



CONTROL DEL PUERTO GPIO DE LA RASPBERRY

PYTHON PARA EL CONTROL DEL GPIO

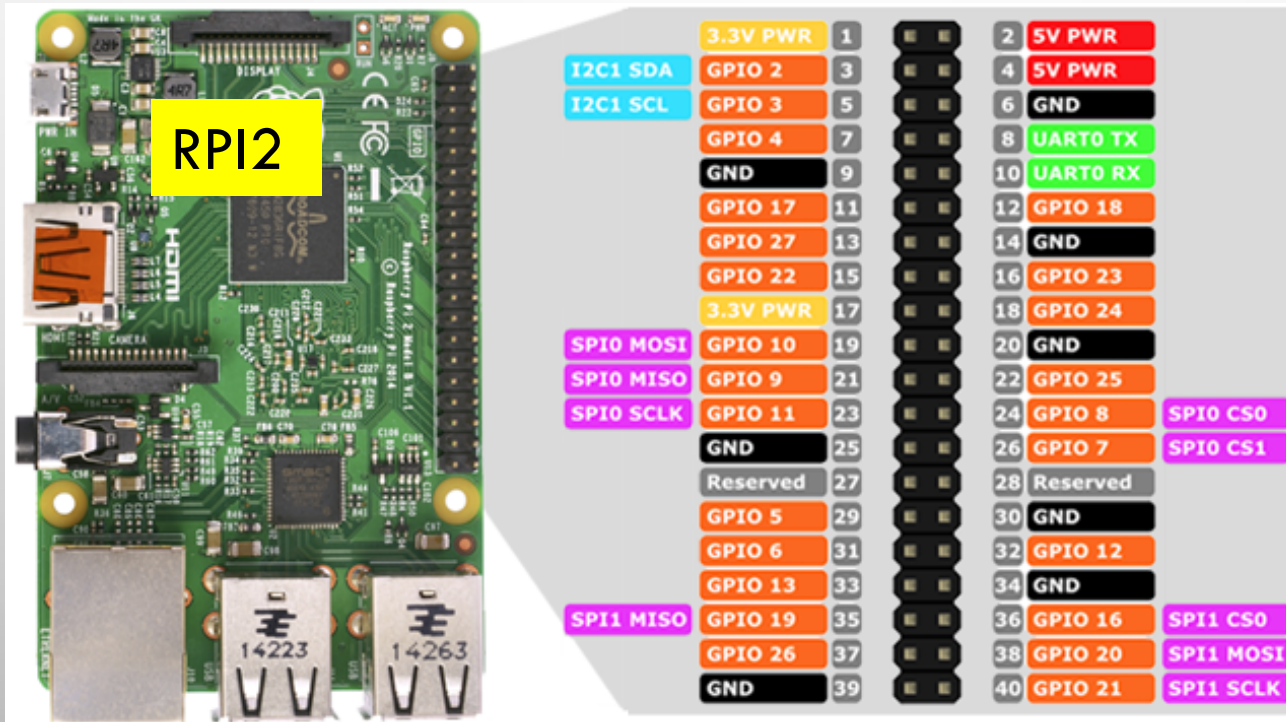
presentado por:

Francisco Javier Ferrández Pastor

CONTROL DEL PUERTO GPIO DE LA RASPBERRY

INTRODUCCIÓN

General Purpose Input Output (GPIO) es un sistema de entrada y salida de propósito general, es decir, consta de una serie de pines o conexiones que se pueden usar como entradas o salidas para múltiples usos. Estos pines están incluidos en todos los modelos de Raspberry Pi aunque con diferencias.



CONTROL DEL PUERTO GPIO DE LA RASPBERRY





















INTRODUCCIÓN

De los pines GPIO disponibles, hay una serie de pines con capacidad de PWM (para control de potencia en pin12 GPIO18).

Sin embargo no se dispone de ningún convertidor de analógico a digital. Esto quiere decir que para medir valores de sensores analógicos necesitaremos utilizar un convertidor (conversor A/D o Arduino).

ADVERTENCIA

Cuando se utilizan los pines de *GPIO* hay que poner mucho cuidado para no dañar la propia Raspberry Pi. Es muy importante comprobar los niveles de tensión y la corriente solicitada. Los pines de GPIO pueden generar y consumir tensiones compatibles con los circuitos de 3.3V. No conectar nunca componentes de 5V a los IO, o se puede quemar la electrónica.

BOARD	GPIO		GPIO	BOARD
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I ² C)		DC Power 5v	04
05	GPIO03 (SCL1 , I ² C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I ² C ID EEPROM)		(I ² C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37			GPIO20	38
39			GPIO21	40

RPI3

CONTROL DEL PUERTO GPIO DE LA RASPBERRY

PROGRAMACIÓN EN PYTHON

Python es un lenguaje interpretado que ofrece recursos para acceder al hardware de la Raspberry con librerías documentadas, fáciles de utilizar. Por defecto, Python viene instalado en el sistema operativo de Raspbian para Raspberry Pi.

EJEMPLO 1:

- Parpadeo continuo de un led con una frecuencia de 2 segundos

Se debe crear fichero con extensión py desde un editor, o bien desde el entorno de programación Python 3 (IDLE) en la interfaz gráfica de la Raspberry Pi y a continuación crearemos un nuevo fichero para escribir el código de programación.

La primera línea a escribir corresponde a la importación de la librería de los pines GPIO, es decir, con esta librería podemos utilizar las funciones implementadas para controlar los pines de nuestra Raspberry Pi.

```
import RPi.GPIO as GPIO
```

A continuación indicamos el modo con el cual nos vamos a dirigir al pin a utilizar ya que podemos dirigirnos al pin por el número en la placa (`GPIO.BOARD`), o por el canal al cual está conectado en el Chip Broadcom (`GPIO.BCM`). En nuestro caso, vamos a utilizar la primera opción, es decir, indicaremos a Python que vamos a utilizar el pin situado en su posición de la placa, 32 en este caso. La siguiente instrucción será la inicialización del pin, es decir, los pines pueden ser utilizados como entradas o salidas. Por último, solamente nos quedará activar (`True`) o desactivar (`False`) el pin seleccionado. Para ello, se va a situar el código dentro de un bucle infinito (recuerda la indentación del código).

EJEMPLO 1:

- Parpadeo continuo de un led con una frecuencia de 2 segundos

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(32, GPIO.OUT)

while True:
    try:
        GPIO.output(32, True)
        time.sleep(2)
        GPIO.output(32, False)
        time.sleep(2)
    except KeyboardInterrupt:
        print 'salida\n'
        GPIO.cleanup() #devuelve los pines a su estado inicial
```

Python es un lenguaje interpretado que ofrece recursos para acceder al hardware de la Raspberry con librerías documentadas, fáciles de utilizar. Por defecto, Python viene instalado en el sistema operativo de Raspbian para Raspberry Pi.

EJEMPLO 2:

- Pulsador para encender/apagar led

El código no tiene mayores complicaciones, sólo para mostrar cómo se programa una Entrada

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(3, GPIO.IN)
GPIO.setup(7, GPIO.OUT)

while True:
    try:
        if GPIO.input(3):
            GPIO.output(7, False)
        else:
            GPIO.output(7, True)
    except KeyboardInterrupt:
        print 'salida\n'
        GPIO.cleanup() #devuelve los pines a su estado inicial
```




INDUSTRIA 4.0

DEFINICIÓN Y CARACTERÍSTICAS

The [Adafruit Python DHT Sensor library](https://github.com/adafruit/Adafruit_Python_DHT) makes interfacing the DHT22 easy.

Please note that you should not use pip for installation because there are platform issues.

Use git clone to install it from your home directory in a terminal window as follows.

You can target Python 3 instead of Python 2 by changing python in the last line to python3.

```
git clone https://github.com/adafruit/Adafruit_Python_DHT.git
cd Adafruit_Python_DHT
sudo apt-get update
sudo apt-get install build-essential python-dev
sudo python setup.py install
```

EJEMPLO 2:

- Sensor de Temperatura y Humedad (DHT22) conectado al GPIO

RPI tiene un conjunto de sensores/actuadores que pueden utilizarse directamente desde el puerto GPIO. Normalmente se descargan las librerías necesarias y se interactúa con programas de ejemplo ya implementados

```
>> git clone  
https://github.com/adafruit/Adafruit_Python_DHT.git  
>>cd Adafruit_Python_DHT  
>>sudo apt-get update  
>>sudo apt-get install build-essential python-dev  
>>sudo python setup.py install
```

```
>> sudo reboot
```


CONTROL DEL PUERTO GPIO DE LA RASPBERRY

PROGRAMACIÓN EN PYTHON

EJEMPLO 2:

- Sensor de Temperatura y Humedad (DHT22) conectado al GPIO

Programa en Python que captura la T y la H utilizando librería de ADAFRUIT y sensor DHT22 conectado al GPIO4

