<Stuff about="code" />

if stuff == "code" then about = True

THURSDAY, 19 SEPTEMBER 2013

Raspberry Pi - GPS Setup and Python

I got myself one of adafruit's ultimate GPS breakout boards as I want to experiment with capturing GPS data in my car projects. Its a seriously good bit of kit and if you looking for a GPS module you could do a lot worse than this. They also have an excellent tutorial on setting it up with the raspberry pi, http://learn.adafruit.com/adafruit-ultimate-gps-on-the-raspberry-pi.

If your in the UK, I noticed that they had them on amazon and pimoroni.

I used the raspberry pi's on board UART to connect to the GPS module, Adafruit advocate using a USB to serial device but that didn't suit my needs (I need the USB for other things).

I also create a GPSController class in python to allow me to communicate with the module easily.

Connecting the GPS module
Wiring it up is pretty simple:

- Raspberry Pi - 5v -> GPS Module - VIN
- Raspberry Pi - GND -> GPS Module - GND
- Raspberry Pi - Tx -> GPS Module - Rx
- Raspberry Pi - Rx -> GPS Module - Tx

SEARCH

Search

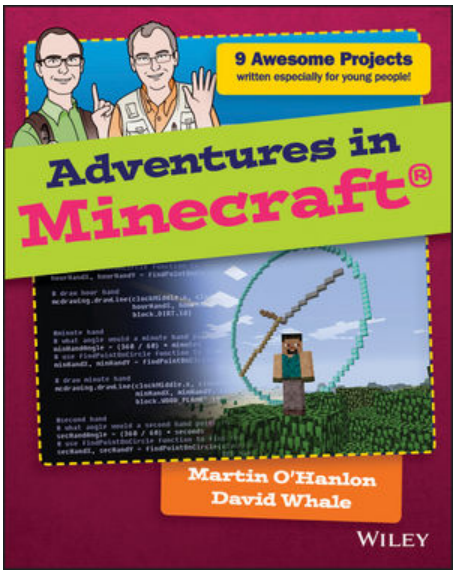
ABOUT ME

Martin O'Hanlon

StuffAboutCode.com YouTube channel

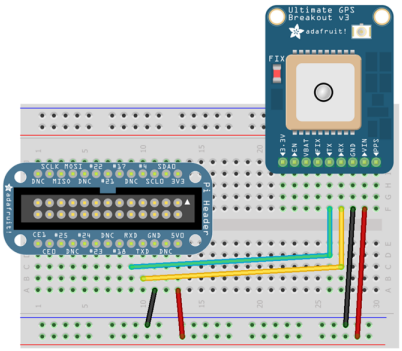
Follow @martinohanlon

Available on Amazon (UK , US)



POSTS

.net (4) Adventures in Minecraft (4) asp (3) c (3) c# (4) camera (7) Car (4) games (18) get_iplayer



Enable the UART
By default the UART is enabled to allow you to connect a terminal window and login, I needed to disable this to free it up for the GPS Module.

Edit the boot options to change the UART so it doesnt provide a terminal connection by default:

```
sudo nano /boot/cmdline.txt
```

The contents looks like this (yours might be slightly different depending on your distribution):

```
dwc_otg.lpm_enable=0 console=ttyAMA0,115200 kgdboc=ttyAMA0,115200
console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=deadline
rootwait
```

remove the following config from the file and save it.

```
console=ttyAMA0,115200
kgdboc=ttyAMA0,115200
```

Change inittab so it doesnt spawn a login to the serial connection:

```
sudo nano /etc/inittab
```

Change:

```
#Spawn a getty on Raspberry Pi serial line
T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100
```

to:

```
#Spawn a getty on Raspberry Pi serial line
#T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100
```

Reboot

```
sudo shutdown -r now
```

Install GPSD
GPSD is an open source project which provides a daemon which streams GPS data via a TCP socket, allowing you to communicate with a whole host of different GPS devices (not just this one):

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

Run gpsd
GPSD needs to be started up, using the following command:

(5) gpio (9) gps (2) html (2) minecraft (39) Python (61) raspberry pi (77) raspbmc / xbmc (4) robot (3) RSS (4) scratch (1) social networking (8) xml (2)

LATEST POSTS

Raspberry Pi CPU Temperature
For a project I am working on I needed a really quick way of reading the Raspberry Pi's CPU temperature, most of the solutions I found ran the ....

Astro Pi Snake Game
I recently made Snake for the Astro Pi, just for fun really, but I am putting it online as I would like to see what others can do with it - it is ....

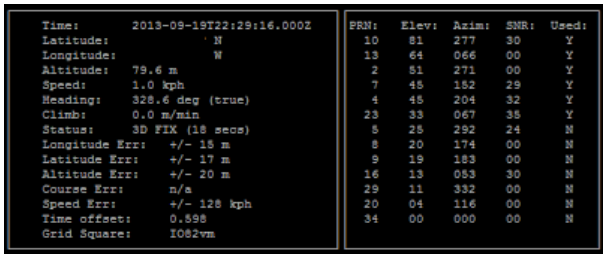
Interactive Minecraft Astro Pi
I have created what I think is a fun way to explore your new Astro Pi - its an interactive Astro Pi in Minecraft.All of the components on the board ....

```
sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd.sock
```

Test gpsd
There is a simple GPS client which you can run to test everything is working:

```
cgps -s
```

It may take a few seconds for data to come through, but you should see a screen like this:



Python code
In order to control and capture the GPS data in python I started looking round for some code, this took me to http://www.danmandle.com/blog/getting-gpsd-to-work-with-python which I used as a starting point for creating this python GPSController class. It uses threading to continuously read the stream of data from GPSD and present it as properties of the class.

There is a more in depth example of how to use the class in the code below, but in simple terms you use it like this:

```
#create controller
gpsc = GpsController()

#start controller
gpsc.start()

#read latitude and longitude
print gpsc.fix.latitude
print gpsc.fix.longitude

#stop controller
gpsc.stopController()
```

Note - you may have to wait a few seconds for the data to start streaming

GPSController.py

```
from gps import *
import time
import threading
import math

class GpsController(threading.Thread):
    def __init__(self):
        threading.Thread.__init__(self)
        self.gpsd = gps(mode=WATCH_ENABLE) #starting the stream of
info
        self.running = False

    def run(self):
```

```
        self.running = True
        while self.running:
            # grab EACH set of gpsd info to clear the buffer
            self.gpsd.next()

    def stopController(self):
        self.running = False

    @property
    def fix(self):
        return self.gpsd.fix

    @property
    def utc(self):
        return self.gpsd.utc

    @property
    def satellites(self):
        return self.gpsd.satellites

if __name__ == '__main__':
    # create the controller
    gpsc = GpsController()
    try:
        # start controller
        gpsc.start()
        while True:
            print "latitude ", gpsc.fix.latitude
            print "longitude ", gpsc.fix.longitude
            print "time utc ", gpsc.utc, " + ", gpsc.fix.time
            print "altitude (m)", gpsc.fix.altitude
            print "eps ", gpsc.fix.eps
            print "epx ", gpsc.fix.epx
            print "epv ", gpsc.fix.epv
            print "ept ", gpsc.gpsd.fix.ept
            print "speed (m/s) ", gpsc.fix.speed
            print "climb ", gpsc.fix.climb
            print "track ", gpsc.fix.track
            print "mode ", gpsc.fix.mode
            print "sats ", gpsc.satellites
            time.sleep(0.5)

    #Error
    except:
        print "Unexpected error:", sys.exc_info()[0]
        raise

    #Ctrl C
    except KeyboardInterrupt:
        print "User cancelled"

    finally:
        print "Stopping gps controller"
        gpsc.stopController()
        #wait for the tread to finish
        gpsc.join()

    print "Done"
```

Posted by Martin O'Hanlon at 23:52

+1 +34 Recommend this on Google

Labels: Car, Python, raspberry pi

26 comments:

vaila ruthvik 23 October 2013 at 07:44

how can you give 5v to raspberry pi.....???

Reply

Replies

Martin O'Hanlon 23 October 2013 at 10:49

Im not sure I understand your question. But, if your talking about how do I wire the GPS unit up. I take the 5v line from the raspberry pi gpio and connect it to the VIN on the gps module.

I dont give the raspberry pi 5v!

Reply

parkview78 9 November 2013 at 12:57

What is the correct way to test if:
1) the GPS unit is present, ie: is there an error code genberated when gpsc.start() is run and even though gpsd is running, there is no GPS unit attached?
2) test to see if the data read is good? At the moment, I just loop until lat/log are both no longer 0.0.

thanks.

Reply

Replies

Martin O'Hanlon 12 November 2013 at 21:07

I dont know a better way to be honest, ive not looked into it, if you find out let me know though

Reply

Jesus Chavez 2 December 2013 at 13:55

Which gps module are you use?

Reply

Replies

Martin O'Hanlon 2 December 2013 at 14:46

adafruit gps ultimate breakout board (see link at the top of the page)

Reply

Faaiz 11 March 2014 at 22:55

hi martin.where should i go to create the python GPSController class

Reply

Frank Ogiamien 12 March 2014 at 00:19

Is there a kit that would allow the GPS to automatically download via wifi when the vehicle returns home? Thx

Reply

Replies

Martin O'Hanlon 28 March 2014 at 08:50

I dont know of a 'kit' but it doesnt sound like a particularly difficult problem to code. A script which runs every x minutes, if it can find a connection, it copies the file of gps data to a share.

Reply

Edgar Omar Cortes Favila 3 April 2014 at 23:31

when i reboot my raspberry doesnt turn on again ... looks like a broke boot particion, i had to reinstall 3 times ._. some idea why ?

Reply

Replies

Martin O'Hanlon 4 April 2014 at 06:50

Knackered SD Card? Have you tried using a different one?

Chris Sutherland 7 April 2014 at 21:33

I get the same problem I'm afraid. After reboot I get a kernel issue and have to reload. Tried another SD and the same thing happens. I don't think all packages were 100% up to date, so im going to give it another shot.

Martin O'Hanlon 7 May 2014 at 14:46

Did you have any luck? I have never experienced this problem before?

hawkins 1 June 2014 at 01:06

Any one else had luck with this. Im getting the same problem

Martin O'Hanlon 8 August 2014 at 20:23

Right I have managed to re-produce the error. Its because the raspbian / noobs setup has changed. Where I originally said:

Change:

dwc_otg.lpm_enable=0 console=ttyAMA0,115200 kgdboc=ttyAMA0,115200 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=deadline rootwait

to:

dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=deadline rootwait

You need to just remove the:

console=ttyAMA0,115200 kgdboc=ttyAMA0,115200

parts.

If you cut and paste my code directly it would have resulted in raspbian being unable to find the root and the kernal error. Ive updated the post.

Rene N 28 September 2014 at 04:31

Hello,

I had a problem with this part of the code:

#Spawn a getty on Raspberry Pi serial line
T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100

to:

#Spawn a getty on Raspberry Pi serial line
#T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100

When I would try to reboot..nothing would happen! It's like I had to re-install everything on the sd-card. I rebooted my pc with a debian live cd and change the above code back. My PI booted back like normal.

In order not to run into this problem again:

on console I entered :

sudo raspi-config In the software configuration tool select option 8 then select option A7 Serial and select NO.
Select OK and close app.
In console enter:

sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd.sock
cgps -s

wait for gps data

That was my work around for serial (none USB method)
The last two steps have to be re-entered everytime you reboot. Just for testing reasons I did not automatically had it run on boot time.

Reply

sun bae yim 16 June 2014 at 14:54

I can't run GPScontroll.py
Why occured this error?

pi@raspberrypi ~ $ python GPSController.py
File "GPSController.py", line 53
time.sleep(0.5)
SyntaxError: default 'except:' must be last

Reply

Replies

Martin O'Hanlon 16 June 2014 at 14:56

It sounds like a code indentation problem. Did you cut and paste the code from the page? Did it cut and paste properly?

Lars 13 July 2014 at 00:35

I got the same error, but managed to fix it. The lines
except:
print "Unexpected error:", sys.exc_info()[0]
raise
must be moved so they come _after_
except KeyboardInterrupt:
print "User cancelled"

Thanks for the excellent write-up Martin!
Cheers, Lars

Reply

Comediante 13 September 2014 at 12:16

Hi,

do you know how to send the data you get from GPSController.py to a html page on webiopi?

Cheers.

Reply

Replies

Martin O'Hanlon 15 September 2014 at 09:40

I have never looked at webiopi. I doubt it would be too difficult though.

Reply

Tyler Marler 6 November 2014 at 09:13

I am trying to run this on a RPi in idle3 with the adafruit ultimate gps but I keep getting a syntax error on the quotation Mark after latitude. Help!

Reply

Replies

Martin O'Hanlon 6 November 2014 at 14:08

Ah Python 3. I wrote this in Python 2. Its probably because Python 3 expects While(True): and print("hello") - note the brackets. I cant guarantee there wont be other python 2 - 3 issues.

Reply

Qisthi Al-Hazmi HR 15 January 2015 at 15:21

This is really useful, great!
but by the way, how to send the output data (longitude and latitude) from py code to website (html)? Is it need other library or package? or if I start the gpsd I can be able to call the longitude and latitude in html without python code like this example http://www.w3schools.com/html/html5_geolocation.asp? thank you

Reply

Srinivas Arcot 5 March 2015 at 16:05

Did you use GPS library module in Python 3? How did you do that?

Reply

Replies

Martin O'Hanlon 5 March 2015 at 16:09

Do you mean the GPS modue I wrote (i.e. the one in the blog post)? If so the code I wrote is python 2 but I dont think it would be that difficult to migrate it to Python 3.

Reply

Enter your comment...

Comment as: Google Accour

Publish Preview

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)