

# NERDS INC.

Sunday, 29 December 2013

## Demystifying the uBlox 6 GPS Module

For an upcoming project I needed to use a uBlox GPS. Specifically, I had to use it with the Raspberry Pi, not an Arduino in Airborne mode (hmm hints about the project? :P) Even though the documentation is very informative, it still lacks a "quick start" guide for us noobs. Even though many blogs have guided me in using this module, I still had to command the GPS to work < 50,000 m (Airborne mode) and had to adjust different configs.

I'll be using a uBlox NEO-6M GPS module for this guide, but any uBlox 6 GPS would work just fine.



### The unofficial uBlox 6 "Quick Start" Guide

#### Making the Connections

The first thing you want to do is connect the GPS to your Pi. The wiring is simple:

uBlox	Raspberry Pi
GND	GND
TX	RX (GPIO 15)
RX	TX (GPIO 14)
VCC	3V3

Once you have made the connections, we need to enable the Pi's serial comm.

Open cmdline.txt file using nano:

```
sudo nano /boot/cmdline.txt
```

Once open and remove 'ttyAMA0', to have this:

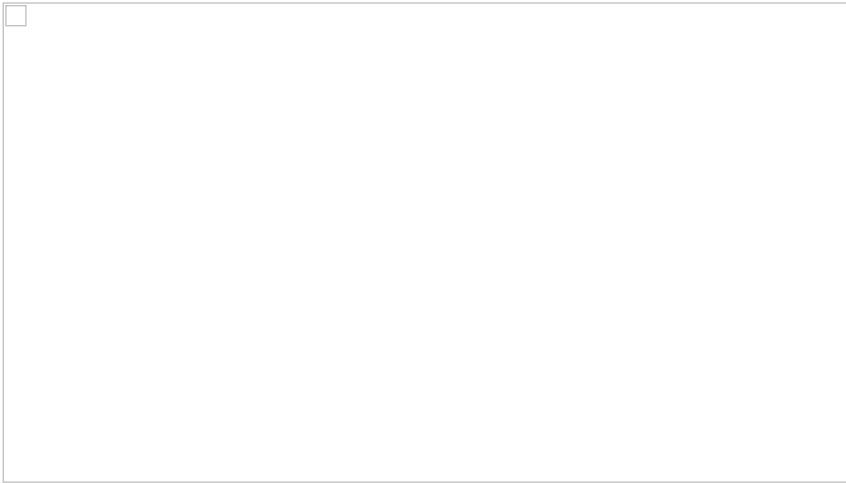
### About Me



Shreyas

I am an artificial intelligence and computer vision enthusiast. I love making crazy things, from rockets to self navigating robots. If you like what I ar doing, please support it, BTC: 1NHpkoYmJsbEkqc3HjGX6x6nPvuTahSr

[View my complete profile](#)



Once you have that use Ctrl-O and Ctrl-X to save and exit.

Then edit the inittab file by:

```
sudo nano /etc/inittab
```

Once open find the "Spawn a getty on Raspberry Pi" line, and put a # before the line just after that. Restart the system.

Install GPSD by typing in:

```
sudo apt-get install gpsd gpsd-clients
```

Then start the serial port,

```
stty -F /dev/ttyAMA0 <baud rate>
```

You can use the baud rate that you want, I used 9600 for my NEO-6M

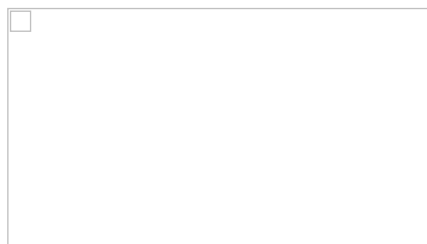
Then starts GPSD,

```
sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd.sock
```

And then display it,

```
cgps -s
```

You should see an output like this:



To allow the GPS to fix and give you a location, go outside or find a place with a clear view of the sky.

Finally. kill gpsd, and let's start programming.

### Using python to read/write a uBlox 6 GPS

Install two python libraries, [pyserial](#) and [pynmea](#) (I suggest you use pip, it is just easier to use that)

Once done, create a new \*.py file. anywhere you want. Make sure you don't create the file as root.

Start the X GUI (startx) and open up the python file in IDLE. And type this in:

```
import serial
from pynmea import nmea

ser = serial.Serial('/dev/ttyAMA0', 9600)
gpqga = nmea.GPQGA()
while 1:
    data = ser.readline()
    if (data.startswith('$GPQGA')):
        gpqga.parse(data)
        print 'Lat: ', gpqga.latitude
```

```
print 'Long: ', gpgga.longitude
print 'Alt: ', gpgga.antenna_altitude, ' ', gpgga.altitude_units
print 'No of sats: ', gpgga.num_sats
```

What the above code starts a serial port, keeps on reading data until sigint and parses the NMEA strings into Latitude, Longitude, Altitude and # of satellites.

What if you want to initialize the GPS in Airborne mode?

Create a new \*.py file and open it. Type in:

```
import serial

ser = serial.Serial('/dev/ttyAMA0',9600)
print 'Sending set Nav airborne < 1g'
command = b'\xB5\x62\x06\x24\x24\x00\xFF\xFF\x06\x03\x00\x00\x00\x00\x10\x27\x00\x00\x05\x00\xFA\x00\xFA\x00\x64'
ser.write(command)
```

The weird hex string tells the GPS to initialize in airborne mode. Here is a list of some commands:

**Revert to default config:**

B5 62 06 09 0D 00 FF FF 00 00 00 00 FF FF 00 00 03 1B 9A

**Enable all debug information:**

B5 62 06 02 0A 00 03 00 00 00 FF FF FF FF 00 10 F0

**Automotive Mode:**

B5 62 06 1A 28 00 03 00 00 00 03 04 10 02 50 C3 00 00 18 14 05 3C 00 03 00 00 FA 00 FA 00 64 00 2C 01 00 00 00 00 00 00 00 00 00 00 00 00 00 6C 95

All you have to do is, replace each space with a '\x' and add a '\x' in the front and you are good to go. These are essentially binary messages and are according to the [UBX protocol](#).

If you want to adjust more settings, download the [u-center software](#). Go to View>Messages View

Expand the UBX node and the CFG node. There you have it. Click on any of the child nodes, adjust the settings to want and copy the hex message in the textbox and send it over to uBlox using python.

## Conclusion

I hope this helped someone, if I missed anything please let me know or if you have any questions feel free to ask me. Hope I saved you a few google searches. And as always,

Happy Coding!

Posted by Shreyas at 10:47

 Recommend this on Google

Labels: [GPS](#), [tutorial](#), [uBlox](#)

## 1 comment:



**Leonardo Junior** 26 May 2016 at 15:45

Great! I searched all over the internet! Thank you so much!

[Reply](#)

Enter your comment...

Comment as: **Bramantio Yuw** ▼

[Sign out](#)

[Publish](#)

[Preview](#)

☐ [Notify me](#)

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

