

**ČESKÉ VYSOKÉ
UČENÍ TECHNICKÉ
V PRAZE**

**FAKULTA
STROJNÍ**



**DIPLOMOVÁ
PRÁCE**

2022

**PAVEL
ABRAHAM**

České vysoké učení technické v Praze

Fakulta strojní

Ústav přístrojové a řídicí techniky

Obor: Automatizační a přístrojová technika

Řízení zobecněného jednoúčelového stroje se třemi pneumatickými
pohony

DIPLOMOVÁ PRÁCE

Vypracoval: Bc. Pavel Abraham

Vedoucí práce: Ing. Marie Martinásková, Ph.D.

Rok: 2022

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Abraham** Jméno: **Pavel** Osobní číslo: **434233**
Fakulta/ústav: **Fakulta strojní**
Zadávající katedra/ústav: **Ústav přístrojové a řídicí techniky**
Studijní program: **Automatizační a přístrojová technika**
Specializace: **Automatizace a průmyslová informatika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Řízení zobecněného jednoúčelového stroje se třemi pneumatickými pohony

Název diplomové práce anglicky:

Virtual one-purpose machine with three pneumatic drives control

Pokyny pro vypracování:

1. Seznamte se s platformou Siemens SIMATIC PLC a vypracujte přehled metodiky odlaďování programů pro PLC se zaměřením na SIMATIC S7-1200.
2. Navrhněte řízení pomocí SIMATIC S7-1200 a odlaďte ho pomocí S7-PLCSIM. V rámci řízení zajistíte možnost parametrizace výrobního cyklu.
3. Navrhněte vizualizaci pomocí WinCC včetně zadání parametrizace výrobního cyklu, chybových hlášení, sběru provozních dat a jejich vyhodnocení.

Seznam doporučené literatury:

Berger, H: Automating with SIMATIC S7-1200, Publicis Erlangen 2013, ISBN: 978-3-89578-385-2
SIMATIC S7-1200 - HW systémový manuál
SIMATIC WinCC - systémový manuál

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Marie Martinásková, Ph.D., U12110.3

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **29.10.2021**

Termín odevzdání diplomové práce: **20.01.2022**

Platnost zadání diplomové práce: _____

Ing. Marie Martinásková, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Michael Valášek, DrSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

_____ Datum převzetí zadání

_____ Podpis studenta

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně s tím, že její výsledky mohou být dále použity podle uvážení vedoucího diplomové práce jako jejího spoluautora. Souhlasím také s případnou publikací výsledků diplomové práce nebo její podstatné části, pokud budu uveden jako její spoluautor.

V Praze dne:

.....
Bc. Pavel Abraham

Poděkování

Děkuji Ing. Marii Martináskové, Ph.D. za připomínky při vedení této diplomové práce a současně i rodině za podporu během celého studia.

Název práce: Řízení zobecněného jednoúčelového stroje se třemi pneumatickými pohony

Autor: Bc. Abraham Pavel

Obor: Automatizační a přístrojová technika

Druh práce: Diplomová práce

Vedoucí práce: Ing. Marie Martinásková, Ph.D.
Ústav přístrojové a řídicí techniky Fakulty strojní
České vysoké učení technické v Praze

Abstrakt: Tato diplomová práce se zabývá řízením zobecněného jednoúčelového stroje. Cílem bylo seznámení se s portfoliem Siemens SIMATIC PLC a návrhem řízení pomocí S7-1200, návrhem vizualizace pomocí WinCC a odladění pomocí S7-PLCSIM. Podle zadání byl vytvořen projekt v TIA Portalu obsahující řídicí program a vizualizaci v podobě HMI panelu.

Klíčová slova: PLC, HMI, PNEUMATIKA, VIZUALIZACE, WINCC, TIA PORTAL

Title: Virtual one-purpose machine with three pneumatic drives control

Author: Bc. Abraham Pavel

Abstract: This diploma thesis deals with control of virtual one-purpose machine with three pneumatic drives. The aim was to learn about Siemens SIMATIC PLC portfolio and design control program using S7-1200, design visualization using WinCC and commission the system with S7-PLCSIM. According to the assignment, as a result, the project was created in TIA Portal including control program and visualization in the form of HMI panel.

Keywords: PLC, HMI, PNEUMATICS, VIZUALIZATION, WINCC, TIA PORTAL

OBSAH

ÚVOD.....	1
1 SIEMENS SIMATIC PLC.....	2
1.1 SIMATIC S7-1200	2
1.2 SIMATIC S7-1500	3
1.3 SIMATIC ET 200	3
1.4 SIMATIC S7-300 A S7-400	4
2 PROSTŘEDÍ TIA PORTAL.....	5
2.1 STEP 7	5
2.2 WINCC	6
2.3 WINCC UNIFIED V17	6
2.4 STARTDRIVE	6
2.5 SCOUT TIA	6
2.6 SIRIUS ES	6
3 CODING GUIDELINES.....	7
3.1 PLCOPEN CODING GUIDELINE	7
3.2 SIEMENS PROGRAMMING STYLE GUIDE	10
4 PROGRAMOVÁNÍ V PROSTŘEDÍ TIA PORTAL.....	11
4.1 CYKLUS PLC.....	11
4.2 DEKLAROVÁNÍ A TYPY PROMĚNNÝCH	12
4.2.1 GLOBÁLNÍ PROMĚNNÉ	12
4.2.2 LOKÁLNÍ PROMĚNNÉ.....	12
4.2.3 PLC DATA TYPE.....	13
4.2.4 DB – DATABLOCKS.....	13
4.2.5 ADRESOVÁNÍ ČÁSTÍ TAGU.....	13
4.3 OB – ORGANIZATION BLOCK.....	14
4.4 SOFTWARE UNITS(S7-1500).....	15
4.5 ŠKÁLOVÁNÍ ANALOGOVÝCH VSTUPŮ/VÝSTUPŮ	16
5 PROGRAMOVACÍ JAZYKY V TIA PORTALU	17

5.1	STL – STATEMENT LIST (S7-1500)	18
5.2	GRAPH (S7-1500)	18
5.3	FBD – FUNCTION BLOCK DIAGRAM	19
5.4	CFC – CONTINUOUS FUNCTION CHART (V17, S7-1500)	20
5.5	CEM(V17) – CAUSE EFFECT MATRIX	20
5.6	SCL – STRUCTURED CONTROL LANGUAGE	20
5.6.1	IF-THEN	21
5.6.2	CASE	22
5.6.3	FOR	23
5.6.4	WHILE	24
5.6.5	REPEAT	25
5.6.6	CONTINUE	25
5.6.7	EXIT	25
5.7	LAD – LADDER DIAGRAM	26
6	ODLAĐOVÁNÍ	27
6.1	OFFLINE	27
6.1.1	KOMPILÁTOR	27
6.1.2	OVERLAPPING	28
6.1.3	MATEMATICKÉ OPERACE	29
6.2	ONLINE	29
6.2.1	MONITOR	29
6.2.2	FORCE TABLE	30
6.2.3	WATCH TABLE	30
6.2.4	TRACE	31
6.2.5	BREAKPOINTS (S7-1500)	31
6.3	S7-PLCSIM	32
6.4	S7-PLCSIM ADVANCED	33
6.5	METODIKA UVÁDĚNÍ DO PROVOZU A ODLAĐOVÁNÍ PROGRAMU	34
7	NÁVRH ŘÍZENÍ ZOBECNĚNÉHO JEDNOÚČELOVÉHO STROJE	36
7.1	PŘEDPOKLAD BEZPEČNOSTI	36
7.2	PNEUMATICKÉ KOMPONENTY	36
7.3	VSTUPY A VÝSTUPY PLC	37
7.4	PROVOZNÍ REŽIMY	39
7.4.1	AUTOMATICKÝ REŽIM	40
7.4.2	DÁVKOVÝ REŽIM	40

7.4.3	MANUÁLNÍ REŽIM	40
7.4.4	KROK ZA KROKEM	40
7.4.5	SERVIS.....	40
7.5	VÝROBNÍ CYKLUS	41
7.5.1	ŘÍZENÍ VÁLCŮ.....	41
7.5.2	PARAMETRIZACE VÝROBNÍHO CYKLU A INICIALIZACE.....	42
7.5.3	VÝROBNÍ CYKLUS.....	42
7.5.4	PŘECHOD V SEKVENCI	44
7.5.5	VÝSTUPY	45
7.5.6	INICIALIZACE	45
7.6	SBĚR DAT	46
7.7	ALARMY.....	46
7.8	ODLAĐOVANÍ PROGRAMU	47
8	NÁVRH VIZUALIZACE ZOBECNĚNÉHO JEDNOÚČELOVÉHO STROJE	48
8.1	MAPA STRÁNEK.....	48
8.1.1	ČÍSLOVÁNÍ STRÁNEK.....	48
8.1.2	NAVIGACE	48
8.1.3	ŠABLONA STRÁNKY (TEMPLATE)	49
8.2	OVLÁDÁNÍ VÝROBY	50
8.3	PARAMETRY CYKLU	50
8.4	PROVOZNÍ DATA	51
8.5	SERVIS	52
8.5.1	ŠABLONA PRVKŮ (FACEPLATE).....	52
8.6	ALARMY.....	53
8.7	SYSTÉM.....	54
8.7.1	ZOBRAZENÍ PŘIHLÁŠENÉHO UŽIVATELE	54
8.7.2	PROVÁZÁNÍ ČASU PLC A HMI	54
9	ZÁVĚR	55
	ZDROJE	56
	SEZNAM OBRÁZKŮ	58
	SEZNAM TABULEK.....	59
	PŘÍLOHA A – SCHÉMATA	I

SEZNAM ZKRATEK

- PLC – Programmable Logic Controller (programovatelný logický automat)
- IL – Instruction List (jazyk seznamu instrukcí)
- STL – Statement List (jazyk výroků)
- ST – Structured text (jazyk strukturovaného textu)
- SCL – Structured Control Language (jazyk strukturovaného řízení)
- LAD – Ladder Diagram (jazyk příčkového diagramu)
- SFC – Sequential Function Chart (jazyk sekvenční funkčního diagramu)
- CFC – Continuous Function Chart (jazyk kontinuálního funkčního diagramu)
- CEM – Cause Effect Matrix (jazyk matice příčiny a následku)
- FBD – Function Block Diagram (jazyk funkčních bloků)
- HMI – Human Machine Interface (rozhraní člověk-stroj)
- UDT – User-Defined Data Type (uživatелеm definovaný datový typ)
- HSC – High Speed Counter (vysokorychlostní čítač)
- IO, I/O – INPUT/OUTPUT (vstup/výstup)
- CPU – Central Processing Unit (centrální procesorová jednotka)
- SCADA – Supervisory Control And Data Acquisition (dispečerské řízení a sběr dat)
- TIA Portal – Totally Integrated Automation Portal

Úvod

V této diplomové práci je věnována pozornost portfoliu SIEMENS SIMATIC PLC. Jsou představeny směrnice poskytnuté firmou SIEMENS a směrnice PLCopen, které slouží ke standardizaci programování PLC a sjednocení struktur programů třetích stran. Následně jsou představeny možnosti programování v prostředí TIA Portal, programovací jazyky, které prostředí nabízí a možnosti odlaďování pomocí S7-PLCSIM, včetně obecné metodiky.

V praktické části je řešeno řízení virtuálního jednoúčelového stroje s parametrizací výrobního cyklu. Dále je navržena vizualizace v podobě operátorského HMI panelu, včetně možnosti zadávání parametrů výrobního cyklu. Program řízení a vizualizace poté byly odladěny pomocí simulačních programů S7-PLCSIM a WinCC Runtime Advanced.

1 Siemens SIMATIC PLC

Široké spektrum řídicích systémů SIMATIC zahrnuje základní, pokročilé, distribuované a softwarové řídicí jednotky, které poskytují možnosti dalšího rozšíření a integraci nejrůznějších funkcí. Díky projektování v prostředí TIA Portal tak lze najít automatizační řešení pro každý typ aplikace.[1]

1.1 SIMATIC S7-1200



OBR 1 SIMATIC S7-1200 [2]

Automatizační systém S7-1200 se skládá ze 4 kontrolérů, a to:

S7-1211C

S7-1212C

S7-1214C

S7-1215C

Dělají se v normální verzi, bezpečnostní verzi F, verzi se zvýšenou odolností SIPLUS a bezpečnostní variantou SIPLUS F.

K programování je použit Step 7 Basic v rámci aplikace TIA Portal. S7-1200 je modulární mikrosystém, který je vhodný k malým až středním aplikacím. Kontroléry mají vlastní integrované vstupy a výstupy, jejichž počet je určen verzí CPU. Vstupy mají funkci HSC, neboli vysokorychlostní čítač. Systém je rozšiřitelný o IO moduly, případně komunikační moduly. [3] [4]

Každá verze CPU má tři druhy podle integrovaných vstupů a výstupů a to:

Napájení	Vstupy	Výstupy
AC	DC	Relé
DC	DC	DC
DC	DC	Relé

TABULKA 1 DRUHY CPU S7-1200

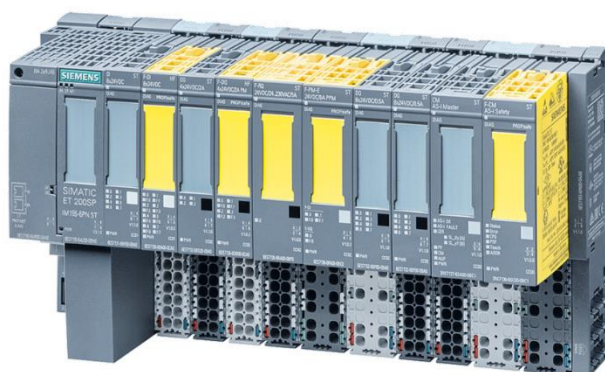
1.2 SIMATIC S7-1500



OBR 2 SIMATIC S7-1500 [5]

S7-1500 je automatizační systém pro střední a vyšší výkonnostní náročnost. Různé verze kontrolérů umožňují vybrat vhodný pro danou aplikaci. Systém se skládá z CPU jednotky a rozšiřujících IO modulů. [6]

1.3 SIMATIC ET 200



OBR 3 ET200SP S IO A BEZPEČNOSTNÍMI IO MODULY [7]

ET 200 slouží jako distribuovaný systém, kde je IO „ostrov“ připojen k centrálnímu CPU pomocí PROFINET IO. ET 200 se dělá ve verzích do rozvaděčů, které jsou rozšiřitelné o IO moduly, pro formát S7-1500 je to ET 200MP, nebo ve verzi modulu, ke kterému se přímo připojují instrumenty a to ET 200pro nebo ET 200 AL. Existuje i CPU jednotka ve formátu ET 200SP. [6] [8]

1.4 SIMATIC S7-300 a S7-400

Předchůdce S7-1500, který je stále v TIA Portalu zpětně podporován.



OBR 4 SIMATIC S7-400 [9]

2 PROSTŘEDÍ TIA PORTAL

TIA Portal, neboli Totally Integrated Automation Portal je platforma, která propojuje celou řadu inženýrských nástrojů. [10]

TIA Portal
Product overview Engineering and Runtime options

SIEMENS
Ingeniunity for life

Totally Integrated Automation Portal

	SIMATIC STEP 7	SIMATIC WinCC	SINAMICS Startdrive	SIMOTION SCOUT TIA	SIRIUS SIRIUS ES
Engineering options	TIA Portal Multiuser Engineering				
	TIA Portal Teamcenter Gateway				
	TIA Portal Cloud Connector				
	TIA User Management Component				
	SIMATIC Energy Suite ES				
	TIA Portal Test Suite				
	SIMATIC S7-PLCSIM Advanced				
	SIMATIC STEP 7 Safety				
	SIMATIC ODK 1500S				
	SIMATIC Target 1500S for Simulink®				
	SIVArc				
	SINAMICS DCC				
Runtime options	SIMATIC ProDiag				
	SIMATIC Energy Suite RT / S7-EE-Monitor				
	SIMATIC OPC UA				
	Safe Kinematics				
WinCC/WebUX ¹⁾					

1) Part of WinCC Runtime Professional DVD Download

OBR 5 SOUČÁSTI TIA PORTAL V16 [11]

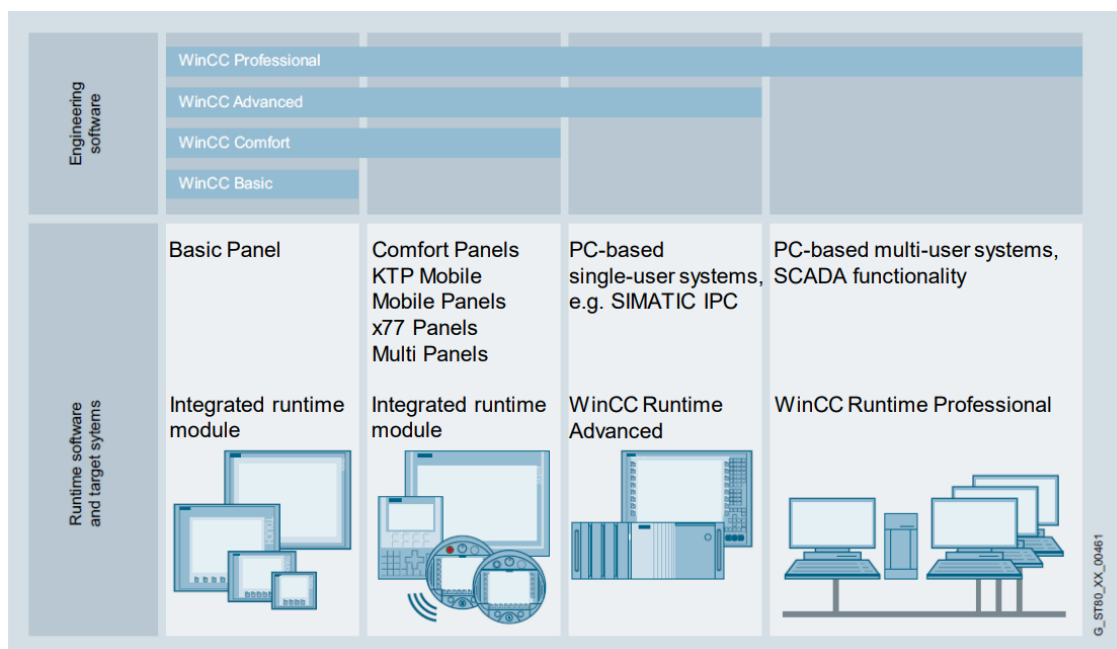
2.1 Step 7

- **Basic**
Slouží k programování S7-1200
Obsahuje jazyky LAD, FBD a SCL[3]
- **Professional**
Slouží k programování S7-1500, 300 a 400
Obsahuje navíc jazyky STL a GRAPH[6]
- **Safety**
Slouží k programování bezpečnostních funkcí v řadě procesorů S7-1200 (Safety Basic)
S7-1500, 300, 400 (Safety Advance)
Programuje se v jazycích FBD a LAD[12]

2.2 WinCC

Slouží programování operátorských HMI panelů a SCADA aplikací.

Řadí se do několika úrovní, viz obr 6.



OBR 6 ROZDĚLENÍ WINCC [13]

2.3 Wincc Unified V17

Jedná se kompletně nový vizualizační systém, který je doplněn o novou řadu operátorských panelů. [14]

2.4 Startdrive

Slouží k intuitivní integraci měničů SINAMICS v rámci projektu v TIA Portalu. [15]

2.5 Scout TIA

SIMOTION SCOUT slouží k implementaci pohybových úloh. [16]

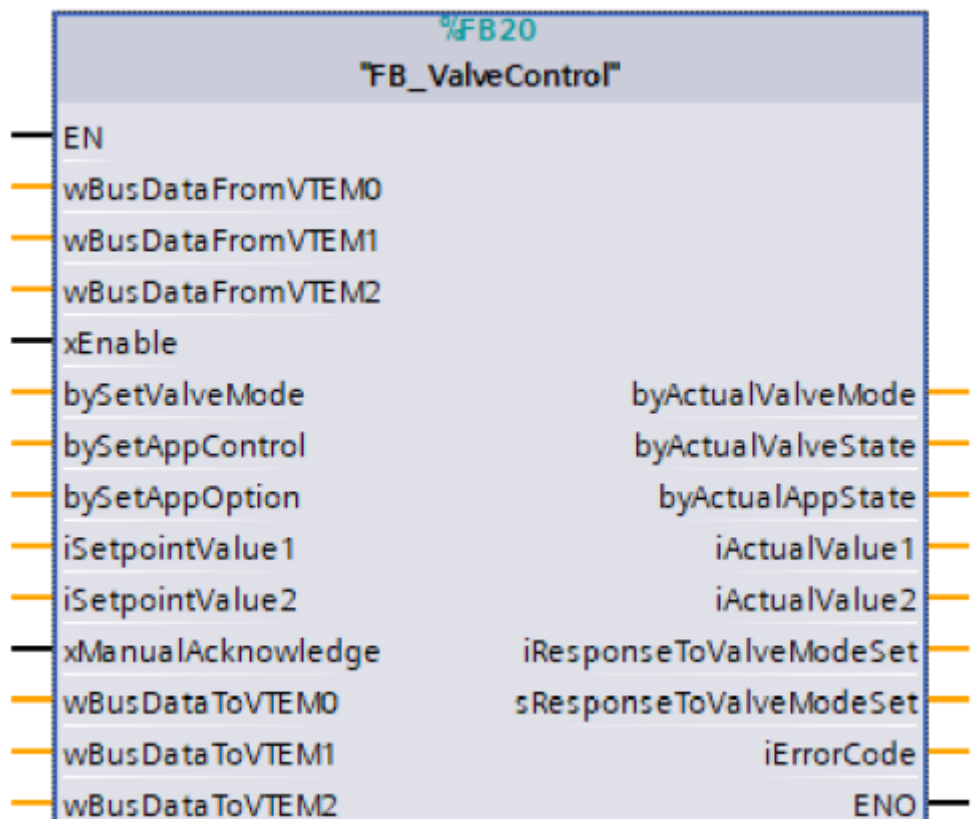
2.6 SIRIUS ES

SIRIUS Safety ES je modul do TIA Portalu, který slouží k programování SIRIUS bezpečnostních relé. [17]

3 Coding guidelines

Používání standardů a směrnic umožňuje částečnou přenositelnost programu mezi platformami, ale hlavně zaručuje čitelnost programu.

Výrobci zařízení se často těmito směrnicemi drží, a tím zaručují snadnou integrovatelnost jejich zařízení do systému, např. funkční blok k ovládní Festo VTEM na obr 7.



OBR 7 FUNKČNÍ BLOK K OVLÁDNÍ FESTO VTEM [18]

3.1 PLCopen Coding Guideline

Tato směrnice definuje pravidla programování, obsahuje ukázky, jak je aplikovat a není vázaná na konkrétní prostředí.

Odkazuje se především na standardy IEC 61131-3, Miscra-C, JSF++ a online nápovědu Codesys.

Skládá se ze čtyř hlavních částí, a to:

1. Pravidla pro pojmenování (N)
2. Pravidla pro komentování (C)
3. Praxe kódování (CP)
4. Jazyky (L)

Výtažek důležitých pravidel:

Pravidla pro pojmenovávání

- N1 – Nepoužívání fyzických adres
- N2 – Používání předpon (prefix) podle datového typu proměnné

Datový typ	Předpona
BOOL	x
INT	i
UINT	ui
REAL	r
TIME	tim
DATE	dt
STRING	str
BYTE	by
WORD	w
STRUCT	st

TABULKA 2 PŘEDPONY PRO DATOVÉ TYPY

Třída proměnné	Předpona
Globální	g
Lokální	l
Dočasná	tmp

TABULKA 3 PŘEDPONY PRO TŘÍDY PROMĚNNÝCH

Druh proměnné	Předpona
Vstup	i
Výstup	o

TABULKA 4 PŘEDPONY PRO DRUHY PROMĚNNÝCH

- N4 – Konvence stylu pojmenovávání proměnných
 - alllowercase
 - lower_snake_case
 - lowerCamelCase
 - UpperCamelCase (PascalForm)
 - ALLUPPERCASE
 - UPPER_SNAKE_CASE
 - OTHER_style

Pravidlo doporučuje používání stylu UpperCamelCase, při použití předpon z pravidla N2 změnu na lowerCamelCase a pro konstanty UPPER_SNAKE_CASE.

Pravidla pro komentování

- C1 – Komentáře by měly popisovat záměr
- C3 – Vyhnout se vnořování komentářů
- C5 – Používání jednořádkových komentářů

Praxe kódování

- CP3 – Všechny proměnné by měly být inicializovány před použitím
- CP4 – Přímé adresování by se nemělo překrývat
- CP6 – Mělo by se vyhnout externím proměnným ve funkčních blocích a funkcích
- CP8 – Porovnávání proměnných s desetinným místem by nemělo být pomocí rovnosti či nerovnosti
- CP12 – Fyzické výstupy by měly být nastavovány jen jednou za cyklus PLC

3.2 Siemens Programming style guide

Tato směrnice se skládá z pravidel a doporučení, které jsou přímo aplikovatelné v prostředí TIA Portal.

Směrnice pokrývá:

1. Nastavení v TIA Portalu (ES)
2. Globalizace (GL)
3. Názvosloví a formátování (NF)
4. Znovu použitelnost (RU)
5. Referencování objektů (AL)
6. Bezpečnost (SE)
7. Návrh programu (DA)
8. Výkon (PE)

Výtažek důležitých pravidel a doporučení:

NF005: Použít styl PascalCasing pro objekty

NF006: Použít styl camelCasing pro elementy (proměnné, datové typy, tagy)

NF007: Používat předpony

Předpona	Typ
Žádná	Vstupní a výstupní parametr
Žádná	InOut parametr
Žádná	PLC tagy a uživatelské konstanty
Žádná	Globální datové bloky
temp	Dočasná proměnná
stat	Statická proměnná
inst	Multi-instance a parametrické instance
Inst	Ojedinelá instance data bloků
type	PLC data type

TABULKA 5 PŘEDPONY PRO TYPY PROMĚNNÝCH

NF011: Použít pouze jednu zkratku v názvu proměnné (maxHodnota)

RU004: Používat pouze lokální proměnné

RU006: Programovat kompletně symbolicky

RU008: Používat šablony

AL002: Definovat pole od 0 do konstantní hodnoty uložené v proměnné

AL004: Specifikovat potřebnou délku stringu

SE001: Validovat aktuální hodnoty před použitím

SE003: Pracovat s ENO

SE005: Vyhodnocovat chybové kódy u FB/FC

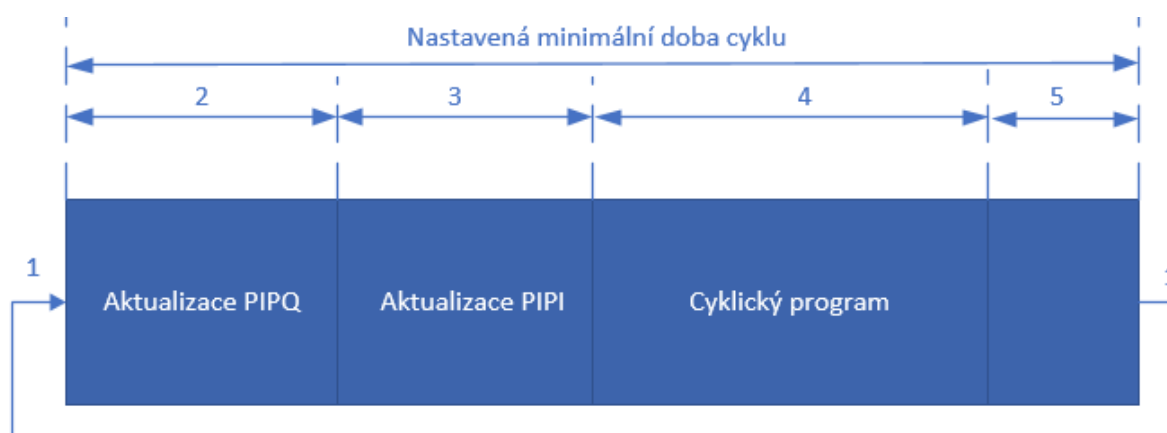
DA002: Použít vhodný programovací jazyk k aplikaci

- Standardní bloky – SCL
- Volání funkcí/ binární logika – LAD, FBD
- Sekvenční řízení – GRAPH

DA004: Používat PLC datové typy

4 Programování v prostředí TIA Portal

4.1 Cyklus PLC



OBR 8 CYKLUS PLC

PIPI – Process image partition input – Obraz vstupů

PIPQ – Process image partition output – Obraz výstupů

1 – Kontrolní bod, od kterého se počítá čas cyklu

2 – CPU z PIPQ do výstupního modulu.

3 – CPU přečte status vstupů a zapíše je do PIPI

4 – CPU vykoná OB

5 – Fáze čekání do minimální doby cyklu

V základu PLC vykonává cyklus, jako je na obr 8, ale je možné direktivně přistupovat ke vstupům a výstupům v průběhu exekuce programu. Používá se to například při přerušení, kdy nelze čekat na dokončení cyklu, ale rovnou se zapíšíou potřebné výstupy. [19]

4.2 Deklarování a typy proměnných

4.2.1 GLOBÁLNÍ PROMĚNNÉ

Deklarují se v tabulce (Tag table) v záložce PLC tags. Dělí se na tři typy, a to:

%I – Vstup

%Q – Výstup

%M – Paměť

U paměťového typu je možné zapnout atribut Retain, který slouží k uchování stavu proměnné i při restartu PLC.

Default tag table								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	příkladVstup	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	příkladVystup	Bool	%Q0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	příkladPamet	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	<Add new>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

OBR 9 PŘÍKLAD DEKLAROVÁNÍ GLOBÁLNÍCH PROMĚNNÝCH

4.2.2 LOKÁLNÍ PROMĚNNÉ

Jsou součástí Funkčního bloku nebo funkce, ve které jsou definovány. Lze k nim externě přistupovat. Dělí se na statické, dočasné, konstanty, vstupní, výstupní a vstupně výstupní.

Statické mají svoji hodnotu, pokud je v nich uložena. Dočasné (temp) mají svou hodnotu pouze do konce aktivního cyklu PLC. Konstanty mají neměnnou hodnotu.

Block_1								
	Name	Data type	Default value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	příkladVstup	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	příkladVystup	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	příkladStaticka	Int	0	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	příkladDocasna	Bool			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
14	Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
15	příkladKonstanta	Int	10		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

OBR 10 PŘÍKLAD DEKLAROVÁNÍ LOKÁLNÍCH PROMĚNNÝCH

4.2.3 PLC DATA TYPE

Jedná se o uživatelské datové typy (user data type – UDT), které se strukturují podobně jako datový typ STRUCT. Používají se především k tvorbě struktur. Tyto typy jsou poté použitelné v rámci definování lokálních a globálních proměnných.

ukazkaUDT							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	prikladInt	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	prikladBool	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	prikladWord	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

OBR 11 PŘÍKLAD DEKLAROVÁNÍ UDT

Block_1								
	Name	Data type	Default value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	prikladVstup	*ukazkaUDT*		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	prikladInt	Int	0	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	prikladBool	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	prikladWord	Word	16#0	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

OBR 12 PŘÍKLAD POUŽITÍ UDT NA VSTUPU FB

4.2.4 DB – DATABLOCKS

Datové bloky se vytváří při vytvoření Instance funkčního bloku a slouží tedy jako prostor pro lokální proměnné. Je možné vytvořit i Globální datový blok, který může sloužit například ke strukturalizaci proměnných pro HMI panel.

PrikladDB								
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	prikladStruct	Struct		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	prikladInt	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	prikladBool	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	prikladUDT	*ukazkaUDT*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	prikladInt	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	prikladBool	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	prikladWord	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

OBR 13 UKÁZKA GLOBÁLNÍHO DATOVÉHO BLOKU

4.2.5 ADRESOVÁNÍ ČÁSTÍ TAGU

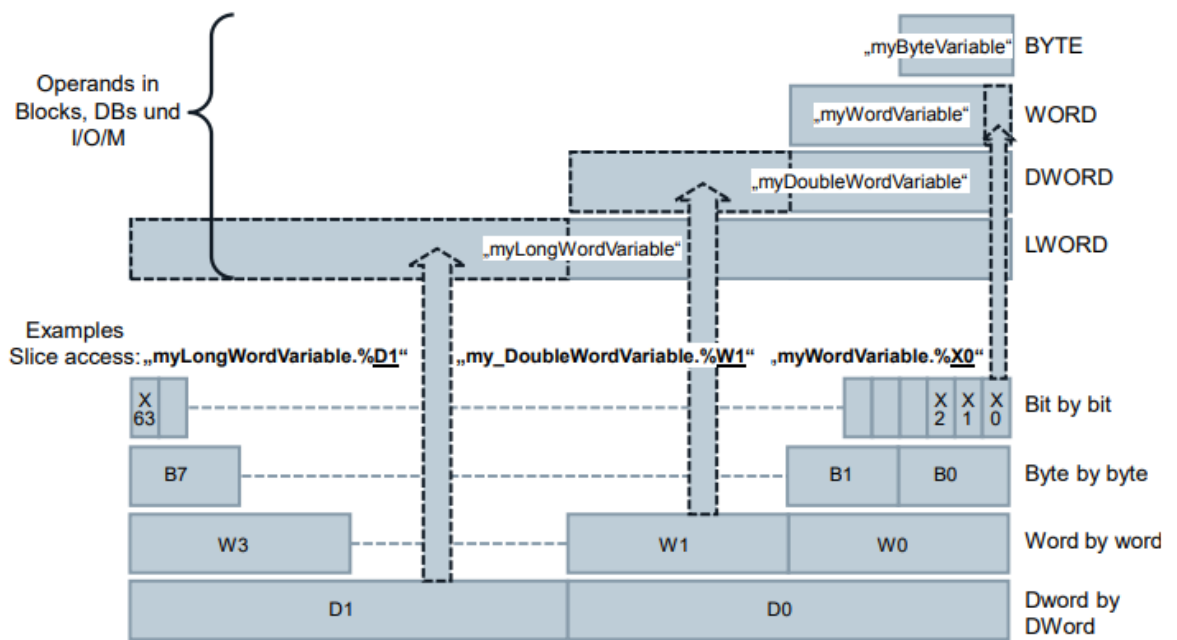
Je možné adresovat části proměnné, která je datového typu LWORD, DWORD, WORD nebo BYTE. Adresování začíná číslem 0. Přístup je pak následovný: [6]

Bit: promenna.%X<čísloBitu>

Byte: promenna.%B<čísloBytu>

Word: promenna.%W<čísloWordu>

Dword: promenna.%D<čísloDwordu>



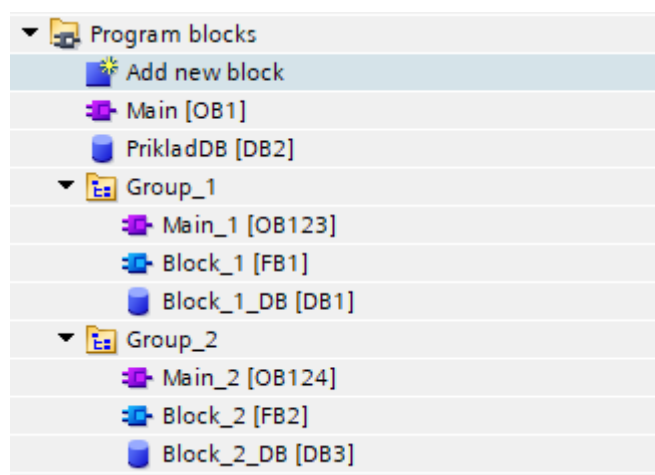
OBR 14 SLICE ACCESS [20]

4.3 OB – Organization Block

OB jsou jednotlivé uživatelské programy, které PLC vykonává. Dělí se podle čísla na několik typů, např.: [21]

- OB1 – Main
- OB100 – Startup
- OB20 – Časové přerušení
- OB40 – HSC Hardwarové přerušení

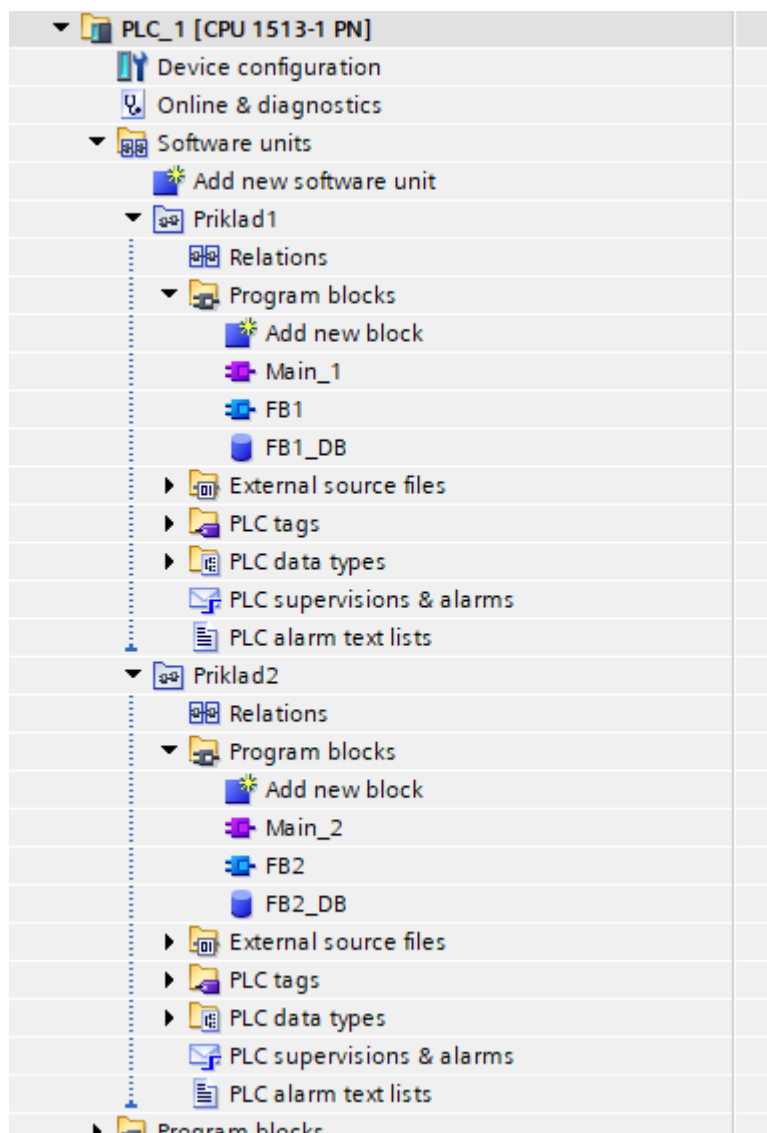
Rozdělením na více částí se dají jednotlivé části zapouzdřit a použít, např. v budoucnu v jiné aplikaci. [22] Příkladem může být program na tisknutí čárových kódů.



OBR 15 PŘÍKLAD STRUKTURALIZACE PROGRAMŮ

4.4 Software Units(S7-1500)

Umožňují mít více projektů na jednom zařízení. Více projektů umožňuje mít více aktivně programujících lidí v jeden moment, ale každý může programovat pouze v rámci své Softwarové jednotky. Mezi jednotlivými jednotkami je možno vytvořit Vztahy a tím je provázat.



OBR 16 PŘÍKLAD SOFTWARE UNITS

Relations			
	Selected unit	Relation type	Accessible element
1	▼ Příklad1		
2	Příklad1	→ Software unit	Příklad2

OBR 17 PŘÍKLAD PROVÁZÁNÍ POMOCÍ VZTAHŮ

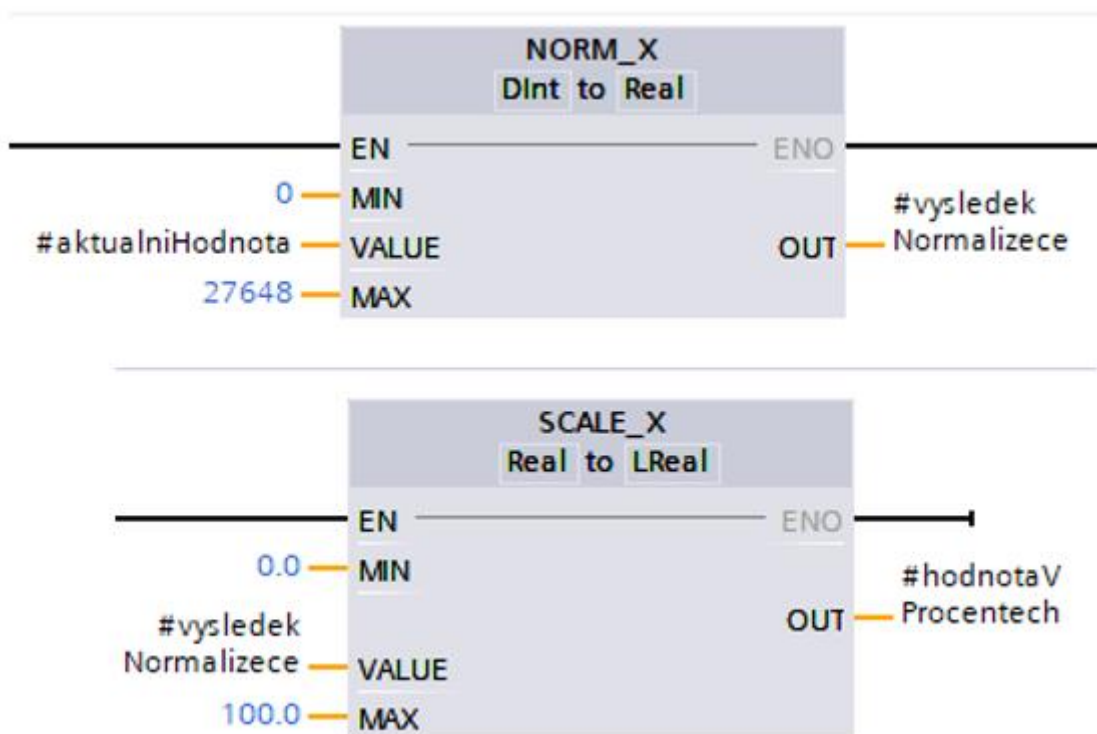
4.5 Škálování analogových vstupů/výstupů

U analogového vstupu/výstupu se v nastavení definuje, o jaký typ se jedná. Podle nastavení je poté pevně definovaný rozsah, viz manuál. [23]

-27648 až 27648 pro symetrické signály, např. -10V až 10V

0 až 27648 pro kladné signály, např. 0V až 10V nebo 4mA až 20mA

Pro škálování v programu je poté možné použít funkce Norm_X a Scale_X

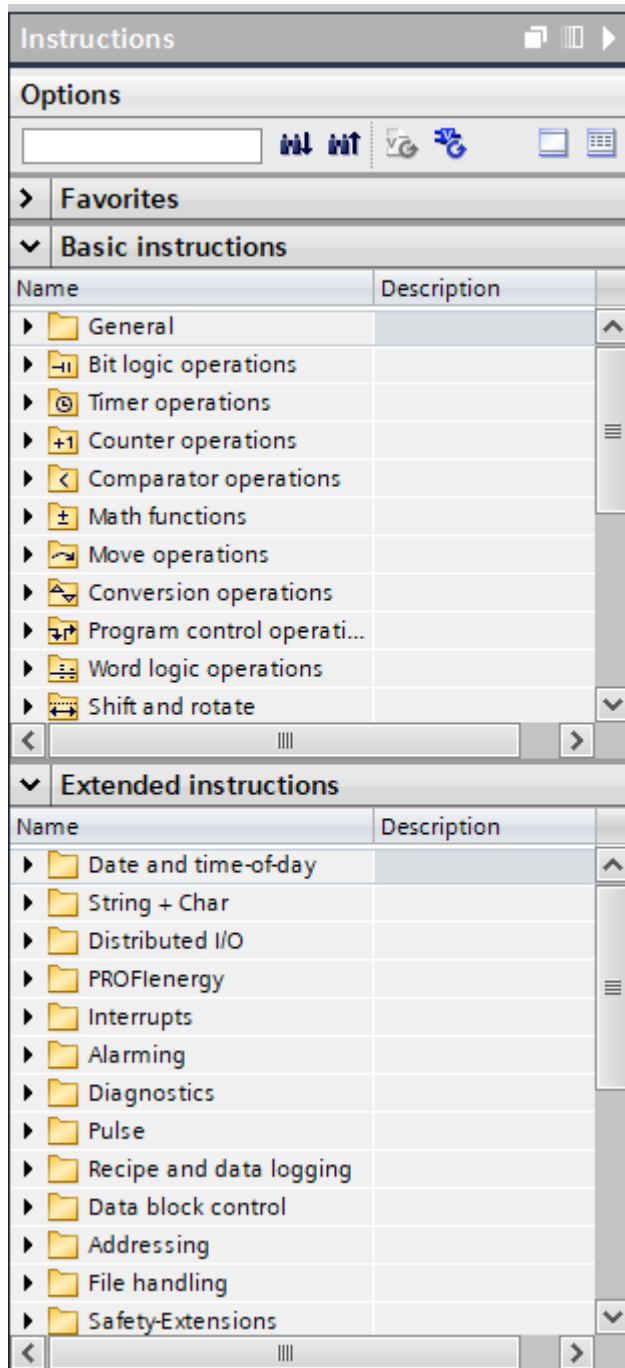


OBR 18 NORM_X A SCALE_X FUNKCE

Norm_X normalizuje hodnotu pomocí definovaných mezí na hodnotu 0 (0 %) až 1 (100 %). Scale_X bere normalizovanou hodnotu a převádí jí na rozsah definovaný uživatelem.

5 Programovací jazyky v TIA Portalu

TIA Portal nabízí několik jazyků, které jsou v základu v souladu s normou 61131-3. V této části nejsou probírané funkce a instrukce, které STEP 7 nabízí, ale jen obecné seznámení s jazyky.



OBR 19 INSTRUKCE V TIA PORTALU

5.1 STL – Statement list (S7-1500)

Jedná se o mnemonický jazyk podobný jazyku IL (Instruction list), ale používá jiné značení operátorů. Používá k operaci dva nebo čtyři registry jako Akumulátory. Na S7-1500 je toto emulováno. Jazyk je podporován pro zpětnou podporu S7-300 a S7-400. [6]

Network 1:				RLO	Value
Comment					
1	A	"Motor_1_Enabled"	1	1	
2	AN	"Motor_1_EmergencyStop"	0	1	
3	JC	n_OK	1		
4	=	"Motor_1_Start"	1	1	
5	AN	"Motor_1_SpeedOK"	0	1	
6	AN	"Motor_1_BreakesEnabled"	0	0	
7	=	"Motor_1_Stop"	0	0	
8	JU	End			
9	n_OK: SET				
10	AN	"Motor_1_BreakesEnabled"			
11	=	"Motor_1_Stop"			
12	End: NOP	0	0		

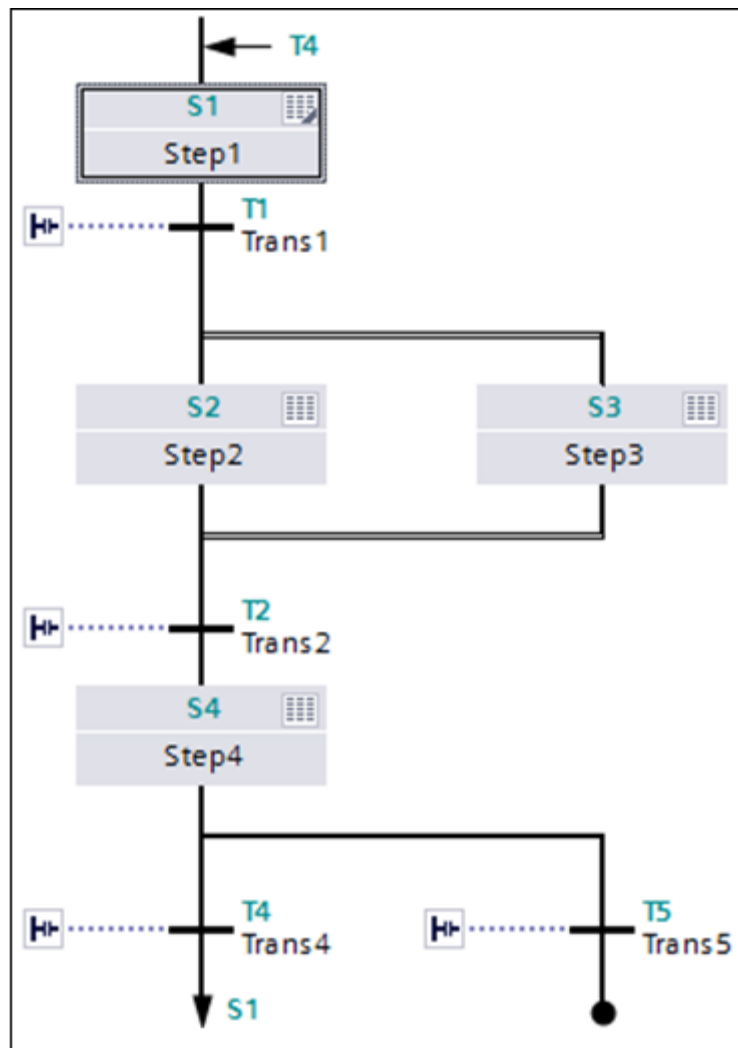
OBR 20 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE STL [24]

5.2 GRAPH (S7-1500)

Vychází z jazyka SFC. Je vhodný k programování sekvencí. Skládá se z kroků, přechodů a akcí. Kroky a přechody definují sekvenci. Přechody se programují v jazyku LAD nebo FBD. Akce mají kvalifikátory, např.: [6]

Kvalifikátor	Vykoná akci
N	Nastaví proměnnou do „1“ po dobu trvání kroku
S	Nastaví proměnnou do „1“
R	Nastaví proměnnou do „0“
D	Nastaví proměnnou do „1“ se zpožděním
L	Nastaví proměnnou do „1“ na definovaný čas

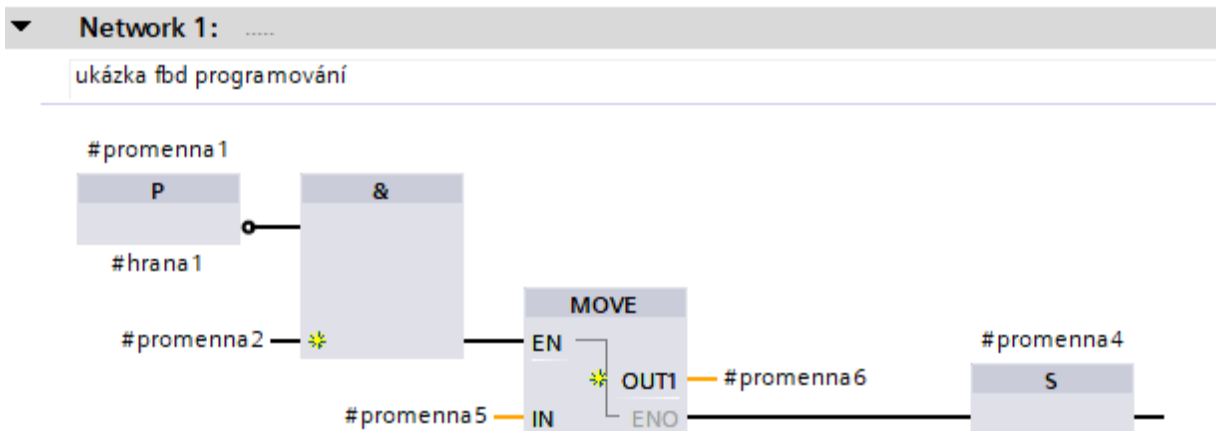
TABULKA 6 KVALIFIKÁTORY AKCÍ GRAPH JAZYKA



OBR 21 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE GRAPH [25]

5.3 FBD – Function Block Diagram

Skládá se z bloků, které se spojují signály. Program je strukturován do Sítí (Network). Každá síť musí být ukončena, tedy obsahovat např. přiřazovací funkci. [6]



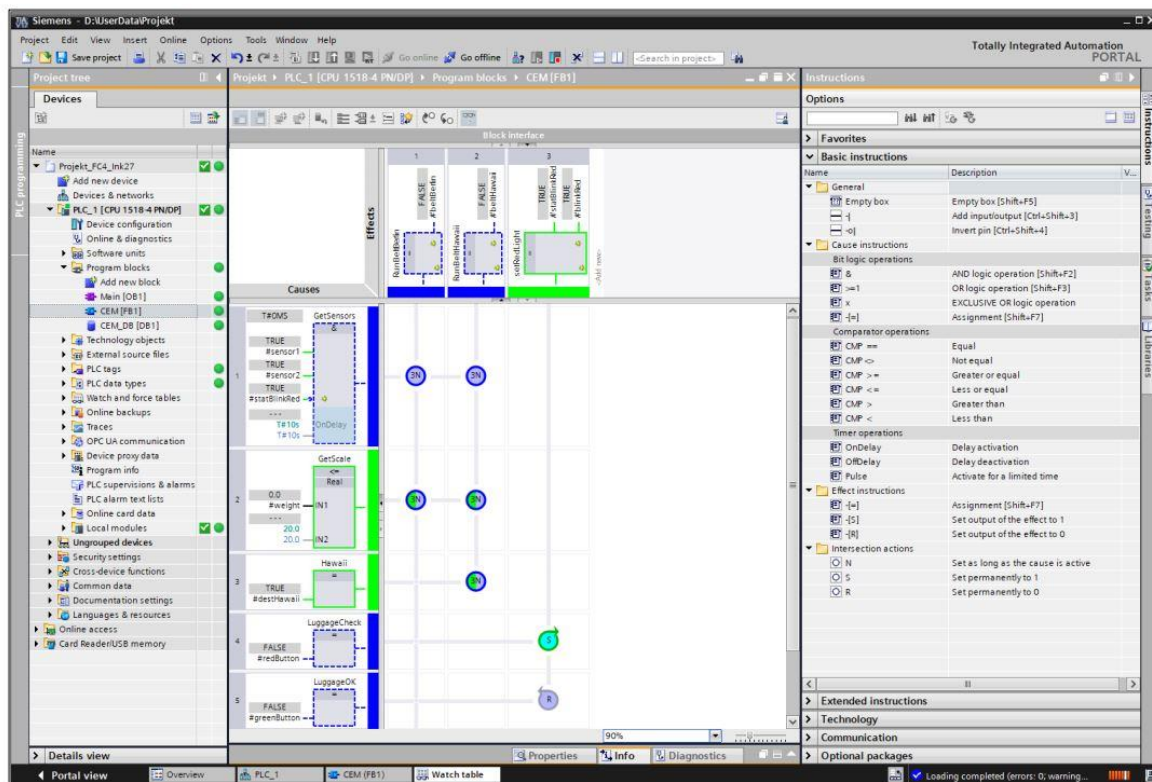
OBR 22 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE FBD

5.4 CFC – Continuous Function Chart (v17, S7-1500)

V tomto jazyce se také používají bloky, jako v jazyce FBD, ale program se vytváří v rámci schématu (chart). Na rozdíl od FBD nemá některé limitace, jako například zpětná vazba. Vyhodnocovací pořadí se dá definovat pro každý blok zvlášť. Je možné vkládat do schématu obrázky jako komentáře. Vhodný pro řízení procesů. [26]

5.5 CEM(v17) – Cause Effect Matrix

Grafický jazyk s maticovou strukturou, díky které je snadno odladitelný. Může běžet na S7-1200 i S7-1500. [26]



OBR 23 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE CEM [27]

5.6 SCL – Structured Control Language

SCL je jazyk, který je podobný Pascalu, vychází z jazyka ST dle normy 61131-3. Je vhodný k programování matematických a datových operací, práci s poli a smyček. Jazyk není moc vhodný pro složitější binární operace, protože je například komplikovanější implementovat náběžné hrany. [3]

```
1 IF #promennal AND NOT #hranal THEN
2     // Výraz
3     ;
4 END_IF;
5 #hranal := #promennal;
```

OBR 24 PŘÍKLAD SCL NÁBĚŽNÁ HRANA

Programové cykly (smyčky):[3]

IF – Větvení závislé na BOOL hodnotě

CASE – Větvení závislé na INT hodnotě

FOR – Smyčka, která vykoná definovaný počet opakování

WHILE – Smyčka se vstupní podmínkou

REPEAT – Smyčka s výstupní podmínkou

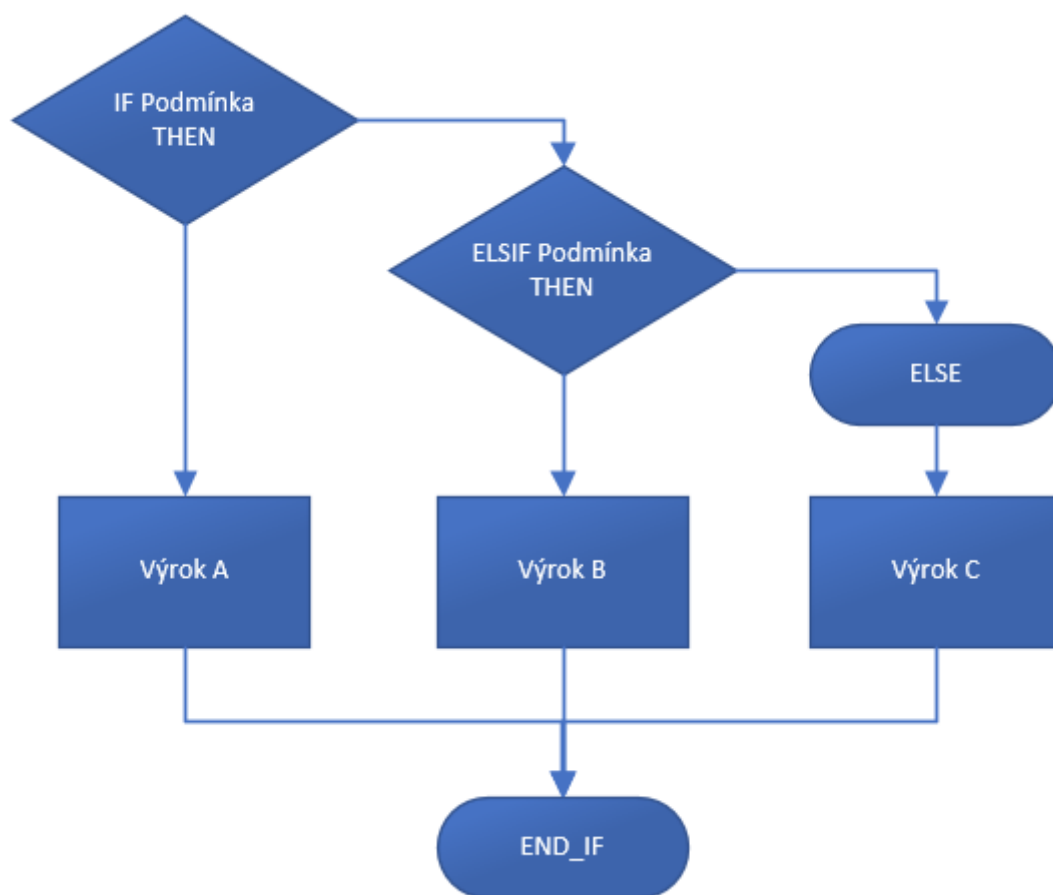
CONTINUE – Ukončí současnou smyčku

EXIT – Opustí smyčku

REGION – Strukturování programu

5.6.1 IF-THEN

IF cyklus vykoná výrok, když je splněna podmínka. Podmínka může být proměnná typu BOOL nebo rovnice s výsledkem typu BOOL. [3] Příklad na obr 24.



OBR 25 VÝVOJOVÝ DIAGRAM FUNKCE IF

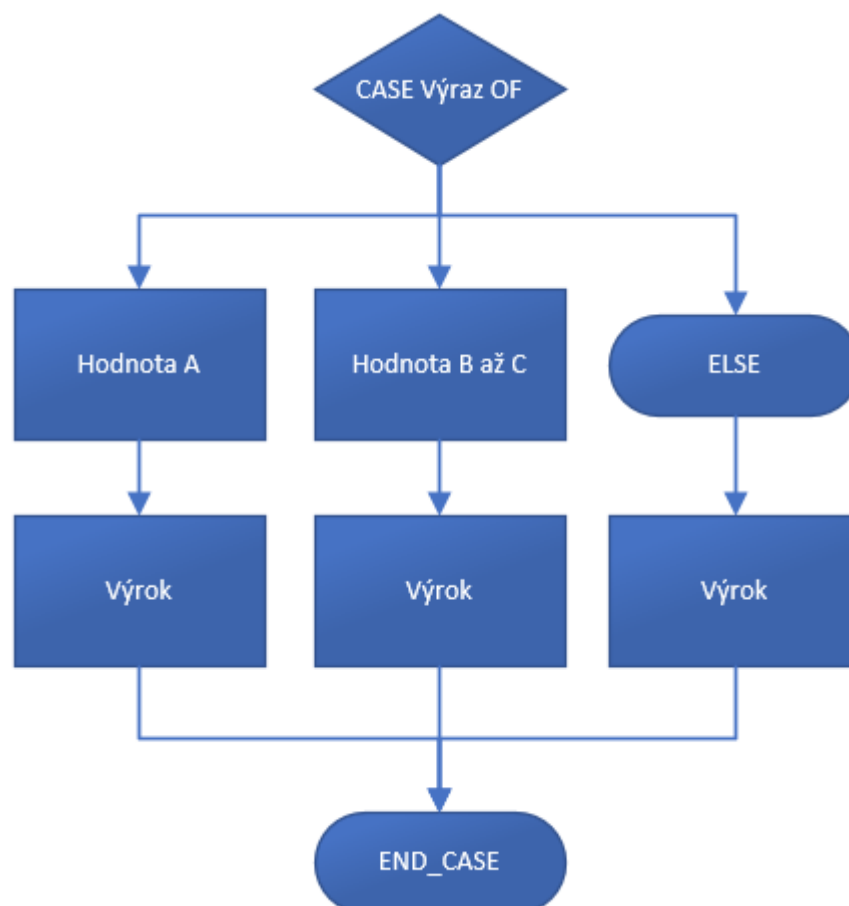
5.6.2 CASE

CASE slouží k vytvoření stavového stroje. Proměnná je porovnávána s více hodnotami. Pokud se hodnota v proměnné shoduje s definovanou hodnotou, výrok k ní připojen bude vykonán. [3]

Používá se i jako sekvencér, kde ve výrocích jsou IF podmínky, a když je jedna splněna, přejde se do dalšího kroku.

```
1 CASE #krok OF
2   1: // Výrok A
3     ;
4   2: // Výrok B
5     ;
6   3..4: // Výrok C
7     ;
8   ELSE // Výrok D
9     ;
10 END_CASE;
```

OBR 26 PŘÍKLAD CASE FUNKCE



OBR 27 VÝVOJOVÝ DIAGRAM FUNKCE CASE

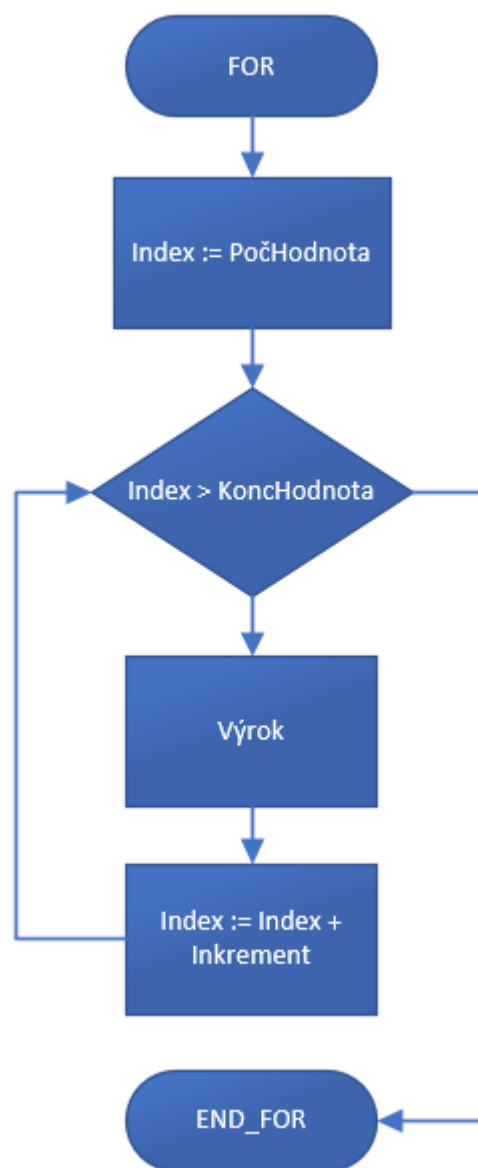
5.6.3 FOR

Instrukce FOR slouží k cyklování ve smyčce při známé hodnotě cyklů.

```
1 FOR #citac := #pocHodnota TO #KoncHodnota BY #inkrement DO
2     // Výrok
3     ;
4 END_FOR;
```

OBR 28 PŘÍKLAD FOR FUNKCE

BY <inkrement> může být vynecháno. Pokud je vynecháno, je inkrement 1. V rámci výroku není možné zasahovat do hodnoty počítání cyklu, kompilátor by zhlásil syntaktickou chybu. [3]



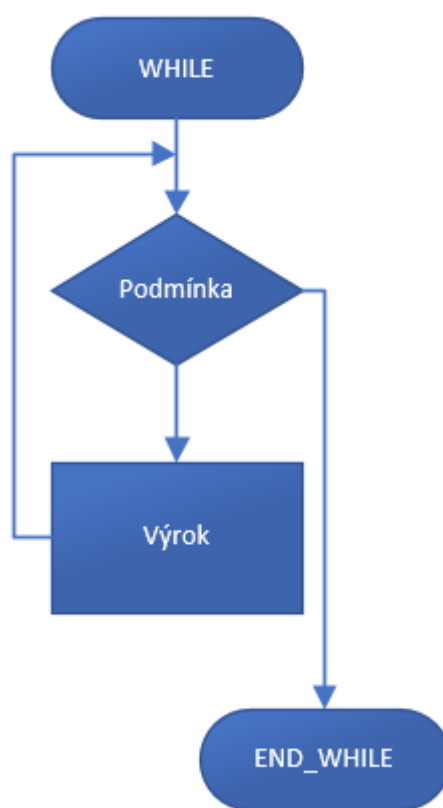
OBR 29 VÝVOJOVÝ DIAGRIAM FUNKCE FOR

5.6.4 WHILE

WHILE cyklus se používá k opakovanému zpracování výroku, dokud je podmínka splněna. Podmínka je proměnná nebo rovnice s datovým typem BOOL. Výraz, který je za DO je opakovaně zpracováván, dokud je podmínka TRUE. Funkce WHILE je možné vnořovat. Funkcemi CONTINUE A EXIT se dá funkce WHILE ukončit. [3]

```
1 WHILE #podmínka DO
2     // Výrok
3     ;
4 END_WHILE;
```

OBR 30 PŘÍKLAD WHILE FUNKCE



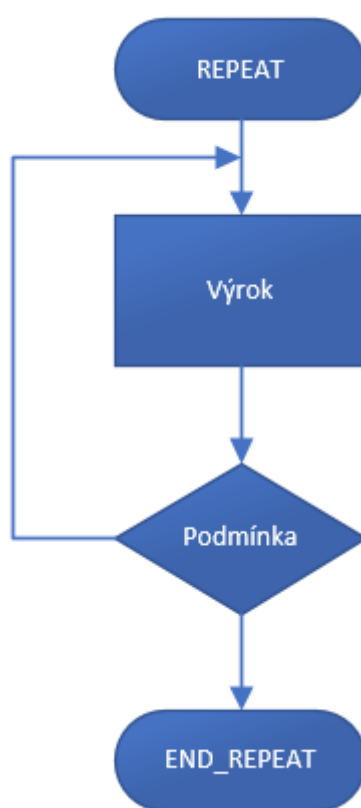
OBR 31 VÝVOJOVÝ DIAGRAM FUNKCE WHILE

5.6.5 REPEAT

REPEAT funguje stejně jako WHILE až na rozdíl, že je podmínka kontrolována po vykonání výroku, to znamená, že výrok bude vždy vykonán při vyvolání funkce. Funkce REPEAT je možné vnořovat. Funkcemi CONTINUE A EXIT se dá funkce REPEAT ukončit. [3]

```
1 REPEAT // Výrok  
2 ;  
3 UNTIL #podmínka  
4 END_REPEAT;
```

OBR 32 PŘÍKLAD REPEAT FUNKCE



OBR 33 VÝVOJOVÝ DIAGRAM FUNKCE REPEAT

5.6.6 CONTINUE

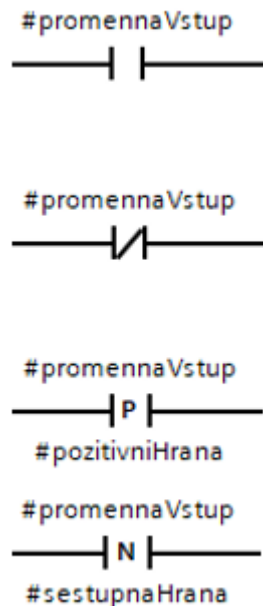
Slouží k ukončení současného cyklu FOR, WHILE nebo REPEAT funkce. [3]

5.6.7 EXIT

Slouží k ukončení celé smyčky FOR, WHILE nebo REPEAT funkce. [3]

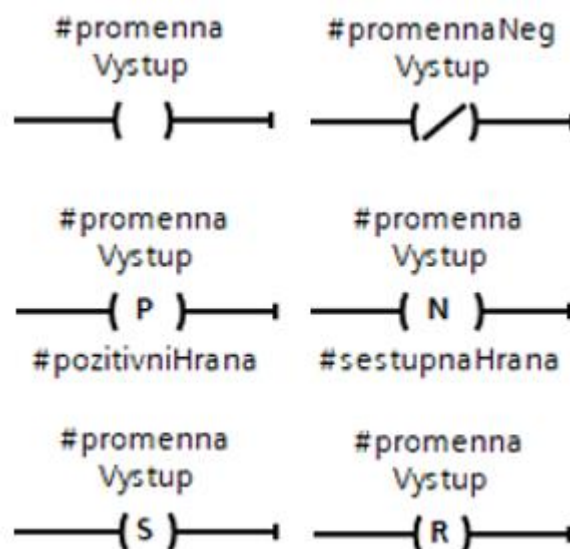
5.7 LAD – Ladder Diagram

Skládá se z proudových cest, do kterých se vkládají komponenty. Komponenty mohou být kontakty, bloky pro funkce a funkční bloky a cívky. Kontakty existují ve dvou variantách, a to Spínací (normálně otevřený) a Rozpínací (normálně zavřený). Případně se dá v rámci kontaktu kontrolovat náběžná a sestupná hrana, kde je potom kontakt aktivní po jeden cyklus PLC. [3]



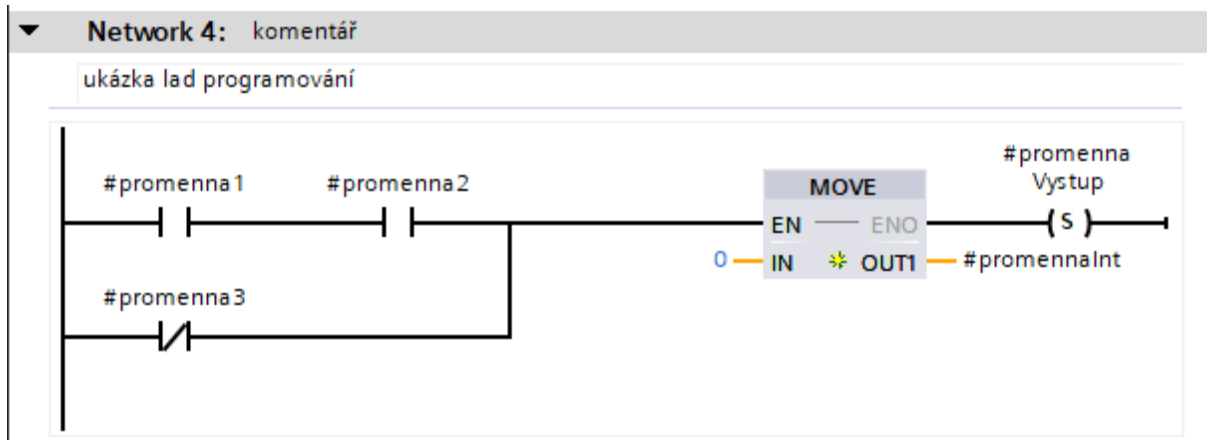
OBR 34 KONTAKTY V JAZYCE LAD

Cívky fungují jako výstupy. Kromě Spínací a Rozpínací cívky je zde i Set a Reset cívka [3] (u jiných výrobců může být jako Latch a Unlatch).



OBR 35 CÍVKY V JAZYCE LAD

Pomocí bloků se volají funkce a funkční bloky. Vyhodnocení programu probíhá zleva doprava a odshora dolů, takže platí dominance polohou. [3]



OBR 36 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE LAD

6 Odladování

K odladování programu je v TIA Portalu několik nástrojů. Počínaje hlášením kompilátoru, konče sekvenčním testováním v PLCSIM. Tyto možnosti se dají rozdělit na Offline a Online.

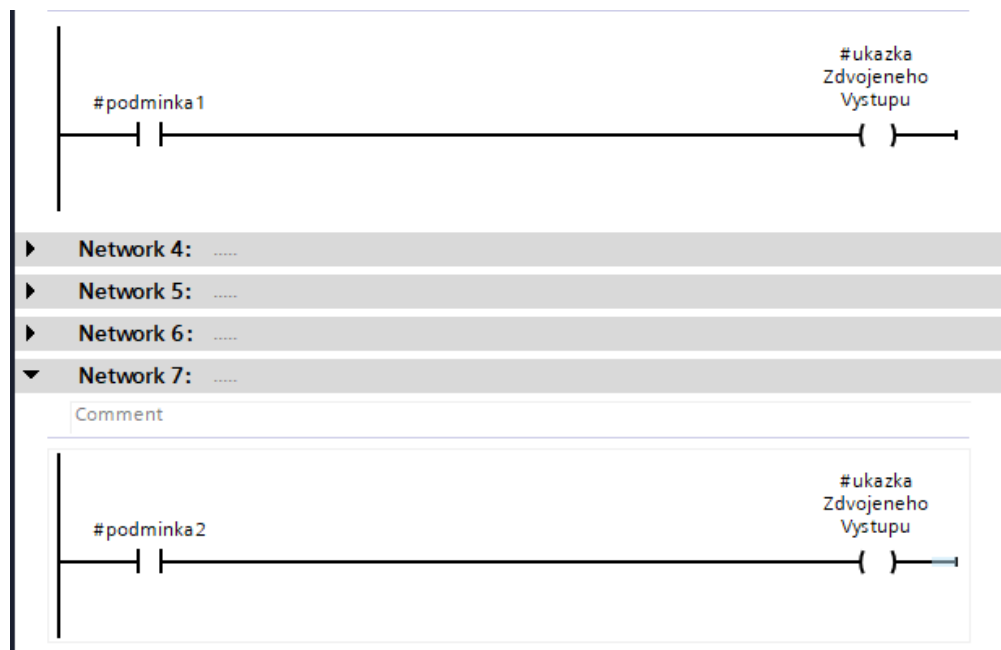
6.1 Offline

Offline kontrola spočívá převážně ve zkušenostech programátora, který může kontrolovat hlášení kompilátoru, správné přiřazení datových typů a operace s nimi.

6.1.1 KOMPILÁTOR

Před každým nahráním programu do PLC se program kompiluje. Kompilátor poté vypíše případné chyby nebo varování. Jedná se převážně o syntaktické chyby nebo např. sériové výstupy v LAD programu. Vyhodnocení varování záleží primárně na zkušenostech programátora.

Některé chyby ale kompilátor nepodchytí. Například zdvojený výstup v jazyce LAD, kde z hlediska vykonávání programu, první řádek nenastaví výstup nikdy.



OBR 37 ZDVOJENÝ VÝSTUP V LAD

!	Path	Description
✓	PLC_1	
✓	Program blocks	
✓	Main (OB1)	Block was successfully compiled.
✓		Compiling finished (errors: 0; warnings: 0)

OBR 38 VÝSLEDEK KOMPILACE ZDVOJENÉHO VÝSTUPU

6.1.2 OVERLAPPING

V rámci definování globálních proměnných je možné jejich překrytí. Kompilátor to nevyhodnocuje jako chybu. Toto je možné zjistit pomocí nástroje Cross-Reference. Nástroj Cross-Reference slouží primárně k ukázaní, kde všude byla proměnná použita, ale je možné zobrazit „Overlapping access“, a tím zobrazit paměťové překrytí.

Default tag table							
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...
1	iBit	Bool	%IO.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	iWord	Word	%IWO	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	<Add new>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

OBR 39 DEFINOVÁNÍ PŘEKRÝVAJÍCÍCH SE PROMĚNNÝCH

Object	Reference location	Reference type	As	Access	Address	Type	Device	Path
iWord					%IW0	Word	PLC_1	PLC_1\PLC tags\Default tag table
Main	@Main ▶ NW3	Used by		Write	%OB1	LAD-Organization...	PLC_1	PLC_1\Program blocks
<Add new source object>								
Overlapping access of: iWord <input type="checkbox"/> Write access only								
Object	Reference location	Reference type	As	Access	Address	Type	Device	Path
iBit					%I0.0	Bool	PLC_1	PLC_1\PLC tags\Default tag table
Main	@Main ▶ NW2	Used by		Write	%OB1	LAD-Organization...	PLC_1	PLC_1\Program blocks

OBR 40 CROSS-REFERENCE

6.1.3 MATEMATICKÉ OPERACE

Operace se provádí v nejvyšším datovém typu, který se v operaci nachází. Pokud má být výsledek typu REAL, musí být tento typ v samotné operaci.

1	#promennaINT := #int8 / #int3;	▼	#promennaINT	2
			#int8	8
			#int3	3
2	#promennaREAL := #int8 / #int3;	▼	#promennaREAL	2.0
			#int8	8
			#int3	3
3	#promennaREAL2 := INT_TO_REAL(#int8) / #int3;	▼	#promennaR...	2.666667
			INT_TO_REAL	8.0
			#int8	8
			#int3	3

OBR 41 UKÁZKA DĚLENÍ RŮZNÝCH DATOVÝCH TYPŮ

6.2 Online

Možností pro online odlaďování je více. Zvláště proto, že je možno sledovat vyhodnocování programu pomocí Monitoringu a modifikovat proměnné pomocí tabulek.

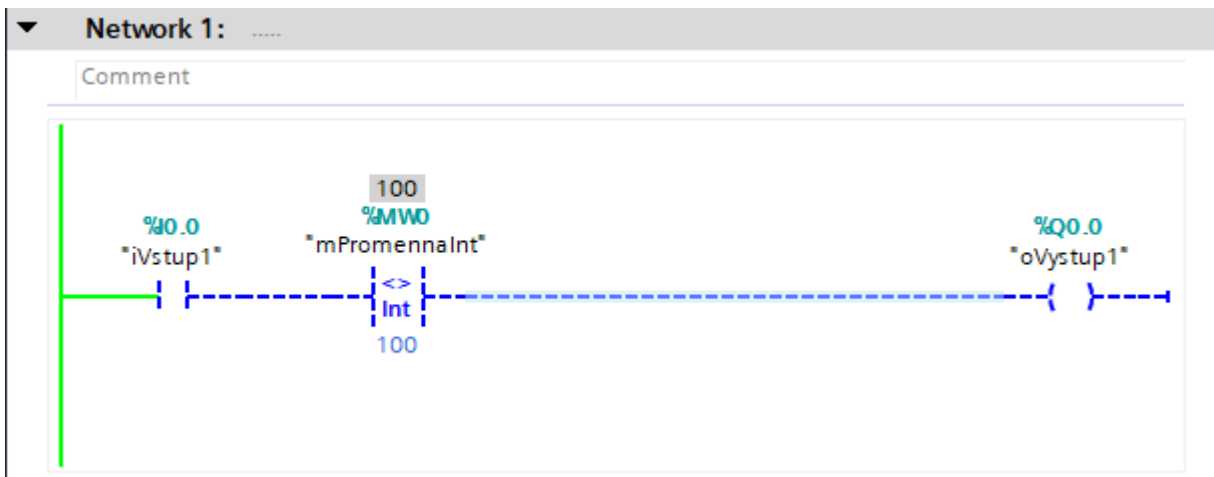
6.2.1 MONITOR

Po navázání komunikace je možné zapnout Monitoring, který zobrazí vyhodnocování programu v reálném čase. [6]

U grafických jazyků je daleko snazší poznat, kde program „stojí“. Barevně je odděleno to, co je splněno (zelená) a to, co ne (modrá).

1	IF "iVstup1" THEN	▶	Result	FALSE
2	"oVystup1" := 1;		"oVystup1"	
3	ELSE			
4	"oVystup1" := 1;		"oVystup1"	TRUE
5	END_IF;			
6				
7				

OBR 42 MONITORING V JAZYCE SCL



OBR 43 MONITORING V JAZYCE LAD

6.2.2 FORCE TABLE

Slouží k nucenému nastavování stavu vstupů a výstupů. V rámci tabulky je možno monitorovat ostatní proměnné. Forcing napevno přiřadí vstupu nebo výstupu hodnotu. Forcing zůstává aktivní i po resetování CPU nebo ztrátě spojení. Vypnout se dá pouze v tabulce. Je nezávislý na stavech PLC. Vnucování vstupu přiřadí dané proměnné žádanou hodnotu a reálný stav vstupu je ignorován. Vnucování výstupu reálně sepne daný fyzický výstup. [6]

	i	Name	Address	Display format	Monitor value	Force value	F
1	<input checked="" type="checkbox"/>	"oVystup1":P	%Q0.0:P	Bool	<input type="checkbox"/>	TRUE	<input checked="" type="checkbox"/>
2		"oVystup2":P	%Q0.1:P	Bool	<input type="checkbox"/>		<input type="checkbox"/>
3		"iVstup1":P	%I0.0:P	Bool	<input type="checkbox"/>		<input type="checkbox"/>
4		// komentar					
5		"mPromennaInt"	%MWO	DEC+/-	100		<input type="checkbox"/>
6			<Add new>				<input type="checkbox"/>

OBR 44 UKÁZKA FORCE TABULKY

6.2.3 WATCH TABLE

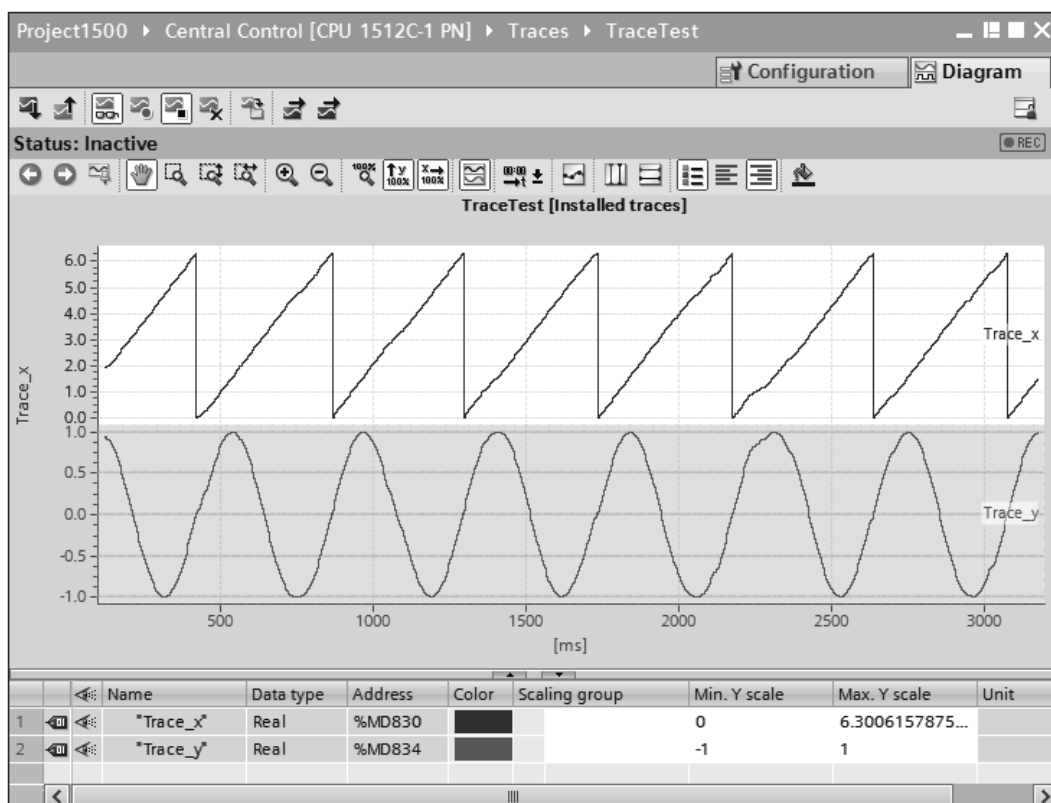
Slouží primárně k monitorování proměnných, ale je možné některé proměnné i modifikovat. K tomu slouží sloupec Modify. U S7-1200 je možné při STOP režimu PLC zapnout rozšířený režim a testovat tak výstupy. [6]

	i	Name	Address	Display format	Monitor value	Monitor with trig...	Modify with trigge	Modify value	
1		"iVstup1"	%I0.0	Bool	<input type="checkbox"/> FALSE	Permanent	Once only, at star...		<input type="checkbox"/>
2		"mPromennaInt"	%MWO	DEC+/-	100	Permanent	Permanent		<input type="checkbox"/>
3		"mPromennaBool2"	%M2.1	Bool	<input type="checkbox"/> FALSE	Permanent	Permanent		<input type="checkbox"/>
4		// komentar							
5	<input checked="" type="checkbox"/>	"oVystup1"	%Q0.0	Bool	<input checked="" type="checkbox"/> FALSE	Permanent	Permanent		<input type="checkbox"/>
6			<Add new>						<input type="checkbox"/>

OBR 45 UKÁZKA WATCH TABULKY

6.2.4 TRACE

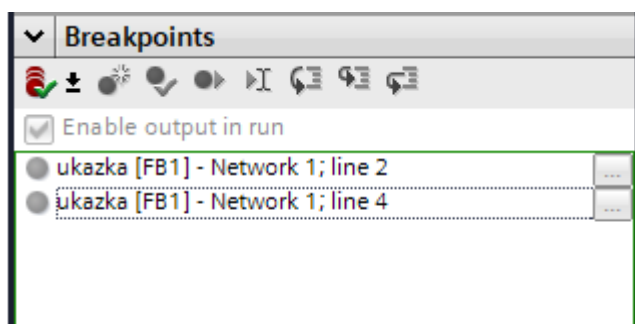
Používá se ke sledování změny proměnné v čase. Nahrávání se dá ukládat a externě zpracovávat. [6]



OBR 46 UKÁZKA NAHRANÝCH HODNOT [6]

6.2.5 BREAKPOINTS (S7-1500)

Slouží k odlaďování programů v textových jazycích SCL a STL. Vložením Breakpointu do programu se na daném řádku přepne PLC do HOLD módu a dochází tím ke krokování programu. Tím je zajištěna možnost odlaďování smyček, jelikož je možné je krokovat po jednotlivých cyklech. Bodů přerušení je možné vložit více.



OBR 47 UKÁZKA BREAKPOINTŮ

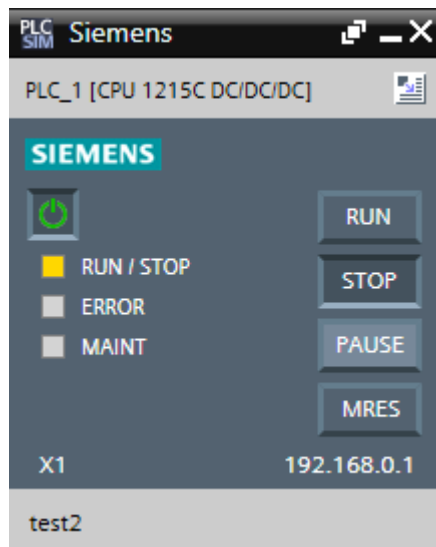
```
1 IF "iVstup1" THEN
2     "oVystup1" := 1;
3 ELSE
4     "oVystup1" := 1;
5 END_IF;
6
7
8
```

OBR 48 KROKOVÁNÍ BREAKPOINTŮ

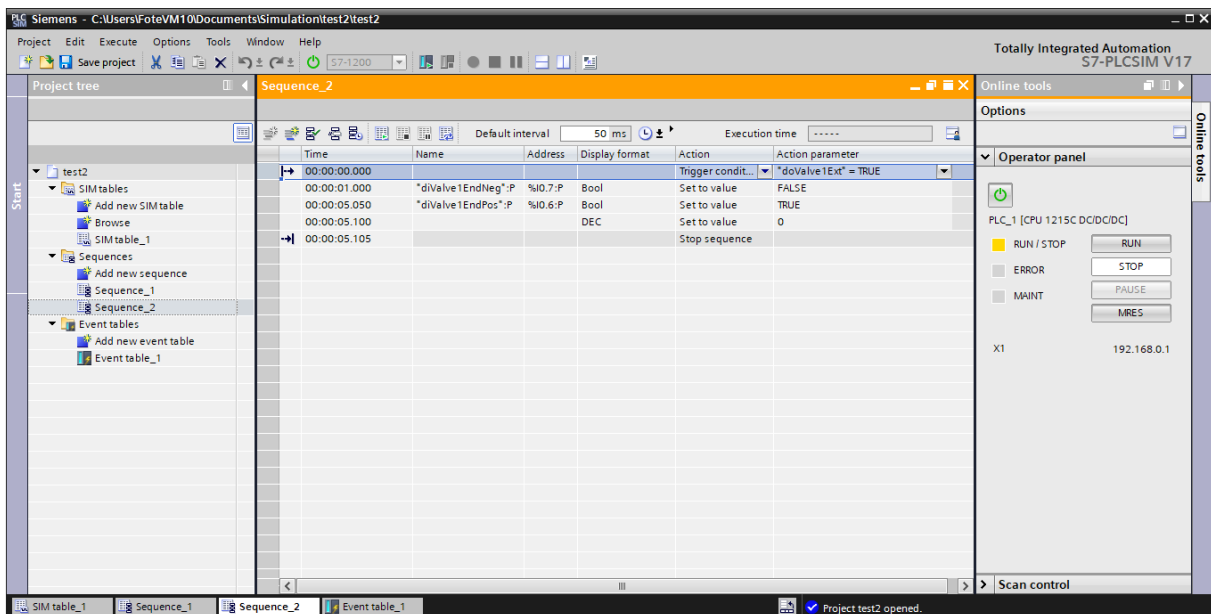
6.3 S7-PLCSIM

Slouží k odladění programu bez nutnosti mít reálný hardware. Některé funkce se simulovat nedají, např. paměťová karta nebo webserver. Uživatelský program, který je zkompilovaný bez chyb, je možné nahrát do simulátoru. Nehodí se na testování časově závislých procesů, protože exekuce simulace závisí na výkonu počítače, na kterém simulace běží.

PLCSIM má dvě rozhraní. Kompaktní rozhraní, které slouží jen jako virtuální PLC a Projektové rozhraní, které umožňuje vytvořit simulační projekt. V rámci projektu je možné modifikovat tagy včetně vstupů a vytvářet sekvence modifikací. [6]



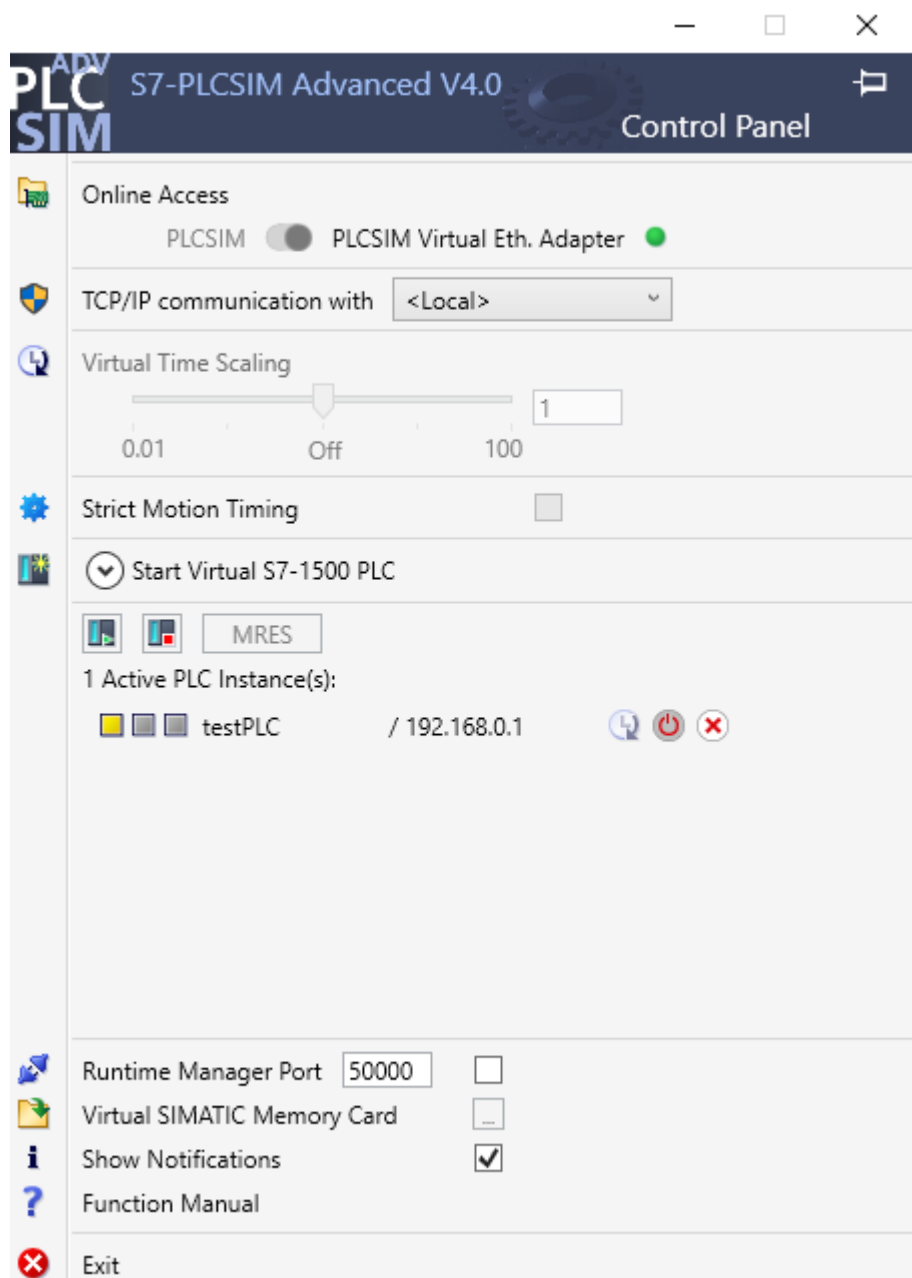
OBR 49 KOMPAKTNÍ ROZHRANÍ - PLCSIM



OBR 50 PROJEKTOVÉ ROZHRANÍ - PLCSIM

6.4 S7-PLCSIM Advanced

Slouží k vytvoření virtuálních S7-1500 a S7-ET 200SP kontrolérů. Nenabízí uživatelské rozhraní k práci s tagy, ale nabízí API, které to umožňuje. V rámci simulace vytvoří instanci kontroléru. Simulace obsahuje např. i OPC UA komunikaci, webserver, propojení s HMI. Dá se propojit se simulačními softwary, např. SIMIT, TX Process Simulate, NX MCD. [28]



OBR 51 S7-PLCSIM ADVANCED

6.5 Metodika uvádění do provozu a odlaďování programu

1. I/O Testing

Testuje se, zdali fungují všechny vstupy a výstupy tak, jak mají. U senzorů probíhá kontrola, jestli jsou v souladu se vstupem PNP nebo NPN. U výstupů, kde se zapíná aktorika, jako například stykač spínající čerpadlo, se kontroluje i směr otáčení čerpadla.

2. Bezpečnost

Kontroluje se odpojení Ovládacího napětí při stisknutí Total stopu, případné „recovery“ funkce v případě procesů nebo použití robotů. Po opětovném zapnutí Ovládacího napětí je důležité, aby stroj nepokračoval v předchozí činnosti, které stisknutí Total stopu vyvolala, ale je žádoucí, aby bylo potřeba stroj uvést do počáteční polohy, neboli reinitializovat.

3. Kontrola programu

V této fázi se testuje program a odlaďují se případné chyby. Testují se i případné nežádoucí stavy, jako přepnutí přepínače Automat/Manuál v průběhu výroby. Všechny tyto stavy musí být ošetřeny.

4. Uvedení do provozu

Spočívá ve spolupráci s operátory a údržbáři v úpravě rozhraní tak, aby splňovalo požadavky v rámci ovládání a servisu. Dále je program uveden do stavu, aby splňoval výrobní požadavky, jako například čas cyklu.

7 Návrh řízení zobecněného jednoúčelového stroje

Řízení stroje bylo navrženo na základě zjednodušené dokumentace. V rámci dokumentace bylo vytvořeno pneumatické schéma, schémata zapojení vstupů/výstupů PLC a schéma zapojení bezpečnostního relé, které jsou přiloženy jako přílohy.

7.1 Předpoklad bezpečnosti

K zajištění bezpečnosti stroje slouží tři prvky, a to:

- 1 – Bezpečnostní relé, které slouží k vyhodnocování stisknutí Total Stopu.
- 2 – Bezpečnostní ventil Festo MS6-SV-1/2-E, který slouží k rychlému odvětrání v případě nouzového zastavení.
- 3 - Použití 5/3 ventilů, které mají střední polohu odvětranou.

Po stisku Total stopu budou tedy válce uvolněny.

7.2 Pneumatické komponenty

S ohledem na bezpečnost bylo vytvořeno pneumatické schéma v programu Festo FluidDraw5, viz příloha.

Název	Počet kusů	Popis
DSBC-32-80-D3-PPVA-N3	3	Pneumatický válec
GRLA-1/8-B	6	Jednosměrný škrťací ventil
CPE14-M1BH-5/3E-1/8	3	Ventil 5/3, střední poloha odvětraná
MS6-EM1	1	Ruční ventil
MS6-SV-1/2-E-10V24-AD1	1	Bezpečnostní softstart ventil
MS6-LFR-3/8-D6-ERMA-AS	1	Redukční ventil

TABULKA 7 SEZNAM PNEUMATICKÝCH KOMPONENT

7.3 Vstupy a výstupy PLC

V programu EPLAN Education 2.9 byla vytvořena schémata zapojení bezpečnostního relé, vstupů a výstupů. Jako PLC byl zvolen S7-1215C DC/DC/DC.

Vstupy a výstupy PLC byly pojmenovány podle směrnic z kapitoly 4.

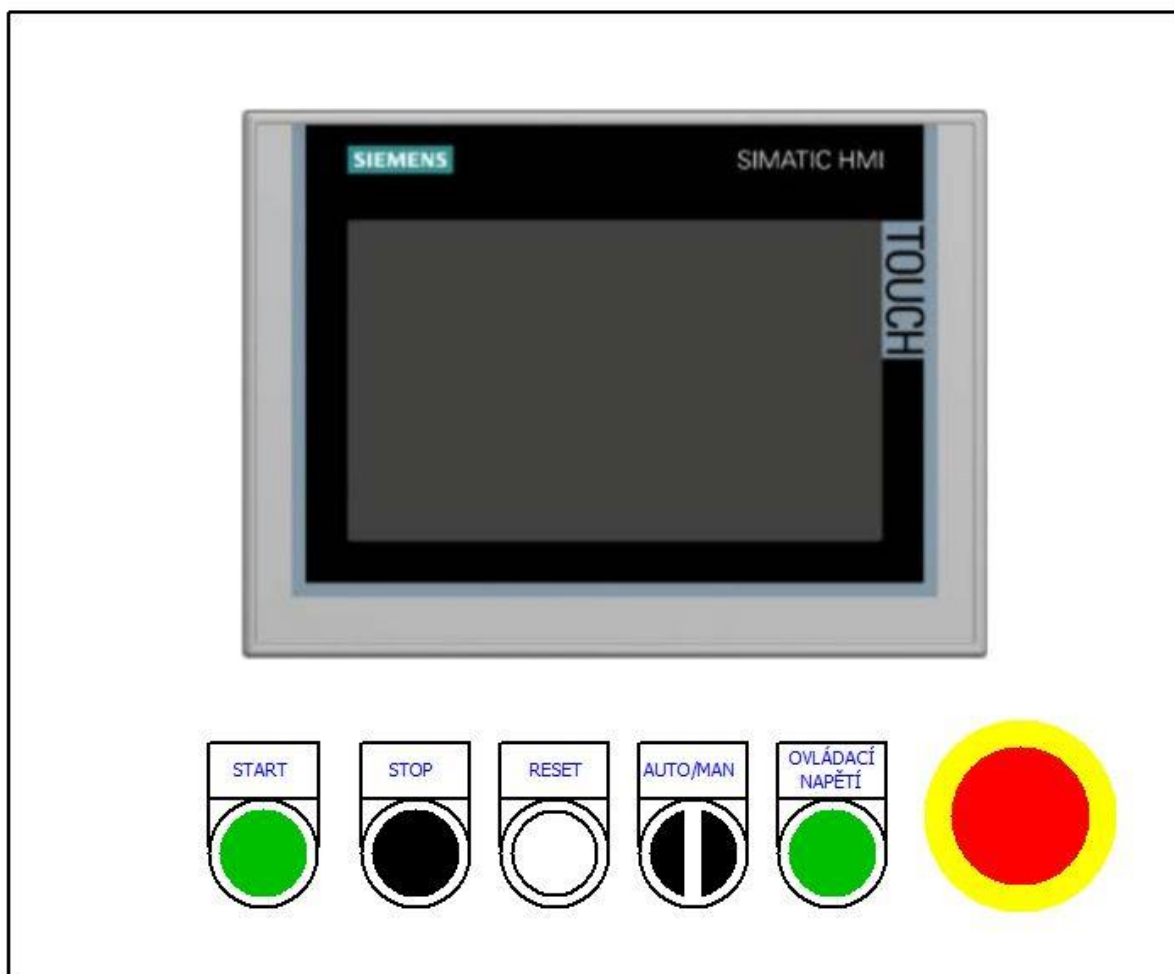
PLC tag	PLC adresa	Označení na schématu	Popis
iBezpecnost	I0.0	I001	Bezpečnost OK
iTlakVzduchu	I0.1	I002	Tlak vzduchu ok
iStart	I0.2	I003	Tlačítko START
iStop	I0.3	I004	Tlačítko STOP rozpínací
iReset	I0.4	I005	Tlačítko Reset
iAuto	I0.5	I006	Přepínač Auto/Man, pozice Auto
iManInvert	I0.6	I007	Přepínač Auto/Man, pozice Man, inv
iValec1KoncPos	I0.7	I008	Konc senzor Válec 1 vysunut
iValec1KoncNeg	I1.0	I009	Konc senzor Válec 1 zasunut
iValec2KoncPos	I1.1	I010	Konc senzor Válec 2 vysunut
iValec2KoncNeg	I1.2	I011	Konc senzor Válec 2 zasunut
iValec3KoncPos	I1.3	I012	Konc senzor Válec 3 vysunut
iValec3KoncNeg	I1.4	I013	Konc senzor Válec 3 zasunut

TABULKA 8 TABULKA VSTUPŮ

PLC tag	PLC adresa	Označení na schématu	Popis
oStartLED	O0.0	O001	LED Signálka
oValec1Vysunout	O0.1	O002	Vysunout válec 1
oValec1Zasunout	O0.2	O003	Zasunout válec 1
oValec2Vysunout	O0.3	O004	Vysunout válec 2
oValec2Zasunout	O0.4	O005	Zasunout válec 2
oValec3Vysunout	O0.5	O006	Vysunout válec 3
oValec3Zasunout	O0.6	O007	Zasunout válec 3

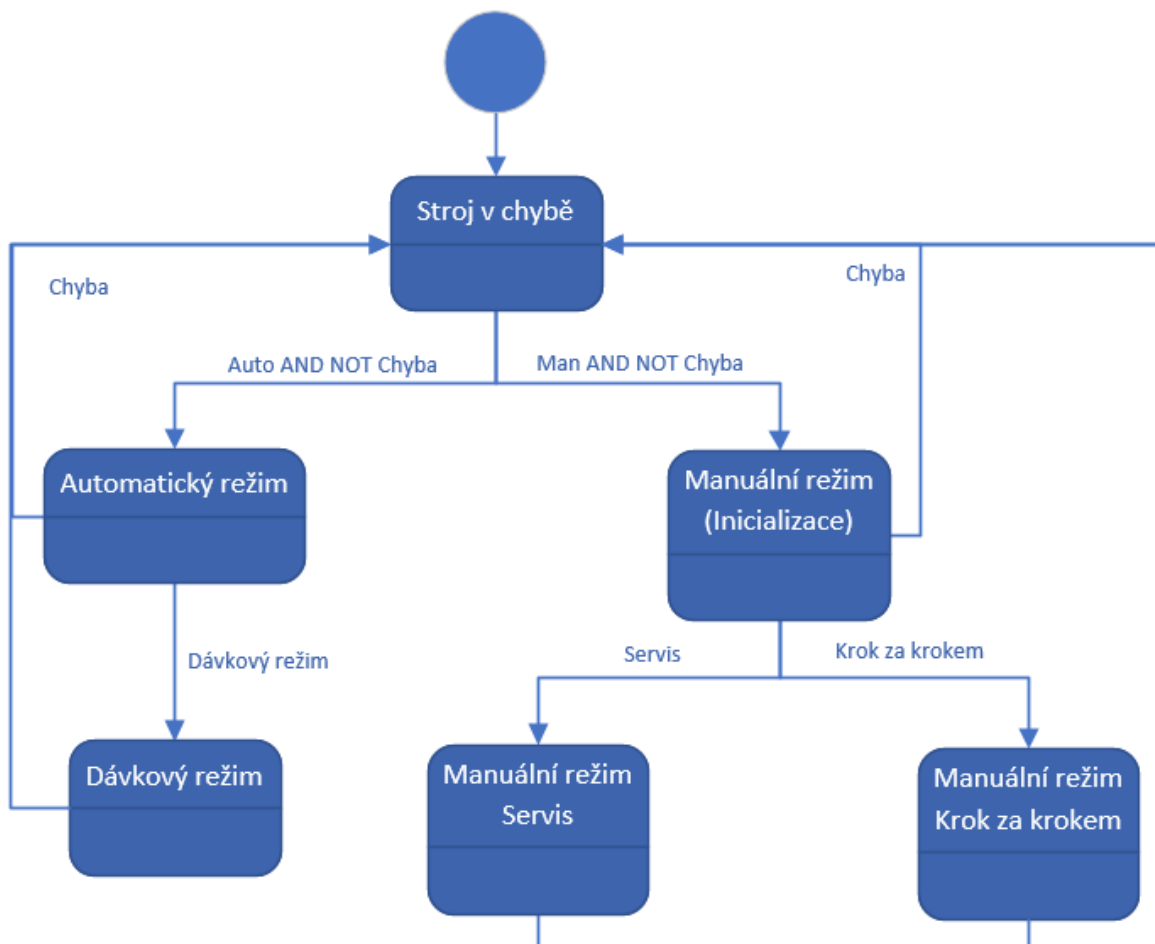
TABULKA 9 TABULKA VÝSTUPŮ

V programu EPLAN Education 2.9 byl vytvořen i návrh ovládacího panelu.



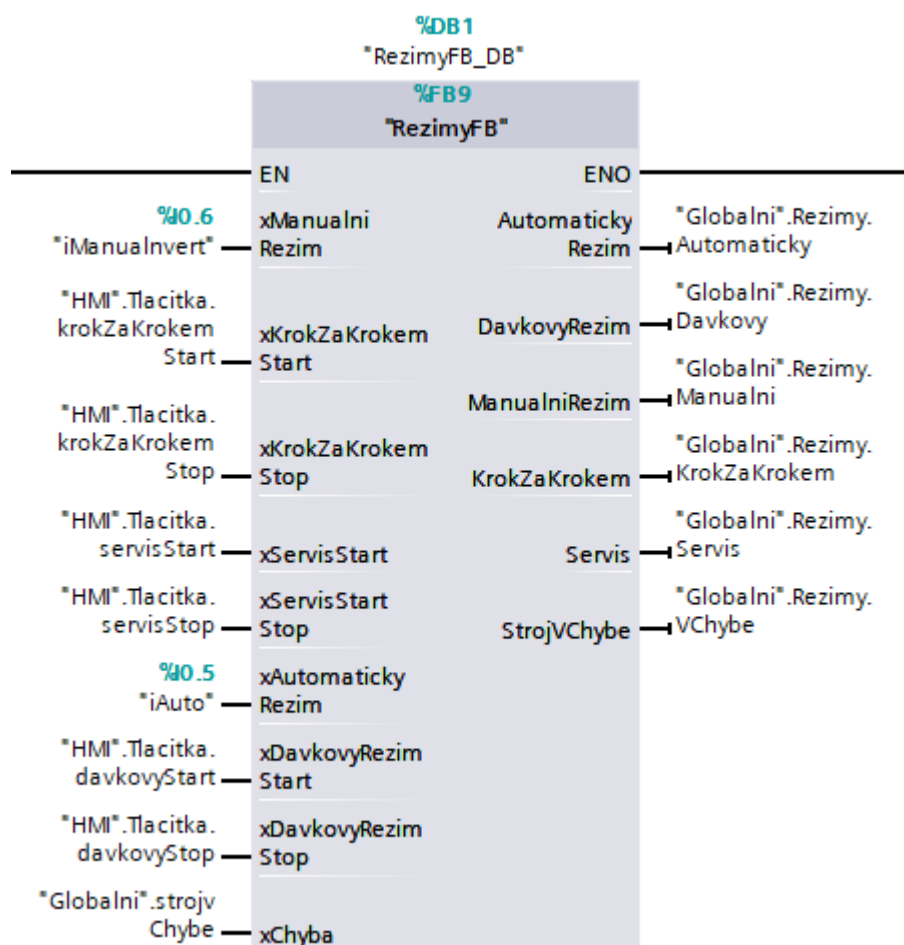
OBR 52 NÁVRH OVLÁDACÍHO PANELU

7.4 Provozní režimy



OBR 53 PROVOZNÍ REŽIMY

K obsluze slouží jeden Funkční blok, který pracuje s reálnými vstupy a vstupy z HMI. Výstupem jsou jednotlivé pracovní režimy. Pouze jeden může být v jeden čas aktivní. Ačkoliv jsou režimy v jedné struktuře, výstupy v této struktuře nejsou z důvodu možnosti zjištění aktuálního režimu při odlaďování pomocí Monitoringu.



OBR 54 FUNKČNÍ BLOK PROVOZNÍCH REŽIMŮ

7.4.1 AUTOMATICKÝ REŽIM

V automatickém režimu probíhá hlavní výrobní cyklus, který se stále opakuje. Když je stroj připraven, signální LED pod Start tlačítkem bliká. když stroj vykonává automatický výrobní cyklus, LED signálka svítí.

7.4.2 DÁVKOVÝ REŽIM

Dávkový režim, na rozdíl od automatického, se opakuje jen pro specifikovaný počet opakování.

7.4.3 MANUÁLNÍ REŽIM

Slouží k uvedení stroje do počáteční polohy pomocí Inicializace.

7.4.4 KROK ZA KROKEM

Tento režim čeká na stisknutí tlačítka operátorem s každým krokem výrobní sekvence.

7.4.5 SERVIS

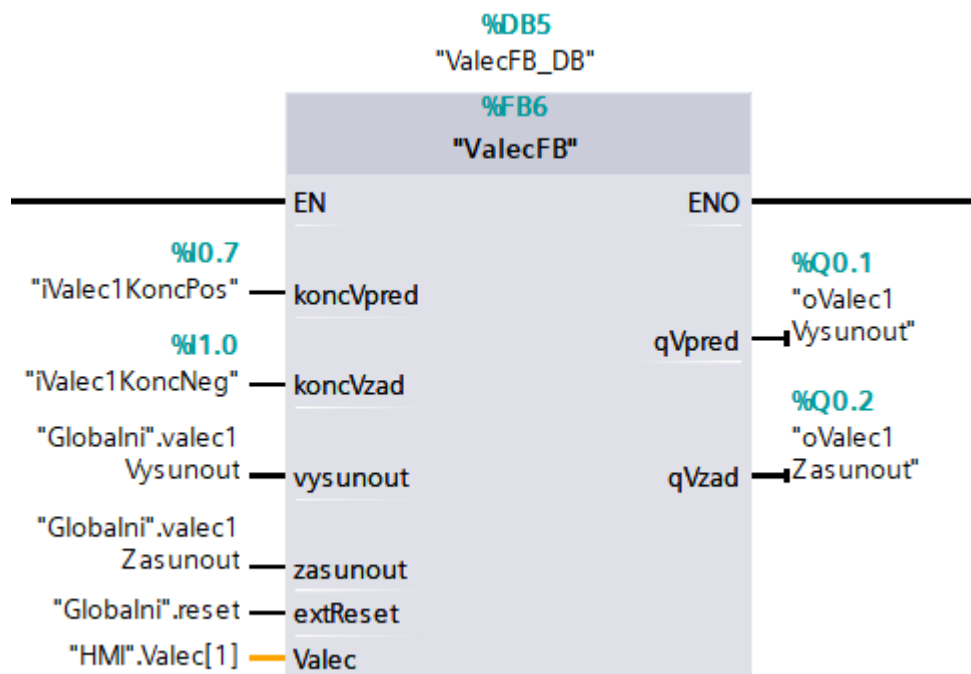
V rámci tohoto režimu je možné manuálně ovládat pneumatické válce přes servisní obrazovku na HMI panelu

7.5 Výrobní cyklus

V rámci výrobního cyklu bylo potřeba zajistit možnost parametrizovat výrobní sekvenci. To způsobilo, že v rámci sekvence je potřeba v jednotlivém kroku sledovat, zdali jsou vstupy od koncových senzorů v souladu s parametrizací a současně se s ní porovnává i vyhodnocování výstupů. Celé to muselo být rozšířeno o inicializaci, aby se stroj mohl uvést do počáteční polohy. Inicializace je též parametrizovatelná, aby byla zajištěna možnost např. vyhození dílu .

Byly tedy vytvořeny Funkční bloky k řešení více zmíněných problémů.

7.5.1 ŘÍZENÍ VÁLCŮ



OBR 55 FUNKČNÍ BLOK ŘÍZENÍ VÁLCE

Funkční blok k řízení válců urychluje odladění, protože stačí odladit pro jeden válec a následně je možné ho použít i pro ostatní válce.

Obsahuje logiku řízení, kontrolu dojetí do konečné polohy, kontrolu senzorů a obsahuje vstupně výstupní proměnnou, která slouží k napárování k servisní obrazovce na HMI. Funkční blok byl psán v jazyce Ladder.

7.5.2 PARAMETRIZACE VÝROBNÍHO CYKLU A INICIALIZACE

Jelikož se jedná o specifickou parametrizaci, byl použit přímý přístup v rámci proměnných namísto použití lokálních. Funkční blok byl psán v jazyce SCL.

```
1 IF ("HMI".zadanyKrok <> "HMI".zvolenyKrok) OR "HMI".nacist THEN
2     "HMI".zadanyKrok := "HMI".zvolenyKrok;
3     "HMI".valec1 := "HMI".kroky["HMI".zadanyKrok].krok.valec1;
4     "HMI".valec2 := "HMI".kroky["HMI".zadanyKrok].krok.valec2;
5     "HMI".valec3 := "HMI".kroky["HMI".zadanyKrok].krok.valec3;
6 END_IF;
7
8 IF "HMI".ulozit THEN
9     "HMI".kroky["HMI".zadanyKrok].krok.valec1 := "HMI".valec1;
10    "HMI".kroky["HMI".zadanyKrok].krok.valec2 := "HMI".valec2;
11    "HMI".kroky["HMI".zadanyKrok].krok.valec3 := "HMI".valec3;
12 END_IF;
13
14
```

OBR 56 PARAMETRIZACE V KROKYFB

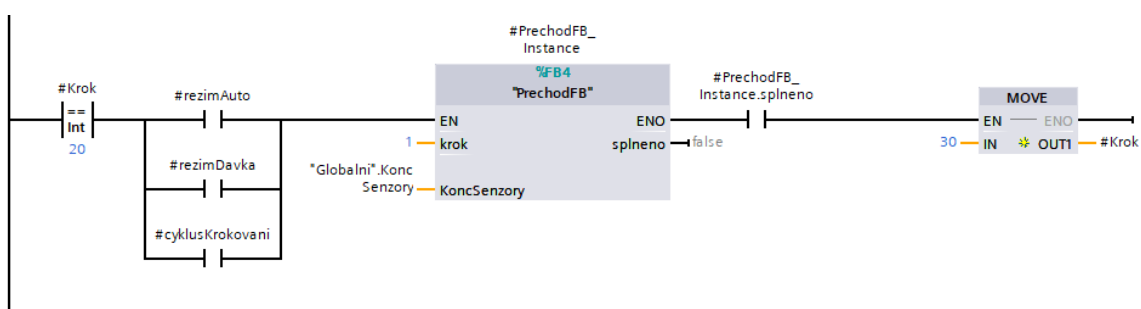
Funkční blok pracuje přímo s prvky na HMI obrazovce. Navolená sekvence je uložena do pole, které se poté používá pro kontrolu přechodů ve výrobním cyklu a aktivaci výstupů.

7.5.3 VÝROBNÍ CYKLUS

Hlavní Funkční blok, kterou slouží primárně ke krokování sekvence. Ke krokování byla použita metoda za pomoci nastavování hodnot v proměnné typu INT. To má výhodu ve snadné editaci a větvení sekvence. Dále je snadné sekvencér vypnout nastavením jedné proměnné, na rozdíl např. od mazacího se taktovacího řetězce.

Krokování sekvence se skládá ze tří částí, a to:

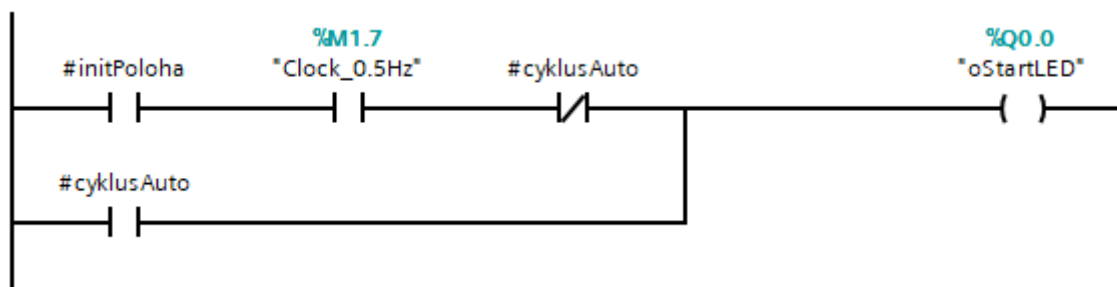
1. Kontrola aktivního kroku
2. Podmínka přechodu do dalšího kroku
3. Nastavení dalšího kroku



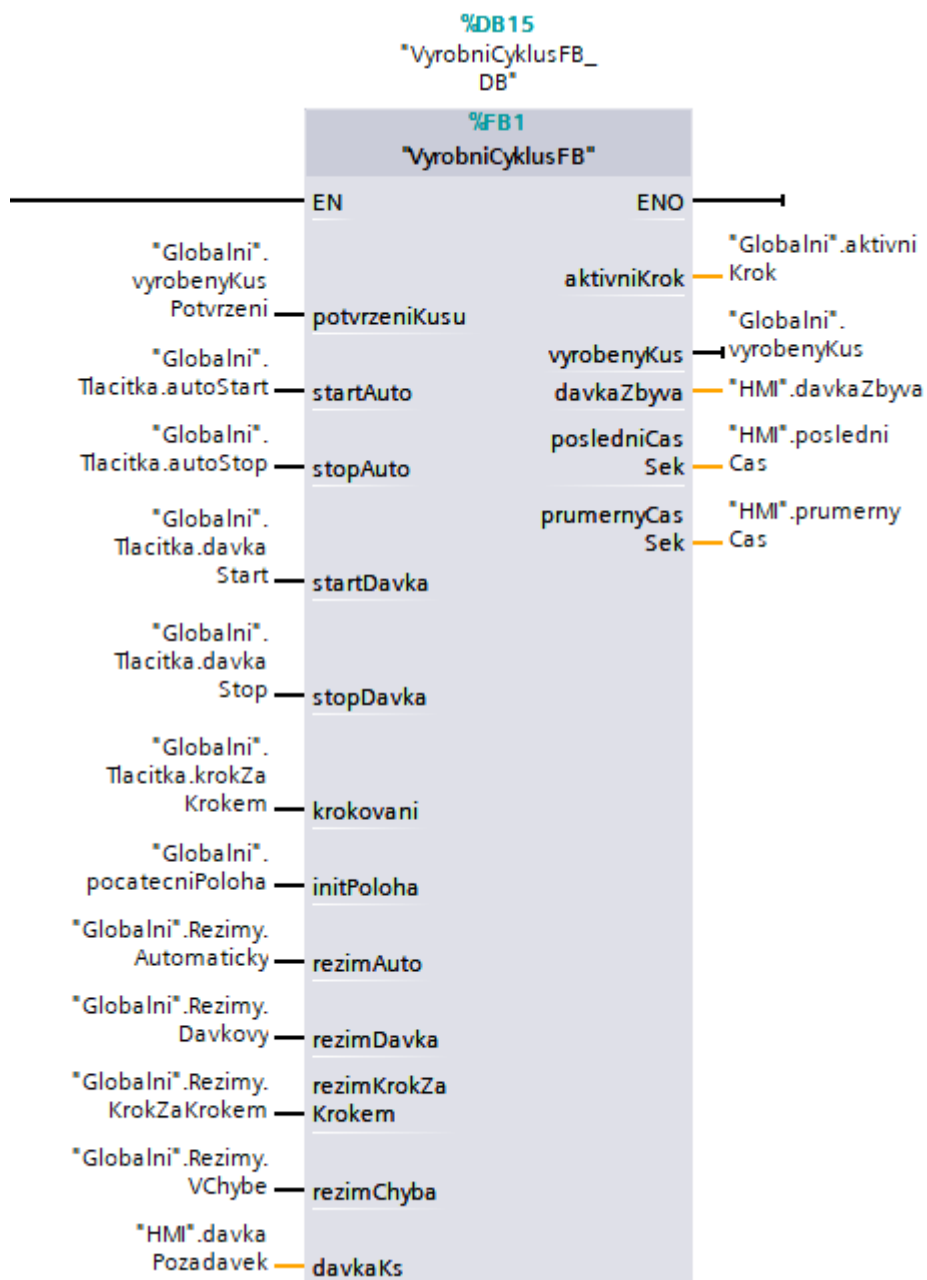
OBR 57 PŘÍKLAD JEDNOHO KROKU SEKVENCE

V rámci Funkčního bloku je řešena i logika blikání/svícení Start signálky. K tomu je použit systémový bit, který se dá nastavit v konfiguraci PLC. Svícení platí pouze pro

automatický cyklus. Ostatní režimy se nevyužívají tak často a mohlo by to plést operátora.



OBR 58 LOGIKA SIGNALIZAČNÍ LED

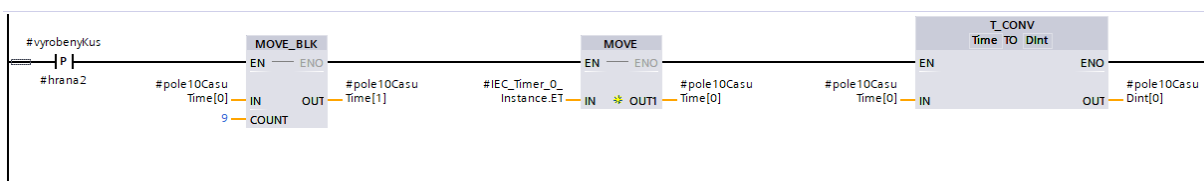


OBR 59 FUNKČNÍ BLOK VÝROBNÍ CYKLU

Dále je v rámci Funkčního bloku řešena logika sběru dat. K měření času je použit TON časovač, ze kterého je na konci výrobního cyklu zkopírovaná hodnota aktuálního

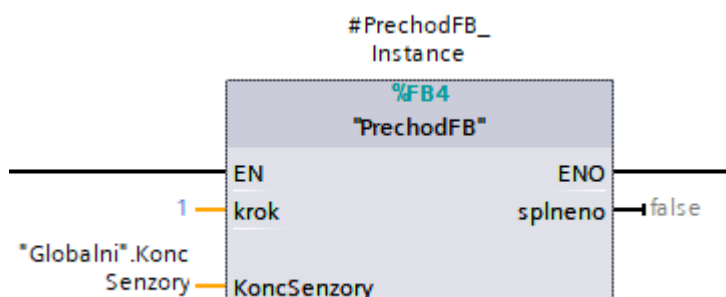
času. Tato hodnota je poté převedena na sekundy a poté uložena do proměnné typu INT.

Pro průměrný čas je uloženo 10 posledních času v poli, které se pak převádí z TIME typů na DINT a pomocí funkce CALCULATE je poté proveden průměr z těchto 10 cyklů.



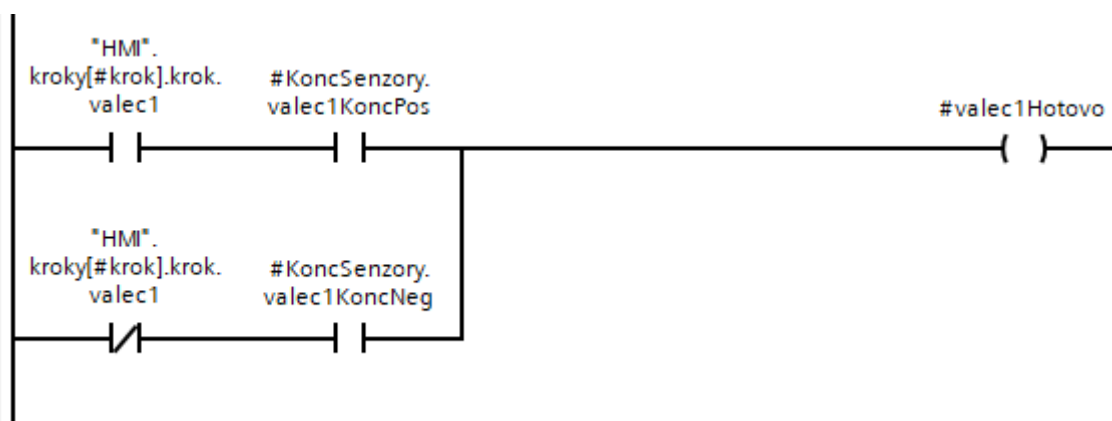
OBR 60 PRÁCE SE POLI VÝROBNÍCH ČASŮ A UKÁZKA KONVERZE DATOVÉHO TYPU

7.5.4 PŘECHOD V SEKVENCI



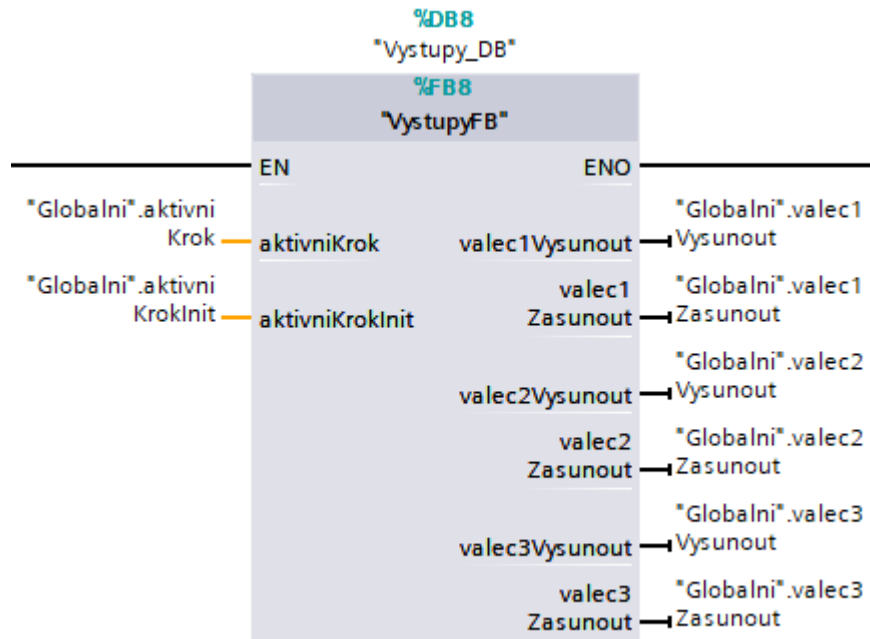
OBR 61 FUNKCNI BLOK PRECHODU

Tento funkční blok porovnává aktuální stav koncových senzorů s požadovaným stavem v konkrétním kroku. Vstupem je současně i číslo kroku, které poté indexuje pole kroků, které se vyhodnocuje. Každý krok výrobního cyklu má instanci tohoto Funkčního bloku. Součástí je časovač, který čeká nastavený čas po skončení pohybů válců.



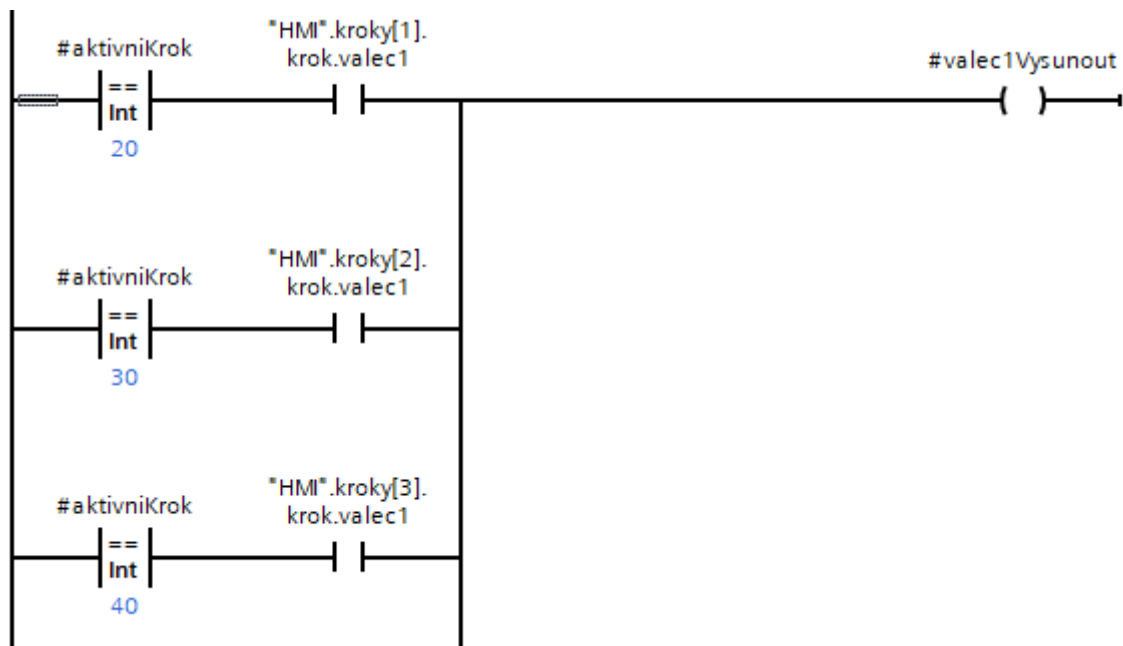
OBR 62 KONTROLA KONEČNÝCH SENZORŮ VE FUNKČNÍM BLOKU PRECHODFB

7.5.5 VÝSTUPY



OBR 63 FUNKČNÍ BLOK VYSTUPY

Funkční blok slouží jako rozhraní mezi Výrobním cyklem, Inicializací a Funkčními bloky řídicích válců. Při aktivním kroku nastavuje výstup podle nastavení.



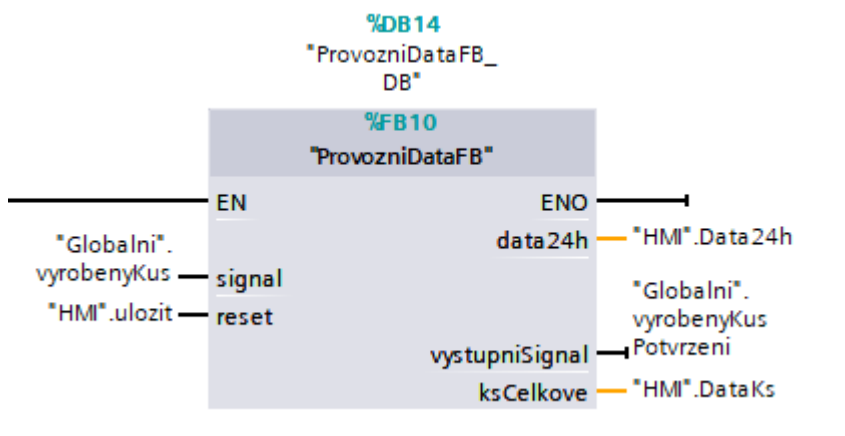
OBR 64 UKÁZKA ČÁSTI NASTAVOVÁNÍ VÝSTUPU PODLE PARAMETRIZACE

7.5.6 INICIALIZACE

Funkční blok, který vychází z Výrobního cyklu, ale je zjednodušen na jedno proběhnutí sekvence. Má vlastní variantu přechodu, která nemá časovač.

Funkční blok, který vyhodnocuje počáteční stav je založen na Funkčním bloku přechodu, ale nemá vstup s krokem, nýbrž fixní hodnotu. Kontroluje tedy páté parametry inicializace oproti konečným sensorům. V případě, že je aktuální poloha válců v souladu s nastavením posledního kroku inicializace, stroj v počáteční poloze.

7.6 Sběr dat



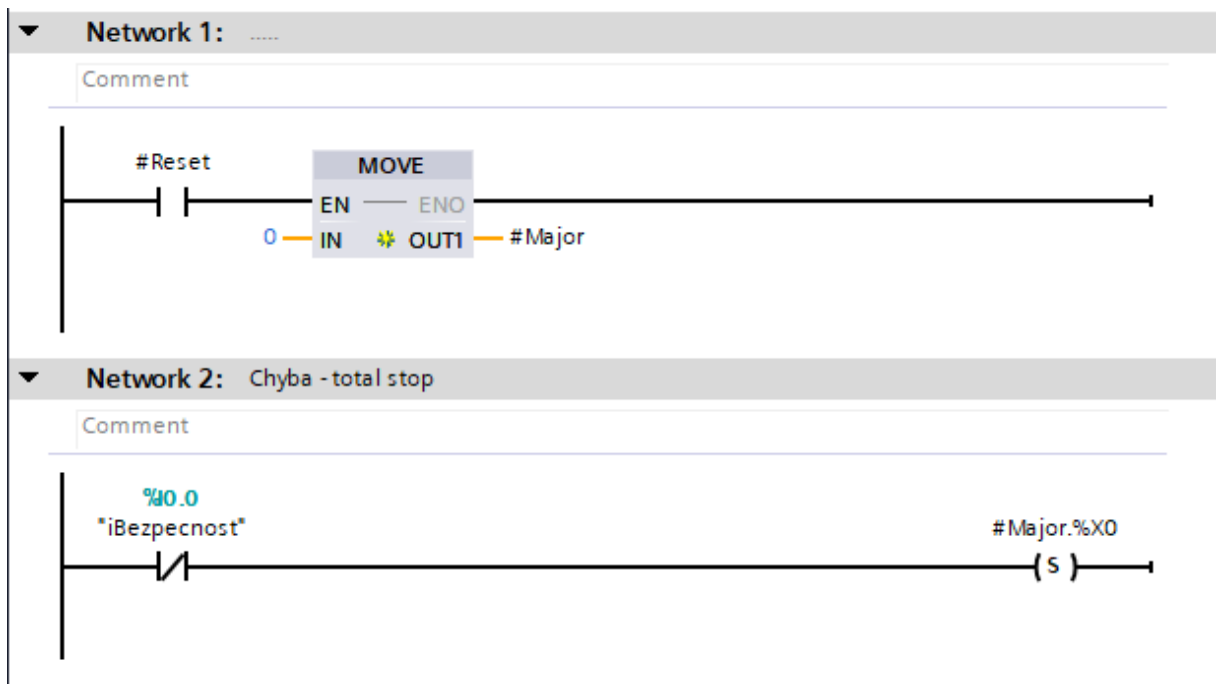
OBR 65 FUNKČNÍ BLOK PROVOZNÍ DATA

Pracuje s poli o 24 INT hodnotách, reprezentujících vyrobené kusy za hodinu. S náběžnou hranou signálu z výrobního cyklu se přičte 1 k aktuální hodnotě v dané hodině, která je brána ze systémového času, a přičte se též 1 k celkovému počtu vyrobených kusů. Současně se nastaví výstup k pokračování ve výrobním cyklu. Celkový počet se resetuje pomocí ukládacího tlačítka pro změnu výrobních parametrů a zároveň je ošetřeno přetečení tím, že při maximální hodnotě 32767 se hodnota vynuluje.

7.7 Alarmy

V rámci vyhodnocování chybových hlášek je potřeba zajistit dominanci aktivních alarmů nad resetovací instrukcí tak, aby jen nestačilo permanentně stisknout resetovací tlačítko.

Metod je více, ale zde byla použita metoda dominance polohou, kde resetovací instrukce se vyhodnocuje jako první tím, že se vynuluje celá proměnná typu WORD. Poté se vyhodnocují jednotlivé chybové stavy, u kterých se nastavuje konkrétní bit.



OBR 66 RESETOVÁNÍ A NATAVENÍ CHYBOVÉ HLÁŠKY

Číslo alarmu	Popis alarmu
1	Není zapnuto ovládací napětí/bezpečnost
2	Chybí tlak vzduchu
3	Válec 1 nedojel na koncový senzor – vysunutá poloha
4	Válec 1 nedojel na koncový senzor – zasunutá poloha
5	Válec 1 – porucha koncových senzorů
6	Válec 2 nedojel na koncový senzor – vysunutá poloha
7	Válec 2 nedojel na koncový senzor – zasunutá poloha
8	Válec 2 – porucha koncových senzorů
9	Válec 3 nedojel na koncový senzor – vysunutá poloha
10	Válec 3 nedojel na koncový senzor – zasunutá poloha
11	Válec 3 – porucha koncových senzorů

TABULKA 10 CHYBOVÉ HLÁŠKY

7.8 Odladování programu

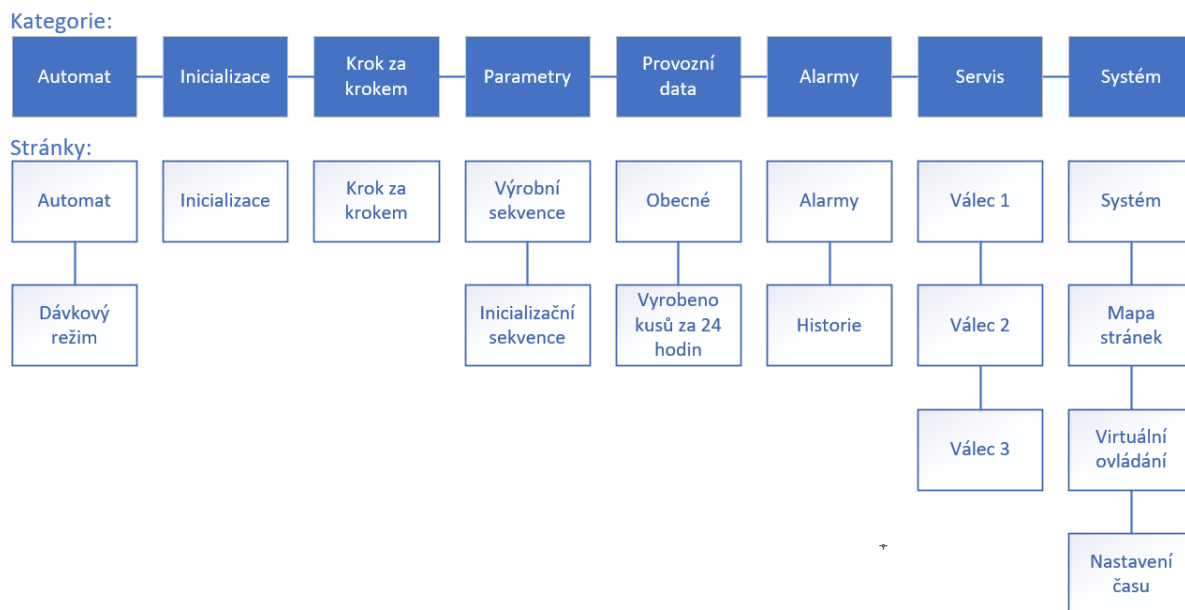
K odladění byl použit PLCSIM společně se simulací HMI pomocí WINCC Runtime Advanced.

Na odladění FB, který slouží k řízení válců, byla použita simulace sekvence vysunutí válce. Následně byla otestována i varianta, kde válec nedojede na koncový senzor v požadovaném čase, se spuštěním příslušné chyby.

Jelikož není možné mít zapnuto více sekvencí současně, výrobní cyklus byl odladován na dvou sekvencích, které nastavovali výstupy v souladu s nastavením. Poté byl otestován stav, kdy je během výrobního cyklu stisknut total stop.

8 Návrh vizualizace zobecněného jednoúčelového stroje

8.1 Mapa stránek



OBR 67 MAPA STRÁNEK

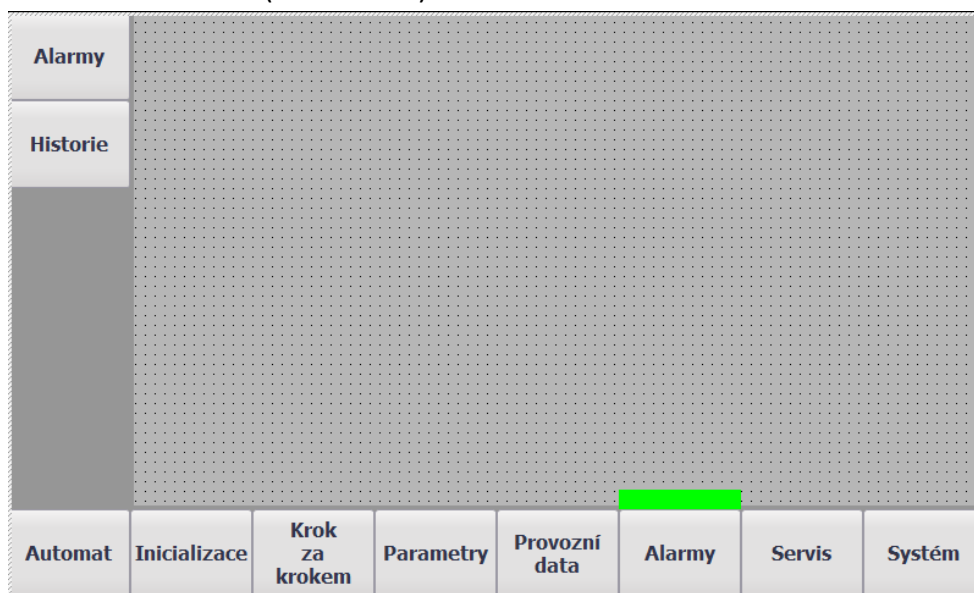
8.1.1 ČÍSLOVÁNÍ STRÁNEK

Kategorie jsou číslovány po desítkách a obrazovky po jednotkách. Např. obrazovka Virtuální ovládání má číslo 83. Tímto je zajištěna snadná orientace v seznamu obrazovek při případné úpravě.

8.1.2 NAVIGACE

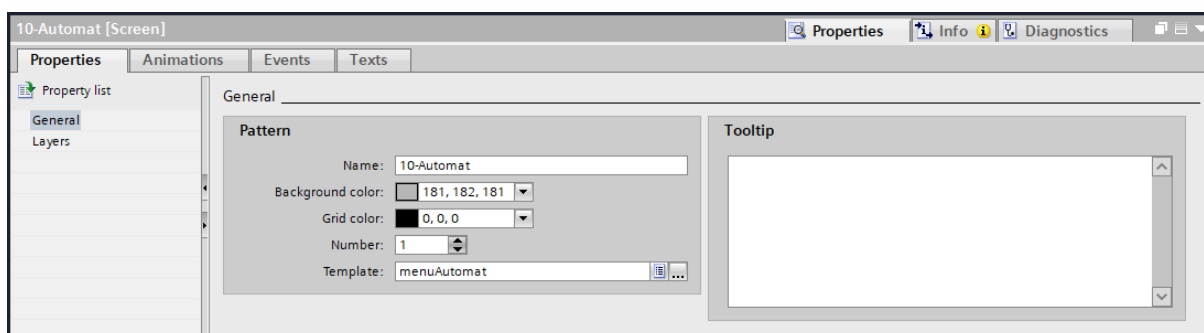
K navigaci slouží horizontální rozdělení kategorií a vertikální rozdělení obrazovek v kategorii. Každá kategorie má svoji šablonu (Template).

8.1.3 ŠABLONA STRÁNKY (TEMPLATE)



OBR 68 ŠABLONA ALARMY

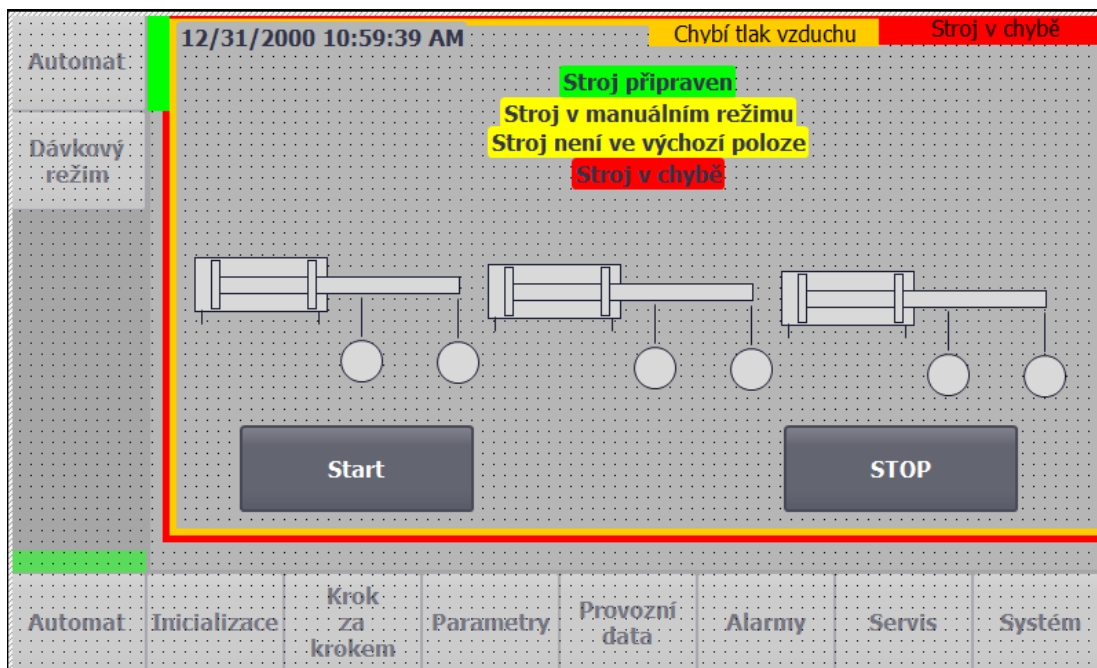
Navigační tlačítka se obsahují na všech stránkách, proto byla využita šablona pro navigaci. Tyto šablony jsou vytvořeny v rámci projektu. Poté se musí u specifické stránky v parametrech zvolit, jakou šablonu má obsahovat. V dané stránce jsou pak prvky šablony součástí pozadí a není možné je upravovat.



OBR 69 NASTAVENÍ ŠABLONY VE VLASTNOSTECH STRÁNKY

8.2 Ovládání výroby

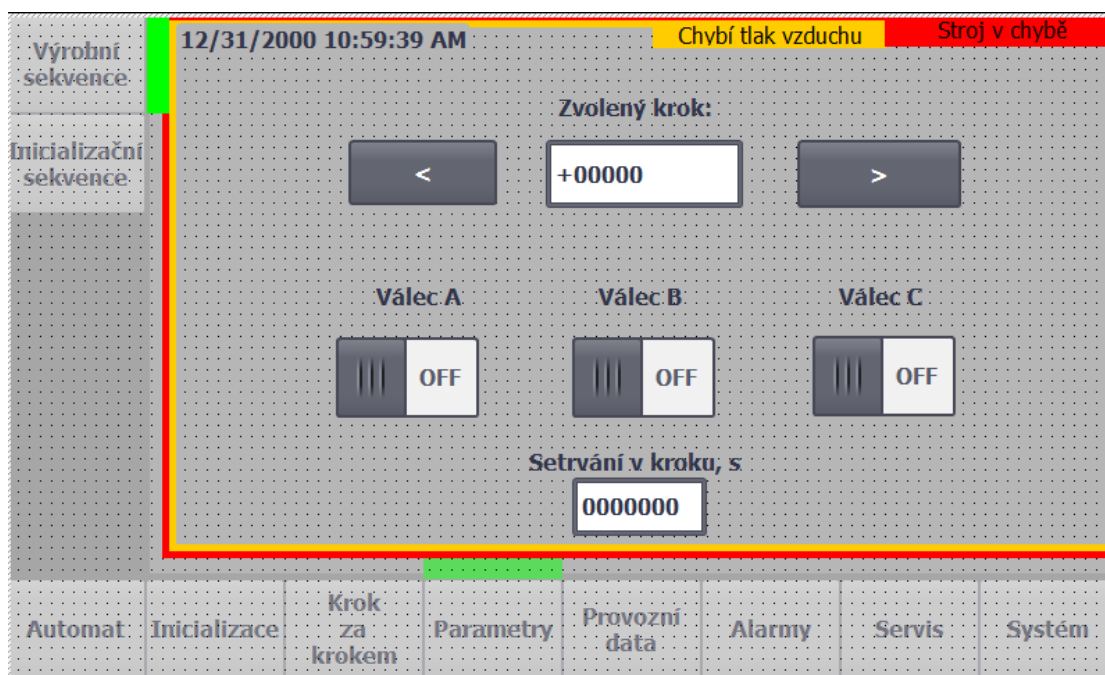
K základnímu řízení stroje slouží obrazovky Automat a Inicializace. Tyto stránky jsou tvořeny pouze Start a Stop tlačítkem a vizualizací stavu válců.



OBR 70 HMI OBRAZOVKA AUTOMATICKÝ REŽIM

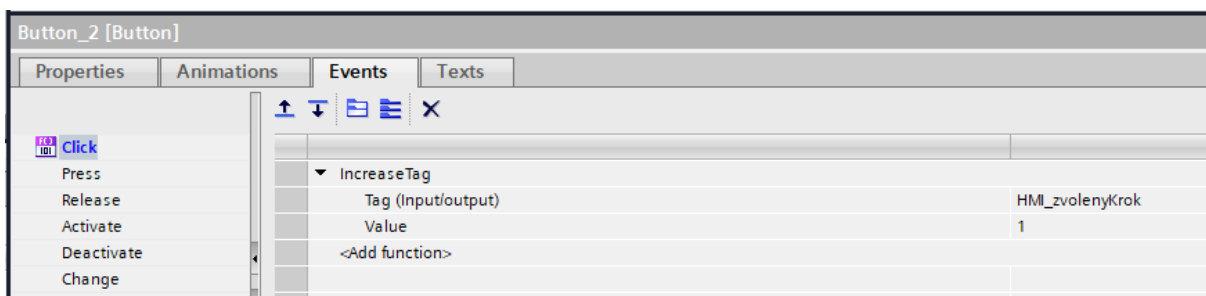
8.3 Parametry cyklu

K nastavení parametrů výroby slouží dvě obrazovky v kategorii Parametry. Jedna slouží k nastavení výrobní sekvence a druhá k nastavení inicializace. Rozdíl v nich je pouze v setrvání v kroku u výrobní sekvence.



OBR 71 HMI OBRAZOVKA PARAMETRY - VÝROBNÍ SEKVENCE

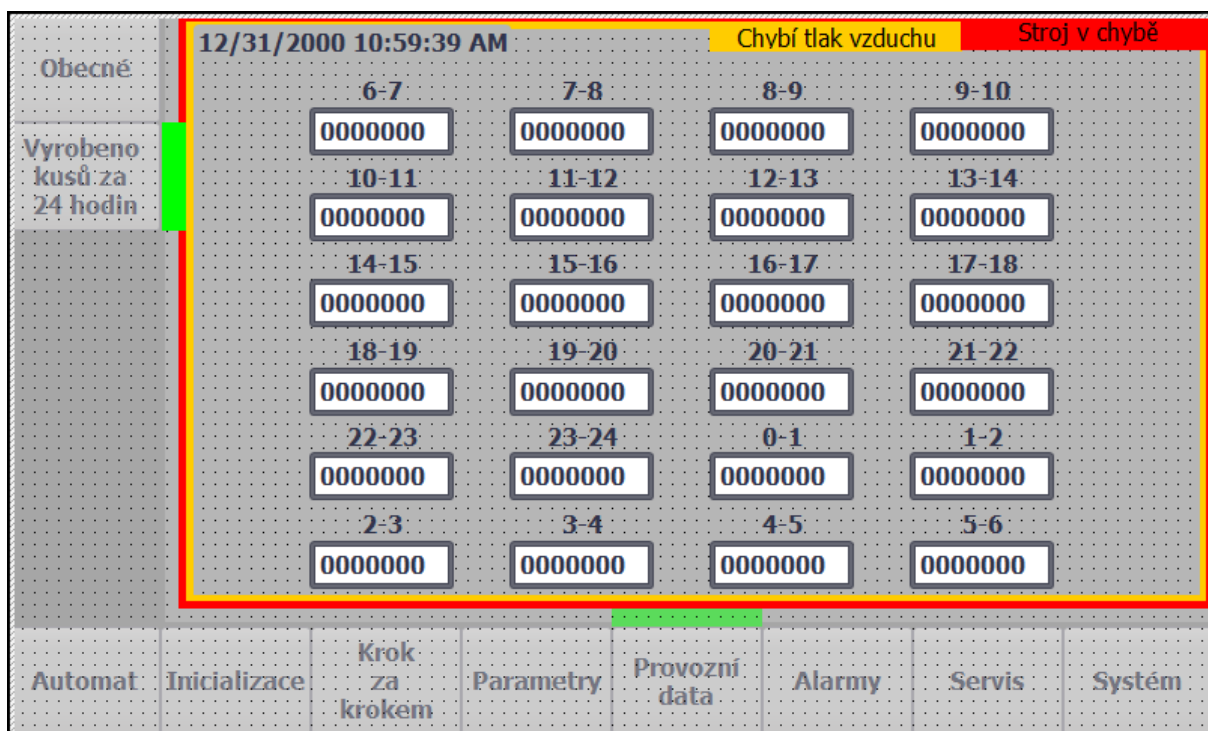
Uživatel volí krok buď pomocí textového vstupu nebo pomocí tlačítek. Tlačítka mají definovanou funkci snižování a zvyšování proměnné. V PLC bylo poté potřeba ošetřit krajní hodnoty zvoleného kroku.



OBR 72 FUNKCE TLAČÍTKA ZVYŠOVÁNÍ PROMĚNNÉ O 1

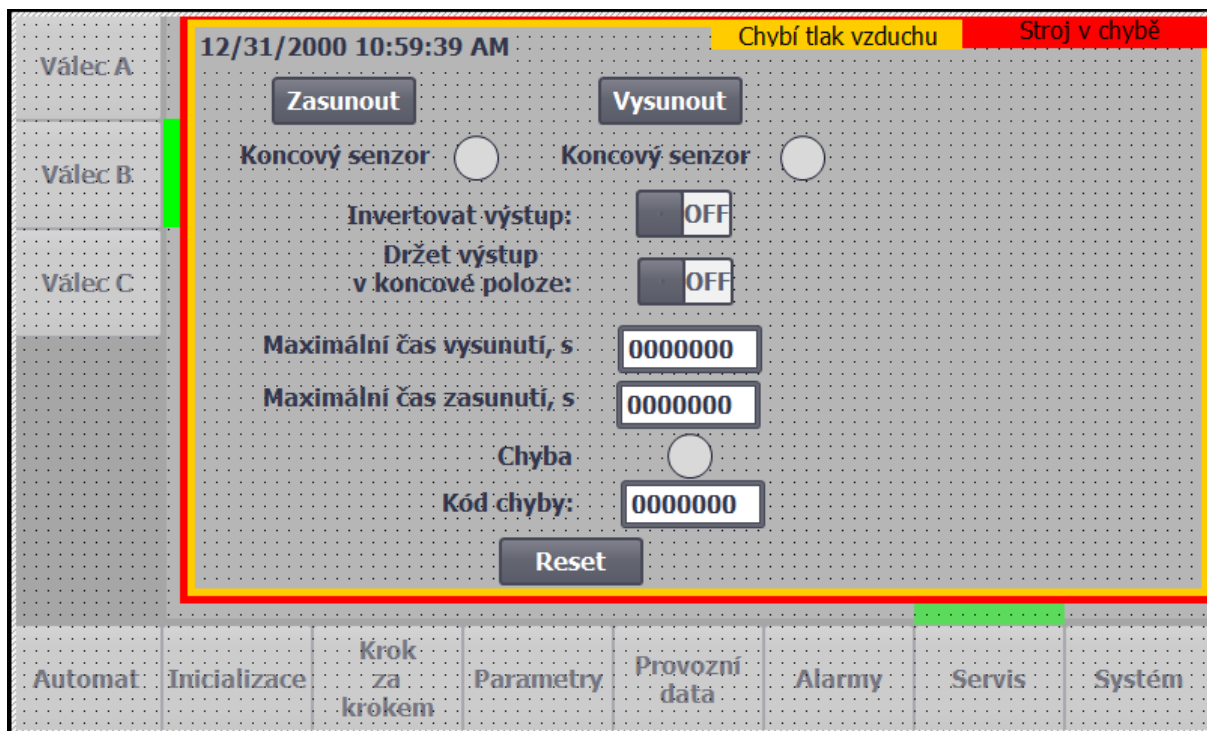
8.4 Provozní data

V kategorii Provozní data jsou zobrazeny počty kusů vyrobených za hodinu v posledních 24 hodinách. Na stránce Obecné je pak zobrazen průměrný čas cyklu, čas posledního cyklu a počet vyrobených kusů od poslední změny sekvence.



OBR 73 HMI OBRAZOVKA VYROBENO KUSŮ ZA 24 HODIN

8.5 Servis

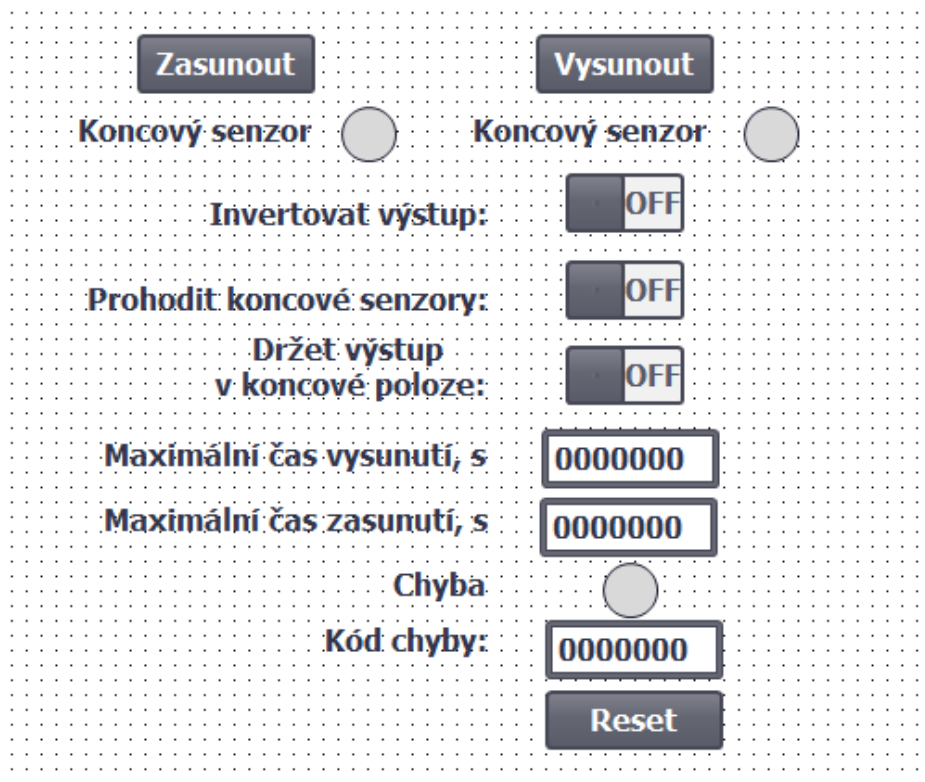


OBR 74 HMI OBRAZOVKA SERVIS

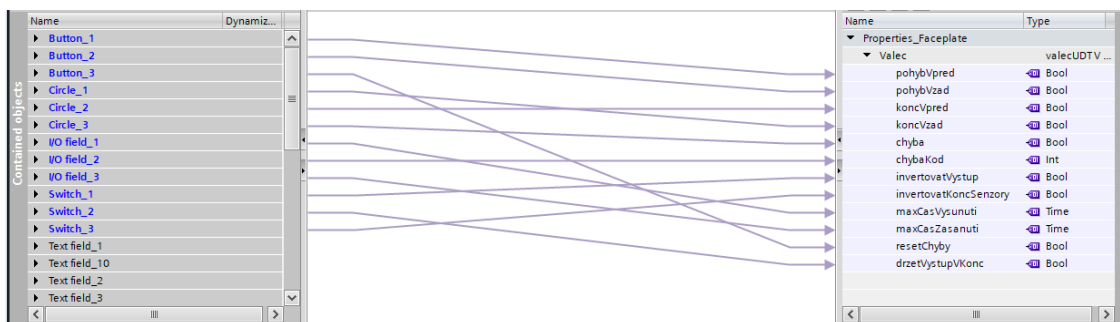
Servisní obrazovka slouží k manuálnímu ovládání válců a k nastavování jejich parametrů. Přístup do stránky je možný pouze po přihlášení servisního uživatele. Pokud není uživatel přihlášen, stisknutím navigačního tlačítka Servis se vyvolá přihlašovací formulář.

8.5.1 ŠABLONA PRVKŮ (FACEPLATE)

V rámci servisu byla použita šablona prvků pro řízení válců, jelikož jsou to prvky, které jsou pro všechny tři válce shodné. Tato šablona je uložena v rámci knihovny a je možné jí v rámci projektu použít kolikrát je potřeba. Šablony je možné verzovat a správa šablon umožňuje aktualizaci všech instancí najednou. Každá instance je napárována na příslušný válec v programu. Šablona se vkládá do již existující obrazovky jako seskupené prvky a je možné ji pozicovat, jak je potřeba.

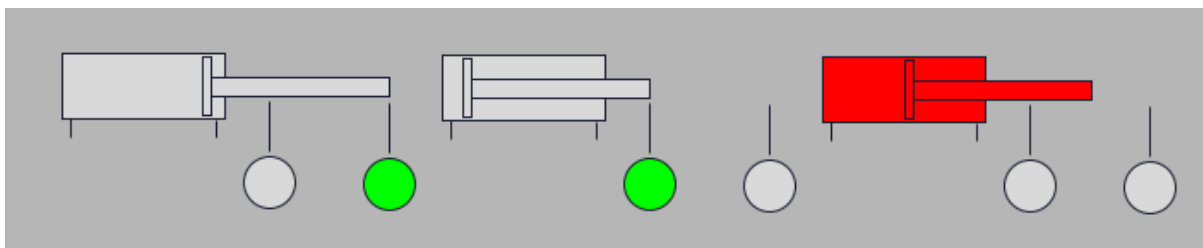


OBR 75 ROZLOŽENÍ PRVKŮ PŘI POUŽITÍ FACEPLATE



OBR 76 NAPÁROVÁNÍ PRVKŮ V RÁMCI FACEPLATE

K navázání šablony k proměnné byla do knihovny vložen uživatelský datový typ (UDT), se kterým pracuje funkční blok v PLC. Stejným způsobem byla vytvořena i vizualizace stavů pneumatických válců.



OBR 77 VIZUALIZACE POHYBU VÁLCE A ZOBRAZENÍ CHYBY

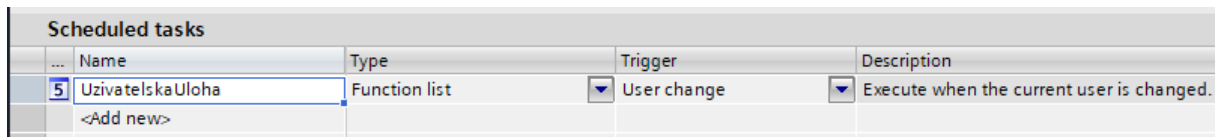
8.6 Alarmy

K zobrazení alarmů slouží dvě obrazovky. První má vyfiltrované pouze aktivní alarmy. Druhá zobrazuje historii.

8.7 Systém

8.7.1 ZOBRAZENÍ PŘIHLÁŠENÉHO UŽIVATELE

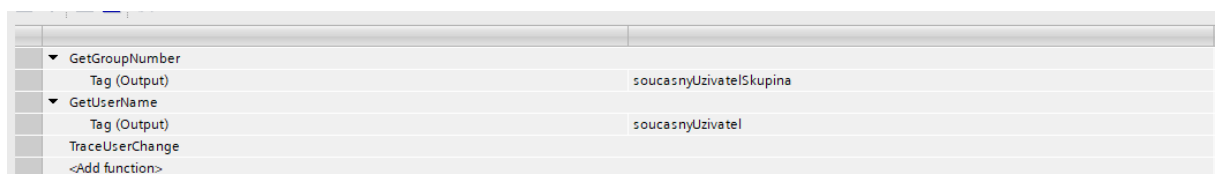
WINCC neumožňuje zobrazit aktuálně přihlášeného uživatele. K tomu je potřeba vytvořit úlohu, která to bude vykonávat.



Scheduled tasks				
...	Name	Type	Trigger	Description
5	UzivatelaskaUloha	Function list	User change	Execute when the current user is changed.
	<Add new>			

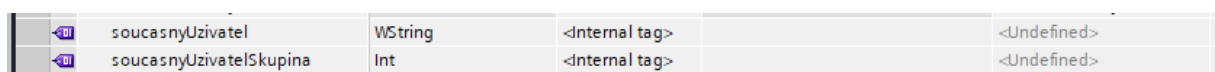
OBR 78 VYTVOŘENÍ UŽIVATELSKÉ ÚLOHY

V rámci úlohy se poté definuje sledování změny uživatele



GetGroupName	Tag (Output)	soucasnyUzivatelSkupina
GetUserName	Tag (Output)	soucasnyUzivatel
TraceUserChange		
<Add function>		

OBR 79 NASTAVENÍ ÚLOHY K UKLÁDÁNÍ AKTIVNÍHO UŽIVATELE



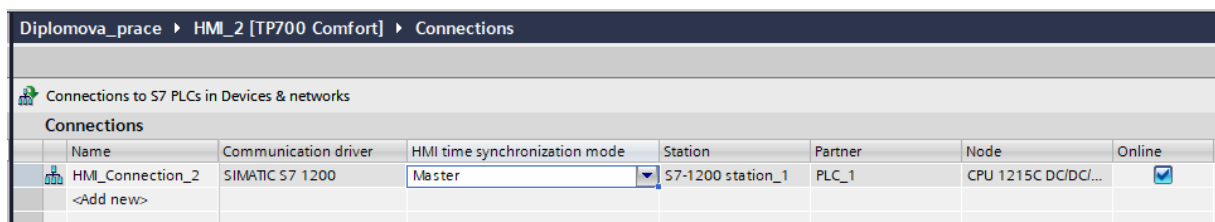
soucasnyUzivatel	WString	<Internal tag>	<Undefined>
soucasnyUzivatelSkupina	Int	<Internal tag>	<Undefined>

OBR 80 PROMĚNNÉ AKTIVNÍHO UŽIVATELE

K tomu slouží proměnná typu WString pro název uživatele a Int pro kód třídy uživatele. Pomocí I/O bloku je poté možno tyto proměnné zobrazit na obrazovce.

8.7.2 PROVÁZÁNÍ ČASU PLC A HMI

V rámci TIA Portalu je možné v nastavení spojení definovat metodu synchronizace času a není tedy potřeba nadále editovat čas v PLC voláním funkce z HMI. V nastavení spojení je možné nastavit, že HMI je „master“ a poté stačí upravovat čas pouze v HMI. K tomu lze použít pole pro zobrazení času, které se dají rozdělit na datum a čas.



Connections to S7 PLCs in Devices & networks							
Connections							
Name	Communication driver	HMI time synchronization mode	Station	Partner	Node	Online	
HMI_Connection_2	SIMATIC S7 1200	Master	S7-1200 station_1	PLC_1	CPU 1215C DC/DCI...	<input checked="" type="checkbox"/>	
<Add new>							

OBR 81 PROVÁZÁNÍ ČASU MEZI PLC A HMI

9 Závěr

V prvních třech kapitolách práce byla věnována pozornost portfoliu SIMATIC PLC, prostředí TIA Portal a směrnicím PLCopen. V rámci směrnic byly vypsány nejdůležitější rady a pravidla. Ve 4. kapitole se práce věnuje programování v TIA Portalu. Jednotlivým jazykům, které TIA Portal nabízí a vhodnosti jejich použití je věnována kapitola 5. V 6. kapitole jsou popsány možnosti odladování online a offline a na konci byla stanovena metodika odladování a tím byl kompletně splněn první bod zadání.

Praktická část začíná kapitolou 8. V této kapitole jsou stanoveny předpoklady virtuálního stroje, definovány jeho provozní režimy. Následně je popsáno řešení řídicího programu. V rámci výrobního cyklu bylo zajištěno parametrizování výrobní sekvence a vkládání časové prodlevy mezi kroky. Program byl odladen pomocí PLCSIM. Touto kapitolou je splněn druhý bod zadání.

Kapitola 9 je věnována poslednímu úkolu, a to návrhu vizualizace. Vizualizace byla vytvořena jako operátorský HMI panel. Byly využity možnosti šablon ve WinCC. V rámci ovládání stroje byly vizualizovány pohyby válců, byla implementována parametrizace výrobního cyklu a inicializace. V rámci chybových hlášení jsou řešeny aktuální i historie. Provozní data jsou zobrazena jako vyrobeno kusů hodinově za poslední den, vyrobeno od poslední změny cyklu, poslední čas cyklu a průměrný čas cyklu, počítaný z posledních deseti cyklů. HMI panel byl simulován v prostředí WinCC Runtime Advanced a byl odladen při propojení s PLCSIM. Tím byl splněn poslední bod zadání.

ZDROJE

- [1] „ŘÍDICÍ SYSTÉMY SIMATIC", *SIEMENS ČESKÁ REPUBLIKA*.
[HTTPS://NEW.SIEMENS.COM/CZ/CS/PRODUCTS/AUTOMATION/SYSTEMS/INDUSTRIAL/PLC.HTML](https://new.siemens.com/cz/cs/products/automation/systems/industrial/plc.html) (VIDĚNO ZÁŘ. 24, 2021).
- [2] „SIMATIC-S7-1200-CPU.PNG (1125×632)", ZÁŘ. 24, 2021.
[HTTPS://ASSETS.NEW.SIEMENS.COM/SIEMENS/ASSETS/API/UUID:AC83E37763FDDDE6CE537648C578934955600791/WIDTH:1125/QUALITY:HIGH/SIMATIC-S7-1200-CPU.PNG](https://assets.new.siemens.com/siemens/assets/api/uuid:ac83e37763fddde6ce537648c578934955600791/width:1125/quality:high/simatic-s7-1200-cpu.png) (VIDĚNO ZÁŘ. 24, 2021).
- [3] H. BERGER, *AUTOMATING WITH SIMATIC S7-1200: CONFIGURING, PROGRAMMING AND TESTING WITH STEP 7 BASIC VISUALIZATION WITH HMI BASIC*, 2., ENL.REV. ED. ERLANGEN: PUBLICIS PUBL, 2013.
- [4] „SIMATIC-ST70-COMPLETE-ENGLISH-2021.PDF - CATALOG ST 70: PRODUCTS FOR TOTALLY INTE... - ID: 109744167 - INDUSTRY SUPPORT SIEMENS".
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/109744167/CATALOG-ST-70-PRODUCTS-FOR-TOTALLY-INTEGRATED-AUTOMATION-SIMATIC?DTI=0&PNID=13685&LC=EN-CZ](https://support.industry.siemens.com/cs/document/109744167/catalog-st-70-products-for-totally-integrated-automation-simatic?dti=0&pnid=13685&lc=en-cz) (VIDĚNO LED. 01, 2022).
- [5] „KOMPAKTNI-PROCESORY-SIMATIC-S7-1500.JPG (1125×632)", ZÁŘ. 24, 2021.
[HTTPS://ASSETS.NEW.SIEMENS.COM/SIEMENS/ASSETS/API/UUID:D81A0739-971C-4159-9171-45BAE27C874E/WIDTH:1125/QUALITY:HIGH/KOMPAKTNI-PROCESORY-SIMATIC-S7-1500.JPG](https://assets.new.siemens.com/siemens/assets/api/uuid:d81a0739-971c-4159-9171-45bae27c874e/width:1125/quality:high/kompaktni-procesory-simatic-s7-1500.jpg) (VIDĚNO ZÁŘ. 24, 2021).
- [6] H. BERGER, *AUTOMATING WITH SIMATIC S7-1500: CONFIGURING, PROGRAMMING AND TESTING WITH STEP 7 PROFESSIONAL*, 2., ENL.REV. ED. ERLANGEN: PUBLICIS PUBL, 2017.
- [7] „SIMATIC-ET-200SP-FAIL-SAFE-WHITE.PNG (1125×632)", ZÁŘ. 24, 2021.
[HTTPS://ASSETS.NEW.SIEMENS.COM/SIEMENS/ASSETS/API/UUID:3043C878-1619-4EAA-B09C-64A00B667DF2/WIDTH:1125/QUALITY:HIGH/SIMATIC-ET-200SP-FAIL-SAFE-WHITE.PNG](https://assets.new.siemens.com/siemens/assets/api/uuid:3043c878-1619-4eaa-b09c-64a00b667df2/width:1125/quality:high/simatic-et-200sp-fail-safe-white.png) (VIDĚNO ZÁŘ. 24, 2021).
- [8] „SIMATIC ET 200", *SIEMENS.COM GLOBAL WEBSITE*.
[HTTPS://NEW.SIEMENS.COM/GLOBAL/EN/PRODUCTS/AUTOMATION/SYSTEMS/INDUSTRIAL/IO-SYSTEMS.HTML](https://new.siemens.com/global/en/products/automation/systems/industrial/io-systems.html) (VIDĚNO LED. 02, 2022).
- [9] „SIMATIC-S7-400.JPG (600×400)".
[HTTPS://ASSETS.NEW.SIEMENS.COM/SIEMENS/ASSETS/API/UUID:A68EDO0A0267DEF8EF60D303BB27DEBD39F09FEF/WIDTH:1125/QUALITY:HIGH/SIMATIC-S7-400.JPG](https://assets.new.siemens.com/siemens/assets/api/uuid:a68edo0a0267def8ef60d303bb27debd39f09fef/width:1125/quality:high/simatic-s7-400.jpg) (VIDĚNO ZÁŘ. 24, 2021).
- [10] „TIA PORTAL", *SIEMENS ČESKÁ REPUBLIKA*.
[HTTPS://NEW.SIEMENS.COM/CZ/CS/PRODUCTS/AUTOMATION/INDUSTRY-SOFTWARE/AUTOMATIZACNI-SOFTWARE/TIA-PORTAL.HTML](https://new.siemens.com/cz/cs/products/automation/industry-software/automatizacni-software/tia-portal.html) (VIDĚNO LED. 10, 2022).
- [11] „TIA_PORTAL_BERBLICK_PRODUKTE_E.PNG (1920×1080)".
[HTTPS://CACHE.INDUSTRY.SIEMENS.COM/DL/FILES/626/109771626/IMG_266269/V1/TIA_PORTAL_BERBLICK_PRODUKTE_E.PNG](https://cache.industry.siemens.com/dl/files/626/109771626/img_266269/v1/tia_portal_berblick_produkte_e.png) (VIDĚNO LED. 01, 2022).
- [12] „SIMATIC STEP 7 SAFETY", *SIEMENS.COM GLOBAL WEBSITE*.
[HTTPS://NEW.SIEMENS.COM/GLOBAL/EN/PRODUCTS/AUTOMATION/TOPIC-AREAS/SAFETY-INTEGRATED/FACTORY-AUTOMATION/OFFERING/SIMATIC-SAFETY/SIMATIC-STEP7-SAFETY.HTML](https://new.siemens.com/global/en/products/automation/topic-areas/safety-integrated/factory-automation/offering/simatic-safety/simatic-step7-safety.html) (VIDĚNO LED. 02, 2022).
- [13] „WINCC MIGRATION GUIDE (TIA PORTAL)", S. 31, 2019.
- [14] „VISUALIZE FUTURE", *SIEMENS.COM GLOBAL WEBSITE*.
[HTTPS://NEW.SIEMENS.COM/GLOBAL/EN/PRODUCTS/AUTOMATION/SIMATIC-HMI/WINCC-UNIFIED.HTML](https://new.siemens.com/global/en/products/automation/simatic-hmi/wincc-unified.html) (VIDĚNO LED. 03, 2022).
- [15] „SINAMICS STARTDRIVE COMMISSIONING SOFTWARE", *SIEMENS.COM GLOBAL WEBSITE*.
[HTTPS://NEW.SIEMENS.COM/GLOBAL/EN/PRODUCTS/DRIVES/SELECTION-AND-ENGINEERING-TOOLS/SINAMICS-STARTDRIVE-COMMISSIONING-SOFTWARE.HTML](https://new.siemens.com/global/en/products/drives/selection-and-engineering-tools/sinamics-startdrive-commissioning-software.html) (VIDĚNO LED. 03, 2022).

- [16] „SIMOTION SCOUT SOFTWARE PACKAGE”.
[HTTPS://MALL.INDUSTRY.SIEMENS.COM/MALL/CS/CZ/CATALOG/PRODUCTS/10027665](https://mall.industry.siemens.com/mall/cs/cz/catalog/products/10027665) (VIDĚNO LED. 03, 2022).
- [17] „3SK SAFETY RELAYS”, *SIEMENS.COM GLOBAL WEBSITE*.
[HTTPS://NEW.SIEMENS.COM/GLOBAL/EN/PRODUCTS/AUTOMATION/INDUSTRIAL-CONTROLS/SIRIUS/SIRIUS-MONITOR/3SK-SAFETY-RELAYS.HTML](https://new.siemens.com/global/en/products/automation/industrial-controls/sirius/sirius-monitor/3sk-safety-relays.html) (VIDĚNO LED. 10, 2022).
- [18] „FESTO_VTEM_DEVCON_TIA_VX.X.35”. VIDĚNO: LED. 01, 2022. [ONLINE].
 DOSTUPNÉ Z: [HTTPS://WWW.FESTO.COM/NET/EN-IL_IL/SUPPORTPORTAL/DEFAULT.ASPX?CAT=5675&TAB=4&S=T](https://www.festo.com/net/en-il_il/supportportal/default.aspx?cat=5675&tab=4&s=t)
- [19] „CYCLE AND RESPONSE TIMES.PDF”. VIDĚNO: PRO. 23, 2021. [ONLINE].
 DOSTUPNÉ Z:
[HTTPS://CACHE.INDUSTRY.SIEMENS.COM/DL/FILES/558/59193558/ATT_112303/V1/S71500_CYCLE_AND_REACTION_TIMES_FUNCTION_MANUAL_EN-US_EN-US.PDF](https://cache.industry.siemens.com/dl/files/558/59193558/att_112303/v1/s71500_cycle_and_reaction_times_function_manual_en-us_en-us.pdf)
- [20] „81318674_PROGRAMMING_GUIDELINE_DOC_V16_EN.PDF - PROGRAMMING GUIDELINE FOR S7-1200/... - ID: 90885040 - INDUSTRY SUPPORT SIEMENS”.
 VIDĚNO: ZÁŘ. 24, 2021. [ONLINE]. DOSTUPNÉ Z:
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/90885040/PROGRAMMING-GUIDELINE-FOR-S7-1200-S7-1500?DTI=0&LC=EN-WW](https://support.industry.siemens.com/cs/document/90885040/programming-guideline-for-s7-1200-s7-1500?dti=0&lc=en-ww)
- [21] „WHICH ORGANIZATION BLOCKS CAN YOU USE IN STEP 7 (TIA PORTAL)? - ID: 40654862 - INDUSTRY SUPPORT SIEMENS”.
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/40654862/WHICH-ORGANIZATION-BLOCKS-CAN-YOU-USE-IN-STEP-7-\(TIA-PORTAL\)-?DTI=0&LC=EN-DZ](https://support.industry.siemens.com/cs/document/40654862/which-organization-blocks-can-you-use-in-step-7-(tia-portal)-?dti=0&lc=en-dz) (VIDĚNO LED. 03, 2022).
- [22] „81318674_PROGRAMMING_GUIDELINE_DOC_V16_EN.PDF - PROGRAMMING GUIDELINE FOR S7-1200/... - ID: 90885040 - INDUSTRY SUPPORT SIEMENS”,
 ZÁŘ. 24, 2021.
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/90885040/PROGRAMMING-GUIDELINE-FOR-S7-1200-S7-1500?DTI=0&LC=EN-WW](https://support.industry.siemens.com/cs/document/90885040/programming-guideline-for-s7-1200-s7-1500?dti=0&lc=en-ww) (VIDĚNO ZÁŘ. 24, 2021).
- [23] „S71200_SYSTEM_MANUAL_EN-US_EN-US.PDF - SIMATIC S7 S7-1200 PROGRAMMABLE CONTROLLER - ID: 91696622 - INDUSTRY SUPPORT SIEMENS”.
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/91696622/SIMATIC-S7-S7-1200-PROGRAMMABLE-CONTROLLER?DTI=0&LC=EN-CZ](https://support.industry.siemens.com/cs/document/91696622/simatic-s7-s7-1200-programmable-controller?dti=0&lc=en-cz) (VIDĚNO LED. 01, 2022).
- [24] „IN STEP 7 (TIA PORTAL), WHY DO YOU HAVE TO INSERT A SET INSTRUCTION AFTER A JUMP C... - ID: 67519310 - INDUSTRY SUPPORT SIEMENS”.
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/67519310/IN-STEP-7-\(TIA-PORTAL\)-WHY-DO-YOU-HAVE-TO-INSERT-A-SET-INSTRUCTION-AFTER-A-JUMP-COMMAND-IN-ORDER-TO-ENSURE-A-FIRST-SCAN-?DTI=0&LC=EN-AU](https://support.industry.siemens.com/cs/document/67519310/in-step-7-(tia-portal)-why-do-you-have-to-insert-a-set-instruction-after-a-jump-command-in-order-to-ensure-a-first-scan-?dti=0&lc=en-au) (VIDĚNO LED. 10, 2022).
- [25] „AUTOMATION OF SEQUENTIAL PROCESSES WITH GRAPH IN THE TIA PORTAL FOR S7-1500 - ID: 109759822 - INDUSTRY SUPPORT SIEMENS”.
[HTTPS://SUPPORT.INDUSTRY.SIEMENS.COM/CS/DOCUMENT/109759822/AUTOMATION-OF-SEQUENTIAL-PROCESSES-WITH-GRAPH-IN-THE-TIA-PORTAL-FOR-S7-1500?DTI=0&LC=EN-DE](https://support.industry.siemens.com/cs/document/109759822/automation-of-sequential-processes-with-graph-in-the-tia-portal-for-s7-1500?dti=0&lc=en-de) (VIDĚNO LED. 01, 2022).
- [26] „V17-LAUNCH-WEBINAR-TIA-PORTAL-V17-TECHNICAL-HIGHLIGHTS.PDF”.
 VIDĚNO: LED. 09, 2022. [ONLINE]. DOSTUPNÉ Z:
[HTTPS://ASSETS.NEW.SIEMENS.COM/SIEMENS/ASSETS/API/UUID:1DBDBF0E-EB1C-429B-B043-2E78E818D8B2/V17-LAUNCH-WEBINAR-TIA-PORTAL-V17-TECHNICAL-HIGHLIGHTS.PDF](https://assets.new.siemens.com/siemens/assets/api/uuid:1dbdbf0e-eb1c-429b-b043-2e78e818d8b2/v17-launch-webinar-tia-portal-v17-technical-highlights.pdf)
- [27] „GET TO GRIPS WITH SIEMENS TIA PORTAL V17”, [HTTPS://WWW.UED.CO.UK](https://www.ued.co.uk),
 ČER. 11, 2021. [HTTPS://WWW.UED.CO.UK/BLOG/GET-TO-GRIPS-WITH-SIEMENS-TIA-PORTAL-V17-275182/](https://www.ued.co.uk/blog/get-to-grips-with-siemens-tia-portal-v17-275182/) (VIDĚNO LED. 10, 2022).
- [28] „SIMATIC S7-PLCSIM ADVANCED | ŘÍDICÍ SYSTÉMY SIMATIC PLCS | SIEMENS CZECH REPUBLIC”, *SIEMENS ČESKÁ REPUBLIKA*.
[HTTPS://NEW.SIEMENS.COM/CZ/CS/PRODUCTS/AUTOMATION/SYSTEMS/INDUSTRIAL/PLC/SIMATIC-S7-PLCSIM.HTML](https://new.siemens.com/cz/cs/products/automation/systems/industrial/plc/simatic-s7-plcsim.html) (VIDĚNO LED. 10, 2022).

SEZNAM OBRÁZKŮ

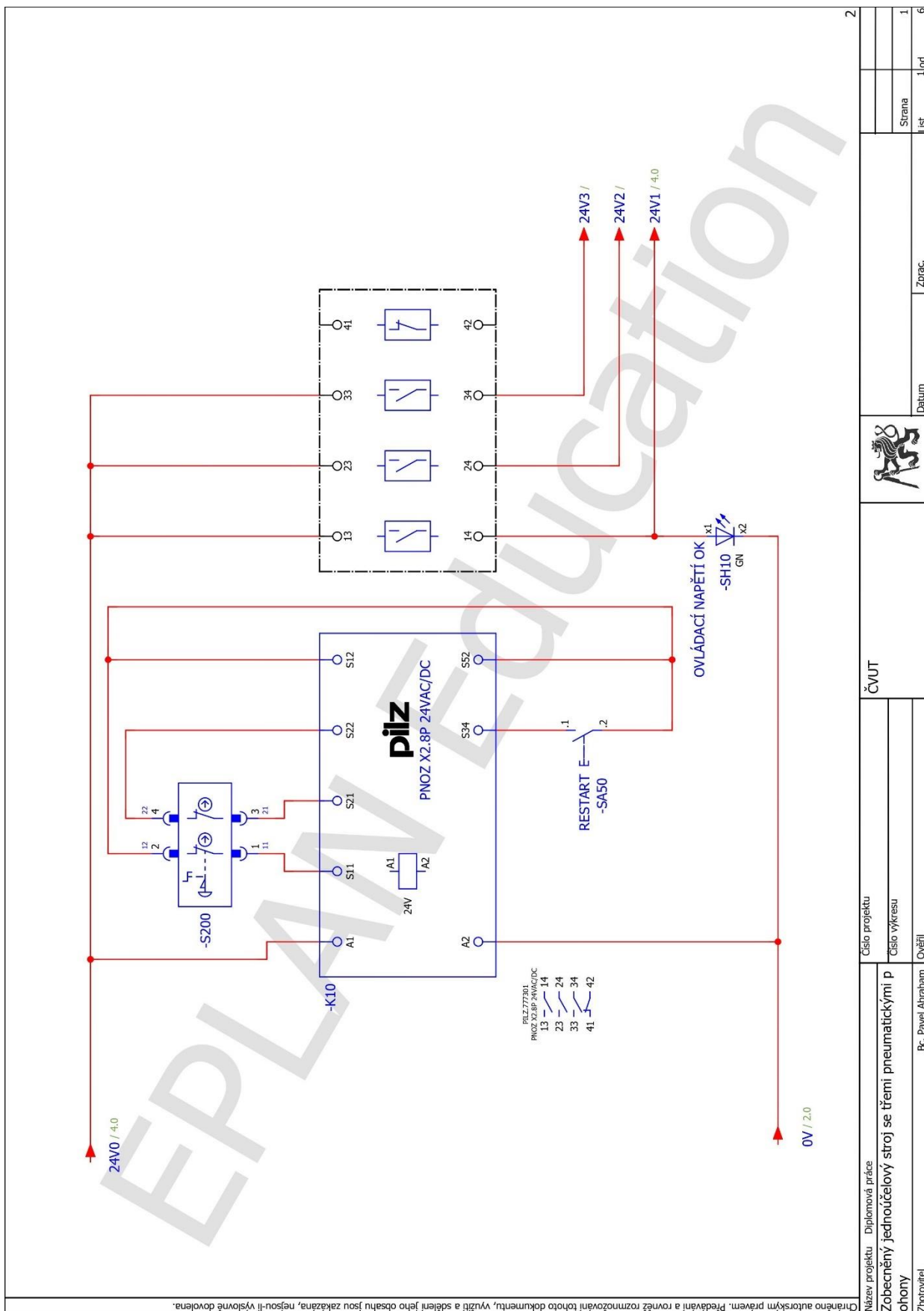
OBR 1 SIMATIC S7-1200 [2]	2
OBR 2 SIMATIC S7-1500 [5]	3
OBR 3 ET200SP S IO A BEZPEČNOSTNÍMI IO MODULY [7]	3
OBR 4 SIMATIC S7-400 [9]	4
OBR 5 SOUČÁSTI TIA PORTAL V16 [11]	5
OBR 6 ROZDĚLENÍ WINCC [13]	6
OBR 7 FUNKČNÍ BLOK K OVLÁDNÍ FESTO VTEM [18]	7
OBR 8 CYKLUS PLC	11
OBR 9 PŘÍKLAD DEKLAROVÁNÍ GLOBÁLNÍCH PROMĚNNÝCH	12
OBR 10 PŘÍKLAD DEKLAROVÁNÍ LOKÁLNÍCH PROMĚNNÝCH	12
OBR 11 PŘÍKLAD DEKLAROVÁNÍ UDT	13
OBR 12 PŘÍKLAD POUŽITÍ UDT NA VSTUPU FB	13
OBR 13 UKÁZKA GLOBÁLNÍHO DATOVÉHO BLOKU	13
OBR 14 SLICE ACCESS [20]	14
OBR 15 PŘÍKLAD STRUKTURALIZACE PROGRAMŮ	14
OBR 16 PŘÍKLAD SOFTWARE UNITS	15
OBR 17 PŘÍKLAD PROVÁZÁNÍ POMOCÍ VZTAHŮ	15
OBR 18 NORM_X A SCALE_X FUNKCE	16
OBR 19 INSTRUKCE V TIA PORTALU	17
OBR 20 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE STL [24]	18
OBR 21 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE GRAPH [25]	19
OBR 22 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE FBD	19
OBR 23 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE CEM [27]	20
OBR 24 PŘÍKLAD SCL NÁBĚŽNÁ HRANA	20
OBR 25 VÝVOJOVÝ DIAGRAM FUNKCE IF	21
OBR 26 PŘÍKLAD CASE FUNKCE	22
OBR 27 VÝVOJOVÝ DIAGRAM FUNKCE CASE	22
OBR 28 PŘÍKLAD FOR FUNKCE	23
OBR 29 VÝVOJOVÝ DIAGRAM FUNKCE FOR	23
OBR 30 PŘÍKLAD WHILE FUNKCE	24
OBR 31 VÝVOJOVÝ DIAGRAM FUNKCE WHILE	24
OBR 32 PŘÍKLAD REPEAT FUNKCE	25
OBR 33 VÝVOJOVÝ DIAGRAM FUNKCE REPEAT	25
OBR 34 KONTAKTY V JAZYCE LAD	26
OBR 35 CÍVKY V JAZYCE LAD	26
OBR 36 PŘÍKLAD PROGRAMOVÁNÍ V JAZYCE LAD	27
OBR 37 ZDVOJENÝ VÝSTUP V LAD	28
OBR 38 VÝSLEDEK KOMPILACE ZDVOJENÉHO VÝSTUPU	28
OBR 39 DEFINOVÁNÍ PŘEKRÝVAJÍCÍCH SE PROMĚNNÝCH	28
OBR 40 CROSS-REFERENCE	29
OBR 41 UKÁZKA DĚLENÍ RŮZNÝCH DATOVÝCH TYPŮ	29
OBR 42 MONITORING V JAZYCE SCL	29
OBR 43 MONITORING V JAZYCE LAD	30
OBR 44 UKÁZKA FORCE TABULKY	30
OBR 45 UKÁZKA WATCH TABULKY	30
OBR 46 UKÁZKA NAHRANÝCH HODNOT [6]	31
OBR 47 UKÁZKA BREAKPOINTŮ	31

OBR 48 KROKOVÁNÍ BREAKPOINTŮ	32
OBR 49 KOMPAKTNÍ ROZHRANÍ - PLCSIM	32
OBR 50 PROJEKTOVÉ ROZHRANÍ - PLCSIM	33
OBR 51 S7-PLCSIM ADVANCED	34
OBR 52 NÁVRH OVLÁDACÍHO PANELU	38
OBR 53 PROVOZNÍ REŽIMY	39
OBR 54 FUNKČNÍ BLOK PROVOZNÍCH REŽIMŮ	40
OBR 55 FUNKČNÍ BLOK ŘÍZENÍ VÁLCE.....	41
OBR 56 PARAMETRIZACE V KROKYFB	42
OBR 57 PŘÍKLAD JEDNOHO KROKU SEKVENCE	42
OBR 58 LOGIKA SIGNALIZAČNÍ LED	43
OBR 59 FUNKČNÍ BLOK VÝROBNÍ CYKLU	43
OBR 60 PRÁCE SE POLI VÝROBNÍCH ČASŮ A UKÁZKA KONVERZE DATOVÉHO TYPU	44
OBR 61 FUNKČNÍ BLOK PŘECHODU	44
OBR 62 KONTROLA KONEČNÝCH SENZORŮ VE FUNKČNÍM BLOKU PŘECHODFB	44
OBR 63 FUNKČNÍ BLOK VÝSTUPY	45
OBR 64 UKÁZKA ČÁSTI NASTAVOVÁNÍ VÝSTUPU PODLE PARAMETRIZACE	45
OBR 65 FUNKČNÍ BLOK PROVOZNÍ DATA	46
OBR 66 RESETOVÁNÍ A NATAVENÍ CHYBOVÉ HLÁŠKY	47
OBR 67 MAPA STRÁNEK.....	48
OBR 68 ŠABLONA ALARMY	49
OBR 69 NASTAVENÍ ŠABLONY VE VLASTNOSTECH STRÁNKY	49
OBR 70 HMI OBRAZOVKA AUTOMATICKÝ REŽIM	50
OBR 71 HMI OBRAZOVKA PARAMETRY - VÝROBNÍ SEKVENCE	50
OBR 72 FUNKCE TLAČÍTKA ZVYŠOVÁNÍ PROMĚNNÉ O 1	51
OBR 73 HMI OBRAZOVKA VYROBENO KUSŮ ZA 24 HODIN	51
OBR 74 HMI OBRAZOVKA SERVIS	52
OBR 75 ROZLOŽENÍ PRVKŮ PŘI POUŽITÍ FACEPLATE	53
OBR 76 NAPÁROVÁNÍ PRVKŮ V RÁMCI FACEPLATE	53
OBR 77 VIZUALIZACE POHYBU VÁLCE A ZOBRAZENÍ CHYBY	53
OBR 78 VYTVOŘENÍ UŽIVATELSKÉ ÚLOHY.....	54
OBR 79 NASTAVENÍ ÚLOHY K UKLÁDÁNÍ AKTIVNÍHO UŽIVATELE	54
OBR 80 PROMĚNNÉ AKTIVNÍHO UŽIVATELE	54
OBR 81 PROVÁZÁNÍ ČASU MEZI PLC A HMI	54

SEZNAM TABULEK

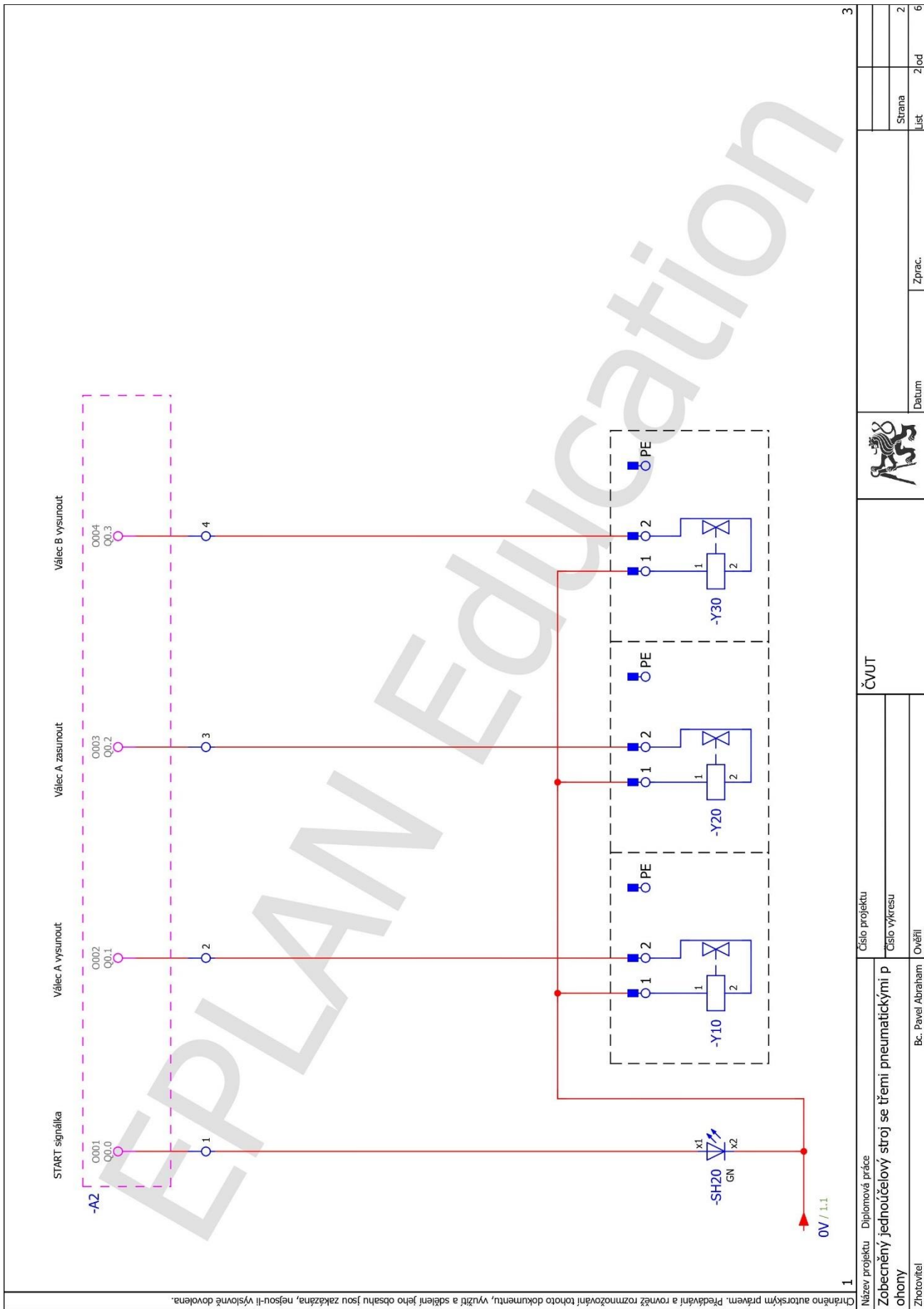
TABULKA 1 DRUHY CPU S7-1200	2
TABULKA 2 PŘEDPONY PRO DATOVÉ TYPY	8
TABULKA 3 PŘEDPONY PRO TŘÍDY PROMĚNNÝCH	8
TABULKA 4 PŘEDPONY PRO DRUHY PROMĚNNÝCH.....	8
TABULKA 5 PŘEDPONY PRO TYPY PROMĚNNÝCH	10
TABULKA 6 KVALIFIKÁTORY AKCÍ GRAPH JAZYKA	18
TABULKA 7 SEZNAM PNEUMATICKÝCH KOMPONENT	36
TABULKA 8 TABULKA VSTUPŮ	37
TABULKA 9 TABULKA VÝSTUPŮ	37
TABULKA 10 CHYBOVÉ HLÁŠKY	47

PŘÍLOHA A – SCHÉMATA



Chráněno autorským právem. Předávání a rozveřtí rozmnožování tohoto dokumentu, využití a sdělení jeho obsahu jsou zakázána, nejsou-li výslovně dovolena.

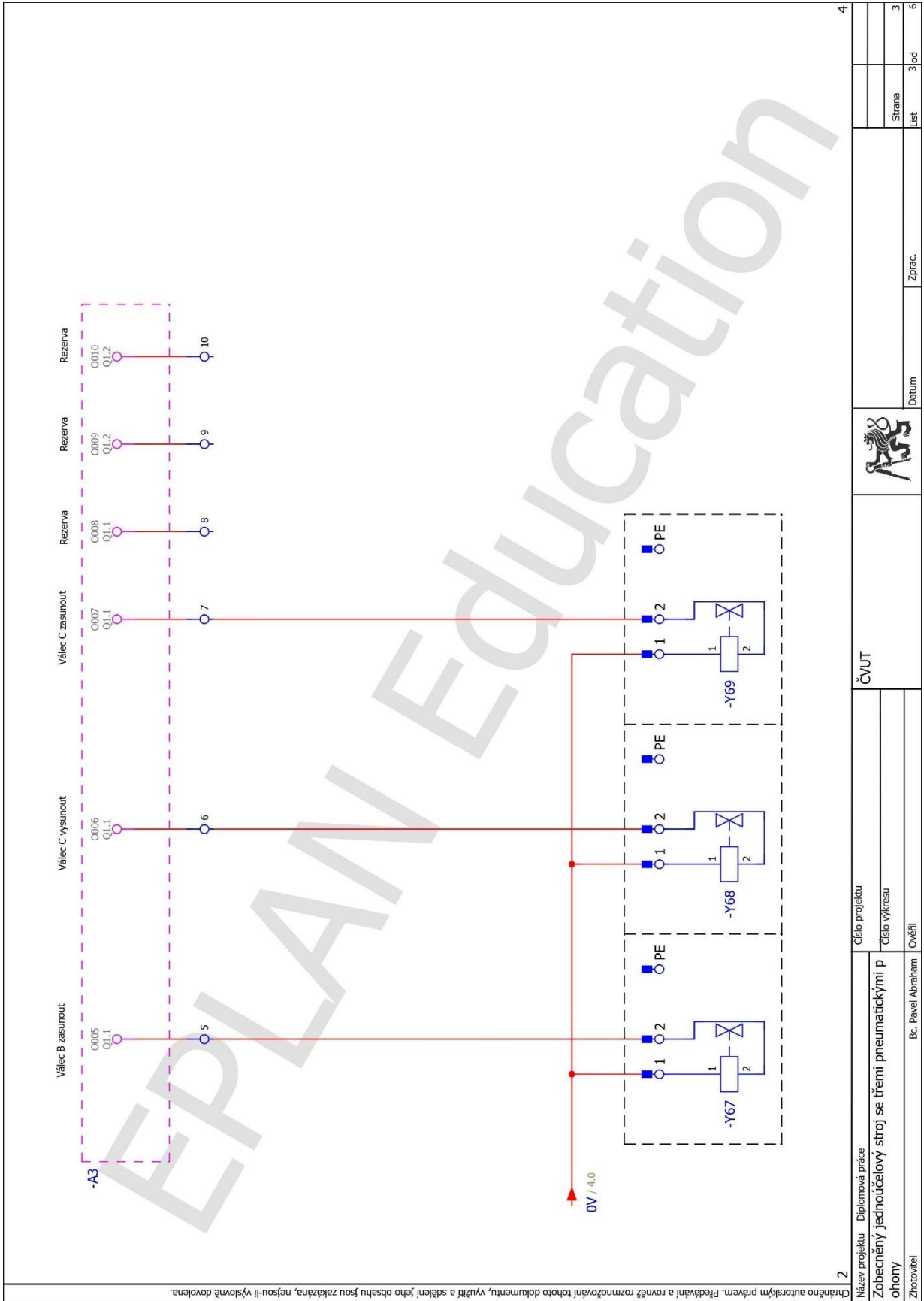
PŘÍLOHA I ZJEDNODUŠENÉ SCHÉMA BEZPEČNOSTI



Chráňného autorským právem. Předávání a rovněž rozmnožování tohoto dokumentu, využití a sdělení jeho obsahu jsou zakázána, nejsou-li výslovně dovolena.

PŘÍLOHA II ZJEDNODUŠENÉ SCHÉMA VÝSTUPŮ

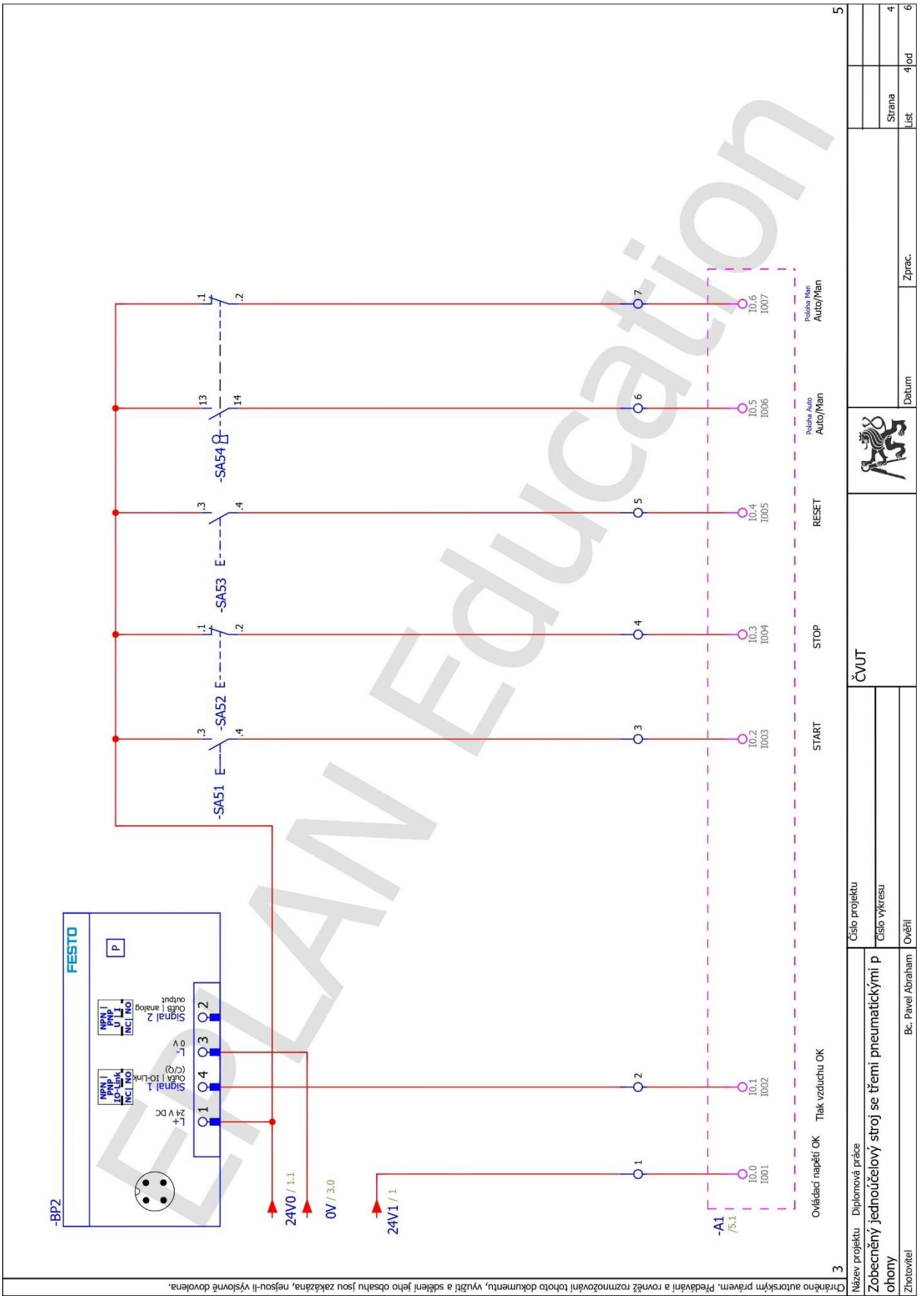
1		3	
Název projektu: Diplomová práce		Číslo projektu	
Zobecněný jednocíselový stroj se třemi pneumatickými p		Číslo výkresu	
ohony		Ověřil	
Zhotovitel: Bc. Pavel Abraham		Datum	
		Zprac.	
		Strana	
		List	
		2	
		2	
		6	



Chráněno autorským právem. Předávání a rovněž rozmnožování tohoto dokumentu, využití a sčlenění jeho obsahu jsou zakázána, nejsou-li výslovně dovolena.

PŘÍLOHA III ZJEDNODUŠENÉ SCHÉMA VÝSTUPŮ

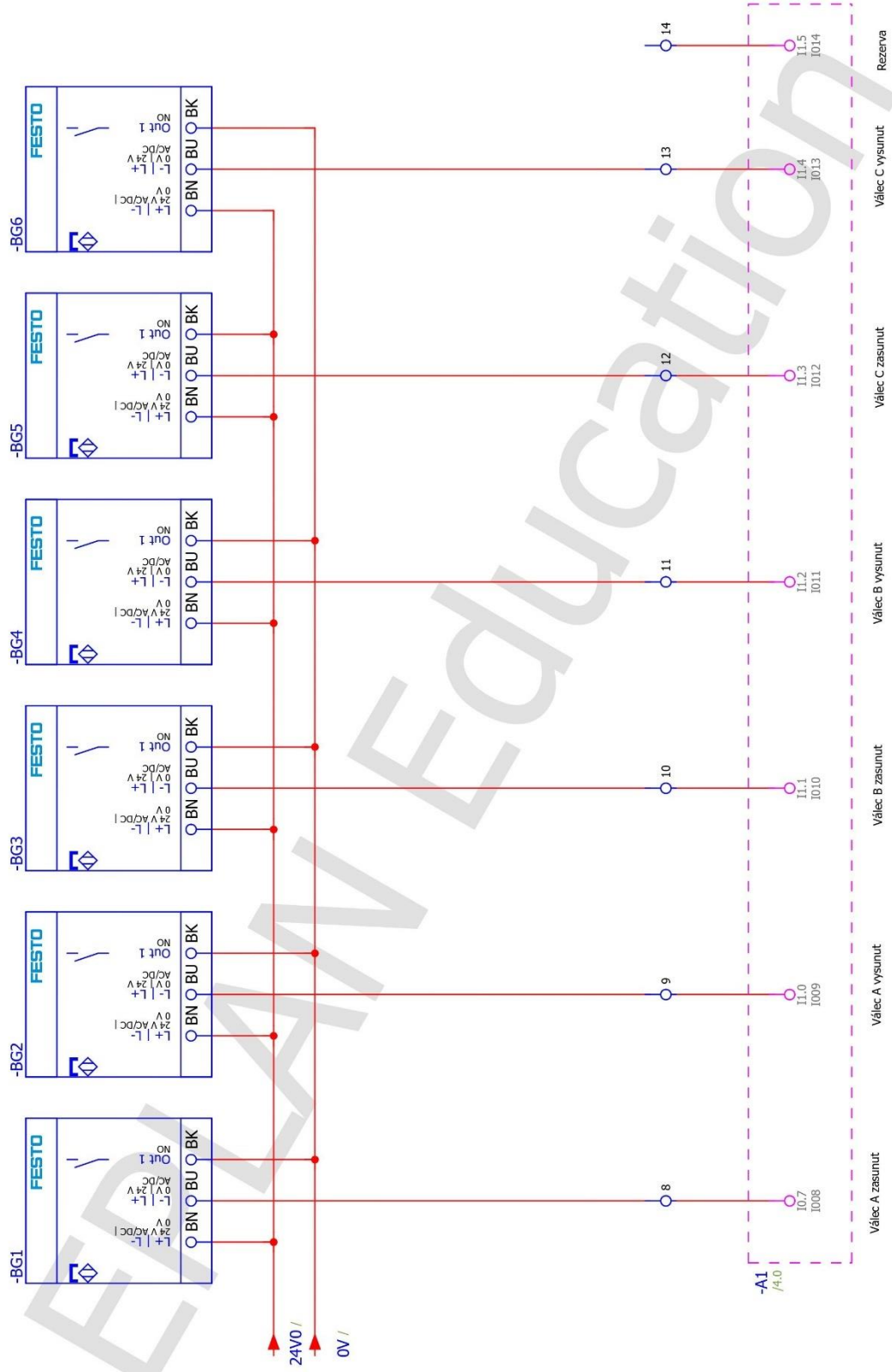
2	Cílo projektu	ČVUT	4
Název projektu	Diplomová práce		
Zobecněný jednoúčelový stroj se třemi pneumatickými p	Cílo výkresu		
ohonů			
Zhotovitel	Bc. Pavel Abraham		
		Datum	
		Zprac.	
		Strana	3
		List	3 od 6



Právního autorským právem. Předáváni a rovněž rozmnožování tohoto dokumentu, využití a sdělení jeho obsahu jsou zakázána, nejsou-li výslovně dovolena.

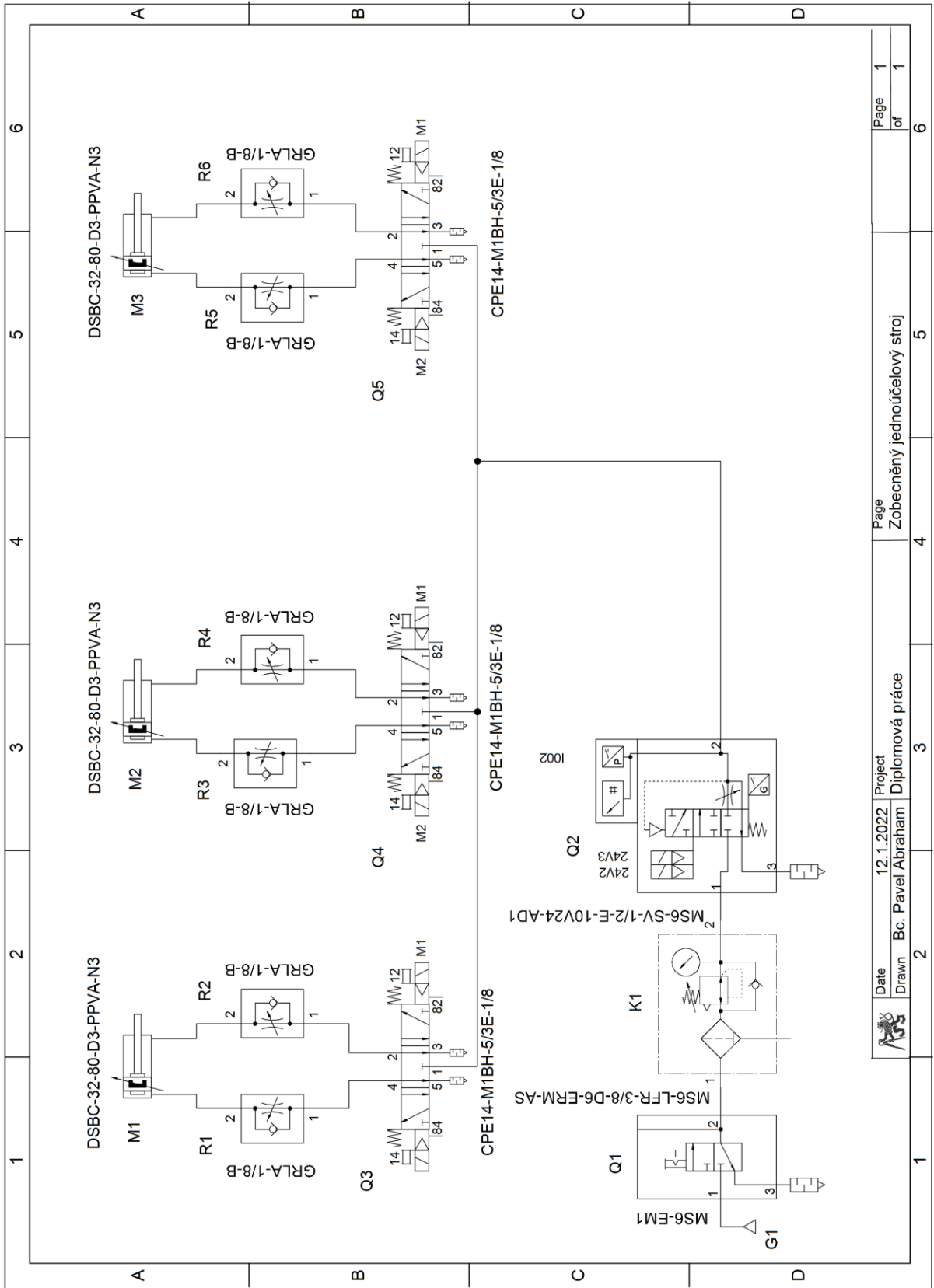
PŘÍLOHA IV ZJEDNODUŠENÉ SCHÉMA VSTUPŮ

Chráněno autorským právem. Předávání a rovněž rozmnožování tohoto dokumentu, využití a sdělení jeho obsahu jsou zakázána, nejsou-li výslovně dovolena.



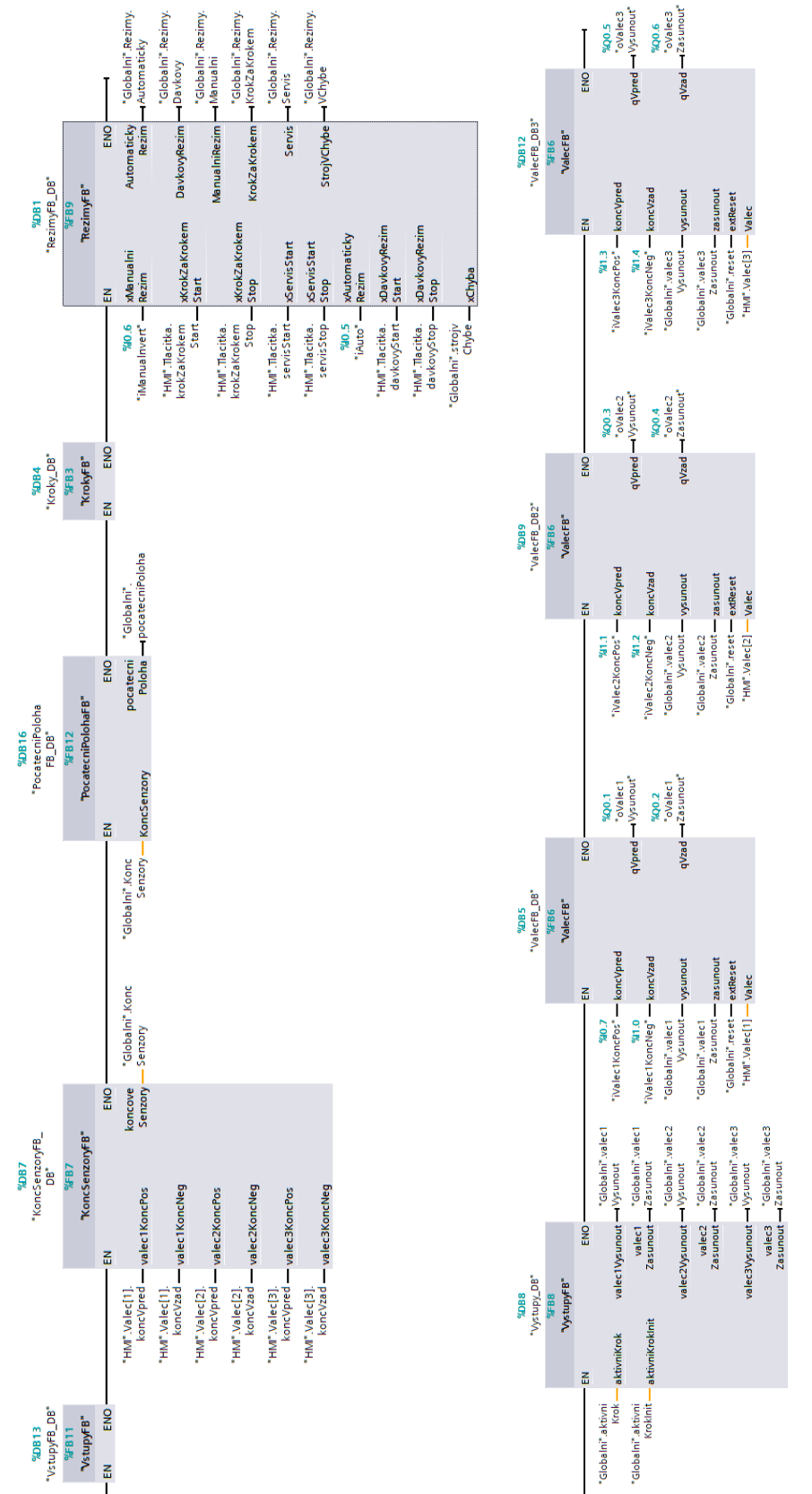
4		6	
Název projektu: Diplomová práce		ČVUT	
Zobecněný jednocelový stroj se třemi pneumatickými p ohony			
Zhotovitel: Bc. Pavel Abraham	Ověřil:	Datum:	Zprac.
		Strana	List
		5	6

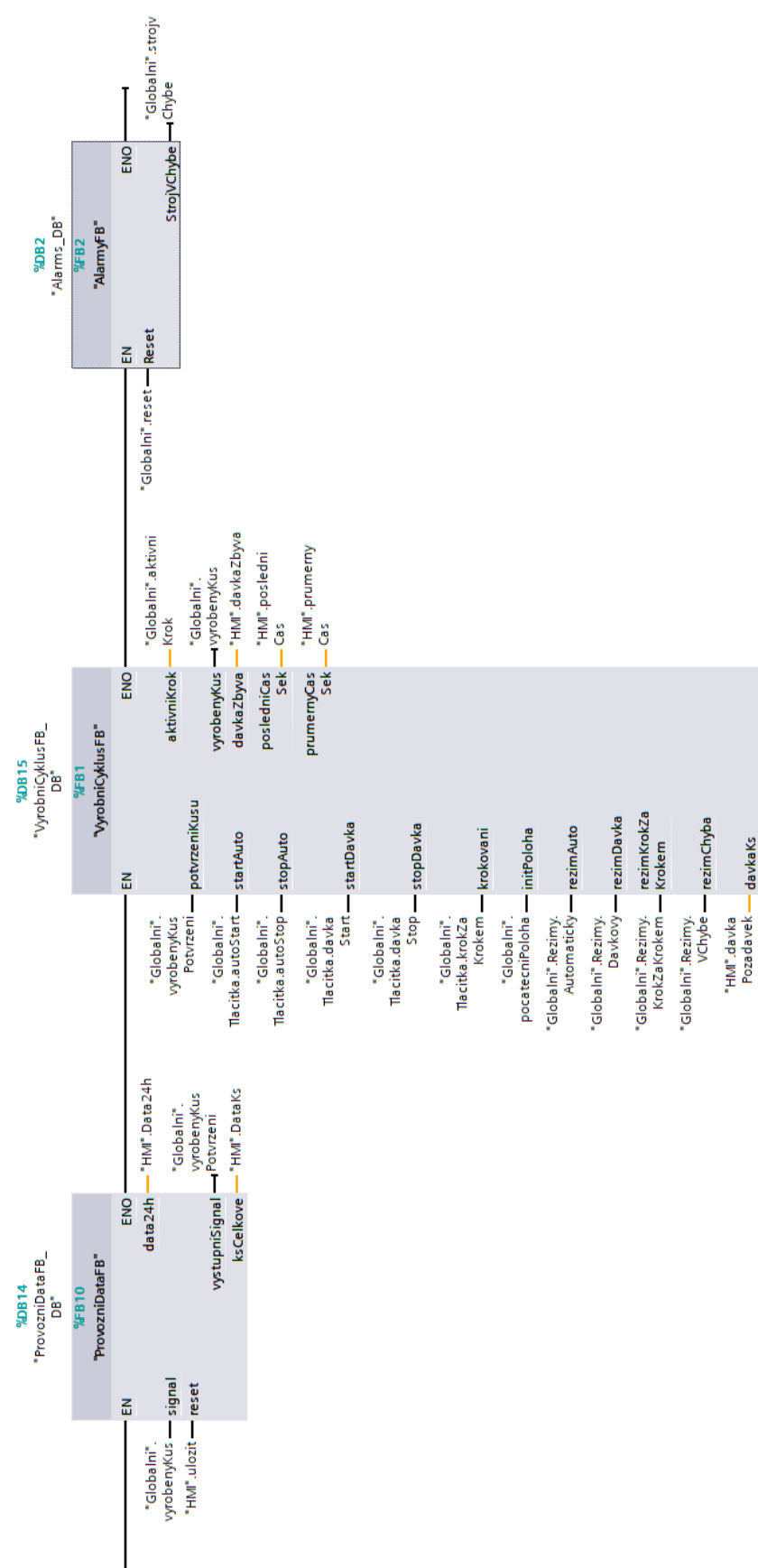
PŘÍLOHA V ZJEDNODUŠENÉ SCHÉMA VSTUPŮ

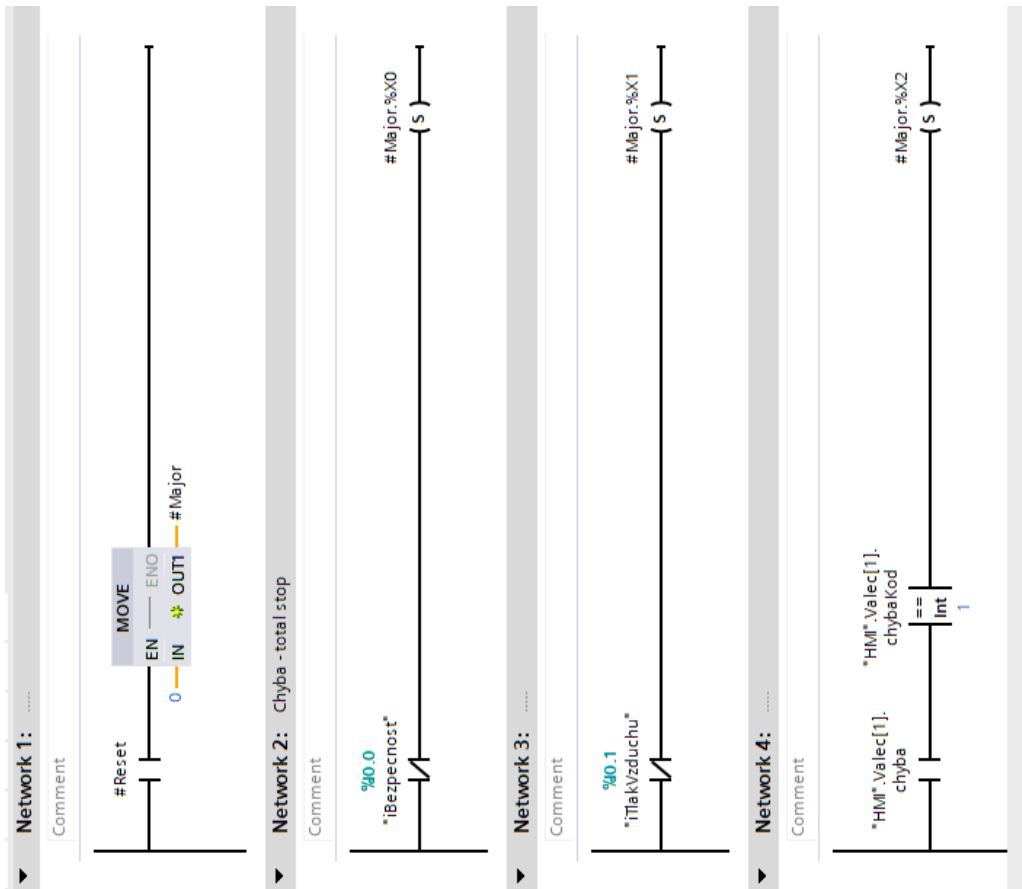


PŘÍLOHA VI PNEUMATICKÉ SCHÉMA

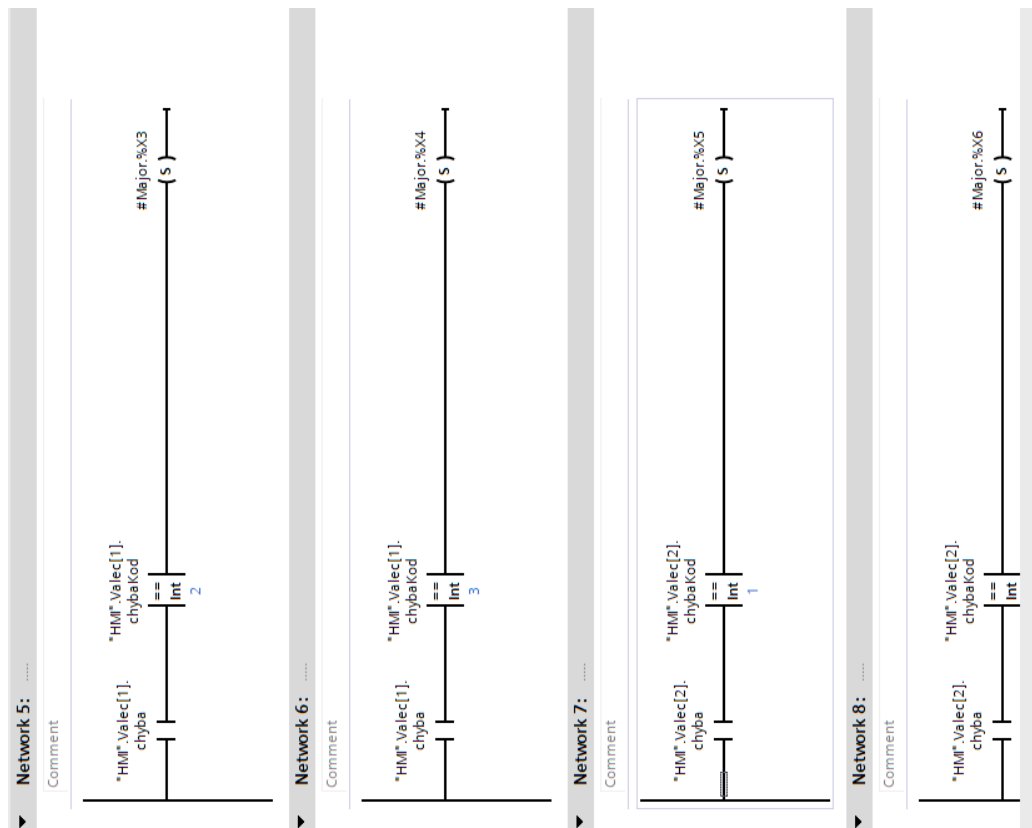
PŘÍLOHA B – PROGRAM



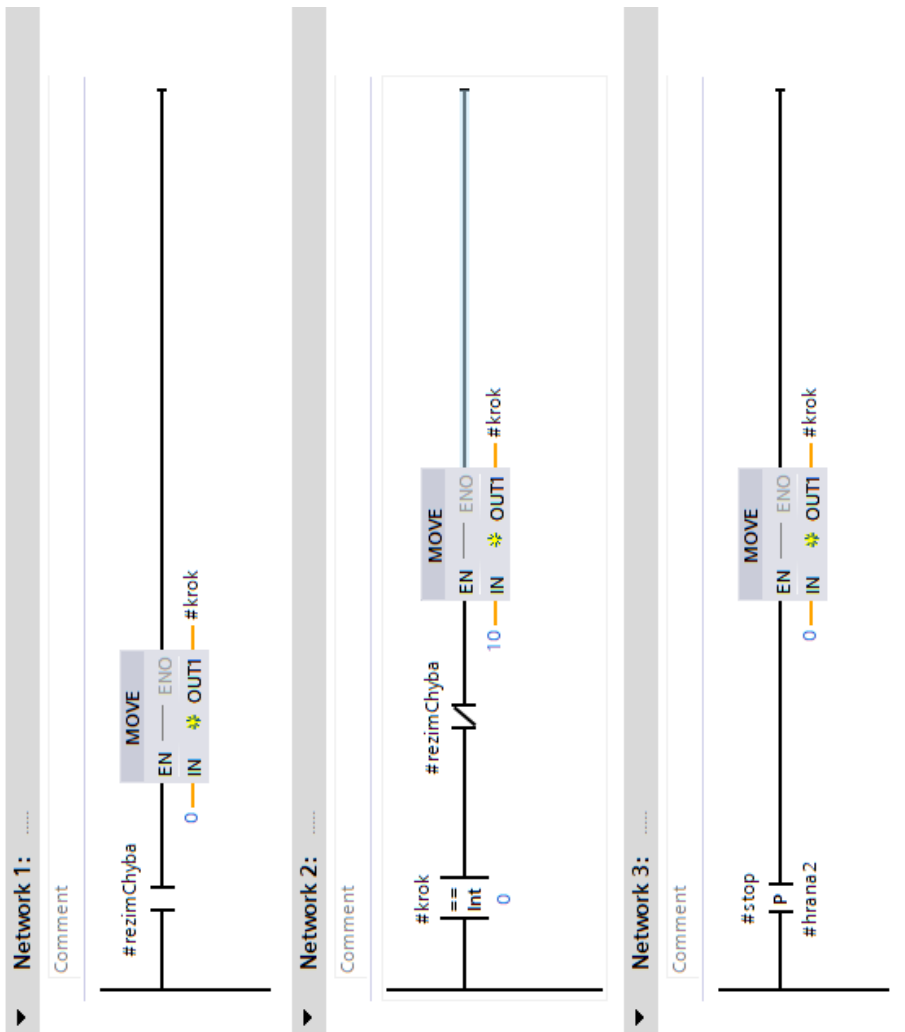




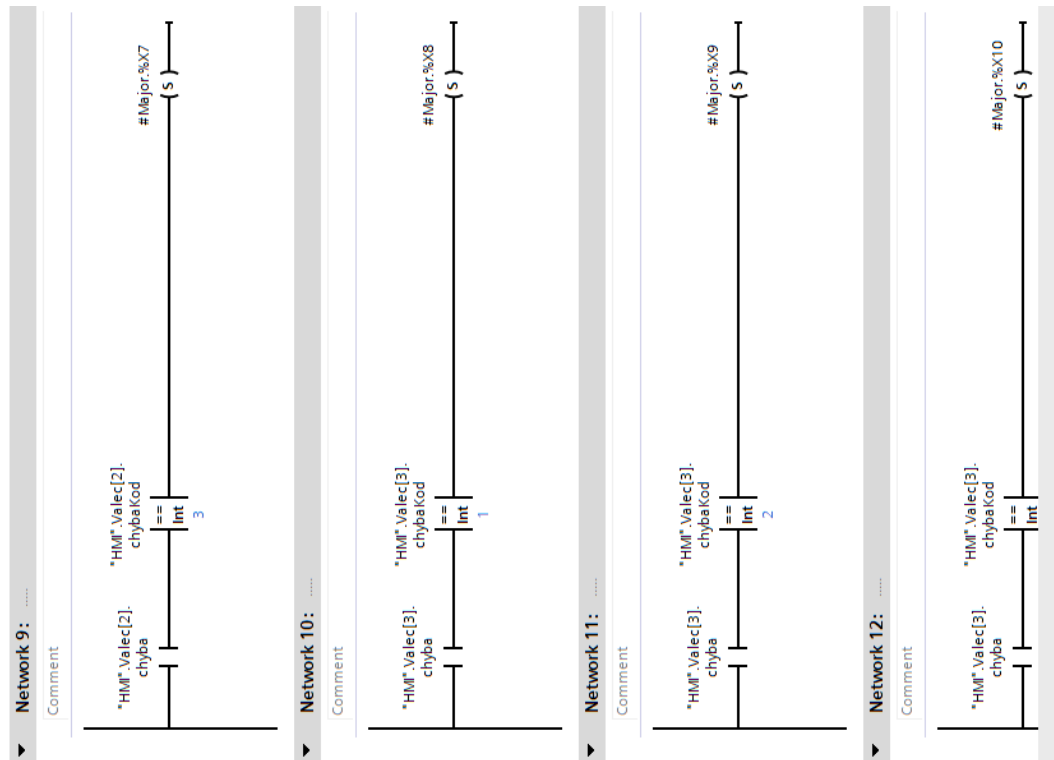
PŘÍLOHA IX ALARMYFB



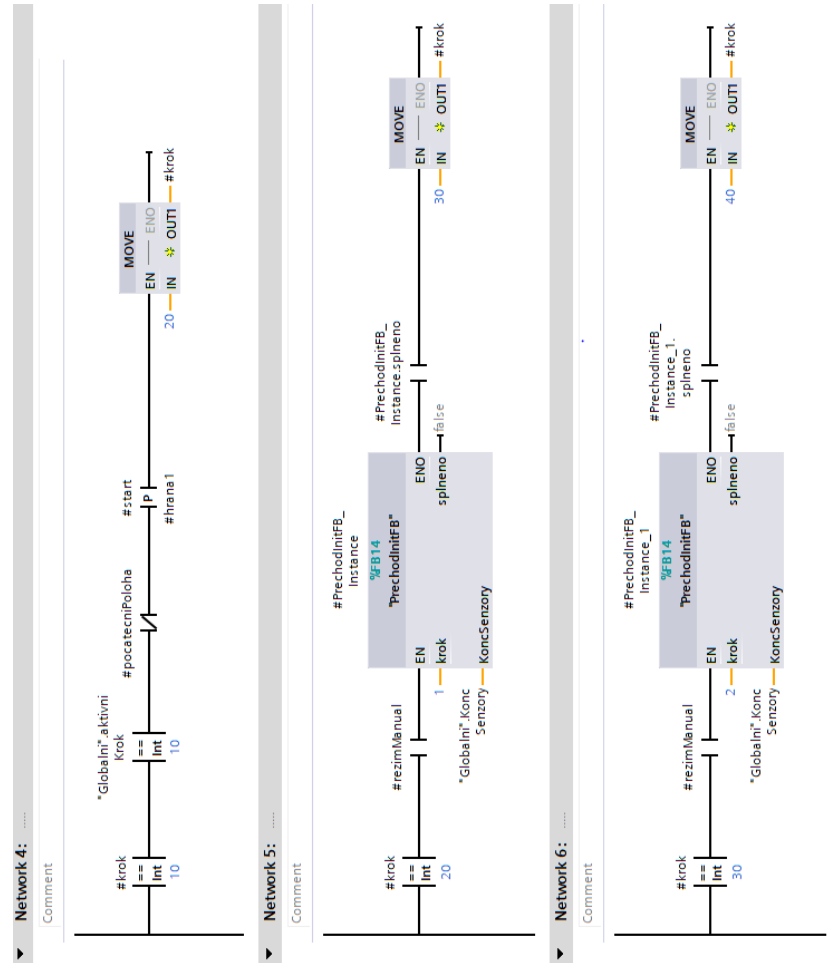
PŘÍLOHA X ALARMYFB



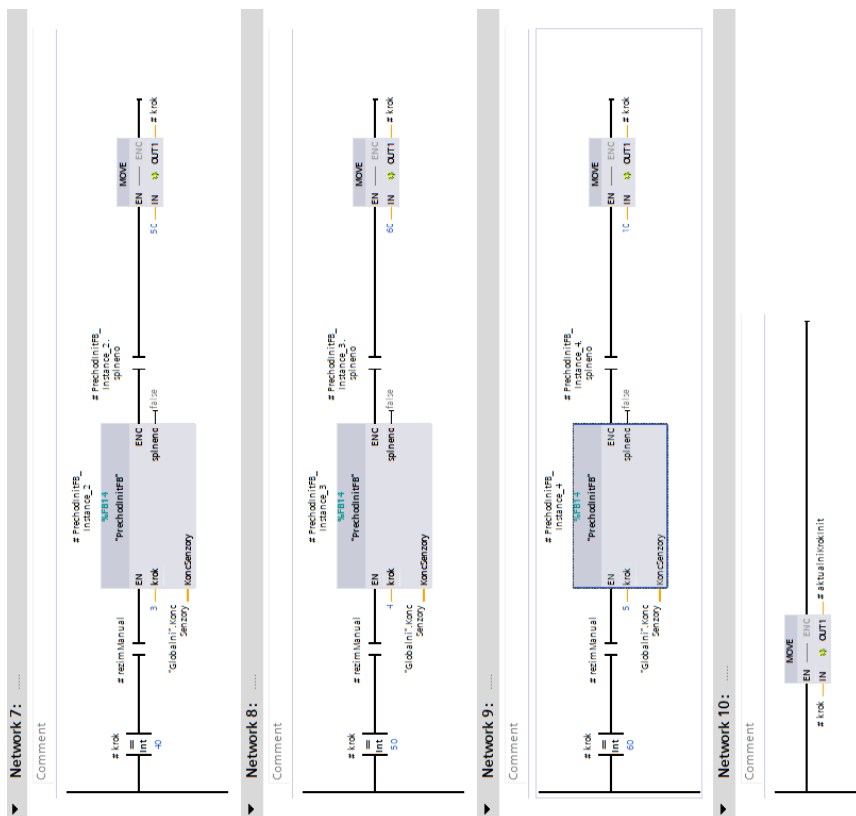
PŘÍLOHA XII ALARMYFB



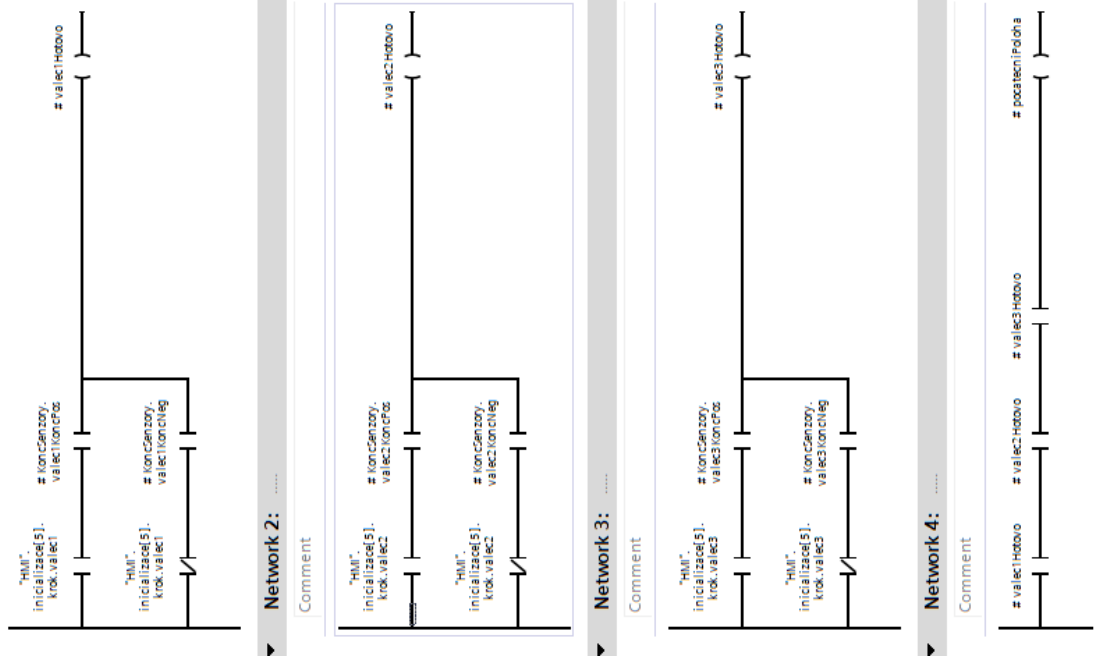
PŘÍLOHA XI ALARMYFB



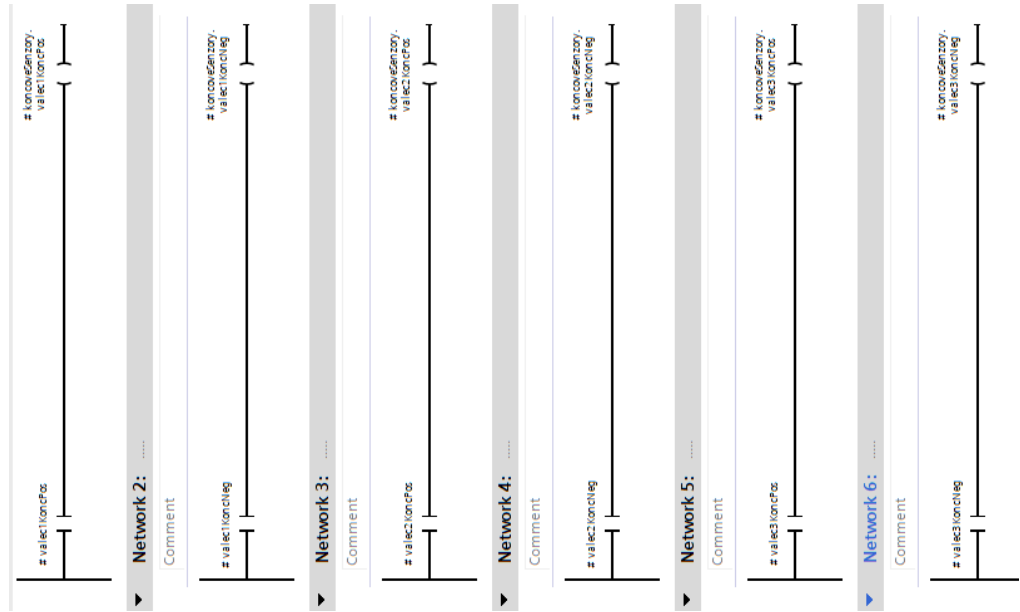
PRÍLOHA XIII ALARMYFB



PRÍLOHA XIV ALARMYFB



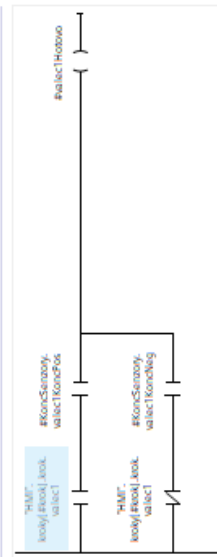
PŘÍLOHA XVI POČATECNIPOLOHAFB



PŘÍLOHA XV KONCENZORYFB

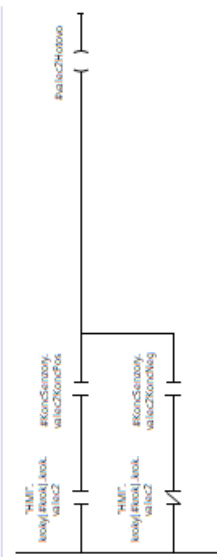
▼ **Network 1:**

Comment



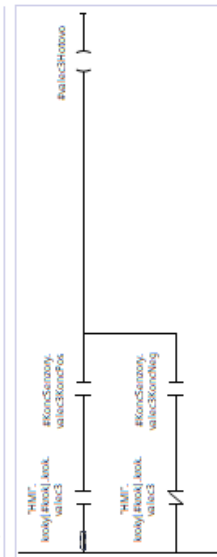
▼ **Network 2:**

Comment



▼ **Network 3:**

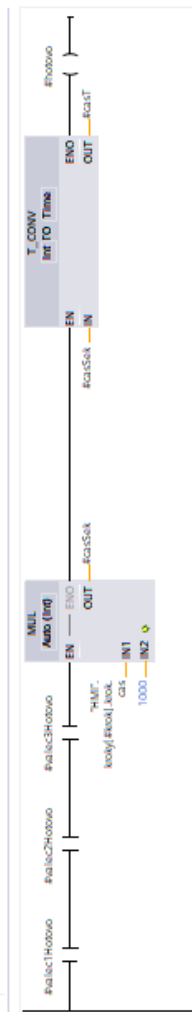
Comment



▼ **PŘÍLOHA XVII PRECHODFB**

▼ **Network 4:**

Comment



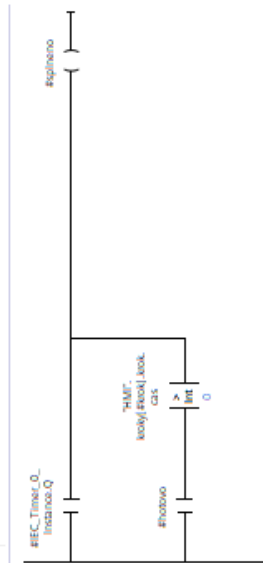
▼ **Network 5:**

Comment

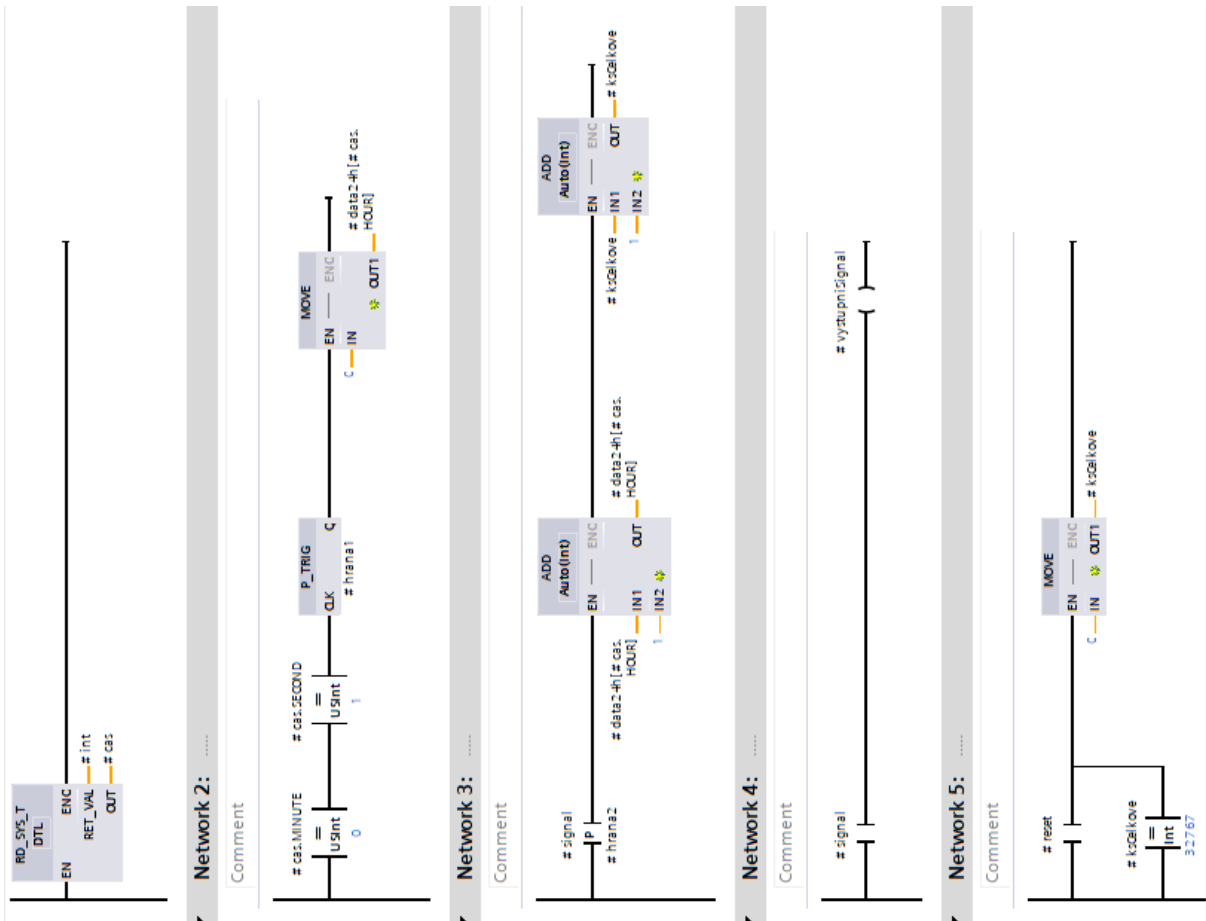


▼ **Network 6:**

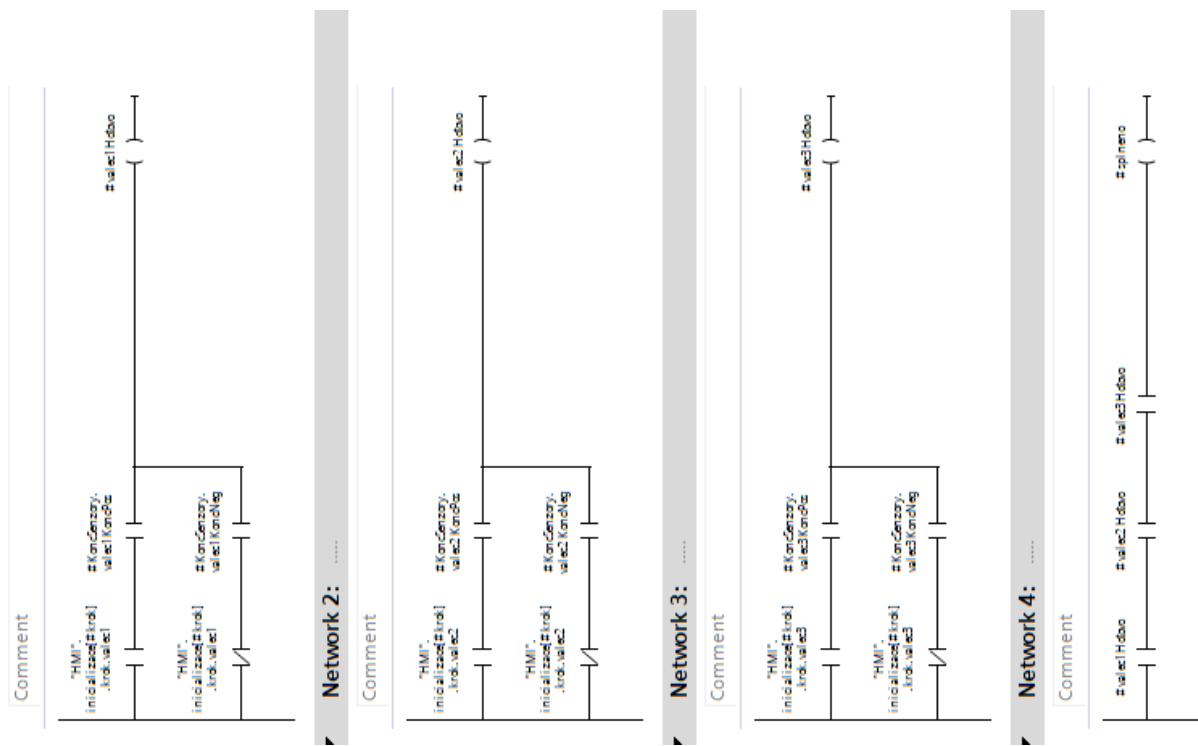
Comment



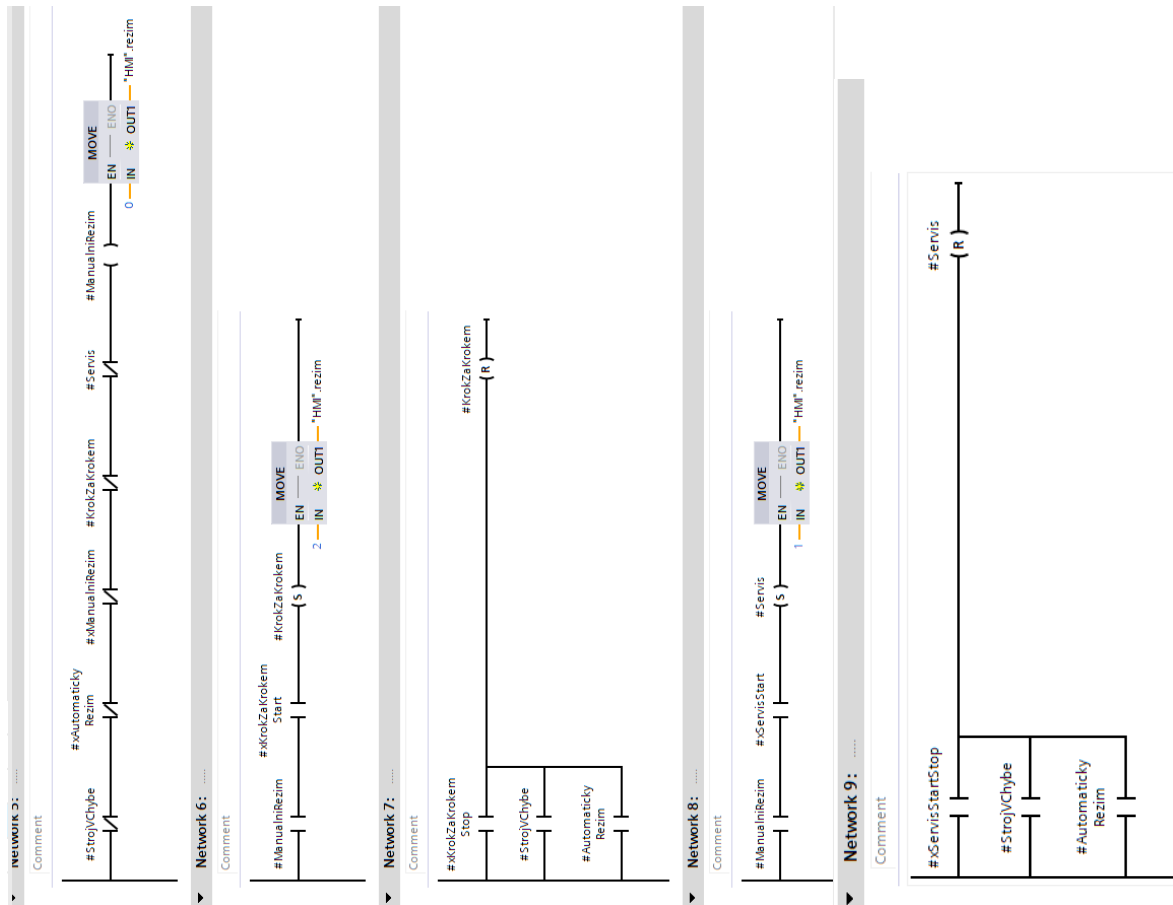
▼ **PŘÍLOHA XVIII PRECHODFB**



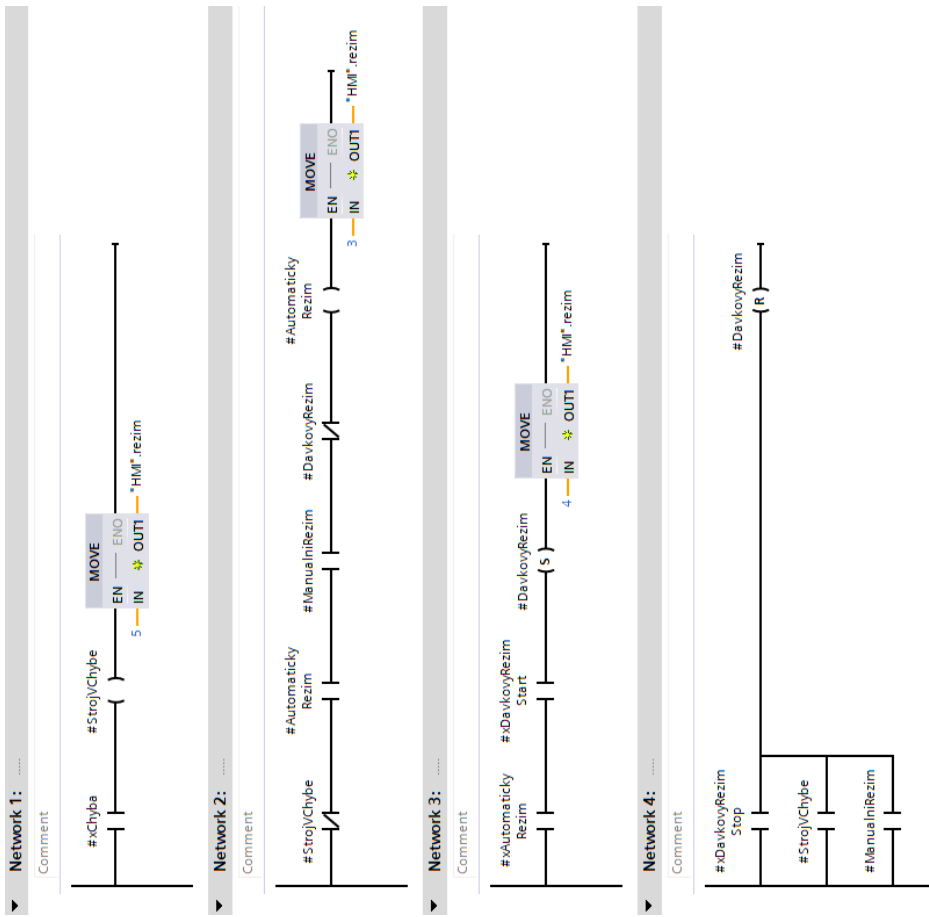
PŘÍLOHA XX PROVOZNI DATAFB



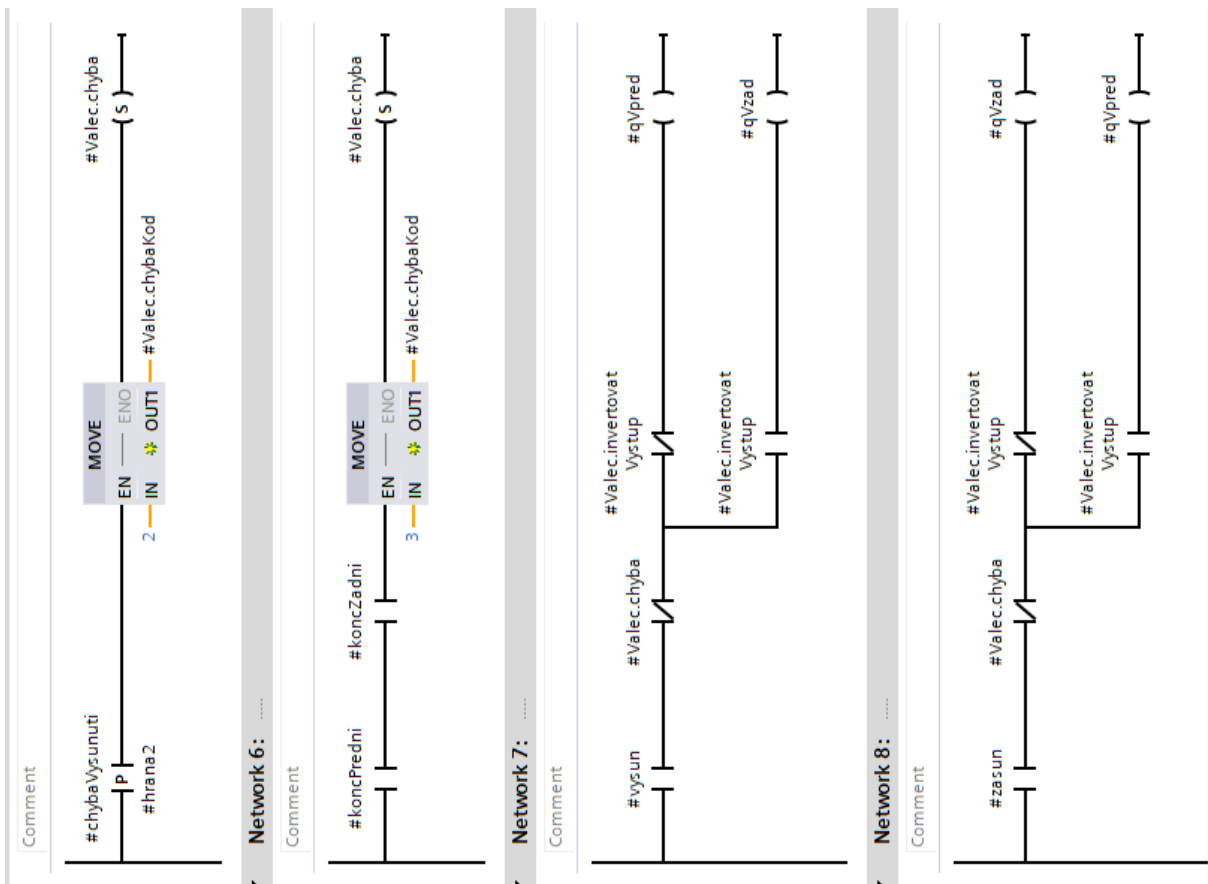
PŘÍLOHA XIX PRECHODINITFB



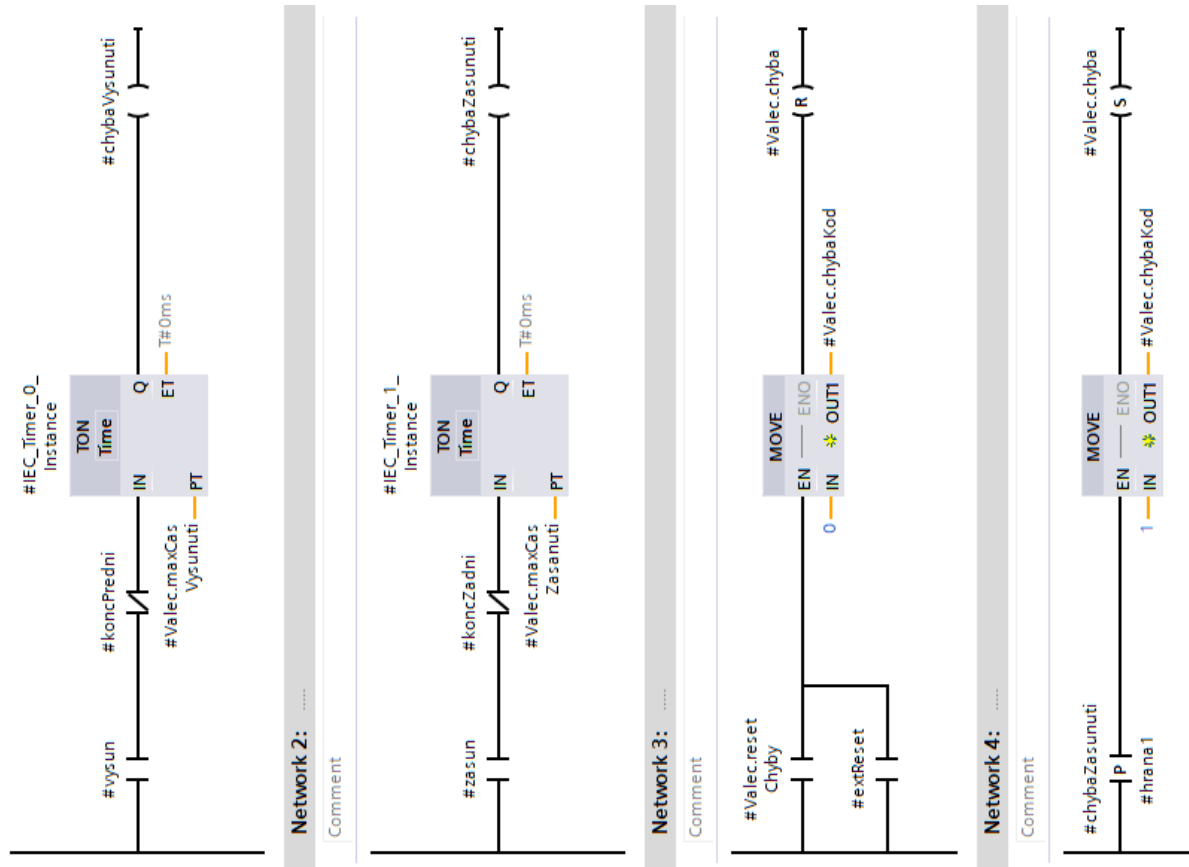
PŘÍLOHA XXII



PŘÍLOHA XXI REZIMYFB



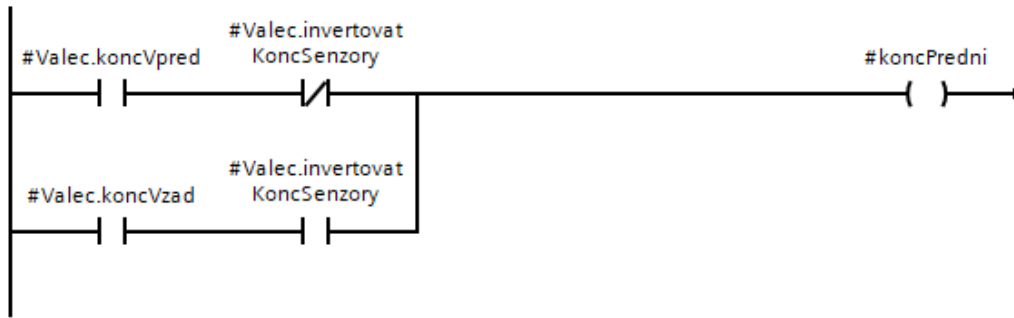
PŘÍLOHA XXIV VALECFB



PŘÍLOHA XXIII VALECFB

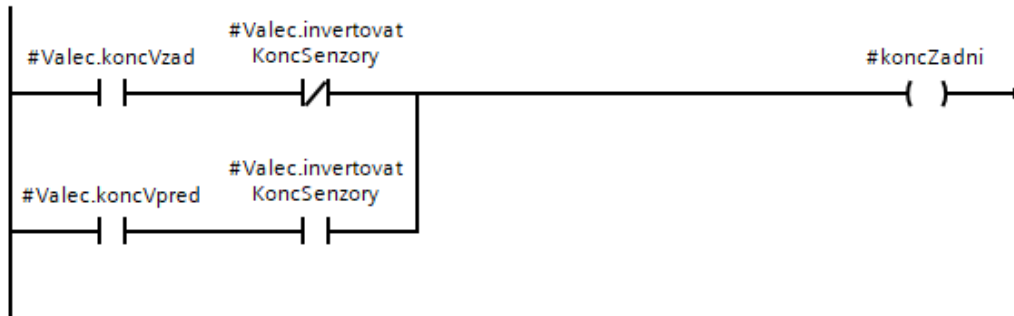
▼ Network 9:

Comment

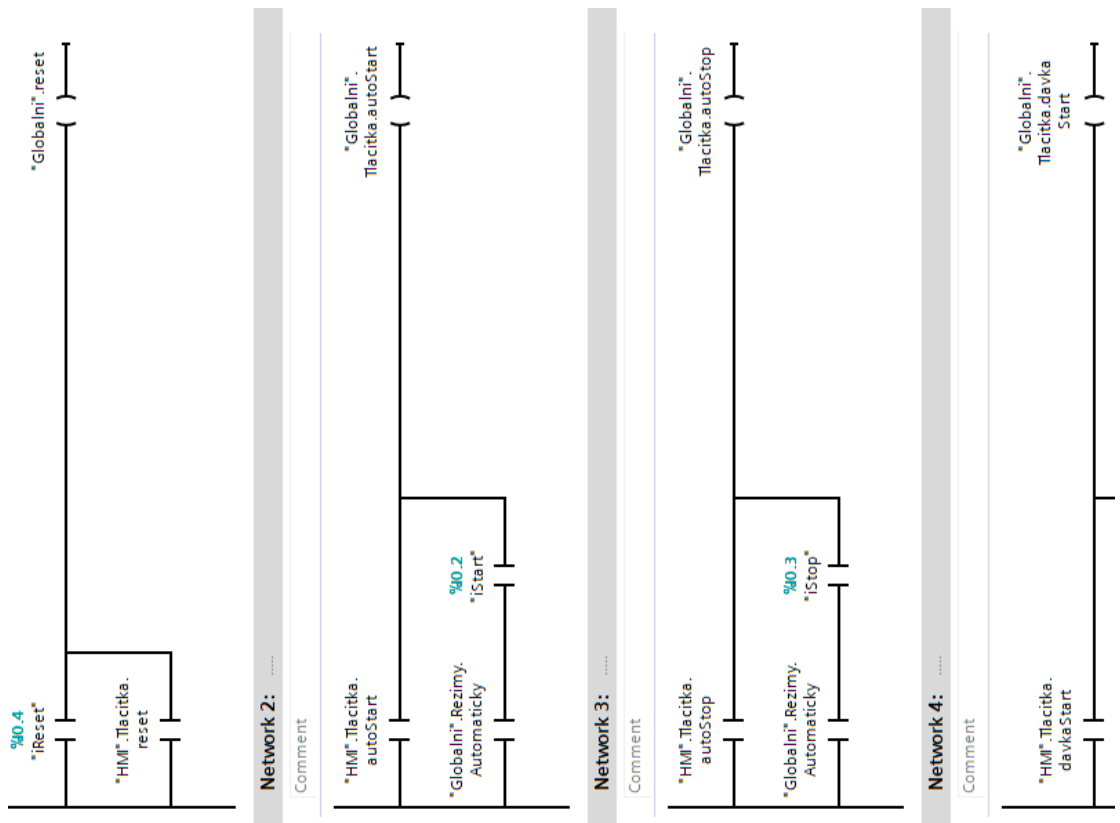


▼ Network 10:

Comment



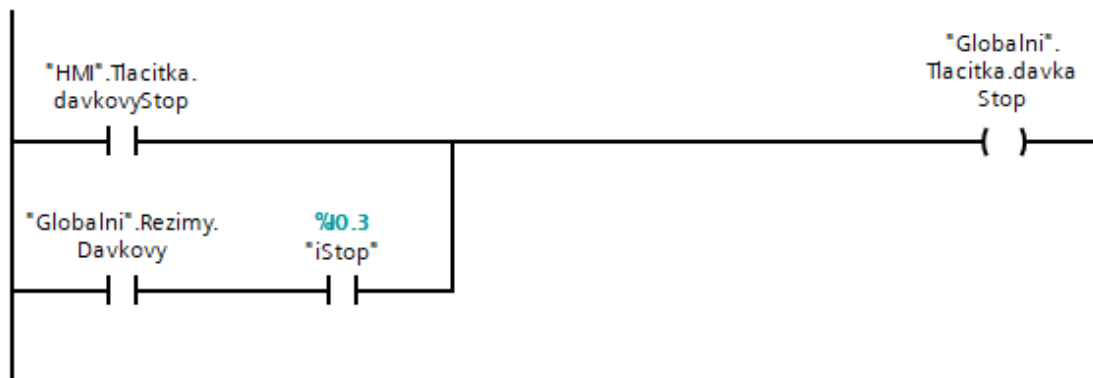
PŘÍLOHA XXV VALECFB



PŘÍLOHA XXVI VSTUPYFB

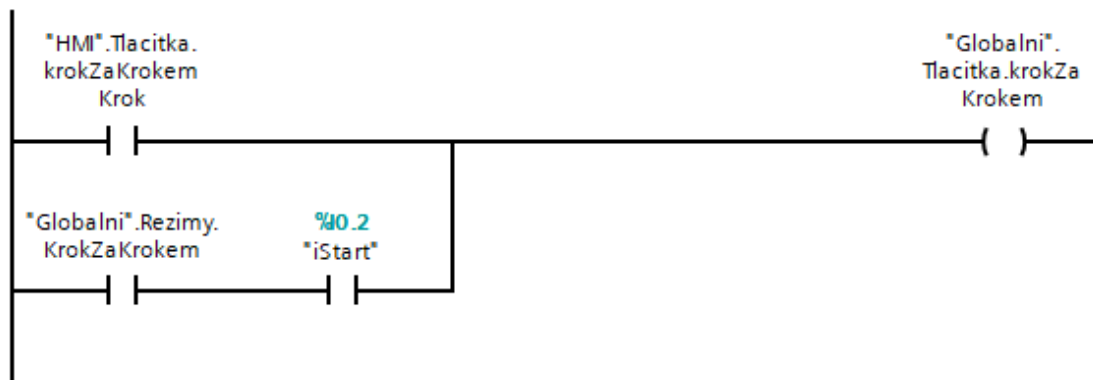
▼ **Network 5:**

Comment

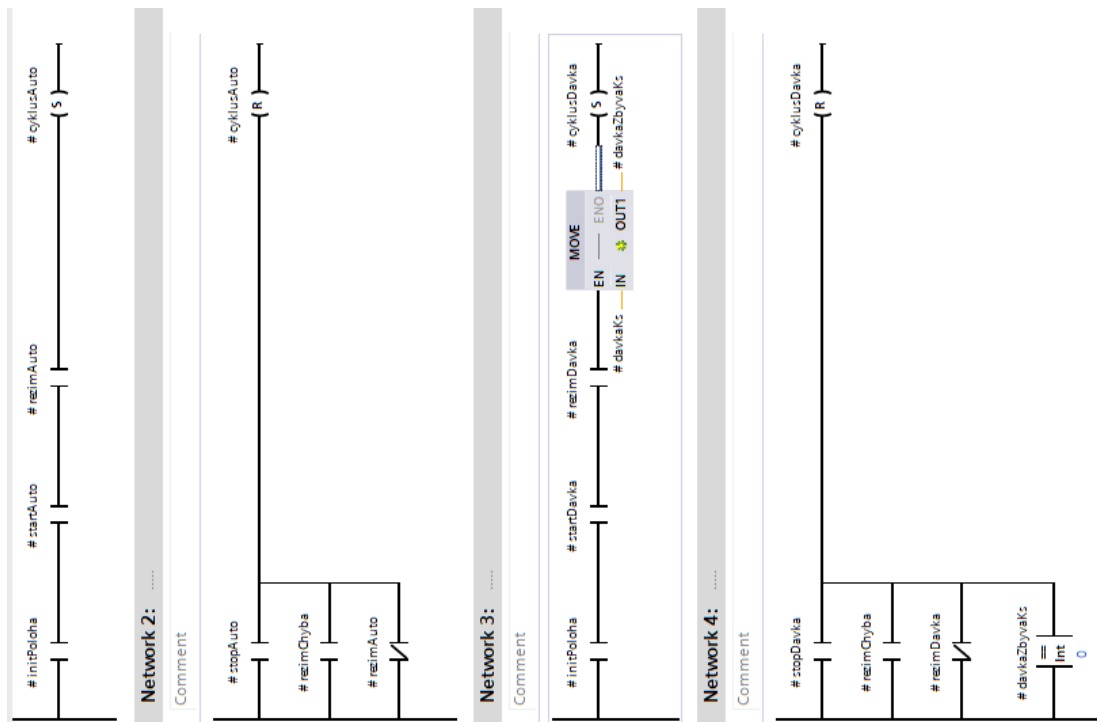


▼ **Network 6:**

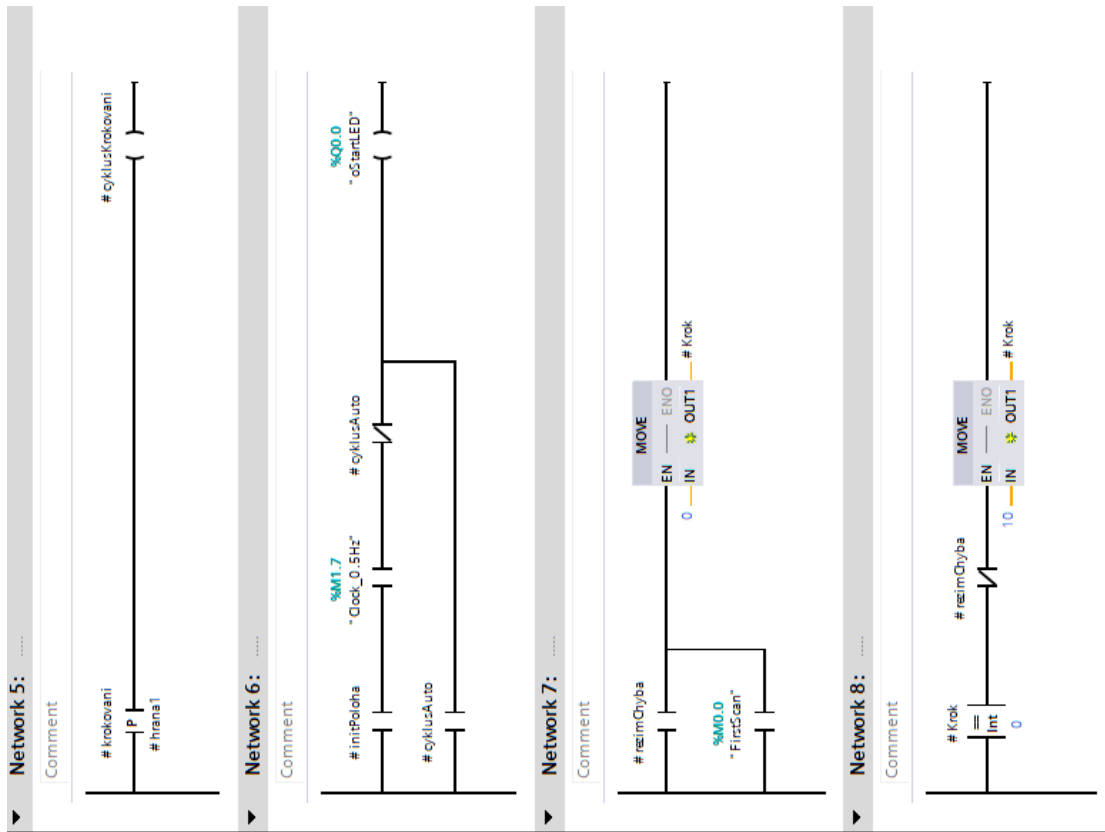
Comment



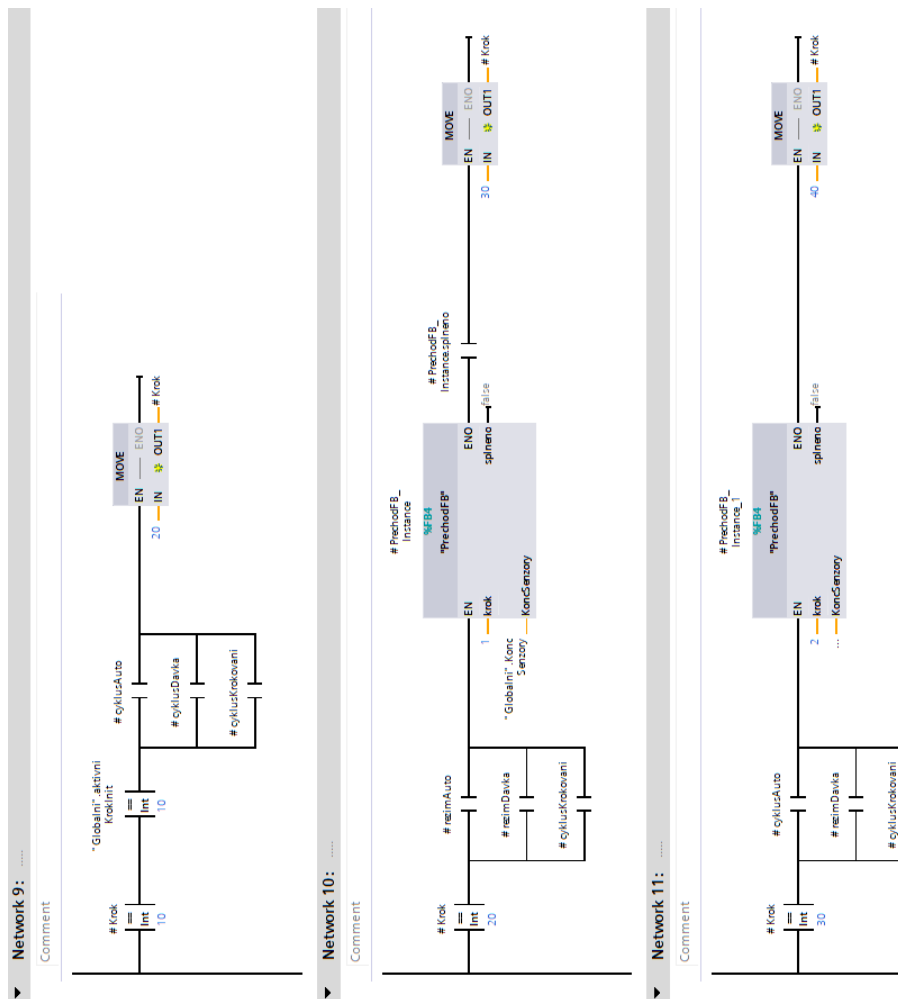
PŘÍLOHA XXVII VSTUPYFB



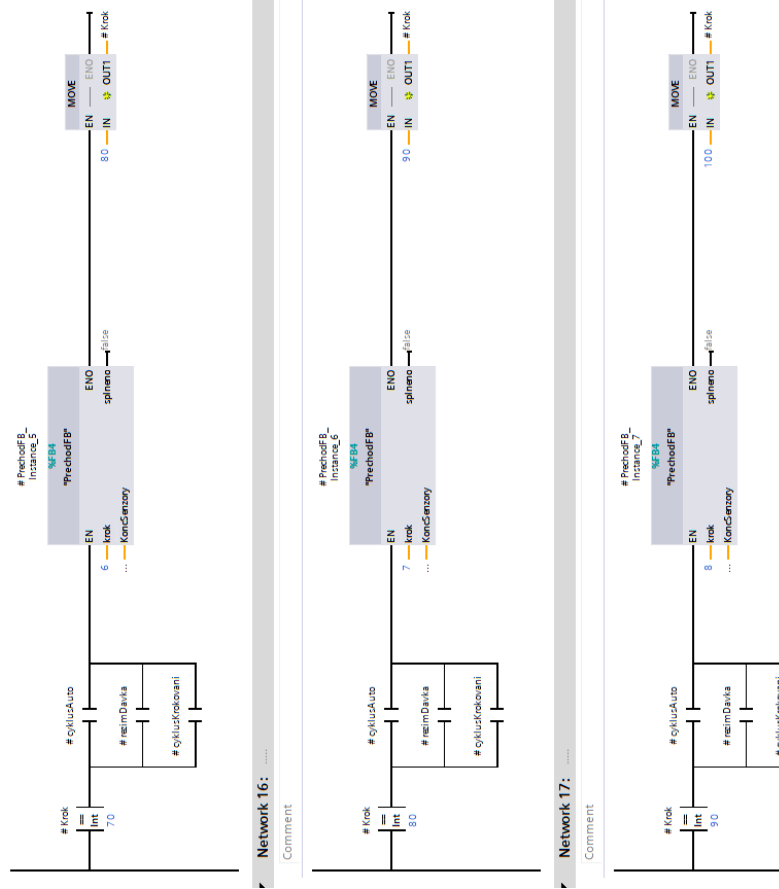
PŘÍLOHA XXVIII VYROBNICYKLUSFB)



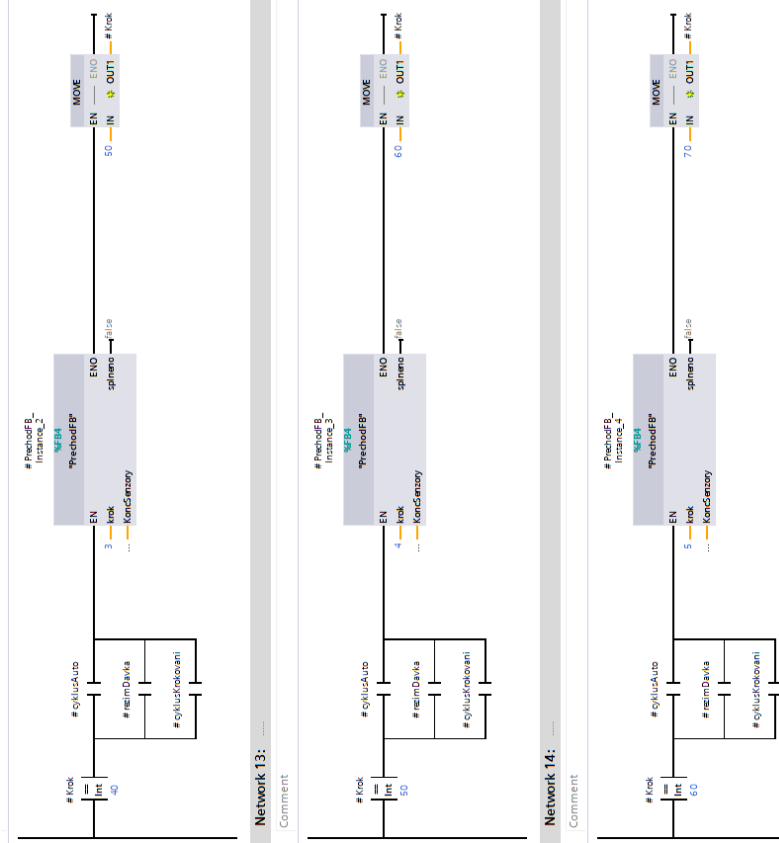
PŘÍLOHA XXIX VYROBNICYKLUSFB



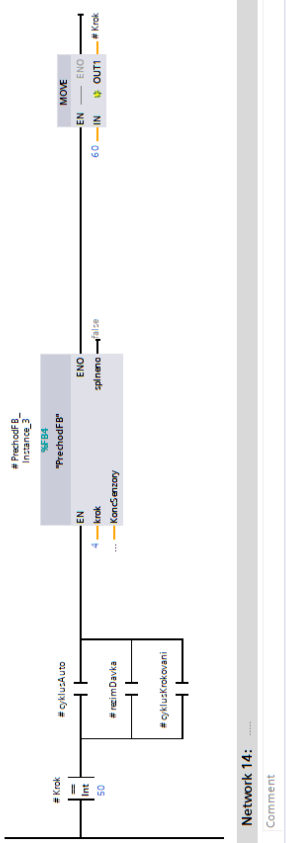
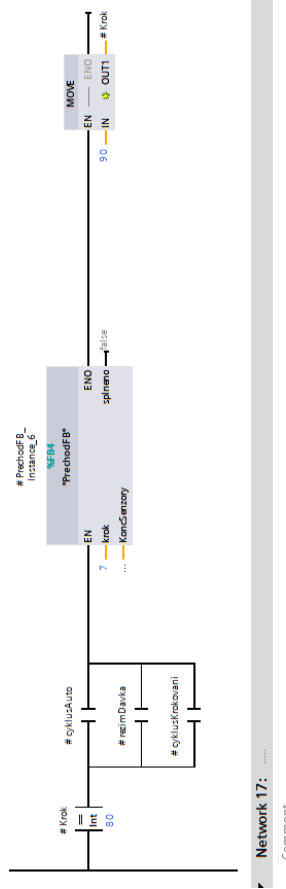
PŘÍLOHA XXX VYROBNICYKLUSFB

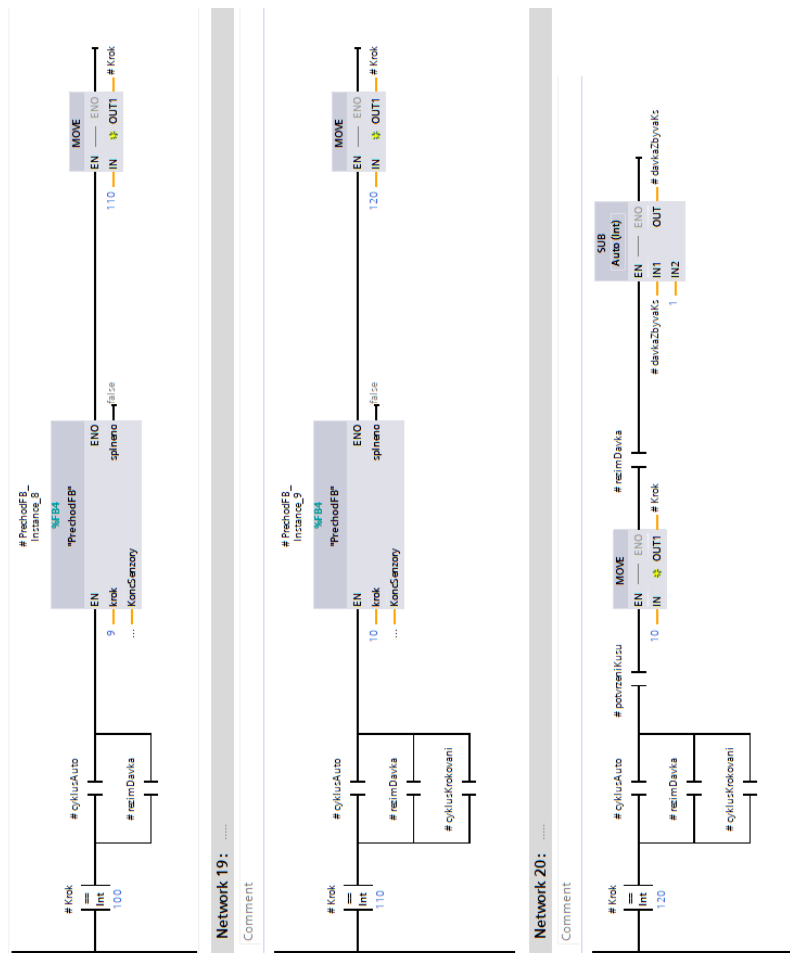


PRÍLOHA XXXII VYROBNICYKLUSFB

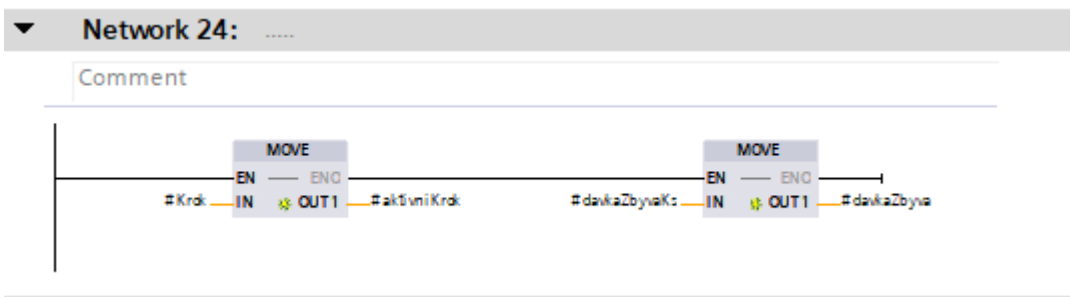


PRÍLOHA XXXI VYROBNICYKLUSFB





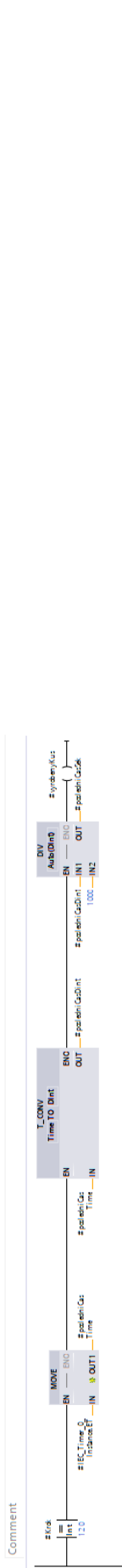
PŘÍLOHA XXXIII VYROBNICYKUSFB



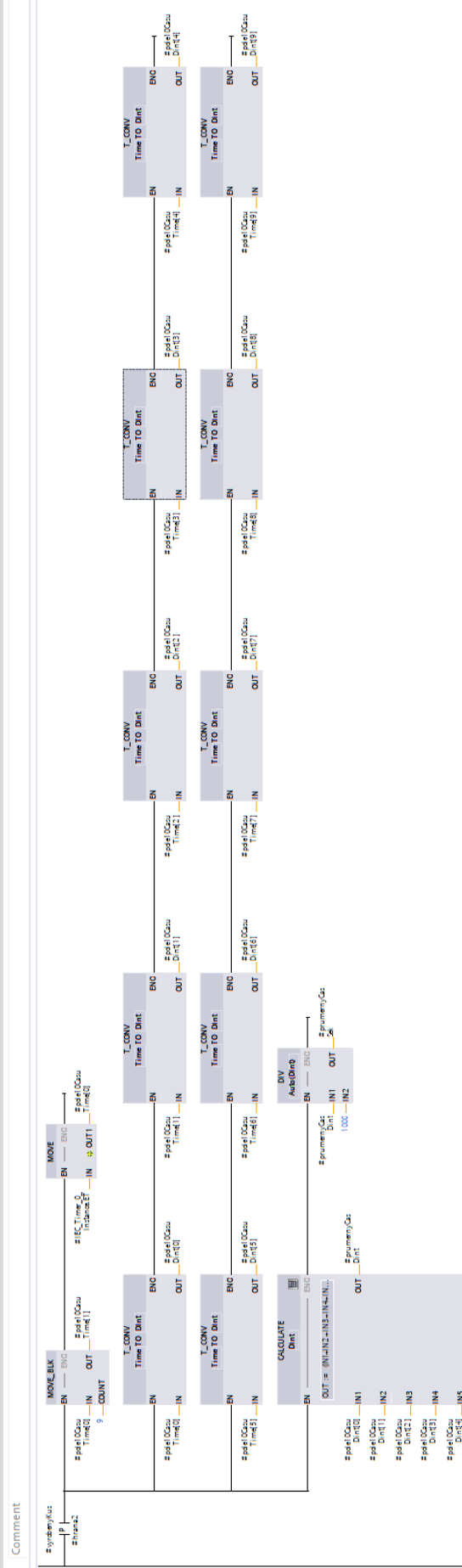
PŘÍLOHA XXXIV VYROBNYCIKLUSFB

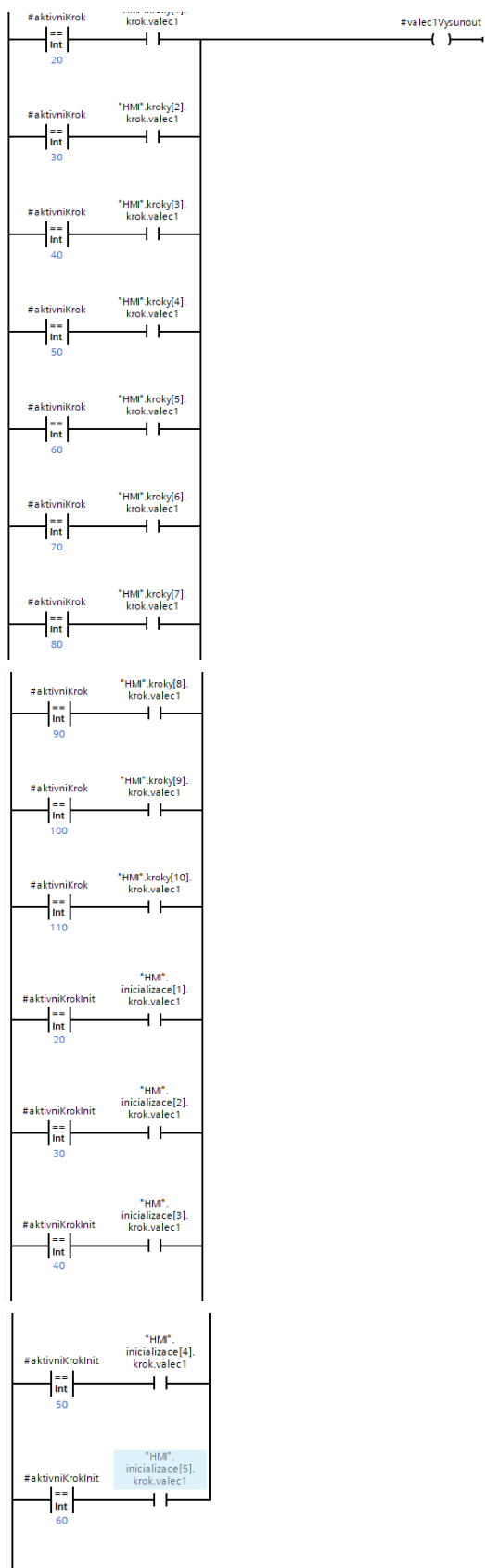


Network 22:



Network 23:





PŘÍLOHA XXXVI VYSTUPNÍFB

```

1 IF ("HMI".zadanyKrok <> "HMI".zvolenyKrok) OR "HMI".nacist THEN
2     "HMI".zadanyKrok := "HMI".zvolenyKrok;
3     "HMI".valec1 := "HMI".kroky["HMI".zadanyKrok].krok.valec1;
4     "HMI".valec2 := "HMI".kroky["HMI".zadanyKrok].krok.valec2;
5     "HMI".valec3 := "HMI".kroky["HMI".zadanyKrok].krok.valec3;
6 END_IF;
7
8 IF "HMI".zadanyKrok > 10 THEN
9     "HMI".zadanyKrok := 10;
10    ;
11 END_IF;
12
13 IF "HMI".zadanyKrok < 1 THEN
14     "HMI".zadanyKrok := 1;
15     ;
16 END_IF;
17
18 IF "HMI".ulozit THEN
19     "HMI".kroky["HMI".zadanyKrok].krok.valec1 := "HMI".valec1;
20     "HMI".kroky["HMI".zadanyKrok].krok.valec2 := "HMI".valec2;
21     "HMI".kroky["HMI".zadanyKrok].krok.valec3 := "HMI".valec3;
22 END_IF;
23
24

```

PŘÍLOHA XXXVII KROKYFB

```

1 IF ("HMI".zadanyInit <> "HMI".zvolenyInit) OR "HMI".nacistInit THEN
2     "HMI".zadanyInit := "HMI".zvolenyInit;
3     "HMI".valec1Init := "HMI".inicializace["HMI".zadanyInit].krok.valec1;
4     "HMI".valec2Init := "HMI".inicializace["HMI".zadanyInit].krok.valec2;
5     "HMI".valec3Init := "HMI".inicializace["HMI".zadanyInit].krok.valec3;
6 END_IF;
7
8 IF "HMI".zadanyInit > 5 THEN
9     "HMI".zadanyInit := 5;
10    ;
11 END_IF;
12
13 IF "HMI".zadanyInit < 1 THEN
14     "HMI".zadanyInit := 1;
15     ;
16 END_IF;
17
18 IF "HMI".ulozitInit THEN
19     "HMI".inicializace["HMI".zadanyInit].krok.valec1 := "HMI".valec1Init;
20     "HMI".inicializace["HMI".zadanyInit].krok.valec2 := "HMI".valec2Init;
21     "HMI".inicializace["HMI".zadanyInit].krok.valec3 := "HMI".valec3Init;
22 END_IF;
23
24

```

PŘÍLOHA XXXVIII KROKYFB