National Textile University, Faisalabad

Department of Computer Science

Name:	**Hamad khan**
Class:	**BSCS 5th B**
Registration No:	**23-NTU-CS-1098**
Course Name:	**Emedded IOT System**
Submitted To:	**Sir Nasir sb**
Submission Date:	**26/10/2025**

https://github.com/ranahamad003/IOT-23-NTU-CS-B-1098/tree/main/Assignment%20_1%2023_NTU_CS_1098

https://wokwi.com/projects/445868683781107713

https://wokwi.com/projects/445876374136970241

```
#include <wire.h>
#include <Adafruit-GFX.h>
#include <Adafruit-SSD1306.h>

#define Button-MODE-PIN 14
#define Button-RESET-PIN27
#define LED1 - PIN 5
#define LED2 - PIN 18
#define LED3 - PIN 19
#define BUZZER-PIN 25

#define SCREEN-WIDTH 128
#define SCREEN-HEIGHT 64
#define OLED-ADDR 0x3C
Adafruit-SSD1306
display (SCREEN-WIDTH,
SCREEN-HEIGHT, & WIRE-1);

int currentMode=0;
const int TOTAL-MODES=4;
unsigned long PreviousBlink Time=0;
int blinkstate =0;
int fadevalue=0;
bool fade Direction=true;

const char* modeName[]=
{
"All LEDs OFF",
"ONE-BY-ONE Blink",
"All LEDs ON",
}

void test All LEDs();
void update Display();
```

Galax

```
void oneByone Blink();
void PwmFade();

void setup(){
Serial.begin(115200);

Pin Mode(BUTTON-MODE-PIN,
  Input-PULLUP);
Pin Mode(BUTTON_RESET-PIN,
  Input-PullUP);

Pin Mode(LED1-PINOutPut);
 Pin Mode(LED2PIN output);
 Pin Mode(LED3PIN,output);
 pin Mode(BUZZER_PIN-Output);

 if(!display.begin(SSD1306-SW)
  ITCHCAPVCC, oLED-ADDR))
 {
  Serial.println("SSD1306 allocation
  failed!");

 Serial println("oLED initialized
   successfully!");
  }

display.clearDisplay();
display.setTextSize(1);
display.setTextColor(white);
display.setCursor(0,0);

test allLEDS();

 resetLEDS();
```

```
updateDisplay ();
Serial.println("Task 1- LED Mode
cycling system started");
serial.println("press BUTTON 1(PIN 14) to
 change mode");
serial println("pres BUTTON2(PIN21)
 to reset to off");
  }
void loop () {

if (digitalRead (BUTTON-MODE-PIN) ==
 Low) {
currentMode = (currentMode + 1) %
 Total - Modes;
 resetLEDs ();
 update display();
serial.Print("Mode changed to: ");
 serial println (modeNames [current
  Mode]);
    delay (300);
    }

  if (digitalRead (BUTTON-RESET-PIN) ==
  Low) {

  currentMode = 0;
   reset LEDS ();
  updateDisplay();
 serial.println("Reset to OFF mode");
   delay (300);
    }

{
 case 0;
```

```
resetLEDs();
break;
case 1:
oneByoneBlink();
break;
case 2:
digitalWrite(LED1-PIN, HIGH);
digitalWrite(LED2-PIN, HIGH);
digitalWrite(LED3-PIN, HIGH);
break;
case 3:
PwmFade();
break;
}
}

void testAllLEDs(){
Serial.println("Testing LED1(Pin5)...");
digitalWrite(LED1-PIN, HIGH);
delay(500);
digitalWrite(LED1-PIN-HIGH);
delay(200);

Serial.println("Testing LED2(Pin18)...");
digitalWrite(LED2-PIN, HIGH);
delay(500);
digitalWrite(LED2-PIN-LOW);
delay(200);
Serial.println("All LEDs tested
successfully!");
}
```

```
void resetLEDs()
{
 digitalWrite(LED1-PIN, LOW);
 digitalWrite(LED2-PIN, LOW);
 digitalWrite(LED3-PIN, LOW);
 analogWrite(LED3-PIN, 0);
   blinkState=0;
    }

void oneByoneBlink(){
 unsigned long currentTime=millis();
if (currentTime-previousBlinkTime>=
 500){

previous blink time = currentTime;

digitalWrite(LED1-PIN, LOW);
digitalWrite(LED2-PIN, LOW);
digitalWrite(LED3-PIN, LOW);

 switch(blinkstate){
  case 0;
digital write(LED1-PIN, HIGH);
serial.println("LED 1 ONLPin5)");
 break;
cas1;
digitalwrite(LED2-PIN-HIGH);
 serial.println("LED2 ONLPin18J");
 break
case2:
digitalwriteLED3-PIN-HIGH);
 serial.println("LED3-OIV(Pin19)");
  break;
   }
```

```
void pwm Fade() {
Static unsigned long last FadeTime=0;
unsigned long currentTime= millis();
if (currentTime- last FadeTime >=30){
  lastFade Time = currentTime;
  fade value += (fadeDirection ? 5 : -5);
if (fade value >=255){
  fade value=255;
  fade Direction = false;
} else if(fadevalue <=0){
  fadevalue=0;
  fade Direction=true;
}

void update display () {
  display.clearDisplay ();
display.setTenTSize(2);
display.println("welcome Hamad");
  display.display();
}
```

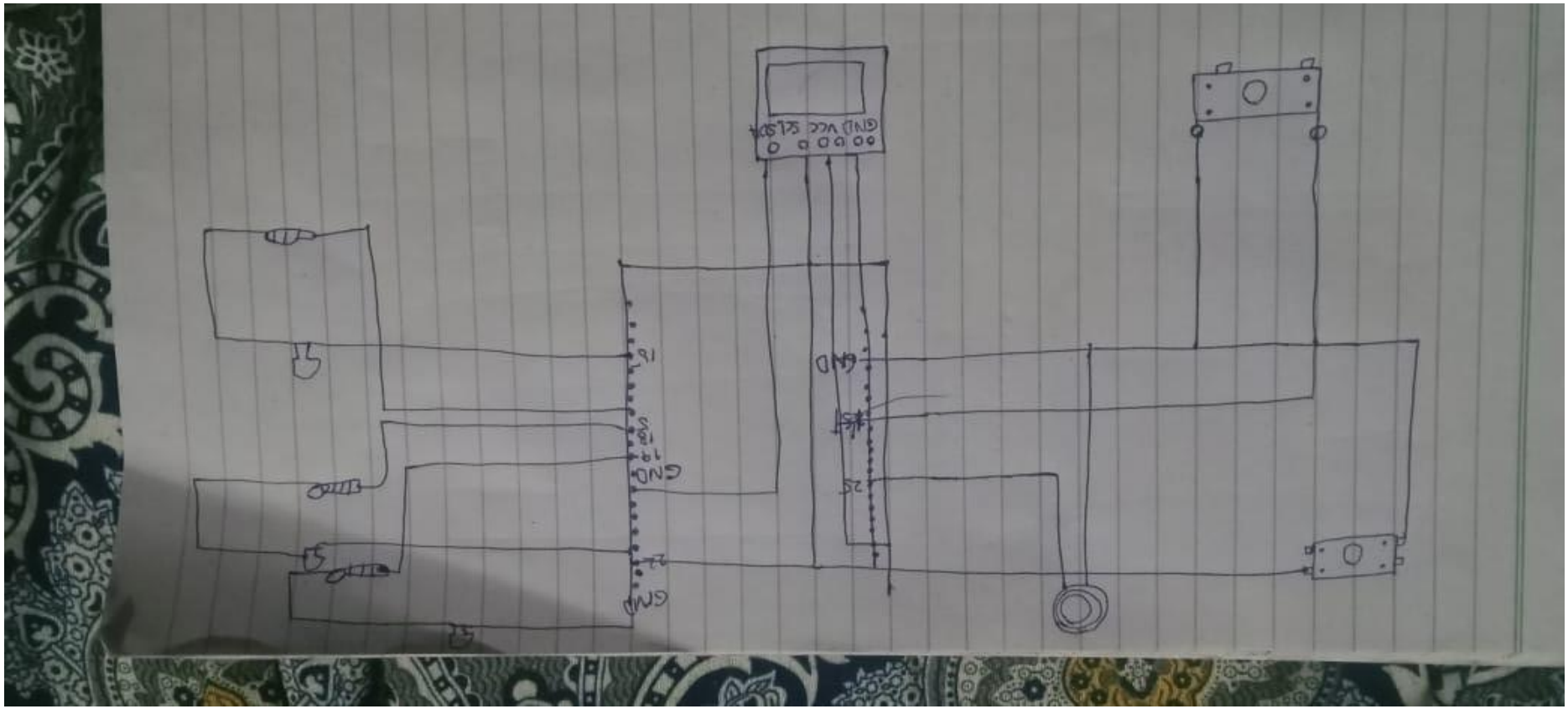
TASK_B

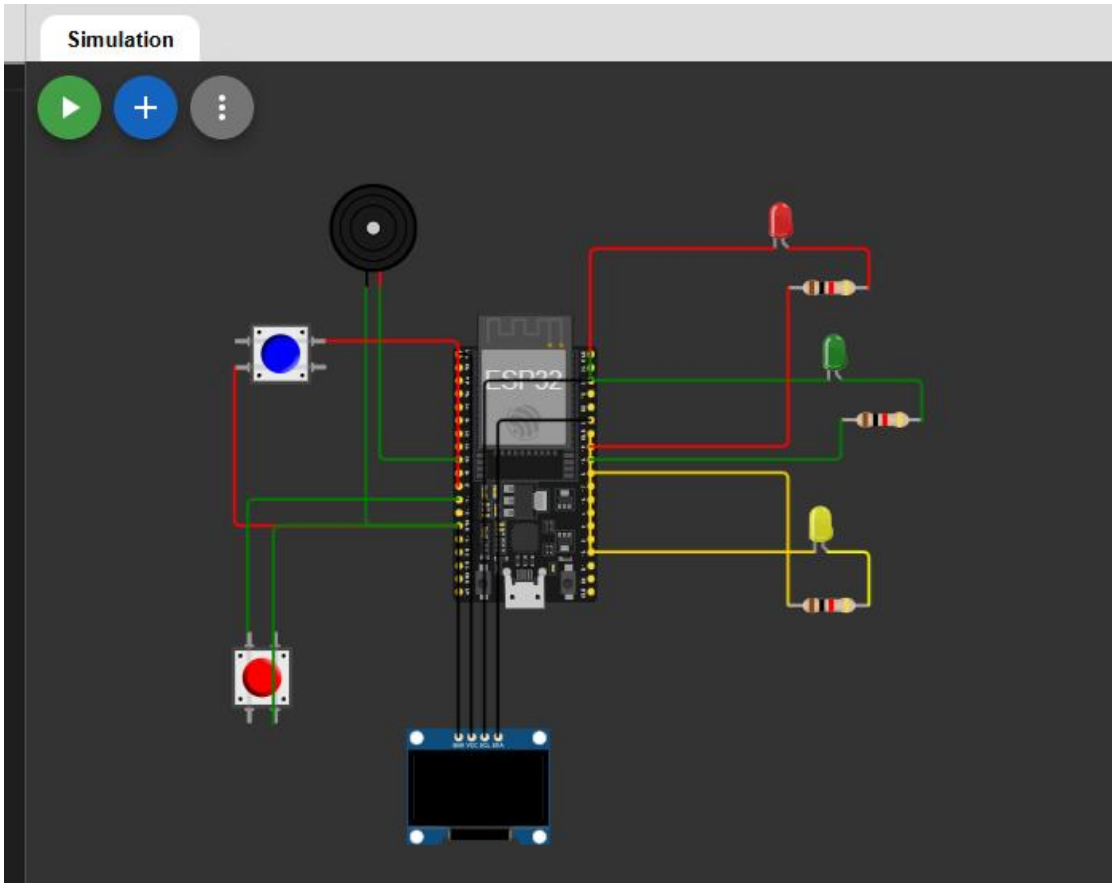
```
#include <wire.h>
#include <Adafruit-GFX.h>
#include <Adafruit-SSD1306.h>

#define BUTTON-PIN 14
#define LED-PIN 5
#define Buzzer-PIN 25

#define SCREEN-WIDTH 128
#define SCREEN-HEIGHT 64
#define OLED-ADDR 0x3C
Adafruit-SSD1306
display (SCREE-WIDTH,
  SCREEN-HEIGHT, &Wire, -1);

bool lastBUTTONState= HEIGH;
bool BUTTON pressed = false;
unsigned long press start Time = 0;
bool ledstate = false;

void setup(){

Serial.begin(115200);
PinMode (BUTTON-PIN,
  Input-PULLUP);
PinMode(LED-PIN, output);
if(!display.begin(SSD1306-SW
  ITCHCAPVCC, OLED-ADDR))
{
serial.println(SSD1306 allocation
    failed!");
}else{

serial.println(OLED initialized
```

```
Successfull!");
}

display.clearDisplay();
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0,0);

digitalWrite(LED_PIN,LOW);
updateDisplay();

Serial.println("Task B-Press Detection
    system started");
serial.println("short press:
        Toggle LED");
serial.println("long press (>1.5S);
Activate buzzer");
}

void loop(){

bool currentButtonState=
digitalRead(BUTTON_PIN);

if(lastButtonState==HIGH &&
currentButtonState==LOW){
pressStartTime=millis();
button Pressed = true;
serial.println("Button pressed-
timing...");
}
```

```
if (last ButtonState == LOW &&
  current ButtonState == HIGH &&
button Pressed ){

  button Pressed = false;
unsigned long pres Duration = millis() -
 press start Time;

 if (Pres Duration <
  Long-PRESS-TIME ){

   ledstate = !ledstate
digital Write(LED→PIN, ledState);
 serial. print ("SHORTPREES-LED");
 serial. println( led stat ? "ON" : "OFF");
  }
 }

 if (button Pressed && (millis() -
press start Time > LONG-PRESS-TIME)){

serial. print in("LONG PRESS - Buzzer
          activated");
 button pressed = false;
  }
last Button state = current Button state;
 delay(50);

void update Display (){
 display. clear Display
```

```
display.setTextsize(2);
display.setcursor(15,15);
display.println("welcome");

display.setText size(2);
display.setCursor(35,40);
display.println("Hamad");
display.display();

}
```


GND VCC SCL SDA
GND
GND


Simulation