Khoi Duong

Prof. Yang

CE450L

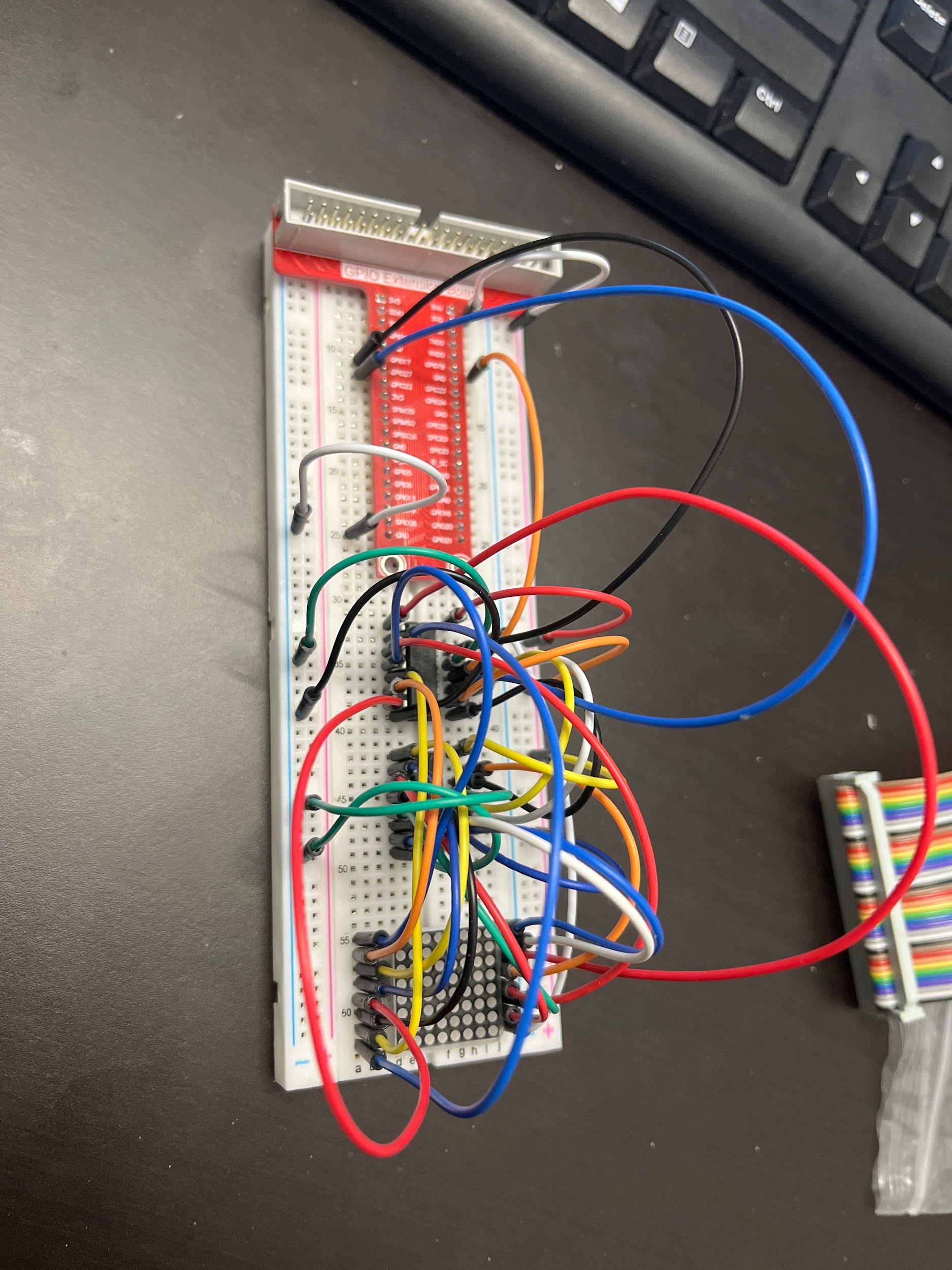
12/17/2022

LAB#11

GitHub link: <https://github.com/MynameisKoi/CE450L/tree/main/Lab%2311>



Breadboard setup:



Source code: <https://github.com/MynameisKoi/CE450L/blob/main/Lab%2311/Dot_matrix.py>

#!/usr/bin/env python3

import RPi.GPIO as GPIO

import time

from sys import version\_info

if version\_info.major == 3:

raw\_input = input

SDI = 17

RCLK = 18

SRCLK = 27

# we use BX matrix, ROW for anode, and COL for cathode

# ROW ++++

code\_H = [0x01,0xff,0x80,0xff,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff]

# COL ----

code\_L = [0x00,0x7f,0x00,0xfe,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xfe,0xfd,0xfb,0xf7,0xef,0xdf,0xbf,0x7f]

def print\_msg():

print ("========================================")

print ("| Dot matrix with two 74HC595 |")

print ("| ------------------------------ |")

print ("| SDI connect to GPIO17 |")

print ("| RCLK connect to GPIO18 |")

print ("| SRCLK connect to GPIO27 |")

print ("| |")

print ("| Control Dot matrix with 74HC595 |")

print ("| |")

print ("| SunFounder|")

print ("========================================")

print ("Program is running...")

print ("Please press Ctrl+C to end the program...")

raw\_input ("Press Enter to begin\n")

def print\_matrix(*matrix*):

for i in range(0,len(*matrix*)):

print (*matrix*[i])

def get\_matrix(*row\_buffer*, *col\_buffer*, *max\_row*=8, *max\_col*=8):

matrix\_msg = [[0 for i in range(*max\_row*)] for i in range(*max\_col*)]

print("row\_buffer = 0x%02x , col\_buffer = 0x%02x"%(*row\_buffer*, *col\_buffer*))

for row\_num in range(0,8):

for col\_num in range(0,8):

#print (row\_num, col\_num), '-->', (((row\_buffer >> row\_num) & 0x01), ((col\_buffer >> col\_num) & 0x01))

if (((*row\_buffer* >> row\_num) & 0x01) - ((*col\_buffer* >> col\_num) & 0x01)):

matrix\_msg[row\_num][col\_num] = 1

print\_matrix(matrix\_msg)

matrix\_msg = [[0 for i in range(*max\_row*)] for i in range(*max\_col*)]

def setup():

GPIO.setmode(GPIO.BCM) # Number GPIOs by its BCM location

GPIO.setup(SDI, GPIO.OUT)

GPIO.setup(RCLK, GPIO.OUT)

GPIO.setup(SRCLK, GPIO.OUT)

GPIO.output(SDI, GPIO.LOW)

GPIO.output(RCLK, GPIO.LOW)

GPIO.output(SRCLK, GPIO.LOW)

# Shift the data to 74HC595

def hc595\_shift(*dat*):

for bit in range(0, 8):

GPIO.output(SDI, 0x80 & (*dat* << bit))

GPIO.output(SRCLK, GPIO.HIGH)

time.sleep(0.001)

GPIO.output(SRCLK, GPIO.LOW)

GPIO.output(RCLK, GPIO.HIGH)

time.sleep(0.001)

GPIO.output(RCLK, GPIO.LOW)

def main():

print\_msg()

while True:

for i in range(0, len(code\_H)):

hc595\_shift(code\_L[i])

hc595\_shift(code\_H[i])

get\_matrix(code\_L[i], code\_H[i])

time.sleep(0.1)

for i in range(len(code\_H)-1, -1, -1):

hc595\_shift(code\_L[i])

hc595\_shift(code\_H[i])

get\_matrix(code\_L[i], code\_H[i])

time.sleep(0.1)

def destroy():

GPIO.cleanup()

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

setup()

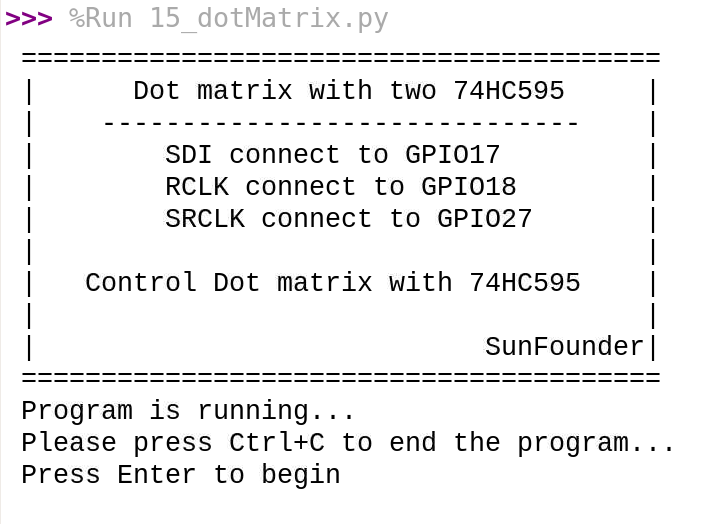
try:

main()

except KeyboardInterrupt:

destroy()

Run program & demonstration:



Video link: <https://youtu.be/GeNxwvT7hhI>



We will use the same breadboard setup as above

Source code: <https://github.com/MynameisKoi/CE450L/blob/main/Lab%2311/A_and_B.py>

#!/usr/bin/env python

import RPi.GPIO as GPIO

import time

from sys import version\_info

if version\_info.major == 3:

raw\_input = input

SDI = 17

RCLK = 18

SRCLK = 27

per\_line = [0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf, 0xbf, 0x7f]

charactors = {

"A" : [0b00111100,0b00111100,0b01100110,0b01100110,0b01111110,0b01111110,0b01100110,0b01100110],

"B" : [0b01111000,0b01111110,0b01100110,0b01111000,0b01111000,0b01100110,0b01111110,0b01111000]}

def print\_msg():

print ("========================================")

print ("| Dot matrix with two 74HC595 |")

print ("| ------------------------------ |")

print ("| SDI connect to GPIO17 |")

print ("| RCLK connect to GPIO18 |")

print ("| SRCLK connect to GPIO27 |")

print ("| |")

print ("| Control Dot matrix with 74HC595 |")

print ("| |")

print ("| SunFounder|")

print ("========================================")

print ("Program is running...")

print ("Please press Ctrl+C to end the program...")

raw\_input ("Press Enter to begin\n")

def setup():

GPIO.setmode(GPIO.BCM) # Number GPIOs by its BCM location

GPIO.setup(SDI, GPIO.OUT)

GPIO.setup(RCLK, GPIO.OUT)

GPIO.setup(SRCLK, GPIO.OUT)

GPIO.output(SDI, GPIO.LOW)

GPIO.output(RCLK, GPIO.LOW)

GPIO.output(SRCLK, GPIO.LOW)

# Shift the data to 74HC595

def hc595\_in(*dat*):

for bit in range(0, 8):

GPIO.output(SDI, 1 & (*dat* >> bit))

GPIO.output(SRCLK, GPIO.HIGH)

time.sleep(0.000001)

GPIO.output(SRCLK, GPIO.LOW)

def hc595\_out():

GPIO.output(RCLK, GPIO.HIGH)

time.sleep(0.000001)

GPIO.output(RCLK, GPIO.LOW)

def flash(*table*):

for i in range(8):

hc595\_in(per\_line[i])

hc595\_in(*table*[i])

hc595\_out()

# Clean up last line

hc595\_in(per\_line[7])

hc595\_in(0x00)

hc595\_out()

def show(*table*, *second*):

start = time.time()

while True:

flash(*table*)

finish = time.time()

if finish - start > *second*:

break

def main():

word = 'AB'

while True:

for table in word:

show(charactors[table],1)

time.sleep(1)

def destroy():

GPIO.cleanup()

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

setup()

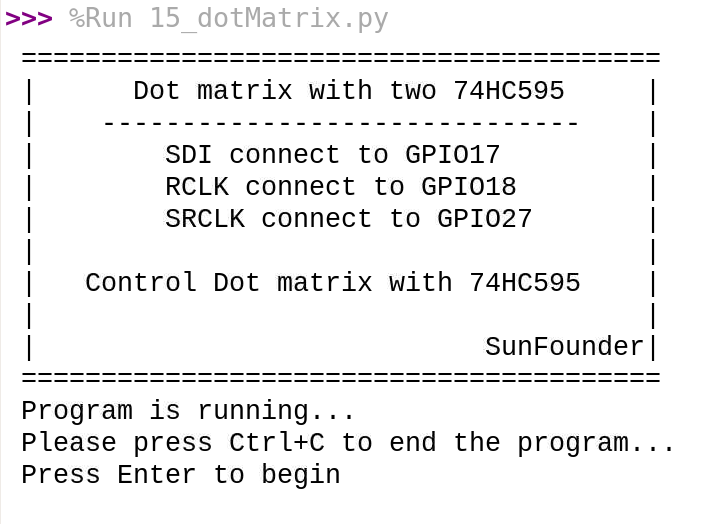
try:

main()

except KeyboardInterrupt:

destroy()

Run program & demonstration:



Video link: <https://youtu.be/2eJWBktG4uw>