

## GPS Overview

SparkFun Electronics Summer Semester

### Introduction to GPS

The goal of this handout is to show you how to use GPS NMEA data and demonstrate some of the features and hardware one finds in embedded GPS modules/receivers. In addition, I will show how the features and hardware can affect tracking accuracy. This is not a buying guide per se, but it will help you get an idea of the differences, shown in Google Earth, between chipsets, antennas, and update rates for the six SparkFun GPS modules seen below.

SUP500F



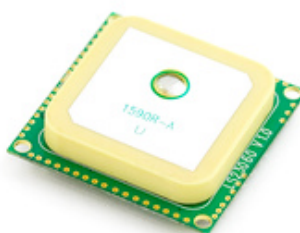
EM406



Copernicus



LS23060



uMini



D2523



I choose modules with different chipsets, antenna/hardware configurations, and update rates in order to get a general idea of the wide variety of features. All modules were left in their default states (except for the SUP500F which I had to configure for 10Hz update rate). The Copernicus and uMini use SMDGPS modules and are assembled by SparkFun. The other four are from outside suppliers.

## GPS Overview

### SparkFun Electronics Summer Semester

GPS Module	Chipset	Antenna	Update Rate
Copernicus	Trimble Trim Core	ceramic patch	1Hz
SUP500F	Venus634	small ceramic patch	10Hz
MN5010 (uMini)	SiRF Star III	chip	1Hz
LS23060	MediaTek MT3318	ceramic patch	5Hz
DS2523T	U-Blox 5	helical	4Hz
EM-406A	SiRF Star III	ceramic patch	1Hz

Refer to individual product pages and datasheets for the detailed technical information for each module.

## GPS Overview

### SparkFun Electronics Summer Semester

## Building the tester

Here is the bill of materials (BOM) for testing each GPS module:

- Breadboard Power Supply Stick 3.3/5V
- LiPo Battery Pack 1000mAh/7.2V
- openlog
- uSD 1GB
- jumper wires and headers

The following are some issues that I had to consider when designing the test rig.

### GPS modules are radio frequency receivers.

For reliable performance, your GPS module needs to receive good unobstructed signals from the satellite constellation. GPS modules have relatively tiny antennas to receive data from satellites moving at relativistic orbital speeds, over 10,000 miles away, and so they need somewhat of a clear view of the sky. I know this might sound really obvious, but I cannot stress this enough, because...

### GPS sensitivity and accuracy is highly dependent on environmental conditions.

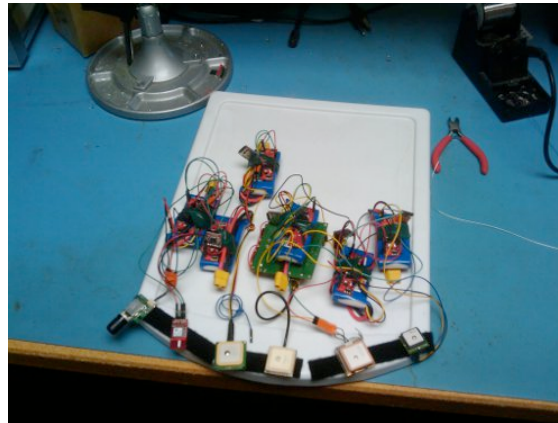
Not only does this include physical objects, like trees, earth, and structures, but also includes atmospheric effects, space weather, the overall health of the GPS constellation, and multipath interactions, all of which contribute to constant variability in receiving the GPS signal. To minimize errors due to these environmental effects, I wanted to test and track all six modules at the same time with varying views of the sky.

What I needed was a mobile logger for each GPS module. SparkFun stocks more than enough supplies to accomplish this.

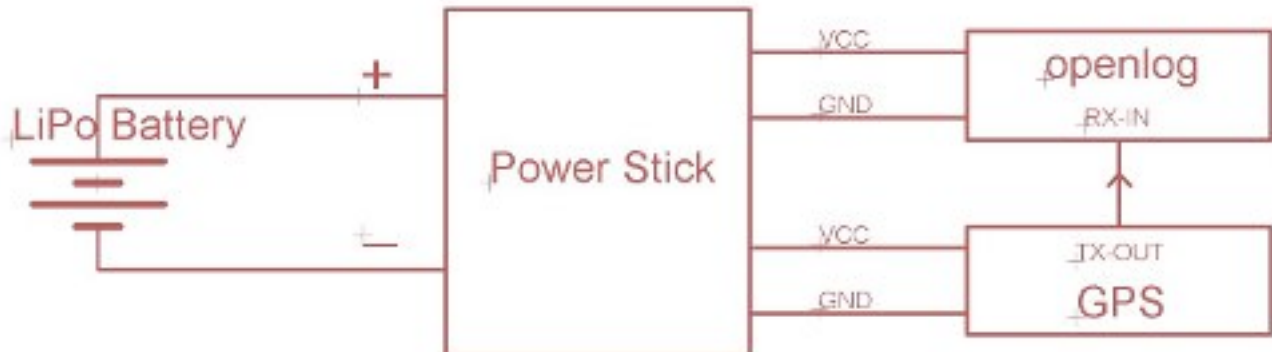
SparkFun Electronics now carries a GPS shield unit that will make hooking up your GPS module to your Arduino as easy as soldering some headers and placing the shield onto your Arduino.

## GPS Overview

### SparkFun Electronics Summer Semester



Tracking Setup. Note: The uMini was oriented vertically for the test and is not shown that way in the picture



Connection Block Diagram

**Each GPS module should have its own clean (regulated and decoupled) DC power supply.**

The GPS power supply feeds into the antenna power supply on some of the modules, so any excessive noise or unexpected loading on the circuit can affect performance. Attaching a high capacity LiPo battery to a regulated Breadboard Power Supply Stick for the power supply for each module is sufficient.

## GPS Overview

### SparkFun Electronics Summer Semester

**Next, I capture the data coming out of the GPS module with the openlog.**

All that needs to be done is to connect the transmit pin (TX) from the GPS module to the receive (RX) pin on the openlog. The openlog will log anything it sees on that pin, just be sure to configure the baud rate on the openlog to match the baud rate coming out of the GPS module.

**A brief word or two on the VBATT pin and the TTFF (time to first fix).**

What is the VBATT pin used for? Some embedded GPS modules contain volatile RAM that is used to store the almanac and ephemeris data of the satellites, which is globally updated on a regular basis. Think of the almanac data as a "bus schedule" of when and where each satellite will be in the sky. The ephemeris data contains the errors in the schedule. If your GPS receiver does not have the schedule or the errors of the satellites, it will have to download that information when there are updates. Since the RAM stores this data, the VBATT pin should be powered or connected to some kind of energy supply (i.e. supply voltage, battery, or supercapacitor), so that your GPS module can quickly start-up. If you don't power the RAM, all of the almanac and ephemeris data is lost with each power cycle.

The TTFF (time to first fix) is determined by how quickly the GPS module can access the almanac and ephemeris data. If the module doesn't have any of this data (some modules don't have this data out-of-the-box) the module might take a bit of time to power up. Same goes for if you don't connect power to the VBATT (RAM power) pin, you will lose all of the data upon a power cycle. If the almanac and ephemeris data can be permanently stored in flash, like the SUP500F can, then you should get a really fast TTFF. Some modules, like the EM406, have a supercapacitor attached to the RAM. This means you will get really quick TTFF for only a period of time after you power the module down. Wait too long, the capacitor will discharge and you will lose your information. The reason for the choice in using a supercap over a battery or flash chip, is that it is expensive, small, and the almanac and ephemeris data needs to be updated regularly anyway, so if your position is changing a great deal over the globe, you don't need to keep all of the same almanac data.

In terms of the TTFF performance for each of these modules, all have similar stated fix times that are under 1 minute, but will differ with environmental conditions and antenna hardware.

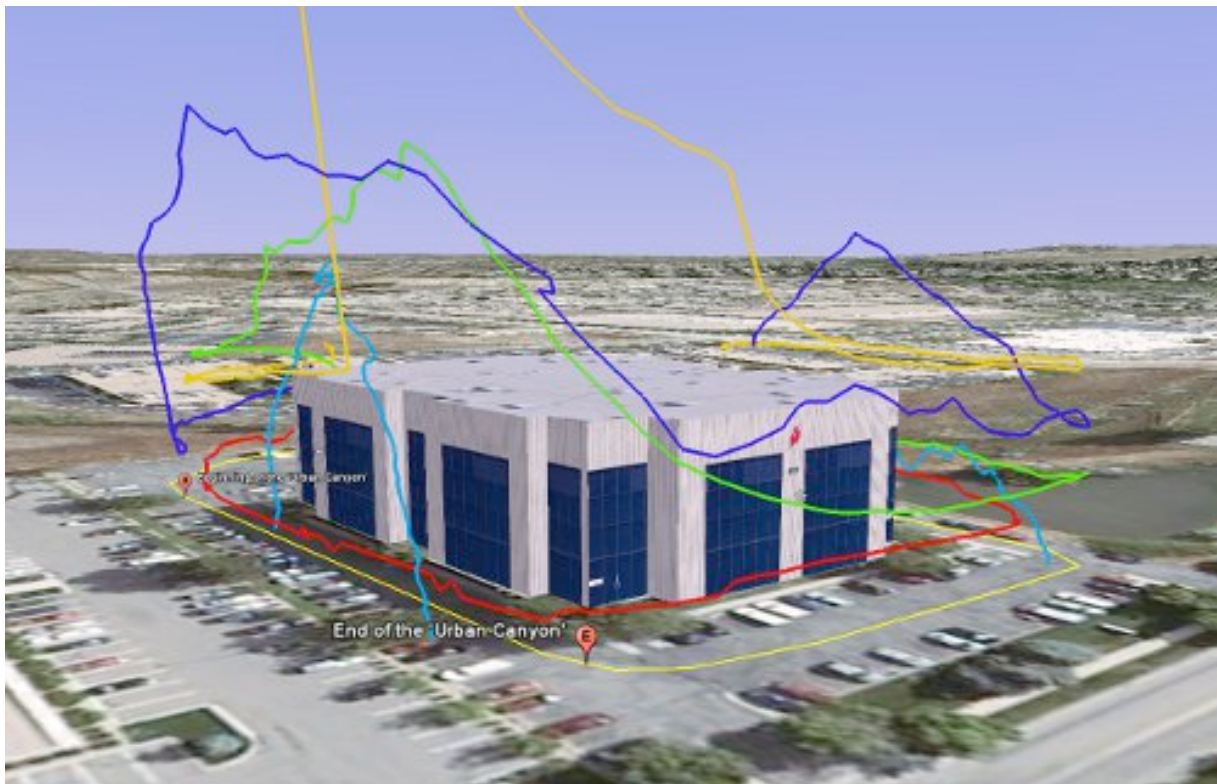
## GPS Overview

SparkFun Electronics Summer Semester

### Tracking

Download Google Earth to view the KMZ files and zoom around my track in 3D (be sure to turn on 3D buildings). Just click the link and Google Earth should open and zoom to the SparkFun building.

- raw KMZ
- edited for locks KMZ



Overview

For my track, I decided to walk the GPS modules around the SparkFun building, instead of testing them in a car. **GPS modules, to an extent, seem to work better the faster they move**, so walking the units around the building should create a test condition where the GPS modules will have a greater chance to show errors.





## GPS Overview

### SparkFun Electronics Summer Semester

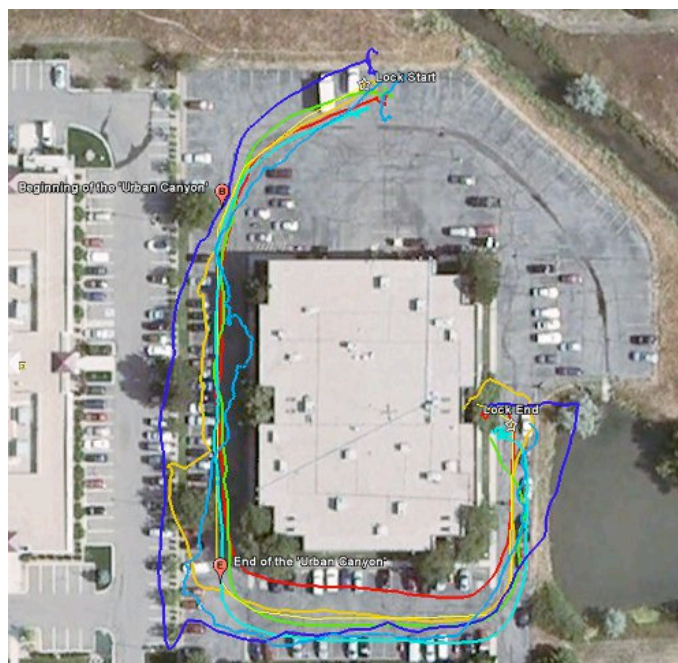
What does it mean when the receiver has a lock or a fix?

**When the receiver has a lock, you get accurate position (3D) and time data.** This is accomplished when the receiver sees at least four satellites in good view. Why four? You would think you only need three for triangulation, right? You forgot about time! Remember, with four unknown variables you need four equations. The fourth signal gives enough information for your GPS unit to compute all three positions (latitude, longitude, elevation) and time.

Getting accurate time is not an easy task for satellites and receivers alike. The GPS satellites need really, I mean really, accurate timing. So accurate, that there are actually atomic clocks on board the satellites in order to keep them in sync. Since the speeds and distances are so great between satellites and receivers, relativistic offsets are programmed into the satellites position and time data. This information, along with additional correction data the satellites receive from permanent ground based stations (this is how WAAS works), turns your GPS receiver into an amazingly accurate and powerful little device with, on average, five meter accuracy.

RED = Copernicus, BLUE = uMini, GREEN = EM406, ORANGE = D2523, LIGHT BLUE = SUP500, GREEN-BLUE = LS23060, YELLOW = actual path

\*North is at the top\*





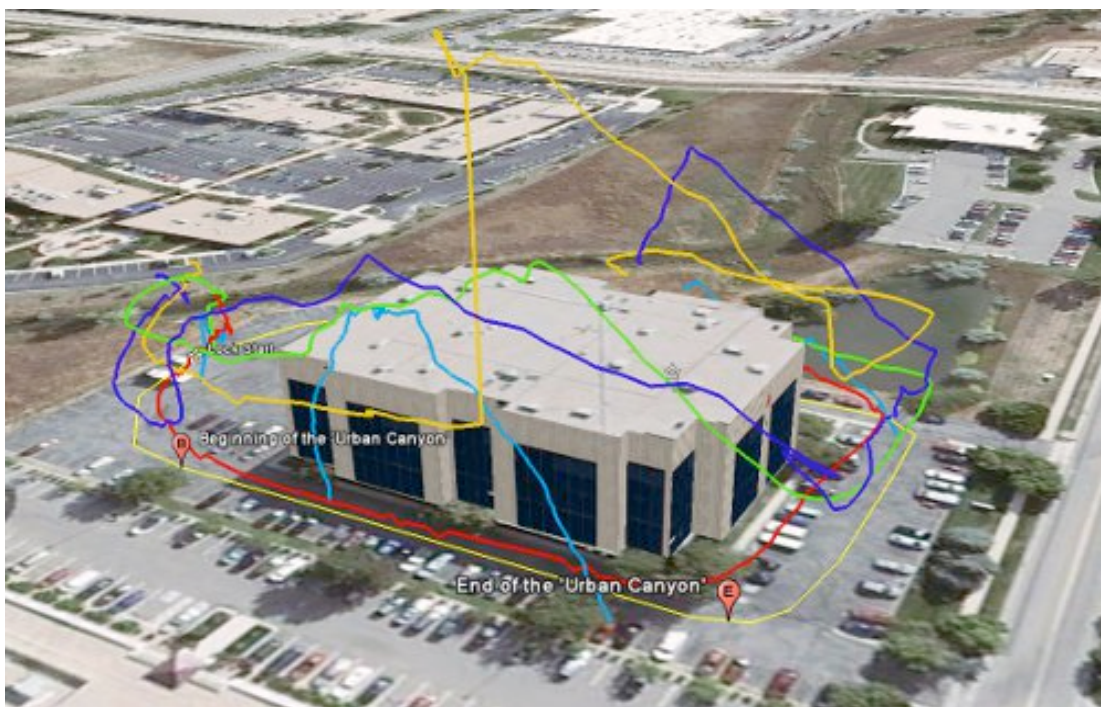
## GPS Overview

### SparkFun Electronics Summer Semester

This view only shows latitude and longitude data (no elevation) and I only am showing the tracks *after* I waited in clear view of the sky to make sure all of the modules had a lock. Notice that all of the tracks in clear view of the sky (to the north and south) are decently accurate; within the claimed 1-5 meter accuracy for C/A code receivers.

There is about a 150 foot span in between the ~25 foot tall SparkFun building and our neighbors ~25 foot tall building to the west that creates a small 'urban canyon' where most of the modules had some expected trouble. In addition, there is tree cover in part of the SparkFun urban canyon, which also hindered the accuracy of the GPS modules.

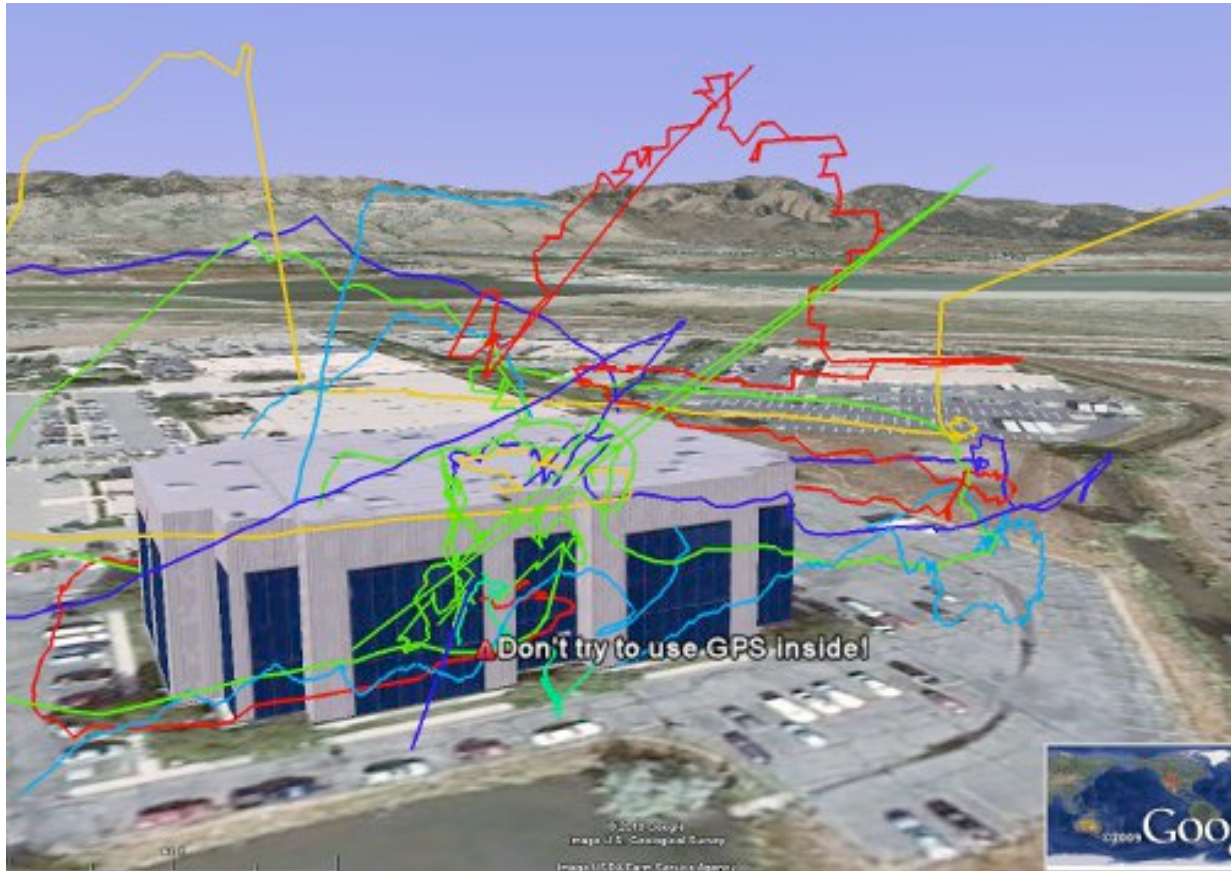
Notice how the SUP500F's (light blue) track is not a straight as the other tracks, this is because the position is being updated at 10 times a second compared to one time a second for most of the other receivers. Modules with faster update rates usually work better within faster moving objects.



This is a 3D view looking at the west side of the building. Looks like the D2523 (orange) had a hiccup in the urban canyon. Also notice the uMini (blue) is not performing as well as the others. This is expected since the uMini is using a passive chip antenna. And wait...I only see five tracks, where is the sixth? The LS23060 module's elevation data is below what Google Earth calls ground height. It is there, just not visible in this view.

## GPS Overview

SparkFun Electronics Summer Semester



Don't try to use GPS inside!

Wow, what is that? Those are the tracks of the GPS modules trying to find their location inside of the building when I first turned the units on. **This is why you don't rely on GPS inside of a structure.**

## GPS Overview

### SparkFun Electronics Summer Semester

#### How to create files in Google Earth

Overlaying your tracks in Google Earth is really easy. After you have logged your positions to the openlog, remove the SD card and plug it into a SD card reader on your computer. Look at the contents on the card. You should see a config file and at least one log file (if you cycled power during logging, new log files would have been created). Open the log file with a text editor, like Notepad in Windows. You can see a bunch of text in the form of NMEA sentences or something in a similar format. The NMEA sentences contain, among other data, your position and time.

Here are my raw text files: [http://www.sparkfun.com/tutorial/GPSTracking/raw\\_5-20-10.zip](http://www.sparkfun.com/tutorial/GPSTracking/raw_5-20-10.zip)

All you need to do now is convert the file to a Google Earth file with a kml or kmz file extension. The website I use to convert the text file to a Google Earth file is GPSVisualizer. If you click on the Google Earth KML link in the middle of the homepage you can configure settings.



## GPS Overview

### SparkFun Electronics Summer Semester

**GPS Visualizer**

Convert your GPS data for use in Google Earth

This form will import your GPS data file (e.g., GPX, or plain-text data (tab-delimited or CSV), and create a KML file that you might also be interested in the [Google Maps](#) input form, which can create an interactive map that can be viewed in [Google Maps](#). To reuse and/or colorize Google Earth markers based on a particular field, use the [data form](#).

**General map parameters** [show advanced options \(+\)](#)

Output file type:  Units:  Add DEM elevation data: ☐ Time offset:  Add time stamps, if possible: ☐ No

**Track options** [show advanced options \(+\)](#)

Altitude mode:  Draw a shadow: ☐ Tickmark interval:  Draw as waypoints: ☐ Track opacity:  Line width:  Colorize by:  Default color:

**Waypoint options** [show advanced options \(+\)](#)

Show waypoints: ☐ All (bounds are defined by waypoints) Wpt. display padding:  Altitude mode:  Default icon color:  Default icon:  Waypoint labels:

**Contact information**

Your e-mail:  This is for prompts tech support, NOT a mailing list!

**Upload your GPS data files here:** (Total size of all files cannot exceed 3 MB)

File #1:  Browse File #2:  Browse File #3:  Browse

[Show additional file input boxes](#)

**Or paste your data here:**

Force plain text to be this type:

**Or provide the URL of data on the Web:**

**Create KML file** ☐ Open in new window

The settings I used are shown on the previous page. Once you have loaded your data into this page, hit the Create KML file. You will then be presented with a link. You can either click it and, if you have Google Earth installed, it will zoom to your tracks or you can right click and save as to save for future use or to show your friends.

If your txt files are over 3MB, you will not be able to use GPSVisualizer. GPSTabel is another site you can try if you have really big log files.

## GPS Overview

### SparkFun Electronics Summer Semester

## Tracking Summary

This is by no means an "end-all-be-all" summarization; it is merely a comparison on how well 6 very different GPS modules performed at the SparkFun location (not in any kind of order).

### Copernicus Pros

- Consistently the most accurate and reliable
- Excellent elevation data in all conditions
- Very good latitude and longitude data in all conditions
- Very good multipath jamming, not as susceptible to interference
- Least troublesome
- Optional antenna
- Excellent support and configuration software

### Copernicus Cons

- SMD, can be hard to solder
- Large, needs an external antenna
- More than the basic (VCC, GND, TX) connections are needed to get up and running

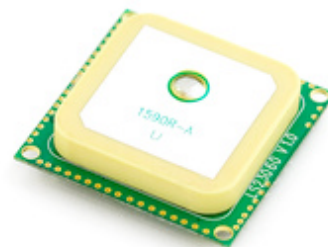


### LS23060 Pros

- One of the best along with the Copernicus
- Excellent latitude and longitude in all conditions
- Good elevation in all conditions
- Compact and self contained
- 5Hz update rate
- Fix indicator LED

### LS23060 Cons

- Solder pads for connection, have to solder on header pins or make your own cable









## GPS Overview

### SparkFun Electronics Summer Semester

<p><b>MN5010 (uMini) Pros</b></p> <ul style="list-style-type: none"> <li>• Very small</li> <li>• Very good accuracy for a chip antenna!</li> <li>• Inexpensive BOM</li> <li>• Simple configuration, minimal pin connections</li> </ul> <p><b>MN5010 (uMini) Cons</b></p> <ul style="list-style-type: none"> <li>• Chip antennas are not as accurate as the ceramic patch antennas</li> <li>• Greater susceptibility to multipath interference with chip antenna</li> </ul>	
<p><b>EM-406A Pros</b></p> <ul style="list-style-type: none"> <li>• Tried-Tested-and-True</li> <li>• Fix indicator LED</li> <li>• Excellent tracking data in all conditions</li> <li>• PCB is completely shielded</li> <li>• Plenty of online support</li> </ul> <p><b>EM-406A Cons</b></p> <ul style="list-style-type: none"> <li>• 5V operation</li> </ul>	

## Conclusion

We now have a pretty good idea of the general performance characteristics for some of SparkFun's GPS modules, however this is obviously not all-inclusive, in terms of the performance capabilities for other GPS modules with similar features out on the market today. GPS modules with combinations of the same chipset, hardware, antennas, and configurations all will have different performance characteristics in different environmental situations. All have their benefits and costs, but it will be up to you to take the data points I have found to help choose the correct GPS module for your project.