YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

LAB-06

OBJECTIVE/ AIM	Interfacing ESP32 with Rotary Encoder and OLED Display.
SOFTWARE REQUIRED	Arduino IDE
CODE	

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <math.h>  // For trigonometric functions

// OLED display parameters
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
#define SCREEN_ADDRESS 0x3C  // I2C address for OLED
(usually 0x3C for 128x64)

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT,
&Wire, OLED_RESET);

// Rotary Encoder Pins for ESP32
#define CLK 18  // GPIO18 for Clock pin
#define DT 19   // GPIO19 for Data pin
#define SW 23   // GPIO23 for Button pin

int counter = 0;
int currentStateCLK;
int lastStateCLK;
int btnState;
String currentDir = "0";
unsigned long lastButtonPress = 0;
int X_CENTER = SCREEN_WIDTH / 2;
int Y_CENTER = SCREEN_HEIGHT - 1;
int x1, y1_pos, x2, y2;
double angle;
float radius = Y_CENTER;
String number[8] = {"0", "0", "100", "75", "50", "25",
"0", "0"};
int x = 0;
```

LAB-06

```
int k = 0;

void setup() {
  // Set up the rotary encoder pins
  pinMode(CLK, INPUT_PULLUP);
  pinMode(DT, INPUT_PULLUP);
  pinMode(SW, INPUT_PULLUP);

  Serial.begin(115200);

  // Initialize the OLED display
  if (!display.begin(SSD1306_SWITCHCAPVCC,
SCREEN_ADDRESS)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;) ; // Don't proceed, loop forever
  }

  lastStateCLK = digitalRead(CLK);
  Serial.print("Direction: ");
  Serial.print(currentDir);
  Serial.print(" | Counter: ");
  Serial.println(counter);

  display.clearDisplay();
  display.display();

  // Initial screen message
  display.setTextColor(SSD1306_WHITE);
  display.setTextSize(2);
  display.setCursor(20, 5);
  display.print("Teach Me");
  display.setCursor(10, 25);
  display.print("Something");
  display.setCursor(23, 45);
  display.print("Project");
  display.display();
  delay(1000);
  display.clearDisplay();
  draw();
  display.display();
```

```
}

void loop() {
  lastStateCLK = currentStateCLK;
  currentStateCLK = digitalRead(CLK);
  btnState = digitalRead(SW);

  if (currentStateCLK != lastStateCLK && currentStateCLK
== 1) {
    if (digitalRead(DT) != currentStateCLK) {
      k = 1;
      currentDir = "CW";
      x++;
    } else {
      k = 2;
      currentDir = "CCW";
      x--;
    }

    Serial.print("Direction: ");
    Serial.print(currentDir);
    Serial.print(" | Counter: ");
    Serial.println(x);
    return;
  } else if (k == 1 || k == 2) {
    if (x <= 0) x = 0;
    if (x >= 100) x = 100;
    k = 0;

    Serial.println(x);

    display.setTextSize(1);
    display.clearDisplay();
    draw();
    display.setTextSize(2);
    display.setCursor(55, 35);
    if (x < 10) display.print("0");
    display.print(x);

    int j = map(x, 0, 100, 270, 89);
```

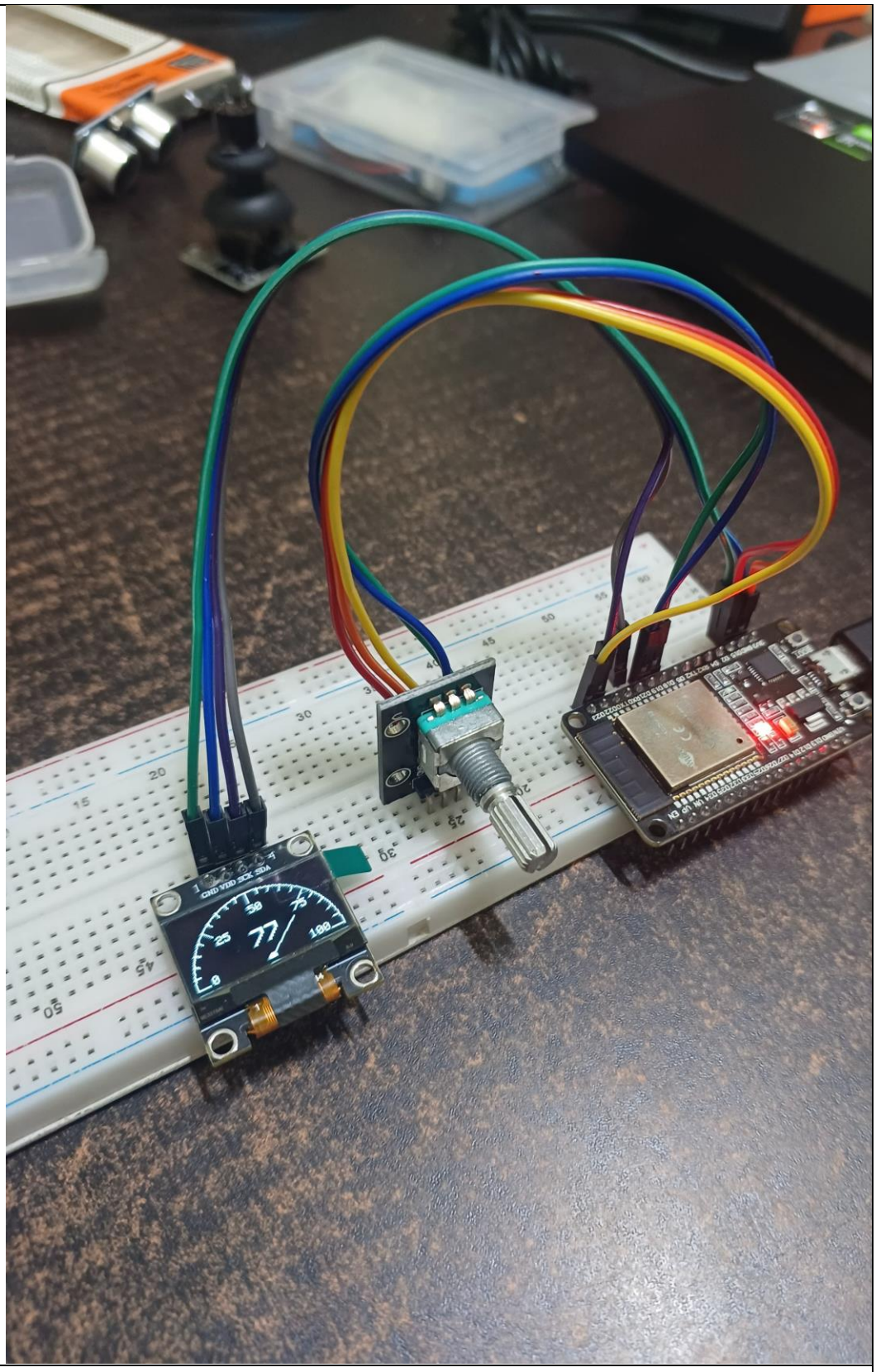
```
    angle = j * 0.0174533;  // Convert degrees to radians

    x1 = X_CENTER - 1;
    y1_pos = Y_CENTER;
    if (x >= 100) y1_pos = Y_CENTER - 1;

    x2 = X_CENTER + (sin(angle) * (radius - 25));
    y2 = Y_CENTER + (cos(angle) * (radius - 25));
    display.drawLine(x1, y1_pos, x2, y2, SSD1306_WHITE);
    display.fillCircle(X_CENTER, Y_CENTER, 3,
SSD1306_WHITE);
    display.display();
  }

  lastStateCLK = currentStateCLK;

  if (btnState == LOW) {
    if (millis() - lastButtonPress > 50) {
      Serial.println("Button pressed!");
      lastButtonPress = millis();
      x = 0;
      k = 1;
    }
  }

  delay(1);
}

void draw() {
  display.drawCircle(X_CENTER, Y_CENTER, radius,
SSD1306_WHITE);

  for (int j = 2; j <= 6; j++) {
    angle = j * 45;
    angle = angle * 0.0174533;  // Convert degrees to
radians

    x1 = X_CENTER + (sin(angle) * radius);
    y1_pos = Y_CENTER + (cos(angle) * radius);
    x2 = X_CENTER + (sin(angle) * (radius - 10));
```

```
    y2 = Y_CENTER + (cos(angle) * (radius - 10));
    display.drawLine(x1, y1_pos, x2, y2, SSD1306_WHITE);

    x2 = X_CENTER + (sin(angle) * (radius - 15));
    y2 = Y_CENTER + (cos(angle) * (radius - 15));

    // Place the number labels
    if (j == 6) display.setCursor(x2 - 2, y2 - 6);
    else if (j == 2) display.setCursor(x2 - 13, y2 - 6);
    else display.setCursor(x2 - 2, y2 - 2);
    display.print(number[j]);
  }

  display.fillCircle(X_CENTER, Y_CENTER, 3,
SSD1306_WHITE);

  // Draw additional small tick marks
  for (int j = 10; j <= 30; j++) {
    angle = j * 9;
    angle = angle * 0.0174533;  // Convert degrees to
radians

    x1 = X_CENTER + (sin(angle) * radius);
    y1_pos = Y_CENTER + (cos(angle) * radius);
    x2 = X_CENTER + (sin(angle) * (radius - 5));
    y2 = Y_CENTER + (cos(angle) * (radius - 5));
    display.drawLine(x1, y1_pos, x2, y2, SSD1306_WHITE);
  }
}
```

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

LAB-06

OUTPUT/ PHOTO	
RESULT	THE PROGRAM HAS BEEN EXECUTED SUCCESSFULLY