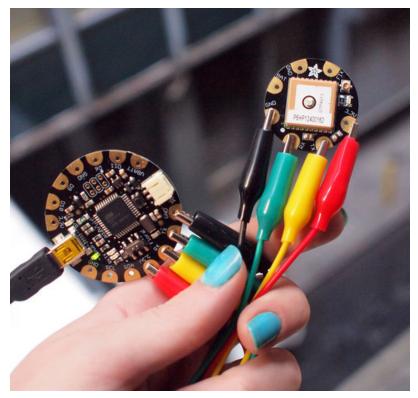


# Flora Wearable GPS

Created by Becky Stern



Last updated on 2019-11-12 03:04:35 AM UTC

### Overview



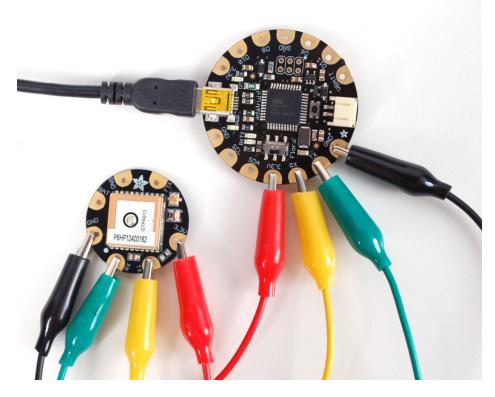


This module is the best way to add a GPS to your wearable project. It's part of the Adafruit Flora series of wearable electronics, designed specifically for use with the Flora motherboard. Installed on the PCB is the latest of our Ultimate GPS modules, a small, super-thin, low power GPS module with built in data-logging capability! This module's easy to use, but extremely powerful:

- -165 dBm sensitivity, 10 Hz updates, 66 channels
- Designed for wearable use with the Flora system
- Only 20mA current draw
- RTC battery-compatible sew a battery on to create a atomic-precision real time clock
- Built-in datalogging
- >25Km altitude
- Internal patch antenna + u.FL connector for external active antenna
- Fix status LED

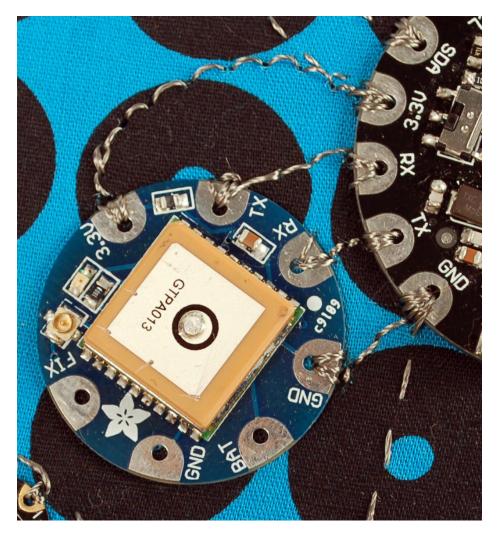
This guide will get you started with the Flora GPS.





Use alligator clips to connect Flora's 3.3V pad to the 3.3V pad on the GPS. Likewise connect RX to TX and TX to RX, then finally GND to GND.

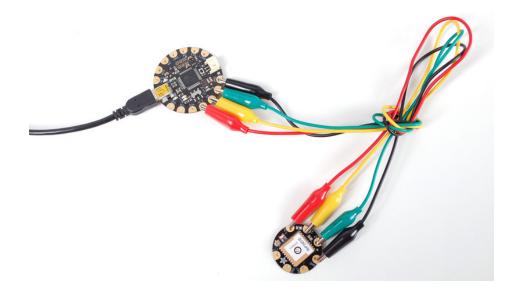
Unlike the GPS breakout and GPS shield we carry, the Flora GPS is for use with 3.3V power and logic only!



The four connectors are all in a row just to the left of Flora's JST battery connector for easy sewing. But test your project with alligator clips before sewing it in!

## Program FLORA





Make sure the USB cable is connecting your computer and Flora.

#### Basic Echo Test

We'll start with the most basic test, where we listen to the raw GPS data, to make sure it shows up! Copy and paste this code into a new sketch window and upload it to your Flora

### Install Adafruit GPS Library

Our helper library will make using the GPS easy since we have working code already. The library is available on GitHub (https://adafru.it/nCR)

You can download the most recent version from the Arduino library manager.

First, open up the Arduino library manager

File Edit Sk	etch Tools Help			
ØÐ	Verify/Compile Upload	Ctrl+R Ctrl+U	Δ	
demo	Upload Using Programmer		Manage Libraries	
demo // Demo #includ #includ	Export compiled Binary	Ctrl+Alt+S	Add .ZIP Library	rn
	Show Sketch Folder	Ctrl+K		
<pre>#includ #includ</pre>	Include Library	ArduinoHttpClient		
	Add File	ArduinoSound		

Search for the Adafruit GPS library and install it

Type All Topic All adafruit gps	
Adafruit GPS Library by Adafruit An interrupt-based GPS library for no-parsing-required use More info Version 1.2.1 v Install	4

We also have a great tutorial on Arduino library installation at: http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use

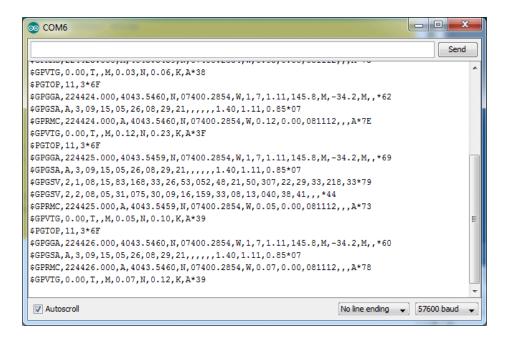
## Load Echo Demo

We'll begin by loading up the HardwareSerial echo test example available in the Adafruit\_GPS library

File) Edi	it Sketch	Tools Help				
	ew pen	Ctrl+N Ctrl+O				
	pen Recent etchbook	•	PS	Using Hardware Serial (e.g.	GPS H	flora or FeatherWing)
	amples			▲ Adafruit FRAM I2C	, ing.	. Best used with
Sa	ose ive ive As	Ctrl+W Ctrl+S Ctrl+Shift+S		Adafruit GPS Library Adafruit HDC1000 Library	1	blank due_parsing
	ige Setup	Ctrl+Shift+P Ctrl+P		Adafruit HMC5883 Unified Adafruit HX8357 Library Adafruit IL19340		due_shield_sdlog echo flora_dumplog
Pr	eferences	Ctrl+Comma		Adafruit INA219 Adafruit IO Arduino		flora_gpslog
	uit <del>ox one u</del> p	Ctrl+Q		Adafruit LED Backpack Library		GPS_HardwareSerial_EchoTest GPS_HardwareSerial_Parsing
// and	d help su	pport open s		Adafruit LIS3DH Adafruit MAX31855 library		leo_echo leo_locus_dumpbasic

Wire up the GPS module according to the hookup. Check your board and serial port settings (https://adafru.it/aRS) and upload this sketch to your Flora using the Upload button in the IDE. Open up the **Serial Monitor**.

You should see something like the following from the serial monitor. You may not have as many numbers, but there should be sentences that start with **\$GPRMC** and **\$GPGGA**, etc. If you see text like that it means your GPS and connection are working fine.





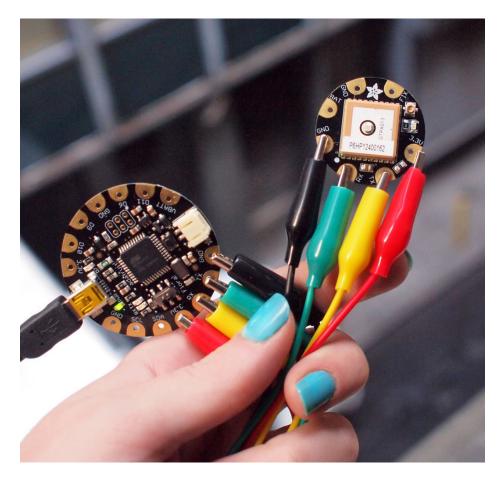
## Getting location data

#### Detailed GPS Test

Now that we know it basically works, we'll try to get 'fix data' from the GPS. For this, you will have to have the GPS outside. It cannot be inside a building, even if its right at the window. The silver antenna must be pointing up with a clear view of the sky!

💿 GPS_Ha	irdwareSe	erial_EchoTest	Ardui	ino 1.6.9		
File Edit	Sketch T	ools Help				
Sketo	n 1 Recent 1 hbook	Ctrl+N Ctrl+O		Neina Hardware Serial		
Exam			<u> </u>	Adafruit FRAM I2C		
Close	•	Ctrl+W		Adafruit GPS Library	-	blank
Save		Ctrl+S		Adafruit HDC1000 Library		due_parsing
Save	As	Ctrl+Shift+S		Adafruit HMC5883 Unified		due_shield_sdlog
Page	Setup	Ctrl+Shift+P		Adafruit HX8357 Library		echo
Print		Ctrl+P		Adafruit ILI9340		flora_dumplog
			-	Adafruit INA219		flora_gpslog
Prefe	rences	Ctrl+Comma		Adafruit IO Arduino		GPS_HardwareSerial_EchoTest
Quit		Ctrl+Q		Adafruit LED Backpack Library		GPS_HardwareSerial_Parsing
// and 1	eln sur	port open a	-	Adafruit LIS3DH	1	leo_echo
,, unu i	icip bup	poro open i		Adafruit MAX31855 library	•	leo_locus_dumpbasic

Wire up the GPS module according to the hookup



Check your board and serial port settings (https://adafru.it/aRS) and upload this sketch to your Flora using the Upload button in the IDE.

Place the GPS module (still connected to the Flora) outside. Once the GPS has located the satellite data, the red LED on the GPS will stop blinking.

If you see the LED blinking once a second, it does not yet have a fix!

It can take many minutes to get a fix if it doesn't see any satellites immediately.

Once it has a fix, you can check the serial monitor for the GPS data, which includes the current date and time in UTC. It will also give you your latitude, longitude and approximate altitude with the Serial monitor

GPS modules will always send data EVEN IF THEY DO NOT HAVE A FIX! In order to get 'valid' (not-blank) data you must have the GPS module directly outside, with the square ceramic antenna pointing up with a clear sky view. In ideal conditions, the module can get a fix in under 45 seconds. however depending on your location, satellite configuration, solar flares, tall buildings nearby, RF noise, etc it may take up to half an hour (or more) to get a fix! This does not mean your GPS module is broken, the GPS module will always work as fast as it can to get a fix.

© СОМ6		3
	Send	
\$PGTOP,11,3*6F		ŀ
\$GPGGA,225356.000,4043.5720,N,07400.2788,W,1,7,1.05,60.4,M,-34.2,M,,*57		
\$GPRMC, 225356.000, A, 4043.5720, N, 07400.2788, W, 0.42, 81.93, 081112, , , A*42		-
\$PGTOP,11,3*6F		Ľ
\$GPGGA,225357.000,4043.5720,N,07400.2786,W,1,7,1.05,60.4,M,-34.2,M,,*58		
\$GPRMC,225357.000,A,4043.5720,N,07400.2786,W,0.39,81.93,081112,,,A*41		
Time: 22:53:56.984		
Date: 8/11/2012		
Fix: 1 quality: 1		
Location: 4043.5717N, 7400.2783W		
Speed (knots): 0.39		
Angle: 81.93		
Altitude: 60.40		
Satellites: 7		
\$PGTOP,11,3*6F		
\$GPGGA,225358.000,4043.5719,N,07400.2785,W,1,7,1.05,60.4,M,-34.2,M,,*5E		
\$GPRMC,225358.000,A,4043.5719,N,07400.2785,W,0.34,81.93,081112,,,A*4A		
Autoscroll No line ending	▼ 57600 baud	-

Now we know where we are. According to the GPS, my location is **4043.5715 N** (Latitude 40 degrees, 43.5815 minutes North) & **07400.2783 W**. (Longitude 74 degrees, 0.2783 minutes West) To look at this location in Google maps, type **+40° 43.5715'**, **-74° 00.2783'** into the google maps search box (https://adafru.it/aMI). Unfortunately gmaps requires you to use +/- instead of NSWE notation. N and E are positive, S and W are negative.

People often get confused because the GPS is working but is "5 miles off" - this is because they are not parsing the lat/long data correctly. Despite appearances, the geolocation data is NOT in decimal degrees. It is in degrees and minutes in the following format: Latitude: DDMM.MMMM (The first two characters are the degrees.) Longitude: DDDMM.MMMM (The first three characters are the degrees.)

## Downloads



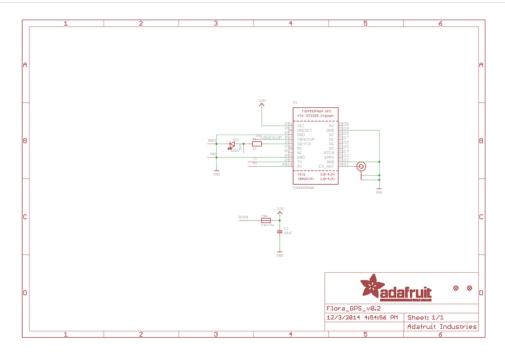
#### Files

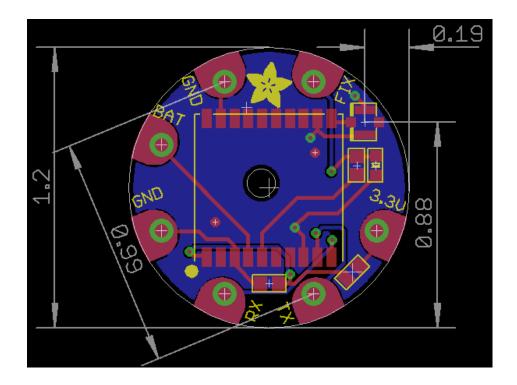
- EagleCAD PCB files on GitHub (https://adafru.it/rPD)
- Fritzing object in the Adafruit Fritzing Library (https://adafru.it/aP3)

#### Downloads:

- MTK3329/MTK3339 command set sheet (https://adafru.it/qif) for changing the fix data rate, baud rate, sentence outputs, etc!
- Datasheet for the PA6H (MTK3339) GPS module itself used in version 3 of this module (https://adafru.it/ria)
- MT3339 GPS PC Tool (windows only) (https://adafru.it/uoD) and the PC Tool manual (https://adafru.it/uoE)
- Mini GPS tool (windows only) (https://adafru.it/rid)

## Schematic & Fabrication Print





# **X-ON Electronics**

Largest Supplier of Electrical and Electronic Components

Click to view similar products for GPS Development Tools category:

Click to view products by Adafruit manufacturer:

Other Similar products are found below :

MAX2769EVKIT A9G SKY65725-11EK1 SKY65728-11EK1 TAU1201-EVK-A00 1059 1090 MIKROE-2045 1272 MDEV-GNSS-TM TEL0051 M20050-EVB-1 GPS-14414 ASD2501-R SIM808 EVK-M8BZOE-0 EVK-M8N-0 EVK-M8U SIM868 746 2324 4279 4415 M10578-A2-U1 ASX00017 AS-RTK2B-F9P-L1L2-NH-02 AS-RTK2B-LIT-L1L2-SMA-00 AS-STARTKIT-BASIC-L1L2-NH-02 AS-STARTKIT-LITE-L1L2-HS-00 AS-STARTKIT-LR-L1L2-EUNH-00 AS-STARTKIT-LR-L1L2-NANH-00 AS-STARTKIT-MCPIE-L1L2-0-00 AS-STARTKIT-MR-L1L2-NH-00 A2235HB04 M5310A-MBR M5312 EVA2035-H EVA2100-A EVA2200-A MAX2659EVKIT+ MAX2669EVKIT+ MIKROE-4673 MIKROE-1032 MIKROE-1714 MIKROE-1850 MIKROE-1887 MIKROE-1895 MIKROE-1912 MIKROE-2382 ML302