2.1 Motores

Vamos a empezar por la parte importante de un rover:



NASA/JPL-Caltech

Los motores estan gobernados con los siguientes GPIO

Interfaces	Puertos GPIO de la Raspberry Pi
IN1	P12
IN2	P13
ENA	P6
IN3	P20
IN4	P21
ENB	P26

Luego una de las primeras líneas que hay que poner en nuestros programas es traducir esos números a letras para que sea más facil utilizarlos en el código, y definir esos pines como pines de salida que van a gobernar a los motores:

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

Copyright 2026

```
IN1=12;IN2=13;ENA=6;IN3=20;IN4=21;ENB=26

GPIO.setmode(GPIO.BCM);GPIO.setwarnings(False)
GPIO.setup(IN1,GPIO.OUT);GPIO.setup(IN2,GPIO.OUT);GPIO.setup(IN3,GPIO.OUT);GPIO.setup(IN4,GPIO
.OUT)
GPIO.setup(ENA,GPIO.OUT);GPIO.setup(ENB,GPIO.OUT)
```

¿Qué significan IN1 IN2 IN3 IN4 ?

IN1	IN2	IN3	IN4	Descripción
1	0	0	1	Motores hacia delante
0	1	1	0	Motores hacia atrás
0	0	0	1	Giro derecha
1	0	0	0	Giro izquierda
0	0	0	0	Stop

¿Y qué significa ENA ENB?

ENA y ENB es la velocidad de los motores A y B respectivamente.

Su valor tiene que ser analógico pero los GPIO son digitales, así que tienen que ser señales PWM.

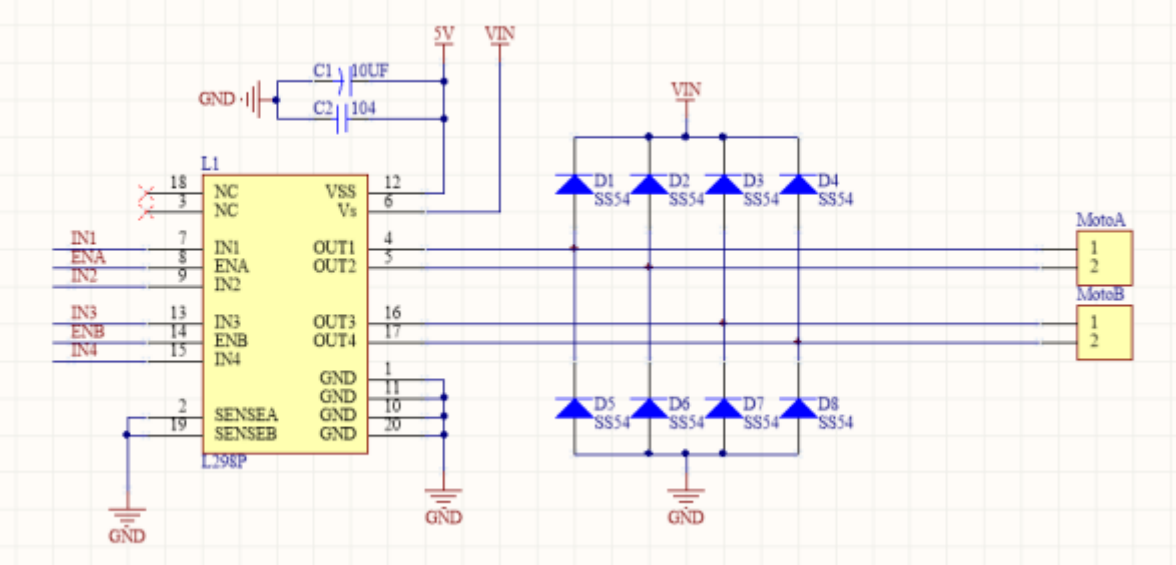
Si vamos a poner una frecuencia de 500Hz y una velocidad media, el código que tenemos que poner al principio de nuestro programa es:

```
PWMA = GPIO.PWM(ENA,500);PWMB = GPIO.PWM(ENB,500)
PWMA.start(50);PWMB.start(50)
```

Bueno, pero ... ¿Cómo son las conexiones?

Copyright 2026

En el AlphaBot están conectados los pines IN1 IN2 IN3 IN4 ENA ENB en los pines de un chip L298P que hace de driver a los motores (nunca conectes un motor a un GPIO de la Raspberry ya lo sabes)



Vale... ¿Y cómo se utiliza?

Podemos definir en nuestros programas unas funciones para simplificar código para utilizar los motores hacia delante, detrás y giros:

```
def FORDWARD():

GPIO.output(IN1,GPIO.HIGH);GPIO.output(IN2,GPIO.LOW);GPIO.output(IN3,GPIO.LOW);GPIO.output(IN4
,GPIO.HIGH)
def BACKWARD():

GPIO.output(IN1,GPIO.LOW);GPIO.output(IN2,GPIO.HIGH);GPIO.output(IN3,GPIO.HIGH);GPIO.output(IN
4,GPIO.LOW)
def LEFT():

GPIO.output(IN1,GPIO.LOW);GPIO.output(IN2,GPIO.LOW);GPIO.output(IN3,GPIO.LOW);GPIO.output(IN4,
GPIO.HIGH)
def RIGHT():

GPIO.output(IN1,GPIO.HIGH);GPIO.output(IN2,GPIO.LOW);GPIO.output(IN3,GPIO.LOW);GPIO.output(IN4
,GPIO.LOW)
```

Copyright 2026

```
def STOP():

GPIO.output(IN1,GPIO.LOW);GPIO.output(IN2,GPIO.LOW);GPIO.output(IN3,GPIO.LOW);GPIO.output(IN4,
GPIO.LOW)
```

Revision #2
Created 2022-03-21 11:57:12 CET by Equipo CATEDU
Updated 2022-03-22 12:04:19 CET by Equipo CATEDU

Copyright 2026