Friday, 3 January 2025 10:04 AM

Lab 10 C

Code

```
#include <wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

Adafruit_SSD1306 display(128,32,-1);

#define Buzzerpin
#define echopin
#define Trigpin
#define DHT Pin 10
#define Servo Pin
#define Oil Pin

void setup() {
  display.begin(SSD1306_SWITCHCAPVCC,
                0x3C);
  display.display();
  display.setTextSize(1);
  display.setColor(WHITE);
  display.clearDisplay();
  Serial.begin(115200);
  pinMode(Buzzerpin,OUTPUT);
  pinMode (echopin,INPUT);
  pinMode (Trigpin,OUTPUT);

void loop() {
  digitalWrite(Trigpin,HIGH);
  delay(10);
  digitalWrite(Trigpin,LOW);
  long duration = pulseIn(echopin,HIGH);
  float dist = duration * 0.034 / 2;

  Serial.print("Distance: ");
  Serial.print(dist);
  Serial.print("cm");
  if (dist > dist_thresh) {
    tone(Buzzerpin, 1000);
    digitalWrite(Servo pin, HIGH);
    }
  else {
    noTone(Buzzerpin);
    digitalWrite(Servo pin, LOW);
    }

  display.clearDisplay();
  display.setCursor(0,0);
  display.print("Dist:");
  display.print(dist);
  display.print("cm");
  display.display();
  delay(100);
  }
```

2nd code:

```
#include <Wire.h>
#  include < Adafruit_SSD1306.h>
#  include <dht.h>
#  include <Adafruit_Sensor.h>

#define DHTTYPE DHT11
#define DHT PIN

DHT dht(DHTPIN, DHTTYPE);

Adafruit_SSD1306 display(128,32,-1);

void setup() {
  Serial.begin(115200);
  display.begin(SSD1306_SWITCHCAPVCC,
                0x3C);
  display.setTextSize(1);
  display.setColor(WHITE);
  display.clearDisplay();
  }

void loop() {
  float Temp = dht.readTemperature();
  float H = dht.readHumidity();
  Serial.print("temperature:");
  Serial.print("Temp");
  Serial.print("*C");
  Serial.print("Humidity ");
  Serial.print(H);
  Serial.print("%");

  display.clearDisplay();
  display.setCursor(0,0);
  display.print("Temp:");
  display.print(Temp);
  display.print("C");
  display.print("*");
  display.print("Hum:");
  display.print(H);
  display.print("%");
  display.display();
  delay(1000);
  }
```

3rd code

```
import board
import Adafruit_dht
import matplotlib.pyplot as plt
import time

Sensor = Adafruit_dht.DHT11(board.D4)
                            GPIO4
s = []
h = []
a = 0
while (a<10):
  temp = Sensor.temperature
  Humi = Sensor.Humidity
  s.append(temp)
  h.append(Humi)
  Sensor.sleep(2)
  a = a

plt.plot(s,r)
plt.xlabel("No of iters")
plt.ylabel("Temp")
plt.title("Temp graph")
plt.show()

plt.bar(range(len(h)), h,width)
plt.xlabel("No of iters")
plt.ylabel("Humidity")
plt.title("Humidity Graph")
plt.show()
```

4th code

```
import board
import requests
import time
import Adafruit_dht

Sensor = Adafruit_dht.DHT(board.D4)

api_key = "
                    "

while (1>0):
    T = Sensor.temperature
    H = Sensor.Humidity
    payload = f"api_key={api_key}&field1={T}&field2={H}"
    req = requests.post("http://api.thingspeak.com/update", data=payload)

    if req.status_code == requests.codes.ok:
        print("Data Sent")
    else:
        print("error")
    time.sleep(20)
```

Step 5:

```
from tkinter import *
from time import sleep
from gpiozero import Angular Servo
servo = AngularServo(17,min=-90,
                     max=90)
def update():
    servo = Angular Servo(17,min=-90,
                          max=90)
    angle = VS.get()
    while True:
        servo.angle = angle
        sleep(10)
        break

VS = double Var()
Title label = Label(root, text = "select angle
                    font="Times",20,"bold",
                    fg="#4BCE115")
Slider = Scale(root, from_=-90, to=90,
               length=300, tickinterval=10, orient=
               HORIZONTAL)
Submit Btn = Button(root, text="update",
                    bg="#ccccc")

Title label.pack(anchor= CENTER)
Slider.pack(anchor = CENTER)
Submit Btn.pack(anchor= CENTER)
root.mainloop()
```

6th code

```
import socket
import time

S = Socket.socket()
port = 12345
S.bind = ("", port)

S.listen(5)

While True:
    c,addr = S.accept()
    c.Send ("Thank you" + encode())
    c.Close()
    break
c.close()
```

7th

```
import socket
s = Socket.socket()
port = 12345
S.bind ("127.0.0.1", port)
print (s.recv(1024).decode())
s.close()
```

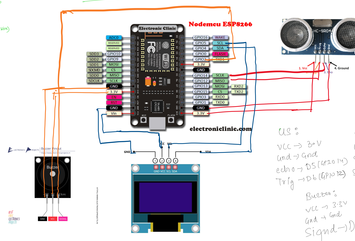
8th

```
from pushbullet import Pushbullet
import cv2
from time import sleep
cam = cv2.VideoCapture (0)
img = 0
def capture:
    ret, img = cam.read()
    R = cv2.imwrite (img,
cv2.imwrite ("/home/mini/Desktop/
                testing.jpg", img)
cv2.destroyAllWindows()
time.release()
sleep(20)
def notification():
    pb = Pushbullet("apikey")
    dev = pb.get_device("device name")
with open ("testing.jpg",'rb') as pic:
    file_upload = pb.upload_file (pic,'pic.jpg')
    push = pb.push_file (**file_upload)
    Value = push['active']

capture ()
notification ()
```

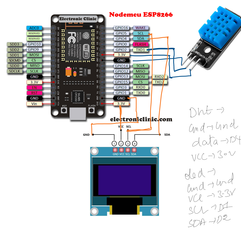
Code 7

```
import cv2
cam = cv2.VideoCapture (0)
while True:
    img = 0
def capture ():
    ret, img = cam.read()
    R = cv2.imwrite (img)
cv2.imwrite ("/home/mini/Desktop/
                testing.jpg", img)
cam.release()
cv2.destroyAllWindows()

def Annotate():
    Img = cv2.imread ("testing.jpg",
                      img)
    ImgGray = Img.copy, ImgGray
    start = (0,0)
    end = (200,100)
cv2.rectangle (ImgGray,start,end,
               (0,255),thickness = 3,
               line_type = cv2.LINE_8)
cv2.imshow ("Annotated Img", ImgGray)
cv2.waitKey (1000)

while (1):
    capture ()
    annotate()
```

Code 1



Code 2:



for prog 3 and 4:

prog 5: