**Teach-In 2014 with Raspberry Pi: Part 3**

by Mike and Richard Tooley

# Warn user that he/she is about to quit

def warn():

print('You have selected Quit')

print('Are you sure?')

warn()

You have selected Quit

Are you sure?

# Function to convert Celsius to Fahrenheit

def c\_to\_f(x):

return(9/5 \* x + 32)

# Get a temperature and convert it!

cS = input('Enter a temperature in deg.C: ')

c = float(cS)

print(cS + ' deg.C is equivalent to ' + str(c\_to\_f(c)) + ' deg.F')

Enter a temperature in deg.C: +120

+120 deg.C is equivalent to 248.0 deg.F

# Function to determine time constant of a C-R circuit

def cr\_time(x, y):

return(x \* y)

# Function to determine the cut-off frequency of a C-R filter

def cr\_freq(x, y):

return(0.159 / (x \* y))

# Get values of C and R

cS = input('Enter value of capacitance in uF: ')

c = float(cS)

rS = input('Enter value of resistance in Mohm: ')

r = float(rS)

# Print the results

print('Time constant is %.3f seconds' % cr\_time(c, r))

print('Cut-off frequency is %.3f Hz' % cr\_freq(c, r))

Enter value of capacitance in uF: 0.22

Enter value of resistance in Mohm: 0.15

Time constant is 0.033 seconds

Cut-off frequency is 4.818 Hz

import sys

# Function to convert Celsius to Fahrenheit

def c\_to\_f(x):

return(9/5 \* x + 32)

# Get a temperature and convert it!

cS = input('Enter a temperature in deg.C: ')

try:

c = float(cS)

except ValueError:

sys.exit('You must enter a number!')

print(cS + ' deg.C is equivalent to ' + str(c\_to\_f(c)) + ' deg.F')

import RPi.GPIO as GPIO

sudo control.py

GPIO.setmode(GPIO.BOARD)

GPIO.setmode(GPIO.BCM)

# Configure the GPIO

GPIO.setmode(GPIO.BOARD)

GPIO.setup(11, GPIO.OUT) # GP0/OP1

GPIO.setup(12, GPIO.OUT) # GP1/OP2

GPIO.setup(13, GPIO.OUT) # GP2/OP3

GPIO.setup(15, GPIO.OUT) # GP3/OP4

GPIO.setup(16, GPIO.OUT) # GP4/OP5

GPIO.setup(18, GPIO.OUT) # GP5/OP6

GPIO.setup(22, GPIO.OUT) # GP6/OP7

GPIO.setup(7, GPIO.OUT) # GP7/OP8

# Initialise all lines to the OFF state

GPIO.output(11, False)

GPIO.output(12, False)

GPIO.output(13, False)

GPIO.output(15, False)

GPIO.output(16, False)

GPIO.output(18, False)

GPIO.output(22, False)

GPIO.output(7, False)

# Turn OP2 on

GPIO.output(12, True)

# Turn OP2 off

GPIO.output(12, False)

def light\_on():

GPIO.output(12, True)

def light\_off():

GPIO.output(12, False)

light\_on()

light\_off()

import RPi.GPIO as GPIO

import time

# Configure GPIO

GPIO.setmode(GPIO.BOARD)

GPIO.setup(11, GPIO.OUT) # GP0/OP1

GPIO.setup(12, GPIO.OUT) # GP1/OP2

GPIO.setup(13, GPIO.OUT) # GP2/OP3

GPIO.setup(15, GPIO.OUT) # GP3/OP4

GPIO.setup(16, GPIO.OUT) # GP4/OP5

GPIO.setup(18, GPIO.OUT) # GP5/OP6

GPIO.setup(22, GPIO.OUT) # GP6/OP7

GPIO.setup(7, GPIO.OUT) # GP7/OP8

# Initialise with all lines OFF

GPIO.output(11, False)

GPIO.output(12, False)

GPIO.output(13, False)

GPIO.output(15, False)

GPIO.output(16, False)

GPIO.output(18, False)

GPIO.output(22, False)

GPIO.output(7, False)

# Light is assigned to OP2

def light\_on():

GPIO.output(12, True)

def light\_off():

GPIO.output(12, False)

# Camera is assigned to OP3

def camera\_on():

GPIO.output(13, True)

def camera\_off():

GPIO.output(13, False)

# Main loop

while True:

light\_on()

camera\_off()

time.sleep(2) # Wait 2 seconds

camera\_on()

time.sleep(2) # Wait 2 seconds

camera\_off()

light\_off()

time.sleep(10) # Wait 10 seconds

def light(switch):

if switch:

GPIO.output(12, True)

else:

GPIO.output(12, False)

light(True)

light(False)

import RPi.GPIO as GPIO

import time

# Configure GPIO

GPIO.setmode(GPIO.BOARD)

GPIO.setup(11, GPIO.OUT) # GP0/OP1

GPIO.setup(12, GPIO.OUT) # GP1/OP2

GPIO.setup(13, GPIO.OUT) # GP2/OP3

GPIO.setup(15, GPIO.OUT) # GP3/OP4

GPIO.setup(16, GPIO.OUT) # GP4/OP5

GPIO.setup(18, GPIO.OUT) # GP5/OP6

GPIO.setup(22, GPIO.OUT) # GP6/OP7

GPIO.setup(7, GPIO.OUT) # GP7/OP8

# Initialise with all lines OFF

GPIO.output(11, False)

GPIO.output(12, False)

GPIO.output(13, False)

GPIO.output(15, False)

GPIO.output(16, False)

GPIO.output(18, False)

GPIO.output(22, False)

GPIO.output(7, False)

# Light is assigned to OP2

def light(switch):

if switch:

GPIO.output(12, True)

else:

GPIO.output(12, False)

# Camera is assigned to OP3

def camera(switch):

if switch:

GPIO.output(13, True)

else:

GPIO.output(13, False)

# Main loop

while True:

light(True)

camera(False)

time.sleep(2) # Wait 2 seconds

camera(True)

time.sleep(2) # Wait 2 seconds

camera(False)

light(False)

time.sleep(10) # Wait 10 seconds

**[Box]**

GPIO.cleanup()

sudo python /home/pi/Desktop/Work\_files/oc\_driver.py

sudo python Desktop/Work\_files/oc\_driver.py

cd Desktop

cd Work\_files

cd Desktop/Work\_files

sudo python oc\_driver.py

**[End box]**