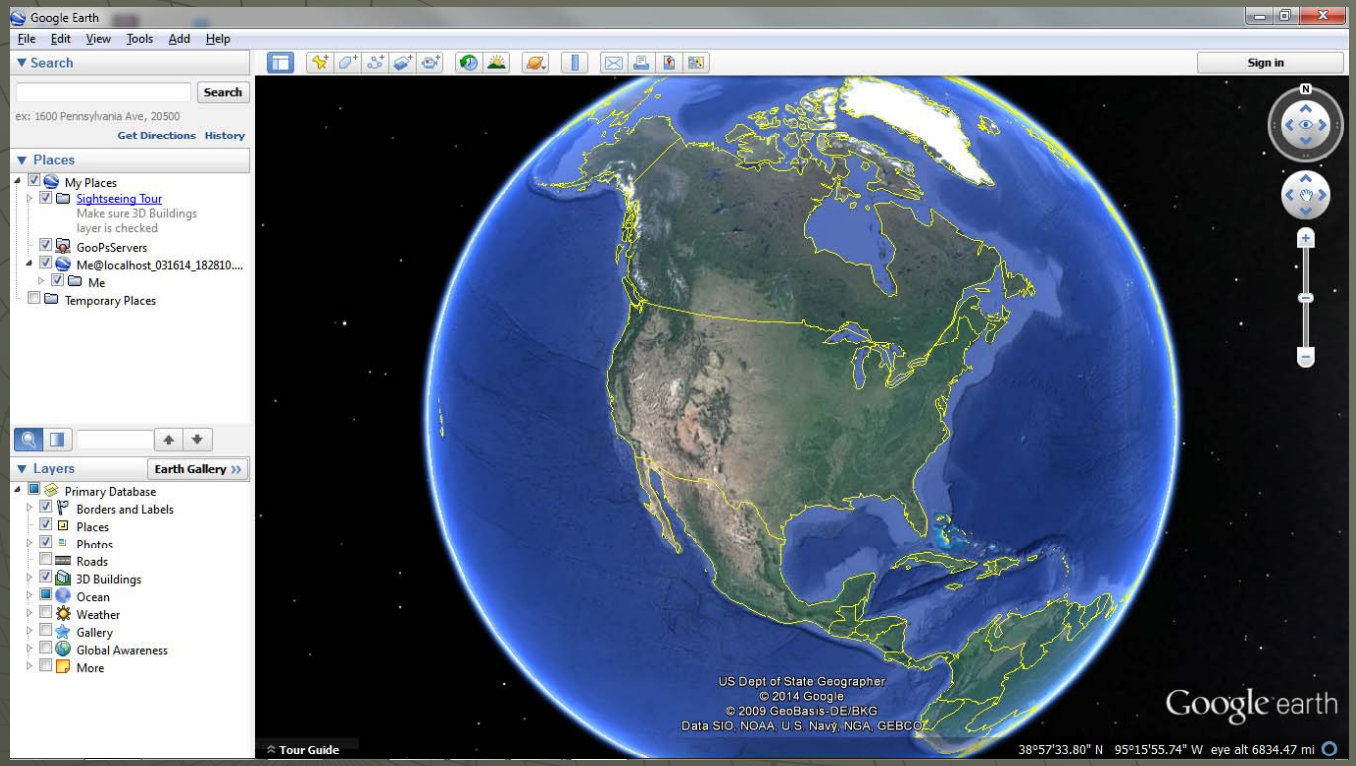
AIAA OC Rocketry

AIAA OC Section – NAR #718

GPS to Google Earth



In this tutorial, you will use an Arduino Uno, GPS Receiver, and an XBee RF downlink to track your rocket's location and altitude in real time on you PC.

Hardware Required (Part I)

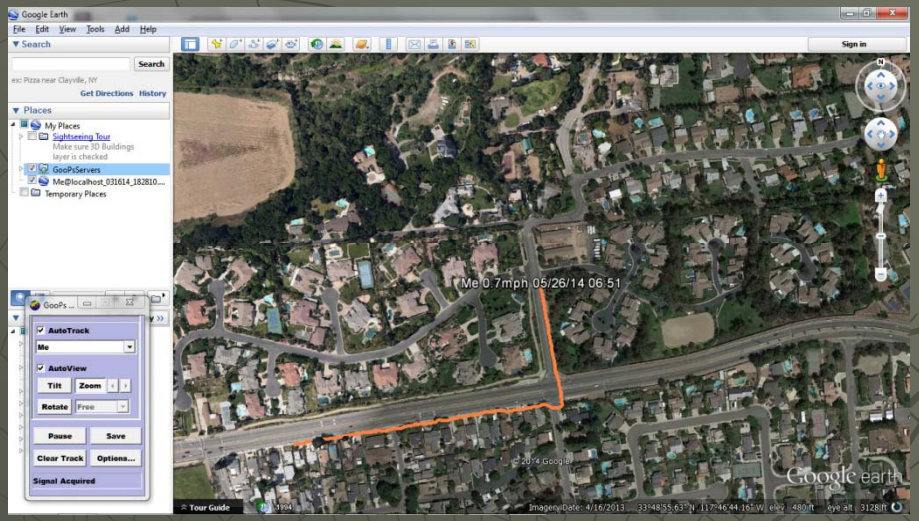
- GPS Receiver such as the EM-506 receiver from USGlobalSatellite: From Sparkfun and many more

- Purchase: https://www.sparkfun.com/products/12751
- Datasheet: http://cdn.sparkfun.com/datasheets/GPS/EM506_um.pdf
- Cable Purchase (6 pin JST for GPS to bare wire): https://www.sparkfun.com/products/10361

-Arduino Uno: From Arduino, Amazon, Sparkfun, MP3Cars, many more:

- Purchase: http://www.amazon.com/Arduino-UNO-board-DIP-ATmega328P/dp/B006H06TVG
- Schematic: http://arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf

- A PC or laptop



Software Required

Arduino Integrated Development Environment (IDE):
http://arduino.cc/en/main/software#.Uy4WgU1OUpA

Google Earth
http://www.google.com/earth/explore/products/desktop.html

GooPs Real Time Tracking for Google Earth (Free Version)
http://goopstechnologies.com/

Arduino Passthrough Sketch
http://aiaaocrocketry.org/AIAAOCRocketryDocs/SPARC2014/Sketches/GPSPass_Through.zip

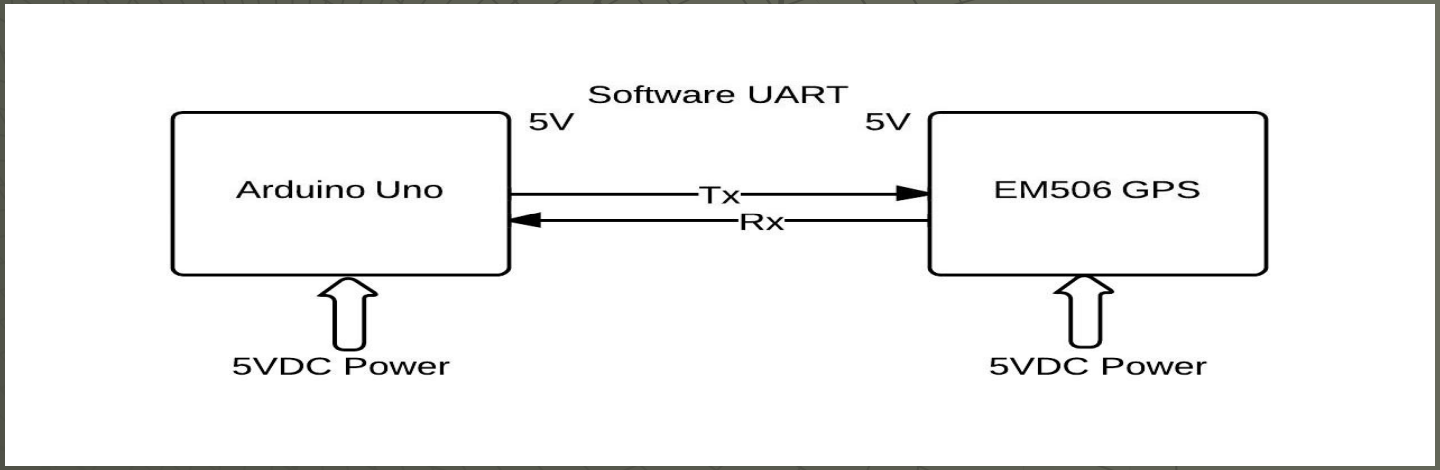
Arduino TinyGPS Library with examples:
https://github.com/mikalhart/TinyGPS (select “Download .zip” on the right below the list of files)

- Install the .zip file as the TinyGPS Library
- Test_with_gps_device is in the examples folder
- Note that this also uses the SoftwareSerial library that is included in the Arduino IDE V1.0 & later

Connecting the EM506GPS to the Arduino Uno

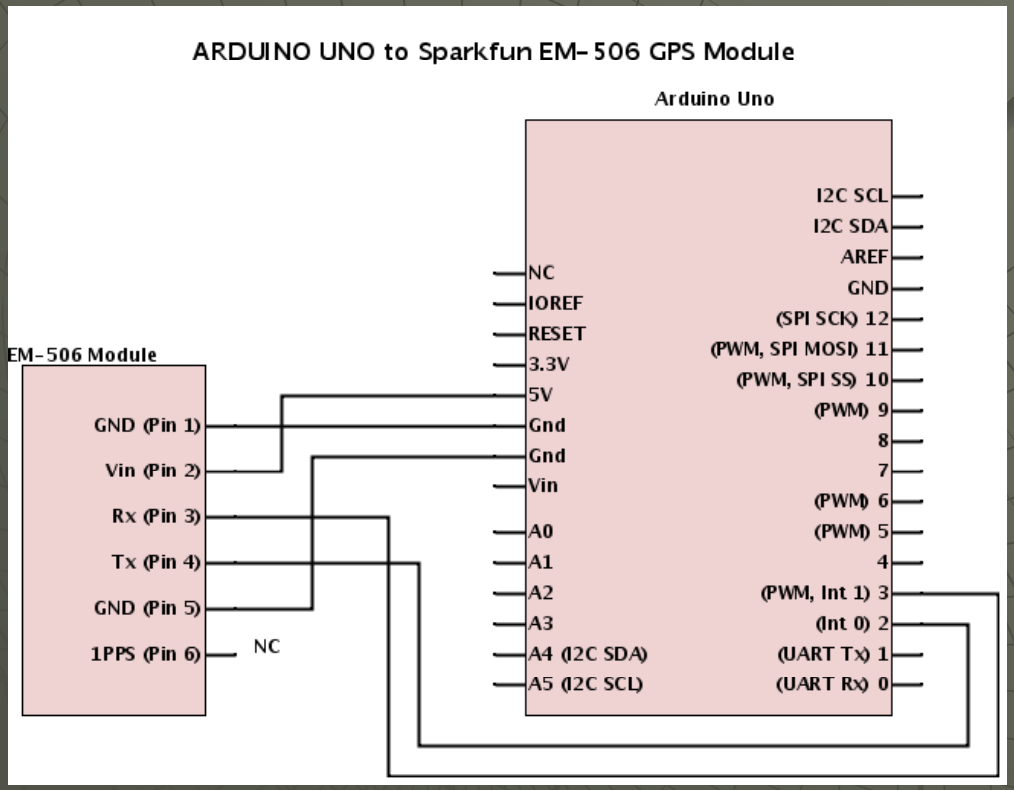
Block Diagram

First step is to connect the EM506 module to an Arduino Uno, then connect the Arduino Uno to a PC using the USB cable. The Arduino Uno will be in pass-through mode (it accepts a character from one device and passes it on to another device unmodified). The results should be identical to those received on the commercial module connected directly to the PC



Connecting the EM506GPS to the Arduino Uno

Schematic



EM-506 Pinout

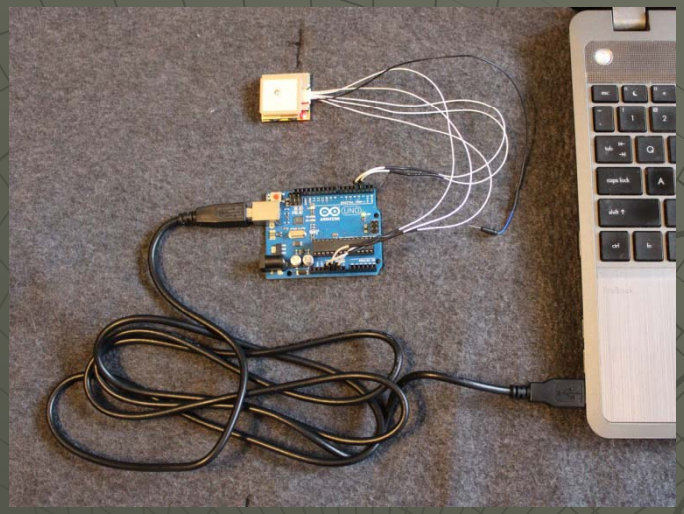
EM-506 Datasheet: http://www.usglobalsat.com/store/download/717/EM506_um.pdf
Both the Arduino Uno and the EM-506 run at 5VDC to no level shifting is required

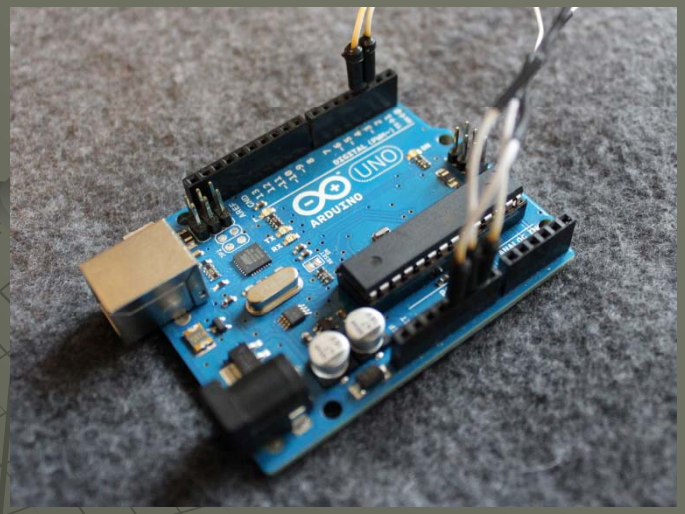
Connecting the EM506GPS to the PC

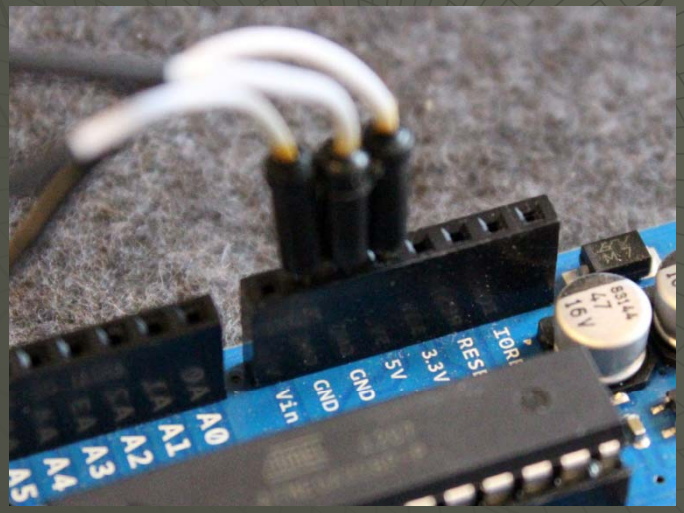
It may make prototyping easier for the GPS if you modify the GPS cable by cutting the jumpers in half and soldering them to the ends of the cable

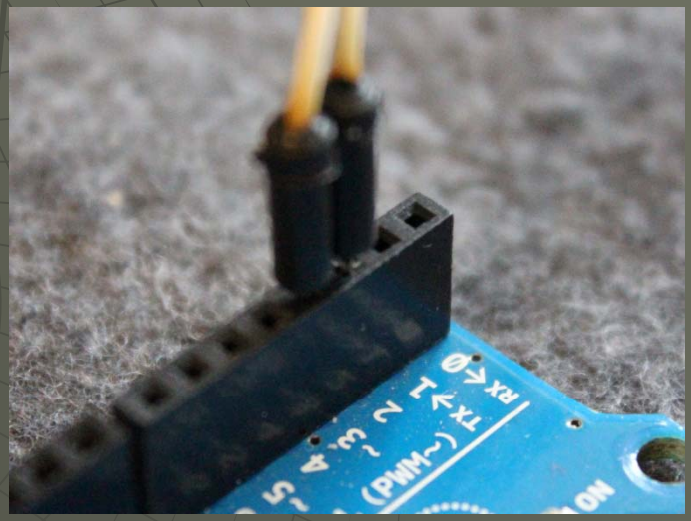
6 pin JST for GPS to bare wire: https://www.sparkfun.com/products/10361
Male to Male jumpers: https://www.sparkfun.com/products/12795
Note: There are many options for jumper wires

Connecting the EM506GPS to the PC





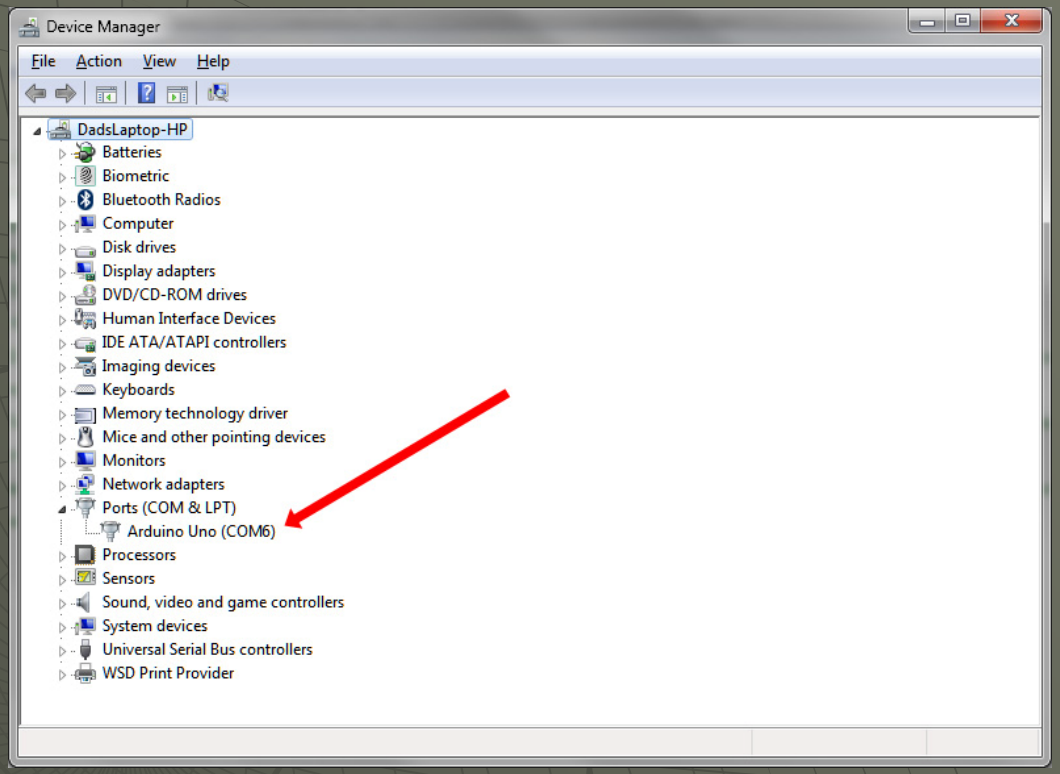




Download and run the GPS_Passthrough sketch from the AIAA OC Rocketry Website:
http://aiaaocrocketry.org/AIAAOCRocketryDocs/SPARC2014/Sketches/GPSPass_Through.zip
Unzip the folder with the .ino file into the Sketches folder under Arduino in (My)Documents.
This program will just pass the data from the GPS receiver to the PC via the USB cable. It uses a software UART on pins 2 and 3 on the Arduino to talk to the GPS receiver and the hardware UART to talk to the PC via USB

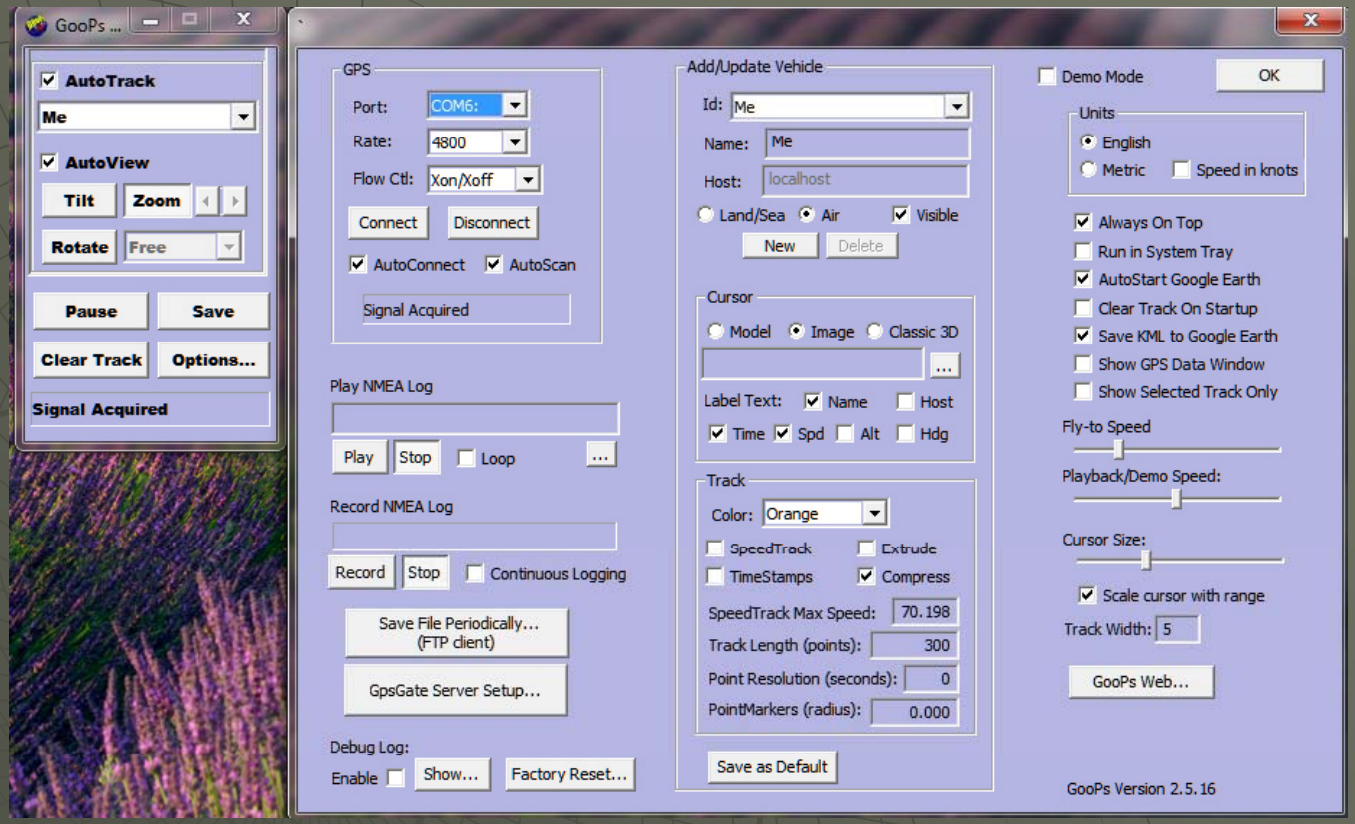
```
#include <SoftwareSerial.h>

// GPS Baud Rate is 4800
#define GPSBAUD 4800
#define RXPIN 2
#define TXPIN 3

SoftwareSerial uart_gps(RXPIN, TXPIN);
void setup()
{
  uart_gps.begin(GPSBAUD);
  Serial.begin(4800);
}
void loop()
{
  byte a;
    if ( uart_gps.available() > 0 )
  {
    a = uart_gps.read(); // get the byte of data from the GPS
    Serial.write(a);
  }
}
```

Determining GPS Virtual COMM Port



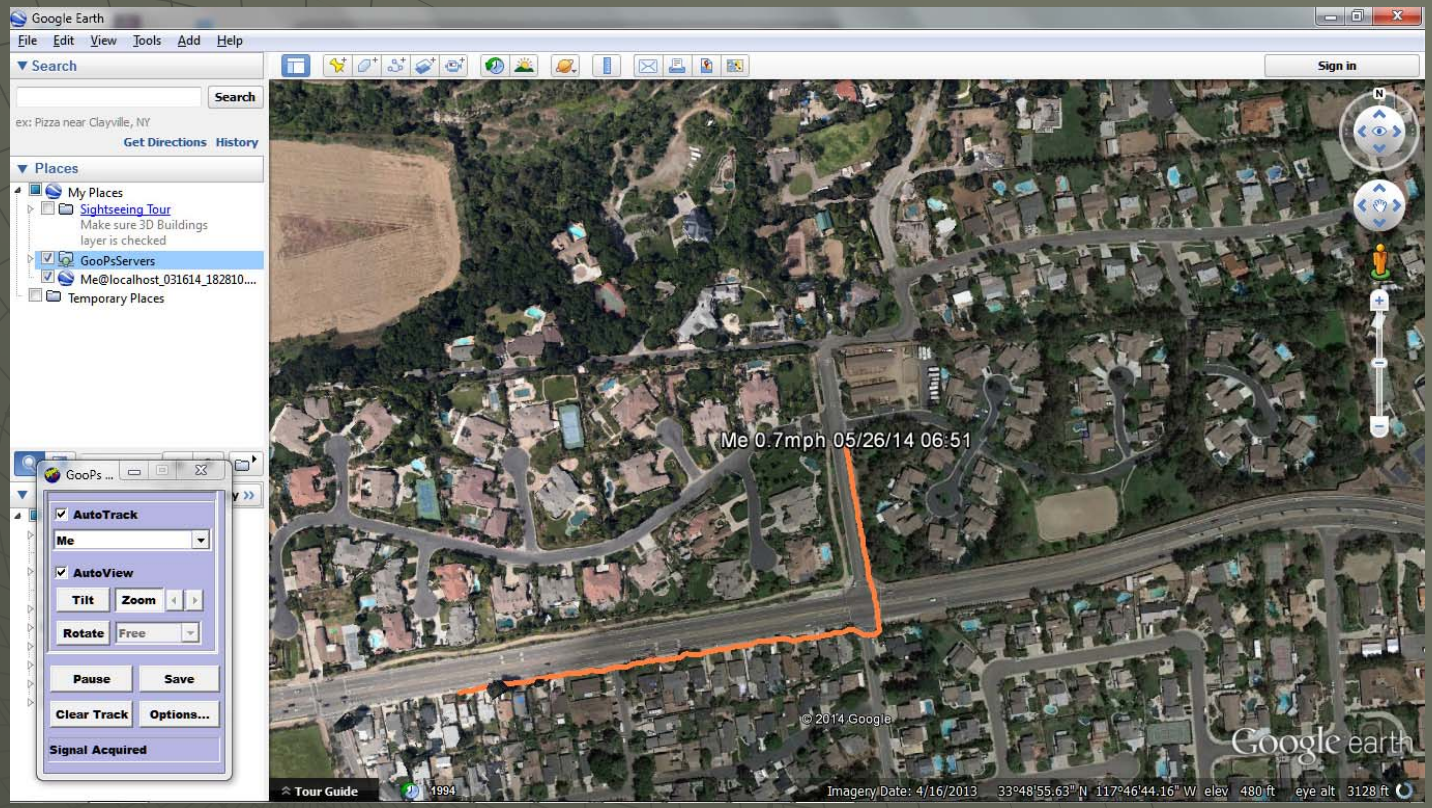
You will need to specify the COM port used for the GooPs program. The Arduino IDE used that same COM port when downloading your sketch. Or you can go to Start->Control Panel->Device Manager, then click on Ports

Set up the GooPs program



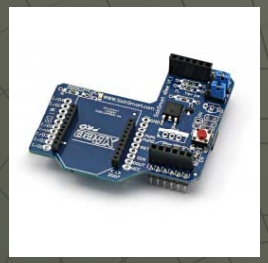
Install and start the GooPs program. Click on Options and set the COM port, Baud rate should be 4800, select autoconnect and autoscan (if you question the COM port), and turn OFF demo mode. Select AutoStart Google Earth since you'll always want that running as well

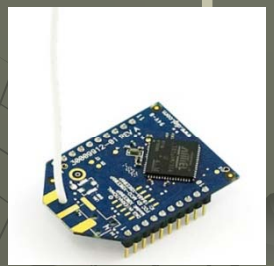
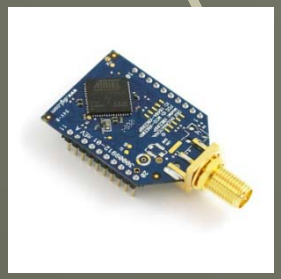
Google Earth – Via USB Cable



Google Earth should now be reporting your location. And as you and your PC walk, Google Earth should show your track. Your speed of travel is reported at your current location, and that latitude and longitude is reported in the lower right of the screen along with your altitude. You will need to maintain internet access via a mobile hot spot to update the map

Hardware Required (Part II)




-XBee Shield for Arduino Uno

- Purchase: http://www.sainsmart.com/sainsmart-xbee-shield-module-for-zigbee-arduino-uno-duemilanove-mega-1280-2560.html
- Schematic: http://www.sainsmart.com/zen/documents/20-011-902/Libelium-Xbee-Shield.pdf

-XBee USB Explorer Dongle

- Purchase: https://www.sparkfun.com/products/9819
- Schematic: http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Wireless/Zigbee/XBee-Explorer-DongleV12.pdf

-XBee Pro 60mW Wire Antenna Series 1

- 900 MHz Purchase: https://www.sparkfun.com/products/9097
- 2.4GHz Purchase: https://www.sparkfun.com/products/8742

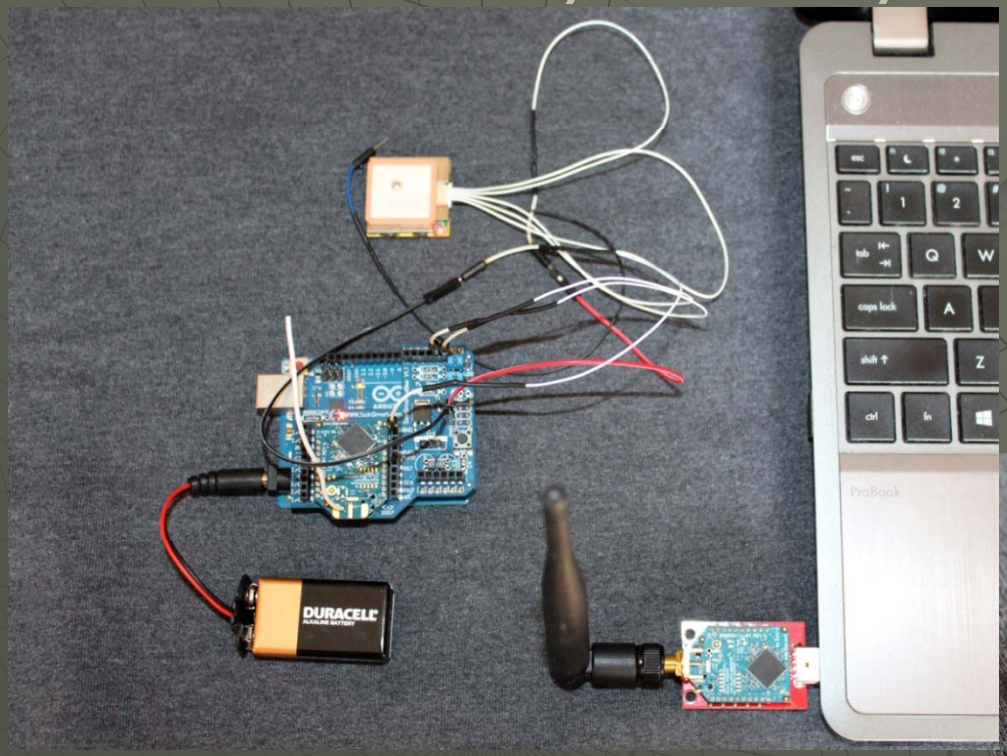
-XBee Pro 60mW RPSMA connector

- 900MHz Purchase: https://www.sparkfun.com/products/9099
- 2.4GHz Purchase (U.FL Connector): https://www.sparkfun.com/products/8710

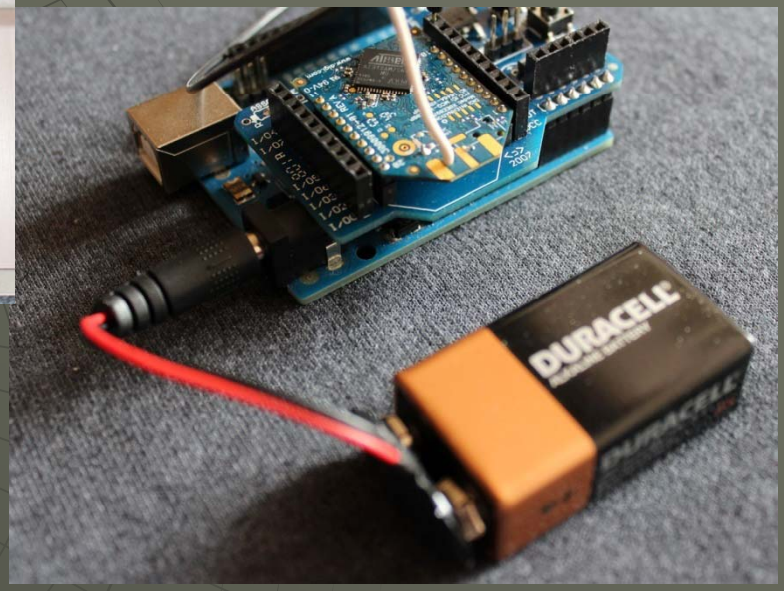
-Rubber Duck Antenna

- Purchase: https://www.sparkfun.com/products/9143
- Purchase (Cable U.FL to RPSMA connector): https://www.sparkfun.com/products/662

Uno, GPS, and XBee

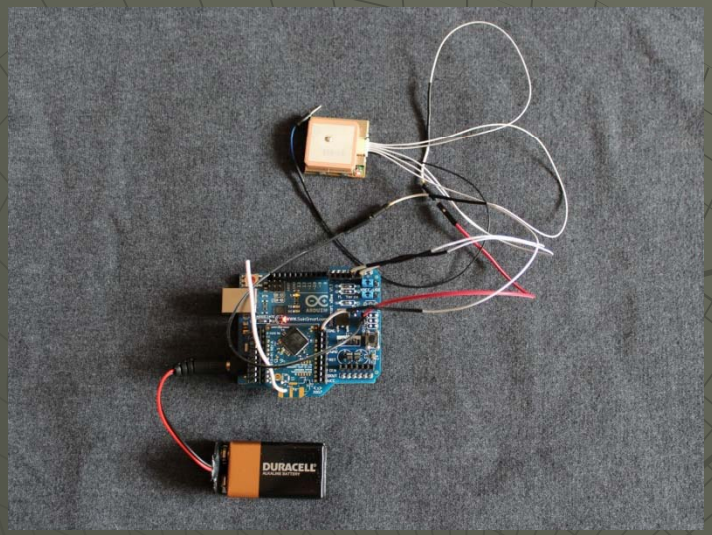


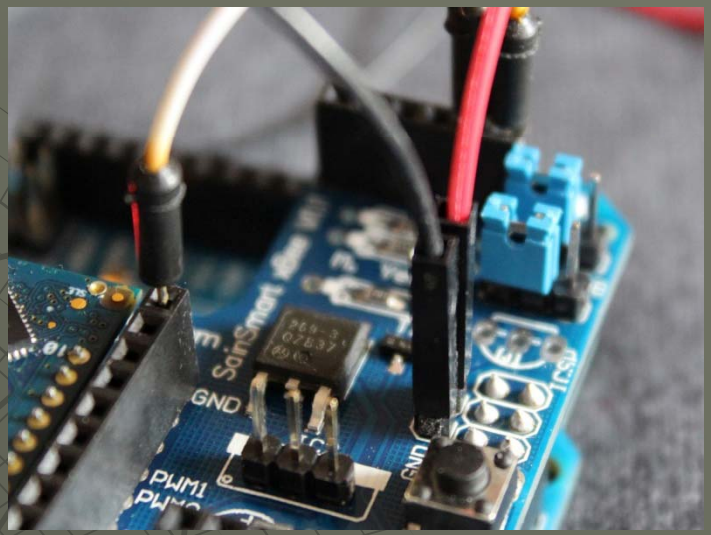
Remove your wiring directly to the Uno and plug in the XBee shield with Xbee (power should be off). Then replace re-attach the wires to the XBee Shield

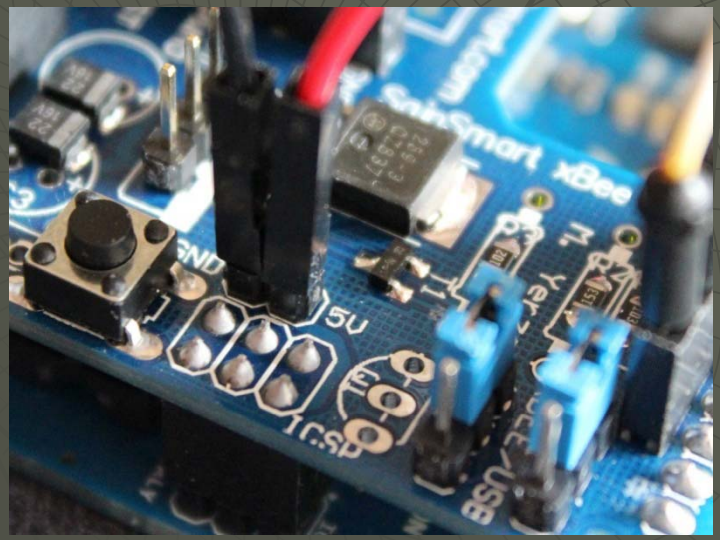


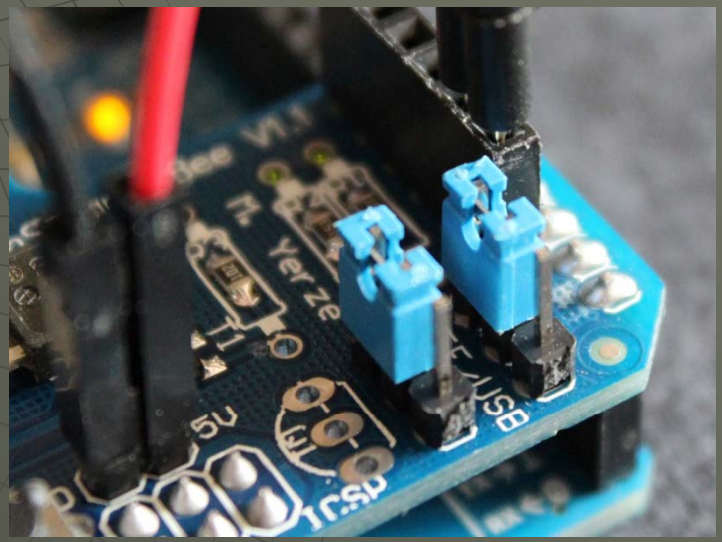
It will be easier to wire to the shield if you solder header strips onto the shield on either side of the XBee

Uno, GPS, and XBee





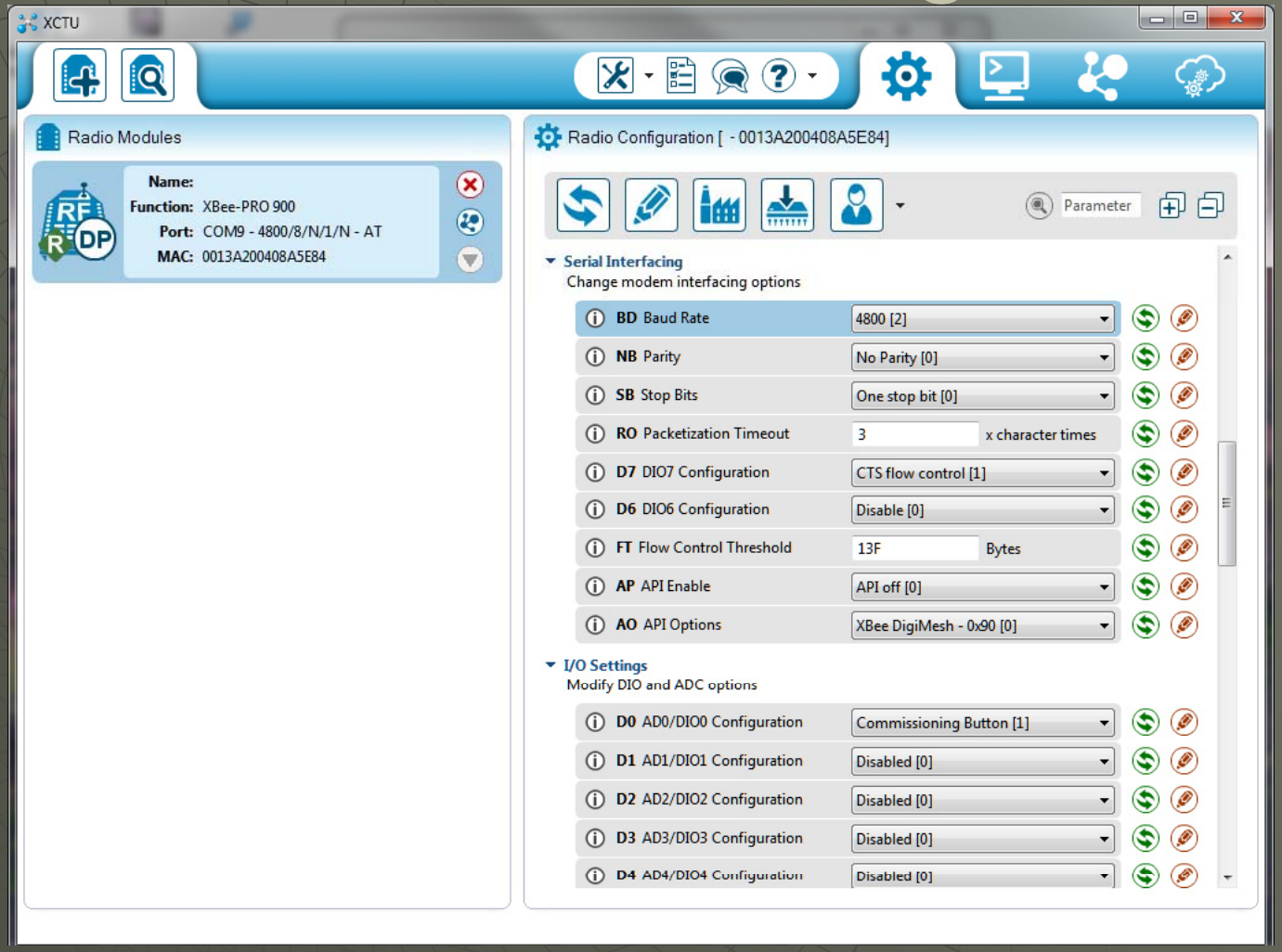




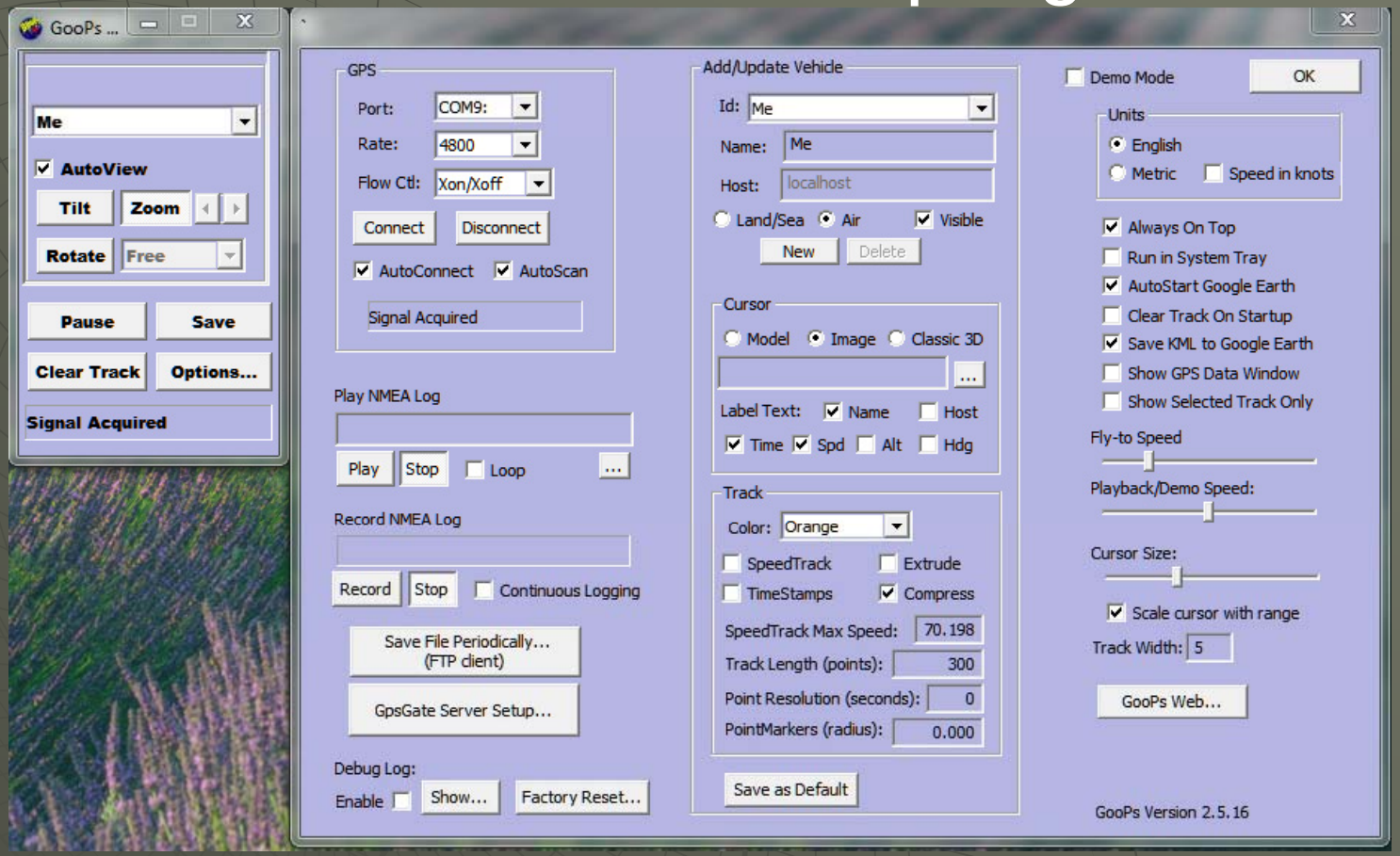
XBee Series 1 Setup

1 – Download the XCTU Next Generation Software and install
http://www.digi.com/support/productdetail?pid=3352&osvid=57&type=utilities
2 – Plug one of the XBee modules into the USB Explorer dongle and plug that into the PC
3 - Set up the XBee modules for 4800 Baud
    A – Start the XCTU Software
    B – Click on the search Icon to find your module (you will need to specify the COM port and BAUD rate
    C – Click on “Add Selected Device”
    D – After the module is found click on the module name (the configuration tab should already be selected)
    E – Set the module BAUD rate to 4800 Baud
    F – Click on the write icon to write to the module
    G – Verify that the BAUD rate changed in the description block that originally appeared when you discovered the module
4 - Repeat for the second module

XBee XCTU Configuration

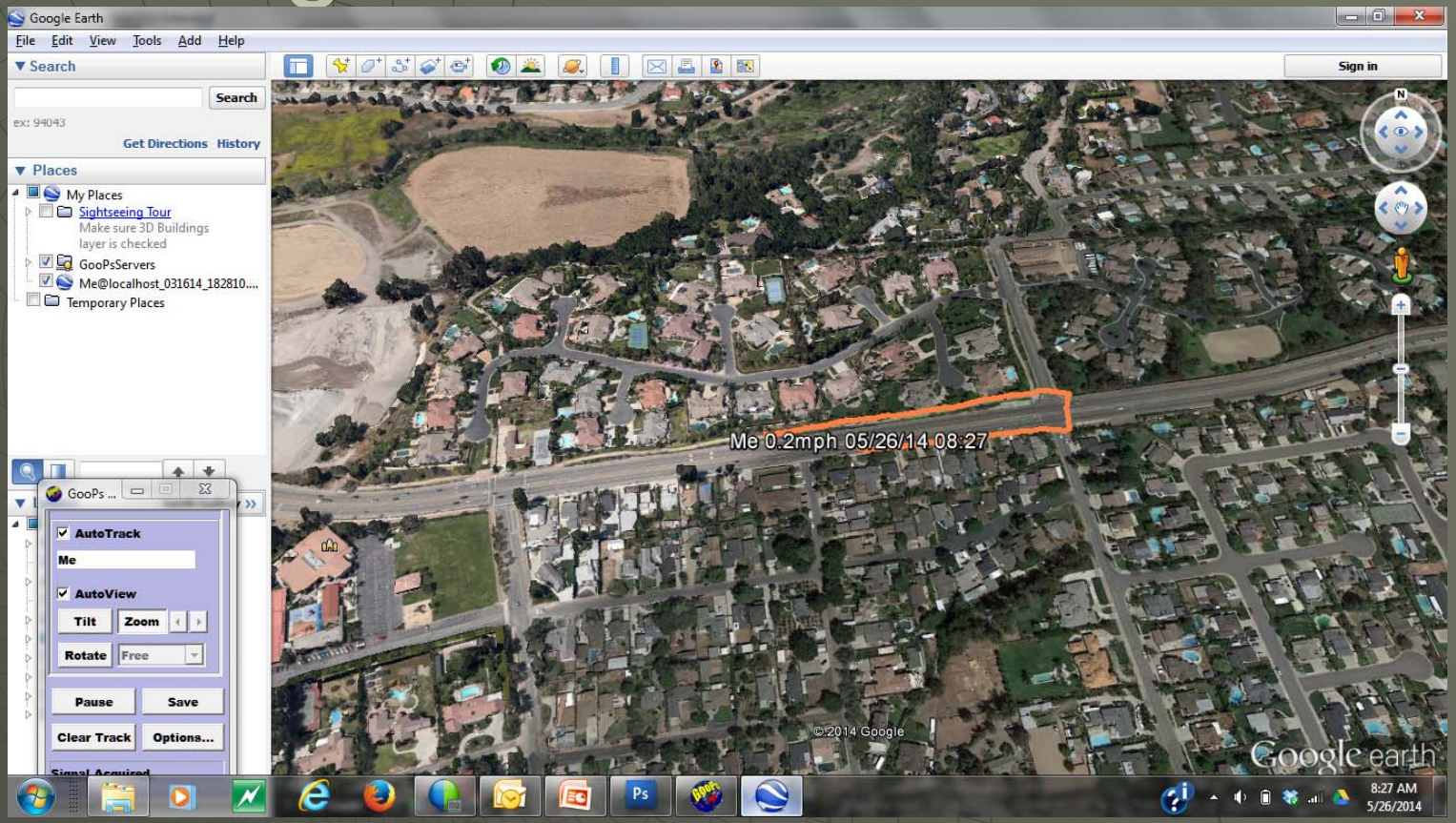


Restart the GooPs program



After you restart the GooPs program, you may need to reset the Port. If you have Autoscan selected, GooPs should automatically scan and find your XBee if you have your Arduino ON and sending data via XBee.

Google Earth – Via XBee RF



Once again, Google Earth should now be reporting your location. But this time you can leave your PC in one location and walk with the Arduino connected to the GPS and powered by a battery. You will see your track on the Google Earth map. If you were flying a rocket instead of walking, you can see your altitude on the 3D track