**Task-1**

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

#define SCREEN\_WIDTH 128

#define SCREEN\_HEIGHT 64

Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &**Wire**, -1);

// Pin definitions

#define LED1 18

#define LED2 19

#define LED3 4

#define BUZZER 27

#define BTN\_MODE 32

#define BTN\_RESET 33

int mode = 0; // 0: OFF, 1: Alternate Blink, 2: ON, 3: Fade

bool lastButtonState = HIGH;

unsigned long lastDebounceTime = 0;

int fadeValue = 0;

void showMode() {

  display.clearDisplay();

  display.setTextSize(1);

  display.setTextColor(SSD1306\_WHITE);

  display.setCursor(0, 10);

  display.print("Mode: ");

  switch (mode) {

    case 0: display.print("All OFF"); break;

    case 1: display.print("Alternate Blink"); break;

    case 2: display.print("All ON"); break;

    case 3: display.print("PWM Fade"); break;

  }

  display.display();

}

void setup() {

  pinMode(LED1, OUTPUT);

  pinMode(LED2, OUTPUT);

  pinMode(LED3, OUTPUT);

  pinMode(BUZZER, OUTPUT);

  pinMode(BTN\_MODE, INPUT\_PULLUP);

  pinMode(BTN\_RESET, INPUT\_PULLUP);

  // OLED init

**Wire**.begin();

  if (!display.begin(SSD1306\_SWITCHCAPVCC, 0x3C)) {

    for (;;);

  }

  display.clearDisplay();

  display.setTextSize(1);

  display.setTextColor(SSD1306\_WHITE);

  display.setCursor(10, 20);

  display.print("System Ready!");

  display.display();

  delay(1000);

  showMode();

}

void loop() {

  bool buttonState = digitalRead(BTN\_MODE);

  bool resetState = digitalRead(BTN\_RESET);

  if (resetState == LOW) {

    mode = 0;

    showMode();

    digitalWrite(LED1, LOW);

    digitalWrite(LED2, LOW);

    digitalWrite(LED3, LOW);

    noTone(BUZZER);

  }

  if (buttonState == LOW && lastButtonState == HIGH && millis() - lastDebounceTime > 300) {

    mode = (mode + 1) % 4;

    showMode();

    tone(BUZZER, 1000, 100);

    lastDebounceTime = millis();

  }

  lastButtonState = buttonState;

  switch (mode) {

    case 0:

      digitalWrite(LED1, LOW);

      digitalWrite(LED2, LOW);

      digitalWrite(LED3, LOW);

      break;

    case 1:

      digitalWrite(LED1, HIGH);

      digitalWrite(LED2, LOW);

      delay(300);

      digitalWrite(LED1, LOW);

      digitalWrite(LED2, HIGH);

      delay(300);

      break;

    case 2:

      digitalWrite(LED1, HIGH);

      digitalWrite(LED2, HIGH);

      digitalWrite(LED3, HIGH);

      break;

    case 3:

      for (fadeValue = 0; fadeValue <= 255; fadeValue += 5) {

        analogWrite(LED1, fadeValue);

        analogWrite(LED2, 255 - fadeValue);

        delay(20);

      }

      for (fadeValue = 255; fadeValue >= 0; fadeValue -= 5) {

        analogWrite(LED1, fadeValue);

        analogWrite(LED2, 255 - fadeValue);

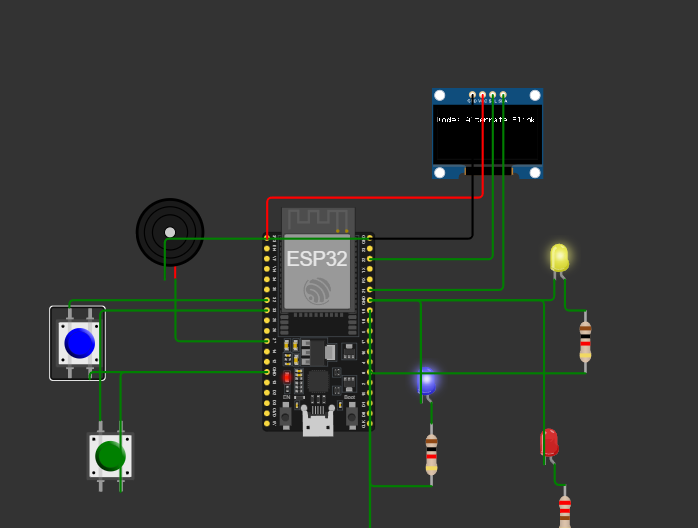
        delay(20);

      }

      break;

  }

}

****