

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**Katedra aplikované elektroniky a telekomunikací**

# **BAKALÁŘSKÁ PRÁCE**

**Testovací jednotka pro programování PLC Siemens  
Simatic S7-1500**

*Originál (kopie) zadání BP/DP*

**Abstrakt**

Tato bakalářská práce se zabývá návrhem a realizací testovací jednotky a tvorbou třech vzorových úloh, na kterých lze funkčnost jednotky demonstrovat. První část práce je zaměřena na popis historie PLC. Další část práce popisuje produkty firmy Siemens, konkrétně PLC Siemens Simatic S7-1500 a vývojové prostředí TIA Portal. Tato část se zaměřuje především na popis základních modulů PLC a na popis programovacích jazyků ve vývojovém prostředí. Zbytek práce je věnován návrhu a realizaci testovací jednotky a třech vzorových úloh. V závěru je shrnuta možnost využití testovací jednotky.

**Klíčová slova**

PLC, TIA Portal, LAD, STL, GRAPH, PLC SIM

**Abstract**

This bachelor thesis deals with the design and implementation of the test unit and the creation of three sample tasks on which the functionality of the unit can be demonstrated. The first part is focused on the history of PLC. The next part of the thesis describes products of Siemens company, namely Siemens Simatic S7-1500 PLC and TIA Portal development environment. This part focuses mainly on the description of basic PLC modules and on the description of programming languages in the development environment. The rest of the work is devoted to the design and implementation of a test unit and three sample tasks. The conclusion summarizes the possibility of using the test unit.

**Key words**

PLC, TIA Portal, LAD, STL, GRAPH, PLC SIM

## **Prohlášení**

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

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

V Plzni dne 11.6.2019

Jiří Hefler

.....

podpis

## **Poděkování**

Tímto děkuji Vyšší odborné škole a Střední průmyslové škole elektrotechnické v Plzni za zapůjčení notebooku s potřebným softwarem pro vypracování této práce.

## Obsah

<b>OBSAH</b> .....	<b>7</b>
<b>ÚVOD</b> .....	<b>9</b>
<b>SEZNAM SYMBOLŮ A ZKRATEK</b> .....	<b>10</b>
<b>SEZNAM OBRÁZKŮ</b> .....	<b>11</b>
<b>SEZNAM TABULEK</b> .....	<b>12</b>
<b>1 POPIS PLC</b> .....	<b>13</b>
1.1 HISTORIE PLC.....	13
1.2 STRUKTURA PLC .....	13
1.2.1 Centrální jednotka (CPU) .....	13
1.2.2 Periferie PLC.....	14
1.2.3 HMI panely.....	15
1.3 DRUHY PLC.....	15
1.3.1 Kompaktní PLC .....	15
1.3.2 Modulární PLC.....	15
<b>2 POPIS LOGICKÉHO AUTOMATU SIEMENS SIMATIC S7-1500</b> .....	<b>16</b>
2.1 MODULY .....	16
2.1.1 I/O moduly.....	16
2.1.2 Technologické moduly.....	16
2.1.3 Komunikační moduly.....	17
2.1.4 Napájecí moduly.....	17
<b>3 TIA PORTAL (TOTALLY INTEGRATED AUTOMATION PORTAL)</b> .....	<b>17</b>
3.1 PROGRAMOVACÍ JAZYKY V TIA PORTAL .....	17
3.1.1 Jazyk reléových schémat (LAD) .....	17
3.1.2 Jazyk funkčních bloků (FBD) .....	18
3.1.3 Sekvenční programovací jazyk (GRAPH).....	18
3.1.4 Jazyk mnemokódů (STL).....	19
3.1.5 Jazyk strukturovaného textu (SCL).....	19
3.2 VYBRANÉ DATOVÉ TYPY V PROSTŘEDÍ TIA PORTAL.....	20
3.2.1 Bool.....	20
3.2.2 INT.....	20
3.2.3 Time.....	20
3.2.4 Time_OF_Day.....	20
3.2.5 Date_and_LTime.....	21
<b>4 NÁVRH A REALIZACE TESTOVACÍ JEDNOTKY</b> .....	<b>21</b>
4.1 NÁVRH TESTOVACÍ JEDNOTKY .....	21
4.1.1 Výpočty potřebné pro návrh jednotky.....	22
4.2 REALIZACE TESTOVACÍ JEDNOTKY .....	23
4.2.1 Cena testovací jednotky.....	23

<b>5</b>	<b>NÁVRH A REALIZACE VZOROVÝCH ÚLOH.....</b>	<b>24</b>
5.1	PRVNÍ ÚLOHA.....	24
5.1.1	Zadání úlohy 1- Řízení kotelny pomocí PLC.....	24
5.1.2	Řešení úlohy 1- Řízení kotelny pomocí PLC.....	25
5.2	DRUHÁ ÚLOHA.....	29
5.2.1	Zadání úlohy 2- Kancelářský kávovar.....	30
5.2.2	Řešení úlohy 2- Kancelářský kávovar.....	31
5.3	TŘETÍ ÚLOHA.....	36
5.3.1	Zadání úlohy 3- Chytrá domácnost.....	37
5.3.2	Řešení úlohy 3- Chytrá domácnost.....	38
<b>6</b>	<b>TESTOVÁNÍ VYTVOŘENÝCH PROGRAMŮ.....</b>	<b>42</b>
<b>7</b>	<b>ZÁVĚR.....</b>	<b>44</b>
	<b>SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ.....</b>	<b>45</b>



## Úvod

Předkládaná práce se zabývá návrhem a realizací testovací jednotky a tvorbou třech vzorových úloh, využitelných společně s jednotkou pro výuku programování PLC na VOŠ a SPŠE Plzeň.

První část práce je věnována popisu PLC Siemens Simatic S7-1500 a vývojového prostředí TIA Portal. Druhá část práce se věnuje návrhu a realizaci testovací jednotky a samotné tvorbě vzorových úloh. Tyto úlohy jsou navrženy a popsány včetně zadání tak, aby sloužily především pro výuku programování PLC.

Důvodem realizace této práce je absence univerzálních výukových přípravků, na kterých by byla možnost otestovat funkčnost programů vytvořených studenty během výuky. Doposud bylo nutné funkčnost programu demonstrovat pomocí HMI panelu, což vedlo v častých případech k nepřehlednosti. Další možností pro ověření funkčnosti programu je využití PLC SIM, kde nastával stejný problém. U obou těchto případů student ale nemá možnost využít rozšiřující moduly PLC. Výukové přípravky, které mají studenti k dispozici, jsou navrženy pouze na konkrétní typy úloh. Testovací jednotka navržená v této práci by měla dosahovat určité univerzálnosti.

## **Seznam symbolů a zkratek**

PLC	Programmable Logic Control. Programovatelný logický automat.
CPU	Central processing unit. Centrální procesorová jednotka.
HMI	Human Machine Interface. Rozhraní mezi člověkem a strojem.
DI	Digital inputs. Digitální vstupy.
DQ	Digital outputs. Digitální výstupy.
AI	Analog inputs. Analogové vstupy.
AQ	Analog outputs. Analogové výstupy.
TIA Portal	Totally Integrated Automation Portal.
STL	Statement List. Programovací jazyk seznam výpisů.
FBD	Function Block Diagram. Programovací jazyk diagram funkčních bloků.
LAD	Ladder Diagram. Programovací jazyk liniových schémat.
SCL	Structured Control Language. Stukturovaný programovací jazyk.
TUV	Teplá užitková voda

## Seznam obrázků

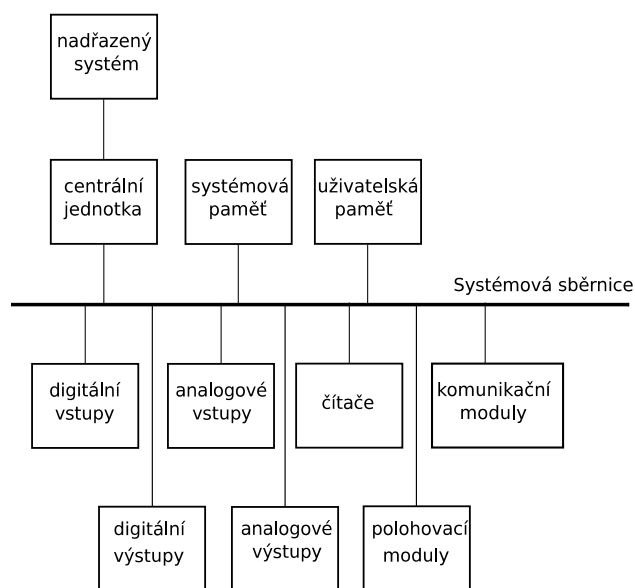
OBRÁZEK 1-STRUKTURA PLC [3] .....	13
OBRÁZEK 2-VYKONÁVÁNÍ PROGRAMU PLC [3] .....	14
OBRÁZEK 3- UKÁZKA LAD KÓDU .....	18
OBRÁZEK 4-UKÁZKA FBD KÓDU .....	18
OBRÁZEK 5-UKÁZKA GRAPH KÓDU .....	19
OBRÁZEK 6-UKÁZKA STL KÓDU .....	19
OBRÁZEK 7-UKÁZKA SCL KÓDU .....	20
OBRÁZEK 8- SCHÉMA ZAPOJENÍ TESTOVACÍ JEDNOTKY .....	22
OBRÁZEK 9- REALIZACE TESTOVACÍ JEDNOTKY PRO PLC .....	23
OBRÁZEK 10-ÚLOHA1- SCHÉMA KOTELNY .....	24
OBRÁZEK 11- ÚLOHA1-FUNKČNÍ BLOKY BLOCK_1_DB A BLOCK_2_DB ZAŘAZENÉ DO HLAVNÍHO PROGRAMU .....	26
OBRÁZEK 12-ÚLOHA 1- OVLÁDÁNÍ OBĚHOVÉHO ČERPADLA .....	26
OBRÁZEK 13-ÚLOHA 1- OVLÁDÁNÍ ŠNEKOVÉHO DOPRAVNÍKU .....	27
OBRÁZEK 14- ÚLOHA 1- BLOCK_1_DB- POKOJOVÁ TEPLOTA.....	27
OBRÁZEK 15- ÚLOHA 1- BLOCK_1_DB- TEPLOTA TUV .....	27
OBRÁZEK 16- ÚLOHA 1- BLOCK_1_DB-TEPLOTA AKUMULAČNÍ NÁDRŽE.....	28
OBRÁZEK 17- ÚLOHA 1- BLOCK_2_DB- INTERVAL SPÍNÁNÍ ŠNEKOVÉHO DOPRAVNÍKU .....	28
OBRÁZEK 18- ÚLOHA 1- BLOCK_2_DB- INTERVAL SPÍNÁNÍ ŠNEKOVÉHO DOPRAVNÍKU .....	29
OBRÁZEK 19- ÚLOHA 1- VIZUALIZACE NA HMI PANELU .....	29
OBRÁZEK 20- ÚLOHA 1- SCHÉMA KÁVOVARU.....	30
OBRÁZEK 21-ÚLOHA 2- HLAVNÍ PROGRAM- ZVOLENÍ CAPPUCCINA .....	32
OBRÁZEK 22- ÚLOHA 2- HLAVNÍ PROGRAM- VÝROBA CAPPUCCINA.....	32
OBRÁZEK 23- ÚLOHA 2- HLAVNÍ PROGRAM- POČÍTÁNÍ NÁPOJŮ PRO SPUŠTĚNÍ PROPLACHU .	33
OBRÁZEK 24- ÚLOHA 2- HLAVNÍ PROGRAM- REALIZACE PROPLACHU.....	34
OBRÁZEK 25- ÚLOHA 2- HLAVNÍ PROGRAM- INDIKACE STAVU NÁPOJE.....	35
OBRÁZEK 26- ÚLOHA 2- FUNKČNÍ BLOK MLÉKO- UKÁZKA PRVNÍHO KROKU A PRVNÍHO PŘECHODU .....	35
OBRÁZEK 27- ÚLOHA 2- FUNKČNÍ BLOK BLOCK3- MNOŽSTVÍ CUKRU A MNOŽSTVÍ NÁPOJE V KELÍMKU .....	36
OBRÁZEK 28- ÚLOHA 2- VIZUALIZACE NA HMI PANELU.....	36
OBRÁZEK 29- ÚLOHA 3- HLAVNÍ PROGRAM- NAČTENÍ AKTUÁLNÍHO ČASU DO PLC .....	39
OBRÁZEK 30- ÚLOHA 3- HLAVNÍ PROGRAM- VLOŽENÍ FUNKČNÍCH BLOKŮ .....	39
OBRÁZEK 31- ÚLOHA 3- HLAVNÍ PROGRAM- SPOUŠTĚNÍ SVĚTLA.....	39
OBRÁZEK 32- ÚLOHA 3- ŽALUZIE_SVĚTLA_TOPENÍ- TESTOVÁNÍ STISKU TLAČÍTKA .....	40
OBRÁZEK 33- ÚLOHA 3- REŽIMY- NASTAVENÍ POŽADOVANÝCH TEPLOT V ZÁVISLOSTI NA REŽIMU .....	40
OBRÁZEK 34- ÚLOHA 3- ŽALUZIE_SVĚTLA_TOPENÍ- OVLÁDÁNÍ ŽALUZIÍ A TOPENÍ.....	41
OBRÁZEK 35- ÚLOHA 3- VIZUALIZACE NA HMI PANELU .....	41
OBRÁZEK 36- PLC SIM- KONFIGURACE PLC .....	42
OBRÁZEK 37- PLC SIM- TABULKA OVLÁDANÝCH PROMĚNNÝCH .....	43
OBRÁZEK 38- TIA PORTAL- ONLINE SLEDOVÁNÍ LOGICKÉ ÚROVNĚ SIGNÁLU .....	43

## Seznam tabulek

TABULKA 1- SEZNAM SOUČÁSTEK VYUŽITÝCH PŘI VÝROBĚ TESTOVACÍ JEDNOTKY .....	23
TABULKA 2- ÚLOHA 1- ZÁVISLOST DOBY CHODU PODAVAČE NA TEPLOTĚ AKUMULAČNÍ NÁDRŽE.....	25
TABULKA 3- ÚLOHA 1- SEZNAM VSTUPNÍCH A VÝSTUPNÍCH PROMĚNNÝCH DO PLC .....	25
TABULKA 4- ÚLOHA 2- NASTAVENÍ VENTILU CUKRU .....	30
TABULKA 5- ÚLOHA 2- SEZNAM VSTUPNÍCH A VÝSTUPNÍCH PROMĚNNÝCH DO PLC.....	31
TABULKA 6- ÚLOHA 3- OVLÁDÁNÍ ŽALUZIE V ZÁVISLOTI NA TEPLOTĚ.....	37
TABULKA 7- ÚLOHA 3- SEZNAM VSTUPNÍCH A VÝSTUPNÍCH PROMĚNNÝCH PLC.....	38

# 1 Popis PLC

PLC (Programmable Logic Controller), česky nazývaný programovatelný logický automat, můžeme chápat jako počítač určený pro průmysl. Je řízený mikroprocesorem a běží na vlastním operačním systému, který je uzpůsoben pro potřeby automatizace. Především se jedná o rychlost odezvy.



Obrázek 1-Struktura PLC [3]

## 1.1 Historie PLC

Před nástupem PLC se pro řízení strojů používaly reléové skříně. S narůstajícími nároky na automatizaci výroby se však toto řešení ukázalo jako nevhodné, a to především z důvodu složitosti a nedostatečné spolehlivosti. Problém vyřešil nástup PLC automatů, které reléové skříně nahradily. Proto existuje jazyk reléových schémat, který připomíná právě reléovou logiku. [1]

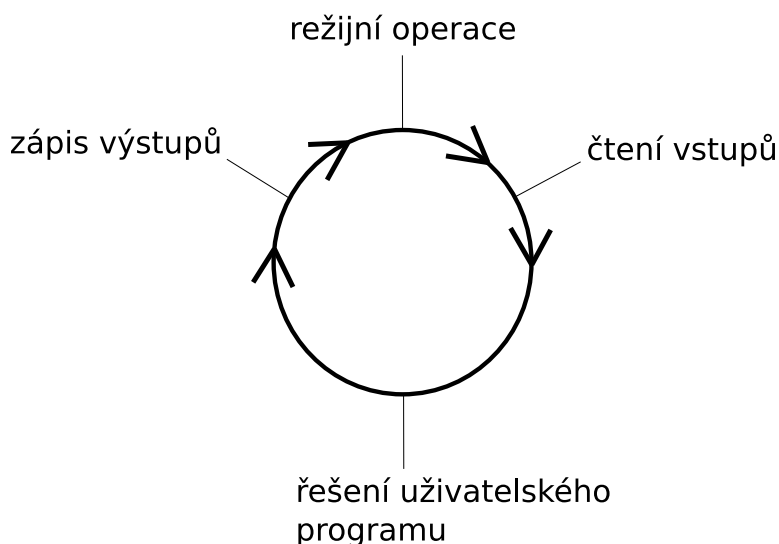
## 1.2 Struktura PLC

### 1.2.1 Centrální jednotka (CPU)

V centrální jednotce běží operační systém pro chod PLC. Ten zajišťuje komunikaci s vlastními či vzdálenými moduly a s nadřazeným systémem. Zpracovává hlavní uživatelský program, načítá vstupní periferie a převádí hodnoty na výstupní periferie. CPU také provádí datovou výměnu mezi programem, systémem a paměťmi a v neposlední řadě

umožňuje komunikaci s programátorem přes programové rozhraní. Mezi hlavní komponenty centrální jednotky patří mikroprocesor, paměť pro uložení programu a uživatelská paměť pro data. Dále CPU obsahuje rozhraní pro komunikaci s programátorem a budič sběrnice pro připojení periférií. [2]

PLC program běží v cyklu. Jeho běh zajišťuje systémový program. Po vykonání poslední instrukce uživatelského programu systémový program provede otáčku cyklu. V ní nejprve aktualizuje hodnoty výstupů a vstupů. Výstupní hodnoty jsou přepsány z paměti do výstupních periférií. Následně jsou provedeny režijní operace systému, aktualizace časových údajů pro časovače a systémové registry. Poté jsou načteny vstupní hodnoty ze vstupních periférií. Po provedení otáčky se opět spustí uživatelský program od první instrukce. [3]



Obrázek 2-Vykonávání programu PLC [3]

### 1.2.2 Periferie PLC

Periferie PLC se používají pro komunikaci s okolím. Lze je rozdělit podle směru toku dat na vstupní a výstupní. Další možné dělení je na digitální a analogové, centrální a decentrální.

Digitální periferie pracují s dvoustavovou hodnotou, tedy vypnuto a zapnuto, často na 24V logice. Tento stav sledují na vstupních perifériích a vytvářejí ho na výstupních

periferiích. Analogové periferie pracují se spojitým signálem. Typickým příkladem může být práce s teplotním čidlem na vstupní periférii. Mezi centrální periferie můžeme zařadit takové, které jsou umístěny v centrálním systému (např. v rozvodné skříni). Decentrální periferie jsou takové, které se nacházejí mimo centrální systém (např. frekvenční měnič umístěný přímo na motoru stroje).

### **1.2.3 HMI panely**

HMI panely slouží pro komunikaci s obsluhou řízeného systému. Nejčastěji se v současné době objevují v podobě dotykového displeje v provedení LCD či TFT s vlastním procesorem a flash pamětí. Lze pomocí nich například přepínat mezi dvěma režimy chodu zařízení (např. dva režimy výroby), zadávat nějaké referenční hodnoty, zobrazovat aktivitu digitálních výstupů, nebo zobrazovat varovné či chybové stavy.

## **1.3 Druhy PLC**

### **1.3.1 Kompaktní PLC**

Kompaktní PLC je druh PLC kdy je CPU umístěno do jednoho modulu, včetně jeho napájení, digitálních vstupů a výstupů a analogových vstupů a výstupů. Počet jednotlivých vstupů a výstupů určuje výrobce. Kompaktní PLC jsou výhodné z hlediska ceny a rychlosti komunikace mezi CPU a periferiemi. Jsou však limitovány svou použitelností, protože mají omezený počet periferií. Kompaktní PLC se vyrábí i v mikro verzi, která přináší ještě nižší cenu. Často však mají pouze 6 digitálních vstupů a 6 výstupů. Mikro PLC ale nelze dodatečně rozšiřovat oproti klasickému kompaktnímu provedení.

### **1.3.2 Modulární PLC**

Modulární PLC je druh PLC, který je složen z jednotlivých modulů. Jeden z nich musí být CPU, ke kterému je nutné ostatní připojit. Výhodou tohoto PLC je jednoznačně možnost konfigurace.

## 2 Popis logického automatu Siemens Simatic S7-1500

Tato řada PLC od firmy Siemens je vhodná pro řízení strojů střední velikosti i pro řízení náročných aplikací. Jedná se o modulární typ PLC, kdy lze k jednomu CPU připojit až 32 modulů. Obsahuje velmi výkonnou systémovou sběrnici a rozhraní PROFINET. Tato kombinace umožňuje, společně s velmi rychlou odezvou, zpracování příkazu v rámci jednotek ns.[4]

Integrované diagnostické funkce a jednotný systém konceptu zobrazení umožňují zobrazení chybových hlášení na HMI panelu, webovém serveru a displeji CPU. Konfigurace diagnostických funkcí a sledování proměnných a jejich stavů napomáhají ke snížení prostojů v případě závady.[4]

Simatic S7-1500 disponuje integrovanými funkcemi, které mohou chránit program před okopírováním nebo přístupem bez oprávnění. Sériové číslo paměťové karty SIMATIC se přiřadí jednotlivým blokům řídicího programu. V případě změn dat, nebo přidání dat ze zdroje, který není povolen, funkce zajišťující ochranu změnu detekují.[4]

### 2.1 Moduly

#### 2.1.1 I/O moduly

Digitální a analogové moduly poskytují přesně vstupy a výstupy, které jsou potřebné pro danou úlohu. K dispozici jsou moduly 25 mm a 35 mm. Analogové moduly mají rozšířené funkce, jako je převzorkování, nastavení měřicího rozsahu nebo škálování naměřených hodnot.[4]

#### 2.1.2 Technologické moduly

Technologické moduly nabízejí hardwarové zpracování signálů pro rychlé počítání, měření a registraci polohy pro různé snímače, stejně jako pro přesný výstup pulzů. Technologické funkce, jako je řízení pohybu a vysokorychlostní čítač, jsou již integrovány do procesorů SIMATIC S7-1500 a lze je naplánovat pomocí STEP 7 (TIA Portal).[4]



### 2.1.3 Komunikační moduly

Rozhraní pro komunikaci přes PROFINET a PROFIBUS jsou již integrována do CPU. Komunikační moduly zvyšují komunikační schopnosti S7-1500 prostřednictvím dalších komunikačních funkcí nebo více rozhraní. Komunikační moduly zvyšují flexibilitu a výkon automatizačních řešení pomocí SIMATIC S7-1500 / ET 200MP. Další rozhraní umožňují realizaci komplexních automatizačních struktur nebo optimalizaci procesů prostřednictvím propojení s úrovní podnikového managementu.[4]

Existují tři typy komunikačních modulů pro S7-1500.

- Komunikační modul pro připojení k průmyslovému ethernetu.
- Komunikační modul pro připojení k PROFIBUS.
- Komunikační moduly pro výměnu dat pomocí point-to-point spojení.

### 2.1.4 Napájecí moduly

Jsou to 24V napájecí zdroje pro CPU a ostatní moduly. Vyrábí se v hodnotách výkonu 25W a 60W.

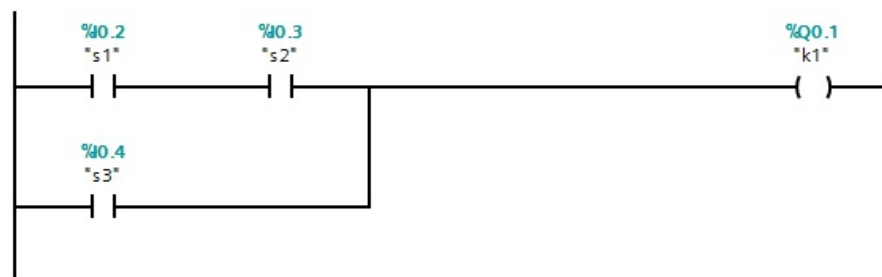
## 3 TIA Portal (Totally Integrated Automation Portal)

TIA Portal je prostředí, které obsahuje kompletní prostředky pro automatizaci a digitalizaci výrobních procesů. To zahrnuje návrh, plánování a projektování. TIA Portal nabízí nástroje pro kompletní simulaci, diagnostiku a monitoring spotřeby. Společně s flexibilnější komunikací, která umožňuje propojení s nadřazenými systémy, TIA Portal umožňuje docílit vyšší produktivity celého závodu. [4]

### 3.1 Programovací jazyky v TIA Portal

#### 3.1.1 Jazyk reléových schémat (LAD)

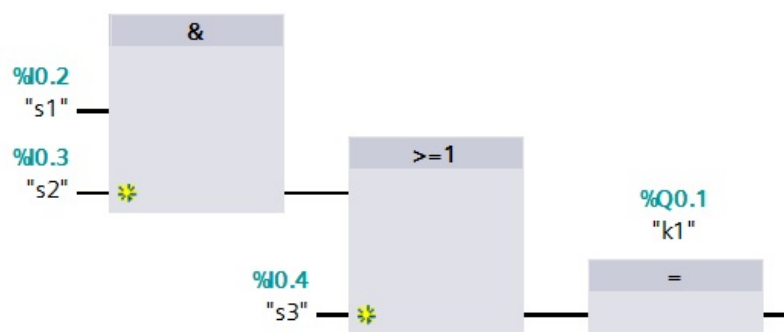
Jedná se o grafický programovací jazyk, který je nejvíce podobný klasickému kreslení schémat při práci s kontaktními prvky. LAD se hodí pro programování jednodušších logických operací. Je přehledný a snadno pochopitelný.



Obrázek 3- Ukázka LAD kódu

### 3.1.2 Jazyk funkčních bloků (FBD)

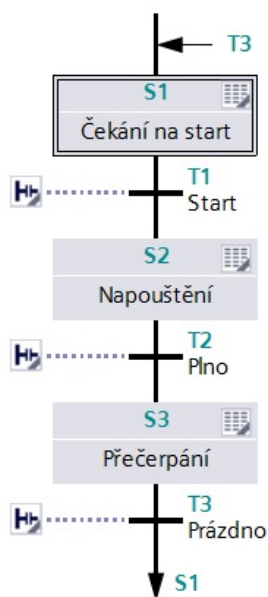
Jedná se o grafický programovací jazyk, který se podobá kreslení logických schémat pro zařízení s integrovanými obvody. Popisuje funkci mezi vstupy a výstupy. FBD se hodí pro podobné úlohy jako LAD.



Obrázek 4-Ukázka FBD kódu

### 3.1.3 Sekvenční programovací jazyk (GRAPH)

Jazyk sekvenčního programování se hodí především pro řešení sekvenčních úloh. Skládá se z jednotlivých kroků programu a přechodů mezi nimi. Kroky programu se popisují v jazyku STL popsáném níže. Přechody mezi kroky se popisují jazykem LAD. Jazyk GRAPH je velmi snadno čitelný a napomáhá k eliminaci chaoticky uspořádaných programů.



Obrázek 5-Ukázka GRAPH kódu

### 3.1.4 Jazyk mnemokódů (STL)

Jedná se o nejdéle používaný jazyk pro programování PLC, kterému se někdy říká jazyk povelů. Tento jazyk bývá často přirovnáván k assembleru. Skládá se ze seznamu instrukcí, z nichž každá začíná na svém řádku.[6]

1	O	"s3"	%I0.4
2	O(		
3	A	"s1"	%I0.2
4	A	"s2"	%I0.3
5	)		
6	=	"k1"	%Q0.1
7			
8			

Obrázek 6-Ukázka STL kódu

### 3.1.5 Jazyk strukturovaného textu (SCL)

Svojí strukturou může připomínat vyšší programovací jazyky pro PC (Pascal, C). Umožňuje velice snadno zapsat požadovaný algoritmus. Proto je vhodný pro popsání funkčních bloků, které mohou být následně použity v kterémkoliv z ostatních programovacích jazyků.

```
1 //pokojová teplota
2
3 "Realna_teplota_pokoje" := "Pokojoiva_teplota" / 10;
4
5 IF "Realna_teplota_pokoje" <= ("Pozadovana_teplota" - 2.0)
6 THEN
7     "start_obeh" := 1;
8 END_IF;
9
10 IF "Realna_teplota_pokoje" >= "Pozadovana_teplota"
11 THEN
12     "start_obeh" := 0;
13 END_IF;
```

Obrázek 7-Ukázka SCL kódu

## 3.2 Vybrané datové typy v prostředí TIA Portal

V této kapitole jsou popsány datové typy, které jsem využil při vypracování řešení úloh.

### 3.2.1 Bool

Proměnná datového typu Bool, reprezentuje hodnotu bitu a může nabývat dvou hodnot, True nebo False, tedy logické 1 nebo 0,

### 3.2.2 INT

Proměnná datového typu INT má délku 16 bitů. Bit na 15 pozici určuje, zda je číslo kladné (log. 0) či záporné (log. 1). Proměnná typu INT může nabývat celočíselných hodnot v rozsahu -32768 až 32767.

### 3.2.3 Time

Obsah proměnné datového typu Time udává počet ms. Má délku 32 bitů a jeho hodnota může nabývat hodnot v rozsahu -24d20h31m23s648ms až 24d20h31m23s647ms. V paměti zabírá 4 byty.

### 3.2.4 Time\_OF\_Day

Datový typ TOD udává počet ms od začátku dne, tedy od (0:00 h). Může nabývat hodnot v rozsahu 00:00:00.000 až 23:59:59.999. V paměti zabírá 4 byty.

### 3.2.5 Date\_and\_LTime

Datový typ LDT ukládá informace o datu a denní době v nanosekundách od 01.01.1970 00:00. Maximální hodnota, které může nabývat je 04.11.2263 23:47:16.854775808. V paměti zabírá 8 bytů.

## 4 Návrh a realizace testovací jednotky

Prvním cílem této práce je navrhnout a realizovat testovací jednotku pro programování PLC. Jednotka má být vybavena 8 digitálními vstupy, 8 digitálními výstupy, 2 analogovými vstupy, 2 analogovými výstupy a vhodnou indikací logické a analogové úrovně. Jednotka má sloužit jako pomůcka pro výuku programování PLC na VOŠ a SPŠE Plzeň. Je tedy kladen velký důraz na robustnost jednotky, která bude v častém kontaktu se studenty. Sledovaným parametrem je také cena testovací jednotky. Pokud se její užívání během výuky osvědčí, dojde k výrobě dalších.

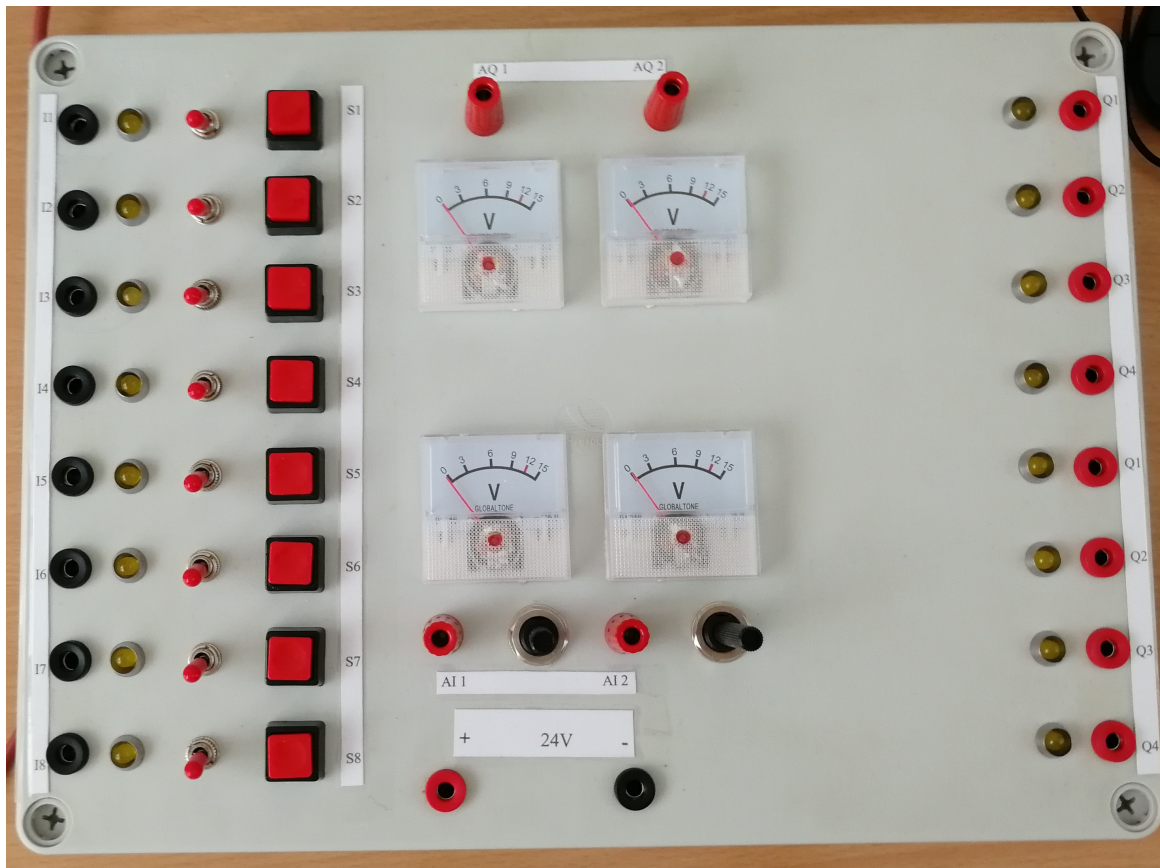
### 4.1 Návrh testovací jednotky

Jako první bylo zapotřebí ujasnit si, jak bude jednotka napájena. Zvolil jsem možnost napájení z napájecího modulu PLC (24V/60W). DI a DQ moduly PLC pracují s 24V logikou. Pro spouštění digitálních výstupů z jednotky je použita paralelní kombinace tlačítka a pákového spínače, která umožní studentům přivést do PLC jak trvalou logickou úroveň, tak jednotkový pulz. Indikace logických úrovní na vstupu i výstupu jednotky je realizována LED diodou s předřadným odporem. AI moduly PLC mohou pracovat ve 2 pro nás využitelných režimech, měření proudu a měření napětí. Já jsem vybral měření napětí v rozsahu -10V až 10V. Pro vytvoření analogové úrovně 0 až 10V je použitý potenciometr. Jako indikace analogové úrovně pro vstupy i výstupy z jednotky je použitý voltmetr.



## 4.2 Realizace testovací jednotky

Pro realizaci jednotky jsem zvolil krabici o rozměrech 300x220x120mm, která poskytuje dostatečné rozměry pro umístění všech komponentů a případné rozšíření.



Obrázek 9- Realizace testovací jednotky pro PLC

### 4.2.1 Cena testovací jednotky

Tabulka 1- Seznam součástek využitých při výrobě testovací jednotky

<b>Součástka</b>	<b>Počet kusů</b>	<b>Cena</b>
<i>Krabice IP65 S-BOX 616</i>	<i>1</i>	<i>377 Kč</i>
<i>Panelový voltmetr 15V</i>	<i>4</i>	<i>220 Kč</i>
<i>Reflektor k LED 5 mm</i>	<i>16</i>	<i>80 Kč</i>
<i>Žlutá LED 5mm</i>	<i>16</i>	<i>51,2 Kč</i>
<i>Tlačítko 250V/1A</i>	<i>8</i>	<i>128 Kč</i>
<i>Zdířka vestavná 4mm</i>	<i>4</i>	<i>276 Kč</i>
<i>Zdířka panelová 4mm</i>	<i>18</i>	<i>270 Kč</i>
<i>Rezistor 1K1 0,6W</i>	<i>16</i>	<i>41,6 Kč</i>
<i>Rezistor 1K5 0,6W</i>	<i>2</i>	<i>5,2 Kč</i>
<i>Potenciometr 1K lineární</i>	<i>2</i>	<i>62 Kč</i>
<b>Celková cena</b>		<b>1511 Kč</b>

## 5 Návrh a realizace vzorových úloh

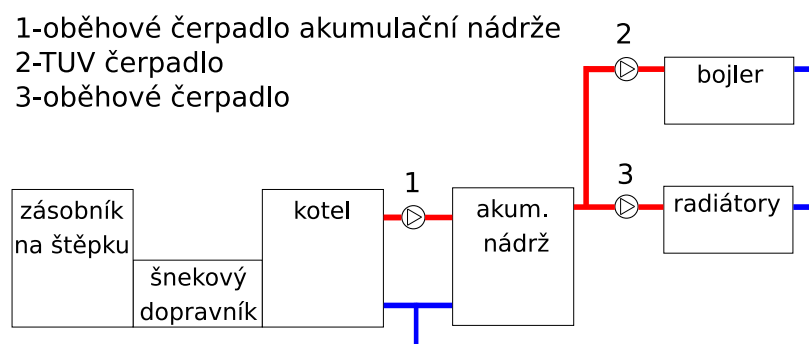
Druhým cílem této práce je navrhnout a realizovat tři vzorové úlohy, na kterých lze demonstrovat funkčnost jednotky. Při realizaci úloh je zapotřebí využít různé programovací jazyky.

### 5.1 První úloha

Jako první úlohu jsem navrhl řízení kotelny pomocí PLC. Cílem této úlohy je, aby si studenti dokázali lépe představit využití PLC v praxi. Úloha je zaměřena spíše na programovací jazyk LAD, lze ji však řešit i jinými způsoby.

#### 5.1.1 Zadání úlohy 1- Řízení kotelny pomocí PLC

Navrhněte program pro řízení kotelny pomocí PLC podle následujících požadavků.



Obrázek 10-Úloha1- Schéma kotelny

V kotelně je použitý kotel se zásobníkem na dřevní štěpku, která je do kotle podávána šnekovým dopravníkem. Ten je spínán v pravidelných intervalech na určitou dobu v závislosti na teplotě akumulční nádrže. Interval chodu podavače bude po spuštění PLC nastaven na 60s. Jeho hodnotu lze změnit pomocí HMI panelu. Doba chodu podavače je určena podle tabulky 1.



Tabulka 2-Úloha 1- Závislost doby chodu podavače na teplotě akumuláční nádrže

<i>Teplota akumuláční nádrže</i>	<i>Doba chodu podavače</i>
<i>&lt;70°C;80°C)</i>	<i>10s</i>
<i>(50°C;70°C)</i>	<i>20s</i>
<i>&lt;50°C</i>	<i>30s</i>

Pokud bude teplota akumuláční nádrže vyšší než 80°C, doba chodu podavače bude 0s. Oběhové čerpadlo akumuláční nádrže poběží, pokud bude teplota akumuláční nádrže menší než 80°C. Teplotu akumuláční nádrže simulujte pomocí analogového výstupu jednotky.

TUV čerpadlo se zapne, pokud bude teplota bojleru pod 50°C a vypne, pokud bude teplota bojleru nad 55°C. Teplotu bojleru simulujte pomocí analogového výstupu jednotky.

Oběhové čerpadlo poběží, pouze pokud bude teplota pokoje nižší nebo rovna požadované teplotě s hysterezí 2°C. Při spuštění PLC bude požadovaná teplota nastavena na 24°C. Následně ji lze nastavit pomocí HMI panelu. Pro snímání pokojové teploty použijte teplotní čidlo Pt-100.

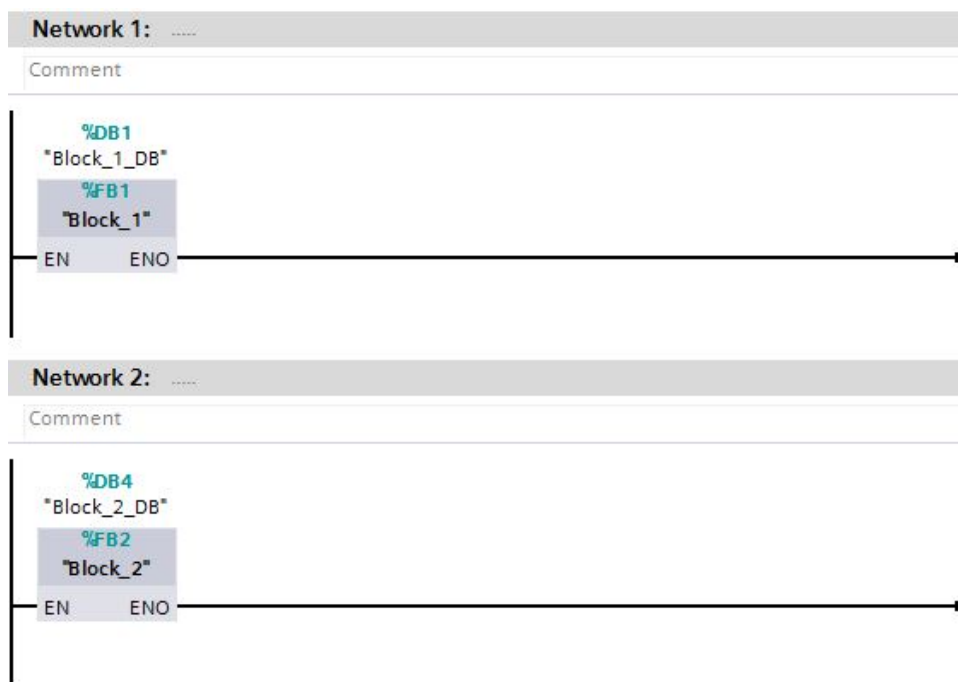
Program bude disponovat servisním režimem, kdy pomocí HMI panelu půjde natvrdo zapnout TUV čerpadlo, oběhové čerpadlo, oběhové čerpadlo akumuláční nádrže a šnekový dopravník.

Tabulka 3-Úloha 1- Seznam vstupních a výstupních proměnných do PLC

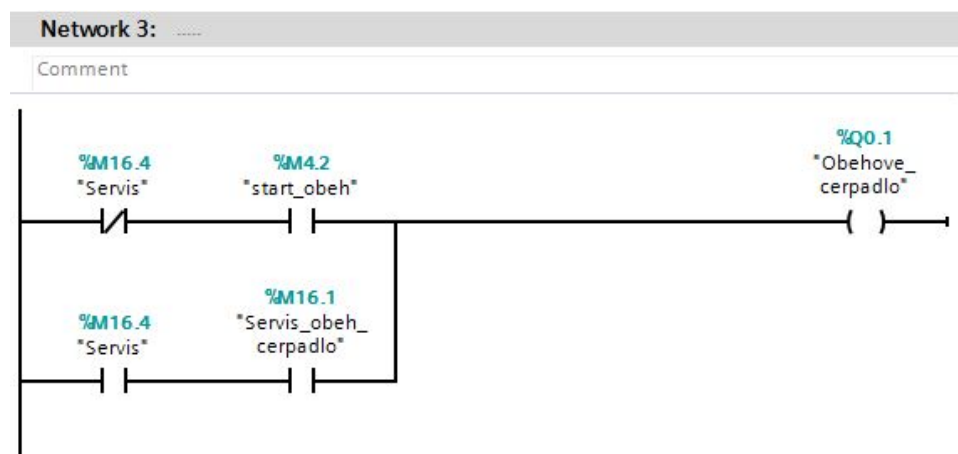
<i>Vstupní proměnné</i>	<i>Výstupní proměnné</i>
<i>Pokojová teplota (AI)</i>	<i>Oběhové čerpadlo (DQ)</i>
<i>Teplota bojleru (AI)</i>	<i>TUV čerpadlo (DQ)</i>
<i>Teplota akumuláční nádrže (AI)</i>	<i>Čerpadlo akumuláční nádrže (DQ)</i>
	<i>Šnekový dopravník (DQ)</i>

### 5.1.2 Řešení úlohy 1- Řízení kotelny pomocí PLC

Následující obrázky popisují hlavní program PLC, který je napsaný v LAD jazyku. Na obrázku 11 můžeme vidět zařazení dvou funkčních bloků do hlavního programu.

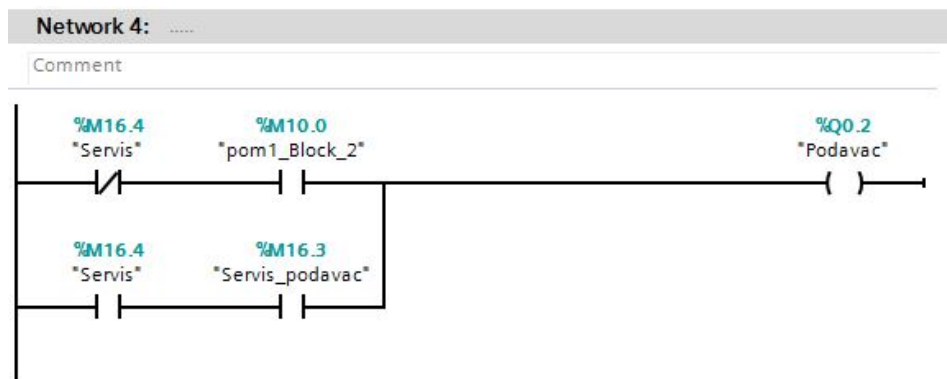


Obrázek 11- Úloha1-Funkční bloky Block\_1\_DB a Block\_2\_DB zařazené do hlavního programu



Obrázek 12-Úloha 1- ovládání oběhového čerpadla

Obrázek 12 popisuje ovládání oběhového čerpadla, které lze spustit v servisním režimu příslušným tlačítkem na HMI, nebo je spouštěno automaticky v závislosti na pokojové teplotě. Ovládání ostatních čerpadel je realizováno stejným způsobem.



Obrázek 13-Úloha 1- ovládání šnekového dopravníku

Obrázek 13 popisuje ovládání šnekového dopravníku, které lze spustit v servisním režimu příslušným tlačítkem na HMI, nebo je spouštěn v pravidelných intervalech v závislosti na teplotě akumulční nádrže. Následující obrázky 14, 15, 16 popisují funkční blok Block\_1\_DB, který je napsaný v SCL jazyku. Cílem tohoto bloku je obsloužit spínání oběhových čerpadel a šnekového dopravníku v závislosti na teplotě.

```

1 //pokojová teplota
2
3 "Realna_teplota_pokoje" := "Pokojoiva_teplota" / 10;
4
5 IF "Realna_teplota_pokoje" <= ("Pozadovana_teplota" - 2.0)
6 THEN
7     "start_obeh" := 1;
8 END_IF;
9
10 IF "Realna_teplota_pokoje" >= "Pozadovana_teplota"
11 THEN
12     "start_obeh" := 0;
13 END IF;

```

Obrázek 14- Úloha 1- Block\_1\_DB- pokojová teplota

```

15 //teplota TUV
16
17 "Realna_teplota_TUV" := "Teplota_bojleru" * 10;
18
19 IF ("Realna_teplota_TUV" < 50.0)
20 THEN
21     "start_TUV" := 1;
22 END_IF;
23
24 IF ("Realna_teplota_TUV" > 55.0)
25 THEN
26     "start_TUV" := 0;
27 END_IF;
28

```

Obrázek 15- Úloha 1- Block\_1\_DB- teplota TUV

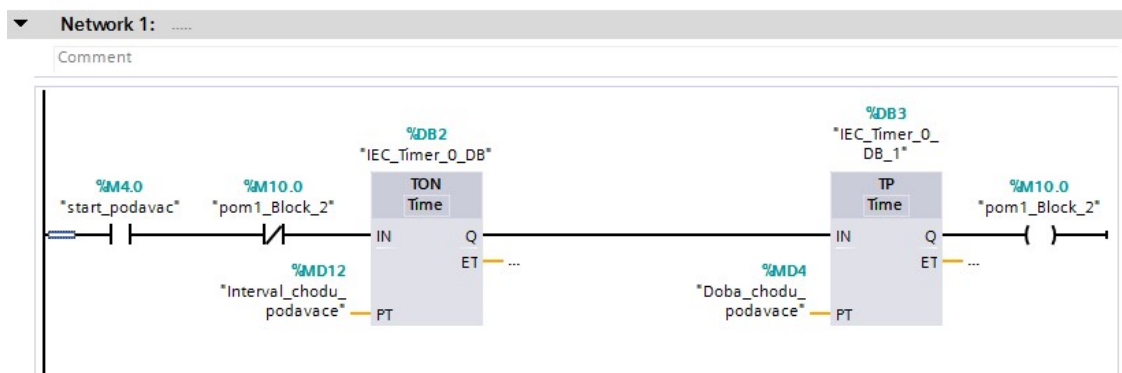
```

29 //teplota AKUM NÁDRŽE
30
31
32 "Realna_teplota_akum_nadrze" := "Teplota_akum_nadrze" * 10;
33
34 IF ("Realna_teplota_akum_nadrze" >= 70.0) AND ("Realna_teplota_akum_nadrze" < 80.0)
35 THEN
36     "start_podavac" := 1;
37     "start_akum" := 1;
38     "Doba_chodu_podavace" := T#10s;
39 ELSIF ("Realna_teplota_akum_nadrze" > 50.0) AND ("Realna_teplota_akum_nadrze" < 70.0)
40 THEN
41     "start_podavac" := 1;
42     "start_akum" := 1;
43     "Doba_chodu_podavace" := T#20s;
44 ELSIF ("Realna_teplota_akum_nadrze" < 50.0)
45 THEN
46     "start_podavac" := 1;
47     "start_akum" := 1;
48     "Doba_chodu_podavace" := T#30s;
49 ELSE
50     "start_podavac" := 0;
51     "start_akum" := 0;
52 END IF:

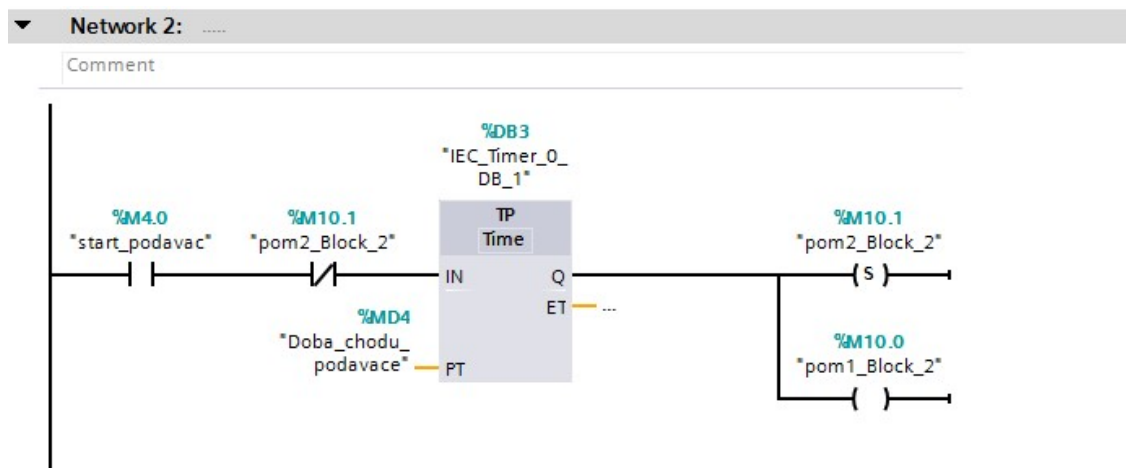
```

Obrázek 16- Úloha 1- Block\_1\_DB-teplota akumulární nádrže

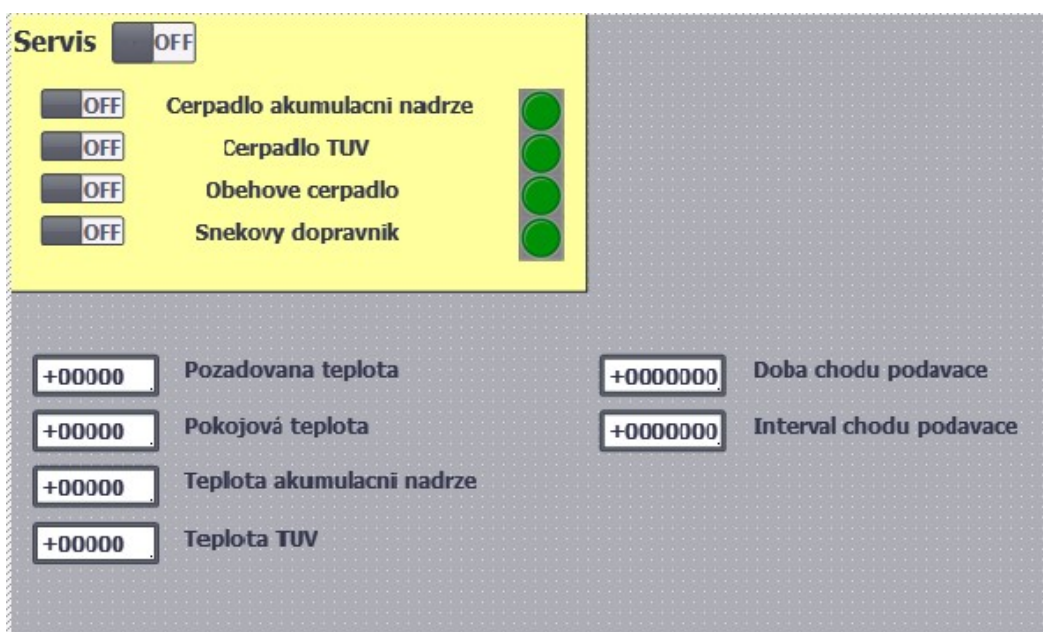
Obrázky 17 a 18 popisují funkční blok Block\_2\_DB, který je naspaný v LAD jazyku. Cílem tohoto bloku je zajištění chodu šnekového dopravníku v pravidelných intervalech.



Obrázek 17- Úloha 1- Block\_2\_DB- interval spínání šnekového dopravníku



Obrázek 18- Úloha 1- Block\_2\_DB- interval spínání šnekového dopravníku



Obrázek 19- Úloha 1- Vizualizace na HMI panelu

Obrázek 19 ukazuje vizualizaci na HMI panelu k ovládání kotelny, kde lze spustit servisní režim a zadávat potřebné hodnoty.

## 5.2 Druhá úloha

Jako druhou úlohu jsem navrhl programování kancelářského kávovaru. Tento způsob použití PLC není úplně typický, ale jako příklad pro výukové potřeby je snadno pochopitelný. Hlavním cílem této úlohy je naučit studenty používat programovací jazyk GRAPH.

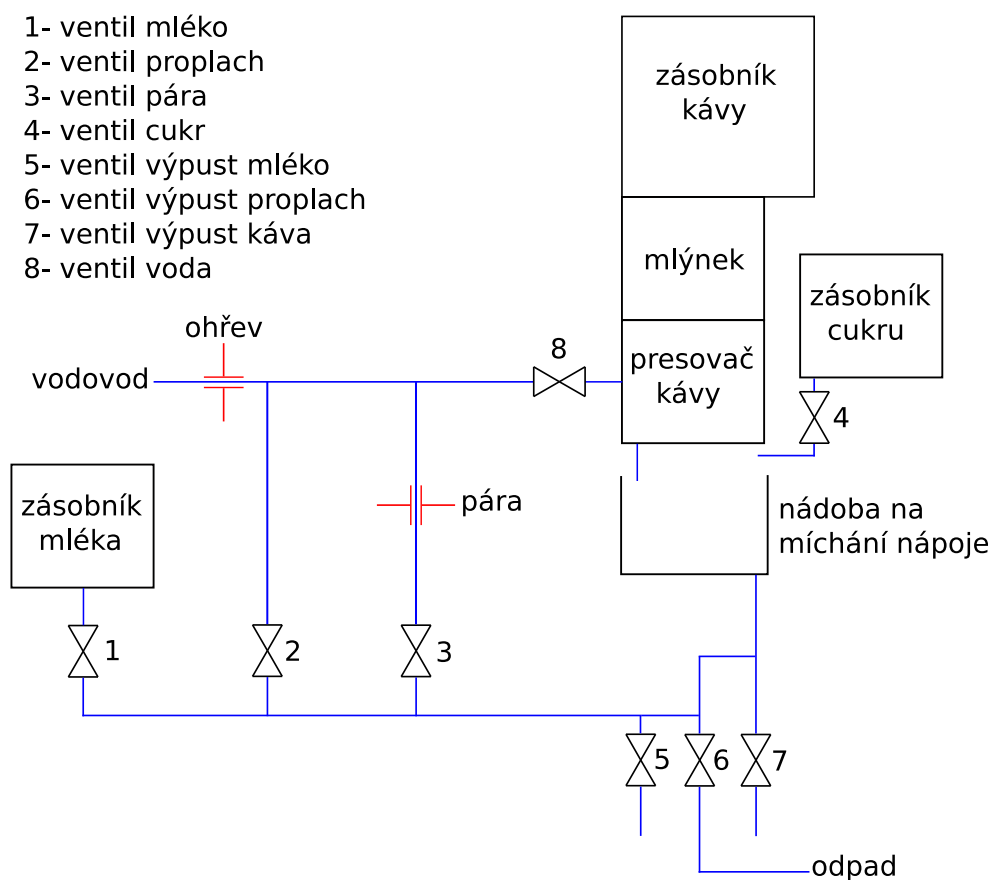
### 5.2.1 Zadání úlohy 2- Kancelářský kávovar

Navrhněte program pro řízení kávovaru pomocí PLC podle následujících požadavků. Kávovar bude nabízet espresso, cappuccino a caffè latte. Množství cukru bude volitelné. To bude určeno dobou, po kterou je otevřen ventil cukru. Nastavení doby realizujte analogovým výstupem z jednotky podle následující tabulky.

Tabulka 4- Úloha 2- Nastavení ventilu cukru

Analogový výstup z jednotky[V]	Doba otevření ventilu cukru
$\leq 1$	0s
(1;4>	1s
(4;7>	2s
$> 7$	3s

Kávovar bude připojen k vodovodu a odpadnímu potrubí. Káva, mléko a cukr budou v jednotlivých zásobnících.



Obrázek 20- Úloha 1- Schéma kávovaru

Kávovar bude zobrazovat množství nápoje v kelímku pomocí HMI panelu a pomocí analogového vstupu jednotky. Po přípravě tří nápojů za sebou nebo po stisknutí tlačítka se provede proplach kávovaru. Jednotlivé nápoje se připravují podle následujících pokynů. Ty popisují, v jakém pořadí a po jakou dobu mají být jednotlivé prvky spuštěny.

**Příprava našlehaného mléka-** ohřev pára (po uplynutí 0,2s) ► ventily páry, mléka a výpusti mléka (6s)

**Espresso-**mlýnek 2s ► presovač kávy 100ms ► ohřev a ventil vody 3s ► cukr ► ventil výpust káva 3s

**Cappuccino-** espresso ► našlehané mléko

**Caffe latte-** našlehané mléko ► espresso

**Proplach-** ventily vody, proplachu mléka a výpusti proplachu a ohřev (4s)

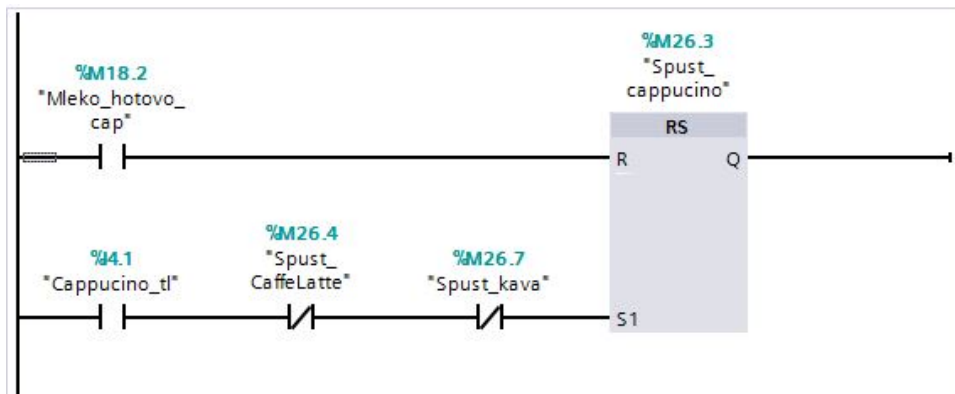
Na HMI panelu indikujte přípravu nápoje, množství cukru, množství nápoje v kelímku a dokončení nápoje.

Tabulka 5- Úloha 2- Seznam vstupních a výstupních proměnných do PLC

<i>Vstupní proměnné</i>	<i>Výstupní proměnné</i>
<i>Espresso tlačítko (DI)</i>	<i>Množství nápoje v kelímku (AQ)</i>
<i>Cappuccino tlačítko (DI)</i>	
<i>Caffe latte tlačítko (DI)</i>	
<i>Proplach tlačítko (DI)</i>	
<i>Množství cukru (AI)</i>	

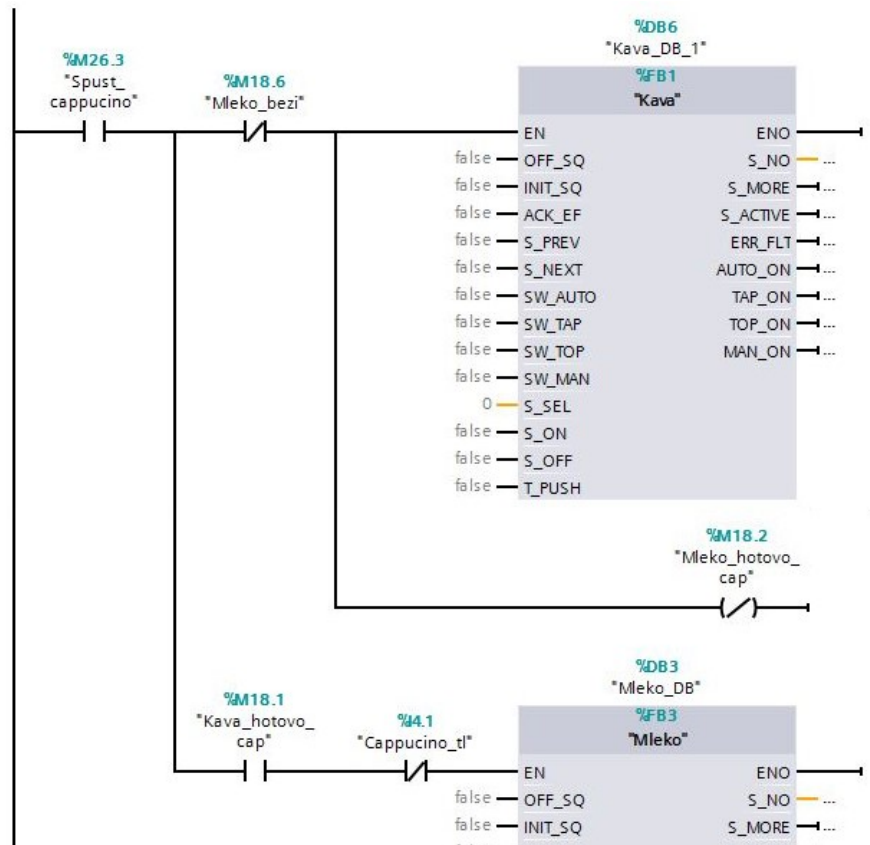
### 5.2.2 Řešení úlohy 2- Kancelářský kávovar

Řešení úlohy si předvedeme na příkladu cappuccina. Ostatní nápoje jsou řešeny obdobným způsobem. Obrázek 21 ukazuje RS klopný obvod, který uchovává hodnotu stisknutého tlačítka pro volbu nápoje. Jeho hodnota je resetována v případě, že je nápoj hotový. Po spuštění přípravy kteréhokoliv z nápojů dochází k blokaci spuštění ostatních.



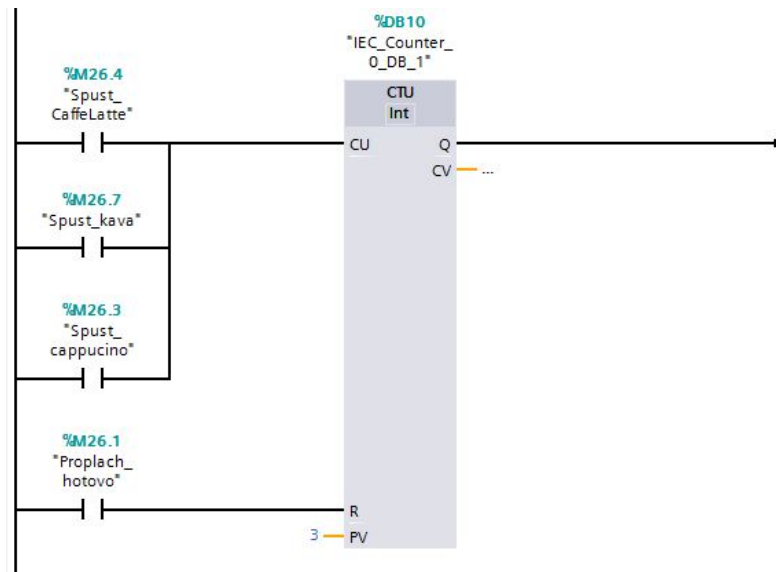
Obrázek 21-Úloha 2- Hlavní program- zvolení cappuccina

Obrázek 22 ukazuje řešení přípravy cappuccina. Jako první se po zvolení nápoje spustí funkční blok káva. Po jeho dokončení se spustí funkční blok mléko. Tyto dva bloky jsou ve vzájemné blokaci, aby nedošlo ke spuštění obou současně.



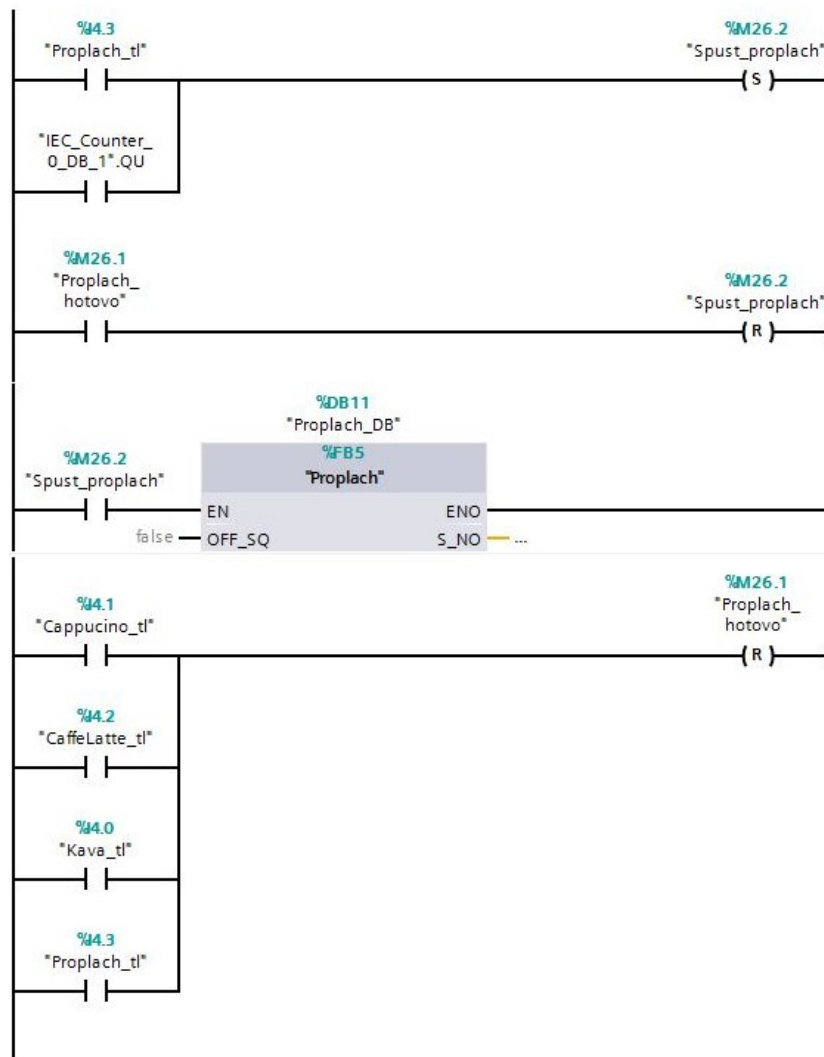
Obrázek 22- Úloha 2- Hlavní program- výroba cappuccina





Obrázek 23- Úloha 2- Hlavní program- počítání nápojů pro spuštění proplachu

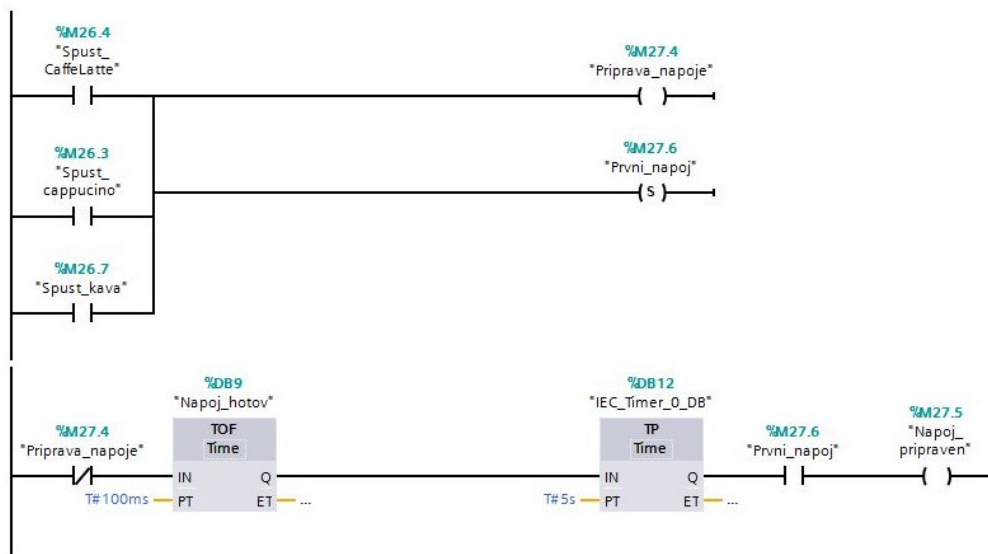
Na obrázku 23 můžeme vidět čítač, který počítá počet připravených nápojů. Poté, co dosáhne hodnoty 3, jeho logický výstup nabyde hodnoty 1 a je spuštěn proplach. Čítač je resetován v případě, je-li cyklus proplachu dokončen.



Obrázek 24- Úloha 2- Hlavní program- realizace proplachu

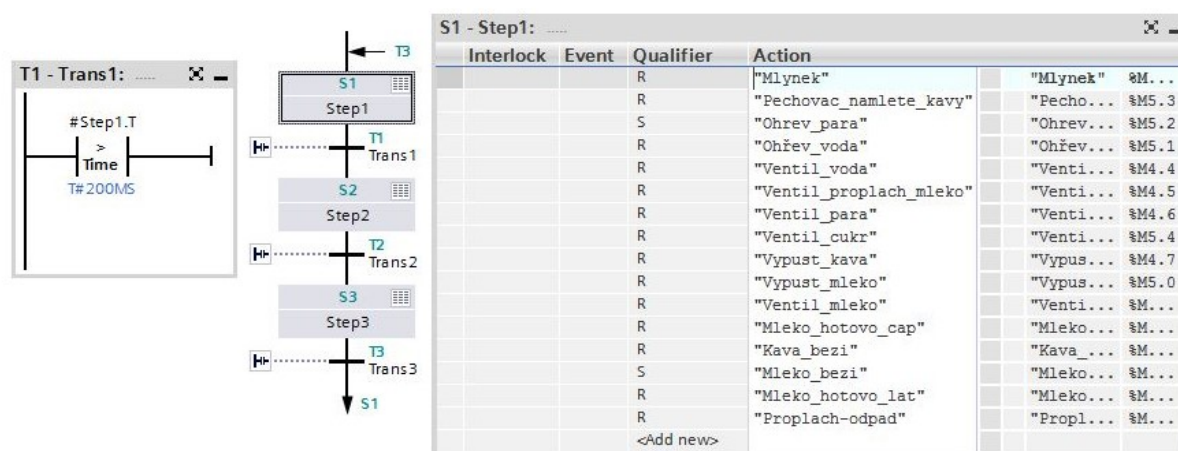
Obrázek 24 popisuje spuštění cyklu proplachu. Ten je spuštěn buď čítačem, nebo příslušným tlačítkem.

Následující obrázek 25 ukazuje, jak je v programu řešena indikace o stavu připravovaného nápoje.



Obrázek 25- Úloha 2- Hlavní program- indikace stavu nápoje

Obrázek 26 ukazuje funkční blok Mléko a jeho první krok s přechodem. Tento blok je napsaný v jazyku GRAPH, který jsem zvolil pro realizaci i u funkčních bloků Káva a Proplach.

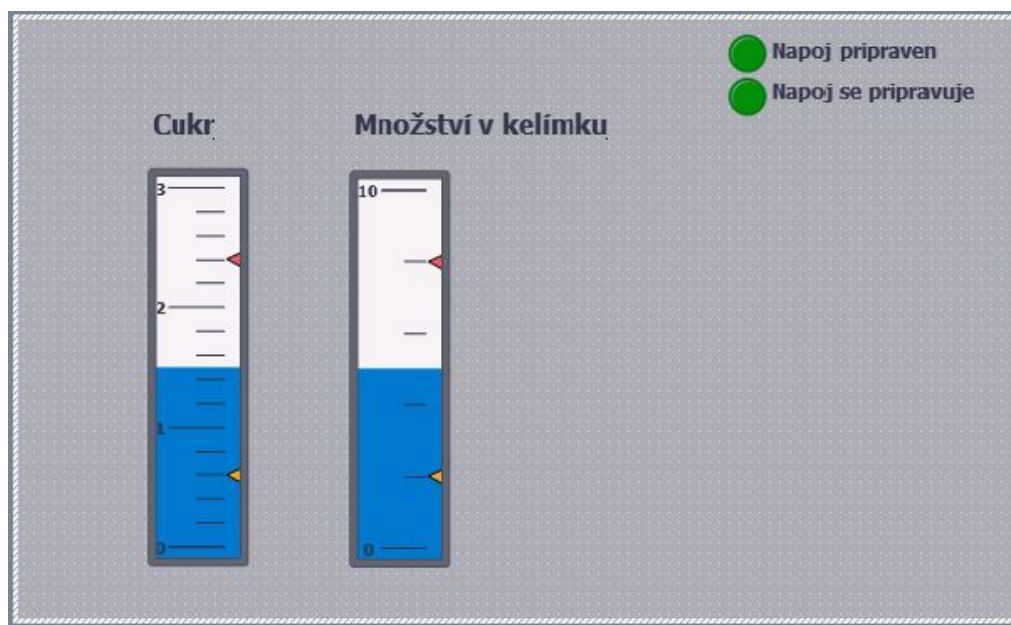


Obrázek 26- Úloha 2- Funkční blok Mléko- ukázka prvního kroku a prvního přechodu

Jelikož pro zjednodušení úlohy pracuji v celém programu s časy, určujícími dobu otevření ventilů a ne například s průtokoměry, bylo zapotřebí pro spojitou indikaci množství nápoje v kelímku vytvořit proměnnou typu INT, do které se převede časová hodnota. K tomu jsem využil integrovanou funkci v TIA Portal, která je pro tyto účely vytvořena. Obrázek 27 také popisuje nastavení množství cukru.

```
1 //Cukr
2 "Mnozstvi_cukru" := "Mnozstvi_cukru_vstup"/10;
3
4 IF "Mnozstvi_cukru" <=1 THEN
5     "Doba_cukr" := T#0s;
6 ELSIF "Mnozstvi_cukru"<=4 THEN
7     "Doba_cukr" := T#1s;
8 ELSIF "Mnozstvi_cukru"<=7 THEN
9     "Doba_cukr" := T#2s;
10 ELSIF "Mnozstvi_cukru"<=10 THEN
11     "Doba_cukr" := T#3s;
12 END_IF;
13
14
15
16 //Kelimek
17 "Mnozstvi_v_kelimku" := TIME_TO_INT("Kava_kelimek" + "Mleko_kelimek");
```

Obrázek 27- Úloha 2- Funkční blok Block3- množství cukru a množství nápoje v kelímku



Obrázek 28- Úloha 2- vizualizace na HMI panelu

### 5.3 Třetí úloha

Jako třetí úlohu jsem zvolil využití PLC pro chytrou domácnost. S tímto typem řešení se v praxi setkáme zřídka, nicméně cílem úlohy je naučit studenty pracovat s aktuálním denním časem uvnitř PLC a pro tyto potřeby je tento typ použití PLC názorný.

### 5.3.1 Zadání úlohy 3 - Chytrá domácnost

Navrhněte program pro řízení chytré domácnosti pomocí PLC. Chytrá domácnost bude ovládat topení, žaluzie a světla. Pro jednoduchost uvažujme pouze 3 místnosti: kuchyni, obývací pokoj a ložnici.

Chytrá domácnost bude pracovat ve dvou pracovních režimech, v automatickém režimu a režimu prázdný dům. Mezi režimy bude uživatel přepínat pomocí HMI panelu. Zvolený režim nastaví požadovanou pokojovou teplotu. Režim prázdný dům bude pracovat pouze s jednou teplotou. Automatický režim bude pracovat se dvěma teplotami. Požadovaná teplota přes den bude nastavena od 7:00 do 22:00. Požadovaná teplota přes noc bude nastavena po zbytek dne. Hodnoty požadovaných teplot bude možné zadávat přes HMI panel. Na ovládání žaluzií a světel, tyto dva režimy nemají vliv.

Pokojovou teplotu snímejte pomocí teplotního čidla Pt-100. Pokud pokojová teplota bude nižší než požadovaná teplota (s hysterezí 3°C), topení bude zapnuté, jinak vypnuté. K regulaci teploty domu napomáhá také úroveň stínění venkovních žaluzií. Jejich ovládání je popsáno níže. Řízení žaluzií je také ovlivněno vnějším soumrakovým spínačem. Pokud je sepnutý, žaluzie jsou spuštěné a úroveň stínění je 100%, nezávisle na teplotě.

Všechny žaluzie v domácnosti jsou ovládány přes jeden řídicí modul. Modul žaluzie obsahuje 2 svorky, svorku A pro spouštění či zvedání a svorku B od servo motoru, který ovládá stínění.

Tabulka 6- Úloha 3- ovládání žaluzie v závislosti na teplotě

<i>Rozdíl pokojové a požadované teploty</i>	<i>Stav žaluzie</i>	<i>Stínění</i>	<i>Svorka A</i>	<i>Svorka B</i>
$>3^{\circ}\text{C}$	<i>nahore</i>	0%	<i>log. 1</i>	<i>0V</i>
$(-2^{\circ}\text{C}; 3^{\circ}\text{C})$	<i>dole</i>	0%	<i>log. 0</i>	<i>0V</i>
$(-3^{\circ}\text{C}; -2^{\circ}\text{C})$	<i>dole</i>	10%	<i>log. 0</i>	<i>1V</i>
$(-4^{\circ}\text{C}; -3^{\circ}\text{C})$	<i>dole</i>	20%	<i>log. 0</i>	<i>2V</i>
$(-5^{\circ}\text{C}; -4^{\circ}\text{C})$	<i>dole</i>	50%	<i>log. 0</i>	<i>5V</i>
$<-5^{\circ}\text{C}$	<i>dole</i>	100%	<i>log. 0</i>	<i>10V</i>

Svorku A simulujte pomocí digitálního vstupu do jednotky. Svorku B simulujte pomocí analogového vstupu jednotky. Vnější soumrakový spínač simulujte pomocí digitálního výstupu z jednotky.

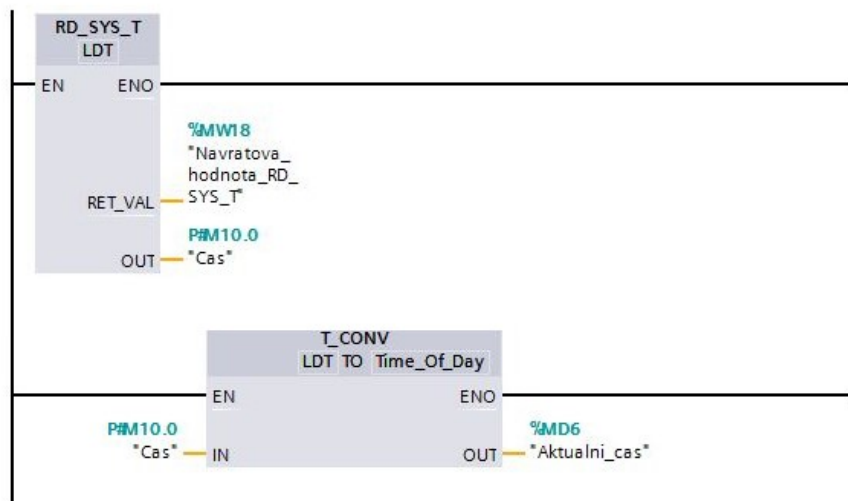
Světla v jednotlivých místnostech půjdou rozsvítit a zhasnout stiskem příslušného tlačítka. Pokud bude sepnutý vnitřní soumrakový spínač, světla začnou reagovat na pohybová čidla. Při zaznamenání pohybu se světlo rozsvítí na 30s. Pokud nezaznamená další pohyb, světlo zhasne. Tato funkce bude vypínatelná pomocí HMI panelu, aby nedocházelo k rozsvěcování například při spánku. Příslušná tlačítka a pohybová čidla simulujte pomocí digitálních výstupů jednotky. Světla simulujte pomocí digitálních výstupů jednotky.

Tabulka 7- Úloha 3- Seznam vstupních a výstupních proměnných PLC

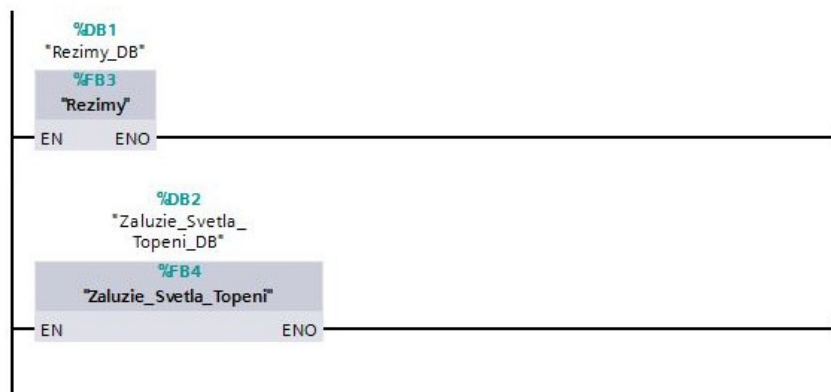
<i>Vstupní proměnné</i>	<i>Výstupní proměnné</i>
<i>Pokojová teplota (AI)</i>	<i>Světlo kuchyň (DQ)</i>
<i>Světlo kuchyň tlačítko (DI)</i>	<i>Světlo obývací pokoj (DQ)</i>
<i>Světlo obývací pokoj tlačítko (DI)</i>	<i>Světlo ložnice (DQ)</i>
<i>Světlo ložnice tlačítko (DI)</i>	<i>Svorka A- stav žaluzie (DQ)</i>
<i>Pohybové čidlo kuchyň (DI)</i>	<i>Svorka B- stínění žaluzie (AQ)</i>
<i>Pohybové čidlo obývací pokoj (DI)</i>	
<i>Pohybové čidlo ložnice (DI)</i>	
<i>Soumrakový spínač vnější (DI)</i>	
<i>Soumrakový spínač vnitřní (DI)</i>	

### 5.3.2 Řešení úlohy 3- Chytrá domácnost

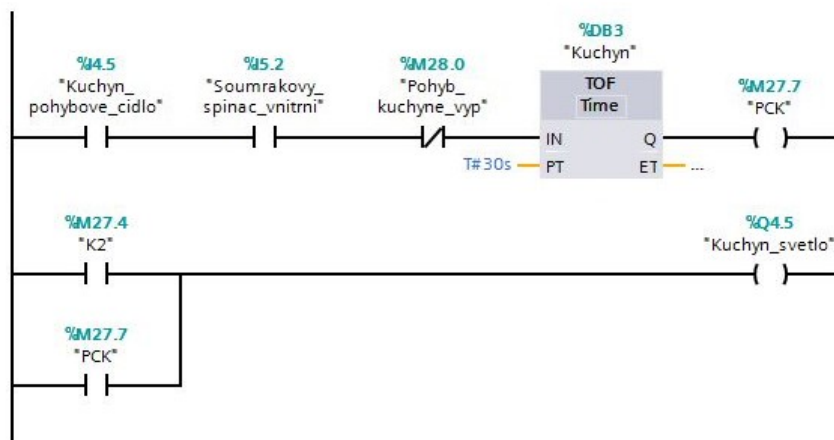
Pro řešení této úlohy jsem zvolil kombinaci programovacích jazyků LAD a STL. Pro správný chod automatického režimu bylo zapotřebí načíst aktuální čas dne. Ten se načel do proměnné typu LDT (Date and Long Time), který bylo následně zapotřebí převést do typu TOD (Time of Day). Toto řešení ukazuje obrázek 29. V obrázku 30 můžeme vidět zařazení funkčního bloku `Žaluzie_Světla_Topení` a bloku `Režimy` do chodu hlavního programu.



Obrázek 29- Úloha 3- Hlavní program- načtení aktuálního času do PLC



Obrázek 30- Úloha 3- Hlavní program- vložení funkčních bloků



Obrázek 31- Úloha 3- Hlavní program- spouštění světla

Obrázek 31 ukazuje ovládání světla v kuchyni. Světlo je rozsvíceno pohybovým čidlem nebo stiskem tlačítka. Testování stisku tlačítka je ukázáno na obrázku 32. Stejným způsobem jsou ovládány světla v ostatních místnostech.

```

47 IF "K1"=NOT("T1_kuchyn") THEN
48     "K1" := NOT("K1");
49 IF "K1"=true THEN
50     "K2" :=NOT("K2");
51     END_IF;
52
53 END_IF;

```

Obrázek 32- Úloha 3- Žaluzie\_světla\_topení- testování stisku tlačítka

```

1 //Režimy
2 IF "Start_automat"=TRUE THEN
3 IF ("Aktualni_cas">=(TIME_OF_DAY#07:00:00.000))AND ("Aktualni_cas"<=(TIME_OF_DAY#22:00:00.000)) THEN
4     "Den" := TRUE;
5     "Noc" := FALSE;
6     "Start_Prazdny_dum" := FALSE;
7     "Pozadovana_teplota" := "Pozadovana_teplota_den";
8 ELSE
9     "Noc" := TRUE;
10    "Den" := FALSE;
11    "Start_Prazdny_dum" := FALSE;
12    "Pozadovana_teplota" := "Pozadovana_teplota_noc";
13 END_IF;
14 ELSE
15 IF "Start_Prazdny_dum" = TRUE THEN
16     "Pozadovana_teplota" := "Pozadovana_teplota_prazdny_dum";
17     "Noc" := FALSE;
18     "Den" := FALSE;
19     "Start_automat" := FALSE;
20 END_IF;
21
22 END_IF;

```

Obrázek 33- Úloha 3- Režimy- Nastavení požadovaných teplot v závislosti na režimu

Obrázek 33 ukazuje funkční blok Režimy. Program testuje zvolený režim chodu domácnosti. Pokud je uživatelem na HMI panelu zvolen automatický režim, program testuje, zda je aktuální denní čas v daném intervalu. Následně je přiřazena příslušná požadovaná hodnota pokojové teploty.

Obrázek 34 ukazuje program funkčního bloku Žaluzie\_Světla\_topení. Ten testuje stav vnějšího soumrakového spínače. Pokud je sepnutý, pokojová teplota je regulována pouze spouštěním topení a žaluzie jsou zatažené. Pokud je snímač rozepnutý, je testován rozdíl mezi pokojovou a požadovanou teplotou a podle jeho hodnoty je nastavována úroveň stínění žaluzií a ovládáno topení.

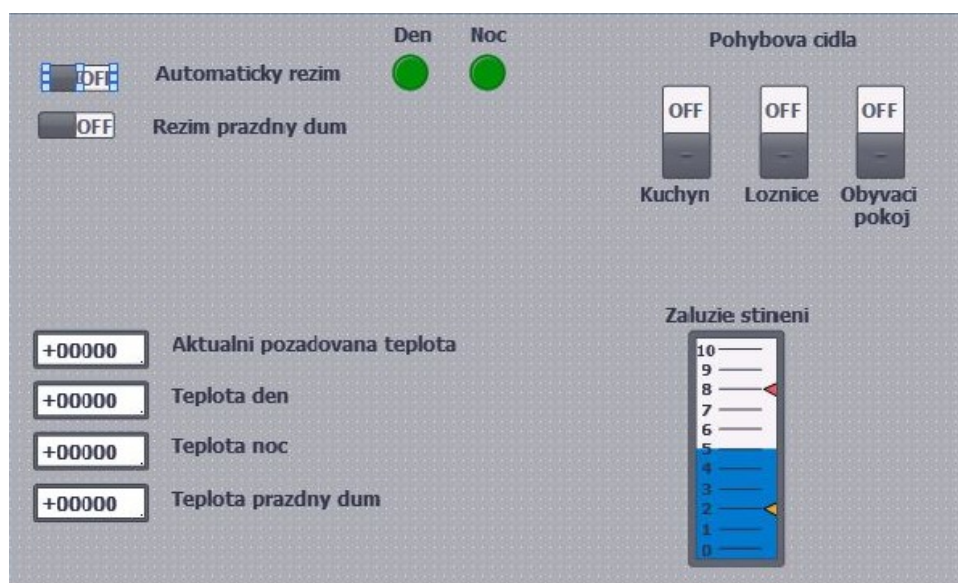


```

1 //Zaluzie_topeni
2 //
3
4 "Realna_teploata_pokoje" := "Pokojeva_teploata" / 10;
5
6 IF "Soumrakovy_spinac_vnejsi" = 0 THEN
7
8   IF "Realna_teploata_pokoje" <= ("Pozadovana_teploata" - 3.0) THEN
9     "Topeni" := TRUE;
10    "Zaluzie_zdvih" := TRUE;
11    "Zaluzie_stineni" := 0;
12   ELSIF "Realna_teploata_pokoje" > ("Pozadovana_teploata" + 5.0) THEN
13     "Topeni" := FALSE;
14     "Zaluzie_zdvih" := FALSE;
15     "Zaluzie_stineni" := 10;
16   ELSIF "Realna_teploata_pokoje" > ("Pozadovana_teploata" + 4.0) THEN
17     "Topeni" := FALSE;
18     "Zaluzie_zdvih" := FALSE;
19     "Zaluzie_stineni" := 5;
20   ELSIF "Realna_teploata_pokoje" > ("Pozadovana_teploata" + 3.0) THEN
21     "Topeni" := FALSE;
22     "Zaluzie_zdvih" := FALSE;
23     "Zaluzie_stineni" := 2;
24   ELSIF "Realna_teploata_pokoje" > ("Pozadovana_teploata" + 2.0) THEN
25     "Topeni" := FALSE;
26     "Zaluzie_zdvih" := FALSE;
27     "Zaluzie_stineni" := 1;
28   END_IF;
29
30
31 ELSE
32   "Zaluzie_zdvih" := FALSE;
33   "Zaluzie_stineni" := 10;
34   IF "Realna_teploata_pokoje" <= ("Pozadovana_teploata" - 3.0) THEN
35     "Topeni" := 1;
36   END_IF;
37
38   IF "Realna_teploata_pokoje" >= "Pozadovana_teploata"
39   THEN
40     "Topeni" := 0;
41   END_IF;
42
43 END_IF;

```

Obrázek 34- Úloha 3- Žaluzie\_světla\_topení- ovládání žaluzií a topení

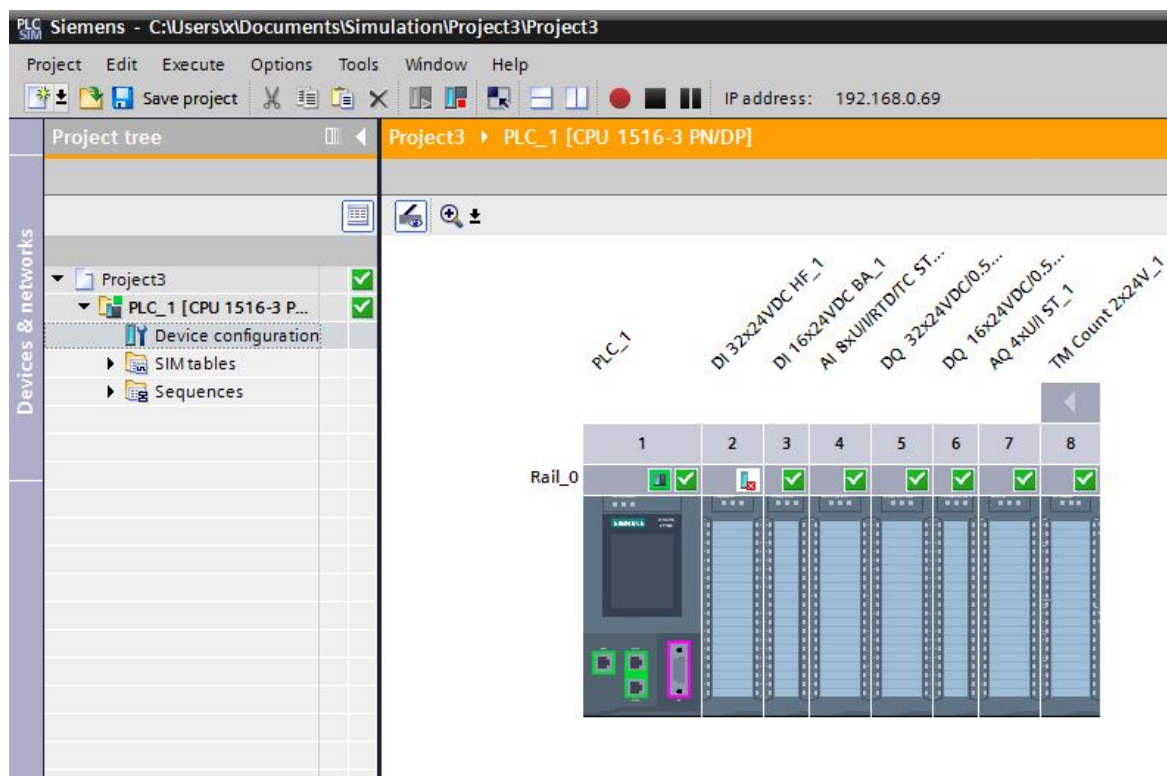


Obrázek 35- Úloha 3- Vizualizace na HMI panelu

## 6 Testování vytvořených programů

K testování a ověření funkčnosti vytvořených programů jsem použil program PLC SIM. Jedná se o online simulátor PLC, ve kterém lze vytvořit virtuální dvojče hardwaru, pro který je program psán.

Simulační systémy poskytují účinnou podporu při vývoji programů a následné skutečné aplikaci. Včasné odhalení programových chyb a optimalizace programových sekcí umožňují optimalizované a bezchybné používání programů ve vlastním systému. Pokud je program modifikován, může být testován před jeho načtením do řídicího systému zařízení. Díky tomu lze v prostředí automatizace snížit například časy potřebné pro oživení řízeného systému, či aktualizaci programu, a tím i snížení prostojů výroby, které zvyšují náklady.



Obrázek 36- PLC SIM- konfigurace PLC

Po spuštění simulace přes prostředí TIA Portal se konfigurace v PLC SIM provede automaticky získáním dat o hardwaru, který je v TIA Portal již nakonfigurován.

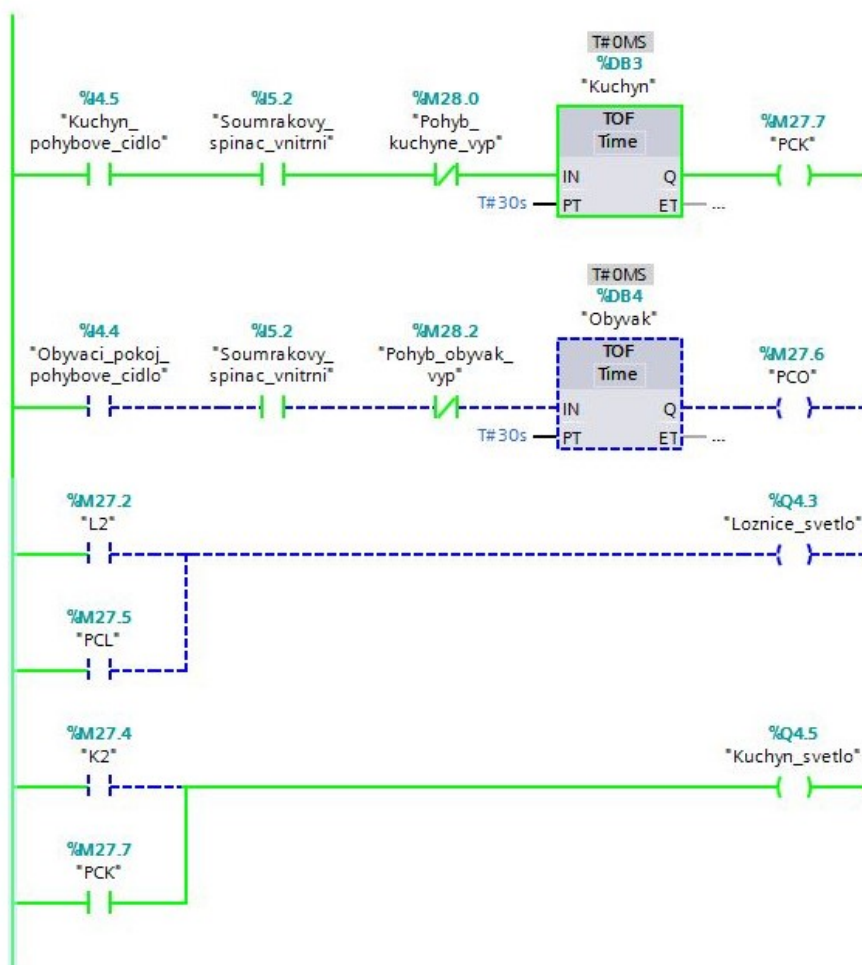
Obrázky 37 a 38 jsou ukázkou simulace úlohy 3, kde byly vybrány vstupní proměnné programu, které lze ovládat, a simulovat tak například skutečné stisknutí tlačítka. Ukázka

simulace testuje funkčnost ovládání světla v kuchyni. V prostředí TIA Portal lze pak online sledovat stav logické úrovně signálu, která je znázorněna graficky.

SIM table\_1

Name	Monit...	Bits	C...
"TI_kuchyn"	FALSE		F...
"TI_loznice"	FALSE		F...
"TI_obyvaci_pokoj"	FALSE		F...
"Soumrakovy_spinac_vnitri"	TRUE		<input checked="" type="checkbox"/> F...
"Soumrakovy_spinac_vnejsi"	FALSE		F...
"Kuchyn_pohybove_cidlo"	TRUE		<input checked="" type="checkbox"/> F...
"Loznice_pohybove_cidlo"	FALSE		F...
"Obyvaci_pokoj_pohybove_cidlo"	FALSE		F...

Obrázek 37- PLC SIM- tabulka ovládaných proměnných



Obrázek 38- TIA Portal- online sledování logické úrovně signálu

## 7 Závěr

V začátku práce je popsána historie a možnosti provedení PLC. V následující části se práce věnuje PLC Siemens Simatic S7-1500, kde jsou popsány integrované funkce společně s nejčastěji používanými moduly. V popisu prostředí TIA Portal se práce zaměřuje na popis programovacích jazyků a popis vybraných datových typů.

Hlavním cílem práce bylo navrhnout a realizovat testovací jednotku pro programování PLC Siemens Simatic S7-1500. Navržená jednotka splňuje všechny parametry zadání a poskytuje tak možnost indikace osmi digitálních a dvou analogových vstupů. Osm digitálních výstupů a dva analogové výstupy umožňují simulovat různé snímače a senzory. Funkčnost jednotky byla ověřena připojením k PLC. Během testování se ukázalo, že testovací jednotka se hodí především pro výukové účely. Její vysoká univerzálnost poskytuje možnost použití v široké škále příkladů z automatizace. Limitujícím parametrem by mohl být omezený počet jednotlivých vstupů a výstupů. Při výrobě jednotky jsem se zaměřil především na robustnost a jednoduchost provedení. To v případě poruchy usnadní případnou opravu. Celková cena testovací jednotky je 1511 Kč. Cena by šla snížit použitím levnějších komponentů.

Druhým cílem práce bylo vytvoření tří vzorových úloh, na kterých lze demonstrovat funkčnost jednotky. Při tvorbě úloh bylo zohledněno jejich možné využití při výuce. Proto jsem vytvořil detailní zadání a následně popsal řešení formou ukázky programu s komentářem. První úlohou je Řízení kotelny pomocí PLC. Druhou úlohou je Kancelářský kávovar. Primárním cílem této úlohy je přiblížení problematiky sekvenčního programování v jazyku GRAPH. Třetí úlohou je Chytrá domácnost. Tato úloha má za úkol naučit studenty pracovat s reálným časem uvnitř PLC. Všechny tři úlohy vhodně demonstrijí funkčnost jednotky a pro jejich vytvoření jsem použil programovací jazyk LAD, STL a GRAPH. Funkčnost vytvořených programů jsem ověřil nejprve v simulačním programu PLC SIM a pak i připojením k PLC v učebně VOŠ a SPŠE Plzeň.

## Seznam literatury a informačních zdrojů

- [1] Luděk ZIMMERMANN. *Co to je PLC a k čemu?* In: ElektriKa.cz [online] Copyright © 1998-2019 [cit. 21. 5. 2019]. Dostupné z: <https://elektriKa.cz/data/clanky/co-to-je-plc-a-k-cemu>
- [2] *PLC - sestava*. PLC AUTOMATIZACE [online] [cit. 21. 5. 2019]. Dostupné z: <http://www.plc-automatizace.cz/knihovna/plc/plc-hw-sestava.htm>
- [3] ŠMEJKAL, Ladislav a Marie MARTINÁSKOVÁ. *PLC a automatizace*. Praha: BEN - technická literatura, 1999. ISBN 80-86056-58-9.
- [4] *Nejrychlejší řídicí systém pro automatizaci*. In: Siemens.com [online] Copyright © 1996 - 2019 Siemens, s.r.o. [cit. 21. 5. 2019] Dostupné z: <https://new.siemens.com/cz/cs/products/automation/systems/industrial/plc/simatic-s7-1500.html>
- [5] Bc. Ludvík KOCHANÍČEK. *Programovací jazyky pro PLC*. [online] 8.4.2010 [cit. 21. 5. 2019]. Dostupné z: <https://coptkm.cz/portal/?doc=3905>
- [7] *SIMATIC S7-PLCSIM V5.4*. In: Siemens.com [online] Copyright © 1996 - 2019 Siemens, s.r.o. [cit. 21. 5. 2019] Dostupné z: <https://w3.siemens.com/mcms/simatic-controller-software/en/step7/simatic-s7-plcsim/Pages/Default.aspx>

## Přílohy

### Příloha A – Schéma zapojení testovací jednotky

