Control remoto

- 5 Control remoto
- 5.1 NEC
- 5.2 VARIABLES.py y NEC.py
- 5.3 Test Control Remoto IR
- 5.4 Control remoto

Copyright 2026

5 Control remoto

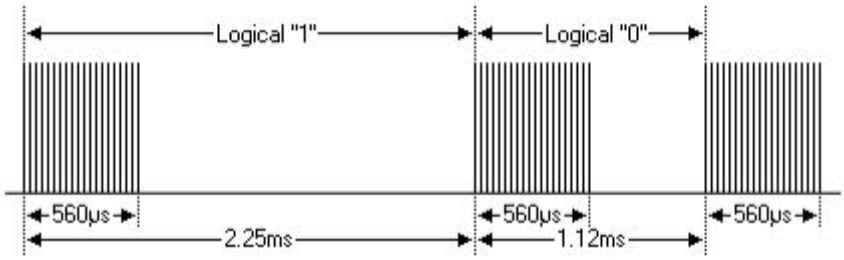


Copyright 2026

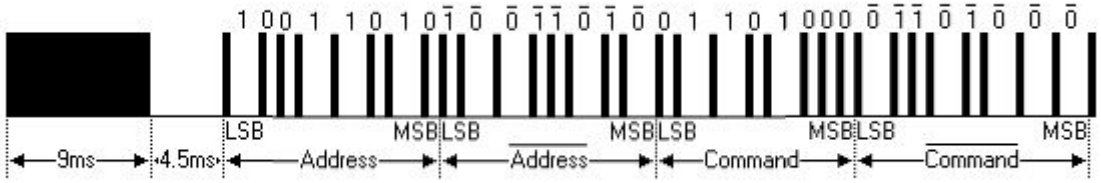
5.1 NEC

Alphabot tiene un receptor LFN0038K compatible con el protocolo estandard NEC y el mando emisor que acompaña a este kit también es compatible con él. Encima, para complicarnos más las cosas, tiene configuración PULL-UP como el resto de sensores de este robot, por lo que si recibe del mando a distancia un 1 lógico, él transmite un 0 y viceversa.

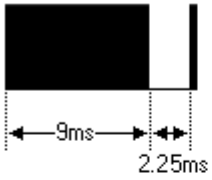
Una señal con el protocolo NEC es una especie de señal PWM donde un '0' es es un ciclo de 1.125ms la emisión de una señal de 0.565ms y un '1' lógico o mismo pero en un intervalo de 2.25ms, se ve mejor con un dibujo:

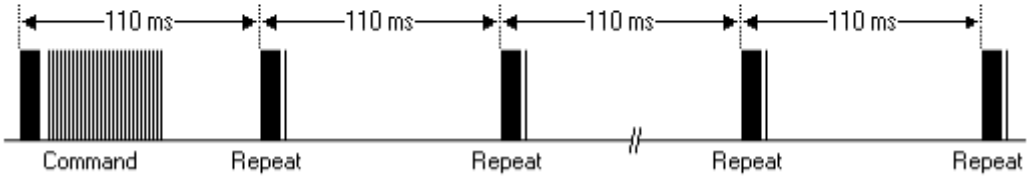


El protocolo establece que primero se emite 9ms de una señal alta, y luego 4.5ms de señal baja para empezar el envío de una señal de 8 bits, empezando por el bit más bajo LSB hasta el más alto MSB y luego su complemento. Todo en una señal de 38kHz donde se modula primero la dirección y luego el comando. Mejor con un dibujo ¿no?



Cada comando se envía una sola vez al menos que mantengas el botón pulsado más de 110ms. El formato de duplicación es según este dibujo:





Copyright 2026

5.2 VARIABLES.py y NEC.py

El sensor IR está unido al GPIO número 18 luego añadimos en el fichero variables.py las siguientes líneas:

> IR = 18
>
> GPIO.setup(IR,GPIO.IN,GPIO.PUD_UP)

lo de PUD_UP es porque su configuración es en PULL-UP

LIBRERIA NEC.py

Creamos este fichero que lo ponemos en la misma carpeta que nuestros ejercicios, el código es complejo, sigue los pasos explicados en el protocolo NEC y lo hemos sacado del código demo de la página https://www.waveshare.com/wiki/AlphaBot :

```cpp+lineNumbers:true
import RPi.GPIO as GPIO from VARIABLES import *

def getkey(): if GPIO.input(IR) == 0: count = 0 while GPIO.input(IR) == 0 and count < 200: #9ms count += 1 time.sleep(0.00006)

    count = 0
    while GPIO.input(IR) == 1 and count < 80:  #4.5ms
        count += 1
        time.sleep(0.00006)

    idx = 0
    cnt = 0
    data = [0,0,0,0]
```

```
    for i in range(0,32):
        count = 0
        while GPIO.input(IR) == 0 and count < 15:    #0.56ms
            count += 1
            time.sleep(0.00006)

        count = 0
        while GPIO.input(IR) == 1 and count < 40:   #0: 0.56mx
            count += 1                               #1: 1.69ms
            time.sleep(0.00006)

        if count > 8:
            data[idx] |= 1<<cnt
        if cnt == 7:
            cnt = 0
            idx += 1
        else:
            cnt += 1
```

print (data)

```
    if data[0]+data[1] == 0xFF and data[2]+data[3] == 0xFF:  #check
        return data[2]

    if data[0] == 255 and data[1] == 255 and data[2] == 15 and data[3] == 255:
        return "repeat"
```

```
```

5.3 Test Control Remoto IR

Vamos a ejecutar un sencillo programa que nos vaya diciendo los códigos que lee las diferentes teclas utilizando la función key de la librería NEC.py

Fichero Test-ControlRemoto-IR.py

```
import RPi.GPIO as GPIO
import time

from VARIABLES import *

import MOVIMIENTOS
import MOVIMIENTOSPASO
import NEC

while True:
    key=NEC.getkey()
    if (key != None):
        print (key)
```

Lo que nos sale por pantalla son los siguiente códigos de las diferentes teclas del mando siguiente:



Estos son los valores (ordenados tal y como están en el mando)

|Tecla|Valor|Tecla|Valor|Tecla|Valor| |-----|-----|-----|-----|-----|-----| |CH-|69|CH|70|CH+|71| |<<|68|>>|64|>|67| |-|7|+|21|EQ|9| |0|22|100+|25|200+|13| |1|12|2|24|3|94| |4|8|5|28|6|90| |7|66|8|82|9|74|

Copyright 2026

5.4 Control remoto

¿Qué esperas? te lo pide el cuerpo !!! vamos a hacer un control remoto del robot !!

Vamos a definir las siguientes teclas

gobernado por el teclado *numérico*:

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

https://www.youtube.com/embed/PfoVh2BTlLY

Solución

La solución es fácil con las librerías que hemos aprendido: * Ponemos las librerías fichero MOVIMIENTOS.py y ahora esta nueva NEC.py en la misma carpeta que vamos a crear este programa y las incorporamos en el programa con **import**. * También incorporamos las variables definidas en **VARIABLES.py** * Utilizaremos los códigos que hemos optenido en Test Control Remoto IR. * Un bucle, si no detecta la tecla 5 que haga los movimientos según las teclas del mando IR.

%accordion%Solución%accordion%

Fichero Control-Remoto-IR.py

```
import RPi.GPIO as GPIO
import time

from VARIABLES import *

import MOVIMIENTOS
```

```
import NEC

vel=50

print ('TECLAS :\nPARAR = tecla 5\nADELANTE=FORDWARD = 2\nATRAS=BACKWARD = 8\nDERECHA=RIGHT =
6\nIZQUIERDA=LEFT = 4')

key=0
while key!=28:
    key=NEC.getkey()
    if (key != None):
        if key==24:
            print ('\nadelante')
            MOVIMIENTOS.FORDWARD(vel)
        if key==82:
            print ('\natrás')
            MOVIMIENTOS.BACKWARD(vel)
        if key==90:
            print ('\nderecha')
            MOVIMIENTOS.RIGHT(vel)
        if key==8:
            print ('\nizquierda')
            MOVIMIENTOS.LEFT(vel)
        if key==28:
            print ('\nFin, has apretado STOP')
            MOVIMIENTOS.STOP()
```

%/accordion%

Copyright 2026