

VYSOKÁ ŠKOLA BÁŇSKÁ - TECHNICKÁ UNIVERZITA OSTRAVA  
FAKULTA STROJNÍ  
KATEDRA AUTOMATIZAČNÍ TECHNIKY A ŘÍZENÍ

# Návrh řešení a příprava funkčních bloků pro řízení procesu

The design and implementation of functional blocks for process  
control

Student:

Bc. Lukáš Křetinský

Vedoucí diplomové práce:

doc. Ing. Lenka Landryová, CSc.

Ostrava 2017



VŠB - Technická univerzita Ostrava  
Fakulta strojní  
Katedra automatizační techniky a řízení

## Zadání diplomové práce

Student: **Bc. Lukáš Křetinský**  
Studijní program: N2301 Strojní inženýrství  
Studijní obor: 3902T004 Automatické řízení a inženýrská informatika  
Téma: **Návrh řešení a příprava funkčních bloků pro řízení procesu**  
**The Design and Implementation of Functional Blocks for Process Control**  
Jazyk vypracování: čeština

Zásady pro vypracování:

1. Seznamte se s řízením technologie plnění vysoké pece, vypište základní terminologii pro automatizaci a systém řízení tohoto procesu.
2. Pro zvolený kontroler prostudujte a popište jeho hardware a funkční bloky pro Step\_7.
3. Seznamte se s konfiguračním a vizualizačním prostředím pro zvolený řídicí systém, analyzujte zařízení dle funkčního popisu a použitých vstupů/výstupů, a jeho vizualizaci (vytvořené ikony, pop-up okna).
4. Vytvořte a otestujte simulační bloky v prostředí WinMod.
5. Shrňte nové poznatky navrhovaného řešení.

Seznam doporučené odborné literatury:

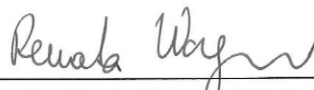
Foltýnek, P. Vizualizace technologických procesů a řízení na platformě PC. Dostupné z URL  
<http://books.fs.vsb.cz/FactoryLink/Index.html>  
Foltýnek, P. Procesní vizualizace v průmyslové automatizaci. Ostrava: katedra ATR-352 VŠB-TUO, 2001.  
53 s. Bakalářská práce, vedoucí: Landryová, L.  
Skýpala, J. Design a konstrukce testovacího zařízení pro AC700F. Ostrava: katedra ATR-352 VŠB-TUO,  
2015. Diplomová práce, vedoucí: Landryová, L.  
Havránková, R. Hardwarový simulátor technologických procesů pro řídicí systém ABB/AC800. Ostrava:  
katedra ATR-352 VŠB-TUO, 2015. Diplomová práce, vedoucí: Landryová, L.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí diplomové práce: **doc. Ing. Lenka Landryová, CSc.**

Datum zadání: 09.12.2016

Datum odevzdání: 15.05.2017



doc. Ing. Renata Wagnerová, Ph.D.  
vedoucí katedry

  
doc. Ing. Ivo Hlavatý, Ph.D.  
děkan fakulty



Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě .....

.....

podpis studenta



Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užít (§ 35 odst. 3).
- souhlasím s tím, že diplomová práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě :.....

.....  
podpis

Křetinský Lukáš  
Olšava 2215  
688 01, Uherský Brod





## **Poděkování**

Na tomto místě bych rád poděkoval vedoucí mé diplomové práce doc. Ing. Lence Landryové, CSc., za odborné vedení a připomínky k práci, stejně tak i vedoucímu automatizačního oddělení firmy Ingeteam a.s., panu Ing. Jaroslavu Tyšerovi za cenné rady a možnosti zúčastnit se projektu ve firmě Ingeteam a.s., díky které vznikla tato práce. Největší poděkování patří panu Ing. Stanislavu Hovancovi a Ing. Vítovi Lukášovi, za odbornou konzultaci a pomoc. Mé poděkování patří i mé rodině a hlavně mé přítelkyni, která mě neustále podporovala a motivovala k úspěšnému dokončení této práce.



## **Anotace diplomové práce**

KŘETINSKÝ, L. *Návrh řešení a příprava funkčních bloků pro řízení procesu : diplomová práce*. Ostrava : VŠB – Technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2017, 81 s. Vedoucí práce: Landryová, L.

Diplomová práce se zabývá řešením a přípravou funkčních bloků pro řízení procesu. Konkrétně se jedná o modernizaci řízení vysoké pece. Práce se zabývá řídicím systémem a jeho hardwarovým vybavením, konfiguračním a vizualizačním prostředím a samotný závěr práce se týká tvorby simulace vysoké pece a částečné vizualizace některých obrazovek a prvků. Práce je zakončena aplikováním všech získaných poznatků naprogramováním řízení dvoucívkového ventilu v prostředí Step 7 a jeho otestování v simulačním prostředí WinMOD.

### **Klíčová slova**

*PLC, WinMOD, Archestra IDE, Step 7, vysoká pec, bezzvonová sazebná, stockhouse, skip*

## **Annotation of the thesis**

KŘETINSKÝ, L. *The design and implementation of functional blocks for process control: master thesis*. Ostrava : VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Automation Engineering and Control), 2017, 81 p. Thesis head: Landryová, L.

This thesis deals with design and preparation of functional blocks for control of the process. Specifically, it is a modernization of blast furnace control. The thesis deals with the control system and its hardware equipment, configuration and visualization environment, and the conclusion of the thesis itself concerns the creation of blast furnace simulation and partial visualization of some screens and elements. The work is completed by applying all the knowledge gained by programming the two-coil valve control in the Step 7 environment and testing it in the WinMOD simulation environment.

### **Keywords**

*PLC, WinMOD, Archestra IDE, Step 7, blast furnace, bell less top charging system, stockhouse, skip*



## Obsah

<b>1</b>	<b>Úvod .....</b>	<b>16</b>
<b>2</b>	<b>Vysoká pec .....</b>	<b>18</b>
<b>3</b>	<b>Řídicí systém a jeho hardware.....</b>	<b>19</b>
3.1	PLC Siemens Simatic S7 – 400 .....	19
3.1.1	ET 200M.....	20
3.1.2	Parametry komponentů PLC .....	21
3.2	Profibus.....	22
3.3	Ethernet.....	23
<b>4</b>	<b>Konfigurační a vizualizační prostředí.....</b>	<b>25</b>
4.1	PLC program.....	25
4.2	Simatic Step7 .....	26
4.3	WinMOD .....	28
4.1	Makra .....	30
4.2	Archestra IDE .....	34
4.2.1	Aplikační server .....	35
4.2.2	Historický server .....	35
<b>5</b>	<b>Definiční listina .....</b>	<b>37</b>
5.1	Deflist BLT .....	37
<b>6</b>	<b>Simulace .....</b>	<b>39</b>
6.1	Import a export vstupně/výstupních dat do aplikace WinMOD .....	41
6.2	Tvorba simulace.....	42
6.3	Provoz přípravy a vážení vysokopecní vsázky .....	43
6.3.1	Přísady .....	43
6.3.2	Zpracování železné rudy .....	45
6.3.3	Zpracování koksu.....	47
6.4	Skip .....	50
6.5	Bezzvonová sazebna .....	51
6.6	Odprášení .....	57
6.7	Hydraulika .....	60
<b>7</b>	<b>Vizualizace .....</b>	<b>63</b>
<b>8</b>	<b>Dvoucívkový ventil a jeho aplikace .....</b>	<b>70</b>
8.1	Organizační blok.....	70
8.2	Funkční blok .....	71
8.2.1	Diagnostika .....	71
8.2.2	Tělo programu.....	72
8.2.3	Statusy a hlášení.....	74
8.3	Simulace ventilu .....	75
<b>9</b>	<b>Závěr .....</b>	<b>77</b>
<b>10</b>	<b>Seznam použité literatury .....</b>	<b>79</b>
<b>11</b>	<b>Příloha na CD.....</b>	<b>81</b>

## Seznam použitých zkratk

AI	Analog Input	Analogový vstup
AO	Analog Output	Analogový výstup
BLT	Bell Less Top	Bezzvonová sazebná
CPU	Central Processing Unit	Centrální procesorová jednotka
DBx	Data Block	Datový blok PLC
DDE	Dynamic Data Exchange	Dynamická výměna dat
DI	Digital Input	Digitální vstup
DO	Digital Output	Digitální výstup
DP	Decentralised Peripherals	Typ průmyslové komunikační sběrnice
FAT	Factory acceptance tests	Funkční testy
FBD	Function Block Diagram	Grafický jazyk PLC
FBx	Function Block	Funkční blok PLC
FCx	Function	Funkce PLC
GSD	General Station Definition	Popis souboru vyvinutý PROFIBUS & PROFINET
HMI	Human–Machine Interface	rozhraní mezi člověkem a strojem.
HW	HardWare	Fyzické vybavení počítače
IEC	International Electrotechnical Commission	Mezinárodní elektrotechnická komise
IEEE	Institute of Electrical and Electronics Engineers	Institut pro elektrotechnické a elektronické inženýrství
IL	Instruction List	Textový jazyk PLC
ISO	International Organization for Standardization	Mezinárodní organizace pro normalizaci
LD	Ladder Diagram	Grafický jazyk PLC
MMC	MultiMediaCard	standard paměťové karty s technologií paměti flash
MPI	Multi-Point Interface	Typ průmyslové komunikační sběrnice
OBx	Organization Block	Organizační blok PLC
OPC	OLE for Process Control	Komunikační protokol
OSI	Open Systems Interconnection	Standard propojení otevřených systémů
PC	Personal Computer	Osobní počítač
PDF	Portable Document Format	Formát pro přenos dokumentů
PLC	Programmable Logic Controller	programovatelný automat
Profibus DP	Decentralized Periphery	
Profibus FMS	Fieldbus Message Specification	
Profibus PA	Process Automation	
PROFINET	Profinet Isochronous Real Time	
IRT	Committee 104	
PROFINET RT	Profinet Real Time	
SFBx	System Function Block	Speciální funkční blok PLC

---

SCADA	Supervisory Control And Data Acquisition	dispečerské řízení a sběr dat
SD	Secure Digital	Označení paměťové karty
SFC	Sequential Function Chart	Grafický jazyk PLC
SFCx	System Function	Speciální funkce PLC
SPC	Statistical Process Control	Statistická regulace procesu
SQL	Structured Query Language	Strukturovaný dotazovací jazyk
ST	Structured Text	Textový jazyk PLC
SW	SoftWare	Programové vybavení počítače
TFT	Thin-Film Transistor	technologie výroby displejů z tekutých krystalů
USB	Universal Serial Bus	univerzální sériová sběrnice
VATx		Tabulka proměnných PLC
Windows CE	Windows Embedded Compact	Operační systém firmy Microsoft pro malé počítače (Handheld PC, PDA)

# 1 Úvod

V dnešní době automatizace představuje komplexní použití řídicích systémů a dalších komponentů automatizační techniky při zásadním požadavku zajistit vysoký stupeň provozuschopnosti a bezpečnosti řízeného technologického zařízení. Ovládání takového plně, nebo polo automatizovaného technologického celku z pohledu člověka - operátora, je jednodušší, lépe kontrolovatelné. Vyznačuje se kratší dobou zásahu v případě poruchy, nebo jiných výjimečných stavů i díky tomu, že ovládání celku je soustředěno do jednoho místa, tzv. velínu. Odtud se celé zařízení řídí a kontroluje se průběh výrobního procesu. Mimo to je vše řízeno a zabezpečeno počítači a řídicími systémy tak, že jsou schopny akční zásah v případě poruchy zabezpečit sami, bez jakéhokoliv zásahu operátora. Ten se o situaci dozví informativně z vizualizačního prostředí díky alarmům.

Velký význam má automatizace v teplárenském, elektrárenském ale hlavně v hutním průmyslu, kde se ve většině případů pracuje v nepřetržitém provozu. Postupně se automatizují kotle, turbogenerátory, výměňkové stanice, nebo celé hutní celky, např. vysoké pece a její přidružené technologie. Takový způsob automatizace, jež řídí technické veličiny (výšky hladin, teploty atd.) výrobního procesu, uplatňující automaty pracující se zpětnou vazbou, nazýváme řízení technologických procesů.

Průmyslovou automatizací a samotným vývojem řídicích systémů se zabývá po celém světě spousta firem. Jednou z nich je i nadnárodní firma Ingeteam a.s., která mimo jiné umožňuje studentům podílet se na reálných projektech nejen z oblasti automatizace. Tuto možnost jsem využil i já a díky tomu jsem mohl nejen získat spoustu cenných a praktických zkušeností, ale vypracovat i tuto diplomovou práci. Táto práce je součástí projektu rekonstrukce automatizovaného řízení vysoké pece v Krakově (Polsko), jehož zadavatelem je společnost ArcelorMittal, a. s.

Cílem práce je seznámit se s řízením technologie plnění vysoké pece a popsat hardware vhodně zvoleného řídicího systému. Seznámit se s konfiguračním a vizualizačním prostředím pro daný řídicí systém a následně vytvořit kompletní simulaci vysoké pece v prostředí WinMOD, která je velmi důležitá pro správný, levný a rychlý vývoj řídicího systému. Právě tvorba simulačních oken v prostředí WinMOD je hlavní náplní této práce. Simulace je rozdělena na několik částí. V první řadě jsou popsány definice jako definiční listina, makro



apod. Dalším krokem je vysvětlení, jak připravit nový projekt a co všechno je potřeba pro to udělat. Následně popisují samotnou tvorbu simulace pro řízení technologických celků, od provozu přípravy a vážení vysokopecní vsázky, skipový vrátek přes sazebnu vysoké pece až po hydrauliku. V samém závěru práce je pak aplikování získaných poznatků, kde jsem naprogramoval ve vývojovém prostředí SIMATIC Step 7 řídicí systém pro dvoucívkový ventil, a následně ho otestoval v simulačním prostředí WinMOD.

## 2 Vysoká pec

Je metalurgické zařízení sloužící pro výrobu surového železa. Jedná se o složité zařízení, skládající se z mnoha přidružených přípravných a pomocných provozů, bez kterých se celý výrobní proces neobejde. Mezi tyto provozy patří například aglomerace a peletizace, kde se zpracovává aglomerát a jiné suroviny (pelety) pro vsázku vysoké pece. Dále koksovna, zařízení na čištění vysokopecního plynu, ohříváče aj.

Dnešní inovace vysokopecní výroby nemění podstatu výroby surového železa ve vysoké peci, ale promítají se hlavně do:

- růstu velikosti vysokých pecí (větší užitečný objem pecí, vytvoření lepších pracovních podmínek na plošině vysoké pece apod.)
- uplatnění nových systémů řízení vysoké pece, zejména počítačové automatizace vysokopecního pochodu jako celku i jednotlivých dílčích subsystémů zahrnujících především:
  - řízení ohříváčů
  - řízení teploty pece
  - modelů řízení ukládání surovin na povrchu vsázkového sloupce ve vysoké peci (radiální i obvodové)
  - řízení intenzity chodu vysoké pece s diagnostikou stavu
  - řízení dalších doplňkových subsystémů
- využívání nejrůznějších druhů náhradních paliv (práškové uhlí, drcené plasty, těžké oleje, hnědouhelné a černouhelné dehty, zemní a karbonský plyn, konverzi plyny případně i jejich kombinací)
- celková ekologizace vysokopecních závodů (odsávání plynů, odprašování apod.)
- využívání nových typů sazeben pro dokonalejší ukládání vsázky (bezzvonové sazebny) [Brož, 1985]

Pro pochopení celé problematiky řízení vysoké pece a její plnění je potřeba znát, jak taková pec vlastně funguje. Jaké druhy sazeben existují a jak fungují, co se v nich ukrývá, co vše lze řídit a jak se plní. To vše je vysvětleno v teoretické části, která je součástí této práce a nachází se v přílohách této práce, viz příloha I.

## 3 Řídicí systém a jeho hardware

Mozkem řízení celé technologie je řídicí systém. Sbírá vstupní data, třídí je, vyhodnocuje a rozhoduje, jak s těmito informacemi naložit. Proto je důležité zvolit si takový řídicí systém, který bude robustní a bude splňovat určitá kritéria, pro řízení tak velkého technologického celku. V dnešní době je spousta typů řídicích systémů od různých výrobců a jedním z nejrozšířenějších po celém světě je Simatic S7-400 a vývojové prostředí Step7 od firmy Siemens, který je oblíbený hlavně kvůli jeho přehlednému vývojovému prostředí.

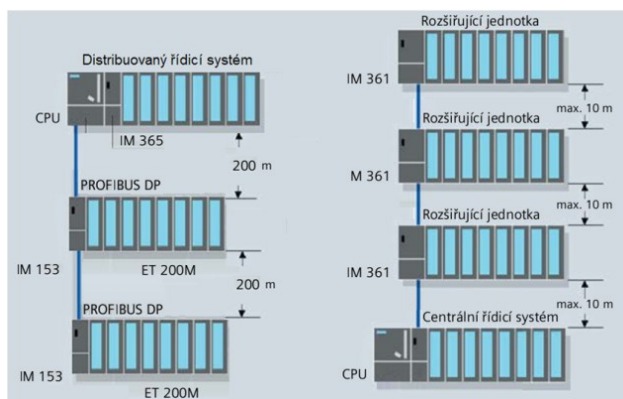
Komunikace mezi perifériemi je zajištěna průmyslovou sběrnici Profibus, a pro monitorování řídicího systému a ovládaní systému operátorem pak Ethernet a jeho modifikací Profinet, která rezervuje část komunikačního kanálu pro Real-Time zprávy.

### 3.1 PLC Siemens Simatic S7 – 400

SIMATIC S7-400 je jedním z nejrozšířenějších řídicích systémů od společnosti Siemens. Využívá se po celém světě a používá se hlavně u rozsáhlých a náročných automatizačních úloh, například právě v hutním průmyslu. Nejnovější CPU, jsou vybaveny rozhraním Profibus a Profinet pro řešení distribuovaného konceptu řízení, ale stále jsou vhodné i pro centralizovaný systém.

U malých aplikací se používá centralizovaný koncept řízení – tzn. 1 CPU a max. 8 modulů vstupů a výstupů ve stejné jednotce – „Rack“. V případech, kdy je potřeba přidat více vstupů, výstupů, čítačů nebo např. modulů pro komunikaci po sběrnici „ModBus“, je možné centralizovaný systém rozšířit až na 4 jednotky s 8 moduly v každé z nich.

Komunikace mezi jednotkami je zajištěna přípojovacími moduly, které komunikují do vzdálenosti maximálně 10 m. Maximální počet takto rozšířeného systému je 1024 I/O. Vyžaduje-li rozlehlost řízené aplikace větší vzdálenosti, je vhodné zvolit distribuovaný koncept. Moduly vstupů a výstupů jsou vsazeny do rozšiřujícího modulu IM (např. IM-153) – vzniká modul ET200s nebo ET200M. S CPU jsou ET200 připojeny přes síťové rozhraní Profibus nebo Profinet. Maximální počet I/O při použití rozhraní Profibus je až 65 536. [Standard CPUs, ©2017]



**Obr. 3.1 Distribuovaný a rozšířený centralizovaný koncept řídicího systému [Koziorek, 2011]**

Systémovou vlastností tohoto automatu je redundantní režim. Je to možnost mít vedle sebe zapojené dvě stejné PLC, z nichž jeden je master a každý program nahaný do něj se automaticky kopíruje i do druhého PLC. V případě poruchy hlavního PLC, převezme řízení druhé PLC, aniž by došlo k nějakému výpadku a řízená technologie se tak nemusí zastavovat.

Velkou výhodou tohoto typu PLC je pak možnost změny konfigurace za chodu systému. Je pravda, že ne úplně každá změna lze provést, ale pokud je systém kompletně zapojen v redundantním režimu včetně modulů, lze vyměnit nebo přidat například právě modul, nebo jakákoliv změna v programu lze aplikovat taktéž bez přerušení systému. Takto je možné pružně reagovat na změny v procesu a snadno jej optimalizovat.



**Obr. 3.2 PLC S7-400 [Standard CPUs, ©2017]**

### 3.1.1 ET 200M

Decentrální periferie SIMATIC ET 200M je modulární vstupně/výstupní stanice skládající se z modulů, které mají možnost až 64 digitálních kanálů. Tato periferie využívá komunikačního rozhraní Profibus nebo Profinet. Lze ji rozšířit až o 12 vstupně/výstupních

modulů. ET 200M je vhodná pro specifické a komplexní automatizační úkoly. Použité moduly mohou být různého typu. Použitím systému SIMATIC S7 400 lze připojovat, měnit nebo konfigurovat moduly za provozu systému bez nutnosti vypnutí řídicího systému.



Obr. 3.3 Modul ET 200M [Standard CPUs, ©2017]

### 3.1.2 Parametry komponentů PLC

- SIMATIC S7-400 CPU 414-3 PN/DP:
  - 4 MB pracovní paměti RAM, z toho 680 K pro instrukce, 2 MB pro data a 2 MB pro program
  - Bit memory 8 KB, counters 2048, timers 2048
  - Interface Ethernet / Profinet – 2 porty, profibus - 2 porty
  - 65 536 DI/DO, 4096 AI/AO
  - Doba zpracování instrukcí: Bit 0.045 $\mu$ s; word 0.045 $\mu$ s, fixed-point 0.045 $\mu$ s floating-point 0.135 $\mu$ s
- Digitální vstupy SM 321 – 32 DI, 24 V
  - Napájení: 20.4 – 28.8 V
  - Optické galvanické oddělení
  - HW přerušení – Ne
  - Diagnostické funkce – Ne
  - Max. napájecí proud pro dvoudrátové senzory: 1.5 mA
- Digitální výstupy SM 322 – 32 DO, 0.5 A
  - Napájení: 20.4 – 28.8 V
  - Optické galvanické oddělení
  - Diagnostické funkce – Ne
  - Rezistivita zátěže 48  $\Omega$  – 4 k $\Omega$
- Analogové vstupy SM 331 – 8 AI
  - Měření s termočlánky – hodnota teploty: 10 t [ $^{\circ}$ C]
  - Připojitelné termometry: Cu10, Ni100, Ni1000, Ni120, Ni200, Ni500, Pt100, Pt1000, Pt200, Pt500

- Další připojitelné rezistivity 0 – 150  $\Omega$ , 0 – 300  $\Omega$ , 0 – 600  $\Omega$
- Parametrizace linearizace – Ano
- Diagnostické funkce – Ano
- Max. napájecí proud pro 2 drátové senzory: 1.5mA
- Analogový výstupní modul SM332 - 4 AO
  - výstupní modul SM332 má 4 analogové výstupy x 12 Bit
  - U, I
  - přední konektor 20 PIN



Obr. 3.4 Modul SM 331 [Standard CPUs, ©2017]

Kapitola čerpána z [Standard CPUs, ©2017].

### 3.2 Profibus

Neboli Process Field Bus je nejrozšířenější sběrnici řídicích systémů v automatizaci. Používá se pro nižší a střední výkony komunikačních přenosů informací mezi perifériemi. Je určena pro všechny oblasti automatizace. Vychází z otevřeného komunikačního modelu ISO/OSI. Vytváří komunikační rozhraní pro připojení malých pasivních zařízení i výkonných automatizačních zařízení PLC, operátorských panelů, řídicích jednotek pohonů atd. Její otevřenost je garantována normou EN 50170. Hlavním důvodem takového rozšíření v automatizaci je nejen množství výrobců a komponent, které tento komunikační standart podporují a využívají.

Profibus prošel od svého vzniku několika vývoji, aby splňoval požadavky současných standardů a norem pro řídicí systémy. Dnes je zastoupen v těchto modifikacích:

- **Profibus - DP** – je určen pro komunikaci na nejnižší úrovni řídicího systému mezi distribuovanými vstupy a výstupy, senzory a pohony. Nositelem informace a komunikačním médiem je kroucená dvojlinka nebo optické vlákno.

Předností této modifikace Profibusu je rychlá odezva a rychlost přenosu informací. Je však určen pro malé objemy dat.

- **Profibus - FMS** – slouží jako standard pro přenášení velkých objemů dat mezi jednotlivými řídicími celky, SCADA systémy, či vizualizačním zařízením operátora. Je určen pro práci v heterogenním prostředí (výměna dat mezi různými výrobci), kde není kladen důraz na rychlost odezvy. V dnešní době je již nahrazován Ethernetovým rozhraním.
- **Profibus - PA** – Je rozšířením Profibusu-DP pro prostředí s nebezpečím výbuchu. K zamezení jiskřivosti je použita pro přenos informací proudová smyčka podle standardu IEC 1158-2.

V mnoha aplikacích komunikují všechny zařízení s PLC metodou Master-Slave, kdy komunikaci řídí CPU v PLC. Pouze usměrňující jednotky ALM si navzájem vyměňují informaci o rozdělení zatížení napájení stejnosměrného meziobvodu měničů. Tato komunikace je přímá Slave-Slave a dosahuje rychlosti odezvy < 2ms. Ostatní zařízení na Profibusu komunikují pouze s PLC podle definovaného interface. V našem případě se pro komunikaci s moduly ET 200M využívá standardu Profibus-DP. Kapitola čerpána z [Koziorek, 2011]

### 3.3 Ethernet

Rozhraní Ethernet vzniklo v 70. letech 20. Století v laboratořích firmy Xerox. Uplatňuje se zde systém náhodného přístupu ke komunikačnímu médiu s detekcí kolize. Ethernet byl postupně standardizován organizací IEEE a normou ISO 8802-3 do nynější podoby

a modifikací, např.:

- IEEE 802.3a 10Base2 – přenos po tenkém koaxiálním kabelu
- IEEE 802.3i 10BaseT – kroucená dvojlinka a zapojení v topologii „Hvězda“
- IEEE 802.3ae 10GBase – optické vlákno
- IEEE 802.3af – využití napájení zařízení po Ethernetu

První Ethernet byl tvořen koaxiálním kabelem, který byl 90. letech 20. století nahrazen krouceným párem, což umožnilo vytváření rozsáhlejších sítí s hvězdicovou topologií. Následně byl vyvinut „Switch“, který umožnil další rozšíření a spojování sítě. V roce 2002 přichází plně duplexní režim 10Gb/s Ethernetu s použitím optických vláken.

V dnešní době je Ethernet již rozšířen i do automatizačních aplikací řídicích systémů. Oproti klasickému komerčnímu Ethernetu a sítím Internet je deterministický. Průmyslový Ethernet je používán pro monitorování řídicího systému nebo jeho ovládání operátorem, popř. v modifikaci Profinet, která rezervuje část komunikačního kanálu pro Real-Time zprávy. Zavedením Standardu Profinet RT a Profinet IRT pro časově kritické procesy se Profinet v dnešní době používá i pro nejnižší vrstvu řídicího systému. Pro distribuované vstupy a výstupy, senzory nebo pro komunikaci s řídicími jednotkami pohonů. [Koziorek, 2011]



## 4 Konfigurační a vizualizační prostředí

Nedílnou součástí projektu je i softwarové prostředí, sloužící pro komfortní obsluhu a sledování parametrů řídicího systému (vizualizační prostředí), nebo jednoduchá, či zjednodušená aplikace programu do systému.

### 4.1 PLC program

V době, kdy byly vyvíjeny první programovatelné automaty, bylo programování PLC pro jednoduchost co nejvíce přiblíženo tvorbě reléových schémat a řešení booleovských rovnic, které přetrvalo dlouhou dobu. Následně se začal s vývojem hardware PLC objevovat jazyk funkčních bloků pro programovatelné automaty.

V 90. letech 20. století zahájila mezinárodní elektrotechnická komise IEC (International Electrotechnical Commission) vývoj standardu neutrálního programování PLC pod názvem IEC 1131 (později IEC 61131). Standard je složen z 8 částí, které definují:

- Definice pojmů a vlastností PLC
- Požadavky na technické vybavení PLC
- Programovací jazyky
- Analýza výběru PLC
- Komunikace a integrace do sítí
- Bezpečnostní funkce – Safety
- Programování Fuzzy logic
- Pokyny pro implementaci programovacích jazyků

Standard „IEC 61131–3 Programovací jazyky“ pojednává o systémově neutrálním programování s následujícími vlastnostmi:

- Nezávislost na technické vybavení – s možností přenesení programu na PLC jiného výrobce
- Možnost ladění SW v rané fázi návrhu
- Jednotný programátorský přístup
- Strukturovanost a modularita

Standard zahrnuje všechny nejobvyklejší programovací jazyky:

- Grafické:
  - **LD** – Ladder diagram
  - **FBD** – Function block diagram
  - **SFC** – Sequential function chart
- Algebraické:
  - **ST** – Structured text
  - **IL** – Instruction list

V mém případě jsem nejdříve používal jazyk LD, ale velmi rychle jsem si osvojil jazyk FBD, především pro jeho přehlednost a také kvůli standardu, kterým se firma Ingeteam a.s. řídí a který využívá především jazyk FBD a IL. Čerpáno z [Zezulka, 2010].

## 4.2 *Simatic Step7*

Simatic Step 7 je vývojové prostředí firmy Siemens pro programování PLC z rodiny Simatic Siemens. Obsahuje mnoho nástrojů a funkcí pro vytvoření ucelené řídicí aplikace. Podporuje všechny zmíněné programovací jazyky standardu IEC 61131–3.

Obsahuje 4 základní editory:

- **Simatic Manager** – je hlavním správcem celého projektu. Slouží pro přístup do jednotlivých částí programu, datových bloků a konfigurace CPU jednotek, pohonů a vizualizací implementovaných v projektu. Definuje rozhraní programátorského PC/PG pro komunikaci s reálným zařízením a obsahuje mnoho nástrojů pro diagnostikování připojeného PLC. Uživatelé mají možnost využít simulační aplikaci a testovat základní funkce bez reálného připojení k PLC. Ze Simatic Manageru se spouští ostatní programovací nástroje pro tvorbu programu a konfigurace řídicího systému.
- **Hardware konfigurační nástroj** – Pokud je v projektu přítomna stanice PLC, v hardware konfigurační nástroji je definován typ daného PLC, verze firmware, komunikační moduly, distribuované periférie a další připojená zařízení (např. frekvenční měniče, enkodéry, atd.). Jsou zde nastaveny adresy vstupů a výstupů do PLC, popřípadě komunikační interface. Pro přidávání zařízení do aplikace je k dispozici katalog se všemi dostupnými GSD soubory v PC.

- **NetPro** – Konfigurátor komunikační sítě je vhodným nástrojem pro nastavení síťových propojení jednotlivých periférií. Některé nastavení lze měnit už v Hardware konfiguraci, ale NetPro nabízí širší možnosti a lepší přehlednost nastavení. Rezervuje konektivitu i pro komunikaci s programátorským PC/PG nebo s případným nadřazeným řídicím systémem.
- **Program Editor** – Slouží pro vytváření řídicího programu, definování datových bloků, tvorbu funkčních bloků a jednoduchých funkcí. Je zde tvořeno a vyvíjeno 95 % řídicí aplikace PLC. Tvorbu jednotlivých funkčních částí programu je možné v rozdílných programovacích jazycích. Dokonce v jednom funkčním bloku je možné střídát jazyk FBD a IL. Editor umožňuje testování jednotlivých částí programu se zobrazením aktuálních hodnot v připojeném nebo simulovaném PLC.

Řídicí program pak rozdělujeme na:

- **Organizační Bloky (OBx)** – Jsou hlavní programové části, které jsou spouštěny procesorem PLC ať už cyklicky, acyklicky, na základě události z HW vstupu nebo z události detekování poruchy.
- **Funkce (FCx)** – Pro zpřehlednění programu je možné zabalit složité matematické nebo logické výpočty do funkce. Při vytváření funkce je nutné definovat vstupní parametry, výstupní hodnoty a vytvoří se vnitřní logika a výpočty. Funkce je volána v OB a jsou jí přiřazeny vstupní a výstupní parametry, s kterými pracuje. Pro funkci nedefinujeme žádný datový prostor. Jednu vytvořenou funkci je možno volat několikrát v programu vždy s rozdílnými vstupními daty.
- **Funkční blok (FBx)** – Má obdobné vlastnosti a použití jako FC, s rozdílem definování statického paměťového prostoru – instančního datového bloku. Je ideálním řešením pro obsluhu periférie, která se v technologii několikrát opakuje. Vytvoříme jeden funkční blok, kterému definujeme vždy jiný paměťový prostor a jiná vstupní data. Vytváříme instance.
- **Systémové funkce a funkční bloky (SFCx, SFBx)** – Jsou předdefinované, uzavřené funkce výrobce PLC Siemens, pro usnadnění vývoje řídicí aplikace.
- **Datové bloky (DBx)** – Paměťové prostory rozdělujeme na:
  - Instanční – Paměťový prostor je definován statickou deklarací paměti funkčního bloku, kterému je přiřazen.

- Volné – Datový blok je možné definovat jako uložení dat nebo tabulku hodnot. Rozložení jednotlivých dat a jejich typ je definován přímo programátorem, strukturou UDT nebo kombinací obou.
- **Struktury (UDTx)** – Jedná se o definované rozložení paměti, používané ve funkčních a datových blocích. Struktury jsou většinou generovány z definičních listin. Každý funkční blok, který obsluhuje periférie nebo technologické části má mimo jiné paměťový prostor definován strukturou konstant, strukturou příkazů, vnitřních stavů a alarmů. [Zezulka, 2010]

### 4.3 WinMOD

Pro simulování řízené technologie, testování řídicího software a možnost předvedení zákazníkovi chování celého systému lze použít software „WinMod“ od německé společnosti Mewes & Partner GmbH. Jedná se o softwarovou platformu pro realizaci výrobních a procesních simulací, která je na trhu již od roku 1995. Obsahuje nástroje a moduly pro simulaci a optimalizaci procesů a je používána po celém světě k realizaci provozu různých strojních zařízení, výroby a také k podpoře inženýrství.

Zkrácení doby pro uvádění řídicího systému do provozu je v dnešním globálním trendu automatizace dominantní úlohou. Software WinMod byl vyvinut pro testování real-time řídicího systému s real-time simulací technologických procesů tam, kde je reálné zařízení stále ve výstavbě nebo není připraveno na uvádění do provozu. Hlavní výhodou je možnost vývoje řídicího softwaru připojeného k simulaci a následné připojení k reálnému zařízení bez nutnosti jakýchkoliv úprav, přeadresování nebo přeprogramování.

Pomocí programu WinMOD mohou být testovány všechny možné stavy technologických procesů. Simulovány poruchy, či havarijní stavy technologií, bez nutnosti vyzkoušet tyto stavy na reálném zařízení, nebo přímo v provozu. Vlastními slovy se jedná o simulaci reálného zařízení, čili máme vytvořenou simulaci v prostředí WinMOD, která je napojena na PLC. Nyní si místo PC se simulací představme, že máme v kanceláři zmenšenou variantu reálné vysoké pece, která je řízena stejným PLC. Simulace v prostředí WinMOD tedy dokáže nahradit reálný projekt, technologií, či proces viz schéma na Obr. 4.1.



Obr. 4.1 Schéma uplatnění softwaru WinMod [WinMod, ©2015]

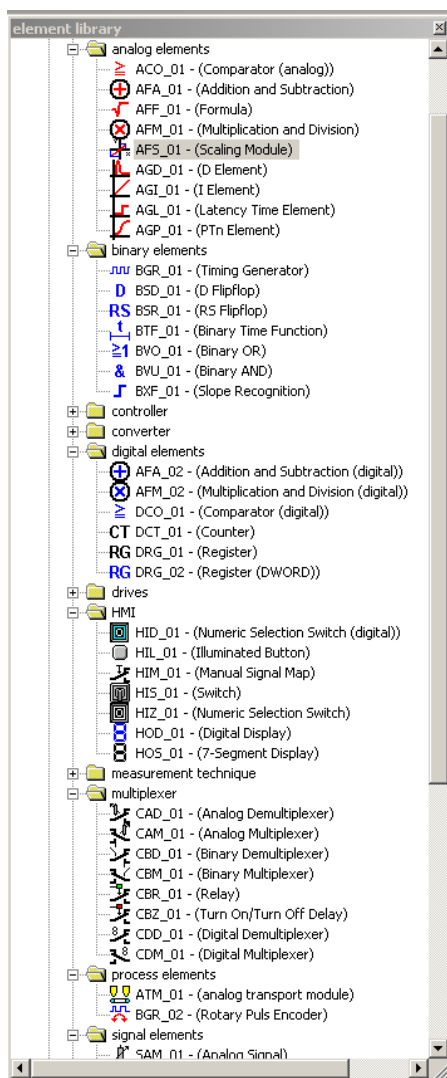
Z PLC jdou reálná data přes sběrnici Profibus do profibusové karty umístěné v simulačním PC. Tato karta pak reálná data převede do softwarového řešení WinMODu a naopak.

Simulační aplikace ve WinMod je tvořena spojením a konfigurací jednoduchých modulů. Funkčnost jednotlivých modulů může být definována matematickými rovnicemi, časovými funkcemi nebo tabulkou naměřených hodnot reálného systému. Jednotlivé moduly a konstanty spojujeme v ucelené bloky tzv. „makra“. [WinMod, ©2015].

Mezi základních funkční elementy aplikace WinMOD patří:

- **Binární elementy** – OR, AND, RS klopný obvod, časovací funkce atd.
- **Digitální elementy** – komparátory, čítače, registry, bloky sčítání, odečítání, násobení a dělení atd.
- **Analogové elementy** – komparátory, integrační a derivační funkce, funkce realizující přechodový děj, funkce sčítání, odečítání násobení a dělení analogových hodnot atd.
- **Elementy konverze datových typů**, multiplexory, demultiplexory, prvky pro ovládací pulty (tlačítka, přepínače, digitální displeje...) a další.

Všechny tyto elementy nalezneme v paletě nástrojů element library viz Obr. 4.2.

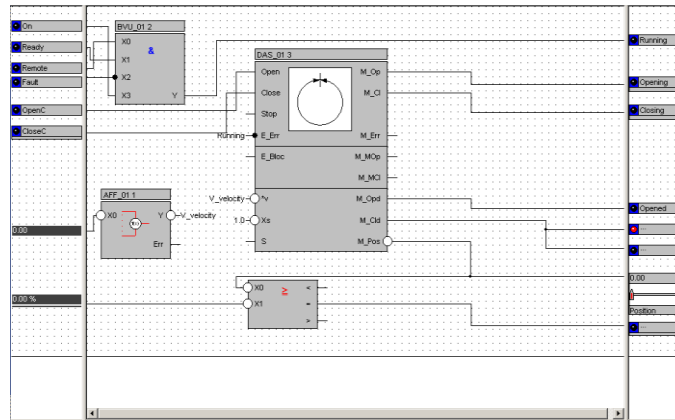


Obr. 4.2 Okno element library

## 4.1 Makra

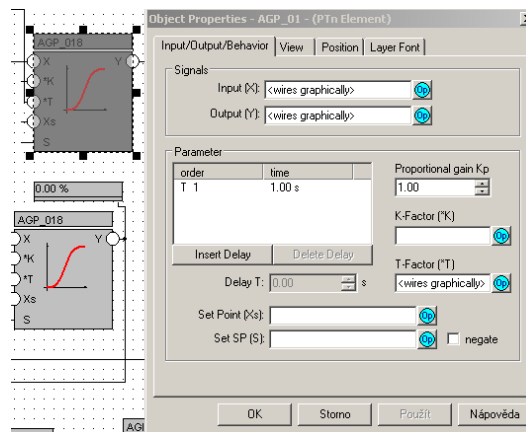
Simulují chování dílčí části technologie např. motor, měření teploty, frekvenční měnič, ventil atd. Uvnitř ukrývají výpočty a celou logiku pro chování technologie. Mohou být vytvořené z předešlých projektů a dají se tedy znovu použít, nebo si je můžeme upravit dle svých představ, popř. vytvořit makra úplně nová.

Makra se vkládají z element library a pomocí pravého tlačítka myši je můžeme rozkliknout a editovat. Jak vypadá takové makro uvnitř, můžeme vidět na Obr. 4.3. Jedná se o jednoduché makro klapky vibračního podavače.



Obr. 4.3 Ukázka makra klapky vibračního podavače

Jednotlivé funkce uvnitř makra se dají ještě dále editovat a různě nastavovat viz Obr. 4.4. V tomhle případě můžeme například měnit proporcionální zesílení  $K_p$ , nastavovat různé časové jednotky apod.

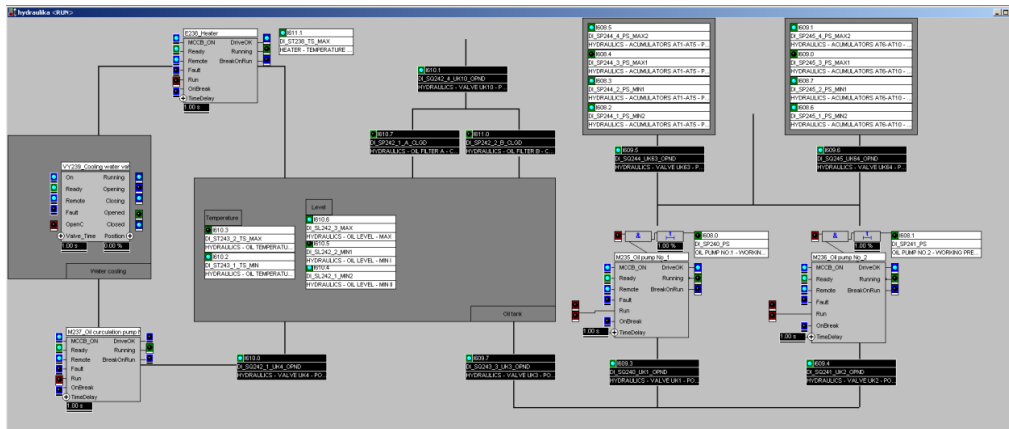


Obr. 4.4 Jednotlivé prvky lze ještě dále různě nastavit, nebo přizpůsobit

Všechna makra ve standardní aplikaci jsou opatřena vstupními signály On, Ready, Remote a Fault. Správná kombinace těchto vstupů dává na výstupu makra signál Running, což je připravenost makra k simulaci.

Seznam standardizovaných maker, které se používají ve firmě Ingeteam a.s. a které jsou použity v této práci můžeme vidět v příloze II.

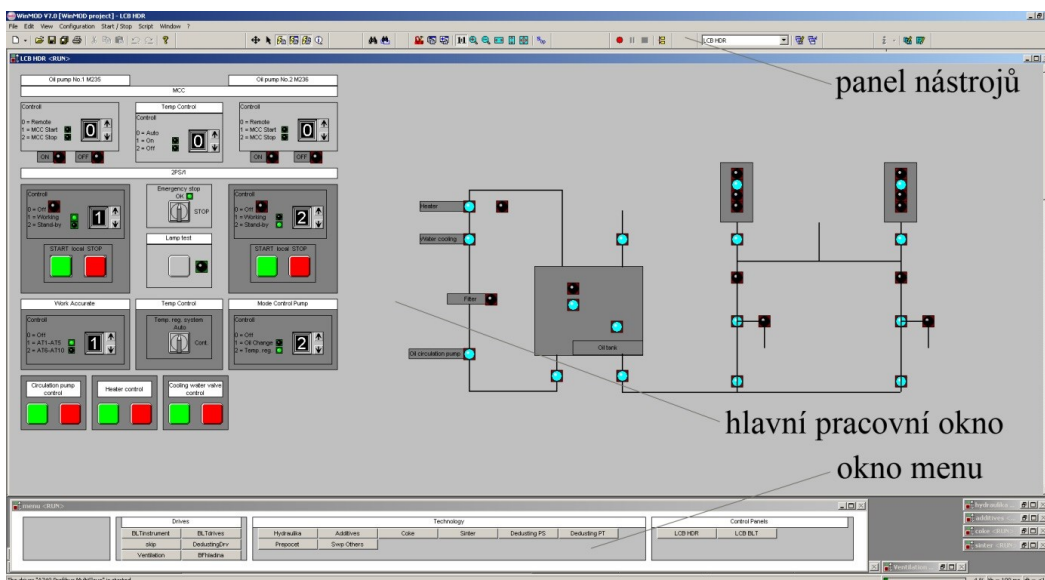
Příklad použití základních výše zmíněných elementů a maker v simulaci je zobrazen na Obr. 4.5.



Obr. 4.5 Příklad použití základních elementů simulace

Pracovní prostředí aplikace WinMOD je celkově ukázáno na Obr. 4.6. Skládá se tedy ze základního panelu nástrojů, přes který se program ovládá. Nachází se zde nejvíce používané prvky, jako spuštění simulace aktuálního okna, spuštění simulace všech oken. Dále je zde tlačítko pro otevření knihovny nástrojů, neboli element library a tlačítka pro práci se vstupně/výstupními hodnotami viz Obr. 4.7.

Pracovní obrazovka je rozdělena na dvě části a to na hlavní pracovní okno a menu. V hlavním pracovním okně se programují jednotlivé technologie - obrazovky. Každá taková obrazovka má pak přidáné svoje tlačítko v okně menu (Obr. 4.8), díky kterému je pak snadné k dané obrazovce okamžitě přistoupit, přepínat mezi jednotlivými okny a mít tak v reálném čase přehled o kompletním stavu simulované technologie.

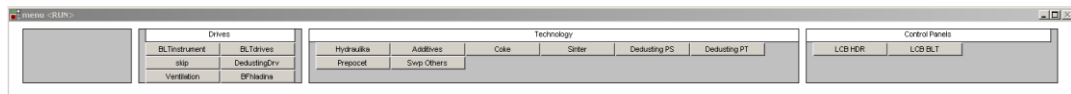


Obr. 4.6 Popis pracovního prostředí aplikace WinMOD



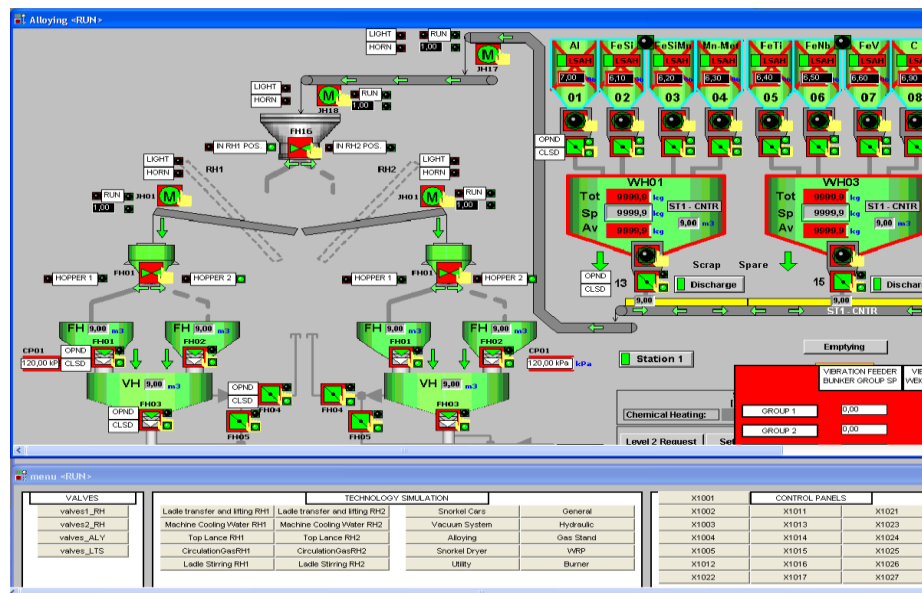


Obr. 4.7 Popis panelu nástrojů

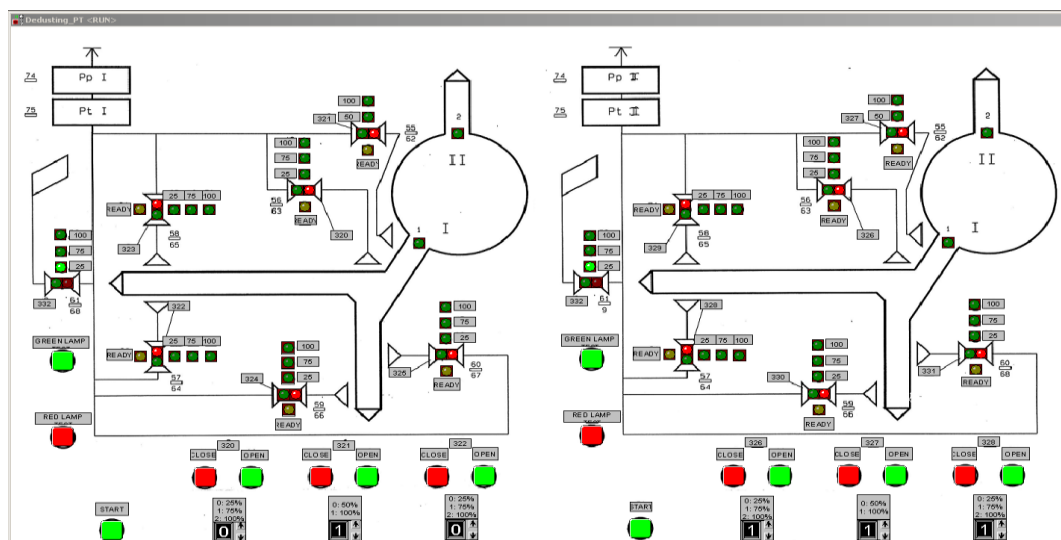


Obr. 4.8 Okno menu

WinMOD dokáže vkládat do svých simulačních oken i pozadí, což může velmi usnadnit pochopení a ovládání simulace dané technologie. Například, když prezentujeme dosavadní výsledky práce zadavateli projektu, či zákazníkovi. Díky pozadí, například z vizualizace, se v simulaci každý lehce zorientuje. Příklady takových simulací jsou zobrazeny na Obr. 4.9 a Obr. 4.10.



Obr. 4.9 Simulace přidávání přísad projektu vakuovacího zařízení pro sekundární metalurgii-  
ilustrační screen [Kocián, 2011]



Obr. 4.10 Simulace odprášení vysoké pece-řídící panel

## 4.2 Archestra IDE

Jednotné výkonné vývojové prostředí, tak by se ve zkratce dala nazvat Archestra IDE (Integrated Development Environment). Jedná se o integrované vývojové prostředí společné pro vývoj aplikací InTouch i pro vývoj řešení na bázi platformy Wonderware System Platform. S jednou sadou konzistentních nástrojů tak mají vývojáři k dispozici nevídaný výkon, flexibilitu a škálovatelnost pro vytváření aplikací od nejmenších jednouživatelských operátorských pracovišť až po rozsáhlá distribuovaná celopodniková řešení.

Funkčně bohaté prostředí Archestra IDE dovoluje mj. i víceuživatelský vývoj stejného projektu díky aplikačnímu serveru, který je součástí. Při sdíleném vývoji toto prostředí zajišťuje kontrolu přístupu k objektům a symbolům a graficky upozorňuje vývojáře na změny, které při sdíleném vytváření aplikace udělali jejich kolegové. Archestra IDE jinými slovy umožňuje automatizované a rychlé uplatnění změn provedených ve vývojovém prostředí (změna vzhledu, chování určitého objektu, přidání oken apod.) do již nasazené aplikace, a to i na více vzdálených uzlů v rámci distribuované síťové aplikace. [Wonderware software, ©2017]

Výhodou prostředí Archestra IDE je vytvoření vlastních objektů s rozšířenou dynamikou, která dříve u prostředí InTouch nebyla možná. Dále jsou přínosné animační vlastnosti pro dynamické zobrazení aktuálních stavů. Vývoj je ve vektorovém zobrazení a samotná vizualizace (Runtime) funguje na InTouchView. Toto prostředí je navrženo přímo pro

---

ArchestrA Industrial Application Server (průmyslový aplikační server). Umožňuje zapojení aplikačního serveru a databáze na jednom počítači. [Wonderware software, ©2017]

### 4.2.1 Aplikační server

Vysoká pec využívá distribučního řízení, kde je řídicí systém rozdělen na několik částí - technologických celků. Každý takový celek je řízen zvlášť samostatně průmyslovým automatem a proto je velmi vhodné je mezi sebou propojit a vizualizovat jako jeden celek. K tomu slouží aplikační server, který je součástí ArchestrA IDE. [Wonderware software, ©2017]

K největším výhodám průmyslového aplikačního serveru patří:

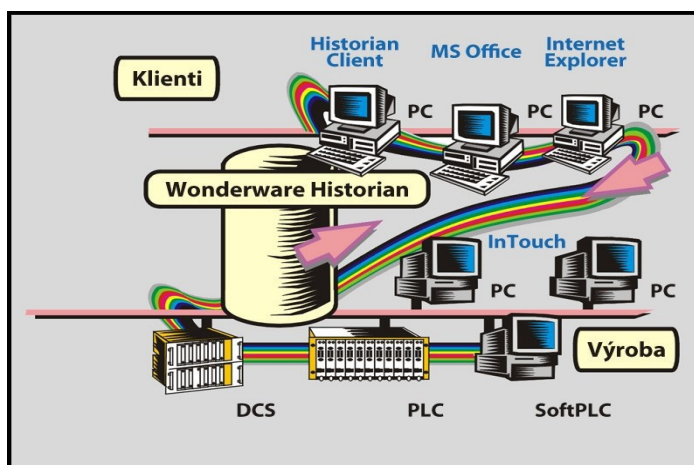
- Robustní komunikace
- Skriptování
- Jednotné zabezpečení
- Centralizované alarmy a události, historizace
- Hierarchicky objektově orientovaný model výrobní technologie
- Flexibilní distribuovaná architektura umožňující škálovatelnost
- Konfigurační prostředí s podporou týmové práce
- Centrální nasazení, správa a diagnostika pro celou aplikaci v rámci jednotného adresního prostoru

### 4.2.2 Historický server

Některá naměřená data je potřeba ukládat. Mezi takové data nejčastěji patří alarmové stavy, a proto bylo nutné do vizualizační sítě připojit Historický server od firmy Wonderware. Tento server se nazývá Wonderware Historian Server a jedná se o relační databázi, která pracuje v reálném čase. Je využívána ve výrobních a řídicích podnicích. Součástí historizační databáze Historian je také Microsoft SQL Server, kde jeho výkonné dotazovací relační jádro poskytuje otevřený univerzální přístup k uloženým datům. [Wonderware software, ©2017]

Wonderware Historian sbírá a archivuje všechna požadovaná procesní data a stavy výrobního průběhu nebo technologických procesů. Historian spolupracuje s aplikačním serverem. Velkou výhodou historického serveru od firmy Wonderware je možnost během

ukládání příliš velkých objemů dat využití kompresi dat, která umožňuje archivaci dat i za dlouhá časová období. Mezi další výhody patří například přístup k uloženým datům z více klientských aplikací, nebo zobrazení historie alarmů, které již byly potvrzeny nebo stále trvají. [Wonderware software, ©2017]



Obr. 4.11 Struktura Wonderware Historian [Wonderware software, ©2017]

## 5 Definiční listina

Jedná se o souhrn XLS souborů, které slouží k popisu řídicího systému a jsou velmi důležité nejen pro tvorbu simulace. Umožňuje vytvořit danou strukturu a udává řád (formát) při tvorbě programu PLC, simulace, ale i vizualizace. Definiční listina je pro každou technologickou část pece vytvořena zvlášť a je i dle části, které se týká pojmenovaná. Například pro bezzvonovou sazebnu je název definiční listiny „Deflist BLT“. Tato definiční listina tedy obsahuje kompletní seznam objektů sazebny, které jsou řízené PLC, určeným právě pro sazebnu.

### 5.1 Deflist BLT

Jak vypadá struktura takové definiční listiny si ukážeme na souboru „Deflist BLT“. Uvnitř listiny jsou definovány názvy a vlastnosti všech objektů, které se v sazebně nacházejí a které jsou řízeny PLC. Objekty se dále dělí dle typu, např. analogové a digitální měření, nebo pohony. Na Obr. 5.1 je ukázka části definiční listiny.

Group	Symbol	Type	Struct	DB	Description
/Bell less top					
BLT	BLT	Group	BLT Main		1100 Bell less top
_BLT	VY201	DevTwoPos	DevTwoPos		2100 Receiving chute
/_BLT	BrdnLvl	Group	BLT_BurdenLevel		1101 Burden level measuring system
/_BLT	M263	ProbeWinch	ProbeWinch		2101 Vertical probe winch No_1
/_BLT	M264	ProbeWinch	ProbeWinch		2102 Vertical probe winch No_2
/_BLT	M265	ProbeWinch	ProbeWinch		2103 Vertical probe winch No_3
/Hopper 1					
BLT	MHP1	Group	BLT_MatHopper		1120 Material hopper 1
_MHP1	MHP1_Instr	Group	Group		1121 Material hopper 1 - Instrumentation
_MHP1_Instr	_SL230_1	DgMeas	DgMeas		2110 MK1 material in the chute above hopper
_MHP1_Instr	_SL230_3	DgMeas	DgMeas		2111 MK1 hopper is full
_MHP1_Instr	_SL230_5	DgMeas	DgMeas		2112 MK1 hopper is empty
_MHP1_Instr	_SL230_7	DgMeas	DgMeas		2113 MK1 material below hopper
_MHP1_Instr	_PT5261	AnMeas	AnMeas		2114 Pressure measurement of MK1
_MHP1_Instr	_PT5262	AnMeas	AnMeas		2115 Differential pressure measurement of MK1 - BF throat
MHP1	MHP1_Vlv	Group	Group		1122 Material hopper 1 - Valves
/_MHP1_Vlv	VY230_1	VlvOneCoil	Vlv		2120 Radioisotope sensor aperture - material pour MK1
/_MHP1_Vlv	VY230_3	VlvOneCoil	Vlv		2121 Radioisotope sensor aperture - MK1 max
/_MHP1_Vlv	VY230_5	VlvOneCoil	Vlv		2122 Radioisotope sensor aperture - MK1 min
/_MHP1_Vlv	VY230_7	VlvOneCoil	Vlv		2123 Radioisotope sensor aperture - material flow MK1
_MHP1_Vlv	_VY210	VlvTwoCoil	Vlv		2124 Upper seal valve of hopper MK1
_MHP1_Vlv	_VY213	VlvTwoCoil	Vlv		2126 Lower seal valve MK1
_MHP1_Vlv	_VY211	VlvTwoCoil	Vlv		2127 Depressurizing valve MK1
_MHP1_Vlv	_VY214	VlvTwoCoil	Vlv		2128 Equalizing valve MK1
_MHP1_Vlv	_VY216	VlvOneCoil	Vlv		2129 Nitrogen filling valve MK1 - secondary equalizing
MHP1	VY212	GenDrive	BLT_MatGate		1123 Lower material gate MK1

Obr. 5.1 Ukázka definiční listiny Deflist\_BLT

Definiční listina se dělí na několik základních částí, rozříděných do sloupců. Patří mezi ně Group, Symbol, Type, DB, Description.

- **Skupina (Group)** - v této části jsou názvy skupin a podskupin, které jsou do sebe vnořeny a vizuálně odděleny do takzvaného stromu. Skupiny nám rozdělují technologickou část daného PLC na několik dílčích částí a jejich názvy

představují zkratku dané části. Uvnitř podskupin se nachází objekty používané v dané části technologie.

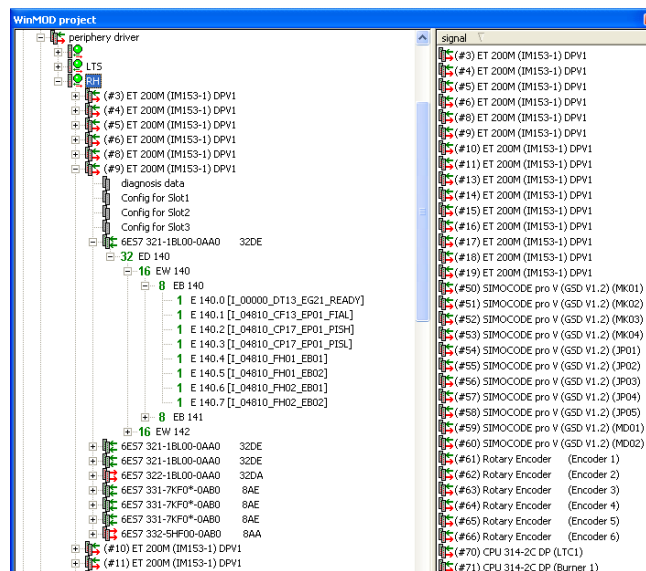
- **Symbol (Symbol)** - představují název objektu v symbolice programu PLC. Skupiny jsou složenina z názvu skupiny nacházející se o úroveň výš a označení dílčí části. Samotné objekty mají individuální názvy, které vznikají na základě dokumentace, nebo vytvořené objekty mají název podle jejich vlastností.
- **Typ (Type)** - každému objektu se přiřazuje typ funkčního bloku, který definuje funkčnost, způsob obsluhy a diagnostikování v programu PLC. Typy se rozdělují podle druhu objektu nebo dle jejich využití.
- **Struktura (Struct)** - struktura definuje vnitřní stavy, poruchy, a rozhraní objektů s vizualizační aplikací objektů. Skládá se z několika uživatelských datových typů, dále jen UDT. Typ struktura se ve většině případů používá pro více objektů.
- **Datový blok (DB)** - tato část udává číslo datového bloku. Každý obsažený objekt, skupina má svůj definovaný paměťový prostor, neboli datový blok, který je vytvořen dle jeho struktury. Tento datový blok obsahuje všechny UDT.
- **Popis (Description)** - popis každého objektu nebo skupiny. Upřesňuje, o jaký druh objektu, nebo technologie se jedná, k čemu se používá a jak funguje. U skupin popisuje chování dílčího technologického celku.

Deflist BLT je rozdělen na hlavní skupinu BLT (bell less top) a pět podskupin. MHP1 (MatHopper 1), MHP2 (MatHopper 2), DstrChute (DistribChute), Nitrogen, HDR1 (Hydraulic). MHP1 a MHP2 mají ještě dvě podskupiny a to MHPx\_Instr (Material hopper-instrumentation), a MHPx\_Vlv (Material hopper-valves). MHPx\_Instr obsahují DgMeas a AnMeas, čili analogové a digitální měření. V podskupině MHPx\_Vlv nalezneme například VlvOneCoil a VlvTwoCoil, čili jednocívkový, nebo dvoucívkový ventil.

Dalšími typy mohou být například VlvHand (ruční ventil), VlvProp (proporcionální ventil), DrvOnOff (pohon zapnuto-vypnuto), VibFeeder (vibrační podavač), Conveyor (dopravník) aj.

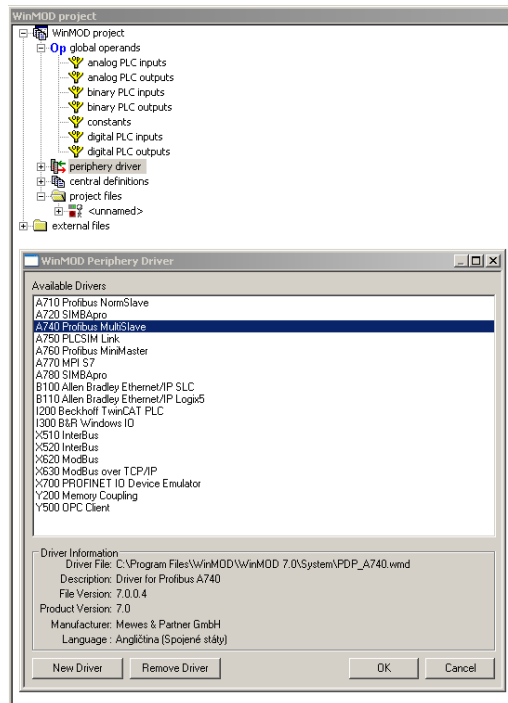
## 6 Simulace

Ještě předtím, než je možné tvořit simulaci, je potřeba nakonfigurovat samotný software, vstupně/výstupní signály a nastavit driver pro Profibus. Konfigurace vstupně/výstupních signálů ve WinMOD vychází z hardwarové konfigurace řídicího systému. Například z hardwarové konfigurace Step 7 firmy Siemens lze vyexportovat „cfg“ soubor konfigurace sběrnice Profibus, který lze následně importovat do WinMODu a vytvořit tím strukturu vstupně/výstupních signálů pro simulaci technologie. Strukturu lze také vytvořit pomocí „gsd“ souborů používaných zařízení, které dodávají výrobci PLC. Příklad importované konfigurace s jednotkami vzdálených vstupů a výstupů na sběrnici Profibus lze vidět na Obr. 6.1. Do takto vytvořené struktury lze importovat seznam symbolů vstupů a výstupů řídicího systému. Příklad adresace (vstupy PLC E140.0 až E140.7) jednotky vzdálených vstupů a výstupů ET200M na Profibus adrese 9 (6ES7-321-1BL00-0AA0). [Kocián, 2011]



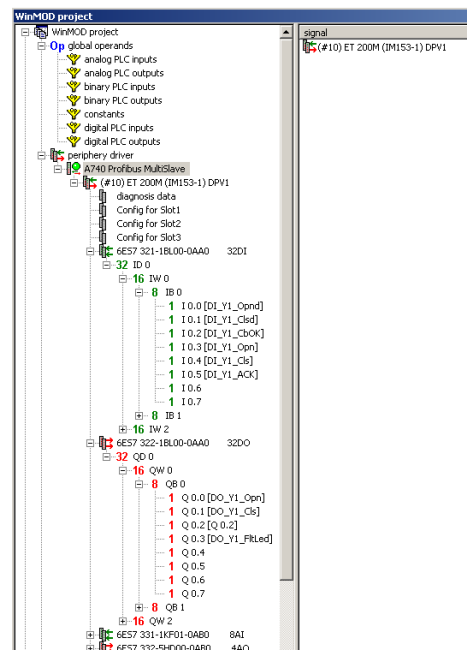
Obr. 6.1 Importovaná HW konfigurace s jednotkami vzdálených vstupů a výstupů [Kocián, 2011]

Konfigurace komunikačního protokolu profibus, po kterém bude komunikace s PLC probíhat je ukázána na Obr. 6.2. Ve WinMOD jsme tedy zvolili komunikační ovladač A740 standardu Profibus Multislave.



Obr. 6.2 Nastavení komunikace

Poté můžeme zkontrolovat všechny jednotky vzdálených vstupů/výstupů (Obr. 6.3). Po nahrání HW konfiguraci a nastavení sběrnice Profibus je simulace téměř připravena.



Obr. 6.3 Kontrola vstupu/výstupu

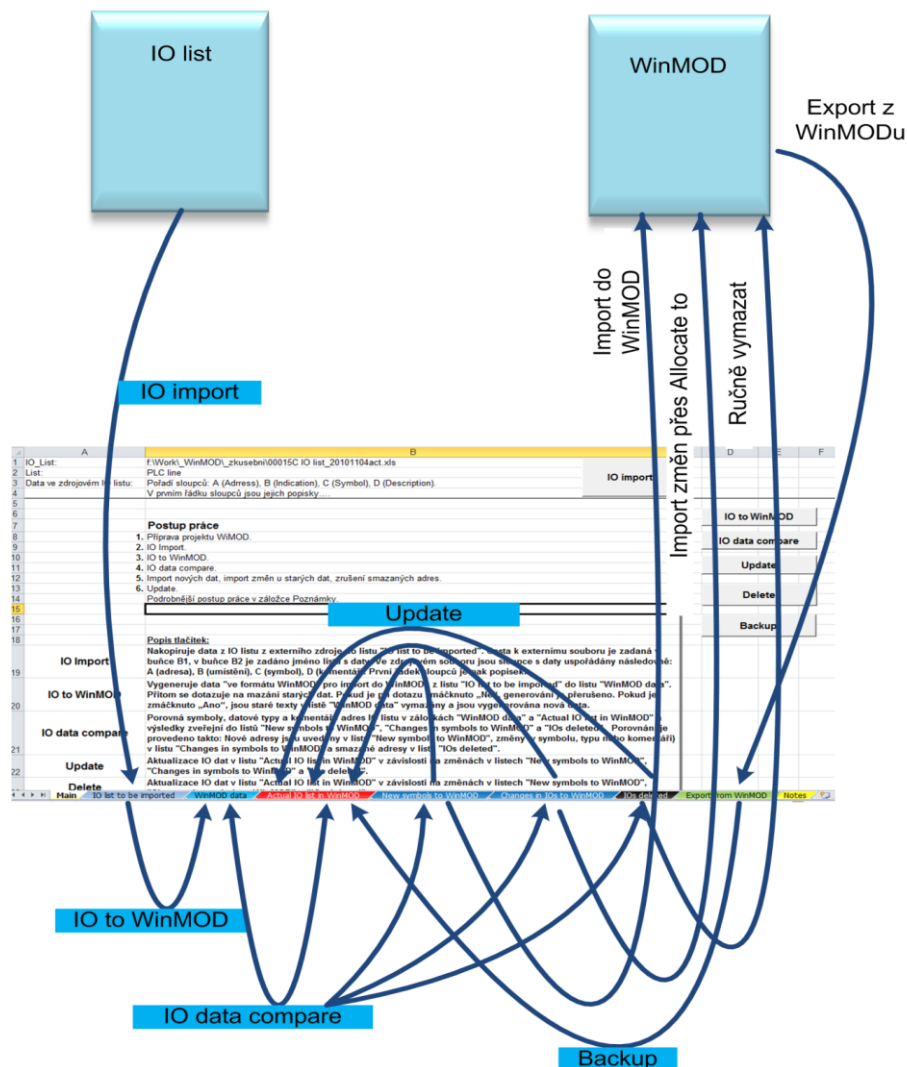
Nyní již můžeme přistoupit k samotnému importu vstupně/výstupních dat.



## 6.1 Import a export vstupně/výstupních dat do aplikace WinMOD

Pro import vstupně/výstupních dat do programu WinMOD jsem používal makry a funkcemi předem definovaný soubor typu XLS, který byl naprogramován tak, aby dokázal generovat data pro import vstupně/výstupních signálů do WinMODu.

Postup práce s excelovským souborem „IOlist\_ToWinMode5.xls“ a importem dat do WinMODu názorně ilustruje Obr. 6.4.



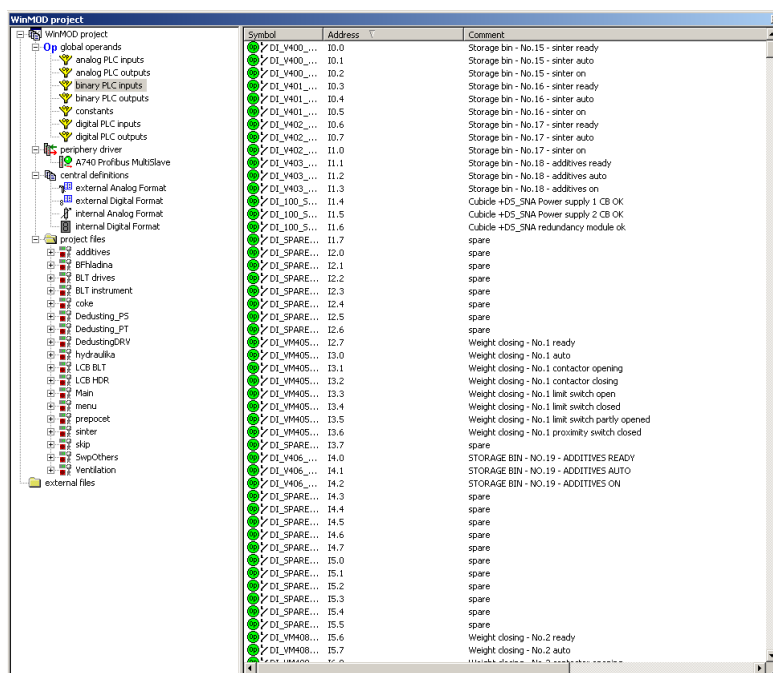
Obr. 6.4 Postup práce při importu dat do WinMODu [Kocián, 2011]

Jelikož se jedná o složitější a rozsáhlý postup, vše je názorně vysvětleno v příloze č. II.

## 6.2 Tvorba simulace

Samotná simulace je tvořena grafickými prvky, jako jsou jednotlivé signály, makra, multiplexory, ovládací prvky, integrační a derivační funkce, bloky sčítání a odečítání. V neposlední řadě pak binární elementy jako funkce OR, AND a mnoho dalších prvků. To vše se skládá a zapojuje pomocí grafického rozhraní aplikace WinMOD.

Jednotlivý prvek vložíme do pracovního prostředí tak, že jej jednoduše přetáhneme myší z element library na pracovní plochu. Pokud se jedná o vložení vstupně/výstupní veličiny, ty nalezneme v okně „WinMOD project“ viz obr. Obr. 6.5. Najdeme požadovanou proměnnou dle pomocného I/O listu (viz Obr. 6.6) a opět jí myší přetáhneme do pracovní plochy. Tím nám vznikne signál s již definovanou adresací.



Obr. 6.5 Nahrané proměnné ve WinMOD

	A	B	C	D
1	Address	Použití	Symbol	Description
2369	I1552.0	ok	DI_W410_RDY	Weight - No.3 ready
2370	I1552.1	ok	DI_W410_WEIGH	Weight - No.3 weighing
2371	I1552.2	ok	DI_W410_BASIC_COND	Weight - No.3 basic condition
2372	I1552.3	ok	DI_W410_EMPT	Weight - No.3 emptying
2373	I1552.4	ok	DI_W410_MIN	Weight - No.3 minimum
2374	I1552.5	ok	DI_W410_MAX	Weight - No.3 maximum
2457	Q1551.0	ok	DO_W410_EMPT	Weight - No.3 start emptying
2458	Q1551.1	ok	DO_W410_WEIGH	Weight - No.3 start weighing
2459	Q1551.2	ok	DO_W410_FINISH	Weight - No.3 weighing finished

Obr. 6.6 Výstřížek z pomocného I/O listu ukazuje, jaké vstupy/výstupy patří k váze č. 410

### **6.3 Provoz přípravy a vážení vysokopecní vsázky**

Po úspěšném importu hardwarové konfigurace a vstupně/výstupních veličin jsem mohl začít pracovat na samotné tvorbě simulace pro jednotlivé technologie. Tak jak se skládá celá technologie výroby surového železa z různých dílčích technologických částí, od provozu přípravy a vážení vsázky, přes velmi důležitou část vysoké pece, tj. skipy či bezzvonovou sazebnu, až po samotné vsypání materiálu do útroh pece, ty samé technologické dílčí celky musí obsahovat i samotná simulace. Na začátku projektu se nejdříve usilovně pracovalo právě na provozu přípravy a vážení vysokopecní vsázky.

Samotný provoz se skládá ze tří hlavních technologických částí, které se zabývají:

- přísadami
- železnou rudou
- koksem

Každá z těchto technologických částí obsahuje zařízení jako vibrační plotny, váhy, pásové dopravníky apod. To vše zakončují skipy, popř. dopravníky, které rozváží jednotlivé materiály dále dle jejich využití.

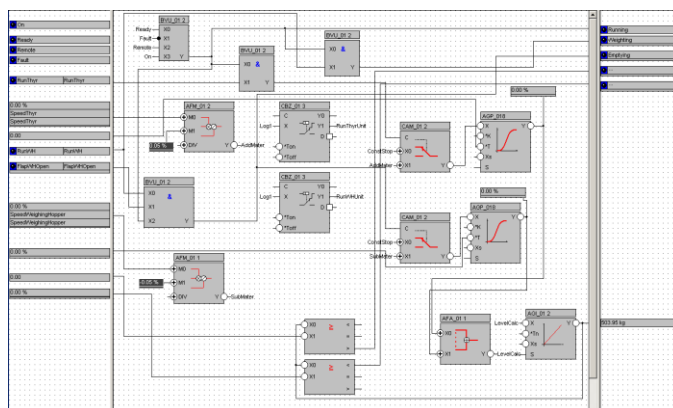
Při programování samotné simulace je potřeba si celou výrobní technologii umět představit, vědět, jak má fungovat a je potřeba úzká spolupráce programátora simulace s programátorem PLC. Simulace provozu přípravy a vážení vsázky celkově obsahuje 74 různých maker.

#### **6.3.1 Přísady**

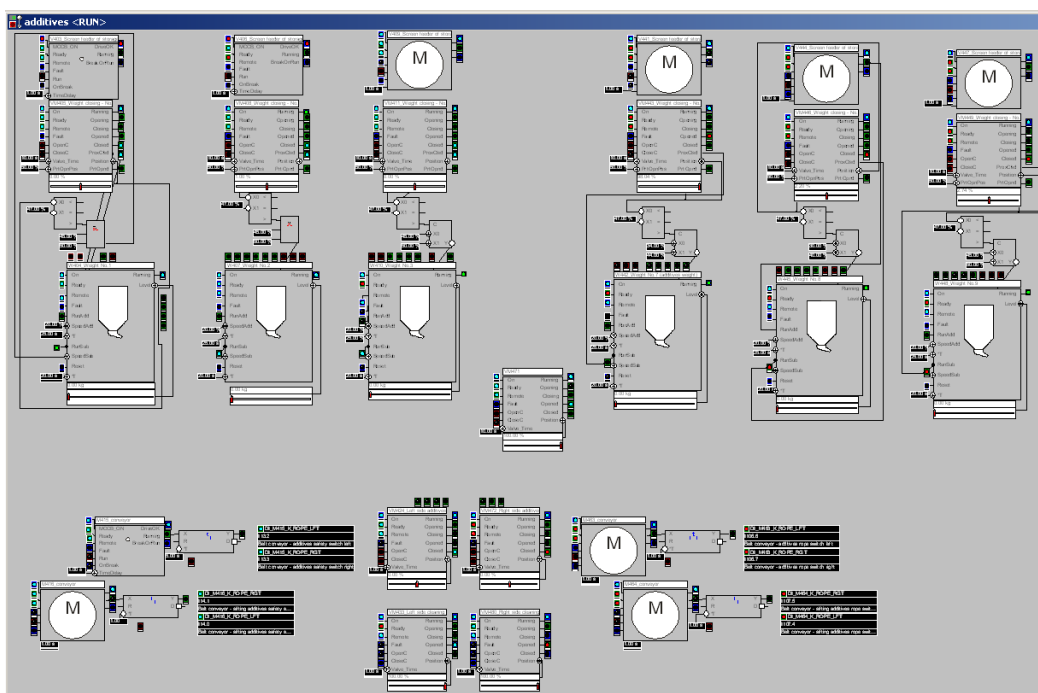
Simulace obsahuje sedmadvacet maker, z toho je deset pohonů, jedenáct ventilů, či klap a šest vah. Na Obr. 6.8 je zobrazen celkový pohled na simulační obrazovku. Nachází se zde (se shora dolů) šest vibračních podavačů a spolu s nimi v přímé návaznosti šest klap pro vysypání materiálu do měřicích vah, které se nachází pod klapami. Klapu můžeme pouze otevřít, či zavřít a vidíme u ní v procentech její polohu otevření/zavření, která určuje, jak rychle se materiál z vibračních podavačů do váhy vysypává.

Jednotlivé váhy jsou řízené PLC a každá z nich má určené maximální přípustné množství přísad. Na Obr. 6.7 je ukázka váhy, která uvnitř makra ukrývá složitější logiku. Jedná se o váhu „w410“, do které se z vibračních podavačů sypou přísady, které se pak přidávají

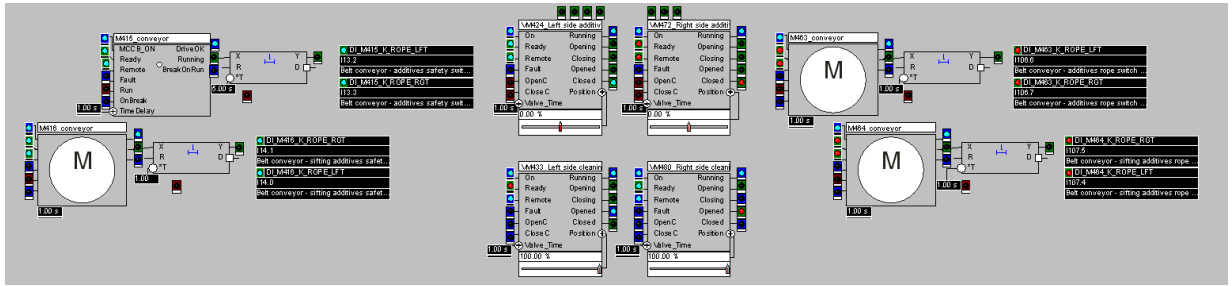
do tavby. Jakmile se dosáhne určité hmotnosti, klapky vibračních podavačů se uzavřou a váha se vysype na dopravník, který materiál dopraví do levé, nebo pravé komory pro přísady, dle jejich aktuálního zaplnění. Makro se skládá například z binárních AND a OR funkcí, analogového násobení a dělení, analogového multiplexoru a v samém závěru logiky se nachází PTn element pro reálnější a plynulejší „dojezd a rozjezd“ analogové veličiny. To vše je zakončeno sečtením analogových veličin a pomocí integrační složky (I elementu), která zaručuje plynulý průběh, vyjádřeno.



Obr. 6.7 Ukázka makra váhy „w410“



Obr. 6.8 Celkový pohled na simulační obrazovku přísad



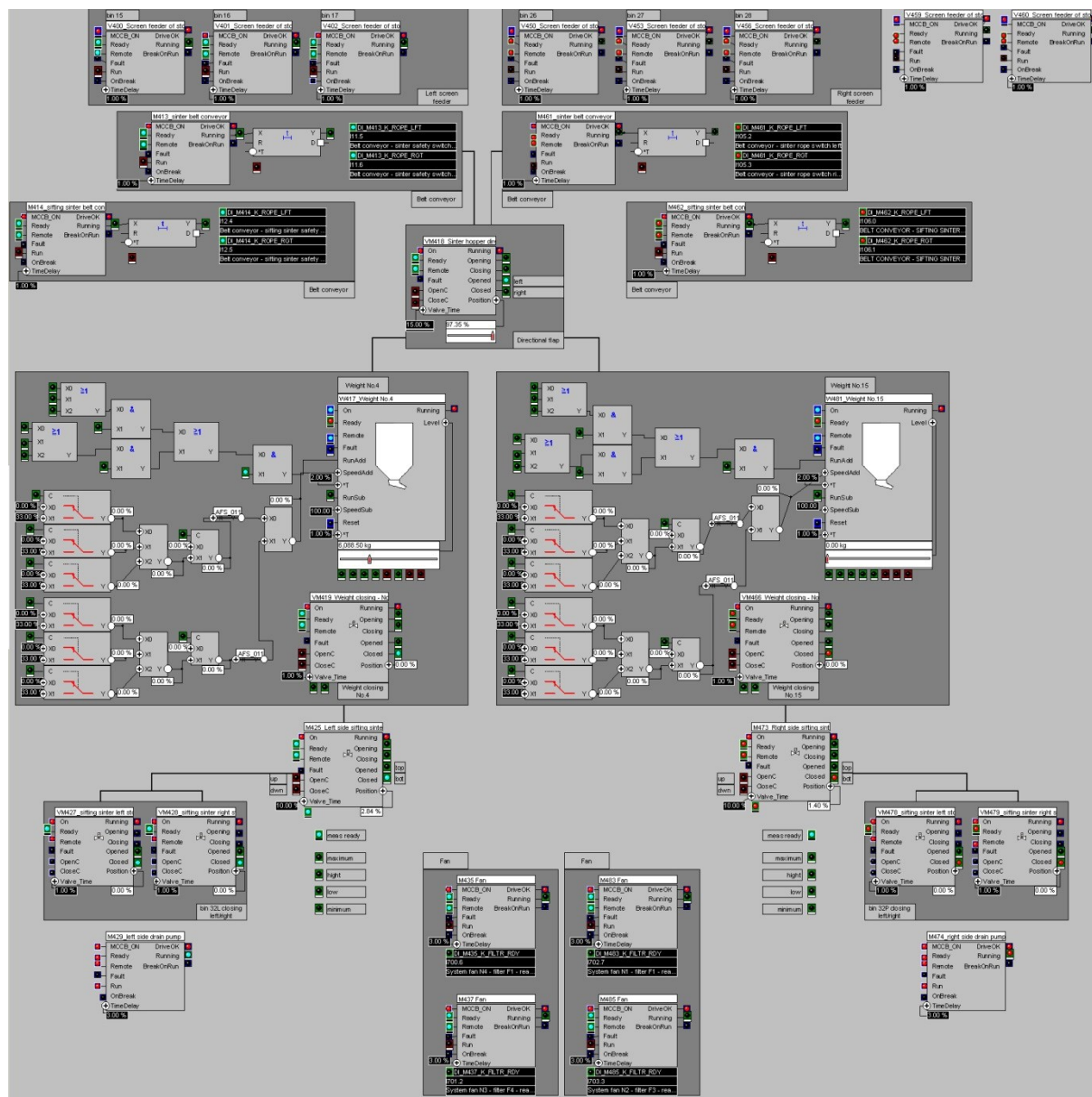
Obr. 6.9 Detail simulace pásových dopravníků pro přísady

### 6.3.2 Zpracování železné rudy

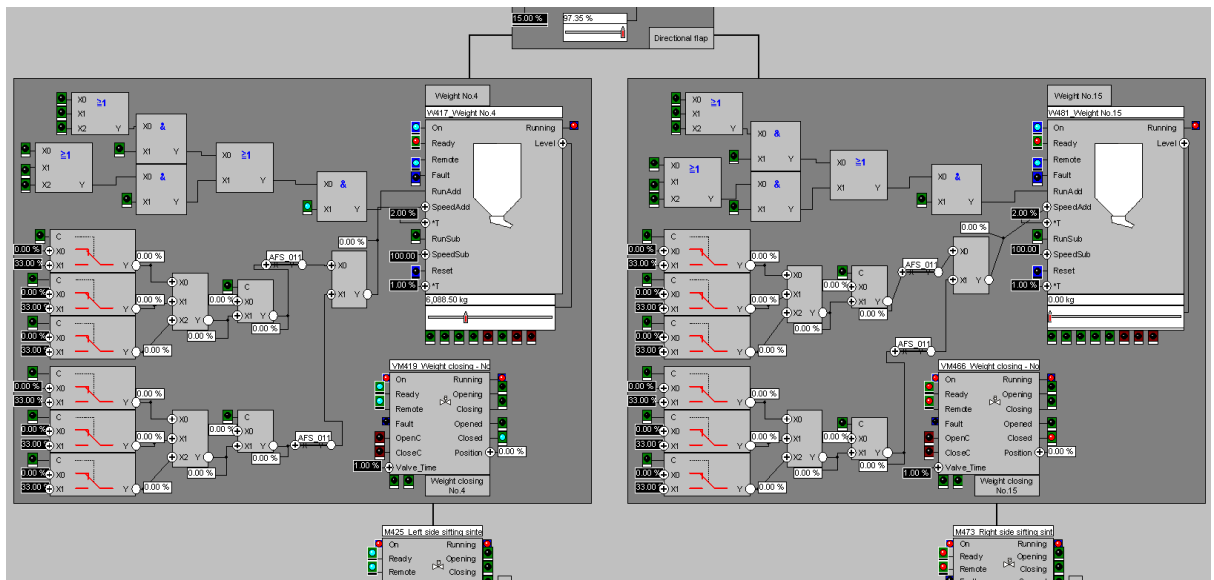
Technologická část zpracování železné rudy se od přísad liší v tom, že jednotlivé vibrační podavače pod sebou nemají žádné kontrolní váhy. Měření materiálu probíhá až v samotných komorách, do kterých se materiál dopraví opět pomocí pásových dopravníků přes směrový rozdělovač. Oproti technologické části s přísadami se zde navíc nachází ještě jeden dopravník (na každé straně jeden) pro jemné částičky rudy, které se tímto dopravníkem dopravují do zvláštní nádoby.

Na simulaci technologie zpracování železné rudy jsou zajímavé samotné komory s váhami. Na Obr. 6.11 můžeme vidět detail těchto komor. Jejich řešení je zvláštní v tom, že kromě naprogramovaného makra samotné váhy je vidět i část logiky mimo makro. Jedná se o samotné sčítání materiálů z jednotlivých podavačů. Přes analogový multiplexor jsou přivedeny signály jednotlivých podavačů a nastaveny hodnoty sčítání v procentech. Pokud je vibrační podavač zapnut, sepne se signál v multiplexoru a tím se začne načítat hodnota 33,3 %. Pokud je podavač vypnutý, signál do multiplexoru je přepnutý automaticky na hodnotu 0 %. Dále se sečtou hodnoty z ostatních multiplexorů. Součet se do váhy promítne jedině v případě, pokud je současně se sčítáním zapnutý pásový dopravník. V opačném případě se výsledný součet do váhy nepropíše. Logika sčítání je rozdělená na pravou a levou stranu, tj. tři podavače s vlastním dopravníkem na straně jedné i na straně druhé.

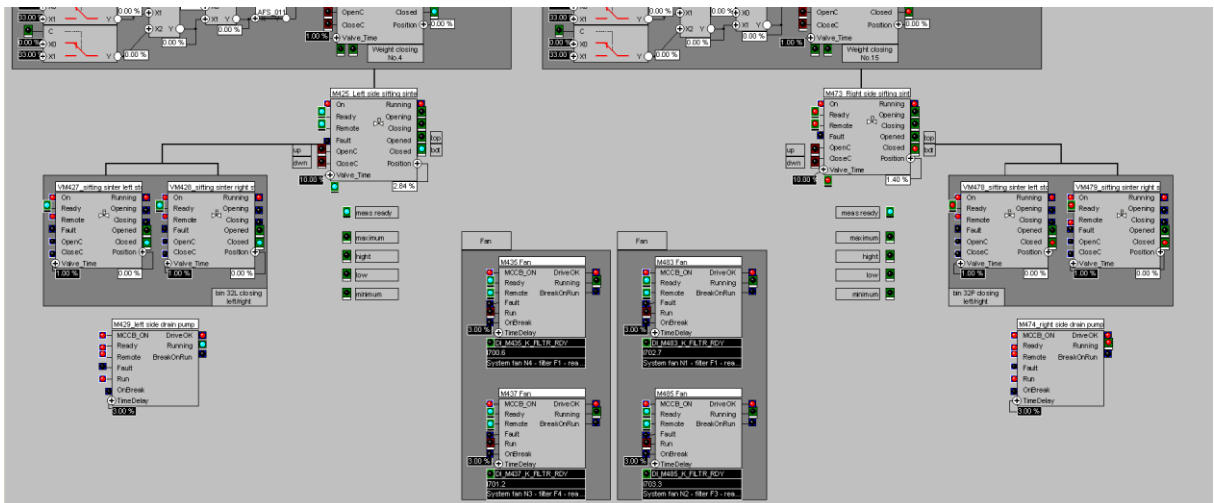
Pod vahami (Obr. 6.12) se nachází ještě simulace klap jednotlivých nádob, ventilátorů a signalizace filtrů pro jemnou rudu, která nesplňuje požadavky pro vysokopecní vsázku. Celkově se v simulaci nachází devětadvacet různých maker. Nejvíce je zde zastoupení maker pohonů, konkrétně se jich zde nachází dvacet. Následuje sedm klap, či ventilů a dvě váhy.



Obr. 6.10 Celkový pohled na simulační okno technologie zpracování železné rudy



Obr. 6.11 Detail výpočtu a simulaci váhy č. 4 a č. 15



Obr. 6.12 Sifting

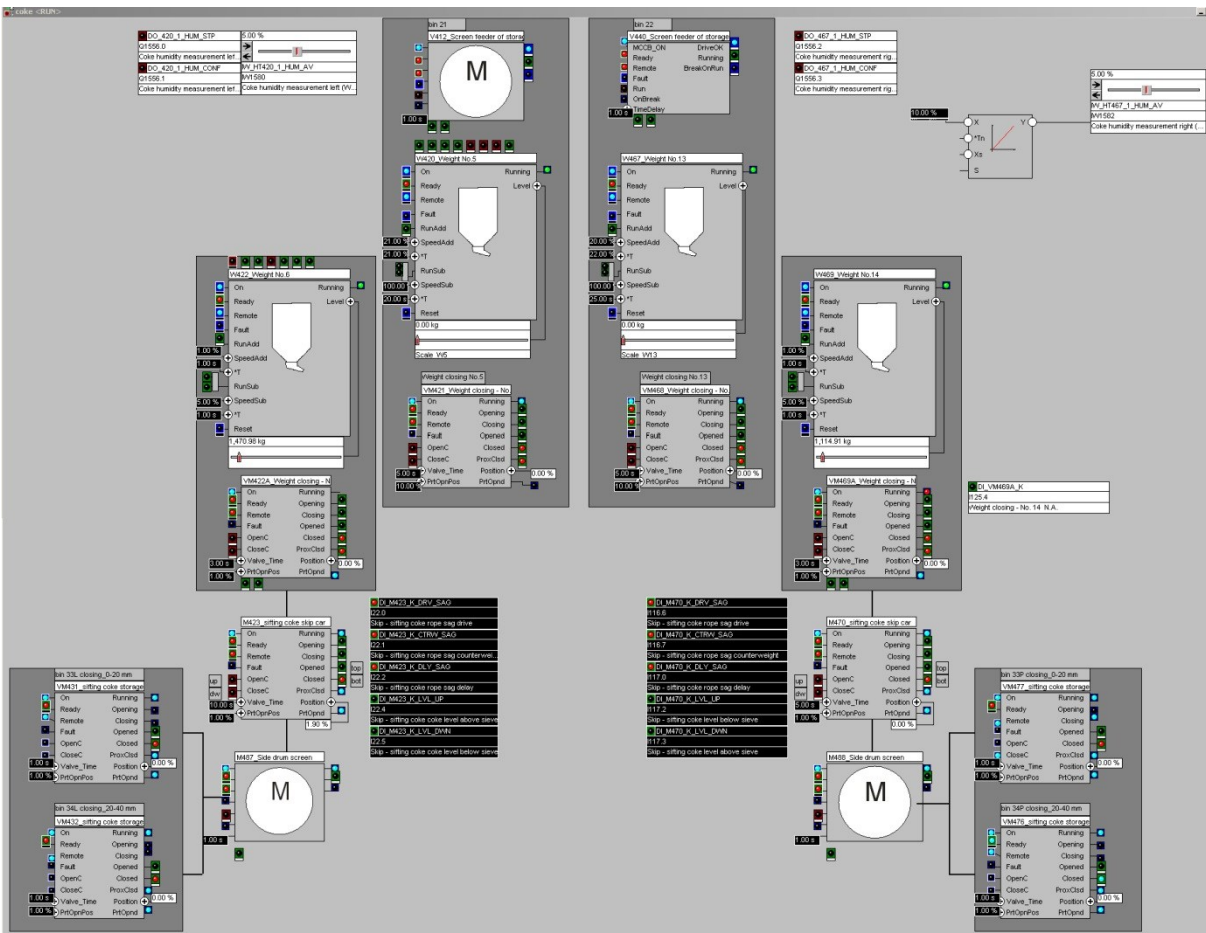
### 6.3.3 Zpracování koksu

Poslední technologickou částí provozu je příprava vsázky koksu. Tak jako u technologie zpracování železné rudy, i zde se zpracování koksu rozděluje na více částí. Na zpracování koksu, který je určený do vsázky vysoké pece a zpracování koksu, který je využit na další účely a rozděluje se dle velikosti zrn. Filtrace zrn nevhodného koksu do vsázky probíhá dle jejich velikosti, a to konkrétně na 0-20 mm a 20-40 mm a to pomocí síta. Takovýto koks je dále využit např. jako tuhé palivo pro vytápění a ohřev vody.



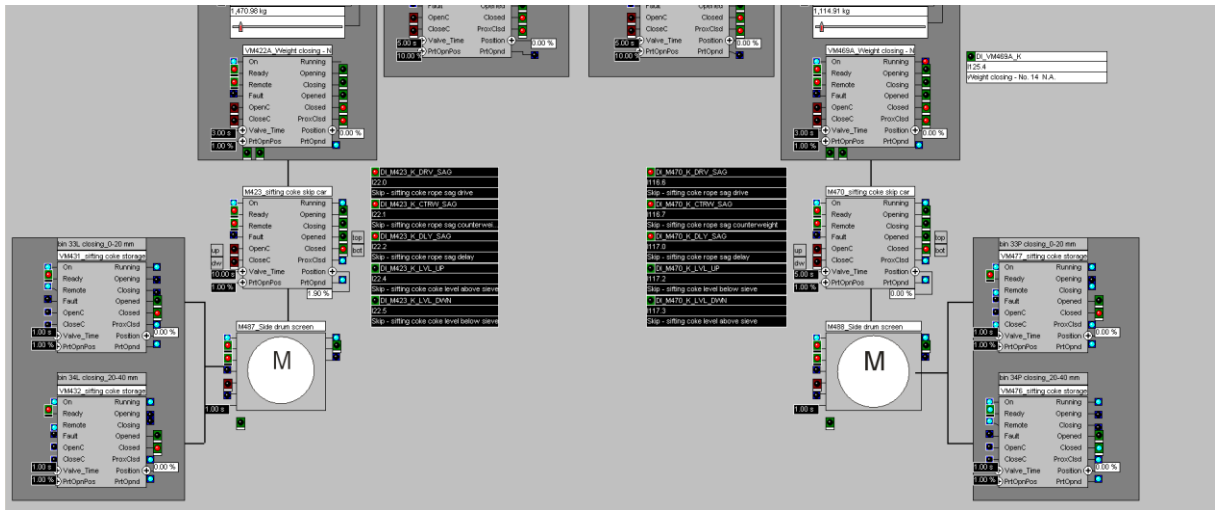
Celou simulaci můžeme vidět na Obr. 6.13. V prostřední části se nachází kontrola vlhkosti koksu, která se nesmí překročit. Dále je zde podavač s váhou a klapou. Po naměření určité hmotnosti se dále koks vysype do hlavního skipového vrátku spolu s přísadami a železnou rudou. Takto naplněný skip předem určeným poměrem přísad, rudy a koksu se následně vysype do sazebný vysoké pece.

V levé a pravé části obrazovky se nachází již zmiňované třídění nevhodného koksu pro vsázku. Každá strana má svou vlastní váhu, která naměří celkovou hmotnost koksu. Obě nádoby mají vlastní skipy, a jakmile jsou plné, vyvezou obsah na třídící síto, které dále rozdělí koks na dvě části dle velikosti zrn, viz poznámka výše. Detail simulace třídění je ukázáno na Obr. 6.14. Dále se zde nachází kontrola prověšení lana skipu, kontrola zpoždění a kontrola polohy skipu (poloha skipu nad sítí a poloha pod sítí), které jsou na obrázku vyobrazeny černými obdélníky, což znamená, že jsou ručně vynucené (tzv. forcing).



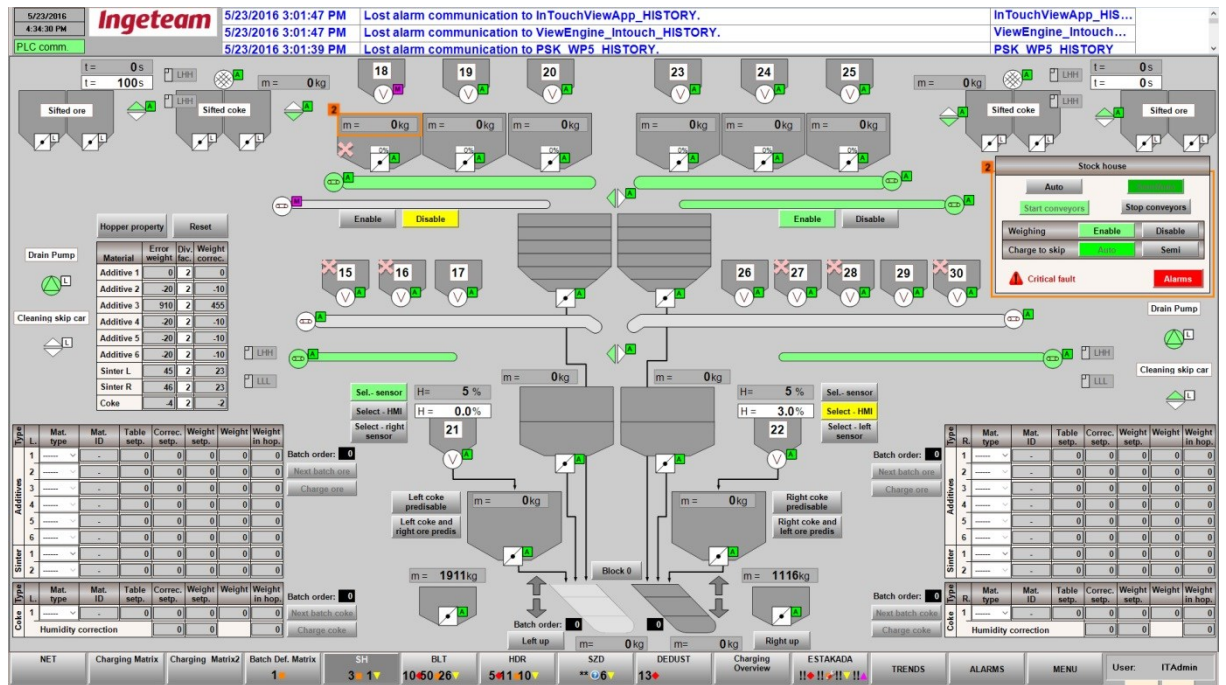
Obr. 6.13 Celkový pohled na simulační obrazovku technologie zpracování koksu





Obr. 6.14 Detail simulace třídění koksu v technologii zpracování koksu

Vizualizaci provozu přípravy a vážení vsázky můžeme vidět na Obr. 6.15. Přísady se nachází ve vrchní části obrazovky (vibrační podavače č. 18 až 25), technologie zpracování železné rudy pak uprostřed (vibrační podavače 15 až 30) a pod nimi technologie zpracování koksu se skipy.



Obr. 6.15 Vizualizace provozu přípravy vsázky

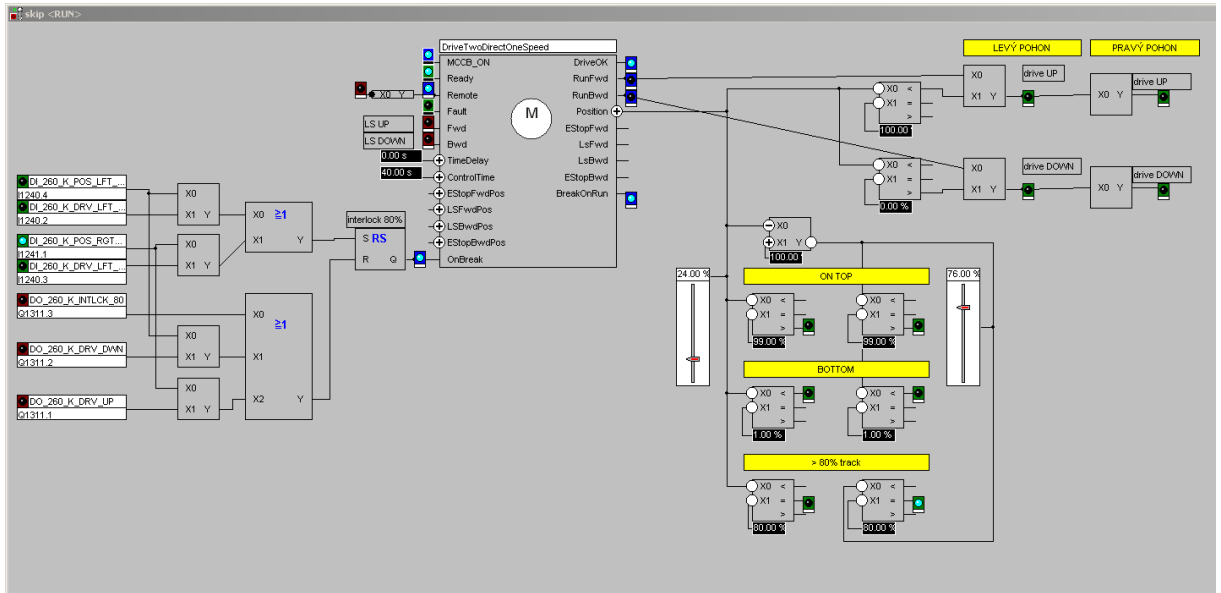
## 6.4 Skip

Skipový vrátek je důležitou součástí výrobního procesu. Je to zařízení spojující provoz přípravy vsázky se samotnou vysokou pecí. Dopravuje vsázku, která se kompletně připravila a navážila, do komor sazební vysoké pece.

Na Obr. 6.16 vidíme simulaci vrátku, kterou jsem rozdělil na čtyři části. Samotné „srdce“ vrátku tvoří jediné makro této simulace a tím je dvousměrný jednorychlostí motor. Bylo tedy použito již nadefinované makro, které bylo jen lehce upraveno pro účely vrátku. Na obrazovce dále můžeme vidět směr pohybu obou vrátků (vpravo nahoře) a kontrolu polohy (vpravo dole).

Směr pohybu je vytvořen spojením signálů RunFwd(vzhůru)/RunBwd(dolů) spolu s analogovou hodnotou pozice levého vrátku. Jakmile je analogová hodnota větší, než 0 %, nebo menší než 100 % u druhé větve, sepne se signál a spolu se signálem RunFwd/RunBwd vyšle signál, který říká, zda levý vrátek stoupá vzhůru, nebo klesá dolů. Pravý vrátek je ošetřen negací, tzn., pokud jede levý vrátek vzhůru, nebo se nachází nahoře, pravý vrátek je automaticky negován a jede směrem dolů, nebo se nachází dole.

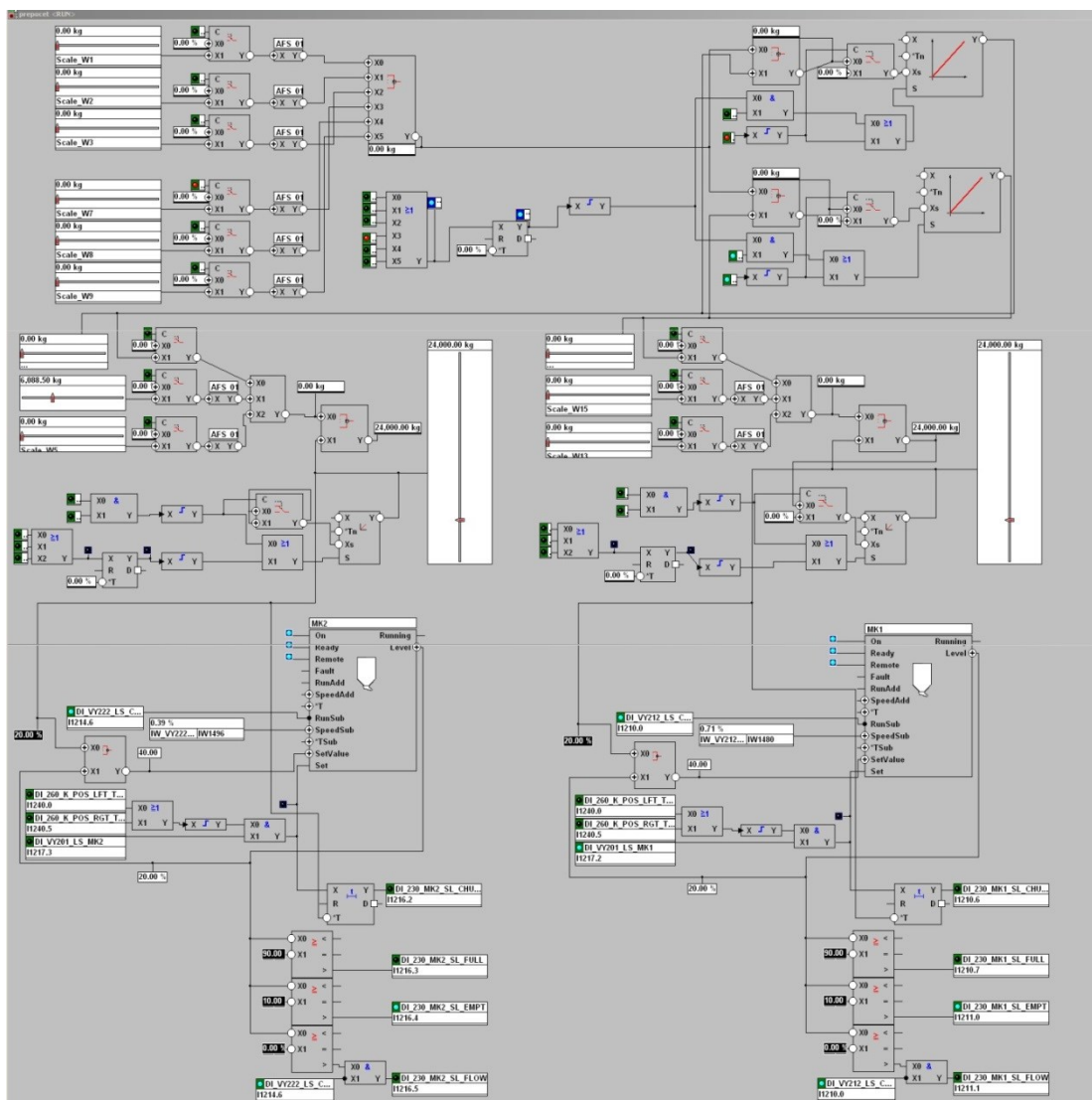
Co se polohy vrátku týče, z makra motoru je klasicky vyvedena analogová veličina pro levý vrátek. Analogový ukazatel pravého vrátku je opět ošetřen negací, čili analogový ukazatel stupnice pravého vrátku bude na opačné straně, jako u vrátku levého viz obrázek. Skip má několik kontrolních poloh opatřenými čidly. Jedná se o přípravnou pozici (80 %) a polohu konečnou nahoře a dole. Přesně tyto čidla máme v simulaci (vpravo dole) a jejich logika spínání je velice jednoduchá. Pomocí analogového komparátoru, který je přiveden na analogovou veličinu pozice určujeme, kdy se má vyslat signál. V případě signálu, že je levý vrátek nahoře v konečné pozici, je komparátor nastaven na porovnání větší než 99 %. Jakmile tedy hodnota pozice překročí 99 %, sepne se signál určující, že skip je na konečné horní pozici. To samé platí v opačném případě, tedy v konečné pozici dole, ale místo nastavení komparátoru větší než 99 %, je nastaven na menší než 1 %. V případě vyslání signálu na přípravnou pozici, tedy 80 % délky směrem vzhůru je komparátor roven hodnotě 80 %.



Obr. 6.16 Naprogramovaná simulace skipového vrátku

## 6.5 Bezzvonová sazebna

Jednoznačně nejobtížnější částí celé mé práce byla tvorba simulace pro bezzvonovou sazebnu vysoké pece. Jednak svým rozsáhlým obsahem, ale i nutností znát technologii sazebny do nejmenších detailů. Jak funguje, jaký je princip sazebny a co se v ní při vsázce děje. Znat všechny ventily a uzávěry a v neposlední řadě je zde navíc realizováno pomocí PLC i měření a řízení tlaku, teploty a mazání převodovky atd. Některé úkoly byly nad mé dosavadní znalosti a možnosti, jako např. kompletní logika přepočtu materiálu převáženého skipy z provozu přípravy a vážení vsázky do sazebny, viz Obr. 6.17. Proto byla téměř celá simulační obrazovka těchto přepočtů realizována odborníky firmy Ingeteam a.s.



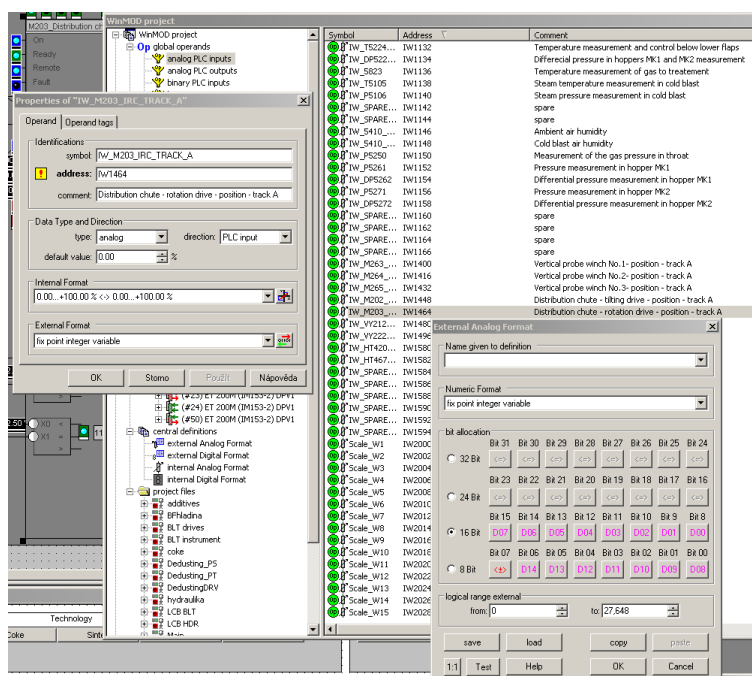
**Obr. 6.17** Přepočítání materiálu přepravovaného skipy z provozu přípravy vsázky do bezzvonové sazební vysoké pece

Simulační okno sazební (Obr. 6.19) se skládá z 24 makr (šest pohonů a osmnáct ventilů/klap) a stejně jako simulaci skipu jsem je rozdělil do několika částí. Konkrétně se jedná o prostřední část, levou a pravou část a část spodní.

V prostřední části se nachází makro dvousměrného motoru pojízdné násypky (tzv. posuvný skluz), který určuje, do které komory se bude sypat vsázka ze skipu. Pod násypkou jsou dvě makra dvousměrného pohonu pro servisní uzávěry, což jsou ve skutečnosti obrovské, několik tun vážící, brýlová šoupátka. Poslední část se týká distribučního žlabu, který je podrobněji popsán v teorii uvedené v příloze č. I. Žlab byl z pohledu tvorby simulace velmi zajímavý, protože díky tomu, že se točí kolem své osy a má ještě možnost měnit úhly sklonu, se muselo vymyslet řešení, jak toto zařízení nasimulovat. Byla tedy

použita dvě makra. První makro bylo jednoduché a obstarávalo naklápění (zdvih) žlabu. Dle projektu měl mít žlab tři výškové pozice. Pozici A (0 %), pozici B (37 %) a pozici C (85 %). Makro mělo na vstupu signály UP a DOWN. V našem případě to tedy znamenalo, že po zapnutí signálu UP se ukazatel pozice posunul z nuly směrem k 100 %. Žlab se ale ve skutečnosti naklápěl do nulové (vodorovné) polohy, tedy do pozice A. Aby se sepnul signál pozice A, odečítá se analogová hodnota z makra od hodnoty 100 a pomocí tří komparátorů, kdy má každý nastaven rovno 0 %, 37 % a 85 % se spínají jednotlivé signály pozic již správně vždy při dosažení těchto hodnot.

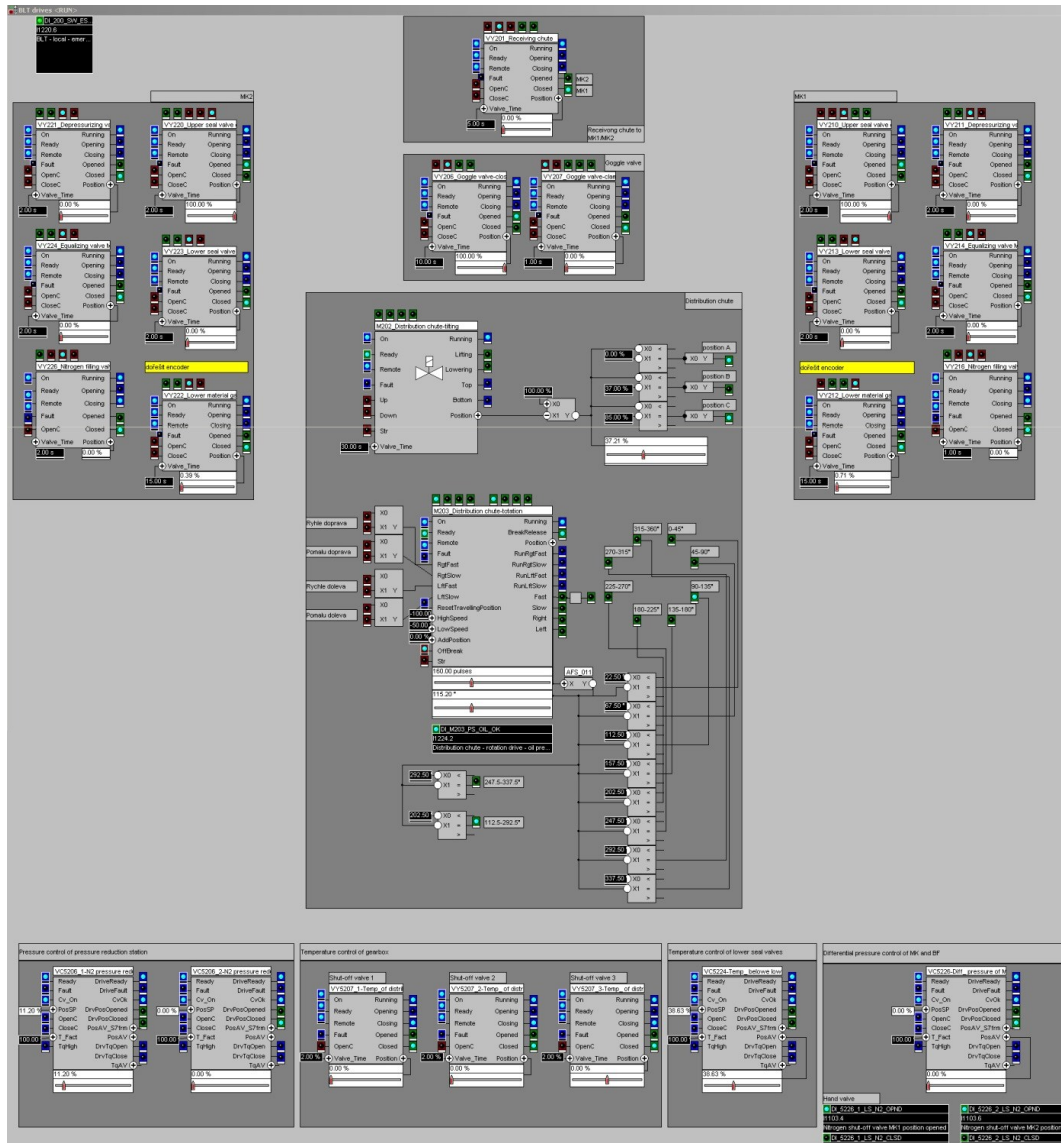
Druhé makro je o poznání složitější a muselo se hodně upravit dle potřeb. Nejen že mělo spínat signály směru doprava a doleva, ale mělo mít i možnost kontroly rychlosti pohybu. Makro se tedy muselo doplnit o tyto nové funkce a navíc bylo doplněno o převodník, který se staral o převod pulsů na stupně. Nastavení IRC je vidět na Obr. 6.18. Právě ukazatel stupňů pro mě byl velmi důležitý a potřebný, protože stejně jako u určování polohy při naklápění, i zde byly signály určující pozici ve stupních. Bylo potřeba připojit celkem osm signálů, každý po 45°. Vytvořil jsem tedy osm komparátorů a ke každému navedl prostřední hodnotu daného rozmezí signálu. V komparátoru jsem nastavil rozsah, kdy má vyslat signál. Tedy pro signál 0°-45° jsem přivedl komparátor s nastavenou hodnotou 22,5° a ve vlastnostech komparátoru jsem nastavil rozsah  $\pm 22,5$ . To znamená, že od hodnoty 0° až do 45° bude komparátor vysílat logickou jedničku a signál 0°- 45° bude po celou dobu sepnutý. Tenhle způsob jsem aplikoval i pro zbylých sedm signálů.



Obr. 6.18 Ukázka nastavení analogového vstupu

V levé a pravé části simulační obrazovky se nachází všechny pohony a signály související s těmito pohony, které jsou obsaženy v pravé (MK1) a levé (MK2) komoře sazebny. Konkrétně se jedná o vrchní a spodní klapové uzávěry, segmentové uzávěry a atmosférický ventil spolu s ventilem vyrovnávacím. Podrobnější popis těchto zařízení je opět popsán v teoretické části přílohy.

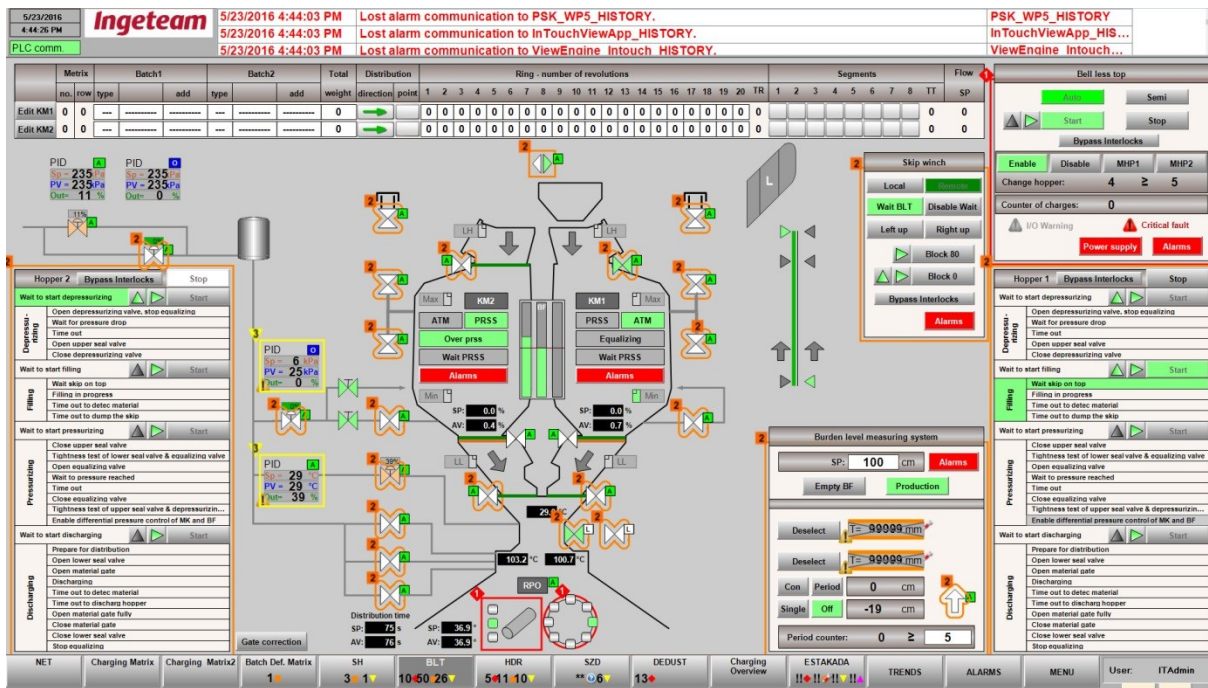
V poslední části simulační obrazovky, tedy ve spodní části se pak nachází regulace tlaku z tlakové redukční stanice, regulace teploty převodovky za pomocí uzavíracích ventilů, řízení teploty spodních těsnících ventilů a regulace rozdílu tlaku mezi oběma komorami MK1/MK2 a pecí s kontrolou otevření, či zavření ventilů.



Obr. 6.19 Celkový pohled na simulační okno bezzvonové sazebny včetně simulace žlabu

Na Obr. 6.20 pak můžeme vidět, jak vypadá vizualizace bezzvonové sazebny a i v tomto případě si můžeme všimnout, že se simulace svým rozvržením velmi podobá vizualizaci. Všechny simulační okna jsem tvořil dle reálného podkladu, dle reálného rozvržení jednotlivých technologických částí a to pro lepší přehlednost a snazší pochopení samotné simulace.

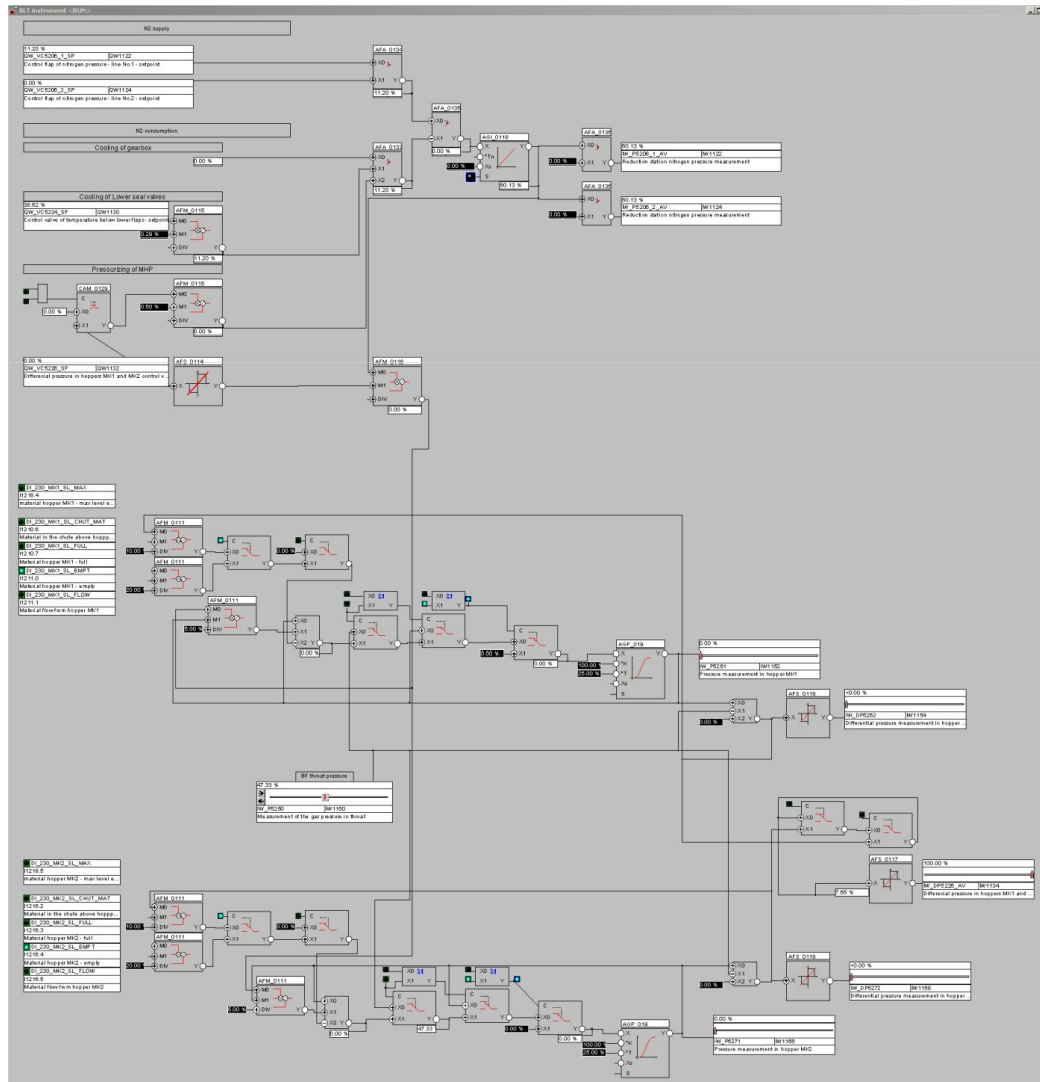




Obr. 6.20 Vizualizace bezzvonové sazebny

Další simulací, která byla vytvořena za pomoci odborníků z firmy je vidět na Obr. 6.21. Jedná se o přepoččet rozdílu tlaku v komorách MK1 a MK2 s tlakem v peci za pomoci dusíku. Po vysypání vsázky do komory a řádnému uzavření a utěsnění vrchní klapky se otevře vyrovnávací ventil na vysokopecní plyn a komora se tlakuje na stejnou hodnotu, jako je v peci. Jakmile je rozdíl tlaků vysokopecního plynu v komoře a v peci nulový, povolí se ventil na přetlakování sazebny dusíkem, čímž se kompenzuje ztráta materiálu při vysypání do pece a čímž se taky zabraňuje průniku vysokopecního plynu z pece do sazebny a do atmosféry.

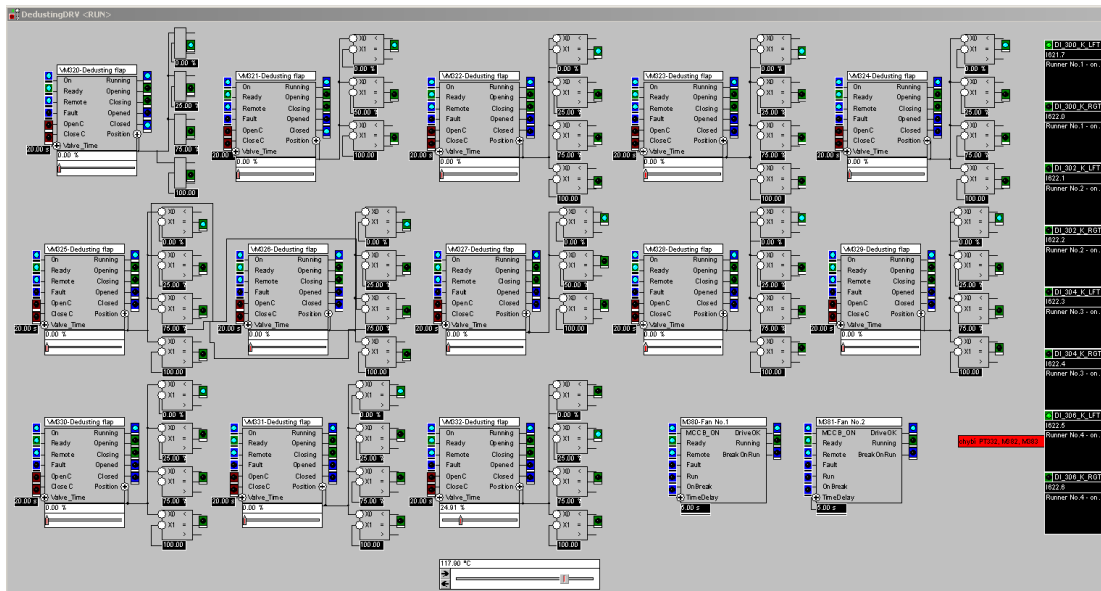




Obr. 6.21 Simulace výpočtu rozdílu tlaku v komorách a v peci

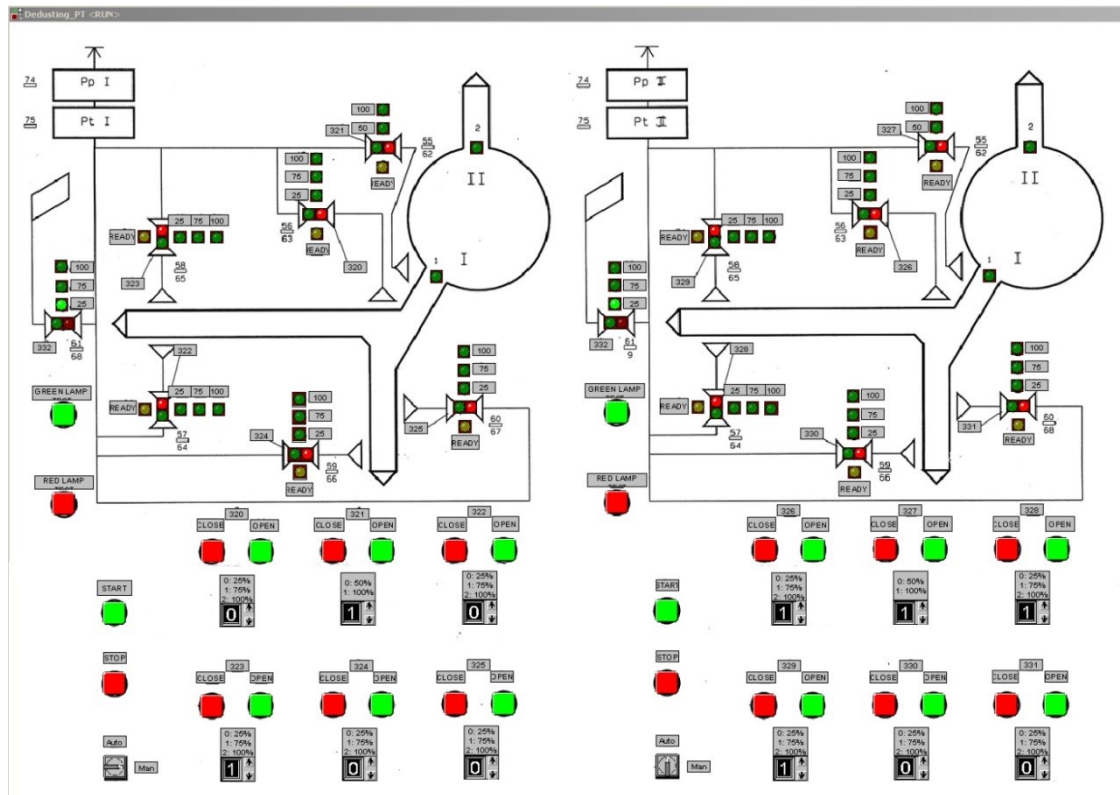
## 6.6 Odprášení

Odprášení patřilo mezi lehčí úkoly. Podle definiční listiny jsem pomocí jednoduchých maker vytvořil třináct klasických dvoucívkových ventilů, což jsou klasické klapky (Obr. 6.22), spolu se signály open/close a opening/closing pro každý ventil. Na výstup z těchto maker, který je analogový, jsem přivedl ještě signály, které se sepnou při určité hodnotě procentuálnosti otevření klapky dle zadání v projektu. Některé ventily mají kontrolu pozice v 0-50-100 %, zbytek ventilů má kontrolu pozic v 0-20-75-100 %. To vše je doplněno o dva ventilátory, které vytváří sací podtlak. Ventilátory jsou vyřešeny pomocí dvou maker jednosměrného motoru.



Obr. 6.22 Simulace technologie odprášení-výkonnová část

Na obrázku Obr. 6.23 je dle reálného panelu graficky znázorněn a nasimulován kontrolní panel odprášení pro operátora. Jeho vyobrazení přesně odpovídá fyzicky reálnému panelu, který se před modernizací používal. V horní části okna se nachází panel s vyobrazením ventilů. U každého ventilu se nachází tři diody (signály). Červeně zbarvený signál zavřeno, zeleně zbarvený signál otevřeno a žlutě zbarvený signál připraveno. Dále se u těchto ventilů ještě nachází již zmiňované kontrolky procentuálního otevření ventilu. Ve spodní části obrazovky je vytvořen ovládací panel všech těchto ventilů s tlačítky otevřít a zavřít a k tomu ještě nastavení hodnoty otevření ventilu v procentech pro každý ventil zvlášť. V praxi to funguje tak, že pokud si u ventilu nastavíme hodnotu 50 % a spustíme ventil tlačítkem open, na simulační obrazovce viz Obr. 6.22 se sepne signál OpenC (červeně zbarvený) na daném ventilu. Rozsvítí se zelený signál Openning a zároveň začne stoupat analogová hodnota. Jakmile tato hodnota dosáhne hodnoty 50 %, PLC automaticky vyšle signál a ventil se zastaví. Rozsvítí se kontrolka vyjadřující hodnotu 50 % jak na simulační obrazovce s ventily (Obr. 6.22), tak na kontrolním panelu u daného ventilu.



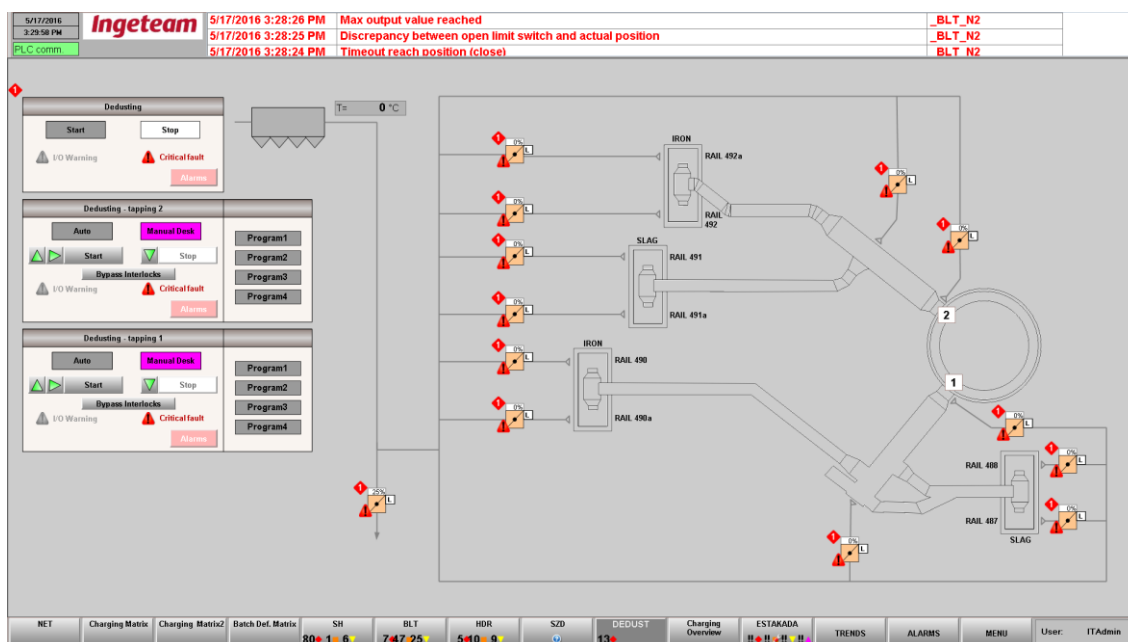
Obr. 6.23 Technologický panel pro operátora

Okamžité zastavení procesu můžeme vyvolat pomocí tlačítka stop na dalším kontrolním panelu, viz Obr. 6.24. Jsou zde umístěny opět kontrolky open, close, ready, včetně ukazatelů polohy a přepínače pro režim auto, nebo manual. Zatímco přechází kontrolní panel se nachází na velině, tento panel slouží operátorovi přímo v provozu.



Obr. 6.24 Operátorský kontrolní panel

Pro srovnání můžeme na Obr. 6.25 vidět jak vypadá vizualizace odprášení. Nalevo se nachází ovládací panely a na zbytku plochy je znázorněna technologie odprášení spolu se všemi třinácti ventily, které jsem popisoval výše.



Obr. 6.25 Vizualizace odprášení

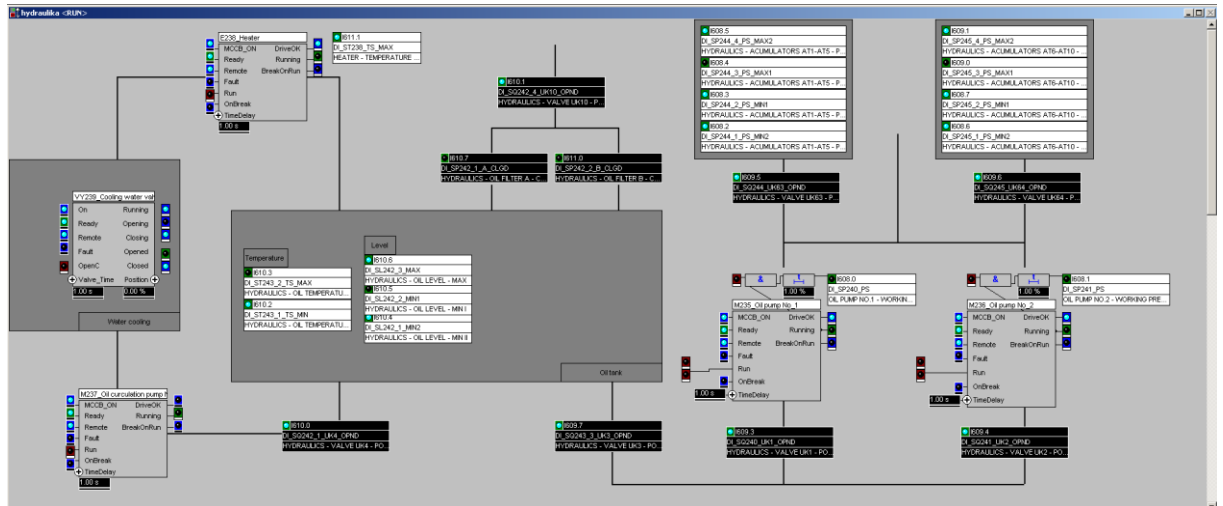
## 6.7 Hydraulika

Žádná moderní bezzvonová sazebná se neobejde bez hydraulického ovládání (využití například pro servisní uzávěru neboli tzv. brýlové šoupátko). Hydraulika je v dnešní době nepostradatelná součást každého většího stroje, každého technologického zařízení. Kromě sazebný má hydraulika využití například i pro hydraulickou odpichovou vrtačku a ucpávačku, pro které jsem taktéž vytvořil simulaci, ale v rámci jiného projektu.

Jelikož se jedná o hydraulický systém vystavený velkému tepelnému zatížení, používá se v takových případech vícerozsahový hydraulický olej bez zinku, který byl vyvinut speciálně pro hydraulické systémy vysokých pecí. Jeho charakteristickou vlastností je velmi nízká závislost viskozity na teplotě. Má dobré schopnosti odlučování vzduchu a deemulgate, nízkou tendenci k pěnění a účinnou ochranu proti korozi a opotřebením. [Hydraulické oleje minerální, bezpopelné a detergentní, 2016]

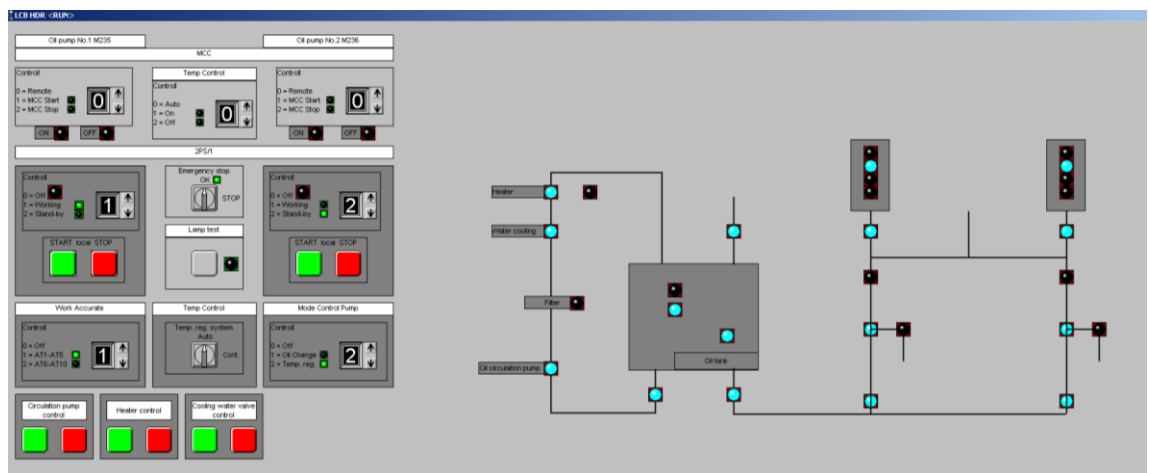
Simulaci hydraulického systému můžeme vidět na Obr. 6.26. Z pohledu simulace se nejedná o nic složitého. Celá hydraulika se skládá ze dvou hydraulických pump (v pravé části dole), hydraulických akumulátorů, které jsou od pump odděleny dvěma ventily (vpravo nahoře). V levé části se pak nachází tzv. olejový okruh. Uprostřed je olejová nádrž s kontrolou teploty a výšky hladiny. Z nádrže jde olej do tzv. heateru, čili ohříváče, který olej zahřívá na určitou teplotu. Dále přes vodní chladicí systém až po cirkulační olejovou pumpu,

kteřá zabezpečuje průtok oleje. Obě olejové pumpy mají signály ukazující, že pracovní tlak je dostatečný pro správnou funkci hydraulického systému. Tyto signály jsou připojeny k signálu running z pumpy přes dopravní zpoždění. Je to z důvodu větší realističnosti chování systému. Po zapnutí pump totiž ve skutečnosti chvíli trvá, než se systém natlakuje na pracovní tlak. Tímto způsobem jsem se snažil vytvořit realistické chování simulace.



Obr. 6.26 Simulace hydrauliky

Ovládací panel hydraulického systému (Obr. 6.27) byl vytvořen přesně dle reálného ovládacího panelu, který je zobrazen na Obr. 6.28. Nachází se zde ovládací prvky (levá část) a signalizace stavu (pravá část).

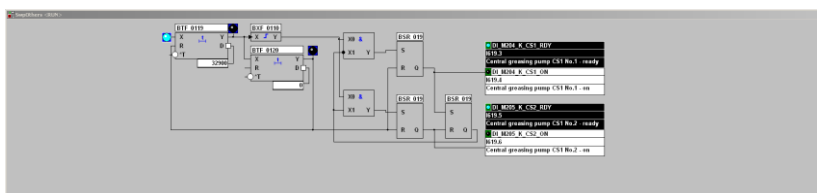


Obr. 6.27 Ovládací panel hydrauliky



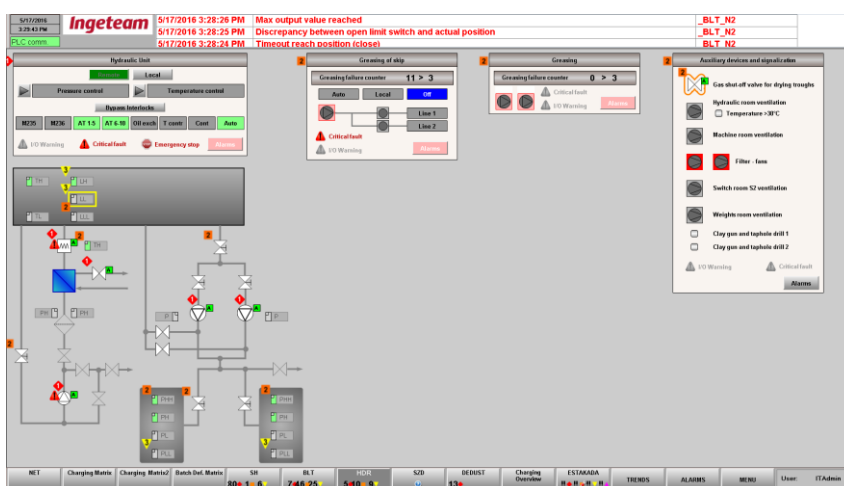
Obr. 6.28 Starý ovládací panel hydrauliky

Mazání technologických částí sazebný, nebo skipu je velmi důležité. Díky složitým pracovním podmínkám, které v oblasti vysoké pece panují (vysoká prašnost, velká teplota) je potřeba častého mazání jednotlivých částí. Mazání tedy probíhá pravidelně v cyklech a je obstaráno dvěma mazacími pumpami. Nasimulování takového cyklu s možností nastavení délky cyklu je realizováno na Obr. 6.29.



Obr. 6.29 Simulace cyklu mazání technologických částí

Na závěr ukázka vizualizace hydrauliky (Obr. 6.30). Rozložení prvků je ve vizualizaci oproti simulaci trochu jiné, ale princip funkčnosti je stejný.



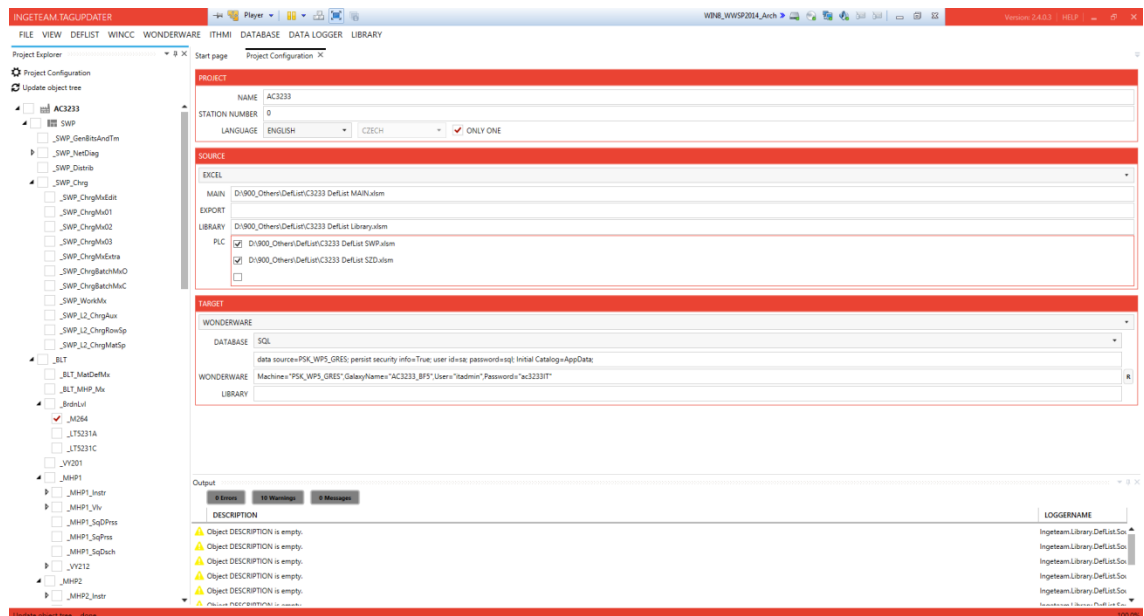
Obr. 6.30 Vizualizace hydrauliky

## 7 Vizualizace

V rámci mé práce jsem se mimo simulaci věnoval částečně i vizualizaci a pomáhal jsem vytvářet dvě vizualizační obrazovky. Konkrétně se jedná o vizualizaci provozu přípravy a vážení vsázky a bezzvonové sazebný vysoké pece. Náhledy finálních podob těchto vizualizací jsou ukázány v kapitolách 6.3.3 a 6.5, kde jsem screeny uváděl pro porovnání rozložení technologických částí simulace s vizualizací. Jedná se tedy o Obr. 6.15 a Obr. 6.20

Vizualizace se skládá z hlavní obrazovky, na které se nachází objekty jako jsou ventily, dopravní pásy, vibrační podavače atd. Po kliknutí na tyto objekty se dále otevře vyskakovací okno, tzv. pop\_up, které se následně otevře v popředí hlavní obrazovky. Pop\_up ve většině případů slouží k ovládání prvku, na který jsme klikli.

Aby se vůbec mohlo začít s realizací vizualizace, je potřeba si předem připravit tzv. proměnné, neboli objekty pro danou strukturu, které se vytváří v prostředí Archestra IDE. Po nadefinování objektů se za pomoci pomocného software, který byl vyvinut firmou Ingeteam a.s., vytvoří instance, které jsou automaticky přiřazené dané struktuře. Následně se může aplikovat jakýkoliv objekt, který je definován v definiční listině, nebo jakákoliv změna v průběhu projektu.

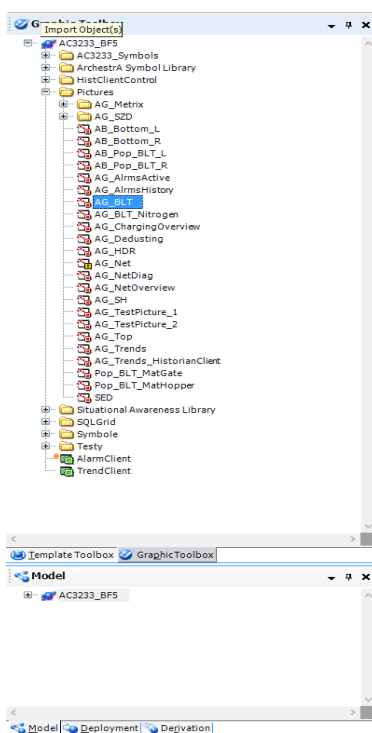


Obr. 7.1 Prostředí TagUpdateru

Důležitým faktem při tvorbě vizualizace jsou určité standardy, které firma Ingeteam a.s. má a kterými se řídí. To znamená, že všechny grafické ikony, či objekty jsou vytvořené dle zavedeného standardu.

Měl jsem možnost podílet se přímo na tvorbě ikon i objektů při přípravě vizualizační části, včetně jejich částečného naskriptování, ale také při tvorbě hlavních obrazovek.

Na Obr. 7.2 je ukázka výběru pracovních obrazovek. V mém případě se jedná o obrazovku AG\_BLT a AG\_SH. Po kliknutí na vybranou položku se otevře pracovní okno, do kterého pak vkládáme předem definované ikony a objekty, popř. vytvoříme vlastní grafické objekty. Po přerušení tvorby a uložení projektu se pak lze tímto způsobem vrátit k rozdělané práci a pokračovat dále.

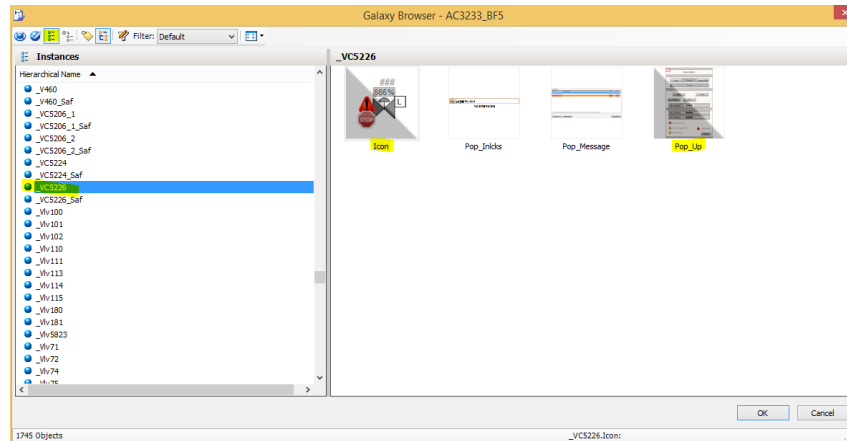


Obr. 7.2 Výběr obrazovek

Objekty a ikony se do obrazovky vkládají výběrem daného prvku ze seznamu dle definiční listiny viz Obr. 7.3. Nejdříve jsem si tedy dle definiční listiny na pracovní plochu vyskládal všechny prvky. Jedná se o pohony, ventily apod. Po konzultaci, jak bude taková obrazovka ve finále vypadat, jsem jednotlivé prvky začal třídit a rozložil jsem je přibližně po pracovní ploše dle nákresu. V druhé fázi jsem vytvořil vlastní grafické prvky. U provozu přípravy vsázky to byly vibrační podavače, váhy, pásové dopravníky a skipy.



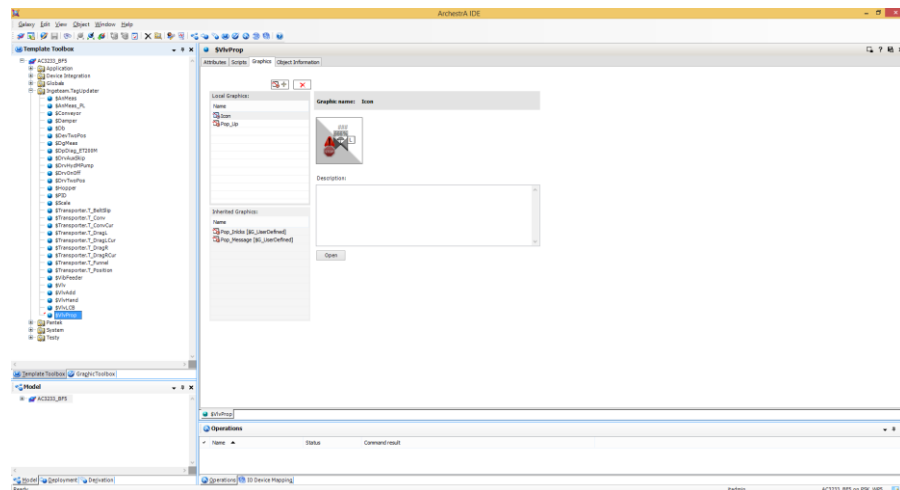
U sazebny se jednalo o nakreslení siluety bezzvonové sazebny, potrubí a násypkový koš. Sazebna je kreslena dle výkresové dokumentace v grafickém editoru Inkscape, a do obrazovky je vložena jako pozadí.



Obr. 7.3 Vložení ikon a objektů

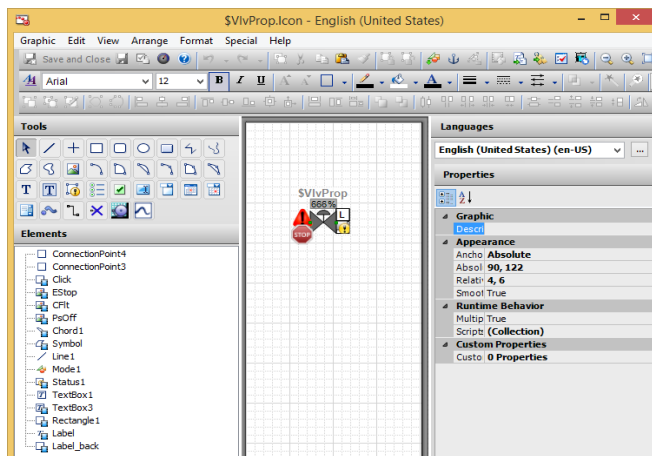
Na Obr. 7.4 jsou vidět v levém panelu všechny vygenerované objekty, např. ruční ventil, vibrační podavač, proporcionální ventil, atd. Právě proporcionální ventil si teď podrobněji popíšeme.

Po kliknutí na ventil se nám zobrazí v pravé části okna na výběr seznam grafických objektů, které se dají editovat (Obr. 7.4). V případě proporcionálního ventilu je na výběr ikona a tzv. pop\_up, tedy okno s ovládacími prvky, které se objeví v popředí po kliknutí na objekt. Po kliknutí na jednu z položek se pak napravo od seznamu zobrazí náhled daného prvku.



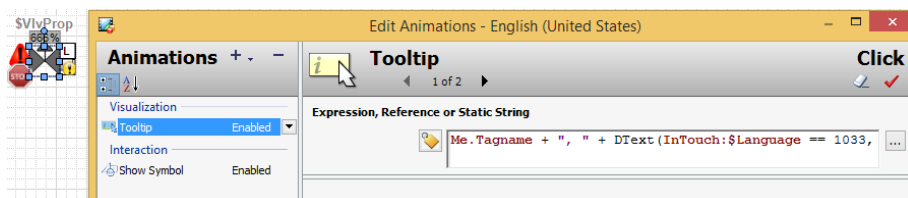
Obr. 7.4 Výběr objektu

Po rozkliknutí jedné z možností se otevře nové okno, ve kterém lze tvořit, nebo editovat ikony, či jiné grafické objekty (viz Obr. 7.5 a Obr. 7.8). V levé části se nachází nástroje pro kreslení a editaci objektu. Pod těmito nástroji je pak seznam použitých nástrojů, jako jsou křivky, textová pole apod. Prostřední částí dominuje editační okno, kde editujeme, či tvoříme daný objekt a v pravé části nalezneme vlastnosti editovaného objektu.



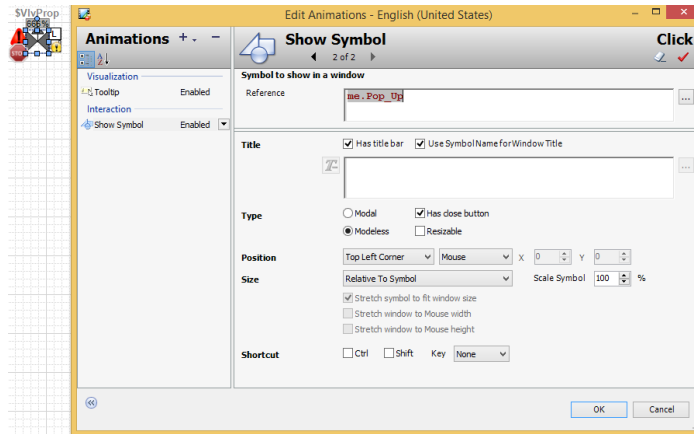
Obr. 7.5 Editace ikony proporcionálního ventilu

Vytvořené grafické prvky mají různé dynamické vlastnosti, jako například otevření pop\_up okna po kliknutí na prvek, či zbarvení obrysu ikony v případě alarmu apod. Tyto dynamické vlastnosti se vytváří za pomoci skriptů, viz Obr. 7.6 a Obr. 7.7. Toto animační (skriptovací) okno otevřeme tak, že označíme objekt, který chceme animovat a pomocí dvojkliku levého tlačítka myši otevřeme animační okno. Pomocí symbolu plus můžeme na objekt navázat další animace z výběru.



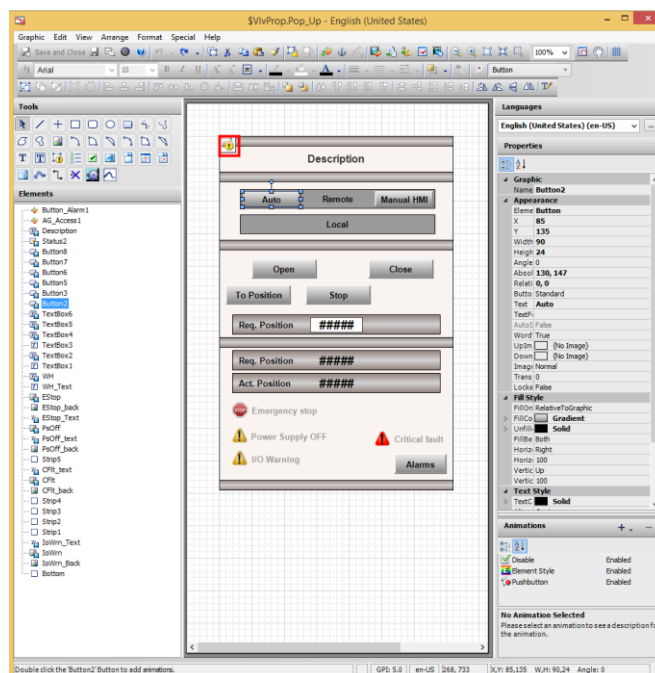
Obr. 7.6 Skripty objektu

Na Obr. 7.7 je ukázka naskriptování vyskakovacího okna. Na objekt ventilu jsem tedy přidal novou animaci „Show Symbol“ a do pole „reference“ jsem napsal skript „me.Pop\_Up“, který zapříčiní otevření pop\_up okna daného objektu po kliknutí na ventil.



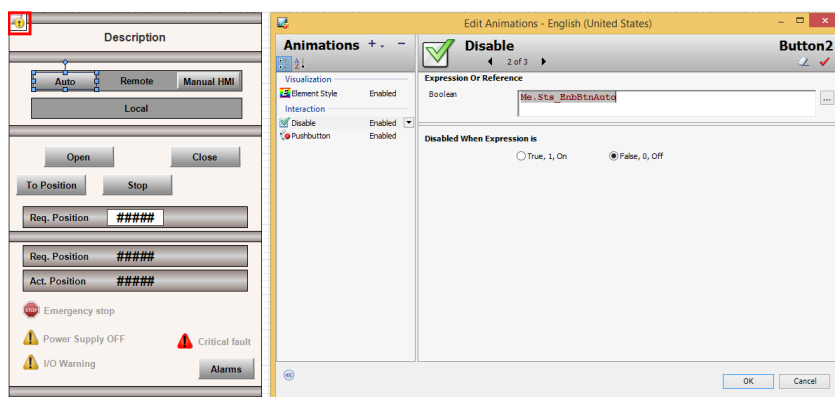
Obr. 7.7 Skripty objektu - nastavení pro pop\_up okno

Pop\_up okno pro proporcionální ventil je vidět na Obr. 7.8. Jedná se prakticky o ovládací panel, pomocí kterého operátor může ručně ovládat daný prvek, v našem případě ventil. Zároveň operátor vidí chybové hlášení a může si zobrazit všechny alarmy, které pak musí každý zvlášť potvrdit. Operátor zde má možnost přepnutí ovládaní pomocí HMI, a ovládat tak ventil manuálně, nebo automatického režimu, který řídí kompletně průmyslový automat. V případě manuálního režimu se rozsvítí status „Local“ a operátor může ventil ovládat pomocí tlačítek Open, Close, Stop nebo v případě částečného otevření, tlačítkem To position, pomocí kterého se ventil otevře na požadovanou hodnotu, kterou operátor nastaví do pole Req. Position.



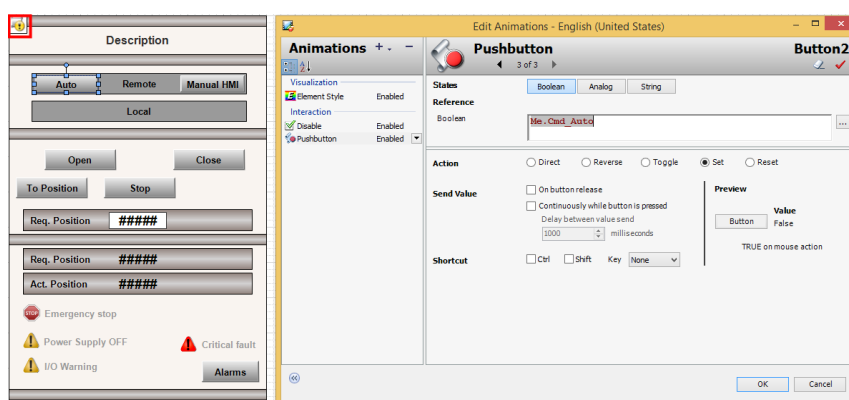
Obr. 7.8 Editace pop\_up okna pro proporcionální ventil

Ke každému tlačítku je potřeba přidat skripty, aby plnilo svou funkci. Pro ukázkou si ukážeme skripty pro tlačítko „Auto“ a status „Local“. Na obrázcích Obr. 7.9 a Obr. 7.10 jsou vidět přidané funkce pro tlačítko „Auto“. V prvním případě se jedná o animaci „disable“ u které je vložen skript „Me.Sts\_EnbBtnAuto“ spolu s nastavením výrazu na „False“. To zapříčiní, že pokud bude například zapnut režim Manual HMI, tlačítko Auto zešedne a deaktivuje se jeho možnost zmáčknutí.



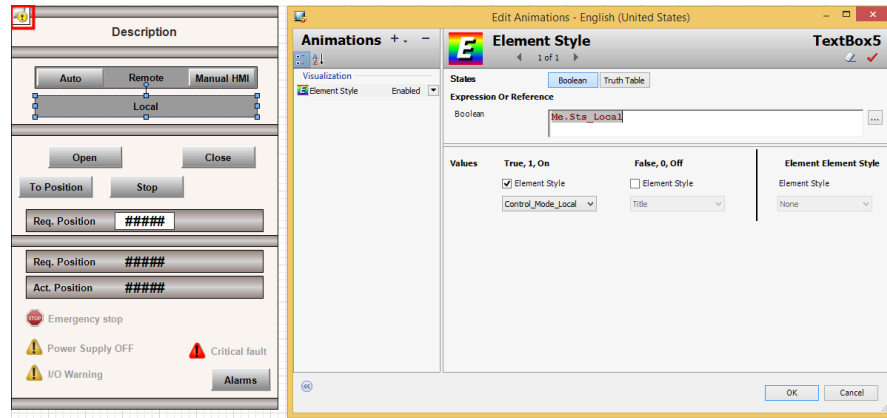
Obr. 7.9 Skript objektu

V případě druhém je animace „Pushbutton“ se skriptem „Me.Cmd\_Auto“ a nastavením na „set“. To znamená, že po zmáčknutí tlačítka „Auto“ se vyšle PLC příkaz pro sepnutí tohoto tlačítka. Můžeme si tedy všimnout, že skripty Me.Sts\_\* jsou statusy a skripty Me.Cmd\_\* příkazy. Stejným, nebo podobným způsobem se takhle musí naskriptovat všechny ostatní tlačítka v Pop\_up okně.



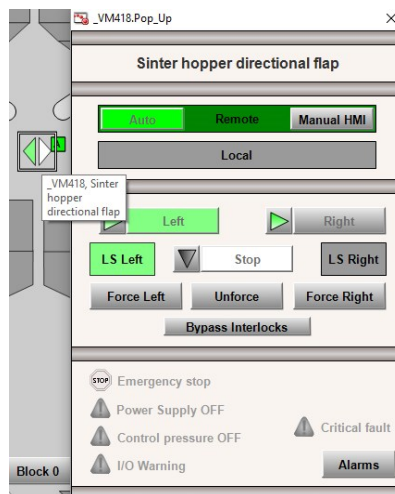
Obr. 7.10 Skript objektu

V poslední ukázce máme skript „Me.Sts\_Local“ v animaci „Element style“, který způsobí zbarvení pole „Local“ v případě, zda se v PLC změní status řízení z „Auto“ na „Local“.



Obr. 7.11 Skript objektu

Obrázek Obr. 7.12 vysvětluje názorně funkčnost skriptů z textu výše. Můžeme si všimnout, že tlačítko „Auto“ je sepnuté a svítí zeleně, ale zároveň je deaktivované. Nejde ho tedy zmáčknout, což zapříčinil právě skript „Me.Sts\_EnbBtnAuto“. Pole „Remote“, ve kterém se nachází právě tlačítko „Auto“ svítí zeleně kvůli skriptu Me.Sts\_Remote. Pokud by se přepnulo řízení klapky na ruční (manuální), pole „Remote“ by zhaslo a rozsvítilo se pole „Local“ díky skriptu „Me.Sts\_Local“.



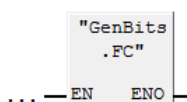
Obr. 7.12 Pop\_up okno s ovládacími prvky směrové klapky

## 8 Dvoucívkový ventil a jeho aplikace

Závěr práce patří aplikování poznatků získaných při tvorbě této práce. Tedy naprogramování vlastního dvoucívkového ventilu ve vývojovém prostředí Step 7 a jeho propojení se simulací ve WinMod tak, aby se dal plnohodnotně ovládat. Ačkoliv se jedná o jednoduchý ventil, je naprogramovaný tak, aby byl bezpečný a systém si dokázal poradit například i s poruchami (chybami) vzniklými ze strany obsluhy, popř. při poruše některé z mechanické, či elektrické části ventilu. Přesto všechno nemohl být mnou vytvořený program řízení ventilu použit do reálného projektu řízení vysoké pece, protože nedosahuje takové technické a programové úrovně, aby byl bezpečný pro použití v reálném systému řízení. Programy řízení ventilů, které se do projektu použily, prošly několikaletým vývojem a jsou nespočetněkrát sofistikovanější a bezpečnější.

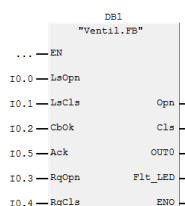
### 8.1 Organizační blok

Celá aplikace ventilu je obsažená v organizačním bloku s označením OB1. Jedná se o systémový organizační blok s cyklickým voláním, kde doba jednoho cyklu může být maximálně 150 ms. V tomto bloku je pak volána funkce FC 100, která slouží k vytvoření logických úrovní a generování hodinového signálu a impulzního signálu s různou frekvencí.



Obr. 8.1 Funkce FC 100-GenBits

Hned za funkcí FC100 následuje funkční blok ventilu FB1, který obsahuje všechnu logiku řízení ventilu.



Obr. 8.2 Organizační blok OB1

## 8.2 Funkční blok

Jedná se o logický blok, který je naprogramován uživatelem a má přiřazenou paměťovou oblast DB1 viz Obr. 8.3, která slouží pro ukládání parametrů.

	Address	Declaratio	Name	Type	Initial valu	Actual valu	Comment
1	0.0	in	LsOpn	BOOL	FALSE	FALSE	Koncový snímac OPEN
2	0.1	in	LsCls	BOOL	FALSE	FALSE	Koncový snímac CLOSE
3	0.2	in	CbOk	BOOL	FALSE	FALSE	Jistic
4	0.3	in	Ack	BOOL	FALSE	FALSE	
5	0.4	in	RqOpn	BOOL	FALSE	FALSE	Request OPEN
6	0.5	in	RqCls	BOOL	FALSE	FALSE	Request CLOSE
7	2.0	out	Opn	BOOL	FALSE	FALSE	Otevira se
8	2.1	out	Cls	BOOL	FALSE	FALSE	Zavira se
9	2.2	out	OUT0	BOOL	FALSE	FALSE	
10	2.3	out	Flt_LED	BOOL	FALSE	FALSE	Status zastaveno
11	4.0	stat	StsOpen	BOOL	FALSE	FALSE	Informace ze je ventil otevren
12	4.1	stat	CbReady	BOOL	FALSE	FALSE	Informace ze jistic je v poradku
13	4.2	stat	StsClose	BOOL	FALSE	FALSE	Informace ze je ventil zavren
14	4.3	stat	LsFlt	BOOL	FALSE	FALSE	POZOR!Chyba ventilu
15	4.4	stat	RqFlt	BOOL	FALSE	FALSE	POZOR!Obe tlačitka zapnuta
16	4.5	stat	AuxOpn	BOOL	FALSE	FALSE	
17	6.0	stat:in	LsOpn_Ton.IN	BOOL	FALSE	FALSE	
18	8.0	stat:in	LsOpn_Ton.PT	TIME	T#0MS	T#0MS	
19	12.0	stat:out	LsOpn_Ton.Q	BOOL	FALSE	FALSE	

Obr. 8.3 Paměťová oblast DB1

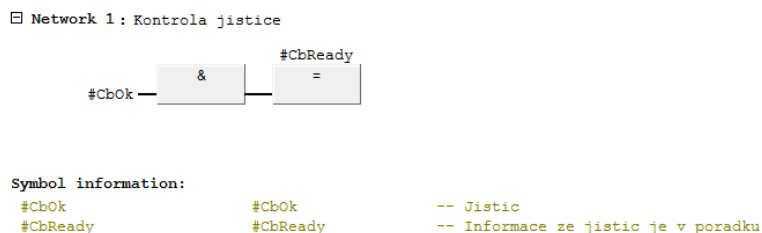
Struktura programu (funkčního bloku FB1) je rozdělena na tři části dle funkce:

- diagnostika
- tělo programu
- statusy-hlášení

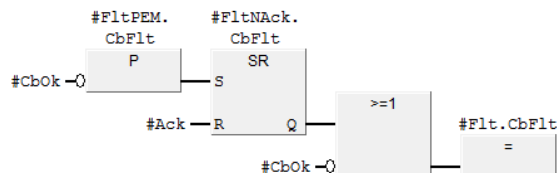
### 8.2.1 Diagnostika

Diagnostika se týká zjištění stavů, před samotným vykonáním těla programu. Čili zda jsou v pořádku například jističe, zda byla správně použita ovládací tlačítka, zda se dosáhlo požadované polohy šoupátka ventilu v daném čase apod. V opačném případě je potřeba vyřadit ventil z provozu, nebo jej vrátit do výchozí polohy, aby nedošlo k poškození technologie, nebo zařízení.

Na Obr. 8.4 a Obr. 8.5 je naprogramována diagnostika jističe a jeho potvrzení chyby v případě poruchy. Na obrázcích vidíme, že pokud je jistič v pořádku (#CbOK) sepne status (#CbReady), říkající, že jistič je připraven. Pokud ale jistič v pořádku není, přes náběžnou hranu a funkci set/reset se sepne chybová hláška (#CbFlt). Ventil se tedy nedá dále zprovoznit, dokud vlastnoručně nepotvrdíme chybu.

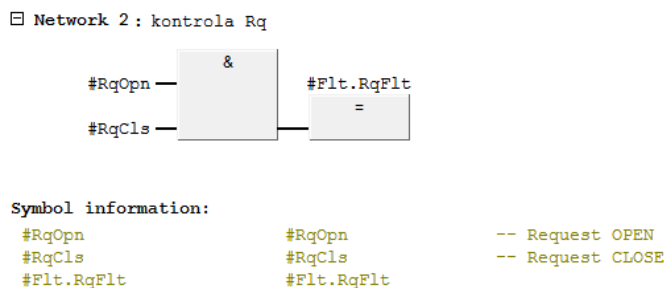


Obr. 8.4 Kontrola jističe



Obr. 8.5 Potvrzení chyby jističe

Na obrázku Obr. 8.6 je zase vidět kontrola sepnutí tlačítek otevřít a zavřít. Pokud jsou sepnuta obě tlačítka zároveň, program tuto skutečnost vyhodnotí jako chybu a ventil zastaví, nebo jej vůbec neuvede do provozu.



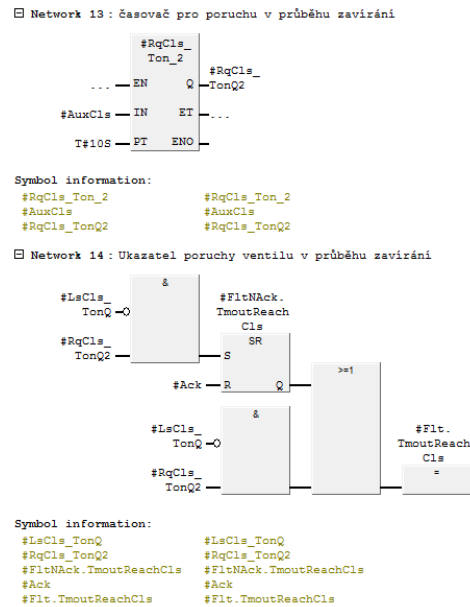
Obr. 8.6 Kontrola požadavku otevření/zavření

## 8.2.2 Tělo programu

Samotné tělo programu obsahuje přímé řízení ventilu, tzn. jeho otvírání a zavírání, nebo automatiku, která při jakékoliv poruše ventil uvede do bezpečné polohy.

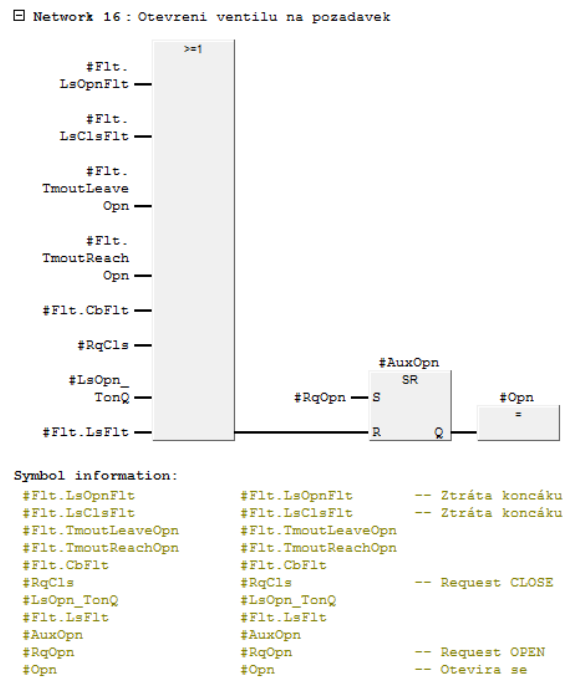
Automatika vyhodnocení chyb ventilu byla vytvořena i za pomoci časovačů viz Obr. 8.7. Na obrázku jde vidět časovač, který při zavírání po 10 sekundách vyše na výstup (#RqCls\_TonQ2) signál a pokud není ventil do té doby v koncové poloze (#LsCls\_TonQ), vyskočí chyba, že se ventil v průběhu zavírání zaseknul. Opět je nutné chybu potvrdit.





Obr. 8.7 Časovač poruchy a ukazatel poruchy v průběhu zavírání

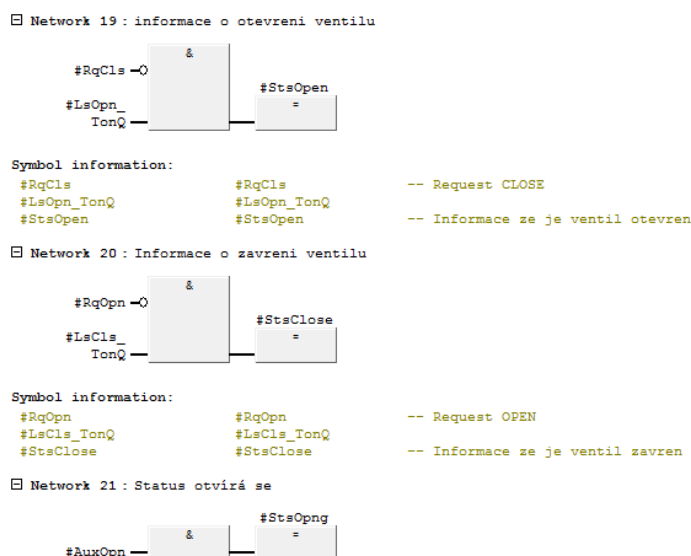
Funkce otevření ventilu je ukázána na Obr. 8.8. Je tvořena funkcí set/reset, tzn., že pokud není aktivní některá z možných chyb, tak se ventil otevře. Jakmile je tomu jinak a minimálně jedna z chyb je aktivní, automaticky se otevírání ventilu přeruší.



Obr. 8.8 Otevření ventilu

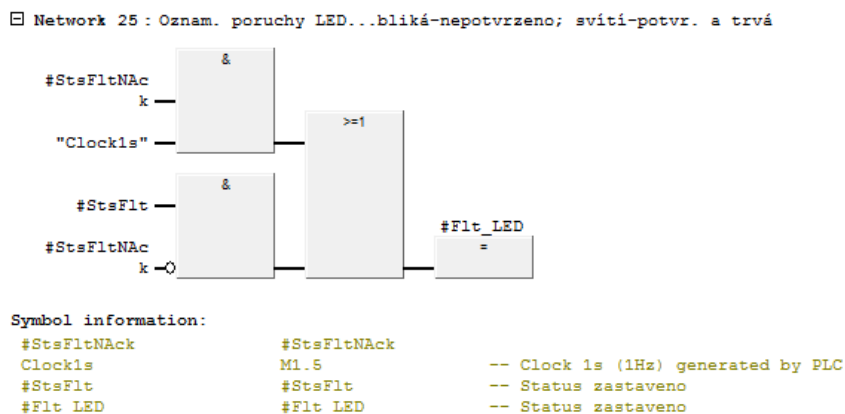
### 8.2.3 Statusy a hlášení

Na konci programu se nachází statusy/hlášení, která informují o stavu zařízení, viz Obr. 8.9. Zda je ventil otevřen nebo uzavřen, zda se otevírá, či uzavírá, nebo chybové hlášení v případě poruchy apod.



Obr. 8.9 Část programu - statusy

Aby se lépe poznalo, že má ventil nějakou poruchu, byla naprogramovaná chybová LED dioda, která se nachází i v simulaci, viz další kapitola. Funkce diody je taková, že pokud nastane nějaká chyba, dioda bude blikat, dokud se chyba neopraví a ručně tlačítkem na ovládacím panelu nepotvrdí. Pokud se chyba potvrdí a chyba neustále trvá, dioda bude svítit.



Obr. 8.10 Naprogramování chybové LED

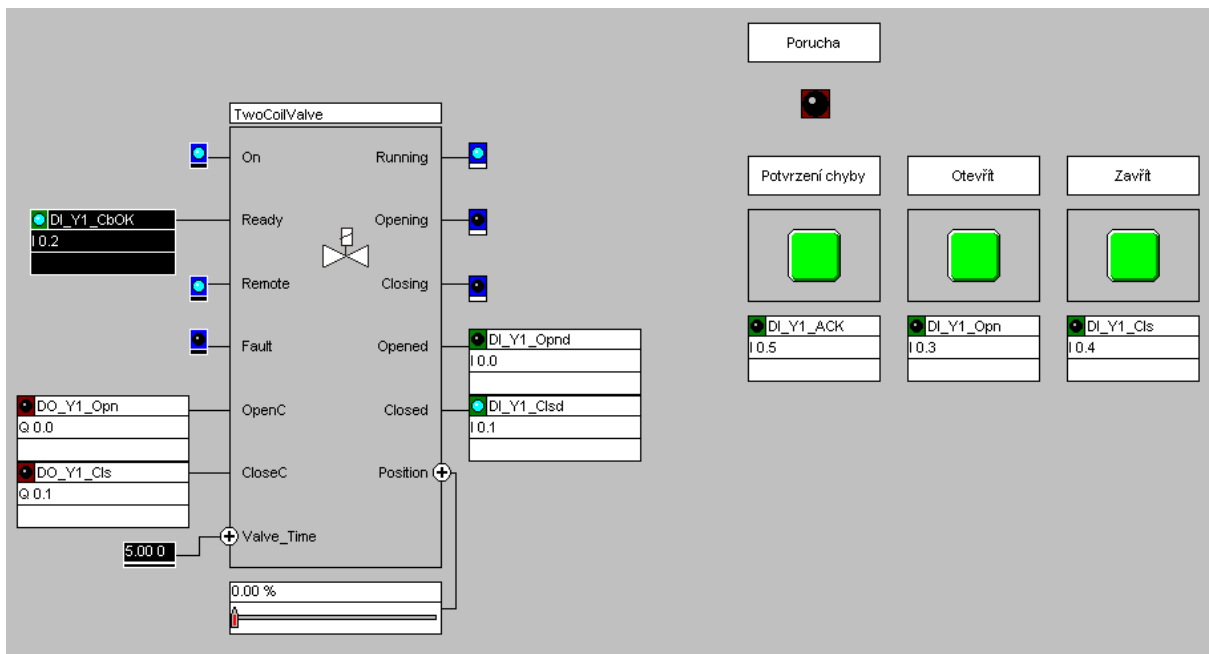
### 8.3 Simulace ventilu

Aby se mohlo reálné chování ventilu otestovat, vyzkoušet chybové stavy a zjistit tak reakci systému na tyto skutečnosti, bylo potřeba vytvořit i simulaci daného ventilu a jeho ovládání v simulačním softwaru WinMOD.

Ve WinMOD se tedy vytvořila HW konfigurace a nainportovaly vstupně/výstupní proměnné. Do simulačního okna se následně vložilo makro pro dvoucívkový ventil, vložily se všechny vstupně/výstupní proměnné a pomocí tlačítek se vytvořil jednoduchý ovládací panel s diodou indikující poruchu. K makru ventilu se připojily vstupně/výstupní proměnné kromě těch, které přímo sloužily pro ovládání ventilu. Konkrétně se jedná se o potvrzení chyby, již zmíněné signalizaci poruchy a vstupní proměnné otevření a zavření ventilu viz pravá část Obr. 8.11.

Nejdříve se vyzkoušelo optimální chování ventilu bez poruchy. Pomocí možnosti vynucení (forcing) se zapnul jistič, což systém vyhodnotil, že je vše připraveno a může se tak ventil ovládat. Pomocí tlačítek otevřít a zavřít se ventil dal do pohybu a na analogovém posuvníku pod makrem ventilu bylo možno vidět pohyb ukazatele, který takhle sděloval otevření nebo uzavření ventilu v procentech. Pokud se ventil otevřel na 100 %, rozsvítil se signál proměnné opened a naopak, když se ventil uzavřel, rozsvítil se signál closed. Takhle se ověřilo klasické ovládání ventilu.

Na závěr bylo potřeba vyzkoušet chování systému v případě poruchy ventilu. Nejdříve se tedy vyzkoušela reakce systému na nefunkční jistič, nebo vypnutí jističe v průběhu otevírání ventilu. Při vypnutém jističi a pokusu o otevření ventilu se rozblíkala alarmová světelná indikace a i při opětovném zapnutí jističe se ventil nedal ovládat, dokud se chyba vlastnoručně nepotvrdila tlačítkem potvrzující chybu. V případě poruchy jističe v průběhu otevírání, či zavírání se ventil zastavil, zapnula se indikace poruchy, a dokud se porucha ručně nepotvrdila, ventil se nedal ovládat. Další z testů bylo například ruční prodloužení doby otevírání, či zavírání ventilu, aniž by se nastavené časové parametry v programu změnily. Tím se simulovala porucha zaseknutí ventilu v průběhu otevírání a zavírání.



Obr. 8.11 Simulace dvoucívkového ventilu

Vizualizaci ventilu již nebylo třeba dělat. Jednak z důvodu, že ventil nebyl použit do projektu řízení vysoké pece a také proto, že k jeho otestování a zjištění tak funkčnosti programu pro jeho řízení stačila simulace v softwaru WinMOD, která slouží právě k těmto účelům testování.

## 9 Závěr

Cílem práce bylo seznámit se s hutním technologickým zařízením vysoké pece pro výrobu surového železa a jeho přidružených technologií. Pochopit princip fungování této výrobní technologie, který je důležitý pro následný vývoj řízení, obsluhu technologie a jejich dílčích technologických celků. Seznámit se a osvojit si vývojové a aplikační prostředí pro vývoj a aplikaci nově vyvíjených programů a řídicích systémů firmou Ingeteam a.s. pro projekt rekonstrukce vysoké pece a jeho technologických součástí v polském ArcelorMittal v Krakowě.

Ačkoliv jsem se velkou měrou zabýval samotnou teorií fungování a plnění vysoké pece, protože při vývoji řídicích systémů je opravdu nutné znát principy fungování řízené technologie, celá sepsaná teoretická část se nachází odděleně v přílohách. Kromě popisu vysoké pece se zde nachází i popis a samotný princip plnění pece. Dozvěděli jsme se například, jak funguje bezzvonová sazebná anebo co je distribuční žlab a jak důležitý je pro funkci plnění pece.

Značná část práce se týkala kompletně praktické části, především pak tvorbě simulace řízení v prostředí WinMOD pro jednotlivé technologické celky vysoké pece včetně pece samotné, což bylo hlavním úkolem této práce. Software WinMOD je real-time simulační prostředí, které dokáže nahradit (simulovat) řízenou technologii. Díky tomu bylo možné průběžně testovat a zkoušet vyvíjenou řídicí aplikaci pro průmyslový počítač. Testovat ji po stránce funkčnosti, ale i po stránce ochrany před vnějšími vlivy, jako jsou nahodilé chyby (poruchy), chyby operátorů apod. Vývoj je tak díky tomu rychlejší, levnější a díky možnosti vyvíjet systém aniž by bylo nutné jej mít připojený k reálné řízené technologii i bezpečnější, především pak v hutních provozech.

Celkem tedy bylo vytvořeno 16 simulačních oken, obsahující přes 121 různých maker, skládajících se z pohonů, vah, ventilů atd., které simulují jednotlivé části vysoké pece a její přidružené technologie, např. skipový vrátek.

Kromě simulace jsem se zabýval částečně i vizualizací, konkrétně tvorbě ikon a pop\_up oken a přípravě vizualizačních obrazovek pro provoz přípravy a vážení vysokopeční vsázky a bezzvonovou sazebnou. U ikon a pop\_up oken jsem kromě samotné tvorby a jejich editace vytvářel i jednotlivé animace pomocí skriptů. U vizualizačních obrazovek jsem nakreslil

a připravil základní strukturu obrazovky a vložil všechny potřebné funkční prvky, jako jsou ventily, klapky apod. U provozu přípravy a vážení vsázky jsem ještě navíc zanimal všechny dostupné dopravníky.

V samotném závěru práce je pak aplikování získaných poznatků v podobě naprogramování řídicí aplikace pro dvoucívkový ventil v prostředí Step 7 a jeho otestování ve vytvořené simulaci v prostředí WinMOD. Je nutno říci, že ventil nebyl použit v reálném projektu řízení vysoké pece, protože nedosahuje takové technické a programové úrovně, aby byl bezpečný pro použití v reálném systému řízení. Programy řízení ventilů, které se do projektu použily, prošly několikaletým vývojem a jsou nespočetněkrát sofistikovanější a bezpečnější.

Díky této práci jsem získal spoustu praktických zkušeností přímo v prostředí firmy zabývající se průmyslovou automatizací. Setkal jsem se s řídicím systémem firmy Siemens, konkrétně se jedná o Simatic S7-400 a osvojil jsem si práci s vývojovým prostředím Step 7 od stejnojmenné firmy. Nejvíce zkušeností jsem však získal prací v simulačním software WinMOD, který dokáže ušetřit peníze, čas i technologii při vývoji řídicího systému. Prvně jsem se také setkal s vývojovým prostředím Archestra IDE od firmy Wonderware, což je integrované vývojové prostředí společné pro vývoj aplikací InTouch a mohl tak vidět, jak se vyvíjí složitá vizualizační aplikace v rámci týmu. Tato práce mi tedy dala možnost ukázat, co taková modernizace řízení vysoké pece a jejich přidružených technologií obnáší a co všechno je potřeba pro to udělat.

## 10 Seznam použité literatury

Automatizační systémy, 2015. *SIEMENS* [online]. [cit. 2015-10-13]. Dostupné z: <http://stest1.etnetera.cz/ad/current/index.php?ctxnh=1d37de2332&ctxp=home>

BROŽ, Ludvík, Jiří BILÍK a Ján KRÉT, 1985. *Vysokopecní výroba surového železa*. 1. Ostrava: VŠB-TU Ostrava.

FARANA, Radim. *Doporučení pro psaní odborných textů z oblasti automatizace a informatiky*. Ostrava: VŠB - Technická univerzita Ostrava, 2008. ISBN 978-80-248-1925-9.

FOJTÍK, David, Jiří TŮMA a Petr JURGA, 2008. Počítačová simulace sypání vsázky do vysoké pece bezzvonnou sazebnou. *Hutnické listy*. (3), 78-84.

FOLTÝNEK, Petr. *Vizualizace technologických procesů a řízení na platformě PC*. Dostupné z URL <http://books.fs.vsb.cz/FactoryLink/Index.html>

FOLTÝNEK, Petr, 2001. *Procesní vizualizace v průmyslové automatizaci*. Ostrava. Bakalářská práce. VŠB-TUO. Vedoucí práce Lenka Landryová.

HAVRÁNKOVÁ, Renáta, 2015. *Hardwarový simulátor technologických procesů pro řídicí systém ABB/AC800*. Ostrava. Diplomová práce. VŠB-TUO. Vedoucí práce Lenka Landryová.

Hydraulické oleje minerální, bezpopelné a detergentní, 2016. In: *Lubstar* [online]. [cit. 2017-03-22]. Dostupné z: <http://www.lubstar.cz/cs/omv/hydraulicke-oleje-mineralni-bezpopelne-a-detergentni/2/>

InTouch (SCADA/HMI), ©2016. In: *Pantek* [online]. [cit. 2016-6-3]. Dostupné z: <http://www.pantek.cz/produkty/intouch/anotace.html>

Iron and steel, ©2016. In: *Viktor Mácha - Industrial: Documentation of iron and steel industries* [online]. [cit. 2015-11-05]. Dostupné z: <http://www.viktormacha.com/galerie/iron-and-steel-11/>

KOCIÁN, Jiří, 2011. *Simulační software WinMOD: Návod*. Ostrava.

KOZIOREK, Jiří. *Distribuované systémy řízení: učební text*. Ostrava: Vysoká škola báňská - Technická univerzita, 2011. ISBN 978-80-248-2599-1.

KRÉT, Ján, 2013. *Základy teorie a technologie výroby železa a oceli*. Studijní opora [online], Dostupné z: [http://katedry.fmmi.vsb.cz/Modin\\_Animace/Opory/02\\_Metalurgicke\\_inzenyrstvi/06\\_Zaklady\\_teorie\\_a\\_techologie\\_vyroby\\_zeleza\\_a\\_oceli\\_I\\_Zelezarstvi/Kret\\_Teorie\\_procesu\\_pri\\_vyrobe\\_zeleza\\_a\\_oceli.pdf](http://katedry.fmmi.vsb.cz/Modin_Animace/Opory/02_Metalurgicke_inzenyrstvi/06_Zaklady_teorie_a_techologie_vyroby_zeleza_a_oceli_I_Zelezarstvi/Kret_Teorie_procesu_pri_vyrobe_zeleza_a_oceli.pdf)

MAKARIUS, Miroslav, 1957. *Výroba surového železa*. Díl I. Státní nakladatelství technické literatury. Praha

OTÁHALOVÁ, Helena, 1985. *Výroba železných kovů I (pro 2. a 3. ročník středních odborných učilišť)*. SNTL – Nakladatelství technické literatury. Praha.

PROGRAMOVATELNÉ AUTOMATY – TECHNICKÉ VYBAVENÍ PLC, ©2016.  
*Učíme v prostoru: Encyklopedie - Drtič kamene* [online]. [cit. 2016-6-04]. Dostupné z: [http://uvp3d.cz/drtic/?page\\_id=3341](http://uvp3d.cz/drtic/?page_id=3341)

SIEMENS, 2006. *SIMATIC: Working with STEP 7*. 3. Dostupné také z: [https://www.fer.unizg.hr/\\_download/repository/STEP7.pdf](https://www.fer.unizg.hr/_download/repository/STEP7.pdf)

SKÝPALA, Jiří, 2015. *Design a konstrukce testovacího zařízení pro AC700F*. Ostrava. Diplomová práce. VŠB-TU Ostrava. Vedoucí práce Lenka Landryová.

Standard CPUs: CPUs, ©2017. *Siemens* [online]. [cit. 2017-02-27]. Dostupné z: [http://w3.siemens.com/mcms/programmable-logic-controller/en/advanced-controller/s7-400/cpu/standard-cpus/pages/default.aspx#SIMATIC\\_20S7\\_400\\_20CPU\\_20414\\_3\\_20](http://w3.siemens.com/mcms/programmable-logic-controller/en/advanced-controller/s7-400/cpu/standard-cpus/pages/default.aspx#SIMATIC_20S7_400_20CPU_20414_3_20)

Technické materiály, ©2015. *Učíme v prostoru: Encyklopedie - drtič kamene* [online]. [cit. 2015-10-06]. Dostupné z: [http://uvp3d.cz/drtic/?page\\_id=2787](http://uvp3d.cz/drtic/?page_id=2787)

TOMAN, Vladimír, *Metalurgie železných kovů: Výroba surového železa*. Dostupné také z: [http://www.ekomonitor.cz/sites/default/files/filepath/prezentace/4-2\\_vyroba\\_zeleza\\_final.pdf](http://www.ekomonitor.cz/sites/default/files/filepath/prezentace/4-2_vyroba_zeleza_final.pdf)

VLACH, Jaroslav. *Řízení a vizualizace technologických procesů*. Praha: BEN - technická literatura, 1999. ISBN 80-86056-66-x.

Vysokopecní provoz železáren, ©2015. *Nejen hornictví.info* [online]. [cit. 2015-10-29]. Dostupné z: <http://www.hornictvi.info/prirucka/zprac/vyspec/vyspec.htm>

*Výroba surového železa* [online], [cit. 2015-10-04]. Dostupné z: <http://www.vyrobazeleza.wz.cz/>

*WinMod: Easy to use* [online], ©2015. [cit. 2015-11-11]. Dostupné z: <http://winmod.de/en/>

Wonderware InTouch: Vaše průlomové řešení pro podnikovou vizualizaci a supervizní řízení SCADA/HMI, ©2017. *Pantek* [online]. [cit. 2017-04-28]. Dostupné z: [http://www.pantek.cz/pdf/produkty/intouch/intouch\\_dtsh.pdf](http://www.pantek.cz/pdf/produkty/intouch/intouch_dtsh.pdf)

Wonderware software: Průmyslové automatizační a informační aplikace pro výrobní inteligenci, ©2017. *Pantek* [online]. Česká republika [cit. 2017-03-27]. Dostupné z: [http://www.pantek.cz/produkty.php?id\\_produkту=9&produkt=&id\\_podkategorie=31&pod\\_kategorie=anotace](http://www.pantek.cz/produkty.php?id_produkту=9&produkt=&id_podkategorie=31&pod_kategorie=anotace)

ZEZULKA, František, 2010. *Programovatelné automaty, učební texty*. VUT Brno



## 11 Příloha na CD

- I. Popis vysoké pece
- II. Příprava simulačního software WinMOD
- III. Program PLC - ventil
- IV. Simulace - ventil
- V. Simulace - vysoká pec