##!/usr/bin/env python2.7

# script by Alex Eames http://RasPi.tv/

# http://raspi.tv/2013/how-to-use-interrupts-with-python-on-the-raspberry-pi-and-rpi-gpio

import time

import RPi.GPIO as GPIO

import subprocess

import os

# GPIO 21 set up as input. It is pulled up to stop false signals

GPIO.setmode(GPIO.BCM)

GPIO. setup(21, GPIO.IN, pull\_up-down=GPIO.PUD-UP)

# GPIO 20 set up as output.

GPIO. setup(20, GPIO.OUT)

# GPIO 16 set up as output.

GPIO. setup(16, GPIO.OUT)

# GPIO 19 set up as output.

GPIO. setup(19, GPIO.OUT)

# GPIO 26 set up as output.

GPIO. setup(26, GPIO.OUT)

print "Make sure you have a button connected so that when pressed"

print "it will connect GPIO port 21 (pin 40) to GND (pin 39)\n"

raw\_input("Press Enter when ready\n>")

print "Waiting for falling edge on port 21"

# now the program will do nothing until the signal on port 21

# starts to fall towards zero. This is why we used the pullup

# to keep the signal high and prevent a false interrupt

# subprocess.call("omxplayer --file=/home/pi/lightshowpi/music/sample/ Noddy\ Theme\ Tone.mp3")

print "During this waiting time, your computer is not"

print "wasting resources by polling for a button press.\n"

print "Press your button when ready to initiate a falling edge interrupt."

try:

count=100

while (count>0):

print 'the count is:', count

omxplayer /home/pi/lightshowpi/py/Make/Way/For/Noddy.mp3

# os.system( -q Make\ Way\ for\ Noddy.mp3 &')

GPIO.wait\_for\_edge(21, GPIO.FALLING)

if GPIO.input(21) ==GPIO.LOW:

print "low"

else:

print "waiting"

# subprocess.call(['killall'])

print "\nFalling edge detected. Now your program can continue with"

print "whatever was waiting for a button press."

# gpio 16 to sound horn for 1 second

GPIO.output(16, 0)

time.sleep(1)

GPIO.output(16, 1)

# gpio 19 to turn wheels on

GPIO.output(19, 0)

# gpio 20 to open and close car doors while lightshow is running

GPIO.output(20, 0)

# red\_led.blink(5, 5, 5)

# gpio 26 to rise and lower PC Plod while lightshow is running

GPIO.output(26, 0)

# green\_led.blink(5, 8, 5)

# subprocess.call("sudo python synchronized\_lights.py --file=/home/pi/lightshowpi/music/sample/Happy\ Birthday\ Emma.mp3", shell=True)

# gpio to stop PC Plod

GPIO.output(26, 1)

# gpio 20 to stop open and close of car doors

GPIO.output(20, 1)

# gpio 19 to turn wheels off

GPIO.output(19, 1)

# wait before next cycle

time.sleep(1)

os.system('mpg123 -q Make\ Way\ for\ Noddy.mp3 &')

GPIO.wait\_for\_edge(21, GPIO.FALLING)

subprocess.call(['killall', 'mpg123'])

print "\nFalling edge detected. Now your program can continue with"

print "whatever was waiting for a button press."

# gpio 16 to sound horn for 1 second

GPIO.output(16,0)

time.sleep(1)

GPIO.output(16, 1)

# gpio 19 to turn wheels on

GPIO.output(19, 0)

# gpio 20 to open and close car doors while lightshow is running

GPIO.output(20, 0)

# red\_led.blink(5, 5, 5)

# gpio 26 to rise and lower PC Plod while lightshow is running

GPIO.output(26, 0)

# green\_led.blink(5, 8, 5)

subprocess.call("sudo python synchronized\_lights.py --file=/home/pi/lightshowpi/music/sample/Happy\ Birthday\ Lucy.mp3", shell=True)

# gpio 26 to stop PC Plod

GPIO.output(26, 1)

# gpio 20 to stop open and close of car doors

GPIO.output(20, 1)

# gpio 19 to turn wheels off

GPIO.output(19, 1)

# wait before next cycle

time.sleep(1)

count= count-1

except KeyboardInterrupt:

GPIO.cleanup() # clean up GPIO on CTRL+C exit

GPIO.cleanup() # clean up GPIO on normal exit