



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

ŘÍZENÍ STEJNOSMĚRNÉHO MOTORU POMOCÍ PLC SIMATIC S7-1200

DC MOTOR CONTROL BY PLC SIMATIC S7-1200

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. ONDŘEJ JIRČÁK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. TOMÁŠ MARADA, Ph.D.

BRNO 2013

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2012/2013

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Ondřej Jirčák

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Aplikovaná informatika a řízení (3902T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Řízení stejnosměrného motoru pomocí PLC Simatic S7-1200

v anglickém jazyce:

DC motor control by PLC Simatic S7-1200

Stručná charakteristika problematiky úkolu:

Cílem této práce je vytvořit laboratorní úlohu pro řízení stejnosměrného komutátorového motoru. Řízení bude realizováno programovatelným automatem Simatic S7-1200. Nezbytnou součástí je návrh a realizace elektroniky pro připojení elektromotoru k PLC. Dalším cílem je vizualizace na panelu HMI.

Cíle diplomové práce:

1. Seznamte se s principem řízení stejnosměrných motorů.
2. Proved'te návrh a realizaci elektroniky pro připojení elektromotoru k PLC.
3. Seznamte se s programovatelnými automaty S7-1200.
4. Seznamte se s operátorským panelem fy Siemens.
5. Proved'te návrh a realizaci programového vybavení pro polohovou a rychlostní regulaci motoru.
6. Proved'te vizualizaci na panelu HMI.

Seznam odborné literatury:

- [1] Firemní materiály o programovatelných automatech fy Siemens S7-1200.
- [2] Firemní materiály o HMI fy Siemens.
- [3] <http://www.blaja.cz>

Vedoucí diplomové práce: Ing. Tomáš Marada, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2012/2013.

V Brně, dne 20.11.2012

L.S.

Ing. Jan Roupec, Ph.D.
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan fakulty

ABSTRAKT

Diplomová práce „Řízení stejnosměrného motoru pomocí PLC Simatic S7-1200“ se zabývá řízením stejnosměrného motoru pomocí programovatelného automatu PLC Simatic S7-1200 CPU1214C od německé firmy Siemens, který generuje PWM signál. PWM je zpracováváno integrovaným obvodem nazývaným H-můstek (LMD18200T), který následně řídí stejnosměrný motor.

ABSTRACT

The thesis „DC motor control by PLC Simatic S7-1200“ focuses on the control of the DC motor using the programmable automatic machine PLC Simatic S7_1200 CPU1214C, designed by the German company Siemens, generating the PWM signal. The PWM is processed by the integrated circuit called the H-Bridge, which subsequently controls the DC motor.

KLÍČOVÁ SLOVA

PLC, DC motor, H – můstek, Simatic S7-1200

KEYWORDS

PLC, DC motor, H – Bridge, Simatic S7-1200

Bibliografická citace mé práce:

JIRČÁK, O. *Řízení stejnosměrného motoru pomocí PLC Simatic S7-1200*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2013. 75s. Vedoucí diplomové práce Ing. Tomáš Marada, Ph.D..

Prohlášení o originalitě

Prohlašuji, že jsem diplomovou práci “Řízení stejnosměrného motoru pomocí PLC Simatic S7-1200“ vypracoval samostatně pod vedením Ing. Tomáš Marada, Ph.D. s využitím materiálů uvedených v seznamu použité literatury.

Dne 24.5.2013

Jirčák Ondřej

Poděkování

Tím to bych chtěl poděkovat panu Ing. Tomáši Maradovi, Ph.D. za pomoc a rady, které mi poskytl při tvorbě práce.

Obsah:

Titulní stránka.....	1
Zadání závěrečné práce.....	3
ABSTRAKT	5
KLÍČOVÁ SLOVA.....	5
Bibliografická citace mé práce:	7
Poděkování	9
1. Úvod	13
2. Stejnoseměrný motor + optický inkrementální snímač.....	15
2.1. Stejnoseměrný Motor	15
2.1.1. Princip činnosti stejnosměrného komutátorového motoru	15
2.1.2. Stejnoseměrný motor použitý v této práci	16
2.2. Optický inkrementální snímač otáček	18
2.2.1. Princip činnosti IRC snímače	19
2.2.2. IRC snímač použitý v této práci	19
3. PLC Siemens Simatic S7 – 1200 a HMI panel	21
3.1. PLC Siemens Simatic S7 - 1200	21
3.1.1. Specifikace CPU 1214C DC/DC/DC	21
3.1.2. Popis S7 – 1200 CPU1214C	23
3.2. HMI Panel	24
3.2.1. Specifikace KTP400 Basic color	24
3.3. TIA portál.....	25
3.4. Switch D-Link DES – 1005D	25
4. Návrh elektroniky.....	27
4.1. Volba H-můstku	27
4.2. Volba stabilizátoru napětí	29
4.3. IRC snímač.....	32
4.4. Destičky pro S7-1200 CPU1214.....	33
4.5. Rozmístění obvodů na DPS	35
4.5.1. Řídicí elektronika pro DC motor	35
4.5.2. Destičky pro S7-1200 CPU1214.....	36
4.6. Výroba DPS	36
4.6.1. Příprava podkladů pro osvit.....	36
4.6.2. Osvit, vyvolání, leptání, pájení	37

4.6.3. Zapojení konektorů	38
4.7. Oživení DPS	40
5. Programování PLC a HMI	45
5.1. Konfigurace zařízení v TIA portálu	45
5.2. Programování regulátoru	47
5.3. Definice proměnných	49
5.4. Konstrukce programu	50
5.5. Jednotlivé režimy řízení	50
5.5.1. Režim ručního řízení	50
5.5.2. Režim rychlostního řízení	51
5.5.3. Režim polohového řízení	55
5.5.4. Režim skokové změny	57
5.5.5. Režim rampového průběhu	57
5.6. Popis programu v jednotlivých blocích	57
5.6.1. Volání po startu (Startup)	58
5.6.2. Časově volaný blok 1 (T_005)	58
5.6.3. Časově volaný blok 2 (T_010)	62
5.6.4. Cyklicky spouštěný blok (Main)	64
5.7. Tvorba HMI vizualizace	67
6. Závěr	69
Citovaná literatura	71
Seznam příloh:	75

1. Úvod

V dnešní době mají stejnosměrné motory velký význam při automatizaci jednotlivých zařízení, při nichž se hojně využívají. Bývají využívány pro ovládání průmyslových zařízení, například k ovládání kohoutů a pojezdů, proto je dobré seznámit studenty s těmito motory a jejich využitím v jednotlivých odvětvích. Díky jednoduchému řízení lze tento motor využívat prakticky kdekoliv. Jelikož většina stejnosměrných motorů pracuje s bezpečným stejnosměrným napětím, bývají s oblibou využívány při výuce ve školách.

Tato diplomová práce se zabývá řízením stejnosměrného motoru pomocí PWM, který je generován programovatelným automatem Simatic S7-1200 CPU1214 DC/DC/DC od německé firmy Siemens. Programovatelný automat je ovládán pomocí HMI panelu KTP400 Basic color, také od firmy Siemens.

V první řadě je nutné naměřit základní parametry stejnosměrného motoru, jelikož v tomto případě jsou dostupné informace značně limitovány z důvodu excesivního stáří motoru.

Následně bude nezbytné navrhnout řídicí elektroniku pro propojení stejnosměrného motoru s programovatelným automatem. Zároveň bude nutné navrhnout připojovací destičky pro jednodušší a rychlejší připojení k PLC, které se budou následně využívat při výuce programovatelných automatů na škole.

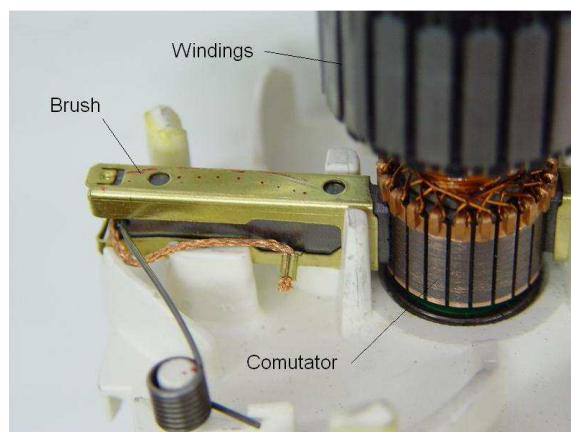
2. Stejnosměrný motor + optický inkrementální snímač

2.1. Stejnosměrný Motor

Stejnosměrný motor je také nazýván DC motor, protože pracuje pouze se stejnosměrným napětím. DC motory mají snadné řízení. Velkou výhodou je, že se zvyšujícím se proudem se zvyšuje krouticí moment. Stejnosměrné motory se dělí do dvou skupin – komutátorové a bezkomutátorové. Hlavními rozdíly mezi oběma typy jsou, že u bezkomutátorového je rotor z permanentního magnetu a stator obsahuje vinutí, ve kterém vzniká magnetické pole. U komutátorového je to přesně naopak: na rotor je přiváděno napětí přes sběrný kartáček (vzniká jiskření). Následně dochází k přenosu napětí přes kartáček na rotor. Otáčením dochází k přepínání jednotlivých vinutí a vzniká magnetické pole. U bezkomutátorového motoru je magnetické pole řízeno elektronikou, která snímá natočení hřídele většinou pomocí hallova snímače a následně snímač předává informaci o natočení rotoru řídicí elektronice, která spíná jednotlivá vinutí.

2.1.1. Princip činnosti stejnosměrného komutátorového motoru

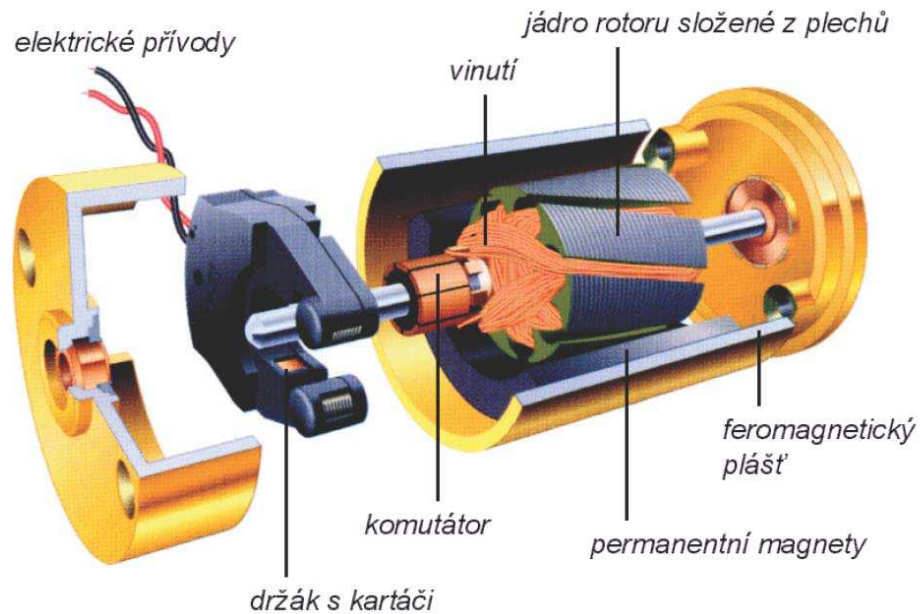
Součástí DC motoru je stator, který obsahuje pravidelně střídající se vzájemně obrácené magnetické pole. To je tvořeno jednotlivými hlavními a komutačními póly, sloužící k přitahování pólů na rotoru. Za hlavním pólem vždy následuje komutační pól stejné polarity. Rotor obsahuje v drážkách vinutí vyvedené na mechanický komutátor. Pomocí komutátoru se přivádí do vinutí proud, který vytváří magnetické pole s hlavními póly, které následně tvoří točivý moment. Pro přenos proudu na komutátor slouží grafitové nebo elektrografitické kartáčky přitlačované na komutátor.



Obr.1 Detail komutátoru DC motoru [1]

Kartáčky jsou umístěny ke středu (magneticky neutrální místo). Motor vždy obsahuje sudý počet kartáčků 2, 4, ... odpovídající počtu pólů. Tvar, složení a nastavení významně ovlivňuje průběh komutace. Proud tekoucí vinutím kotvy vytváří reakční magnetické pole. To deformuje a snižuje magnetické pole hlavních pólů, a tudíž ovlivňuje i magnetické pole

komutačních pólů. Pro odstranění reakčního pole slouží kompenzační vinutí, vložené do drážek pólových nástavců. [2]



Obr.2 Podélný řez DC motorem [3]

2.1.2. Stejnoseměrný motor použitý v této práci

V této práci byl využit stejnosměrný motor od firmy Litton, což je firma, která již více než deset let na trhu nepůsobí. Použitý motor tedy patří mezi starší typy. Velice podobné motory v současnosti vyrábí firma Hitachi. K motoru byl připevněn štítek označující výhradně model a typ motoru. Další parametry nebyly uvedeny. Na internetu byl nalezen pouze motor od firmy Hitachi. Bylo proto nutné pro kontrolu provést kontrolní měření, zda použitý motor odpovídal výše zmíněnému motoru značky Hitachi. Měření probíhalo následovně. Na svorky se ze zdroje stejnosměrného napětí přivedlo napětí +24V. Poté se zabránilo výstupní hřídeli, aby se otáčela, čímž se přivedla do zkratu. Následně se z ampérmetru na zdroji odečetla hodnota proudu 2,2A. Pomocí IRC snímače byly odečteny naměřené otáčky, které se pohybovaly okolo 2000 otáček za minutu v nezátíženém stavu. U nalezeného motoru byly parametry následující: maximální odebíraný proud 2,2A a 1580 otáček za minutu v zatíženém stavu. Z toho vyplývá, že se jedná o podobné motory, nikoliv o totožné. [4]

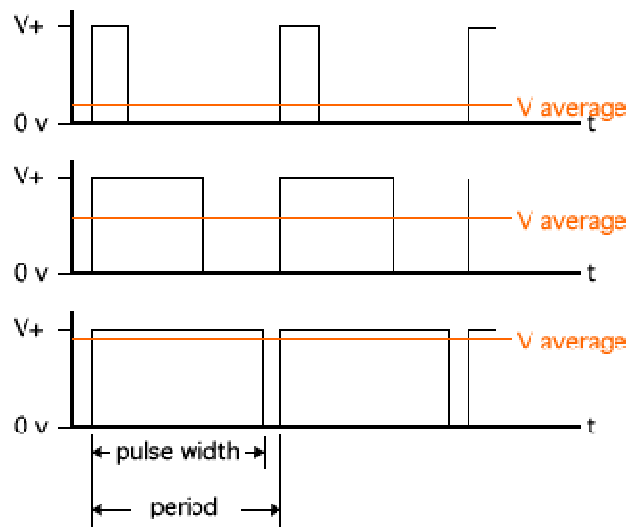
Model	JDTH-2250-CQ1C
Napájecí napětí	24 VDC
P/N	DEC 1214659-00, REV.D
Maximální proud	2,2A
Otáčky motoru	1580 ot/min

Tab. 1 Přehled štítku motoru s parametry



Obr.3 DC motor využitý v práci s IRC snímačem

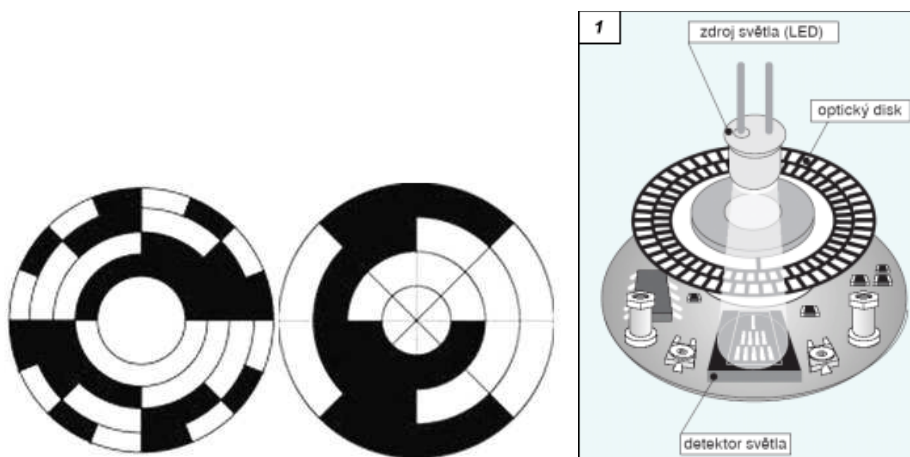
Motor je řízen pomocí PWM (pulzní šířková modulace). V podstatě se přes digitální výstup přenáší analogová veličina v podobě napětí. Velikost napětí se mění pomocí střídy, což je poměr mezi zapnutým a vypnutým signálem. Velikost napětí je tedy dána poměrem $\frac{\text{délka pulzu}}{\text{délka periody}}$ [5]



Obr.4 Zobrazení PWM signálu a výstupní napětí vzniklé z PWM [6]

2.2. Optický inkrementální snímač otáček

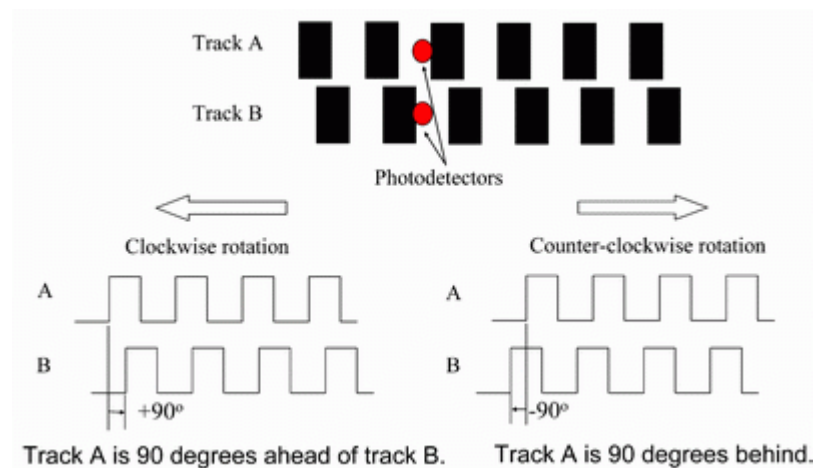
Optické inkrementální snímače bývají často označovány jako IRC snímače. Hlavními prvky snímače jsou černý disk s průhlednými okénky, zdroj světla a prvek citlivý na světlo. Optické snímače se dělí podle principu snímání na přírůstkový (inkrementální) a absolutní. Absolutní snímač bývá dražší než inkrementální. Na druhou stranu ale poskytuje uživateli velkou výhodu. Je u něho totiž vždy známa poloha natočení hřídele motoru, a to jak po zapnutí, tak po resetu zařízení. U inkrementálního snímače je nutný referenční bod, na který se tato hřídel natočí (výchozí poloha hřídele). Tím dojde k zjištění přesné polohy natočení hřídele. Při odpojení napájení nebo resetu vyhodnocovací elektroniky je nutno opět najet na referenční bod. Ale ve zkoumaném případě není referenční bod potřeba, protože se odměřuje rychlost otáčení hřídele motoru.



Obr.5 Disky pro snímání polohy, vlevo pro absolutní snímač a vpravo pro inkrementální snímač, kde je vyobrazena i potřebná elektronika [7], [8]

2.2.1. Princip činnosti IRC snímače

Činnost snímače je založena na principu optické závory. Při otáčení prochází světelný paprsek propustnou částí disku až do momentu, kdy paprsek narazí na nepropustnou část disku, která způsobí jeho přerušení. K jeho opětovnému propojení dochází otočením disku na propustnou část. Takto vznikají jednotlivé pulzy. Ty jsou následně vyhodnocovány pro potřeby regulace nebo získání informací o činnosti. Vždy bývají na disku umístěny dva kanály, tedy dvě optické závory s posunutím mechanicky o jednu čtvrtinu délky a elektricky o 90° . Pomocí těchto signálů lze určit směr otáčení. U snímače lze zvýšit citlivost až na čtyřnásobek (tzv. kvadrurní zapojení) pomocí detekce každé hrany signálů. [9]



Obr.6 Světelné paprsky a graf jednotlivých signálů [8]

2.2.2. IRC snímač použitý v této práci

V této práci byl využit IRC snímač, který byl již součástí DC motoru. Snímač neobsahoval štítek o svých vlastnostech, proto bylo nutné jeho parametry změřit. Odběr snímače byl změřen pomocí zdroje stejnosměrného napětí, který dodával napětí +5V. Ke vstupním svorkám snímače bylo přivedeno napájení a z ampérmetru na zdroji byla odečtena hodnota odebíraného proudu 28 mA. Počet pulzů na jednu otáčku byl změřen pomocí programovatelného automatu Simatic S7 – 1214 DC/DC/DC. Na jednu otáčku jednoho kanálu bylo naměřeno 660 pulzů. V této práci je PLC nastaveno na kvadrurní příjem signálu, tedy 2640 pulzů na jednu otáčku hřídele motoru, viz obr. motoru. V pravé části zobrazen snímač.

3. PLC Siemens Simatic S7 – 1200 a HMI panel

3.1. PLC Siemens Simatic S7 - 1200

Programovatelný automat je běžně označován zkratkou PLC, která pochází z anglického názvu Programmable Logic Controller. Německá firma Siemens patří mezi největšího výrobce programovatelných automatů na světě s 30% podílem na trhu. Simatic 7 – 1200 patří mezi malé programovatelné automaty. PLC jsou vyráběny ve 4 verzích CPU, které se dále dělí na 3 odlišné verze se vstupním napětím a spínáním výstupního napětí. Jednotlivé verze CPU se liší rozměry, pamětí, počtem vstupů a výstupů. Automaty lze rozšířit nepřeberným množstvím rozšiřujících modulů, například digitální a analogové vstupy a výstupy, komunikační moduly (RS 232, RS485, GPRS, ...), atd. Verze CPU: 1211C, 1212C, 1214C, 1215C, které lze pořídit ve verzích: DC/DC/DC, DC/DC/RELAY, AC/DC/RELAY. **V této práci je využita verze CPU 1214C DC/DC/DC.** [10], [11]

3.1.1. Specifikace CPU 1214C DC/DC/DC

Vstupy (D/A)	Výstupy (D/A)	Napájení DC	Rozšiřující signální karta	Rozšiřující komunikační modul
14/2	10/1	24V	8	3

Tab. 2 Parametry PLC Siemens Simatic S7 – 1200 CPU1214C DC/DC/DC

- Rozměry: 110 x 100 x 75 mm
- Ztrátový výkon: 12W
- Uživatelská paměť: Pracovní paměť 50 Kbytes
Pro program 2Mbytes
- Počet vysokorychlostních čítačů: 6
- Pulzní výstupy: 4
- Komunikační port: 1x Profinet

Analogové výstup PLC standartně neobsahuje, do automatu byl dodán pomocí signálního modulu.



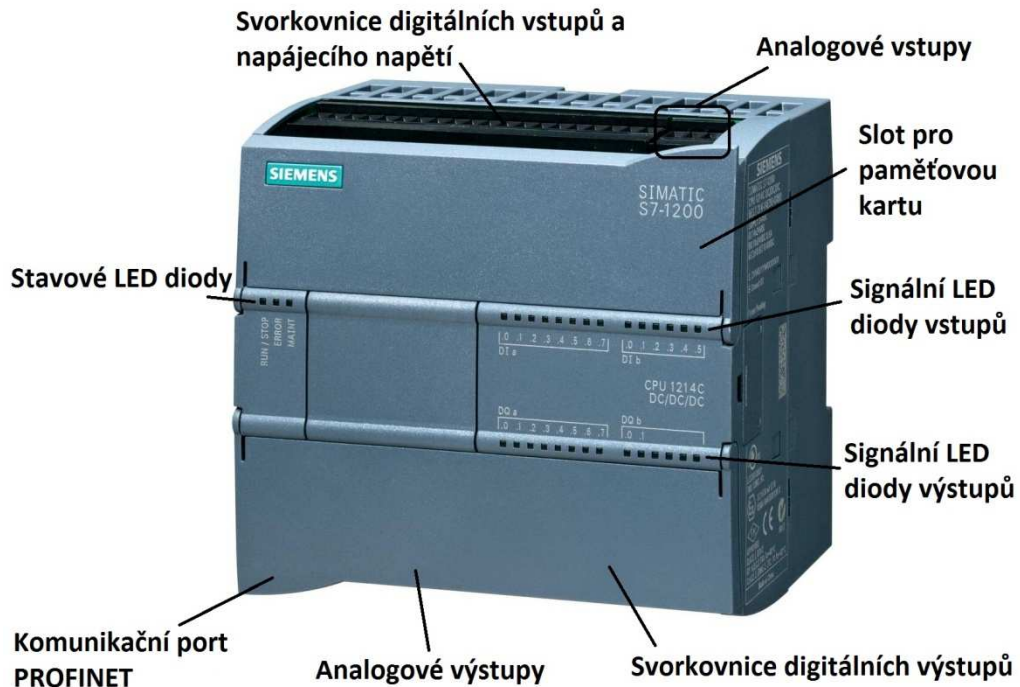
Obr.7 Vkládání signálního modulu do S7 – 1200 CPU1211C

Zprava se připojují rozšiřující signální karty a na levou stranu automatu rozšiřující komunikační moduly.



Obr.8 Rozšiřující signální karty a komunikační moduly S7 – 1200 CPU1214C [12]

3.1.2. Popis S7 – 1200 CPU1214C



Obr.9 Popis S7 – 1200 CPU1214C bez osazeného signálního modulu

- Komunikační port PROFINET slouží ke komunikaci s vývojovým prostředím a s panelem HMI
- Stavové LED diody zobrazují, v jakém režimu se automat nachází (RUN / STOP, ERROR, MAINT)
- Svorkovnice digitálních vstupů a vstupního napětí slouží pro přivedení napájecího napětí pro automat a 14 digitálních vstupních hodnot
- Analogové vstupy slouží pro přivedení až 2 analogových vstupních signálů
- Signální LED diody vstupů a výstupů slouží pro zobrazení, zda je přiveden vstupní signál nebo výstupní signál.
- Slot pro paměťovou kartu slouží pro přenos programu z automatu a pro rozšíření paměti
- Analogové výstupy slouží pro odvod 1 analogového signálu z automatu
- Svorkovnice digitálních výstupu slouží pro připojení zařízení, které mají být vkládány pomocí digitálních hodnot [12]

3.2. HMI Panel

Název HMI pochází z anglického názvu Human-Machine Interface (dotykový ovládací display). HMI jsou vyráběna v rozdílných velikostech, v černobílém či barevném provedení. Display slouží pro zobrazování informací nebo pro ovládání zařízení, která jsou připojena k automatu. Do zařízení není problém namodelovat aplikaci nebo vložit obrázek s ovládacími prvky pro reálný obraz ovládaného zařízení. HMI je napájeno stejnosměrným napětím +24V. Pro přenos informací mezi zařízením a automatem slouží rozhraní PROFINET. K jednomu automatu může být připojeno více HMI panelů. K tomu už je ale většinou zapotřebí použít switch, jelikož automaty v základních verzích ve většině případů obsahují jeden port PROFINET nebo rozšiřující komunikační kartu s dalším portem PROFINET. **V práci je využit panel KTP400 Basic color.**

3.2.1. Specifikace KTP400 Basic color

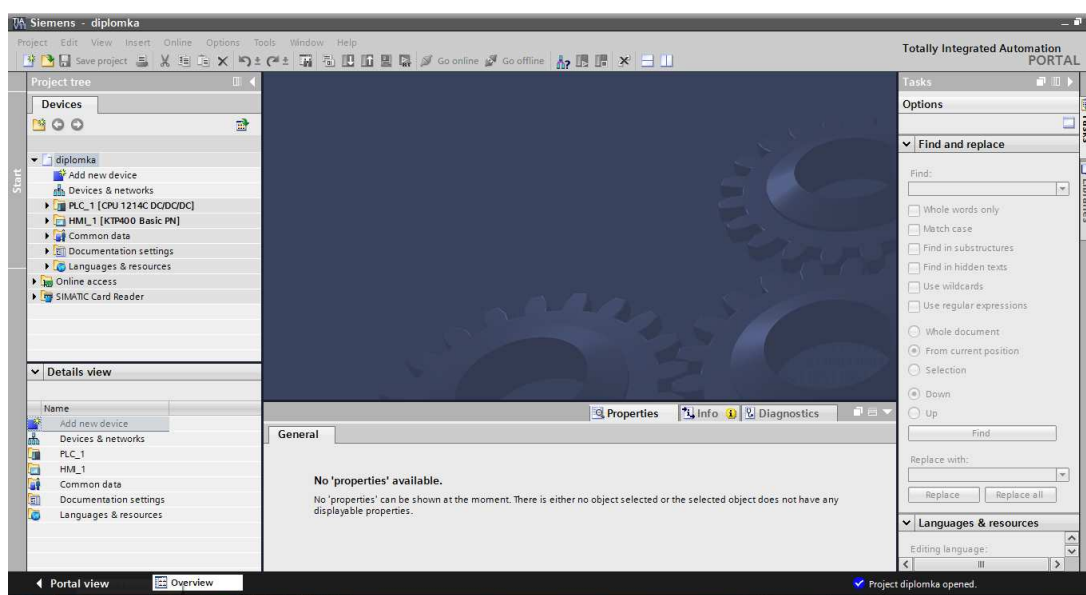
Display	- 4,3 palcový TFT širokoúhlý displej, 256 barev
Rozlišení	- 480 x 272 pixelů
Ovládací prvky	- Dotyková obrazovka rezistivní, 4 volně konfigurovatelné tlačítkové klávesy
Uživatelská paměť	- 512kB
Rozhraní	- 1 x RJ45 pro PROFINET
Rozměry	- 140 x 116 x 34 mm (Š x V x H)
Programovací software	- WinCC Basic (TIA Portal) [13]



Obr.10 KTP400 Basic color

3.3. TIA portál

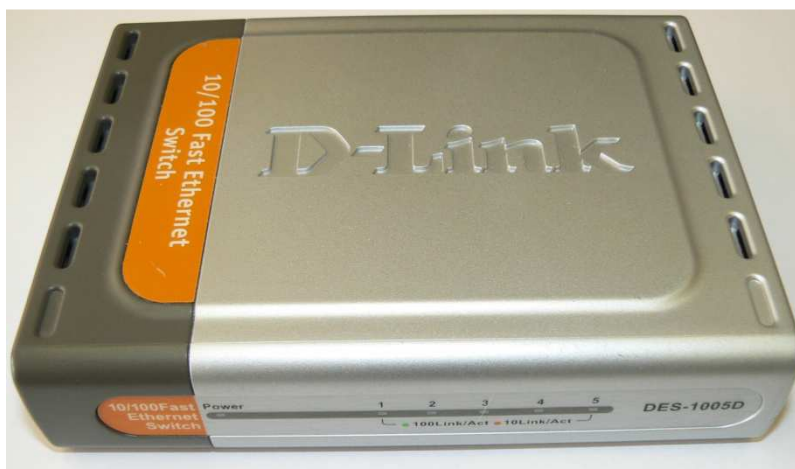
TIA portál je software pro programování programovatelných automatů řady simatic a také pro HMI panely. Software vyvinula německá společnost Siemens pro své PLC. TIA portál je především tvořen dvěma moduly STEP 7 a WinCC, které jsou samostatně dostupné, ale TIA je obsahuje společně. STEP 7 slouží pro programování simaticu a WinCC pro tvorbu vizualizací na HMI panelu a provázání proměnných s automatem. Do programu lze doinstalovat mnoho přídatných modulů, které jsou k nalezení na stránkách výrobce.



Obr.11 Tia portál v11

3.4. Switch D-Link DES – 1005D

V této práci bylo nezbytné využít switch pro komunikaci mezi programovatelným automatem, HMI a počítačem a pro naprogramování zařízení. Pomocí počítače také byly programy odladěny a získány hodnoty pro přechodovou charakteristiku. Využit byl switch od firmy **D-link DES – 1005D**. Switch podporuje pouze ethernet, který postačuje pro komunikaci u této aplikace a je využit pouze při vývoji programů. Při provozu aplikace, když už není zapotřebí PC pro získávání hodnot a ladění, je HMI panel propojen přímo s programovatelným automatem (viz obrázek č. 12 na následující straně).



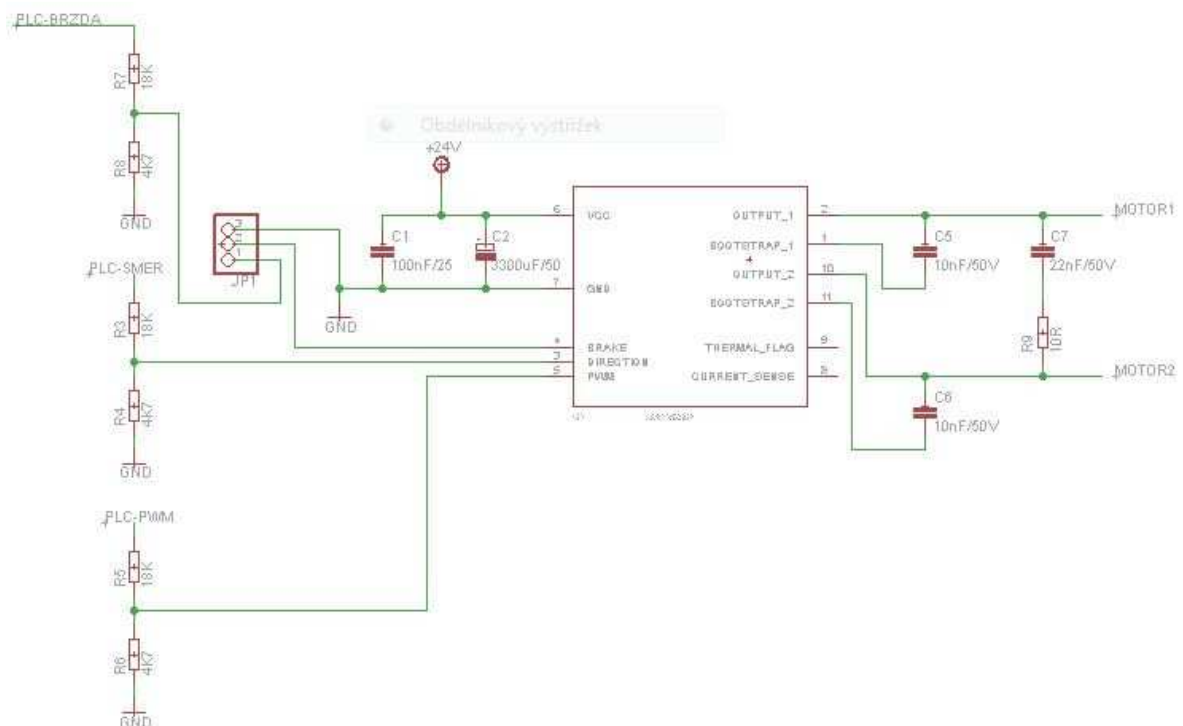
Obr.12 Switch D-Link DES – 1005D

4. Návrh elektroniky

Návrh veškeré elektroniky byl realizován v programu Eagle verze 6.3 od německé společnosti Cadsoft.

4.1. Volba H-můstku

Původně zamýšlený H-můstek L298 byl zavrhnut po zjištění maximálního odběru elektromotoru 2,2A, jelikož tento můstek dodává maximálně 2A. Z tohoto důvodu bylo nutné vybrat jiný můstek, který by byl schopen trvale dodávat elektromotoru požadovaný maximální proud. Proto byl zvolen integrovaný obvod LMD18200T s parametry: 3A trvalý až 6A špičkový proud a napětí do 55V. [14]

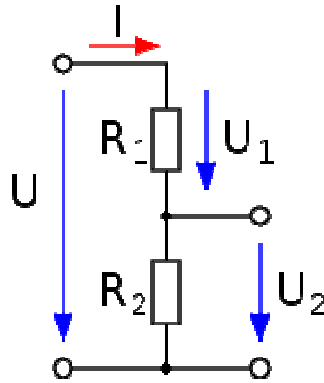


Obr.13 Schéma zapojení H-můstku LMD18200T

Napájecí napětí +24V bylo přivedeno na napájecí svorku můstku číslo 6. Toto napájecí napětí bylo filtrováno elektrolytickým kondenzátorem C2 s kapacitou 3300uF, vedle kterého byl přidán blokovací kondenzátor C1 s kapacitou 100nF. Velikost filtračního a blokovacího kondenzátoru byla zvolena na základě diplomové práce Bc. Pavla Nekuli [15], kde tato kapacita byla ověřena. Na vstup číslo 7 byla připojena zem GND. Velikost blokovacích kondenzátorů na výstupu k motoru byla zvolena na základě katalogového listu, kde byl ukázkový příklad zapojení. Dva smd0805 keramické kondenzátory C5 a C6 s kapacitou 10nF byly připojeny na piny můstku 1, 2 a 10, 11. Na vstupy 3, 4, 5 byly přivedeny vstupní signály: směr, brzdění a PWM. Všechny tyto signály byly pomocí

odporového děliče sníženy z 24V na 5V. Výpočty odporů pro dělič napětí byly vypočítány podle vztahu:

$$U_2 = U * \frac{R_2}{R_1 + R_2} \Rightarrow 24 * \frac{4700}{18000 + 4700} = 4,969V$$



Obr.14 Schéma odporového děliče [16]

Velikost vypočítaných odporů R3, R5 a R7 byla tedy 18kΩ a odpory R4, R6 a R8 měly hodnotu 4,7kΩ. Pro tyto hodnoty bylo třeba provést kontrolní výpočet zatížení odporů.

$$I = \frac{U}{R} = \frac{24}{18000 + 4700} = 1,057mA$$

$$P = U * I = 24 * 0,001057 = 0,03W$$

Z katalogového listu vyplývá, že tyto odpory byly dostačující [17].

Ke vstupnímu signálu brzda byl přidán jumper pro přepínání a ovládání brzdění. Když není zapotřebí ovládat brzdění motoru, tak je svorka propojena s GND. Mezi výstupy vedoucími k elektromotoru bylo nutné paralelně přidat odpor a kondenzátor. Tyto prvky byly přidány z důvodu potlačení jiskření komutátoru motoru. Hodnoty byly zvoleny podle diplomové práce Bc. Pavla Nekuli, kde tyto hodnoty byly ověřeny. Velikost kondenzátoru C7 byla 22nF a velikost odporu R9 činila 10Ω. [15]

Výpočet chladiče: [18]

C_1 – materiálová konstanta (1)

C_2 – vyzařovací konstanta (0,43)

V_a – teplota okolí (45°C)

V_c – teplota pouzdra

V_j – maximální teplota přechodu (150°)

K_1 – vnitřní tepelný odpor (2°C/W)

K_2 – vnější tepelný odpor

Odebíraný výkon:

$$P_{odebíraný} = U * I_{odebíraný}$$

Teplota pouzdra:

$$V_C = V_j - (K_1 * P)$$

Vnější odpor:

$$K_2 = \frac{V_c - V_a}{P}$$

Plocha chladiče:

$$S = \frac{C_1 * 760 * C_2}{K_2 - C_1 * 1,73}$$

$$P_{ztrátový} = 25W$$

$$V_C = 150 - (1 * 25) = 125^\circ C$$

$$K_2 = \frac{125 - 45}{25} = 3,2^\circ C/W$$

$$S = \frac{1 * 760 * 0,43}{3,2 - 1 * 1,73} = 222,3cm^2$$

Integrovaný obvod byl osazen chladičem V7131, který má plochu 28,25cm². Pro školní účely je tento chladič dostačující.

4.2. Volba stabilizátoru napětí

Stabilizátor napětí byl využit v této práci pro napájení IRC snímače, který byl napájen +5V. Na výběr bylo ze dvou typů stabilizátorů napětí: lineární a spínaný. Lineární stabilizátor napětí má jednoduché zapojení, avšak jeho nevýhodou je, že srážené napětí přeměňuje na teplo. Pro účely této diplomové práce byl lineární stabilizátor nevhodný z důvodu velkého sráženého napětí 19V, které by bylo zapotřebí zchladit $P = U * I = 19 * 0,28 = 5,32W$. Proto byl zvolen spínaný stabilizátor napětí **LM2576T-5**. Spínání stabilizátoru funguje na principu již dříve zmiňované pulzní šířkové modulace.

Parametr	Hodnota	Jednotky
U _{out}	5	[V]
I _{out}	3	[V]
U _{in max}	45	[V]
Frekvence	52	[kHz]
Účinnost	77	[%]
Teplotní rozsah	-40 .. 125	[°C]

Tab. 3 Parametry LM2576T-5 [19]

Integrovaný obvod byl napájen +24V, které byly přivedeny na vstup. Toto vstupní napětí bylo filtrováno elektrolitickým kondenzátorem C3 s kapacitou 100uF. Tato hodnota byla zvolena podle příkladu v datasheetu. Na vstup signálu pro zapnutí nebo vypnutí obvodu byla napojena zem, která byla zinvertována na opačnou hodnotu. Z tohoto důvodu byl stabilizátor trvale zapnut. Na výstupní svorku byla připojena tlumivka. Hodnota tlumivky byla zvolena podle grafu zatížení. Touto hodnotou se ovlivňuje účinnost integrovaného obvodu.

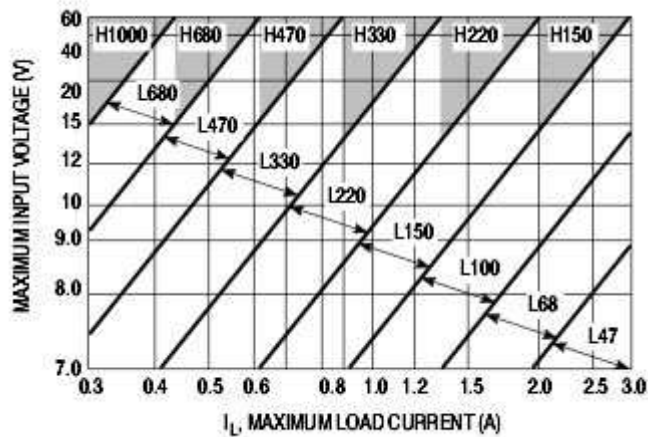


Figure 19. LM2576-5

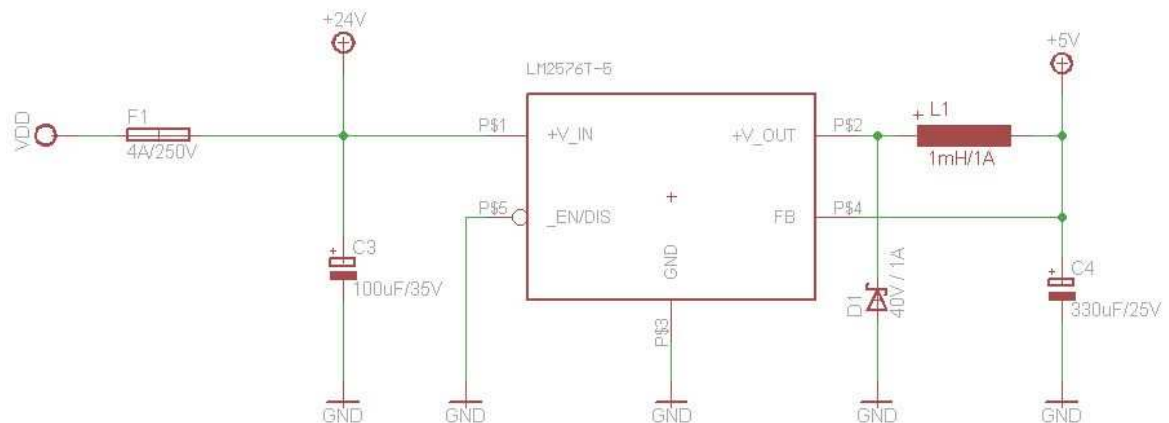
Obr.15 Diagram pro výběr tlumivky LM2576T-5

Vstupní napětí 24V, odebíraný proud 0,28A => Hodnota tlumivky H1000. Použitá tlumivka L1 měla parametry 1mH/1A.



Obr.16 Tlumivka 1mH/1A [20]

Na výstup před tlumivku byla připojena schottkyho dioda D1 o parametrech 40V/1A. Hodnoty této diody byly zvoleny podle typického příkladu připojení v datasheetu. K výstupnímu napětí +5V byl přidán elektrolitický kondenzátor C4 s kapacitou 330uF, který byl umístěn za tlumivkou a připojen proti GND. Před kondenzátorem C4 byla připojena zpětná vazba spínaného stabilizátoru napětí.



Obr.17 Schéma zapojení stabilizátoru napětí LM2576T-5

Výpočet chladiče:

C_1 – materiálová konstanta (1)

C_2 – vyzařovací konstanta (0,43)

V_a – teplota okolí (45°C)

V_c – teplota pouzdra

V_j – maximální teplota přechodu (150°)

K_1 – vnitřní tepelný odpor (2°C/W)

K_2 – vnější tepelný odpor

Odebíraný výkon:

$$P_{\text{odebíráný}} = U * I_{\text{odebíráný}}$$

Celkový výkon:

$$P_{\text{celkový}} = \frac{P_{\text{odebíráný}}}{\eta}$$

Ztrátový výkon:

$$P_{\text{ztrátový}} = P_{\text{celkový}} - P_{\text{odebíráný}}$$

Teplota pouzdra:

$$V_c = V_j - (K_1 * P)$$

Vnější odpor:

$$K_2 = \frac{V_c - V_a}{P}$$

Plocha chladiče:

$$S = \frac{C_1 * 760 * C_2}{K_2 - C_1 * 1,73}$$

$$P_{\text{odebíráný}} = 5 * 0,028 = 0,14W$$

$$P_{celkový} = \frac{0,14}{0,77} = 0,18W$$

$$P_{ztrátový} = 0,18 - 0,14 = 0,04W$$

$$V_C = 150 - (2 * 0,04) = 149,92^{\circ}C$$

$$K_2 = \frac{149,92 - 45}{0,04} = 2623^{\circ} \frac{C}{W}$$

$$S = \frac{1 * 760 * 0,43}{2623 - 1 * 1,73} = 0,125cm^2$$

Kontrolní výpočet chladiče přes celkový výkon:

$$P_{celkový} = U * I = 5 * 3 = 15W$$

$$P_{maxodebíraný} = P_{celkový} * \eta = 15 * 0,77 = 11,55W$$

$$P_{ztrátový} = 15 - 11,55 = 3,45W$$

$$V_C = 150 - (2 - 3,45) = 143,1^{\circ}C$$

$$K_2 = \frac{143,1 - 45}{3,45} = 28,4^{\circ}C/W$$

$$S_{celkový} = \frac{1 * 760 * 0,43}{28,4 - 1 * 1,73} = 12,25cm^2$$

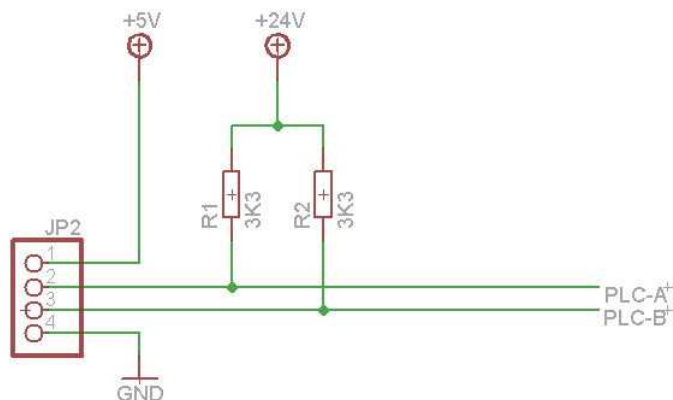
$$S = \frac{12,25}{100} = 0,1225cm^2$$

Pro výpočet chladiče z celkového výkonu bylo nutné vydělit plochu číslem 100, protože odebíraný proud byl 100násobně menší než celkový. Vypočtené hodnoty ploch vyšly obdobně, tudíž je lze, s přihlédnutím k metodě výpočtu, považovat za shodné. Drobná odchylka totiž vznikla při výpočtech na kalkulačce, kde docházelo k zaokrouhlování jednotlivých výpočtů. Spínaný stabilizátor byl osazen chladičem DO3A, který byl schopen uchladiť stabilizátor při maximálním zatížení. Z výpočtů vyplývá, že by nebylo zapotřebí používat tak velký chladič.

4.3. IRC snímač

IRC snímač je nezbytné napájet napětím +5V, proto bylo nutné přivést napájecí napětí ke svorkám snímače. Kanály A a B, které přenášely signál do PLC, byly připojeny k napětí +24V přes odpory R1 a R2. Oba odpory měly shodnou hodnotu 3,3 kΩ. Vývod

JP2 sloužil k připojení ke vstupním svorkám IRC snímače. PLC-A a PLC-B byly připojeny k vysoko rychlostnímu čítači PLC.



Obr.18 Schéma propojení a napájení IRC snímače

4.4. Destičky pro S7-1200 CPU1214

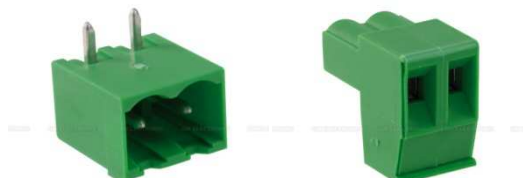
Pro potřeby výuky programovatelných automatů byly navrženy výstupní destičky s konektory MLW10 (ve schématu označované jako SV), které umožňují rychlé přepojování výukových modelů, jako například EDU modely.



Obr.19 Konektor MLW10 [21]

Vstupy PLC:

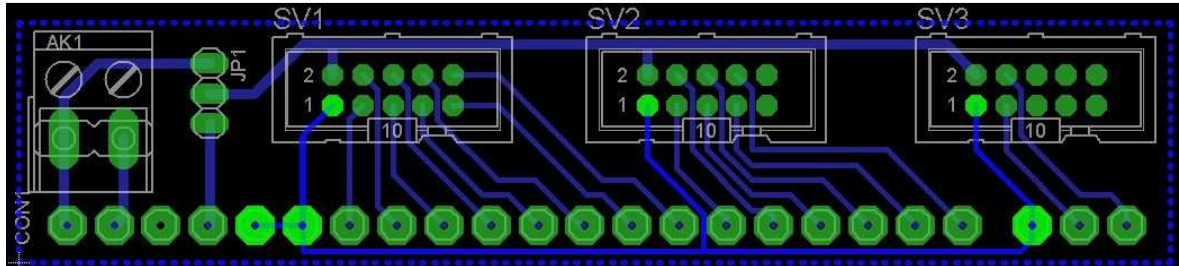
Pro napájení byl zvolen konektor PRT (ve schématu označený jako AK1), který umožňuje svým tvarem zabránit prepólování napájecího napětí do automatu. Na rozdíl od MLW10 je také stavěn pro potřeby napájení. Při výměně různých automatů to tedy znamená urychlení výměny PLC. Nemusí se totiž dávat pozor, kde je plus napájecí napětí a kde mínus. Stačí zapojit jednou.



Obr.20 Konektor PRT STLZ950/2G-5.08H s proti kusem [22], [23]

U destičky byl navržen přepínací konektor JP1 pro přepínání napájecího napětí pro zařízení, které vyžaduje větší odběr proudu připojeného k PLC, a to buď z PLC nebo

přímo z napájecího zdroje. Přepínání napájecího napětí bylo realizováno pomocí jumperu. Vývody z automatu byly propojeny následovně.



Obr.21 Schéma propojení destičky pro vstupy PLC S7-1214

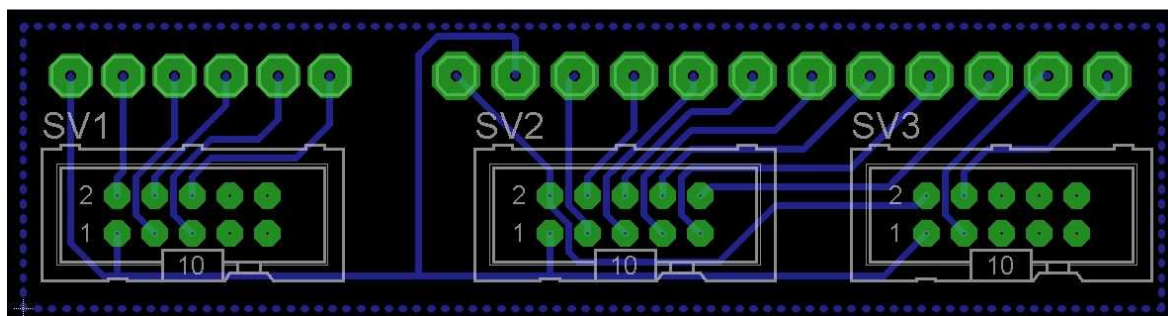
Konektor SV1		Konektor SV2		Konektor SV3	
1	GND	1	GND	1	GND
2	+24V	2	+24V	2	+24V
3	I0.0	3	I1.0	3	AI0
4	I0.1	4	I1.1	4	AI1
5	I0.2	5	I1.2	5	-
6	I0.3	6	I1.3	6	-
7	I0.4	7	I1.4	7	-
8	I0.5	8	I1.5	8	-
9	I0.6	9	-	9	-
10	I0.7	10	-	10	-

Tab.4 Zapojení vstupní destičky pro PLC S7-1214

Veškeré konektory MLW10 byly deseti pinové i přes to, že některé vývody nebyly vůbec zapojeny. Konektory SV1 a SV2 sloužily pro připojení digitálních vstupů a SV3 pro připojení analogových vstupů.

Výstupy PLC:

I v tomto případě byly použity konektory MLW10 pro připojení výstupu. Konektor SV1 byl připojen k signál boardu, který v tomto případě obsahoval pouze jeden výstup. Jednalo se o výměnnou část, kterou bylo možné vyměnit například za board s digitálními výstupy. V případě použití digitálních výstupů by obsahoval více výstupů, a proto by bylo neuvážené jednotlivé vývody nepřipojit také. Tím byla zajištěna univerzálnost destičky.



Obr.22 Schéma propojení destičky pro výstupy PLC S7-1214

Konektor SV1		Konektor SV2		Konektor SV3	
1	M	1	GND	1	GND
2	AO0	2	+24V	2	+24V
3	3	3	Q0.0	3	Q1.0
4	4	4	Q0.1	4	Q1.1
5	5	5	Q0.2	5	-
6	6	6	Q0.3	6	-
7	-	7	Q0.4	7	-
8	-	8	Q0.5	8	-
9	-	9	Q0.6	9	-
10	-	10	Q0.7	10	-

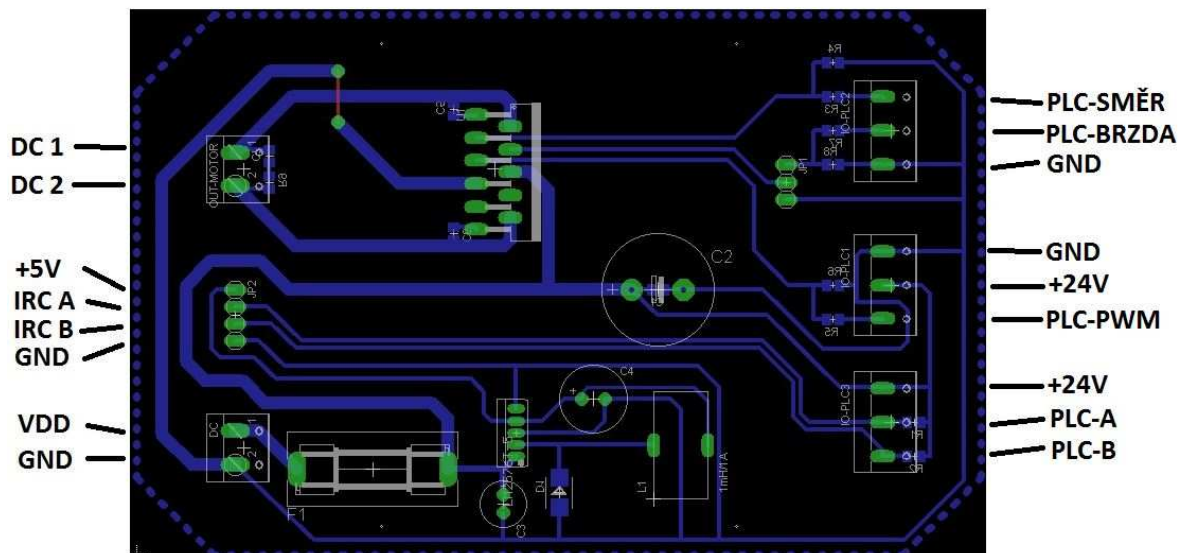
Tab.5 Zapojení výstupů destičky pro PLC S7-1214

4.5. Rozmístění obvodů na DPS

4.5.1. Řídicí elektronika pro DC motor

Obvody byly na DPS rozmístěny tak, aby na jedné straně byly vstupy a výstupy k automatům a na straně druhé vstupy a výstupy k motoru a k napájení plošného spoje. Na straně k automatu a zároveň k motoru byla snaha pokud možno vstupy a výstupy mít uspořádané vedle sebe, nebo co nejbližší ke konektorům. Po několika pokusech o rozmístění prvků se nakonec povedlo nalézt optimální rozmístění, které umožňovalo, aby vstupy a výstupy navazovaly na konektory a zároveň byly i čitelné. Také bylo nutné rozmístit jednotlivé součástky (kromě SMD součástek) s přihlédnutím k tomu, že na integrované obvody budou namontovány chladiče. Zkratka VDD slouží k označení vstupního napájecího napětí +24V, které bylo přivedeno na 3A pojistku. Pojistka by měla zabránit nenávratnému poškození plošného spoje nebo součástek, kdyby náhodou došlo

ke vzniku zkratu ať už zaviněného součástkami, chybnou obsluhou nebo nešetrným zacházením.



Obr.23 Schéma zapojení řídicí elektroniky pro DC motor

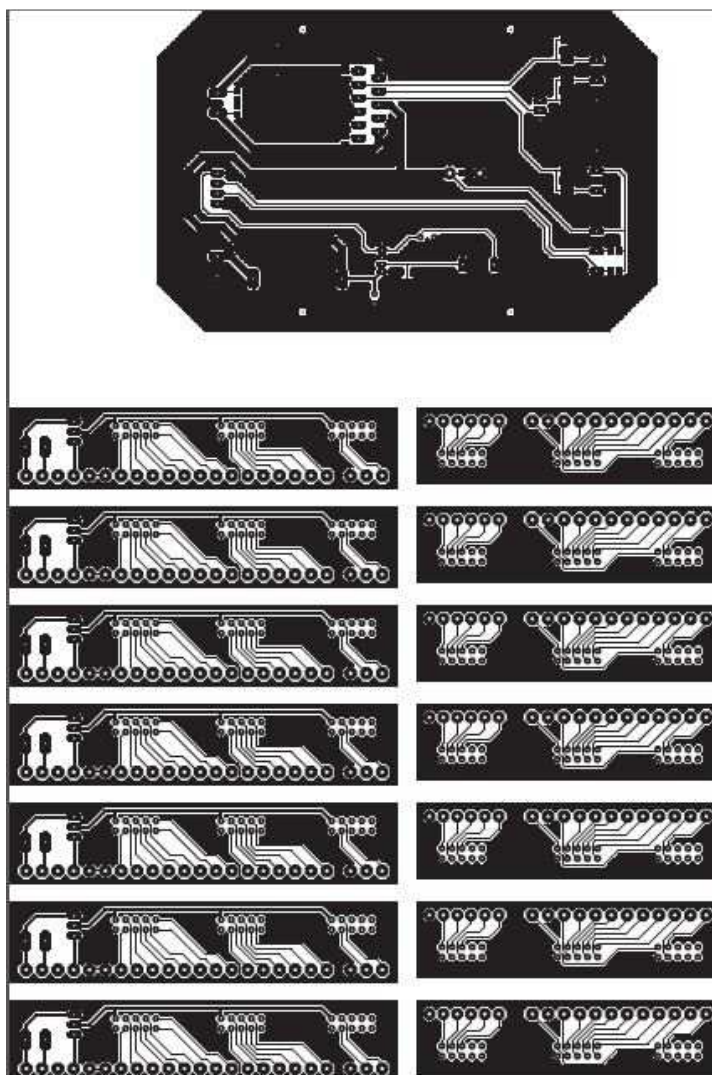
4.5.2. Destičky pro S7-1200 CPU1214

Také zde byly jednotlivé prvky rozmístěny tak, aby byly respektovány pozice vstupů a výstupů na programovatelném automatu. Na příklad konektor SV1 byl umístěn na proti digitálním vstupům IO a SV2, proti I1. obr. 21

4.6. Výroba DPS

4.6.1. Příprava podkladů pro osvit

Dokončené návrhy plošných spojů byly uloženy z eaglu jako postscript (zkratka .eps) a následně otevřeny v Adobe Ilustrátoru. Tam byly jednotlivé podklady vloženy do jednoho souboru a rozrovnány tak, aby byl prostor co nejlépe využit. Výsledkem byl jediný uložený soubor. Výrobce osvitových filmů uvádí na svých stránkách nejen informace o tom, jaké náležitosti by měl soubor mít a jak jich docílit, ale také pokyny pro nastavení k tisknutí. Pomocí Adobe Distilleru 11 byl soubor upraven a uložen jako pdf. Velkou výhodou představovalo, že na stránkách byl k nalezení konfigurační soubor pro Distiller. Ten stačilo naimportovat do programu a pomocí tohoto programu přeložit. Po tomto kroku byly podklady pro výrobu osvitové folie hotovy a odeslány do firmy na výrobu osvitové folie. [24]



Obr.24 PDF podklady pro výrobu osvitové folie

4.6.2. Osvit, vyvolání, leptání, pájení

Osvit byl proveden na vyrobené osvitové jednotce z UV diod.

Jako fotosenzitivní matrice byla vybrána jednovrstvá deska Fotocuprexit 300x200x1,5, na kterou byla položena osvitová folie. To celé bylo následně vloženo mezi dvě skla.

Po tomto úkonu následovalo osvětlení fotocitlivé desky po dobu pěti a půl minuty ve vzdálenosti 15 cm. Po ukončení osvětlení bylo prováděno vyvolání ve vývojce (1,5% NaOH) až do té doby, dokud se uvolňovala měď. Po vyvolání byla deska umyta vodou, usušena a připravena pro další krok: leptání. Doba leptání je obecně závislá na teplotě leptacího roztoku chloridu železitého (FeCl_3). Ve zkoumaném případě byla doba leptání zhruba 1,5 hodiny. Po uplynutí této doby byl plošný spoj vyjmut z leptacího roztoku a opláchnut. Po oschnutí byla deska rozstříhávána na jednotlivé destičky, které byly následně očištěny acetonem od zbývajících nečistot a poté nastříkány kalafunou na pájení. Jako závěrečný krok byly destičky osázeny konektory a součástkami a zapájeny.

4.6.3. Zapojení konektorů

Pro připojení stejnosměrného motoru byly zvoleny tři konektory s různým počtem pinů, aby nebylo možné zaměnit jednotlivé konektory mezi sebou. Z tohoto důvodu byly zvoleny konektory typu XLR, které se dají pořídit ve firmě GES.

Pro propojení motoru a jeho napájení s řídicí elektronikou byly vybrány konektory XLR 3 ES a XLR 3 KU.



Obr.25 Konektory XLR 3 ER a XLR 3 KU

Pin	Zapojení signálů
1	DC 1
2	DC 2
3	-

Tab.6 Zapojení signálů konektoru XLR 3

IRC snímač byl propojen s řídicí elektronikou pomocí konektoru XLR 5.



Obr.26 Konektory XLR 5 ER a XLR 5 KU

Pin	Zapojení signálů
1	+5V
2	CHN A
3	-
4	CHN B
5	GND

Tab.7 Zapojení signálů konektoru XLR 5

Pro přivedení napájecího napětí ze zdroje k řídicí elektronice byl vybrán konektor XLR 4. Z důvodu větší zátěže, která by byla vyvinuta na jeden pin, byly piny zdvojeny. Výsledkem bylo propojení prvního pinu s druhým a třetího se čtvrtým.



Obr.27 Konektory XLR 4 ER a XLR 4 KU

Pin	Zapojení signálů
1	+24V
2	+24V
3	GND
4	GND

Tab.8 Zapojení signálů konektoru XLR 4 [25]

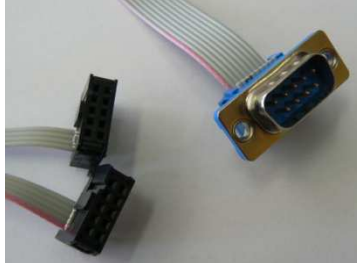
Pro propojení řídicí elektroniky s programovatelným automatem byl zvolen devíti pinový konektor Canon (pro plochý kabel).



Obr.28 Konektory 9 pinový Canon

Na konektor na obr. 29 (pravá strana) byl nakrimpován plochý devíti žilový kabel, který byl na druhé straně rozdělen takovým způsobem, který dal vzniknout dvěma

svazkům, jeden čítající 5 žil a druhý 4 žíly. Na konce oddělených svazků byly nakrimpovány konektory MLW10 pro vstupní a výstupní signály z řídicí elektroniky.



Obr.29 Propojovací kabel PLC – řídicí elektronika

Pin	Zapojení signálů	Pin	Zapojení signálů
1	GND	6	+24V
2	PWM	7	SMĚR
3	BRZDA	8	GND
4	+24V	9	A
5	B	-	-

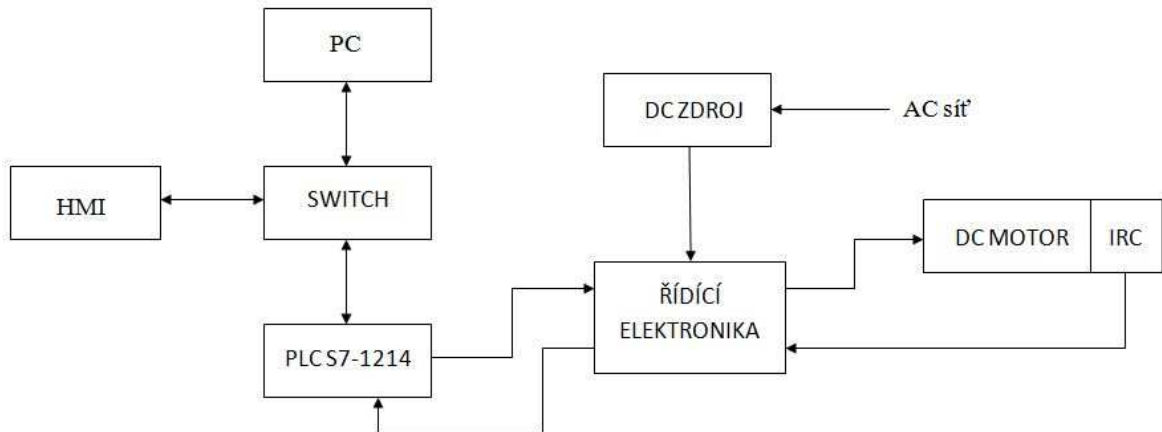
Tab.9 Zapojení signálů u řídicí elektroniky

PLC OUT		PLC IN	
Pin	Zapojení signálů	Pin	Zapojení signálů
1	GND	1	GND
2	+24V	2	+24V
3	PWM	3	A
4	SMĚR	4	B
5	BRZDA	-	-

Tab.10 Zapojení signálů u PLC

4.7. Oživení DPS

Při oživení řídicí elektroniky byl použit laboratorní zdroj na otestování správné funkčnosti spínaného stabilizátoru napětí. Pro ověření správné funkčnosti stabilizátoru bylo na zdroji nastaveno přibližně +8V. Zároveň byly proměřeny cesty na plošném spoji, zda-li nedošlo při výrobě nebo pájení součástek k jejich propojení, což by mohlo být případně příčinou zkratu. Když byla ověřena správná funkčnost plošného spoje po elektrické stránce, bylo třeba ověřit řízení motoru pomocí h-můstku (PWM signál). Pro ověření správné činnosti PWM bylo nejprve nutné provést zapojení veškerých zařízení pro tuto činnost, viz obr. 30.



Obr.30 Blokové schéma zapojení zařízení

Po zapojení zařízení dle schématu byla v TIA portálu provedena potřebná nastavení pro PWM (nastavení PWM níže) a byl napsán jednoduchý program pro ověření činnosti.

```

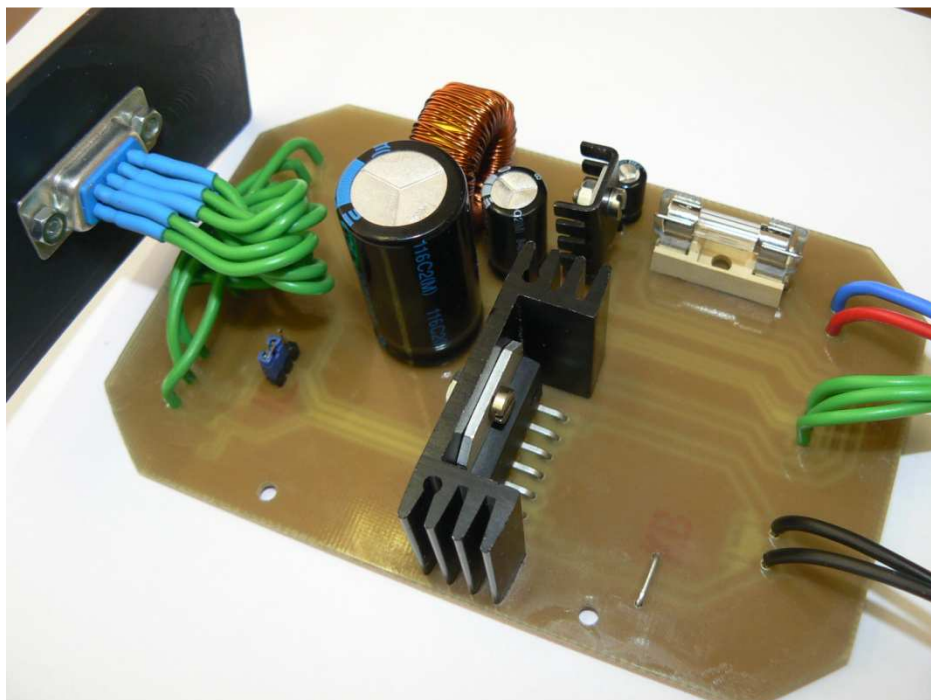
"PWM_channel_1" := 80; //přířazení žádané rychlosti
"CTRL_PWM_DB"(PWM:="PWM1[PTO/PWM]", ENABLE:=True); // spustení bloku PWM

```

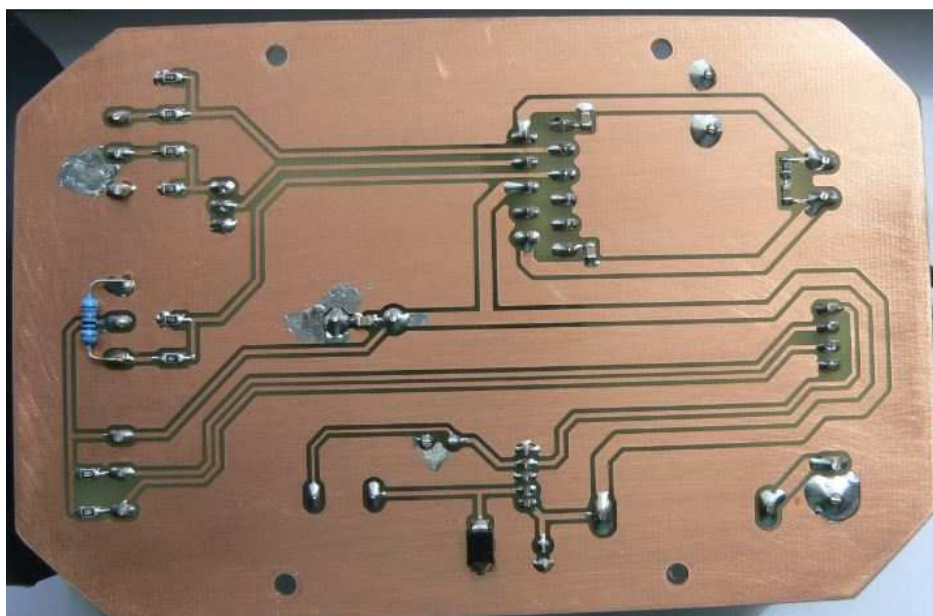
Při frekvenci 1 kHz mělo PWM ostré hrany a pracovalo dle očekávání. Ovšem při frekvenci 25kHz nastal problém a to ten, že vznikaly postupně klesající hrany (jako by se vybíjel kondenzátor), ale ten v konstrukci nebyl nikde zabudován. Při zkoumání této chyby byla odhalena chybějící odporová zátěž mezi PLC a h-můstkem, která by zajistila ostré hrany PWM signálu. Proto bylo nutné paralelně přidat odpor s hodnotou 1kΩ (viz obr. 32). Tato hodnota byla zvolena na základě výpočtu, protože výstup automatu může být maximálně zatížen 0,5A.

$$I = \frac{U}{R} = \frac{24}{1000} = 0,024A$$

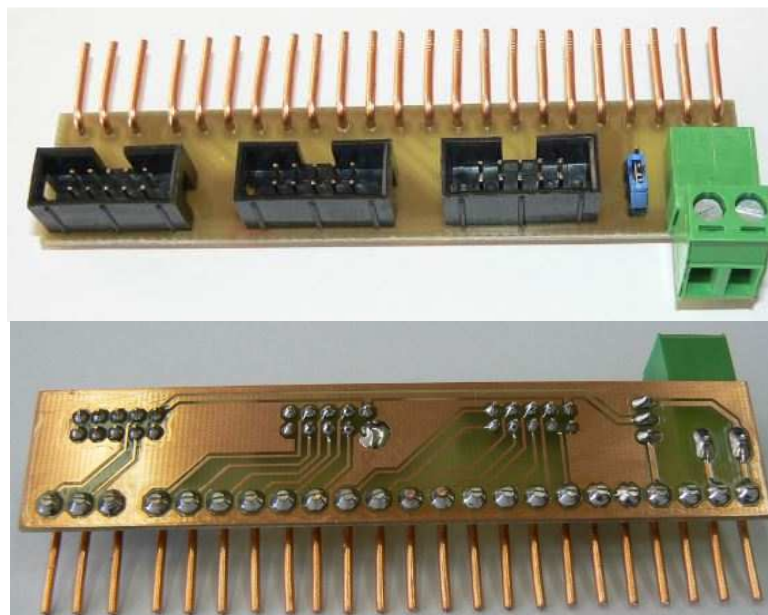
Po přidání odporu byla funkčnost PWM znovu ověřena. Výsledkem tohoto testu bylo zjištění, že funguje optimálně. Takto byla ověřena kompletní funkčnost DPS. Oživení destiček pro vstupy a výstupy k PLC spočívalo pouze v připojení k automatu.



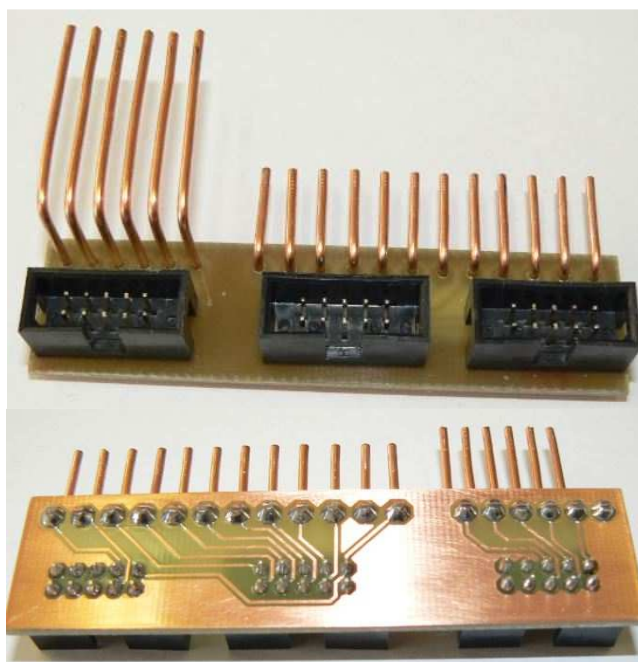
Obr.31 Deska řídicí elektroniky pohled z vrchní strany



Obr.32 Deska řídicí elektroniky pohled ze spodní strany



Obr.33 Vstupní deska PLC IN

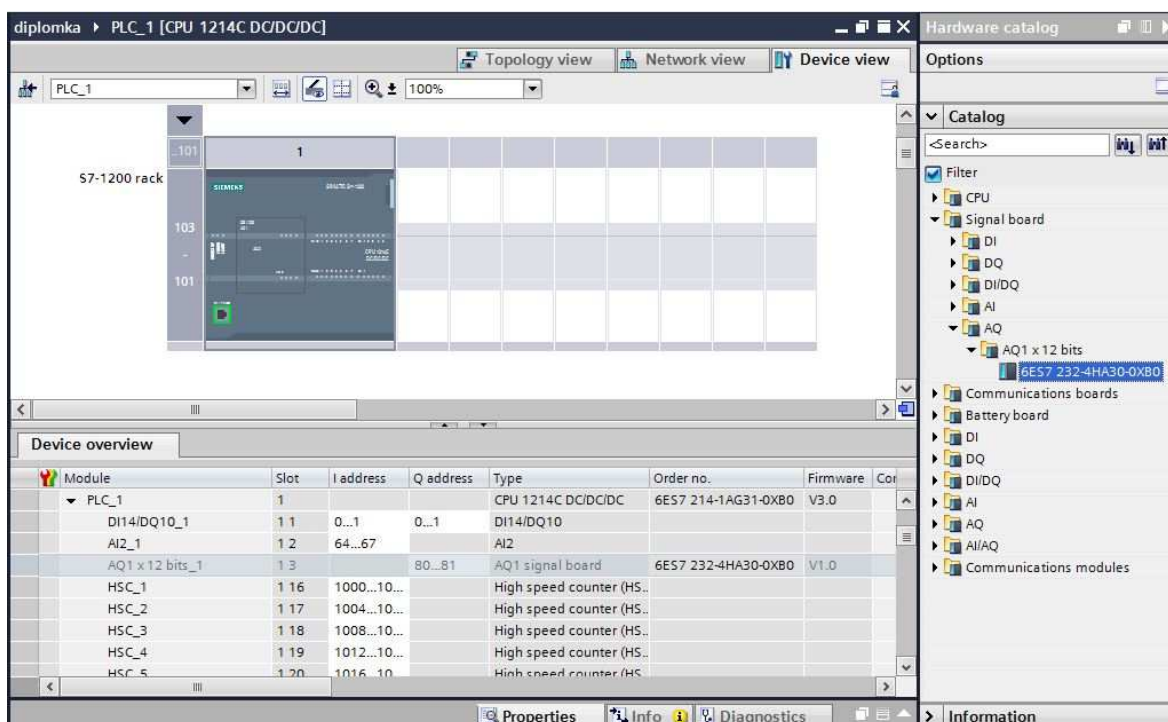


Obr.34 Výstupní deska PLC OUT

5. Programování PLC a HMI

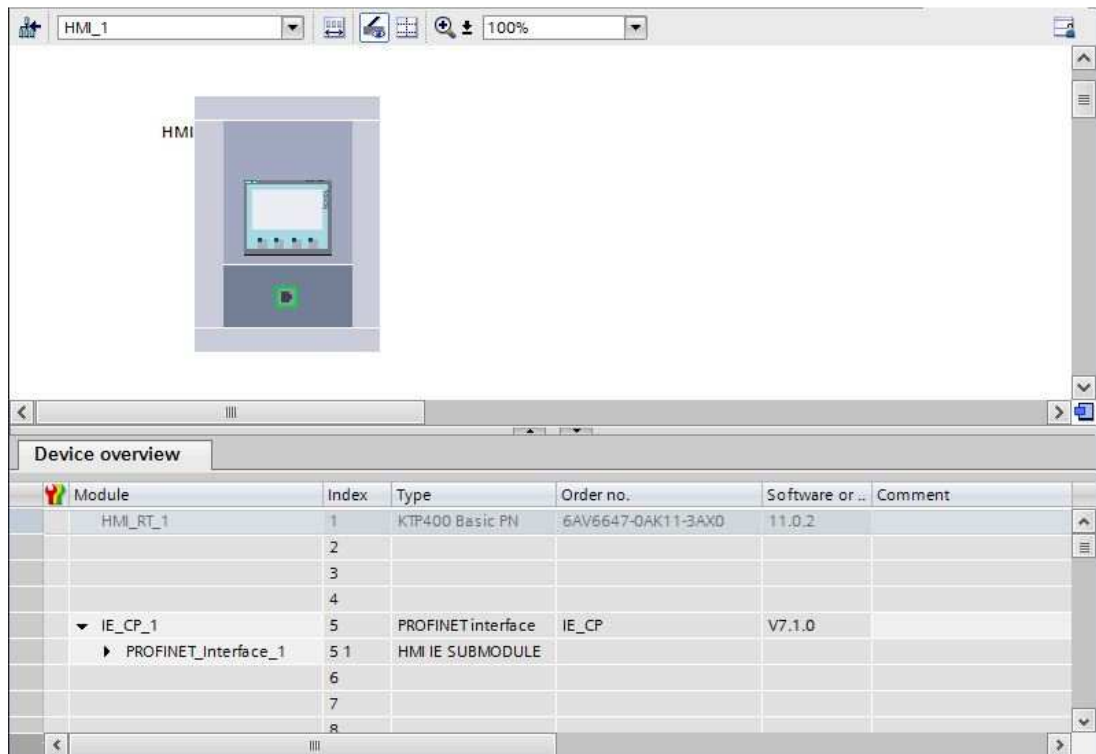
5.1. Konfigurace zařízení v TIA portálu

Při zakládání nového projektu bylo nejprve zapotřebí vybrat hardware. Volba nakonec spočinula na PLC 1214DC/DC/DC verze 6ES7 214-1AG31-0XB0 firmware v3.0. Tento automat byl následně nalezen v seznamu, přidán a nakonfigurován. Následně bylo ještě zapotřebí přidat signál board. Toto zařízení se přidává samostatně a bylo taktéž vybráno ze seznamu, verze 6ES7 232-4HA30-0XB0 firmware v1.0



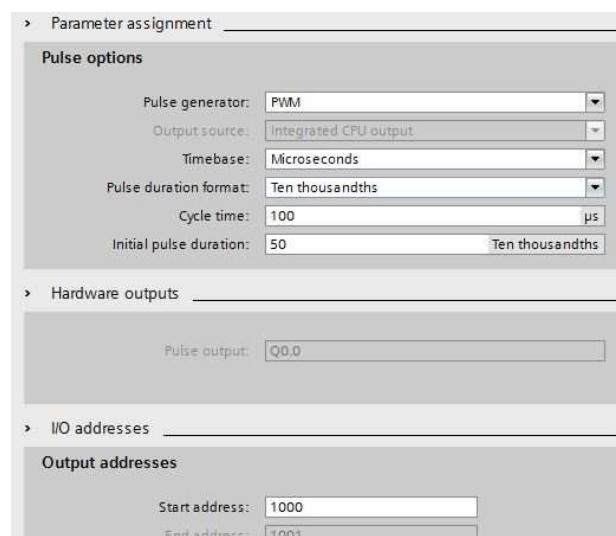
Obr.35 TIA portál nastavení PLC1214DC/DC/DC

Dále bylo nutné přidat HMI panel, ale při jeho výběru bylo zjištěno, že není v seznamu zařízení. Z toho důvodu bylo jedinou možnou variantou vyhledat zařízení KTP400 Basic color na stránkách výrobce a stáhnout potřebný ovladač. Následně byl HMI panel doinstalován do TIA portálu a mohlo se přistoupit k výběru zařízení: verze 6AV6647-0AK11-3AX0 firmware v11.0.2.



Obr.36 TIA portál nastavení KTP400 Basic color

Před zahájením programování bylo třeba nakonfigurovat PWM výstup a vysokorychlostní čítač. PWM výstup 1 byl nejprve v nastavení povolen a jeho název nastaven na PWM1. Bylo i nezbytné zvolit mezi PTO a PWM, přičemž byla vybrána PWM. Drobné úpravy se týkaly změny časové základny na mikrosekundy, počet pulzů na desetitisíce, periodu spínání 100 μ s (10kHz) a výstupní adresu 1000 (QW100). Původně zamýšlená hodnota periody řízení PWM 25kHz byla zamítnuta z důvodu větších ztrát při přemagnetování, i když její výhodou byl pro lidské ucho neslyšitelný zvuk. Z důvodu nižších ztrát byla zvolena frekvence řízení 10kHz.



Obr.37 Nastavení PWM

HSC1 (vysokorychlostní čítač 1) bylo třeba v nastavení povolit a jeho název změnit na HSC_1. Dalším krokem byl výběr typu načítání na pulzy a operačního módu na AB kvadraturní. I když bylo na výběr z více možností, touto volbou bylo docíleno největší citlivosti. Na závěr byla nastavena adresa vstupu na 1000 (ID1000).

Enable this high speed counter

Project information

Name: HSC_1

Comment:

> Function

Type of counting: Count

Operating phase: AB quadrature phase 4X

Input source: Integrated CPU input

Counting direction is specified by: Input (external direction control)

Initial counting direction: Count up

Frequency measuring period: + sec

Obr.38 Nastavení vysokorychlostního čítače

5.2. Programování regulátoru

Programovatelný automat obsahoval vestavěný PID regulátor, avšak tento regulátor nebyl využit. Místo něho byl zvolen postup naprogramování vlastního regulátoru, kterého bylo možné využít vícekrát (stačilo naprogramovat pouze jednou), formou funkčního bloku. Při programování byl nastaven vstupní interface, přes který vstupovaly hodnoty pro výpočet do regulátoru. Regulátor byl naprogramován podle této rovnice:

$$G_r(s) = r_0 \left(1 + \frac{1}{T_i s} + T_d s \right) [26]$$

Interface				
	Name	Data type	Def...	Comment
1	▼ Input			
2	xEnable	Bool	false	0=nic, 1=reset
3	xReset	Bool	true	0=nic, 1=reset
4	xInv	Bool	false	0=nic, 1=inverze
5	rZadanaHodnota	Real	0.0	žádaná hodnota regulátoru
6	rSkutecnaHodnota	Real	0.0	skutečná hodnota regulátoru
7	rIntegracniSlozkaMin	Real	0.0	dolní omezení I složky
8	rIntegracniSlozkaMax	Real	0.0	horní omezení I složky
9	rAkcniZasahMin	Real	0.0	dolní omezení akčního zásahu
10	rAkcniZasahMax	Real	0.0	horní omezení akčního zásahu
11	rKp	Real	0.0	zesílení P složky
12	rTi	Real	0.0	časová konstanta I složky
13	rTd	Real	0.0	časová konstanta D složky
14	rTvz	Real	0.0	vzorkovací perioda v sekundach
15	▼ Output			
16	rAkcniZasah	Real	0.0	akční zásah regulátoru
17	► InOut			
18	▼ Static			
19	rCi	Real	0.0	integrační konstanta regulátoru
20	rCd	Real	0.0	derivační konstanta regulátoru
21	rRegulacniOdchylka_1	Real	0.0	regulační odchylka v case t-1
22	rRegulacniOdchylka	Real	0.0	regulační odchylka v case t
23	rProporcionalniSlozka	Real	0.0	akční zásah P složky
24	rIntegracniSlozka_pre...	Real	0.0	předchozí integrační složka
25	rIntegracniSlozka	Real	0.0	nová integrační složka
26	rDerivacniSlozka	Real	0.0	akční zásah D složky

Obr.39 Interface funkčního bloku PID regulátoru

Naprogramovaný regulátor:

```
//slouží propřepnutí resetu
IF #xReset THEN
  //přiřazení hdonoty
  #xReset := FALSE;
END_IF;

//výpočet integrační konstanty regulátoru
#rCi := #rKp * #rTvz / #rTi;
//výpočet derivační konstanty regulátoru
#rCd := #rKp * #rTd / #rTvz;
//výpočet regulační odchylky
//přiřazení hodnoty v čase T-1
#rRegulacniOdchylka_1 := #rRegulacniOdchylka;
//výpočet odchylky v čase T
#rRegulacniOdchylka := #rZadanaHodnota - #rSkutecnaHodnota;
//výpočet proporcionalni složky
#rProporcionalniSlozka := #rKp * #rRegulacniOdchylka;
// výpočet integrační složky
// přiřazení hodnoty v čase T-1
#rIntegracniSlozka_predchozi := #rIntegracniSlozka;
// výpočet I složky v čase T
```



```

#rIntegracniSlozka := #rIntegracniSlozka_predchozi + (#rCi * #rRegulacniOdchylka);
//omezeni integracni slozky
// podmínka pro dolní omezení I složky
IF #rIntegracniSlozka < #rIntegracniSlozkaMin THEN
  //přiřazení dolní hranice omezení
  #rIntegracniSlozka := #rIntegracniSlozkaMin;
END_IF;
//podmínka pro horní omezení I složky
IF #rIntegracniSlozka > #rIntegracniSlozkaMax THEN
  //přiřazení horní hranice omezení
  #rIntegracniSlozka := #rIntegracniSlozkaMax;
END_IF;

```

5.3. Definice proměnných

Nejdůležitější částí každého programování programovatelných automatů je vhodná volba proměnných, včetně datových typů a jejich nadefinování. Cílem tohoto kroku je, aby byl program přehledný a srozumitelný i pro jiné osoby, které by s tímto kódem pracovali. Je žádoucí patřičné okomentování proměnných i kódu.

PLC tags					
	Name	Tag table	Data type	Address ▲	Comment
1	xStartMereni	Default tag table	Bool	%I1.0	Tlačítko pro zahájení měření
2	xResetPomocne	Default tag table	Bool	%I1.1	Tlačítko pro reset proměnné
3	xStartkopirovani	Default tag table	Bool	%I1.2	Tlačítko pro zapnutí kopírování dat
4	xResetProPrekopirovani	Default tag table	Bool	%I1.3	Tlačítko pro reset proměnných pro kopírování dat
5	HSC1_in	Default tag table	DWord	%ID1000	HSC hodnota na snímači
6	xDirection	Default tag table	Bool	%Q0.1	Přepínání směru otáčení motoru
7	xBreak	Default tag table	Bool	%Q0.2	Spínání brzdy
8	PWM_channel_1	Default tag table	Word	%QW1000	Zápis hodnoty rychlosti do registru
9	xStart	Default tag table	Bool	%M0.0	Zapínání / vypínání motoru
10	xZapis	Default tag table	Bool	%M0.1	Pomocná proměnná při kopírování dat do csv (přechod)
11	xSetStartkopirovani	Default tag table	Bool	%M0.2	Proměnná pro zahájení kopírování dat do csv
12	xZapisSkok	Default tag table	Bool	%M0.3	Pomocná proměnná při kopírování dat do csv (skok)
13	iRezim	Default tag table	Int	%MW1	Volba regulace
14	ix	Default tag table	Int	%MW5	Pomocná proměnná pro přechodovou charakteristiku
15	iy	Default tag table	Int	%MW7	Pomocná proměnná pro přenos dat
16	ip	Default tag table	Int	%MW9	Pomocná proměnná při kopírování dat do csv (přechod)
17	iz	Default tag table	Int	%MW11	Pomocná proměnná při kopírování dat do csv (skok)
18	dwPosition	Default tag table	DWord	%MD100	Poloha v pulzech
19	dwPosition_1	Default tag table	DWord	%MD104	Poloha v pulzech v bloku T_005
20	rSpeed	Default tag table	Real	%MD108	Rychlost v pulzech za sec
21	rZadanaHodnotaPWM	Default tag table	Real	%MD112	Žádaná hodnota otáčení od 0 do 100%
22	rZadanaHodnotaSpeed	Default tag table	Real	%MD116	Žádaná hodnota rychlosti
23	rZadanaHodnotaPosition	Default tag table	Real	%MD120	Žádaná hodnota počtu otáček
24	rZasah	Default tag table	Real	%MD124	Regulační zásah
25	rOtacky	Default tag table	Real	%MD128	aktuální hodnota otáček
26	rOtacky_1	Default tag table	Real	%MD132	Předchozí hodnota otáček
27	rZasahpoloha	Default tag table	Real	%MD136	Akční zásah polohové regulace
28	rIRCPuls	Default tag table	Real	%MD140	Nastavení počtu pulsu na otáčku IRC snímače
29	rMotorotacky	Default tag table	Real	%MD144	Nastavení počtu otáček motoru
30	rRychlosZaSekundu	Default tag table	Real	%MD148	Proměnná pro zobrazení rychlosti za minutu
31	dwPozice_1	Default tag table	DWord	%MD152	Poloha v pulzech v čase t-1 blok T_010
32	dwPozice	Default tag table	DWord	%MD156	Poloha v pulzech v čase t blok T_010

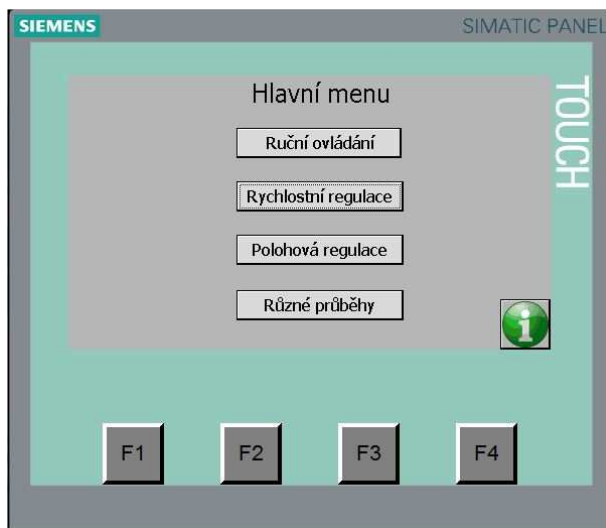
Obr.40 Tabulka definovaných proměnných

5.4. Konstrukce programu

Program obsahuje několik bloků, které jsou navzájem provázány. O počáteční nastavení proměnných se stará blok „Startup,” který se provede pouze jednou, a to po zapnutí automatu. Hlavní blok nazývaný „Main,” který je cyklicky volán, je využíván pro vytvoření a kopírování dat (pro přechodovou charakteristiku) do csv souborů. Dále jsou v programu dva bloky časově volané, a sice „T_005,” který je volán v časovém intervalu 5ms a „T_010,” který je volán v 10ms intervalu. Blok „T_010” je využíván pro polohovou regulaci (polohová smyčka) a pro zvyšování proměnných. V bloku „T_005” se nachází zbytek programu počínaje ručním řízením, rychlostní regulací, různými režimy řízení, spouštěním a přiřazováním hodnot do PWM, po rychlostní smyčce pro polohovou regulaci. Dále tu jsou datové oblasti DB sloužící pro ukládání dat při výpočtech nebo pro překopírování do csv. Datové oblasti „PID_DB_x” si systém vytvořil sám při vkládání funkčního bloku PID regulátoru. Datové oblasti „Datalog” a „DataProPrechod” byly vytvořeny ručním zadáním parametru do datové oblasti. „Datalog” slouží pro ukládání parametrů a stavů při vytváření a kopírování dat do csv souborů. „DataProPrechod” funguje za účelem ukládání dat (například rychlost otáčení a skoková změna), které mají být kopírovány do csv.

5.5. Jednotlivé režimy řízení

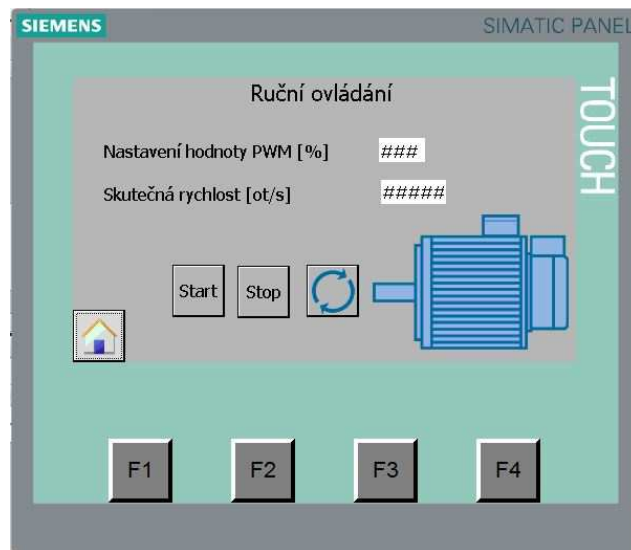
Tato část práce se zabývá pouze jednotlivými režimy bez programu. Program je samostatně zmíněn dále v práci. Volbu mezi jednotlivými režimy lze provádět pomocí HMI panelu („Hlavní menu”). Kliknutím na konkrétní tlačítka se lze dostat do jednotlivých režimů řízení, kterými lze ovládat motor.



Obr.41 HMI vizualizace - Hlavní menu

5.5.1. Režim ručního řízení (hodnoty využítá v programu)

Režim ručního řízení slouží pro ruční ovládání motoru. Lze jej nastavit na HMI panelu, velikost PWM od 0% do 100%. Při zadání vyšší hodnoty je automaticky brána hodnota 100%.

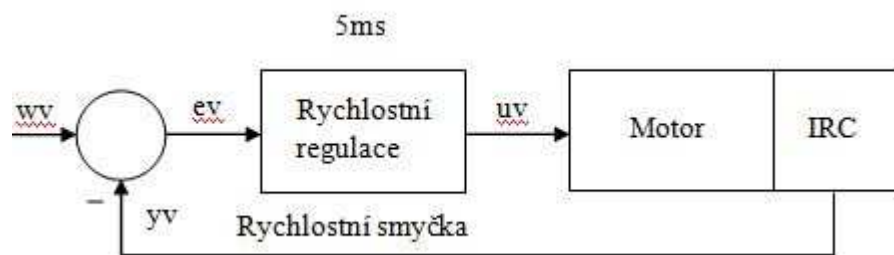


Obr.42 HMI vizualizace – Ruční ovládání

Na display lze sledovat skutečnou rychlost otáček motoru za sekundu. Směr otáčení motoru lze ovládat pomocí tlačítka (tlačítko s dvěma modrými šipkami napravo od tlačítka Stop).

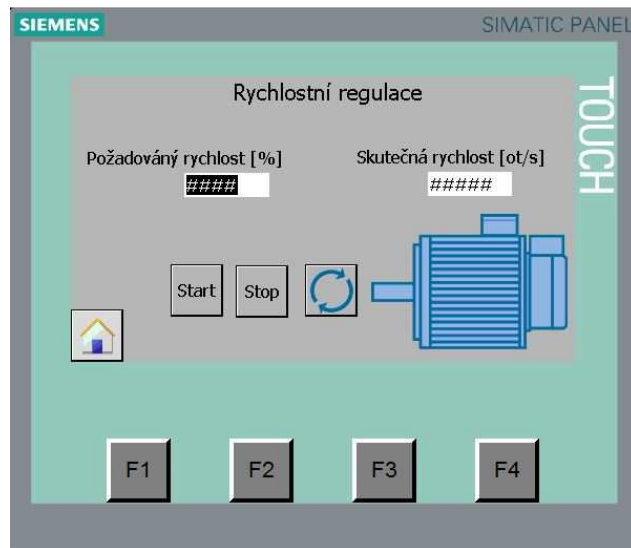
5.5.2. Režim rychlostního řízení

Režim rychlostního řízení slouží pro ovládání rychlosti otáček hřídele pomocí PSD regulace. Regulátor je téměř stejný jako PID. Jediný rozdíl je v tom, že je zapotřebí udávat vzorkovací periodu. Rychlostní smyčka pracuje v 5ms intervalu.



Obr.43 Regulační smyčka rychlostní regulace

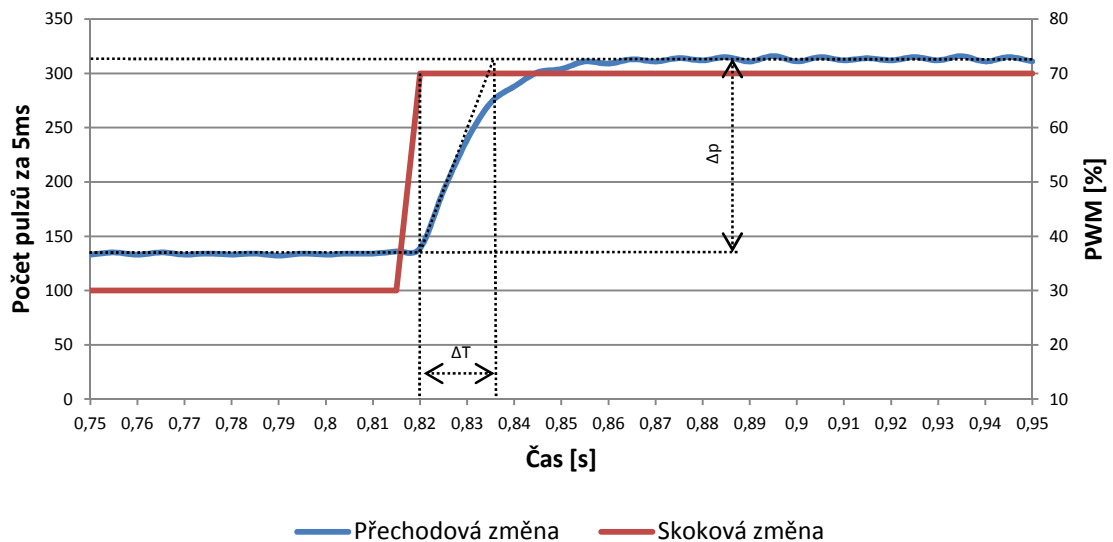
Rychlost otáčení motoru se zadává pomocí procentuálního vyjádření rychlosti otáčení. Pro změnu směru otáčení je třeba před požadovanou hodnotu zadat znaménko mínus. Výstupní hodnota, která je zobrazována na display, je počet otáček za sekundu.



Obr.44 HMI vizualizace – Rychlostní regulace

Hodnoty pro nastavení regulátoru byly vypočteny z přechodové charakteristiky. Ta byla odměřena pomocí ručního řízení. Měření probíhalo tak, že PWM bylo zvýšeno ze 30% na 70%. Tyto hodnoty byly zvoleny záměrně blízko sebe z důvodu toho, aby nebylo dosaženo maximálního odebraného proudu, nebo jiného omezení. Hrozilo by totiž zkreslení přechodové charakteristiky.

Přechodová charakteristika ze 30% na 70% PWM



Graf 1- Přechodová charakteristika ručního řízení (zvýšení rychlosti)

Skoková změna by v grafu měla být kolmá. Náklon skokové změny je dán měřením v intervalu 5 ms. Navíc je zobrazován velice krátký časový úsek (200 ms).

Výpočet přenosu z přechodové charakteristiky probíhal na základě těchto vzorců:

Výpočet času náběhu:

$$\Delta T = T_2 - T_1$$

Výpočet změny počtu pulzů:

$$\Delta p = p_2 - p_1$$

Výpočet skokové změny:

$$\Delta z = z_2 - z_1$$

Výpočet zesílení:

$$K_s = \frac{\Delta p}{\Delta z}$$

Výpočet přenosu:

$$G(z) = \frac{K_s}{\Delta T(z) + 1}$$

$$\Delta T = 0,836 - 0,82 = 0,016s$$

$$\Delta p = 312 - 132 = 180$$

$$\Delta z = 70 - 30 = 40$$

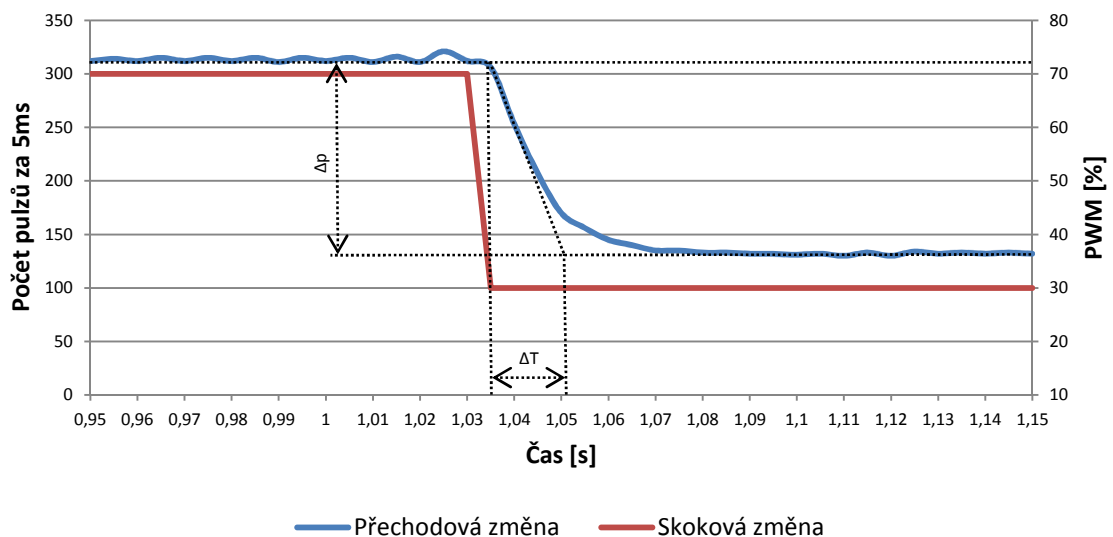
$$K_s = \frac{180}{40} = 4,5$$

Výsledný přenos:

$$G(z) = \frac{4,5}{0,016z + 1}$$

Pro kontrolu, že se řídicí elektronika (h-můstek) chová stejně jako při rozběhu, byla naměřena kontrolní přechodová charakteristika (snížení PWM ze 70% na 30%).

Přechodová charakteristika ze 70% na 30% PWM



Graf 2- Přechodová charakteristika ručního řízení (snížení rychlosti)

$$\Delta T = 1,051 - 1,035 = 0,016s$$

$$\Delta p = 132 - 312 = -180$$

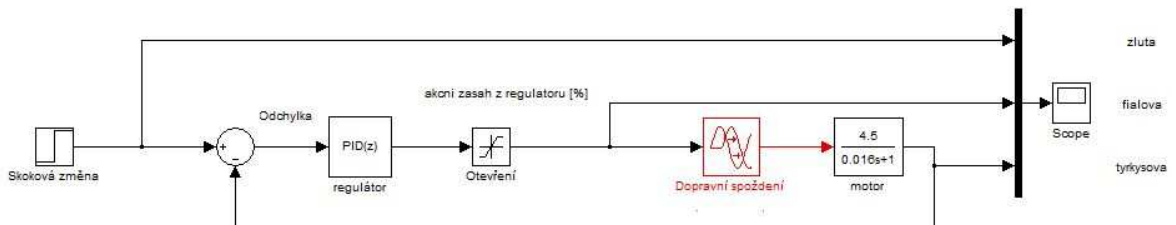
$$\Delta z = 30 - 70 = -40$$

$$K_s = \frac{-180}{-40} = 4,5$$

$$G(z) = \frac{4,5}{0,016z + 1}$$

Výsledné přenosy z přechodových charakteristik vyšly identicky, lze tedy říci, že h-mústek pracuje aktivně i při snížení PWM (brzdění).

Vypočítaný přenos byl použit při výpočtu hodnot nastavení regulátoru. Pro výpočet byl využit program Matlab s modulem Simulink, kde byl vytvořen model pro simulaci přenosu. V tomto modelu se vyskytuje PSD regulátor, který funguje v režimu diskretního řízení. Avšak tato práce se dále vyjadřuje o tomto regulátoru jako o PID regulátoru přepnutém do režimu diskretního řízení. Přitom se ale stále jedná o regulátor PSD. Regulátor obsahuje nástroj pro zjištění hodnot PID Tuning tool, kterým lze zjistit hodnoty a upravit dynamiku regulace.



Obr.45 Model vytvořeny v Simulinku pro výpočet parametrů regulátoru

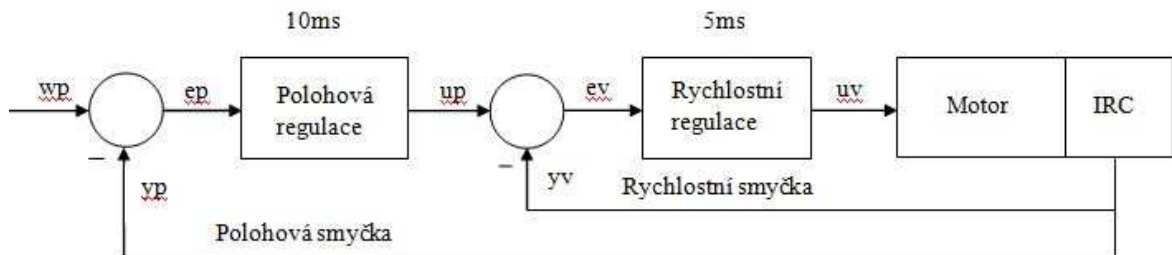
Po dosažení hodnoty přenosu a zjištěných hodnot nastavení regulátoru vyšly tyto hodnoty $P = 0,57515$ a $I = 75,44$, $D = 0$. Složku I bylo nezbytné přepočítat, protože naprogramovaný regulátor obsahoval integrační složku ve jmenovateli, zatímco vytvořený model v Simulinku obsahoval tuto složku v čitateli. Tudíž bylo nutné vydělit tuto hodnotu zesílením z přenosu.

$$\frac{4,5}{75,44} \cong 0,0597$$

Parametry rychlostního regulátoru jsou tyto **P = 0,57515**, **I = 0,0597** a **D = 0**.

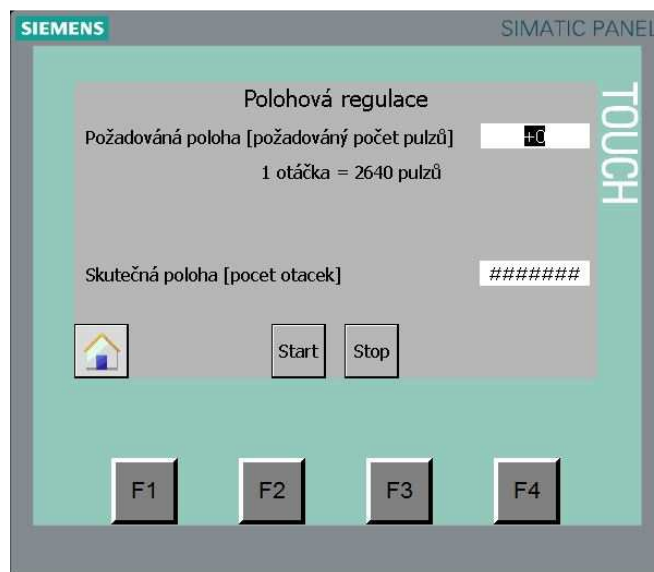
5.5.3. Režim polohového řízení

Režim polohového řízení slouží k natočení motoru na přesnou polohu nebo k docílení přesného počtu otáček, a tím docílení polohy (např. 30 otočení). Při ovládání regulace je zapotřebí dvou smyček regulace: rychlostní a polohové. Rychlostní smyčka pracuje s 5 ms vzorkovací periodou a polohová smyčka s 10 ms.



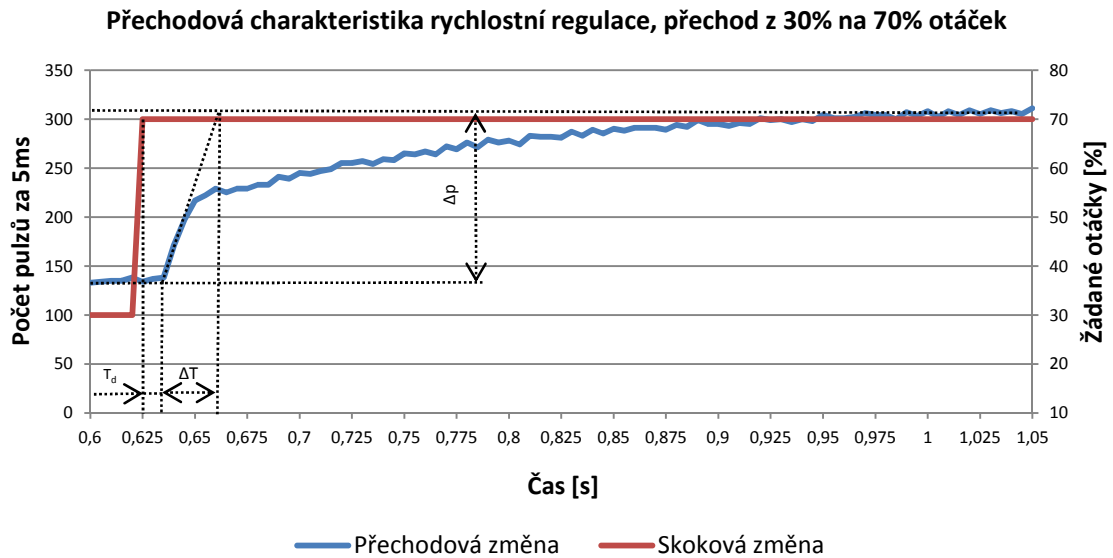
Obr.46 Regulační smyčky polohové regulace

Vstupní parametr udávaný na HMI panelu je požadovaný počet pulzů pro pootočení motoru. Směr otočení lze změnit tak, že se před udanou hodnotu vloží znaménko plus či mínus. Na display je dále zobrazována hodnota pulzů, o které se motor pootočil.



Obr.47 HMI vizualizace – Polohové regulace

Výpočet parametrů pro polohový regulátor probíhal obdobně jako výpočet pro rychlostní regulátor. Pomocí rychlostní regulace byla vytvořena přechodová charakteristika pro výpočet hodnot regulátoru z přenosu.



Graf 3- Přechodová charakteristika rychlostní regulace

Z přechodové charakteristiky vyplývá, že rychlostní regulace obsahuje dopravní zpoždění $T_d = 0,005s$.

Výpočet přenosu:

$$\Delta T = 0,665 - 0,630 = 0,035s$$

$$\Delta p = 312 - 132 = 180$$

$$\Delta z = 70 - 30 = 40$$

$$K_s = \frac{180}{40} = 4,5$$

$$G(z) = \frac{4,5 * e^{0,005}}{0,035z + 1}$$

Po vypočtení přenosu soustavy byla hodnota přenosu dosazena do Simulinku, kde pomocí PID Tuning tool byly zjištěny hodnoty nastavení regulátoru. Zjištěné hodnoty regulátoru jsou: $P = 0,1614$, $I = 50$ a $D = 0$. Opět bylo nutné přepočítat I složku jako v předešlém případě:

$$\frac{4,5}{50} \cong 0,09$$

Výsledné parametry polohového regulátoru pro polohovou smyčku jsou tyto **$P = 0,1614$, $I = 0,09$ a $D = 0$** . Parametry pro rychlostní regulátor byly použity stejné jako u rychlostního regulátoru, tedy následující: **$P = 0,57515$, $I = 0,0597$ a $D = 0$** .

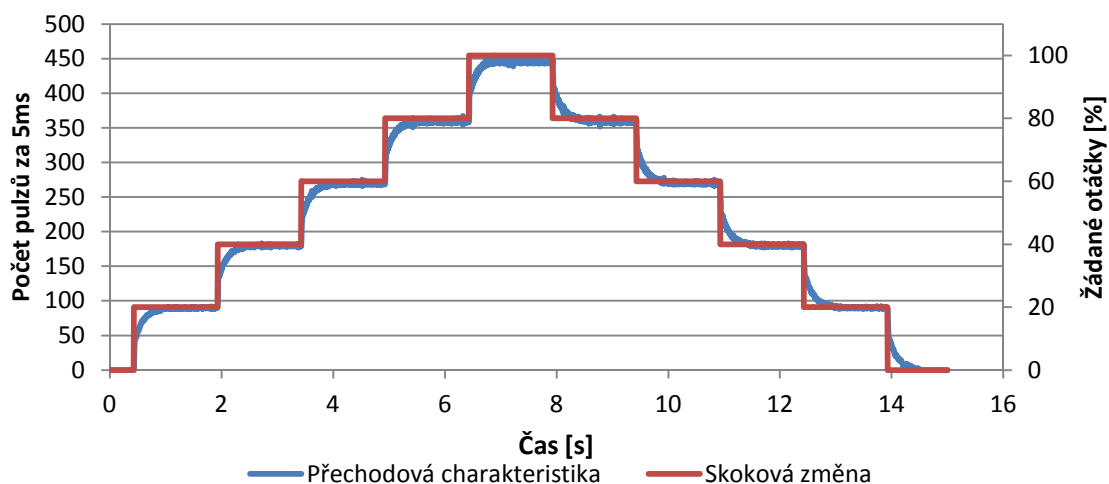
Režimy skokové změny a rampového průběhu byly naprogramovány, jako demonstrační příklad funkčnosti rychlostní regulace, kdy po uplynutí naprogramované

sekvence dojde k návratu na začátek cyklu a tím k opětovnému opakování. Jednotlivé režimy lze vybrat v hlavním menu tlačítkem „Různé průběhy,” kde je na výběr z výše zmiňovaných průběhů.

5.5.4. Režim skokové změny

Režim skokové změny byl naprogramován za účelem zobrazení své regulace v přechodové charakteristice, která vznikla naměřením tohoto skokového průběhu. Skoková změna byla naprogramována od 0% do 100% otáček motoru, a to po 20% skocích. Následně docházelo ke snižování, opět po 20%, až na 0%. Výdrž na jednotlivých skocích byla 1,5s. Parametry nastavení regulátoru byly stejné jako u rychlostní regulace.

Přechodová charakteristika skokových změn



Graf 4 - Přechodová charakteristika rychlostní regulace skokového průběhu

5.5.5. Režim rampového průběhu

Rampový průběh byl naprogramován pro demonstraci pozvolného zvyšování žádané hodnoty otáček rychlostního regulátoru. Pozvolným zvyšováním žádané hodnoty (od 0% otáček) po dobu 2s se dosáhlo maximálních otáček motoru. Na 100% otáček se setrvalo po dobu 4s. Následně byly otáčky postupně snižovány na nulu, opět po dobu 2s.

5.6. Popis programu v jednotlivých blocích

Celkem byly vytvořeny 4 bloky programů s rozdílným typem volání: po startu, časové volání s periodou 5ms a 10ms a cyklické.

5.6.1. Volání po startu (Startup)

Tento blok je volán pouze jednou, a sice ihned při spuštění programu v PLC, kdy dojde k inicializaci počátečního nastavení.

```
//počáteční nastavení PWM hodnoty
"rZadanaHodnotaPWM" :=0;
//proměnná pro volbu režimu
"iRezim":=0;
//proměnná pro polohovou regulaci
"rOtacky" :=0;
//proměnná pro polohovou regulaci t-1
"rOtacky_1":=0;
//počáteční nastavené pro kopírování dat do datové oblasti
"ix" :=0;
//počáteční nastavené pro kopírování dat do csv pro přechodovou char.
"iy" :=0;
//počáteční nastavené pro kopírování dat do csv pro skok
"ip" :=0;
// počáteční nastavení púlzu na otáčku u IRC
"rIRCPuls" := 660;
//počáteční nastavené proměnné pro různé průběhy
"iz":=0;
```

5.6.2. Časově volaný blok 1 (T_005)

Tento blok je cyklicky volán vždy v časovém intervalu 5ms (má vyšší prioritu než cyklicky volaný program). V tomto bloku byly naprogramovány veškeré režimy včetně rychlostní smyčky u polohové regulace.

```
// spuštění bloku HSC pro měření pulzu
"CTRL_HSC_0_DB"(HSC:="HSC_1");

//výpočet skutečné rychlosti
//uložení předchozí hodnoty polohy do proměnné
"dwPosition_1" := "dwPosition";
//uložení nýnější hodnoty polohy do proměnné
"dwPosition" := "HSC1_in";
//výpočet rychlosti s převode do real
"rSpeed" := DINT_TO_REAL(DWORD_TO_DINT("dwPosition" - "dwPosition_1"));

//Přepočet do rozsahu 0-90000 imp/s
//vynásobení rychlosti 200
"rSpeed" := "rSpeed" * 200.0;
//výpočet otáček za sekundu
"rRychlosZaSekundu":=("rSpeed"/(4*"rIRCPuls"));

//Přepočet do rozsahu 0-100 procent
"rSpeed" := "rSpeed" / 900.0;

// reset pomocné poroměnné pro zběr dat do datové oblasti
IF "xResetPomocne" = TRUE THEN
  //nastavení proměnné na 0
  "ix" :=0;
END_IF;

//Zahájení měření dat pro přechodovou charakteristiku
IF "xStartMereni" = TRUE THEN
  //podmínka pro zběr dat
  IF "ix" < 3300 THEN
    //uložení hodnoty přechodu do datové oblasti
    "DataProPrechod".Motor.poloha["ix"] :="dwPosition";
    //uložení hodnoty skoku do datové oblasti
```

```

        "DataProPrechod".MotorZmenaZadane.Zmena["ix"]:= "rZadanaHodnotaSpeed";
        //zvýšení proměnné o jedna
        "ix" := "ix" + 1;
    END_IF;
END_IF;

//Režim ručního řízení
IF "iRezim" = 0 THEN
    //přířazení žádané rychlosti
    "rZasah" := "rZadanaHodnotaPWM" * 100;
END_IF;

//Režim rychlostní
IF "iRezim" = 1 THEN
    //zavolání funkčního bloku regulátoru
    "PID_DB_1" (//povolení regulátoru
        xEnable:=TRUE,
        //reset regulátoru
        xReset:=FALSE,
        //inverze regulátoru
        xInv:=FALSE,
        //žádaná hodnota vynásobená 100
        rZadanaHodnota:=("rZadanaHodnotaSpeed"*100),
        //skutečná hodnota vynásobena 100
        rSkutecnaHodnota:= ("rSpeed"*100),
        //dolní omezení I složky
        rIntegracniSlozkaMin:=-10000,
        //horní omezení I složky
        rIntegracniSlozkaMax:=+10000,
        //dolní omezení akčního zásahu
        rAkcniZasahMin:=-10000,
        //horní omezení akčního zásahu
        rAkcniZasahMax:=+10000,
        //P konstanta
        rKp:=0.57515,
        //I konstanta
        rTi:=0.0597,
        //D konstanta
        rTd:=0.0,
        //časová konstanta
        rTvz:=0.005,
        //přířazení akčního zásahu do zásahu
        rAkcniZasah=>"rZasah");

    //nastavení směru otáčení motoru
    IF "rZasah" >= 0 THEN
        //otáčení ve směru hodinových ručiček
        "xDirection" := FALSE;
    ELSE
        //otáčení proti směru hodinových ručiček
        "xDirection" := TRUE;
    END_IF;
END_IF;

//Režim polohy
IF "iRezim" = 2 THEN
    //zavolání funkčního bloku regulátoru rychlostní smyčky
    "PID_DB_3" (//povolení regulátoru
        xEnable:=TRUE,
        //reset regulátoru
        xReset:=FALSE,
        //inverze regulátoru
        xInv:=FALSE,
        //přířazení žádané hodnoty
        rZadanaHodnota:="rZasahpoloha",
        //skutečná hodnota vynásobena 100
        rSkutecnaHodnota:= ("rSpeed"*100),
        //dolní omezení I složky
        rIntegracniSlozkaMin:=-10000,
        //horní omezení I složky
        rIntegracniSlozkaMax:=+10000,
        //dolní omezení akčního zásahu

```

```
        rAkcniZasahMin:=-10000,
        //horní omezení akčního zásahu
        rAkcniZasahMax:=+10000,
        //P konstanta
        rKp:=0.57515,
        //I konstanta
        rTi:=0.0597,
        //D konstanta
        rTd:=0.0,
        //časová konstanta
        rTvz:=0.005,
        //přiřazení akčního zásahu do zásahu
        rAkcniZasah=>"rZasah");
END_IF;

//Režim skokové změny
IF "iRezim" = 3 THEN
    //spuštění PWM
    IF "xStart" = true THEN
        //přičtení 1
        "iz":="iz" + 1;
        //do času 1,5s
        IF "iz" <= 300 THEN
            //přiřazení hodnoty 20
            "rZadanaHodnotaSpeed":=20;
        END_IF;
        //od času 1,5s do času 3s
        IF "iz" > 300 AND "iz" <= 600 THEN
            //přiřazení hodnoty 40
            "rZadanaHodnotaSpeed":=40;
        END_IF;
        //od času 3s do času 4,5s
        IF "iz" > 600 AND "iz" <= 900 THEN
            //přiřazení hodnoty 60
            "rZadanaHodnotaSpeed":=60;
        END_IF;
        //od času 4,5s do času 6s
        IF "iz" > 900 AND "iz" <= 1200 THEN
            //přiřazení hodnoty 80
            "rZadanaHodnotaSpeed":=80;
        END_IF;
        //od času 6s do času 7,5s
        IF "iz" > 1200 AND "iz" <= 1500 THEN
            //přiřazení hodnoty 100
            "rZadanaHodnotaSpeed":=100;
        END_IF;
        //od času 7,5s do času 9s
        IF "iz" > 1500 AND "iz" <= 1800 THEN
            //přiřazení hodnoty 80
            "rZadanaHodnotaSpeed":=80;
        END_IF;
        //od času 9s do času 10,5s
        IF "iz" > 1800 AND "iz" <= 2100 THEN
            //přiřazení hodnoty 60
            "rZadanaHodnotaSpeed":=60;
        END_IF;
        //od času 10,5s do času 12s
        IF "iz" > 2100 AND "iz" <= 2400 THEN
            //přiřazení hodnoty 40
            "rZadanaHodnotaSpeed":=40;
        END_IF;
        //od času 12s do času 13,5s
        IF "iz" > 2400 AND "iz" <= 2700 THEN
            //přiřazení hodnoty 20
            "rZadanaHodnotaSpeed":=20;
        END_IF;
        //od času 13,5s DO času 15s
        IF "iz" > 2700 AND "iz" <= 3000 THEN
            //přiřazení hodnoty 0
            "rZadanaHodnotaSpeed":=0;
        END_IF;
        //vyšší čas než 15s
```

```

IF "iz"> 3000 THEN
  //přiřazení hodnoty 0
  "iz":=0;
END_IF;
//zavolání funkčního bloku regulátoru
"PID_mereni"(/povolení regulátoru
  xEnable:=TRUE,
  //reset regulátoru
  xReset:=FALSE,
  //inverze regulátoru
  xInv:=FALSE,
  //žádaná hodnota vynásobená 100
  rZadanaHodnota:=( "rZadanaHodnotaSpeed"*100),
  //skutečná hodnota vynásobena 100
  rSkutecnaHodnota:= ( "rSpeed"*100),
  //dolní omezení I složky
  rIntegracniSlozkaMin:=-10000,
  //horní omezení I složky
  rIntegracniSlozkaMax:=+10000,
  //dolní omezení akčního zásahu
  rAkcnizasahMin:=-10000,
  //horní omezení akčního zásahu
  rAkcnizasahMax:=+10000,
  //P konstanta
  rKp:=0.57515,
  //I konstanta
  rTi:=0.0597,
  //D konstanta
  rTd:=0.0,
  //časová konstanta
  rTvz:=0.005,
  //přiřazení akčního zásahu do zásahu
  rAkcnizasah=>"rZasah");
//nastavení směru otáčení motoru
IF "rZasah" >= 0 THEN
  //otáčí ve směru hodinových ručiček
  "xDirection":=FALSE;
ELSE
  //otáčení proti směru hodinových ručiček
  "xDirection":=TRUE;
END_IF;
END_IF;

//Režim rampového průběhu
IF "iRezim" = 4 THEN
  //spuštění PWM
  IF "xStart" = True THEN
    //přičtení 1
    "iz":="iz" + 1;
    //do času 2s
    IF "iz" <= 400 THEN
      //přiřazení hodnoty 0
      "rZadanaHodnotaSpeed":=0;
    END_IF;
    //od času 2s do času 4s
    IF "iz" > 400 AND "iz" <=800 THEN
      //zvýšení hodnoty o 0,25
      "rZadanaHodnotaSpeed":="rZadanaHodnotaSpeed" + 0.25;
    END_IF;
    //od času 4s do času 8s
    IF "iz" > 800 AND "iz" <=1600 THEN
      //přiřazení hodnoty 0
      "rZadanaHodnotaSpeed":=100;
    END_IF;
    //od času 8s do času 10s
    IF "iz" > 1600 AND "iz" <=2000 THEN
      //snížení hodnoty o 0,25
      "rZadanaHodnotaSpeed":="rZadanaHodnotaSpeed" - 0.25;
    END_IF;
    //vyšší čas než 10s
    IF "iz" > 2001 THEN
      //přiřazení hodnoty 0

```

```

    "iz":=0;
    //přiřazení hodnoty 0
    "rZadanaHodnotaSpeed":=0;
END_IF;
//zavolání funkčního bloku regulátoru
"PID_mereni" (//povolení regulátoru
    xEnable:=TRUE,
    //reset regulátoru
    xReset:=FALSE,
    //inverze regulátoru
    xInv:=FALSE,
    //žádaná hodnota vynásobená 100
    rZadanaHodnota:=( "rZadanaHodnotaSpeed"*100),
    //skutečná hodnota vynásobena 100
    rSkutecnaHodnota:= ( "rSpeed"*100),
    //dolní omezení I složky
    rIntegracniSlozkaMin:=-10000,
    //horní omezení I složky
    rIntegracniSlozkaMax:=+10000,
    //dolní omezení akčního zásahu
    rAkcnizasahMin:=-10000,
    //horní omezení akčního zásahu
    rAkcnizasahMax:=+10000,
    //P konstanta
    rKp:=0.57515,
    //I konstanta
    rTi:=0.0597,
    //D konstanta
    rTd:=0.0,
    //časová konstanta
    rTvz:=0.005,
    //přiřazení akčního zásahu do zásahu
    rAkcnizasah=>"rZasah");
ELSE
    //přiřazení hodnoty 0
    "iz":=0;
END_IF;
END_IF;

//přiřazení absolutní hodnoty zásahu
"rZasah":=ABS("rZasah");
//uložení hodnoty do PWM
"PWM_channel_1" := DWORD_TO_WORD(REAL_TO_UINT("rZasah"));
//spuštění bloku PWM
"CTRL_PWM_DB" (PWM:="PWM1[PTO/PWM]", ENABLE:="xStart");

```

5.6.3. Časově volaný blok 2 (T_010)

Tento blok je cyklicky volán vždy v časovém intervalu 10ms (má vyšší prioritu než cyklicky volaný program). V tomto bloku byla naprogramována polohová regulace (polohová smyčka) a dále podmínky sloužící ke kopírování dat.

Naprogramovaná polohová regulace

```

//režim polohové regulace
IF "iRezim" = 2 THEN

    //Výpočet počtu otáček pro polohovou smyčku
    //uložení předchozí hodnoty polohy do proměnné T-1
    "dwPozice_1" := "dwPozice";
    //uložení nynějšší hodnoty polohy do proměnné T
    "dwPozice" := "HSC1_in";
    //výpočet rychlosti otáčení
    "rOtacky" := DINT_TO_REAL(DWORD_TO_DINT("dwPozice" - "dwPozice_1"));
    //Připočítávání otáček v každém cyklu
    "rOtacky_1" := "rOtacky_1" + "rOtacky";

```

```

//podmínka pro připočítávání otáček
IF "rZadanaHodnotaPosition" >= 0 THEN
  //podmínka když jsou otáčky záporné
  IF "rOtacky_1" < "rZadanaHodnotaPosition" THEN
    //přiřazení hodnoty
    "xDirection" := FALSE;
  ELSE
    //přiřazení hodnoty
    "xDirection" := TRUE;
  END_IF;
ELSE
  //podmínka když jsou otáčky kladné
  IF "rOtacky_1" >= "rZadanaHodnotaPosition" THEN
    //přiřazení hodnoty
    "xDirection" := TRUE;
  ELSE
    //přiřazení hodnoty
    "xDirection" := FALSE;
  END_IF;
END_IF;
//zavolání funkčního bloku regulátoru polohové smýčky
"PID_DB_4" (//povolení regulátoru
  xEnable := TRUE,
  //reset regulátoru
  xReset := FALSE,
  //inverze regulátoru
  xInv := FALSE,
  //žádaná hodnota vynásobená 100
  rZadanaHodnota := ("rZadanaHodnotaPosition" * 100),
  //skutečná hodnota vynásobena 100
  rSkutecnaHodnota := ("rOtacky_1" * 100),
  //dolní omezení I složky
  rIntegracniSlozkaMin := -10000,
  //horní omezení I složky
  rIntegracniSlozkaMax := 10000,
  //dolní omezení akčního zásahu
  rAkcniZasahMin := -10000,
  //horní omezení akčního zásahu
  rAkcniZasahMax := 10000,
  //P konstanta
  rKp := 0.1614,
  //I konstanta
  rTi := 0.09,
  //D konstanta
  rTd := 0.0,
  //časová konstanta
  rTvz := 0.010,
  //přiřazení akčního zásahu do zásahu
  rAkcniZasah => "rZasahpoloha");
END_IF;

```

Podmínka slouží k přepnutí na následující krok - kopírování dat skok a uzavření souboru přechod.

```

IF "iy" = 3300 AND "Datalog".StepSkok = 20 THEN
  //přiřazení hodnoty
  "Datalog".Step := 30;
  //přiřazení hodnoty
  "Datalog".StepSkok := 25;
END_IF;

```

Podmínka pro vytváření pulzů pro REQ (zmíněno dále):

```

IF "xSetStartkopirovani" = true THEN
  //podmínka pro kopírování přechodu
  IF "iy" < 3300 THEN
    //přiřazení hodnoty
    "xZapis" := True;
  END_IF;
  //podmínka pro kopírování skoku
  IF "ip" < 3300 THEN
    //přiřazení hodnoty

```

```

    "xZapisSkok" := True;
  END_IF;
END_IF;

```

Podmínka pro přepnutí na další krok (uzavření souboru skok) a ukončení kopírování dat.

```

IF "ip" = 3300 THEN
  //přiřazení hodnoty
  "Datalog".StepSkok := 30;
  //přiřazení hodnoty
  "xSetStartkopirovani" := false;
END_IF;

```

5.6.4. Cyklicky spuštěný blok (Main)

Tento blok slouží výhradně k zajištění kopírování dat do csv souborů, protože jsou vytvářeny dva soubory, jeden pro přechodovou a druhý pro skokovou charakteristiku. Tím pádem veškeré příkazy jsou zdvojeny. Rozdíl mezi nimi je, že každý odkazuje na jiný soubor. Z toho důvodu je níže zmíněna pouze první část kódu znázorňující práci se souborem a kopírováním dat. Druhá, zde nezmíněná, část kódu je však obdobná.

Naprogramované vytvoření a kopírování dat:

```

//podmínka pro tvorbu csv souboru přechod
IF "Datalog".Step = 0 THEN
  "DataLogCreate_prechod" (//pro spuštění procesu
    REQ="Datalog".CreateB.REQ,
    //počet hodnot v souboru
    RECORDS:=3300,
    //volba formátu
    FORMAT:=1,
    //časové razítko
    TIMESTAMP:=0,
    //název souboru
    NAME:="Datalog".navez,
    //hlášení o dokončení
    DONE=>"Datalog".CreateB.DONE,
    //hlášení o průběhu
    BUSY=>"Datalog".CreateB.BUSY,
    //chybové hlášení
    ERROR=>"Datalog".CreateB.ERROR,
    //hláškové stavy
    STATUS=>"Datalog".CreateB.STATUS,
    //Id souboru
    ID:="Datalog".ID_prechod,
    //hlavička souboru
    HEADER:="Datalog".Header,
    //data pro ukládání
    DATA:="DataProPrechod".NatoceniHridelu);

  //přiřazení opačné hodnoty
  "Datalog".CreateB.REQ := NOT "Datalog".CreateB.BUSY;
  // pokud se soubor vytvorí tak preskoci na dalsi krok
  IF "Datalog".CreateB.DONE THEN
    //přiřazení hodnoty
    "Datalog".Step := 10;
  END_IF;

  //zjistí zda existuje uz csv soubor kdyz existuje tak preskoci na dalsi krok
  IF "Datalog".CreateB.STATUS = W#16#8093 THEN
    //přiřazení hodnoty
    "Datalog".Step := 10;
  END_IF;
END_IF;

```



```

//podmínka pro otevření souboru
IF "Datalog".Step = 10 THEN
    "DataLogOpen_prechod" (//pro spuštění procesu
        REQ:="Datalog".OpenB.REQ,
        //typ otevření souboru pro zápis
        MODE:="Datalog".Mode,
        //název souboru
        NAME:="Datalog".navez,
        //hlášení o dokončení
        DONE=>"Datalog".OpenB.DONE,
        //hlášení o průběhu
        BUSY=>"Datalog".OpenB.BUSY,
        //chybové hlášení
        ERROR=>"Datalog".OpenB.ERROR,
        //hláškové stavy
        STATUS=>"Datalog".OpenB.STATUS,
        //Id souboru
        ID:="Datalog".ID_prechod);

    //přiřazení opáčné hodnoty
    "Datalog".OpenB.REQ := NOT "Datalog".OpenB.BUSY;

    // pokud se soubor otevře tak přeskočí na další krok
    IF "Datalog".OpenB.DONE THEN
        //přiřazení hodnoty
        "Datalog".Step := 20;
    END_IF;

    //podmínka zjišťující zda je soubor vytvořen
    IF "Datalog".OpenB.STATUS = W#16#8092 THEN
        //přiřazení hodnoty
        "Datalog".Step := 0;
    END_IF;
END_IF;

//podmínka pro otevření souboru přechod
IF "Datalog".Step = 20 THEN
    "DataLogWrite_prechod" (//pro spuštění procesu
        REQ:="Datalog".WriteB.REQ,
        //hlášení o dokončení
        DONE=>"Datalog".WriteB.DONE,
        //hlášení o průběhu
        BUSY=>"Datalog".WriteB.BUSY,
        //chybové hlášení
        ERROR=>"Datalog".WriteB.ERROR,
        //hláškové stavy
        STATUS=>"Datalog".WriteB.STATUS,
        //Id souboru
        ID:="Datalog".ID_prechod);

    //přiřazení hodnoty
    "Datalog".WriteB.REQ := "xZapis" AND NOT "Datalog".WriteB.BUSY;
    //přiřazení dat
    "xZapis" := false;

    //podmínka pro zvýšení hodnoty při úspěšném zápisu dat
    IF "Datalog".WriteB.STATUS = W#16#0000 THEN
        //zvýšení proměnné o 1
        "iy" := "iy" + 1;
        //přiřazení dat
        "DataProPrechod".NatoceniHridelu := "DataProPrechod".Motor.poloha["iy"];
    END_IF;
END_IF;

//podmínka pro uzavření souboru
IF "Datalog".Step = 30 THEN
    "DataLogClose_DB" (//pro spuštění procesu
        REQ:="Datalog".CloseB.REQ,
        //hlášení o dokončení
        DONE=>"Datalog".CloseB.DONE,
        //hlášení o průběhu
        BUSY=>"Datalog".CloseB.BUSY,

```

```
        //chybové hlášení
        ERROR=>"Datalog".CloseB.ERROR,
        //hláškové stavy
        STATUS=>"Datalog".CloseB.STATUS,
        //Id souboru
        ID:="Datalog".ID_prechod);

//přiřazení opáčné hodnoty
"Datalog".CloseB.REQ := NOT "Datalog".CloseB.BUSY;

//podmínka pro ukončení zavírání souboru
IF "Datalog".CloseB.DONE THEN
    //přiřazení hodnoty
    "Datalog".Step := 40;
END_IF;

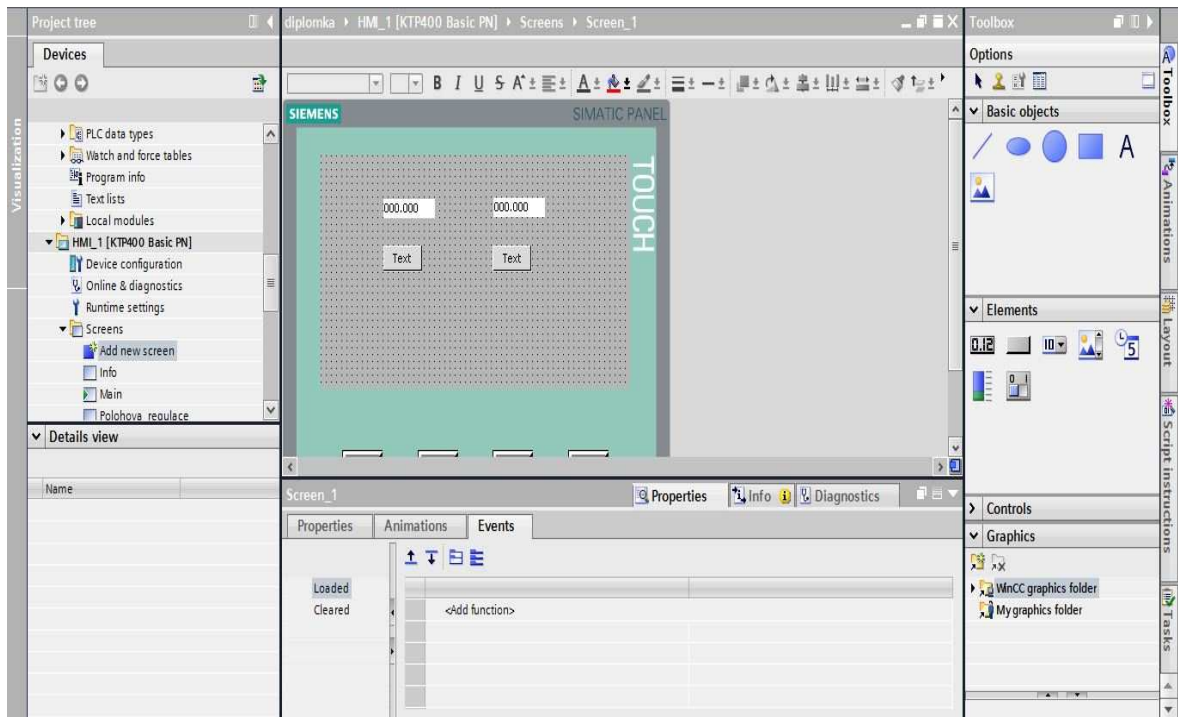
END_IF;

//podmínka startu kopírování dat (set)
IF "xStartkopirovani" = true THEN
    //přiřazení hodnoty
    "xSetStartkopirovani" := True;
END_IF;

//Podmínka resetu kopírování dat (návrat na začátek pro nové kopírování)
IF "xResetProPrekopirovani" = True THEN
    //přiřazení hodnoty
    "iy" := 0;
    //přiřazení hodnoty
    "Datalog".Step := 0;
    //přiřazení hodnoty
    "ip" := 0;
    //přiřazení hodnoty
    "Datalog".StepSkok :=0;
END_IF;
```

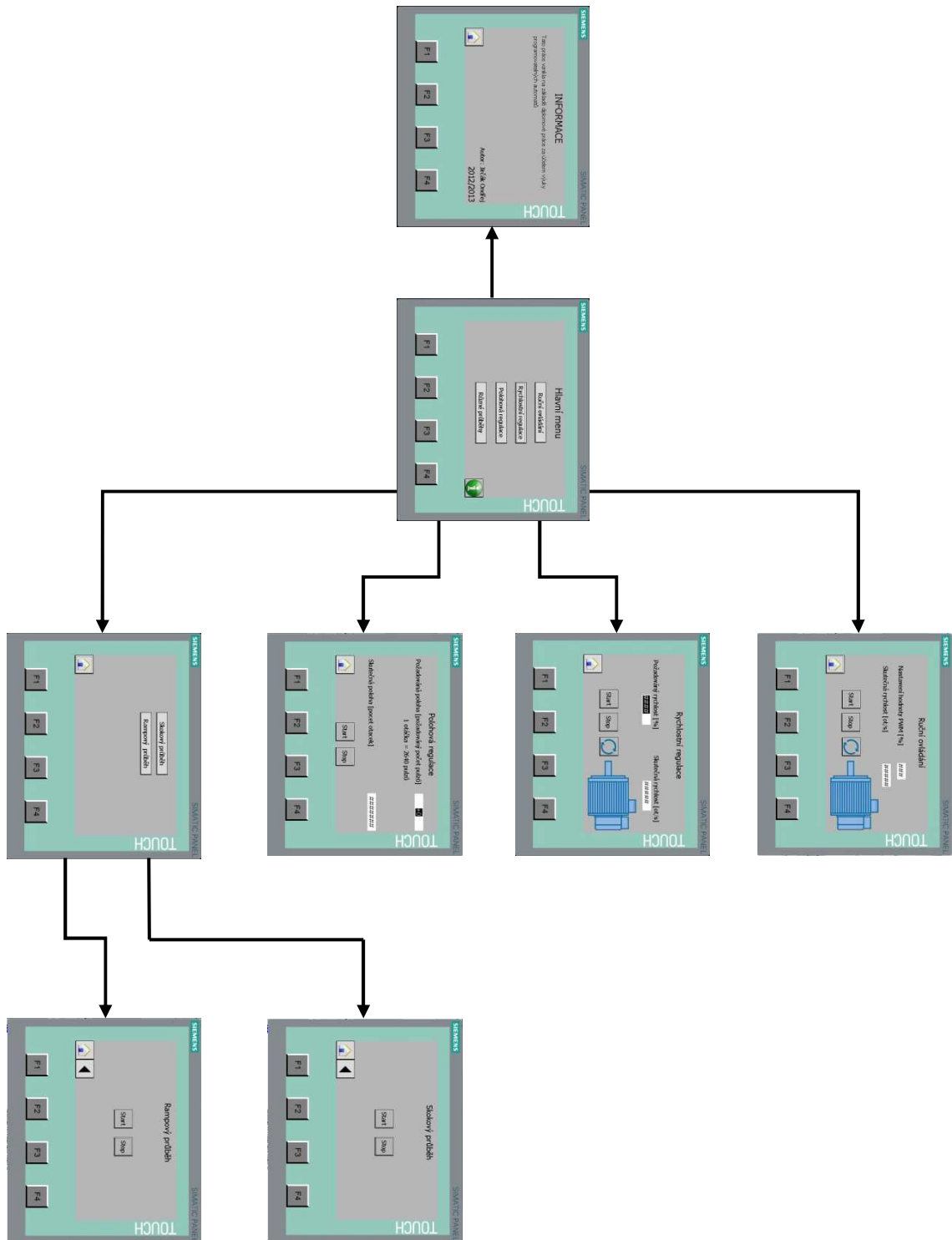
5.7. Tvorba HMI vizualizace

Tvorba vizualizace probíhala v programu TIA portál modul WinCC, který obsahuje prvky pro tvorbu vizualizace a navázání těchto prvků na proměnné v programovatelném automatu. V tomto programu lze nadefinovat i vlastní prvky a funkce pro vizualizaci, nebo přidávat takové prvky, které WinCC neobsahuje.



Obr.48 TIA portál – Tvorba vizualizace

Schéma vytvořených vizualizací:



6. Závěr

Úkolem této diplomové práce bylo seznámit se s principem řízení stejnosměrného motoru a navrhnout řídicí elektroniku pro připojení stejnosměrného motoru k programovatelnému automatu. Dále bylo úkolem seznámit se s programovacím prostředím pro programovatelný automat Simatic S7-1200 od německé firmy Siemens a navrhnout v tomto prostředí program pro řízení motoru pomocí polohové a rychlostní regulace. V neposlední řadě bylo cílem navrhnout vizualizaci na HMI, kterým by bylo PLC ovládáno a zároveň by zobrazovalo údaje o motoru.

První část práce se zabývala hardwarovým vybavením, kde byly zjištěny parametry jednotlivých komponentů a princip jejich řízení.

Druhá část práce se zabývala vývojem a výrobou řídicí elektroniky. Srdcem této elektroniky byl h-můstek (LMD 18200T), který zpracovával PWM signál vysílaný programovatelným automatem a následně po zpracování signálu řídil stejnosměrný motor. Pro potřeby regulace bylo zapotřebí vyřešit napájení IRC snímače, který snímal natočení výstupního hřídele motoru. IRC snímač byl napájen napětím +5V, které bylo dodáváno spínaným stabilizátorem napětí LM2576T-5.

Poslední část se zabývala tvorbou programů pro řízení stejnosměrného motoru. Pro řízení bylo navrženo pět režimů: ruční ovládání bez regulace, rychlostní regulace, polohová regulace, skokový průběh a rampový průběh, přičemž poslední dva zmiňované režimy vznikly v rámci demonstrace rychlostní regulace. Pro výpočet parametrů rychlostního regulátoru bylo nutné vytvořit přechodovou charakteristiku z ručního řízení. Z přechodové charakteristiky byl získán přenos a následně dosazen do Simulinku v Matlabu, kde byl vytvořen model regulované soustavy. V této soustavě byl PID regulátor přepnut na diskrétní řízení, pomocí kterého byly získány parametry pro nastavení regulátoru. Zjištěné parametry regulátoru pro rychlostní regulaci jsou následující: **$P = 0,57515$, $I = 0,0597$ a $D = 0$** . Ze zjištěných hodnot vyplývá, že byl použit PI regulátor. Polohová regulace byla tvořena dvěma smyčkami: rychlostní a polohovou. Nastavení hodnot regulátoru pro rychlostní smyčku bylo stejné jako pro rychlostní regulaci. Za účelem nalezení hodnot regulátoru pro polohovou smyčku bylo využito rychlostního regulátoru, pomocí kterého byla naměřena přechodová charakteristika. Z této charakteristiky byl opět získán přenos a dosazen do modelu. Následně byly získány hodnoty pro regulátor polohové smyčky. Nalezené hodnoty jsou následující: **$P = 0,1614$, $I = 0,09$ a $D = 0$** .

Zadané cíle práce se tedy podařilo splnit; oba regulátory byly nastaveny tak, že regulace byla funkční.

Citovaná literatura

- [1] **Seeker, SV.** dc_motor.jpg.
[Online] 28. 4. 2013.
Dostupné z: <http://www.submarineboat.com/images/rov/dc_motor.jpg>.
- [2] **Elektrika.cz.** Princip stejnosměrných motorů.
[Online] 23. 4. 2013.
Dostupné z: <<http://elektrika.cz/data/clanky/princip-stejnosmernych-motoru>>.
- [3] **fei1.vsb.cz.** kap22.pdf.
[Online] 28. 4. 2013.
Dostupné z: <<http://fei1.vsb.cz/kat430/old/Studium/Materialy/VS/kap22.pdf>>.
- [4] **Recycledgoods.** Hitachi DEC 1214659-00 DC Permanent Magnet Motor.
[Online] 28. 4. 2013.
Dostupné z: <http://www.recycledgoods.com/product/34981_hitachi_dec-1214659-00_dc-permanent-magnet-motor>.
- [5] **DH.** Pulzně šířková modulace.
[Online] 4. 5. 2013.
Dostupné z: <<http://www.dhservis.cz/psm.htm>>.
- [6] **Rhaaa.** DIY – Régulateur PWM.
[Online] 4. 5. 2013.
Dostupné z: <<http://www.rhaaa.fr/diy-regulateur-pwm>>.
- [7] **Snímače.ppt.**
[Online] 2. 5. 2013.
Dostupné z: <<http://www.google.cz/url?sa=t&rct=j&q=&esrc=s&source=web&cd=7&ved=0CE8QFjAG&url=http%3A%2F%2Flearning.sps-dopravni.cz%2Fmod%2Fresource%2Fview.php%3Fid%3D2283&ei=Q9SCUdnpGoaztAaJkYAw&usg=AFQjCNGYOzIMtnPqA8ApSQTpQO4iWocpNg&bvm=bv.45960087,d.>>>.
- [8] **jonatan.spse.pilsedu.cz.** Princip optických enkodérů polohy pro řízení motorů.
[Online] 2. 5. 2013.
Dostupné z: <http://jonatan.spse.pilsedu.cz/~mazanec/opticke_ekodery.htm>.
- [9] **Senzory.** Optický inkrementální senzor.
[Online] 5. 4. 2013.
Dostupné z: <<http://senzory.aspone.cz/ois.htm>>.
- [10] **Automatizace.cz.** Simatic S7-1200.
[Online] 21. 3. 2013.
Dostupné z: <<http://www.automatizace.cz/article.php?a=480>>.

- [11] **Siemens.** Řídicí systém Simatic S7-1200.
[Online] 21.. 3 2013.
Dostupné z:
<<http://www1.siemens.cz/ad/current/index.php?ctxnh=5dc8474325&ctxp=home>>.
- [12] **Siemens.** SIMATIC S7-1200 Easy Book.
[Online] 26. 3. 2013.
Dostupné z:
<http://www1.siemens.cz/ad/current/content/data_files/automatizacni_systemy/mikrosystemy/simatic_s71200/manualy/manual_s71200-easy-book_2009-11_en.pdf>.
- [13] **Siemens.** SIMATIC HMI KTP400 Basic color.
[Online] 20. 4. 2013.
Dostupné z:
<<http://www.automation.siemens.com/mcms/human-machine-interface/en/operator-interfaces/basic-panel/simatic-hmi-ktp400-basic-color/Pages/default.aspx>>.
- [14] **GMelectronic.** LMD18200T.
[Online] 4. 5. 2013.
Dostupné z:
<<http://www.gme.cz/obvody-pro-rizeni-motoru-mustkove-budice/lmd18200t-p399-186/#popis>>.
- [15] **Nekula, Bc. Pavel.** Diplomová práce.
[Online] 2009.
Dostupné z:
<https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=16397>.
- [16] **Wikipedia.** Dělič napětí.
[Online] 5. 5. 2013.
Dostupné z:
<http://cs.wikipedia.org/wiki/D%C4%9Bli%C4%8D_nap%C4%9Bt%C3%AD>.
- [17] **GMelectronic.** R0805 4K7 1%.
[Online] 22. 5. 2013.
Dostupné z: <<http://www.gme.cz/smd-rezistory-0805/r0805-4k7-1-p901-188/>>.
- [18] **Pandatron.cz.** Výpočet chladiče.
[Online] 11. 5. 2013.
Dostupné z: <http://pandatron.cz/?219&vypocet_chladice>.
- [19] **GMelectronic.** LM2576T-5.
[Online] 5. 5. 2013.
Dostupné z: <<http://www.gme.cz/spinane-regulatory-napeti-dc-dc-menice-pevne/lm2576t-5-p330-113/#popis>>.

- [20] **GMelectronic**. DPU1000A1.
[Online] 5. 5. 2013.
Dostupné z: <<http://www.gme.cz/tlumivky-vyvodove/dpu1000a1-p611-143/>>.
- [21] **GMelectronic**. Konektor XINYA MLW10G.
[Online] 11. 5. 2013.
Dostupné z: <<http://www.gme.cz/konektory-pro-ploche-kabely-do-dps/konektor-xinya-mlw10g-p800-035/>>.
- [22] **GMelectronic**. Násuvná svorkovnice PTR AKZ950/2-5.08-GREEN.
[Online] 11. 5. 2013.
Dostupné z: <<http://www.gme.cz/nasuvne-svorkovnice-na-kabel/nasuvna-svorkovnice-ptr-akz950-2-5-08-green-p821-050/>>.
- [23] **GMelectronic**. Násuvná vidlice PTR STLZ950/2G-5.08-H-GREEN.
[Online] 11. 5. 2013.
Dostupné z: <<http://www.gme.cz/nasuvne-vidlice-zasuvky-do-dps/nasuvna-vidlice-ptr-stlz950-2g-5-08-h-green-p821-081/>>.
- [24] CTP tisk. *Osvity na film* .
[Online] 4. 2. 2013.
Dostupné z: <<http://www.ctptisk.cz/osvit.htm>>.
- [25] **GES**. XLR.
[Online] 12. 5. 2013.
Dostupné z: <<http://www.ges.cz/cz/hledat/?search=xlr>>.
- [26] **Ivan Švarc, Miloš Šeda, Miluše Vítečková**. *AUTOMATICKÉ ŘÍZENÍ*. Brno : Vysoké učení technické v Brně, 2007.

Seznam příloh:

CD obsahuje: Tento dokument v PDF
 Podklady pro výrobu osvitové folie
 Návrh plošných spojů v Eaglu
 Zdrojový kód pro PLC a HMI