



GNSS WITH DEAD-RECKONING (DR) AND REAL-TIME KINETICS (RTK)

GNSS + DR + RTK USB User Manual



Purpose of the Document

The purpose of this document is to explain the GNSS board with Dead-Reckoning and Real-Time Kinetics feature. This document contains the features of GNSS board and how to use it to use it for high accuracy and high-precision positioning.

Document History

Version	Author	Date	Description
A	5G HUB	06.20.2024	Initial Document

Table of Contents

Purpose of the Document	2
Document History	2
1 Package Contents.....	5
1.1 GNSS Sensor board	5
1.2 Download	5
1.3 Documentation	5
2 Introduction	6
3 Typical Applications	7
4 GNSS Constellations.....	8
4.1 GPS	8
4.2 BeiDou	8
4.3 GLONASS	8
4.4 Galileo	8
4.5 IRNSS	8
4.6 QZSS	8
5 Augmentation System.....	9
5.1 SBAS	9
5.2 AGNSS.....	9
5.3 Real-Time Kinematic (RTK).....	9
5.4 Odometer (ODO).....	9
5.5 Dead Reckoning Function	9
6 GNSS USB Diagram.....	10
7 Connecting using the UART or I2C or SPI	11
8 DR Configuration.....	13
8.1 Setting the Orientation	13
9 Mounting.....	16
10 Dead-Reckoning Calibration.....	17
11 Enable Dead-Reckoning and 6-axis Sensor	17
12 Sensor Messages.....	19
13 Using QGNSS Tool	19
13.1 Dual Band vs. single Band Antenna	21
14 Using Arduino IDE	22
15 Updating Firmware (Optional)	24

15.1 Calibrating the IMU and enable IMU output..... 26

1 Package Contents

1.1 GNSS Sensor board

- GNSS USB

1.2 Download

GNSS Tool is here:

[5G-NB-IoT/Tools at master · 5ghub/5G-NB-IoT \(github.com\)](https://github.com/5ghub/5G-NB-IoT)

Arduino sketches for the GNSS USB can be downloaded from the following website:

<https://github.com/5ghub/5G-NB-IoT/tree/master/KitSketches>

To use the board with Arduino IDE and starts running Arduino projects and sketches, install the following software:

Install Arduino IDE for Windows from the following website:

<https://www.arduino.cc/en/Main/Software>

Download and install Arduino library (**5G-NB-IoT_Arduino.zip**) here:

<https://github.com/5ghub/5G-NB-IoT>

1.3 Documentation

Dead-Reckoning and Real-Time Kinetics Application Note:

[LC29H\(BA\) DR&RTK Application Note](#)

2 Introduction

The GNSS board is a compact board for GNSS applications. It features a concurrent multi-constellation GNSS receiver with dual GNSS bands, an integrated 6-axis sensor, fusion with Real-Time Kinematic (RTK) and Dead-Reckoning (DR). The module can achieve sub-meter-level accuracy in open-sky areas.

The GNSS board can work on L1 and L5 bands for GPS, Galileo and QZSS, L1 band for GLONASS and BeiDou, and L5 band for IRNSS. This greatly increases the number of satellites which can be involved in tracking and positioning, thereby significantly reducing the multipath effect from tall buildings in urban environments, reducing signal acquisition time and improving positioning accuracy. In addition to this the GNSS module's on-board LNAs and SAW filters serve to ensure better positioning in weak signal areas and harsh environments.

The GNSS USB supports two DR modes: ADR (Automotive Dead Reckoning) and UDR (Untethered Dead Reckoning). In ADR mode, the module relies on speed data from the vehicle and the onboard 6-axis sensor for enhanced accuracy in environments with nonexistent GNSS coverage. The UDR mode does not require speed data. The firmware automatically switches to UDR mode if no speed data is injected upon module power-up. The module obtains vehicle speed data through wheel-ticks or direct vehicle speed data output (m/s). There are two wheel-tick injection methods: 1) injection through the WHEELTICK pin, with a maximum distance increment of 0.05 m per pulse; 2) cumulative wheel-tick injection through the UART interface, with a minimum injection frequency of 10 Hz, and a maximum distance increment of 0.05 m per pulse. The direct vehicle speed output can only be injected through the UART interface, with the minimum injection frequency of 20 Hz, and the maximum error of 0.1 m/s between the injected speed and actual speed.

The GNSS USB supports the Dead Reckoning technology. By combining satellite navigation data with wheel speed, gyroscope and accelerometer data, the module obtains continuous and high accuracy positioning in weak signal environments such as tunnels and urban canyons when the vehicle state (e.g., speed, forward direction or vertical displacement) changes, or even when the satellite signal is partially or completely blocked.

The GNSS module's combination of compact design, low power consumption and high performance makes it a popular choice for real-time tracking systems for vehicles, people and assets, as well as for sharing economy applications

Feature Highlights

- Support dual-band and multi-constellation
- Supports GPS/BeiDou/GLONASS/Galileo/IRNSS/QZSS
- High-performance, high reliability positioning engine. It facilitates a fast and precise GNSS positioning capability.
- Supports serial communication interfaces UART and SPI.
- Integrates a 6-axis IMU sensor and supports sophisticated Real-Time Kinematic (RTK) and Dead-Reckoning (DR) algorithms to fuse the sensor data, GNSS raw data and speed data, etc. to provide sub-meter level positioning accuracy in an open-sky environment
- Includes an RTK position engine in order to provide cm-level positioning accuracy.
- Can be used as a base station to generate RTK differential correction data that can be transmitted over radio or over cellular connectivity to become a part of a NTRIP network.

- Support AGNSS feature that significantly reduces the modules' TTFF, especially under lower signal conditions.
- Embedded flash memory provides the capacity for storing not only user-specific configurations, but also future firmware updates.
- Can be embedded in your application

3 Typical Applications

- High-precision GNSS for Tracking and Positioning
- Navigation where GNSS signal is not available
- 6-axis IMU sensor (3-axis accelerator and 3-axis gyroscope)
- Internet of things

4 GNSS Constellations

The GNSS module is a dual-band GNSS receiver that can receive and track GPS, BeiDou, GLONASS, Galileo, QZSS, IRNSS signals.

4.1 GPS

The module is designed to receive and track GPS L1 C/A signals (1574.397–1576.443 MHz) and L5 signals (1166.22–1186.68 MHz) provided by GPS.

4.2 BeiDou

The module is designed to receive and track BeiDou B1I (1559.052–1563.144 MHz) and B2a (1155.99–1196.91 MHz). The ability to receive and track BeiDou signals in conjunction with GPS results in higher coverage, improved reliability, and better accuracy.

4.3 GLONASS

The module is designed to receive and track GLONASS L1 signals (1597.781–1605.656 MHz) provided by GLONASS.

4.4 Galileo

The module is designed to receive and track Galileo E1 (1573.374–1577.466 MHz) and E5a (1166.22–1186.68 MHz) signals provided by Galileo.

4.5 IRNSS

The Indian Regional Navigation Satellite System (IRNSS) or NavIC is a regional navigation satellite system that transmits additional L5 signals for complying with the requirements of an independent accurate positioning system for users in India. The GNSS module is designed to receive and track IRNSS L5 signals (1175.427–1177.473 MHz) from IRNSS satellites.

4.6 QZSS

The Quasi-Zenith Satellite System (QZSS) is a regional navigation satellite system that transmits additional GPS L1 C/A signals for the Pacific region covering Japan and Australia. The GNSS module can detect and track these signals concurrently with GPS signals, resulting in better availability especially under challenging conditions, e.g., in urban canyons.

5 Augmentation System

5.1 SBAS

The GNSS module supports SBAS (Satellite-Based Augmentation System) broadcast signal reception, and GPS data are complemented by additional regional or wide area GPS enhancement data. The system enhances the data through satellite broadcasting, and this information can be used in GNSS receivers to improve the accuracy of the results. SBAS satellites can also be used as additional signals for ranging or distance measurement, further improving availability. Supported SBAS systems: WAAS, EGNOS, MSAS and GAGAN.

5.2 AGNSS

The module supports AGNSS feature that significantly reduces the module's TTFF, especially under lower signal conditions. To implement AGNSS feature, the module should get the assistance data including the current time, rough position, and LTO data.

5.3 Real-Time Kinematic (RTK)

The GNSS module modules support RTK functionality as rovers. Before supporting the RTK navigation technique, the module needs to receive the RTK correction messages via its UART port. RTK correction messages can be delivered either using a cellular module or other terrestrial network technology. In default configuration, the module will attempt to achieve the best positioning accuracy based on the correction data that it receives. When the module receives an input stream of RTCM messages, it enters RTK float mode. Once it fixes carrier phase ambiguities, the module enters the RTK fixed mode.

The module may be expected to achieve sub-meter level accuracy only when it is in RTK fixed mode. If the module loses the carrier phase lock, at the minimum semaphore required to maintain the RTK fixed mode, it returns to the RTK float mode. The module will also continue to try to resolve carrier phase ambiguity and return to RTK fixed mode after restoring the minimum semaphore.

The current mode of operation is set by the associated NMEA messages.

5.4 Odometer (ODO)

The FWD hardware input is used to input vehicle forward/backward status signals. When it is at low voltage level, the vehicle is moving forward, and when it is at high level, it is moving backward.

The WHEELTICK hardware input is used to input odometer signals from a vehicle. It can be sampled from the wheel revolution sensors or the transmission of the vehicle.

Only cars need to be connected to the FWD signal, electric bicycles do not.

5.5 Dead Reckoning Function

Dead Reckoning is the process of estimating the module's current position based on the last position obtained from GNSS, speed, heading sensor data, etc. With this combined sensor inputs, the system plots the navigation trace when the satellite signals are partially or completely blocked while satellite signals provide updates and correction for sensor drift. With this technology, the system obtains continuous and high-accuracy positioning in environments such as tunnels and urban canyons.

In addition, the module supports output of sensor raw data through UART to support your applications such as driving behavior analysis.

6 GNSS USB Diagram



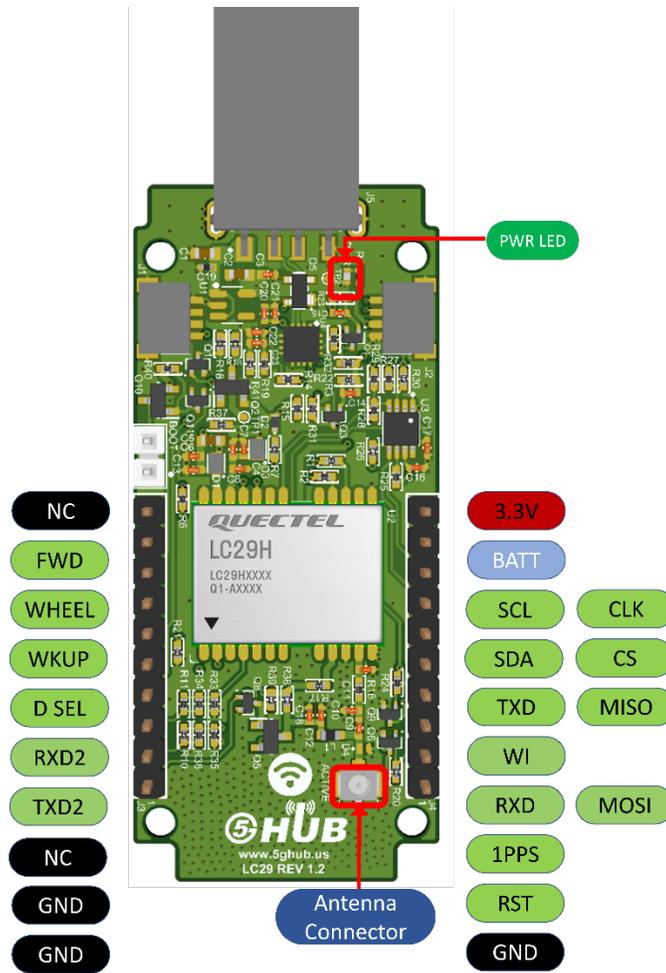
Pin #	Feature	Direction	Description
J3-1	NC		
J3-2	FWD	Input	Forward/Backward direction
J3-3	WHEELTICK	Input	Odometer/wheel-tick input
J3-4	WAKEUP	Input	wake up the module from the Backup mode
J3-5	D_SEL	Input	Selects I2C, or SPI, or UART interface
J3-6	RXD2	Input	UART2 Receives data. Use with voltage domain 1.8V
J3-7	TXD2	Output	UART2 Transmits data. Use with voltage domain 1.8V
J3-8	NC		
J3-9	GND		
J3-10	GND		
J4-1	GND		
J4-2	RESET_N	Input	Resets GNSS USB
J4-3	1PPS	Output	One pulse per second
J4-4	MOSI (RXD)	Input	SPI slave output master out or UART1 Receive data
J4-5	WI (NC)	Output	Warning Indicator. Currently NC
J4-6	MISO (TXD)	Output	SPI master out or UART Transmit data
J4-7	CS (SDA)	Input/Output	SPI Chip-Select or I2C data
J4-8	CLK (SCL)	Input	SPI clock or I2C clock
J4-9	BATT	Input	Back-up battery
J4-10	3.3V	Output	Output 3.3V

7 Connecting using the UART or I2C or SPI

The GNSS USB can be interfaced with either UART or SPI. Plug the GNSS USB into a USB port to power it ON. GNSS data comes automatically on USB interface. The assertion of D_SEL (either 0V or 3.3V) will select either USB (UART1) or SPI or I2C.

D_SEL	USB	SPI	I2C
0 (Default)	Can be used for communication and downloading		Can only be used for communication
1		Can be used for communication	

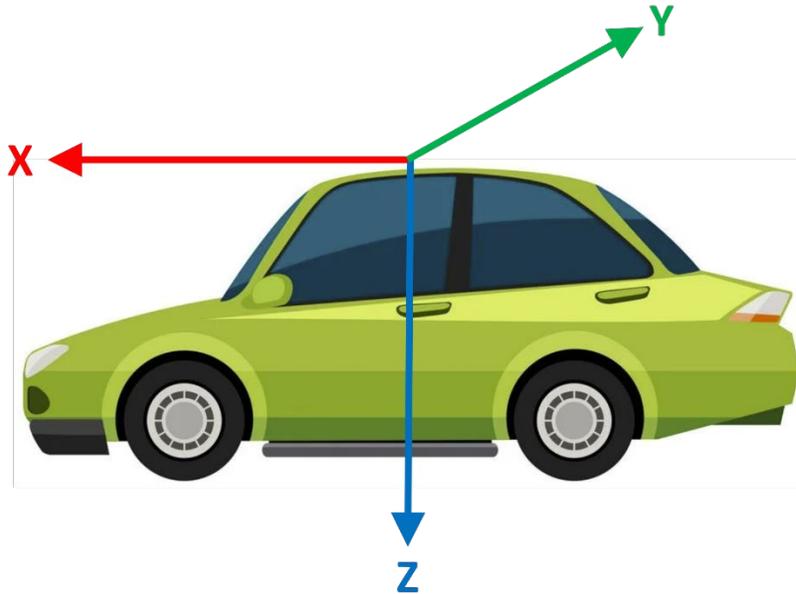
Connect a GNSS antenna to the Antenna U.FL connector.



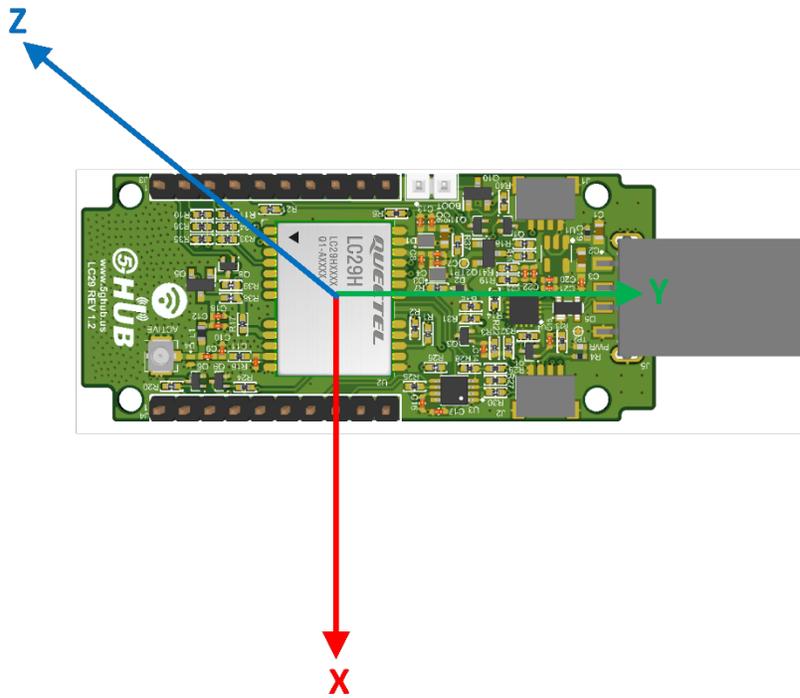
8 DR Configuration

8.1 Setting the Orientation

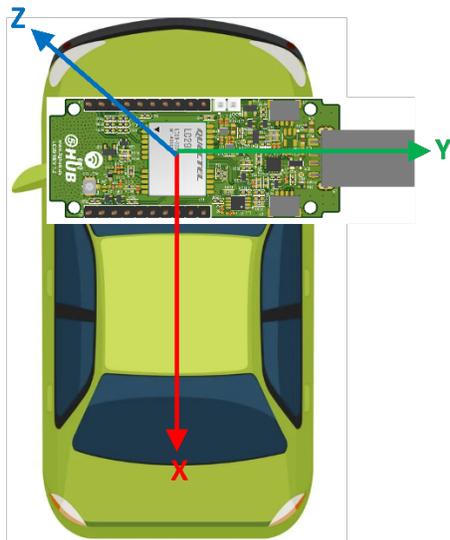
The reference frame axes definitions are shown below. The X axis is the vehicle forward direction, the Y axis is the right side of the vehicle, and the Z axis is the downward direction.



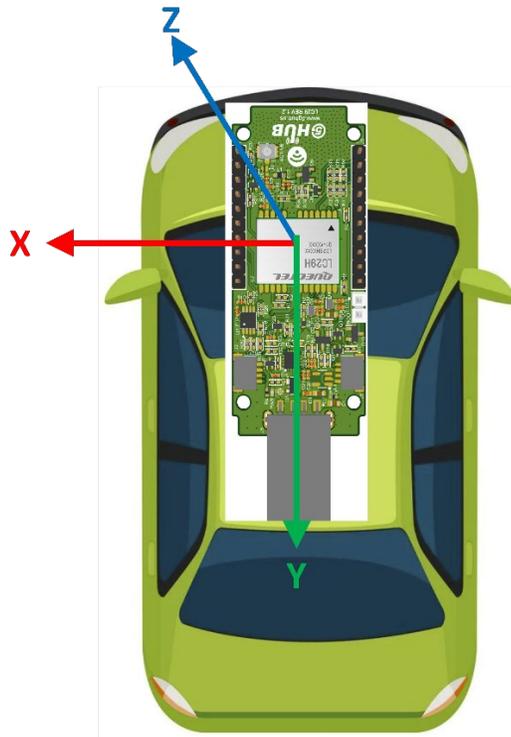
The orientation of the module is shown below:



Below show some examples for board installation:



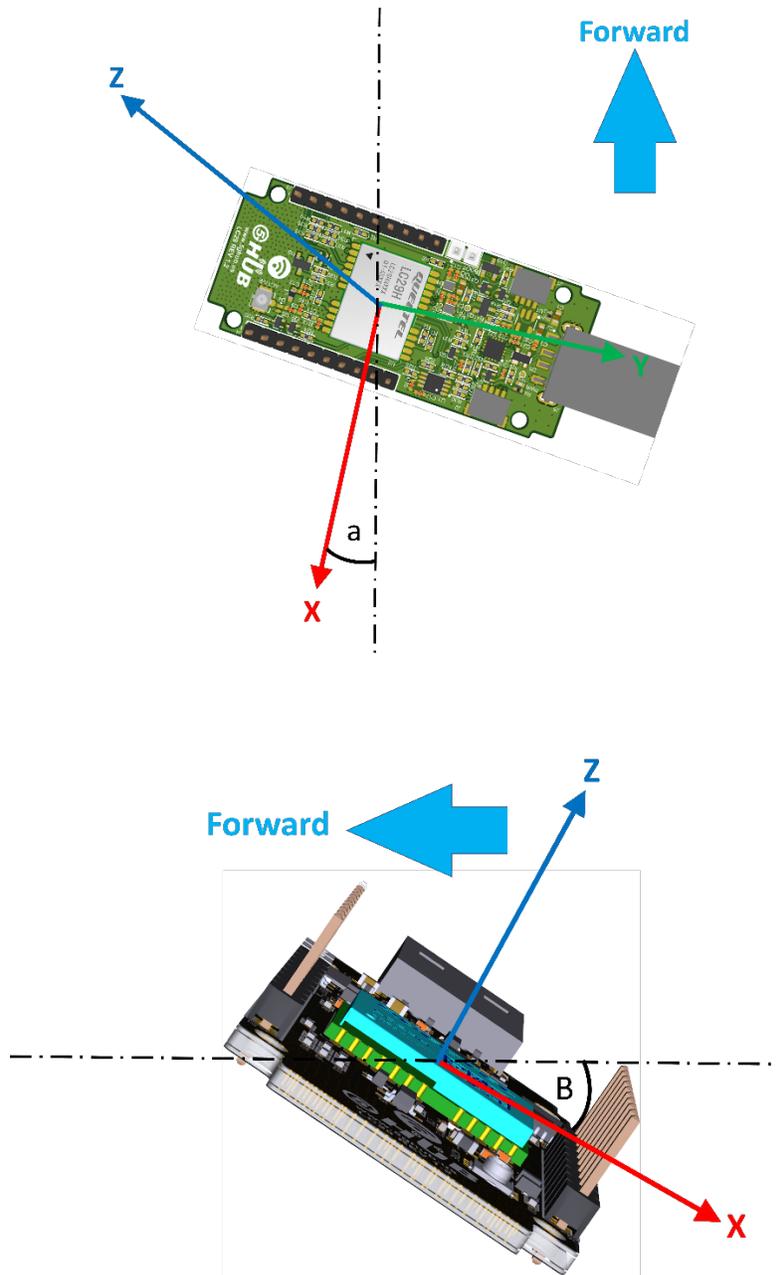
if the board installation direction like the above picture, the X axis is inverted with reference, the Y axis is same with the reference frame, and the Z axis is also inverted with the reference frame, so the configuration should be **-X Y -Z**, the command is **\$PQTMCFGORIENTATION,1,-X Y -Z*66**.



If board installation direction like the above picture, the X axis is inverted with Y axis of reference frame, the Y axis is inverted with the X axis of reference frame, and the Z axis is also inverted with the Z axis of reference frame, so the configuration should be **-Y -X -Z**, the command is **\$PQ TMCFGORIENTATION,1,-Y -X -Z*4B**.

9 Mounting

When mounting the LC29H on the carrier, need to keep the yaw, pitch and roll angle within 5 degrees to the reference frame.



In the real mounting, need to make sure that $-5^\circ \leq a \leq 5^\circ$, $5^\circ \leq B \leq 5^\circ$.

10 Dead-Reckoning Calibration

- 1) Fix the module on the vehicle frame. Any displacement, turn or tilt of the device, although small, will cause performance issues and/or void the calibration process making it fail.
- 2) The calibration process should be performed on good GNSS signal conditions and clear sky view.
- 3) Power up the module then start the vehicle on a plain surface and keep it still for at least 30 seconds.
- 4) Start driving the vehicle in good GNSS signal conditions. The module will start the self-calibration process which would be completed in a few minutes.
- 5) The calibration process ends when the <SolType> of \$PQTMINS message indicates combined solution(GNSS + DR).

11 Enable Dead-Reckoning and 6-axis Sensor

You can enable or disable the GNSS and the raw data from the internal 6-axis sensor and for dead-reckoning. Use the following command to enable or disable GNSS and/or sensor.

```
$PQTMCFGEINSMMSG,<RW>,<INS>,<IMU>,<GPS>,<Rate>*<Checksum><CR><LF>
```

If it succeeds, it will output:

```
$PQTMCFGEINSMMSGOK*16
```

After issuing the above command, it has to be saved using the following command

```
$PQTMSAVEPAR*5A
```

Table 1: Config GNSS and sensor parameters.

Field	Unit	Meaning
<RW>		0 = Read current configuration. 1 = Config.
<INS>		Enable/disable PQTMINS message. 0 = Disable. 1 = Enable.
<IMU>		Enable/disable PQTMIMU message. 0 = Disable. 1 = Enable.
<GPS>		Enable/disable PQTMGPS message. 0 = Disable. 1 = Enable.
<Rate>		set the output rate of PQTMINS and PQTMIMU messages. It can be 10, 20, 50, 100. When set to 100, the maximum output rate is dependent on baud rate.

Issue the following two commands and then restart the board (power it OFF and then ON)

Command: **\$PQTMCFGEINSMMSG,1,1,1,1,10*3F**
response: **\$PQTMCFGEINSMMSGOK*16**

Command: **\$PQTMSAVEPAR*5A**
Response: **\$PQTMSAVEPAROK*5E**

The following read current configuration:

Command: **\$PQTMCFGEINSMMSG,0*0E**
Response: **\$PQTMEINSMMSG,0,1,1,1,10*7C**

The following can be done through QCOM tool or any serial terminal connected to the board. Enter each command above and click **Send Command**.

The screenshot shows the QCOM_V1.6 software interface. On the left, the 'COM Port Setting' panel is visible, with 'COM Port' set to 3, 'Baudrate' at 115200, 'StopBits' at 1, 'Parity' at None, 'ByteSize' at 8, and 'Flow Control' at No Ctrl Flow. The 'Open Port' button is highlighted. Below this, a log window displays a series of AT command responses, including '\$PQTMINS,51290,0,10.143889,3.315430,0.058844*49' and '\$PQTMINMU,51403,0.147376,-0.041480,0.946964,-1.067500,-2.852500,-0.875000,0.0*58'. A message at the bottom of the log window reads '[2021-07-06_18:32:12:324] Open COM Port Success'. Below the log is an 'Operation' section with checkboxes for 'DTR', 'RTS', 'View File', 'Show Time', 'HEX String', 'Show In HEX', and 'Send With Enter'. The 'Input String' field contains '\$PQTMSAVEPAR*5A' and is highlighted with a red box. The 'Send Command' button is also visible. On the right, the 'Command List' panel shows a table of 29 AT commands with checkboxes for 'Choose All Commands', 'HEX', 'Enter', and 'Delay(mS)'. The 'Run Times' is set to 10 and 'Delay Time(mS)' is set to 1000. At the bottom right, there are buttons for 'Load Test Script', 'Clear All Commands', 'Save As Script', 'Run', and 'Stop'.

12 Sensor Messages

The GNSS receiver emits this message. The message contains the IMU Raw Data: Acceleration, Angular Rate and Hardware Wheel Ticks.

```
$PQTMIMU,<Timestamp>,<ACC_X>,<ACC_Y>,<ACC_Z>,<AngRate_X>,<AngRate_Y>,<AngRate_Z>,<TickCount>,<LastTick_Timestamp>*<Checksum><CR><LF>
```

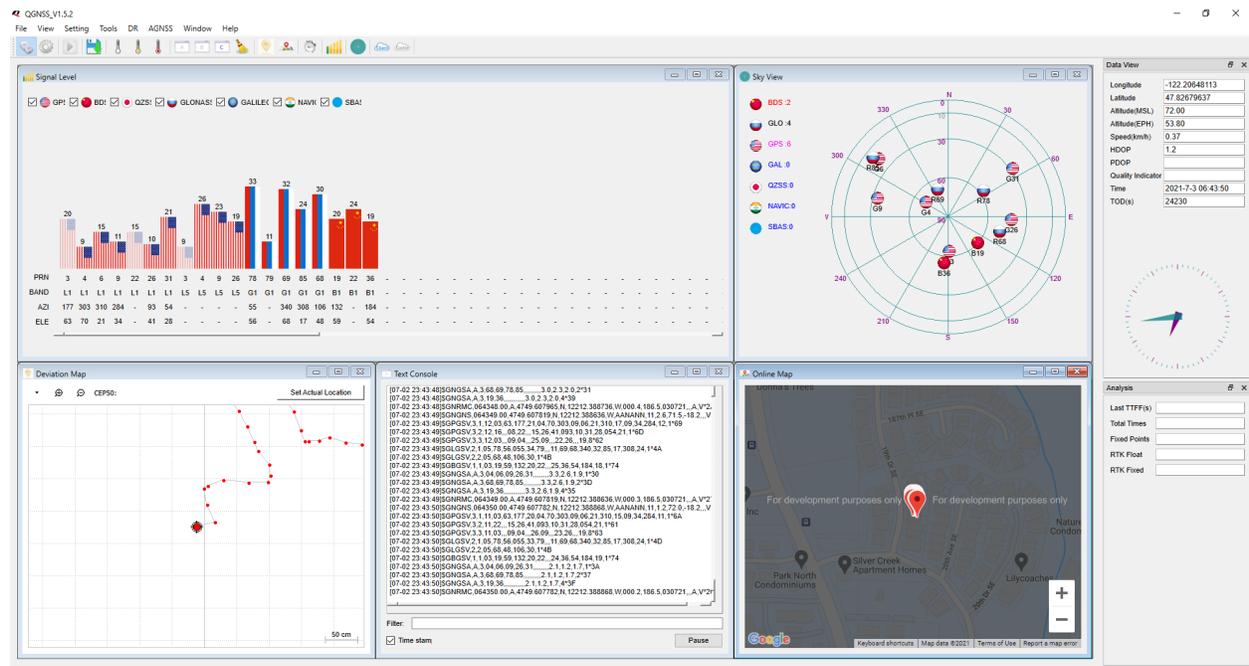
Table 2: PQTMIMU message parameters.

Field	Format	Unit	Meaning
<Timestamp>	Numeric	ms	Milliseconds since turn on. 32-bit unsigned integer
<ACC_X>	Numeric	G	X-Acceleration
<ACC_Y>	Numeric	G	Y-Acceleration
<ACC_Z>	Numeric	G	Z-Acceleration
<AngRate_X>	Numeric	deg/s	X-Angular Rate
<AngRate_Y>	Numeric	deg/s	Y-Angular Rate
<AngRate_Z>	Numeric	deg/s	Z-Angular Rate

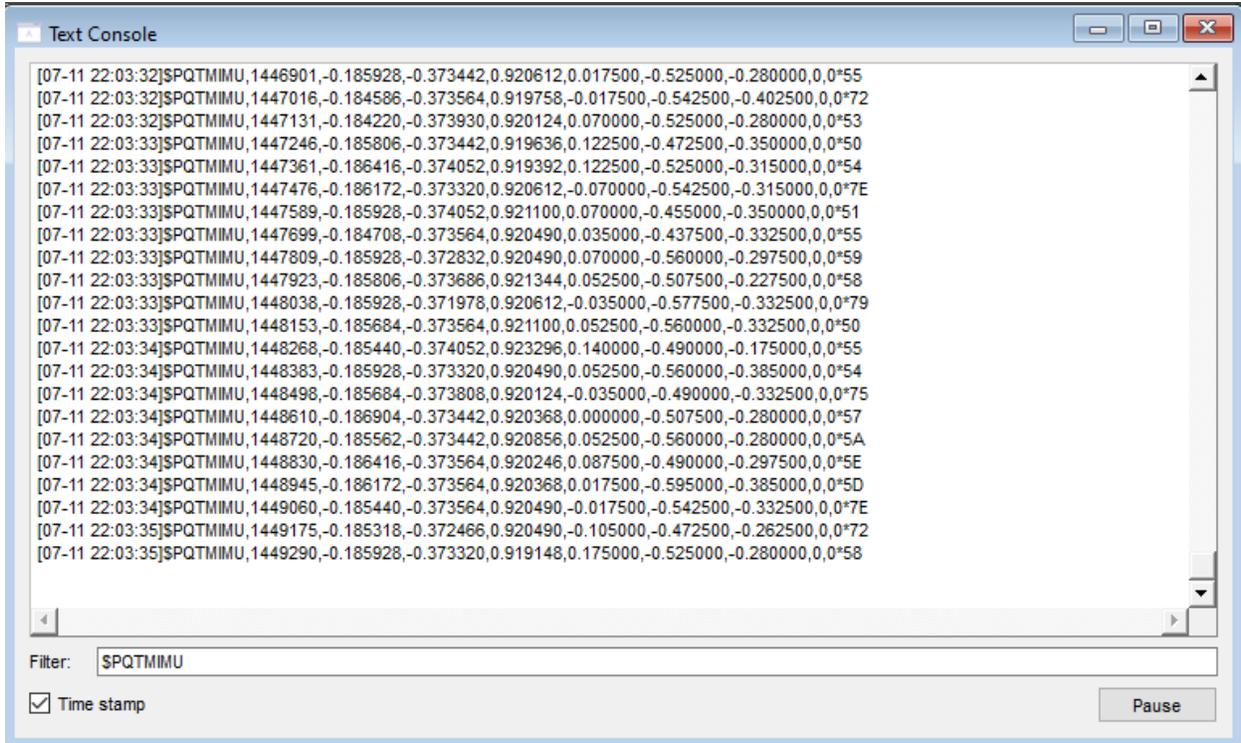
13 Using QGNSS Tool

QGNSS tool is a visual tool where you can see different information about GNSS position including satellites acquired, GNSS NMEA sentences, online Google map, Sky View, Deviation maps and Data View.

Connect the QGNSS to the COM port and click **File->Connect** and you will be able to see a lot of visual data.



You can view the sensor messages only in the GNSS tool by filtering those messages only as in this screen-shoot:



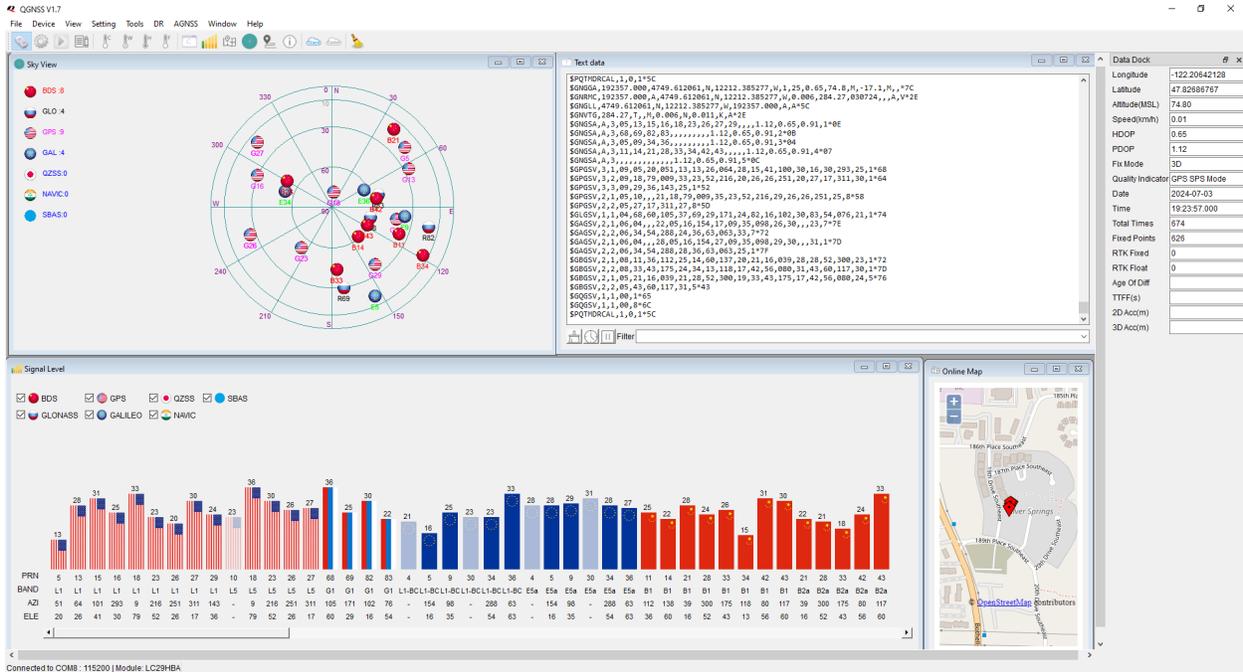


Figure 1: Satellite constellation detected using dual-band antenna

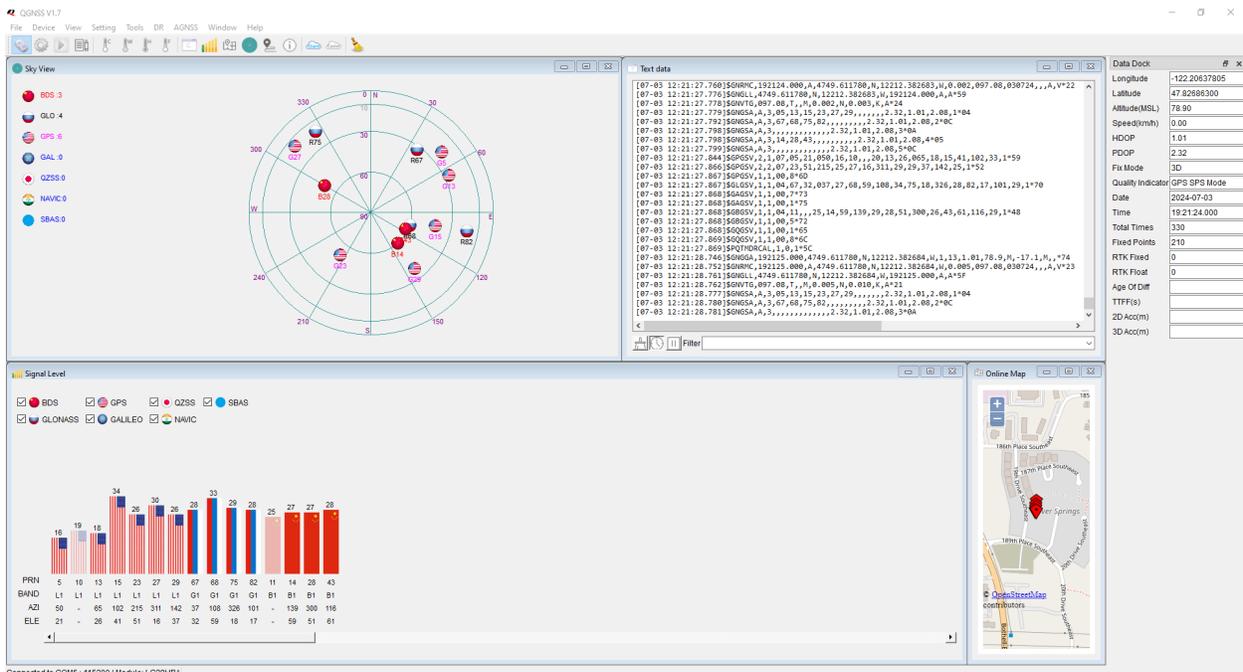


Figure 2: Satellite constellation detected using single-band antenna

14 Using Arduino IDE

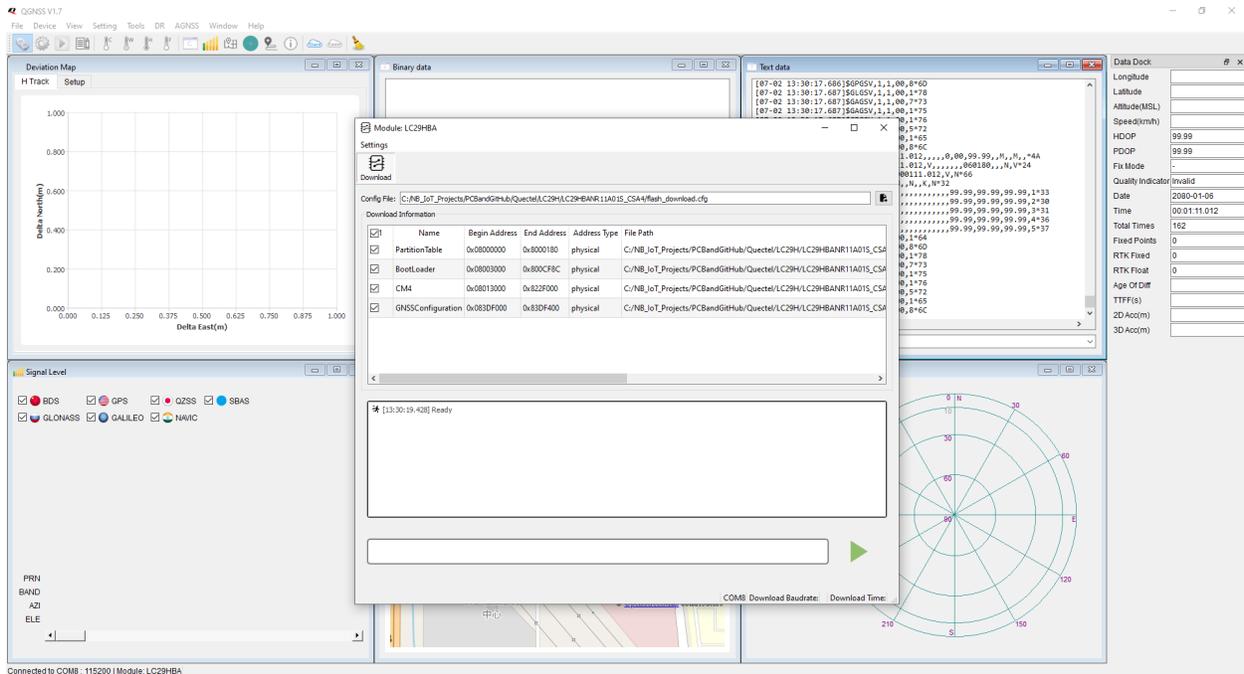
Download the Arduino sketch for the LC29H module from the GitHub and upload it to the NB-IoT board. Connect the UART of the GNSS board to the UART of the NB-IoT board. You shall see the GNSS output messages on Arduino serial monitor.

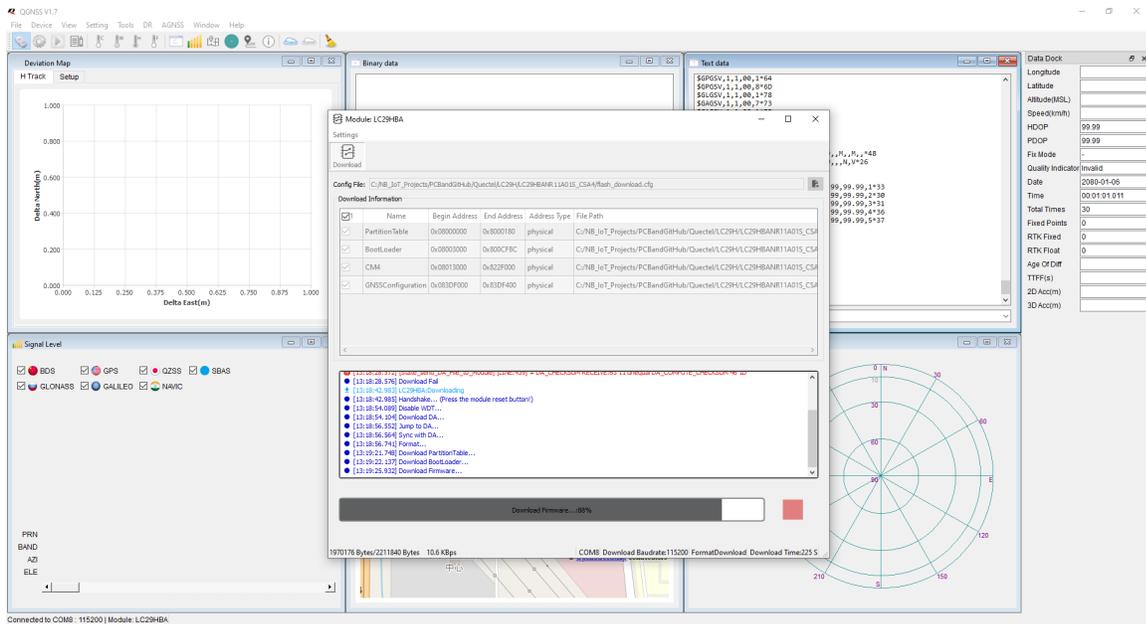
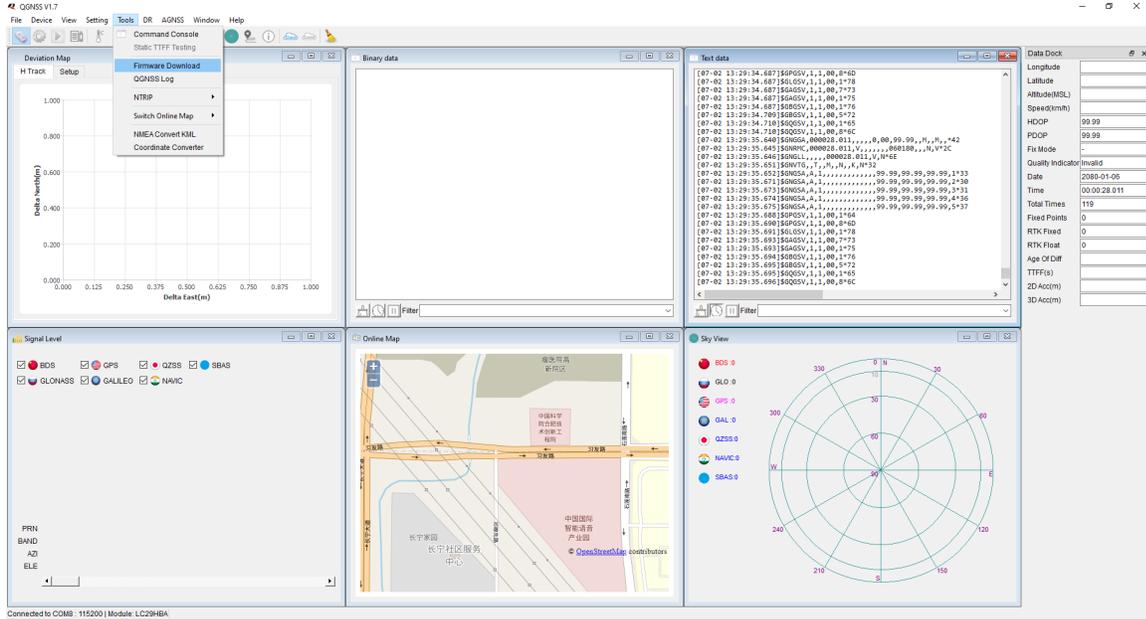
15 Updating Firmware (Optional)

The GNSS USB is shipped with the latest firmware . If you need to perform any other firmware update for this USB, follow the following steps.

Launch QGNSS and choose **Tools->Firmware Download**. Click on the button to start the firmware download. The QNSS prompts to reset the LC29HBA. Reset the LC29H module by connecting RESET_N pin to GND for more than 100ms.

After reset, the firmware download continues until it finishes successfully.





15.1 Calibrating the IMU and enable IMU output

The IMU is enabled by default in the GNSS USB. If you want to enable the IMU, do the following steps:

To see the \$PQTMIMUTYPE message, do the following:

- 1- Send \$PAIR6010,3,1*0F . You should get response \$PAIR001,6010,0*0C
- 2- Send \$PQTMSAVEPAR*5A
- 3- Power cycle the USB and you shall be able to see the \$PQTMIMUTYPE message output shortly after power up.

After running \$PQTMSAVEPAR*5A, you need to reset the module to see the \$PQTMIMUTYPE message as it is only output one time, shortly after the module boots up.

To calibrate the module and see the \$PQTMDCAL message (Section 3.1.1 in DR Application Note), do the following:

1. Send \$PAIR6010,2,1*0E
2. Send \$PQTMSAVEPAR*5A
3. Power cycle module, you should see output afterwards.

After that, you have successfully enabled the message. Now you need to calibrate the DR by driving, the DR will be calibrated once the <CalState> of \$PQTMDCAL (2nd parameter) shows "2" (see Section 3.1.1. in DR&RTK Application Note). Please refer to Section 2.1.3 in the DR&RTK Application Note for calibrating the module successfully.