# Code Python

## RobotMain.py

from tkinter import LEFT

from MotorModule import Motor

import KeyPressModule as kp

motor= Motor(2,3,4,17,22,27)

kp.init()

def main():

    if(kp.getKey("UP")):

        motor.move(0.6,0 ,0.1)

        print("Tiến\n")

    elif(kp.getKey("DOWN")):

        motor.move(-0.6, 0, 0.1)

        print("Lùi\n")

    elif(kp.getKey("LEFT")):

        motor.move(0.5,0.2,0.1)

        print("Trái\n")

    elif(kp.getKey("RIGNT")):

         motor.move(0.5, -0.2, 0.1)

         print("Phải\n")

    else:

        motor.stop(2)

        print("Dừng\n")

if \_\_name\_\_ == '\_\_main\_\_':

    while True:

        main()

## KeyPressModule.py

import pygame

def init():

    pygame.init()

    win = pygame.display.set\_mode((100,100))

def getKey(keyName):

    ans = False

    for eve in pygame.event.get():pass

    keyInput = pygame.key.get\_pressed()

    myKey = getattr(pygame,'K\_{}'.format(keyName))

    if keyInput [myKey]:

        ans = True

    pygame.display.update()

    return ans

## MotorModule.py

import RPi.GPIO as GPIO

from time import sleep

GPIO.setmode(GPIO.BCM)

GPIO.setwarnings(False)

class Motor():

    def \_\_init\_\_(self,EnaA,In1A,In2A,EnaB,In1B,In2B):

        self.EnaA = EnaA

        self.In1A = In1A

        self.In2A = In2A

        self.EnaB = EnaB

        self.In1B = In1B

        self.In2B = In2B

        GPIO.setup(self.EnaA,GPIO.OUT)

        GPIO.setup(self.In1A,GPIO.OUT)

        GPIO.setup(self.In2A,GPIO.OUT)

        GPIO.setup(self.EnaB,GPIO.OUT)

        GPIO.setup(self.In1B,GPIO.OUT)

        GPIO.setup(self.In2B,GPIO.OUT)

        self.pwmA = GPIO.PWM(self.EnaA, 100)*;*

        self.pwmA.start(0)*;*

        self.pwmB = GPIO.PWM(self.EnaB, 100)*;*

        self.pwmB.start(0)*;*

    def move(self,speed=0.5,turn=0,t=0):

        speed \*=100

        turn \*=100

        leftSpeed = speed - turn

        rightSpeed = speed + turn

        if leftSpeed>100: leftSpeed=100

        elif leftSpeed<-100: leftSpeed= -100

        if rightSpeed>100: rightSpeed=100

        elif rightSpeed<-100: rightSpeed= -100

        self.pwmA.ChangeDutyCycle(abs(leftSpeed))

        self.pwmB.ChangeDutyCycle(abs(rightSpeed))

        if leftSpeed>0:

            GPIO.output(self.In1A,GPIO.HIGH)

            GPIO.output(self.In2A,GPIO.LOW)

        else:

            GPIO.output(self.In1A,GPIO.LOW)

            GPIO.output(self.In2A,GPIO.HIGH)

        if rightSpeed>0:

            GPIO.output(self.In1B,GPIO.HIGH)

            GPIO.output(self.In2B,GPIO.LOW)

        else:

            GPIO.output(self.In1B,GPIO.LOW)

            GPIO.output(self.In2B,GPIO.HIGH)

        sleep(t)

    def stop(self,t=0):

        self.pwmA.ChangeDutyCycle(0)*;*

        self.pwmB.ChangeDutyCycle(0)*;*

        sleep(t)