



# Závěrečný projekt MIT

Název projektu

MĚŘÍCÍ STANICE

Číslo projektu

1

Zadání:

- Na univerzální desce plošných spojů sestrojte zařízení, které bude přijímat teploty přes I2C sběrnici
- Vytvoření funkční aplikace s programem pro vybraný mikrokontroler
- Vytvoření shield modulu pro desku DeroBoard
- Vytvoření prezentace (dle pravidel), odevzdání tisknuté dokumentace a zaslání emailu s kompletním výpisem programu nebo zasláním odkazu na repositář na GitHubu

Použité periferie:

- 1x LCD1602 – zobrazovací periferie (sběrnice I2C)
- 1x PCF8574 – převodník na I2C pro zobrazovací periferii
- 3x CJMCU-75 – teploměr (sběrnice I2C)

Použitý SOFTWARE:

- Microsoft Word 2016 (verze č. 2302 Build 16.0.16130.20186)
- ST Visual Programmer (verze č. ST-068)
- Visual Studio Code (verze č. March 2023 (version 1.77))

Poř. č.

16

Příjmení a jméno

OBLOUK Petr

Třída

4.A

Školní rok

2022/23

Datum vypracování

5.4.2023

Datum odevzdání

13.4.2023

Počet listů

5

Klasifikace

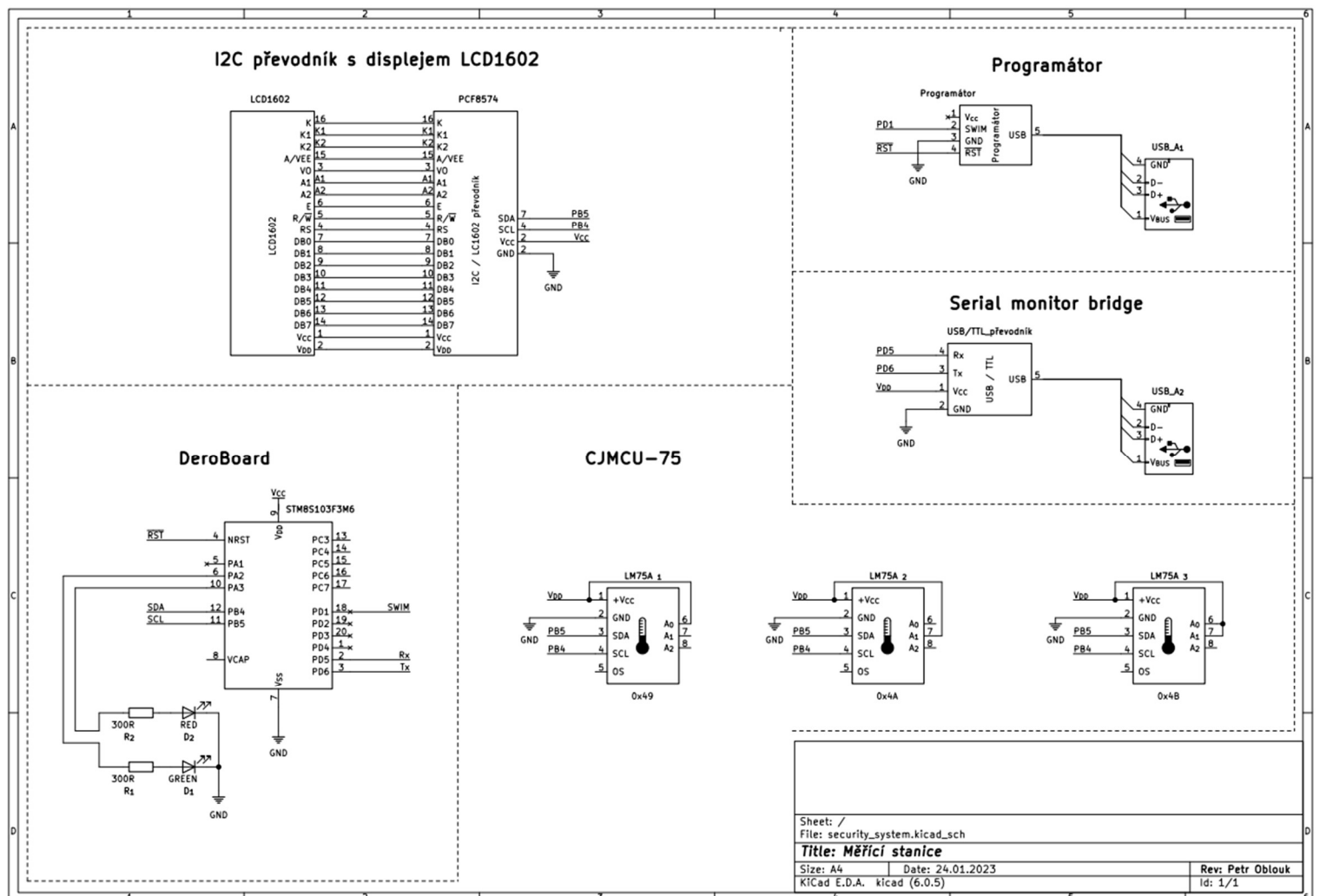
## 1. Slovní popis

Po zapnutí proběhne inicializace displeje, který má adresu 0x27, pokud inicializace proběhne pořádku, na displeji zhasne první řada a objeví se „Teplota“. Poté by mělo proběhnout měření teploty pomocí snímače CJMCU-75 na adrese 0x49 poté následovně po 3 sekundách by mělo následovat čtení z teploměru na adrese 0x4A a opět po 3 sekundách měření na adrese 0x4B. Zároveň každé čtení teploty se zobrazí na displeji.

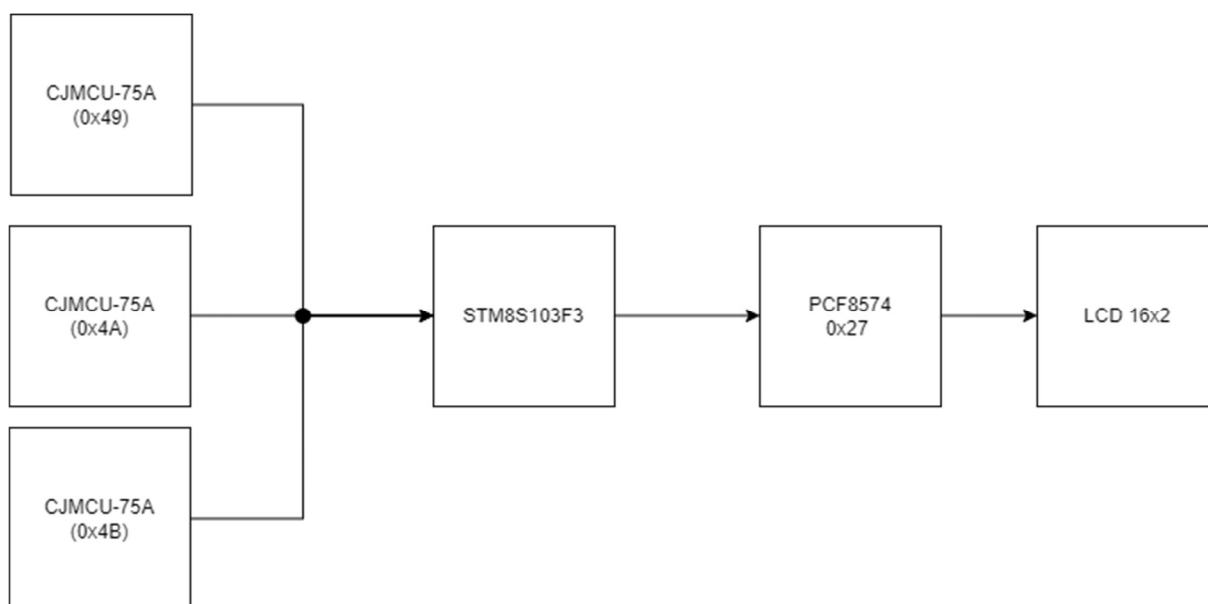
## 2. Periférie

- 1x LCD1602
  - zobrazovací periférie (sběrnice I2C)
  - má 2 řádky a 16 míst
- 1x PCF8574
  - převodník na I2C pro zobrazovací periférii
  - adresa I2C je 0x27
- 3x CJMCU-75
  - teploměr (sběrnice I2C)
  - adresa I2C je 0x49, 0x4A, 0x4B
  - schopný měřit s přesností:
    - - 25°C až 100°C:  $\pm 2^\circ\text{C}$
    - - 55°C až 125°C:  $\pm 3^\circ\text{C}$

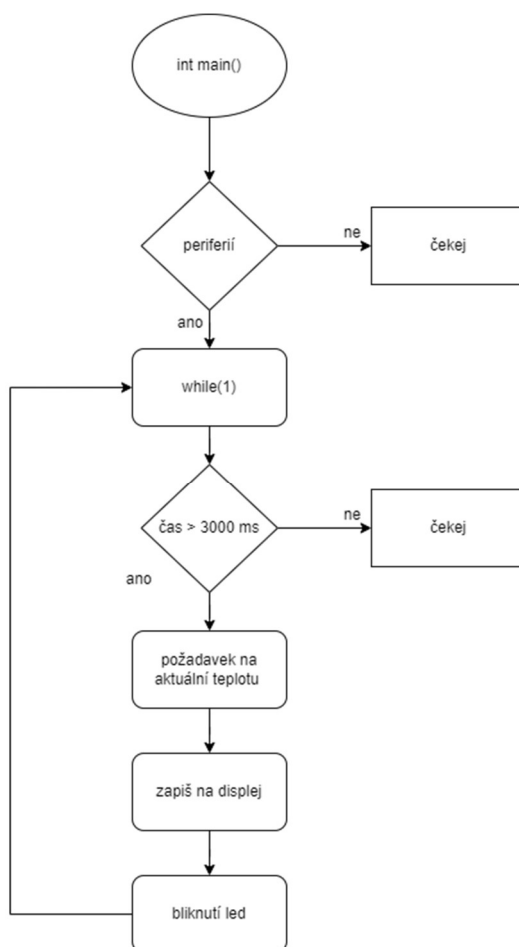
## 3. Schéma zapojení



#### 4. Blokové schéma



#### 5. Vývojový diagram kódu



## 6. Blok kódu

```
1 #include "stm8s.h"
2 #include "delay.h"
3 #include "LCD_I2C.h"
4 #include "milis.h"
5 #include "lm75a.h"
6 #include "uart_bridge.h"
7 ///////////////////////////////////////////////////
8 //! Makra
9 // Indikační LED
10 #define LED_PORT GPIOA
11 #define LED_PIN_GREEN GPIO_PIN_2
12 #define LED_PIN_RED GPIO_PIN_3
13 #define BUILD_IN_LED GPIOC, GPIO_PIN_4
14
15 #define TEPLOMER1 0x49
16 #define TEPLOMER2 0x4A
17 #define TEPLOMER3 0x4B
18
19 ///////////////////////////////////////////////////
20 //! Uživatelské funkce
21 void setup(void)
22 {
23     CLK_HSIPrescalerConfig(CLK_PRESCALER_HSIDIV1); // Předdělička DIV1
24     delay_init(); // Inicializace časovače TIM4
25     init_milis(); // Inicializace millis TIM2
26     GPIO_Init(LED_PORT, LED_PIN_RED, GPIO_MODE_OUT_PP_LOW_SLOW); // Pin LED RED
27     LCD_I2C_Init(0x27, 16, 2); // Inicializace LCD
28     LCD_I2C_Print("Inicializace..."); // Úvodní obrazovka na displej
29     LM75A_Init(TEPLOMER1, TEPLOMER2, TEPLOMER3); // Inicializace teploměrů
30     GPIO_Init(BUILD_IN_LED, GPIO_MODE_OUT_PP_LOW_SLOW); // Inicializace LED
31     GPIO_Init(LED_PORT, LED_PIN_GREEN, GPIO_MODE_OUT_PP_LOW_SLOW); // PIN Inicializace proběhla úspěšně
32     GPIO_WriteHigh(LED_PORT, LED_PIN_RED); // Konec inicializace
33     delay_ms(1500); // Počkej 1,5 s
34     GPIO_WriteHigh(LED_PORT, LED_PIN_GREEN); // Zhasni LED
35     LCD_I2C_Clear(); // Vyčisti displej
36 }
37 ///////////////////////////////////////////////////
38 //! Main program loop
39 int main(void)
40 {
41     //? Lokální proměnné
42     uint8_t temperature_data[2]; // Proměnná pro uložení teploty
43     uint16_t mtime_key = 0; // Proměnná pro millis
44     uint8_t adresy[3] = {TEPLOMER1, TEPLOMER2, TEPLOMER3}; // Proměnná pro skladování adres
45     uint8_t cislo = 0; // Proměnná pro scrolling čísla
46     char buffer[48]; // Proměnná pro zápis na displej
47     setup(); // Inicializace všech periférií
48     LCD_I2C_SetCursor(0, 0); // Nastavení kurzoru
49     LCD_I2C_Print("Teplota :"); // Úvodní obrazovka na displej
50     while (1)
51     {
52         if ((get_milis() - mtime_key) > 3000) // každých 1500 ms
53         {
54             mtime_key = get_milis(); // Milis now
55             LM75A_ReadTemperature(adresy[cislo], temperature_data); // Čtení teploty
56             GPIO_WriteReverse(BUILD_IN_LED); // Rozsviť BUILD_IN_LED
57             delay_ms(20); // Počkej 20 ms
58             GPIO_WriteReverse(BUILD_IN_LED); // Zhasni BUILD_IN_LED
59             LCD_I2C_SetCursor(0, 1); // Nastavení kurzoru
60             sprintf(buffer, "T%d = %d.%d C", cislo + 1, temperature_data[0], temperature_data[1]); // Zformátování stringu
61             LCD_I2C_Print(buffer); // Vytiskni na displej
62             cislo++; // Přičti k proměnné
63             if (cislo >= 3) // Přetečení teploměrů
64             {
65                 cislo = 0;
66                 GPIO_WriteLow(LED_PORT, LED_PIN_RED);
67                 delay_ms(50);
68                 GPIO_WriteHigh(LED_PORT, LED_PIN_RED);
69             }
70             if (temperature_data[0] >= 20) // Pokud je teplota větší jak 20 stupňů tak rozsviť LED
71             {
72                 GPIO_WriteLow(LED_PORT, LED_PIN_GREEN);
73             }
74             else
75             {
76                 GPIO_WriteHigh(LED_PORT, LED_PIN_GREEN);
77             }
78         }
79     }
80 }
81 ///////////////////////////////////////////////////
```

## **7. Závěr**

Projekt se mi zprvu nedařilo vůbec rozjet, měl jsem problémy se zobrazení na displeji a zároveň jsem navrhoval desku. Dost práce mi dalo samotnou desku připravit na pájení a pak si vyhrát se spojení. Zároveň jsem tiskl maličkosti na 3D tiskárně, které tak doplnily celkový vzhled projektu. Měření teploty se mi podařilo velice rychle zprovoznit, vše bylo krásně popsáno a měl jsem k dispozici logický analyzátor, který bych všem vřele doporučil.