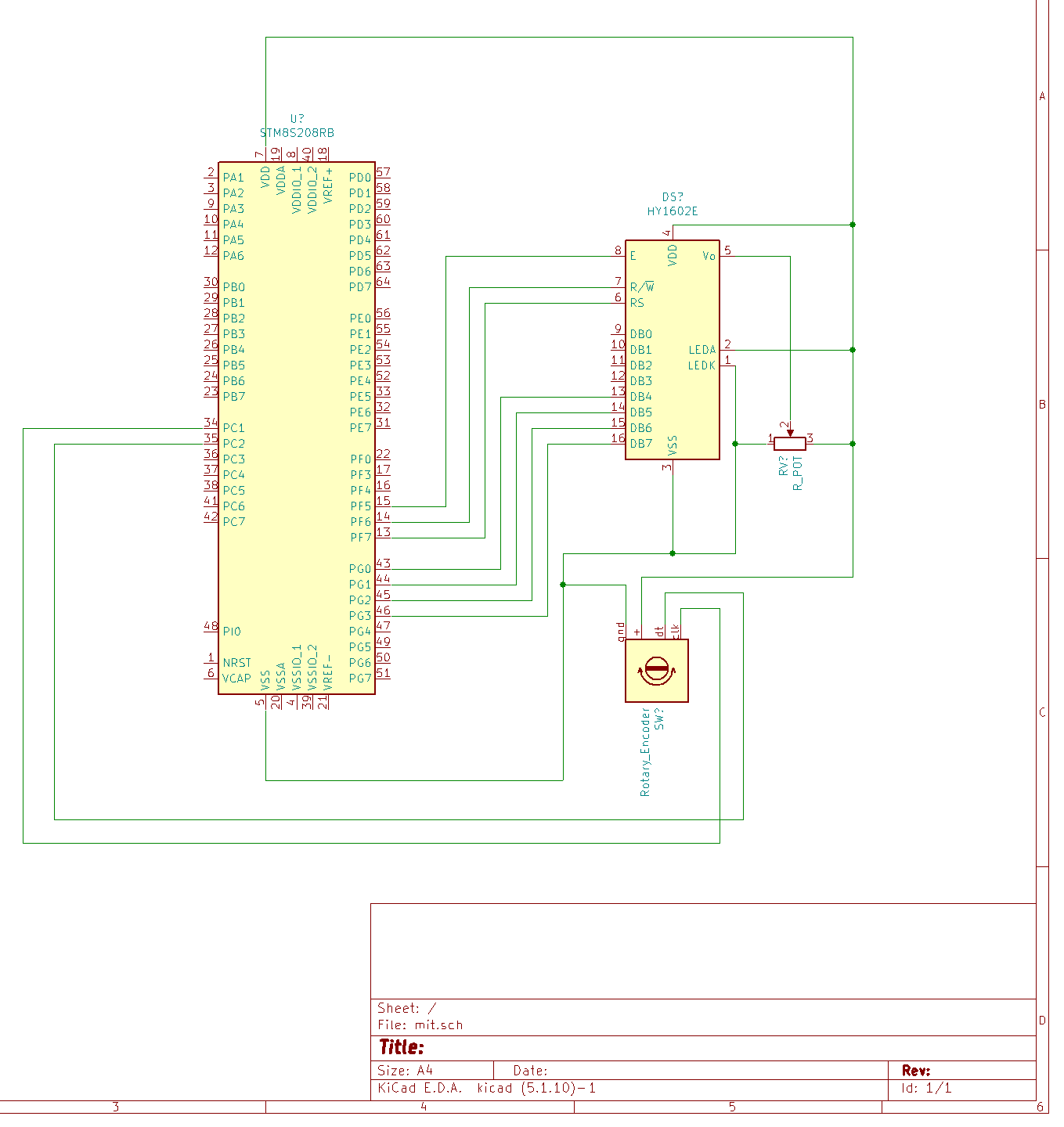
| Vyšší odborná škola a Střední průmyslová škola elektrotechnická  Božetěchova 3, Olomouc  Laboratoře elektrotechnických měření | | | | | | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | | | | | | | | |
| Název úlohy | |  | | | | | | | | | | | | | Číslo úlohy | |  |
| **Stroboskop** | | | | | | | | | | | | | | | 15 | | |
| Zadání:  Program a zapojení díky kterému bude led dioda na stm8 blikat různou rychlostí. Ovládání rychlosti led diody bude pomocí rotačního n-coderu. Perioda rychlosti blikání se bude zobrazovat na zeleném alfanumerickém displeji. | | | | | | | | | | | | | | | | | |
| Poř. č. | Příjmení a jméno | | | |  | | | | | Třída | Skupina | | | | Školní rok | |  |
|  | URBÁNEK Lukáš | | | | | | | | | 4B | 3 | | | | 2021/22 | | |
|  | | |  | Datum odevzdání | |  | Počet listů | |  | Klasifikace | | | | | | | |
|  | | | | 1.3.2022 | | |  | | |  | |  | |  | |  | |
|  | | | | | | | |  | | | | |  | | | | |
|  | | | | | | | |  | | | | |  | | | | |
|  | | | | | | | |  | | | | |  | | | | |
|  | | | | | | | |  | | | | |  | | | | |

**Schéma zapojení**



**Slovní popis zapojení**

Piny displeje D4 - D7 jsou připojeny na piny na stm PG0 - PG3. Displej je připojený taky na napájení a je nachystáno i připojení napájení podsvícení displeje, občas mi ale čitelnost displeje přišla lepší bez podsvícení. Pin RS je připojený na PF7, RW na PF6 a E na PF5. Rotační encoder je připojený následovně: CLK je připojený na stm na PC1, DT na PC2, dále pak pouze napájení. Vše napájím přes náš vývojový kit.

**Slovní popis funkce programu**

Pomocí encoderu měníme proměnou která mění rychlost blikání led diody. Perioda se poté zobrazuje na displeji

**Zhodnocení**

Největší přínos pro mě měl tenhle projekt v tom že jsem si uvědomil jak moc špatně na tom jsem, nešlo mi to, stále mi něco nefungovalo, častokrát jsem se přistihl jak uprostřed noci sháním někoho z mých chytřejších spolužáků aby mi s něčím pomohl, naštěstí naše třída nikdy nespí a všichni moji spolužáci jsou velice ochotní. Bohužel narozdíl třeba od programování mě práce na projektu nikam neposunula, nic nového jsem se nenaučil a začínám být velmi naštvaný sám na sebe protože nejsem schopný se do problematiky mikroprocesorové techniky více ponořit a nedaří se mi ji pochopit.

Nevím jak mám shrnout výhody a nevýhody, jsem rád že jsem alespoň něco zvládl a že se to minimálně tváří že to tak nějak funguje a dělá to alespoň část z toho co jsem si představoval, že bych chtěl aby to dělalo.

Projekt má určitě spoustu nedostatků. Z mého pohledu je to spíš jeden velký smutný nedostatek.

Ale abych nebyl pouze negativní tak se musím pochválit za to jak nízké cíle si jsem schopný zvolit. Kdybych si vybral o něco složitější zadání tak bych ho určitě nezvládl vůbec. Takhle když jsem si zvolil velmi jednoduchý projekt tak jsem ho nakonec po dlouhých hodinách snažení zvládl poskládat.

#include "stm8s.h"

#include "milis.h"

#include "stm8\_hd44780.h"

#include "delay.h"

#include <stdio.h>

#define \_ISOC99\_SOURCE

#define \_GNU\_SOURCE

void encoder\_init(void){

GPIO\_Init(GPIOC, GPIO\_PIN\_1,GPIO\_MODE\_IN\_PU\_NO\_IT);

GPIO\_Init(GPIOC, GPIO\_PIN\_2,GPIO\_MODE\_IN\_PU\_NO\_IT);

TIM1\_DeInit();

TIM1\_TimeBaseInit(16, TIM1\_COUNTERMODE\_UP, 500, 16); //inicializace enkoderu

TIM1\_EncoderInterfaceConfig(TIM1\_ENCODERMODE\_TI12,TIM1\_ICPOLARITY\_FALLING,TIM1\_ICPOLARITY\_FALLING);

TIM1\_Cmd(ENABLE);

}

void setup(void)

{

CLK\_HSIPrescalerConfig(CLK\_PRESCALER\_HSIDIV1); // taktovani MCU na 16MHz

lcd\_init(); //inicializace LCD

init\_milis(); //inicializace mmilisu

GPIO\_Init(GPIOC,GPIO\_PIN\_5,GPIO\_MODE\_OUT\_PP\_LOW\_SLOW); // nastavíme PC5 jako výstup typu push-pull (LEDka)

encoder\_init();

}

int main(void)

{

uint32\_t time = milis(); //inicializace proměnných

unsigned int present\_value = 0x0000;

unsigned int previous\_value = 0x0001;

char text[32];

uint32\_t perioda;

setup();

lcd\_gotoxy(0,0);

lcd\_puts("perioda");

while (1) {

if(milis() - time > perioda/2){

GPIO\_WriteReverse(GPIOC,GPIO\_PIN\_5);

time = milis();

}

present\_value = TIM1\_GetCounter();

if(present\_value != previous\_value)

{

lcd\_gotoxy(12, 0);

sprintf(text,"%1u ",present\_value\*4);

lcd\_puts(text);

perioda = present\_value \* 4;

}

previous\_value = present\_value;

}

}

/\*------------------------------- Assert -----------------------------------\*/

#include "\_\_assert\_\_.h"