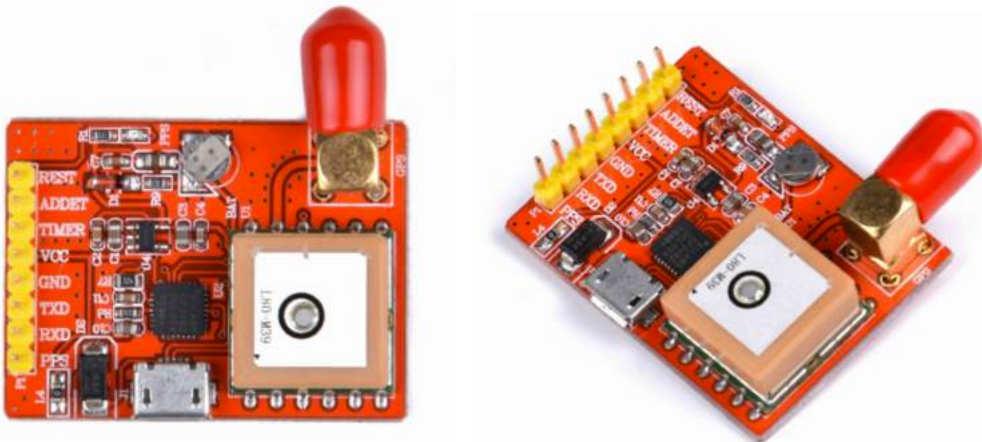


How to use GPS module for real-time location and PPS clock synchronization service

Wide Applications of USB-Port-GPS:

GPS Technology has been widely applied in the industries of Transportation, Measure, Emergency Rescue, Agriculture, Entertainment and Military.

Here is the picture for the module:



Principle:

USB-Port-GPS module acquires latitude and longitude, altitude, speed and other information about the current location by searching satellite and gets the command to print to the screen.

Tasks:

- Learn to use the serial port GPS
- Learn to use the software GPSD
- Use command to get the current latitude, longitude and speed information

Electronic Components:

- 1*Raspberry Pi with Power Supply
- 1*USB-PORT-GP Module
- 1* GPS Antenna

Operating System:



Distributor ID: Raspbian
Description: Raspbian GNU/Linux 8.0 (Jessie)
Release: 8.0
Codename: Jessie

Testing software: gpsd gpsd-clients python-gps

You may purchase all the components from www.52pi.com

Experimental Procedures

1. Wiring Connection

Connect the antenna to the antenna interface of GPS and keep the antenna exposed outside the room, preferably directly toward the sky. Then connect the GPS MicroUSB interface to Raspberry Pi USB interface with adapter cable.

2. Install and configure the software

Login the system and open a virtual terminal. Type:

```
sudo apt-get update
```

Then update the system. Install the packages:

```
sudo apt-get -y install gpsd gpsd-clients python-gps
```

Reconfigure the GPSD service. (If you need to use gpsd service as a daemon, you can use it to debug.)

```
sudo dpkg-reconfigure gpsd
```

This command is to start GPSD service and monitor.

```
sudo gpsd /dev/ttyUSB0 -F /var/run/gpsd.sock
```

You may run the below command to check if the GPS module is recognized:

```
sudo lsusb
```

If it is being recognized, you may see the similar information like below:

```
pi@raspberrypi:~$ sudo lsusb
Bus 001 Device 004: ID 10c4:ea60 Cygnal Integrated Products, Inc. CP210x UART Bridge / myAVR mySmartUSB light
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

If you need to restart the gps service, you can execute the following command:

```
sudo killall gpsd
```

If you want to configure the GPSD service as a background service, you can edit the configuration file / etc / default / gpsd configured as follows:

(Note the contents highlighted in red and type in the name of your own your device)

```
pi@raspberrypi:~$ grep -v "#" /etc/default/gpsd |grep -v "^$"
START_DAEMON="true"
USBAUTO="true"
DEVICES="/dev/ttyUSB0"
GPSD_OPTIONS="-F /var/run/gpsd.socket"
```

Once it is finished, you can start the following command:

Boot automatically and start the service

```
sudo systemctl enable gpsd.sock
```

```
sudo systemctl start gpsd.sock
```

Stop the service and turn off auto start

```
sudo systemctl disable gpsd.sock
```

```
sudo systemctl stop gpsd.sock
```

3 minutes after the startup, you may get GPS signal information through the following command:

```
sudo cgps -s
```

Then if you can get access to the information, it will display like the below picture. If not, try some more times. There will be output as long as there is GPS satellite synchronization.

Time:	2016-07-20T08:28:29.000Z	PRN:	Elev:	Azim:	SNR:	Used:
Latitude:	31.101534 N	193	74	099	32	Y
Longitude:	121.439684 E	17	71	047	17	Y
Altitude:	57.8 m	19	61	344	00	Y
Speed:	0.0 kph	6	54	305	15	Y
Heading:	328.7 deg (true)	137	45	139	36	Y
Climb:	n/a	28	37	183	39	Y
Status:	3D FIX (426 secs)	3	27	043	11	Y
Longitude Err:	+/- 4 m	9	23	122	22	Y
Latitude Err:	+/- 2 m	23	22	092	34	N
Altitude Err:	+/- 4 m	2	21	273	00	N
Course Err:	n/a	22	06	042	17	N
Speed Err:	n/a	12	06	325	20	N
Time offset:	0.327					
Grid Square:	PM01rc					

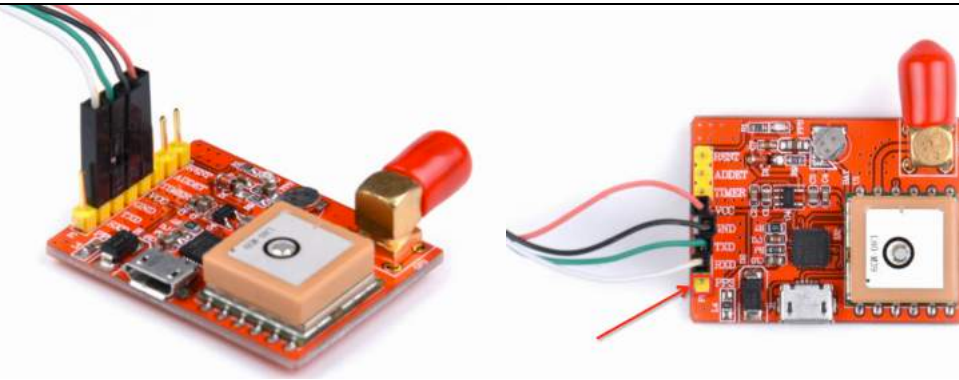
3. Start the PPS service to display satellite time

If you want to start PPS service, you have to achieve it through the serial Port PINS. Remove the USB cable of GPS and replace it with DuPont wires. You need a USB-to-TTL adapter cable to connect GPS and Raspberry Pi. What is more, you need a DuPont cable to connect the PPS to the GPIO NO.18 PIN. Here is the picture to help you connect the wires:

GPS module	Wire color
VCC	Red wire
GND	Black wire
TXD	Green wire
RXD	White wire
PPS	Yellow wire

Here is the picture to show you the wire connections:





Connect the PPS PIN to the GPIO number 18 PIN of Raspberry Pi:

```
pi@raspberrypi:~$ gpio readall
```

Pi 3											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA. 1	ALTO	1	3	4		5V			
3	9	SCL. 1	ALTO	1	5	6		0v			
4	7	GPIO. 7	IN	1	7	8	0	TxD	15	14	
		0v			9	10	1	RxD	16	15	
17	0	GPIO. 0	IN	0	11	12	0	GPIO. 1	1	18	
27	2	GPIO. 2	IN	0	13	14		0v			
22	3	GPIO. 3	IN	0	15	16	0	GPIO. 4	4	23	
		3.3v			17	18	0	GPIO. 5	5	24	
10	12	MOSI	IN	0	19	20		0v			
9	13	MISO	IN	0	21	22	0	GPIO. 6	6	25	
11	14	SCLK	IN	0	23	24	1	CEO	10	8	
		0v			25	26	1	CE1	11	7	
0	30	SDA. 0	IN	1	27	28	1	SCL. 0	31	1	
5	21	GPIO. 21	IN	1	29	30		0v			
6	22	GPIO. 22	IN	1	31	32	0	GPIO. 26	26	12	
13	23	GPIO. 23	IN	0	33	34		0v			
19	24	GPIO. 24	IN	0	35	36	0	GPIO. 27	27	16	
26	25	GPIO. 25	IN	0	37	38	0	GPIO. 28	28	20	
		0v			39	40	0	GPIO. 29	29	21	

Then login and execute this command:

```
sudo apt-get update
```

```
sudo apt-get dist-upgrade # Optional
```

```
sudo rpi-update # Optional
```

```
sudo reboot # Optional
```

Install pps-tools

```
sudo apt-get -y install pps-tools libcap-dev
```




Ensure that the latest package is available

```
pi@raspberrypi:~$ sudo apt-get -y install pps-tools libcap-dev
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
libcap-dev 已经是最新的版本。
pps-tools 已经是最新的版本。
升级了 0 个软件包，新安装了 0 个软件包，要卸载 0 个软件包，有 8 个软件包未被升级。
```

```
sudo dpkg-reconfigure tzdata # You can modify the time zone
```

Then edit the startup configuration file:

```
sudo vim.tiny /boot/config.txt
```

Add

```
dtoverlay = pps-gpio, gpiopin = 18
```

Exit and save

```
pi@raspberrypi:~$ grep -v "#" /boot/config.txt |grep -v "^$"
disable_overscan=1
hdmi_force_hotplug=1
hdmi_group=2
hdmi_mode=87
hdmi_cvt=800 480 60 6 0 0 0
device_tree=bcm2710-rpi-3-b.dtb
dtparam=i2c_arm=on
dtoverlay=pps-gpio, gpiopin=18
dtparam=audio=on
start_x=1
gpu_mem=128
```

Restart raspberry pi and check the loading status of the module by running lsmod command:

```
lsmod |grep pps
```

```
pi@raspberrypi:~$ lsmod |grep pps
pps_gpio                2655  1
pps_core                6779  2 pps_gpio
```

```
dmesg |grep pps
```

```
pi@raspberrypi:~$ dmesg |grep pps
[ 3.134770] pps_core: LinuxPPS API ver. 1 registered
[ 3.137111] pps_core: Software ver. 5.3.6 - Copyright 2005-2007 Rodolfo Giometti <giometti@linux.it>
[ 3.153410] pps pps0: new PPS source pps.-1
[ 3.155425] pps pps0: Registered IRQ 184 as PPS source
```



You may try the following command to test if you can get pps resources:

```
sudo ppstest /dev/pps0
```

If you can see the below picture. it means that you can get access to the information: trying PPS source "/dev/pps0"

```
found PPS source "/dev/pps0"
```

```
ok, found 1 source(s), now start fetching data...
```

```
source 0 - assert 1418933982.9980424, sequence: 970 - clear 0.0000000, sequence: 0
```

```
source 0 - assert 1418933983.9980454, sequence: 971 - clear 0.0000000, sequence: 0
```

Then you can download the latest NTP source package and compile it:

```
wget http://archive.ntp.org/ntp4/ntp-4.2/ntp-4.2.8p6.tar.gz # Download NTP Code
```

```
tar zxvf ntp-4.2.8p6.tar.gz # Extract the source
```

```
cd ntp-4.2.8p6 # Enter the source path
```

```
./configure --enable-linuxcaps # Precompiled
```

```
make -j 5 # Compile
```

```
sudo make install # Compile and install
```

```
sudo service ntp stop # Stop the NTP server
```

```
sudo cp /usr/local/bin/ntp* /usr/bin/ && sudo cp /usr/local/sbin/ntp* /usr/sbin/
```

Replace the local application file

```
sudo vim.tiny /etc/ntp.conf
```

Add those parameters into the file::

```
server 127.127.22.0 minpoll 4 maxpoll 4
```

```
fudge 127.127.22.0 refid PPS
```

Add "prefer"

```
server 0.debian.pool.ntp.org iburst prefer
```

Then save and exit. Restart the NTP service:

```
sudo service ntp restart
```

Wait for a while and then execute the following command:

```
ntpq -pn
```



If you can get pps0 information, it means that it is working. NTP server can debug the system time with satellites.

PS, if you want to see the version information, execute the following command:

```
ntpq -crv -pn
```

That is all for this tutorial. Thanks for watching.