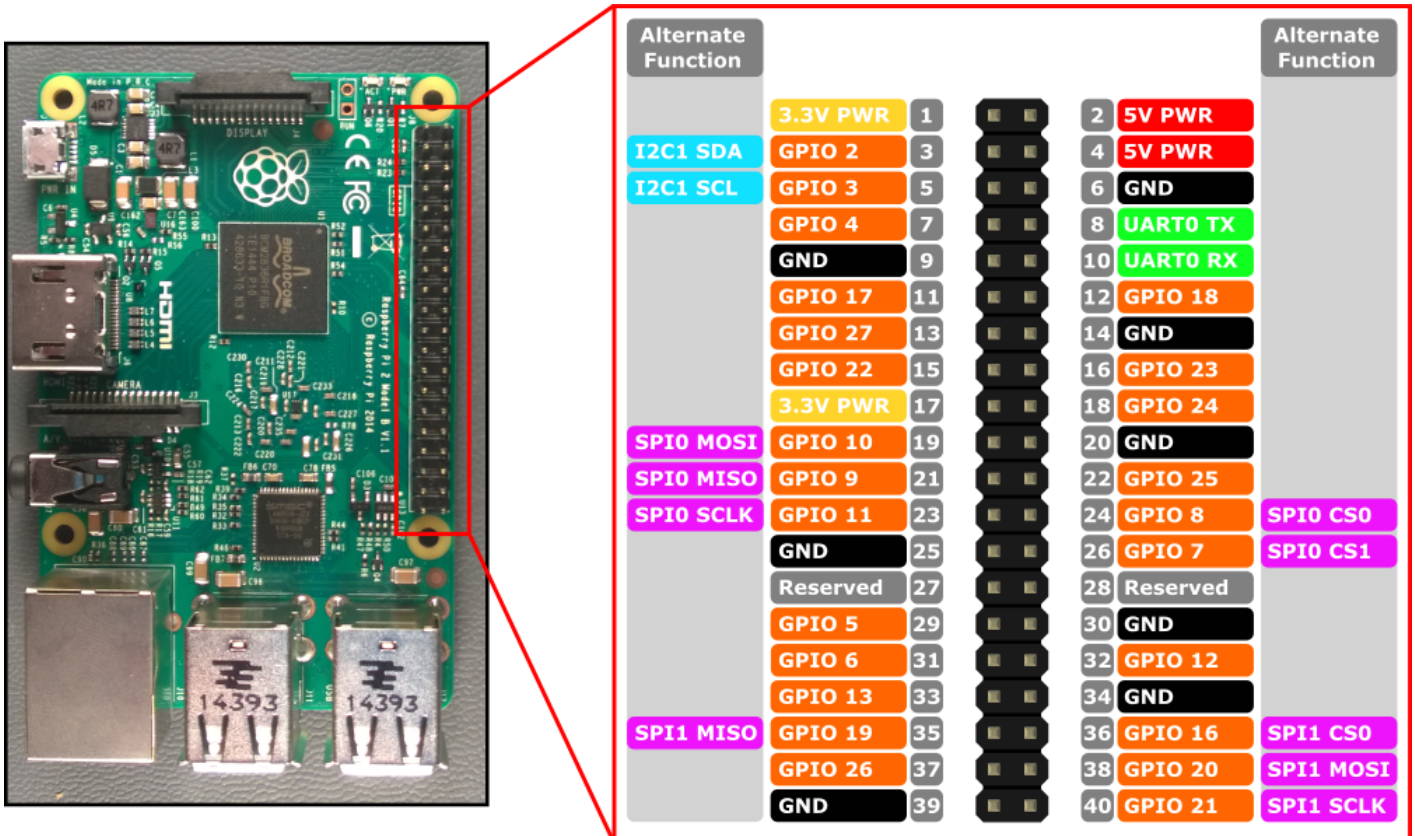


# 1.6 GPIO

Vamos a recordar lo que vimos [aquí](#), dos cosas:

- Estos son los pines GPIO con la numeración BCM:



- Están diseñados para 3.3V sólo proporcionan 3mA cada pin luego NO conectes directamente componentes de 5V ni que consuman más corriente o de lo contrario ESTROPEARÁS LA RASPBERRY DE FORMA IRREVERSIBLE, o sea, directamente sólo LEDs con una resistencia de mínimo 1.1K tal [y como vimos aquí](#), todo lo demás a través de chips drivers, por eso utilizamos este kit comercial.

En nuestro rover la conexión es la siguiente:

BCM	wPi	RPI1			wPi	BCM
		3.3V	1	2	5V	
2	8	P SDA	3	4	5V	
3	9	P SCL	5	6	GND	
4	7	P4	7	8	P TX	15
		GND	9	10	P RX	16
17	0	P17	11	12	P18	1
27	2	P27	13	14	GND	18
22	3	P22	15	16	P23	4
		3.3V	17	18	P24	5
10	12	P MOSI	19	20	GND	
9	13	P MISO	21	22	P25	6
11	14	P SCK	23	24	P CE0	10
		GND	25	26	P CE1	11
		ID SD	27	28	ID SC	
5	21	P5	29	30	GND	
6	22	P6	31	32	P12	26
13	23	P13	33	34	GND	
19	24	P19	35	36	P16	27
26	25	P26	37	38	P20	28
		GND	39	40	P21	29
						21

## Librería RPi.GPIO

Necesitamos una librería GPIO que Raspbian lo tiene por defecto, pero por si acaso ejecuta estas instrucciones:

```
sudo apt-get install python-dev
sudo apt-get install python-rpi.gpio
```

Normalmente te dirá que las tienes instaladas en su última versión.

Para utilizar la librería, simplemente tenemos que poner esta instrucción:

**import RPi.GPIO as GPIO**

# GPIO.setmode y GPI.setup

Hay dos formas de utilizar la numeración de las GPIO, respetando la misma numeración que los pines de la placa, entonces la instrucción que tenemos que poner en nuestros programas es `GPIO.setmode(GPIO.BOARD)` o utilización de la numeración `BCM GPIO.setmode(GPIO.BCM)` nosotros elegiremos esta última por ser más sencilla, aunque tiene la desventaja de que si cambian en el futuro la numeraciones en los BCM nuestro programa no servirá.

Una vez definido qué numeración usamos, tenemos que especificar en nuestro programa si tal GPIO es entrada o salida, por ejemplo la siguiente instrucción define el GPIO número 4 como entrada (7 en numeración BOARD) `GPIO.setup(4, GPIO.IN)`

## ¿Qué es lo que hay conectado en cada puerto GPIO de este rover?

Pues aquí lo tienes, cada uno lo vamos a ver a lo largo de este curso :

Interfaces	Puertos GPIO de la Raspberry Pi nomenclatura BCM
IN1 motores	12
IN2 motores	13
ENA motores	6
IN3 motores	20
IN4 motores	21
ENB motores	26
Sensor velocidad derecha	7
Sensor velocidad izquierda	8
Sensor IR obstáculos derecha	16
Sensor IR obstáculos izquierda	19
Sensor IR mando distancia	18
Siguelíneas CS	5
Siguelíneas Clock	25



Interfaces	Puertos GPIO de la Raspberry Pi nomenclatura BCM
Siguientes Address	24
Siguientes DataOut	23
Servo brazo eje X	22
Servo brazo eje Z	27

## Ejemplo de utilización de la librería RPI.GPIO

El siguiente ejemplo enciende un LED puesto en el GPIO 4, durante 2 segundos

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setup(4, GPIO.OUT) ## GPIO 4 como salida
GPIO.output(4,True) ##encendemos
time.sleep(2)          ## espera 2 segundos
GPIO.output(4,False)  ##APAGAMOS
```

**No lo hagas** pues en el 4 de este rover no hay puesto nada, mira el [plano esquemático](#) y en P4 (pin 7 real del conector RPI1) no hay nada conectado.

BCM	wPi	RPI1				wPi	BCM
		3.3V	1	2	5V		
2	8	P SDA	3	4	5V		
3	9	P SCL	5	6	GND		
4	7	P4	7	8	P TX	15	14
		GND	9	10	P RX	16	15
17	0	P17	11	12	P18	1	18
27	2	P27	13	14	GND		
22	3	P22	15	16	P23	4	23
		3.3V	17	18	P24	5	24
10	12	P MOSI	19	20	GND		
9	13	P MISO	21	22	P25	6	25
11	14	P SCK	23	24	P CE0	10	8
		GND	25	26	P CE1	11	7
		ID SD	27	28	ID SC		
5	21	P5	29	30	GND		
6	22	P6	31	32	P12	26	12
13	23	P13	33	34	GND		
19	24	P19	35	36	P16	27	16
26	25	P26	37	38	P20	28	20
		GND	39	40	P21	29	21

Revision #3

Created 21 March 2022 11:57:08 by Equipo CATEDU

Updated 28 March 2022 09:37:33 by Equipo CATEDU