

Diferenčný regulátor teploty

Vypracovala: Mgr. Iveta Piteľová

Predmet: Internet vecí

Diferenčný regulátor teploty je zariadenie, ktoré podľa nastavených parametrov ovláda svoj výstup na základe rozdielu teplôt z dvoch snímačov teploty T1 a T2. V tomto prípade je to PIN D23. Na tento PIN je pripojená LED dióda, ktorá signalizuje stav výstupu (ON / OFF). Snímače sú pripojené na PIN D4.

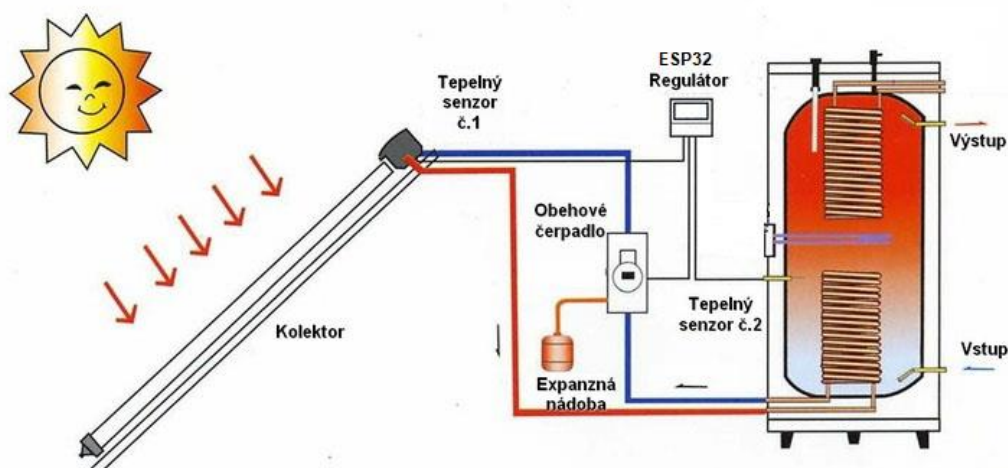
Teploty T1, T2, rozdiel teplôt a stav výstupu sú zobrazené na OLED displeji.

Rozdiel teplôt, pri ktorom je výstup v stave ON alebo OFF sa dá nastaviť. V tomto prípade ak $T1 - T2 > 3$ výstup je ON, ak $T1 - T2 < 1$ výstup je OFF.

Pri poruche niektorého zo snímačov je jeho stav hlásený na displeji.

Praktické využitie: Tento regulátor umožňuje po pridaní ďalšieho hardvéru (reléového modulu) efektívne riadenie teploty vody v zásobníku teplej vody (bojleri) v spolupráci s teplovodnými slnečnými kolektormi zapínaním a vypínaním obehového čerpadla.

Schéma zapojenia slnečných kolektorov



Použité komponenty

- 1x ESP32
- 2x snímač teploty DS18B20
- 1x OLED display 1,3", 128x64 bodov
- 1x rezistor 4,7 kΩ
- 1x rezistor 270 Ω
- 1x LED dióda
- Prepojovacie vodiče
- Napájací kábel USB C

Snímač DS18B20

Je to digitálny teplotný senzor, ktorý na komunikáciu využíva rozhranie One-Wire, čo znamená, že ho možno pripojiť k mikrokontroléru jediným dátovým vodičom, okrem napájania a uzemnenia.

DS18B20 má presnosť $\pm 0,5^{\circ}\text{C}$ v rozsahu -10°C až $+85^{\circ}\text{C}$. Dokáže merať teploty od -55°C do $+125^{\circ}\text{C}$ s nastaviteľným rozlíšením (typicky 9 až 12 bitov).

Každý snímač DS18B20 má jedinečnú 64-bitovú adresu. To nám umožní pripojiť viacero senzorov na rovnakú zbernicu One-Wire a jednoducho ich rozlíšiť. Sensor odosiela údaje o teplote digitálne, čo znižuje možnosť chýb alebo rušenia bežného v analógových senzorochoch.

Zapojenie DS18B20: červený vodič – Vcc (3,3V)
 čierny vodič – GND
 žltý vodič – PIN D4

OLED displej

Rozlíšenie: 128X64, pozorovací uhol: viac ako 160°, spotreba energie: 0,08W, rozsah napájania: DC 3V-5V, rozsah prevádzkových teplôt -30°C – 70°C, prevádzkový režim: I2C, OLED obrazovka 1,3“, interný čip: SH1106.

Zapojenie displeja: VDD – Vcc
 GND – GND
 SCK – PIN D22
 SDA – PIN D21

ESP32 Pinout

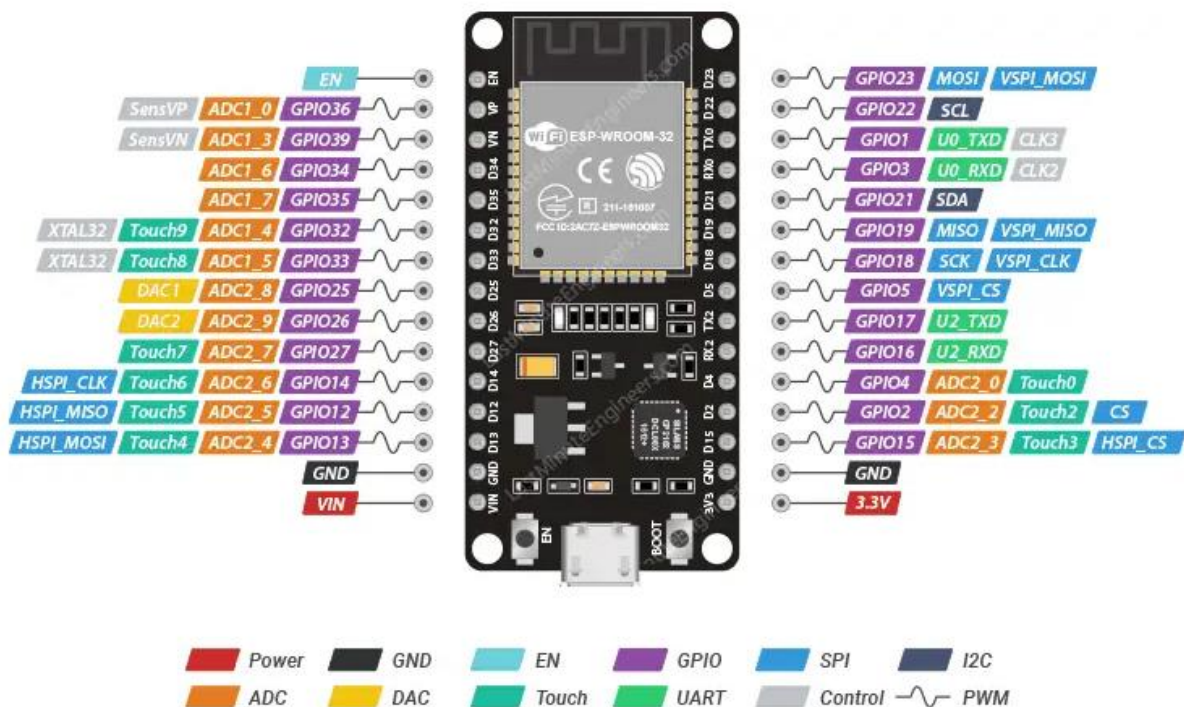
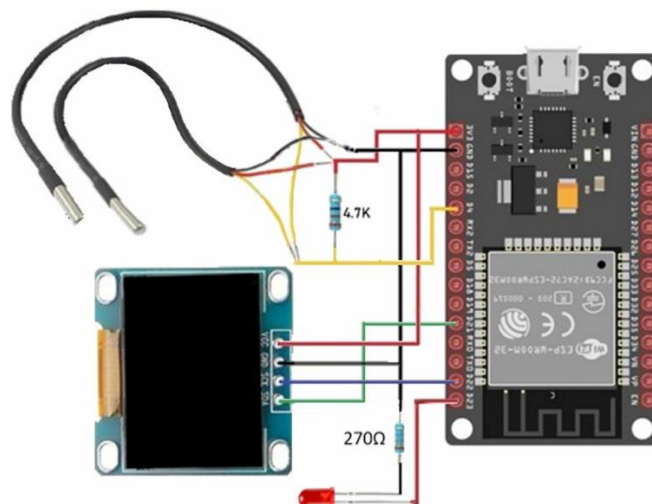
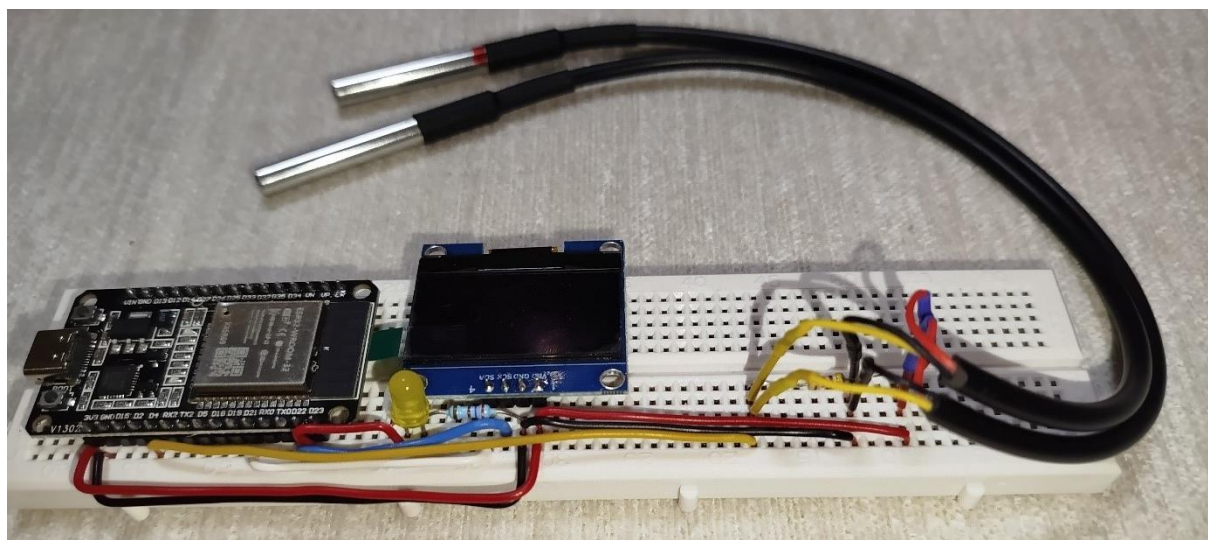


Schéma zapojenia



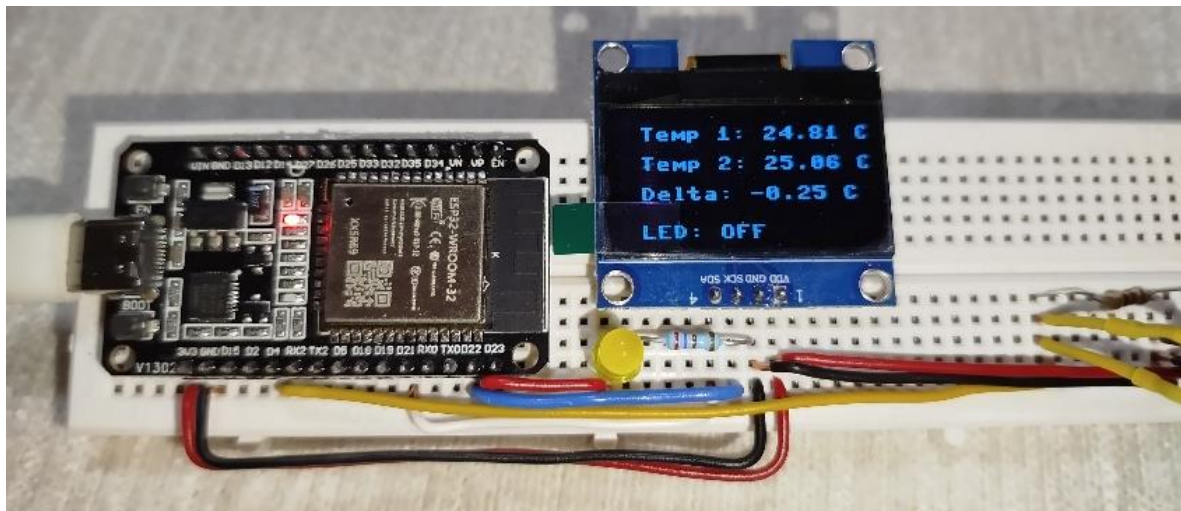
Zapojenie



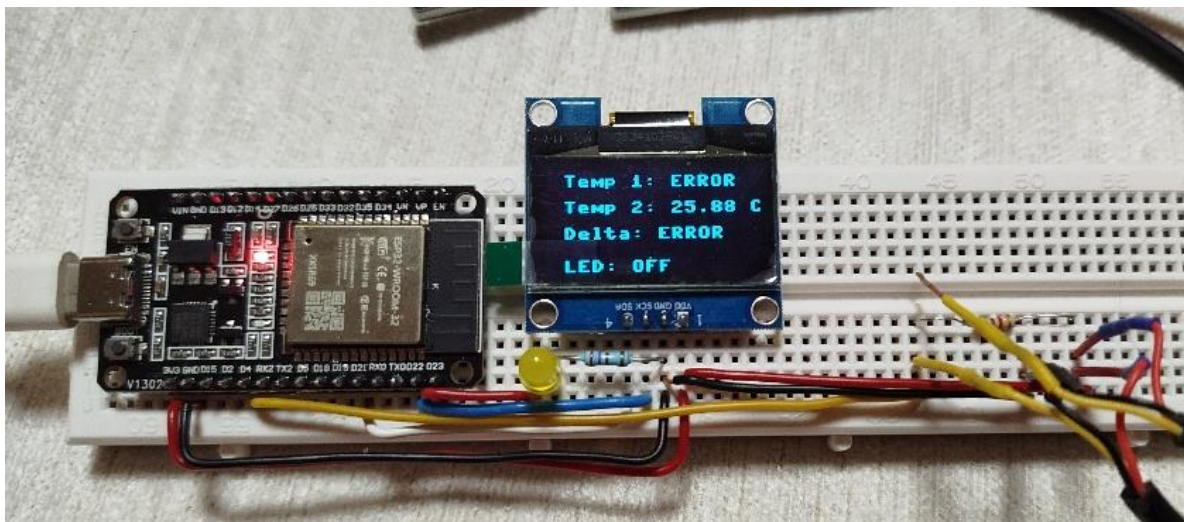
Prevádzkový stav: delta $T > 3^{\circ}\text{C}$ (LED zapnutá)



Prevádzkový stav: delta $T < 1^{\circ}\text{C}$ (LED vypnutá)



Prevádzkový stav: porucha snímača T1



Postup tvorby projektu

- Vytvoríme virtuálny COM port pre komunikáciu s ESP32 pomocou ovládača CP210x
- Nainštalujeme programovacie prostredie Thonny pre ESP32 v MicroPythone
- Zapojíme jednotlivé komponenty podľa schémy
- Pred spustením Thonny pripojíme ESP32 cez USB kábel k počítaču
- Pomocou Thonny nahráme firmware do ESP32, vytvoríme súbor boot.py
- Nahráme knižnicu sh1106 pre displej do ESP32 pomocou Thonny (sh1106.py)
- Vytvoríme script pre skenovanie adries snímačov (adresy.py) a uložíme do ESP32
- Spustíme script adresy.py pre získanie adries snímačov
- Vytvoríme hlavný program (main.py), adresy snímačov zapíšeme a uložíme do ESP32

adresy.py

```
from machine import Pin
import time
import onewire
import ds18x20

# Konfigurácia pinu pre OneWire
dat = Pin(4) # Teplotný senzor pripojený na GPIO4

# Inicializácia teplotného senzora DS18B20
ds = ds18x20.DS18X20(owewire.OneWire(dat))

# Unikátne adresy senzorov (získané skenovaním zbernice)
T1_address = b'\x28\xff\x4c\x6b\x93\x16\x03\x7c' #
T2_address = b'\x28\xff\x6a\x7c\x93\x16\x01\x23' #

# Zoznam všetkých senzorov na zbernici
roms = ds.scan()
print("Nájdene senzory:")
for rom in roms:
    print(''.join(f'{b:02X}' for b in rom))
```

Výpis adres snímačov



Thonny - Zariadenie MicroPython :: /adresy.py @ 1:2

Súbor Upraviť Zobrazíť Spustiť Nástroje Pomocník

<bez názvu> x [adresy.py] x

```
1 from machine import Pin
2 import time
3 import onewire
4 import ds18x20
5
6 # Konfigurácia pinu pre OneWire
7 dat = Pin(4) # Teplotný senzor pripojený na GPIO4
8
9 # Inicializácia teplotného senzora DS18B20
10 ds = ds18x20.DS18X20(owewire.OneWire(dat))
11
12 # Unikátne adresy senzorov (získané skenovaním zbernice)
13 T1_address = b'\x28\xff\x4c\x6b\x93\x16\x03\x7c' #
14 T2_address = b'\x28\xff\x6a\x7c\x93\x16\x01\x23' #
15
16 # Zoznam všetkých senzorov na zbernici
17 roms = ds.scan()
18 print("Nájdene senzory:")
19 for rom in roms:
20     print(''.join(f'{b:02X}' for b in rom))
21
```

Shell <

```
>>> %Run -c $EDITOR_CONTENT

MPY: soft reboot
Nájdene senzory:
287B6536000000FC
2843AC37000000EE
```

main.py

```
from machine import Pin, SoftI2C
import sh1106
import time
import machine
import onewire
import ds18x20

# Konfigurácia pinov pre I2C a OneWire
dat = machine.Pin(4)          # Teplotný senzor pripojený na GPIO4
i2c = SoftI2C(scl=Pin(22), sda=Pin(21), freq=100000) # Inicializácia I2C pre OLED display

# Inicializácia OLED displeja
lcd = sh1106.SH1106_I2C(128, 64, i2c) # Vytvorenie objektu pre LCD s rozmermi 128x64

# Inicializácia teplotného senzora DS18B20
ds = ds18x20.DS18X20(onewire.OneWire(dat)) # Vytvorenie OneWire objektu pre DS18B20
roms = ds.scan() # Vyhľadanie zariadení na zbernici

# Unikátne adresy senzorov (získané skenovaním zbernice)
T1_address = b'\x28\x7B\x65\x36\x00\x00\x00\xFC' # adresa senzora 287B6536000000FC
T2_address = b'\x28\x43\xAC\x37\x00\x00\x00\xEE' # adresa senzora 2843AC37000000EE

# Inicializácia LED na GPIO23 ako výstup
led = Pin(23, Pin.OUT)

# Premenná pre sledovanie stavu LED (či už bola zapnutá)
led_on = False

# Overenie, či sú pripojené oba senzory
# if T1_address not in roms:
#     print("Senzor T1 nie je pripojený!")
# if T2_address not in roms:
#     print("Senzor T2 nie je pripojený!")

# Hlavná slučka programu
try:
    while True:
        ds.convert_temp() # Spustenie merania teploty
        time.sleep_ms(750) # Čas potrebný na zmeranie teploty (min. 750 ms)

        # Čítanie teplôt pre pevne priradené senzory T1 a T2
        try:
            T1_temp = ds.read_temp(T1_address)
        except Exception as e:
            T1_temp = None

        try:
            T2_temp = ds.read_temp(T2_address)
        except Exception as e:
            T2_temp = None

        # Vyčistenie displeja
        lcd.fill(0)
```

```

# Zobrazenie teploty z oboch senzorov
if T1_temp is not None:
    lcd.text(f"Temp 1: {T1_temp:.2f} C", 10, 4)
else:
    lcd.text("Temp 1: ERROR", 10, 4)

if T2_temp is not None:
    lcd.text(f"Temp 2: {T2_temp:.2f} C", 10, 20)
else:
    lcd.text("Temp 2: ERROR", 10, 20)

# Výpočet rozdielu teplôt (delta temp) a jeho zobrazenie
delta_temp = None
if T1_temp is not None and T2_temp is not None:
    delta_temp = T1_temp - T2_temp
    lcd.text(f"Delta: {delta_temp:.2f} C", 10, 36)
else:
    lcd.text("Delta: ERROR", 10, 36)

# Logika pre LED a text na displeji
led_state = "OFF"
if delta_temp is not None:
    if abs(delta_temp) > 3: # Ak delta je väčšia ako 3°C
        if not led_on: # Ak LED ešte nie je zapnutá
            led.value(1) # Zapnúť LED
            led_on = True # Uložiť stav LED ako zapnutý
            led_state = "ON"
        elif abs(delta_temp) < 1: # Ak delta klesne pod 1°C
            if led_on: # LED sa už zapla predtým
                led.value(0) # Zhasnúť LED
                led_on = False # Uložiť stav LED ako vypnutý
            led_state = "OFF"
    else:
        led_state = "ON" if led_on else "OFF" # Zobrazenie aktuálneho stavu LED
# Zobrazenie stavu LED na displeji
lcd.text(f"LED: {led_state}", 10, 56)

# Aktualizácia displeja
lcd.show()
time.sleep(2) # Krátka pauza pred ďalším cyklom

except KeyboardInterrupt:
    led.value(0) # Zhasnutie LED pri prerušení

```

Použité zdroje (dostupné na internete):

Ovládač CP210X – ovládač na vytvorenie virtuálneho COM portu pre komunikáciu s ESP32

<https://www.silabs.com/developer-tools/usb-to-uart-bridge-vcp-drivers?tab=downloads>

THONNY – programovacie prostredie pre ESP32 v MicroPythone

<https://github.com/thonny/thonny/releases/tag/v4.1.6>

Knižnica pre display

<https://github.com/robert-hh/SH1106/blob/master/sh1106.py>

Návod na zapojenie DS18B20, displeja, micropython

<https://how2electronics.com/micropython-ds18b20-temperature-sensor-esp32/>

Datasheet interného čipu displeja SH1106

https://cdn.velleman.eu/downloads/29/infosheets/sh1106_datasheet.pdf

Návod na použitie displeja

https://robu.in/wp-content/uploads/2022/04/1.3inch_OLED_UserManual.pdf

Datasheet snímača teploty DS18B20

<https://www.analog.com/media/en/technical-documentation/data-sheets/DS18B20-PAR.pdf>

<https://www.quick-teck.co.uk/Management/EEUploadFile/1420644897.pdf>