that.

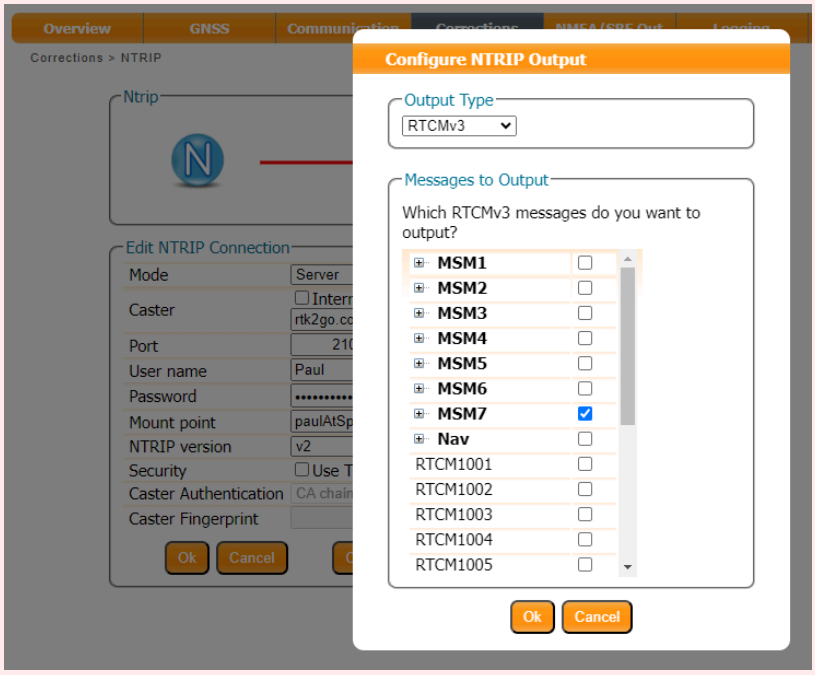


Configure Output

Ignore - Step from Original Instructions

Ignore this step, documented in the **Output RTK Corrections** section:

~~We also need to select which type of RTCM message to send. Here we select **MSM7** since these are the longest highest precision messages. The message length does not matter since we are using Ethernet and WiFi, but it could be important if we were using Serial (UART) or LoRa to carry the correction messages.~~

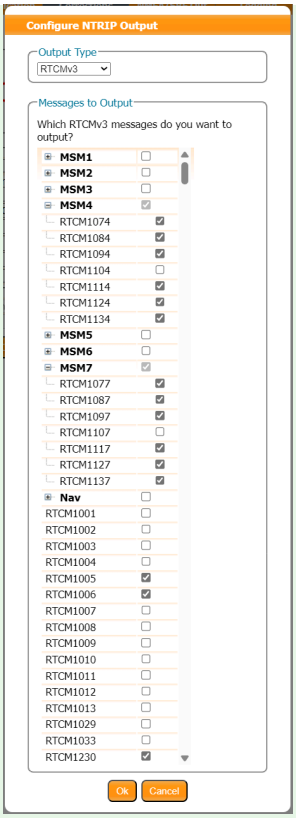


~~*Configure Output*~~

SUCCESS

Instead, select the RTCM messages listed in the Onocoy documentation. Below, are the RTCM messages we selected:

- Enable RTCM3, MSM Messages `1005`, `1077`, `1087`, `1097`, `1117`, `1127`, `1137`, and `1230` in the reference station receiver.
- onocoy supports MSM4/5/6/7 messages



RTCMv3 messages

INFO

Remember to click the `OK` button at the bottom of the page, after making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

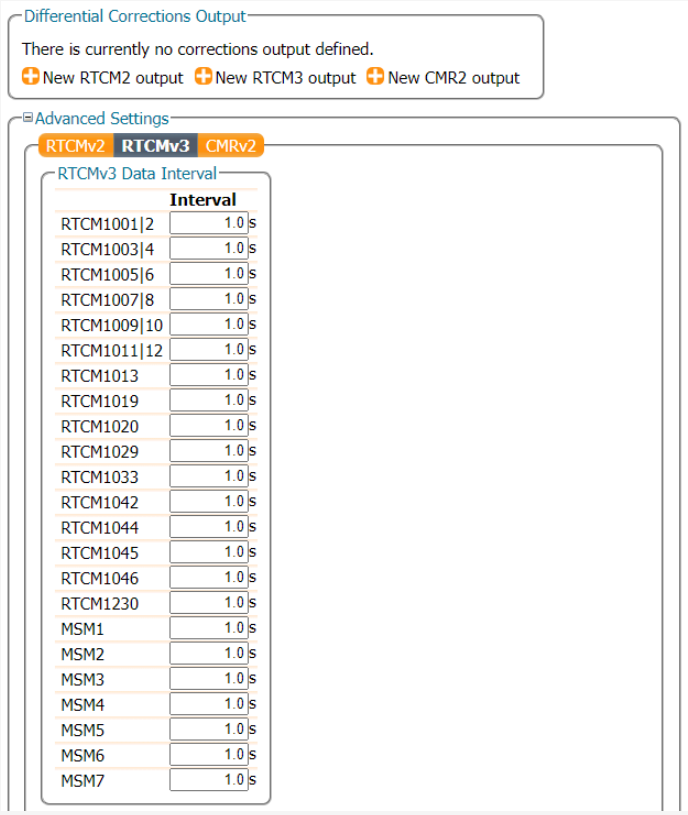
Enable L5 Corrections

The last step in configuring the NTRIP server, users will need to enable the GPS L5 signal to be included in the MSM output messages. This will allow the SXM-E to provide RTK corrections for the GPS L5 frequency band. These steps, are documented in the **Output L5 Corrections** section of this tutorial:

OUTPUT L5 CORRECTIONS

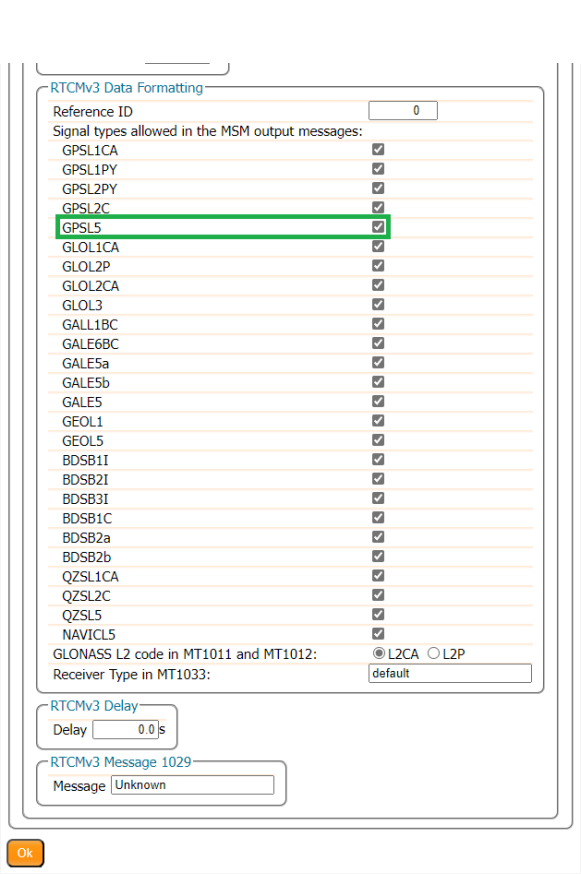
Below, are instructions for outputting RTK corrections for the GPS L5 frequency band, from the SXM-E.

From the drop-down menu of the **Corrections** tab of the web interface, select **Corrections** > **Corrections Output** to bring up the corrections output settings. On this page, locate the **Advanced Settings** menu, where users can define the data intervals and formatting. Clicking on this will prompt a security dialog box to pop up, click the `Proceed` button to make changes to these settings. RTK corrections for the GPS L5 frequency band are only available in the RTCMv3 messages; therefore, users will need to select the **RTCMv3** tab.



Open the Advanced Settings for RTCMv3 Output

Scroll down to the **RTCMv3 Data Formatting** section, select the box next to the `GPSL5` to enable the RTK corrections for the GPS L5 frequency band. In the example below, we have enable all the available signals to be included in the MSM output messages.



ENable the corrections for the GPS L5 signal in the MSM output

> **TIP**
>
> Make sure to click the `OK` button at the bottom of each page, where you are making changes to update the current configuration. Also, don't forget to save it to boot, if you want the configuration to persist when you cycle power.

Receive Rewards

Once users have connected the SXM-E as an NTRIP server on the Onocoy network, there are a few additional steps in that need to be completed, a specified in their online documentation.

Happy Mining!

Support

🧑‍🦲 Technical Support

A guide on how to get help

🗃️ Resources

Resources for the product, including datasheets, application notes, and manuals

Technical Support

If you need technical assistance or more information on a product that is not working as you expected, please head over to our SparkPNT Forum. Feel free to check out our troubleshooting tips, below, for some suggestions on common topics. By posting on the forum you benefit other users with similar questions. It also gives us feedback as a whole about where our products and documentation need improvement.

> **ACCOUNT REGISTRATION REQUIRED**
>
> On your first visit to our forum, you'll need to create a Forum Account to post questions.

Order and Shipping Issues

If your order was damaged or is missing any parts, please email our support team at `support@sparkpnt.com`. Please, include your order number and a clear picture of your package's condition on arrival or showing any missing items.

> **NOTE**
>
> If your order was placed through one of our distributors, please contact them first to resolve any issues.

> **TIP**
>
> By including information and images of the issue, our support staff can transfer you to the appropriate service team, who can get a jump start on a resolution to the problem.

Returns Process

For more details on our returns process, check out our full Return Policy.

Troubleshooting Tips

Below are some tip to troubleshoot common issues that users may come across.

What should be in the Box?

The SparkPNT SXM-E comes packaged as a complete kit, with all the accessories you'd need to set up an RTK base station or NTRIP server. Inside the kit, users will find the SXM-E, GNSS antenna, and another box containing additional accessories. Inside, the accessory box, users will find the CAT-6 Ethernet cable, USB cable, SMA to TNC cable, USB power supply, WiFi antenna, a pack of silicone bumpers, and 32GB SD card.



The contents of the SXM-E package.

Kit contents

Products in the accessories box.

TIP

If you are missing any of these items, please email our support team at `support@sparkpnt.com`. Please, include your order number and a clear picture of your package's condition if it was damaged or showing any missing items.

WiFi Connectivity

For WiFi connectivity issues, here are some simple troubleshooting tips:

- Verify that the SXM-E is operating in **Mode:** `2` for WiFi functionality.
- Double-check the WiFi credentials used in the ESP32 configuration.
- Make sure that the configured WiFi network is broadcasting on a 2.4GHz band. The ESP32 cannot access the 5GHz band.
- Make sure that the Ethernet cable is not coiled around/near the WiFi antenna. This will cause data packets to be lost and the mosaic-X5 web page to hang/freeze.

WiFi Provisioning

The ESP32 firmware we provide is only compatible with basic `SSID` and `Password` WiFi authentication. The firmware is not compatible with networks that implement other provisioning methods such as a captive portal, a QR code, or Wi-Fi protected setup. Our intention was that users could easily develop their, own firmware to suit their needs.

> **TIP**
>
> Here are some resources that might be useful for getting started with other provisioning methods:
>
> - ESP-IDF: Wi-Fi Provisioning
> - ESP-IDF: Example - Wi-Fi Provisioning Manager
> - AutoConnect: Captive Portal
> - Video Tutorials:
>   - QR Code
>   - Access Point/Web Server and Guide

Data Logging

For data logging issues, here are some simple troubleshooting tips:

- Make sure that your SD card is formatted to a `FAT32` file system.
  - The `FAT32` file system also limits the maximum capacity of the card to less than **32GB** *(i.e. a 256GB SD card will not work)*.
- Make sure that the mosaic-X5 module has a configured data stream output.
- Use the mosaic-X5 web page to verify that the SD card is mounted as a storage drive.

OLED Dead Pixels

Got a few dead pixels on your OLED display? Please email our support team at `support@sparkpnt.com` and include your order number with a clear picture showing the dead pixels.

Vehicle Power - Dead Battery

Is the SXM-E killing your vehicle's battery?

Make sure that the external power source for the SXM-E is not directly tied to the vehicle's battery, `Always On`, or accessory circuits. Otherwise, users will risk killing the battery while the engine is off. We recommend locating the *ignition on* or *switched power* circuit, which is only powered when the key is in the `On` position *and the engine is running*.

> **NOTE**
>
> The `On` position, is where a key normally rests after the engine is started. However, users can still move the key from the `Off` position and into the `On` position without starting the engine. In this case, the alternator is not running and keeping the battery charged.
>
> Modern *eco-efficient* vehicles may automatically shut down the engine if the vehicle is idling too long. Therefore, cutting off the vehicle's alternator that keeps the battery charged. Luckily, most vehicles with this *automatic start/stop* technology will monitor the battery's voltage and restart the engine when required. With this in mind, users may want to initially monitor their battery voltage, in case their vehicle isn't "*so smart*" 😄.

L5 Band

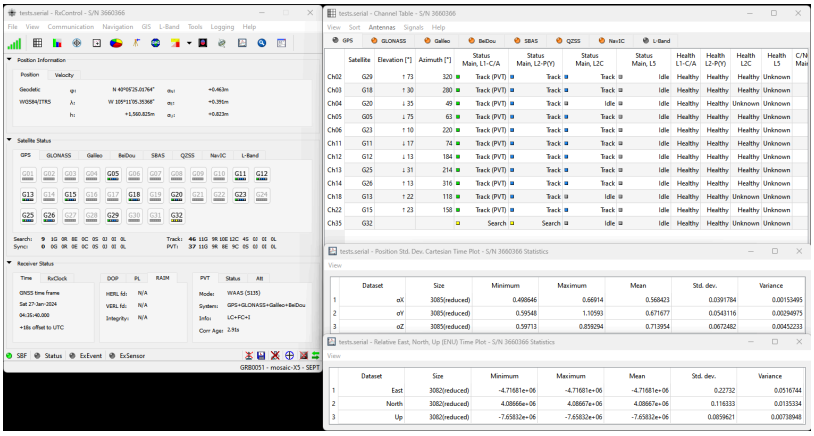
By default, the L5 band is disabled on the mosaic-X5. To enable reception of the L5 band on the mosaic-X5, we have provided instructions on the Enable the L5 Band page.

Expected Accuracy

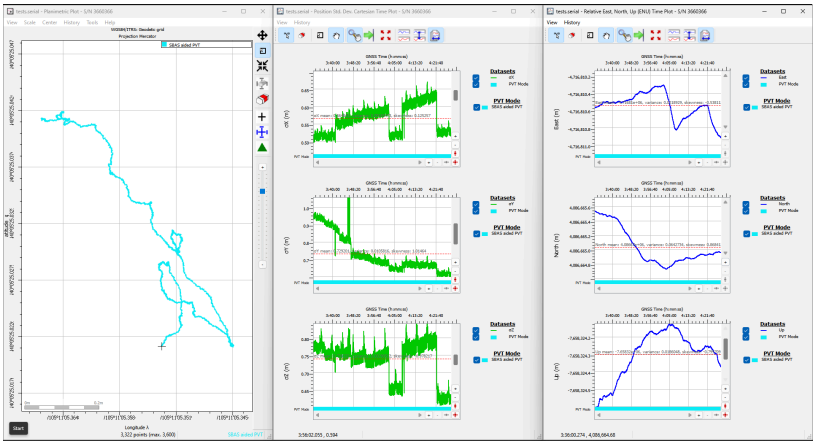
Here is a reference of the accuracy that can be expected from the SXM-E at a stationary position in `Rover` mode.

Factory Settings | **L5 Band Enabled** | **RTK Corrections** | **RTK & L5 Band**

With the factory settings for the mosaic-X5, our location had a standard deviation of σ_x = 0.499m and σ_y = 0.595m



A summary of the GPS signals and the standard deviation of the antenna's location.



Data recorded from the SXM-E with the default/factory settings.

Conditions: L1/L2/L5 band antenna, statically mounted to the rooftop of our building, featuring a significant ground plane, with no obstructions for multi-path signals. The SXM-E was connected with a low-loss cable, fully powered at 5V, and default settings.

Reprogramming the ESP32

In case users accidentally reprogram or corrupt the flash memory on the ESP32, the factory firmware is available in our GitHub repository.

The batch files in the linked directory are used to flash our factory firmware onto the ESP32 in the SXM-E. The batch file relies on the tools from the Arduino IDE and the ESP32 Arduino core and therefore, they will need to be installed before the batch file can be executed.

NOTE

The batch file is only compatible with Windows computers. For Linux and MacOS, users will need to manually flash the binary files onto the ESP32.

NOTE

To compile the firmware, users will need to install the ESP-IDF, Espressif's development framework for the ESP32. However, the compilation instructions are outside the scope of our support for this product. However, users can refer to the programming guide for the ESP-IDF to figure out how to compile the firmware.

mosaic-X5 Security Credentials

WARNING

Beginning with firmware `v4.15.0+`, Septentrio has introduced a mandatory user login requirement. For more details, please refer to the following articles:

- Cybersecurity guidelines: general
- Cybersecurity guidelines: Log-in procedure

FACTORY CREDENTIALS

- Username:

```
RxAdmin
```

- Password:

```
S3pt3ntr10
```

NOTE

Once configured, the users credentials will be stored in memory. These settings will be

NOTE

The serial ports (COM and USB-COM) remain accessible by default, without login credentials.

DANGER

This access can be restricted to logged-in users, if enhanced physical access control is required. However, we recommend absolute caution before implementing these access control restrictions.

We do not have any methods to restore the credentials to their factory defaults, if users can't access the serial ports.

Enclosure Disassembly

Due to the ESD sensitivity of the mosaic-X5 module, we don't recommend disassembling the SXM-E. However, if users must access the PCB to troubleshoot an issue, make a modification, or repair a component, we highly recommend that they take the necessary ESD precautions to avoid damaging the mosaic-X5 module.

READ BEFORE HANDLING PCB!

ESD SENSITIVITY

The mosaic-X5 module is sensitive to ESD. Use a proper grounding system to make sure that the working surface and the components are at the same electric potential.

ESD PRECAUTION

As recommended by the manufacturer, we highly encourage users to take the necessary precautions to avoid damaging their module.

- The PCB is not grounded through the enclosure.
- The PCB features ESD protection on the USB-C connectors and ethernet jacks.
- The mosaic-X5 module also has ESD protection from the `ANT_1` antenna input.

iFixit Anti-Static Wrist Strap

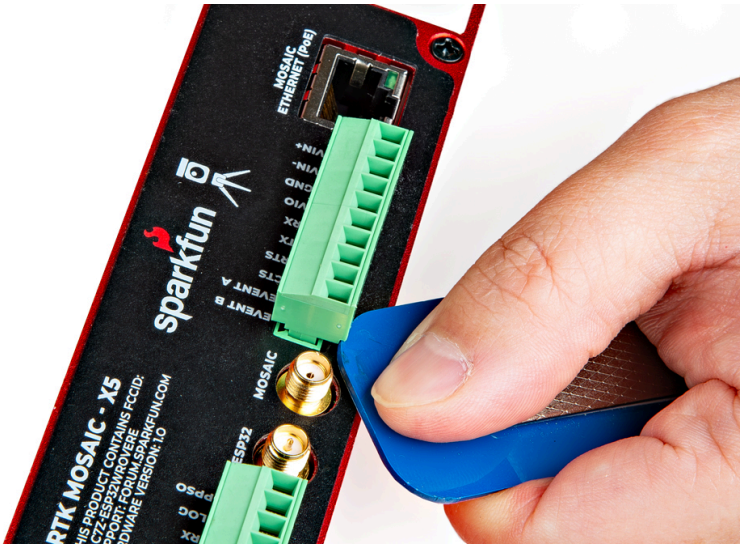


Purchase from SparkFun

Before disassembling the enclosure, users should disconnect the power and all cables from the SXM-E. This will prevent users from creating any shorts and facilitate the teardown process further on.

The internal PCB is only held in place by the front and rear panels of the enclosure. While users can remove the panels first, we recommend that users disconnect the pluggable screw terminal blocks first. Users may find it more difficult to separate the pluggable terminal block from the PCB later in the disassembly process.

1. Users may find it easiest to disconnect the pluggable terminal blocks first; as opposed to later in the teardown process. Users can wiggle or use a soft/rigid object to carefully pry the terminal block off.



Carefully pry the pluggable terminal blocks off.

> **TIP**
>
> In the picture above, a plastic name tag (~1.5mm thick) is used to carefully pry the terminal block up. We have also found the edge of a PCB ruler works great too.

2. Once the pluggable terminal blocks have been removed, users can remove the front and rear panels of the enclosure. They are held in place with four, M3, Phillips head screws on the corners of each panel.

> **TIP**
>
> We recommend removing the front panel first to prevent the Qwiic cable from being yanked off the OLED display or main PCB. Once the front panel is free, carefully lift the panel from the enclosure and disconnect the Qwiic cable from the top connector on the OLED display.

3. At this point, if users have previously disconnected all the cables and the pluggable terminal blocks from the back, the SXM-E PCB should slide out of the enclosure. Users can then, remove the rear panel from the enclosure to complete the teardown process.

To re-assemble the SXM-E, simply reverse the process above.

Resources

Product Resources

A list of resources for the SXM-E and its components

Product Specifications

The hardware specifications for the device

Product Comparison

A comparison of similar products utilizing the Septentrio mosiac-T GNSS module

Product Resources

Product Resources

- Product Page
- Quick Start Guide
- Design Files:
  - Schematic
  - Board Dimensions
  - Enclosure Dimensions
  - ESP32 Firmware
- mosaic-X5 Datasheet
- SFE Product Showcase
- Hardware Repo

Additional Resources

- SparkPNT Forum

Manufacturer's Resources

Septentrio: mosaic-X5

Septentrio also provides great resources for the mosaic-X5:

- mosaic-X5 Product Page
  - Resources: Datasheet, firmware, manual, software tools, USB driver, etc.
- Support Portal:
  - Septentrio Knowledge Base
- Septentrio GitHub Organization:
  - mowi (mosaic wireless) open-source design
  - Set Up Guide to use Third Party OSR and SSR correction services
  - Using Septentrio's Receiver and uBlox's Pointperfect correction services for precise positioning
  - Using Septentrio's Receiver and Swift Navigation's Skylark correction services for precise positioning

- YouTube Channel
- JammerTest - Anti-Jamming and Anti-Spoofing Performance

> **WARNING**
>
> Beginning with firmware `v4.15.0+`, Septentrio has introduced a mandatory user login requirement. For more details, please refer to the following articles:
>
> - Cybersecurity guidelines: general
> - Cybersecurity guidelines: Log-in procedure

Recommended Articles

General Knowledge

- mosaic - Getting started
- Resetting a receiver
- Resetting and clearing the non-volatile memory
- Generating a diagnostic report of a Septentrio receiver
- [How to upgrade the firmware of a Septentrio receiver](https://customersupport.septentrio.com/s/article/How-to-upgrade-the-firmware-of-a-Septentrio-receiver)
- RTCMv3 messages supported by the Septentrio receivers
- Interference Detection, Mitigation, and Flagging

Providing Corrections

- How to set up a base station
- RTK positioning with a moving baseline
- Set up an (internal) NTRIP caster on a Septentrio receiver

Applying Corrections

- Receive corrections over IP
- Receive corrections via NTRIP
- Using uBlox's correction services for precise positioning

Antenna Corrections

- Antenna calibration
- Applying ARP/APC offsets

Detecting and Mitigating Interference

- How to Activate Wideband Interference Mitigation
- How to mitigate interference with notch filters
- Pulse-Blanking
- Front-end Saturation Caused by Interference
- Visualizing interference using the spectrum plot
- SBF Analyzer: Carrier to Noise

Recommended Videos

- Getting started with the mosaic receiver module
- How to upgrade the firmware of a Septentrio receiver
- Output NMEA data on the mosaic receiver module
- Receiving NTRIP corrections on the mosaic module
- Logging data to the Septentrio mosaic receiver module
- Receiving corrections over an IP connection
- Share the USB internet connection on your receiver
- Septentrio's Agnostic GNSS Corrections program
- Test-run for Septentrio's anti-jamming feature
- How to Activate Wideband Interference Mitigation
- How to set a notch filters
- How to use a spectrum plot
- mosaic - GNSS module receiver range from Septentrio

Espressif: ESP32-WROVER

Espressif also provides great resources for their ESP32 module:

- ESP32 Product Resource Page
- Espressif's Forum for the ESP32
- Espressif GitHub Organization
  - ESP32 Arduino Core
    - `.json` file needed for Epressif's ESP32 Arduino Core:

      ```
      https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
      ```

  - ESP-IDF -- IoT Development Framework
    - Programming Guide

Product Specifications

Below are the full specifications for this device:

Features

- Septentrio mosaic-X5 multi-constellation, multi-frequency GNSS Receiver
  - Highly secure against jamming and spoofing with AIM+ unique interference mitigation technology
  - Update rate up to 100 Hz
  - 448 hardware channels for simultaneous tracking of all visible supported satellite signals:
    - GPS: L1C/A, L1PY, L2C, L2P, L5
    - GLONASS: L1CA, L2CA, L2P, L3 CDMA
    - Galileo: E1, E5a, E5b, E5 AltBoc, E6
    - Beidou: B1I, B1C, B2a, B2b, B2I, B3
    - QZSS: L1C/A, L1 C/B, L2C, L5
    - NavIC: L5
    - SBAS: Egnos, WAAS, GAGAN, MSAS, SDCM (L1, L5)
    - On-module L-band
  - Accuracy:
    - Standalone:
      - Horizontal: 1.2m
      - Vertical: 1.9m
    - RTK:
      - Horizontal: 6mm + 0.5ppm
      - Vertical: 10mm + 1ppm
  - Time to Fix: <45s
  - Operating temperature: -40 to 85 °C
  - USB-C interface (UART and Ethernet-over-USB)
- ESP32-WROVER processor
  - Flash: 16MB
  - PSRAM: 8MB
  - USB-C interface (UART via CH340)
- Enclosure
  - Materials: Extruded Aluminum
  - Three Push Buttons

    - Six LED Indicators
    - Two USB-C Ports
    - microSD Slot for Data Logging
        - Connected directly to the mosaic-X5 for fast data logging
    - OLED display
        - 128x64 pixels
- Dimensions: 180.6 x 101.8 x 41mm (7.1 x 4.0 x 1.6in)
- Weight: 415.15g (0.92lbs)

Connections

- SMA Connections:
    - GNSS Antenna (L1/L2/L5/L-Band) - provides 3.3V for an active antenna
    - WiFi + Bluetooth Antenna (2.4GHz)
- 3.5mm Screw Cage Connections:
    - 9V-36V DC input (isolated)
    - GND
    - mosaic-X5 - 3.3V/5V switchable
        - UART (TX/RX/CTS/RTS): `COM1`
        - Event A/B Inputs
        - PPSO
        - LOG
    - ESP32 - 3.3V/5V switchable
        - UART (TX/RX/CTS/RTS)
            - GPIO: (RX: `32`, TX: `34`, CTS: `35`, RTS: `33`)
        - I^2C (SCL/SDA)
- Ethernet:
    - KSZ8041NLI Ethernet PHY interface
    - 10Base-T / 100Base-TX with auto-negotiate and Auto MDI/MDI-X
- Power Options:
    - USB-C Ports
    - Power-over-Ethernet (PoE)
    - 9V-36V DC input (fully isolated)

Antennas

- GNSS Survey Antenna
    - Frequency Bands: L1, L2, L5, L6
    - Gain: ≥2.5dBi

    - APC (NGS Calibrated: ANTEX, ANTINFO):
        - L1: 48.0mm
        - L2/L5: 35.9mm
        - Average: 42.0mm
    - Dimensions: 150 x 58.4mm (5.9 x 2.3in)
    - Weight: 473g (1.05lbs)
- Whip Antenna
    - WiFi/Bluetooth (SPP and BLE)
    - 2.4GHz

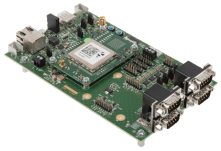
Capabilities

Feature	Enabled	Notes
SBAS	✓	Default
DGNSS Rover	✓	Permitted
DGNSS Base	✓	Permitted
RTK Rover	✓	Permitted
RTK Base	✓	Permitted
RTCM v2.3	✓	Depends on RTK/DGNSS permissions
RTCM v3.x	✓	Depends on RTK permission
CMR v2.0	✓	Depends on RTK/DGNSS permissions
Time Sync	✗	Not permitted
xPPS Output	✓	Default
Event tracking	✓	Permitted
Logging on disk(s)	✓	Permitted with advanced options
APME	✓	Permitted

Feature	Enabled	Notes
RAIM	✓	Permitted
Moving Base	✓	Permitted
Measurement Availability	✓	Available
Interference Mitigation	✓	Permitted
DSK1	✓	
Frequency Sync	✗	Not permitted
PPP	✗	Not permitted
- PPPAtomiChron	✗	
Augmentation Data Service	✗	None
- LBAS2	✗	

Product Comparison

Below is a simple comparison table between our mosaic-X5 products and Septentrio's development and evaluation kits:

	mosaic-T Development Kit	**mosaic-go Evaluation Kit**	**mosaic-X5 GNSS Breakout**	**SXM-E**	**mosaic-X5 Flex Module**	**RTK Facet mosaic**
						
GNSS Antenna	Dual	Single (mosaic-X5) Dual (mosaic-H)	Single	Single	Single	Integrated
USB Connector	micro-B	micro-B	Type-C	Type-C	N/A*	Type-C
Ethernet	Yes *10/100 Base-T*	No	No	Yes *10/100 Base-T*	2x10 Header*	No
WiFi	No	No	No	Yes - Network Bridge *10 Base-T*	No	Yes - Network Bridge *10 Base-T*

COM Ports	4	2	4	1 - mosaic-X5 1 - ESP32	4	1 - mosaic-X5 1 - ESP32
µSD Card Slot	Yes	Yes	Yes	Yes	2x10 Header*	Yes
Reset/Log Buttons	Yes	No*	Yes	Yes	No	Yes
Logic-Level	1.8V 3.3V	3.3V	3.3V	3.3V 5V	3.3V	3.3V
PPS Signal	Header Pin	6-Pin JST Connector	SMA Connector	Screw Terminal	2x10 Header*	No
Enclosure Material	N/A	Metal	N/A	Aluminum	N/A	Plastic
Dimensions		71 x 59 x 12mm ± 1mm	70.9 x 50.8 x 8mm	180.6 x 101.8 x 41mm *Enclosure Only*		
Weight		58g ± 1g	22.6g	415.15g *Enclosure Only*		

MOSAIC-GO EVALUATION KIT

The reset pin is exposed on 4-pin JST connector and the log pin is connected to the latch pin of the SD card slot.

MOSAIC-5 FLEX MODULE

SparkPNT GNSS Flex modules are modular, *plug-in* boards that utilize a *carrier* board to access the pins of the GNSS Flex headers.