

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**Katedra elektromechaniky a výkonové elektroniky**

**DIPLOMOVÁ PRÁCE**

**Rozšíření laboratorních úloh s PLC Simatic**

**vedoucí práce: Ing. Martin Pittermann, Ph.D.**  
**autor: Jan Vitek**

**2013**



## **Anotace**

Předkládaná diplomová práce řeší problematiku programovatelného logického kontroleru od společnosti SIEMENS řady S7-200 použitého spolu s rozšiřujícím ethernet modulem, výukovým modelem kódovací linky s ražením a dotykovým panelem TP177 micro pro řadu výukových úloh. Tyto úlohy se zabývají vytvořením jednotlivých částí složitějšího programu, tak aby každá úloha představila nové programovací prvky.

## **Klíčová slova**

Programovatelný logický kontroler řady S7-200, dotykový panel, výukový model, síťový rozšiřující modul, STEP 7 MicroWIN, WinCC flexible

## **Abstract**

The diploma thesis presents the principles of programmable logic controller S7-200 class from SIEMENS company, used with ethernet expansion module, educational model of checking station with punch and touchpanel TP177 micro also from SIEMENS, for number of educational examples. Each of these examples presents new program elements so that together they create one complex program.

## **Key words**

Programmable logic controller S7-200 class, touchpanel, teaching model, ethernet expansion module, STEP 7 MicroWIN, WinCC flexible

## **Prohlášení**

Předkládám tímto k posouzení a obhajobě diplomovou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této diplomové práce, je legální.

V Plzni dne

Jan Vítek

.....

## **Poděkování**

Tímto bych rád poděkoval vedoucímu diplomové práce Ing. Martinu Pittermannovi, Ph.D. Za umožnění vypracování této práce pro Gymnázium a SOŠ Rokycany, Mládežníků 1115 bych rád poděkoval Ing., Bc. Lence Likeové.

# Obsah

<b>OBSAH</b> .....	<b>7</b>
<b>SEZNAM SYMBOLŮ</b> .....	<b>9</b>
<b>SEZNAM ZKRATEK</b> .....	<b>10</b>
<b>ÚVOD</b> .....	<b>11</b>
<b>1 STEP 7-MICRO/WIN DŮLEŽITÉ PROGRAMOVÉ PRVKY</b> .....	<b>12</b>
1.1 ZÁKLADNÍ PRVKY .....	12
1.2 ČÍTAČE A ČASOVAČE .....	12
1.3 OPERACE PRO PŘÍRAZENÍ, PŘEVOD A NAVÝŠENÍ HODNOTY .....	13
<b>2 NASTAVENÍ ETHERNET MODULU</b> .....	<b>14</b>
2.1 ZADÁNÍ ÚLOHY .....	14
2.2 ŘEŠENÍ .....	14
2.2.1 <i>Spojení PLC - PC pomocí ethernet modulu</i> .....	14
2.2.2 <i>Spuštění SMTP serveru a odeslání testovacího e-mailu</i> .....	17
2.2.3 <i>Vytvoření jednoduchého programu pro odeslání emailu s proměnnou hodnotou</i> .....	18
2.2.4 <i>Ovládání programu</i> .....	19
2.2.5 <i>Praktické využití</i> .....	20
<b>3 IDENTIFIKACE VSTUPŮ A VÝSTUPŮ MODELU</b> .....	<b>22</b>
3.1 ZADÁNÍ ÚLOHY .....	22
3.2 ŘEŠENÍ .....	22
3.2.1 <i>Identifikace vstupů modelu</i> .....	22
3.2.2 <i>Identifikace vstupů PLC</i> .....	23
<b>4 ZÁKLADNÍ PROGRAM</b> .....	<b>24</b>
4.1 ZADÁNÍ ÚLOHY .....	24
4.2 ŘEŠENÍ .....	24
4.2.1 <i>Program pro PLC</i> .....	24
4.2.2 <i>Program HMI</i> .....	26
<b>5 PROGRAM PRO ČTENÍ KÓDU VÝROBKU</b> .....	<b>27</b>
5.1 ZADÁNÍ ÚLOHY .....	27
5.2 ŘEŠENÍ .....	27
5.2.1 <i>Program pro PLC</i> .....	27
5.2.2 <i>Program pro HMI</i> .....	28
<b>6 PAMĚŤ</b> .....	<b>31</b>
6.1 ZADÁNÍ ÚLOHY .....	31
6.2 ŘEŠENÍ .....	31
6.2.1 <i>Program pro PLC</i> .....	31
6.2.2 <i>Program pro HMI</i> .....	32
<b>7 PROGRAM RAŽENÍ</b> .....	<b>34</b>
7.1 ZADÁNÍ ÚLOHY .....	34
7.2 ŘEŠENÍ ÚLOHY .....	34
7.2.1 <i>Program pro PLC, podprogram ražení</i> .....	34
7.2.2 <i>Program pro PLC, úpravy hlavního programu</i> .....	36

<b>8 ZOBRAZENÍ PRVKŮ NA PANELU .....</b>	<b>37</b>
8.1 ZADÁNÍ ÚLOHY .....	37
8.2 ŘEŠENÍ.....	37
8.2.1 Program pro PLC .....	37
8.2.2 Program pro HMI .....	39
<b>9 CHYBOVÉ HLÁŠENÍ .....</b>	<b>42</b>
9.1 ZADÁNÍ ÚLOHY .....	42
9.2 ŘEŠENÍ.....	42
9.2.1 Program pro PLC .....	42
9.2.2 Program pro HMI .....	43
<b>ZÁVĚR .....</b>	<b>45</b>
<b>POUŽITÁ LITERATURA.....</b>	<b>47</b>
<b>SEZNAM OBRÁZKŮ .....</b>	<b>48</b>
<b>SEZNAM TABULEK.....</b>	<b>49</b>
<b>PŘÍLOHA A CELÝ PROGRAM.....</b>	<b>50</b>
HLAVNÍ PROGRAM .....	50
PODPROGRAM PŘÍŘAZENÍ PAMĚTI.....	53
PODPROGRAM RAŽENÍ .....	53
PODPROGRAM PRO ZOBRAZENÍ NA PANELU .....	54
PODPROGRAM SPUŠTĚNÍ RAŽENÍ.....	55
OVLÁDACÍ PANEL .....	55
TABULKY ADRES A SYMBOLŮ .....	57



## Seznam symbolů

Symbol	Adresa	Popis	Symbol	Adresa	Popis
AIN	I0.0	Analogové tlačítko vstup	P1PB	M2.3	Pozice 1 přídržný bit ražení
AOUT	I0.1	Analogové tlačítko výstup	P2PB	M2.4	Pozice 2 přídržný bit ražení
SeIN	I0.2	Senzor na vstupu	P3PB	M2.5	Pozice 3 přídržný bit ražení
SeRZ	I0.3	Senzor u ražení	CHR	M3.0	Pomocný bit resetu ERRC
SP	I0.4	Pravý senzor kódování	CRES	M3.1	Úplný reset
SL	I0.5	Levý senzor kódování	PBZCH	M3.2	Bit zobrazení chyby
SRUP	I0.6	Ražení nahoře	PBCHZ	M3.3	Bit chyby chod zpět
SRDN	I0.7	Ražení dole	PRUN	M4.0	Spuštění z HMI
PFW	Q0.0	Pás chod vpřed	PSTOP	M4.1	Vypnutí z HMI
PBCK	Q0.1	Pás chod vzad	PM0	MB0	Vstupní paměť matice
RUP	Q0.2	Ražení chod nahoru	PM10	MB10	Paměť matice M10
RDN	Q0.3	Ražení chod dolů	PM11	MB11	Paměť matice M11
RL	Q0.4	Ražení kontrolka	PM12	MB12	Paměť matice M12
X1	M0.0	Pozice v matici	PVD1	MW13	Počet dílů
X2	M0.1	Pozice v matici	PVD2	MW15	Počet dílů pomocný word
X3	M0.2	Pozice v matici	CPLD	MW17	Celkový počet L dílů
X4	M0.3	Pozice v matici	CPWD	MW19	Celkový počet W dílů
X5	M0.4	Pozice v matici	CPCD	MW21	Celkový počet C dílů
X6	M0.5	Pozice v matici	TP1	T33	Časovač pozice 1
RUN	M1.0	Bit pro spuštění	TP3	T34	Časovač pozice 3
MRES	M1.1	reset paměti	TVST	T37	Časovač vstupního senzoru
TBPB	M1.2	Pomocný bit běhu pásu	TCT	T38	Časovač na senzorech čtení
PB1	M1.3	Přídržný bit L dílu	SDT	T39	Časovač pro blikání
PB2	M1.4	Přídržný bit W dílu	TPM	T40	Časovač pro paměť
PB3	M1.5	Přídržný bit C dílu	TBP	T42	Časovač běhu pásu
NULL	M1.6	Vynulování počtu dílů	CPM	C1	Čítač pozice v matici
PMBRX1	M2.0	Pomocný bit ražení pozice 1	CRAZ	C2	Čítač pro zastavení ražení
PMBRX2	M2.1	Pomocný bit ražení pozice 2	ERRC	C3	Čítač chybové hlášky
PMBRX3	M2.2	Pomocný bit ražení pozice 3			

Tabulka symbolů a adres pro PLC program [zdroj: vlastní tvorba]

Symbol	Adresa	Popis	Symbol	Adresa	Popis
X10_1	M10.0	Paměť pozice v matici	X11_4	M11.3	Paměť pozice v matici
X10_2	M10.1	Paměť pozice v matici	X11_5	M11.4	Paměť pozice v matici
X10_3	M10.2	Paměť pozice v matici	X11_6	M11.5	Paměť pozice v matici
X10_4	M10.3	Paměť pozice v matici	X12_1	M12.0	Paměť pozice v matici
X10_5	M10.4	Paměť pozice v matici	X12_2	M12.1	Paměť pozice v matici
X10_6	M10.5	Paměť pozice v matici	X12_3	M12.2	Paměť pozice v matici
X11_1	M11.0	Paměť pozice v matici	X12_4	M12.3	Paměť pozice v matici
X11_2	M11.1	Paměť pozice v matici	X12_5	M12.4	Paměť pozice v matici
X11_3	M11.2	Paměť pozice v matici	X12_6	M12.5	Paměť pozice v matici

Tabulka symbolů a adres přidávaných pro dotykový panel [zdroj: vlastní tvorba]

## Seznam zkratk

PLC	Programovatelný kontroler (Programable Logic Control)
HMI	Rozhraní mezi člověkem a strojem (Human Machine Interface)
IEC	Mezinárodní elektrotechnická komise (International Electrotechnical Commission)
PPI	Parallel Peripheral Interface
SMTP	Simple Mail Transfer Protocol
RJ45	Rozhraní pro připojení rozšiřujících modulů k PLC
LPT	Starý konektor k připojení tiskáren (Line Printer Terminal)
Bit	Základní jednotka informace (binary digit)
BYTE	Osmiciferné binární číslo (8 bitů)
WORD	1 WORD = 16 bit
DWORD	Double WORD (DWORD = 2 WORD)
Integer	Celé číslo např. 1024, -2048

*Tabulka seznamu seznamu zkratk [zdroj: vlastní tvorba]*

## Úvod

Od jejich vzniku na počátku sedmdesátých let minulého století se programovatelné logické kontroléry (PLC) rozšířily do všech odvětví průmyslu. Používají se k řízení procesů elektráren, výrobních linek, pohonů a domácích spotřebičů. Právě kvůli jejich širokému využití se dnes práce s nimi vyučuje i na středních odborných školách.

Tato diplomová práce byla vytvořena k výukovým účelům pro Gymnázium a SOŠ Rokycany, Mládežníků 1115 ve spolupráci s touto školou. Jedná se o pokračování bakalářské práce "Výukový mechatronický systém s PLC" [6], která mimo jiné popisuje funkce programovacích prostředí STEP 7-Micro/WIN a WinCC flexible. V tomto didaktickém materiálu se již předpokládá znalost prostředí a ovládání základních funkcí softwaru.

Cílem této diplomové práce je vytvoření řešených příkladů, na kterých jsou názorně předvedeny a vysvětleny různé možnosti programování PLC. Příklady jsou soustředěny na vytvoření programu pro řízení linky kódovacího zařízení s ražením od společnosti Staudinger GMBH [3]. Jejíž řízení se realizuje pomocí PLC řady S7-200 CPU 222 od společnosti SIEMENS společně s dotykovým panelem TP177micro od stejné společnosti.

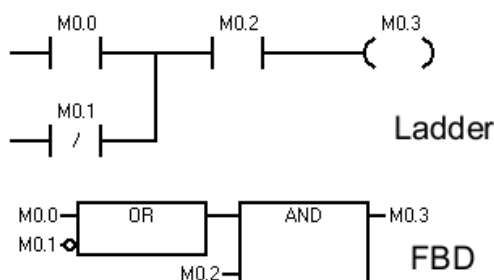
Složitější systémy, jako například elektrárny, mají centrální stanoviště (server), ze kterého se kontrolují jednotlivé procesy. Připojení kontrolérů těchto procesů k serveru se řeší pomocí průmyslové sítě (Industrial Ethernet). Tato síť kromě přenosu naprogramovaných informací umožňuje v případě potřeby nahrání programu do PLC. Vysvětlení nastavení některých funkcí rozšiřujícího ethernet modulu CP243-1 se věnuje druhá kapitola. Z důvodu hodinových dotací cílového předmětu nebylo možné vytvořit systém spojení více PLC, proto jsou ukázány funkce ethernet modulu, které umožňuje dostupné vybavení.

V průběhu vytváření práce byly její části konzultovány s vyučující odborného předmětu, pro který je tato práce určena. Zároveň byly vytvořené části testovány přímo ve výuce a podněty ze strany vyučující a žáků byly použity k vytvoření řešení problematických partií.

## 1 Step 7-Micro/WIN důležité programové prvky

V této kapitole jsou ukázány základní prvky použité k vytvoření řešení jednotlivých úloh. Aplikace těchto prvků se vysvětluje v kapitolách, ve kterých se vyskytují.

### 1.1 Základní prvky



Obr. 1 Rozlišení mezi Ladder a FBD editorem [zdroj: vlastní tvorba]

V programu STEP 7 v záložce "view" se nachází volby "STL", což je programování formou textu, "Ladder" a "FBD". Ladder a FBD jsou přehlednější a tedy vhodnější na názorné ukázky. Z obrázku 1 je zřejmé, že zatímco v Ladder zobrazení jsou vodivé cesty přerušeny spínači nebo vypínači, v FBD je toto znázornění tvořeno formou logických členů a případně negovaných vstupů.

### 1.2 Čítače a časovače

Rozdělení časovačů:

Časovače			
Typ	Nastavení	Krok časovače	Popis
TON	T32, T96	1ms	Přivedením signálu časovač běží po vypnutí se resetuje.
	T33-T36, T97-T100	10ms	
	T37-T63, T101-T255	100ms	
TONR	T0, T64	1ms	Přivedením signálu časovač běží po vypnutí drží hodnotu.
	T1-T4, T65-T68	10ms	
	T5-T31, T69-T95	100ms	
TOF	T32, T96	1ms	Přivedením signálu se časovač resetuje při vypnutí se spustí.
	T33-T36, T97-T100	10ms	
	T37-T63, T101-T255	100ms	

Tab. 1 Rozdělení časovačů [zdroj: vlastní tvorba]

Rozdělení čítačů:

TYP	Popis
CTU	Nastavení čítače: C1, C2,... (Counter up) nastavením řídicí svorky CU se přičte 1 k současné hodnotě jedná se o Integer
CTUD	Nastavení čítače: C1, C2,... (Counter up) nastavením řídicí svorky CU se přičte 1 k současné hodnotě a nastavením řídicí svorky CD se od současné hodnoty odečte 1 jedná se o Integer

Tab. 2 Rozdělení použitých čítačů [zdroj: vlastní tvorba]

### 1.3 Operace pro přiřazení, převod a navýšení hodnoty

Prvky MOVE přiřadí hodnotu vstupního BYTE, WORD, DWORD do výstupního, pokud je přivedeno napájení do vstupu EN.

Typ	Hodnota	Popis
MOV_B	BYTE	Přiřadí hodnotu ze vstupního BYTE na výstupní BYTE
MOV_W	WORD	Přiřadí hodnotu ze vstupního WORD na výstupní WORD
MOV_DW	DWORD	Přiřadí hodnotu ze vstupního DWORD na výstupní DWORD

Tab. 3 Rozdělení přiřazování hodnot [zdroj: vlastní tvorba]

V záložce "convert" jsou prvky programu používané pro převod mezi hodnotami. Například Integer - BYTE. Tyto prvky fungují stejně jako přiřazení hodnoty, přivedením napájení na vstup EN se provede operace převodu.

A pro navýšení stávající hodnoty o 1 slouží funkce INC (increment). Přičte hodnotu 1 k současné hodnotě, podobně jako čítač. INC na rozdíl od čítače navyšuje hodnotu v každém cyklu programu, je-li přivedeno napájení do vstupu EN, čítač však navyšuje hodnotu pouze jednou při každém sepnutí řídicího kontaktu. Stejným způsobem funguje i snížení hodnoty DEC.

Typ	Hodnota	Popis
INC_B	BYTE	Navýší hodnotu vstupního BYTE a uloží do výstupního BYTE
INC_W	WORD	Navýší hodnotu vstupního WORD a uloží do výstupního WORD
INC_DW	DWORD	Navýší hodnotu vstupního DWORD a uloží do výstupního DWORD

Tab. 4 Rozdělení navyšování hodnot [zdroj: vlastní tvorba]

## 2 Nastavení Ethernet modulu

### 2.1 Zadání úlohy

Nahradit PPI kabel připojením přes ethernet modul a nastavit emailovou komunikaci. Vytvořit spojení PLC - PC pomocí ethernetového modulu. Spustit SMTP server a odeslat testovací e-mail. Vytvořit jednoduchý program, který odešle přednastavený e-mail s proměnnou hodnotou.

### 2.2 Řešení

#### 2.2.1 Spojení PLC - PC pomocí ethernet modulu

Aby bylo možné nastavit ethernet modul, musí se nejprve zajistit komunikace mezi STEP 7 – Micro/WIN (zkráceně STEP 7) a PLC. Nejjednodušší způsob je využití PPI (Parallel Peripheral Interface) kabelu. Nastavení spojení v STEP 7 se nachází v záložce z hlavní nabídky "Communications", "set PG/PC interface" zvolit "PC/PPI cable(PPI)" a připojit v záložce komunikace.

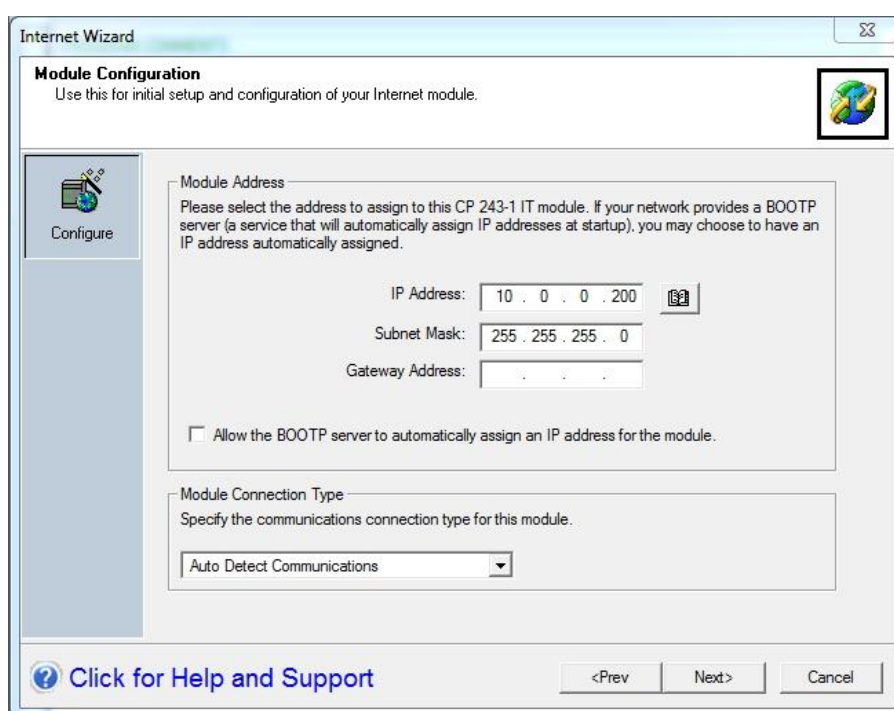
Ethernet modul se k PLC připojuje konektorem RJ45 a napájí se stejně jako zbytek sestavy 24V. K nastavení ethernet modulu slouží průvodce nastavením "Internet wizard" v STEP 7, ten je v záložce "Wizards, Internet". První obrazovka informuje o funkcích, které wizard nastavuje a upozorňuje, že projekt musí být zkompileovaný v symbolickém adresovacím módu.

V druhém okně se nastavuje pozice ethernet modulu vůči PLC. K PLC CPU 222 se mohou připojit pouze dva rozšiřující moduly, pozice 0 a 1 jsou vyhrazeny pro tyto moduly a pozice 2 je přiřazena PLC. Rozšiřující moduly se připojují sériově a jejich pozice je určena vzdáleností od PLC. Tedy modul připojen přímo k PLC má pozici 0. Po stisknutí tlačítka "Read Modules" načte program informaci o připojených modulech z PLC, v tomto případě CP243-1 na pozici 0.

Následující obrazovku znázorňuje Obr. 2, v té se nastavuje IP adresa ethernet modulu a maska podsítě. Do IP adresy se musí vyplnit neobsazená adresa sítě se stejným formátem. Například v tomto případě je ethernet modul připojen na router s IP adresou 10.0.0.138 a k němu připojen počítač s IP adresou 10.0.0.3. Proto se použije adresa modulu 10.0.0.200. IP adresa počítače se zjistí v příkazovém řádku příkazem "ipconfig". Masku podsítě bývá zpravidla 255.255.255.0. Políčko "Allow the BOOTP server to automatically assign an IP address for the module" musí být odškrtnuté, jedná se o starý systém přiřazování

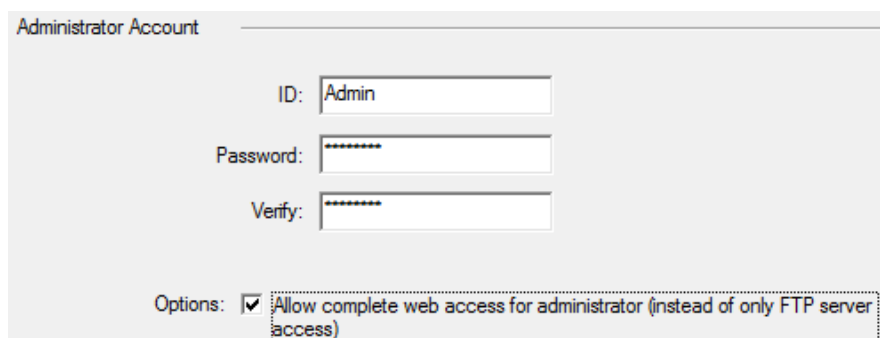
IP adres, který dnes nebývá využíván. V části nastavení "Module Connection Type", postačí zvolit "Auto Detect Communications".

Další obrazovka umožňuje nastavení více připojení k tomuto PLC. To znamená spolupráce s více PLC, každý s vlastním ethernet modulem. Ethernet modul řady S7-200, dovoluje spojení maximálně osmi PLC. Také je možné připojení k serveru nebo nastavit PLC jako server. Toho se využívá k vytvoření složitějšího systému, který umožňuje předávat data mezi PLC. V takovém spojení mohou být i PLC vyšších řad. Pro práci s jedním kontrolérem se volí počet paralelních spojení 0.



Obr. 2 Ethernet IP adresa [zdroj: vlastní tvorba]

V následujícím okně "CRC protection" se nastavuje ochrana proti nechtěnému přepisu dat. To znamená, že není možné upravovat hodnoty pomocí STEP 7, jako například stav bitu, pokud je PLC v run módu. Vhodné využití pro praktické aplikace, k zabránění nechtěného stavu provádí-li se kontrola programu, nicméně pro vývoj programu nevhodné. Právě možnost změny hodnoty bitů usnadní simulaci různých stavů, bez připojení ovládacího zařízení. Je tedy nutné zvolit možnost "No, do not generate CRC protection for this configuration". Druhým nastavitelným parametrem na této obrazovce je "Keep Alive Interval", který udržuje komunikaci mezi PLC a PC, jiným PLC nebo serverem. Výchozí nastavení 30s postačí pro tuto aplikaci.



Administrator Account

ID: Admin

Password: \*\*\*\*\*

Verify: \*\*\*\*\*

Options:  Allow complete web access for administrator (instead of only FTP server access)

Obr. 3 Administrator account [zdroj: vlastní tvorba]

Nastavení administrátorského účtu, který umožňuje přístup k datům v modulu pomocí internetového prohlížeče a odeslání zkušebního e-mailu zobrazuje Obr. 3. Heslo musí mít alespoň 8 znaků. Možnost "Allow complete web access for administrator (instead of only FTP server access)" musí být zaškrtnuta, aby se mohl odeslat zkušební e-mail.

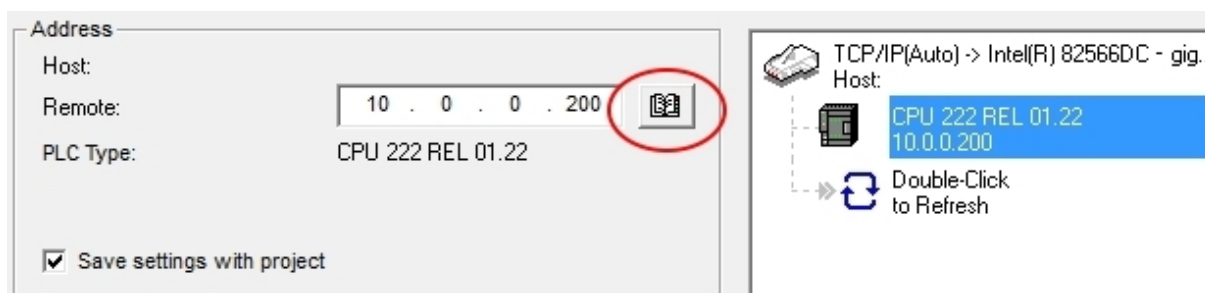
V dalším okně se musí zaškrtnout možnosti "Enable E-mail services", pro přidání podprogramu na odesílání e-mailů. A "Enable HTTP (web) services" umožňuje přístup k ethernet modulu pomocí internetového prohlížeče.

Následně se volí umístění vytvořeného nastavení v paměti. A předposlední obrazovka ukazuje, které podprogramy budou přidány. Na poslední obrazovce je možné nastavit uživatelské účty, volitelný e-mail, a FTP nastavení. Nyní postačí ukončit průvodce nastavením.

Posledním krokem před připojením PLC k PC pomocí ethernet modulu je stažení takto nastaveného programu do PLC a spuštění RUN módu. Tím dojde k aplikaci konfigurace v ethernet modulu a je možné změnit nastavení vlastností spojení.

V STEP 7 v okně "Communications" volba "Set PG/PC interface" se nachází několik komunikačních protokolů. Kromě již zmíněného "PC/PPI cable(PPI)", jsou zde protokoly nastavené v počítači na kterém je STEP 7 nainstalován, včetně "TCP/IP(Auto)" který umožní připojení. Po změně komunikačního protokolu, se v okně "Communications" nastaví IP adresa - viz Obr. 4. IP adresa PLC se zadává v okně, které se otevře pomocí naznačeného tlačítka. Jednoduše volbou "new adress" se zadá IP adresa a potvrzením dojde k uložení. Poté "Refresh", čímž načte PLC, které se musí zvolit a potvrzením je nastavení ukončeno.





Obr. 4 Communications výřez [zdroj: vlastní tvorba]

## 2.2.2 Spuštění SMTP serveru a odeslání testovacího e-mailu

K odeslání e-mailu je nezbytný přístup na SMTP server (Simple Mail Transfer Protocol) a znát IP adresu tohoto serveru. Odesílání zpráv do veřejné sítě umožňují pouze veřejné IP adresy. V tomto ohledu se práce soustředí na lokální server. Na internetu lze najít široký výběr softwaru pro vytvoření SMTP serveru. Zde je použit volně dostupný software ArGoSoft Mail Server [2].

Instalace s průvodcem instalace je velice jednoduchá. Po nainstalování stačí program spustit k uvedení SMTP serveru do online režimu. Dále se musí vytvořit nový účet uživatele v "tools, users, Add new user", pro přijímání e-mailu. Adresa takto vytvořeného účtu může být například `janvitek@localhost.cz`

Po přípravě tohoto serveru se nastaví ethernet modul, pro odeslání testovacího emailu. V STEP 7 v záložce "wizards, Internet, INT configuration for 0 (module0)" jsou nastavení, která byla vytvořena při dokončení průvodce nastavení. V "E-mail configuration" se nachází nastavení na odeslání e-mailu. Do "e-mail address" je možné vložit libovolnou adresu ve formátu `ethernet@ethernet.cz`. IP adresa počítače, na kterém se nachází server, se zadává do "SMTP server 1".

V druhém okně se nastavuje formát zpráv, které jsou odesílány pomocí programu, až 31 zpráv může být přednastaveno. Hlavním rozdílem proti běžným emailovým zprávám je možnost odeslat hodnotu z paměti programu. V nabídce "Insert data" se musí vyplnit informace pro správné zobrazení těchto dat. Mohou se zobrazit pouze hodnoty typu BYTE, WORD nebo DOUBLEWORD například MW0. Není možné zobrazit hodnotu čítače, jelikož se jedná o INTEGER. Posledním nastavitelným parametrem je umístění desetinné čárky a rozsah čísla.

Zkušební e-mail se odesílá pomocí internetového prohlížeče. Zadáním IP adresy ethernet modulu (10.0.0.200) se otevřou stránky, které jsou součástí továrního nastavení modulu. K přihlášení na tyto stránky se využijí přihlašovací údaje vyplněné v průvodci

nastavení modulu. Na těchto stránkách se nachází informace o stavu PLC a přídatných modulů viz obr. 5. V záložce "Status Chart" lze zobrazit hodnoty jednotlivých adres, například bit, byte, word atd. "IT information" zobrazuje informace o ethernet modulu, verze firmware, IP adresa, stav emailového klienta, stav připojení STEP 7 atd. Pro odeslání zkušebního emailu se musí v záložce "Send Testmail" vyplnit příjemce, tedy vytvořený účet v ArGoSoftu janvitek@localhost.cz a potvrdit tlačítkem "Send Mail". Po úspěšném odeslání je uživatel informován velkým textem "OK Email successfully delivered to email server". Nedojde-li k úspěšnému odeslání zprávy, vypíše se na stránce zdroj problému.

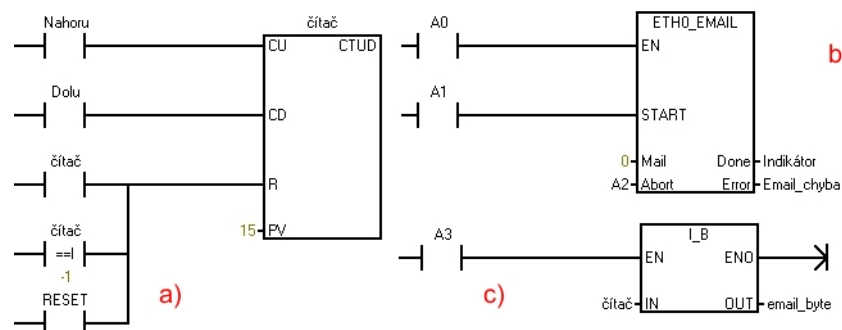
Slot	Module Type	Module Status
PLC	CPU 222 REL 01.22: digital 8 inputs / 8 outputs	ERROR <input type="checkbox"/> RUN <input checked="" type="checkbox"/> STOP <input type="checkbox"/>
0	CP243-1 INTERNET	WARNING <input type="checkbox"/> LINK <input checked="" type="checkbox"/> RUN <input checked="" type="checkbox"/> CFG <input checked="" type="checkbox"/>

Obr. 5 Výřez z webové stránky ethernet modulu [zdroj: vlastní tvorba]

Zadáním adresy localhost:81 do prohlížeče se otevře vytvořený email server. Po přihlášení bude v přijatých zprávách testovací zpráva.

### 2.2.3 Vytvoření jednoduchého programu pro odeslání emailu s proměnnou hodnotou

K vytvoření jednoduchého programu postačí několik spínačů, čítač, podprogram odeslání emailu a převod viz obr. 6. Symboly a přiřazené adresy programu se nachází v tabulce 5. Jeden ze způsobů, jak vytvořit proměnnou hodnotu, je pomocí čítače obrázek 6 a). Čítač CTUD (count Up/Down) umožňuje zvyšování a snižování hodnoty. Přivedením signálu na CU (count up) se navýší hodnota o 1 a přivedením signálu na CD (count down) se hodnota sníží o 1.



Obr. 6 Program pro odeslání emailu [zdroj: Vlastní tvorba]

Za problém lze považovat skutečnost, že v emailu lze odeslat proměnné hodnoty pouze typu BYTE, WORD a DWORD, přičemž hodnota čítače je integer. Jednoduché řešení

se nabízí ve funkci "convert" z integer na BYTE. Na výstupu však není možné mít zápornou hodnotu, neboť to převod neumožňuje. Tento stav tedy nesmí nastat, zajištěním resetování čítače v případě, že hodnota klesne pod 0, se nechtěnému stavu předejde. To zajišťuje porovnání hodnota čítače rovna -1 - viz obr. 6 a). Dále se také může omezit horní hranice tak, že při dosažení hodnoty čítače 15, dojde k resetu. A samozřejmostí je možnost ručního vynulování čítače.

K odesílání e-mailu se použije podprogram ETH0\_EMAIL, který se nachází v záložce instrukcí v "Call Subroutines". Jako u většiny funkcí, i zde se nachází napájecí vstup EN, který musí být aktivní, aby bylo možné s podprogramem pracovat. Přivedením signálu na START, dojde k odeslání přednastavené zprávy. Tato zpráva se nastavuje podle výše uvedeného návodu. Vstup "Mail" určuje, která z připravených zpráv bude odeslána, zde se jedná o zprávu na pozici 0. Pokud bude binární vstup "Abort" aktivní, tak zabráni odeslání zprávy. Po odeslání zprávy, se binární výstup "Done" změní na 1, tím je možné zobrazit informaci o úspěšném odeslání zprávy na displeji. Pokud nastane chyba při odesílání zprávy, tak výstup "Error" umožňuje identifikaci chyby a její zobrazení na operátorském stanovišti, jedná se o hodnotu BYTE.

Tabulka symbolů			
Symbol	Adresa	Symbol	Adresa
A0	M0.0	Indikátor	M0.6
A1	M0.1	RESET	M0.7
A2	M0.2	Čítač	C1
A3	M0.3	Email_chyba	MB5
Nahoru	M0.4	Email_byte	MB1
Dolu	M0.5		

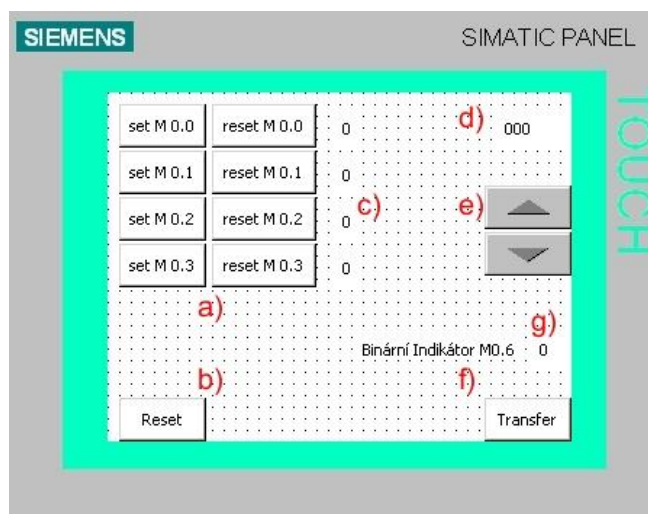
Tab. 5 Tabulka symbolů programu [zdroj: vlastní tvorba]

## 2.2.4 Ovládání programu

Ovládání se může uskutečnit pomocí STEP 7 a v RUN módu upravovat jednotlivé bity. Elegantnější řešení se však nabízí pomocí dotykového panelu, který se programuje pomocí SIMATIC WinCC flexible 2007. Vytvoření přehledové obrazovky, umožní sledovat chování programu. Což je vhodné zejména pro vývoj a testování programu.

Příklad přehledové obrazovky se nachází na obrázku 7. Ovládání se skládá z tlačítek na nastavení bitových spínačů a), set nastavuje při stisku 1 a reset 0. Textové pole, které zobrazí aktuální hodnotu bitu c). Reset b) a zvýšení/snížení hodnoty e) nastavují daný bit do jedničky po dobu stisku, při uvolnění nastavují opět nulu. Textové pole d) vypisuje hodnotu čítače. Binární indikátor g) zobrazuje aktuální hodnotu bitu stejně tak c). Transfer f) se

použije, pro nahrazení programu v panelu. Při vývoji je důležité umístit tuto funkci do ovládání, v opačném případě se musí vypnout a opět zapnout panel.



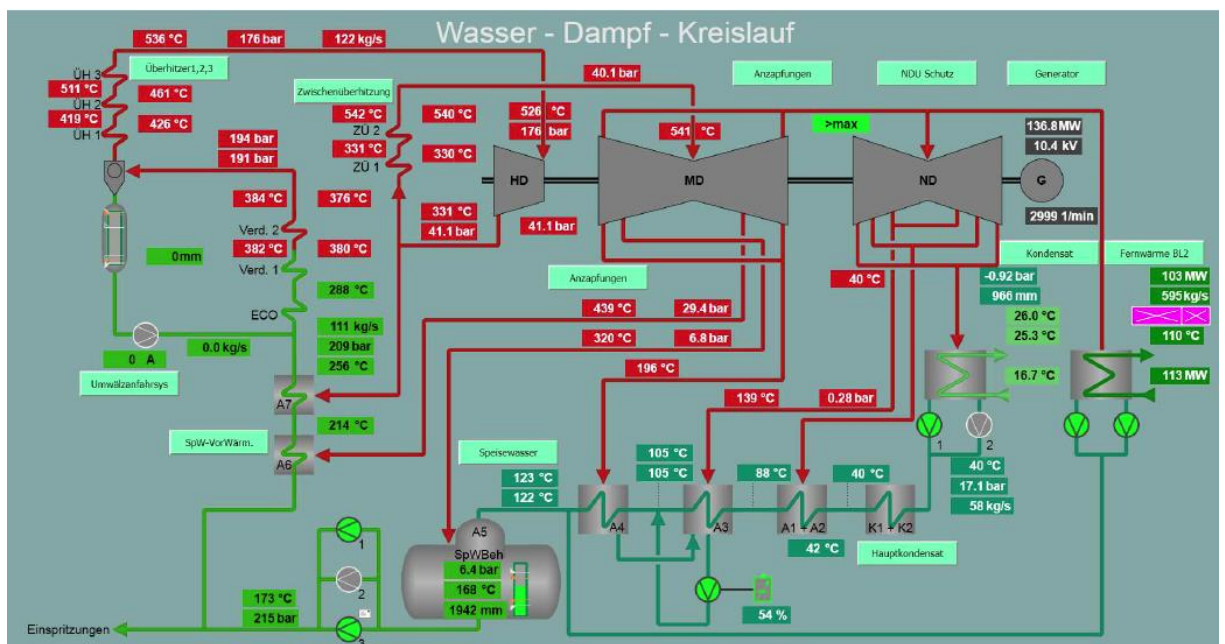
Obr. 7 Základní obrazovka [zdroj: vlastní tvorba]

Tabulka 5 platí i pro panel, adresy musí být shodné se STEP 7. Na symbolismu není závislost, nicméně pro přehlednost je vhodnější zachovávat stejné symboly.

### 2.2.5 Praktické využití

U vyšších řad PLC od SIEMENS S7-300, S7-400 bývá ethernet modul součástí automatu. Složité systémy, jako je například elektrárna, kde jsou na sobě závislé jednotlivé procesy, využívají informace jednotlivých podsystémů. Není prakticky možné, aby se elektrárna ovládala z jednotlivých stanovišť. K tomu slouží právě operátorské stanoviště, kde má operátor k dispozici informace ze všech částí elektrárny a v případě potřeby může zasáhnout do tohoto procesu.

Jednotlivé kontrolery jsou spojeny pomocí industrial ethernet (průmyslové sítě). Bývá to uzavřená síť, ke které není možné připojení pomocí internetu, právě z bezpečnostních důvodů. V dnešní době se využívá k propojení optických kabelů, díky jejich rychlosti a odolnosti vůči rušení datového toku.



Obr. 8 přehledová obrazovka oběhu vody a páry z elektrárny HKW West Wolfsburg vytvořeno programem PGtool [zdroj: Firmitní materiály společnosti ABB s.r.o. [5]]

Příklad přehledové obrazovky z operátorského stanoviště elektrárny ukazuje obr. 8. Jedná se o obrázek z živého procesu s načtenými hodnotami senzorů a čidel. Jak je vidět, snímají se například teplota, průtok, tlak, otáčky atd. Detaily jednotlivých částí jsou pak podrobněji zobrazeny na obrazovkách příslušných částí. Obrazovka byla vytvořena programem PGtool, který využívá společnost ABB s.r.o. k vytvoření grafických reprezentací pro operátorská stanoviště. Jedná se o konkurenční firmu společnosti SIEMENS.

### 3 Identifikace vstupů a výstupů modelu

#### 3.1 Zadání úlohy

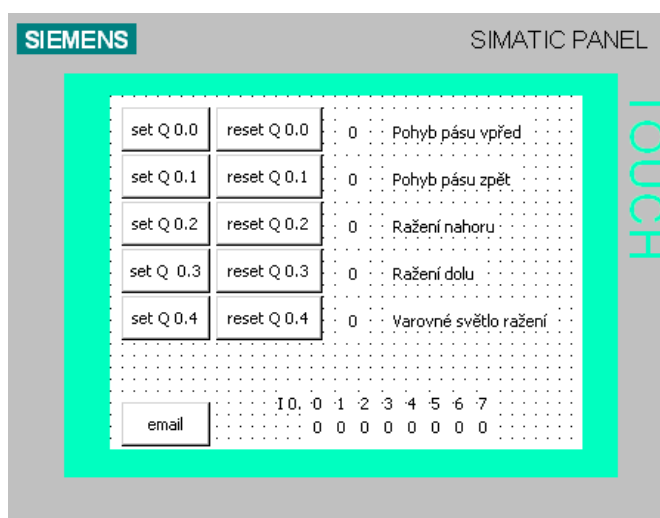
PLC je osazeno vodiči, které jsou připojeny k modelu pomocí LPT konektoru. Cílem je zjistit ke kterým vstupům na modelu jsou připojené jednotlivé výstupy PLC. A ke kterým vstupům na PLC jsou připojené jednotlivé výstupy modelu.

#### 3.2 Řešení

##### 3.2.1 Identifikace vstupů modelu

Vstupy modelu ovládají jednotlivé funkce, jako například pohyb pásu. Pro snadnou identifikaci postačí program ovládající výstupy PLC. Je důležité mít k dispozici možnost rychlého odpojení tohoto výstupu, aby se předešlo poškození modelu v případě, že se chodem dostane do mezního stavu.

PLC má zapojeno 5 výstupů, program tedy musí být schopen ovládat tyto výstupy jednotlivě. Upravením nastavení ovládacího panelu z předešlé úlohy se zajistí snadné ovládání. Přidáním nové obrazovky se zpřehlední uživatelské prostředí HMI. PLC dovoluje nastavovat hodnotu výstupu přímo z panelu. Proto se nemusí vytvářet program v STEP 7, ale adresa bitu výstupu se přiřadí přímo k tlačítku. Při takové úpravě nejsou stavy modelu nijak ošetřeny, a proto se při zjišťování stavu musí postupovat opatrně. Na obrázku 9 je vzor obrazovky pro identifikaci vstupů a výstupů modelu. Změnou výstupního bitu dojde k sepnutí části modelu. Bezpečný postup je sepnutí vždy jen jednoho bitu a sledovat změny na modelu. Jednotlivé výstupní bity mají popis činnosti modelu přímo v obrázku 9.

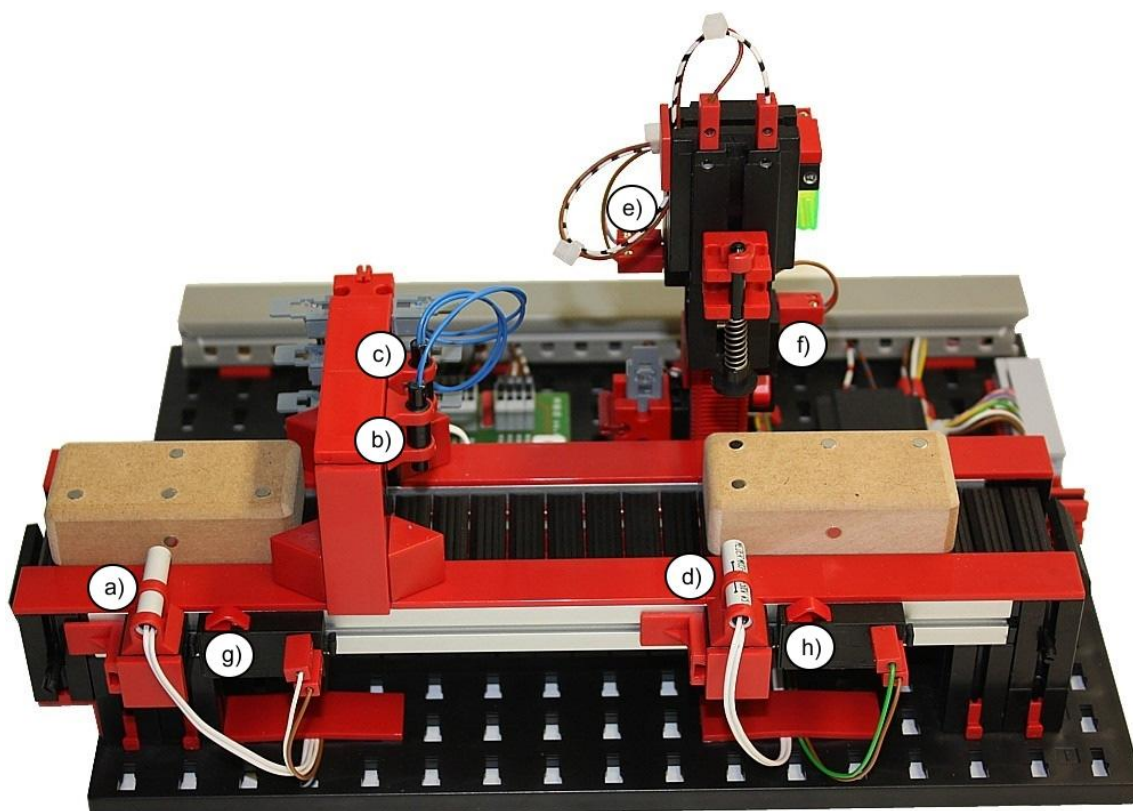


Obr. 9 Ovládání výstupů [zdroj: vlastní tvorba]

### 3.2.2 Identifikace vstupů PLC

Když jsou známy možnosti ovládání modelu, je důležité zjistit, na které vstupy PLC jsou připojeny senzory. Tyto vstupy lze identifikovat dvěma způsoby. Sledováním indikačních diod na vstupech PLC nebo, jak znázorňuje obrázek 9 zobrazením na panelu. Základem je spuštění každého senzoru zvlášť. Na tomto modelu jsou dva typy senzorů. Elektromagnetické a kontaktní senzory. Obrázek 10 ilustruje umístění senzorů na modelu. Elektromagnetické senzory a), d) snímají polohu výrobku na vstupu a u ražení a senzory b), c) snímají kód výrobku. Kontaktní senzory e), f) indikují mezní polohy razícího ramena. A k indikaci aktivace analogových tlačítek zajišťují g), h).

Spuštěním pohybu razícího ramena nahoru se identifikuje vstup senzoru horní polohy a pohybem dolů sensor dolní polohy. Posuvem výrobku přes senzory vstupu, čtení a polohy u ražení sepnou vstupy PLC, na které jsou tyto senzory připojeny. Výsledek se nachází tabulce 6, s přiřazenými vstupy na PLC k senzorům podle umístění v obrázku 10.



Obr. 10 Model s označenými senzory [zdroj: [www.staudinger-est.de](http://www.staudinger-est.de) s vlastní úpravou]

Vstup	Senzor	Vstup	Senzor
I0.0	g)	I0.4	c)
I0.1	h)	I0.5	b)
I0.2	a)	I0.6	e)
I0.3	d)	I0.7	f)

Tab. 6 Tabulka senzorů [zdroj: vlastní tvorba]

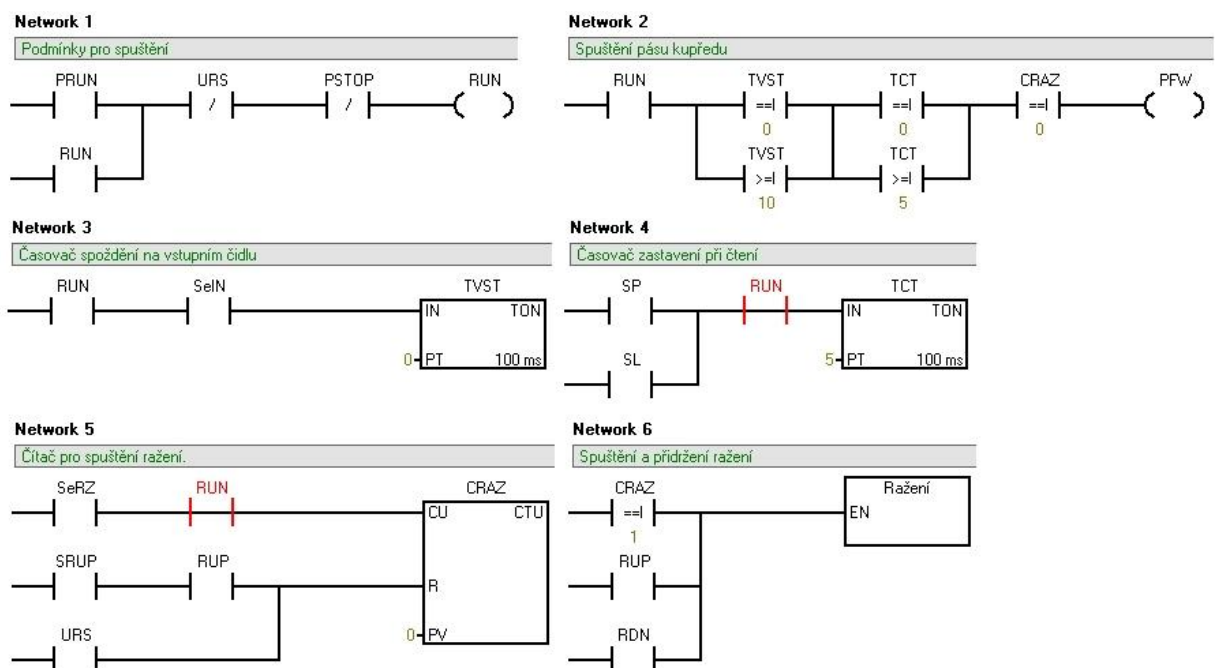
## 4 Základní program

### 4.1 Zadání úlohy

Základní chování programu, zastavování běhu pásu. Při sepnutí čidla vstupu zastavit chod pásu na 1s. Zastavit pás po dobu 500ms při kontaktu se senzory čtení. Sepnutím senzoru u razicího zařízení se pás zastaví a spustí se proces ražení. Ražení znamená spuštění razicí hlavice do spodní pozice a poté navrácení do horní pozice. Při ražení bude blikat varovné světlo ražení. Realizace ovládání pomocí dotykového panelu bude mít k dispozici START pro spuštění programu a STOP pro zastavení.

### 4.2 Řešení

#### 4.2.1 Program pro PLC



Obr. 11 Network 1 až Network 6 hlavní program [zdroj: vlastní tvorba]

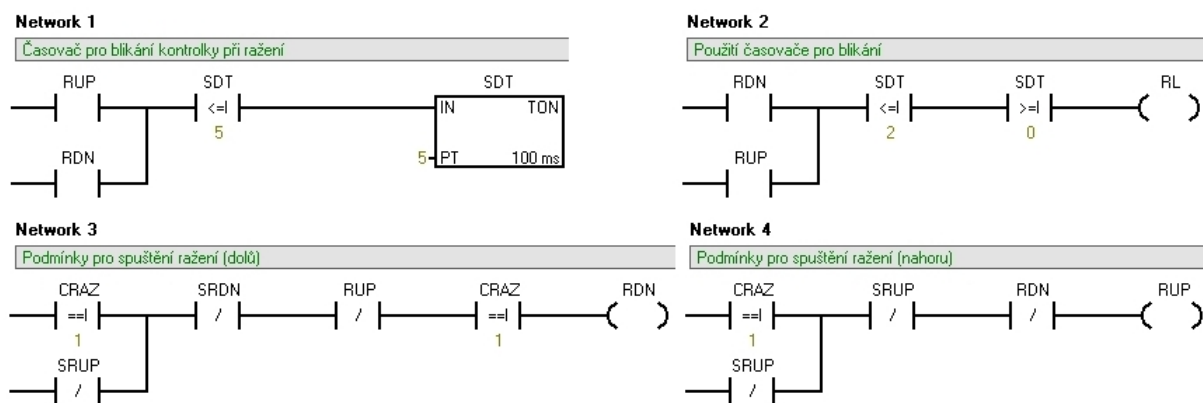
Základní program se dělí na dvě části, hlavní program viz obr. 11 a podprogram ražení viz obr. 12. V hlavním programu Network 1 nastavuje pomocný bit RUN, který zajišťuje běh programu. Bitem PRUN se RUN aktivuje, bity pro reset URS a zastavení PSTOP vypíná. Jediný žádoucí posun výrobku je pouze pohybem pásu. K zajištění proti spuštění části programu vlivem nechtěného pohybu výrobku, je potřeba umístit kontakt s bitem RUN do každého Networku, který zajišťuje aktivaci prvku modelu. To znamená do Networku 5 za kontakt senzoru ražení, kde zamezí přičtení hodnoty čítače a aktivaci



podprogramu ražení. Zamezením spuštění časovačů v Networkích 3 a 4 se zajistí rychlý průběh cyklu programu v stop režimu. Umístění tohoto spínače je na obrázku 11 vyznačeno červeně. Chybovým a nežádoucím stavům se musí předcházet správným návrhem programu.

Network 2 zajišťuje spuštění a zastavení pásu. Porovnání hodnot časovačů TVST a TCT, zastavují pohyb pásu na požadovanou dobu, při sepnutí senzorů SeIN a SP/SL. Časovače mají krok 100ms, proto k zastavení pásu na 1s musí být porovnání k deseti krokům TVST = 10 x 100ms. Porovnání CRAZ = 0 vypíná pohyb pásu při ražení.

Networky 3 a 4 spouští časovače TVST pro vstupní senzor SeIN a TCT pro senzory čtení SP/SL. V Networku 4 se pro spuštění TCT musí počítat s možností sepnutí pouze jednoho senzoru. Network 6, při hodnotě čítače CRAZ = 1, aktivuje podprogram ražení, který je na obrázku 12. Pokud se neošetří podmínky vypnutí podprogramu, může nastat problém v průběhu podprogramu. Z důvodu chování podprogramů, kdy se při odpojení napájení neresetují výstupní bity, ale zachovávají se hodnoty při odpojení. Bez spínačů výstupů RUP a RDN v Networku 6, v případě resetu čítače CRAZ uprostřed razícího cyklu, dojde k zachování hodnoty výstupního bitu. Tím by zůstal sepnut výstup RUP nebo RDN, které ovládají pohyb razící hlavice, a mohlo by dojít k poškození modelu. Proto je důležité mít k dispozici tyto přídržné kontakty nebo přiřadit požadované hodnoty v případě vypnutí podprogramu jiným způsobem. Není-li k podprogramu připojeno napájení, dojde k jeho přeskočení v průběhu programu.



Obr. 12 Podprogram ražení [Zdroj: vlastní tvorba]

Vytvoření podprogramu je vhodné z několika důvodů. Vyjmutím části programu z hlavního programu a vložením do podprogramu zpřehledňuje program. Pokud se část programu využívá jen za určitých podmínek, tak vložením do podprogramu se zrychlí

program, protože není tato část vyvolávána v každém cyklu. Je možné opakovaně použít části programu bez potřeby kopírování příkazů, ale pomocí vyvolání podprogramu.

Networky 1 a 2 v podprogramu ražení z obrázku 12 zajišťují blikání světelné signalizace chodu ražení pomocí spínačů, časovače, porovnání a výstupního bitu RL. Spuštění časovače SDT se realizuje aktivací pohonu razící hlavice a resetuje se každých 500ms. Networky 3 a 4 zajišťují chod ražení dolů a nahoru. Podmínky jsou pomocí spínačů, negací a porovnání čítače CRAZ nastaveny tak, aby v případě resetování ražení došlo k zastavení chodu dolů a navrácení do výchozí horní pozice.

#### 4.2.2 Program HMI

Tato úloha se soustředí na program v PLC, ovládací prvky jsou pouze START, STOP a RESET. Postačí tedy vytvoření čtyř tlačítek na dotykovém panelu, kde START nastavuje bit PRUN, STOP bit PSTOP, RESET bit CRES a TRANSFER. Tabulka 7 znázorňuje adresy použité v tomto příkladu, je vhodné zahrnout do tabulky i nevyužité vstupy a výstupy.

Symbol	Adresa	Popis	Symbol	Adresa	Popis
AIN	I0.0	Analogové tlačítko vstup	RDN	Q0.3	Ražení chod dolů
AOUT	I0.1	Analogové tlačítko výstup	RL	Q0.4	Ražení kontrolka
SeIN	I0.2	Senzor na vstupu	RUN	M1.0	Bit pro spuštění
SeRZ	I0.3	Senzor u ražení	CRES	M3.1	Úplný reset
SP	I0.4	Pravý senzor kódování	PRUN	M4.0	Spuštění z HMI
SL	I0.5	Levý senzor kódování	PSTOP	M4.1	Vypnutí z HMI
SRUP	I0.6	Ražení nahoře	TVST	T37	Časovač vstupního senzoru
SRDN	I0.7	Ražení dole	TCT	T38	Časovač na senzorech čtení
PFW	Q0.0	Pás chod vpřed	SDT	T39	Časovač pro blikání
PBCK	Q0.1	Pás chod vzad	CRAZ	C2	Čítač pro spuštění ražení
RUP	Q0.2	Ražení chod nahoru			

Tab. 7 Tabulka symbolů a adres pro základní úlohu [zdroj: vlastní tvorba]

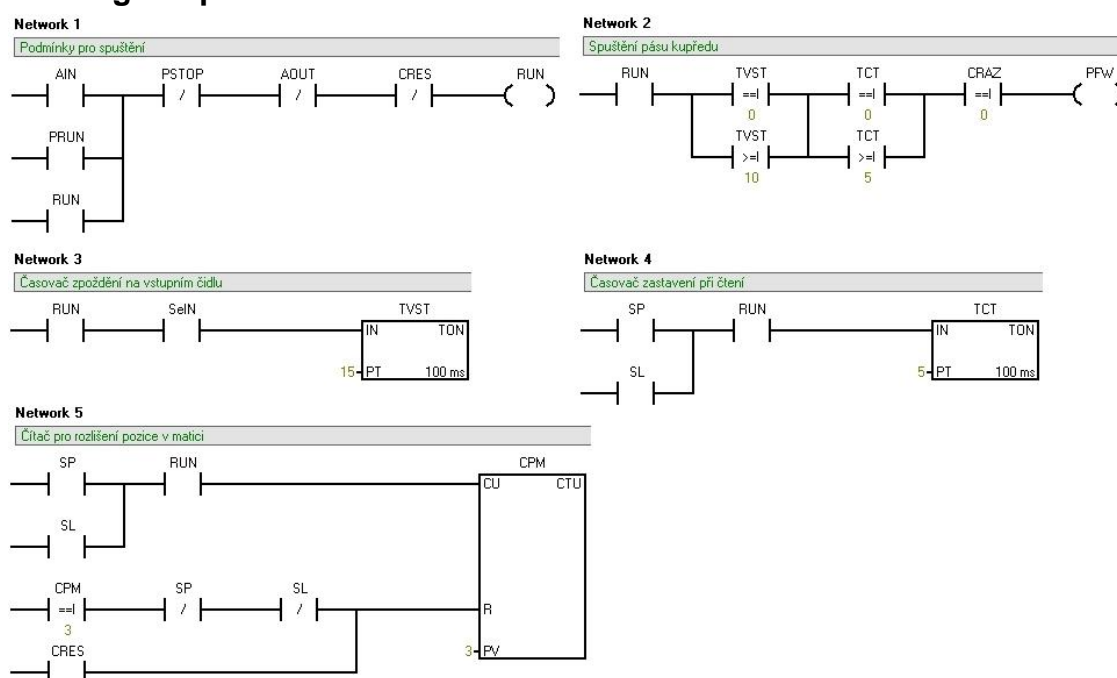
## 5 Program pro čtení kódu výrobku

### 5.1 Zadání úlohy

Rozšířit předchozí program o načítání kódu výrobku a zobrazení načteného kódu na panelu. Použít analogová tlačítka modelu připojená na vstupy PLC I0.0 a I0.1, pro spuštění a nouzové zastavení.

### 5.2 Řešení

#### 5.2.1 Program pro PLC



Obr. 13 Network 1 - Network 5 [zdroj: vlastní tvorba]

Pro využití analogových tlačítek postačí upravit Network 1, přidáním spínačů s bity AIN a AOUT na spuštění a zastavení programu. Networky 2 až 4 jsou shodné se základním programem.

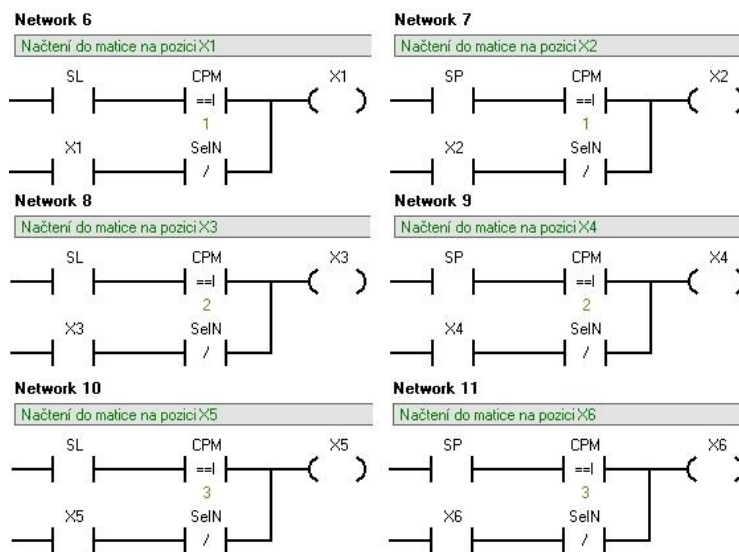
Network 5 z obrázku 13 je klíčový pro načtení kódu. Kód na výrobku se dá

představit jako matice  $\begin{bmatrix} X_5 & X_6 \\ X_3 & X_4 \\ X_1 & X_2 \end{bmatrix}$ . Cílem je tedy vytvořit souřadnicový systém načítání bitů

matice. Kde načtená matice je ve skutečnosti BYTE, tedy 8 bitů, jež se dá představit jako řada  $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8$ . Kde bity  $X_7$  a  $X_8$  jsou nevyužity a tedy vždy 0. Čítač CPM určuje řadu v matici a senzory SP a SL sloupec při načítání. Spínače SP, SL a RUN zajišťují, že se hodnota čítače zvýší o 1 při sepnutí obou nebo jednoho ze senzorů čtení v RUN módu.

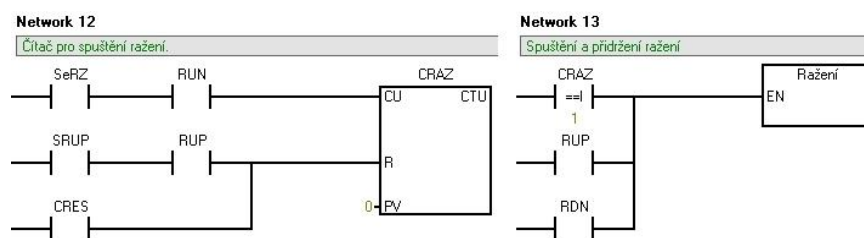
A čítač se resetuje při vypnutí senzorů po načtení třetí řady, nebo nastavením resetovacího bitu CRES.

Obrázek 14 ukazuje způsob nastavení bitů matice. Networky 6 - 11 přiřazují hodnoty bitům X1 - X6. Při sepnutí levého senzoru čtení SL a hodnotě čítače CPM = 1 se nastaví bit X1 = 1 viz Network 6. Pro zachování hodnoty bitu je použit negovaný spínač vstupního senzoru SeIN, který zajišťuje reset bitu, při sepnutí vstupního senzoru. Bity X2 - X6 se nastavují stejným způsobem - viz Networky 7 - 11. Symboly přiřazené adresám jsou v tabulce 8.



Obr. 14 Přiřazení hodnot do matice [zdroj: vlastní tvorba]

Networky 12 a 13 jsou shodné s Networky 5 a 6 z předešlého příkladu. Spuštění ražení a podprogram ražení se v tomto případě nastavuje shodně s předešlým programem.



Obr. 15 Ražení [zdroj: vlastní tvorba]

### 5.2.2 Program pro HMI

Tlačítka START, STOP a RESET nastavují stejné bity jako v předchozím příkladu. Problém nastává při volbě způsobu grafického zobrazení. Úpravou programu PLC pro přiřazení pomocného bitu ke každému kódu výrobku a využitím tohoto bitu k aktivaci grafiky na panelu znázorňující daný kód, se dá tato úloha realizovat. Pro nastavení pomocných bitů

lze využít, buď série spínačů a negací znázorňující hodnoty jednotlivých bitů kódu, například

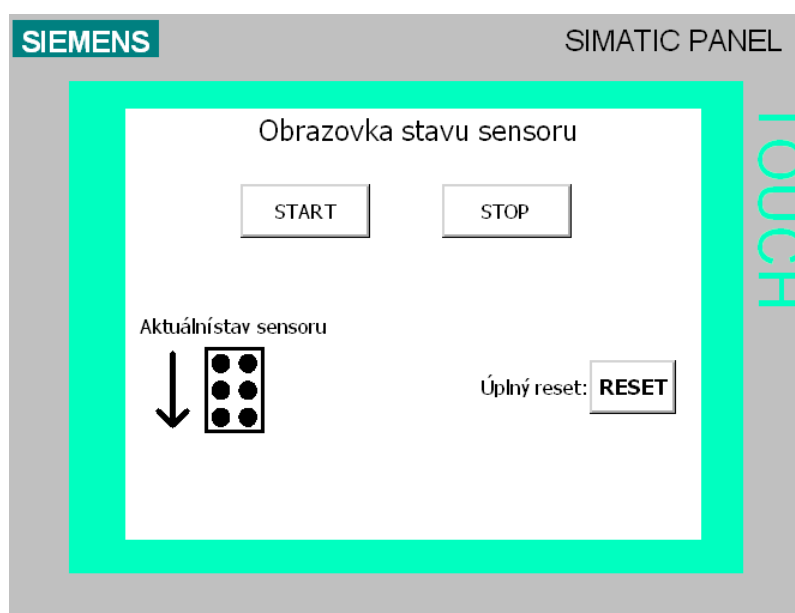
$$|X1| |\overline{X2}| |X3| |X4| |X5| |\overline{X6}| = \begin{vmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{vmatrix}, \text{ nebo porovnání. Číselná hodnota kódového BYTE}$$

se může aplikovat do funkce porovnání jako decimální číslo, které bude jedinečné pro každý kód. Je třeba mít na paměti, že výrobek může být na pás vložen dvěma směry a tudíž bude mít jeden výrobek dva kódy. Tento fakt se musí zohlednit v programu vložením číselného porovnání obou kódů, pro nastavení pomocného bitu daného výrobku. Tato řešení však nejsou univerzální, protože zobrazí pouze nastavené kódy a v případě přidání nových kódů, se musí zasáhnout do programů PLC a HMI k jejich zobrazení.

Ideální způsob zobrazení kódů je tedy takový, který nevyžaduje vlastní nastavení v programu PLC, ani jednotlivých grafických reprezentací na panelu. V programu PLC se již nachází potřebná data v podobě jednotlivých bitů kódu, cílem je tedy jejich znázornění na displeji. Toho lze docílit vytvořením dynamických grafických prvků znázorňujících pozice magnetů na výrobku, umístěných podle stejné matice jako v programu pro načítání. Grafika aktuálního stavu senzoru na obrázku 16, má černé tečky nastavené pro zobrazení na panelu

pokud je přiřazený bit 1. Bity jsou přiřazeny podle matice  $\begin{vmatrix} X_5 & X_6 \\ X_3 & X_4 \\ X_1 & X_2 \end{vmatrix}$ . Adresace použitých

symbolů je v tabulce 8.



Obr. 16 Obrazovka ovládání [zdroj: vlastní tvorba]

Symbol	Adresa	Popis	Symbol	Adresa	Popis
AIN	I0.0	Analogové tlačítko vstup	X3	M0.2	Pozice v matici
AOUT	I0.1	Analogové tlačítko výstup	X4	M0.3	Pozice v matici
SeIN	I0.2	Senzor na vstupu	X5	M0.4	Pozice v matici
SeRZ	I0.3	Senzor u ražení	X6	M0.5	Pozice v matici
SP	I0.4	Pravý senzor kódování	RUN	M1.0	Bit pro spuštění
SL	I0.5	Levý senzor kódování	CRES	M3.1	Úplný reset
SRUP	I0.6	Ražení nahoře	PRUN	M4.0	Spuštění z HMI
SRDN	I0.7	Ražení dole	PSTOP	M4.1	Vypnutí z HMI
PFW	Q0.0	Pás chod vpřed	TVST	T37	Časovač vstupního senzoru
PBCK	Q0.1	Pás chod vzad	TCT	T38	Časovač na senzorech čtení
RUP	Q0.2	Ražení chod nahoru	SDT	T39	Časovač pro blikání
RDN	Q0.3	Ražení chod dolů	TPM	T40	Časovač pro načtení
RL	Q0.4	Ražení kontrolka	CPM	C1	Čítač pozice v matici
X1	M0.0	Pozice v matici	CRAZ	C2	Čítač pro spuštění ražení
X2	M0.1	Pozice v matici			

Tab. 8 Tabulka symbolů a adres pro čtení [zdroj: vlastní tvorba]

## 6 Paměť

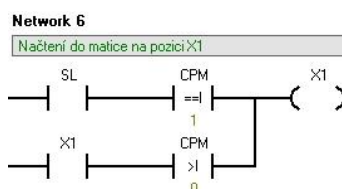
### 6.1 Zadání úlohy

Rozšířit předchozí úlohy o paměť načtených výrobků. Zobrazit na panelu kódy tří předchozích výrobků.

### 6.2 Řešení

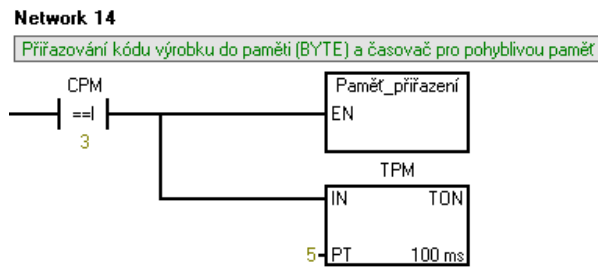
#### 6.2.1 Program pro PLC

Oproti předešlému programu je upraven způsob resetování bitů načítání. BYTE 0, do kterého je načítán kód, již není využíván pro zobrazení po načtení, ale pouze v průběhu načítání. K resetu bitu dochází společně s resetem čítače pozice v matici CPM - viz obr. 17. Stejná úprava platí pro ostatní bity matice.

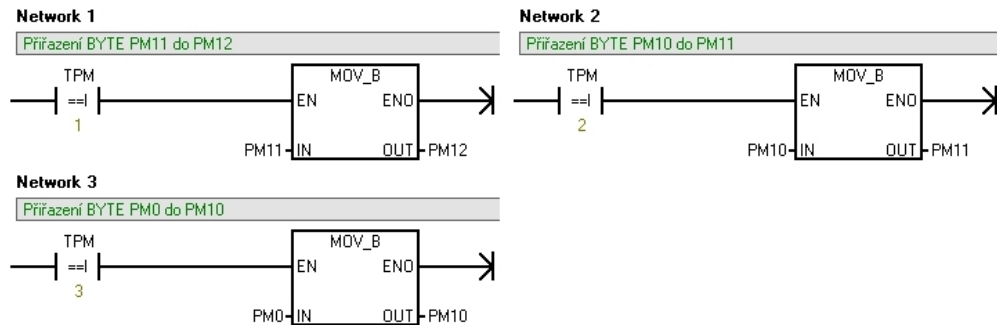


Obr. 17 Network 6 hlavního programu [zdroj: vlastní tvorba]

Pro zobrazení kódů předchozích výrobků se musí vytvořit pohyblivá paměť. Za tímto účelem se může využít čítač a přiřazování hodnoty, které bude pomocí čítače cyklovat, což nedovoluje přehledné zobrazení, jelikož aktuální výrobek nebude vždy na stejné pozici. Efektivnější způsob zajišťuje využití časovače a přiřazení hodnot. Obrázek 18 ukazuje časovač spolu s podmínkou spuštění podprogramu paměti přiřazení. Vzhledem k tomu, že reset čítače CPM nastává po dokončení čtení, je vhodný k použití pro aktivaci, to zajistí jen krátkou dobu spuštění podprogramu a časovače. V Networku 3 z obrázku 19 se přiřazuje načteného BYTE PM0 do paměťového BYTE PM10. Přiřazení hodnoty BYTE MOV\_B z Networků 2 a 1 posouvají hodnotu na paměťové BYTE PM11 a PM12. Časovač, bez kterého by se všechny paměťové BYTE přepsaly na načtenou hodnotu, zajišťuje správné pořadí přiřazení tedy PM11 -> PM12 poté PM10 -> PM11 a jako poslední PM0 -> PM10. Vytvořením podprogramu přiřazení paměti se zajistí zpřehlednění a zrychlení programu. Není nutné přiřazovat hodnoty v každém cyklu, ale jen při změně, která nastává po načtení kódu.

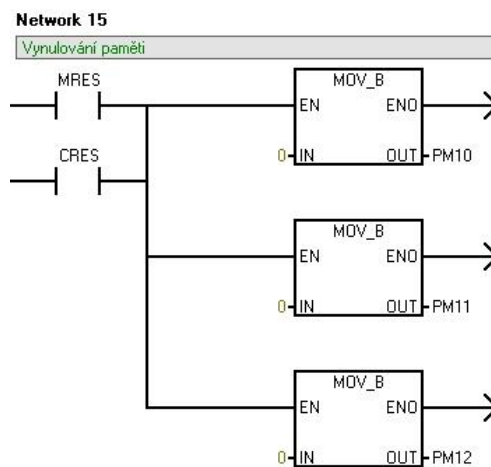


Obr. 18 Hlavní program, spuštění podprogramu přiřazení paměti [zdroj: vlastní tvorba]



Obr. 19 Podprogram přiřazení paměti [zdroj: vlastní tvorba]

Přiřazením nuly do paměťových BYTE se realizuje vynulování paměti. Bit MRES resetuje pouze paměť a CRES je celkový reset, jak ukazuje obrázek 20.

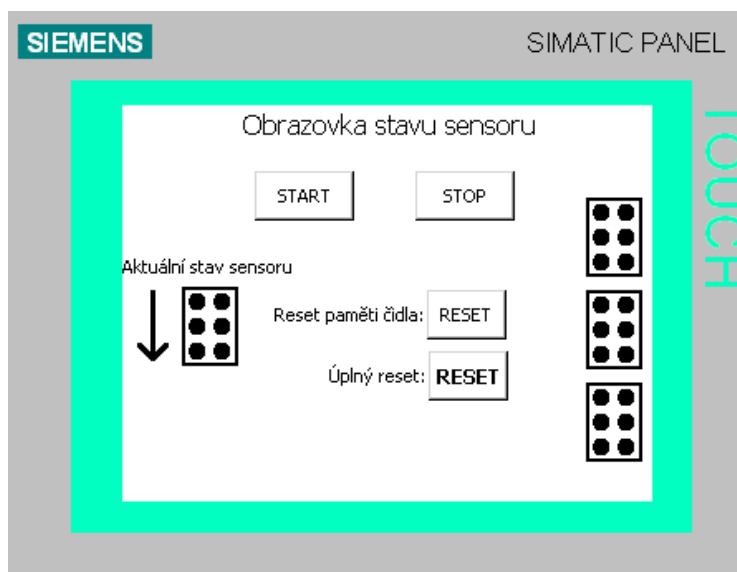


Obr. 20 Network 15 hlavního programu [zdroj: vlastní tvorba]

### 6.2.2 Program pro HMI

Na obrázku 21 se nachází obrazovka stavu senzoru, úprava ovládání oproti předešlé úloze spočívá v přidání tlačítka resetu paměti čidla. Zobrazení paměti je téměř stejné jako zobrazení průběhu načítání. Pro zobrazení každého z předešlých výrobků se musí načíst 6 adres jednotlivých bitů paměťových BYTE. V tabulce 9 jsou pouze adresy bitů a BYTE, které nebyly použity v předchozích příkladech.





Obr. 21 Ovládání programu [zdroj: vlastní tvorba]

Symbol	Adresa	Popis	Symbol	Adresa	Popis
PM0	MB0	BYTE pro načtení	X10_5	M10.4	Paměť pozice v matici
PM10	MB10	Paměť výrobku	X10_6	M10.5	Paměť pozice v matici
PM11	MB11	Paměť výrobku	X11_1	M11.0	Paměť pozice v matici
PM12	MB12	Paměť výrobku	X11_2	M11.1	Paměť pozice v matici
MRES	M1.1	Reset paměti	X11_3	M11.2	Paměť pozice v matici
X1	M0.0	Pozice v matici	X11_4	M11.3	Paměť pozice v matici
X2	M0.1	Pozice v matici	X11_5	M11.4	Paměť pozice v matici
X3	M0.2	Pozice v matici	X11_6	M11.5	Paměť pozice v matici
X4	M0.3	Pozice v matici	X12_1	M12.0	Paměť pozice v matici
X5	M0.4	Pozice v matici	X12_2	M12.1	Paměť pozice v matici
X6	M0.5	Pozice v matici	X12_3	M12.2	Paměť pozice v matici
X10_1	M10.0	Paměť pozice v matici	X12_4	M12.3	Paměť pozice v matici
X10_2	M10.1	Paměť pozice v matici	X12_5	M12.4	Paměť pozice v matici
X10_3	M10.2	Paměť pozice v matici	X12_6	M12.5	Paměť pozice v matici
X10_4	M10.3	Paměť pozice v matici			

Tab. 9 Tabulka symbolů a adres pro zobrazení paměti [zdroj: vlastní tvorba]

## 7 Program ražení

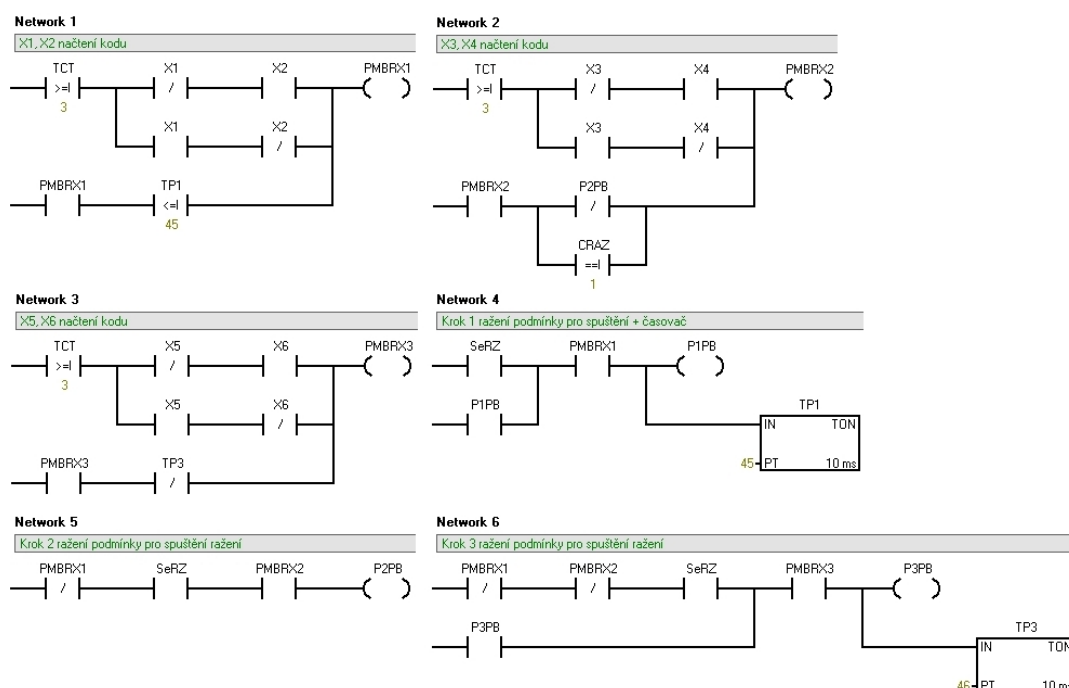
### 7.1 Zadání úlohy

Cílem je vytvoření podmínek pro spuštění ražení výrobku. Zajistit spuštění ražení v řadách, kde se nachází jen jeden magnet.

### 7.2 Řešení úlohy

#### 7.2.1 Program pro PLC, podprogram ražení

Jako první se musí vytvořit program, který rozhodne, ve které řadě dojde k ražení. Na obrázku 22 je podprogram nastavující pomocné paměťové bity používané v hlavní části programu k sepnutí ražení, pohybu pásu vpřed a vzad, za účelem ražení ve správné pozici na výrobku. Při sepnutí senzoru u ražení SeRZ se střed výrobku nachází pod razicí hlavicí, jedná se o výchozí pozici pro ražení. Požaduje-li se tedy ražení v první řadě výrobku, musí dojít ke spuštění zpětného chodu pásu a pro třetí řadu chod pásu vpřed. Jelikož nejsou na modelu senzory pro snímání polohy každé řady výrobku, použije se časovače, který zajistí, že se výrobek posune o potřebnou vzdálenost. Podprogram se dělí na dvě části, první tři Networky zjišťují, ve kterých řadách se provede ražení. A Network 4 - 6 nastavuje postup ražení a časovače pro posun vpřed a vzad.



Obr. 22 Program podmínek pro ražení [zdroj: vlastní tvorba]

Pomocné bity PMBRX1 - PMBRX3 jsou nastavovány pomocí bitů načtení X1 - X6. Požadavek zní provést ražení v řadách s jedním magnetem. V těchto případech se tedy musí

	X1	X2	PMBRX1	
nastavovat bity, jedná se v podstatě o pravdivostní tabulku	0	1	1	, negace a
	1	0	1	

spínače zajišťují splnění podmínek této tabulky - viz Network 1. Vlivem polohy výrobku na pásu nebo magnetů na výrobku nemusí dojít k sepnutí obou senzorů ve stejném okamžiku, což vede ke špatnému nastavení pomocného bitu. Zpoždění nastavení bitu porovnáním hodnoty časovače TCT zajišťuje nastavení pomocného bitu ražení PMBRX1 v ustáleném stavu, tedy při zastaveném pásu a magnety umístěny plně pod senzory. Stejně se nastavují pomocné bity ražení dvou zbylých řad - viz Networky 2 a 3. Jelikož se BYTE načtení MB0 resetuje po dokončení načítání, je hodnota bitu PMBRX1 přidržena pomocí kontaktu daného bitu a podmínkami pro reset. Resetování těchto bitů se provádí při spuštění ražení dané řady.

Bity P1PB - P3PB a časovače TP1 a TP3 použité v Networkích 4 - 6 spouští v hlavním programu ražení a chod pásu. Network 4 zajišťuje ražení první řady výrobku, to znamená, že musí zahájit chod pásu zpět na dobu 450ms, čímž se zajistí správná pozice pro ražení a poté spustit ražení. Kontakt výrobku se senzorem ražení SeRZ v případě, že první řada výrobku vyžaduje ražení (PMBRX1), nastaví bit P1PB a spustí časovač TP1. Kontakt P1PB přidržuje bit a chod časovače. Po uplynutí doby  $46 \times 10 \text{ms} = 460 \text{ms}$  se v Networku 1 resetuje bit PMBRX1, čímž dojde k vypnutí časovače TP1 a resetu bitu P1PB.

Networkem 5 se realizuje ražení druhé řady, která je pod hlavicí ražení ve chvíli sepnutí senzoru ražení SeSZ. Negace PMBRX1 zajišťuje, že nedojde k nastavení bitu P2PB, pracuje-li Network 4. Je-li pomocný bit PMBRX1 deaktivován a pokud došlo k aktivaci pomocného bitu ražení druhé řady PMBRX2, pak se sepnutím kontaktu SeRZ nastaví bit P2PB. Při aktivaci P2PB se spustí ražení a po dokončení ražení se resetuje bit PMBRX2 viz Network 2.

Ražení třetí řady řeší Network 6, kde negace PMBRX1 a PMBRX2 nedovolují nastavení bitu P3PB a spuštění časovače TP3, pracuje-li Network 4 nebo Network 5. Sepnutím spínače SeRZ se aktivuje bit P3PB a časovač TP3, je-li aktivní bit PMBRX3. Opět dochází k vypnutí spínače SeRZ, protože výrobek musí dojet na pozici ražení třetí řady, k zamezení deaktivace bitu P3PB slouží přidržný kontakt. Při čase  $TP3=460 \text{ms}$  dochází k resetu PMBRX3 a deaktivaci časovače a bitu P3PB.

Z Networků 4 - 6 je zřejmé, že senzor ražení SeRZ, se používá pro identifikaci pozice výrobku u ražení a odtud se rozhoduje, zda se má spustit chod vzad, vpřed nebo razit

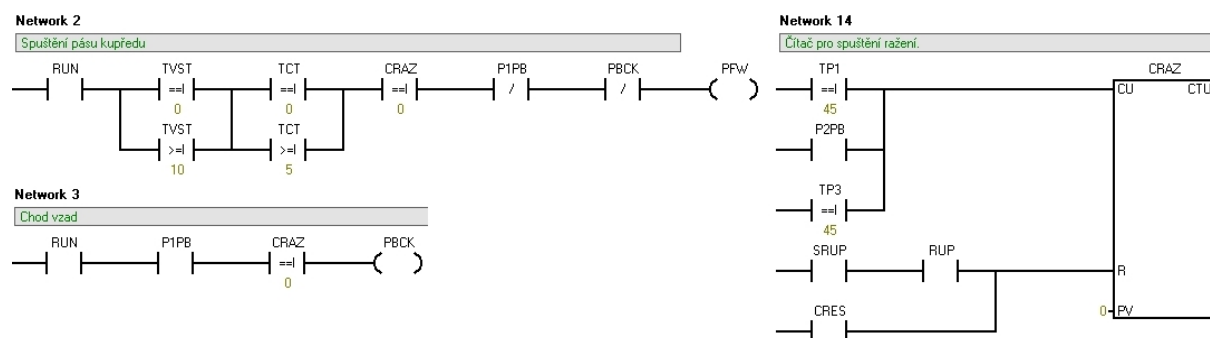
na současné pozici. Bity PMBRX1 - PMBRX3 se zajišťuje ražení vybrané části výrobku. Bude-li požadavek na upravení podmínek ražení, je nutný zásah pouze do Networků 1 - 3.

## 7.2.2 Program pro PLC, úpravy hlavního programu

Na obrázku 23 jsou upravené Networky z předchozího programu. Do Networku 2 jsou přidány negované kontakty P1PB a PBCCK a Network 3 z předchozího programu se odsunul na Network 4. Místo toho je vložena podmínka pro spuštění pásu vzad. Dále Network 14 má upravené podmínky pro spuštění čítače, který slouží jako spínač pro podprogram ražení.

Negace kontaktu P1PB prvotně deaktivuje chod vpřed při nastavení tohoto bitu v podprogramu ražení. Chod pásu vpřed je negací PBCCK zajištěn tak, aby nebylo možné spuštění obou stavů zároveň. Zpětný chod pásu nastavuje spínač P1PB a vypíná čítač ražení CRAZ = 1, případně deaktivací bitu RUN, dojde-li k vypnutí programu.

Spouštění čítače v Networku 14 je změněno, není aktivován senzorem ražení SeRZ. Ale časovačem TP1 pro první, bitem P2PB druhou a časovačem TP3 pro třetí řadu výrobku. Zbytek programu PLC a ovládání i zobrazení na HMI jsou shodné s předchozím příkladem. V tabulce 10 jsou pouze přidány adresy a symboly oproti předešlým příkladům.



Obr. 23 Úpravy hlavního programu [zdroj: vlastní tvorba]

Symbol	Adresa	Popis	Symbol	Adresa	Popis
PMBRX1	M2.0	Pomocný bit ražení pozice 1	P2PB	M2.4	Pozice 2 přídržný bit ražení
PMBRX2	M2.1	Pomocný bit ražení pozice 2	P3PB	M2.5	Pozice 3 přídržný bit ražení
PMBRX3	M2.2	Pomocný bit ražení pozice 3	TP1	T33	Časovač pozice 1
P1PB	M2.3	Pozice 1 přídržný bit ražení	TP3	T34	Časovač pozice 3

Tab. 10 Tabulka symbolů a adres podmínek ražení [zdroj: vlastní tvorba]

## 8 Zobrazení prvků na panelu

### 8.1 Zadání úlohy

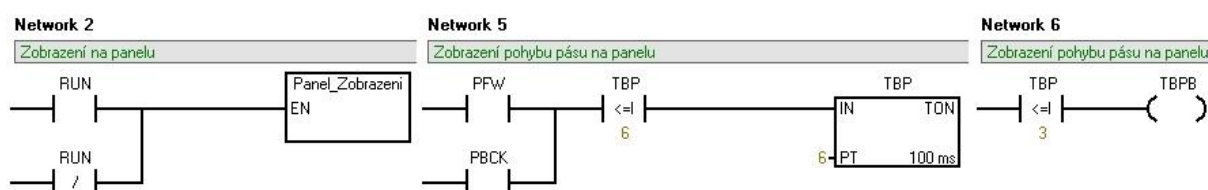
Zobrazit na dotykovém panelu počet zpracovaných výrobků. Jednotlivé výrobky s grafickou vizualizací kódu a celkový počet. Zobrazit prvky modelu, pohyb pásu, pozici výrobku při načítání kódu a ražení. Upravit program PLC pro umožnění zobrazení těchto informací. Vytvořit úvodní obrazovku, kde bude možnost aktivace přenosu dat, nastavení světlosti, kalibrace obrazovky a odkazy na jednotlivé obrazovky.

### 8.2 Řešení

#### 8.2.1 Program pro PLC

Pozici výrobku při načítání kódu lze zobrazit využitím čítače CPM a k signalizaci chodu razicího zařízení se mohou aplikovat vstupy PLC. Není proto nutné upravovat program pro tyto prvky.

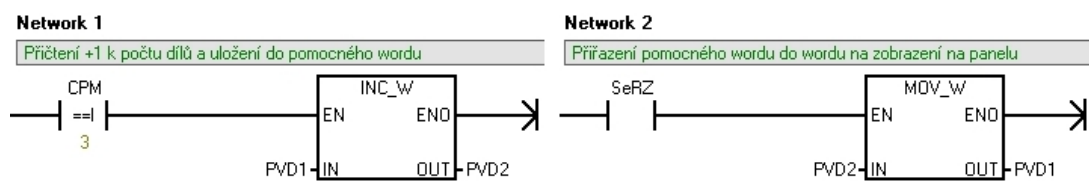
Ve WinCC flexible Micro nejsou k dispozici dynamické prvky pro znázornění chodu pásu, musí se tedy najít alternativní způsob. Za tímto účelem se do hlavního programu přidá časovač - viz obrázek 24 Networky 5 a 6, pro imitaci pohybu pásu viz program pro HMI. Network 5 spouští a resetuje časovač TBP při spuštění chodu pásu vpřed nebo vzad. A v Networku 6 se nastavuje pomocný paměťový bit TBPB, který je použit pro zobrazení na panelu. Cyklus je 600ms kde 300ms je bit aktivní a zbylý čas neaktivní. Reset časovače a nastavování bitu řeší porovnání.



Obr. 24 Přidané Networky do hlavního programu [zdroj: vlastní tvorba]

Network 2 udržuje aktivní podprogram Panel\_Zobrazeni, ve kterém se nachází zbylé funkce nutné pro zobrazení požadovaných prvků. Jak bylo ukázáno v druhé kapitole, je možné použít čítač pro přidání hodnoty. V této úloze jsou další dva způsoby zvyšování hodnoty. Výhodou těchto dvou metod je, že výstupní hodnota není nezbytně jen Integer, ale BYTE, WORD, DOUBLEWORD nebo REAL. Na obrázku 25 v Networku 1 a 2 podprogram zvyšuje hodnotu celkového množství výrobků. Network 1 říká, že při dokončení načítání kódu výrobku CPM = 3, dojde k navýšení hodnoty WORDu PVD1 o 1 a výsledná hodnota se

uloží do PVD2. A sepnutím senzoru polohy u ražení SeRZ, se hodnota PVD2 uloží do PVD1, jak zobrazuje Network 2, tím je zajištěn cyklus navyšování hodnoty. U funkce INC\_W se může použít shodný výstupní WORD a vstupní WORD adresa, je však problematické zajistit správnou funkci, neboť při každém cyklu programu dojde k navýšení o 1. To při špatném ošetření spouštěcí podmínky, může vést k výslednému navýšení o stovky až tisíce místo o jedna. Hodnotu PVD1 zobrazuje panel jako celkový počet výrobků. Je taktéž možné provést součet jednotlivých výrobků, což by při načtení jiného než nastaveného kódu vedlo ke špatné hodnotě.



Obr. 25 Podprogram Panel\_Zobrazení Network 1 a 2 [zdroj: vlastní tvorba]

Zbylou část podprogramu znázorňuje obrázek 26, kde Network 6 resetuje hodnoty WORDů přiřazením nuly, stejně jako u vynulování paměti. Networky 3 - 5 navyšují počet jednotlivých výrobků o 1 při splnění podmínek.

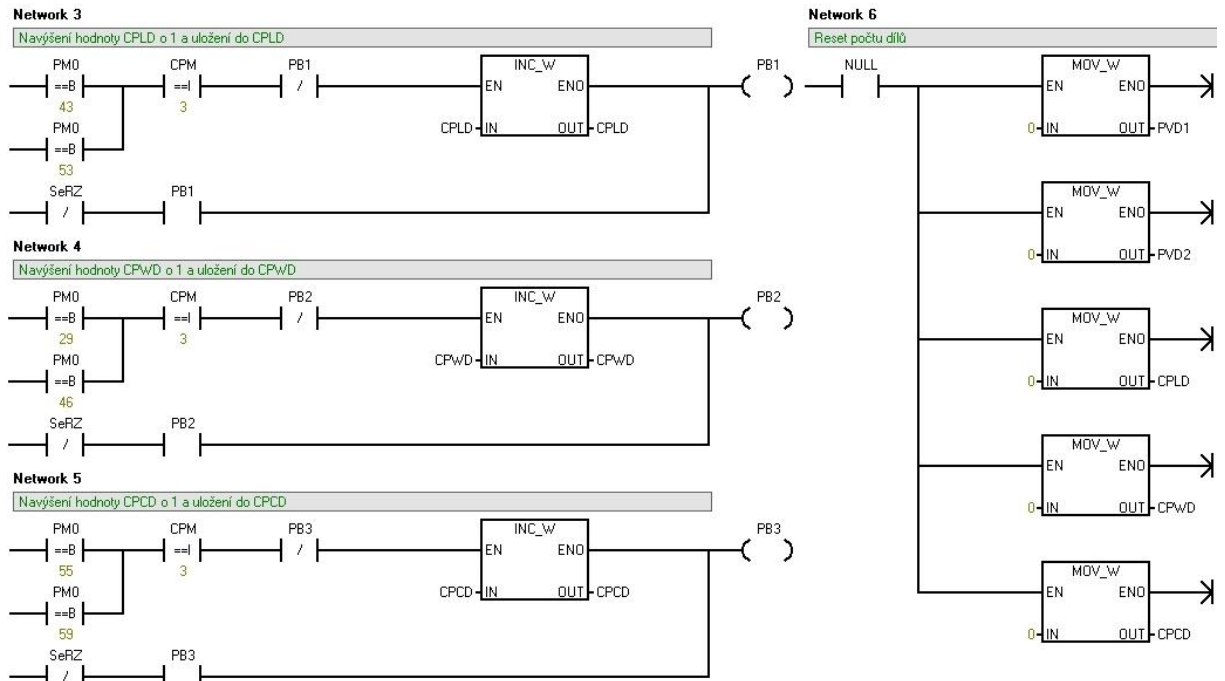
Nejprve je potřeba rozhodnout o způsobu rozlišení jednotlivých výrobků. V programu doposud takové rozlišení není, pouze schopnost zobrazit daný kód. Vzhledem k tomu, že se kód může zobrazit, jako dekadická hodnota nabízí se relativně snadné řešení. A to porovnání, je nutné mít opět na paměti, že výrobek může být do čtečícího zařízení vložen dvěma směry. To znamená, každý výrobek může mít dvě hodnoty. Ty se dají zjistit dvěma způsoby, výpočtem nebo programem, který jí zobrazí. Výpočet se provede převedením matice na řadu a poté převod z binárního čísla na dekadické například:

$$\begin{vmatrix} X_5 & X_6 \\ X_3 & X_4 \\ X_1 & X_2 \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 1 \end{vmatrix} \quad X_1 \times 1 + X_2 \times 2 + \dots + X_6 \times 32 = 1 + 2 + 0 + 8 + 0 + 32 = 43$$

Upravením stávajícího programu panelu tak, že zobrazí hodnotu BYTE M0, při načítání kódu, se dá hodnota zjistit bez výpočtu.

Zjištěné hodnoty každého výrobku se porovnají s paměťovým BYTE M0 viz Network 3 z obrázku 26. Porovnání čítače CPM při načtení třetí řady aktivuje INC\_W čímž se k WORDu CPLD přičte 1 a opět uloží do CPLD. Ihned po uložení se aktivuje pomocný bit PB1, který odpojí napájení k INC\_W, čímž se předejde nechtěnému navýšení. Tento bit musí být přidržen, alespoň po dobu kdy má čítač hodnotu 3. Bit je resetován po sepnutí senzoru

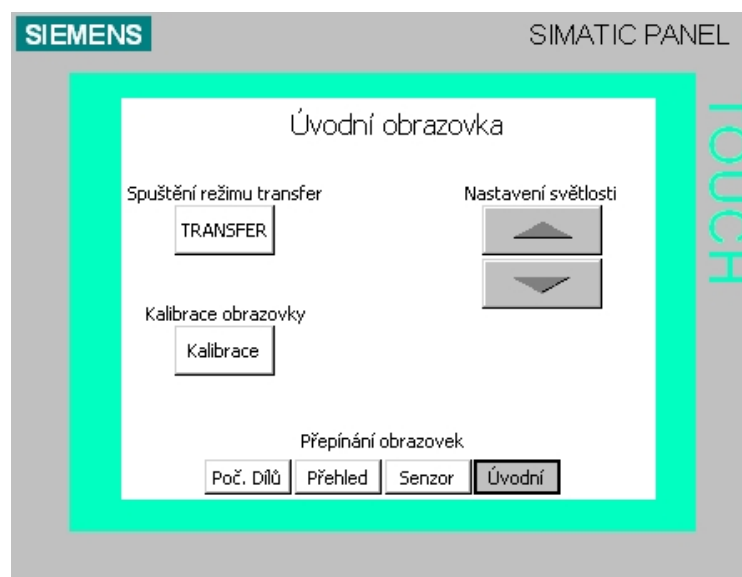
ražení SeRZ, čímž se celý network deaktivuje. Networky 4 a 5 nastavují stejným způsobem počet zbylých dvou výrobků. Hodnoty CPLD, CPWD a CPCD jsou zobrazovány na panelu.



Obr. 26 Podprogram Panel\_zobrazení Networky 3 - 6 [zdroj: vlastní tvorba]

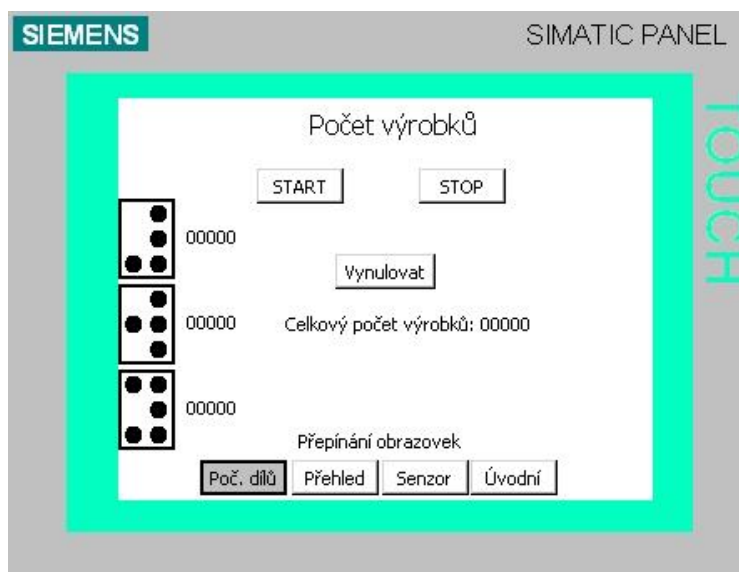
## 8.2.2 Program pro HMI

Na obrázku 27 je úvodní obrazovka s prvky Transfer pro přenos dat, Kalibrace a nastavení světlosti. Přepínání obrazovek je ve spodní části, označení aktivní obrazovky je řešeno statickým objektem. Tyto funkce se nachází v nastavení akce tlačítka, popis těchto základních nastavení řeší bakalářská práce Výukový mechatronický systém s PLC [6].



Obr. 27 Úvodní obrazovka [zdroj: vlastní tvorba]

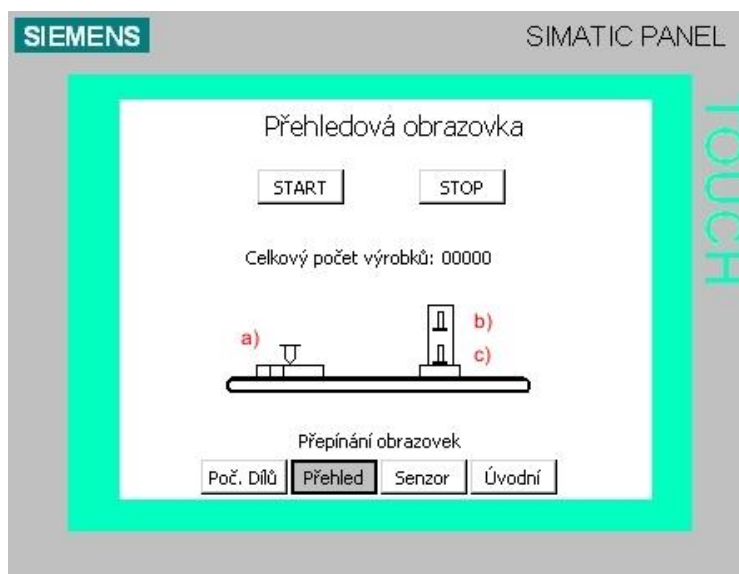
Na obrázku 28 se nachází obrazovka zobrazení počtu zpracovaných výrobků. Jako na všech pracovních obrazovkách jsou i zde ovládací prvky START a STOP. Grafické znázornění kódu výrobků jsou statické objekty. Dynamická textová pole u těchto objektů vypisují hodnoty množství zpracovaných výrobků z WORDů CPLD, CPWD a CPCD z programu PLC. Tlačítko vynulovat nastavuje bit symbolu NULL. A textové pole celkového počtu výrobků vypisuje hodnotu WORDu PVD1.



Obr. 28 Obrazovka počtu výrobků [zdroj: vlastní tvorba]

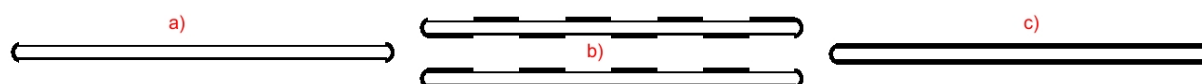
Přehledová obrazovka z obrázku 29 obsahuje základní ovládací prvky START a STOP, přepínání obrazovek a informaci o celkovém počtu zpracovaných výrobků. K identifikaci pozice výrobku při čtení jsou použity tři obdélníky s přiřazeným symbolem čítače CPM. Každý se zobrazuje podle nastavené hodnoty čítače 1 - 3 viz obrázek 29 a). Indikace ražení se realizuje obdobně, při hodnotě čítače ražení CRAZ = 0 se zobrazí pouze hlavice ražení - viz obrázek 29 b) a při hodnotě čítače CRAZ = 1 se hlavice zobrazí ve spodní poloze spolu s výrobkem - viz obrázek 29 c).





Obr. 29 Přehledová obrazovka [zdroj: vlastní tvorba]

Bit nastavovaný v programu PLC pro pohyb pásu se využívá k vytvoření optického klamu. Pás se skládá ze tří částí statická - viz obr 30 a) a dvě dynamické viz obr 30 b). Horní obrázek dynamické části se zobrazuje, pokud platí pomocný bit  $TBPB = 0$  a spodní část při  $TBPB = 1$ . Rychlou změnou těchto hodnot se vytvoří iluze pohybu. Problém může být v množství prvků, které se použije k vytvoření takového objektu. Z obrázku je vidět 20 prvků a maximální počet objektů na obrazovku je 50, proto se musí počítat s tímto faktem při návrhu obrazovky a vkládat složité dynamické objekty na zvláštní obrazovku. Tabulka 11 symbolů a adres opět rozšiřuje předešlé úlohy.



Obr. 30 Složení pohyblivého pásu [zdroj: vlastní tvorba]

Symbol	Adresa	Popis	Symbol	Adresa	Popis
TBPB	M1.2	Pomocný bit běhu pásu	PVD2	MW15	Počet dílů pomocný word
PB1	M1.3	Přidržený bit L dílu	CPLD	MW17	Celkový počet L dílů
PB2	M1.4	Přidržený bit W dílu	CPWD	MW19	Celkový počet W dílů
PB3	M1.5	Přidržený bit C dílu	CPCD	MW21	Celkový počet C dílů
NULL	M1.6	Vynulování počtu dílů	TBP	T42	Časovač běhu pásu
PVD1	MW13	Počet dílů			

Tab. 11 Tabulka symbolů a adres pro zobrazení na panelu [zdroj: vlastní tvorba]

## 9 Chybové hlášení

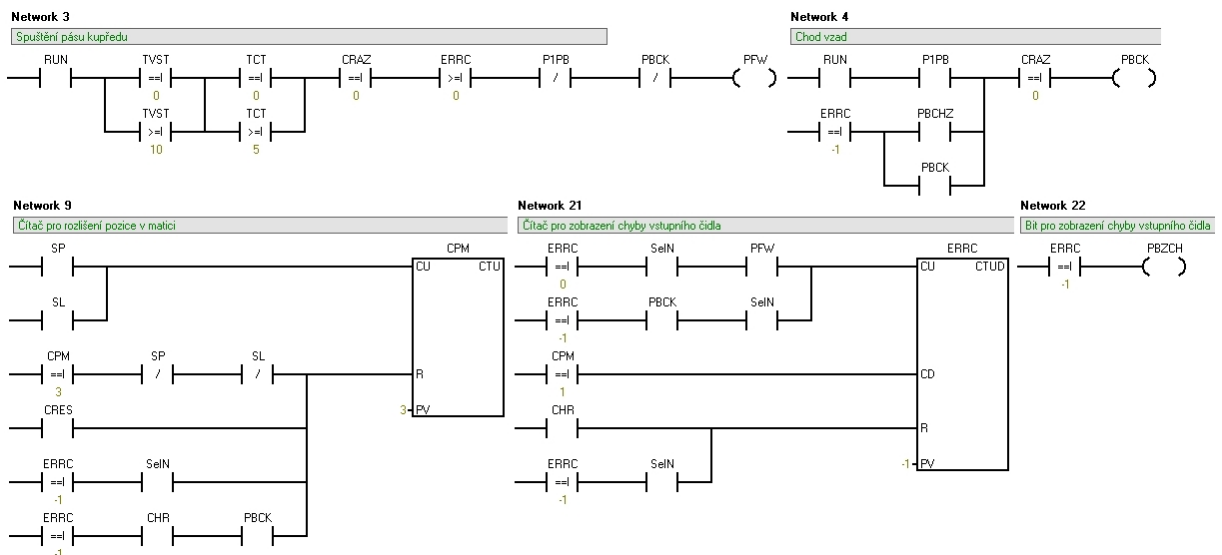
### 9.1 Zadání úlohy

Zobrazení chybového hlášení na panelu ve chvíli, kdy dojde k aktivaci senzorů čtení, bez předešlého sepnutí vstupního senzoru SeIN. Toto hlášení umožní volbu pokračovat, kdy program bude pokračovat ve čtení a běžném průběhu, a možnost pro opakování. Při volbě možnosti opakování dojde ke zpětnému chodu pásu, dokud nesejne vstupní senzor nebo stisk tlačítka pokračování.

### 9.2 Řešení

#### 9.2.1 Program pro PLC

V ideálním případě je systém zajištěn proti jakýmkoliv nežádoucím stavům. V reálném případě to však není možné a musí se tedy vytvořit varování, které obsluhu informuje o problému. Nejdříve se musí zjistit, kde mohou nastat chyby. Například pokud začne čtení kódu výrobku bez předchozího sepnutí vstupního senzoru SeIN, může to znamenat několik důvodů pro chybu. Vstupní senzor SeIN je poškozen, čtecí senzory udávají falešnou hodnotu, což značí jejich poškození, na výrobku je chybně umístěn magnet, který spíná vstupní senzor apod.



Obr. 31 Upravené networky v programu PLC [zdroj: vlastní tvorba]

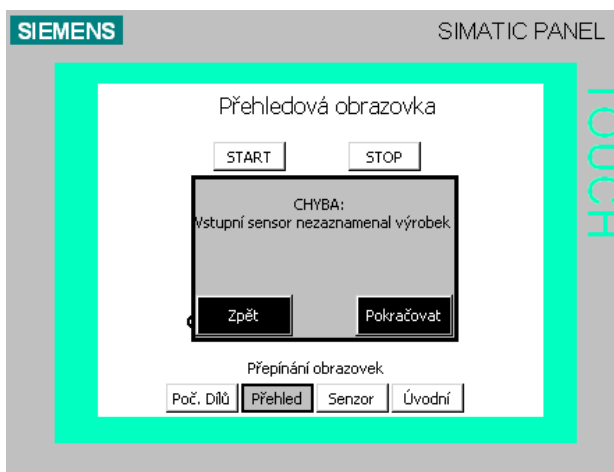
Pro aktivaci chybové hlášky se upraví program v PLC k nastavení pomocného bitu zobrazení PBZCH do 1, nastane-li nežádoucí stav. Zároveň se musí zajistit zastavení programu a určitých procesů, které by mohly zapříčinit chybu programu. Network 21 z

obrázku 31 pracuje s čítačem ERRC, který zastavuje proces programu a ovládání v chybovém stavu. Čítač má v klidu hodnotu  $ERRC = 0$ , sepnutím vstupního senzoru SeIN při pohybu pásu vpřed dojde ke změně hodnoty na  $ERRC = 1$ . Poté při načtení první řady kódu CPM se hodnota změní na  $ERRC = 0$ . Takto čítač funguje, pokud proces proběhne v pořádku a chyba se neprojeví.

Nesepne-li však vstupní senzor a dojde k načtení první řady kódu CPM, změní se hodnota čítače na  $ERRC = -1$ . To způsobí zobrazení chyby nastavením bitu PBZCH viz Network 22, a pás se zastaví, viz porovnání ERRC v Networku 3. Je-li zvolena volba pokračovat, nastaví se bit CHR, čítač se resetuje na  $ERRC = 0$  a program pokračuje v průběhu. Při volbě zpět dojde k nastavení bitu PBCHZ, tím se sepne zpětný chod pásu - viz Network 4. Tento stav trvá, dokud nesepne vstupní senzor SeIN nebo nestiskne pokračovat na panelu, čímž se čítače ERRC a CPM resetují a program opět pokračuje v procesu. Při sepnutí SeIN dojde k resetu čítačů CPM - viz Network 9 a ERRC - viz Network 21, po sepnutí chodu pásu vpřed se přičte 1 k ERRC tzn. bezchybný stav. Pokud se použije tlačítko pokračovat, resetují se oba čítače stejně, nicméně nepřičte se 1 k ERRC při rozběhnutí pásu vpřed a dojde tedy opět k zastavení programu při načtení první řady kódu výrobku.

### 9.2.2 Program pro HMI

Zobrazení chyby musí být zřejmé a efektivní, k tomu je vhodné vyskakovací okno. Při změně bitu PBZCH z 0 na 1 se zobrazí okno s chybou a ovládáním viz obrázek 32. Text a rámeček jsou statické objekty, nastavené na zobrazení při  $PBZCH = 1$ , taktéž tlačítka mají přiřazen tento bit pro jejich zobrazení. Tlačítko Pokračovat nastavuje bit CHR pro reset čítače a tlačítko Zpět nastavuje bit PBCHZ. Rozšíření adresy a symbolů se nachází v tabulce 12.



Obr. 32 Zobrazení chyby [zdroj:vlastní tvorba]

<b>Symbol</b>	<b>Adresa</b>	<b>Popis</b>	<b>Symbol</b>	<b>Adresa</b>	<b>Popis</b>
ERRC	C3	Čítač pro chybovou hlášku vstupního senzoru	PBZCH	M3.2	Pomocný bit zobrazení chyby
CHR	M3.0	Pomocný bit resetu čítače chybové hlášky	PBCHZ	M3.3	Pomocný bit chyby chod zpět

*Tab. 12 Tabulka symbolů a adres chyby [zdroj: vlastní tvorba]*

## Závěr

Diplomová práce se skládá ze tří částí, kde první popisuje vybrané programovací prvky k vytvoření programu pro PLC, druhá vysvětluje funkce ethernet modulu a třetí část obsahuje typické úlohy s řešením vysvětlující složitější operace s PLC.

Ethernet modul zajišťuje komunikaci mezi dílčími prvky systému pomocí datových linek. Zde použité PLC řady S7-200 od firmy SIEMENS, na rozdíl od vyšších řad, nemá integrovaný ethernet, tento nedostatek se řeší rozšiřujícím modulem CP243-1. V práci jsou popsány a vysvětleny některé funkce modulu CP243-1, jako je přenos dat mezi PLC a STEP 7-MicroWIN, odeslání nastavitelného emailu s proměnnou hodnotou a přístup k některým funkcím modulu pomocí internetu. Popis nastavení, úloha a příklad praktického využití tohoto modulu se nachází v druhé kapitole.

Třetí část práce obsahuje zadání a detailní řešení typických úloh k vypracování. Řešení úloh se zpravidla skládá ze dvou podkapitol, kde jedna popisuje program pro PLC S7-200 CPU222 a druhá program pro HMI TP177micro. Příklady jsou sestaveny tak, že navazují a postupně vytvářejí složitější celek, který ovládá výukový model linky kódovacího zařízení s ražením od společnosti Staudinger GMBH [3]. Způsoby jakým byly vytvořeny jednotlivé části programu, jsou prezentovány v podobě obrázků s jejich podrobným popisem. A případy, ve kterých se nabízí jiná než zvolená řešení, jsou popsány pouze textem v příslušných kapitolách, z důvodu velikosti obrázků potřebných pro jejich popis.

Cílem diplomové práce bylo vytvoření výukového materiálu s řešenými úlohami pro Gymnázium a SOŠ Rokycany, Mládežníků 1115. Inspirací byly konzultace s vyučující odborných předmětů této školy. Vzhledem k výši hodinových dotací předmětu "Řídící systémy" není dostatek prostoru na ukázková řešení většího počtu úloh. Aby byla práce ucelená a přínosná, tak jsem nejdříve zmapoval problémy, které žáci mají při tvorbě samostatného projektu. Studijní materiál jsem cíleně koncipoval tak, aby byl pro žáky nápomocným vodítkem a měli z čeho čerpat při tvorbě svých ročníkových prací. Zjistil jsem, že během tvorby obsáhlejších projektů mají žáci problém s ethernetovým modulem, jak s nastavením, tak s využitím. Dalším problémem se přímo na tomto modelu ukázalo rozlišení kódů, proto většina žáků model linky kódovacího zařízení s ražením používala jen v omezeném rozsahu bez rozlišení kódu. Využil jsem tohoto poznatku a v části svojí diplomové práce jsem se zabýval možnostmi rozlišení kódu pomocí matic. Z konzultací také vyplynulo, že žáci využívají HMI, ovšem dynamické prvky, pohyblivé obrázky či využití paměti jsou pro ně

velmi náročné a snaží se vyhnout komplexnějšímu řešení, proto jsem chybějící dovednosti žáků s HMI rozpracoval v této diplomové práci.

Veškeré části programu i s rozbohem byly během tvorby diplomové práce testovány přímo ve výuce na SOŠ. Díky cenným podnětům žáků i vyučující jsem měl možnost věnovat se problematičným partiím a hledat co nejjednodušší řešení, na kterém by danou problematiku lépe pochopili. Práce se tedy soustředí na tyto obtížnější prvky a detailně popisuje jejich řešení, které částečně pomohou žákům i v řešení úloh na jiných modelech. Mohu říci, že vznikl ucelený učební materiál s řešenými úlohami a rozborů těchto úloh pro žáky SOŠ orientovaný na náročnější pasáže.

## Použitá literatura

- [1] Technické dokumentace pro výrobky společnosti *SIEMENS*. [online]. [staženo 20.11.2012]. <http://www.siemens.com>
- [2] *Argosoft SMTP*. [online]. [staženo 10.2.2013]: <http://www.argosoft.com/>
- [3] *Staudinger GMBH*. [online]. [staženo 10.2.2013]: <http://www.staudinger-est.de/>
- [5] Firemní materiály společnosti *ABB s.r.o.*
- [6] Bakalářská práce: Jan Vitek. *Výukový mechatronický systém s PLC*. FEL ZCU, Plzeň

## Seznam obrázků

OBR. 1	ROZLIŠENÍ MEZI LADDER A FBD EDITOREM [ZDROJ: VLASTNÍ TVORBA] .....	12
OBR. 2	ETHERNET IP ADRESA [ZDROJ: VLASTNÍ TVORBA] .....	15
OBR. 3	ADMINISTRATOR ACCOUNT [ZDROJ: VLASTNÍ TVORBA] .....	16
OBR. 4	COMMUNICATIONS VÝŘEZ [ZDROJ: VLASTNÍ TVORBA] .....	17
OBR. 5	VÝŘEZ Z WEBOVÉ STRÁNKY ETHERNET MODULU [ZDROJ: VLASTNÍ TVORBA] .....	18
OBR. 6	PROGRAM PRO ODESLÁNÍ E-MAILU [ZDROJ: VLASTNÍ TVORBA] .....	18
OBR. 7	ZÁKLADNÍ OBRAZOVKA [ZDROJ: VLASTNÍ TVORBA] .....	20
OBR. 8	PŘEHLEDOVÁ OBRAZOVKA OBĚHU VODY A PÁRY Z ELEKTRÁRNY HKW WEST WOLFSBURG VYTVOŘENO PROGRAMEM PGTOOL [ZDROJ: FIRERMNÍ MATERIÁLY SPOLEČNOSTI ABB S.R.O. [5]] .....	21
OBR. 9	OVLÁDÁNÍ VÝSTUPŮ [ZDROJ: VLASTNÍ TVORBA] .....	22
OBR. 10	MODEL S OZNAČENÝMI SENZORY [ZDROJ: WWW.STAUDINGER-EST.DE S VLASTNÍ ÚPRAVOU] .....	23
OBR. 11	NETWORK 1 AŽ NETWORK 6 HLAVNÍ PROGRAM [ZDROJ: VLASTNÍ TVORBA] .....	24
OBR. 12	PODPROGRAM RAŽENÍ [ZDROJ: VLASTNÍ TVORBA] .....	25
OBR. 13	NETWORK 1 - NETWORK 5 [ZDROJ: VLASTNÍ TVORBA] .....	27
OBR. 14	PŘÍŘAZENÍ HODNOT DO MATICE [ZDROJ: VLASTNÍ TVORBA] .....	28
OBR. 15	RAŽENÍ [ZDROJ: VLASTNÍ TVORBA] .....	28
OBR. 16	OBRAZOVKA OVLÁDÁNÍ [ZDROJ: VLASTNÍ TVORBA] .....	29
OBR. 17	NETWORK 6 HLAVNÍHO PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	31
OBR. 18	HLAVNÍ PROGRAM, SPUŠTĚNÍ PODPROGRAMU PŘÍŘAZENÍ PAMĚTI [ZDROJ: VLASTNÍ TVORBA] .....	32
OBR. 19	PODPROGRAM PŘÍŘAZENÍ PAMĚTI [ZDROJ: VLASTNÍ TVORBA] .....	32
OBR. 20	NETWORK 15 HLAVNÍHO PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	32
OBR. 21	OVLÁDÁNÍ PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	33
OBR. 22	PROGRAM PODMÍNEK PRO RAŽENÍ [ZDROJ: VLASTNÍ TVORBA] .....	34
OBR. 23	ÚPRAVY HLAVNÍHO PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	36
OBR. 24	PŘIDANÉ NETWORKY DO HLAVNÍHO PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	37
OBR. 25	PODPROGRAM PANEL_ZOBRAZENÍ NETWORK 1 A 2 [ZDROJ: VLASTNÍ TVORBA] .....	38
OBR. 26	PODPROGRAM PANEL_ZOBRAZENÍ NETWORKY 3 - 6 [ZDROJ: VLASTNÍ TVORBA] .....	39
OBR. 27	ÚVODNÍ OBRAZOVKA [ZDROJ: VLASTNÍ TVORBA] .....	39
OBR. 28	OBRAZOVKA POČTU VÝROBKŮ [ZDROJ: VLASTNÍ TVORBA] .....	40
OBR. 29	PŘEHLEDOVÁ OBRAZOVKA [ZDROJ: VLASTNÍ TVORBA] .....	41
OBR. 30	SLOŽENÍ POHYBLIVÉHO PÁSU [ZDROJ: VLASTNÍ TVORBA] .....	41
OBR. 31	ÚPRAVENÉ NETWORKY V PROGRAMU PLC [ZDROJ: VLASTNÍ TVORBA] .....	42
OBR. 32	ZOBRAZENÍ CHYBY [ZDROJ: VLASTNÍ TVORBA] .....	43



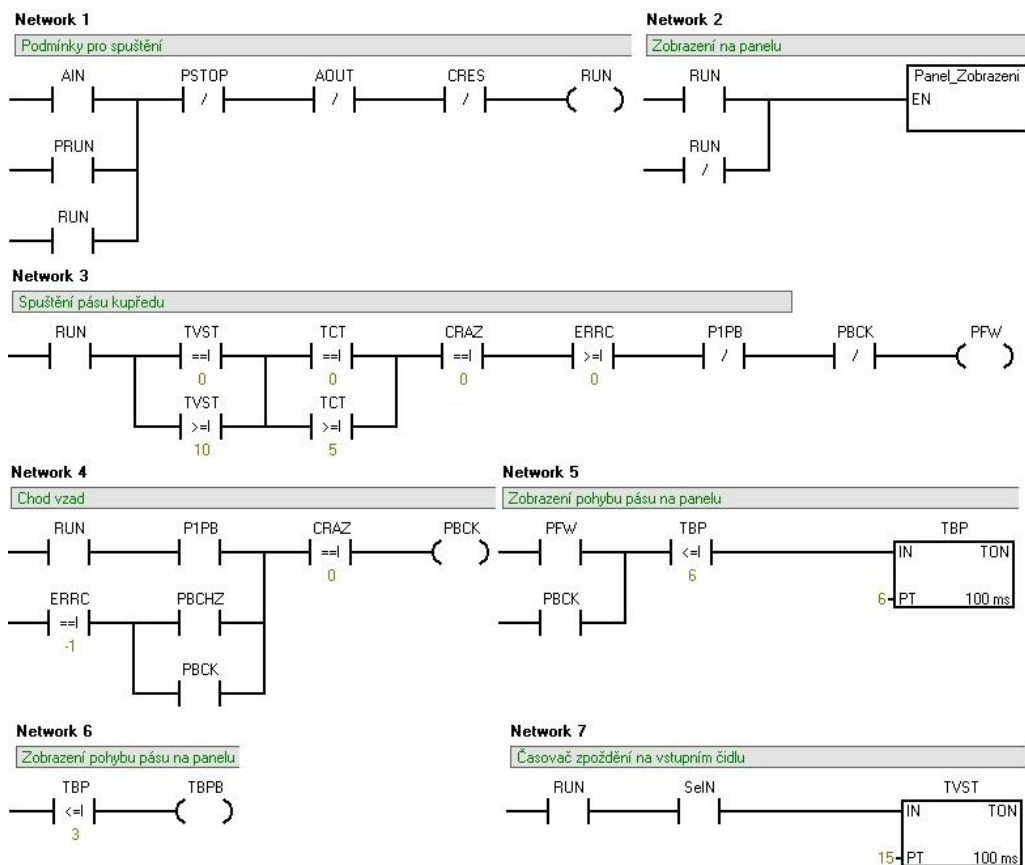
## Seznam tabulek

TAB. 1	ROZDĚLENÍ ČASOVAČŮ [ZDROJ: VLASTNÍ TVORBA] .....	12
TAB. 2	ROZDĚLENÍ POUŽITÝCH ČÍTAČŮ [ZDROJ: VLASTNÍ TVORBA] .....	12
TAB. 3	ROZDĚLENÍ PŘÍRAZOVÁNÍ HODNOT [ZDROJ: VLASTNÍ TVORBA] .....	13
TAB. 4	ROZDĚLENÍ NAVYŠOVÁNÍ HODNOT [ZDROJ: VLASTNÍ TVORBA] .....	13
TAB. 5	TABULKA SYMBOLŮ PROGRAMU [ZDROJ: VLASTNÍ TVORBA] .....	19
TAB. 6	TABULKA SENZORŮ [ZDROJ: VLASTNÍ TVORBA] .....	23
TAB. 7	TABULKA SYMBOLŮ A ADRES PRO ZÁKLADNÍ ÚLOHU [ZDROJ: VLASTNÍ TVORBA] .....	26
TAB. 8	TABULKA SYMBOLŮ A ADRES PRO ČTENÍ [ZDROJ: VLASTNÍ TVORBA] .....	30
TAB. 9	TABULKA SYMBOLŮ A ADRES PRO ZOBRAZENÍ PAMĚTI [ZDROJ: VLASTNÍ TVORBA] .....	33
TAB. 10	TABULKA SYMBOLŮ A ADRES PODMÍNEK RAŽENÍ [ZDROJ: VLASTNÍ TVORBA] .....	36
TAB. 11	TABULKA SYMBOLŮ A ADRES PRO ZOBRAZENÍ NA PANELU [ZDROJ: VLASTNÍ TVORBA] .....	41
TAB. 12	TABULKA SYMBOLŮ A ADRES CHYBY [ZDROJ: VLASTNÍ TVORBA] .....	44

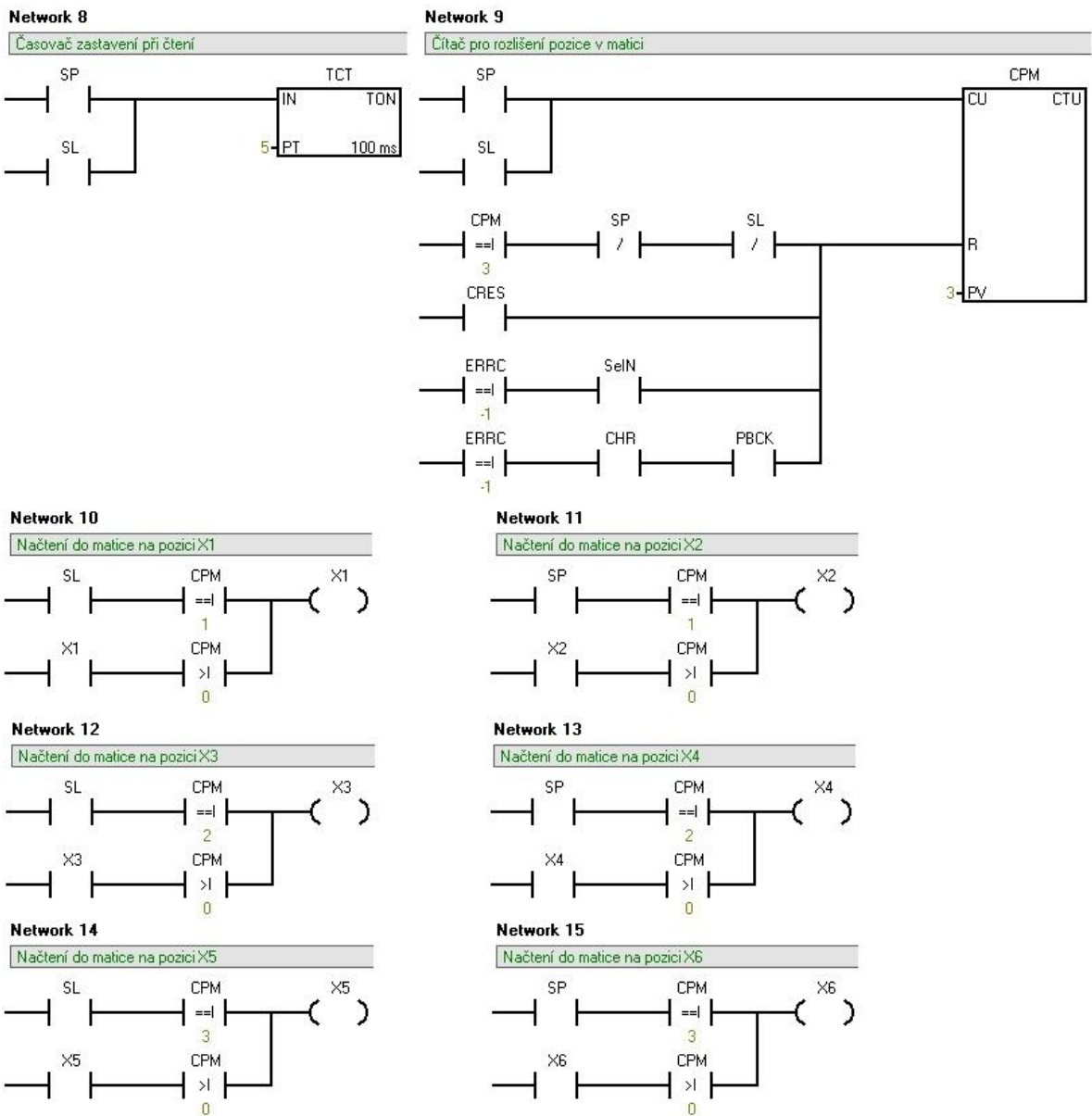
## Příloha A celý program

Obrázky jsou rozděleny na hlavní program a podprogramy. Části hlavního programu jsou rozděleny na stránky po jednotlivých částech, které mají největší návaznosti při zachování číslování networků.

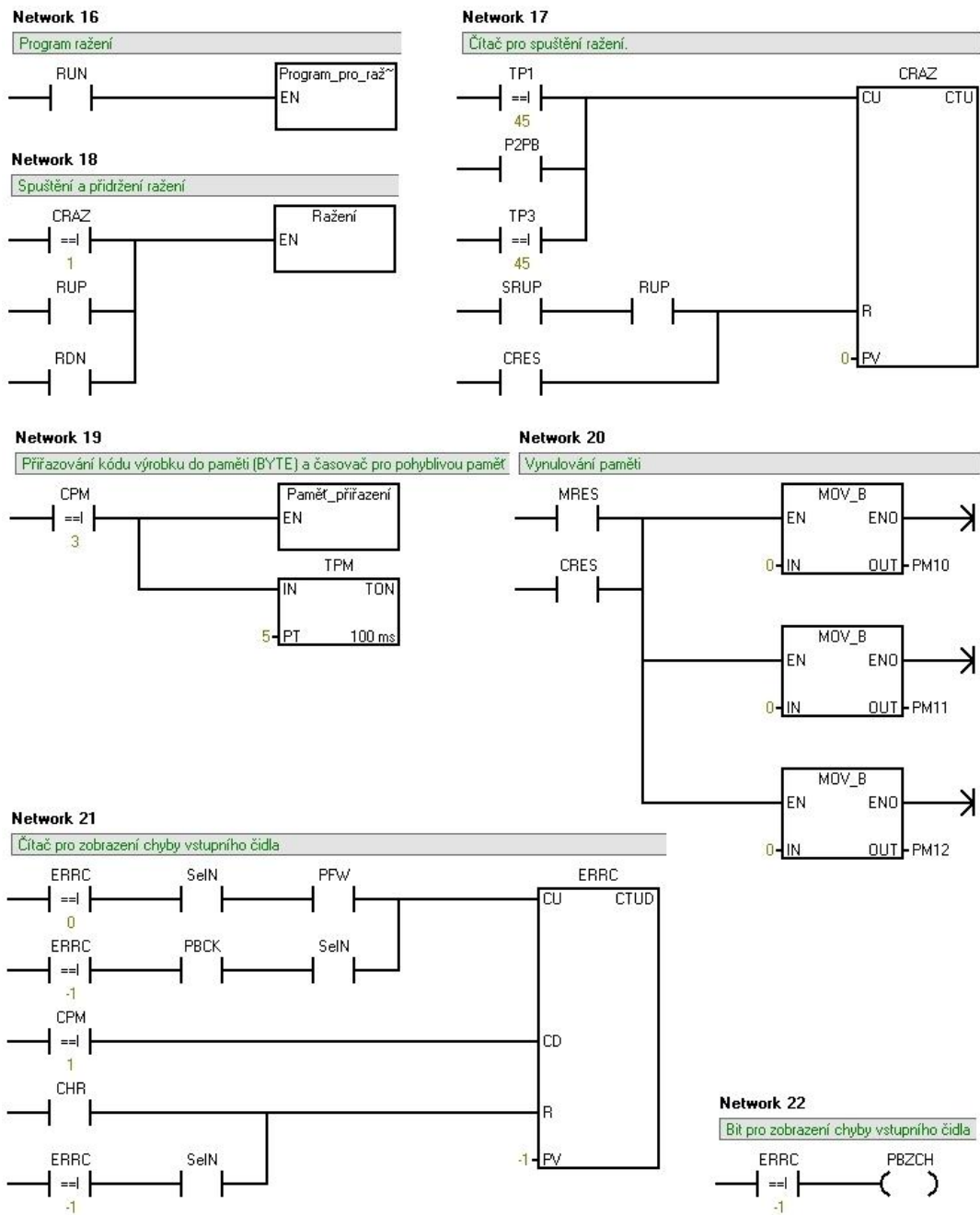
### Hlavní program



Obr. A1 Hlavní program Networky 1 - 7 [zdroj: vlastní tvorba]

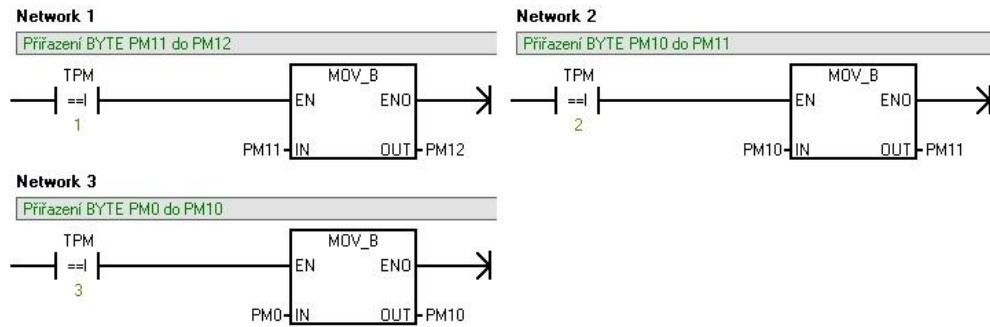


Obr. A2 Hlavní program Networky 8 - 13 [zdroj: vlastní tvorba]



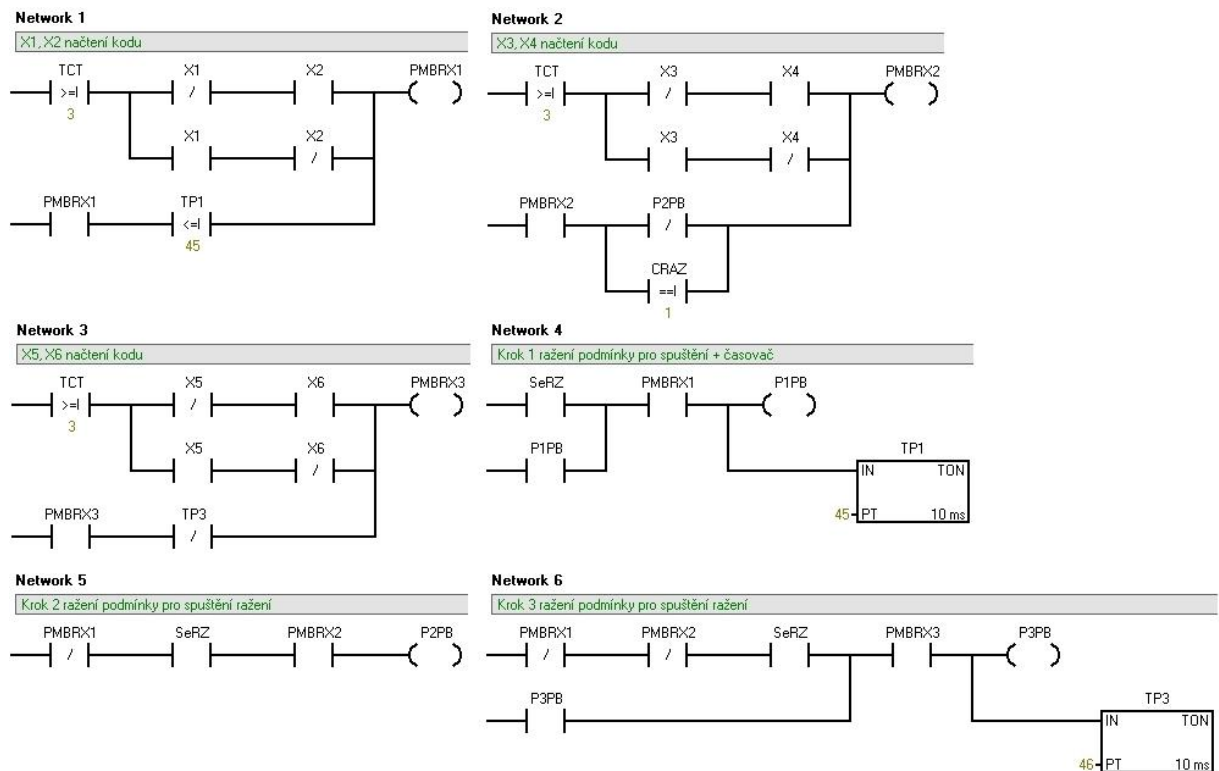
Obr. A3 Hlavní program Networky 14 - 22 [zdroj: vlastní tvorba]

## Podprogram přiřazení paměti



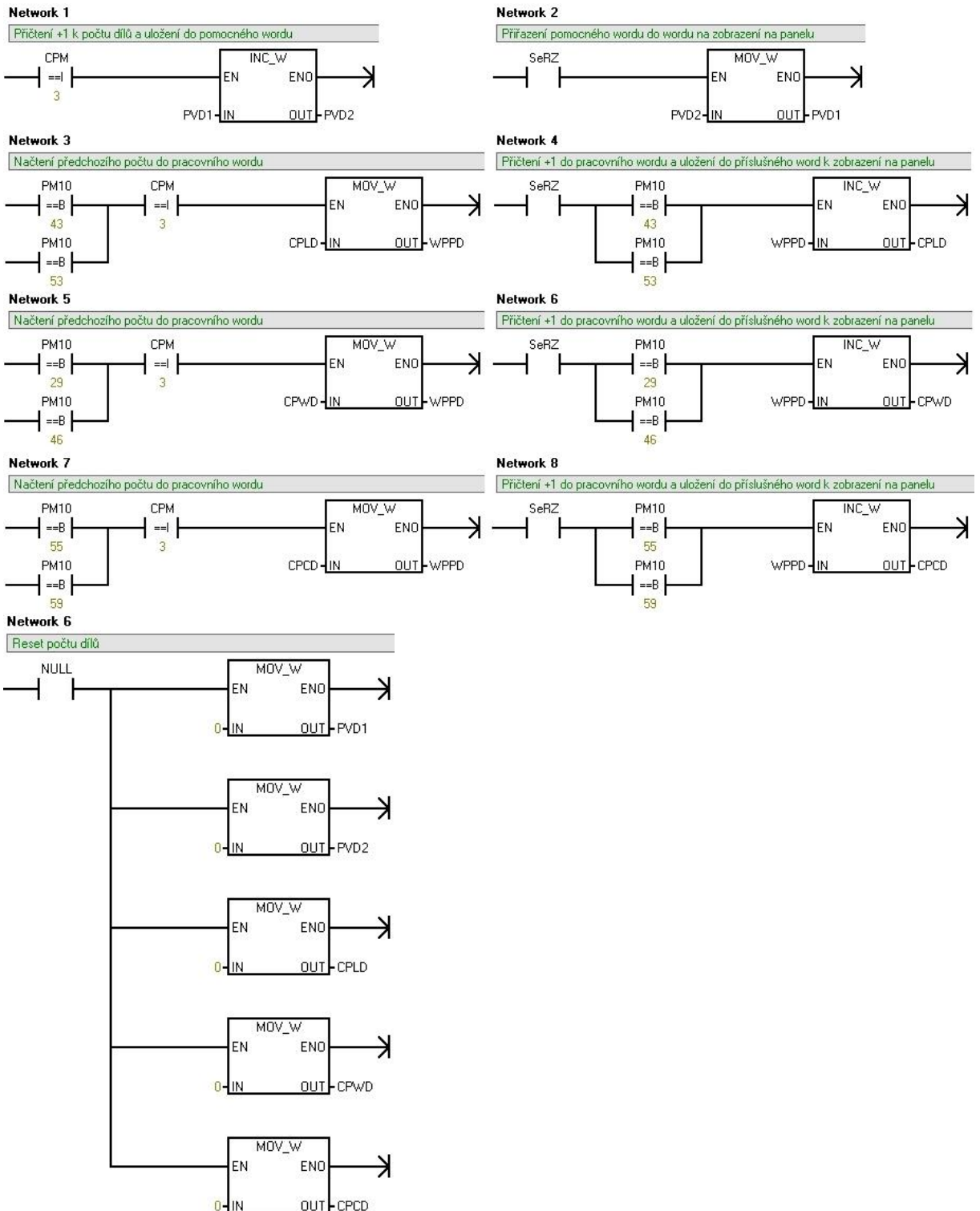
Obr. A4 Podprogram přiřazení paměti Networky 1 - 3 [zdroj: vlastní tvorba]

## Podprogram ražení



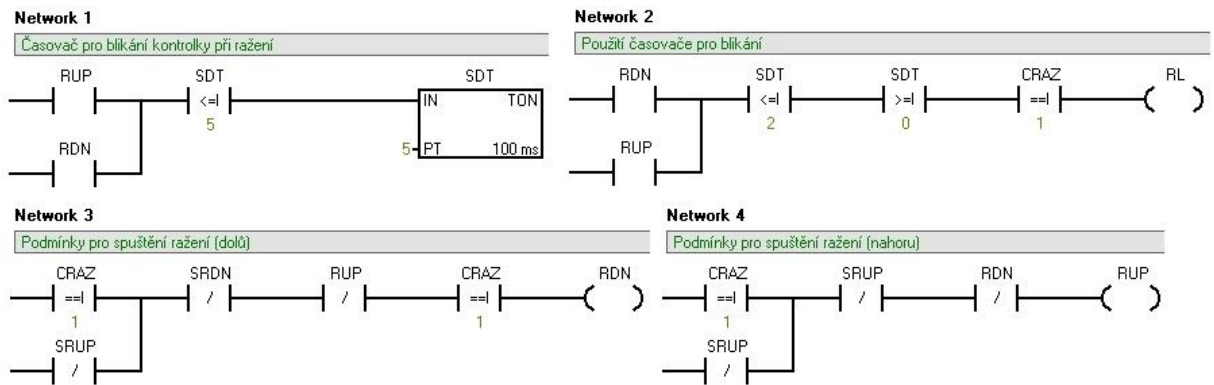
Obr. A5 Podprogram ražení Networky 1 - 6 [zdroj: vlastní tvorba]

## Podprogram pro zobrazení na panelu



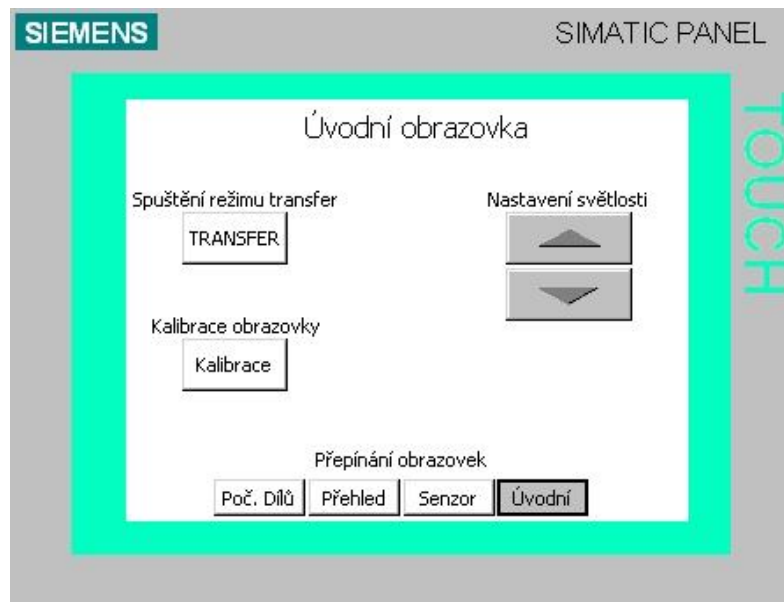
Obr. A6 Podprogram pro zobrazení na panelu Networky 1 - 6 [zdroj: vlastní tvorba]

## Podprogram spuštění ražení

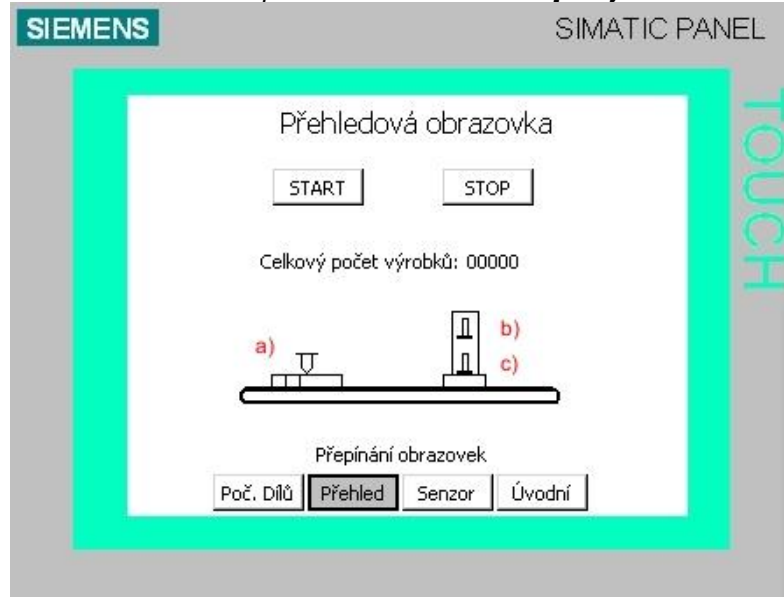


Obr. A7 Podprogram spuštění ražení Networky 1 - 4 [zdroj: vlastní tvorba]

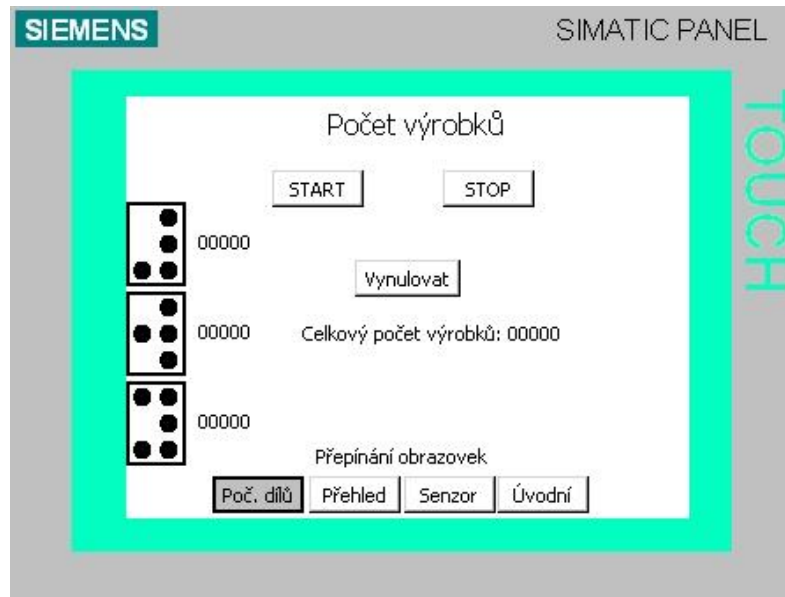
## Ovládací panel



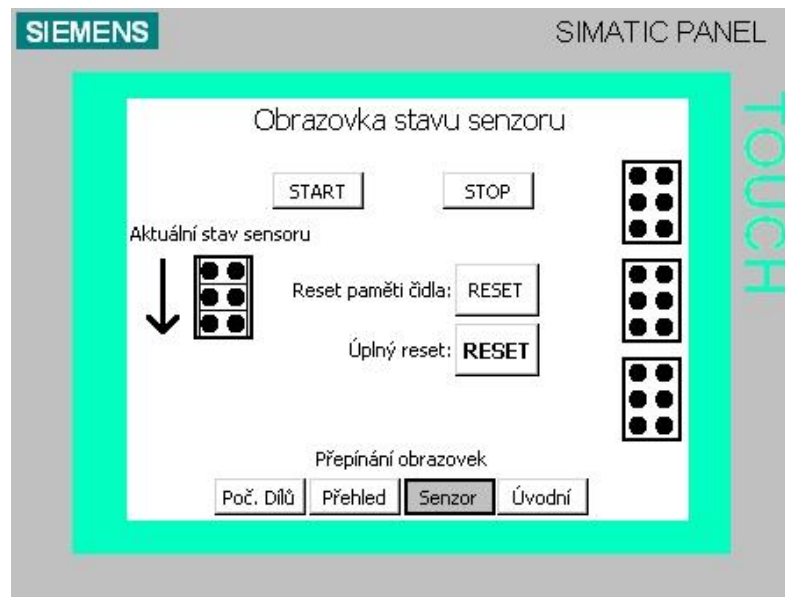
Obr. A8 Ovládací panel úvodní obrazovka [zdroj: vlastní tvorba]



Obr. A9 Ovládací panel přehledová obrazovka [zdroj: vlastní tvorba]



Obr. A10 Ovládací panel počet výrobků [zdroj: vlastní tvorba]



Obr. A11 Ovládací panel obrazovka stavu senzoru [zdroj: vlastní tvorba]



## Tabulky adres a symbolů

Symbol	Adresa	Popis	Symbol	Adresa	Popis
AIN	I0.0	Analogové tlačítko vstup	P1PB	M2.3	Pozice 1 přídržný bit ražení
AOUT	I0.1	Analogové tlačítko výstup	P2PB	M2.4	Pozice 2 přídržný bit ražení
SeIN	I0.2	Senzor na vstupu	P3PB	M2.5	Pozice 3 přídržný bit ražení
SeRZ	I0.3	Senzor u ražení	CHR	M3.0	Pomocný bit resetu ERRC
SP	I0.4	Pravý senzor kódování	CRES	M3.1	Úplný reset
SL	I0.5	Levý senzor kódování	PBZCH	M3.2	Bit zobrazení chyby
SRUP	I0.6	Ražení nahoře	PBCHZ	M3.3	Bit chyby chod zpět
SRDN	I0.7	Ražení dole	PRUN	M4.0	Spuštění z HMI
PFW	Q0.0	Pás chod vpřed	PSTOP	M4.1	Vypnutí z HMI
PBCK	Q0.1	Pás chod vzad	PM0	MB0	Vstupní paměť matice
RUP	Q0.2	Ražení chod nahoru	PM10	MB10	Paměť matice M10
RDN	Q0.3	Ražení chod dolů	PM11	MB11	Paměť matice M11
RL	Q0.4	Ražení kontrolka	PM12	MB12	Paměť matice M12
X1	M0.0	Pozice v matici	PVD1	MW13	Počet dílů
X2	M0.1	Pozice v matici	PVD2	MW15	Počet dílů pomocný word
X3	M0.2	Pozice v matici	CPLD	MW17	Celkový počet L dílů
X4	M0.3	Pozice v matici	CPWD	MW19	Celkový počet W dílů
X5	M0.4	Pozice v matici	CPCD	MW21	Celkový počet C dílů
X6	M0.5	Pozice v matici	TP1	T33	Časovač pozice 1
RUN	M1.0	Bit pro spuštění	TP3	T34	Časovač pozice 3
MRES	M1.1	reset paměti	TVST	T37	Časovač vstupního senzoru
TBPB	M1.2	Pomocný bit běhu pásu	TCT	T38	Časovač na senzorech čtení
PB1	M1.3	Přídržný bit L dílu	SDT	T39	Časovač pro blikání
PB2	M1.4	Přídržný bit W dílu	TPM	T40	Časovač pro paměť
PB3	M1.5	Přídržný bit C dílu	TBP	T42	Časovač běhu pásu
NULL	M1.6	Vynulování počtu dílů	CPM	C1	Čítač pozice v matici
PMBRX1	M2.0	Pomocný bit ražení pozice 1	CRAZ	C2	Čítač pro zastavení ražení
PMBRX2	M2.1	Pomocný bit ražení pozice 2	ERRC	C3	Čítač chybové hlášky
PMBRX3	M2.2	Pomocný bit ražení pozice 3			

Tab. A1 Tabulka symbolů a adres pro PLC program [zdroj: vlastní tvorba]

V programu pro dotykový panel TP177 jsou shodné adresy a symboly. Avšak pro zobrazení kódu výrobku jsou přidány bitové adresy paměťových BYTE viz tabulka A2.

Symbol	Adresa	Popis	Symbol	Adresa	Popis
X10_1	M10.0	Paměť pozice v matici	X11_4	M11.3	Paměť pozice v matici
X10_2	M10.1	Paměť pozice v matici	X11_5	M11.4	Paměť pozice v matici
X10_3	M10.2	Paměť pozice v matici	X11_6	M11.5	Paměť pozice v matici
X10_4	M10.3	Paměť pozice v matici	X12_1	M12.0	Paměť pozice v matici
X10_5	M10.4	Paměť pozice v matici	X12_2	M12.1	Paměť pozice v matici
X10_6	M10.5	Paměť pozice v matici	X12_3	M12.2	Paměť pozice v matici
X11_1	M11.0	Paměť pozice v matici	X12_4	M12.3	Paměť pozice v matici
X11_2	M11.1	Paměť pozice v matici	X12_5	M12.4	Paměť pozice v matici
X11_3	M11.2	Paměť pozice v matici	X12_6	M12.5	Paměť pozice v matici

Tab. A2 Tabulka symbolů a adres přidávaných pro dotykový panel [zdroj: vlastní tvorba]