

# GPS Module Interfacing with Raspberry Pi

## **Hardware Guide:**

For completing this lesson, you will require the following things along with your initial raspberry pi setup

1. GPS module
2. USB to TTL converter
3. Connecting wires

## **GPS Module:**

Global Positioning System (GPS) makes use of signals sent by satellites in space and ground stations on Earth to accurately determine their position on Earth.

Radio Frequency signals sent from satellites and ground stations are received by the GPS. GPS makes use of these signals to determine its exact position.

The signals received from the satellites and ground stations contain time stamps of the time when the signals were transmitted.

Using information from 3 or more satellites, the exact position of the GPS can be triangulated.



GPS receiver module gives output in standard (National Marine Electronics Association) NMEA string format. It provides output serially on Tx pin with default 9600 Baud rate.

This NMEA string output from GPS receiver contains different parameters separated by commas like longitude, latitude, altitude, time etc. Each string starts with '\$' and ends with carriage return/line feed sequence.

E.g.

\$GPGGA,184237.000,1829.9639,N,07347.6174,E,1,05,2.1,607.1,M,-64.7,M,,0000\*7D

\$GPGSA,A,3,15,25,18,26,12,,,,,,5.3,2.1,4.8\*36

\$GPGSV,3,1,11,15,47,133,46,25,44,226,45,18,37,238,45,26,34,087,40\*72

\$GPGSV,3,2,11,12,27,184,45,24,02,164,26,29,58,349,,05,26,034,\*7F

\$GPGSV,3,3,11,21,25,303,,02,11,071,,22,01,228,\*40

\$GPRMC,184237.000,A,1829.9639,N,07347.6174,E,0.05,180.19,230514,,,A\*64

## USING BUILT IN UART

### Enabling Built in UART

For enabling built in UART refer to page No. 80 of fingerprint section of this manual.

### Wiring up your circuit:

In this Raspberry Pi GPS Module interfacing project we connect GPS sensor to Raspberry Pi UART. So, first of all, we need to make the all the required connection as shown below. Connections are simple, we have just connected GPS module to Raspberry Pi UART port

1. Connect the VCC Pin of GPS Module to 3.3V Pin Pin 1 of Raspberry Pi.
2. Connect the GND Pin of GPS Module to GND Pin 6 of Raspberry Pi.
3. Connect the Tx Pin of GPS Module to Rx Pin 10 of Raspberry Pi.
4. Connect the RXD Pin GPS Module TXD Pin 8 of Raspberry Pi.

### Software Guide:

We can find whether our GPS module is working properly and the connections are correct by typing the following command: `sudo cat /dev/ttys0`

## USING USB TO TTL Converter Module

### Wiring up your Circuit:

1. Connect the VCC Pin of GPS Module to 3.3V Pin of USB to TTL converter
2. Connect the GND Pin of GPS Module to GND Pin of USB to TTL converter
3. Connect the Tx Pin of GPS Module to Rx Pin of USB to TTL converter
4. Connect the Rx Pin of GPS Module to Tx Pin of USB to TTL converter.
5. Lastly connect the USB to TTL converter to USB port of Raspberry Pi.

### Software Guide:

Open Terminal Window and type the following command to know to which USB port the GPS module is attached: `ls /dev/ttys0*`

We can find whether our GPS module is working properly and the connections are correct by typing the following command: `sudo cat /dev/ttys0*`

### Use 'gpsd':

You can always just read that raw data, but it's much nicer if you can have some Linux software prettify it. We'll try out `gpsd` which is a GPS-handling Daemon (background-helper)

### Installing a GPS Daemon (gpsd)

The first step is installing some software on your Raspberry Pi that understands the serial data that your GPS module is providing via `/dev/ttys0`.

Thankfully other people have already done all the hard work for you of properly parsing the raw GPS data, and we can use (amongst other options) a nice little package named '`gpsd`', which essentially acts as a layer between your applications and the actual GPS hardware, gracefully handling parsing errors, and providing a common, well-defined interfaces to any GPS module.

To install `gpsd`, make sure your Pi has an Internet connection and run the following commands from the console:

1. `sudo apt-get update`
2. `sudo apt-get install gpsd gpsd-clients python-gps`

And install the software as it prompts you to do.