Home
Install or Upgrade PrivateEyePi

Projects
Home Alarm System project
Wireless Projects
Temperature Sensor Project using DS18B20
Temperature and Humidity using the DHT22 sensor
Rfu-328 - Programmable Wireless Processor
Webcam Surveillance Projects
On/Off Project - Switch a light on/off using your smart phone
Siren Project
Arm/Disarm Using a Switch
Home Alarm System for Raspberry Pi - How it works
Automate your alarm and sensors

Support
Discussion Forum
Having Problems?

g+1   7

Other
Contact Us
Donations
Previous Versions
Sitemap
Recent site activity

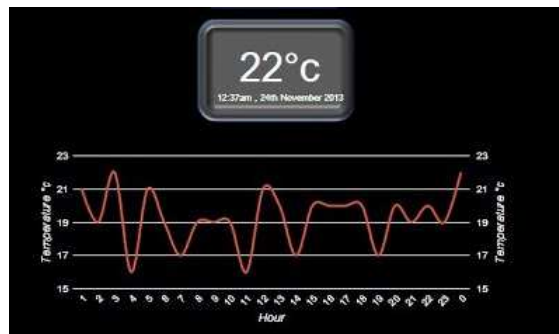Home >

Monitor your home temperature using your Raspberry Pi

Note: There is a newer tutorial that replaces this one that supports multiple temperature sensors.

At the end of the project you will have a WWW dashboard that looks similar to this:

Multiple Sensors

Display a temperature history graph per sensor

What you need:

- Raspberry Pi Model A or B
- The following parts are available through the PrivateEyePi Store:
    - DS18B20 sensor(s) (TS01), one per temperature reading you want on the dashboard
    - 4.7k resistor (one per sensor) (TS01)
    - Jumper wire cables and breadboard or wire, soldering iron and solder (JW01, BB01)

For support, queries or suggestions please email support@privateeyepi.com or leave comments in our blog.

Step 1: Build and test the Electronics:

Parts can be purchased from our store: http://www.privateeyepi.com/store

- Purchase a DS18B20 digital thermometer

- Purchase a 4.7k Ω (ohm) resistor
- As per *Figure 1 p*ut the resistor between pins 2 and 3 as in the diagram
- Connect Pin 3 of the temperature gauge to 3.3v GPIO Pin
- Connect Pin 2 of the temperature gauge to GPIO4 (Pin 7)
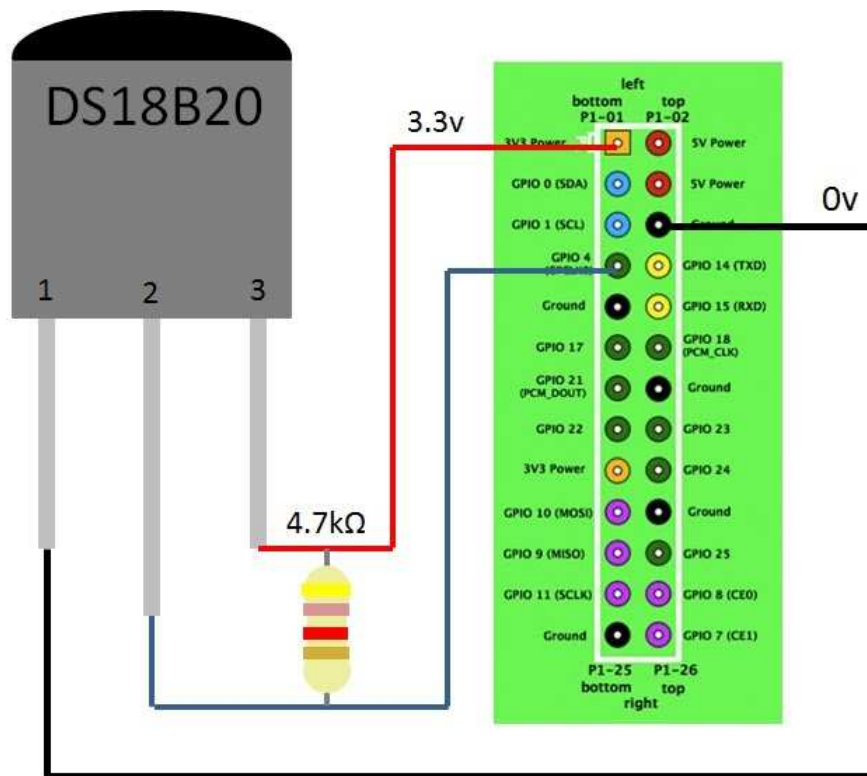- Connect Pin 1 of the temperature gauge to ground (Pin 6)

Figure 1. DS18B20 temperature sensor circuit diagram

If you want to add multiple sensors daisy chain the middle pin of the sensors. This will work because every DS18B20 sensor has a unique ID that is used to differentiate the signals coming from the sensors
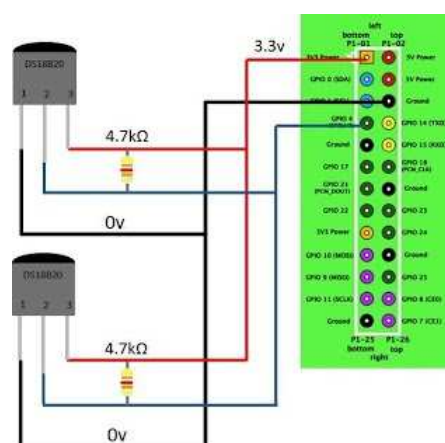See *Figure 2* below.

Figure 2. Multiple DS18B20 temperature sensor circuit diagram

Once built, follow the following steps to test that it is working:

Log in to your Raspberry Pi

At the prompt pi@raspberrypi / $

Type:

sudo modprobe w1-gpio

sudo modprobe w1-therm
cd /sys/bus/w1/devices/
ls

You will see a listing of the current directory. There should be a directory that is the serial number of your temperature gauge. The serial number of my thermometer 28-0000040be5b6, but every one will be unique. Make a note of yours as you will need it later in step 4. If you have multiple sensors there will be multiple directories listed.

If you don't see a directory with lots of numbers and letters like the one above then:

- Check your circuit wiring.
- Make sure you have the correct resistor (this is very important - yellow, violet, red, gold).
- Feel the temperature gauge with your finger. If it feels hot then you have it wired back to front.

If you do see the directory then type cd followed by the directory name:

cd 28-0000040be5b6
cat w1_slave

You will now see a dump of the w1_slave file that contains the temperature data in celcius (refer *Figure 3)*. 20812 is 20.812 degrees celcius. The dashboard does support a Fahrenheit setting that we will cover later if that is your preference.
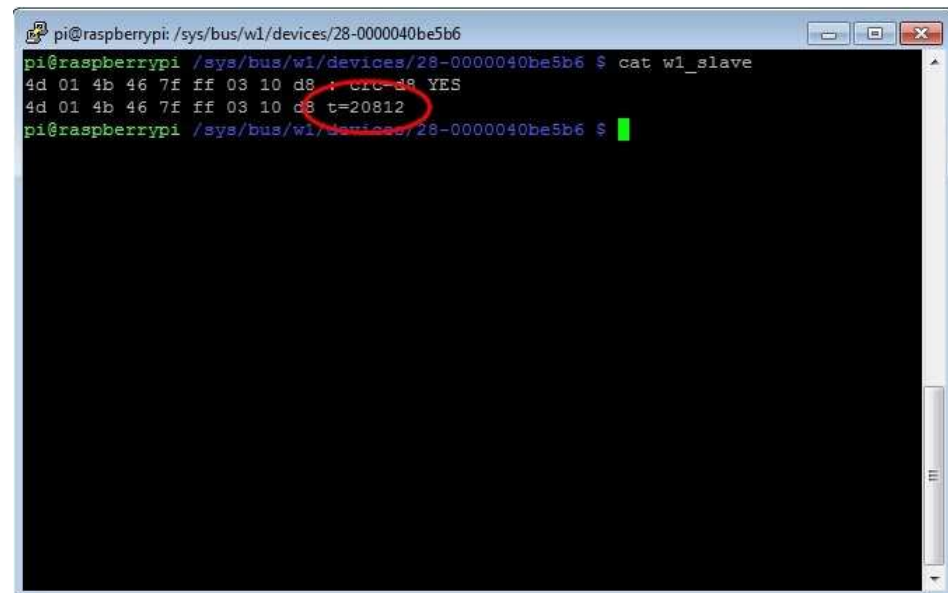

Figure 3. The contents of w1_slave file that contains the temperature for the DS18B20 sensor

Step 2: Register user and password

If you have not yet registered a user and password with PrivateEyePi follow the following steps, otherwise proceed to Step 3. Note this is a completely free service and has no trial period.

- Go to www.privateeyepi.com
- Click on "New User" option top right of the screen.
- Enter your the details on the screen and click update

Step 3: Configure GPIO 4

This configuration tells the system that you have connected a temperature gauge to GPIO 4 and to display it on your dashboard.

- As per *Figure 4* click on the GPIO menu option
- Click "Add"
- Type 7 in the "number" field
- Give it a Description of "GPIO 4", or anything else you like

Figure 4. PrivateEyePi setting for a new sensor

- Click "Update"
- As per *Figure 5* click the "Location" menu item
- Enter a Description, e.g. "Temperature Gauge"
- Select GPIO Port 4
- Select Type "Temperature"
- Click "Update"

Figure 5. PrivateEyePi location settings

If you have multiple sensors then repeat this step and add other GPIO and Locations settings for each sensor. In *Figure 6* I have configured three sensors, one for basement, ground floor and first floor. At least one must have a Number of 7, and the others can have any number you choose. Choose a number that does not conflict with any of the GPIO number of the Raspberry Pi. I have used 88 and 89. In step 4 you will use these numbers to associate the sensors on the Raspberry Pi with the temperature gauges on the dashboard.
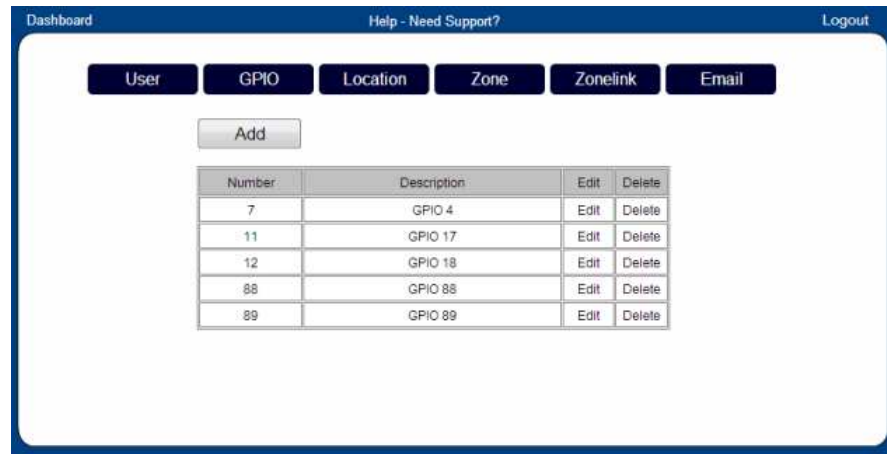
Figure 6. Multiple temperature gauge settings

Step 4: Install/Upgrade your PrivateEyePi software

Follow the following link to the tutorial -> Install PrivateEyePi Software

Step 4: Configure the program

- Follow the following link to enter your username and password in the globals.py file

    Enter username and password in the globals.py file

- We have assumed you installed the PrivateEyePi software in the /home directory otherwise you should replace /home in the following steps with your directory.
- Edit the globals.py file in order to enter your user and password by typing the following commands at the command prompt:

cd /home
sudo nano globals.py

Page down until you see the DallasSensorNumber settings (as per Figure 8). You need to place the GPIO number that you configured in step 3 (Figure 5) in between the brackets (e.g. DallasSensorNumber(7)), replacing the numbers you see in the file with your own numbers. If you have multiple sensors then repeat the line with the other GPIO numbers (as per Figure 8). Then do the same for the directory name replacing the directory names you see with your directory names that you made a note of in step 1.
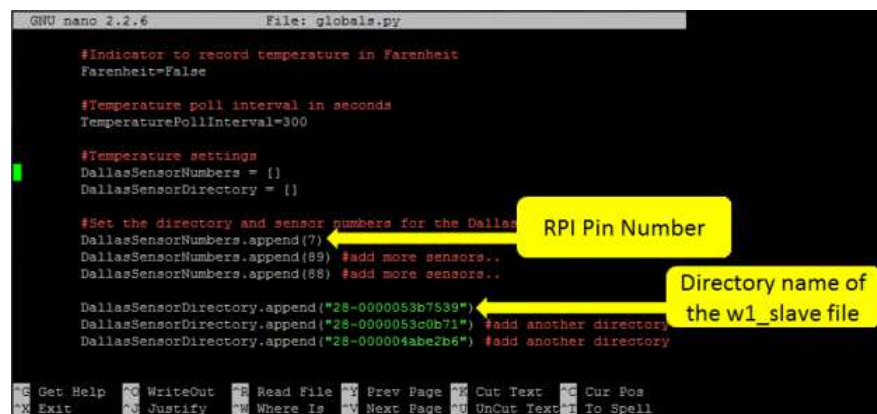

Figure 8. Enter lines for each sensor number and w1_slave directory

If you want your temperature to be display in Fahrenheit instead of Celsius then find the line that says Farenheit=False and change it to Farenheit=True (refer the top two lines of Figure 8).

Press CTRL-X
Then press Y to save the file
Press ENTER

Step 5: Run dallas.py program

pi@raspberrypi / $

Type
  cd /home
  sudo python dallas.py

If you are getting errors or problems see here for diagnostic steps.

Step 6: View your dashboard

- Go to www.privateeyepi.com
- Log in, or click on the "Dashboard" link on top left
- You should now see the temperature displayed