



**Università degli studi di
Roma Tor Vergata**

Facoltà di Ingegneria

Roma2LUG
Linux User Group

Roma2LUG Incontra

Music On Linux

Speaker
Giulia Cassarà

Speaker
Emanuele Savo

Raspberry Pi

Introduction to the Raspberry Pi 3 Model B Board

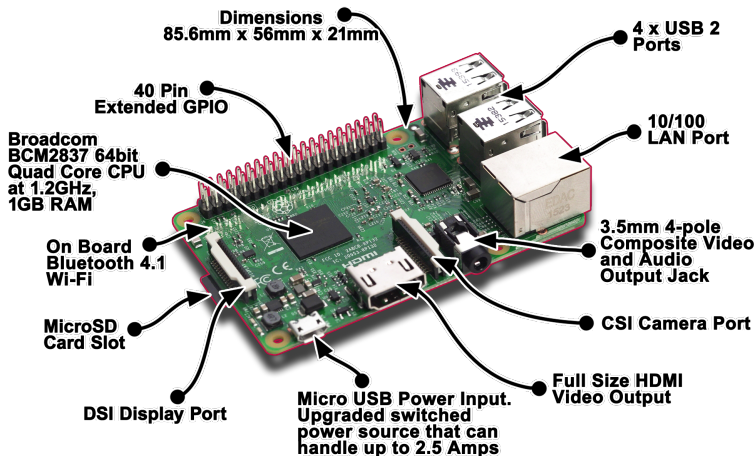


General features

- Born as a MiniPC
- Can reproduce HD movies
- Pi is versatile, you can use it in many ways: web servers, print servers, robot, camera... We have use it as a speaker :)

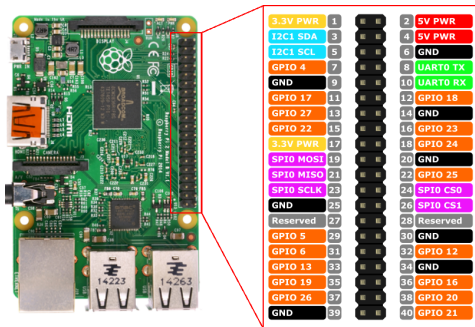
Raspberry Pi

Specs of the Raspberry Pi 3 Model B Board



GPIO (General Purpose Input Output)

GPIO ports mapping of the Raspberry Pi 3 Model B Board



- The main difference between Raspberry Pi and a normal PC are GPIO pins.
- These pins are a physical interface between the Pi and the outside world. Raspberry can take an input from outside (input mode) or the Pi can "generate" something outside (output mode)
- **Be aware!** GPIO pins don't tolerate more than 3.3 V!

Raspbian OS installation

Download and unzip OS



- Download Raspbian OS lite version for the Raspberry Pi

```
$ wget https://downloads.raspberrypi.org/  
raspbian_lite_latest
```

- Unzip Raspbian OS for the Raspberry Pi

```
$ unzip xxxx-xx-xx-raspbian-jessie-lite.zip
```

After Download

Prepare SD card from Linux



- Insert SD card into the PC
- Search for device name of the SD card with this command:

```
$ sudo fdisk -l
```

- Search for info about your SD card. *Warning, be careful!*

```
Disk /dev/mmcblk0: 14,5 GiB, 15523119104 bytes, 30318592 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytesa
Disklabel type: dos
Disk identifier: 0x6f92008e
```

- Replace mmcblk0 with device name of your SD

```
$ sudo dd \
  if=/xxxx-xx-xx-raspbian-jessie-lite.img \
  of=/dev/mmcblk0
```

Boot

Boot the system and update packages



- Connect ethernet cable to the Raspberry Pi
- Connect HDMI cable to the Raspberry Pi
- Connect micro USB power cable to the Raspberry Pi
- Waiting for complete boot...
- Login
 - user: pi
 - password: raspberry
- Repository update:

```
$ sudo apt-get update
$ sudo apt-get dist-upgrade -y
$ sudo apt-get install rpi-update -y
```

Configuration

Expand filesystem and configure your raspberry



- Config Raspbian OS with this tool

```
$ sudo raspi-config
```

- Expand Filesystem
- Internationalisation Options
 - Change Locale
 - Change Timezone
 - Change Keyboard Layout
 - Change wifi Country

```
$ sudo reboot
```

- Update Raspberry Pi firmware

```
$ sudo rpi-update
```

```
$ sudo reboot
```


WiringPi and GIT

Install necessary software



- Install library for gpio and other tools

```
$ sudo apt-get install -y wiringpi git vim
```

- Download the scripts

```
$ git clone https://github.com/Roma2Lug-  
Projects/MusicOnLinux.git
```

- Open the script

```
$ cd MusicOnLinux/Scripts  
$ vim keyboard.sh  
$ vim smario.sh
```

Final steps

Script's permission and execution



- Give execute permission

```
$ chmod +x keyboard.sh  
$ chmod +x smario.sh
```

- Execute the scripts!

```
$ ./keyboard.sh  
$ ./smario.sh
```

Tone function



```
#!/bin/bash
tone () {
    local note="$1"
    local duration="$2"
    if test "$note" -eq 0; then
        gpio -g mode 18 in
    else
        local frequency=$(python -c "print '{0:.0f}'
            .format(600000.0/440.0/2**((($note-69)
            /12.0)))")
        gpio -g mode 18 pwm
        gpio pwmr "$(( frequency ))"
        gpio -g pwm 18 "$(( frequency/2 ))"
        gpio pwm-ms
        sleep $duration
        tone 0
    fi
}
```

Tone function in details (1)



```
tone () {  
    local note="$1"  
    local duration="$2"  
    if test "$note" -eq 0; then  
        gpio -g mode 18 in  
        ...  
    fi  
}
```

- First parameter: note.
- Second parameter: duration of the note.
- Test if the note is 0 then put the GPIO in input mode, so the speaker doesn't make any sound.

Tone function in details (2)



```
...  
else  
    local frequency=$(python -c "print '{0:.0f  
        }'.format(600000.0/440.0/2**(($note-69)  
        /12.0))")  
...
```

- We use the formula below to obtain the frequency of the speaker

$$K \cdot \frac{440}{2^{\frac{X-69}{12}}}$$

- $K = \frac{19.2MHz}{32} = 600kHz$ is the base frequency on which the notes are calculated. It is dependent to the hardware of Pi.
- The so called *twelfth root of two* or $\sqrt[12]{2}$ is an algebraic irrational number. It is most important in music theory, where it represents the frequency ratio of a semitone in twelve-tone equal temperament.
- X is the range of the note, encoded in ASCII.

Tone function in details (3)



```
...
gpio -g mode 18 pwm
gpio pwmr "$(( frequency ))"
gpio -g pwm 18 "$(( frequency/2 ))"
gpio pwm-ms
sleep $duration
tone 0
fi
}
```

- These lines of code tell to Pi to give HIGH signal through GPIO port 18 at a rate *frequency* to the connected speaker with a modulation algorithm called Pulse Width Modulation (PWM).
- The speaker beeps the "note" for a time "duration".
- Finally the last command mute the sound by recalling the tone function with 0. Without this line the speaker will sound indefinitely(!!!).



[DEMO]





Grazie per l'attenzione!

