Control velocidad

- 3 Control velocidad
- 3.1 ¿Cómo funciona?
- 3.2 Prueba velocidad
- 3.3 Variables.py
- 3.4 MOVIMIENTOSPASO.py
- 3.5 Movimientos con paso

Copyright 2026

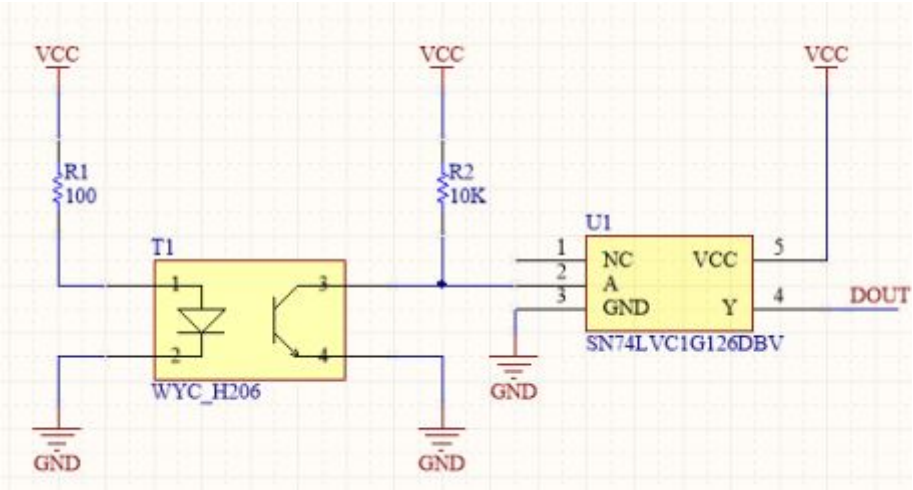
3 Control velocidad



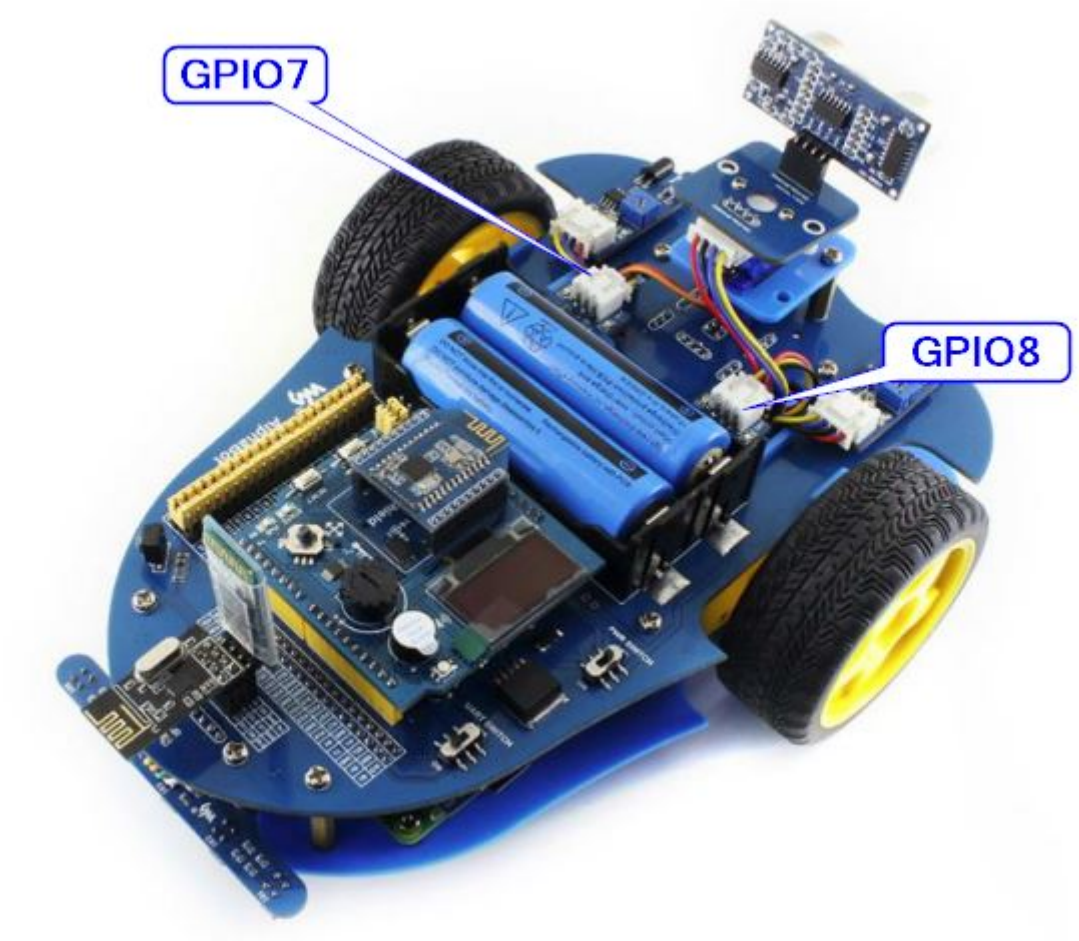
Copyright 2026

3.1 ¿Cómo funciona?

Las ruedas tienen un disco con agujeros, una parte es un diodo emisor de IR y el otro es un sensor fotoeléctrico tipo WYC-H206 que detecta los agujeros:

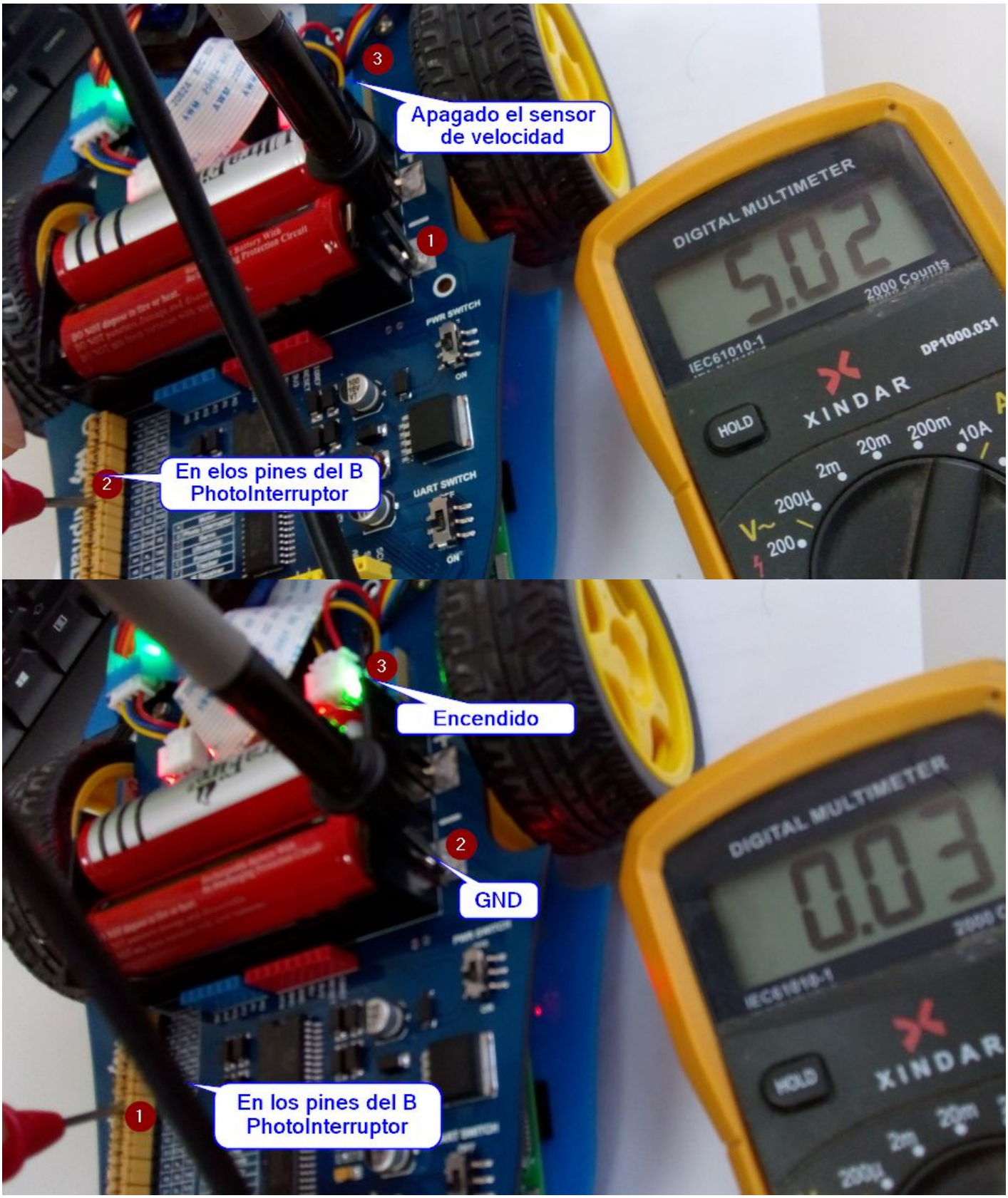


Están conectados a los siguientes GPIO: * Motor derecha GPIO7 * Motor izquierda GPIO8



Si te fijas en el esquema anterior, las resistencias, están con la configuración **PULL-UP** (aquí para saber + en el curso Arduino) ¿qué significa esto? pues que van al revés, cuando el circuito está encendido, o sea detecta agujero, estado ON transmite un 0 lógico, y al revés, cuando está apagado OFF transmite un 1 lógico, lo puedes ver mejor en estas fotografías:

Copyright 2026



Copyright 2026

3.2 Prueba velocidad

En el siguiente vídeo vemos como cuando el sensor está encendido, el programa detecta un 0 y si el sensor está apagado, el programa detecta un 1:

https://www.youtube.com/embed/anWM7Y54u98

Fichero Pruebasensorvelocidad.py

El programa es el siguiente:

```
import RPi.GPIO as GPIO

DataMotorR = 7
DataMotorL = 8

GPIO.setmode(GPIO.BCM)

GPIO.setup(DataMotorR,GPIO.IN)
GPIO.setup(DataMotorL,GPIO.IN)



for i in range(100000):
    print('\nMotor derecha   :',GPIO.input(DataMotorR))
    print('\nMotor izquierda :',GPIO.input(DataMotorL))
```

Segundo test de contador

En el segundo vídeo vídeo vemos como un simple contador puede detectar el paso del 1 al 0:

https://www.youtube.com/embed/Xm1aZvEp9fE

El programa es el siguiente:

Fichero Pruebasensorvelocidad-2.py

```cpp+lineNumbers:true

import RPi.GPIO as GPIO

DataMotorR = 7 DataMotorL = 8

GPIO.setmode(GPIO.BCM)

GPIO.setup(DataMotorR,GPIO.IN) GPIO.setup(DataMotorL,GPIO.IN)

contador=0 repetido=0 num = 100 while (contador<num):
if((GPIO.input(DataMotorR)==1)and(repetido==0)): contador=contador+1 print('\nContador
:',contador) repetido=1 if(GPIO.input(DataMotorR)==0): repetido=0 ```

Copyright 2026
```

3.3 Variables.py

Añadimos ahora las varialbes de paso siguientes a este fichero

VARIABLES.py

import RPi.GPIO as GPIO

DataMotorR = 7 DataMotorL = 8

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

########CONFIGURACION GPIO ENTRADAS SALIDAS

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)

GPIO.setup(DataMotorR,GPIO.IN) GPIO.setup(DataMotorL,GPIO.IN)

########################## VELOCIDAD DE LOS MOTORES

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

3.4 MOVIMIENTOSPASO.py

Vamos a hacer una pequeña función donde le pasemos dos argumentos por cada motor (en total 4 argumentos): velocidad y número de pasos. Tiene que hacer:

- Si el número de pasos es positivo va hacia delante el motor.
- Si el paso es negativo es que el motor va hacia atrás.
- Los motores funcionarán con la velocidades dadas en los argumentos.
- En total 4 argumentos tiene la función, dos para cada motor R y L: *velR,numR,velL,numL* donde *vel* es la velocidad del motor y *num* el número de pasos.

¿Te atreves? Sino, mira la solución:

%accordion%Solución%accordion%

Fichero MOVIMIENTOSPASO.py

```
import RPi.GPIO as GPIO
import time
import MOVIMIENTOS
from VARIABLES import *

######################################################################
####################FUNCIóN AMBOS#####################################
######################################################################
def BOTH(velR,numR,velL,numL):
    repetidoR=0
    repetidoL=0
    if (numR>0):
        GPIO.output(IN1,GPIO.HIGH)
        GPIO.output(IN2,GPIO.LOW)
    else:
        numR=-numR
        GPIO.output(IN1,GPIO.LOW)
        GPIO.output(IN2,GPIO.HIGH)
```

```
    if (numL>0):
         GPIO.output(IN4,GPIO.HIGH)
         GPIO.output(IN3,GPIO.LOW)
    else:
        numL=-numL
        GPIO.output(IN4,GPIO.LOW)
        GPIO.output(IN3,GPIO.HIGH)
    contadorR=0
    contadorL=0
    while ((contadorR
```

Copyright 2026

3.5 Movimientos con paso

Vamos a hacer un programa para utilizar la librería anterior MOVIMIENTOSPASO.py y gobernado por el teclado *numérico*:

- PARAR = tecla 5
- ADELANTE=FORDWARD = 8
- ATRAS=BACKWARD = 2
- DERECHA=RIGHT = 6
- IZQUIERDA=LEFT = 4

Fijaremos de antemano las velocidades y el paso a 50 y 10 por ejemplo.

https://www.youtube.com/embed/n0tOHIHl0Rs

Solución

- Ponemos las librerías fichero MOVIMIENTOS.py y MOVIMIENTOSPASO.py en la misma carpeta que vamos a crear este programa y las incorporamos en el programa con **import**.
- Vamos llamando a las distintas funciones de movimientos según la tecla pulsada, fijamos la velocidad al 50%, por pantalla va saliendo el estado de los contadores.
- Todo dentro de un bucle de manera que si pulsamos la tecla espacio sale del buble no sin antes parar el robot.

%accordion%Solución%accordion%

Fichero 3-5-Movimientos-paso.py

```
import RPi.GPIO as GPIO
import time

import MOVIMIENTOS
import MOVIMIENTOSPASO

velR=50
numR=10
```

```
velL=50

numL=10

print ('TECLAS ¡en minúscula!:\nPARAR = tecla 5\nADELANTE=FORDWARD = 8\nATRAS=BACKWARD =
2\nDERECHA=RIGHT = 6\nIZQUIERDA=LEFT = 4')
tecla='x'
while tecla!='5':
    tecla = input('\nPresiona una tecla y después enter : ')
    if tecla != '5':
        print ('\nHas presionado ', tecla)
        if tecla=='8':
            print ('\nadelante')
            MOVIMIENTOSPASO.BOTH(velR,numR,velL,numL)
        if tecla=='2':
            print ('\natrás')
            MOVIMIENTOSPASO.BOTH(velR,-numR,velL,-numL)
        if tecla=='6':
            print ('\nderecha')
            MOVIMIENTOSPASO.BOTH(velR,-numR,velL,numL)
        if tecla=='4':
            print ('\nizquierda')
            MOVIMIENTOSPASO.BOTH(velR,numR,velL,-numL)

    else:
        print ('\nFin, has apretado STOP')
        MOVIMIENTOS.STOP()
```

%/accordion%

Copyright 2026