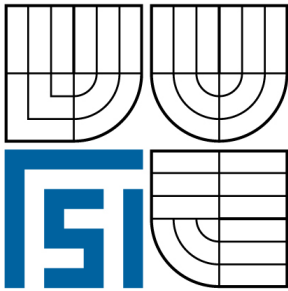


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

## UKÁZKA ŘÍZENÍ OTÁČEK MOTORU FREKVENČNÍM MĚNIČEM

MOTOR SPEED CONTROL EXHIBIT WITH FREQUENCY CHANGER

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JIŘÍ MÖHWALD

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. ZDENĚK NĚMEC, CSc.

BRNO 2009

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2008/2009

## **ZADÁNÍ DIPLOMOVÉ PRÁCE**

student(ka): Bc. Jiří Möhwald

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:

### **Ukázka řízení otáček motoru frekvenčním měničem**

v anglickém jazyce:

### **Motor speed control exhibit with frequency changer**

Stručná charakteristika problematiky úkolu:

Jde o laboratorní ukázkou řízení otáček asynchronního třífázového motoru pomocí frekvenčního měniče. Časové změny otáček jsou ovládány programovatelným automatem.

Cíle diplomové práce:

1. Osvojit si práci s daným třífázovým asynchronním motorem o výkonu 180W a s frekvenčním měničem Micromaster 440. Též se seznámit s daným programovatelným automatem Simatic S7-224 a s jeho programováním.
2. Vyřešit spojení mezi frekvenčním měničem a programovatelným automatem. Zvážit možnost obsluhovat automat operátorským panelem z vybavení pracoviště ÚAI.
3. Navrhnout a realizovat řízení proměnlivých otáček motoru. Umožnit obsluhu volit různé varianty časových průběhů otáček (změny sinusové, skokové, pilové, atd.).
4. Navrhnout a eventuálně odzkoušet pomocné zařízení k názorné demonstraci účinku proměnných otáček.

Seznam odborné literatury:

[1] Švarc, I.: Automatizace-Automatické řízení. Brno: CERM, 2005.

[2] Firemní dokumentace od fy Siemens k frekvenčním měničům a k programovatelným automatům.

Vedoucí diplomové práce: doc. Ing. Zdeněk Němec, CSc.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2008/2009.

V Brně, dne

L.S.

---

doc. RNDr. Ing. Miloš Šeda, Ph.D.  
Ředitel ústavu

---

doc. RNDr. Miroslav Doupovec, CSc.  
Děkan fakulty

## **ABSTRAKT**

Diplomová práce se zabývá řízením otáček asynchronního motoru frekvenčním měničem MICROMASTER 440 pomocí programovatelného automatu S7-200.

Začátkem mé diplomové práce jsem se zabýval studováním firemní dokumentace fy. Siemens, která je výrobcem daného frekvenčního měniče a programovatelného automatu. V průběhu řešení jsem začal studovat i firemní dokumentaci fy. Mitsubishi Electric, která je výrobcem operátorského panelu.

V první části mé práce jsem se zabýval obecným popisem daných zařízení a v druhé části jsem již řešil danou problematiku.

Mým úkolem bylo navržení komunikace mezi měničem a automatem, kterou jsem zvolil pomocí USS protokolu po sériové lince RS-485. Způsob ovládání automatu jsem vyřešil operátorským terminálem řady GOT1000, který je k němu připojen.

Má práce se tedy hlavně zabývala tvorbou řídicího programu pro automat a vytvoření grafického ovládání na terminálu.

Na závěr jsem zhodnotil poznatky z řešení a navrhnul další postupy.

## **ABSTRACT**

The following diploma thesis deals with the management of asynchronous motor revolutions by a frequency converter MICROMASTER 440 via the programmable controller S7-200.

At the beginning, I researched the company documentation of Siemens, which is the producer of the frequency converter and the programmable controller mentioned above. As my work proceeded, I also started to study the company documentation of Mitsubishi Electric, which produces the operating pannel.

The first part of my thesis provides a general description of the stated devices and the second part deals with the solution of the task itself.

My task was to design a way of communication between the converter and the controller. I chose a USS protocol via serial port RS-485. The operation of the controller was performed by using an operating terminal type GOT1000 which was connected to the controller.

In summary, my diploma thesis primarily deals with the construction of an operating programme for the controller and with creating of a graphical control on the terminal.

In the end, I evaluated the knowledge gained during the work on this subject and suggested further procedures.

## **KLÍČOVÁ SLOVA**

Frekvenční měnič, programovatelný automat, operátorský terminál, USS4 protokol, S7-200, GOT1000, STEP 7-Micro/WIN, GT Designer 2.

## **KEYWORDS**

Frequency converter, programmable controller, operating terminal, USS4 protocol, S7-200, GOT 1000, STEP 7-Micro/WIN, GT Designer 2.

## **PODĚKOVÁNÍ**

Tímto bych chtěl poděkovat vedoucímu své diplomové práce docentu Ing. Zdeňku Němcovi CSc., za jeho čas který mi věnoval a za veškeré podněty a připomínky, které mi pomohli při vypracování mé práce.

**Obsah:**

<b>1.</b>	<b>Úvod .....</b>	<b>3</b>
<b>2.</b>	<b>Postup práce .....</b>	<b>5</b>
<b>3.</b>	<b>Frekvenční měnič MICROMASTER 440 .....</b>	<b>7</b>
3.1	Charakteristické vlastnosti .....	7
3.2	Elektrická instalace .....	9
3.2.1	Blokové schéma měniče .....	9
3.2.2	Silová svorkovnice .....	10
3.2.3	Zapojení vynutí motoru.....	10
3.2.4	Řídící svorkovnice .....	11
3.2.5	Volba provozu Evropa/USA .....	12
3.2.6	Konfigurace analogových vstupů .....	12
3.3	Ovládací panely měniče .....	13
3.3.1	Uvedení měniče do provozu s ovládacím panelem BOP .....	13
3.3.2	Změna hodnot parametrů pomocí ovládacího panelu BOP .....	15
3.3.3	Místní a dálkové ovládání měniče .....	15
<b>4.</b>	<b>Programovatelný automat S7 – 200 .....</b>	<b>17</b>
4.1	Rozdělení S7 – 200 podle CPU .....	17
4.2	Rozšiřovací moduly .....	18
4.3	Připojení S7 – 200 .....	18
4.3.1	připojení napájení k S7 – 200 .....	18
4.3.2	Možnosti komunikace .....	19
4.4	Jak S7 – 200 vykonává řídicí program .....	20
4.4.1	S7 – 200 spojuje úložný program s fyzickými vstupy a výstupy .....	20
4.4.2	S7 – 200 provádí své úkony v programovém cyklu .....	20
4.5	Přístup k datům .....	21
4.6	Použití STEP 7 – Micro/WIN pro vytvoření programu .....	25
4.6.1	Vlastnosti editoru STL .....	26
4.6.2	Vlastnosti editoru LAD .....	26
4.6.3	Vlastnosti editoru FBD .....	26
<b>5.</b>	<b>Operátorský terminál GOT 1000 .....</b>	<b>27</b>
5.1	Výkonná technologie .....	27
5.2	Typy operátorských terminálů GOT 1000 .....	28
5.3	Grafický software GT – Designer 2 .....	30
<b>6.</b>	<b>Asynchronní stroje .....</b>	<b>31</b>
6.1	Princip funkce asynchronních strojů.....	31
6.2	Kmitočtová regulace rychlosti asynchronních strojů.....	31
<b>7.</b>	<b>Spojení mezi FM a S7-200 .....</b>	<b>33</b>
7.1	Operátorský panel BOP .....	33
7.2	Možnosti řízení MICROMASTER a Simatic S7 – 200 .....	33
7.3	Zvolené řízení .....	34
<b>8.</b>	<b>USS4 protokol.....</b>	<b>35</b>
8.1	Požadavky použití USS4 protokolu .....	35
8.2	Použití instrukcí USS4 .....	36
8.3	Instrukce USS4 _INIT .....	36
8.4	Instrukce USS4 _DVR_CTRL .....	38
8.5	Instrukce USS4 _RPM_x .....	42
8.6	Instrukce USS4 _WPM_x .....	44

<b>9.</b>	<b>Návrh uživatelského ovládání .....</b>	<b>47</b>
9.1	Operátorský terminál GT1155 – QSBD .....	47
9.2	Výměna programovatelného automatu .....	48
<b>10.</b>	<b>Zapojení a nastavení .....</b>	<b>49</b>
10.1	Blokové schéma zařízení .....	49
10.2	Napájení .....	49
10.2.1	Zapojení napájení měniče a motoru .....	49
10.2.2	Napájení automatu a operátorského terminálu .....	50
10.3	Zapojení komunikace mezi měničem a automatem pro USS4 protokol... ..	50
10.4	Nastavení USS4 protokolu na měniči .....	51
10.5	Zapojení a nastavení komunikace mezi automatem a terminálem .....	52
10.5.1	Instalace OS knihovny pro S7 – 200 .....	52
10.5.2	Nastavení komunikace mezi automatem a operátorským terminálem .....	54
10.6	Zapojení a nastavení komunikace mezi automatem a PC .....	57
10.6.1	PPI Multi – Master kabel .....	58
10.6.2	Nastavení přenosné rychlosti a síťové adresy pro S7 – 200 .....	59
10.6.3	Vyhledávání S7 – 200 v síti .....	59
<b>11.</b>	<b>Software STEP 7–Micro/WIN V4.0 .....</b>	<b>61</b>
11.1	Program ve STEP 7-Micro/WIN V4.0 .....	61
11.2	Download programu ze STEP 7 do automatu .....	63
<b>12.</b>	<b>Software GT Designer 2 .....</b>	<b>65</b>
12.1	Multi Action Switch .....	65
12.2	Bit Lamp .....	66
12.3	Numerical Display a Numerical Input .....	66
12.4	Download programu z GT Designer 2 do operátorského terminálu .....	67
<b>13.</b>	<b>Postup pro laboratorní cvičení .....</b>	<b>69</b>
13.1	Úvod .....	69
13.2	Cíl .....	69
13.3	Rozbor úlohy .....	69
13.4	Schéma zapojení .....	70
13.5	Pokyny pro cvičení .....	70
13.5.1	Uvedení do provozu .....	70
13.5.2	Menu ZADEJ OTÁČKY .....	71
13.5.3	Menu ZADEJ PRŮBĚH SKOKOVĚ .....	73
13.5.4	Menu ZADEJ PRŮBĚH S RAMPOU .....	75
13.6	Vypnutí zařízení .....	76
13.7	Závěr .....	76
<b>14.</b>	<b>Návrhy na zlepšení.....</b>	<b>77</b>
14.1	Konstrukční návrh pro demonstraci účinku proměnných otáček.....	77
14.2	Návrhy na zlepšení.....	77
14.3	Ukázka mého laboratorního stolu.....	78
<b>15.</b>	<b>Závěr.....</b>	<b>79</b>
	<b>Seznam použité literatury.....</b>	<b>81</b>
	<b>Seznam příloh.....</b>	<b>83</b>

## 1. ÚVOD

Frekvenční měniče s asynchronním motorem spolu tvoří regulační pohon, který si v poslední době získal dominantní postavení v automatizovaném průmyslu.

K dosažení širokého regulačního rozsahu musíme napájet asynchronní stroj, ze zdroje proměnného kmitočtu, umožňujícího současnou změnu výstupního napětí. K tomuto účelu slouží frekvenční měniče.

V mé diplomové práci jsem frekvenční měnič řídil programovatelným automatem po sériové lince RS-485, pomocí USS protokolu (speciální komunikační knihovna pro řízení pohonů). Do řídicího obvodu jsem zapojil i dotykový operátorský terminál, kterým se celé zařízení ovládá.

Navržené zařízení tak tvoří jednotný celek, který je schopný ovládat frekvenční měnič.





## 2. POSTUP PRÁCE

Začátkem mé diplomové práce jsem se zabýval studováním firemní dokumentace fy. Siemens, která je výrobcem daného frekvenčního měniče a programovatelného automatu a firemní dokumentace fy. Mitsubishi Electric, která je výrobcem operátorského panelu.

V první části mé práce jsem se zabýval obecným popisem daných zařízení a v druhé části jsem již řešil danou problematiku.

Pro moji diplomovou práci mi poskytlo pracoviště ÚAI frekvenční měnič MICROMASTER 440. Jedná se o frekvenční měnič s jednofázovým napájecím napětím 230V o výkonu pro motory do 0,25 kW. Tudíž je použitelný pro řízení daného asynchronního motoru o výkonu 0,18kW.

Jako programovatelný automat je použit typ Simatic S7-200 CPU 224.

Mým úkolem bylo vyřešit spojení mezi frekvenčním měničem a programovatelným automatem, pro řízení motoru. A zvážit možnost obsluhovat automat operátorským panel BOP (Basic Operator Panel) z pracoviště ÚAI, kterým se daný měnič programuje.

Jelikož panel BOP nemá sériové rozhraní RS-485, [má ho pouze panel AOP (Advanced Operator Panel)], kterým bych ho připojil na port automatu nebo jinou komunikační možnost s automatem, od této volby jsem tedy musel upustit. A zabýval se možností přímo, řídit měnič automatem. Z třech možných řešení, který automat nabízí jsem zvolil řízení měniče pomocí USS protokolu.

Dále jsem přemýšlel jak nejlépe z uživatelského hlediska ovládat řízení motoru a jako nejvýhodnější řešení se ukázalo použít operátorský terminál fy. Mitsubishi Electric z řady GOT1000, typ GT1155-QSBD, který mi opět poskytlo pracoviště ÚAI.

Jelikož jsem se rozhodl pro použití USS protokolu a operátorského terminálu, vyžádalo si toto řešení nutnou výměnu programovatelného automatu za typ S7-200 CPU 224xp, který má 2 komunikační porty RS-485.

Následně jsem vytvořil schéma zapojení, podle kterého jsem vše zapojil.

Nakonfiguroval jsem všechny tři zařízení ( operátorský terminál, programovatelný automat a frekvenční měnič ), tak aby spolu komunikovaly a tvořily ucelené jednotné zařízení.

Vytvářel jsem ve dvou vývojových prostředích (STEP 7-Micro/WIN [automat] a GT-Designer 2 [operátorský terminál]) uživatelské programy a testoval chod celého zařízení. Finální programy jsem nahrál do automatu a op. terminálu, kdy může uživatel pohodlně řídit chod motoru přes dotykový panel.

V neposlední řadě jsem navrhl schéma pro pomocné zařízení k názorné demonstraci účinku proměnných otáček, jelikož změna otáček je doposud sledována pouze na samotné hřídeli motoru. A také laboratorní úlohu, ve které si mohou studenti řízení motoru vyzkoušet.

Na závěr jsem vše zhodnotil a navrhnul možnosti pro vylepšení mé diplomové práce.



### 3. FREKVENČNÍ MĚNIČ MICROMASTER 440

MICROMASTER 440 jsou měniče kmitočtu s napětovým meziobvodem určené pro napájení třífázových asynchronních a synchronních elektromotorů ve výkonovém rozsahu od 120 W do 75 kW. Podle typu měniče je možné jednofázové nebo třífázové napájení.

Obvody řízení a regulace jsou realizovány pomocí digitální techniky s mikroprocesorovým řízením a výkonovými tranzistory typu IGBT. Top činí měniče spolehlivými zařízeními s možností přizpůsobení vlastností velkému množství aplikací. Metodou pulzně šířkové modulace s přepínatelným spínacím kmitočtem je dosaženo tichého a rovnoměrného chodu motoru. Ochranné funkce měniče a motoru zajišťují dokonalou ochranu pohonu.

Tovární nastavení měniče je vhodné pro široký rozsah jednoduchých aplikací. Změnou parametrů je možné měniče MICROMASTER 440 přizpůsobit náročným aplikacím. Měniče je možné použít jako samostatná zařízení nebo je začlenit do automatizačních celků.

#### 3.1 Charakteristické vlastnosti

##### Základní vlastnosti

- Velice snadné připojení, nastavení a uvedení do provozu.
- Měniče jsou navrženy v souladu s požadavky EMC.
- Možnost provozu na izolované síti.
- Rychlá odezva na řídicí signály.
- Obvody mikroprocesorového řízení a regulace zabezpečují vysokou spolehlivost a flexibilitu zařízení.
- Množství parametrů umožňuje dokonalé přizpůsobení pohonu s měničem kmitočtu dané aplikaci.
- Vysoký spínací kmitočet pulzně šířkové modulace zajišťuje tichý chod motoru.
- Možnost výběru způsobu ovládní přes řídicí svorkovnici se základním ovládacím panelem (BOP), rozšířeným ovládacím panelem (AOP), sériovou linkou z PC nebo komunikační sběrnici PROFIBUS /DEVICE NET.
- Požadovanou hodnotu výstupního kmitočtu (a tedy i hodnotu otáček motoru) lze zadávat těmito způsoby:
  1. přímým číselným zadáním hodnoty kmitočtu
  2. analogovým napětovým 0...10V, -10...+ 10V nebo proudovým signálem 0...20mA, 4...20mA,
  3. externím potenciometrem,
  4. motorpotenciometrem,
  5. až 15 pevně přednastavenými hodnotami kmitočtu,
  6. prostřednictvím sériového rozhraní (USS protokol, PROFIBUS),
  7. dle vnějšího snímače pomocí vestavěného technologického regulátoru.
- Dva analogové vstupy, dva analogové výstupy, 6 až 8 digitálních vstupů, 3 reléové výstupy.
- Přednastavené hodnoty parametrů odpovídají požadavkům evropských a severoamerických norem.

**Rozšířené vlastnosti**

- Vektorové řízení bez zpětné otáčkové vazby (SLVC).
- Vektorové řízení (VC) se zpětnou otáčkovou vazbou s inkrementálním snímačem rychlosti.
- Řízení s aktivní regulací magnetizačního proudu (FCC) pro zvýšení účinnosti pohonu.
- Rychlá ochrana proti nadměrnému vzrůstu proudu (FCL) umožňuje reakci měniče dříve tak, aby nedošlo k vyhodnocení poruchového stavu a zastavení pohonu.
- Možnost brždění motoru stejnosměrným proudem.
- Kompaundní způsob brždění motoru umožňuje řízené zastavení pohonu s velkým momentem setrvačnosti.
- Dynamické brždění pomocí vestavěné brzdě jednotky.
- Nastavitelná doba rozběhu a doběhu s počátečním koncovým zaoblením rozběhové křivky pro měkký rozběh a zastavení pohonu.
- Vestavěný technologický PID regulátor umožňuje řízení procesu bez nutnosti nadřazeného řídicího systému.
- Možnost přepínání mezi 3 různými sadami nastavení měniče a 3 různými sadami nastavení pohonu umožňuje pružné použití jednoho měniče pro různé aplikace.
- Speciální funkce pro:
  - řízení polohování při zastavení pohonu
  - režim kontroly zatěžovacího momentu (hlídání správného mechanického chodu pohonu)
  - propojení pomocí technologie BICO

**Ochranné funkce**

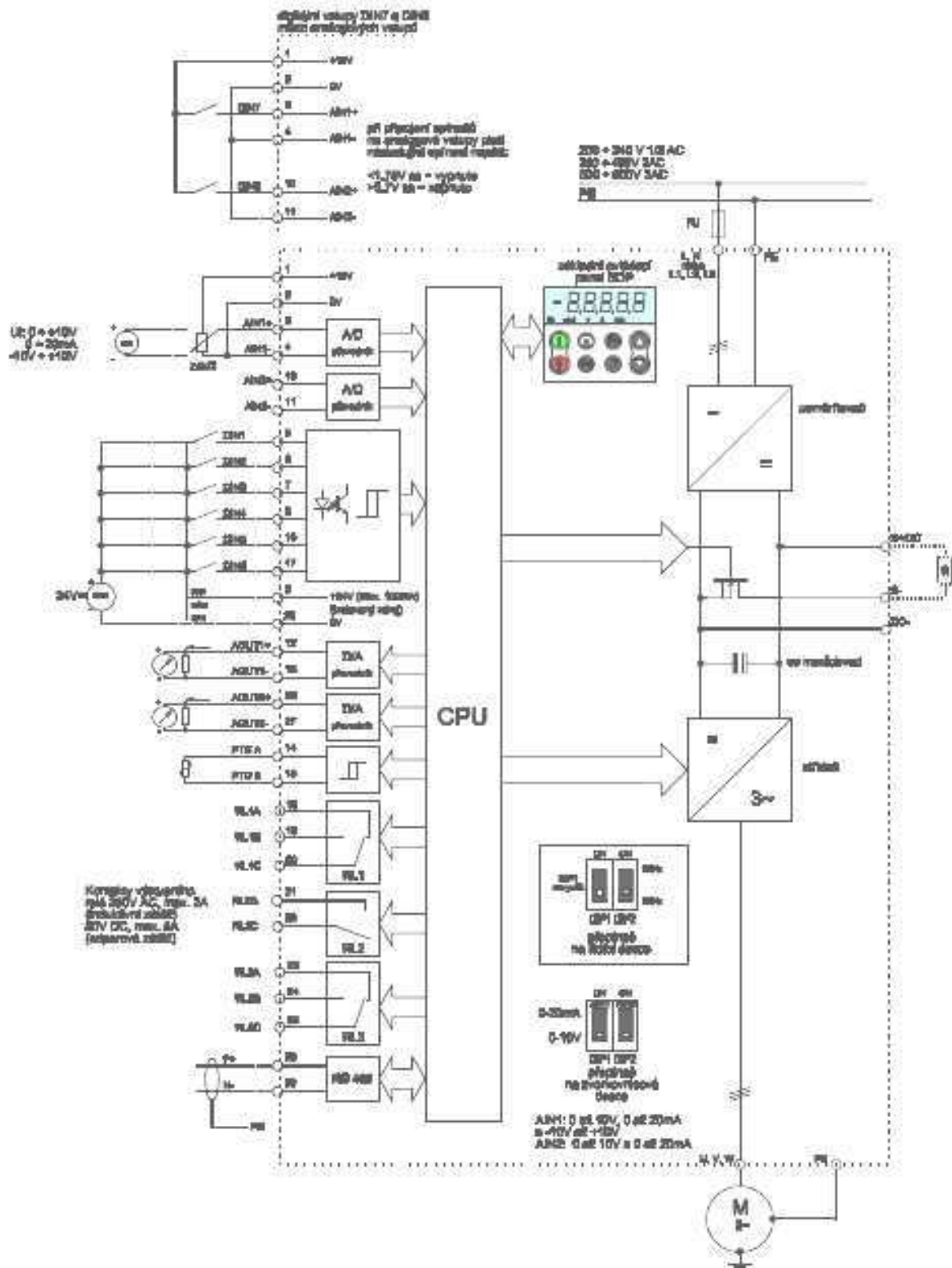
- Kompletní ochrana měniče i motoru před přetížením.
- Ochrana proti přepětí a podpětí.
- Ochrana proti zemnímu a mezifázovému spojení.
- Ochrana proti překročení teploty měniče.
- Tepelná ochrana  $I^2t$  motoru.
- Ochrana motoru pomocí PTC/KTY čidla ve vinutí motoru.



Obr. 1. Frekvenční měnič MICROMASTER 440

### 3.2 Elektrická instalace

#### 3.2.1 Blokové schéma měniče

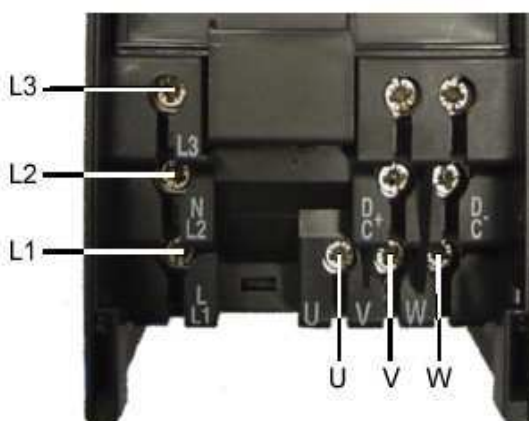


Obr. 2 Blokové schéma měniče

### 3.2.2 Silová svorkovnice

Síťové napětí připojte třížilovým kabelem na silové svorky L/L1, N/L2 a na zemní svorku PE u jednofázového měniče nebo čtyřžilovým kabelem na svorky L/L1, N/L2, L3 a na zemní svorku PE u třífázového měniče. Pro připojení motoru použijte čtyřžilový kabel. Kabel se připojí na silové svorky U, V, W a na zemní svorku PE tak, jak je uvedeno v následující tabulce 1.

Zapojení silové svorkovnice	
Silová svorkovnice	Funkce
PE	uzemnění síťového přívodu
L/L1	síťový přívod
N/L2	síťový přívod
L3	síťový přívod (svorka se zapojuje jen u měničů s třífázovým napájením 3x 400 V, popř. 3x 230 V, 3x 500 V)
PE	uzemnění motorového přívodu
U	přívod k motoru
V	přívod k motoru
W	přívod k motoru
DC+ / B+	připojení brzdného odporníku / kladný pól ss meziobvodu
B-	připojení brzdného odporníku
DC-	záporný pól ss meziobvodu

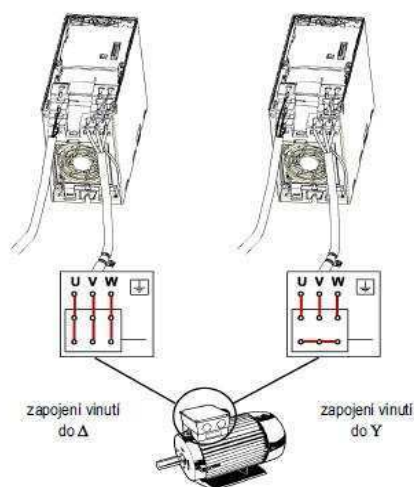


Tab. 1 a Obr. 3 Zapojení silové svorkovnice

### 3.2.3 Zapojení vynutí motoru

Správná hodnota napájecího napětí motoru musí odpovídat zapojení vynutí motoru.

- Pokud se jedná o měniče s jednofázovým napájením 1 x 230V, zapojte vynutí motoru na napětí 3 x 230 V (obvyklé zapojení malých motorů do trojúhelníku  $\Delta$ ).
- Pokud se jedná o měnič s třífázovým napájením 3 x 400V, zapojte vynutí motoru na napětí 3 x 400V (obvyklé zapojení malých motorů do hvězdy Y a motorů nad 3 kW do trojúhelníku  $\Delta$ ).



Obr. 4. Zapojení vynutí motoru do trojúhelníku nebo do hvězdy

## 3.2.4 Řídící svorkovnice

Zapojení řídicí svorkovnice				
Svorky na řídicí svorkovnici	Označení	Hodnota	Funkce	Poznámka
1	10V+	+10 V	referenční napětí	≤ 10 mA
2	0 V	0 V	referenční napětí	vztažný potenciál
3	AIN1 +	0 + 10 V / - 10 V + +10 V <sup>1)</sup>	analogový vstup 1	kladný potenciál
4	AIN1 -	0 + 20 mA / 4 + 20 mA <sup>2)</sup>	viz P0756[1] až P0761[1]	záporný potenciál
5	DIN 1	programovatelné vstupy viz P0701+P0704	digitální vstup 1	24 V <sup>3)</sup>
6	DIN 2		digitální vstup 2	
7	DIN 3		digitální vstup 3	
8	DIN 4		digitální vstup 4	
9	izol. +24 V	+24 V	pomocné napájecí napětí (izolovaný zdroj)	≤ 50 mA
10	AIN2 +	0 + 10 V <sup>1)</sup>	analogový vstup 1	kladný potenciál
11	AIN2 -	0 + 20 mA / 4 + 20 mA <sup>2)</sup>	viz P0756[2] až P0761[2]	záporný potenciál
12	AOUT1 +	0 + 20 mA / 4 + 20 mA <sup>4)</sup>	analogový výstup	kladný potenciál
13	AOUT1 -		viz P0771[1] až P0781[1]	záporný potenciál
14	PTC A		vstup tepelné ochrany motoru PTC / KTY	viz: P0601
15	PTC B			
16	DIN 5	programovatelné vstupy viz P0705+P0706	digitální vstup 5	24 V <sup>3)</sup>
17	DIN 6	digitální vstup 6		
18	RL1 A	230 V~ / 2 A <sup>4)</sup> 30V= / 5 A <sup>5)</sup>	programovatelné relé 1 viz P0731	rozpinací kontakt
19	RL1 B			spínací kontakt
20	RL1 C			střední kontakt
21	RL2 B	230 V~ / 2 A <sup>4)</sup> 30V= / 5 A <sup>5)</sup>	programovatelné relé 2 viz P0732	spínací kontakt
22	RL2 C			střední kontakt
23	RL3 A			230 V~ / 2 A <sup>4)</sup> 30V= / 5 A <sup>5)</sup>
24	RL3 B	spínací kontakt		
25	RL3 C	střední kontakt		
26	AOUT2 +	0 + 20 mA / 4 + 20 mA <sup>4)</sup>	analogový výstup	kladný potenciál
27	AOUT2 -		viz P0771[2] až P0781[2]	záporný potenciál
28	izol. 0V	0 V	pomocné napájecí napětí (izolovaný zdroj)	společný potenciál ke sv. 9 a DIN 1 až DIN 6
29	P +		sériová linka RS485	
30	N -			

1) Vstupní impedanace 720k $\Omega$ 

ve funkci digitálního vstupu DIN7, DIN8 je log. 1 &gt;3,7V, log.0 &lt; 1,75V

2) Vstupní impedanace 120 $\Omega$ 

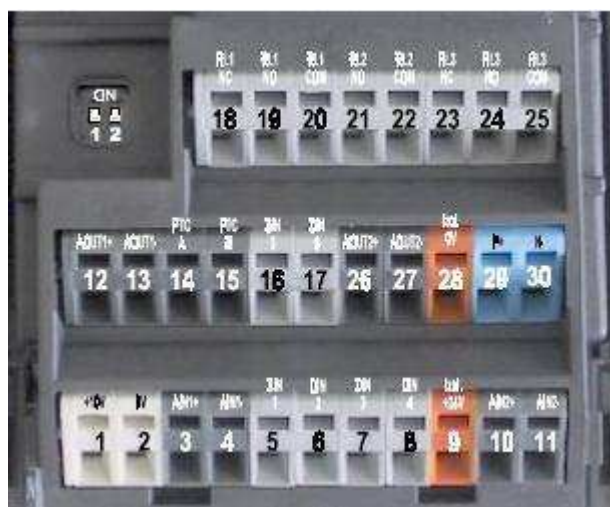
3) Logická úroveň 1 = +15 až +30V, vstupní proud max. 5mA

3) Odporná zábrž

4) Max. zatěžovací impedanace 500 $\Omega$ .

5) Odporná zábrž

Tab. 2 Zapojení řídicí svorkovnice



Obr. 5. Řídící svorkovnice

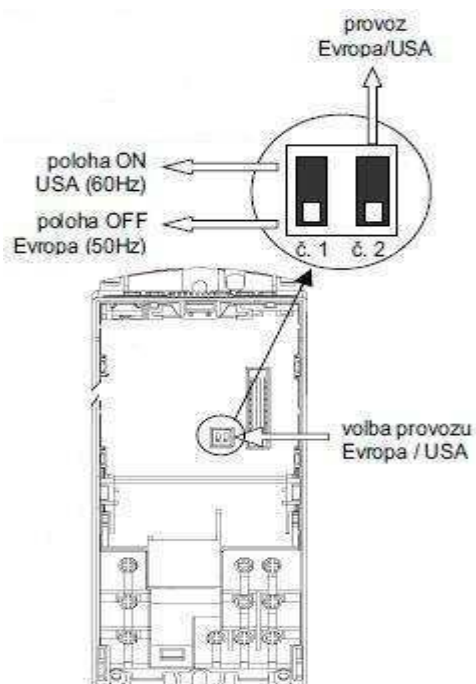


### 3.2.5 Volba provozu Evropa/USA

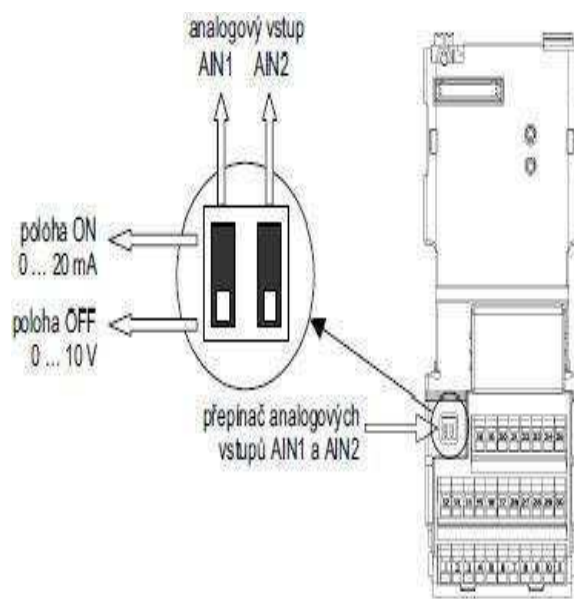
Přepínačem DIP, který je umístěn na řídicí desce (viz. obr.6), je možné volit provoz měniče na síti 50Hz nebo 60Hz.

OFF provoz měniče na síti 50Hz (Evropa)

ON provoz měniče na síti 60Hz (USA)



Obr. 6. Přepínač DIP na řídicí desce



Obr. 7. Přepínač DIP na svorkovnicové desce

### 3.2.6 Konfigurace analogových vstupů

Přepínačem DIP, který je umístěn na svorkovnicové desce (viz obr.7), je možné zvolit konfiguraci analogových vstupů AIN1 a AIN2.

OFF napět'ový vstup 0 ... 10V

ON proudový vstup 0 ... 20mA

### 3.3 Ovládací panely měniče

Měniče kmitočtu MICROMASTER 440 mohou být vybaveny třemi různými ovládacími panely – viz. obr. 8. Měniče jsou standardně dodávány s ovládacím panelem SDP (Status Display Panel). Jako doplněk je dodáván panel BOP (Basic Operator Panel ) nebo panel AOP (Advanced Operator Panel).

















Obr. 8. Ovládací panely

#### 3.3.1 Uvedení měniče do provozu s ovládacím panelem BOP

Ovládací panel BOP je doplněk měniče MICROMASTER 440. Panel BOP umožňuje uživateli přístup k parametrům měniče a jejich změnou přizpůsobení měniče různým aplikacím a způsobům ovládání. S ovládacím panelem BOP je možné změnit parametry měniče, aniž by poté musel být panel BOP při provozu na měniči umístěn. Tímto způsobem je možné snížit náklady.

Při továrním nastavení je ovládání měniče z panelu BOP zablokováno. Odblokování ovládacích tlačítek je možné následujícími parametry:












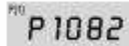







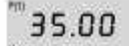














- start/stop chodu motoru tlačítka  a  - nastavte P0700 = 1
- reverzace chodu otáčení motoru tlačítkem  - nastavte P0700 = 1
- zvyšování a snižování žádané hodnoty otáček tlačítka  a  - nastavte P1000 = 1

Význam ovládacích tlačítek na panelu BOP		
Ovládací tlačítko	Popis tlačítka	Funkce tlačítka
	tlačítko „1“	Tlačítko „1“ slouží k zapnutí chodu motoru. Chcete-li pohon zapnout, stisknete tlačítko „1“. Tato funkce je standardně zablokována. Odblokování tlačítka „1“ je možné nastavením parametru P0700 = 1.
	tlačítko „0“	Tlačítko „0“ slouží k vypnutí chodu motoru. Chcete-li pohon vypnout takovým způsobem, aby motor dobíhal po doběhové rampě, stisknete tlačítko „0“ jedenkrát. Tato funkce je standardně zablokována. Její odblokování je možné nastavením parametru P0700 = 1. Pokud stisknete tlačítko „0“ dvakrát nebo tlačítko podržíte déle stisknuté, dojde k okamžitému vypnutí výstupních tranzistorů měniče a volnému doběhu motoru.
	tlačítko „reverzace“	Tlačítko pro změnu směru otáčení motoru. Je-li zvolen smysl otáčení doleva, objeví se na displeji znaménko minus, popř. začne blikat desetinná tečka. Funkce tlačítka je standardně zablokována. Odblokování tlačítka „reverzace“ je možné nastavením parametru P0700 = 1.
	tlačítko „krokování“	Bude-li tlačítko „krokování“ stisknuto při stojícím pohonu, začne se pohon rozbíhat v závislosti na nastavených hodnotách parametrů. Po uvolnění tlačítka se pohon zastaví. Stisk tlačítka při běžícím pohonu nemá žádný účinek.
	tlačítko „Fn“	<ul style="list-style-type: none"> <li>♦ Při chodu pohonu tlačítko „Fn“ slouží k zobrazení: <ul style="list-style-type: none"> <li>- hodnoty napětí stejnosměrného meziobvodu (indikováno zobrazením <b>d</b>)</li> <li>- hodnoty výstupního proudu (indikováno zobrazením <b>A</b>)</li> <li>- hodnoty výstupního kmitočtu (indikováno zobrazením <b>Hz</b>)</li> <li>- hodnoty výstupního napětí (indikováno zobrazením <b>o</b>)</li> <li>- hodnoty určené parametrem P0005</li> </ul>           Jednotlivé hodnoty jsou postupně zobrazovány při opakovaném stisku tlačítka.         </li> <li>♦ Při změně parametrů stisknutím tlačítka „Fn“ dojde k zobrazení parametru r0000, při opětovném stisknutí k zobrazení posledně měněného parametru.</li> <li>♦ Nulování poruchového stavu měniče při hlášení poruchy Fxxx.</li> </ul>
	tlačítko „P“	Tlačítko „P“ slouží k přepínání mezi číslem parametru a hodnotou parametru.
	tlačítko „Δ“	Tlačítko „Δ“ slouží k zvětšování čísel a hodnot parametrů nebo na změnu požadované hodnoty výstupního kmitočtu. Zvyšování požadované hodnoty kmitočtu je standardně zablokováno. Odblokování je možné nastavením parametru P1000 = 1.
	tlačítko „∇“	Tlačítko „∇“ slouží ke snižování čísel a hodnot parametrů nebo na změnu požadované hodnoty výstupního kmitočtu. Snižování požadované hodnoty kmitočtu je standardně zablokováno. Odblokování je možné nastavením parametru P1000 = 1.
		Displej na ovládacím panelu měniče slouží k indikaci čísel parametrů (r0000 ... P9999), indexů parametrů (in001), hodnot parametrů (např. 12.40) nebo kódů poruchových (F0011) a výstražných hlášení (A0501).

Tab. 3. Význam ovládacích tlačítek na panelu BOP

### 3.3.2 Změna hodnot parametrů pomocí ovládacího panelu BOP

Následující příklad popisuje postup při změně hodnoty parametru P1080 (maximální výstupní kmitočet z továrně nastavené hodnoty 50.00 Hz na hodnotu 35 Hz). Tento příklad slouží jako postup při změně parametrů měniče pomocí ovládacího panelu BOP.

Krok	Činnost	Výsledek činnosti zobrazený na displeji panelu BOP
1	Stisknete tlačítko  pro přístup k parametrům.	
2	Stisknete opakovaně tlačítko  dokud se nezobrazí parametr P0010.	
3	Stisknete tlačítko  Zobrazí se hodnota parametru P0010 (volba stavu měniče).	
4	Tlačítkem  změňte hodnotu parametru P0010 = 1 (nastavení měniče).	
5	Stisknutím tlačítka  dojde k uložení zvolené hodnoty parametru a zobrazení čísla parametru.	
6	Stisknete a podržete tlačítko  dokud se nezobrazí parametr P1082 (max. kmitočet).	
7	Stisknete tlačítko  Zobrazí se pořadí indexu parametru P1082. Tlačítkem  , popř.  , můžete změnit číslo indexu parametru (v tomto případě číslo sady dat měniče).	
8	Stisknete tlačítko  Zobrazí se hodnota parametru P1082.	
9	Stisknete a podržete tlačítko  dokud se nezmění hodnota parametru na 35.00 Hz.	
10	Stisknutím tlačítka  dojde k uložení zvolené hodnoty parametru a zobrazení čísla parametru P1082.	
11	Stisknete a podržete tlačítko  dokud se nezobrazí parametr P0010 (volba stavu měniče).	
12	Stisknete tlačítko  Zobrazí se hodnota parametru P0010.	
13	Tlačítkem  změňte hodnotu parametru P0010 = 0 (parametr P0010 je nutné před zadáním povelu ZAP nastavit vždy na hodnotu 0, jinak se pohon nerozběhne).	
14	Stisknutím tlačítka  dojde k uložení zvolené hodnoty parametru a zobrazení čísla parametru P0010.	
15	Stisknete opakovaně tlačítko  dokud se nezobrazí parametr r0000 (zobrazení stavu měniče).	
16	Stisknete tlačítko  pro zobrazení stavu měniče. Na displeji bude střídavě blikat nulová hodnota a požadovaná hodnota kmitočtu.	

Tab. 4. Postup při změně parametru

### 3.3.3 Místní a dálkové ovládání měniče

Měniče MICROMASTER 440 lze ovládat buď z místa přes ovládací panel BOP/AOP nebo řídicí svorkovnicí nebo dálkově prostřednictvím sériového komunikačního rozhraní a protokolu USS nebo přes sběrnici PROFIBUS s doplňkovým modulem.



## 4. PROGRAMOVATELNÝ AUTOMAT S7-200

Série S7-200 je řada malých -programovatelných automatů (mikro - PLC) určených k řízení v různých automatizačních aplikacích.

Zařízení S7-200 monitoruje vstupy a řídí výstupy pomocí uživatelského programu, který může obsahovat Booleovu logiku, počítání, časování, složité matematické operace a komunikaci s jinými inteligentními zařízeními. Kompaktní design, flexibilní konfigurace a výkonný instrukční soubor jsou důvody, proč je zařízení S7-200 optimálním řešením pro řízení široké škály aplikací. Široká nabídka různých modelů S7-200 a programovací nástroj na bázi Windows - poskytují flexibilitu potřebnou pro řešení vašich automatizačních potřeb.

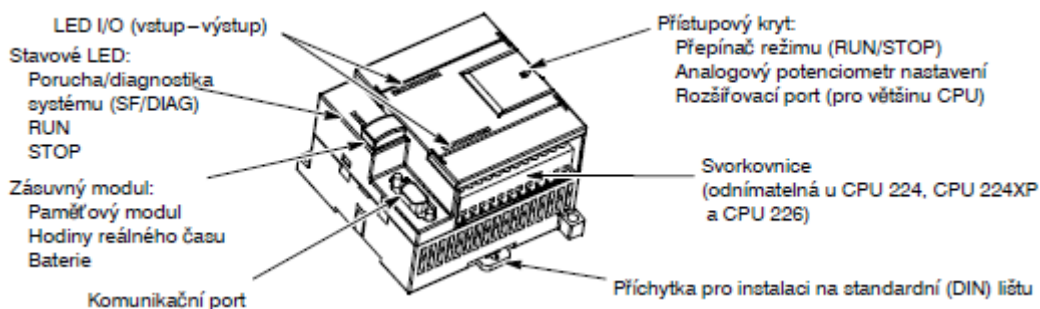
S7-200 obsahuje mikroprocesor, integrovaný napájecí zdroj, vstupní a výstupní obvody v kompaktním pouzdru, které tak tvoří výkonný programovatelný automat.

### 4.1 Rozdělení S7-200 podle CPU

Siemens dodává různé modely S7-200 s různými parametry a schopnostmi, které pomáhají vytvořit efektivní řešení pro vaše aplikace. Tabulka 5. stručně srovnává některé parametry CPU.

Parametr	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Rozměry (mm)	90 x 80 x 62	90 x 80 x 62	120,5 x 80 x 62	140 x 80 x 62	190 x 80 x 62
Paměť pro program: s editací v režimu RUN bez editace v režimu RUN	4096 bytů 4096 bytů	4096 bytů 4096 bytů	8192 bytů 12288 bytů	12288 bytů 16384 bytů	16384 bytů 24576 bytů
Paměť pro data	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Zálohování dat	standardně 50 hodin	standardně 50 hodin	standardně 100 hodin	standardně 100 hodin	standardně 100 hodin
Integrované I/O digitální analogové	6 vst./4 výst. –	8 vst./6 výst. –	14 vst./10 výst. –	14 vst./10 výst. 2 vst./1 výst.	24 vst./16 výst. –
Rozšiřovací moduly	0 modulů	2 moduly <sup>1</sup>	7 modulů <sup>1</sup>	7 modulů <sup>1</sup>	7 modulů <sup>1</sup>
Vysokorychlostní čítače Jednofázové Dvoufázové	4 při 30 kHz 2 při 20 kHz	4 při 30 kHz 2 při 20 kHz	6 při 30 kHz 4 při 20 kHz	4 při 30 kHz 2 při 200 kHz 3 při 20 kHz 1 při 100 kHz	6 při 30 kHz 4 při 20 kHz
Pulzní výstupy (DC)	2 při 20 kHz	2 při 20 kHz	2 při 20 kHz	2 při 100 kHz	2 při 20 kHz
Analogové potenciometry	1	1	2	2	2
Hodiny reálného času	Zásuvný modul	Zásuvný modul	Integrované	Integrované	Integrované
Komunikační porty	1 RS-485	1 RS-485	1 RS-485	2 RS-485	2 RS-485
Matematika s pohyblivou řádovou čárkou	Ano				
Velikost registru obrazu digitálních I/O	256 (128 vst., 128 výst.)				
Rychlost booleovských instrukcí	0,22 mikrosekund/instrukce				

Tab. 5. Porovnání modelů S7-200 podle CPU



Obr. 9. Popis S7-200

## 4.2 Rozšiřovací moduly S7-200

Pro lepší řešení požadavků vaší aplikace obsahuje řada S7-200 širokou škálu rozšiřovacích modulů. Těmito rozšiřovacími moduly můžete do S7-200 přidat další funkce. V tabulce 6. je seznam rozšiřovacích modulů, které jsou v současné době k dispozici.

Rozšiřovací moduly	Typy			
Digitální moduly	vstup	8 x DC vst	8 x AC vst	16 x DC vst
	výstup	4 x DC	4 x relé	
		8 x DC výst	8 x AC výst	8 x relé
	kombinace	4 x DC vst / 4 x DC výst	8 x DC vst / 8 x DC výst	16 x DC vst / 16 x DC výst
		4 x DC vst / 4 x relé	8 x DC vst / 8 x relé	16 x DC vst / 16 x relé
Analogové moduly	vstup	4 x analogový vstup	4 x termočlánek vstup	2 x RTD vstup
	výstup	2 x analogový výstup		
	kombinace	4 x analogový vstup / 1 x analogový výstup		
Inteligentní moduly		Polohovací Ethernetový	Modemový Internetový	PROFIBUS-DP
	Jiné moduly	AS interface		

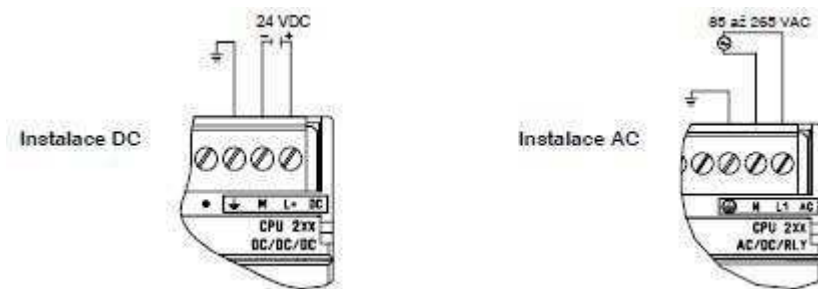
Tab. 6. Rozšiřovací moduly S7-200

## 4.3 Připojení S7-200

Připojení S7-200 je snadné. Pro tento příklad je potřeba pouze připojit napájení k CPU S7-200 a pak připojit komunikační kabel mezi programovací zařízení a S7-200.

### 4.3.1 Připojení napájení k S7-200

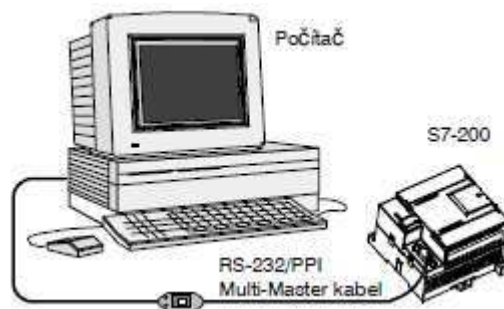
Prvním krokem je připojení S7-200 k napájecímu zdroji. Obrázek 10. ukazuje zapojení pro stejnosměrný nebo střídavý model S7-200. Před montáží nebo demontáží elektrického zařízení zajistěte, aby bylo odpojeno napájení. Vždy dodržujte příslušné bezpečnostní předpisy. Zajistěte, aby bylo napájení k S7-200 vypnuto před pokusem instalovat nebo demontovat jednotku S7-200.



Obr. 10. Připojení napájení S7-200

### 4.3.2 Možnosti komunikace

Siemens nabízí dvě programovací možnosti připojení počítače k S7-200: přímé připojení pomocí PPI Multi-Master kabelu nebo kartou komunikačního procesoru (CP) a MPI kabelem. PPI Multi-Master programovací kabel je nejběžnější a nejekonomičtější způsob připojení počítače k S7-200. Tímto kabelem se propojí komunikační port S7-200 se sériovým komunikačním portem RS-232 nebo USB portem počítače. PPI Multi-Master programovací kabel se může rovněž použít pro připojení dalších zařízení k S7-200.



Obr. 11. Připojení např. RS-232/PPI Multi-Master kabelem



#### 4.4 Jak S7-200 vykonává řídicí program

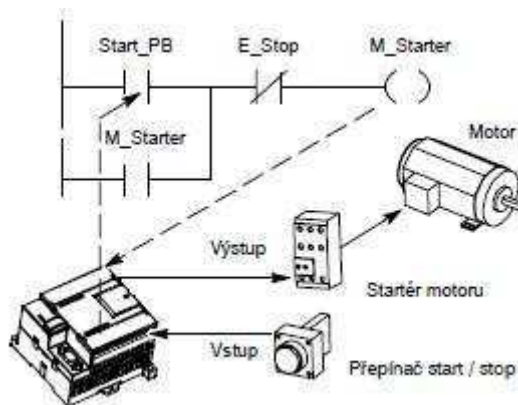
Programovatelný automat cyklicky provádí uložený řídicí program, čte a zapisuje data.

##### 4.4.1 S7-200 spojuje uložený program s fyzickými vstupy a výstupy

Základní provoz S7-200 je velmi jednoduchý:

- S7-200 přečte stav vstupů
- program uložený v automatu S7-200 použije tyto vstupy při zpracování řídicího programu, při běhu programu S7-200 aktualizuje svoje data
- automat S7-200 zapíše data na výstup

Obrázek 12. zobrazuje jednoduchý diagram vztahu kontaktního schématu a S7-200. V tomto příkladu je stav přepínače pro start motoru kombinován se stavem ostatních vstupů. Výpočtem těchto stavů pak S7 200 určí stav výstupu, který je přenesen na akční člen, jenž spustí motor.



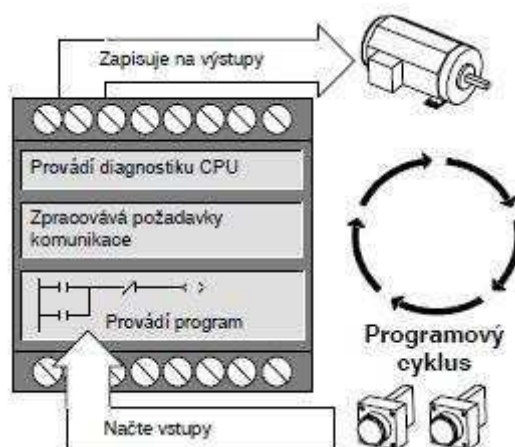
Obr. 12. Řízení vstupů a výstupů

##### 4.4.2 S7-200 provádí své úkony v programovém cyklu

S7-200 cyklicky zpracovává řadu úloh. Toto cyklické provádění jednotlivých úloh se nazývá programový cyklus. Jak je znázorněno na obrázku 13., S7-200 provádí během programového cyklu všechny následující úlohy nebo jejich větší část:

- čtení vstupů: S7-200 kopíruje stav fyzických vstupů do registru obrazu vstupů
- provádění řídicí logiky programu: S7-200 provede instrukce programu a hodnoty uloží do různých oblastí paměti
- zpracování požadavků komunikace: S7-200 provede všechny úlohy požadované pro komunikaci
- provádění autodiagnostiky CPU: S7-200 kontroluje, zda firmware, paměť pro program a všechny rozšiřovací moduly pracují správně
- zapisování na výstupy: hodnoty uložené v registru obrazu výstupů jsou zapsány na fyzické výstupy

Provádění uživatelského programu závisí na tom, zda se S7-200 nalézá v režimu STOP nebo RUN. V režimu RUN program běží, v režimu STOP nikoliv.



Obr. 13. Programový cyklus automatu S7-200

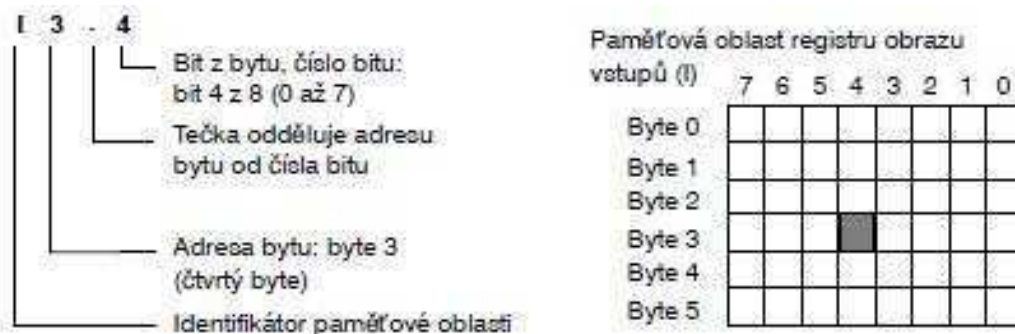
#### 4.5 Přístup k datům

Programovatelný automat S7-200 uchovává informace v různých paměťových oblastech, které mají jedinečné adresy. Můžete explicitně určit konkrétní adresu paměti. To programu umožní přímý přístup k informacím. Tabulka 7. ukazuje rozsahy, které mohou být reprezentovány daty různých velikostí.

Reprezentace	Byte (B)	Word (W)	Double word (D)
Unsigned integer	0 až 255 0 až FF	0 až 65 535 0 až FFFF	0 až 4 294 967 295 0 až FFFF FFFF
Signed integer	-128 až +127 80 až 7F	-32 768 až +32 767 8000 až 7FFF	-2 147 483 648 až +2 147 483 647 8000 0000 až 7FFF FFFF
Real IEEE 32bitová pohyblivá řádová čárka	Ne lze	Ne lze	+1.175495E-38 až +3.402823E+38 (kladné) -1.175495E-38 až -3.402823E+38 (záporné)

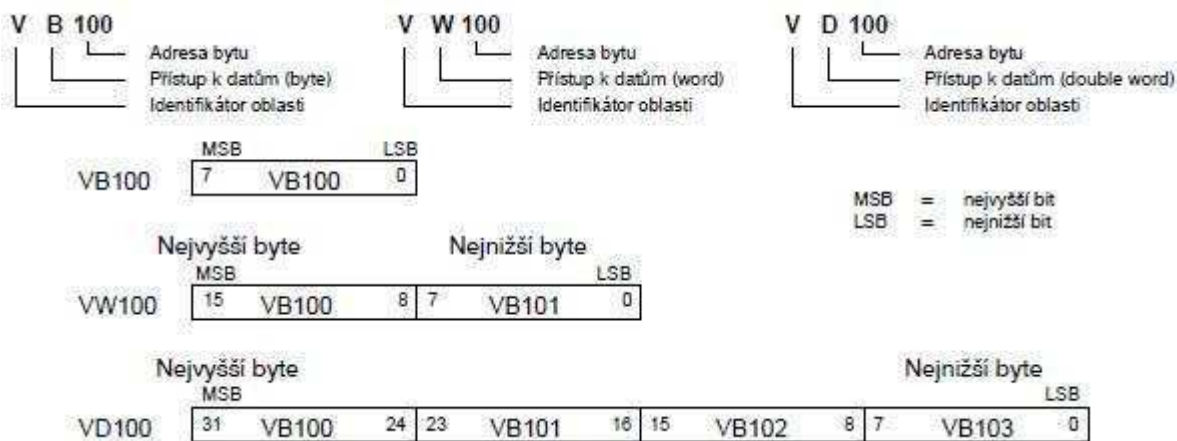
Tab. 7. Desítkové a šestnáctkové rozsahy pro různé velikosti dat

Když chcete přistoupit k nějakému bitu v paměťové oblasti, specifikujete adresu, která se skládá z identifikátoru paměťové oblasti, adresy bytu a čísla bitu. Obrázek 14. ukazuje příklad přístupu k bitu (často se nazývá adresování "byte.bit"). V tomto příkladu za paměťovou oblastí a adresou bytu (I = vstup a 3 = byte 3) následuje tečka ("."), která odděluje bitovou adresu (bit 4).



Obr. 14. Adresování byte.bit

K datům ve většině paměťových oblastí (V, I, Q, M, S, L, a SM) typu byte, word nebo double word můžete přistupovat v tzv. bytovém adresním formátu. Pro přístup k bytu, word nebo double word v paměti je nutné vytvořit adresu podobně, jako byla vytvořena adresa pro přístup k bitu. Adresa se skládá z identifikátoru oblasti, z označení velikosti dat a z adresy počátečního bytu hodnoty bytu, word nebo double word, jak to ukazuje obrázek 15.



Obr. 15. Porovnání přístupu do stejné oblasti paměti k byte, word, double word

Popis	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Velikost uživatelského programu s editací v režimu RUN bez editace v režimu RUN	4096 bytů 4096 bytů	4096 bytů 4096 bytů	8192 bytů 12288 bytů	12288 bytů 16384 bytů	16384 bytů 24576 bytů
Velikost uživatelských dat	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Registr obrazu vstupů	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7
Registr obrazu výstupů	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7
Analogové vstupy (čtení)	AIW0 až AIW30	AIW0 až AIW30	AIW0 až AIW62	AIW0 až AIW62	AIW0 až AIW62
Analogové výstupy (zápis)	AQW0 až AQW30	AQW0 až AQW30	AQW0 až AQW62	AQW0 až AQW62	AQW0 až AQW62
Paměť pro proměnné (V)	VB0 až VB2047	VB0 až VB2047	VB0 až VB8191	VB0 až VB10239	VB0 až VB10239
Lokální paměť (L) <sup>1</sup>	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63
Bitová paměť (M)	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7
Speciální paměť (SM)	SM0.0 až SM179.7	SM0.0 až SM299.7	SM0.0 až SM549.7	SM0.0 až SM549.7	SM0.0 až SM549.7
Pouze pro čtení	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7
Časovače	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)
Remanentní zpožděné zapnutí					
1 ms	T0, T64	T0, T64	T0, T64	T0, T64	T0, T64
10 ms	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68
100 ms	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95
Zpožděné zapnutí/vypnutí					
1 ms	T32, T96	T32, T96	T32, T96	T32, T96	T32, T96
10 ms	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100
100 ms	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255
Čítače	C0 až C255	C0 až C255	C0 až C255	C0 až C255	C0 až C255
Vysokorychlostní čítače	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5
Sekvenční řídicí relé (S)	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7
Registry akumulátoru	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3
Skoky/návěští	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
Volání/podprogramy	0 až 63	0 až 63	0 až 63	0 až 63	0 až 127
Přerušovací podprogramy	0 až 127	0 až 127	0 až 127	0 až 127	0 až 127
Náběžné/cestupné hrany	256	256	256	256	256
Smyčky PID	0 až 7	0 až 7	0 až 7	0 až 7	0 až 7
Porty	Port 0	Port 0	Port 0	Port 0, Port 1	Port 0, Port 1

<sup>1)</sup> LB60 až LB63 jsou rezervovány pro STEP 7- Micro/WIN, verze 3.0 nebo pozdější.

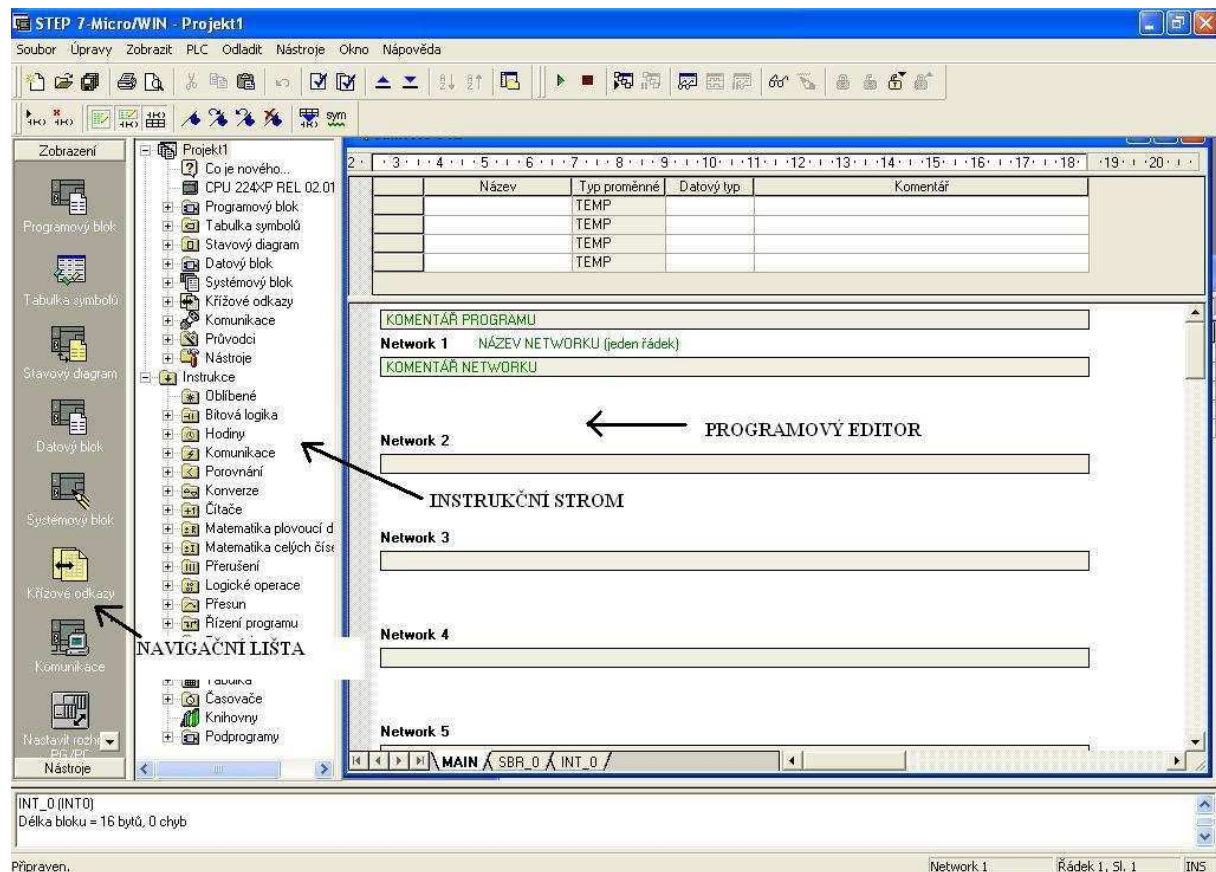
Tab. 8. Rozsahy paměti S7-200

Způsob přístupu	CPU 221	CPU 222	CPU 224	CPU 224 XP	CPU 226
Přístup po bitech (byte.bit)					
I	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7
Q	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7
V	0,0 až 2047,7	0,0 až 2047,7	0,0 až 8191,7	0,0 až 10239,7	0,0 až 10239,7
M	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7
SM	0,0 až 165,7	0,0 až 299,7	0,0 až 549,7	0,0 až 549,7	0,0 až 549,7
S	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7
T	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
C	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
L	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7
Přístup po bytech					
IB	0 až 15	0 až 15	0 až 15	0 až 15	0 až 15
QB	0 až 15	0 až 15	0 až 15	0 až 15	0 až 15
VB	0 až 2047	0 až 2047	0 až 8191	0 až 10239	0 až 10239
MB	0 až 31	0 až 31	0 až 31	0 až 31	0 až 31
SMB	0 až 165	0 až 299	0 až 549	0 až 549	0 až 549
SB	0 až 31	0 až 31	0 až 31	0 až 31	0 až 31
LB	0 až 63	0 až 63	0 až 63	0 až 63	0 až 63
AC	0 až 3	0 až 3	0 až 3	0 až 255	0 až 255
KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)
Přístup po word					
IW	0 až 14	0 až 14	0 až 14	0 až 14	0 až 14
QW	0 až 14	0 až 14	0 až 14	0 až 14	0 až 14
VW	0 až 2046	0 až 2046	0 až 8190	0 až 10238	0 až 10238
MW	0 až 30	0 až 30	0 až 30	0 až 30	0 až 30
SMW	0 až 164	0 až 298	0 až 548	0 až 548	0 až 548
SW	0 až 30	0 až 30	0 až 30	0 až 30	0 až 30
T	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
C	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
LW	0 až 62	0 až 62	0 až 62	0 až 62	0 až 62
AC	0 až 3	0 až 3	0 až 3	0 až 3	0 až 3
AIW	0 až 30	0 až 30	0 až 62	0 až 62	0 až 62
AQW	0 až 30	0 až 30	0 až 62	0 až 62	0 až 62
KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)
Přístup po double word					
ID	0 až 12	0 až 12	0 až 12	0 až 12	0 až 12
QD	0 až 12	0 až 12	0 až 12	0 až 12	0 až 12
VD	0 až 2044	0 až 2044	0 až 8188	0 až 10238	0 až 10238
MD	0 až 28	0 až 28	0 až 28	0 až 28	0 až 28
SMD	0 až 162	0 až 296	0 až 546	0 až 546	0 až 546
SD	0 až 28	0 až 28	0 až 28	0 až 28	0 až 28
LD	0 až 60	0 až 60	0 až 60	0 až 60	0 až 60
AC	0 až 3	0 až 3	0 až 3	0 až 3	0 až 3
HC	0 až 5	0 až 5	0 až 5	0 až 5	0 až 5
KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)

Tab. 9. Rozsahy operandů S7-200

#### 4.6 Použití STEP 7-Micro/WIN pro vytvoření programu

Chcete-li otevřít STEP 7-Micro/WIN, dvakrát klikněte na ikonu STEP 7-Micro/WIN nebo vyberte příkaz menu **Start > SIMATIC > STEP 7 MicroWIN 32 V4.0**. Jak je vidět na obrázku 16., okno projektu STEP 7-Micro/WIN nabízí přehledné pracovní prostředí pro vytváření řídicího programu.



Obr. 16. STEP 7-Micro/WIN

Nástrojové lišty obsahují tlačítka pro zkratky k často používaným příkazům menu. Všechny nástrojové lišty můžete zobrazit nebo skrýt.

Strom s instrukcemi zobrazuje všechny objekty projektu a instrukce pro tvorbu řídicího programu. Jednotlivé instrukce můžete do programu vložit metodou drag&drop nebo můžete na instrukci dvakrát kliknout, čímž ji vložíte na současnou pozici kurzoru v programovém editoru.

Programový editor obsahuje program a tabulku lokálních proměnných, ve které můžete přiřadit symbolické názvy dočasným lokálním proměnným. Podprogramy a přerušení jsou zobrazeny jako záložky ve spodní části okna programového editoru. Chcete-li přecházet mezi podprogramy, přerušeními a hlavním programem, klikněte na příslušnou záložku.

STEP 7-Micro/WIN obsahuje tři editory pro vytváření uživatelského programu: kontaktní schémata (LAD), výpis příkazů (STL) a funkční bloky (FBD). S určitými omezeními mohou být programy, psané v kterémkoliv z těchto programových editorů, prohlíženy a editovány ostatními programovými editory.

### 4.6.1 Vlastnosti editoru STL

Editor STL zobrazuje program jako znakově orientovaný programovací jazyk. Umožňuje vytvářet řídicí programy vkládáním textových instrukcí. Editor STL také umožňuje tvorbu programů, které by pomocí editoru LAD nebo FBD nešly vytvořit. Je to proto, že v STL programujete v jazyku S7-200 a nikoli v jazyku grafického editoru, kde platí určitá omezení, aby byly diagramy správně nakresleny. Jak je vidět na obrázku 17., je tato znakově orientovaná koncepce velmi podobná programování ve strojovém kódu.

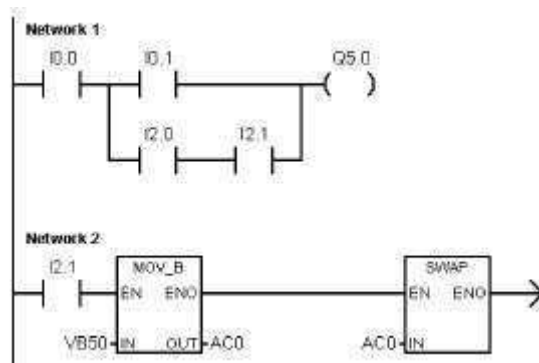
```
LD  I0.0    //Načtení jednoho vstupu
A   I0.1    //Logický součin s jiným
=   Q1.0    //Zapsání hodnoty na výstup 1
```

Obr. 17. Příklad programu STL

### 4.6.2 Vlastnosti editoru LAD

Editor LAD zobrazuje program v grafické formě podobné schématům. Programy v kontaktním schématu umožňují simulovat tok elektrického proudu z napájecího zdroje přes řadu logických vstupních podmínek, které následně aktivují výstupní logické podmínky.

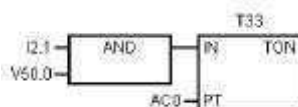
Program LAD obsahuje levou napájecí lištu, která je pod napětím. Kontakty, které jsou sepnuté, umožňují tok energie do dalšího prvku; kontakty, které jsou rozepnuté, tok energie blokují.



Obr. 18. Příklad programu LAD

### 4.6.3 Vlastnosti editoru FBD

Editor FBD zobrazuje program v grafické formě, která připomíná běžná logická schémata. Neobsahuje kontakty ani cívký, které se nalézají v editoru LAD, ale ekvivalentní instrukce, které se objevují jako blokové instrukce.



Obr. 19. Příklad programu FBD

## 5. OPERÁTORSKÝ TERMINÁL GOT1000

Společnost Mitsubishi Electric touto řadou dotykových operátorských terminálů GOT1000 opět zavedla nový standard v oblasti komunikace člověk – stroj. Při vývoji se nevycházelo pouze z přání a očekávání zákazníků, ale rovněž z pokrokových technologií a obrovských zkušeností, jež se odvíjejí z dlouholeté výroby operátorských terminálů. Výsledkem je produkt, který usnadňuje práci jak programátorům, tak pracovníkům údržby a obsluze.

### 5.1 Výkonná technologie

Nové terminály nejen lahodí oku, ale jsou též vybaveny řadou výkonných funkcí:

#### **Vysokorychlostní zpracování:**

64bitový procesor RISC použitý v řadě GOT1000 je kombinován s nově vyvinutým velmi rychlým grafickým procesorem. Výsledkem této kombinace je působivě krátká doba odezvy a vytvoření obrazu.

#### **Rozhraní USB:**

Díky USB rozhraní na přední straně přístroje je možné přenášet projektová data, aniž by bylo nutné otevírat rozvaděč nebo řídicí konzolu. Rozhraní je opatřeno bezpečnostním krytem a odpovídá parametrům třídy ochrany IP67. V transparentním režimu lze toto rozhraní využít rovněž k programování nebo diagnostice MELSEC PLC a dalších automatizačních komponent.

#### **Větší flexibilita díky přenosným přístrojům:**

V rozsahu řady GOT jsou zahrnuty také přenosné verze (Handy-GOT), které umožňují flexibilní použití operátorského terminálu s příslušnou aplikací.

Inovativní systém chybových hlášení řady GOT1000 zajišťuje rychlé rozpoznání chyb, čímž se minimalizují prostoje. Předem definované obrazovky nabízejí přímý přístup ke vstupům, výstupům a k datovému registru PLC a dokonce k vyrovnávací paměti speciálních funkčních modulů.



Obr. 20. Operátorský terminál GOT1000



Obrazovka s vysokým rozlišením až 65.536 barvami je schopná zobrazit jak komplexní grafické projekty, tak fotografie, nebo CAD výkresy.

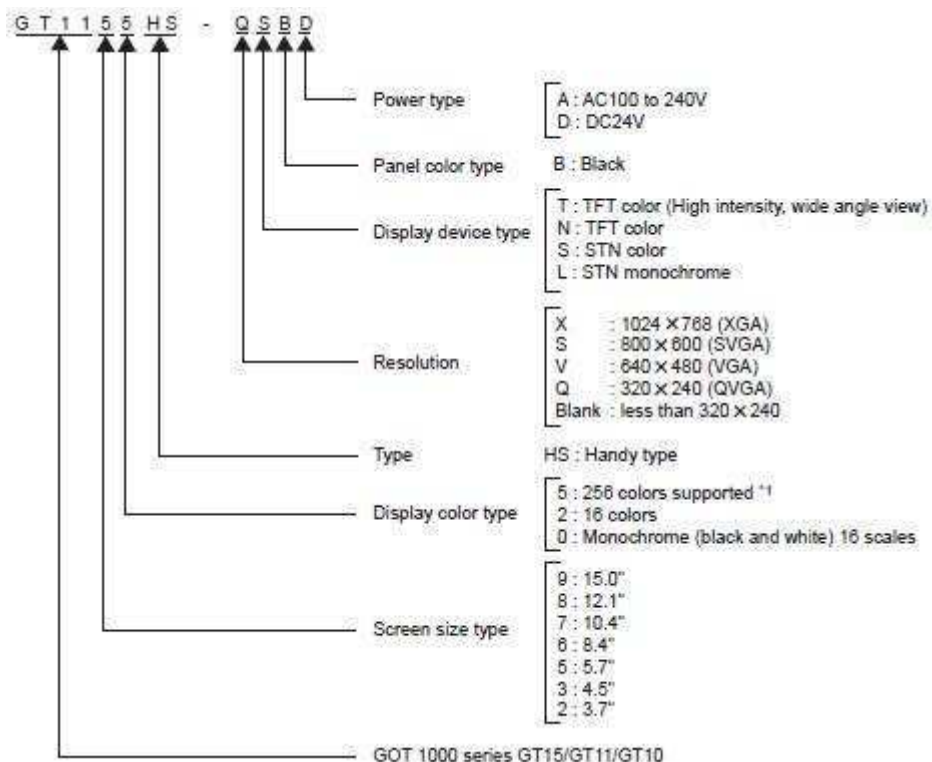
### Rozšířená podpora protokolů:

Kromě širokého spektra podporovaných produktů – Mitsubishi PLC, frekvenčních měničů a servo zesilovačů lze řadu GOT1000 připojit také k automatizačním zařízením jiných výrobců jejichž počet narůstá. Díky tomuto mohou uživatelé pro své aplikace využít vizualizační řešení přesahující jednotlivé systémy.

## 5.2 Typy operátorských terminálů GOT1000

GOT1000 Series	GT SoftGOT1000		Abbreviation of GT SoftGOT1000
	GT1895	GT1895M-X	Abbreviation of GT1895M-XTBA, GT1895M-XTBD
	GT1885	GT1885M-S	Abbreviation of GT1885M-STBA, GT1885M-STBD
	GT18□□, GT18		Abbreviation of GT1895, GT1885
	GT1595	GT1595-X	Abbreviation of GT1595-XTBA, GT1595-XTBD
	GT1585	GT1585V-S	Abbreviation of GT1585V-STBA, GT1585V-STBD
		GT1585-S	Abbreviation of GT1585-STBA, GT1585-STBD
	GT157□	GT1575V-S	Abbreviation of GT1575V-STBA, GT1575V-STBD
		GT1575-S	Abbreviation of GT1575-STBA, GT1575-STBD
		GT1575-V	Abbreviation of GT1575-VTBA, GT1575-VTBD
		GT1575-VN	Abbreviation of GT1575-VNBA, GT1575-VNBD
	GT1572-VN	Abbreviation of GT1572-VNBA, GT1572-VNBD	
	GT156□	GT1565-V	Abbreviation of GT1565-VTBA, GT1565-VTBD
		GT1562-VN	Abbreviation of GT1562-VNBA, GT1562-VNBD
	GT155□	GT1555-V	Abbreviation of GT1555-VTBD
		GT1555-Q	Abbreviation of GT1555-QTBD, GT1555-QSBD
		GT1550-Q	Abbreviation of GT1550-QLBD
	GT15□□, GT15		Abbreviation of GT1595, GT1585, GT157□, GT156□, GT155□
	GT115□	GT1155-Q	Abbreviation of GT1155-QTBDQ, GT1155-QSBDQ, GT1155-QTBDA, GT1155-QSBDA, GT1155-QTBD, GT1155-QSBD
		GT1150-Q	Abbreviation of GT1150-QLBDQ, GT1150-QLBDA, GT1150-QLBD
	Handy GOT	GT1155HS-Q	Abbreviation of GT1155HS-QSBD
		GT1150HS-Q	Abbreviation of GT1150HS-QLBD
	GT11□□, GT11		Abbreviation of GT115□, GT11 Handy GOT
	GT105□	GT1055-Q	Abbreviation of GT1055-QSBD
		GT1050-Q	Abbreviation of GT1050-QBBD
	GT1030		Abbreviation of GT1030-LBD, GT1030-LBD2, GT1030-LBDW, GT1030-LBDW2
GT1020		Abbreviation of GT1020-LBD, GT1020-LBD2, GT1020-LBL, GT1020-LBDW, GT1020-LBDW2, GT1020-LBLW	
GT10□□, GT10		Abbreviation of GT105□, GT1030, GT1020	

Tab. 10. Přehled operátorských terminálů GOT1000



Obr. 21. Kódování výrobního čísla z GOT1000

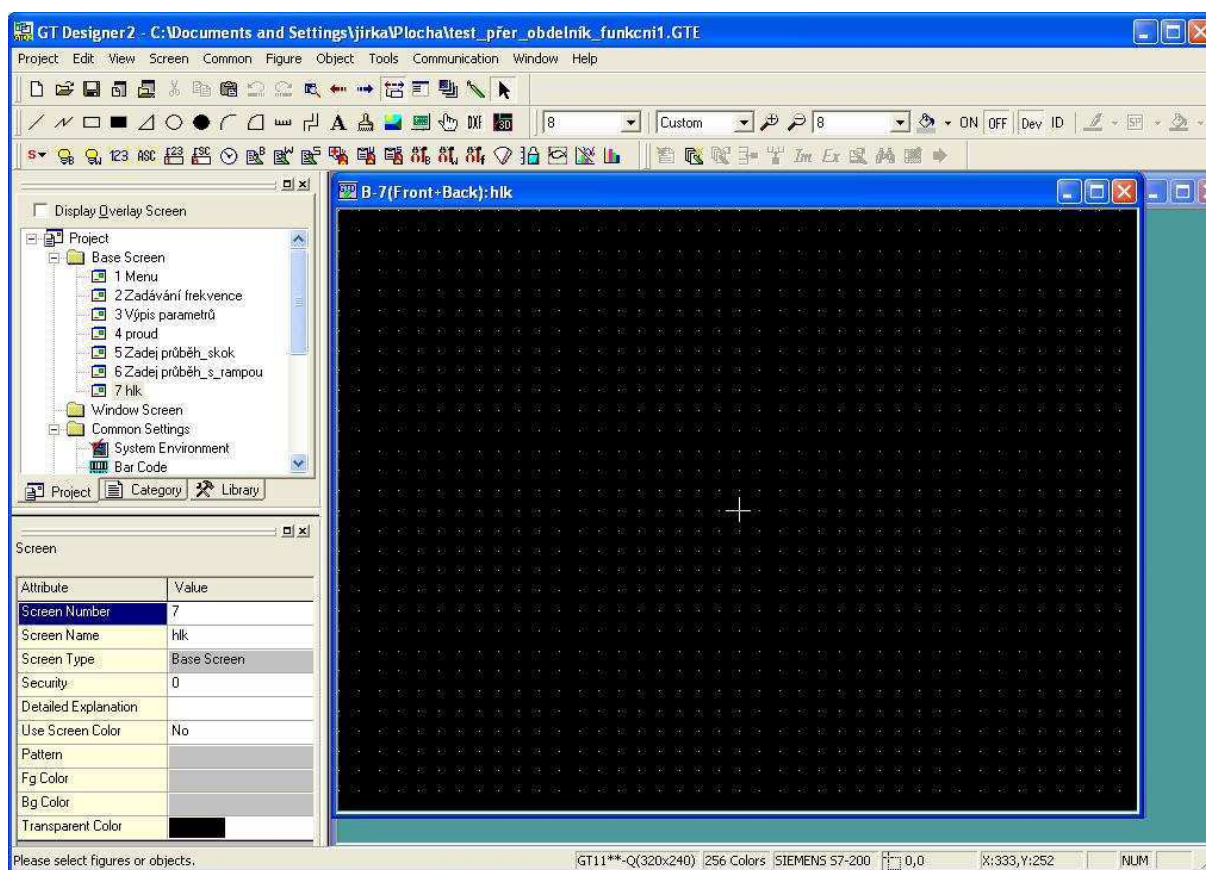
Technické údaje		GT10 (10 Zařízení)	GT11 (4 Zařízení)	GT15 (24 Zařízení)
Displej	Provedení	STN	STN	STN, TFT
	Rozměry (diagonální)	3,7" – 4,5"	5,7"	5,7" – 15"
	Text (Řádky x Znak)	Volně definovatelné	Volně definovatelné	Volně definovatelné
	Výška znaku (mm)	Volně definovatelné, fonty Windows	Volně definovatelné, fonty Windows	Volně definovatelné, fonty Windows
	Max. rozlišení (Pixel)	160 x 64 / 288 x 96	320 x 240	320 x 240 – 1024 x 768
Napájecí zdroj	5 V DC / 24 V DC	24 V DC	24 V DC / 220 V AC	
Kapacita paměti	512 KB / 1,5 MB	3 MB	5 – 9 MB (rozšiřitelná do 57 MB)	
Externí paměťová karta	—	1 (Compact-Flash, max. 2 GB)	1 (Compact-Flash, max. 2 GB)	
Typ klávesnice	Dotykový panel	Dotykový panel	Dotykový panel	
Funkční klávesy	Dotykové klávesy	Dotykové klávesy + 6 funkčních kláves	Dotykové klávesy	
Indikátory LED	—	1 (Power ON)	1 (Power ON)	
Rozhraní	Sériové	2 x RS232, RS422 / RS232	RS232C, RS422	RS232
	Další	—	USB	USB (vpředu)
Připojení do sítě	Sériové	Sériové	Ethernet (TCP/IP), CC-Link, RS422, RS485, A-Bus, Q-Bus, MELSECNET/10/H	
Krytí (přesní strana)	IP67	IP67 / IP65 (Tada Handy-GOT)	IP67	

Tab. 11. Technické údaje

### 5.3 Grafický software GT-Designer 2

Softwarový balíček GT Designer 2 s rozsáhlou knihovnou grafických prvků v celé řadě stylů umožňuje snadno a rychle vytvářet informativní graficky atraktivní obrazovky. S GT Designer 2 si vzhled obrazovky přizpůsobíte podle vlastních představ. Integrovaná funkce průvodce pomáhá uživatelům při zakládání nového projektu krok za krokem. Tato uživatelsky přívětivá pomůcka začátečníkům usnadňuje první kroky a profesionálům pomáhá významně redukovat činnosti nezbytné pro nastavení.

Závěrečná kontrola vzhledu a funkcí obrazovky je stejně jednoduchá jako nastavování. Se softwarem GT Designer 2 lze vytvářet náhledy obrazovek, přepínat jazyky a testovat bezpečnostní nastavení. Všechny uvedené funkce jsou součástí tohoto výkonného softwarového balíčku. Lze jej využít rovněž pro simulaci provozu všech zařízení řady GOT1000.



Obr. 22. Grafický software GT Designer 2

## 6. ASYNCHRONNÍ STROJE

Ve srovnání se stejnosměrnými stroji vyniká jednoduchá konstrukce a prakticky bezúdržbový provoz asynchronních strojů. Jejich rozšíření je spojeno s rozvojem distribučních a napájecích sítí. Především v neregulovaných pohonech se využívá možnost jednoduchého spouštění přímým připojením na síť, i když problémy může činit vzniklý proudový náraz (pěti až sedminásobek jmenovitého proudu) a velký odběr jalového výkonu. Snaha o omezení záběrového (spouštěcího) proudu a splnění požadavků na regulaci vedla ke konstrukci speciálních strojů buď vinutým (kroužkovým) rotorem, čímž však zanikla hlavní výhoda jednoduchých asynchronních strojů s klecovým rotorem, případně strojů více rychlostních umožňujících za cenu větší složitosti statorového vinutí stupňovou regulaci otáčivé rychlosti. Teprve nové poznatky v oblasti výkonové elektroniky a regulace zcela vyřešili problémy s kmitočtovým řízením rychlosti asynchronních strojů, které tak dnes získali dominantní postavení i v oblasti regulovaných pohonů všeobecného použití.

### 6.1 Princip funkce asynchronních strojů

Podstatou funkce střídavých točivých strojů je vznik točivého magnetického pole. Nutná podmínka pro jeho vybudování je existence  $n$  vinutí, prostorově posunutých po obvodu stroje, napájených  $n$ -fázovou soustavou střídavých proudů s časovým posunem jejich okamžitých hodnot v jednotlivých obvodech. V konkrétním případě se jedná o trojfázové systémy se souměrným prostorovým uspořádáním os vynutí po  $120^\circ$ , protékajícími proudy, tvořícími souměrný trojfázový systém.

### 6.2 Kmitočtová regulace rychlosti asynchronních strojů

Všechny způsoby regulace otáčivé rychlosti asynchronního stroje vycházejí ze vztahu:

$$n = (60 \cdot f_1) \cdot p \cdot (1 - s)$$

Asynchronní stroj nakrátko, připojený na síť konstantního kmitočtu a napětí, má pouze jednu momentovou charakteristiku, a libovolnému požadovanému momentu z rozsahu od chodu naprázdno do maximálního momentu odpovídá pouze jeden stabilní bod této charakteristiky, ve kterém stroj může pracovat.

Odpovídající otáčkový rozsah tvoří pouze zlomek intervalu od nulových otáček do otáčivé rychlosti, odpovídající chodu naprázdno. Máme-li však možnost měnit nezávisle napájecí kmitočet a napětí, můžeme požadovaný moment a otáčky dosáhnout za různých podmínek. Jsou tedy k dispozici různé charakteristiky. Z toho vyplývá, že k dosažení širokého regulačního rozsahu musíme napájet AS ze zdroje proměnného kmitočtu, umožňujícího současnou změnu výstupního napětí.



## 7. SPOJENÍ MEZI FM A S7-200

Na začátku mého řešení bylo nejdůležitější vyřešit spojení (komunikaci) mezi frekvenčním měničem a programovatelným automatem. A zvážit možnost obsluhovat automat operátorským panel BOP (Basic Operator Panel) z pracoviště ÚAI, kterým se daný měnič programuje.

Pro moji diplomovou práci mi poskytlo pracoviště ÚAI frekvenční měnič MICROMASTER 440. Jedná se o frekvenční měnič s jednofázovým napájecím napětím 230V o výkonu pro motory do 0,25 kW. Tudíž je použitelný pro řízení daného asynchronního motoru o výkonu 0,18kW.

Jako programovatelný automat je použit typ Simatic S7-200 CPU 224.

### 7.1 Operátorský panel BOP (Viz. kapitola 3.3 Ovládací panely měniče.)

Jelikož panel BOP nemá sériové rozhraní RS-485, [má ho pouze panel AOP (Advanced Operator Panel)], kterým bych ho připojil na port automatu nebo jinou komunikační možnost s automatem, od této volby jsem tedy musel upustit. A zabýval se možností přímo ovládání měniče automatem.

### 7.2 Možnosti řízení MICROMASTER a Simatic S7-200

#### Řízení pomocí vstupů / výstupů

- až 6 výstupů na pohon
- drátování
- statické ovládání
- žádná možnost volby parametrů
- ✓ levné
- ✓ jednoduché



Obr. 23. Digitální řízení

#### Analogové řízení

- drátování
- jeden analogový výstup na pohon
- žádná možnost volby parametrů
- ✓ jednoduché



Obr. 24. Analogové řízení

#### Řízení pomocí USS protokolu

- knihovna USS
- ✓ jednoduché
- ✓ až 31 pohonů
- ✓ dynamická kontrola
- ✓ možnost změny parametrů
- ✓ menší náklady
- ✓ méně drátování



Obr. 25. Řízení USS protokolem

### 7.3 Zvolené řízení

Z těchto tří možností které jsou k dispozici, se jako nejlepší a nejvýhodnější jeví použití USS protokolu. Jedná se o knihovnu do STEP 7-Micro/WIN, která je speciálně navržena pro komunikaci s pohonem. Pomocí instrukcí USS můžete řídit fyzický pohon a číst, popř. zapisovat jeho parametry.

Avšak jak se později ukázalo tato knihovna bohužel není standardně součástí STEP 7-Micro/WIN a musí se do ní dokoupit. S prosbou o věnování této knihovny jsme se obrátily na Siemens v Brně kde nám vyšli vstříc a tuto knihovnu věnovaly, s tím že bude využívána pro studijní účely fakulty FSI. Dostaly jsme knihovnu USS4, která slouží přímo k řízení měničů typu MICROMASTER řady 4.

Podrobný popis protokolu USS4 viz kapitola 8.

## 8. USS4 PROTOKOL

Knihovny instrukcí STEP 7-Micro/WIN usnadňují řízení pohonu MICROMASTER, protože obsahují již nakonfigurované podprogramy přerušení a jiné podprogramy, které jsou speciálně navrženy pro použití USS protokolu při komunikaci s pohonem. Pomocí instrukcí USS můžeme řídit fyzický pohon a číst, popř. zapisovat jeho parametry.

### 8.1 Požadavky použití USS4 protokolu

Knihovny instrukcí ve STEP 7-Micro/WIN obsahují 14 podprogramů, 3 podprogramy přerušení a 8 instrukcí, které podporují USS4 protokol. Instrukce USS4 používají v automatu S7-200 následující prostředky:

- Inicializace USS4 protokolu vyhradí port 0 pro komunikace USS4.  
Instrukce USS4\_INIT se používá pro volbu USS4 nebo PPI pro port 0. (USS4 se vztahuje na pohony SIMOTION MICROMASTER řady 4.) Poté, co zvolíte pro komunikaci s pohony použití USS4 protokolu, nemůžete port 0 používat pro žádný jiný účel včetně komunikace se STEP 7-Micro/WIN.  
Během programování aplikace používající USS protokol byste měli používat CPU 224XP, CPU 226 nebo modul EM 277 PROFIBUS-DP s připojením na kartu PROFIBUS CP. Tento druhý komunikační port umožňuje STEP 7-Micro/WIN monitorovat aplikaci během chodu USS protokolu.
- Instrukce USS4 ovlivňují všechna místa paměti SM spojená s komunikací Freeport na portu 0.
- Instrukce USS4 používají 14 podprogramů a 3 podprogramy přerušení.
- Díky instrukcím USS4 se zvětšuje část paměti potřebná pro uživatelský program až o 3000 bytů. V závislosti na konkrétních použitých instrukcích USS4 mohou podpůrné podprogramy pro tyto instrukce zvýšit režii pro řídicí program nejméně o 2300 bytů, až na 3600 bytů.
- Proměnné pro instrukce USS4 potřebují 400bytový blok v paměti V. Počáteční adresa tohoto bloku začíná na adrese VB4723 a je rezervována pro proměnné USS4.
- Některé z instrukcí USS4 také vyžadují 16bytový komunikační zásobník. Uživatel přiděluje počáteční adresu v paměti V pro tento zásobník jako parametr instrukce. Doporučuje se přiřadit vlastní zásobník každé instrukci USS4.
- Při provádění výpočtu používají instrukce USS4 akumulátory AC0 až AC3. Ve vašem programu můžete také použít akumulátory, ale hodnoty v akumulátorech budou instrukcemi USS4 změněny.
- Instrukce USS4 není možné používat v podprogramu přerušení.

Poznámka:

Chcete-li změnit činnost portu 0 zpět na PPI, abyste mohli komunikovat se STEP 7-Micro/WIN, použijte další instrukci USS4\_INIT, která změní přiřazení portu 0.



## 8.2 Použití instrukcí USS4

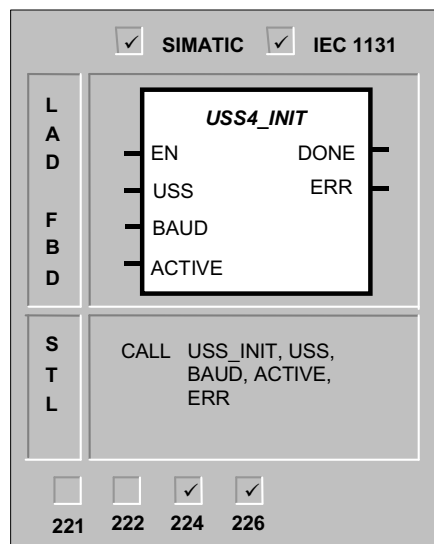
Pro použití instrukcí USS protokolu v programu programovatelného automatu S7-200 proveďte tyto kroky:

1. Vložte do svého programu instrukci USS4\_INIT a provádějte ji pouze po dobu jednoho programového cyklu. Instrukci USS4\_INIT můžete použít buď pro inicializaci, nebo pro změnění parametru komunikace USS4.  
Když vložíte instrukci USS4\_INIT, přidá se k vašemu programu automaticky několik skrytých podprogramů a podprogramů přerušení.
2. Pro každý aktivní pohon vložte do svého programu pouze jednu instrukci USS4\_CRV\_CTRL.  
Instrukcí USS4\_RPM\_x a USS4\_WPM\_x můžete vložit tolik, kolik potřebujete, ale pouze jedna z nich může být v určitou dobu aktivní.
3. Nakonfigurujte parametry pohonu tak, aby odpovídaly přenosové rychlosti a adrese použité programem.
4. Zapojte komunikační kabel mezi automatem S7-200 a pohony. Zajistěte, aby všechna řídicí zařízení, jako je například S7-200, která jsou propojena s pohonem, byla spojena krátkým silným kabelem se stejným uzemněním nebo nulovým bodem jako pohon.

## 8.3 Instrukce USS4\_INIT

Instrukce USS4\_INIT se používá pro povolení a inicializaci nebo pro ukončení komunikace s pohony MICROMASTER.

Instrukce USS4\_INIT musí být dokončena bez chyb před použitím jakékoliv jiné instrukce USS4. Dříve než může být provedena další instrukce, musí být prováděná instrukce ukončena a nastaven bit DONE.



Obr. 26 Volání instrukce USS4\_INIT

Tato instrukce je provedena v každém programovém cyklu, když je zapnutý vstup EN. Instrukci USS4\_INIT proveďte pro každou změnu ve stavu komunikace pouze jednou. Proto by vstup EN měl být zapojený pulzně, přes vyhodnocující náběžnou hranu. Chcete-li změnit parametry inicializace, proveďte novou instrukci USS4\_INIT.

Parametr USS volí komunikační protokol:

vstupní hodnota 1 přiřazuje port 0 USS4 protokolu a aktivuje protokol

vstupní hodnota 0 přiřazuje port 0 protokolu PPI a blokuje USS4 protokol.

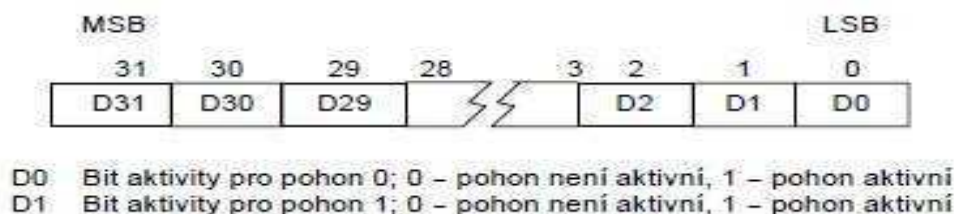
BAUD nastavuje přenosovou rychlost na 1200, 2400, 4800, 9600, 19200, 38400.

ACTIVE indikuje, které pohony jsou aktivní. Některé pohony podporují pouze adresy 0 až 30.

Vstupy/Výstupy	Operandy	Typ dat
USS	VB, IB, QB, MB, SB, SMB, LB, AC, Constant, *VD, *AC, *LD	BYTE
BAUD	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC *VD, *AC, *LD	WORD
ACTIVE	VD, ID, QD, MD, SD, SMD, LD, AC, Constant, *VD, *AC, *LD	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

Tabulka 12. Operandy platné pro USS4\_INIT

Obrázek 27. ukazuje popis a formát vstupu aktivního pohonu. Kterýkoliv pohon označený jako ACTIVE je automaticky dotazován na pozadí tak, aby řídil pohon, shromažďoval stav a zabránil časovému odpojení sériové linky pohonu.



Obr. 27. Formát parametru aktivního pohonu

Když je dokončena instrukce USS4\_INIT, zapne se výstup DONE. Výstupní byte ERR obsahuje výsledek provádění této instrukce. Tabulka 16. definuje chyby, které mohou vyplynout z provedení této instrukce.

Ukázka volání instrukce USS4\_INIT:

```
LD SM0.1
CALL USS4_INIT:SBR50, 1, 9600, 16#1, Q0.0, VB0
```

- Popis:
- 1 = přiřadí portu 0 USS4 protokol
  - nastaví přenosovou rychlost na 9600 bps
  - 16#1 = označí pohon 0 za aktivní
  - Q0.0 = DONE signalizuje dokončení instrukce USS4\_INIT
  - VB0 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

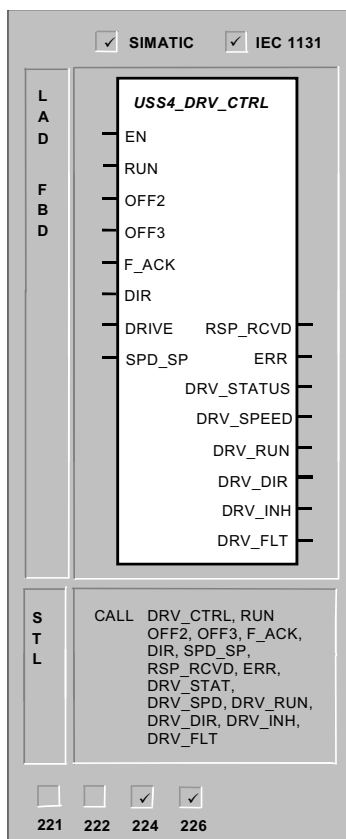
#### 8.4 Instrukce USS4\_DRV\_CTRL

Instrukce USS4\_DRV\_CTRL se používá na řízení aktivního pohonu MICROMASTER. Tato instrukce uloží vybrané příkazy do komunikačního zásobníku, který je pak odeslán adresovanému pohonu (parametr DRIVE - pohon), pokud byl tento pohon vybrán v parametru ACTIVE instrukce USS4\_INIT.

Každému pohonu by měla být přiřazena pouze jedna instrukce USS4\_DRV\_CTRL. Aby byla povolena instrukce USS4\_DRV\_CTRL, musí být zapnutý bit EN. Tato instrukce by měla být vždy povolena.

RUN (START/STOP) indikuje, zda je pohon zapnutý (1), nebo vypnutý (0). Když je bit RUN zapnutý, dostane pohon MICROMASTER příkaz, aby se rozběhl se zadanou rychlostí a směrem. Aby byl pohon uveden do chodu, musí platit následující:

- Pohon musí být vybrán jako ACTIVE v USS4\_INIT.
- OFF2 a OFF3 musí být nastaveny na 0.
- DRV\_FLT (Porucha) a DRV\_INH (Blokování) musí být 0.



Obr. 28. Volání instrukce USS4\_DRV\_CTRL

Když je bit RUN vypnutý, je pohonu MICROMASTER poslán příkaz, aby se zastavil zpomalením.

Bit OFF3 se používá, aby umožnil pohonu MICROMASTER volně doběhnout až do zastavení.

Bit OFF2 se používá, jestliže chceme pohonu MICROMASTER přikázat, aby zastavil okamžitě.

Bit F\_ACK (Potvrzení poruchy) se používá pro potvrzení poruchy pohonu. Pohon vymaže poruchu (DRV\_FLT), když F\_ACK přejde z 0 na 1.

Bit DIR (Směr) ukazuje, kterým směrem by se měl pohon pohybovat.

Vstup DRIVE (Adresa pohonu) je adresa pohonu MICROMASTER, na kterou má být poslán příkaz USS4\_DRV\_CTRL. Platné adresy: 0 až 31.

SPD\_SP (Nastavená hodnota rychlosti) je rychlost motoru jako procento plných otáček. Záporné hodnoty SPD\_SP způsobí, že se pohon bude pohybovat v opačném směru. Dosah: -200,0 % až 200,0 %.

Bit RSP\_RCVD (Odpověď přijata) potvrzuje odpověď pohonu. Všechny aktivní pohony jsou cyklicky dotazovány na aktuální informace o stavu pohonu. Pokaždé, když automat S7-200 dostane odpověď od pohonu, zapne se bit RSP\_RCVD po dobu jednoho programového cyklu a všechny hodnoty jsou aktualizovány.

ERR (Chyba) je chybový byte, který obsahuje výsledek posledního požadavku na komunikaci s modulem. Tabulka 16. definuje chyby, které mohou vyplynout z provedení této instrukce.

DRV\_STATUS (Stav) je nezpracovaná hodnota stavového slova, odeslaného zpět pohonem. Obrázek X-X ukazuje stavové bity pro DRV\_STATUS .

DRV\_SPEED (Rychlost) je rychlost pohonu jako procento plných otáček. Dosah: -200,0 % až 200,0 %.

DRV\_RUN (Povolení RUN) indikuje, zda je pohon v chodu (1), nebo vypnutý (0).

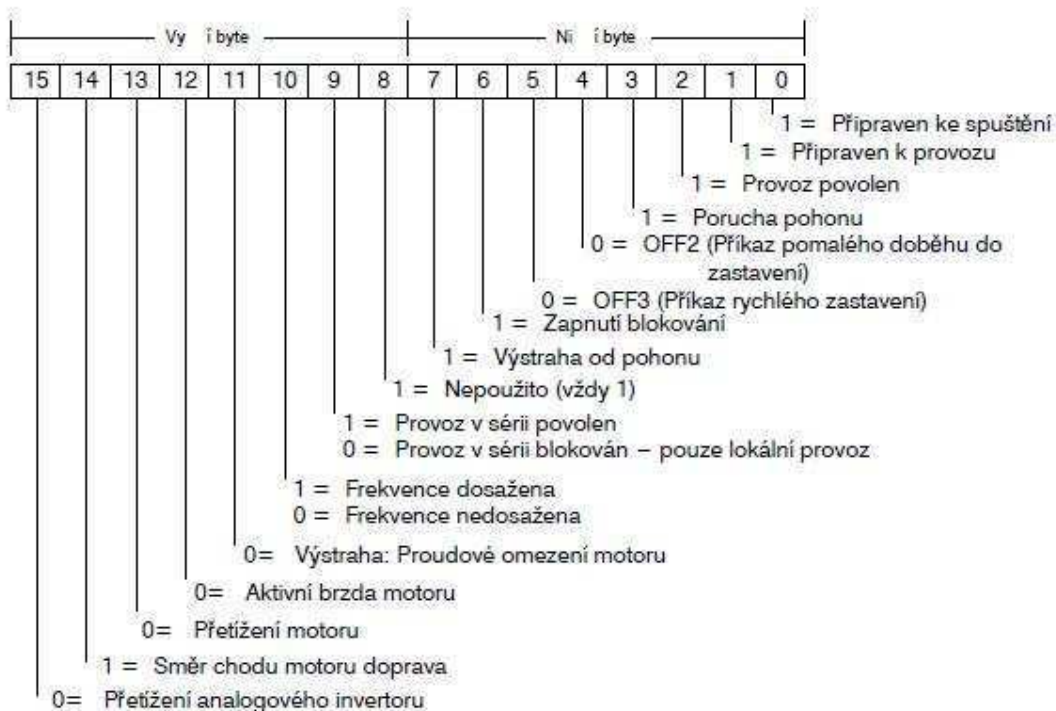
DRV\_DIR indikuje směr pohybu pohonu.

DRV\_INH indikuje stav blokovacího bitu pro pohon (0- není blokován, 1 - blokován). Chcete-li vymazat blokovací bit, musí být vypnutý bit DRV\_INH; vstupy RUN, OFF2 a OFF3 musí být také vypnuté.

DRV\_FLT indikuje stav bitu pro poruchu (0 - bez poruchy, 1 - porucha). Poruchový kód odráží stav pohonu. (Viz příručka pro pohon.) Abyste vymazali bit DRV\_FLT, opravte příčinu poruchy a nastavte bit F\_ACK.

Vstupy/Výstupy	Operandy	Typy dat
RUN	I, Q, M, S, SM, T, C, V, L, Power Flow	BOOL
OFF2	I, Q, M, S, SM, T, C, V, L, Power Flow	BOOL
OFF3	I, Q, M, S, SM, T, C, V, L, Power Flow	BOOL
F_ACK	I, Q, M, S, SM, T, C, V, L, Power Flow	BOOL
DIR	I, Q, M, S, SM, T, C, V, L, Power Flow	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, Constant, *VD, *AC, *LD	BYTE
SPD_SP	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD, Konstant	REAL
RSP_RCVD	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
DRV_STATUS	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD	WORD
DRV_SPEED	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL
DRV_RUN	I, Q, M, S, SM, T, C, V, L	BOOL
DRV_DIR	I, Q, M, S, SM, T, C, V, L	BOOL
DRV_INH	I, Q, M, S, SM, T, C, V, L	BOOL
DRV_FLT	I, Q, M, S, SM, T, C, V, L	BOOL

Tabulka 13. Operandy platné pro USS4\_DRV\_CTRL



Obr. 29. Stavové bity DRV\_STATUS

Ukázka volání instrukce USS4\_DRV\_CTRL:

```
LD  SM0.0
=   L60.0
LD  M2.0
=   L63.7
LD  M2.1
=   L63.6
LD  M2.2
=   L63.5
LD  M2.3
=   L63.4
LD  M2.4
=   L63.3
LD  L60.0
CALL USS4_DRV_CTRL:SBR51, L63.7, L63.6, L63.5, L63.4, L63.3, 0, VD100, V50.0,
      VB1, VW2, VD4, M0.0, M0.1, M0.2, M0.3
```

Popis:

- M2.0 = RUN spouštění pohonu
- M2.1 = OFF2 okamžité zastavení pohonu
- M2.2 = OFF3 volné doběhnutí motoru
- M2.3 = F\_ACT porucha
- M2.4 = DIR směr

\*\*\*\*\*

0 = DRIVE adresa pohonu

VD100 = nastavená hodnota rychlosti

V50.0 = RSP\_RCVD odpověď přijata

VB1 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

VW2 = DRV\_STATUS stavové slova odeslané zpět pohonem

VD4 = aktuální rychlost

\*\*\*\*\*

M0.0 = DRV\_RUN indikace povolení RUN

M0.1 = DRV\_DIR indikuje směr otáčení

M0.2 = DRV\_INH indikuje stav blokovacího bitu pro pohon

M0.3 = DRV\_FLT indikuje stav bitu pro poruchu

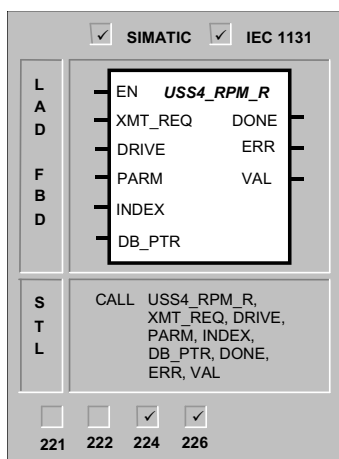
## 8.5 Instrukce USS4\_RPM\_x

V protokolu USS4 existují tři instrukce pro čtení:

- Instrukce USS4\_RPM\_W čte parametr word bez znaménka.
- Instrukce USS4\_RPM\_D čte parametr double word bez znaménka.
- Instrukce USS4\_RPM\_R čte parametr s plovoucí desetinnou čárkou.

V jednu dobu může být aktivní pouze jedna instrukce pro čtení (USS4\_RPM\_x) nebo zápis (USS4\_WPM\_x).

Transakce USS4\_RPM\_x je ukončena, když pohon MICROMASTER potvrdí příjem příkazu nebo když je ohlášena chyba. Programový cyklus se nadále provádí i v době, kdy tento proces čeká na odezvu.



Obr. 30. Volání instrukce USS4\_RPM\_R

Aby byl povolen přenos požadavku, musí být bit EN zapnutý a musí zůstat zapnutý do nastavení bitu DONE, který signalizuje ukončení procesu. Například požadavek USS4\_RPM\_x je vysílán pohonu MICROMASTER v každém programovém cyklu, kdy je zapnutý vstup XMT\_REQ. Proto by měl být vstup XMT\_REQ spouštěn pulzem od náběžné hrany, aby byl požadavek vyslaný pouze jednou v každém kladném přechodu vstupu EN.

Vstup DRIVE je adresa pohonu MICROMASTER, na kterou má být poslán příkaz USS4\_RPM. Platné adresy jednotlivých pohonů jsou 0 až 31.

PARM je číslo parametru. INDEX je indexová hodnota parametru, který má být načten.

Adresa 16bytového zásobníku musí být zadána na vstup DB\_PTR. Tento zásobník používá instrukce USS4\_RPM\_x na uložení výsledku příkazu daného pohonu MICROMASTER.

Po ukončení instrukce USS4\_RPM\_x se zapne výstup DONE a výstupní byte ERR obsahuje výsledky provedení instrukce. Tabulka 16. definuje chyby, které mohou vyplynout z provedení této instrukce. Výstupy ERR a VAL nejsou platné, dokud se nezapne výstup DONE. Výstup VAL obsahuje výsledek provedení instrukce.

Vstupy/Výstupy	Operandy	Typy dat
XMT_REQ	I, Q, M, S, SM, T, C, V, L, Power Flow conditioned by a rising edge detection element.	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, Constant, *VD, *AC, *LD	BYTE
PARAM	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC *VD, *AC, *LD	WORD
INDEX	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC *VD, *AC, *LD	WORD
DB_PTR	&VB	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
VAL	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL

Tabulka 14. Operandy platné pro USS4\_RPM\_R

Ukázka volání instrukce USS4\_RPM\_R:

```
LD M5.1
= L60.0
MOVW 22, Param:VW202
MOVW 0, Index:VW204
LD M5.1
EU
= L63.7
LD L60.0
CALL USS4_RPM_R:SBR54, L63.7, 0, Param:VW202, Index:VW204, &VB1100, V51.0,
VB60, VD306
```

Popis:

- M5.1 = bit pro povolení čtení otáček motoru
- Param = 22 číslo parametru otáček motoru
- Index = 0 index parametru pro čtení

\*\*\*\*\*

0 = DRIVE číslo pohonu ze kterého chceme číst

&VB1100 = DB\_PTR ukazatel na zásobník

V51.0 = DONE po ukončení instrukce se zapne

VB60 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

VD306 = VAL obsahuje výsledek provedení instrukce



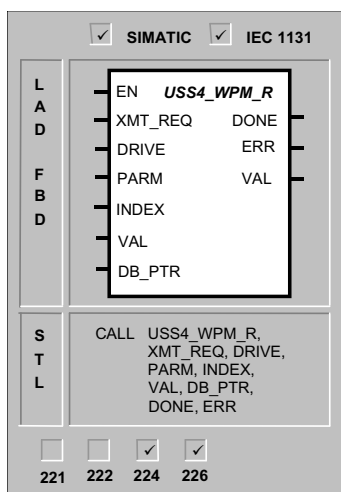
## 8.6 Instrukce USS4\_WPM\_x

V protokolu USS4 existují tři instrukce pro zápis:

- Instrukce USS4\_WPM\_W zapisuje parametr word bez znaménka.
- Instrukce USS4\_WPM\_D zapisuje parametr double word bez znaménka.
- Instrukce USS4\_WPM\_R zapisuje parametr s plovoucí desetinnou čárkou.

V jednu dobu může být aktivní pouze jedna instrukce pro čtení (USS4\_RPM\_x) nebo zápis (USS4\_WPM\_x).

Transakce USS4\_WPM\_x je ukončena, když pohon MICROMASTER potvrdí příjem příkazu nebo když je ohlášena chyba. Programový cyklus se nadále provádí i v době, kdy tento proces čeká na odezvu.



Obr. 31. Volání instrukce USS4\_WPM\_R

Aby byl povolen přenos požadavku, musí být bit EN zapnutý a musí zůstat zapnutý do nastavení bitu DONE, který signalizuje ukončení procesu. Například požadavek USS4\_WPM\_x je vysílán pohonu MICROMASTER v každém programovém cyklu, kdy je zapnutý vstup XMT\_REQ. Proto by měl být vstup XMT\_REQ spouštěn pulzem od náběžné hrany, aby byl požadavek vyslán pouze jednou v každém kladném přechodu vstupu EN.

Vstup DRIVE je adresa pohonu MICROMASTER, na kterou má být poslán příkaz USS4\_RPM. Platné adresy jednotlivých pohonů jsou 0 až 31.

PARM je číslo parametru. INDEX je indexová hodnota parametru, který má být zapsán.

VAL je hodnota parametru, která má být zapsána do RAM v pohonu.

Adresa 16bytového zásobníku musí být zadána na vstup DB\_PTR. Tento zásobník používá instrukce USS4\_WPM\_x na uložení výsledku příkazu daného pohonu MICROMASTER.

Po ukončení instrukce USS4\_WPM\_x se zapne výstup DONE a výstupní byte ERR obsahuje výsledky provedení instrukce. Tabulka 16. definuje chyby, které mohou vyplynout z provedení této instrukce.

Vstupy/Výstupy	Operandy	Typ dat
XMT_REQ	I, Q, M, S, SM, T, C, V, L, Power Flow conditioned by a rising edge detection element.	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, Constant, *VD, *AC, *LD	BYTE
PARAM	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC *VD, *AC, *LD	WORD
INDEX	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC *VD, *AC, *LD	WORD
VAL	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD, Constant	REAL
DB_PTR	&VB	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

Tabulka 15. Operandy platné pro USS4\_WPM\_R

Ukázka volání instrukce USS4\_WPM\_R:

```
LD M8.1
= L60.0
MOVW +1120, Param:VW202
MOVW +0, Index:VW204
MOVR 0.2, VD320
LD M8.1
EU
= L63.7
LD L60.0
CALL USS4_WPM_R:SBR57, L63.7, 0, Param:VW202, Index:VW204, VD320,
&VB1250, V51.5, VB65
```

Popis:

- M8.1 = bit pro povolení zápisu doby rozběhu
- Param = 1120 číslo parametru doby rozběhu
- Index = 0 index parametru pro zápis

\*\*\*\*\*

0 = DRIVE číslo pohonu na který chceme zapisovat

VD320 = hodnota kterou chceme zapsat 0.2s

&VB1250 = DB\_PTR ukazatel na zásobník

V51.5 = DONE po ukončení instrukce se zapne

VB65 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

Chybové kódy	Popis
0	Bez chyby
1	Pohon neodpověděl
2	Byla zjištěna chyba kontrolního součtu v odpovědi pohonu
3	Byla zjištěna chyba parity v odpovědi pohonu
4	Chyba způsobená zásahem uživatelského programu
5	Pokus o nepřípustný příkaz
6	Zadána nepřípustná adresa pohonu
7	Komunikační port nebyl nastaven pro USS protokol
8	Komunikační port je zaneprázdněn zpracováním instrukce
9	Vstupní rychlost pohonu je mimo rozsah
10	Nesprávná délka odpovědi pohonu
11	Nesprávný první znak v odpovědi pohonu
12	Znak délky v odpovědi pohonu není podporován instrukcemi USS
13	Odpověděl nesprávný pohon
14	Zadaná adresa DB_PTR je nesprávná
15	Zadané číslo parametru je nesprávné
16	Byl vybrán neplatný protokol
17	USS je aktivní; změna není povolena
18	Byla specifikována neplatná přenosová rychlost
19	Nekomunikuje se: pohon není AKTIVNÍ
20	Parametr nebo hodnota v odpovědi pohonu jsou nesprávné nebo obsahují chybový kód
21	Namísto požadované hodnoty word byla vrácena hodnota double word
22	Namísto požadované hodnoty double word byla vrácena hodnota word

Tabulka 16. Chybové kódy při běhu programu

## 9. NÁVRH UŽIVATELSKÉHO OVLÁDÁNÍ

Pro ovládání řízení automatem jsem musel vymyslet, také jak nejlépe z uživatelského hlediska ovládat celé zařízení.

Jelikož má uživatel obsluhovat řízení, jako nejvýhodnější řešení se ukázalo použít operátorský terminál fy. Mitsubishi Electric z řady GOT1000, typ GT1155-QSBD, který mi opět poskytlo pracoviště ÚAI.

### 9.1 Operátorský terminál GT1155-QSBD

Jedná se o terminál, který je vybaven knihovnou která umožňuje komunikaci s mnoha jinými výrobci programovatelných automatů, mimo jiné i s automaty Siemens řad S7-200 a S7-300/400. Na přední straně je mini USB port, kterým se provádí nahrávání uživatelského programu. Na zadní straně jsou k dispozici 2 komunikační porty RS-232 a RS-422.

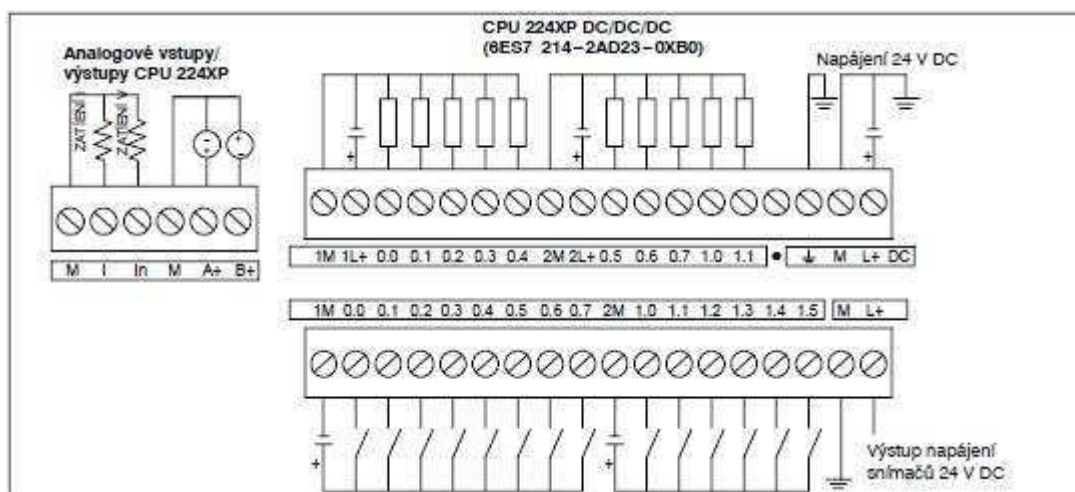
Tento terminál má velikost obrazovky 14,5 cm, displej je dotykový typu STN s 256 barvami o rozlišení 320 x 240 (QVGA). Napájení DC24V.



Obr. 32. GOT1155-QSBD

## 9.2 Výměna programovatelného automatu

Jelikož jsem se rozhodl pro použití USS protokolu a operátorského terminálu, vyžádalo si toto řešení nutnou výměnu programovatelného automatu za typ S7-200 CPU 224xp, který má 2 komunikační porty RS-485. Proto aby bylo možné USS protokol připojit na port0 a operátorský terminál připojit na port1 daného automatu.



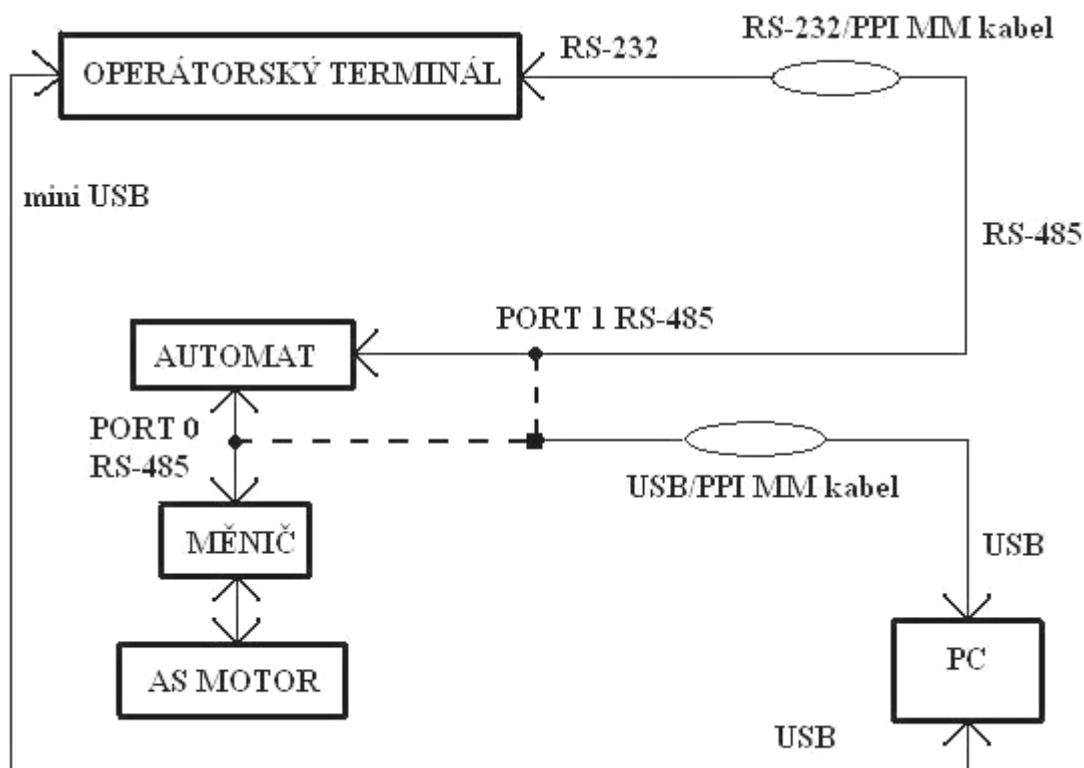
Obr. 33. Schéma zapojení CPU 224xp



Obr. 34. S7-200 CPU 224xp

## 10. ZAPOJENÍ A NASTAVENÍ

### 10.1 Blokové schéma řízení



Obr. 35. Blokové schéma řízení

### 10.2 Napájení

#### 10.2.1 Zapojení napájení měniče a motoru

Daný frekvenční měnič MICROMASTER 440 má jednofázové napájecí napětí 230V o výkonu pro motory do 0,25 kW. Údaje o motoru viz. jeho štítek.

Síťové napětí jsem připojil třížilovým kabelem na silové svorky L/L1, N/L2 a na zemní svorku PE. Viz. obr. 3.

Zapojení kabelu:

L/L1	- fáze	- modrý kabel
N/L2	- nulák	- hnědý kabel
PE	- zem	- zelenožlutý kabel

Vynutí motoru je napájeno 3 x 230 V, proto jsem vynutí na motoru zapojil do  $\Delta$ . Viz. obr. 4. Kabel jsem použil čtyřžilový, jeho zapojení je:


U	- šedý kabel
V	- černý kabel
W	- hnědý kabel

### 10.2.2 Napájení automatu a operátorského terminálu

Pro napájení automatu a operátorského terminálu, jsem použil zdroj stejnosměrného napětí DC 24V.

### 10.3 Zapojení komunikace mezi měničem a automatem pro USS4 protokol

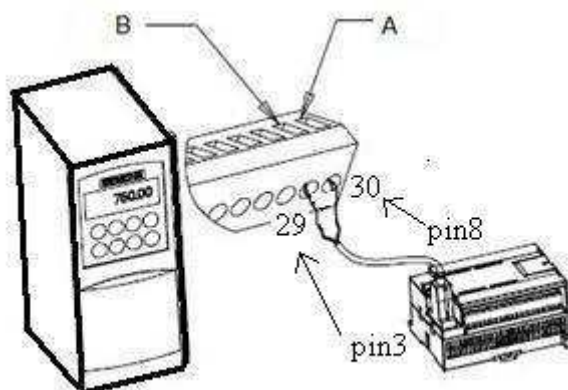
Připojení jsem realizoval pomocí standardního sériového kabelu. Na straně automatu je konektor D-SUB Canon 9pin typu samec, který se zapojí do portu 0 (USS protokol tento port potřebuje pro svoji činnost) na automatu. Zapojení vývodů portů na straně automatu je v tab. 17 .

Konektor	Číslo vývodu	PROFIBUS Signal	Port 0/Port 1
	1	Stínění	Uzemnění rozvaděče
	2	24 V zpětný	Logický společný
	3	RS-485 signál B	RS-485 signál B
	4	Požadavek na vyslání	RTS (TTL)
	5	5 V zpětný	Logický společný
	6	+5 V	+5 V, 100 Ω sériový odpor
	7	+24 V	+24 V
	8	RS-485 signál A	RS-485 signál A
	9	Nepoužitý	10bitová volba protokolu (vstup)
	Pouzdro konektoru	Stínění	Uzemnění rozvaděče

Tab. 17. Rozmístění vývodů v komunikačním portu automatu S7-200

Z portu 0 na automatu jsem vyvedl dva sériové vodiče k měniči. Jedná se o pin3 (RS-485 signál B) a pin8 (RS-485 signál A). Na řídicí svorkovnici viz. obr. 5. měniče jsou svorky 29 a 30 pro sériovou komunikaci. Celé zapojení je znázorněno na obr. 36.

- pin3 - svorka 29 - červený kabel
- pin8 - svorka30 - zelený kabel



Obr. 36. Připojení k řídicí svorkovnici MICROMASTER M440

#### 10.4 Nastavení USS4 protokolu na měniči

Aby byl měnič řízen USS protokolem je třeba na něm nastavit několik parametrů. Pro nastavení parametrů použijte operátorský panel BOP na měniči:

1. Resetujte pohon na tovární nastavení: P0010 = 30  
P0970 = 1

Pokud tento krok přeskočíte, zajistěte, aby byly následující parametry nastaveny na tyto hodnoty:

- Délka USS PZD: P2012 Index 0 = 2  
Délka USS PKW: P2013 Index 0 = 127

2. Povolte čtení/zápis všech parametrů (režim "Expert"): P0003 = 3

3. Zkontrolujte nastavení motoru vašeho pohonu:

- P0304 = Jmenovité napětí motoru (V) = 230  
P0305 = Jmenovitý proud motoru (A) = 0,9  
P0307 = Jmenovitý výkon motoru (kW) = 0,15  
P0308 = Účinník motoru  $\cos \varphi$  (-) = 0,85  
P0310 = Jmenovitá frekvence motoru (Hz) = 50  
P0311 = Jmenovitá rychlost motoru (RPM) = 2820

Nastavení se budou měnit podle použitého motoru. Abyste mohli nastavovat parametry P304, P305, P307, P308, P310 a P311, musíte nejprve nastavit parametr P010 na 1 (režim rychlého uvedení do provozu). Pokud jste dokončili nastavování parametrů, nastavte parametr P010 na 0. Parametry P304, P305, P307, P310 a P311 mohou být měněny pouze v režimu rychlého uvedení do provozu.

4. Nastavte režim řízení: P0700 Index 0 = 5

5. Výběr zdroje žádané hodnoty: P1000 Index 0 = 5

6. Doba rozběhu (volitelné): P1120 = 0 až 650,00 = 10 s

Je to doba v sekundách, kterou motor potřebuje na to, aby zrychlil na maximální frekvenci.

7. Doba doběhu (volitelné): P1121 = 0 až 650,00 = 0,2 s

Je to doba v sekundách, kterou motor potřebuje na to, aby zpomalil až do úplného zastavení.

8. Nastavte referenční frekvenci sériové linky: P2000 = 1 až 650 Hz = 50 Hz

9. Nastavte normalizaci USS: P2009 Index 0 = 0



10. Nastavte přenosovou rychlost sériového rozhraní RS-485:

P2010 Index 0 = 4 (2400 baud) = 6  
 5 (4800 baud)  
**6 (9600 baud)**  
 7 (19200 baud)  
 8 (38400 baud)  
 9 (57600 baud)  
 12 (115200 baud)

11. Vložte adresu zařízení typu slave: P2011 Index 0 = 0 až 31

Každý pohon (maximálně 31) může být řízen přes sběrnici.

12. Nastavte časovou prodlevu sériové linky: P2014 Index 0 = 0 až 65 535 ms = 0 s

(0 = časová prodleva blokována) Je to maximální povolený interval mezi dvěma příchozími datovými zprávami. Tato funkce se používá pro vypnutí invertoru v případě poruchy komunikace. Odpočítávání času začne od příjmu platné datové zprávy. Pokud ve specifikovaném časovém intervalu není přijata další datová zpráva, invertor se vypne a zobrazí se chybový kód F0070. Nastavení hodnoty na nulu vypne řízení.

13. Přesuňte data z RAM do EEPROM:

P0971 = 1 (Začátek přesunu) Uložte změny nastavení parametrů do EEPROM.

## 10.5 Zapojení a nastavení komunikace mezi automatem a operátorským terminálem

Pro komunikaci mezi operátorským panelem GT1155-QSBD a programovatelného automatu Simatic S7-200, jsem použil kabel RS-232/PPI Multi-Master, který se používá pro spojení PC s automatem. Stranu označenou pro PC zapojíme do portu RS-232 na terminálu a druhou PPI zapojíme na port 1 automatu. Přenosovou rychlost kabelu nastavíme na 9600 bps, pomocí pinů na přepínači.

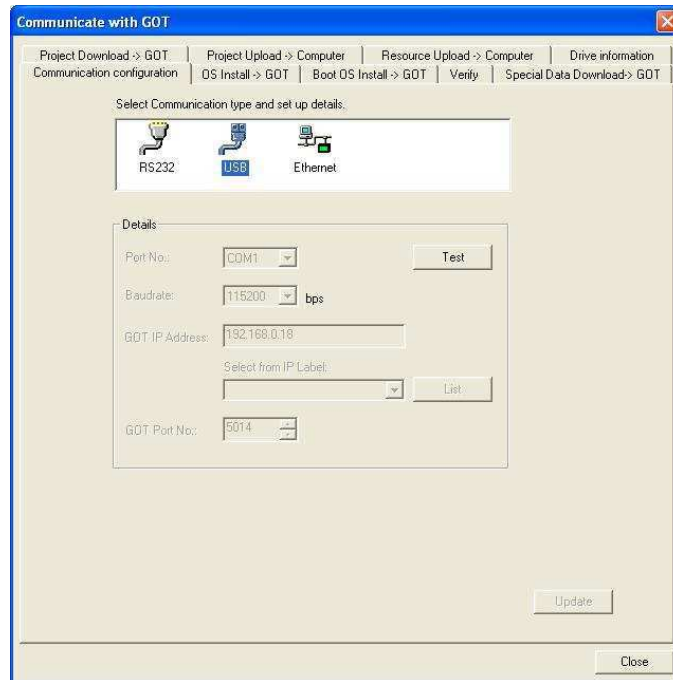
### 10.5.1 Instalace OS a knihovny pro S7-200

Na začátku mé práce byl operátorský terminál bez operačního systému, který je potřeba do něj nahrát spolu s komunikační knihovnou pro automat S7-200.

Prvním krokem je zajištění vůbec komunikace terminálu s PC, se kterým jsem ho spojil přes mini USB s USB portem PC. Nyní musíme nastavit komunikaci přes USB a poté můžeme teprve nahrát OS spolu s komunikační knihovnou.

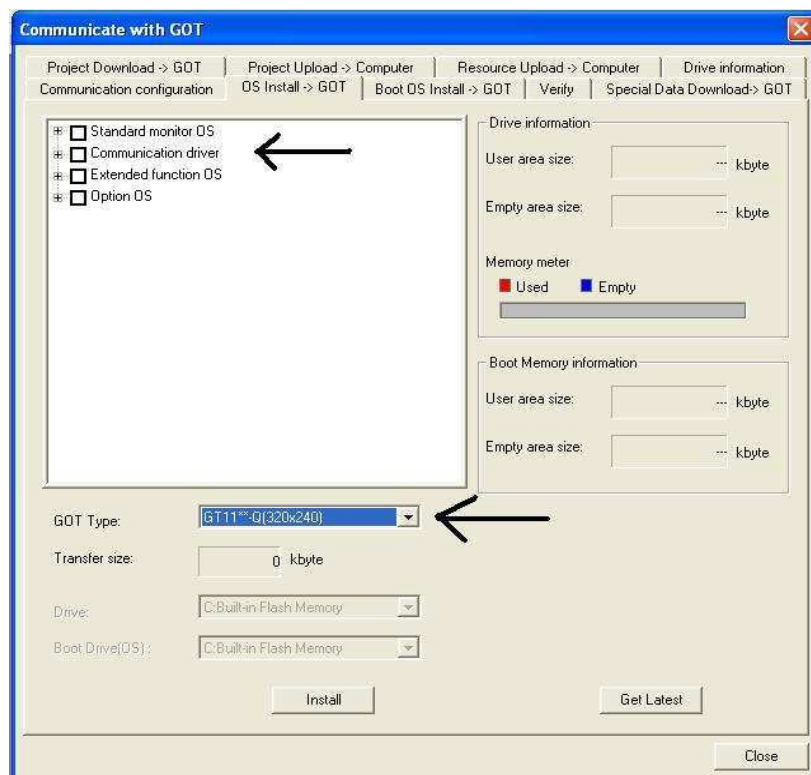
Postup:

- zapojíme USB kabel mezi terminálem a PC
- spustíme program GT Designer2
- nyní se nás to zeptá, jestli chceme založit nový projekt či otevřít, tohle okénko nyní ignorujem a zavřeme křížkem
- najedeme na záložku Communication a zvolíme To/From GOT a vybereme pro komunikaci GOT-1000
- v okně se záložkami, které se nám objeví zvolíme záložku Communication configuration viz. obr. 37., a zvolíme USB.



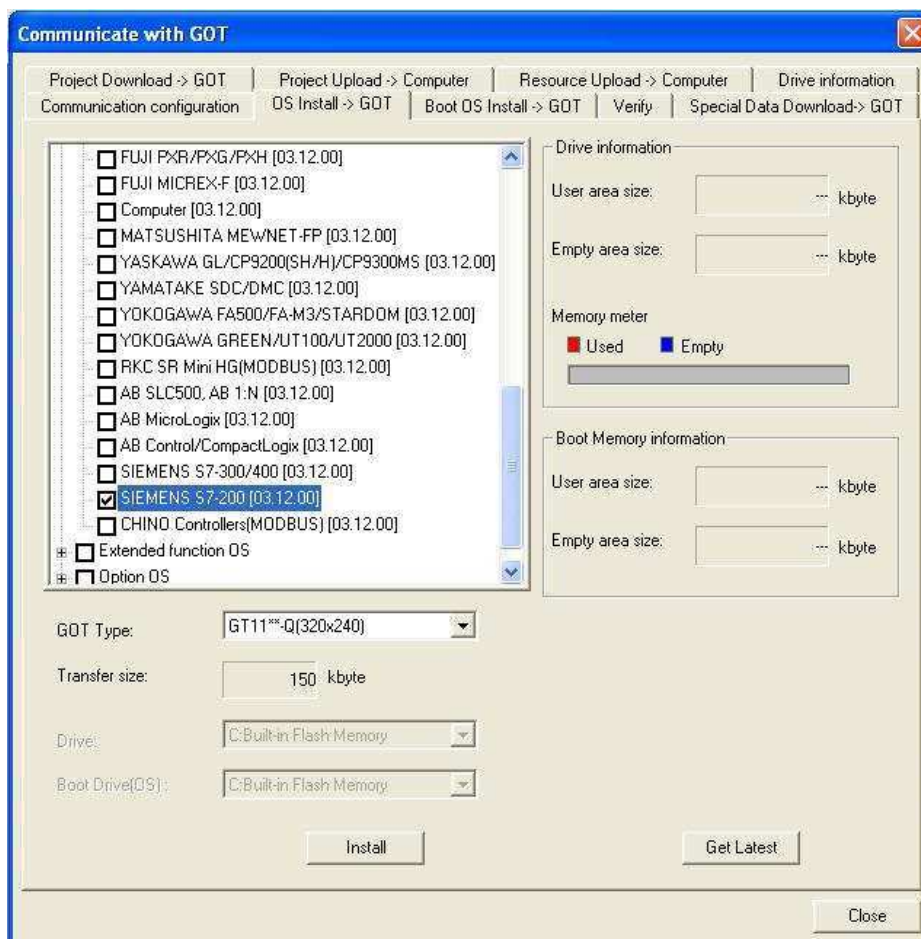
Obr. 37. Konfigurace USB portu

- dáme otestovat komunikaci tlačítkem Test, za předpokladu že je zapojen USB kabel a zapnuto napájení terminálu proběhne test bez problémů
- nyní můžeme přistoupit k samotné instalaci OS
- zvolíme záložku OS Instal -> GOT, viz obr. 38.



Obr. 38. Instalace OS

- v ní rozklikneme roler s GOT type a vybereme náš GT11\*\*-Q(320x240)
- a dále rozklikneme v levé horním okénku Communication driver, v něm zaškrtneme náš automat S7-200, viz obr. 39.
- v tuto chvíli nám již nic nebrání k nahrání OS, zvolíme tedy Install
- terminál se restartuje a sám si nahraje, to co jsme si zvolily
- nyní máme vše potřebné pro další práci nastaveno



Obr. 39. Výběr S7-200

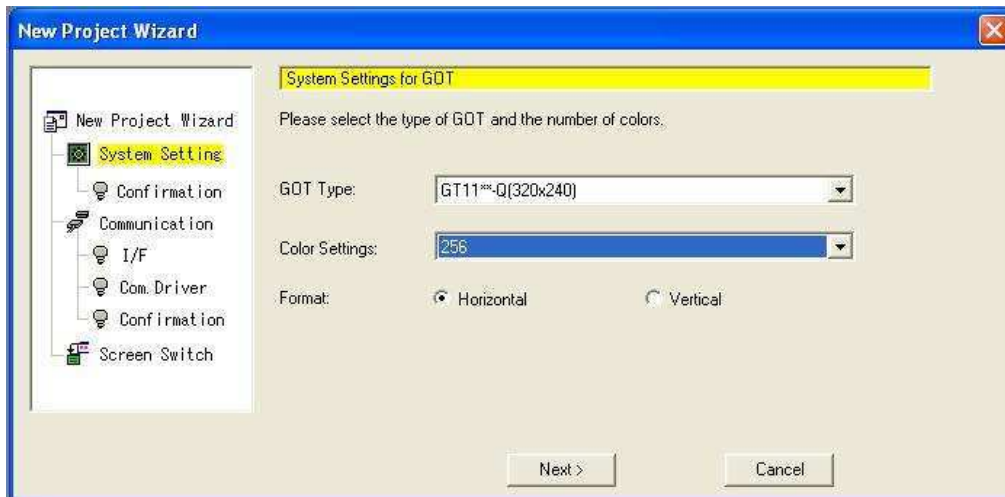
### 10.5.2 Nastavení komunikace mezi automatem a operátorským terminálem

K tomu abychom mohli nakonfigurovat komunikaci si prvně založíme nový projekt, ve kterém vše potřebné nastavíme a potom ho použijeme pro samotné programování.

Postup je následující:

- spustíme program GT Designer 2
- zvolíme založení nového projektu tlačítkem New
- následující okénko odklikneme tlačítkem Next, zde nic nenastavujeme
- v následující okénku musíme zvolit náš typ GT11\*\*-Q(320x240), viz obr. 40. a dáme dvakrát Next
- v dalším okénku musíme vybrat typ automatu, se kterým má terminál komunikovat, v našem případě je to S7-200, viz. obr. 41. a dáme Next

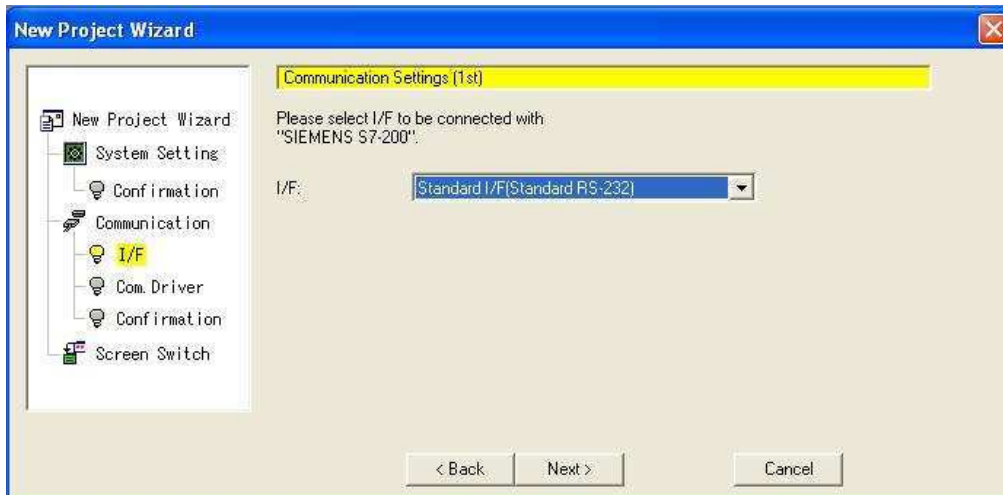
- v tomto okénku musíme nastavit komunikační kabel RS- 232, viz. obr. 42. a dáme Next
- v dalším okénku viz. obr. 43. se již nastavuje u zvoleného automatu, přímo komunikace tlačítkem Details, kde pouze změníme přenosovou rychlost na 9600 bps a poté klikneme na OK a třikrát Next, pak tlačítko Finish
- potom již naskočí okénko, ve kterém si zvolíme název první obrazovky, např. Menu a dáme OK, viz. obr. 44.
- tímto je nastavení hotovo a můžeme se pustit do programování



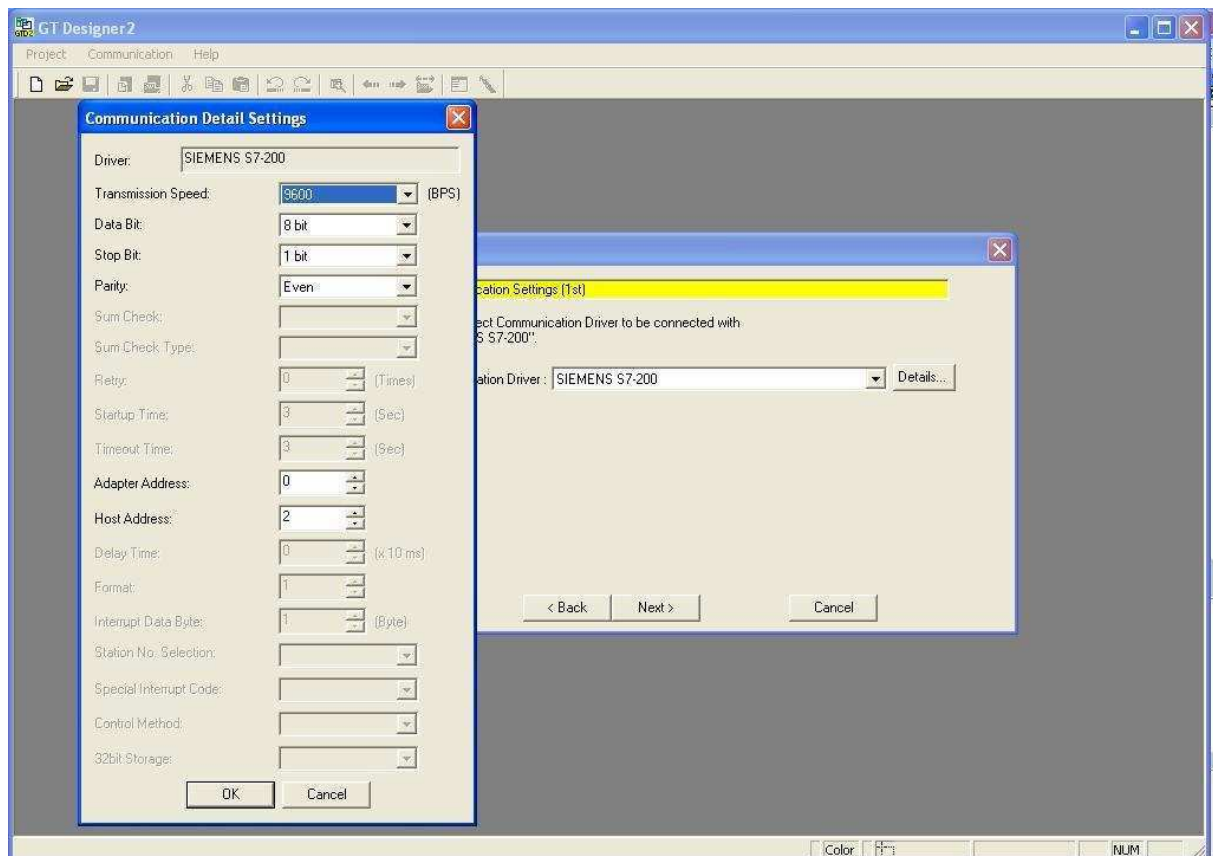
Obr. 40. Výběr typu terminálu



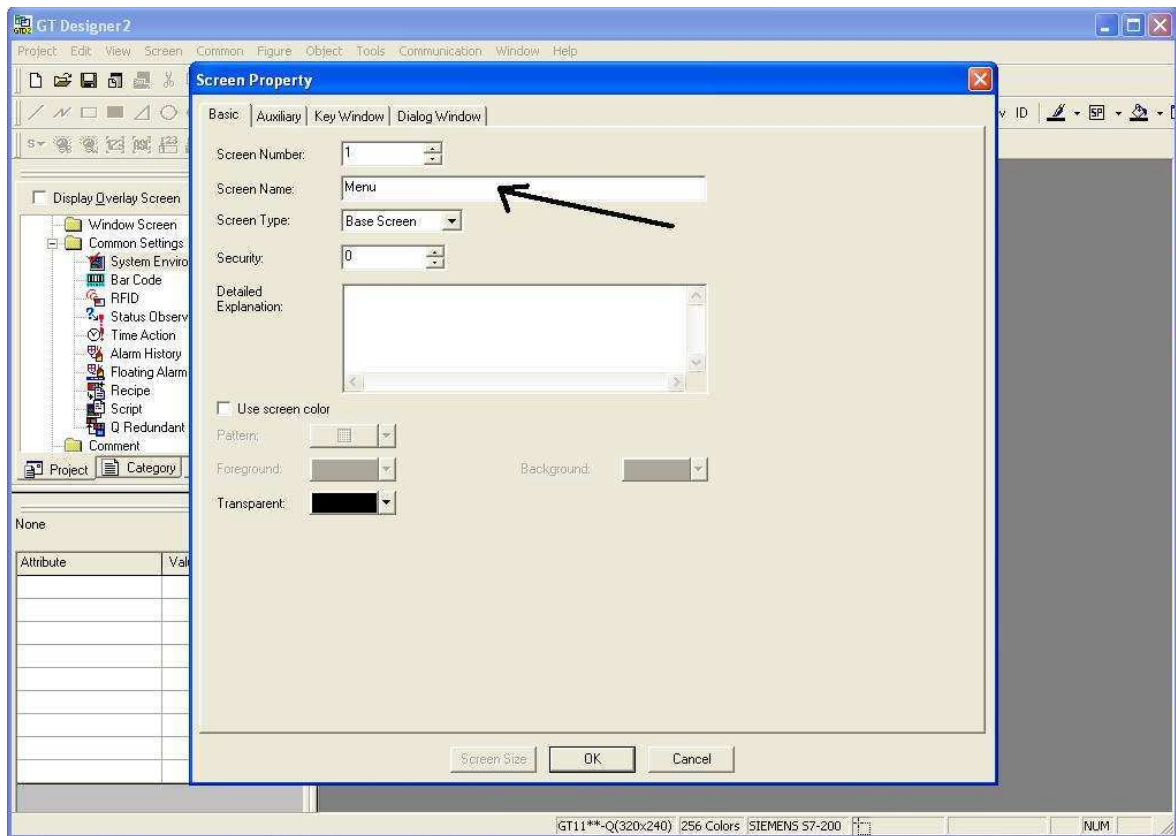
Obr. 41. Výběr automatu S7-200



Obr. 42. Výběr komunikačního kabelu



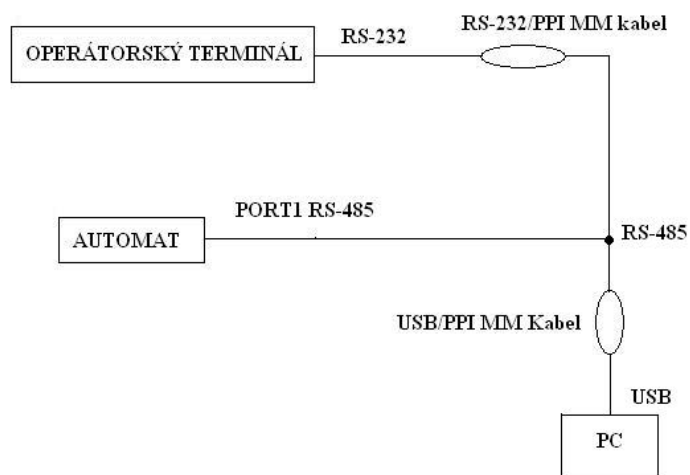
Obr. 43. Nastavení komunikace s S7-200



Obr. 44. Volba názvu obrazovky

### 10.6 Zapojení a nastavení komunikace mezi automatem a PC

Nastavení komunikace se provádí v programu pro automat STEP 7-Micro/WIN. Pro komunikaci PC s automatem, jsem použil USB/PPI Multi-Master kabel. Jelikož mám již obsazeny oba porty automatu, rozhodl jsem se sdílet port 1 spolu s operátorským terminálem, viz. obr. 45.



Obr. 45. Zapojení sdílení postu1 na automatu

Toto zapojení jsem navrhl proto, abych mohl pomocí PC programem STEP 7 kontrolovat chod programu v automatu (kontrola hodnot zadaných z terminálu, vykonání požadovaných instrukcí v automatu, chod USS protokolu).

Bohužel tento návrh sice vypadá hezky, ale ve skutečnosti není toto zapojení reálné, tudíž nemůže nikdy fungovat, jelikož tam jsou použity dva PPI Multi-Master kabely. Problém je v tom, že není možné napájet oba PPI kabely současně. Tímto jsem byl při kontrolách chodu programu v automatu omezen:

- bylo možné buď sledovat komunikaci s terminálem, kdy jsem PPI kabel zapojil na port 0 automatu
- nebo jsem ho připojil na port 1 a kontroloval chod USS protokolu

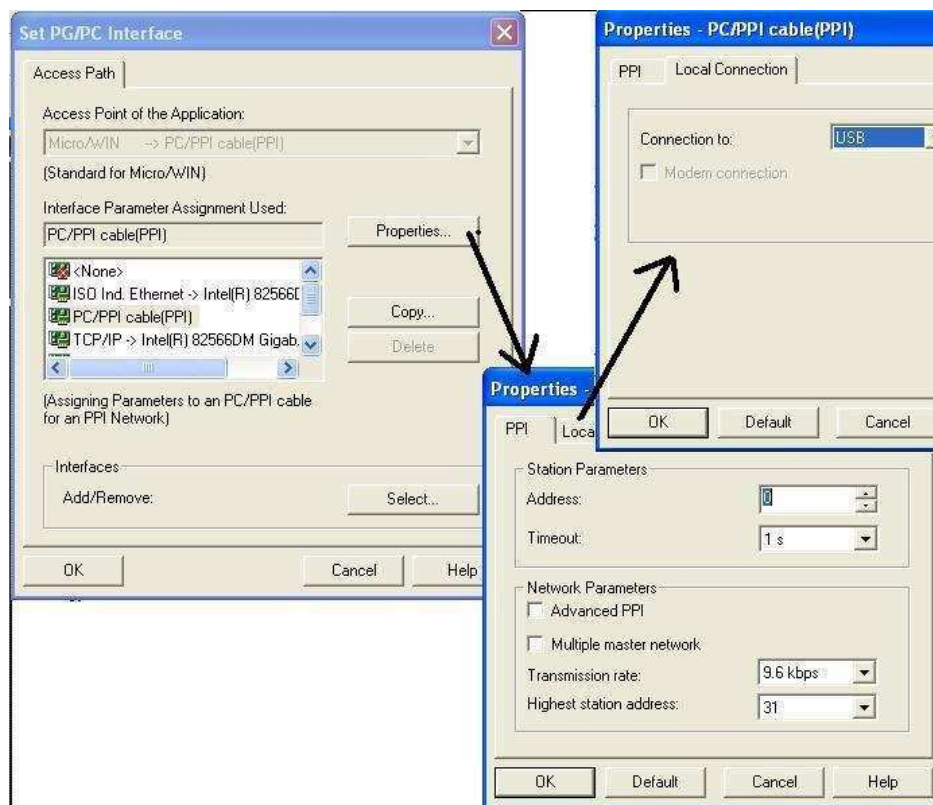
Proto, aby to fungovalo podle navrženého obrázku, musel by být operátorský terminál vybavený portem RS-485 (např. operátorské terminály od fy. Siemens) a tím by odpadl kabel RS-232/PPI. A zůstal by jen jeden kabel USB/PPI, pak už by vše fungovalo tak jak by mělo.

### 10.6.1 PPI Multi-Master kabel

Musíme nastavit kabel USB/PPI Multi-Master.

Postup nastavení:

- spustíme program STEP7
- v navigační liště zvolíme ikonu Nastavit rozhraní PG/PC a klikneme na tlačítko Properties (vlastnosti)
- v záložce PPI nastavíme rychlost 9600 bps a adresu kabelu na 0
- v záložce Local Connection vybereme požadované USB
- viz. obr. 46.



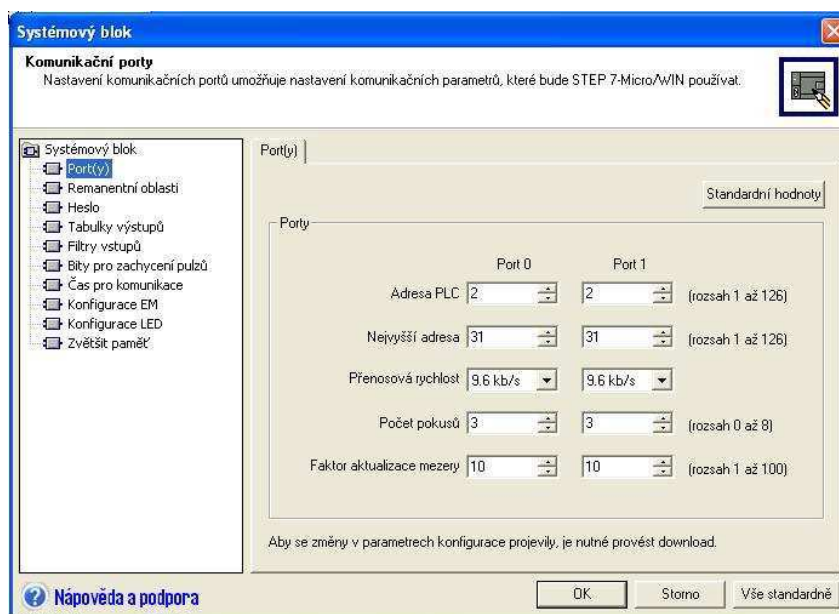
Obr. 46. Nastavení PPI Multi-Master kabelu

### 10.6.2 Nastavení přenosové rychlosti a síťové adresy pro S7-200

Přenosovou rychlost a síťovou adresu je nutné nastavit také pro S7-200. Rychlost i adresa jsou uloženy v systémovém bloku S7-200. Po nastavení parametrů pro S7-200 musíte systémový blok do S7-200 downloadovat.

Postup nastavení:

- spustíme program STEP7
- v navigační liště zvolíme ikonu systémový blok
- zde nastavíme adresy PLC pro oba porty na 2
- přenosové rychlosti nastavíme obě 9600 bps
- viz. obr. 47.



Obr. 47. Systémový blok

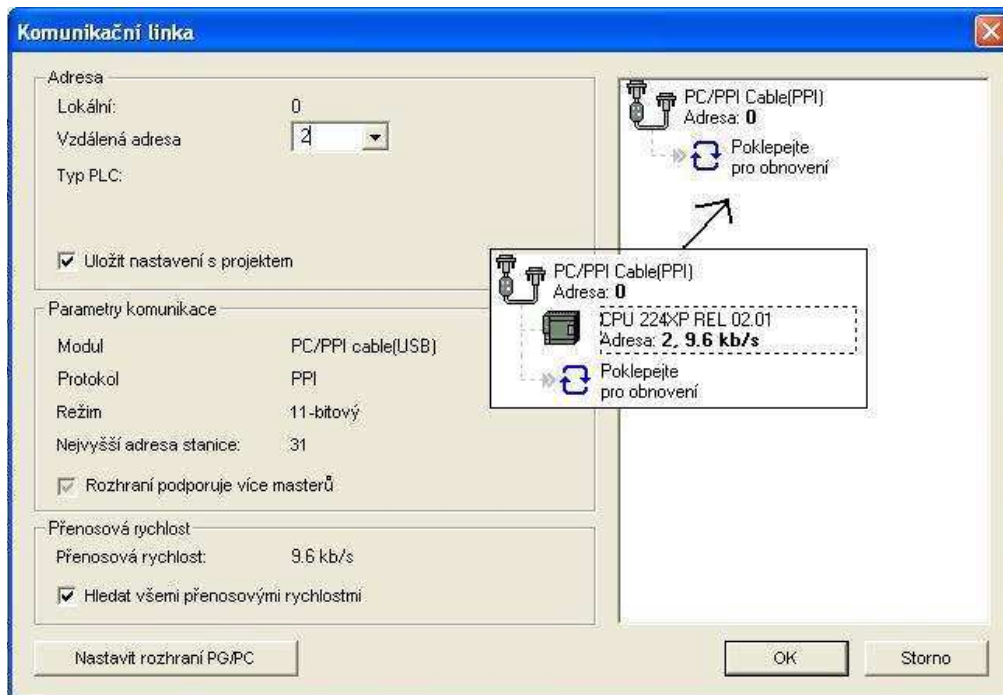
### 10.6.3 Vyhledávání S7-200 v síti

Automat S7-200, který je k síti připojen, je možné vyhledat a identifikovat. Při vyhledávání je možné prohledávat síť určenou buď přenosovou rychlostí, nebo všemi přenosovými rychlostmi.

Postup vyhledání:

- spustíme STEP 7
- v navigační liště zvolíme ikonu Komunikace a dvakrát klikněte na obnovit
- po prohledání najde STEP 7 automat v síti, tímto je nastavení komunikace hotové, potvrdíme tlačítkem OK
- viz. obr. 48.





Obr. 48. Vyhledání S7-200 v síti

## 11. SOFTWARE STEP 7-MICRO/WIN V4.0

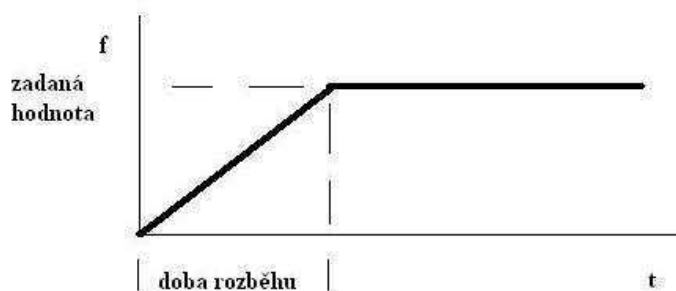
### 11.1 Program ve STEP 7- Micro/WIN V4.0

Chod celého zařízení je řízen pomocí programovatelného automatu Simatic S7-200 CPU224xp. V prostředí STEP 7 jsem vytvořil řídicí program postavený na koncepci použitého USS protokolu. Celý program jsem napsal v editoru výpisu příkazů STL.

Vstupní data jsou zadávány přes operátorský terminál, ty zpracuje automat a na základě jejich vyhodnocení automat řídí frekvenční měnič. Na terminálu je možné volit mezi třemi způsoby řízení otáček. U všech třech způsobů se hodnota vstupních otáček zadává v procentech. Rozsah  $-100 \div 100\%$  ( $100\% = 50 \text{ Hz}$ ).

#### 1. ZADEJ OTÁČKY:

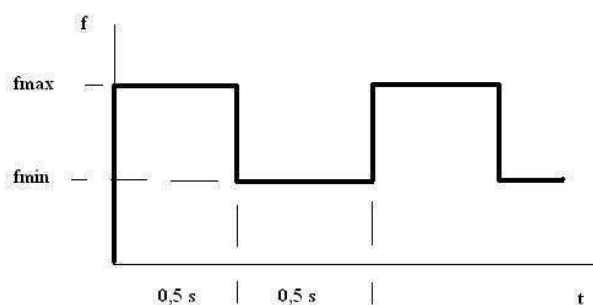
- zadáváme otáčky
- pohon můžeme zapnout/vypnout, změnit směr otáčení, pozvolný doběh motoru až do zastavení
- umožňuje servisní výpis hodnot výstupních otáček, proudu a výstupní frekvence
- výpis aktuální hodnoty otáček
- doba rozběhu na max. otáčky 10s
- doba doběhu pro zastavení 0,2s
- průběh otáček viz. obr. 49.



Obr. 49. Průběh otáček pro ZADEJ OTÁČKY

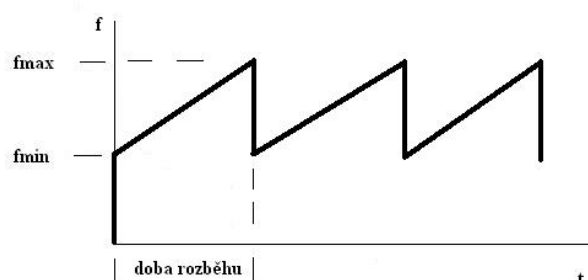
#### 2. ZADEJ PRUBĚH SKOKOVĚ

- zadáváme otáčky maximální a minimální
- pohon můžeme zapnout/vypnout
- otáčky se střídavě mění dle zadaných hodnot otáček
- výpis aktuální hodnoty otáček ?
- doba rozběhu na max. otáčky a doběhu na min. otáčky 0,2s
- průběh otáček viz. obr. 50.



Obr. 50. Průběh otáček pro ZADEJ PRUBĚH SKOKOVĚ

3. ZADEJ PRUBĚH S RAMPOU
- zadáváme otáčky maximální a minimální
  - pohon můžeme zapnout/vypnout
  - otáčky se střídavě mění dle zadaných hodnot otáček
  - výpis aktuální hodnoty otáček ?
  - doba rozběhu na max. otáčky 10s
  - doba doběhu na min. otáčky 0,2s
  - průběh otáček viz. obr. 51.



Obr. 51. Průběh otáček pro ZADEJ PRUBĚH S RAMPOU

V programu jsem použil pomocí USS protokolu instrukce na čtení z měniče, pro výpis hodnot výstupních otáček, proudu a výstupní frekvence, které se pomocí přerušení jednotlivě přepínají (protože v jedné době může být aktivní pouze jedna instrukce pro čtení nebo zápis). Také jsem použil dva podprogramy pro čtení těchto parametrů a pro zjišťování aktuální hodnoty otáček.

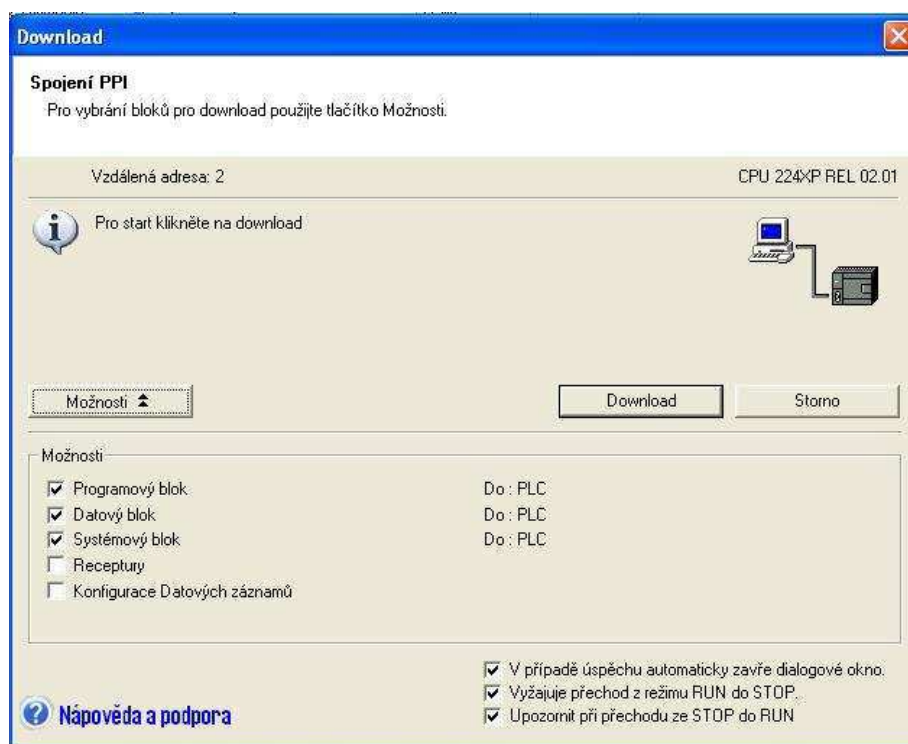
Využil jsem i instrukce pro zápis hodnot parametrů do měniče, konkrétně se jednalo o dobu rozběhu, kterou jsem volil buď 0,2s nebo 10s. Doba doběhu je pevně nastavena na 0,2s.

Podrobnější popis viz. příloha č. 1. Jelikož knihovna pro USS je velmi rozsáhlý a složitý program nebylo možné ho zobrazit v tištěné podobě a nejdeme ho v mém programu, který je na příloženém CD.

## 11.2 Download programu ze STEP 7 do automatu

Pro nahrání programu do automatu, klikneme na záložku Soubor ve které zvolíme Download. Objeví se okénko pro download viz. obr 52. , zvolíme si všechno co chceme nahrát, hlavně náš nastavený systémový blok viz. kap. 10.6.2 a program nahrajeme do automatu.

Po vykonání downloadu je automat připraven vykonávat nahraný program, stačí ho jen zapnout buď ve STEP7 nebo přímo na automatu páčkou do polohy RUN. Já měl po celou dobu programování páčku v pozici TERM, což umožňuje automat zapínat a vypínat pomocí STEP 7.



Obr. 52. Download projektu



## 12. Program v GT DESIGNER 2

Vzhledem k tomu, že v tomto prostředí se programuje plně graficky nebylo možné vložit program do přílohy, ten je obsažen pouze jako soubor na přiloženém CD. Proto se zde budu snažit popsat nejdůležitější poznatky a objekty při vytváření projektu.

Princip řízení automatu operátorský terminálem spočívá v tom, že terminál vidí na určitá místa (I,Q,M,V,SM atd.) v paměti automatu a může do nich zapisovat i číst z nich hodnoty. V mém projektu jsem využíval bitovou paměť M a paměť pro proměnné V.

V první řadě je potřeba založit nový projekt viz. kap. 10.5.2.

Ve svém projektu jsem používal 3 objekty:

- Multi Action Switch
- Bit Lamp
- Numerical Display a Numerical Input

### 12.1 Multi Action Switch

Tento multifunkční switch jsem využil u všech obsažených tlačítek.

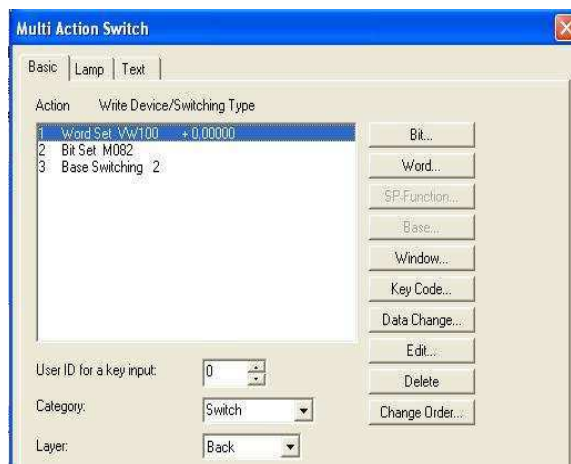
Umožňuje hlavně:

- nastavení bitu, byte, word i double word
- umožňuje přepínání obrazovek
- na jedno stisknutí tlačítka se umožňuje nastavit několik akcí, která mají být vykonána

Najdeme ho viz. obr. 53., ukázka použití viz obr. 54.



Obr. 53. Umístění Multi Action Switch



Obr. 54. Ukázka funkce MAS

Na obr. 54. je ukázka co se má provést po zmáčknutí Multi Action Switche:

1. nastaví adresu VD100 na hodnotu +0.0
2. nastaví bit M8.2 na hodnotu 1
3. přepne se na obrazovku 2

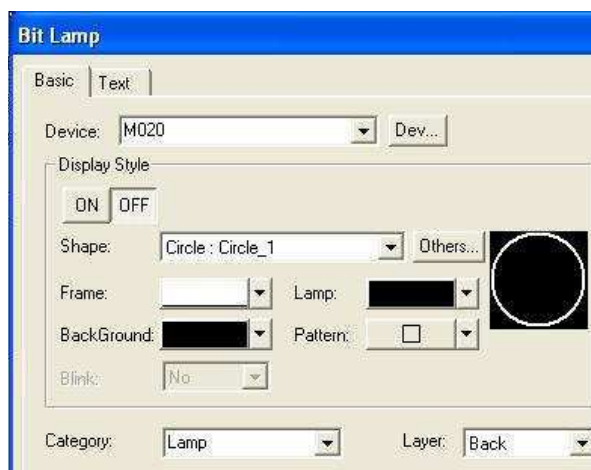
## 12.2 Bit Lamp

Jak již z názvu vyplývá jedná se o signalizaci stavu zadaného bitu. Je-li zadaný bit v log. 1, lampička se rozsvítí.

Najdeme ho viz. obr. 55., ukázka použití viz obr. 56.



Obr. 55. Umístění Bit Lamp



Obr. 56. Ukázka funkce BL

Na obr. 56. je ukázka stavení Bit Lampy na bit M20, je-li daný bit v log. 1 lampička se rozsvítí.

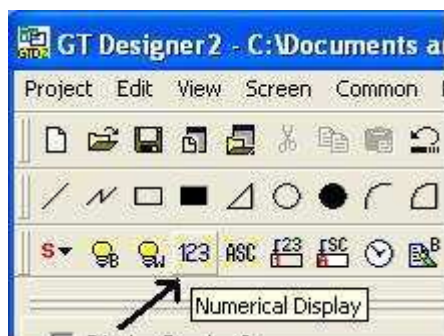
Pozn. :

jak u Multi Action Switch tak u Bit Lampy, může volit mezi několika typy vzorů vzhledů, barev a textů které mají být zobrazeny na displeji, záleží jen na nás jak si je zvolíme. Objekt se na obrazovku vloží tak že klikneme na požadovaný objekt a potom na místo kde má být na displeji umístěn. Na vlastnosti objektu kde se vše nastavuje se dostaneme dvojklikem na objekt.

## 12.3 Numerical Display a Numerical Input

Tento objekt může buď zadávat hodnoty do automatu (Numerical Input) nebo je z něj číst respektive zobrazovat (Numerical Display).

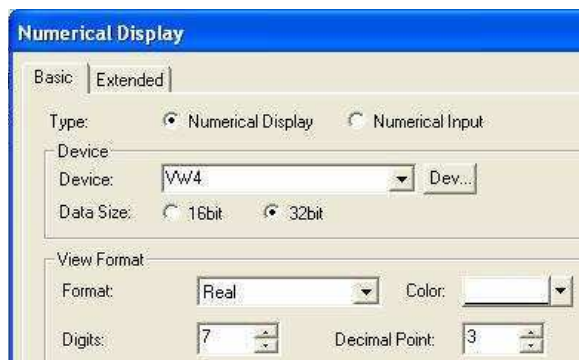
Najdeme ho viz. obr. 57, ukázka použití viz obr. 58. a obr. 59.



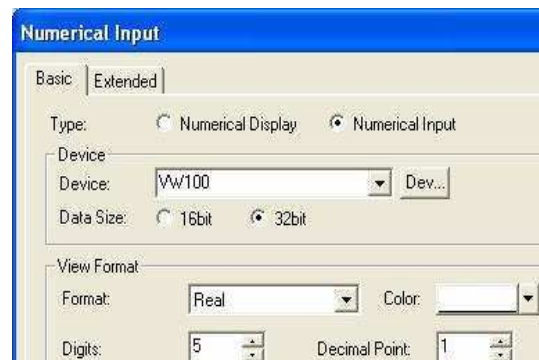
Obr. 57. Umístění Numerical Display

Pozn. :

To jestli se jedná o Numerical Display nebo Numerical Input, volíme až ve vlastnostech displeje. U Numerical Input (zadávaní hodnot) se po zmáčknutí tlačítka objeví klávesnice, kterou zadáváme požadovanou hodnotu pro zápis do automatu .



Obr. 58 Numerical Display



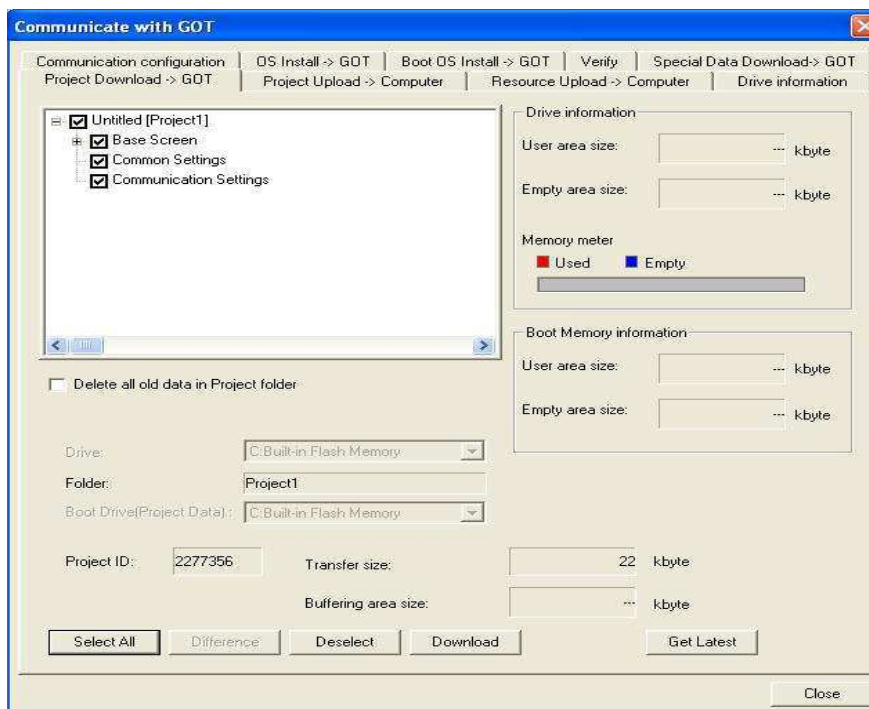
Obr. 59. Numerical Input

Na obr. 58. je ukázka pro zobrazení proměnné VD4 v automatu.

Na obr. 59. je ukázka pro zápis hodnoty do proměnné VD100 v automatu.

## 12.4 Download programu z GT Designer 2 do operátorského terminálu

Pro download zvolíme záložku Communication a v ní klikneme na To/From GOT, v zobrazeném okně zvolíme Project Download -> GOT. Zvolíme možnost Select All a poté již dáme download. Terminál se restartuje a naběhne již s nahaným projektem.



Obr. 60. Download projektu





## 13. POSTUP PRO LABORATORNÍ CVIČENÍ

Jelikož moje diplomová práce má sloužit studentům k ukázce možnosti řízení motoru měničem za pomoci programovatelného automatu a operátorského terminálu, popsal jsem v této kapitole postup jak s daným zařízením zacházet a ovládat ho.

### Ukázka řízení motoru frekvenčním měničem

#### 13.1 Úvod

Frekvenční měnič s asynchronním motorem spolu tvoří regulační pohon, který si v poslední době získal dominantní postavení v automatizovaném průmyslu. Frekvenční měniče jsou využívány pro nejrůznější způsoby řízení otáček asynchronních motorů.

#### 13.2 Cíl

Cílem této laboratorní úlohy je seznámit se a vyzkoušet si řízení otáček asynchronního motoru pomocí frekvenčního měniče fy. Siemens MICROMASTER 440, který je řízen programovatelným automatem fy. Siemens Simatic S7-200 CPU 224xp a vstupní hodnoty jsou zadávány přes dotykový operátorský terminál fy. Mitsubishi Electric GT1155-QSBD.

#### 13.3 Rozbor úlohy

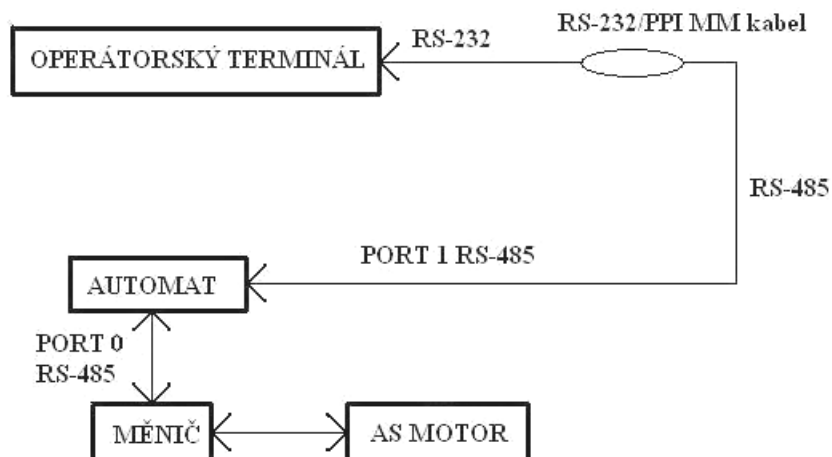
K dosažení širokého regulačního rozsahu musíme napájet asynchronní stroj, ze zdroje proměnného kmitočtu, umožňujícího současnou změnu výstupního napětí. K tomuto účelu slouží frekvenční měniče.

Celý řídicí program je postaven na koncepci USS protokolu (je to speciální komunikační knihovna, určená pro řízení frekvenčních měničů řady MICROMASTER pomocí programovatelných automatů). Pomocí instrukcí USS můžeme řídit fyzický pohon a číst, popř. zapisovat jeho parametry.

Veškerá komunikace probíhá po sériové lince RS-485 kombinovaná s RS-232.

Daný frekvenční měnič MICROMASTER 440 má jednofázové napájecí napětí 230V o výkonu pro motory do 0,25 kW.

### 13.4 Schéma zapojení

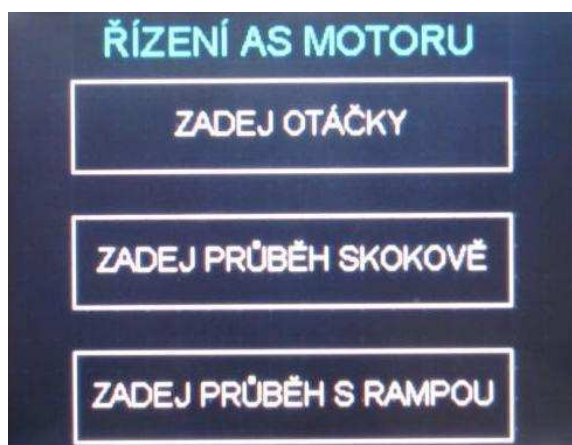


Obr. 1. Schéma zapojení

### 13.5 Pokyny pro cvičení

#### 13.5.1 Uvedení do provozu

1. Zapojte frekvenční měnič do zásuvky 230V.  
Zapněte zdroj stejnosměrného napětí DC24V pro automat a operátorský terminál.
2. Po naběhnutí všech tří zařízení, přepněte na automatu páčku do polohy RUN.
3. Nyní již máme pro uvedení do provozu vše potřebné nastaveno.
4. Na displeji terminálu se objeví 3 tlačítka, pro volby řízení viz. obr. 2.



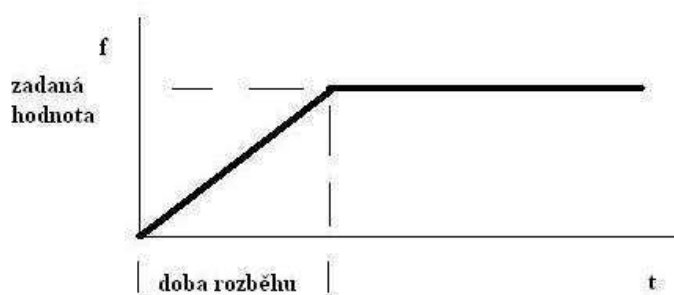
Obr. 2. Menu řízení

Pozn.: Pro všechny tři způsoby řízení se hodnota žádaných otáček zadává v % (100% odpovídá frekvenci 50Hz).

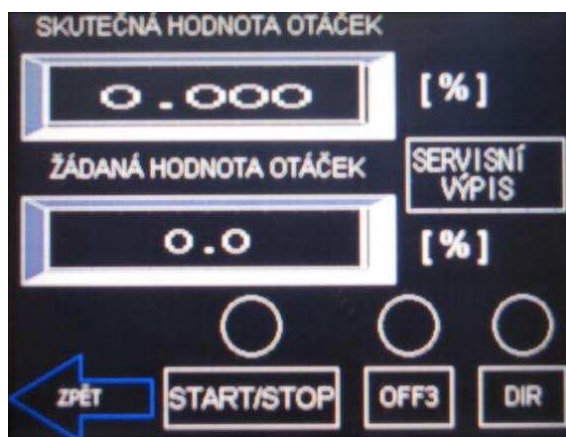
### 13.5.2 Menu ZADEJ OTÁČKY

Zmáčknutím toho tlačítka se dostaneme do menu, které umožňuje:

- zadávat hodnotu otáček a sledovat její skutečnou hodnotu
- zapínat/vypínat pohon, nechat motor volně doběhnout až do zastavení
- přepínat směr otáčení
- servisní výpis hodnot výstupních otáček, proudu a výstupní frekvence
- doba rozběhu na max. otáčky 10s
- doba doběhu pro zastavení 0,2s
- průběh otáček viz. obr. 3., vzhled menu viz. obr. 4.



Obr. 3. Průběh otáček pro ZADEJ OTÁČKY



Obr. 4. Menu ZADEJ OTÁČKY

Popis tlačítek:

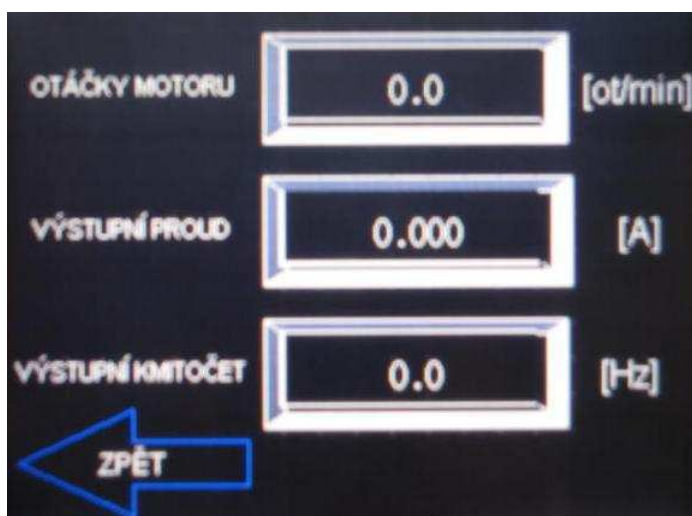
- **START/STOP** (zapnutí/vypnutí pohonu)  
Toto tlačítko je alternativní, tzn. jedním stisknutím se pohon zapne a druhým vypne. Kontrolní lampa (kolečko nad tlačítkem) signalizuje stav pohonu. Jestliže svítí je pohon zapnutý.
- **OFF3** (volné doběhnutí motoru do úplného zastavení)  
Jestliže se motor točí se zadanými otáčkami, stisknutím tohoto tlačítka se začne motor pozvolna zastavovat až do úplného zastavení. Kontrolní lampa signalizuje stav, je-li aktivní OFF3 lampa se rozsvítí a pohon se po doběhnutí motoru zastaví. Uvedením do provozu tlačítkem START/STOP se OFF3 vypne.

- DIR (volí směr otáčení)  
Toto tlačítko volí buď směr otáčení vpravo nebo vlevo. Stisknutím tlačítka za chodu pohonu se motor rozběhne na druhou stranu. Kontrolní lampka signalizuje směr chodu. Je-li zhasnutá motor se otáčí vpravo, svítí-li je směr otáčení vlevo.
- ŽÁDANÁ HODNOTA OTÁČEK (volí hodnotu otáček v %)  
Stisknutím tohoto tlačítka se nám objeví dotyková klávesnice viz. obr. 5., pomocí které zadáme žádanou hodnotu otáček  $-100 \div 100\%$  ( $100\% = 50 \text{ Hz}$ ) a potvrdíme Enter. Pak už stačí jen zmáčknout START/STOP a motor se rozběhne se zadanou hodnotou otáček.



Obr. 5. Dotyková klávesnice

- SKUTEČNÁ HODNOTA OTÁČEK (zobrazí aktuální hodnotu v %)  
Toto okénko s displejem slouží pro zobrazení skutečné hodnoty otáček, která se jen nepatrně liší od námi zadaných.
- SERVISNÍ VÝPIS  
Toto tlačítko umožňuje nahlédnout do servisního menu, ve kterém se zobrazí hodnoty výstupních otáček, proudu a výstupní frekvence. Viz. obr. 6.



Obr.6. Menu SERVISNÍ VÝPIS

Pozn.: Ze všech menu se vracíme tlačítkem ZPĚT, které všechny nastavené parametry vrací zpět do výchozích hodnot. Žádanou hodnotu otáček lze měnit i za chodu pohonu..

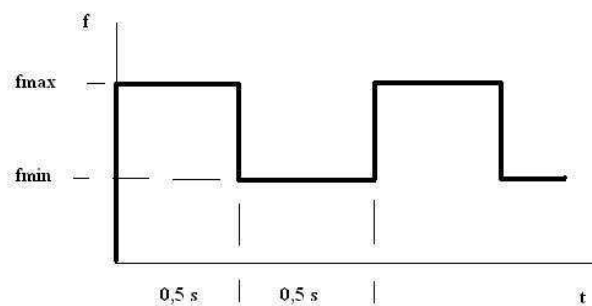
Postup:

- přepneme se na menu ZADEJ OTÁČKY stisknutím dotykového tlačítka
- na začátku je pohon vypnutý, OFF3 neaktivní, DIR na 0 (směr vpravo)
- všechny tři kontrolní lampy jsou tedy zhasnuty
- pomocí tlačítka ŽÁDANÁ HODNOTA OTÁČEK zadáme hodnotu otáček
- na displeji SKUTEČNÁ HODNOTA OTÁČEK se zobrazí aktuální otáčky
- nyní si můžeme vyzkoušet funkci tlačítek START/STOP, OFF3 a DIR
- můžeme se také přepnout do SERVISNÍHO VÝPISU (pozn.: pro servisní výpis musí být stejná hodnota znaménka jak u žádané hodnoty otáček tak i u skutečné, tzn. nepoužívat DIR, ale směr zadat z klávesnice pomocí + a -)

### 13.5.3 Menu ZADEJ PRŮBĚH SKOKOVĚ

Zmáčknutím toho tlačítka se dostaneme do menu, které umožňuje:

- zadávat maximální a minimální žádané otáčky
- zapínat/vypínat pohon
- směr otáčení je pevně nastaven vpravo
- doba rozběhu na max. otáčky 0,2s
- doba doběhu na min. otáčky 0,2s
- průběh otáček viz. obr. 7., vzhled menu viz. obr. 8.



Obr. 7. Průběh otáček pro ZADEJ PRŮBĚH SKOKOVĚ



Obr. 8. Menu ZADEJ PRŮBĚH SKOKOVĚ

Popis tlačítek:

- **MAX OTÁČKY**  
Žádaná hodnota maximálních otáček, kterých má motor dosáhnout a točit se s nimi po dobu 0,5s. Stisknutím tohoto tlačítka se nám objeví dotyková klávesnice viz. obr. 5., pomocí které zadáme žádanou hodnotu otáček 0 ÷ 100% (100% = 50 Hz) a potvrdíme Enter.
- **MIN OTÁČKY**  
Žádaná hodnota minimálních otáček, na které má motor spadnout a točit se s nimi po dobu 0,5s. Stisknutím tohoto tlačítka se nám objeví dotyková klávesnice viz. obr. 5., pomocí které zadáme žádanou hodnotu otáček 0 ÷ 100% (100% = 50 Hz) a potvrdíme Enter.
- **START/STOP**  
Toto tlačítko je alternativní, tzn. jedním stisknutím se pohon zapne a druhým vypne. Kontrolní lampa (kolečko nad tlačítkem) signalizuje stav pohonu. Jestliže svítí je pohon zapnutý.

Pozn.: Ze všech menu se vracíme tlačítkem ZPĚT, které všechny nastavené parametry vrací zpět do výchozích hodnot. Žádanou hodnotu otáček MAX i MIN lze měnit i za chodu pohonu.

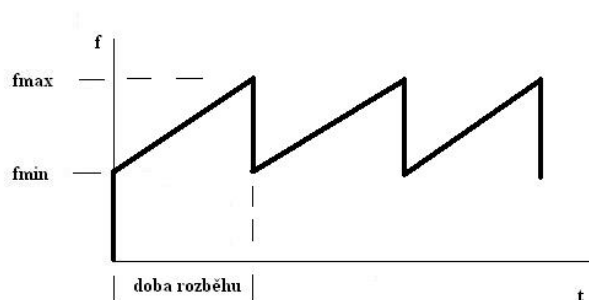
Postup:

- přepneme se na menu ZADEJ PRŮBĚH SKOKOVĚ stisknutím tlačítka
- na začátku je pohon vypnutý, kontrolní lampa zhasnuta
- pomocí tlačítek MAX a MIN OTÁČKY zadáme žádané hodnoty otáček
- nyní si můžeme vyzkoušet tlačítkem START/STOP pohon zapnout

### 13.5.4 Menu ZADEJ PRŮBĚH S RAMPOU

Zmáčknutím toho tlačítka se dostaneme do menu, které umožňuje:

- zadávat otáčky maximální, minimální a sledovat jejich aktuální hodnotu
- zapínat/vypínat pohon
- směr otáčení je pevně nastaven vpravo
- doba rozběhu na max. otáčky 10s
- doba doběhu na min. otáčky 0,2s
- průběh otáček viz. obr. 9., vzhled menu viz. obr. 10.



Obr. 9. Průběh otáček pro ZADEJ PRŮBĚH S RAMPOU



Obr. 10. Menu ZADEJ PRŮBĚH SKOKOVĚ

Popis tlačítek:

- **MAX OTÁČKY**  
Žádaná hodnota maximálních otáček, kterých má motor dosáhnout. Stisknutím tohoto tlačítka se nám objeví dotyková klávesnice viz. obr. 5., pomocí které zadáme žádanou hodnotu otáček  $0 \div 100\%$  ( $100\% = 50 \text{ Hz}$ ) a potvrdíme Enter.



- **MIN OTÁČKY**  
Žádaná hodnota minimálních otáček, na které má motor spadnout, po dosažení maximálních otáček. Stisknutím tohoto tlačítka se nám objeví dotyková klávesnice viz. obr. 5., pomocí které zadáme žádanou hodnotu otáček  $0 \div 100\%$  ( $100\% = 50 \text{ Hz}$ ) a potvrdíme Enter.
- **SKUTEČNÁ HODNOTA OTÁČEK** (zobrazí aktuální hodnotu v %)  
Toto okénko s displejem slouží pro zobrazení skutečné hodnoty otáček, která se jen nepatrně liší od námi zadaných.
- **START**  
Tímto tlačítkem se pohon zapne. Kontrolní lampa (kolečko vedle tlačítka) signalizuje stav pohonu. Jestliže svítí je pohon zapnutý.
- **STOP**  
Tímto tlačítkem se pohon vypne.

Pozn.: Ze všech menu se vracíme tlačítkem ZPĚT, které všechny nastavené parametry vrací zpět do výchozích hodnot. Žádanou hodnotu otáček MAX i MIN nelze měnit za chodu pohonu, před změnou musíme vždy pohon prvně zastavit tlačítkem STOP.

Postup:

- přepneme se na menu ZADEJ PRŮBĚH S RAMPOU stisknutím tlačítka
- na začátku je pohon vypnutý, kontrolní lampa zhasnuta
- pomocí tlačítek MAX a MIN OTÁČKY zadáme žádané hodnoty otáček
- na displeji SKUTEČNÁ HODNOTA OTÁČEK se zobrazí aktuální otáčky
- nyní si můžeme vyzkoušet tlačítkem START pohon zapnout
- tlačítkem STOP pohon vypneme

### 13.6 Vypnutí zařízení

Vypnutí celého zařízení provedeme opačným postupem jako jeho zapnutí.

1. Přepněte na automatu páčku do polohy STOP.
2. Vypněte zdroj stejnosměrného napětí DC24V pro automat a operátorský terminál.
3. Vypojte frekvenční měnič ze zásuvky 230V.

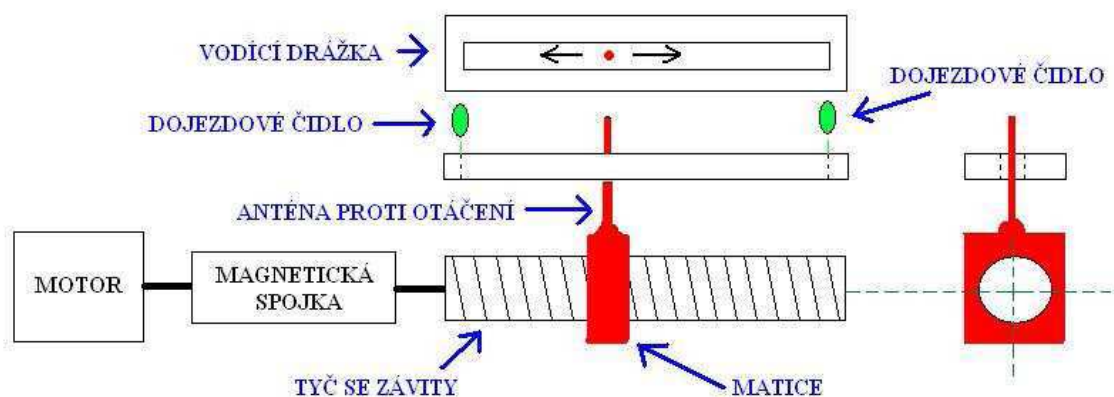
### 13.7 Závěr

Pomocí této laboratorní úlohy si studenti osvojí jak vůbec takový frekvenční měnič vypadá, seznámí se s programovatelným automatem a s dotykovým operátorským terminálem. Mohou si sami udělat závěr o tom co znamená řízení otáček motoru frekvenčním měničem a programovatelným automatem v neposlední řadě s možností ovládat toto zařízení operátorským terminál.

## 14. NÁVRHY NA ZLEPŠENÍ

### 14.1 Konstrukční návrh pro demonstraci účinku proměnných otáček

Jelikož ukázka účinku proměnných otáček motoru na samotné hřídeli není dosti názorná, pro zobrazení vlivu schopnosti řízení otáček motoru frekvenčním měničem, rozhodl jsem se navrhnout jak by mohla taková demonstrace účinku proměnných otáček vypadat.



Obr. 61. Návrh pro demonstraci účinku proměnných otáček

Popis funkce:

- hřídel motoru je spojena se závitovou tyčí magnetickou spojkou, která je schopna se při přetížení rozpojit a prokluzovat
- na závitové tyči je našroubovaná matice, na které je přivařena anténa, která zabráňuje otáčení matice při šroubování tyče
- anténa se pohybuje ve vodící drážce
- na začátku a na konci jsou dojezdová čidla polohy, které by zajišťovaly vypnutí motoru nebo jeho reverzaci

### 14.2 Návrhy na zlepšení

Rozhodně prvním návrhem, který bych zde uvedl je výměna operátorského terminálu. Někdy by jeho schopnosti byly nedostačující to spíše naopak, ale z důvodů jeho komunikačního portu RS-232, kvůli kterému jsem nemohl sdílet port 1 na automatu pro terminál a PC, viz. obr. 45. Tímto jsem byl při kontrolách chodu programu v automatu omezen:

- bylo možné buď sledovat komunikaci s terminálem, kdy jsem PPI kabel zapojil na port 0 automatu
- nebo jsem ho připojil na port 1 a kontroloval chod USS protokolu

Proto, abychom mohli sdílet port 1, bych doporučil použít např. operátorské terminály od fy. Siemens, které jsou vybaveny portem RS-485, pak už by vše fungovalo a uživatel by mohl plnohodnotně kontrolovat chod programu v programovatelném automatu.

Také by se mohl u zadávaných průběhů volit čas pro doby běhu, rozběhu a doběhu, které jsem pro jednotlivé průběhy neumožnil obsluze nastavovat a automat si je zadává sám.

Poslední mým návrhem by bylo sestavení a zprovoznění mého návrhu pro demonstraci účinku proměnných otáček

#### 14.2 Ukázka mého laboratorního stolu



Obr. 62. Ukázka mého laboratorního stolu

## 15. ZÁVĚR

Diplomová práce se zabývá možností řídit frekvenční měnič programovatelným automatem po sériové lince RS-485, pomocí USS protokolu. Tento speciální komunikační protokol umožňuje pomocí instrukcí USS řídit fyzický pohon a číst, popř. zapisovat jeho parametry. Proto jsem ho podrobně popsal v kapitole 8. a uvedl ukázky zdrojových kódů. Ve své práci jsem také zapojil ot který v náspí do mého práce pohodlné uživatelské ovládání.

Na závěr jsem zhodnotil poznatky z řešení a navrhnul další postupy.







## SEZNAM PŘÍLOH

Příloha P1: Výpis mého programu v prostředí STEP 7-Micro/WIN V4.0.

Příloha P2: CD, které obsahuje kompletní zdrojový kód v STEP 7-Micro/WIN V4.0 pro programovatelný automat S7-200, kompletní program v prostředí GT Designer2 pro operátorský terminál GOT1000, věnovanou knihovnu instrukcí USS4 firmou Siemens, elektronickou podobu diplomové práce.





**Příloha P1**

PROGRAM RIZENI\_OTACEK\_MOTORU\_FREKVEVENCNIM\_MENICEM

\*\*\*\*\* Hlavní program MAIN \*\*\*\*\*

**Network 1: inicializace USS4\_INIT**

```
LD SM0.1
CALL USS4_INIT:SBR50, 1, 9600, 16#1, Q0.0, VB0
```

- provede se v prvním programovém cyklu
- 1 = přiřadí portu 0 USS4 protokol
- nastaví přenosovou rychlost na 9600 bps
- 16#1 = označí pohon 0 za aktivní
- Q0.0 = DONE signalizuje dokončení instrukce USS4\_INIT
- VB0 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

**Network 2: počáteční inicializace**

```
LD SM0.1
R M2.0, 1 //RUN
R M2.1, 1 //OFF2
R M2.2, 1 //OFF3
R M2.3, 1 //F_ACT
R M2.4, 1 //DIR
S M5.0, 1 //bit pro nastavení přerušení
R M5.1, 1 //bit pro povolení čtení výstupních otáček
R M5.2, 1 //bit pro povolení čtení výstupního proudu
R M5.3, 1 //bit pro povolení čtení výstupní výstupního kmitočtu
R M6.0, 1 //kontrolní bit u přerušení
MOVB 0, VB20 //adresa pro počítání počtu přerušení
MOVR 0.0, VD306 //adresa pro uložení hodnoty výstupních otáček
MOVR 0.0, VD310 //adresa pro uložení hodnoty výstupního proudu
MOVR 0.0, VD314 //adresa pro uložení hodnoty výstupního kmitočtu
MOVR 0.0, VD100 //adresa pro zadávání žádané rychlosti
MOVW 0, Param:VW202 //Param je číslo parametru které se má číst nebo zapsat
MOVW 0, Index:VW204 //Index je indexové číslo daného parametru
R M8.1, 1 //bit pro povolení zápisu doby rozběhu 0.2s
R M8.2, 1 //bit pro povolení zápisu doby rozběhu 10.0s
R M9.0, 1 //pomocné bity pro zjištění aktuální výstupní rychlosti
R M9.1, 1 //pomocné bity pro zjištění aktuální výstupní rychlosti
R M9.2, 1 //pomocné bity pro zjištění aktuální výstupní rychlosti
MOVR 0.0, VD220 //adresa hodnoty zadaných MAX otáček s rampou
MOVR 0.0, VD250 //adresa hodnoty zadaných MIN otáček s rampou
MOVR 0.0, VD210 //adresa hodnoty zadaných MAX otáček skokově
MOVR 0.0, VD200 //adresa hodnoty zadaných MIN otáček skokově
```

```

MOVR 1.0, VD230 //pomocná proměnné pro zjištění aktuální rychlosti
MOVR -1.0, VD240 //pomocná proměnné pro zjištění aktuální rychlosti
MOVR 1.0, VD260 //pomocná proměnné pro zjištění aktuální rychlosti
MOVR -1.0, VD270 //pomocná proměnné pro zjištění aktuální rychlosti

```

### Network 3: řízení aktivního pohonu 0 instrukcí USS4\_DRV\_CTRL

```

LD SM0.0
= L60.0
LD M2.0
= L63.7
LD M2.1
= L63.6
LD M2.2
= L63.5
LD M2.3
= L63.4
LD M2.4
= L63.3
LD L60.0
CALL USS4_DRV_CTRL:SBR51, L63.7, L63.6, L63.5, L63.4, L63.3, 0, VD100, V50.0,
VB1, VW2, VD4, M0.0, M0.1, M0.2, M0.3

```

- M2.0 = RUN spouštění pohonu
- M2.1 = OFF2 okamžité zastavení pohonu
- M2.2 = OFF3 volné doběhnutí motoru
- M2.3 = F\_ACT porucha
- M2.4 = DIR směr

\*\*\*\*\*

0 = DRIVE adresa pohonu  
VD100 = nastavená hodnota rychlosti  
V50.0 = RSP\_RCVD odpověď přijata  
VB1 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce  
VW2 = DRV\_STATUS stavové slova odeslané zpět pohonem  
VD4 = aktuální rychlost

\*\*\*\*\*

M0.0 = DRV\_RUN indikace povolení RUN  
M0.1 = DRV\_DIR indikuje směr otáčení  
M0.2 = DRV\_INH indikuje stav blokovacího bitu pro pohon  
M0.3 = DRV\_FLT indikuje stav bitu pro poruchu

### Network 4: přístup do Menu ZADEJ PRŮBĚH SKOKOVĚ

```

LD M7.0
A M7.1
S M7.5, 1

```

- M7.0, M7.1, M7.5 pomocné řídicí bity pro řízení z Menu ZADEJ PRŮBĚH SKOKOVĚ

**Network 5: hodnoty MAX a MIN otáček pro Menu ZADEJ PRŮBĚH SKOVĚ**

```
LD SM0.5
A M7.5
MOVR VD210, VD100
LDN SM0.5
A M7.5
MOVR VD200, VD100
```

- VD210 = hodnota MAX
- VD200 = hodnota MIN
- hodnoty se pomocí speciálního bitu SM0.5 navzájem střídají a zapisují do žádané hodnoty rychlosti VD100

**Network 6: přístup do Menu ZADEJ PRŮBĚH s RAMPOU**

```
LD M9.0
CALL zjisti_otacky:SBR1
```

- M9.0 povolení zjištění otáček, jeli aktivní vyvolá se podprogram zjisti\_otacky

**Network 7: Menu SERVISNÍ VÝPIS**

```
LD M5.0
CALL vypis_par:SBR0
```

- M5.0 povolení nastavení přerušení, je-li aktivní vyvolá se podprogram vypis\_par

**Network 8: Menu SERVISNÍ VÝPIS**

```
LD M5.0
A M6.0
R M5.1, 1
R M5.2, 1
R M5.3, 1
MOVB 0, VB20
```

- M5.0 a M6.0 jsou kontrolní řídicí bity
- M5.1, M5.2, M5.3 jsou povolovací bity pro jednotlivá čtení parametrů
- VB20 počítadlo přerušení

**Network 9: čtení Otáček motoru**

```

LD M5.1
= L60.0
MOVW 22, Param:VW202
MOVW 0, Index:VW204
LD M5.1
EU
= L63.7
LD L60.0
CALL USS4_RPM_R:SBR54, L63.7, 0, Param:VW202, Index:VW204, &VB1100, V51.0,
      VB60, VD306

```

- M5.1 = bit pro povolení čtení otáček motoru

- Param = 22 číslo parametru otáček motoru

- Index = 0 index parametru pro čtení

\*\*\*\*\*

0 = DRIVE číslo pohonu ze kterého chceme číst

&VB1100 = DB\_PTR ukazatel na zásobník

V51.0 = DONE po ukončení instrukce se zapne

VB60 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

VD306 = VAL obsahuje výsledek provedení instrukce

**Network 10: čtení Výstupního proudu motoru**

```

LD M5.2
= L60.0
MOVW 27, Param:VW202
MOVW 0, Index:VW204
LD M5.2
EU
= L63.7
LD L60.0
CALL USS4_RPM_R:SBR54, L63.7, 0, Param:VW202, Index:VW204, &VB1100, V51.1,
      VB61, VD310

```

- M5.2 = bit pro povolení čtení výstupního proudu motoru

- Param = 27 číslo parametru výstupního proudu motoru

- Index = 0 index parametru pro čtení

\*\*\*\*\*

0 = DRIVE číslo pohonu ze kterého chceme číst

&VB1100 = DB\_PTR ukazatel na zásobník

V51.1 = DONE po ukončení instrukce se zapne

VB61 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

VD310 = VAL obsahuje výsledek provedení instrukce

**Network 11: čtení Aktuálního výstupního kmitočtu**

```

LD M5.3
= L60.0
MOVW 24, Param:VW202
MOVW 0, Index:VW204
LD M5.3
EU
= L63.7
LD L60.0
CALL USS4_RPM_R:SBR54, L63.7, 0, Param:VW202, Index:VW204, &VB1100, V51.2,
      VB62, VD314

```

- M5.3 = bit pro povolení čtení aktuálního výstupního kmitočtu
- Param = 24 číslo parametru aktuálního výstupního kmitočtu
- Index = 0 index parametru pro čtení

\*\*\*\*\*

0 = DRIVE číslo pohonu ze kterého chceme číst  
 &VB1100 = DB\_PTR ukazatel na zásobník  
 V51.2 = DONE po ukončení instrukce se zapne  
 VB62 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce  
 VD314 = VAL obsahuje výsledek provedení instrukce

**Network 12: zápis Doby rozběhu 0.2s**

```

LD M8.1
= L60.0
MOVW +1120, Param:VW202
MOVW +0, Index:VW204
MOVR 0.2, VD320
LD M8.1
EU
= L63.7
LD L60.0
CALL USS4_WPM_R:SBR57, L63.7, 0, Param:VW202, Index:VW204, VD320,
      &VB1250, V51.5, VB65

```

- M8.1 = bit pro povolení zápisu doby rozběhu
- Param = 1120 číslo parametru doby rozběhu
- Index = 0 index parametru pro zápis

\*\*\*\*\*

0 = DRIVE číslo pohonu na který chceme zapisovat  
 VD320 = hodnota kterou chceme zapsat 0.2s  
 &VB1250 = DB\_PTR ukazatel na zásobník  
 V51.5 = DONE po ukončení instrukce se zapne  
 VB65 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

**Network 13: zápis Doby rozběhu 10.0s**

```

LD M8.2
= L60.0
MOVW +1120, Param:VW202
MOVW +0, Index:VW204
MOVR 10.0, VD324
LD M8.2
EU
= L63.7
LD L60.0
CALL USS4_WPM_R:SBR57, L63.7, 0, Param:VW202, Index:VW204, VD324,
&VB1250, V51.6, VB66

```

- M8.2 = bit pro povolení zápisu doby rozběhu

- Param = 1120 číslo parametru doby rozběhu

- Index = 0 index parametru pro zápis

\*\*\*\*\*

0 = DRIVE číslo pohonu na který chceme zapisovat

VD324 = hodnota kterou chceme zapsat 10.0s

&VB1250 = DB\_PTR ukazatel na zásobník

V51.6 = DONE po ukončení instrukce se zapne

VB66 = ERR obsahuje chyby, které mohou vyplynout z provedení této instrukce

\*\*\*\*\* Podprogram vypis\_par \*\*\*\*\*

**Network 1: nastavení přerušení pro SERVISNÍ VÝPIS**

```

LD SM0.0
MOVB 100, SMB34
ATCH INT_0:INT0, 10
ENI

```

- přerušení je nastaveno na 100ms

- každých 100ms se inkrementuje počítadlo VB20 pro čtení parametrů

**Network 2: nastavení pomocné proměnné**

```
S M6.0, 1
```

- M6.0 = pomocný bit k přerušení

\*\*\*\*\* Podprogram zjistí\_otacky \*\*\*\*\*

**Network 1: nastavení pomocných hodnot pro zjištění aktuálních otáček**

```
LD M9.1
+R VD220, VD230
+R VD220, VD240
R M9.1, 1
```

```
LD M9.2
+R VD250, VD260
+R VD250, VD270
R M9.2, 1
```

- aktivací M9.1 a M9.2 se povoluje nastavení pomocných proměnných pro toleranci + nebo - 0.5 Hz

**Network 2: zjištění aktuálních otáček motoru u Menu ZADEJ PRŮBĚH S RAMPOU**

```
MOVR VD220, VD100
LDR>= VD4, VD240
S M10.0, 1
LDR<= VD4, VD230
S M10.1, 1
LD M10.0
A M10.1
MOVR VD250, VD100
R M10.2, 2
LDR>= VD4, VD270
S M10.2, 1
LDR<= VD4, VD260
S M10.3, 1
LD M10.2
A M10.3
MOVR VD220, VD100
R M10.0, 2
```

- jsou-li aktuální otáčky stejné se zadanou hodnotou v MAX u Menu ZADEJ PRŮBĚH S RAMPOU v toleranci + nebo - 0.5Hz nastaví se žádaná hodnota na zadanou hodnotu v MIN u Menu ZADEJ PRŮBĚH SKOKOVĚ
- jsou-li aktuální otáčky stejné se zadanou hodnotou v MIN u Menu ZADEJ PRŮBĚH S RAMPOU v toleranci + nebo - 0.5Hz nastaví se žádaná hodnota na zadanou hodnotu v MAX u Menu ZADEJ PRŮBĚH SKOKOVĚ
- takhle se to postupně střídá abychom dosáhly pilového průběhu



\*\*\*\*\*Přerušení INT\_0\*\*\*\*\*

**Network 1: počítání počtu přerušení**

```
LD SM0.0
INCB VB20
LDB= VB20, 5
S M5.1, 1
LDB= VB20, 10
R M5.1, 1
LDB= VB20, 15
S M5.2, 1
LDB= VB20, 20
R M5.2, 1
LDB= VB20, 25
S M5.3, 1
LDB= VB20, 30
R M5.3, 1
MOVB 0, VB20
```

- pomocí přerušení se mezi sebou přepínají parametry pro čtení
- jelikož v jednu dobu může být aktivní pouze jedna instrukce pro čtení