

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

OBJECTIVE/ AIM	Interface ESP32 with Joystick and 0.96' OLED Display.
SOFTWARE REQUIRED	Arduino IDE
CODE	<pre>#include &lt;Wire.h&gt; #include &lt;Adafruit_GFX.h&gt; #include &lt;Adafruit_SSD1306.h&gt;  // Define the OLED display dimensions and pins for ESP32 #define SCREEN_WIDTH 128 #define SCREEN_HEIGHT 64 #define OLED_RESET -1 // ESP32 doesn't use reset pin #define SDA_PIN 21 #define SCL_PIN 22  Adafruit_SSD1306 oled(SCREEN_WIDTH, SCREEN_HEIGHT, &amp;Wire, OLED_RESET);  #define RIGHT 0 #define LEFT 1 #define UP 2 #define DOWN 3  // Joystick pins #define pinX 34 #define pinY 35 #define pinSw 32 // Optional, for joystick button if you use it  int valueX = 0; int valueY = 0; unsigned char keyValue = 0;  // Snake game variables const uint8_t block[] PROGMEM = {   0xf0, //B11110000   0xb0, //B10110000   0xd0, //B11010000</pre>

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

```
0xf0 //B11110000
};

uint8_t snake_head_x = 4;
uint8_t snake_head_y = 4;
uint8_t x[100];
uint8_t y[100];
uint8_t snake_len = 2;
uint8_t snake_dir = RIGHT;

uint8_t food_x;
uint8_t food_y;
bool food_eaten = true;

bool game_over = false;
int score = 0;
int level = 1;
int snake_speed = 150;
int i;

// Joystick scanning function
void keyScan(void) {
    static unsigned char keyUp = 1;

    valueX = analogRead(pinX);
    valueY = analogRead(pinY);

    if (keyUp && ((valueX <= 1000) || (valueX >= 3000) ||
(valueY <= 1000) || (valueY >= 3000))) {
        delay(10);
        keyUp = 0;
        if (valueX <= 1000) {
            if (snake_dir != UP) {
                snake_dir = DOWN;
            }
        }
        else if (valueX >= 3000) {
            if (snake_dir != DOWN) {
                snake_dir = UP;
            }
        }
    }
}
```

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

```
    }
    else if (valueY <= 1000) {
        if (snake_dir != RIGHT) {
            snake_dir = LEFT;
        }
    }
    else if (valueY >= 3000) {
        if (snake_dir != LEFT) {
            snake_dir = RIGHT;
        }
    }
} else if ((valueX > 1000) && (valueX < 3000) && (valueY
> 1000) && (valueY < 3000)) {
    keyUp = 1;
}
}

void draw_snake(int x, int y) {
    oled.drawBitmap(x, y, block, 4, 4, 1);
}

void show_score(int x, int y, int data) {
    oled.setCursor(x, y);
    oled.println(data);
}

void screen(void) {
    oled.clearDisplay();
    oled.setTextSize(1);
    oled.drawRect(0, 1, 102, 62, 1);
    oled.drawRect(0, 0, 102, 64, 1);
    oled.setCursor(104, 12);
    oled.println("LEVE");
    oled.setCursor(104, 40);
    oled.println("SCOR");

    show_score(110, 25, level);
    show_score(110, 53, score);

    for (i = 0; i < snake_len; i++) {
```

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

```
        draw_snake(x[i], y[i]);
    }

    draw_snake(food_x, food_y);
    oled.display();
}

void draw_food(void) {
    int food_out = 0;

    if (food_eaten) {
        while (food_out == 0) {
            food_out = 1;
            food_x = (uint8_t)(random(4, 100) / 4) * 4;
            food_y = (uint8_t)(random(4, 60) / 4) * 4;

            for (int i = snake_len - 1; i > 0; i--) {
                if (food_x == x[i] && food_y == y[i]) {
                    food_out = 0;
                }
            }
        }
    }

    food_eaten = false;
}

void snake_move(void) {
    switch (snake_dir) {
        case RIGHT:
            snake_head_x += 4;
            break;
        case UP:
            snake_head_y -= 4;
            break;
        case LEFT:
            snake_head_x -= 4;
            break;
        case DOWN:
            snake_head_y += 4;
```

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

```
        break;
    }

    if ((snake_head_x == food_x) && (snake_head_y == food_y))
    {
        food_eaten = true;
        snake_len++;
        score++;
        level = score / 5 + 1;
        snake_speed -= level;
    }

    for (i = snake_len - 1; i > 0; i--) {
        x[i] = x[i - 1];
        y[i] = y[i - 1];
    }
    x[0] = snake_head_x;
    y[0] = snake_head_y;

    check_snake_die();
}

void draw_game_over() {
    oled.clearDisplay();
    oled.setTextSize(2);
    oled.setCursor(10, 10);
    oled.println("GAME OVER");
    oled.setTextSize(1);
    oled.setCursor(30, 35);
    oled.println("LEVE:");
    oled.setCursor(30, 55);
    oled.println("SCOR:");

    show_score(80, 35, level);
    show_score(80, 55, score);
    oled.display();
}

void check_snake_die(void) {
    if (snake_head_x < 4 || snake_head_x > 96 || snake_head_y
```

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

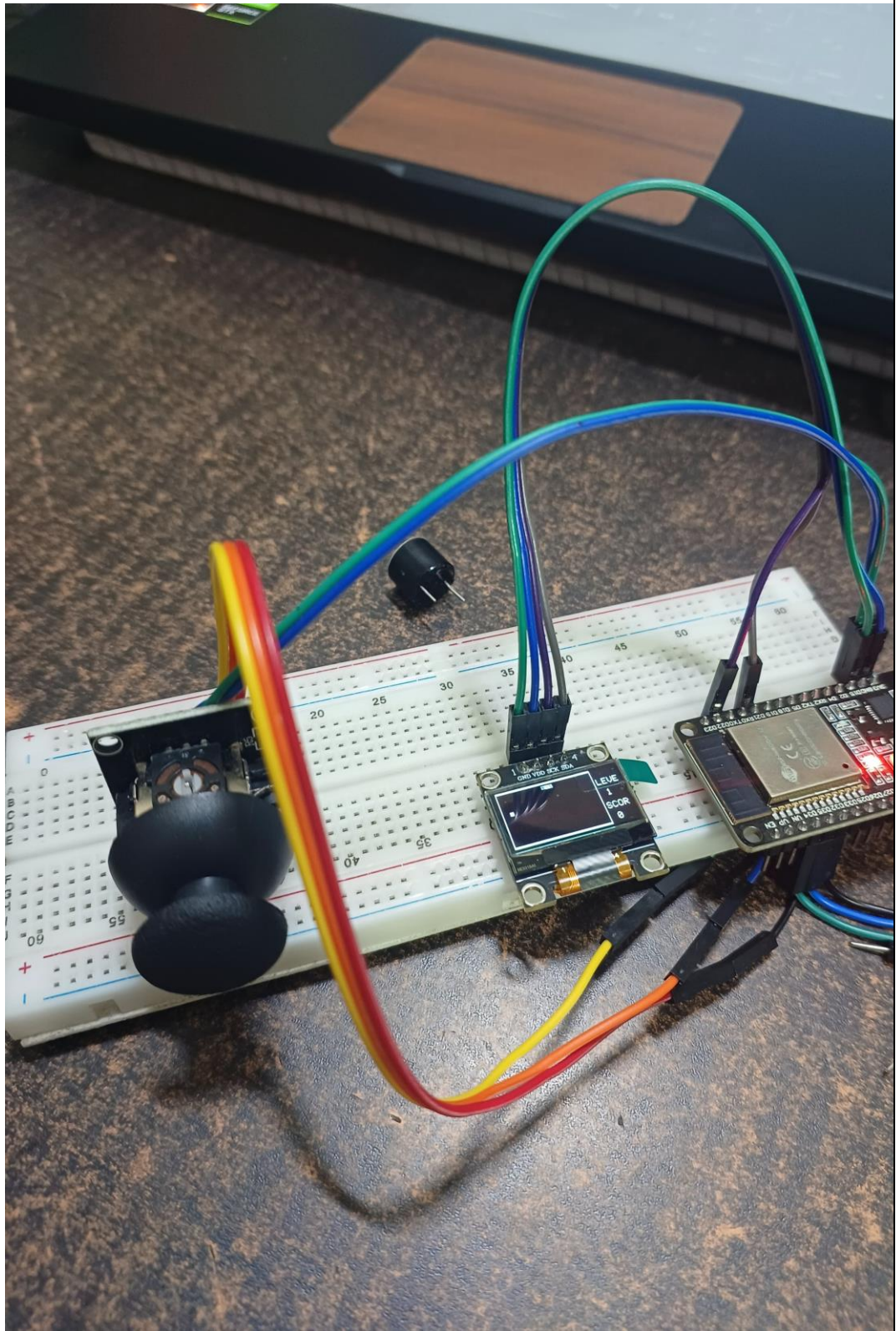
```
< 1 || snake_head_y > 56) {  
    game_over = true;  
}  
  
if (snake_len > 4) {  
    for (int i = 1; i < snake_len; i++) {  
        if (snake_head_x == x[i] && snake_head_y == y[i]) {  
            game_over = true;  
        }  
    }  
}  
}  
  
void setup() {  
    Wire.begin(SDA_PIN, SCL_PIN); // Initialize I2C for  
ESP32  
    oled.begin(SSD1306_SWITCHCAPVCC, 0x3C); // OLED I2C  
address  
    oled.setTextColor(WHITE);  
    randomSeed(analogRead(3)); // Initialize random seed  
}  
  
void loop() {  
    if (game_over) {  
        draw_game_over();  
    } else {  
        keyScan();  
        snake_move();  
        draw_food();  
        screen();  
    }  
    delay(snake_speed);  
}
```

# YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

IIOT 2404: Internet of Things Application Lab

## LAB-07

OUTPUT/  
PHOTO



RESULT

THE PROGRAM HAS BEEN EXECUTED SUCCESSFULLY