5G NB-IOT & GNSS KIT

BG96 Technical Specifications & User Manual

Purpose of the Document

The purpose of this document is to explain the technical specifications and manual for using the 5G NB-IoT & GNSS board.

Document History

Version	Author	Date	Description
A	5G HUB	04.05.2019	Initial Document
B	5G HUB	06.08.2019	Add Arduino IDE instructions
C	5G HUB	06.10.2019	Add Serial Interface
D	5G HUB	09.11.2019	Add more information about Arduino IDE
E	5G HUB	01.02.2020	Update Information regarding Input voltage and Current
F	5G HUB	07.11.2020	Update pictures and information
G	5G HUB	02.28.2021	Update pictures and add download Section

Table of Contents

1 Package contents:

1.1 NB-IoT Kit Package:

- Hardware board
- One USB cable
- One LTE & GPS antenna

1.2 Download

Arduino software 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 LTE&GNSS modem driver for Windows OS:
https://github.com/5ghub/5G-NB-IoT/tree/master/Driver

Download and Install QNavigator and QCOM tools for Quectel BG96 here:
https://github.com/5ghub/5G-NB-IoT/tree/master/Tools

Download and install Arduino library (**5G-NB-IoT_Arduino.zip**) here:
https://github.com/5ghub/5G-NB-IoT

All the following software can be installed from the GitHub location here:
https://github.com/5ghub/5G-NB-IoT

LTE cellular connectivity on Windows OS

2 General Description

2.1 Overview

The NB-IoT kit is a cellular and GPS kit that can be used for the 5G wireless technology. The kit includes a hardware board, LTE&GPS antenna, and USB cables. The board is a powerful board that features a microcontroller and wireless modem. The microcontroller is an Atmel's SAMD21G18A MCU which features a 32-bit ARM Cortex® M0+ core. The wireless modem is BG96 which is an embedded IoT (LTE Cat-M1, LTE Cat-NB1 and EGPRS) wireless communication module. BG96 wireless modem provides a maximum data rate of 375Kbps downlink and 375Kbps uplink. It features ultra-low power consumption, provides data connectivity on LTE-TDD/LTE-FDD/GPRS/EDGE networks, and supports half-duplex operation in LTE networks. It also provides GNSS to meet customers' specific application demands

The board provides rich sets of Internet protocols, industry-standard interfaces (USB/UART/I2C/Status Indicator) and abundant functionalities. The board offer a high integration level and enables integrators and developers to easily design their applications and take advantage of the board low power consumption, many functionalities, and USB drivers for Windows 7/8/8.1/10, Linux and Android.

The kit board is a rich hardware board that can be used for the latest 5G wireless technology and enables a variety of smart and 5G applications for devices, and acts as a great educational tool for learning about 5G and 32-bit application development. It enables large number of applications such as wireless POS, smart metering, tracking, smart transportation, smart buildings, smart city, and smart homes.

The board is also compatible with Arduino and Arduino software (IDE). Arduino sketches and examples are provided with the kit and additional sketches can be developed and uploaded to the board.

Figure 1. Hardware Board – unboxed with one LTE/GPS antenna and one USB cable

2.2 Key Features

- Atmel ATSAMD21G18 MCU
- Quectel BG96 NB-IoT module
- External GPS antenna Connector
- External LTE antenna connector
- Supports LTE NB-IoT and Machine Type Communications (MTC)
- Supports EGPRS
- Global Frequency Band B1/B2/B3/B4/B5/B8/B12/B13/B18/B19/B20/B26/B28/B39 (B39 for Cat.M1 only) for LTE and 850/900/1800/1900MHz for EGPRS
- Supports the protocols TCP/UDP/PPP/ SSL/ TLS/ FTP(S)/ HTTP(S)/ NITZ/ PING/ MQTT
- Supports SMS
- Supports GNSS technology (GPS, GLONASS, BeiDou/Compass, Galileo, QZSS)
- Compact board size of 58mm x 42mm
- Nano USIM card slot
- Arduino IDE Compatible
- Works with Windows, Linux, or Android
- Ready for smart applications and development (smart home, smart city, smart transportation, smart metering, smart farming, smart waste management, asset tracking, location, navigation, mapping, and timing applications). Application such as Gas Detector, Soil PH Tester, Optical Sensor, Machinery Alarm System, Irrigation Controller, Elevator, Asset Tracking Electronics, Person/Pet Tracking, Water/Gas Metering, Smart Parking System, Fire Hydrant, Smoke Alarm, Trash Bin, Street Lighting
- The board can operate on an external power supply of 3.3V to 5V. The recommended voltage is 5V.
- The board can be powered via the USB connector or with an external DC power supply. The power source is selected automatically.
- External DC Power supply (non-USB) can be provided from an AC-to-DC adapter (such as a wall-wart) or battery, and can be connected using a 2.1mm center-positive plug connected to the board's power jack, or directly to the GND and VIN pins.
- Each of the 14 general purpose I/O pins on the board can be used for digital input or digital output using pinMode(), digitalWrite(), and digitalRead() functions. Pins used for PWM can be using analogWrite() function. All pins operate at 3.3 volts. Each pin can source or sink a maximum of 10 mA and has an internal pull-up resistor (disconnected by default) of 20-60 K ohm.

2.3 Overview Diagrams

Figure 2. Hardware Board Overview Diagram – Top and Bottom View

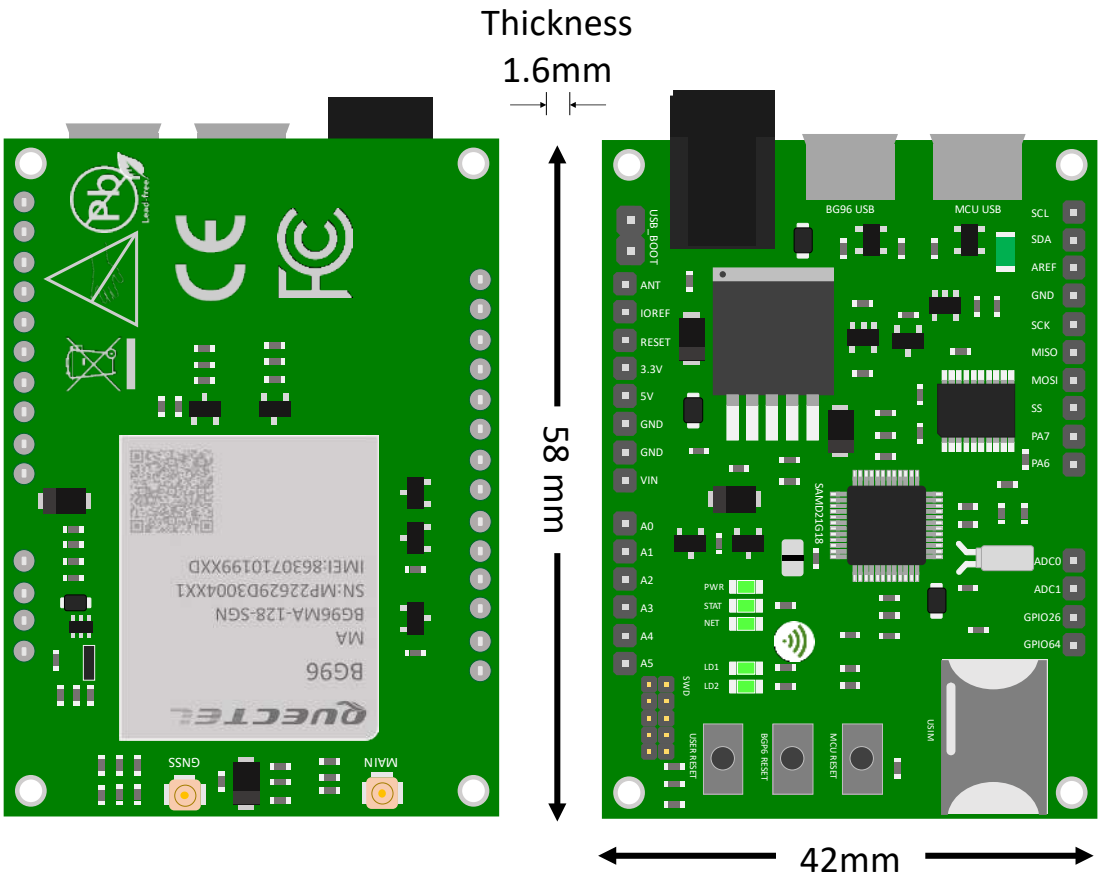


Figure 3. Top and Bottom Views

2.4 Physical Characteristics

The width and length of the board is 48mm (width) by 52 mm (length). The board have four screw holes in each corner that allows the board to be attached to a surface or case.

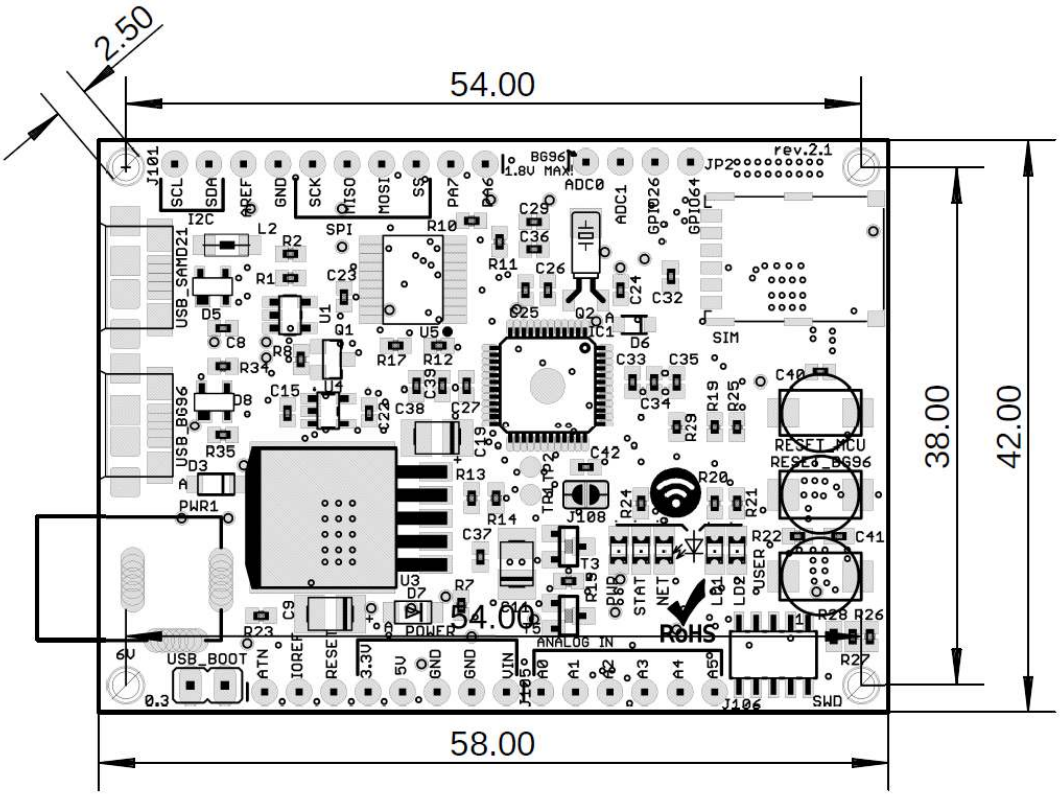


Figure 4. Hardware Board Top View – External Dimensions

2.5 Peripherals – Key Components

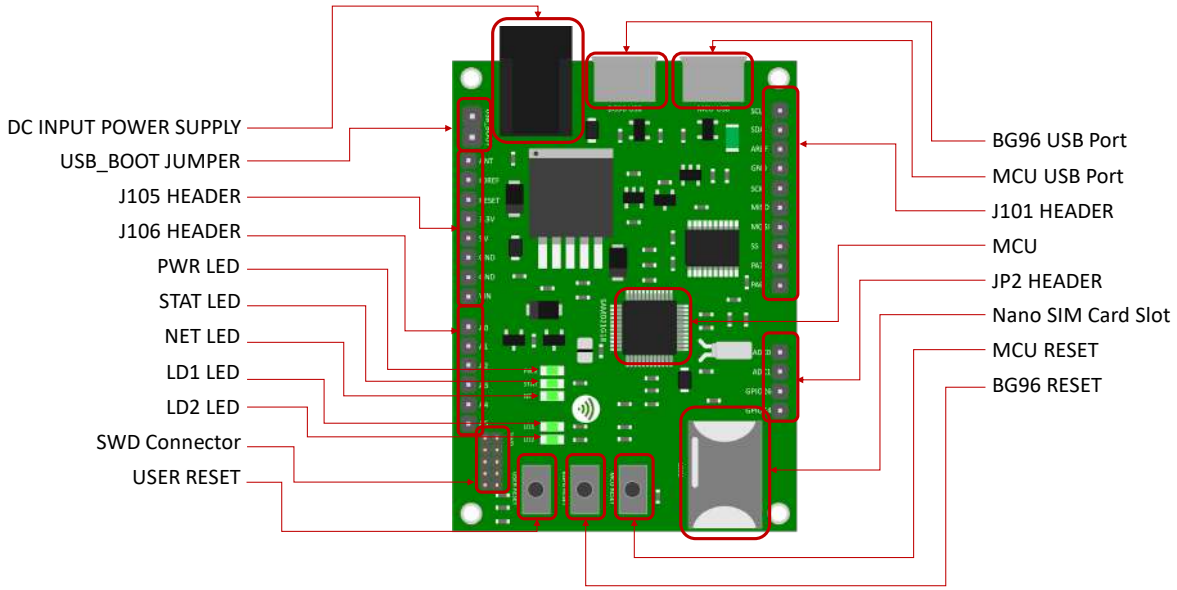


Figure 5. Board Top Side – Key Components

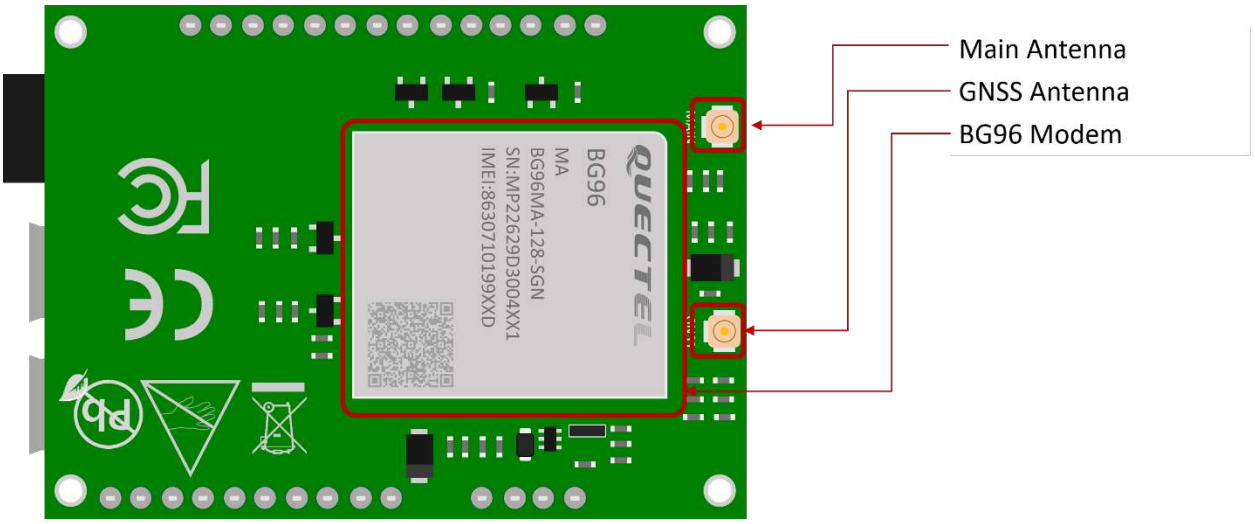


Figure 6. Board Bottom Side – Key Components

2.6 Peripherals – IO Connections

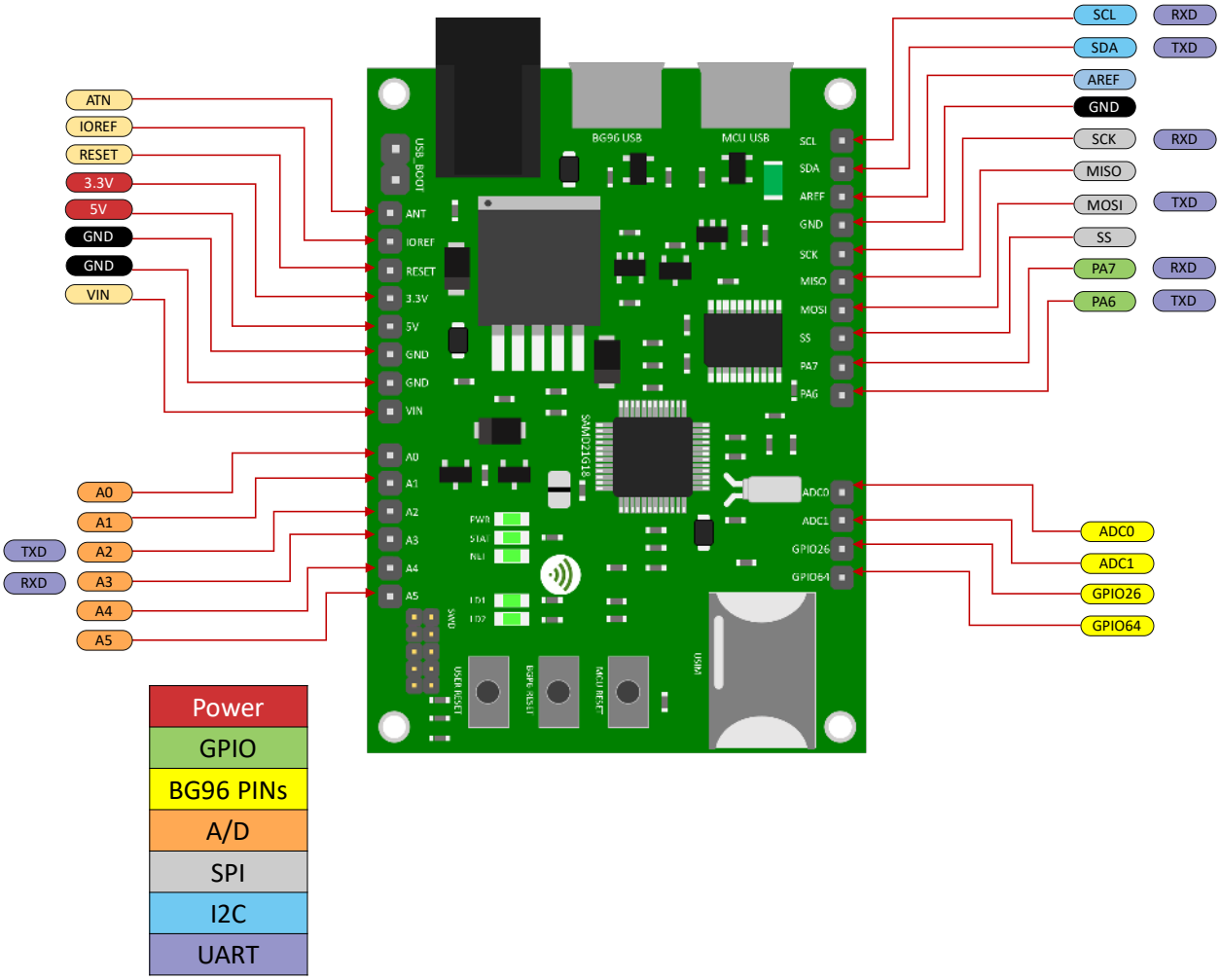


Figure 7. Board Extension Connectors

* I2C interface lines might be configured as USART interface SDA line can work then as USART TXD and SCL line can work as USART RXD)

** MOSI and SCK lines might be configured as USART interface (MOSI line can work then as USART TXD and SCK line can work as USART RXD)

2.7 Hardware Specification

Technical Specification	
Microcontroller (MCU)	Atmel ATSAMD21G18, 32-Bit ARM Cortex M0+
Clock Speed	48 MHz
Flash Memory	256 KB
SRAM	32 KB
NB-IoT Module	Quectel BG96
Dimension	48mm (width) by 52 mm (length)
Weight	18 grams
Power Supply	DC Power Supply (3.8-5V), USB (5V), VIN (3.8-5V), or Battery
LED	LED1, LED2, Power LED, Status LED, Netlight LED
Interfacing Logic Voltage Level (Operating Voltage)	3.3V
Voltage output	5V, 3.3V
RESET buttons	Two; one for MCU and one for BG96
User-defined Button	1 connected to MCU
General-purpose digital I/O Pins	14 (A0-A5, PA6, PA7, SS, MOSI, MISO, SCK, SDA, SCL)
GPIO	2 connected to BG96
ADC	2 connected to BG96
USB	2
I^2C	1
SPI	1
UART	1
ADC pins	6 (8/10/12-bit ADC channels)
DAC pin	1 (10-bit DAC)
External interrupts	14 (All general-purpose PINs)
PWM pin	6
DC Current per I/O Pin	10 mA
JTAG Debug	Cortex Debug Connector (Single Wire Debug)
USIM	Nano
GNSS	GPS, GLONASS, BeiDou/Compass, Galileo, QZSS
Antenna	1 main antenna and 1 GPS antenna
Band	LTE-FDD, B1/B2/B3/B4/B5/B8/B12/ B13/B18/B19/B20/B26/B28 LTE-TDD: B39 (for Cat M1 only)
Certification	FCC, CE
Mobile Operator Certification	Verizon and currently for AT&T

Notes:

- UART can be programmed through any of general-purpose pins.
- SPI can be programmed through any of general-purpose pins.

2.8 PIN Description

PIN	DIRECTION	Description
DC Power Jack	I	The board can be supplied with power either from the DC power jack (3.8V-5V), the USB connector (5V), or the VIN pin of the board (3.8V-5V)
LED (PWR)	O	LED is lighted on when the board is power on from the MCU USB port
LED1 (USER)	O	LED which can be controlled from MCU (D25). When the pin is HIGH value, the LED is on, when the pin is LOW, it is off
LED2 (USER)	O	LED which can be controlled from MCU (D26). When the pin is HIGH value, the LED is on, when the pin is LOW, it is off
LED (NET)	O	Indicate the BG96 operation status
LED (STAT)	O	Indicate the BG96 network activity status
MCU RESET button	I	Reset the MCU
BG96 RESET button	I	Reset the BG96 module
User Button	I	Connected to digital pin, D0, of MCU and can be used for user-defined purposes
IOREF	O	Provides the voltage reference with which the MCU operates. A device can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V
3.3V	O	3.3V generated by the on-board regulator. Maximum current drawn is 300 mA. The regulator also provides power to the MCU and BG96
5V	O	5V generated from the board. The board can be supplied with power either from the DC power jack (3.3V - 5V), the USB connector (5V), or the VIN pin of the board (3.3-5V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator and can damage the board if it is not sufficiently regulated (This is not recommended)
GND		Ground
VIN	I	Input voltage to the board when it uses an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or if supplying voltage via the power jack, access it through this pin
A0	IO	Six analog inputs which can provide up to 12 bits of resolution (i.e. 4096 different values). By default, each input measures from ground to 3.3 volts, though is it possible to change the upper end of their range using the AREF pin A0 can also be used as a DAC output and provides a 10 bit voltage output with analogWrite() function Analog pins can be used as GPIOs
A1	IO	
A2	IO	
A3	IO	
A4	IO	
A5	IO	
SCL	IO	I2C. The SCL (clock line). Can be used as GPIO
SDA	IO	I2C. The SDA (data line). Can be used as GPIO

AREFA	I	Input reference voltage for the analog inputs used for either he ADC or the DAC
SCK	IO	SPI Interface. Can be used as GPIO
MISO	IO	SPI Interface. Can be used as GPIO
MOSI	IO	SPI Interface. Can be used as GPIO
SS	IO	SPI Interface. Can be used as GPIO
PA7	IO	GPIO. Can be used as GPIO
PA6	IO	GPIO. Can be used as GPIO
Cortex Debug Connector	IO	Using Single Wire Debug to burn bootloader and debug the board
ADC0	I	Connected to BG96. General purpose analogue to digital converter
ADC1	I	Connected to BG96. General purpose analogue to digital converter
GPIO26	IO	Connected to BG96. General purpose IO
GPIO64	IO	Connected to BG96. General purpose IO
USIM	I	Used to insert a Nano USIM. Connected to BG96
USB Boot	I	Connected to BG96. Force the BG96 to enter emergency download mode
USB1	IO	Connected to MCU
USB2	IO	Connected to BG96

Precaution

The board runs at 3.3V. The maximum voltage that the I/O pins can tolerate is 3.3V. Applying voltages higher than 3.3V to any I/O pin could damage the board

2.9 BG96 chipset

All functionality of the BG96 shipset shall be implemented excluding the following features. That is, the following features are not supported [1][2].

- Audio, Earphone, and Codes are not supported.
- PCM and I2C are not supported
- PSM_IND and AP_READY are not supported

2.10 Interface between SAM21D and BG96

The Microcontroller communicates with the BG96 through UART interfaces:

- **UART1:** (PA12/PA13/PA14/PA15). Used for data transmission and AT command communication 115200bps by default. The default frame format is 8N1 (8 data bits, no parity, 1 stop bit) Support RTS and CTS hardware flow control.
- **UART3:** (PB23/PB22). Used for outputting GNSS data or NEMA sentences 115200bps baud rate.

3 Using the Board with Arduino IDE

3.1 Installing the Software

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

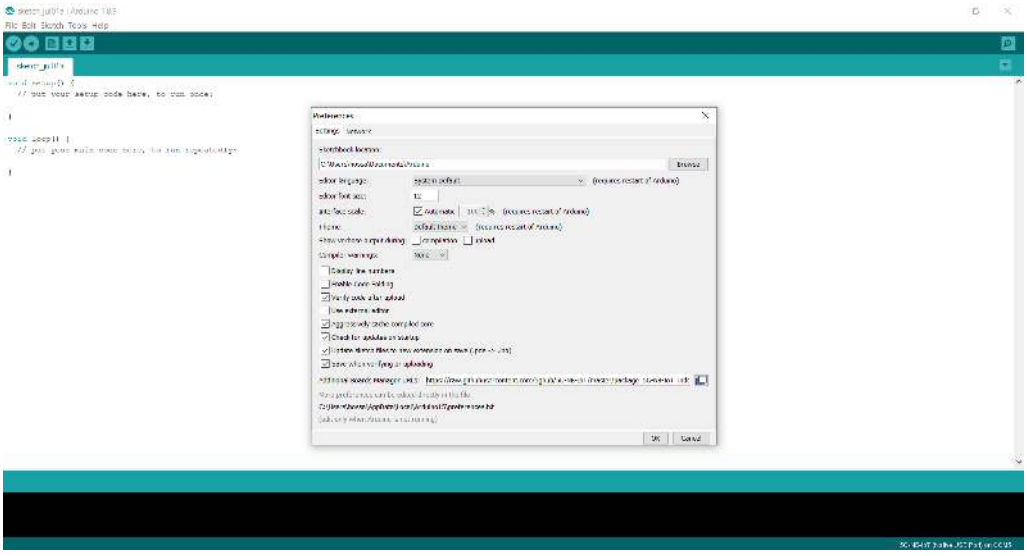
1- Install Arduino IDE for Windows from the following web site
https://www.arduino.cc/en/Main/Software
2- Download and Install Quectel Driver here:
This will install Quectel driver on Windows.
https://github.com/5ghub/5G-NB-IoT/tree/master/Driver
3- Download and Install QNavigator tool for Quectel BG96 here:
https://github.com/5ghub/5G-NB-IoT/tree/master/Tools
4- Download and save the file **5G-NB-IoT_Arduino.zip** here:
https://github.com/5ghub/5G-NB-IoT

All the following software can be installed from the GitHub location here:

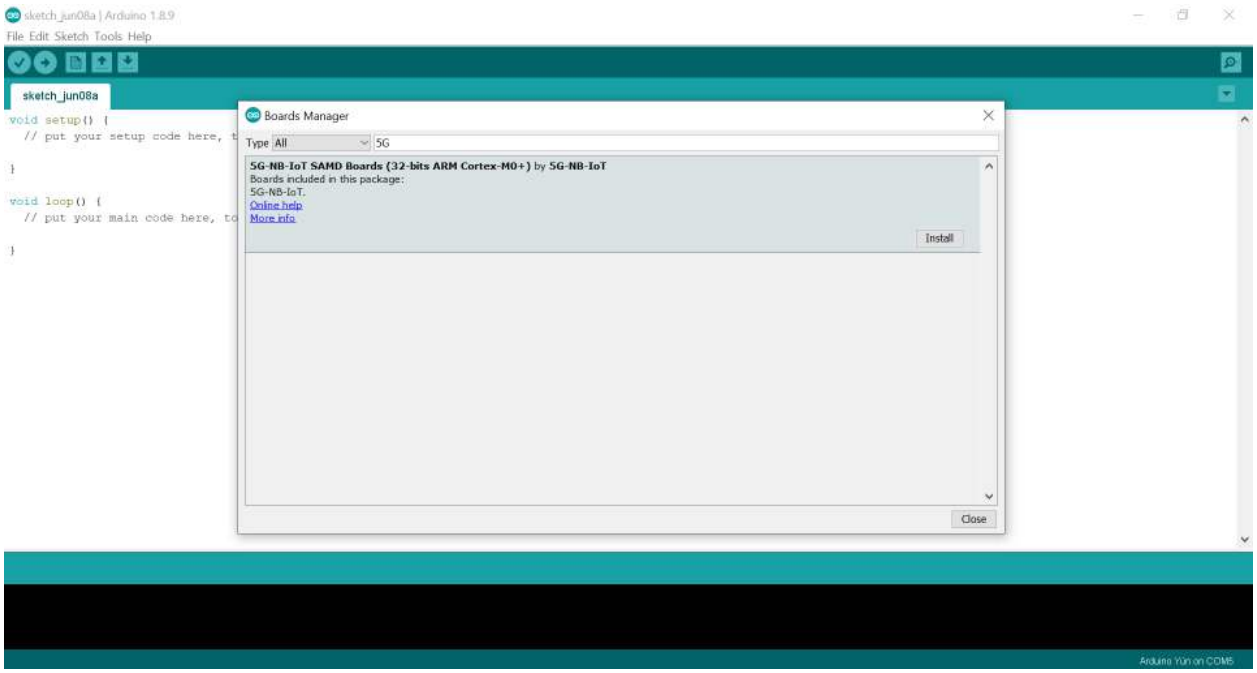
https://github.com/5ghub/5G-NB-IoT

3.2 Setting Up Arduino IDE

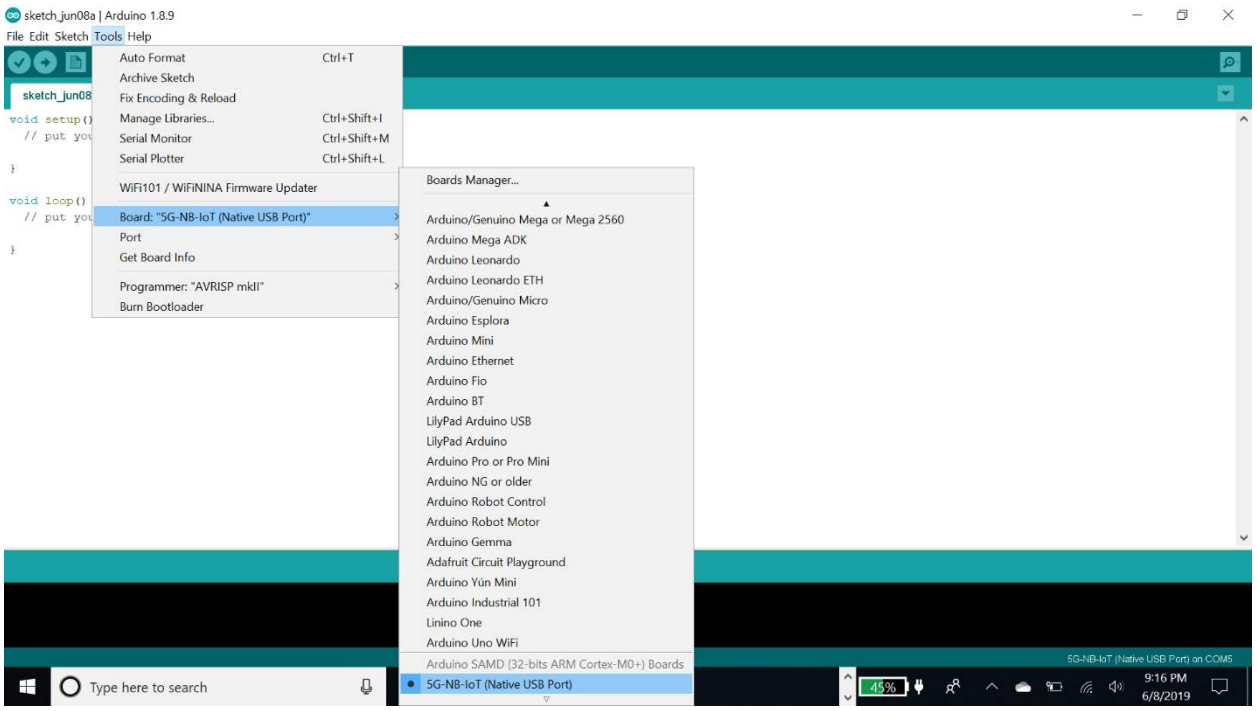
1- Connect a USB cable from the **computer USB** to the **USB_SAMD21 port** to power on the board.
2- Connect a USB cable from the **computer USB** to the **USB_BG96 port** to connect to the BG96.
3- Launch Arduino IDE and choose **File->Preferences**. In the Additional Boards Manager URLs, insert the following URL:
https://raw.githubusercontent.com/5ghub/5G-NB-IoT/master/package_5G-NB-IoT_index.json



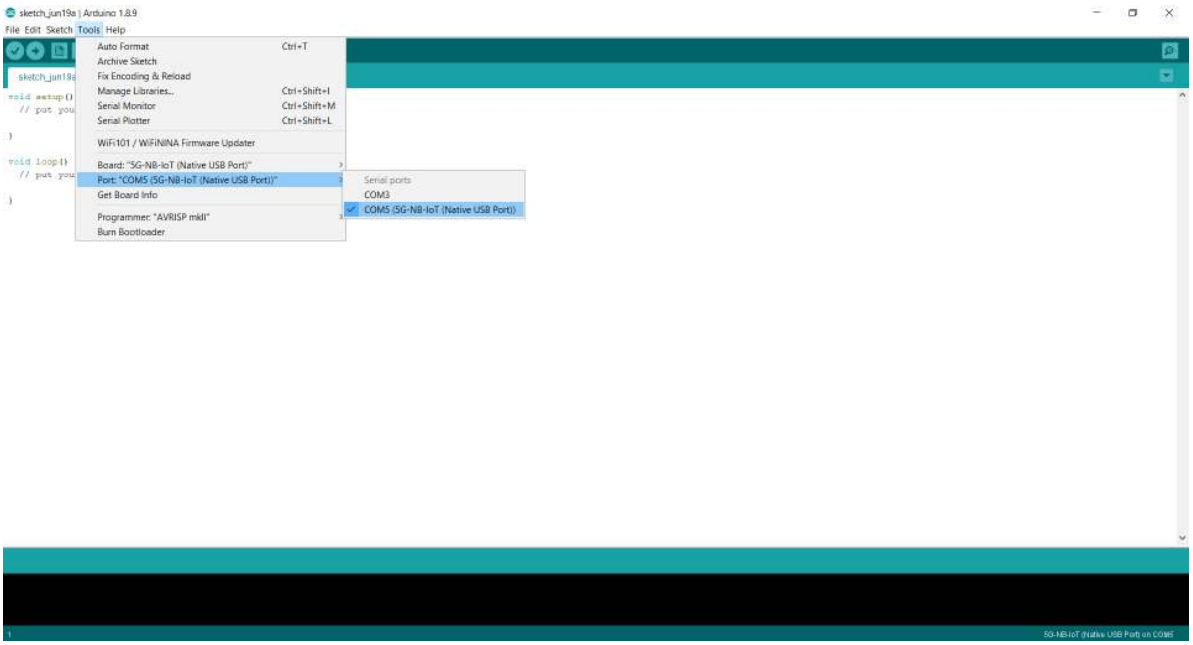
4- In Arduino IDE, choose **Tools->Board->Boards Manager**, select and install "**5G-NB-IoT SAMD Boards**".



5- Choose "**5G NB-IoT (Native USB Port)**"

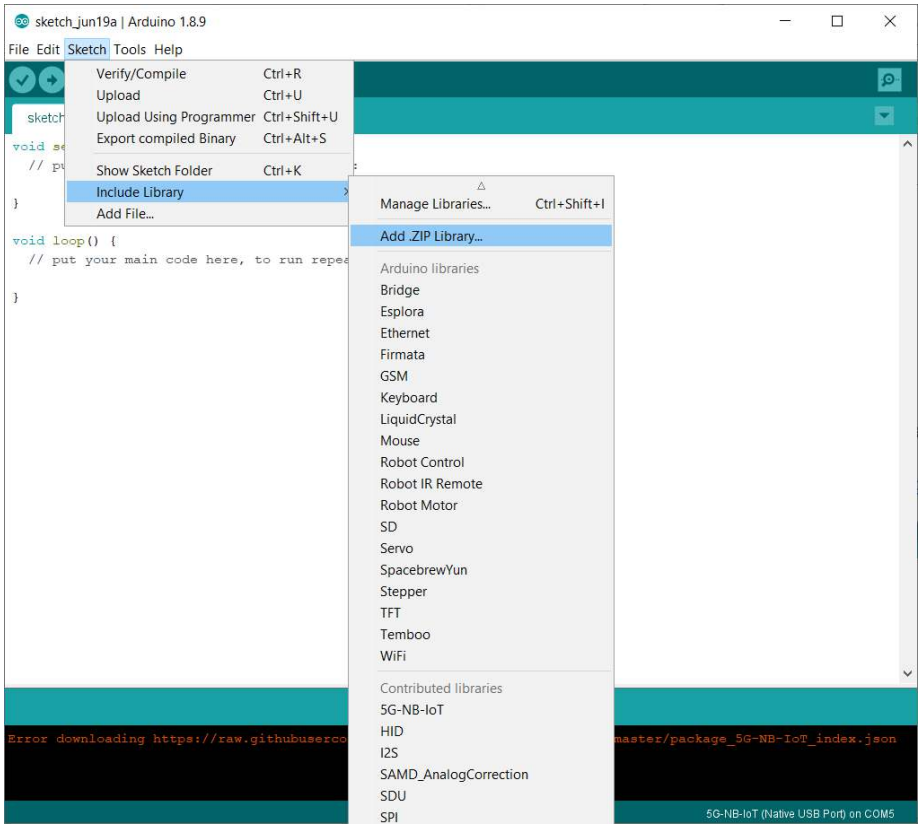


6- In the Arduino IDE, Choose **Port** and select the serial port where the board appears.



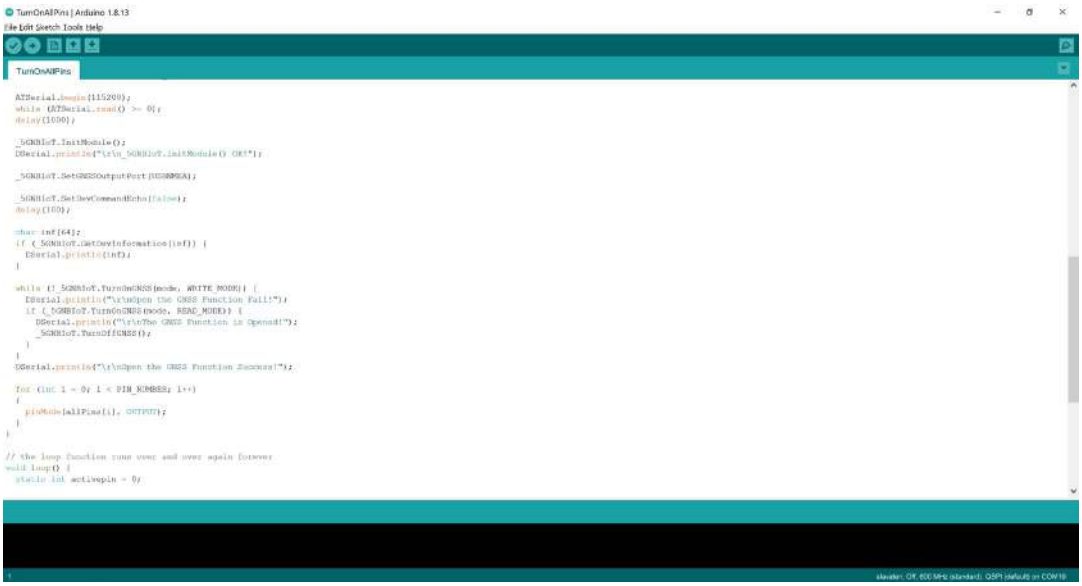
7- In the Arduino IDE, Choose **Sketch->Include Library->Add .Zip Library** and select the file **5G-NB-IoT_Arduino.zip**

You are now ready to start running Arduino sketches and projects.

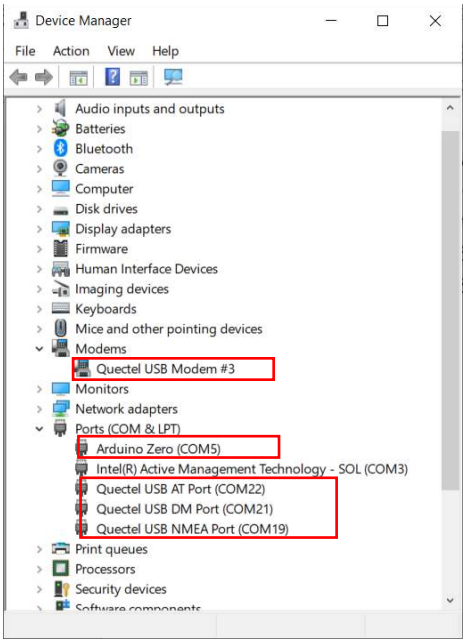


3.3 **Running Arduino Sketch**

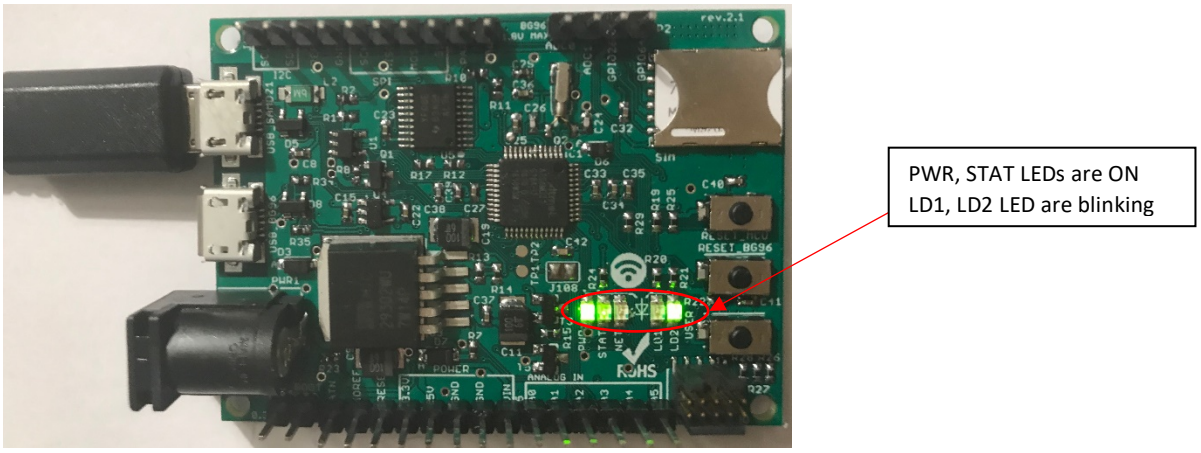
1- Using Arduino IDE, open the Arduino sample **TurnOnAllPins.ino**, choose **Sketch->Upload**.



2- In the Device manager, the Arduino and Quectel USB Modem and ports shall show up as in this screen shoot:



After compiling and uploading the Arduino sketch, the LEDs will blink and the BG96 is enabled as in the following picture:



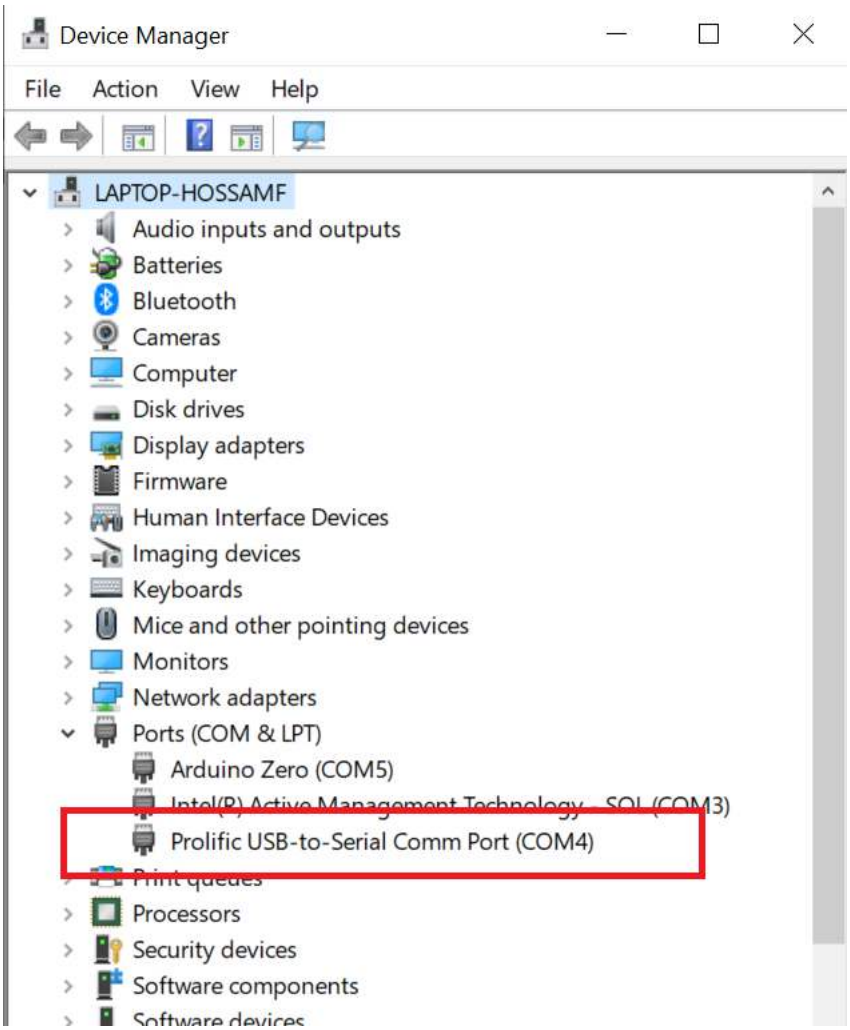
The following table show different five LEDs status after uploading this sketch

LED	Behavior	Description
PWR	ON	Indicates the MCU is powered on
STATE	ON	Indicates the BG96 is powered on
NET	Flicker slowly (200ms High/1800ms Low)	Network Searching
	Flicker slowly (1800ms High/200ms Low)	Idle
	Flicker slowly (125ms High/125ms Low)	Data transfer is ongoing
LD1	Blinking	User controlled LED
LD2	Blinking	User controlled LED

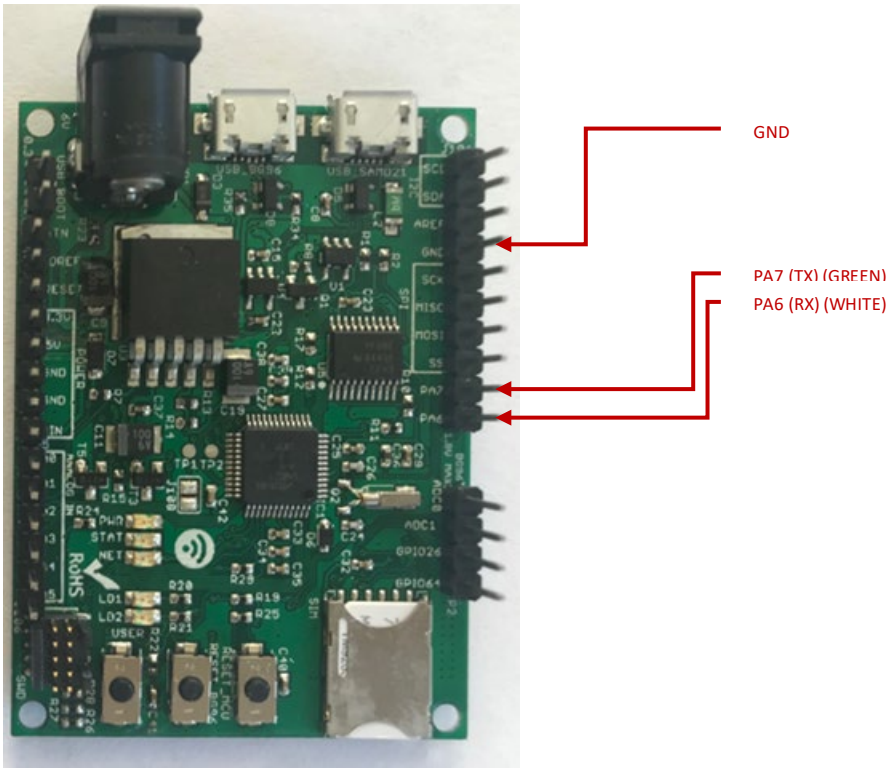
4 Using the Arduino Sketches with Serial Interface

The board has a serial interface where output can be sent via the serial interface to the computer. Use USB-to-Serial cable such as the Prolific USB-to-Serial cable.

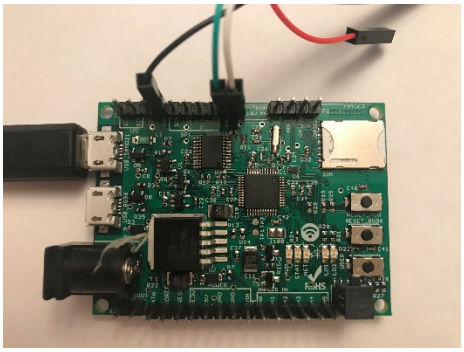
Connect the USB-to-Serial cable to the PC, and the device manager shall display a new COM port.



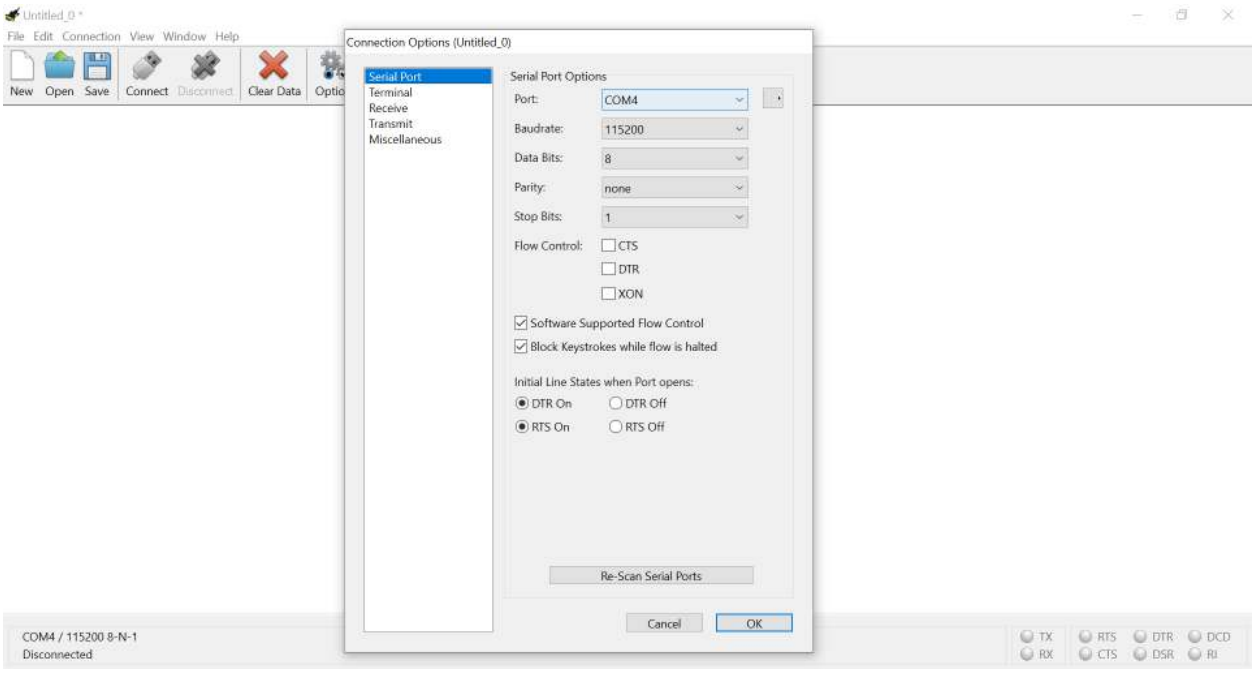
The board has the serial interface on the J101 as in the following figure.



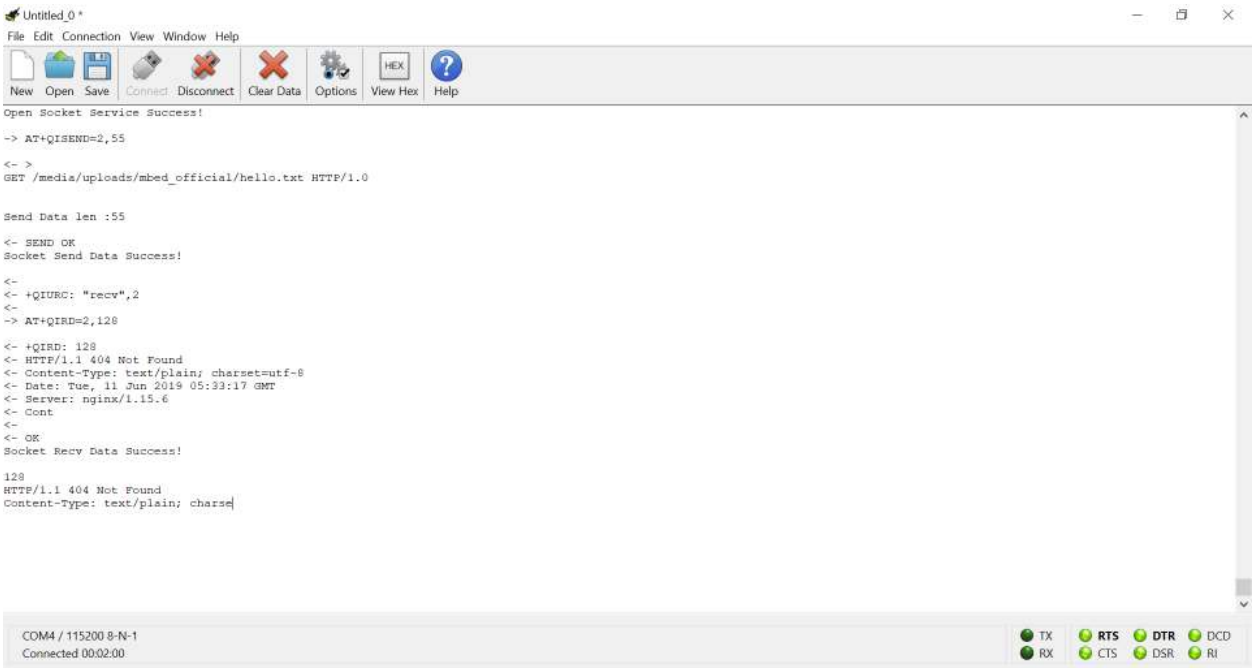
Connect the USB-to-Serial cable from the PC to the board as shown.



Launch any serial port software to connect to the board serial board. Set the COM speed to 115200.

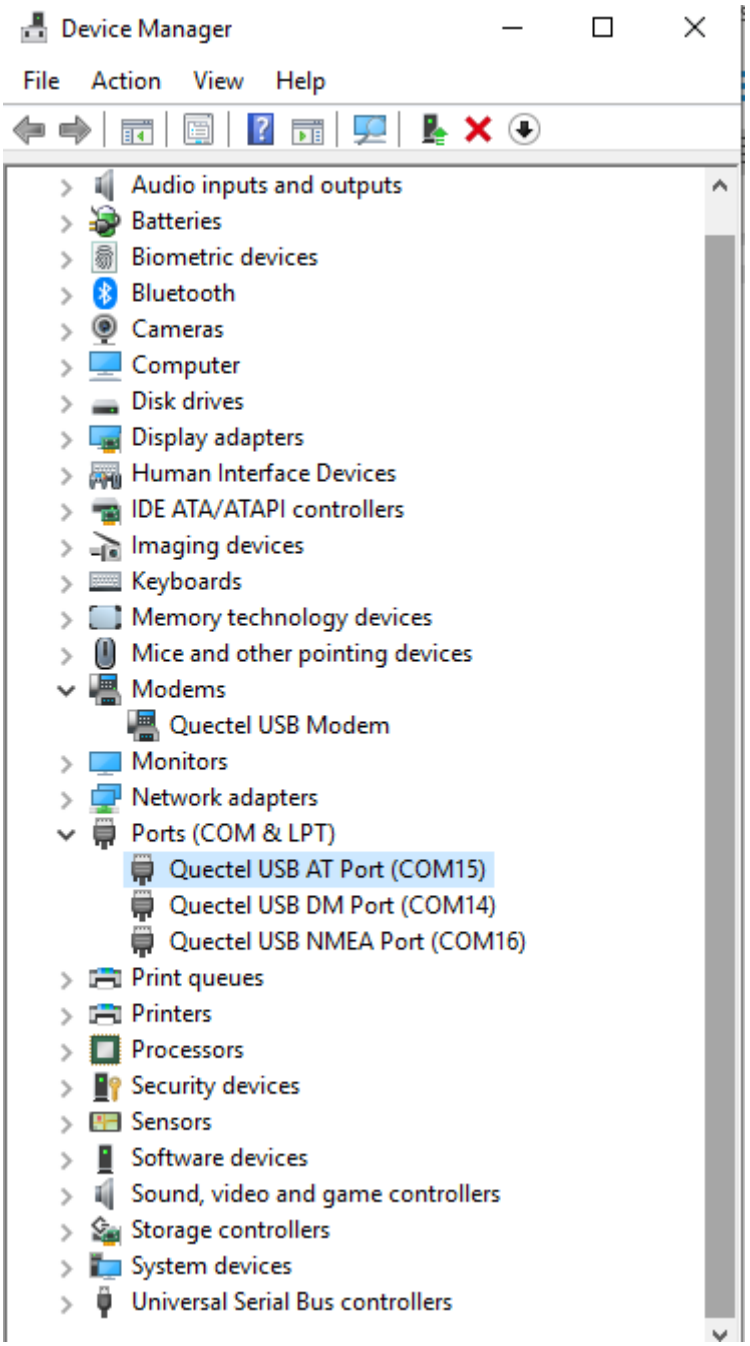


Click connect and the Arduino sketch output shall be displayed on the terminal software

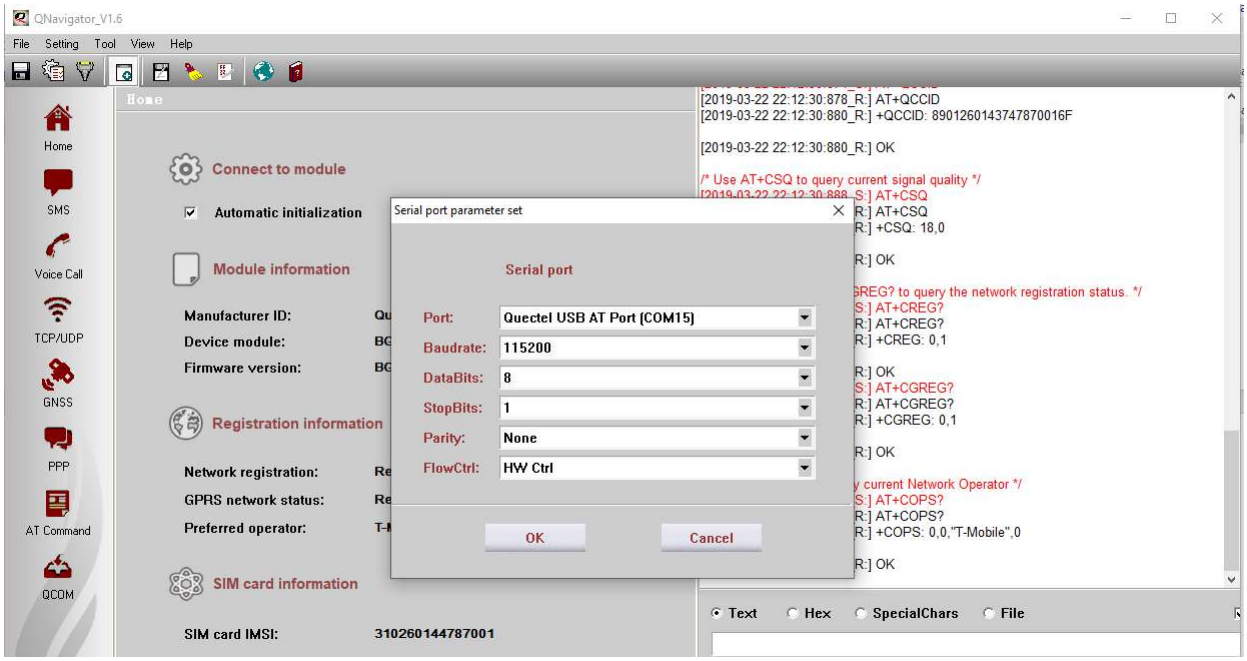


5 Procedure for Operating the BG96 Wireless Unit

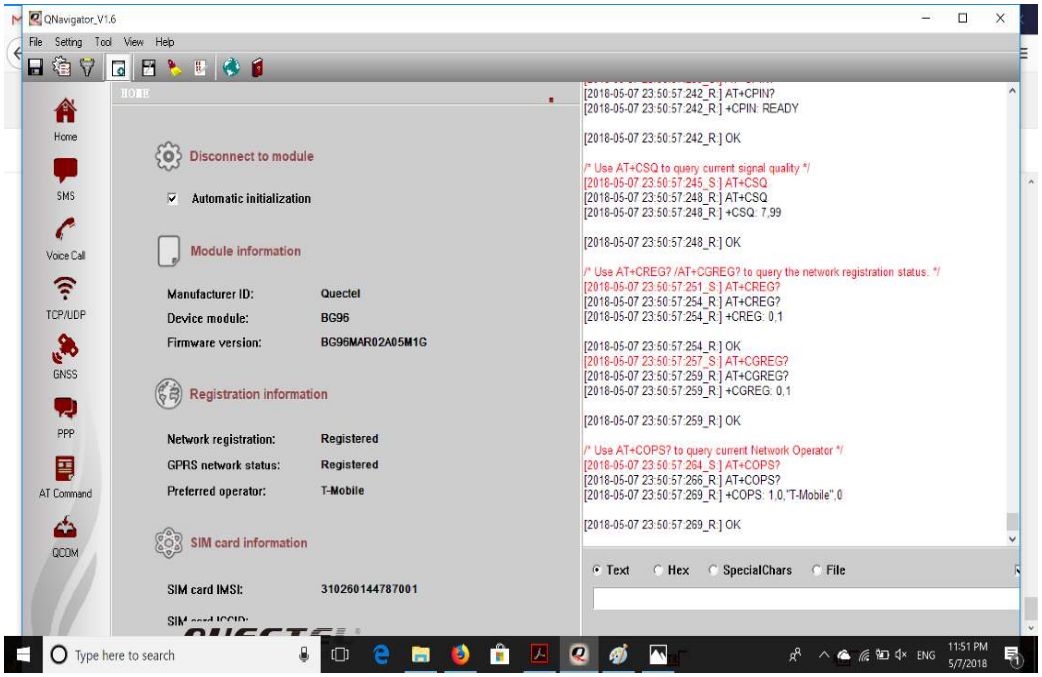
1. Insert a USIM (or a test USIM) into the USIM slot on the hardware board.
2. Connect the LTE/GPS Antenna to the MAIN and GNSS antenna ports respectively on the board.
3. Connect the two USB cables to the board and to the PC (Windows 10 PC).
4. In the Device manager, the Quectel USB Modem and ports are shown up.



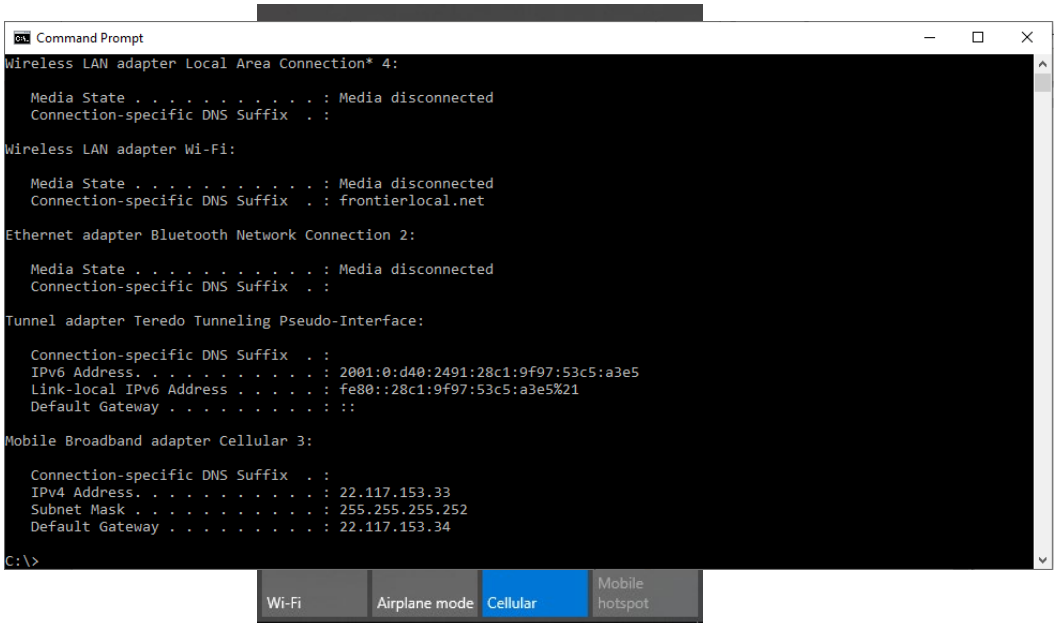
5. Launch QNavigator. Choose the correct Quectel USB AT port in Settings->Serial Port Parameter Set.

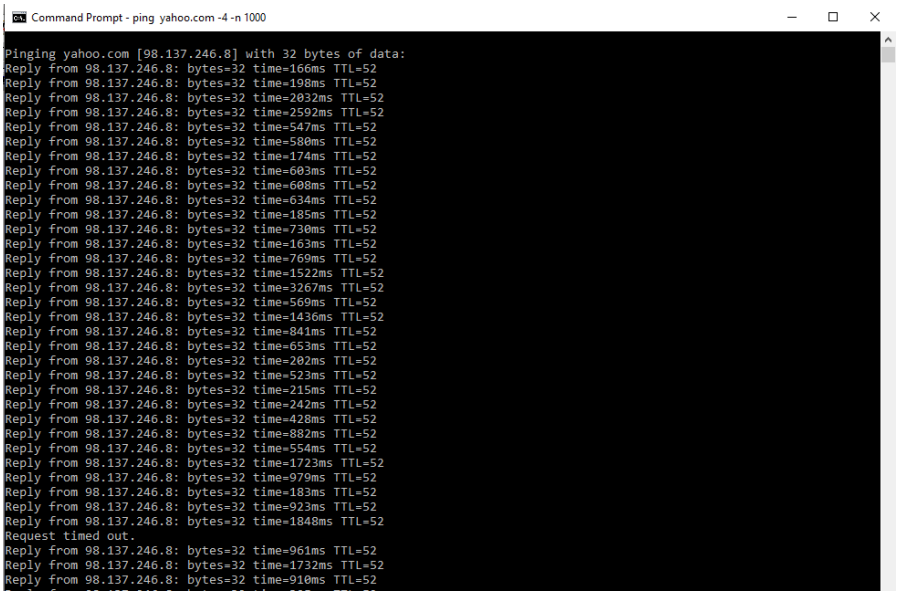


6. Click “Connect”. The board will connect to Mobile operator. Screen shot below shows it connects to T-Mobile network.



7. On the Windows PC, disable all network connections (WiFi or Ethernet) except the Quectel cellular Modem. You can, for example, do a “Ping” to any IP address and this illustrates transmitting and receiving from the T-Mobile cellular network. This is illustrated as in this screen shot:

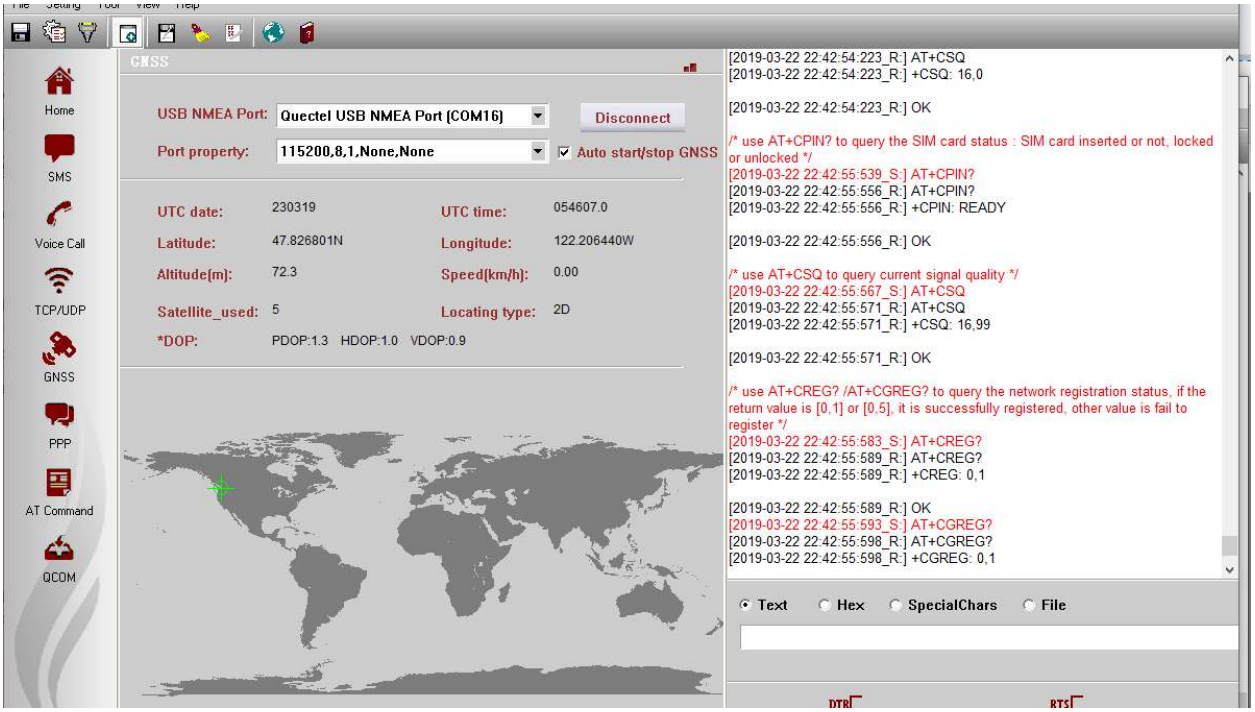


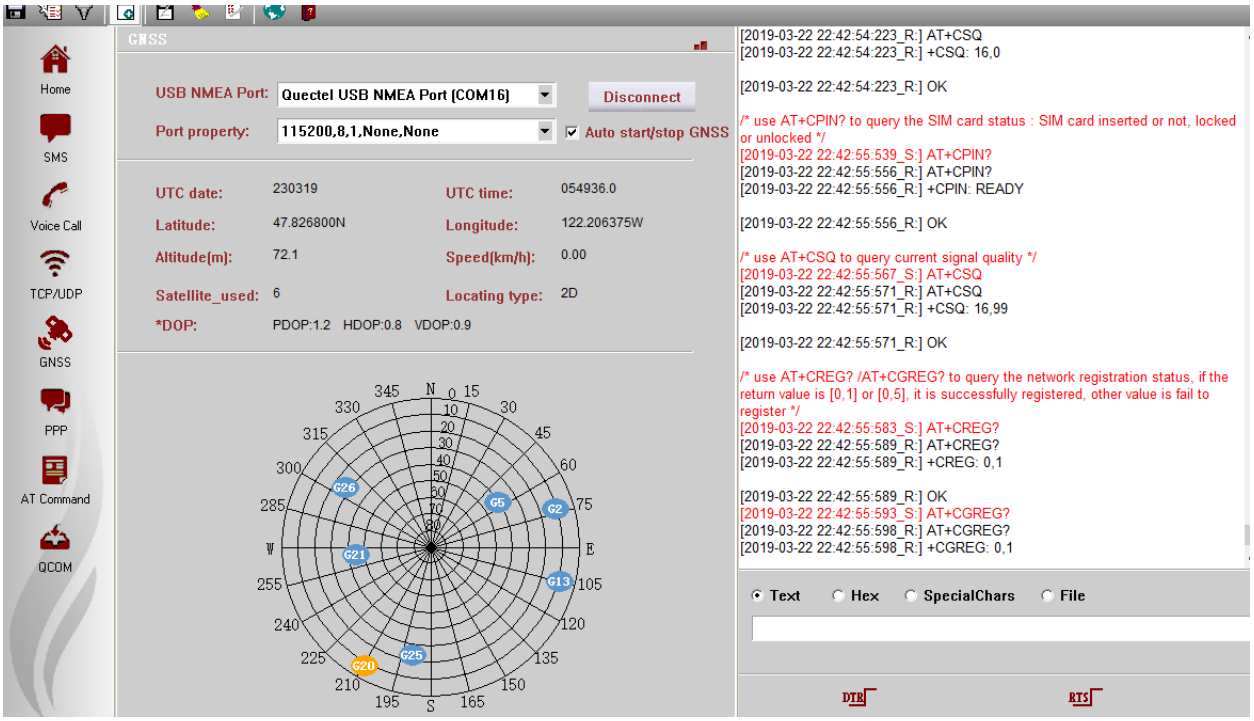


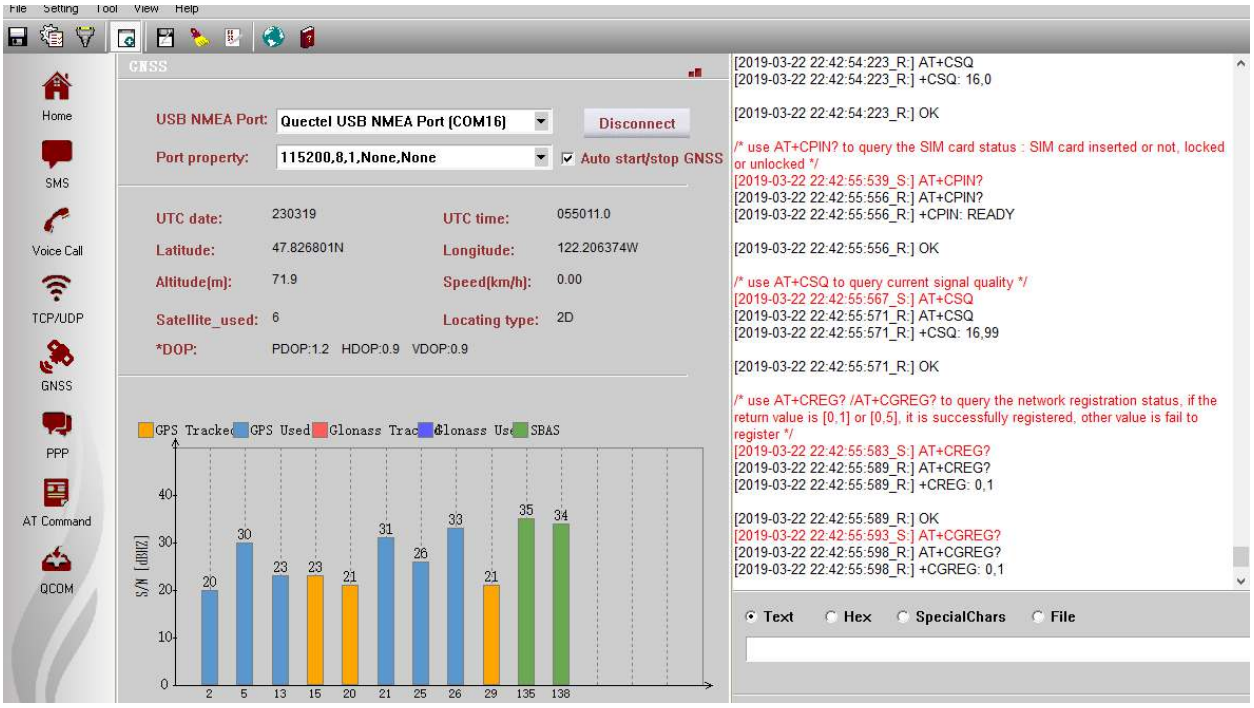
6 Running the GNSS

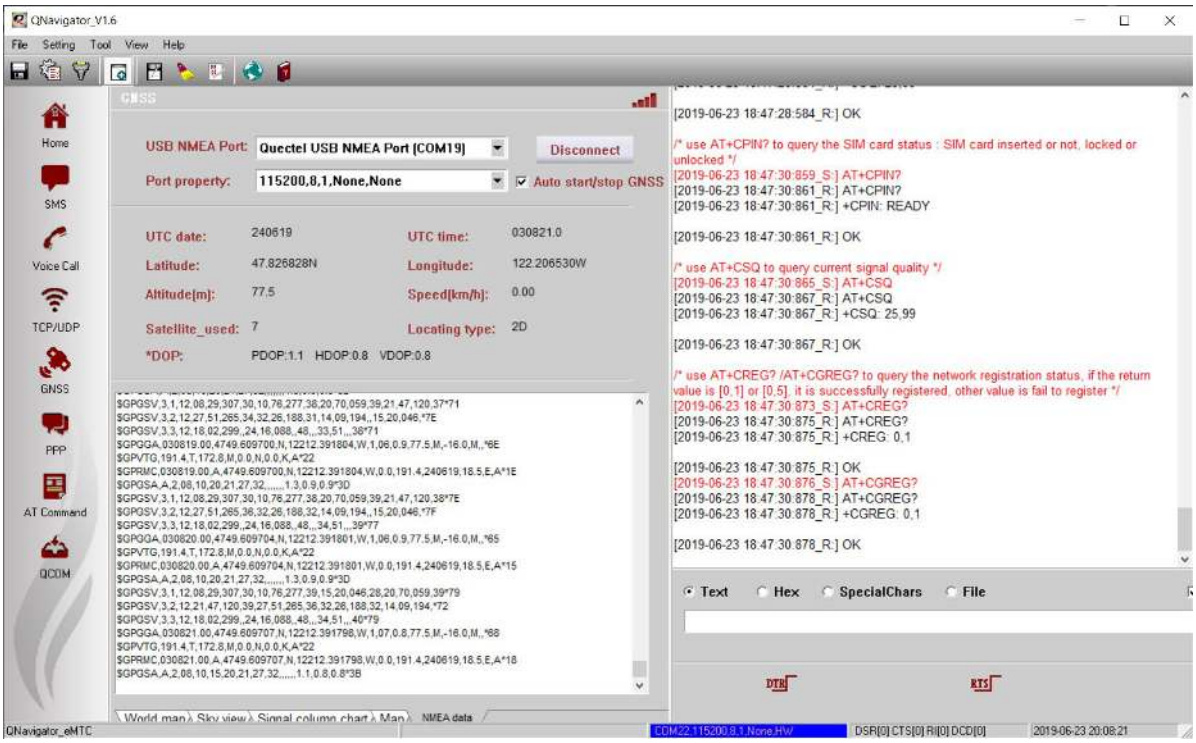
You can use the GNSS module to get location and position information:

1. Connect the second Antenna to the GNSS antenna port on the board. In the QNavaigator, click “GNSS” and then “Connect”. You will get location and positioning information. Sample screen shots as below:









7 AT Commands

The QNavigator can be used to communicated with the transmitter and receiver. For example, you can use the following AT commands:

a. **ATI:** The ATI command displays the module information, such as the name and version
b. **AT+CFUN?:** This commands sets up phone functionality in the module. In our case it is 1, because we require full functionality.
c. **AT+CSQ:** This command is used to get the signal strength and it makes sure that the USIM is receiving signals.
d. **AT+CREG?:** This command is used to enquire about the network registration status. Response 1 means registered.
e. **AT+CGREG?:** This command is also used to enquire the network registration status.
f. **AT+COPS?:** This commands returns the current mode and currently selected operator. (e.g. T-Mobile).
g. **AT+QCFG = "BAND", f,400a0e189f,a0e189f:** In order for the USIM to get registered with the 5G NB network we set up the device to look for the above frequency band configuration.
h. **AT+QCFG = "NWSCANMODE",1:** This command is used to set the scan mode. We set it to 1 to scan for GSM networks to register.
i. **AT+QCFG = "iotopmode",1:** This command is used to specify the network category which is to be searched under the LTE network mode. We set it to 1 to search for LTE Cat.NB1.
j. **AT+QCFG = "NWSCANSEQ",010302:** This command configures the sequence of searching for the network. Our sequence means GSM→LTE Cat.NB1→LTE Cat.M1
k. **AT+CGDCONT=1,"IPV4V6":** This command sets the PDP context, we have set the PDP context to IPV4 and V6.

l. **AT+CGPADDR:** This command returns a list of PDP addresses.
m. **AT+QNWINFO:** This command queries the network information such as access technology, operator's numeric code, the band and channel ID.
n. **AT&W0**: Use this command to stores the current AT command settings to a user defined profile in non-volatile memory

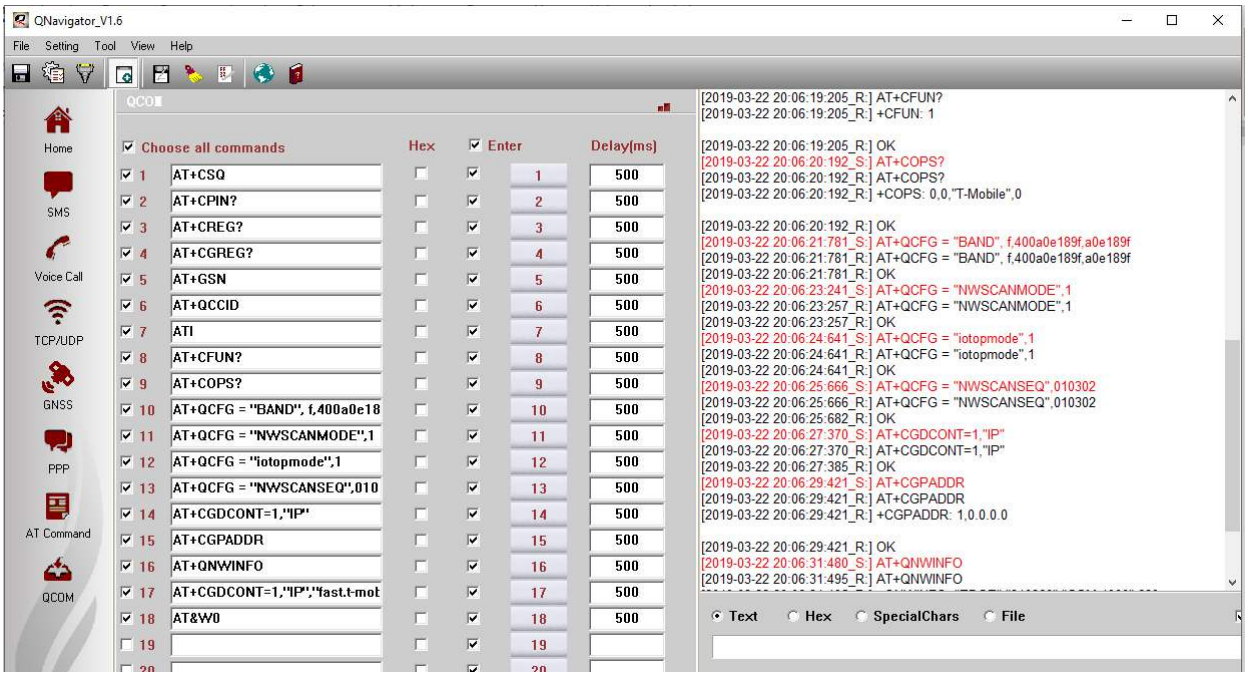
For GNSS to work with QNavigator, make sure the following is issued:

o. **AT+QGPSCFG="outport","usbnmea"**: Use this command to configure various GNSS settings, including NMEA sentences output to the BG96 USB port.

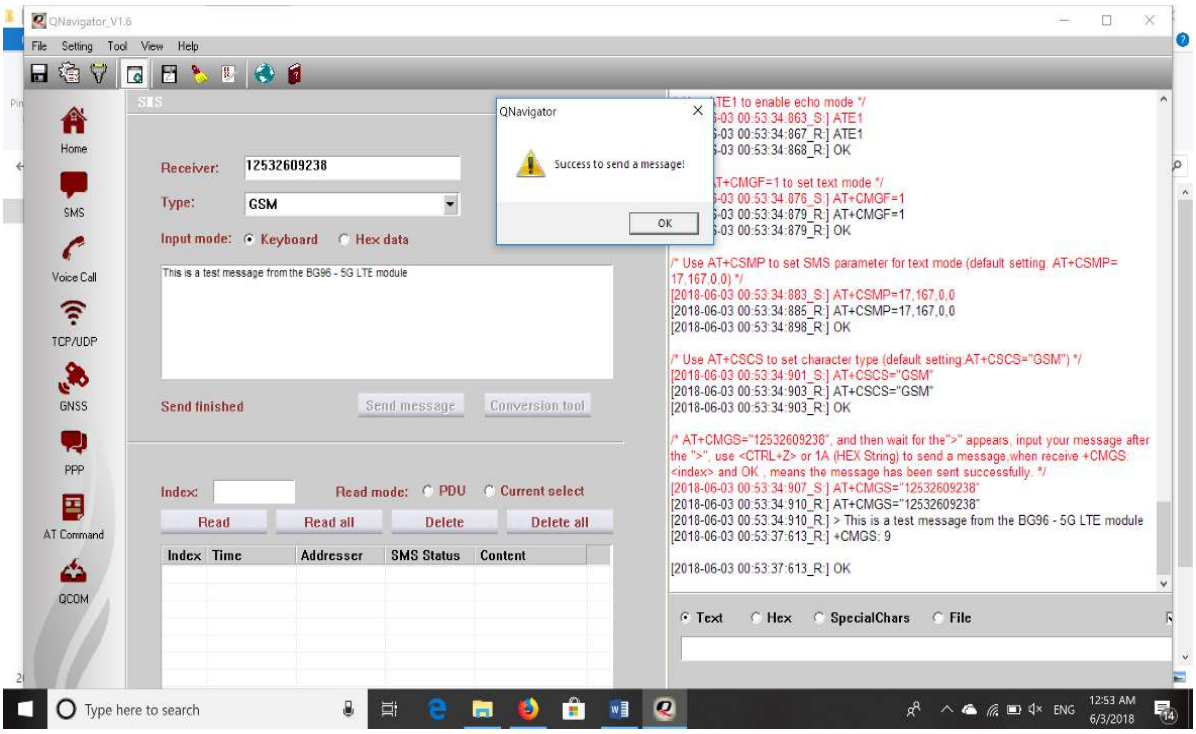
The BG96 is connected to two GPIOs (GPIO26 and GPIO64) which produces a maximum of 1.8V. You can set them using the following commands:

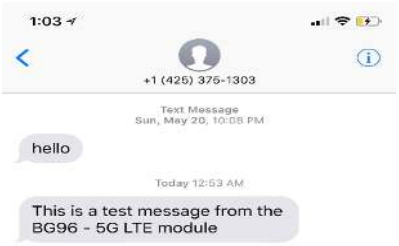
AT+QCFG="gpio",1,26,1,0,7 //configure pin 26 as output, no pull, 17mA drive, clean last configuration
AT+QCFG="gpio",3,26,1 // write pin 26 high value, clean last configuration
AT+QCFG="gpio",1,64,1,0,7 //configure pin 64 as output, no pull, 17mA drive, clean last configuration
AT+QCFG="gpio",3,64,1 // write pin 64 high value, clean last configuration

The following are screen shots of some of these AT commands:



2. You can send SMS from the SMS pane of the QNavigator. Screenshots below shows the SMS transmitted by the device and received on the phone.





8 References

[1] Quectel_BG96_Hardware_Design_V1.2.pdf
[2] Quectel_BG96_Reference_Design_Rev.A_20170814.pdf
[3] Quectel_Antenna_Design_Note_V2.0.pdf
[4] Quectel_RF_Layout_Application_Note_V2.2.pdf
[5] Quectel_QFlash_User_Guide_V2.3
[6] Arduino IDE, https://www.arduino.cc/en/Main/Software
[7] Arduino IDE, https://www.arduino.cc/en/Guide/ArduinoZero
[8] Microchip, “Low-Power, 32-bit Cortex-M0+ MCU with Advanced Analog and PWM”

APPENDIX A – SCHEMATIC