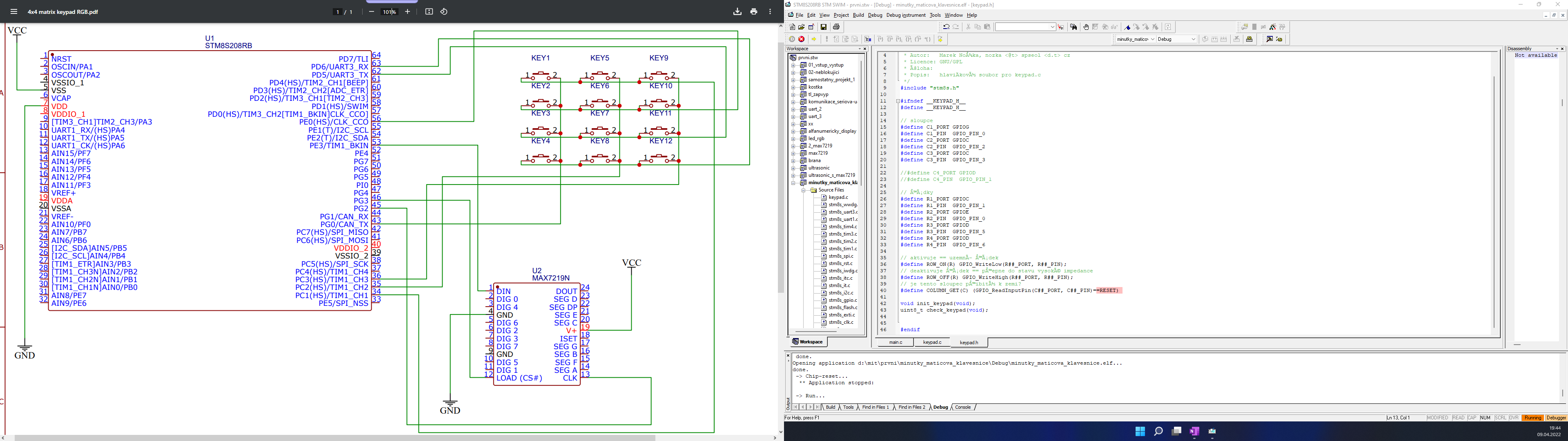
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Vyšší odborná škola a Střední průmyslová škola elektrotechnickáBožetěchova 3, OlomoucMikroprocesorová technika | | | | | | | | | | | | | | | | | |
| **ZÁVĚREČNÝ PROJEKT MIT** | | | | | | | | | | | | | | | | | |
| **Obsah**  [**Schéma ZAPOJENÍ** 2](#_Toc100429900)  [**SLOVNÍ POPIS ZAPOJENÍ** 2](#_Toc100429901)  [**BLOKOVÉ SCHÉMA ZAPOJENÍ** 2](#_Toc100429902)  [**SLOVNÍ POPIS FUNKCE PROGRAMU** 3](#_Toc100429904)  [**VÝVOJOVÝ DIAGRAM PROGRAMU** 4](#_Toc100429905)  [**ZÁVĚR** 5](#_Toc100429907)  [**UKÁZKA PROGRAMU** 6](#_Toc100429909) | | | | | | | | | | | | | | | | | |
| Poř. č. | Příjmení a jméno | | |  | | | | | | Třída | | Skupina | | Školní rok | |  | |
| 14 | PŘIKRYL Jan | | | | | | | | | 4B | | 1 | | 2021/22 | | | |
| Datum zpracování | |  | Datum odevzdání | |  | Počet listů | |  | Klasifikace | | | | | | | |
| 9. 4. 2022 | | | 12. 4. 2022 | | | 8 | | | Textová část | | Obhajoba | | | | Funkčnost | |
| Protokol o měření obsahuje: | | | | | | | teoretický úvod | | | | | | tabulky | | | | |
|  | | | | | | | schéma | | | | | | příklad výpočtu | | | | |
|  | | | | | | | použité přístroje | | | | | | grafy | | | | |
|  | | | | | | | postup měření | | | | | | závěr | | | | |

**Schéma ZAPOJENÍ**

**Schéma č.1:** Zapojení klávesnice a displeje na STM8



**SLOVNÍ POPIS ZAPOJENÍ**

Maticová klávesnice je připojena přes tyto piny k STM8: PG0, PC2, PC3 – jako sloupce tlačítek. Řádky jsou připojeny přes tyto piny: PC1, PE0, PD5, PD6.

Jako výstup je mimo komunikaci s PC (UART) zapojen sedmi segmentový displej max7219. Ten je připojen přes piny PG2, PG3 a PE3.

Displej je napájen pěti volty (pin VCC), uzemněn GND.

**BLOKOVÉ SCHÉMA ZAPOJENÍ**

Max7219

Maticová klávesnice

STM 8

UART 1

(komunikace s PC)

**SLOVNÍ POPIS FUNKCE PROGRAMU**

Nejprve se importují všechny potřebné knihovny (to jsou milis.h, stdio.h, stm8s.h, assert.h, keypad.h, delay.h). Následuje založení maker pro piny, porty a stavy displeje a všechny potřebné funkce (makra a funkce pro fungování klávesnice jsou v knihovně keypad.h).

Poté se vytvoří funkce pro UART komunikaci, inicializaci (taktování procesoru na 16 MHz, rozběhnutí časovače milis, povolení komunikace s počítačem a nastavení displeje).

Ve funkci main (hlavní funkce každého programu v jazyce C) jsou založeny proměnné pro minulé a aktuální zmáčknuté tlačítko a pozici.

Poté se zavolá inicializační funkce a vypíše se uvítací hláška.

Poté následuje funkce while (1), což způsobí nekonečný cyklus opakování.

Opakovat se bude následující: Každých 55 ms se uloží aktuální čas milis.

Zkontroluje se stisknuté tlačítko. Nerovná se minulému stisknutému tlačítko, bylo tedy stisknuto, vypíše se do PC stisknutá klávesa.

Byla-li stisknuta klávesa \* (netisknutelný znak), na displej se vypíše prázdné místo.

Byla-li stisknuta klávesa # (opět netisknutelný znak), všechny pozice na displeji se nastaví na prázdné místo a pozice se nastaví na 8 (první místo displeje).

Byla-li stisknuta jakákoliv jiná klávesa, vypíše se na pozici na místě pozice.

Poté se do proměnné pozice přičte 1. Je-li toto číslo 0, nastaví se opět na 8.

Nakonec se minule stisknutá klávesa bude rovnat aktuálně stisknuté klávese.

**VÝVOJOVÝ DIAGRAM PROGRAMU**

START

Importuj všechny potřebné knihovny.

Vytvoř makra pro piny a porty displeje a jeho funkce.

Vytvoř funkce pro komunikaci s PC, pro komunikaci s displejem a nastavení prázdného místa na displej.

Povol taktování mikroprocesoru na 16 MHz.

Inicializuj milis, UART a keypad.

Pravda

KONEC

-

+

Vypiš stisknutou klávesu v PC a na displeji, odečti 1 od pozice

mtime\_key = milis();

key\_now = check\_keypad();

Definuj

-

Uběhlo 55 ms

-

+

-

+

key\_last == 0Xff

&&

key\_now != 0xFF

-

+

pozice == 0

key\_now = 0xFF

key\_last = 0xFF

mtime\_key = 0

pozice = 8;

pozice = 0

key\_last = key\_now

Vypiš uvítací hlášku

**ZÁVĚR**

Shrnutí a zhodnocení

* Vytvořil jsem program na zaznamenávání čísel (poznámkový blok) s STM8. Čísla jsou zadávána na maticové klávesnici. Čísla jsou zobrazována na displeji max7219 a pro přehlednost i v počítači, což se hodí v případě absence či poškození displeje).
* Program jsem psal v prostředí ST Visual Develop.

Výhody

* Výstup je zobrazen na dvou místech.
* Displej se dá celý vymazat.

Nevýhody

* Nelze vkládat a zobrazovat písmena.

Co jsem se naučil a v čem to pro mě mělo přínos?

* Naučil jsem se používat maticovou klávesnici.

**UKÁZKA PROGRAMU**

#include "stm8s.h"

#include <stdio.h>

#include "assert.h"

#include "milis.h"

#include "keypad.h"

#include "delay.h"

*//Max7219*

#define CLK\_PORT GPIOG

#define CLK\_PIN GPIO\_PIN\_2

#define CS\_PORT GPIOG

#define CS\_PIN GPIO\_PIN\_3

#define DIN\_PORT GPIOE

#define DIN\_PIN GPIO\_PIN\_3

#define SET(BAGR) GPIO\_WriteHigh(BAGR##\_PORT, BAGR##\_PIN)

#define CLR(BAGR) GPIO\_WriteLow(BAGR##\_PORT, BAGR##\_PIN)

#define NOOP 0

#define DIGIT0 1

#define DIGIT1 2

#define DIGIT2 3

#define DIGIT3 4

#define DIGIT4 5

#define DIGIT5 6

#define DIGIT6 7

#define DIGIT7 8

#define DECODE\_MODE 9

#define INTENSITY 10

#define SCAN\_LIMIT 11

#define SHUTDOWN 12

#define DISPLAY\_TEST 15

*// argumenty pro SHUTDOWN*

#define DISPLAY\_ON 1 // zapne displej

#define DISPLAY\_OFF 0 // vypne displej

*// argumenty pro DISPLAY\_TEST*

#define DISPLAY\_TEST\_ON 1 // zapne test displeje

#define DISPLAY\_TEST\_OFF 0 // vypne test displeje

*// argumenty pro DECODE\_MOD*

#define DECODE\_ALL 0b11111111

#define DECODE\_NONE 0

*//UART komunikace*

char putchar (char c)

{

*/\* Write a character to the UART1 \*/*

UART1\_SendData8(c);

*/\* Loop until the end of transmission \*/*

while (UART1\_GetFlagStatus(UART1\_FLAG\_TXE) == RESET);

return (c);

}

char getchar (void) *//funkce čte vstup z UART*

{

int c = 0;

while (UART1\_GetFlagStatus(UART1\_FLAG\_RXNE) == RESET);

c = UART1\_ReceiveData8();

return (c);

}

*//Povoleni UART1*

void init\_uart1(void)

{

UART1\_DeInit(); *// smazat starou konfiguraci*

UART1\_Init((uint32\_t)115200, *//Nova konfigurace*

UART1\_WORDLENGTH\_8D,

UART1\_STOPBITS\_1,

UART1\_PARITY\_NO, UART1\_SYNCMODE\_CLOCK\_DISABLE, UART1\_MODE\_TXRX\_ENABLE);

}

void max7219(uint8\_t address, uint8\_t data)

{

uint16\_t mask;

CLR(CS);

mask = 1<<7;

while(mask){

CLR(CLK);

if(address & mask){

SET(DIN);

} else {

CLR(DIN);

}

SET(CLK);

mask >>=1;

CLR(CLK);

}

mask = 1<<7;

while(mask){

CLR(CLK);

if(data & mask){

SET(DIN);

} else {

CLR(DIN);

}

SET(CLK);

mask >>=1;

CLR(CLK);

}

SET(CS);

}

void nic(void) { *//Vypsání prázdných míst*

max7219(1, 15);

max7219(2, 15);

max7219(3, 15);

max7219(4, 15);

max7219(5, 15);

max7219(6, 15);

max7219(7, 15);

max7219(8, 15);

}

void setup(void)

{

CLK\_HSIPrescalerConfig(CLK\_PRESCALER\_HSIDIV1);*// taktovat MCU na 16MHz* init\_milis(); *//Initializace milis*

init\_uart1(); *//Povoleni komunikace s PC*

init\_keypad(); *//Initializace klávesnice*

*//max7219*

GPIO\_Init(CLK\_PORT, CLK\_PIN, GPIO\_MODE\_OUT\_PP\_LOW\_SLOW);

GPIO\_Init(CS\_PORT, CS\_PIN, GPIO\_MODE\_OUT\_PP\_LOW\_SLOW);

GPIO\_Init(DIN\_PORT, DIN\_PIN, GPIO\_MODE\_OUT\_PP\_LOW\_SLOW);

max7219(DECODE\_MODE, DECODE\_ALL);

max7219(SCAN\_LIMIT, 7);

max7219(INTENSITY, 3);

max7219(DISPLAY\_TEST, DISPLAY\_TEST\_OFF);

max7219(SHUTDOWN, DISPLAY\_ON); *// zapneme displej*

nic();

}

int main(void)

{

uint32\_t mtime\_led = 0;

uint8\_t key\_now = 0xFF;

uint8\_t key\_last = 0xFF;

uint32\_t mtime\_key = 0;

uint8\_t pozice = 8;

setup();

printf("Dobry den,**\n\r**");

printf("vitejte v programu poznamkovy blok.**\n\r**");

printf("Tlacitky klavesnice muzete vypisovat postupne na jednotlive segmenty cisla.**\n\r**");

printf("Tlacitkem \* napisete prazdne misto, tlacitkem # napisete vsude prazdna mista a kurzor posunete na zacatek.**\r\n**");

while (1) {

if (milis() - mtime\_key > 55) {*// detekce stisknuté klávesy*

mtime\_key = milis();

key\_now = check\_keypad();

if (key\_last == 0xFF && key\_now != 0xFF) {

char x[2];

sprintf(x, "%x", key\_now);

printf("Klavesa: %c**\n\r**", x[0]);

if (x[0] == 'a') { max7219(pozice, 15);

}else if (x[0] == 'b') {

nic();

pozice = 9;

}else {

max7219(pozice, x[0]);

}

pozice = pozice - 1;

if(pozice == 0) {

pozice = 8;

}

}

key\_last = key\_now;

}

}

}