

CURSO DE RASPBERRY PI

Introducción

Prof. Edwin Yllanes Cucho
e.yllanescucho@gmail.com



8 de febrero de 2015



- 1 Raspberry Pi Model B+
- 2 Fuente para Raspberry 5V-2A
- 3 Caja de acrilico para Raspberry
- 4 Memoria SD de 4GB
- 5 WIFI para Raspberry
- 6 1 Protoboard
- 7 cables de conexión
- 8 leds
- 9 resistencias
- 10 botones
- 11 ultrasonido

SEMANA 1

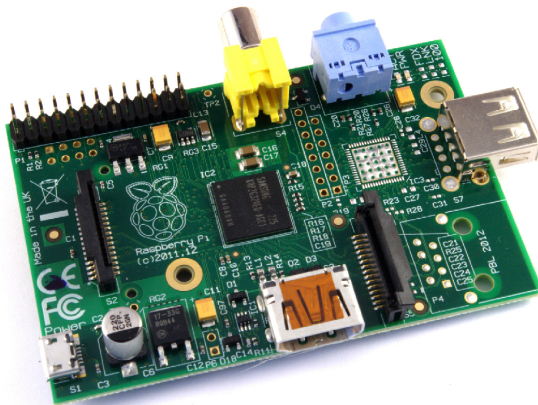


Figura: Model A

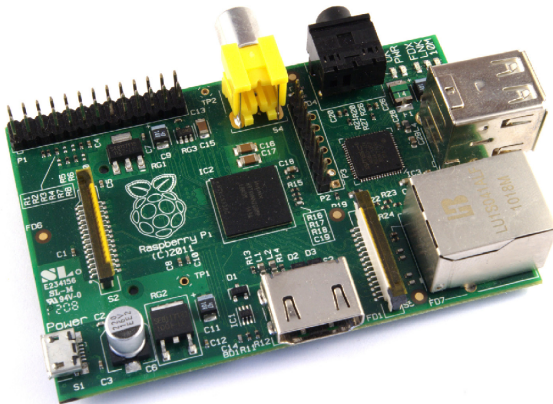


Figura: Model B

Algunos de los miles de comandos que existen en el mundo GNU-linux...

Comando	función	Uso
man	manual	\$ man man
ls	listing	\$ ls /home/pi
cd	change directory	\$ cd ..
mv	move	\$ mv carpeta1 carpeta2
rm	remove	\$ rm archivo.txt
rmdir	remove directory	\$ rmdir carpeta
mkdir	make directory	\$ mkdir carpeta
cp	copy	\$ cp archivo1.txt archivo2.txt
find	find	\$ find archivo.txt
locate	locate	\$ locate archivo.txt

Cuadro: Comandos básicos

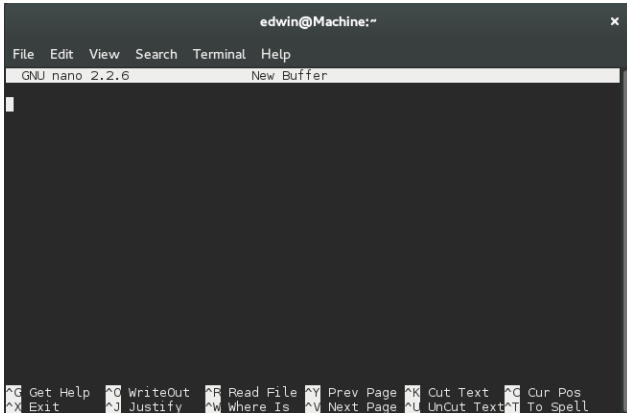
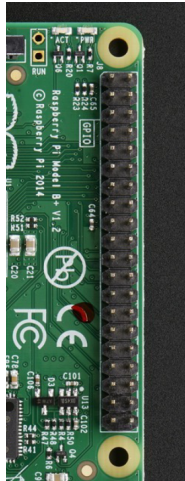


Figura: Editor Nano



(a)

Pi Model B+			
3V3 Power	1	2	SV Power
GPIO2 SDA1 I2C	3	4	SV Power
GPIO3 SCL1 I2C	5	6	Ground
GPIO4	7	8	GPIO14 UART1_TXD
Ground	9	10	GPIO15 UART1_RXD
GPIO17	11	12	GPIO18 PWM_SDA
GPIO27	13	14	Ground
GPIO22	15	16	GPIO23
3V3 Power	17	18	GPIO24
GPIO10 SPD_B001	19	20	Ground
GPIO9 SPD_B005	21	22	GPIO25
GPIO11 SPD_SCLX	23	24	GPIO8 SPD_C00_N
Ground	25	26	GPIO7 SPD_C01_N
ID_SD (DC to 5VPP0M)	27	28	ID_SC (DC to 5VPP0M)
GPIO5	29	30	Ground
GPIO6	31	32	GPIO12
GPIO13	33	34	Ground
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
Ground	39	40	GPIO21
Pi Model B+			

www.raspberrypi-sgp.co.uk

(b)

Figura: a)Gpio b) Pinout

Bus	Nombre	Posición	MODE BCM
I ² C	SDA	Pin3	GPIO2
	SCL	Pin5	GPIO3
	ID_SD	Pin27	ID_SD
SPI	ID_SC	Pin28	ID_SC
	MOSI	pin19	GPIO10
	MISO	pin21	GPIO9
	SCLK	pin21	GPIO11
	CE0	pin24	GPIO8
UART	CE1	pin26	GPIO7
	TX	pin8	GPIO14
	RX	pin10	GPIO15

SEMANA 2

- Instalar Win32DiskImager

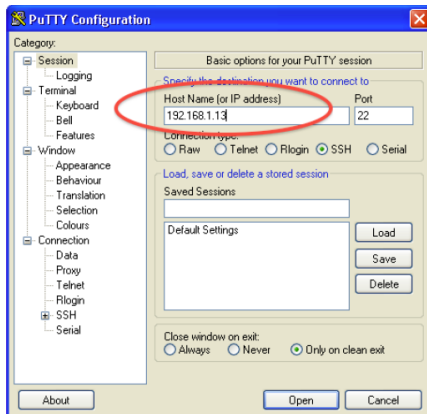
<http://sourceforge.net/projects/win32diskimager/files/latest/download>

- Instalar putty

<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

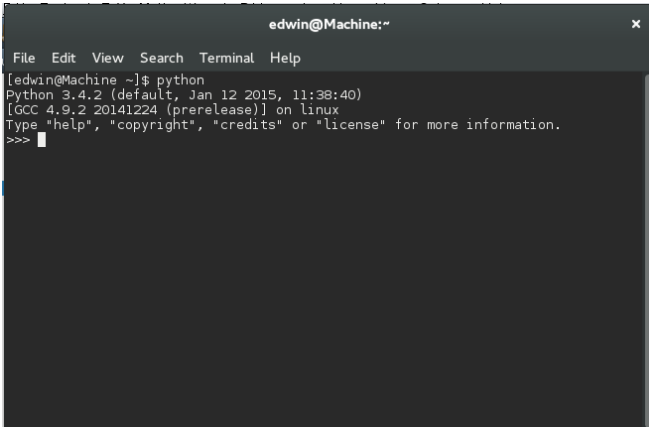


Abrir el programa putty y colocar la configuración siguiente:



```
$ sudo apt-get install python-dev  
$ sudo apt-get install python-setuptools  
$ sudo easy_install -U distribute  
$ sudo apt-get install python-pip  
$ sudo pip install rpi.gpio
```

\$ python

A terminal window titled "edwin@Machine:~" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal output shows the command "python" being executed, resulting in the following text: "Python 3.4.2 (default, Jan 12 2015, 11:38:40)", "[GCC 4.9.2 20141224 (prerelease)] on linux", and "Type 'help', 'copyright', 'credits' or 'license' for more information." followed by a prompt ">>>" and a cursor.

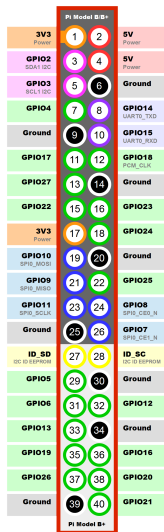
```
edwin@Machine:~  
File Edit View Search Terminal Help  
[edwin@Machine ~]$ python  
Python 3.4.2 (default, Jan 12 2015, 11:38:40)  
[GCC 4.9.2 20141224 (prerelease)] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> █
```



\$ print("Hello World!!!")

```
edwin@Machine:~  
File Edit View Search Terminal Help  
[edwin@Machine ~]$ python  
Python 3.4.2 (default, Jan 12 2015, 11:38:40)  
[GCC 4.9.2 20141224 (prerelease)] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> print("Hello World!!!")  
Hello World!!!  
>>> █
```





www.raspberrypi-spy.co.uk

Figura: Pinout B+



Descargar el código fuente desde la siguiente página:
<https://github.com/eyllanesc/Raspberry-Course>

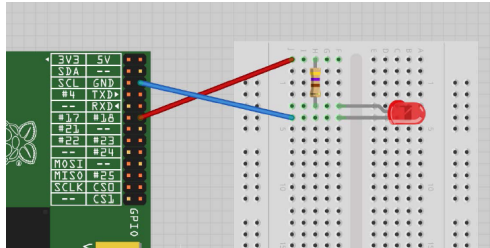


Figura: Circuito para el encendido de Led

```
import RPi.GPIO as GPIO
import time

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(18, GPIO.OUT)

state = True

while True:
    GPIO.output(18, True)
    time.sleep(1)
    GPIO.output(18, False)
    time.sleep(1)
```

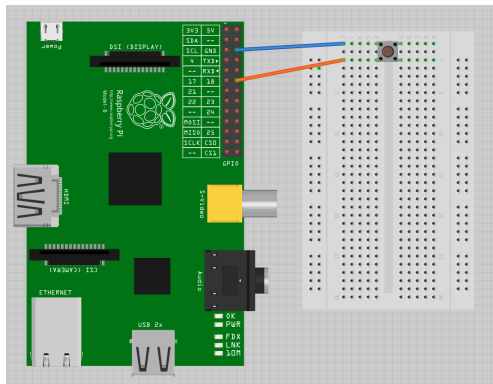


Figura: Circuito para el uso de pulsadores

```
import RPi.GPIO as GPIO #import GPIO library
import time #import time for managed time

GPIO.setmode(GPIO.BCM)

GPIO.setup(18, GPIO.IN, pull_up_down=GPIO.PUD_UP) #GPIO 18 como entrada

while True:
    input_state = GPIO.input(18) #leemos la entrada
    if input_state == False: #si esta en nivel bajo
        print('Button Pressed') #imprime esto
    time.sleep(0.2) #tiempo muerto
```

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
switch_pin = 18
led_pin = 23
GPIO.setup(switch_pin, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(led_pin, GPIO.OUT)
led_state = False
old_input_state = True # pulled-up
while True:
    new_input_state = GPIO.input(switch_pin)
    if new_input_state == False and old_input_state == True:
        led_state = not led_state
    old_input_state = new_input_state
    GPIO.output(led_pin, led_state)
```

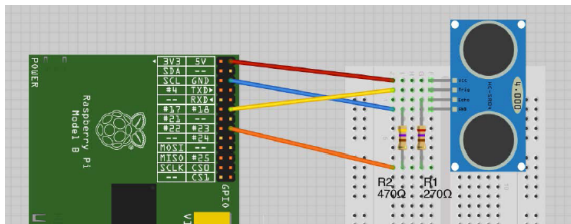


Figura: Conexión


```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
TRIG = 23
ECHO = 24
print "Distance Measurement In Progress"
GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)
GPIO.output(TRIG, False)
print "Waiting For Sensor To Settle"
time.sleep(2)
GPIO.output(TRIG, True)
time.sleep(0.00001)
GPIO.output(TRIG, False)
while GPIO.input(ECHO)==0:
    pulse_start = time.time()
while GPIO.input(ECHO)==1:
    pulse_end = time.time()
pulse_duration = pulse_end - pulse_start
distance = pulse_duration * 17150
distance = round(distance, 2)
print "Distance:",distance,"cm"
GPIO.cleanup()
```



SEMANA 3

```
$ sudo apt-get install python-serial  
$ sudo usermod -a -G tty pi  
$ sudo usermod -a -G dialout pi
```

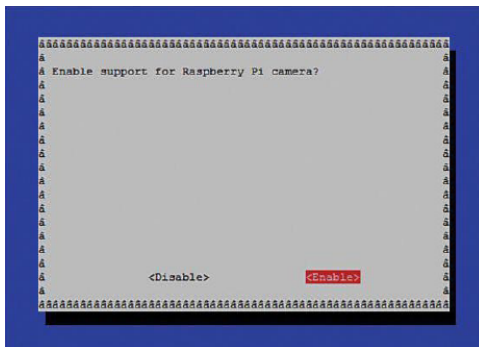


SEMANA 4

Instalación:

\$ sudo apt-get update && sudo apt-get upgrade

\$ sudo raspi-config



```
$ sudo apt-get install python-opencv python-scipy python-numpy python-pip
```

SEMANA 5

\$ sudo apt-get update

\$ sudo apt-get install apache2 php5 php5-mysql mysql-server







SEMANA 6

A high-level language, primarily intended for numerical computations.



```
$ sudo apt-get install octave  
$ octave
```







Gracias
Por su
Atención!!!

Consultas a:
e.yllanescucho@gmail.com