



## Raspberry pi loves Sensors and LEDs

by [sajingeo](#) on October 22, 2012

### Table of Contents

Raspberry pi loves Sensors and LEDs .....	1
Intro: Raspberry pi loves Sensors and LEDs .....	2
Step 1: Parts .....	2
Step 2: Tools .....	4
Step 3: Temp. on a Site .....	5
Step 4: Let Raspberry pi check my email .....	5
Step 5: More fun with raspberry pi .....	6
Related Instructables .....	6
Advertisements .....	6

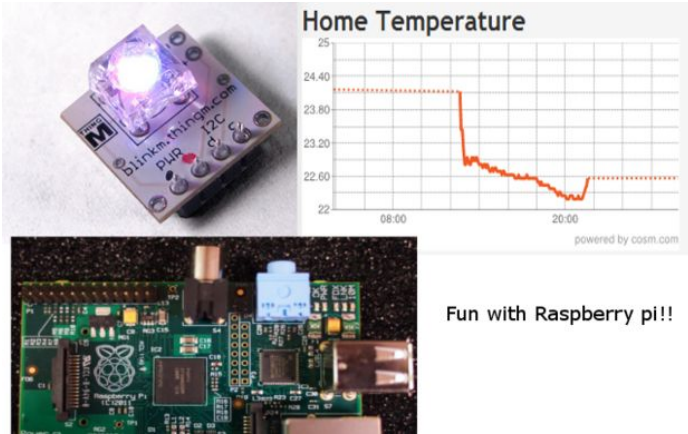


Author: **sajingeo Mysrobonics**  
DIY, electronics...and loves getting creative...

## Intro: Raspberry pi loves Sensors and LEDs

This tutorial teaches you how to talk to I2C devices on a raspberry pi, this is based on the ADAfruit tutorials but i have made the code more generic and you can easily modify them to work with any sensors.

This tutorial uses the distro provided by adafruit, if you dont have the distro up an running re-visit one of my old tutorial on raspberry pi.



### Step 1: Parts

i am using a **Mblink** and **PMODtemp** (from digilent) for this tutorial, this approach would work for most I2C devices. connect the I2C sensor to the I2C pins refer the connection diagram from [here](#) , make sure you connect the a 3.3v I2C sensor.

#### connect

```
SCL>>> SCL
SDA >>> SDA
GND >>>GND
VCC >>>VCC
```

the diagram below is from [http://elinux.org/RPi\\_Low-level\\_peripherals](http://elinux.org/RPi_Low-level_peripherals)



- Image Notes**
1. Data
  2. Serial Clock
  3. GND
  4. 3.3v

## Step 2: Tools

make sure the tools are installed... to test if the tool are installed run the following command..

```
sudo i2cdetect -y 0
```

this will scan all devices connected to the I2C bus. it should look like the pic below. if you see that i2ctools are not installed run the following comand

```
sudo apt-get install i2c-tools
```

also install

```
sudo apt-get install python-smbus
```

```
sudo apt-get install python-pip
```

```
sudo pip install rpi.gpio
```

```
sudo /home/pi/Desktop
```

```
wget -O geekman-python-eeml.tar.gz https://github.com/geekman/python-eeml/tarball/master
```

```
tar zxvf geekman-python-eeml.tar.gz
```

```
cd geekman-python-eeml*
```

```
sudo python setup.py install
```

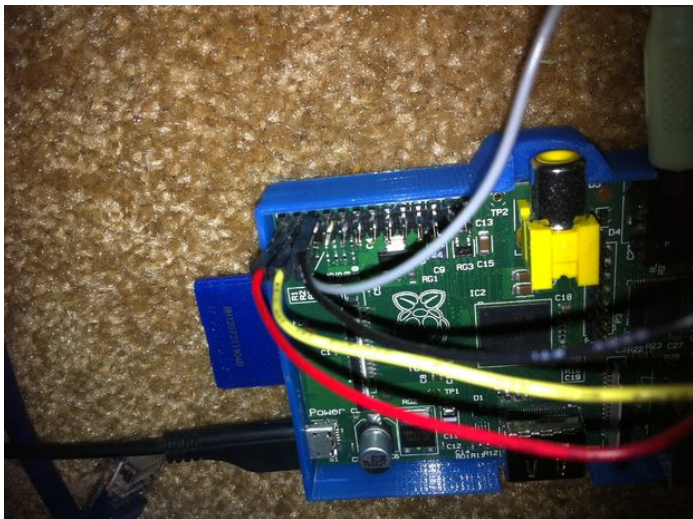
note down the address of the I2C device that you have connected. in this case i have connected a temp. sensor PMODTEMP2 from digilent. we use the smbus function for I2C communication.

```
pi@raspberrypi: ~
pi@raspberrypi ~$ sudo kill 2350
pi@raspberrypi ~$ ls
Desktop  python_games  recipe-502241-1.py
pi@raspberrypi ~$ i2cdetect
Error: No I2C-bus specified!
Usage: i2cdetect [-y] [-a] [-q|-r] I2CBUS [FIRST LAST]
i2cdetect -F I2CBUS
i2cdetect -1
I2CBUS is an integer or an I2C bus name
If provided, FIRST and LAST limit the probing range.
pi@raspberrypi ~$ i2cdetect -y 0
Error: Could not open file '/dev/i2c-0': Permission denied
Run as root?
pi@raspberrypi ~$ sudo i2cdetect -y 0
00:  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
pi@raspberrypi ~$
```



### Image Notes

1. address of your device





### Step 3: Temp. on a Site

now you can setup your COSM account for receiving the sensor data... note the key and feed ID down, you can also refer this tutorial for more detailed steps

to make things easier you can download the git code to home folder. doing the following...

```
cd /home/pi/Desktop
```

run the code

```
git clone https://github.com/sajingeo/raspberry-pi-saj.git
```

to get a copy of my repository on your desktop.

```
cd raspberry-pi-saj
```

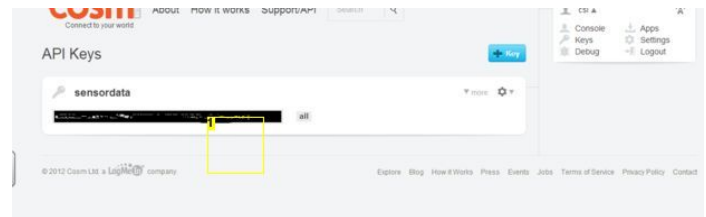
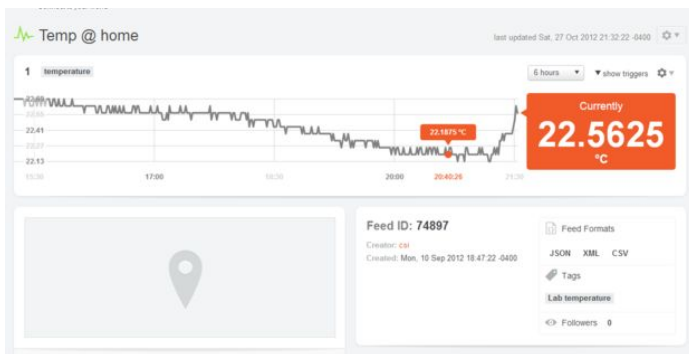
```
cd cosm-temp-logger
```

here you will see two files, open the i2cpy.py using nano

```
nano i2cpy.py
```

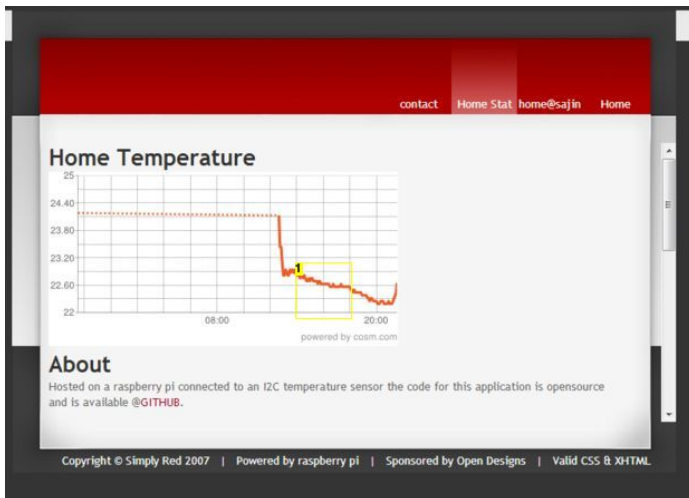
read the code for comments, the Adafruit library that is included in the files is very good for I2C operations and should be used to avoid common I2C mistakes...thanks to ADAFRUIT...

you should be able to now embed the temperature graph on your website, and host it on the raspberry pi refer this tutorial.



#### Image Notes

1. you key



#### Image Notes

1. website hosted on raspberry pi showing temp graph

### Step 4: Let Raspberry pi check my email

this uses raspberry pi to check my email and control Mblink to notify me of a new email...

connect the Mblink as shown in **step 1** . and find the address as shown in **step 2** ...

go to /mail-checker folder in my git repo and open i2cLED.py

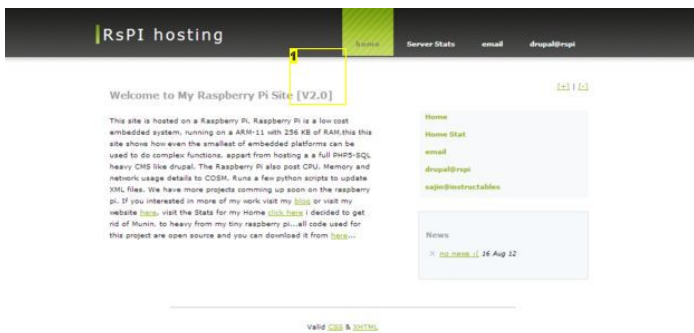
change the **user name** and **password** for your mail account and set the **MAILSNOW** to your current inbox unread mails number and **MAILCHECK** as the time duration for checking mails. (make sure you are connected to the internet)

run the code and you will see that the mail LED goes green when you have now new emails and RED when you GOT MAIL...



### Step 5: More fun with raspberry pi

Do check out other projects in the GIT repo, that includes a MP3 player and a CPU usage logger(2 types), the README file in the folders should explain on how to get them running...happy hacking with raspberry pi... you can check out my raspberry pi website here



#### Image Notes

1. <http://rspi.no-ip.org>

#### Related Instructables



**Use ssh to talk with your Raspberry Pi.** by antares72



**Install a webserver on Raspberry Pi.** by antares72



**RaspberryPi Media center - XBMC (video)** by hackitbuildit



**Connect Raspberry Pi to Projector or TV** by tim.ding



**Raspberry Pi as webserver.** by antares72



**Blinking LED with Raspberry Pi** by rahulkar