

Pseudocode for ESP32

START

//Initialize Pins for use in ESP32

```
#define ONE_WIRE_BUS 4      // Pin for DS18B20 temperature sensor
#define PELTIER_PIN 12     // PWM pin to control Peltier module
#define OLED_SDA 21        // I2C SDA pin for OLED
#define OLED_SCL 22        // I2C SCL pin for OLED
#define OLED_ADDR 0x3C     // OLED I2C address (commonly 0x3C or 0x3D)
```

//Rotary Encoder Pins

```
#define ENCODER_PIN 34
```

// PID variables

```
#define Setpoint 20.0 // Desired temperature
```

```
#define Input // Current temperature reading
```

```
#define Output // PID control output
```

```
#define Kp = 2.0, Ki = 5.0, Kd = 1.0 // PID constants (tune these)
```

```
#define integral = 0, previousError = 0
```

//Initialize functioning of ESP32

FUNCTION SETUP ESP32()

 //Initialize Serial Communication

 Serial.begin(115200)

 //Initialize PWM for Peltier

 //Initialize OLED Display

END FUNCTION

FUNCTION LOOP()

```

// Check encoder position and adjust Setpoint
RotaryValue = analogRead(Rotary_Encoder)
//Map to Temperature Range from 10 to 25 Celsius
setpoint = map(RotaryValue, 0, 4096, 10, 25)

// Get current temperature reading
Input = readTemperature()

// Run PID control if temperature is within range
IF Input >= MIN_TEMP AND Input <= MAX_TEMP THEN
    // Calculate PID error, integral, and derivative
    // Constrain PID output
    // Set Peltier PWM based on PID output
    // Print status to serial monitor
    // Display on OLED
ELSE
    // Safety shutdown if temperature out of bounds
    // Turn off Peltier
    // Print error message
END IF

// Delay between readings
END FUNCTION

FUNCTION readTemperature()
    //Request Temp. Measurement
    //Read temperature data
    //Calculate temperature in Celsius
END FUNCTION

FUNCTION RESETCOMMUNICATION()
    //Reset Pulse
    //delay
END FUNCTION

//Read and Write functions for Communication with sensor

```

```

FUNCTION WRITEBYTE()
    //Write the PWM Signal
END FUNCTION
FUNCTION READBYTE()
    //Read the PWM Signal
END FUNCTION

//I2C Functions for OLED
FUNCTION WRITEBYTE()
    //Write the data on the SCL and SDA pin
END FUNCTION

FUNCTION INIT()
    //START_I2C_COMMUNICATION
    //SET_COMMAND_MODE
    //TURN_OFF_DISPLAY
    //SET_DISPLAY_PARAMETERS
    //TURN_ON_DISPLAY
    //STOP_I2C_COMMUNICATION
END FUNCTION

FUNCTION CLEAR()
    //Clear the Data in the OLED by writing using the WRITEBYTE
END FUNCTION

FUNCTION PRINT()
    //START I2C
    //Initialize data mode
    //Send Text Data
    //Write the data
    //Stop I2C
END FUNCTION

```