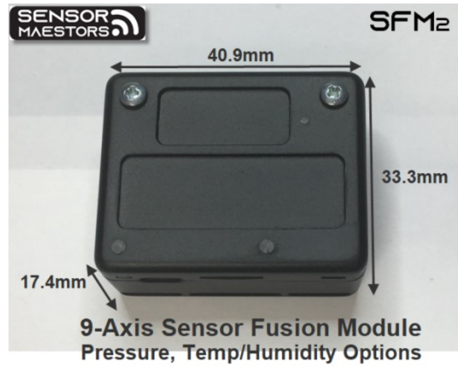
Sensor Maestros

www.sensormaestros.com

SFM2 Sensor Fusion Module User Manual

The SFM2 is the smallest form-factor and lowest power wireless 9 DOF Sensor Fusion Module with optional Pressure, Temperature and Humidty sensors. The SFM2 offers both a Bluetooth Low Energy and a USB/COM port interface. The SFM2 provides an extremely flexible BLE and USB interfaces and allows complete customization of sensor output streams and sensor configurations. **The SFM2 can easily be worn using the Velcro strap ran through slot in the bottom of the enclosure.**



Shown with Black Enclosure. For other color options please inquire

1.0 Product Highlights

- 40.9 x 33.3mm (1.61” x 1.31”) Form Factor
- Wearable: Enclosure has slot for Velcro strap.
- USB and BLE Interfaces: **Connect 6+ SFM2’s running 208Hz Fusion ODR and Quaternion data with newer iOS and Android Devices.**
- Field Updatable using ‘Over the Air’ update
- 9 DOF: 3-Axis Gyro, Accel, Mag
  - Optional Pressure, Temp, and Humidity
- Up to 833Hz Sensor Fusion Operation. Multiple Streams can be enabled at same time.
  - Tared & Un-Tared Quaternion Output
  - Fusion Compass Heading and Tilt Output
  - Fusion Linear Acceleration Output
  - Fusion Euler Angle Output
- Sensor Fusion Output via BLE or USB Interface
  - 833, 417, 208, 104, 52, 26Hz ODR Rate Selections
- Raw Sensor Output availability for all sensors.
- USB(1Mbps) and BLE Interfaces(DLE and 2Mb PHY capability)
- iOS and Android App for sensor and data output configuration.
- USB/COM command set for sensor and data output configuration.
- Python API Library to quickly and easily get application up and running
- TARE function to zero orientation for quaternion output.
- GLOBAL REFERENCE enable/disable command
- TIMESTAMP sensor data with 25us resolution and user definable OFFSET for multiple sensor and multiple SFM2 synchronization.
- Rechargable Battery (typical 250mAh)
  - ~156hour battery life with Quaternion output @ 26Hz using BLE interface
  - ~100hour battery life with Quaternion output @ 104Hz using BLE interface
  - ~58hour battery life with Quaternion output @417Hz using BLE interface
- Individual Control of BLE Output/Notification for every Sensor Output
  - Every Sensor Output has its own characteristic/Notification
  - Sensors and Sensor Fusion can be completely customized including Sensor Filters.
- Custom OEM/Logo Enclosure available
- Example Application Software

2.0 Table of Contents

3.0 Custom Development and OEM Solutions

Sensor Maestros can provide customized solutions using the SFM2 as a foundation. Below are some examples of the common customizable options for the SFM2.

- Customized Hardware
  - Additional Sensors
    - Temperature/Humidity
    - Gas: CO2, CO, VOC's
    - PIR
  - Customized Battery options
- Custom Motion Algorithms
  - Pattern Recognition
  - Machine Learning
  - Vibration Analysis
  - Fitness/Sports Analysis
- Custom Software Applications
  - iOS
  - Android
  - PC Applications
- Custom Cloud Applications
- Custom Enclosure Options

4.0 Future Options

- SENSR-POD
  - BLE to Cellular Bridge
- Bluetooth Long-Range:
  - 2400m+ between SFM2 and SENSR-POD
  - 3000m+ with external antenna(~5-6dbi) between SFM2 and SENSR-POD
- On-board Data-Logging
- Pattern Recognition
- Vibration Analysis
- Cloud Enablement

5.0 SFM2 Hardware Description

The SFM2 is comprised of a 64MHz Cortex M4F BLE SOC, Motion Sensors, and optional Pressure, Relative Humidity and Temperate Sensors. A block diagram is shown below with further details below.

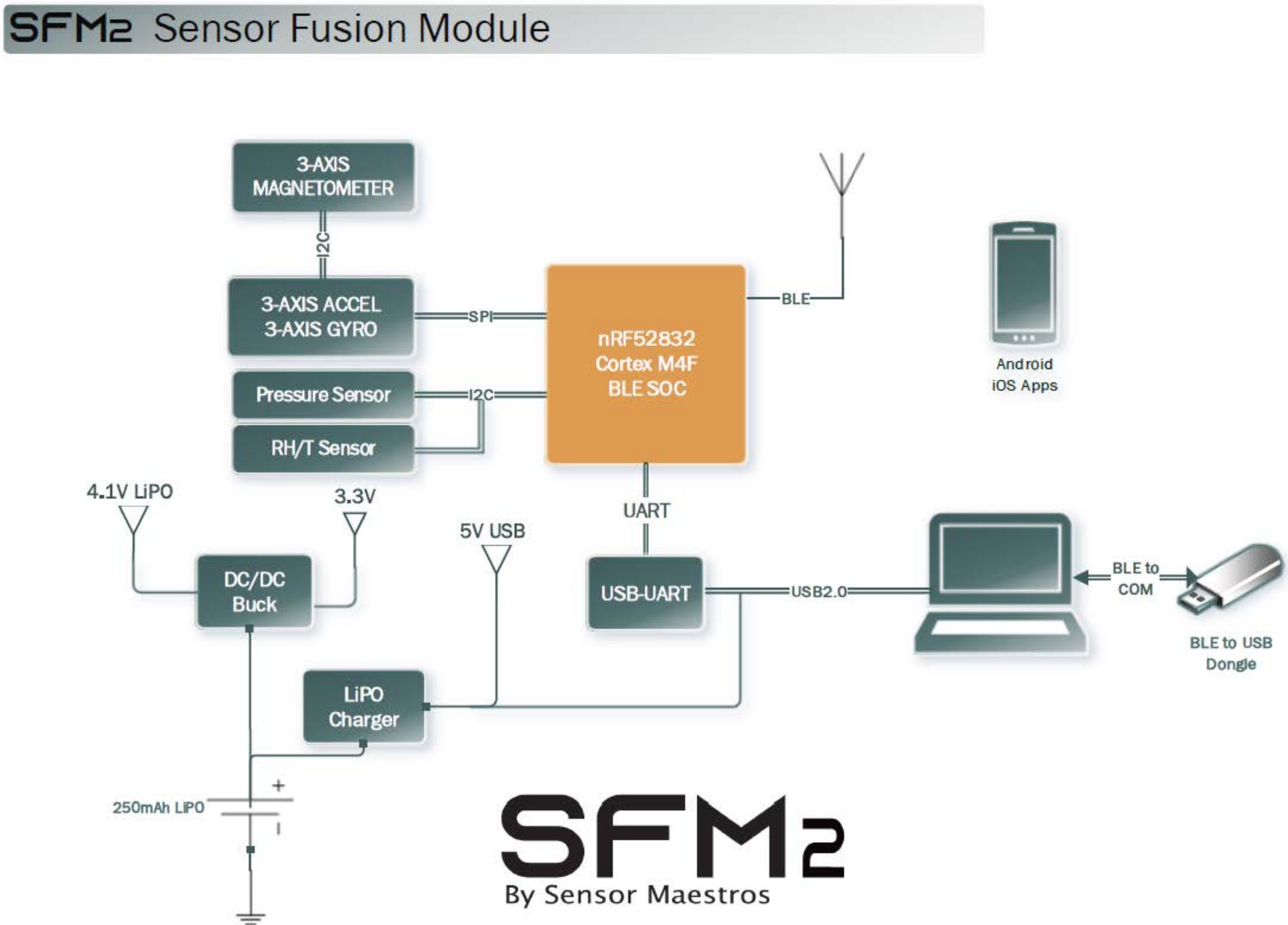


Figure 1: SFM2 Block Diagram

- 64MHz Cortex M4F MCU
- 9DOF Sensors
  - LSM6DSO – 6 Axis Accelerometer/Gyroscope
  - LIS2DML – 3 Axis Magnetometer
- Optional Environmental Sensors
  - LPS22HH – Pressure Sensor
  - ENS210 – Relative Humidity and Temperature Sensor
- BLE interface to provide easy means for application developers to make use of the SFM2 for developing custom applications.
- Example Software provided for BLE Client side for how to interface to the SFM2
  - Example Python 3D Cube application used to interface to the SFM2 from a USB/CDC Connection.
    - Python API Library
  - Native C++ Library(coming soon)
- USB2.0 Support:
  - SFM2 can be controlled/configured from a COM Terminal program using Serial Commands described in 3.0 Serial Commands.
  - Sensor Fusion Data can be sent via USB2.0
  - USB provides Battery charging
- Highly efficient BLE/Characteristic architecture
  - Every Sensor has its own Sample Rate, Full-Scale Range, Data Enable
  - Every Sensor has a Notification Enable

6.0 ACCEL/GYRO/SENSOR FUSION Specifications

Symbol	Parameter	Test conditions	Min.	Typ.(1)	Max.	Unit
FS_ODR	Sensor Fusion Output Data Rate		12.5		833	*Hz*
LA_FS	Linear acceleration measurement range			±2		*g*
				±4		
				±8		
				±16		
G_FS	Angular rate measurement range			±125		dps
				±250		
				±500		
				±1000		
				±2000		
LA_So	Linear acceleration sensitivity(2)	FS = ±2 *g*		0.061		mg/LSB
		FS = ±4 *g*		0.122		
		FS = ±8 *g*		0.244		
		FS = ±16 *g*		0.488		
G_So	Angular rate sensitivity(2)	FS = ±125 dps		4.375		mdps/LSB
		FS = ±250 dps		8.75		
		FS = ±500 dps		17.50		
		FS = ±1000 dps		35		
		FS = ±2000 dps		70		
LA_SoDr	Linear acceleration sensitivity change vs. temperature(4)	from -40° to +85°		±0.01		%/°C
G_SoDr	Angular rate sensitivity change vs. temperature(4)	from -40° to +85°		±0.007		%/°C
LA_TyOff	Linear acceleration zero-g level offset accuracy(5)			±20		m*g*
G_TyOff	Angular rate zero-rate level(5)			±1		dps
LA_OffDr	Linear acceleration zero-g level change vs. temperature(4)			±0.1		m*g*/ °C
G_OffDr	Angular rate typical zero-rate level change vs. temperature(4)			±0.010		dps/°C
Rn	Rate noise density in high-performance mode(6)			3.8		mdps/√Hz
RnRMS	Gyroscope RMS noise in normal/low-power mode(7)			75		mdps
An	Acceleration noise density in high-performance mode(8)	FS = ±2 *g*		70		μ*g*/√Hz
		FS = ±4 *g*		75		
		FS = ±8 *g*		80		
		FS = ±16 *g*		110		

Symbol	Parameter	Test conditions	Min.	Typ.[1]	Max.	Unit
RMS	Acceleration RMS noise in normal/low-power mode[9] [10]	FS = ±2 *g*		1.8		m*g*(RMS)
		FS = ±4 *g*		2.0		
		FS = ±8 *g*		2.4		
		FS = ±16 *g*		3.0		
LA_ODR	Linear acceleration output data rate			12.5 26 52 104 208 416 833 1666		Hz
G_ODR	Angular rate output data rate			12.5 26 52 104 208 416 833 1666		

1. *Typical specifications are not guaranteed.*
2. *Sensitivity values after factory calibration test and trimming.*
3. *Subject to change.*
4. *Measurements are performed in a uniform temperature setup and they are based on characterization data in a limited number of samples. Not measured during final test for production.*
5. *Values after factory calibration test and trimming.*
6. *Gyroscope rate noise density in high-performance mode is independent of the ODR and FS setting.*
7. *Gyroscope RMS noise in normal/low-power mode is independent of the ODR and FS setting.*
8. *Accelerometer noise density in high-performance mode is independent of the ODR.*
9. *Accelerometer RMS noise in normal/low-power/ultra-low-power mode is independent of the ODR.*
10. *Noise RMS related to BW = ODR/2.*

7.0 Magnetometer Specifications

Symbol	Parameter	Test conditions	Min.	Typ.	Max.	Unit
FS	Magnetic dynamic range			±49.152		gauss
So	Sensitivity		-7%	1.5	+7%	mgauss/ LSB
TcyOff	Magnetic Sensor Offset	With offset cancellation	-60	1.5	+60	mgauss
RMS	RMS Noise	High Performance Mode		3		mgauss (RMS)

8.0 Pressure Sensor Specifications

Symbol	Parameter	Test conditions	Min.	Typ.	Max.	Unit
P	Pressure Operating Range		260		1260	hPa
P_{bits}	Pressure bits of data		-7%	24	+7%	mgauss/ LSB
P_{sens}	Pressure Sensitivity			4096		LSB/hPa
P_{AccRel}	Relative Accuracy over pressure	P = 800-1100 hPa T = 25°C		±0.025		hPa
P_{AccT}	Absolute accuracy over temperature	Pop, T = -20 to 80°C		±0.5		hPa
ODR_{Pres}	Pressure output data rate			1 10 25 75 100 200		Hz

9.0 Relative Humidity Sensor Specifications

Symbol	Parameter	Test conditions	Min.	Typ.	Max.	Unit
P	Pressure Operating Range		260		1260	hPa
P_{bits}	Pressure bits of data		-7%	24	+7%	mgauss/ LSB
P_{sens}	Pressure Sensitivity			4096		LSB/hPa
P_{AccRel}	Relative Accuracy over pressure	P = 800-1100 hPa T = 25°C		±0.025		hPa
P_{AccT}	Absolute accuracy over temperature	Pop, T = -20 to 80°C		±0.5		hPa
ODR_{Pres}	Pressure output data rate			1 10 25 75 100 200		Hz

10.0 Temperature Sensor Specifications

Symbol	Parameter	Test conditions	Min.	Typ.	Max.	Unit
P	Pressure Operating Range		260		1260	hPa
P_{bits}	Pressure bits of data		-7%	24	+7%	mgauss/ LSB
P_{sens}	Pressure Sensitivity			4096		LSB/hPa
P_{AccRel}	Relative Accuracy over pressure	P = 800-1100 hPa T = 25°C		±0.025		hPa
P_{AccT}	Absolute accuracy over temperature	Pop, T = -20 to 80°C		±0.5		hPa
ODR_{Pres}	Pressure output data rate			1 10 25 75 100 200		Hz

11.0 LED Operation Description

COLOR	FUNCTION	NOTES
YELLOW	Connected	Will blink twice every ~2.5 seconds to indicate SFM2 is connected to a BLE Client
YELLOW	Advertising	Will blink once every ~1 second to indicate SFM2 is in advertising mode
RED	Sensor/Sensors Enabled	Connected: Blink twice every ~2.5 seconds Unconnected: Blink once every ~1 second This LED is sequenced with the YELLOW LED if a Sensor is Active/ON.
BLUE	Active Battery Charging	Solid BLUE Led indicates battery is charging via USB.

Table 1: LED Operation

12.0 BLE Operation

The SFM2 provides BLE Peripheral(slave) operation and allows any BLE Client(master) to connect to the SFM2. The SFM2 does not require Bonding. Description of operation is provided below.

1) **BLE ADVERTISING:** Upon powering the device the SFM2 Performs BLE Advertising and advertises the Device name "SFM2".
    a. **YELLOW BLE LED** Blinks at ~1 second interval
    b. Advertising Interval = 500ms
    c. Currently the device does not allow user adjustment of the Advertising Interval.

    NOTE: The Device Advertises as a Connectable BLE Peripheral and does not require any Bonding/Secure Pairing.
2) **BLE CONNECTED:** Once a BLE Client has connected to the SFM2 the device will transition to what is shown below and allows the BLE Client to read all the Services, Characteristics, and Descriptors to allow an application to be developed.
    a. **YELLOW BLE LED** Blinks at ~2.5 second interval
    b. All BLE Services, Characteristics, Descriptors are available to be read from the Client.
    c. The SFM2 Max/Min Connection interval is set to 7.5ms for Maximum throughput via a BLE Connection. Not all BLE Clients in particular Mobile Devices will allow for a 7.5ms Connection interval. Sensor Sample Rates and Sensor Fusion Output Rates should be configured according to the maximum throughput allowed by the Client BLE device.
3) **BLE DISCONNECTED:** Upon a BLE Client(master) disconnecting from the SFM2 the device will return to the **BLE ADVERTISING** mode.

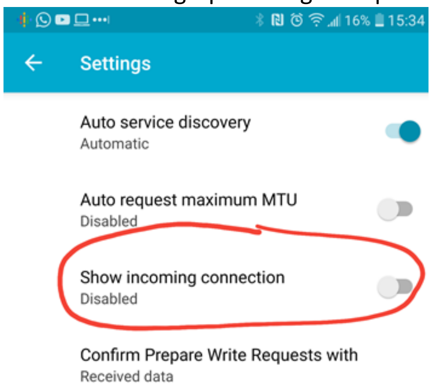
13.0 SFM2 Firmware Update

The SFM2 embedded firmware is not available in source code as open source. Licensing can be made available upon requests. Developers/users of the SFM2 can make use of the USB Commands and/or the BLE Characteristics to configure and retrieve data from the SFM2. Sensor Maestros can provide design services for customization of the embedded firmware for specific use cases on a case by case basis. The SFM2 firmware can be updated via a BLE Connection. Further details/instructions are provided in the "SFM2 Fimware Updates" section of this User Manual.

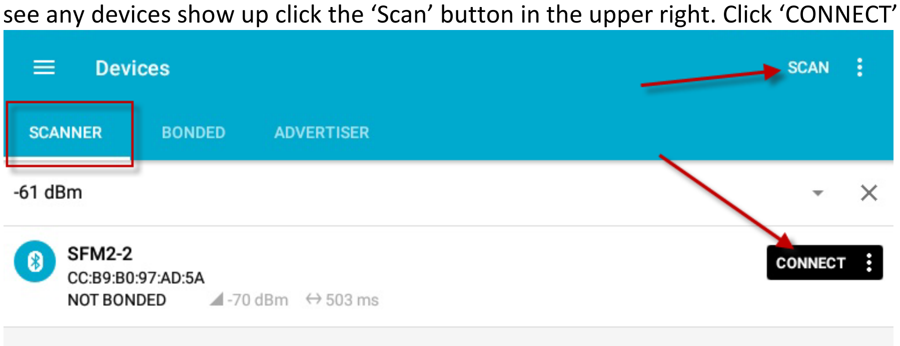
BLE Update: The SFM2 can be updated via a BLE connection using the nRF Connect mobile app. This is a free app that Nordic Semiconductor provides and is available for both Android and iOS platforms. To perform an update you will need to have a OTA(Over the Air) Zip file which contains the firmware update. All firmware updates are provided and available on the Google Drive that Sensor Maestros will provide access to upon a purchase of the SFM2. The steps for performing this update are very simple and outlined below.

1) Download the 'nRF Connect' App for your preferred mobile OS.
    a. Android: Google Play Store: https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp
    b. iOS: Go to the App Store and search for nRF Connect

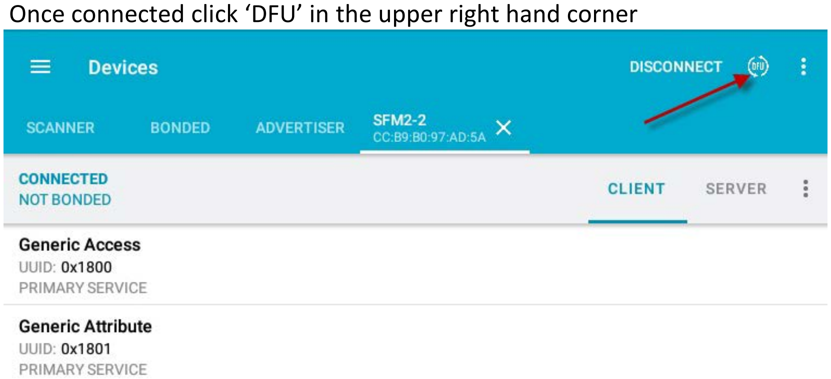
2) IF YOU ARE USING THE ANDROID VERSION!! After installing nRF Connect open the app and first go into Settings->Connectivity and disable ‘Show incoming connection’ as shown below. This will aid in allowing nRF Connect to show a smooth graph during the update. NOTE: Settings are accessed from the Side Nav bar



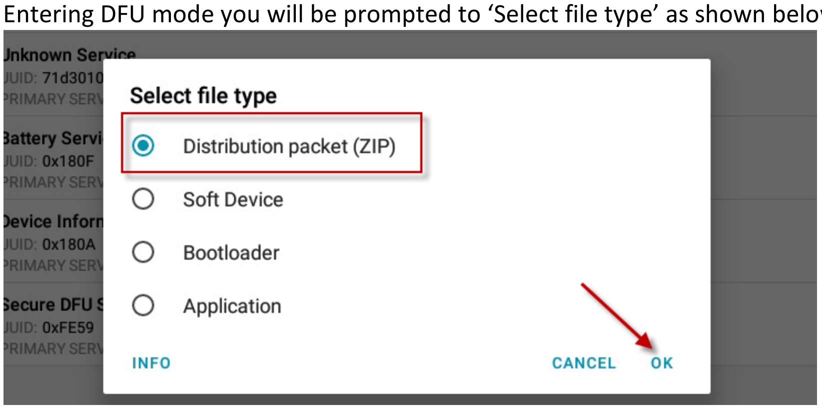
3) After installing nRF Connect open the app and go to the ‘Scanner’ tab and look for the SFM2 as shown below. If you do not see any devices show up click the ‘Scan’ button in the upper right. Click ‘CONNECT’



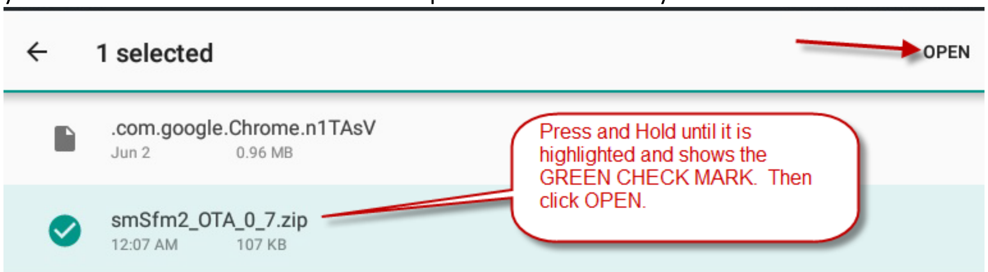
4) Once connected click ‘DFU’ in the upper right hand corner



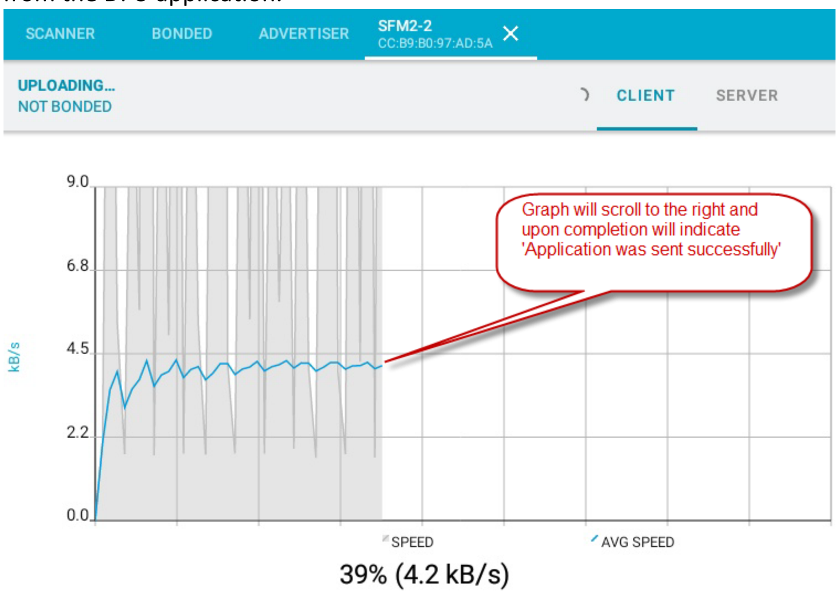
5) Entering DFU mode you will be prompted to ‘Select file type’ as shown below. Select ‘Distribution packet(ZIP) and click OK.



6) Now select the OTA Zip file that you have been provided or downloaded from Sensor Maestros. Typically if you have downloaded this from a Google Drive or accessed it from email it is in the 'Downloads' folder on Android devices. On iOS you will have to have stored this OTA Zip file somewhere on your device or be able to access it from your iCloud folders.



7) You will now see a screen that shows a graph as the SFM2 begins to receive the firmware update. This process can take a bit of time. Upon completion you will see a pop-up indicating 'Application was sent successfully' and will then Disconnect from the DFU application.



8) You can now disconnect the SFM2 from the nRF Connect App
9) Upon connecting the SFM2 to the Sensor Maestros smSENSR mobile App it will display the new firmware version.

14.0 Sensor Fusion

The SFM2 provides for 9DOF Sensor Fusion output that can be enabled by the user. The following Sensor Fusion output streams are provided by the SFM2 and can be individually enabled/disabled using either the BLE or USB interface.

- Quaternion
- Tared Quaternion
- Compass Heading and Tilt
- Linear Acceleration
- Euler Angle

Sensor Fusion output rates (ODR) can be selected by the user from 0(Disabled), 12.5, 26, 52, 104, 208, 417, and 833Hz.
The Sensor Fusion ODR(SFOR) can be set to an ODR rate equal to the highest ODR rate of either the GYRO or ACCEL. For example if 417Hz SFOR ODR rate is desired either the ACCEL or GYRO or both should be configured also to 417Hz. For optimum results it is recommended to set the ACCEL & GYRO ODR rates to 2X the SFOR ODR rate. Technically the SFOR ODR can be greater than the ODR rate of either the ACCEL or GYRO but not both. This configuration is NOT recommended though. The Magnetometer does not need to be ODR rate correlated with the Sensor Fusion ODR rate though for any Sensor Fusion ODR rates 104Hz or greater it is highly recommended to set the Magnetometer to its max value of 104Hz. The MAG ODR rate does have a requirement that the ODR rate can't be greater than both the ACCEL and GYRO. An acceptable configuration is shown below though NOT recommended.

ASR = 12.5Hz GSR = 208Hz MSR = 104Hz

14.1 Coordinate Systems

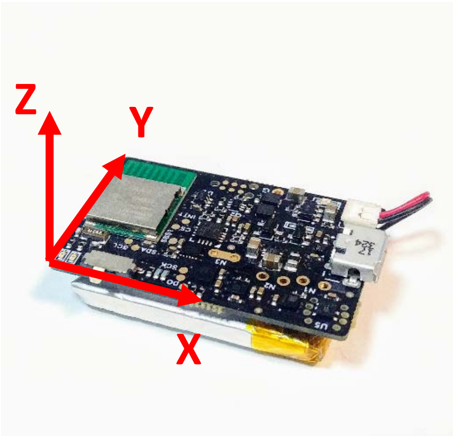
The SFM2 uses two different Coordinate Systems(CS):

- Local Coordinate System(LCS)
- Global Coordinate System(ENU)

The Global Coordinate System is fixed with the external world and it is ENU standard(East—North-Up, acceleration positive) the same that is used in the Android coordinate system. It means that Global Coordinate System axes are oriented as follows:

- X is East
- Y is North
- Z is Up

The LCS is a rigidly fixed coordinate system affixed to the SFM2 device and by default is as shown below:



Using the 'TARE' function this coordinate system can be re-oriented to any position the SFM2 has been moved to and will be effectively Zero'd at the position it is in prior to using the 'TARE' function.

This means that when the SFM2 rotates(changes its orientation) the LCS rotates with it in the exact same manner. All the SFM2 attitude streams except SFQ provide attitude of current LCS provided in ENU. SFQ provides attitude of default LCS given in ENU. The sensor measurement vectors(acceleration, angular rate, magnetic) and SFLA(Sensor Fusion Linear Acceleration) are by default given in LCS but can also be given in ENU by enabling GLOBAL REFERENCE by setting GLOBREF = 1.

14.2 Tare Quaternion

The SFM2 variable SFTARE! provides the ability to change the LCS orientation. Executing SFTARE! creates a new LCS that is the same as ENU in the moment of executing the command. After taring the SFM2 the LCS orientation is exactly the same as ENU so all angles are zero. This allows you to change the default LCS to any other you want. This is **EXTREMELY USEFUL** if the SFM2 has been mounted or moved the user can use the SFTARE! function to re-orient the LCS to this new orientation. This provides an easy means for the user to always start from a known orientation starting point regardless of the physical orientation of the SFM2.

Example: Attach the SFM2 to some piece of equipment that is stationary during the attachment. The orientation of the SFM2 with respect to the equipment doesn't matter. Orient the equipment in such a way that its X axis point East, Y axis points North and Z axis points Up(the orientation of equipment is the same as ENU). With the SFM2 in a stable position execute the SFTARE! command or execute this via the Mobile App. Now the SFM2 outputs are given in the new LCS which is the same as the equipment's coordinate system.

14.3 Headtare

Sometimes it is not practical to rotate the equipment to match the ENU axes for taring. Then you can 'TARE' in any orientation(making sure the equipment is not tilted) and then manually correct the SFM2 using the Headtare function (HEADTARETARE=<angle_in_deg>) according to the current equipment heading angle. The angle is measured between the West-East axis and the equipment axis is in the counter-clockwise manner(looking from above). Headtare allows for a manual alignment adjustment to the SFM2 in degrees.

14.4 Global Reference

To understand the Global Reference option, please get familiar with the 'Coordinate Systems' first being LCS and ENU.

All sensor measurement vectors(acceleration, angular rate, magnetic) and SFLA(Sensor Fusion Linear Acceleration) are by default given in LCS.

Tracking Acceleration Example: It is useful for example when you want to measure the acceleration of your car. You fix the SFM2 to your car, tare and adjust the heading using headtare if needed. Then all the accelerations related to car acceleration or breaking appear on the X axis. When the car accelerates there are positive readings on the X axis, when it is braking there are negative readings on the X axis. It is no matter how the car is oriented in the external world the X axis is always affixed to forward and backward motion. No matter whether you are driving South-North or East-West the reading are always on the X axis.

In the presented above car example using measurements given in the LCS is the best solution because it gives you exactly what you want.

The case is completely different when you want to track the trajectory of a ball.

Tracking Trajectory Example: In this case you would put the SFM2 into the ball and affix it in some manner and throw it. Now you want to draw the flight trajectory. In the simplest form to get the position(relative to the start point)you have to integrate the acceleration twice. It would be easy if the ball would not be spinning. When the ball(and SFM2) is spinning its LCS axis are changing orientation all the time so the acceleration measured on the X axis sometimes refers to North-South movement, sometimes East-West and sometimes Up and Down. It all depends what is the actual SFM2 orientation in the external world at that moment in time. Calculating trajectory using measurements given in LCS are not easy and require a bit of math.

The trajectory calculation would be GREATLY SIMPLIFIED if the measurement would be given in some external(global) coordinate system that is fixed NOT with the SFM2/Ball BUT with the external world. This can be achieved using the 'Global Reference' option by setting GLOBREF=1.

When GLOBREF=1 is set all the measurements are given in the ENU. It means that all the acceleration on the X axis corresponds to East-West movement, on the Y axis to North-South movement and on the Z axis to Up-Down movement no matter what the ball orientation is. Calculating the trajectory using measurements given in ENU is MUCH EASIER because you don't need to take the ball spin into account.

- When GLOBREF=1 all vector outputs(raw sensor data and SFLA) are represented in the global reference frame and are not affected if the SFM2 is 'tared'.
- When GLOBREF=0 and the device is 'tared' all vector outputs are rotated to match the new orientation. This means that after taring the SFM2 the Accelerometer output AD should always read approximately x=0, y=0, z=1, until the orientation is changed.

15.0 USB Commands

This protocol is used by the SFM2 to connect to an external computer via a standard COM port. COM port Baud Rate is 1Mbaud.

15.1 Command Format:

The top-level protocol elements are commands, queries, responses, and data. Commands and responses are of the general form:
FOO=37
where "FOO" is the designator for a specific command or response, "=" is a separator between the designator and parameter(s), and "37" is a parameter.
Designators such as "FOO" in the above example are considered case insensitive. The receiving side MUST accept "FOO", "foo", and even "FOO" interchangeably, e.g., by using the C standard library strcasecmp() or strncasecmp() functions for case insensitive string comparison, or by converting the received characters to a specific case before comparison. Where the designators are programmatically generated, they SHOULD be generated in upper case.

A command such as "FOO=37" will result in a similar response, typically also "FOO=37". However, in a case where the parameter value is not acceptable, the response may have a different parameter value, which is the actual parameter value that will be used henceforth. This may be the previous value of the parameter, or a new value that is an approximation of the requested parameter

value. For instance, if the previous value of the parameter for FOO was 5, and 37 is not an acceptable value, but 35 is acceptable, then a "FOO=37" command might yield a response of "FOO=5" or a response of "FOO=35".
Under some circumstances a command may yield multiple responses. For instance, if the parameters to FOO and BAR are related, such that only certain combinations are permitted, then a command "FOO=37" might yield two consecutive responses "FOO=35" and "BAR=42".

Under some circumstances responses may be generated spontaneously rather than in response to a command. This might happen if a parameter change occurs based on an alternate communication channel. For instance, if the USB CDC channel sends a "FOO=37" command, a "FOO=35" response may be sent both over the USB CDC channel and to the nRF52. This is necessary so that the nRF52 firmware will update the appropriate characteristic values, and send corresponding notifications if enabled.
Queries are similar to commands, but rather than having a "=" followed by parameter(s), they have a "?" and no parameters. The response to a query is formatted in the same way as a command response.
Data is formatted similarly to reponses, but with a ":" separating the designator from the data.

15.2 Special characters:

\r carriage return, 0x13
\n newline, 0x10
All commands and queries MUST be sent with a terminating carriage return. For a command, the carriage return MAY be followed by a line feed (which is highly recommended), but the receiving side **MUST NOT** require the line feed.
Responses and data **MUST** be sent terminated with a carriage return followed by a line feed. The receiving side **MUST NOT** require the line feed. In general, the receiving side of any protocol element should ignore the presence (or absence) of line feed characters.

15.3 Metavariables:

In the descriptions of commands, repsonses, and data, items enclosed in angle brackets, e.g., "<sample_rate-int>" are metavariables. The angle brackets are only for syntactic specification, and are **NOT** actually sent or received on the serial link. Within the metavariable specification, the portion before the hyphen identifies the nature of the data item, and the portion after the hypen indicates the data type.

15.4 Data types:

boolean:
"0" (false) or "1" (true)
int16:
16-bit integer represented as up to five decimal digits, without leading zeros, and with an optional leading "-" for negative values
int32:
32-bit signed integer
uint32:
32-bit unsigned integer can be preceded by a “-“ to indicate negative values
float:
Floating-point value represented as a mantissa portion with optional leading "-" sign and optional "." radix mark, and optional exponent part

15.5 System Commands:

NAME=<=Alpha-Numeric String>
CONFIG?
Shows current configuration of the SFM2 and provides a list of available USB commands.
SFRESET!
Reset sensors to 0Hz ODR including Sensor Fusion output
SELFTEST!
Performs automated Self-Test on Accel, Gyro and Mag sensors.
CALIBSTORE!
Performs SFM2 calibration and stores sensor offsets into Non-Volatile memory. To run the calibration enable each sensor with a >0Hz ODR rate and enable SFOR > 0Hz.
Success => CALIBSTORE=VALID

CALIBCLEAR!

Clears previously stored calibration data.

Success => CALIBSTORE=EMPTY

GLOBREF=<enable-boolean>

Enable/Disable Global Reference Frame

BINMODE=<enable-boolean>

1 = COM Sensor output in Binary

0 = COM Sensor in ASCII format(Default)

15.6 Sensor Configuration Commands:

ASR=<sample_rate-int16>

Set accelerometer sample rate in Hz (0 for sampling disabled)

AFR=<full_scale_range-int16>

Set accelerometer full-scale range in g

AFASTSET=<enable-boolean>

Set accelerometer Fast Settling

ALPF2=<enable-boolean>

Enable Accelerometer 2^{nd} stage Low Pass Filter

ADE=<enable-boolean>

Set accelerometer data output enable

GSR=<sample_rate-int16>

Set gyro sample rate in Hz (0 for sampling disabled)

GFR=<full_scale_range-int16>

Set gyro full-scale range in dps

GDE=<enable-boolean>

Set gyro data output enable

MSR=<sample_rate-int16>

Set magnetometer sample rate in Hz (0 for sampling disabled)

MFR=<full_scale_range-int16>

Set magnetometer full-scale range in uT

MDE=<enable-boolean>

Set magnetometer data output enable

15.7 Sensor Fusion Commands:

SFOR=<fusion_rate-int16>

Set sensor fusion output rate in Hz

SFQDE=<enable-boolean>

Set sensor fusion untared quaternion data output enable

SFQTDE=<enable-boolean>

Set sensor fusion tared quaternion data output enable

SFCHTDE=<enable-boolean>

Set sensor fusion compass heading and tilt data output enable

SFLADE=<enable-boolean>

Set sensor fusion linear acceleration data output enable

SFEADE=<enable-boolean>

Set sensor fusion euler angle data output enable

SFTARE!

Tare sensor fusion

SFTARE=<w>,<x>,<y>,<z>   (coming soon!!)

Set sensor fusion tare quaternion

15.8 Timestamp Commands:

TIME!

Resets the Timestamp value to 0

TIME=<uint32>

Set the Timestamp value to an integer. Can proceed value with "-" sign to indicate negative value.

TOFFSET!

Resets the Timestamp value to 0

TOFFSET=<int32>

Set the Timestamp value to an integer value where 1 LSB = 25us. Can proceed value with “-“ sign to indicate negative value.

15.9 System Queries:

Config?

Shows current configuration of the SFM2 and provides a list of available USB commands.

NAME?

Provides name of SFM2

CALIBSTORE?

Returns

=VALID (Calibration has been performed and stored into NV Memory)

OR

=EMPTY (Calibration has not been performed)

15.10 Sensor Queries:

ASR?

Get accelerometer sample rate

AFR?

Get accelerometer full-scale range

ADE?

Get accelerometer data output enable

GSR?

Get gyro sample rate

GFR?

Get gyro full-scale range

GDE?

Get gyro data output enable

MSR?

Get magnetometer sample rate

MFR?

Get magnetometer full-scale range

MDE?

Get magnetometer data output enable

SSAT?

Get sensor saturation

15.11 Sensor Fusion Queries

SFOR?

Get sensor fusion output rate in Hz

SFQDE?

Get sensor fusion untared quaternion data output enable

SQTDE?

Get sensor fusion tared quaternion data output enable

SFCHTDE?

Get sensor fusion compass heading and tilt data output enable

SFEADE?

Get sensor fusion euler angle data output enable

SFLADE?

Get sensor fusion linear acceleration data output enable

SFTARE?

Get sensor fusion tare quaternion

15.12 Responses:

SRESET=<self_test-bool>
Sensor reset response
ASR=<sample_rate-int>
Accelerometer sample rate in Hz (0 for sampling disabled)
AFR=<full_scale_range-int>
Accelerometer full-scale range in g
ADE=<enable-boolean>
Accelerometer data output enable
GSR=<sample_rate-int>
Gyro sample rate in Hz (0 for sampling disabled)
GFR=<full_scale_range-int>
Gyro full-scale range in dps
GDE=<enable-boolean>
Gyro data output enable
MSR=<sample_rate-int>
Magnetometer sample rate in Hz (0 for sampling disabled)
MFR=<full_scale_range-int>
Magnetometer full-scale range in uT
MDE=<enable-boolean>
Magnetometer data output enable
SSAT=<accel>,<gyro>,<mag>
Sensor saturation
SFOR=<fusion_rate-int>
Sensor fusion output rate in Hz
SFQDE=<enable-boolean>
Sensor fusion untared quaternion data output enable
SFQTDE=<enable-boolean>
Sensor fusion tared quaternion data output enable
SFCHTDE=<enable-boolean>
Sensor fusion compass heading and tilt data output enable
SFEADE=<enable-boolean>
Sensor fusion euler angle data output enable
SFLADE=<enable-boolean>
Sensor fusion linear acceleration data output enable
SFTARE=<w>,<x>,<y>,<z>
Sensor fusion tare quaternion

15.13 Data Stream Format:

AD:<x-int>,<y-int>,<z-int>
Accelerometer data
GD:<x-int>,<y-int>,<z-int>
Gyro data
MD:<x-int>,<y-int>,<z-int>
Magnetometer data
SFQ:<w>,<x>,<y>,<z>
Sensor fusion untared quaternion
SFQT:<w>,<x>,<y>,<z>
Sensor fusion tared quaternion
SFCHT:<hdg>,<tilt>
Sensor fusion compass heading and tilt
SFEA:<roll>,<pitch>,<yaw>
Sensor fusion euler angle data
SFLA:<x>,<y>,<z>
Sensor fusion linear acceleration

16.0 SFM2 BLE Services and Characteristics

All custom service and characteristic UUIDs are 128-bit UUIDs of the form
71D3xxxx-E8E7-4F91-AA3C-4A68051247BC, with unique 16-bit values replacing the "xxxx" in bytes 2 and 3.
In the descriptions below, only the unique 16 bits of the UUIDs are given, between hyphens, expressed in hexadecimal, for example, -0102-represents 71D3***0102***-E8E7-4F91-AA3C-4A68051247BC.

Bluetooth specification, which is not explicitly listed below.
https://developer.bluetooth.org/gatt/descriptors/Pages/DescriptorViewer.aspx?u=org.bluetooth.descriptor.gatt.client_characteristic_configuration.xml
Indications may optionally be supported.
Note: the standard Bluetooth Device Information Service must be implemented to provide the identification strings and hardware/firmware/software versions.
https://developer.bluetooth.org/gatt/services/Pages/ServiceViewer.aspx?u=org.bluetooth.service.device_information.xml

16.1 SFM2 BLE Characteristic Map

The SFM2 has a total of 28 characteristics, 24 of them support notifications.

Name	ID	Read	Notify	Write
SFQ_stream	0x0101		X	
SFQT_stream	0x0102		X	
SFLA_stream	0x0103		X	
SFEA_stream	0x0104		X	
SFCHT_stream	0x0105		X	
AD_stream	0x0106		X	
GD_stream	0x0107		X	
MD_stream	0x0108		X	
SFM_stream	0x0109		X	
Sensor_Fusion_settings	0x0122		X	
Reference_settings	0x0128	X	X	X
Tare	0x0129			X
Time	0x0131			X
Time_offset	0x0132	X	X	X
Name	0x0133	X	X	X
Settings_storage	0x0134			X
Calibration_storage	0x0135			X
Stats	0x0137	X	X	X
Accelerometer_settings	0x0201	X	X	X
Gyroscope_settings	0x0202	X	X	X

Magnetometer_settings	0x0203	X	X	X
SF_Kalman_settings	0x0204	X	X	X
Calibration	0x0206	X	X	
Calibration_settings	0x0207	X	X	X
Self-test	0x0208	X	X	X
RH/T_stream	0x0301		X	
Pressure_stream	0x0302		X	
Env._sensors_settings	0x0310	X	X	X

16.2 Streaming characteristics

Streaming characteristics are used to send data from SFM2. The transmission is enabled by enabling BLE notifications of the *Data characteristic* (writing its CCCD). These characteristics support notifications, but can't be directly read.

To maximize throughput each notification packet can contain multiple samples.

IMU related streams (raw sensor data and SF outputs) are timestamped. The timestamp is a uint32 value denoting time in 25us resolution, 1 LSB = 25us. It can be used to synchronize different streams with each other.

16.2.1 SFQ stream

Untared quaternion data output from Sensor Fusion.

Size		Name	Description
20 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	w	The real component of unit rotation quaternion, Float32
	4 bytes	x	The i vector component of unit rotation quaternion, Float32
	4 bytes	y	The j vector component of unit rotation quaternion, Float32
	4 bytes	z	The k vector component of unit rotation quaternion, Float32

16.2.2 SFQT stream

Tared quaternion data output from Sensor Fusion.

Size		Name	Description
20 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	w	The real component of unit rotation quaternion, Float32
	4 bytes	x	The i vector component of unit rotation quaternion, Float32
	4 bytes	y	The j vector component of unit rotation quaternion, Float32
	4 bytes	z	The k vector component of unit rotation quaternion, Float32

16.2.3 SFLA stream

Linear acceleration data output from Sensor Fusion.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	x	The acceleration in X-Axis in g, Float32
	4 bytes	y	The acceleration in Y-Axis in g, Float32
	4 bytes	z	The acceleration in Z-Axis in g, Float32

16.2.4 SFEA stream

Euler angles data output from Sensor Fusion.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	roll	The roll angle in degrees, NED, Float32
	4 bytes	pitch	The pitch angle in degrees, NED, Float32
	4 bytes	yaw	The yaw angle in degrees, NED, Float32

16.2.5 SFCHT stream

Heading+tilt data output from Sensor Fusion.

Size		Name	Description
12 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	heading	Heading in degrees, Float32
	4 bytes	tilt	Tilt in degrees, Float32

16.2.6 AD stream

Raw accelerometer data.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	x	The acceleration in X-Axis in g, Float32
	4 bytes	y	The acceleration in Y-Axis in g, Float32
	4 bytes	z	The acceleration in Z-Axis in g, Float32

16.2.7 GD stream

Raw gyroscope data.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	x	The angular rate in Pitch axis (around X) in °/s, Float32
	4 bytes	y	The angular rate in Roll axis (around Y) in °/s, Float32
	4 bytes	z	The angular rate in Yaw axis (around Z) in °/s, Float32

16.2.8 MD stream

Raw magnetometer data.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	x	The magnetometer reading in X-Axis in uT, Float32
	4 bytes	y	The magnetometer reading in Y-Axis in uT, Float32
	4 bytes	z	The magnetometer reading in Z-Axis in uT, Float32

16.2.9 SFM stream

Calibrated magnetometer data. Output from Sensor Fusion.

Size		Name	Description
16 bytes	4 bytes	ts	Timestamp, Uint32
	4 bytes	x	The magnetometer reading in X-Axis in uT, Float32
	4 bytes	y	The magnetometer reading in Y-Axis in uT, Float32
	4 bytes	z	The magnetometer reading in Z-Axis in uT, Float32

16.2.10 RH/T stream

Output data from relative humidity and temperature sensor (ENS210).

Size		Name	Description
8 bytes	4 bytes	t	Temperature in °C, Float32
	4 bytes	rh	Relative humidity in %, Float32

16.2.11 Pressure stream

Output data from pressure sensor (LPS22HH).

Size		Name	Description
16 bytes	4 bytes	p	Pressure in hPa, Float32
	4 bytes	t	Temperature in °C, Float32
	4 bytes	alt	Altitude in m, Float32

16.3 Settings Characteristics

Settings characteristics are used to configure the device. They support reading, writing and emit notifications.

16.3.1 Sensor Fusion Settings

Size	Name	Description
1 byte	rate	Data rate

Data rate: Data rate is a single byte setting that is used to configure data rate.

Value	Rate	Value	Rate
0x00	0 Hz	0x05	208 Hz
0x01	12.5 Hz	0x06	417 Hz
0x02	26 Hz	0x07	833 Hz
0x03	52 Hz	0x08	1667 Hz
0x04	104 Hz		

16.3.2 Reference Settings

Size		Name	Description
17 bytes	4 bytes	w	The real component of tare quaternion, Float32
	4 bytes	x	The i vector component of tare quaternion, Float32
	4 bytes	y	The j vector component of tare quaternion, Float32
	4 bytes	z	The k vector component of tare quaternion, Float32
	1 byte	globref	Global reference frame. 1-enabled, 0-disabled

16.3.3 Accelerometer Settings

Size		Name	Description
7 bytes	1 byte	rate	Data rate
	1 byte	full scale	Full scale
	1 byte	lpf2	LPF2 second stage selected, 1-enabled, 0-disabled
	1 bytes	filter	Filter configuration
	1 byte	fast_settling	Filter fast settling. 1-enabled, 0-disabled
	1 byte	power_mode	Power mode
	1 byte	self_test	Self-test disturbance configuration

Data rate: Data rate is a single byte setting that is used to configure data rate.

Value	Rate	Value	Rate
0x00	0 Hz	0x05	208 Hz
0x01	12.5 Hz	0x06	417 Hz
0x02	26 Hz	0x07	833 Hz
0x03	52 Hz	0x08	1667 Hz
0x04	104 Hz		

Accelerometer full scale: Used to configure full scale of accelerometer

Value	Rate
0x00	2 G
0x01	4 G
0x02	8 G
0x03	16 G

Accelerometer filter configuration

Value	Filter	Value	Filter
0x00	HP_PATH_DISABLE_ON_OUT	0x34	HP_REF_MD_ODR_DIV_100
0x10	SLOPE_ODR_DIV_4	0x35	HP_REF_MD_ODR_DIV_200
0x11	HP_ODR_DIV_10	0x36	HP_REF_MD_ODR_DIV_400
0x12	HP_ODR_DIV_20	0x37	HP_REF_MD_ODR_DIV_800
0x13	HP_ODR_DIV_45	0x01	LP_ODR_DIV_10
0x14	HP_ODR_DIV_100	0x02	LP_ODR_DIV_20
0x15	HP_ODR_DIV_200	0x03	LP_ODR_DIV_45

0x16	HP_ODR_DIV_400	0x04	LP_ODR_DIV_100
0x17	HP_ODR_DIV_800	0x05	LP_ODR_DIV_200
0x31	HP_REF_MD_ODR_DIV_10	0x06	LP_ODR_DIV_400
0x32	HP_REF_MD_ODR_DIV_20	0x07	LP_ODR_DIV_800
0x33	HP_REF_MD_ODR_DIV_45		

Accelerometer power mode

Value	Rate
0x00	High performance
0x01	Normal
0x02	Ultra Low Power

Self-test configuration

Self-test disturbance configuration.   Used for both accelerometer and gyro.

Value	Rate
0x00	Off
0x01	Positive
0x02	Negative

16.3.4 Gyroscope Settings

Size		Name	Description
6 bytes	1 byte	rate	Data rate
	1 byte	full scale	Full scale
	1 byte	lpf1	LPF1 configuration
	1 bytes	high_perf	High performance mode. 1-enabled, 0-disabled
	1 byte	hpf	High pass filter configuration
	1 byte	self_test	Self-test disturbance configuration

Data rate: Data rate is a single byte setting that is used to configure data rate.

Value	Rate	Value	Rate
0x00	0 Hz	0x05	208 Hz
0x01	12.5 Hz	0x06	417 Hz
0x02	26 Hz	0x07	833 Hz
0x03	52 Hz	0x08	1667 Hz
0x04	104 Hz		

Gyroscope full scale: Used to configure full scale of accelerometer

Value	Rate
0x00	125 °/s
0x01	250 °/s
0x02	500 °/s
0x03	1000 °/s
0x04	2000 °/s

Gyroscope LPF1 configuration

Value	Rate	Value	Rate
0x00	Ultra Light	0x05	Very strong
0x01	Very Light	0x06	Aggressive
0x02	Light	0x07	Extreme
0x03	Medium	0x08	Off
0x04	Strong		

Gyroscope HPF configuration

Value	Rate
0x00	None
0x80	16 mHz
0x81	65 mHz
0x82	260 mHz
0x83	1.04 Hz

16.3.5 Magnetometer Settings

Size		Name	Description
5 bytes	1 byte	rate	Data rate
	1 byte	temp_comp	Temperature compensation. 1-enabled, 0-disabled
	1 byte	low_power	Low power mode. 1-enabled, 0-disabled
	1 byte	low_pass_filter	Low pass filter. 1-enabled, 0-disabled
	1 byte	self-test	Self-test disturbance enabled. 1-enabled, 0-disabled

Data rate: Data rate is a single byte setting that is used to configure data rate.

Value	Rate
0x00	0 Hz
0x01	12.5 Hz

0x02	26 Hz
0x03	52 Hz
0x04	104 Hz

16.3.6 SF Kalman Settings

Size		Name	Description
20 bytes	4 bytes	qvy	Gyro sensor noise variance units (deg/s)^2
	4 bytes	qvg	Accelerometer sensor noise variance units g^2 defining minimum deviation from 1g sphere.
	4 bytes	qvb	Magnetometer sensor noise variance units uT^2 defining minimum deviation from geomagnetic sphere.
	4 bytes	qwb	Gyro offset random walk units (deg/s)^2
	4 bytes	max_bpl	Maximum absolute permissible power on gyro offsets(deg/s)

16.3.7 Calibration Settings

Size		Name	Description
20 bytes	1 byte	mode	Magnetometer calibration auto storage mode
	1 byte	threshold	Magnetometer calibration auto storage threshold

Magnetometer calibration auto storage mode

Value	Mode	Comment
0x00	Off	Auto storage disabled
0x01	Once	Magnetometer calibration will be stored once.
0x02	Always	Magnetometer calibration will be stored each time new calibration is calculated.

Magnetometer calibration type

Value	Threshold
0x00	None
0x01	4 element
0x02	7 element
0x03	10 element

16.3.8 Env. sensors settings (on –ENV Version)

Size		Name	Description
17 bytes	4 bytes	zero_pressure	Pressure at 0 m. Used to tare altitude, in hPa. Float32.
	4 bytes	alarm_up	High altitude alarm threshold, in m. Set to NAN to disable. Float32.
	4 bytes	alarm_down	Low altitude alarm threshold, in m. Set to NAN to disable. Float32.
	1 byte	lps22hh_rate	LPS22HH data rate
	1 byte	lps22hh_low_noise	LPS22HH low noise mode. 1-enabled, 0-disabled.
	1 byte	ens210_mode	ENS210 sampling mode
	2 bytes	ens210_interval	ENS210 sampling interval. Used only when sensor is in *Interval* sampling mode. In ms. Uint16

LPS22HH data rate

Value	Rate
0x00	0 Hz
0x01	1 Hz
0x02	10 Hz
0x03	25 Hz
0x04	50 Hz
0x05	75 Hz
0x06	100 Hz

ENS210 sampling mode

Value	Mode	Comment
0x00	Off	Sensor disabled
0x01	Interval	Sampling at configured interval
0x02	Continuous	Sampling as fast as possible

17.0 Storage characteristics

Storage characteristic control storing data in non-volatile memory.

17.1.1 Settings storage

Settings storage is a write-only characteristic. It can be written with single byte commands. Each command triggers an action, depending on the value written.

Value	Action
0x01	Store settings in non-volatile memory
0x02	Restore settings from non-volatile memory
0x03	Restore settings to factory defaults. This does not overwrite non-volatile memory.

17.1.2 Calibration storage

Calibration storage is a write-only characteristic. It can be written with single byte commands. Each command triggers an action, depending on the value written.

Value	Action
0x01	Store magnetometer calibration in non-volatile memory
0x02	Clear magnetometer calibration storage.

17.2 Time characteristics

Time characteristics are used to control the IMU timestamp with 25us resolution.

17.2.1 Time

Writing the time characteristic sets time. Internally, it works by modifying the time offset. The characteristic is write-only.

Size	Name	Description
4 bytes	time	Time in 25us resolution. UInt32

17.2.2 Time offset

Time offset characteristic can be used to shift the time by a fixed amount. The time wraps around after reaching its max value. This means that setting the time offset to a large value effectively shifts the time backwards.

Size	Name	Description
4 bytes	offset	Time offset in 25us resolution. UInt32

17.3 Other characteristics

17.3.1 Tare

Tare characteristic can be used to tare the sensor fusion in current orientation. It's write-only.

Size	Name	Description
1 bytes	tare	Tare command. 1 - full tare, 2 - heading tare

17.3.2 Name

Name characteristic can be used to read or configure the device's name. The name gets updated immediately after write.

The name is ASCII encoded, without the null terminator. The characteristic is of variable length, with a maximum length of 16.

17.3.3 Stats

Once every 2 seconds the device's statistics are transmitted through the stats characteristic. It can also be read directly to immediately get the statistics. Writing any value to this characteristics clears the stats.

Size		Name	Description
24 bytes	4 bytes	runtime	Device runtime in ticks (1/1024 s). UInt32
	4 bytes	ble_sent	Count of bytes sent through BLE. UInt32.
	4 bytes	ble_dropped	Count of bytes dropped when sending via BLE. UInt32.
	4 bytes	serial_sent	Count of bytes written to serial. UInt32
	4 bytes	serial_dropped	Count of bytes dropped when writing to serial.
	2 bytes	max_queue	Maximum count of element in IMU queue.
	2 bytes	calibration_stores	The number of times the magnetometer calibration was stored.

17.3.4 Calibration

Calibration characteristic tells the current magnetometer calibration status.

Size		Name	Description
20 bytes	4 bytes	error	Calibration error in %. Float32
	1 byte	type	Calibration type.

Magnetometer calibration type

Value	Threshold
0x00	None
0x01	4 element
0x02	7 element
0x03	10 element

17.3.5 Self-test

Self-test characteristic can be used to trigger and monitor self-test. The characteristic can be read and emits notifications. Writing any value to it starts self-test.

Size		Name	Description
4 bytes	1 byte	phase	Self-test phase
	1 byte	accel	Accelerometer result. (1 - ok, 0 - failure)
	1 byte	gyro	Gyroscope result. (1 - ok, 0 - failure)
	1 byte	mag	Magnetometer result. (1 - ok, 0 - failure)

Self-test phase

Value	Phase
0x00	Never run
0x01	Base measurement
0x02	Positive disturbance measurement
0x03	Negative disturbance measurement
0x04	Completed

18.0 Example Software

Sensor Maestros will provide example software to assist customers getting their applications developed quickly to interface to the SFM2. Example projects are provided for Python and for C++ as indicated below.

18.1.1 Python API Library

The Python API Library can be used with Python 3.7 or higher. It provides an easy means for users to connect to the SFM2 via a COM/USB port and quickly get an application up and running. The Python Library provides high level API functions that utilize the AT Commands. The Python API Library provides numerous examples and maximizes throughput by enabling Binary Data Streams which will allow for multiple SFM2 to be connected simultaneously. It is certainly feasible to connect 5+ SFM2's depending on the exact data streams that have been enabled and Fusion ODR rate. This has also been tested on a Raspberry Pi 4.

- Numerous examples to quickly and easily get started in the /samples folder
- Data-log timestamped sensor streams to CSV file
- COM Port Throughput Test
- SFM2 Self-Test
- Binary Mode to optimize throughput
- Show 3D Cube representation of SFM2
- Provides 4 Standard SFM2 configurations:
    - OFF
    - Low Power (ACC=26Hz, GYRO=26Hz, MAG=26Hz, SFOR=26Hz)
    - Balanced (ACC=104Hz, GYRO=104Hz, MAG=104Hz, SFOR=104Hz)
    - Performance (ACC=833Hz, GYRO=833Hz, MAG=104Hz, SFOR=417Hz)
- Provides means to measure COM port throughput

18.2 Mobile App

The SFM2 has both an Android and an iOS App that can be used to Configure, Test, and Data-log the SFM2. Features of the SFM2 Mobile App are listed below.

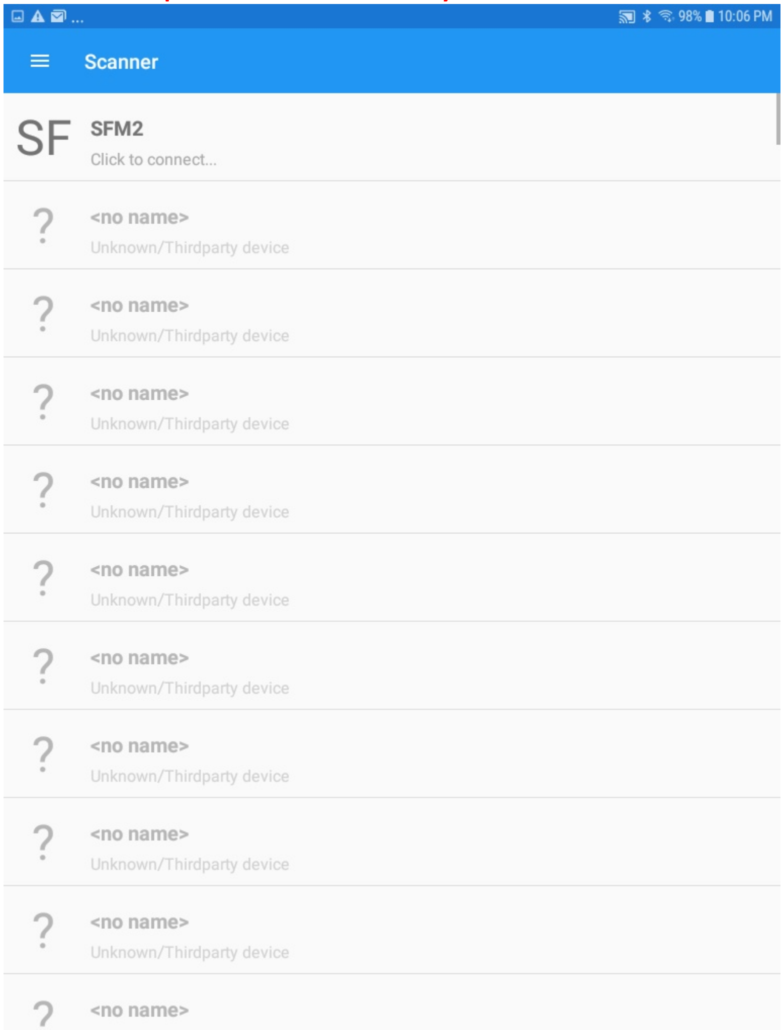
- Allow multiple SFM2's to be connected simultaneously
- Provides 4 Standard SFM2 configurations:
  - OFF
  - Low Power (ACC=26Hz, GYRO=26Hz, MAG=26Hz, SFOR=26Hz)
  - Balanced (ACC=104Hz, GYRO=104Hz, MAG=104Hz, SFOR=104Hz)
  - Performance (ACC=833Hz, GYRO=833Hz, MAG=104Hz, SFOR=417Hz)
- 3D Cube Representation: Sensor Fusion Output for SFQTE and SFLA
  - Tare SFM2 from 3D View and Manual rotate image of SFM2 to match real-world positioning of SFM2
- BLE & COM Port Throughput Testing
- SFM2 Self-Test
- Individual Sensor Configuration
  - Including Filter Settings for Accel, Gyro. Mag
- Individual Data Enable for each Sensor Stream
  - Both Raw Sensor Output and Sensor Fusion Outputs
- Global Reference Enable/Disable
- Headtare Adjustment: Allows for Tare Offset
- Tare SFM2
- Manual Calibration of SFM2
- Manual Kalman Filter Tuning
- Renaming of individual SFM2
- Store Current Configuration
- Restore Factory Defaults
- Real-Time Graphing for Pressure, Temperature, Humidity, and Altitude
- Display of real-time stream data
- BLE and COM Port Stats
- TIME and TOFFSET Settings
- Data Log to csv file

18.2.1 Scanner

SFM2 App will show all SFM2's that are active(ON) and in the range of the mobile device.

To connect click on SFM2 shown in list that is desired to be connected to.

NOTE: If multiple SFM2's are in use they should be renamed so that they can be distinguished between each other.

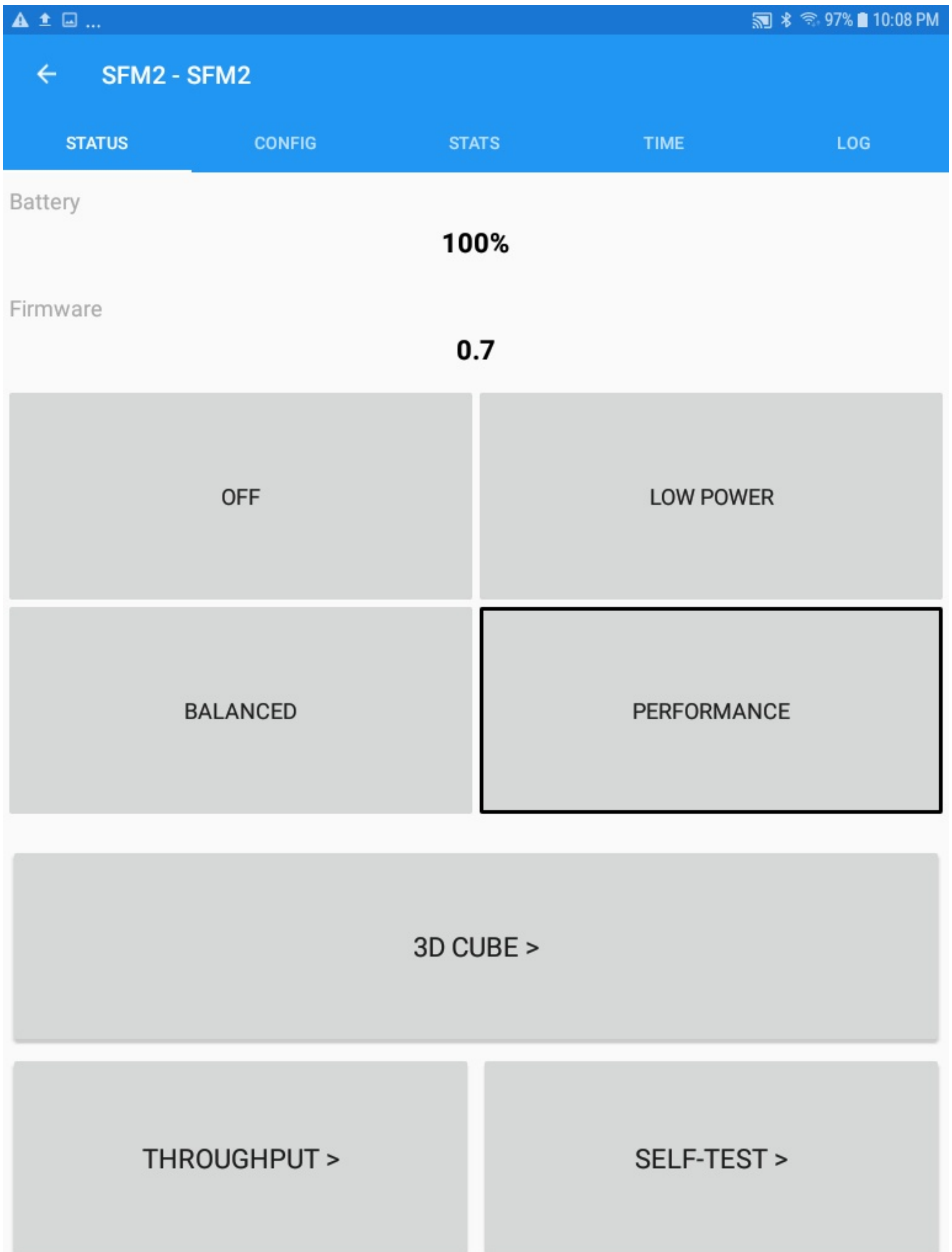


18.2.2 Home Tab

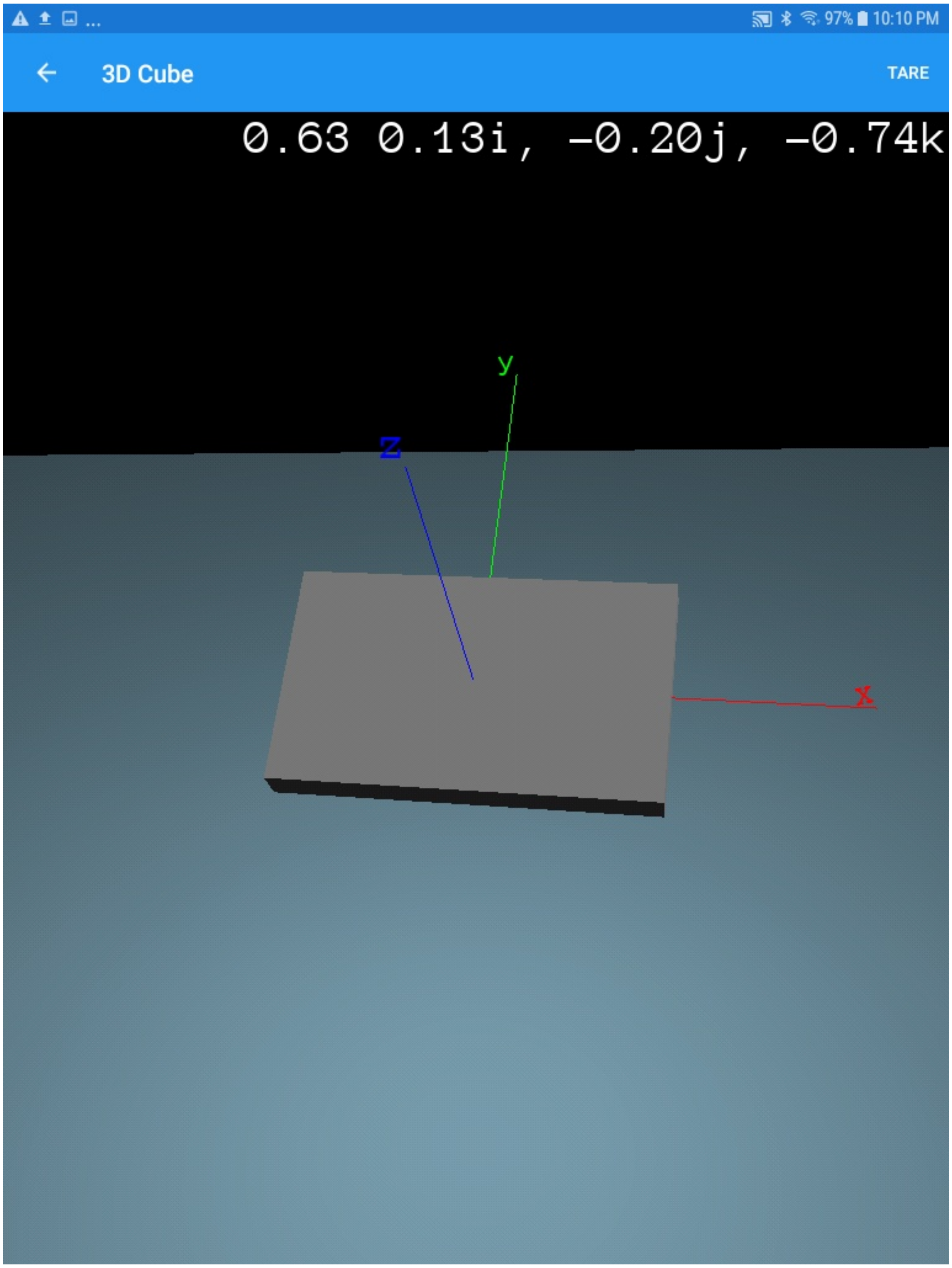
Status tab allows for quick SFM2 configuration and status

- Battery Percentage
- Firmware Version
- Quick Configuration between 4 Default Modes
- 3D Cube Display with Tare and manual drag for orientation
- Throughput testing(BLE and COM Port)
- Perform SFM2 Self-Test

HOME – ‘Performance’ mode selected SFM2



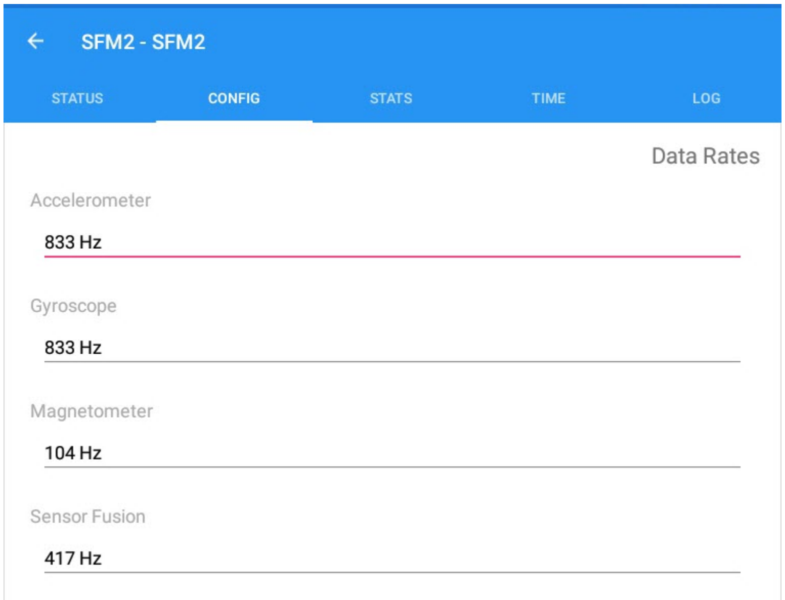
3D SFM2 View



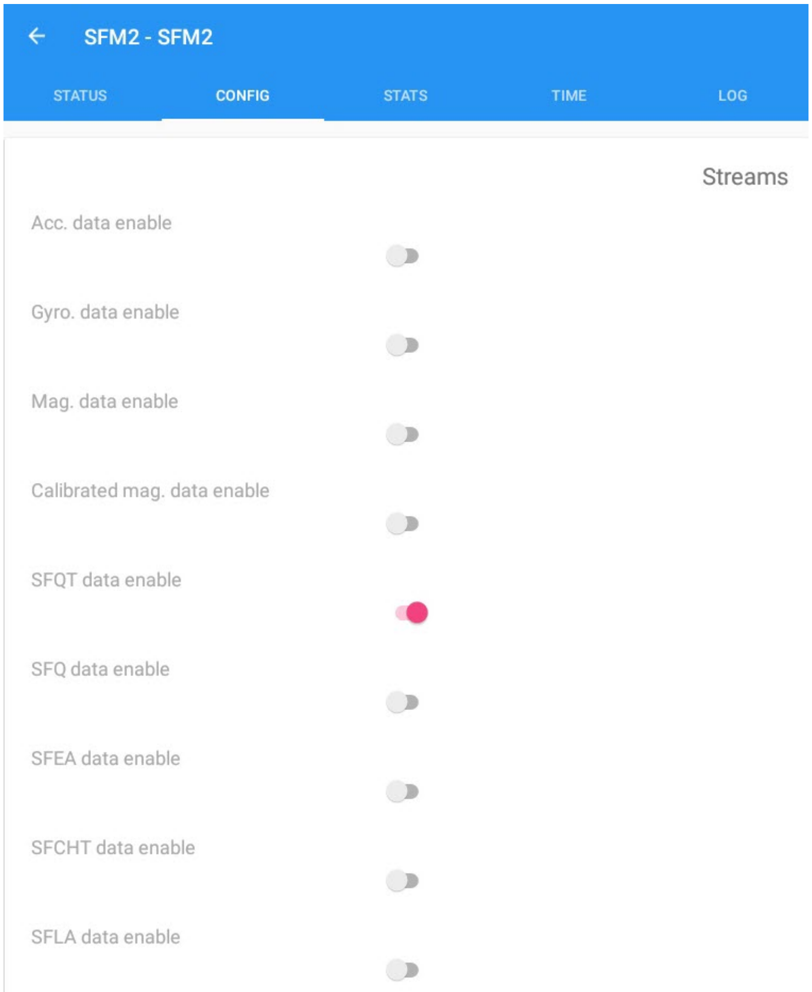
18.2.3 Config Tab

Config tab provides for the following configurations.

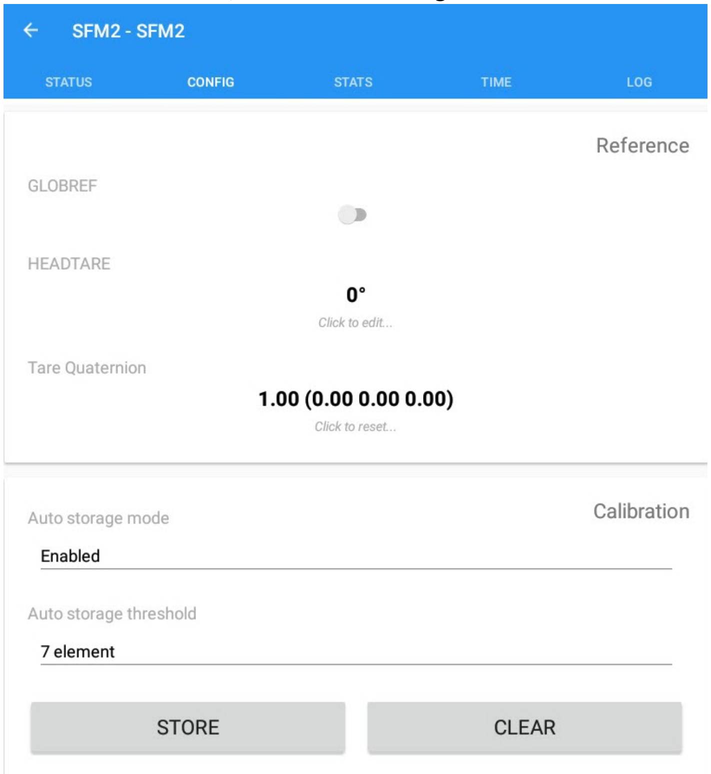
CONFIG – Data Rates for individual Motion Sensors and Sensor Fusion Output



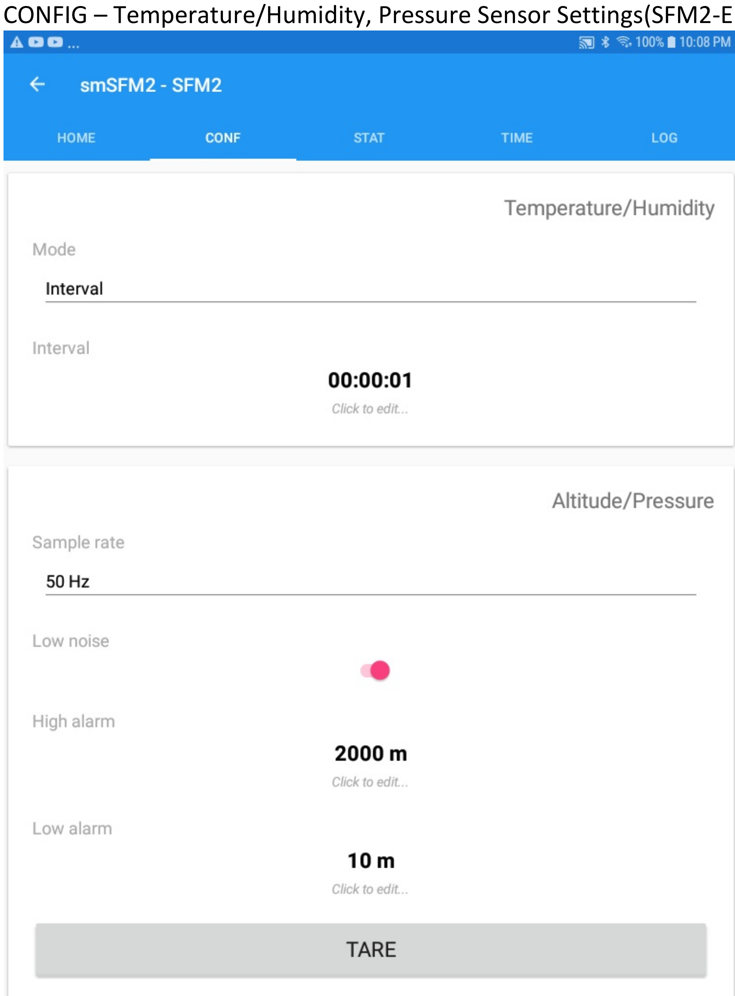
CONFIG – STREAMS



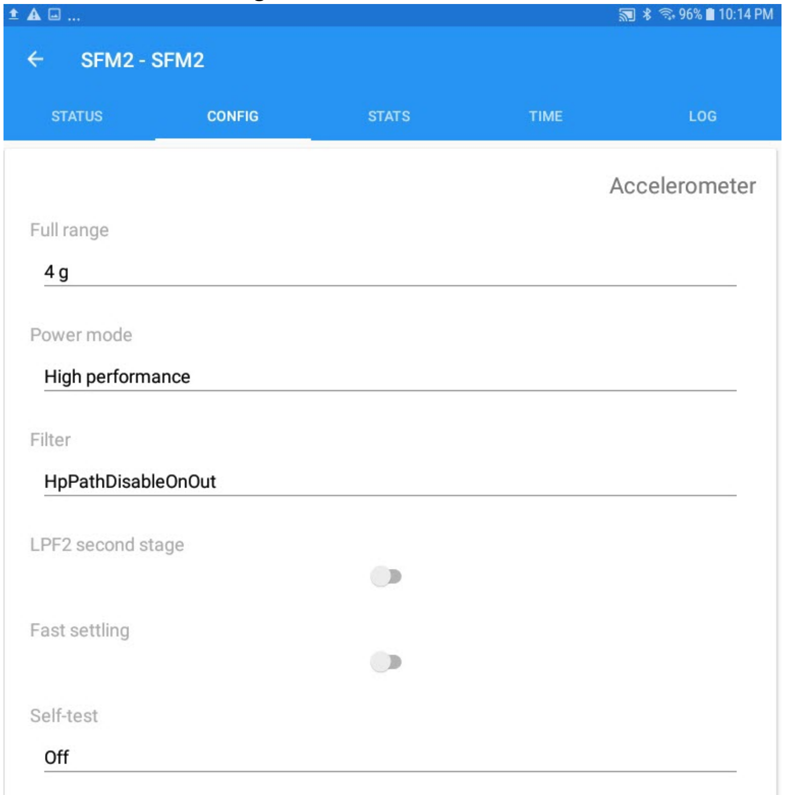
CONFIG – Reference / Calibration Settings



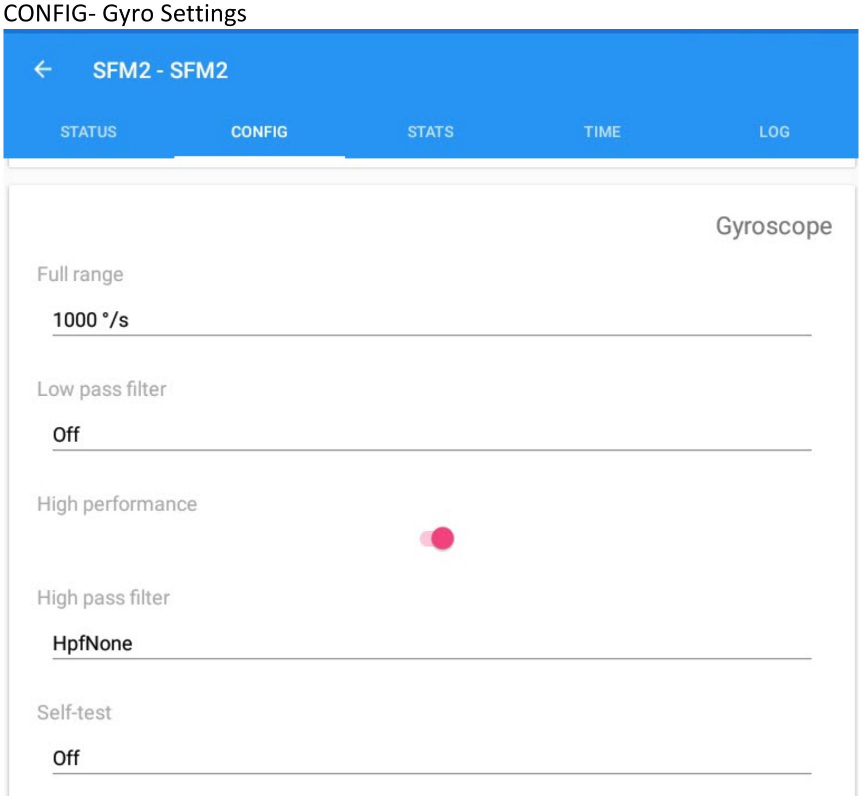
CONFIG – Temperature/Humidity, Pressure Sensor Settings(SFM2-EH Version)



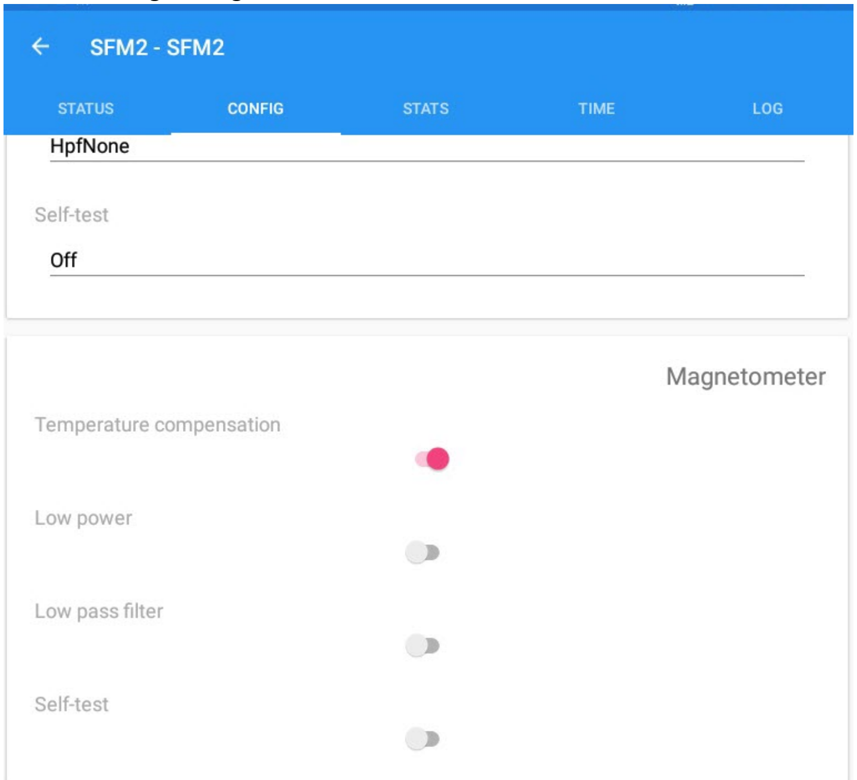
CONFIG – ACCEL Settings



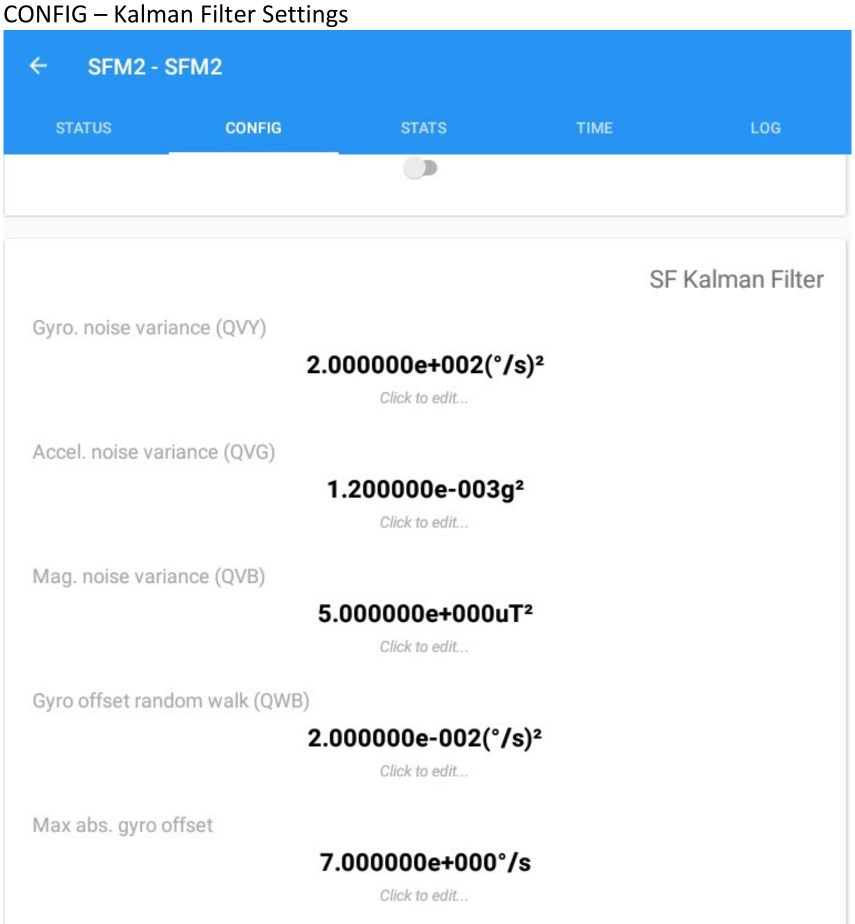
CONFIG- Gyro Settings



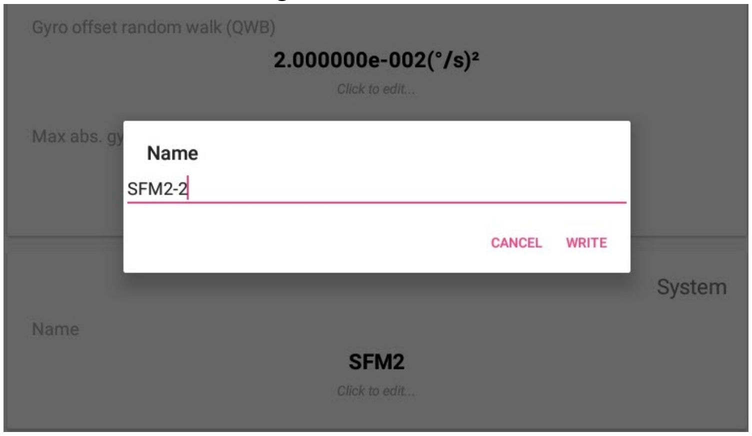
CONFIG – Mag Settings



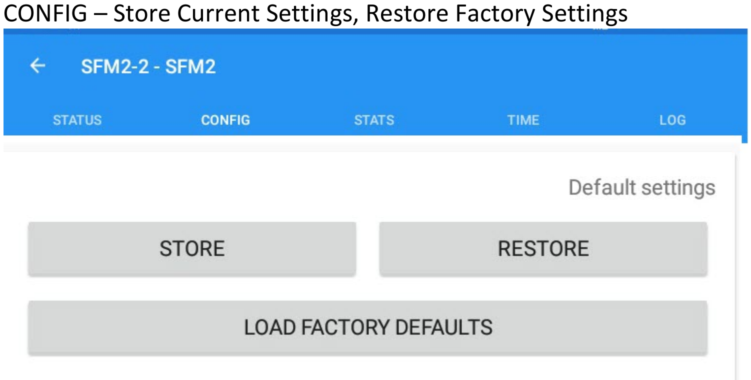
CONFIG – Kalman Filter Settings



CONFIG – SFM2 Renaming



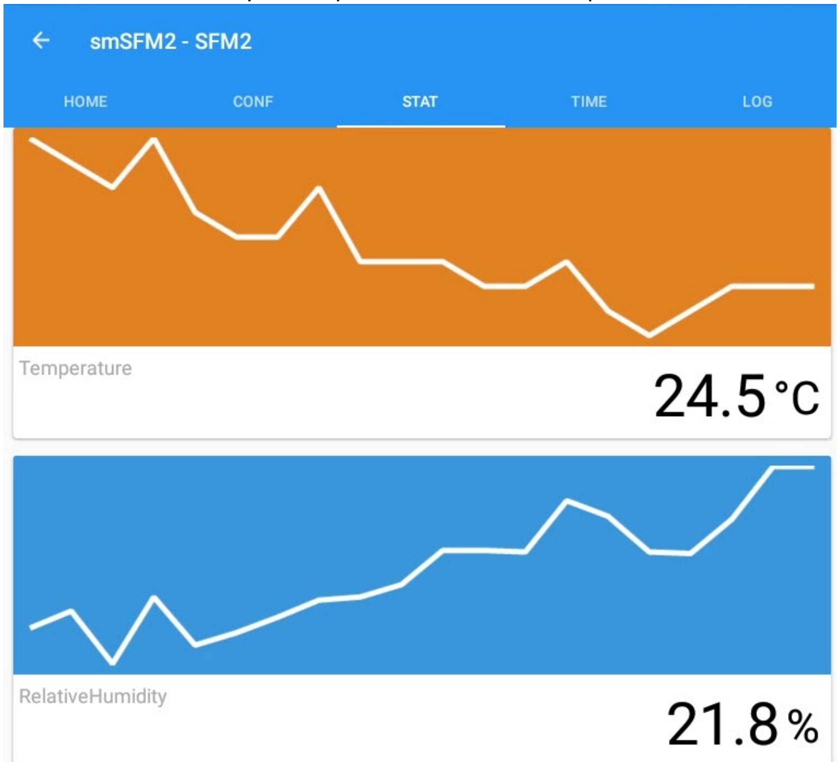
CONFIG – Store Current Settings, Restore Factory Settings



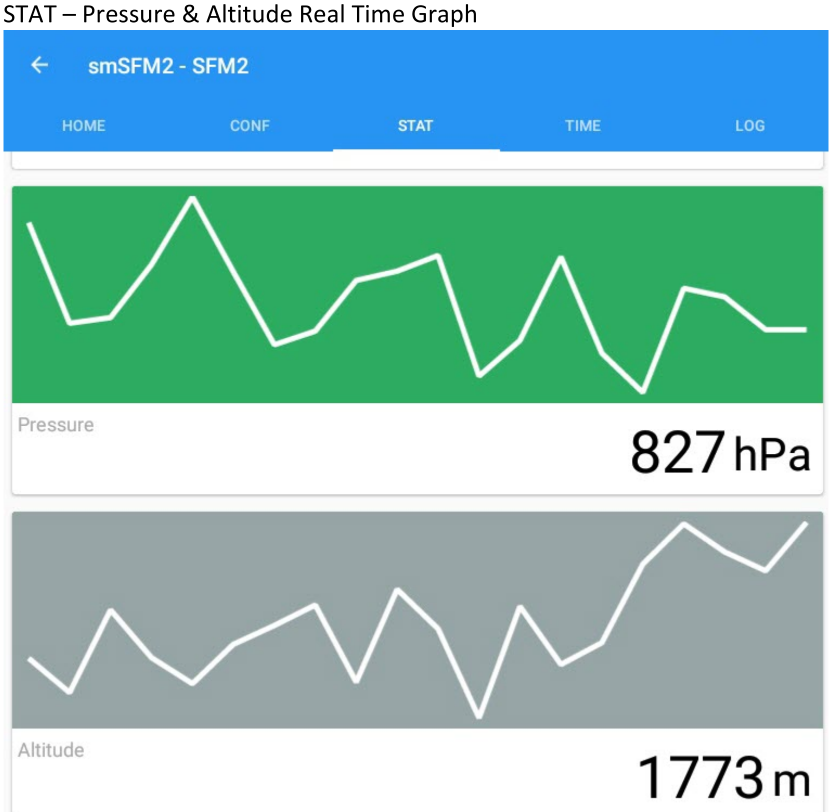
18.2.4 STAT Tab

The STAT tab shows real sensor information including scrolling graphs and also other information such as Calibration and BLE/COM port data statistics.

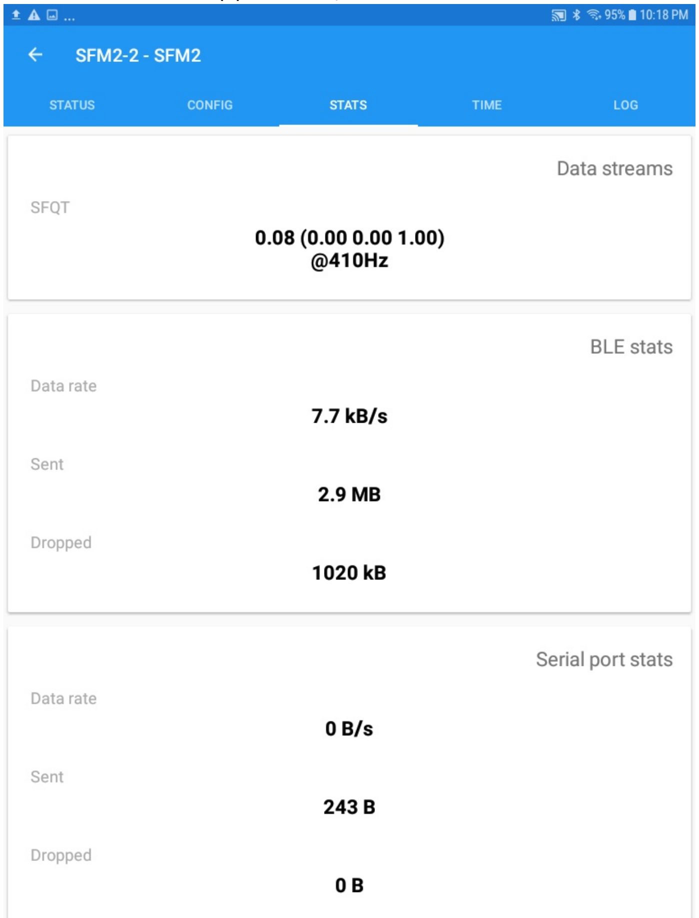
STAT-Relative Humidity & Temperature Real Time Graph



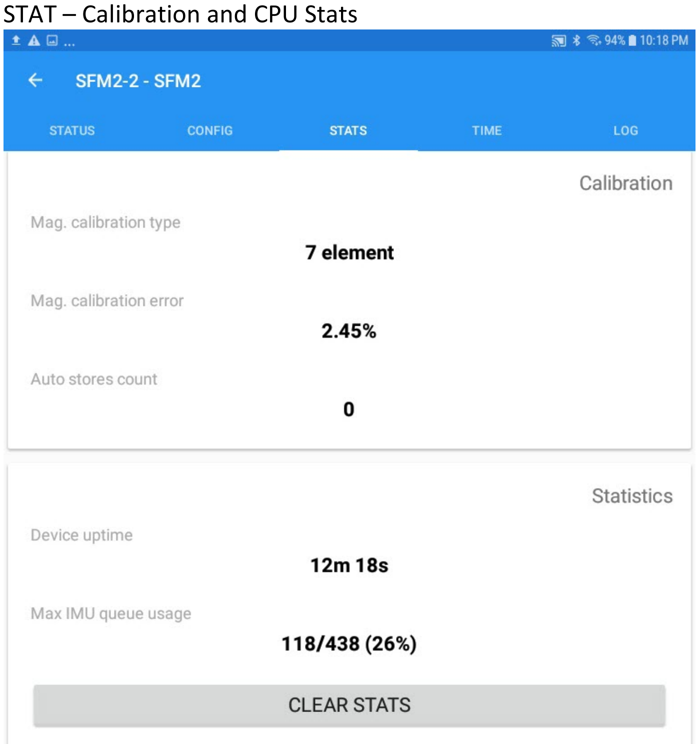
STAT – Pressure & Altitude Real Time Graph



STAT – Data Stream(s) and BLE, COM Port Stats

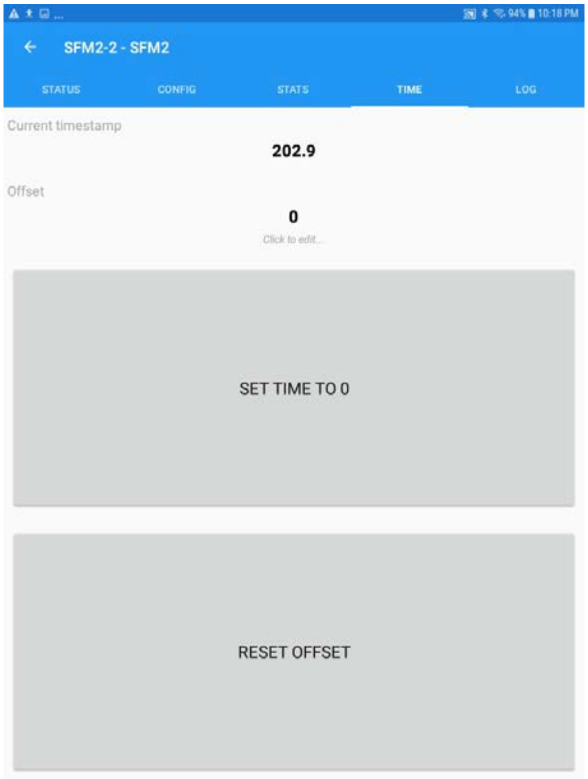


STAT – Calibration and CPU Stats



18.2.5 Time Tab

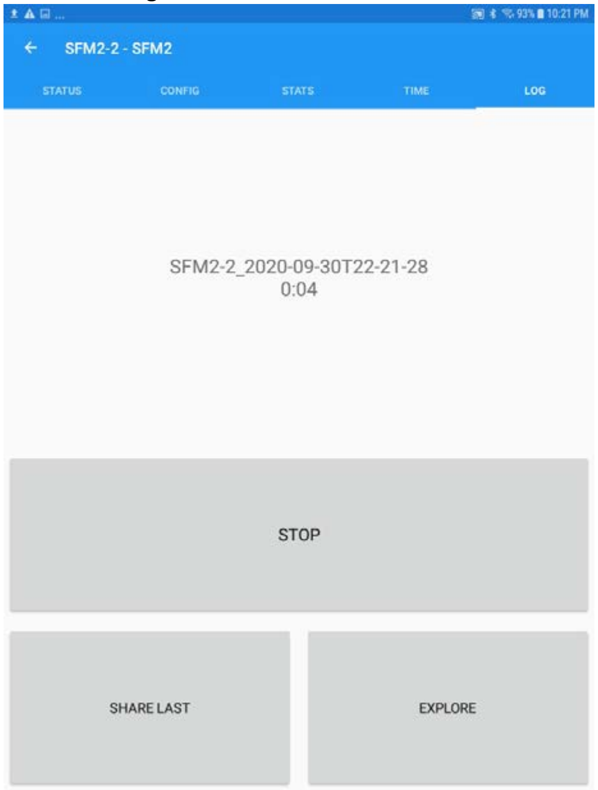
The Time Tab shows the current Timestamp and allows for user to manually set a time offset, TOFFSET, Reset the timestamp to 0 and Reset the TOFFSET value to 0.



18.2.6 Log Tab

The log tab is where the user can start logging any of the sensor data streams to a CSV file and push to a remote drive or store locally on the mobile device. Users can also pull up prior stored data logs that are stored on the mobile device.

LOG - Running



LOG – Stored Log Files