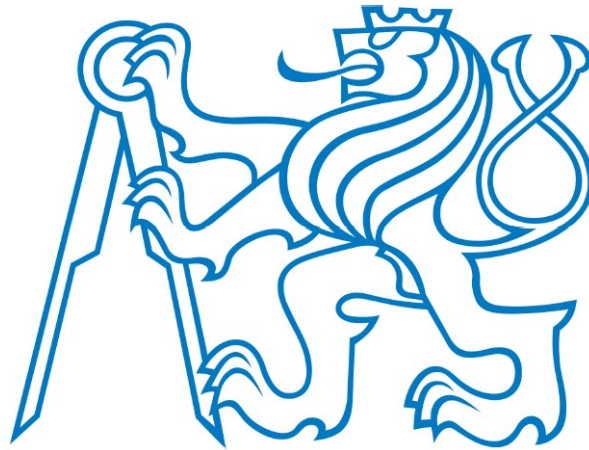


# Vestavěné systémy

Semestrální práce



Vypracovali:

Vlk Jan

Hazdra Martin

Cvičící: Ing. Zdeněk Novák Ph.D.

Ročník: 1

Skupina: L4

# Zadání

Zadání č. 2:

Sestavte program pro dvoupolohový regulátor pro regulaci otáček stejnosměrného motoru. Signál skutečné hodnoty otáček se snímá pulzním čidlem, žádaná hodnota otáček se zadává pomocí potenciometru. Jsou-li žádané otáčky vyšší než skutečné, spíná se hlavní tranzistor pulzního měniče, jsou-li žádané otáčky menší než skutečné, hlavní tranzistor se vypíná.

# Řešení

## Deklarace proměnných

```
1 //Vysledek adc konverze – potenciometr
2 uint32_t potResult = 0;
3
4 //Hodnota casovace
5 uint32_t uSClock = 0;
6
7 //Hodnota casovace na posledni hrane enkoderu
8 uint32_t uSlastPulseTime = 0;
9
10 //Hodnota casovace na aktualni hrane enkoderu
11 uint32_t uStimeNow = 0;
12
13 //Rozdil hodnot casovace mezi hranami
14 uint32_t uStimeDelta = 108882172;
15
16 //Posledni stav senzoru
17 uint8_t lastPinState = 1;
18 uint8_t currentPinState = 0;
19
20 //Prepocitana hodnota z potenciometru na ocekavanou periodu
21 uint32_t requestedSpeed = 0;
```

## Nastavení převodníku a časovače

```
1 //Nastaveni automatickeho zapisovani stavu potenciometru do pameti
2 HAL_ADC_Start_DMA(&hadc3, &potResult, 1);
3 //Zapnuti casovace – mereni v mikrosekundach
4 HAL_TIM_Base_Start(&htim2);
```

## Hlavní smyčka programu

```
1 while (1)
2 {
3     //precteni hodnoty casovace
4     uSClock = _HAL_TIM_GET_COUNTER(&htim2);
5
6     //Vyhodnoceni regulatoru
7     if(requestedSpeed < uStimeDelta){
8         //Motor se toci pomaleji nez je nastaveno
9         HAL_GPIO_WritePin(HSS_Pin_GPIO_Port, HSS_Pin_Pin, 1);
10        HAL_GPIO_WritePin(LSS_Pin_GPIO_Port, LSS_Pin_Pin, 1);
11
12    }
13    if(requestedSpeed > uStimeDelta){
14        //Motor se toci rychleji nez je nastaveno
15        HAL_GPIO_WritePin(HSS_Pin_GPIO_Port, HSS_Pin_Pin, 0);
16        HAL_GPIO_WritePin(LSS_Pin_GPIO_Port, LSS_Pin_Pin, 0);
17
18    }
```

```

19 //precteni stavu senzoru polohy
20 currentPinState = HAL_GPIO_ReadPin(RPM1_GPIO_Port, RPM1_Pin);
21 //Nastaveni casu v aktualnim cyklu
22 uStimeNow=uSClock;
23 if(currentPinState == 1 && lastPinState == 0){
24     // Detekovana hrana
25     //Ulozeni stavu senzoru
26     lastPinState = 1;
27     //Ulozeni casu posledni hrany
28     uSlastPulseTime = uStimeNow;
29
30 }
31 if(currentPinState == 0){
32     //Resetovani stavu senzoru
33     lastPinState = 0;
34 }
35
36 //Vyhodnoceni rychlosti otaceni
37 uStimeDelta=uStimeNow-uSlastPulseTime;

```

## Čtení hodnoty z potenciometru

```

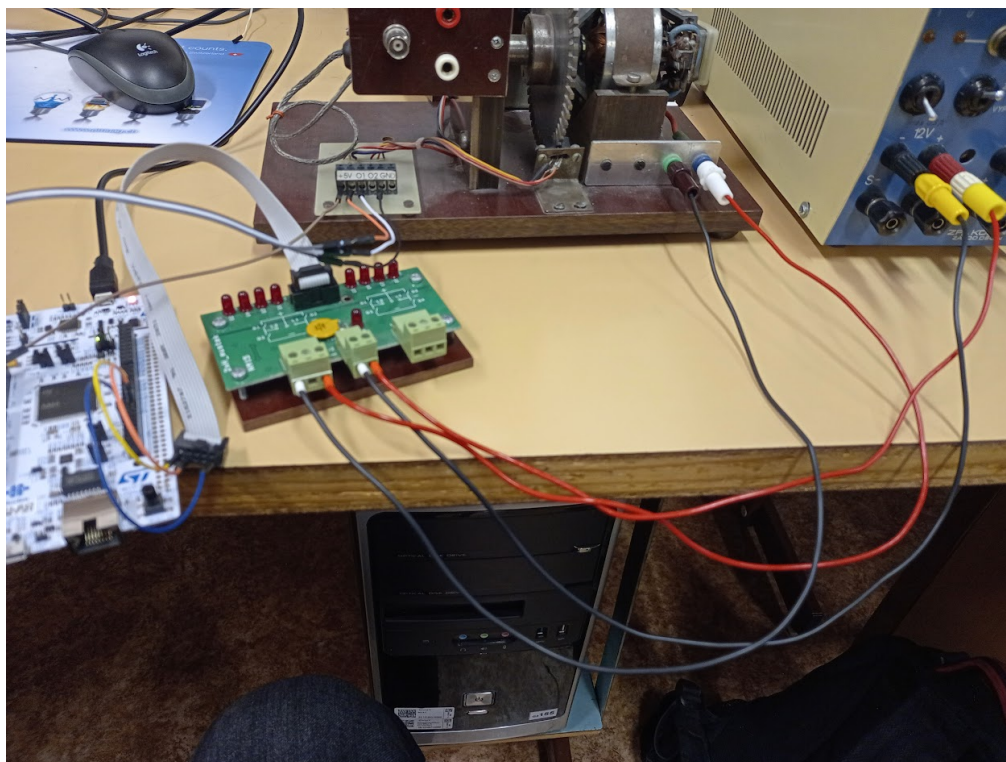
1 void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc){
2     //Konverze dokoncena, ulozeni hodnot a nastaveni vyžadovaných
   otacek
3     potResult = HAL_ADC_GetValue(hadc);
4     requestedSpeed = potResult*2;
5 }

```

## Zapojení

Q1	Q3	Q1	Q3	3v3
Q2	Q4	Q2	Q4	gnd

Obrázek 1: Rozložení pinů na H-můstku



Obrázek 2: Zapojení obvodu

## 1 Závěr

Naším úkolem bylo vytvořit dvupolohové řízení otáček stejnosměrného motoru. Žádanou hodnotu otáček jsme zadávali pomocí potenciometru připojeného na ADC převodník a skutečnou hodnotu otáček jsme zjišťovali pomocí pulzního čidla a čísovače vestavěného v procesoru. Problémem tohoto způsobu řízení byla nemožnost nastavit potenciometr na nulu, tedy zastavit motor úplně.