Raspberry Pi Alarm System

Technical Documentation

Guillaume Vigeant

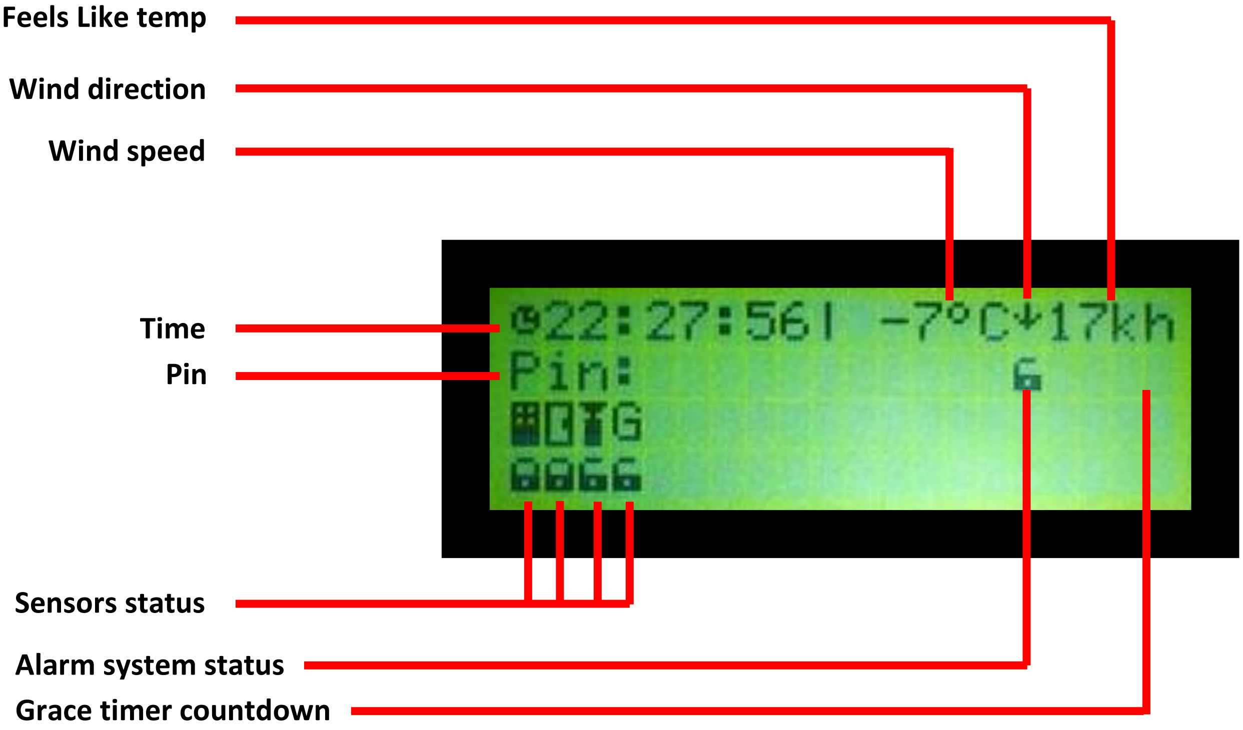
4/20/2013

The Raspberry Pi Alarm system is a DIY system built around the Raspberry Pi computer. It provides normal home alarm system functionalities as well as some added features such as sms message notifications, remote configuration and administration through ssh and weather display on interface unit.

Overview

The Raspberry Pi Alarm system is a DIY system built around the Raspberry Pi computer. It provides normal home alarm system functionalities as well as some added features such as sms message notifications, remote configuration and administration through ssh and weather display on interface unit.

Interface



Operation

The alarm system operates in very much the same way as a normal alarm system. It is armed and disarmed by entering the nip.

Arming sequence

If the alarm is in the idle state (*Alarm system status* icon shows an unlocked lock), it can be armed by entering the nip. When arming, the system immediately takes a snapshot of the sensors and will essentially only monitor the sensors that were closed at the time of arming and ignore the rest. This allows arming the system even if you want to leave the back door or the garage open for example. The arming sequence then enters a grace period of 30 seconds (configurable) to allow the user to leave the house. During this grace period, sensors are allowed to change state without triggering the alarm. The grace timer is displayed on the LCD and a audible beep is emitted every seconds. The last five seconds play three beeps to warn that the grace period is almost over.

Disarming sequence

When the system is armed, it can be disarmed by entering the pin again. If a monitored sensor is open while in the armed mode, it starts a grace timer of 30 seconds to let the user disarm the system by entering the pin. If the user does not successfully disarm the system before the 30 seconds, it transitions into alarm mode where a loud siren is played for 15 minutes and an SMS is sent to the user indicating what sensor triggered the alarm. When in the alarm state, the locked lock icon is replaced with the word *ALERT.* Note that at any point in time the keypad+LCD unit may be unplugged or disconnected without affecting the operation of the alarm system. This prevents the alarm system from stopping if the burglar cuts the wires. If the keypad and LCD are reconnected in operation, it will reinitialize itself and will be usable again.

Installation Instructions

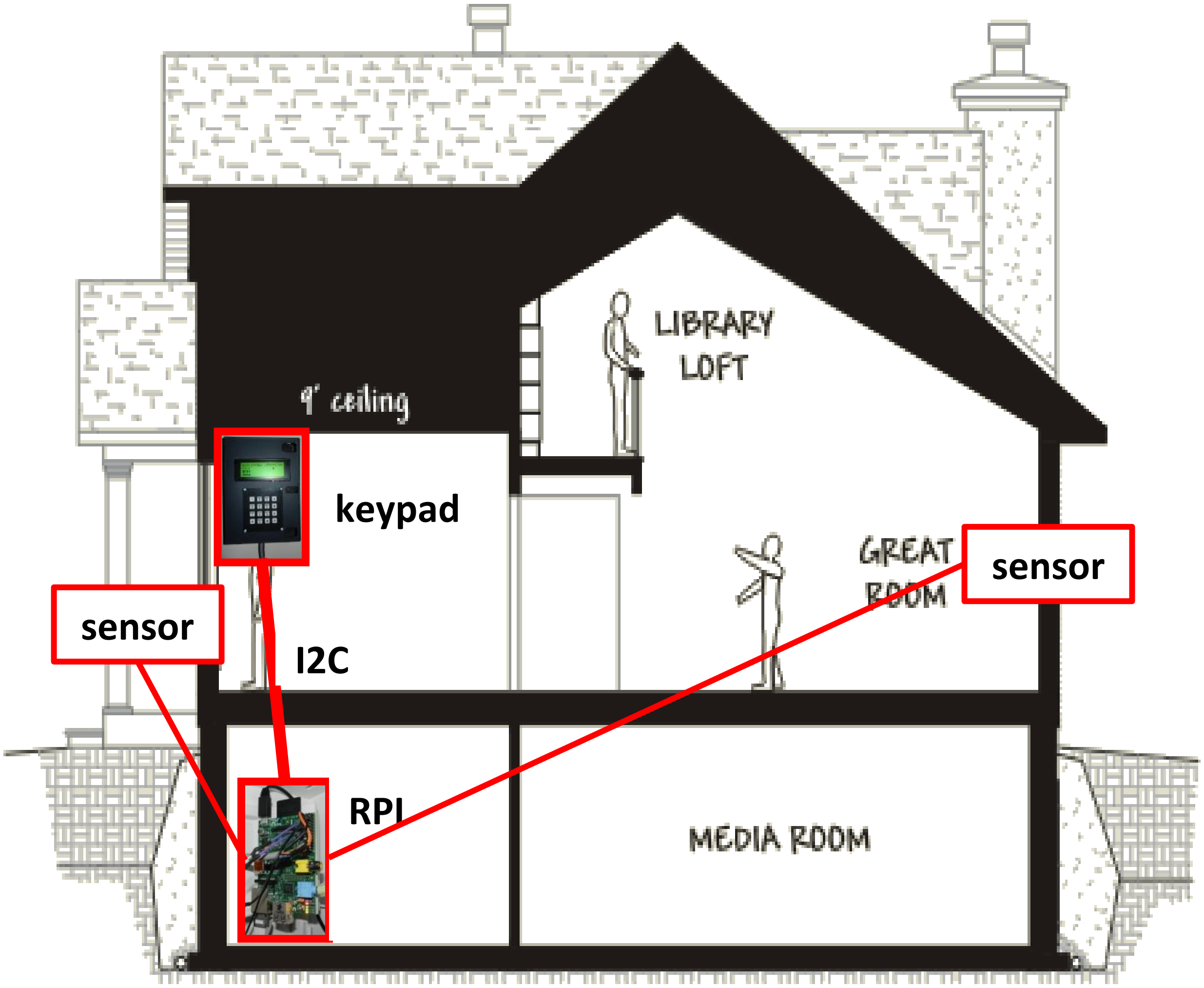
The whole system relies on the GPIO pins of the RaspberryPi to read the different sensors and the I2C bus to communicate with the LCD. The LCD03 also provides the interface to read the keypad. For more information on the LCD03 , refer to the LCD03 technical documentation.

Software installation

Before you go and fix your raspberry pi in the farthest corner of your basement, you may want to plug it in with a monitor, keyboard and mouse so you can do some initial installation and testing. On your first boot up, you will want to enable ssh login as this is how we will access it once it is installed in your basement. At this point you may proceed with the software setup by following the steps in the **installation__.sh** file. Note that it is not an executable shell script but notepad++ will display it in pretty colors when using this extension. (This step will likely change as I plan on packaging the whole system)

Hardware installation

The general layout of the alarm system is as follow:



Shopping list

LCD03 type display.	<20$
3x4 or 4x4 keypad.	>=2$
Raspberry Pi	>=35$
Windows and door sensors	<20$
Wiring	<10$
Wiring connectors (wire to header)	<5$

Installing the sensors is probably the most difficult task as wiring all the sensors may require some time. If your house already has the wiring, this is a walk in the park. Note that for the I2C wire, you will need to use cat5E or cat6 cabling and pair(twisted pair) the SDA with GND as well as SCLK with GND. Depending on the rating of the power supply you use for the Raspberry Pi, you may even use a pair of this cable to power the LCD from the Raspberry Pi. For my prototype, I decided to use a separate power supply just for the LCD unit. It may also be a good idea to have your alarm system connected to a UPS along with your internet modem and router to ensure sms can be sent during a power failure or if power is maliciously cut. If the internet connection is also lost the sound alarm will be the only thing standing between the burglars and your house.

You will have to use your imagination to build the LCD+keypad unit. I used an old DVD box but it's a bit lacking on the rigidity so I lined it with wood inside. There are probably better solutions to this but this does the work for me for now.

Sensors

All the sensors do is to close a connection when the two parts a close enough together otherwise the wires are essentially open loop. Therefore, the GPIOs are setup to pullup and are simply polled to see if the switch is closed or open. This means that you will have to connect one lead of every wire to a ground pin on the Raspberry pi and the other lead to an available GPIO pin. The mapping of pin to sensor is done in software using a convenient configuration file.

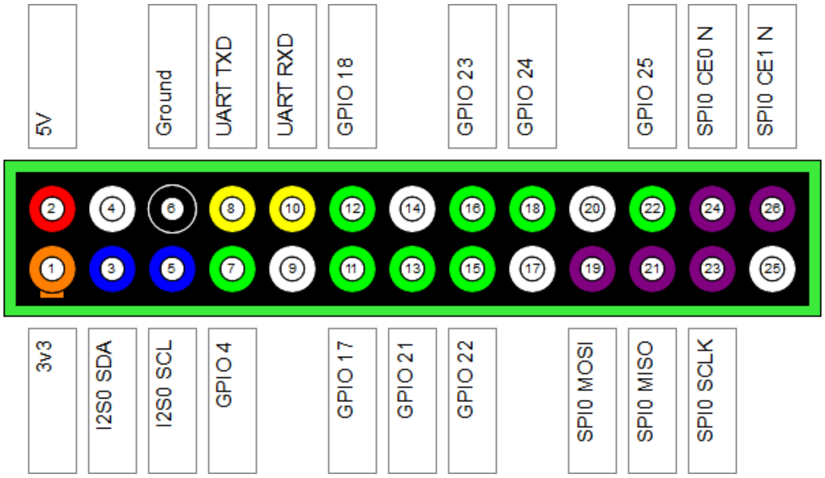


Figure 1 RPi header pinout

Make sure you connect an Ethernet cable to your RPI as well as speakers in the 3.5mm jack if you want sounds. From there, we can keep going with the Alarm system configuration. When you're done it should look something like this:

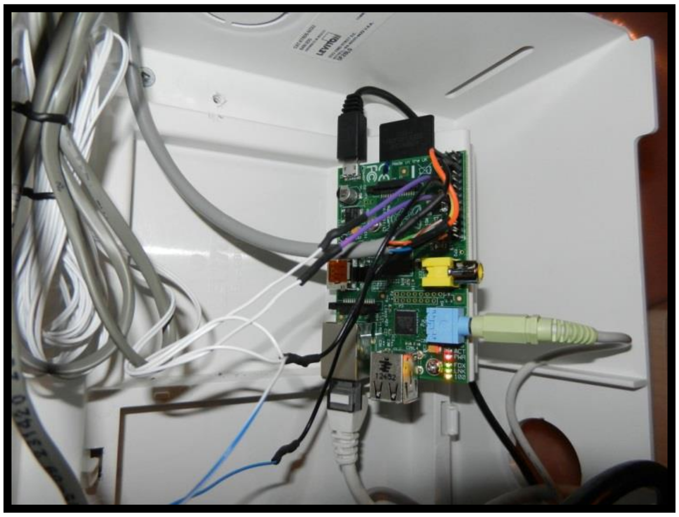


Figure 2 Raspberry pi buried in my basement

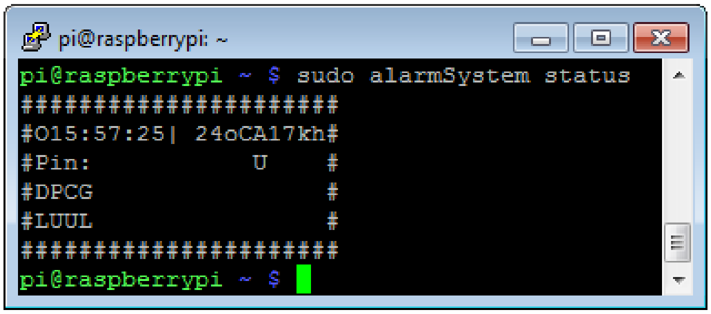
Alarm system configuration

At this point, all your hardware and software is installed and all that remains is to configure your system so that it maps the GPIO pins to the right sensors, sends sms messages through your Google calendar account as well as choose a nip to arm and disarm your system. All of this is done by editing the alarm_config.json file in the project folder. The file should be self-explanatory. Make sure your Google calendar account is set up to allow sms reminders for events as this is the mechanism used to send sms.

The weather is obtained through the web api of wunderground.com. Make sure you obtain a free api key and enter it in the configuration file as well. This will allow you to receive free weather updates that will display on the LCD of your alarm system. Also enter the appropriate zone such as your postal code in the appropriate field.

SSH monitoring

Once your system is operational, you can interact with your system by logging via ssh and using the provided alarmSystem script to either stop, start, restart, arm, and get the status of your system. The status returned is essentially a copy of what is on the LCD.



```
pi@raspberrypi ~ $ sudo alarmSystem status
######################
#O15:57:25| 24oCA17kh#
#Pin:           U    #
#DPCG                #
#LUUL                #
######################
pi@raspberrypi ~ $
```