



## Diplomová práce

# Automatizace vyhodnocení možnosti rozšíření svařovacích zařízení při integraci nových modelů ve Škoda Auto

*Studijní program:*

N0714A270010 Mechatronika

*Autor práce:*

**Bc. David Krčmář**

*Vedoucí práce:*

Ing. Lukáš Hubka, Ph.D.

Ústav mechatroniky a technické informatiky

Liberec 2024



## Zadání diplomové práce

# Automatizace vyhodnocení možnosti rozšíření svařovacích zařízení při integraci nových modelů ve Škoda Auto

<i>Jméno a příjmení:</i>	<b>Bc. David Krčmář</b>
<i>Osobní číslo:</i>	M22000046
<i>Studijní program:</i>	N0714A270010 Mechatronika
<i>Zadávací katedra:</i>	Ústav mechatroniky a technické informatiky
<i>Akademický rok:</i>	2023/2024

### Zásady pro vypracování:

1. Analyzujte možnosti PLC Simatic 1517F z hlediska přístupu k systémovým datům CPU a Profinet kontroléru a vyberte důležité proměnné pro rozšiřitelnost stávajících výrobních zařízení.
2. Naprogramujte funkční blok dle standardu VASS V6 v jazyce SCL, který zpracuje vybraná data a připraví je pro vizualizaci (HMI, ZENON).
3. Vytvořte obrazovku pro vizualizaci výrobního zařízení dle standardu VASS V6 a shrňte pravidla pro generování obrazovky pomocí SiVArc.
4. Vytvořte a implementujte funkci pro předávání dat z CPU do nadřazeného systému ZENON prostřednictvím TCP/IP.
5. Navržené řešení integrujte a ověřte jeho funkčnost.

*Rozsah grafických prací:* dle potřeby dokumentace  
*Rozsah pracovní zprávy:* 40 až 50 stran  
*Forma zpracování práce:* tištěná/elektronická  
*Jazyk práce:* čeština

### **Seznam odborné literatury:**

- [1] BERGER, Hans. Automating with Simatic S7-1500: Configuring, Programming and Testing with Step 7 Professional. Publicis MCD Werbeagentur, 2017. ISBN 9783895784606.
- [2] Programming Guideline for S7-1200/S7-1500: STEP 7 (TIA Portal) and STEP 7 Safety in TIA Portal [online]. 12/2018. Siemens [cit. 2023-04-10]. Dostupné z:  
[https://cache.industry.siemens.com/dl/files/040/90885040/att\\_970576/v1/81318674\\_Programming\\_guideline\\_DOC\\_v16\\_en.pdf](https://cache.industry.siemens.com/dl/files/040/90885040/att_970576/v1/81318674_Programming_guideline_DOC_v16_en.pdf).
- [3] Siemens AG. VASS V6 Standard. Standardization guideline. Siemens AG, 2019.

*Vedoucí práce:* Ing. Lukáš Hubka, Ph.D.  
Ústav mechatroniky a technické informatiky

*Datum zadání práce:* 12. října 2023  
*Předpokládaný termín odevzdání:* 14. května 2024

prof. Ing. Zdeněk Plíva, Ph.D.  
děkan

L.S.

doc. Dr. Ing. Jaroslav Hlava  
garant studijního programu

V Liberci dne 12. října 2023

## Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědom toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

# Automatizace vyhodnocení možnosti rozšíření svařovacích zařízení při integraci nových modelů ve Škoda Auto

## Abstrakt

Účelem této práce je analyzovat PLC Simatic 1517F z hlediska diagnostických schopností přístupu dat CPU a vybrat ty, které mají velkou váhu při integraci stávajících svařovacích linek ve Škoda-Auto a vytvořit nástroj pro zobrazení těchto informací. Je zde řešena problematika dostupnosti interních informací o PLC, jeho programování dle standardu VASS a vytváření nových uživatelských obrazovek v SiVArC.

**Klíčová slova:** Siemens Simatic, TIA Portal, SiVarc, WinCC, Škoda-Auto

# Automation of Evaluation of Expansion Possibility of Welding Equipment During Škoda Auto's Integration of New Models

## Abstract

The purpose of this work is to analyze the PLC Simatic 1517F in terms of diagnostic capabilities of CPU data access and select those that are of great importance in the integration of existing welding lines in Škoda-Auto and to create a tool for displaying this information. The issue of accessibility of internal PLC information, its programming according to the VASS standard and the creation of new user screens in SiVArC is addressed.

**Keywords:** Siemens Simatic, TIA Portal, SiVarc, WinCC, Škoda-Auto

## Poděkování

Rád bych poděkoval všem, kteří přispěli ke vzniku tohoto díla, zejména mému konzultantovi a kolegovi Ing. Tomáši Snětivému za podporu a pevnou ruku. Dále Ing. Lukáši Hubkovi, Ph.D. za odborné rady při psaní diplomové práce.

# Obsah

Seznam obrázků . . . . .	8
Seznam tabulek . . . . .	9
Seznam zkratk . . . . .	11
<b>1 Úvod</b>	<b>12</b>
<b>2 Průmyslové automatizační systémy</b>	<b>13</b>
2.1 Simatic S7-1500 . . . . .	13
2.1.1 Charakteristiky systému . . . . .	13
2.1.2 Moduly . . . . .	15
2.2 Simatic IPC panely . . . . .	16
2.2.1 Hardwarová Architektura . . . . .	16
2.2.2 HMI software . . . . .	17
2.2.3 IPC 477 E Pro . . . . .	18
2.3 Profinet . . . . .	19
2.4 Tia Portal . . . . .	20
2.5 WinCC . . . . .	22
2.6 Programovací standardy . . . . .	23
<b>3 VASS standard</b>	<b>24</b>
3.1 Vytvoření projektu . . . . .	25
3.2 Profinet . . . . .	26
3.3 VASS definice funkcí a funkčních bloků . . . . .	26
3.4 Pravidla pro generování vizualizace . . . . .	27
3.5 Struktura projektu TIA Portal . . . . .	28
3.6 SiVArc . . . . .	30
3.6.1 Přípravné kroky . . . . .	31
3.6.2 Generace SiVArc . . . . .	32
3.6.3 Generovací matice . . . . .	33
<b>4 Vytvoření diagnostického nástroje</b>	<b>34</b>
4.1 Analýza bloků v TIA Portalu . . . . .	34
4.1.1 Diagnostické bloky . . . . .	35
4.1.2 VASS bloky . . . . .	36
4.1.3 Stahování dat vytíženosti paměti . . . . .	37
4.2 Vytvoření VASS modulu . . . . .	38

4.2.1	Práce v základním VASS projektu . . . . .	38
4.2.2	Vytvoření FB modulu v jazyce SCL . . . . .	38
4.2.3	Základní příkazy pro řízení programu . . . . .	40
4.2.4	Vytvoření diagnostického bloku . . . . .	42
4.2.5	ZAU . . . . .	43
4.3	Práce v základním projektu WinCC . . . . .	44
4.3.1	Vytvoření faceplate . . . . .	44
4.3.2	Vytvoření tlačítka . . . . .	46
4.3.3	Layouty . . . . .	47
4.3.4	Přiřazení makra . . . . .	48
4.3.5	Pravidla generování . . . . .	49
4.4	Zprovoznění na výrobní lince . . . . .	51
<b>5</b>	<b>Závěr</b>	<b>55</b>
	<b>Použitá literatura</b>	<b>57</b>



## Seznam obrázků

2.1	CPU 1517F-3 [10]	15
2.2	IPC 477E 19" [4]	18
2.3	WinCC Unified System [19]	22
3.1	Diagram vytvoření projektu	25
3.2	Struktura názvů zařízení PROFINET	26
3.3	Strom projektu	28
3.4	SiVArc [15]	30
4.1	Testovací soustava	34
4.2	Knihovna diagnostických bloků	35
4.3	Blok RT_INFO	35
4.4	Blok GetSMCinfo	36
4.5	Počet účastníků sítě	36
4.6	Vytvoření data type pro obrazovku a FB	40
4.7	Přiřazení proměnných v ZAU	43
4.8	Vytvoření obrazovky s objekty	45
4.9	Přiřazení animací	45
4.10	Přiřazení proměnných	46
4.11	Vytvoření tlačítka	46
4.12	Přiřazení obrazovky tlačítka	47
4.13	Layout 1	47
4.14	Layout 2	48
4.15	Přiřazení makra	48
4.16	Pravidla generování	50
4.17	Knihovna systemanalyse	51
4.18	Nahrání bloku FB <sub>systemanalyse</sub>	52
4.19	Nastavení bloku FB <sub>systemanalyse</sub>	52
4.20	Web server	53
4.21	Cesta k VB skriptům	53
4.22	Vytvoření HMI tagu	54
4.23	Vizualizace na panelu	54

## Seznam tabulek

4.1	Číslování FB bloků . . . . .	39
4.2	Povolené barvy . . . . .	44

## Seznam zkratek

<b>ARG</b>	Arbeitsgruppe; Pracovní skupina
<b>BMK</b>	Betriebsmittelkennzeichnung; Označení provozního prostředku
<b>CFAST</b>	Compact Flash ATA Serial Transport; Paměťová karta
<b>CPU</b>	Central Processing Unit, Centrální procesorová jednotka
<b>FB</b>	Function Block; Funkční blok
<b>FC</b>	Function Call; Volání funkce
<b>FM</b>	Fakulta mechatroniky, informatiky a mezioborových studií Technické univerzity v Liberci
<b>GBA</b>	Großbildanzeige; Velkoplošná obrazovka
<b>GSDML</b>	General Station Description Markup Language; Inženýrská data zařízení
<b>HMI</b>	Human machine interface; Rozhraní člověk - stroj
<b>IEEE</b>	Institute of Electrical and Electronics Engineers; Mezinárodní nezisková profesní organizace, usilující o vzestup technologie související s elektrotechnikou
<b>ISO/OSI</b>	International Organization for Standardization/Open Systems Interconnection; Referenční model pro komunikaci
<b>IO</b>	Input/Output; Vstupně/výstupní
<b>IP</b>	Internet Protocol; Síťový protokol
<b>IP</b>	Ingress Protection; Označení ochrana zařízení před pevnými látkami a tekutinami
<b>IPC</b>	Industrial PC; Průmyslový počítač
<b>OPC</b>	OLE (Object Linking and Embedding) for Process Control; Standard softwarového rozhraní průmyslových zařízení
<b>PCI</b>	Peripheral Component Interconnect; Počítačová sběrnice
<b>PLC</b>	Programable Logic Controller; Programovatelný logický automat
<b>PoT</b>	Produktion ohne Teil; Produkce bez dílu
<b>RAM</b>	Random Access Memory; Operační paměť
<b>SiVArc</b>	SIMATIC Visualization Architect
<b>SK</b>	Scgutzkreis; Pracovní okruh
<b>SSD</b>	Solid-state drive; Polovodičový disk
<b>TIA</b>	Totally Integrated Automation; Úplně integrovaná automatizace
<b>TUL</b>	Technická univerzita v Liberci
<b>USB</b>	Universal Serial Bus; Univerzální sériová sběrnice
<b>VASS</b>	Volkswagen Audi SEAT Škoda
<b>VB</b>	Visual Basic; Programovací jazyk
<b>ZAU</b>	Zentrale Anlagenüberwachung; Nadřazený systém

# 1 Úvod

Motivací pro tuto práci bylo vytvořit nástroj pro usnadnění práce projektantům z plánování svařoven ve Škoda Auto. Myšlenkou bylo vytvořit informační obrazovku na IPC panelu, která by obsahovala všechny potřebná data k analýze před integrací výrobní linky, respektive jejím rozšířením. Dalším bodem je tyto data připravit pro odesílání do nadřazené sítě. Došlo by tedy k odstranění nutnosti chodit na halu k výrobním linkám. Výsledkem by byla úspora času a usnadnění práce.

Vlastní práce by tedy spočívala v získání potřebných dat, respektive analýze diagnostických schopností TIA Portalu. Následně jejich zpracování v novém programovém bloku. K vytvoření nové obrazovky je zapotřebí vytvoření pravidel pro generování a podkladových obrazovek, které budou obsahovat získaná data.

## 2 Průmyslové automatizační systémy

### 2.1 Simatic S7-1500

Jedná se o nejvyšší řadu průmyslových automatů značky Siemens a jsou určeny k řízení náročných aplikací vyžadující vysokou výkonnost, spolehlivou komunikaci, flexibilitu a technologické funkce.

**Klíčové vlastnosti :**

- Vysoký výkon
- Integrovaná systémová diagnostika
- Funkční design
- Řízení technologických systémů
- Integrovaná safety
- Ochrana programu

#### 2.1.1 Charakteristiky systému

**Technologické funkce**

K dispozici je integrovaná podpora Motion Control pro připojení pohonů SINAMICS přímo do procesorové jednotky nebo pomocí sběrnice pro PROFIdrive pohony. Pro pohony připojené přes PROFIdrive jsou k dispozici softwarové funkční bloky, které odpovídají standardům PLCopen a usnadňující práci. Dále je možné vyhodnocovat signály z enkodéru na otáčkových či polohovacích osách. Pro usnadnění a urychlení nasazení pohonů a jejich řídicích algoritmů do provozu jsou uživatelům k dispozici sledovací funkce (trace), které umožňují diagnostikovat a optimalizovat běh řídicích algoritmů a chování pohonů v reálném čase. Další technologickou vlastností je PID regulace, která využívá snadno nastavitelných funkčních bloků, což zajišťuje vysokou účinnost regulačních procesů. Automaticky jsou nastavovány optimální hodnoty parametrů PID regulátorů.

## **Zabezpečení**

Bezpečnostní koncept integrovaný do systému Simatic S7-1500 zahrnuje ochranu jednotlivých bloků až po zajištění integrity celé komunikace. Tento systém aktivně podporuje uživatele při ochraně jejich investic. Klíčovou roli v tomto zabezpečení hrají výkonné funkce, které chrání know-how, například zabránění neoprávněného kopírování programu. Každý programový blok PLC je propojen s jedinečným sériovým číslem použité paměťové karty, což zaručuje, že programy mohou být použity pouze s korektně nakonfigurovanou paměťovou kartou, brání tak neoprávněnému kopírování. Ochrana aplikačního programu před neoprávněnými změnami konfigurace je zajištěna pomocí chráněného přístupu. Systém umožňuje přidělování různých práv různým uživatelským skupinám a obsahuje speciální mechanismy pro rozpoznávání modifikovaných dat, což efektivně brání neoprávněné manipulaci s daty a informacemi řídicí jednotky.

## **Funkční bezpečnost**

Funkční bezpečnost v řídicím systému Simatic S7-1500 je integrována jako klíčová součást s celkově komplexním přístupem. Zajištění bezpečnosti strojů a technologických zařízení je častým požadavkem a použitím jedné platformy lze výrazně snížit náklady. Stačí vybrat bezpečné varianty, tzv. F-verze (failsafe), nových řídicích jednotek, které využívají stejné vývojové prostředí a způsob ovládání jako standardní programy, což platí i pro programy související s funkční bezpečností.

## **Konstrukce a ovládání**

Při vývoji systému Simatic S7-1500 bylo kladen důraz na jednoduché ovládání a maximální uživatelskou přívětivost. To vedlo k mnoha vylepšením, včetně přehledného displeje pro snadné ovládání a zobrazení detailních informací. Standardizovaný čelní konektor umožňuje bezpečné připojení signálních vodičů a jejich následné přesunutí do pracovní polohy. Doplňkové komponenty, jako jsou jističe nebo relé, lze snadno montovat na přístrojovou lištu DIN, která je integrována do profilové lišty systému S7-1500. Každá jednotka CPU řady S7-1500 může být rozšířena až o 32 rozšiřujících modulů, což umožňuje modulární konfiguraci podle potřeb uživatele.

## **Systémová diagnostika**

Součástí systému Simatic S7-1500 je široká škála diagnostických funkcí, které jsou k dispozici bez potřeby jakéhokoliv dalšího programování. Potřebné diagnostické funkce lze jednoduše konfigurovat místo programování. Zároveň je zaveden jednotný způsob zobrazení diagnostických informací. Hlášení, jako například o chybách v pohonech nebo o závadách, se zobrazují jako textové zprávy na displeji CPU a stejně tak i všude jinde - v prostředí TIA Portal, na uživatelském rozhraní (HMI) a na webovém serveru.[8]

## 2.1.2 Moduly

### Procesorové jednotky

Procesorové jednotky jsou označené kódy, které značí odstupňování výkonnosti. Všechny jsou dostupné také v safety verzi F pro bezpečnostní úlohy. Jednotky se od sebe liší například počtem rozhraní, výpočetní rychlostí, rozměry displeje a kapacitou paměti pro uložení dat.[13]



Obrázek 2.1: CPU 1517F-3 [10]

### Signální moduly

Signálové nebo periferní moduly (ET 200 MP) slouží jako rozhraní mezi řídicím systémem a řízeným systémem. [14]

#### Charakteristiky :

- Využívají PROFINET IRT (isochronous real-time) pro dosažení maximální kvality řízení s nejdelším hodinovým pulzem 250 mikrosekund.
- Moduly digitálních vstupů mají nejkratší periodu při čtení 50 mikrosekund.
- Rychlé analogové moduly mají dobu převodu 62,5 nebo 125 mikrosekund každé pro 8 kanálů.
- Analogové vstupní moduly mají nezávislou linearizaci charakteristik pro měření teploty a limitních monitorování

## Technologické moduly

Technologické moduly (TM) předpracovávají signál na hardwarové úrovni, zajišťují rychlé čítání, měření a zaznamenávání polohy pro řadu kódovacích zařízení. Jsou určeny pro SIMATIC S7-1500 a ET 200MP. Moduly mají časově řízené vstupy a výstupy pro přesné funkce spínání s dobou odezvy několik mikro sekund. Modul PTO (Pulse Train Output) slouží pro řízení krokových motorů. [13]

## Komunikační moduly

Komunikační moduly rozšiřují komunikační možnosti systému SIMATIC S7-1500 pomocí dalších komunikačních funkcí či více rozhraní. Komunikační moduly zvyšují flexibilitu a automatizační výkon u systémů využívající SIMATIC S7-1500 nebo ET 200MP. Přidáním komunikačního modulu přibude i další použitelná adresa gateway.

### Typy komunikačních modulů :

- Moduly s připojením pro průmyslový ethernet
- Moduly s připojením pro PROFIBUS
- Moduly pro výměnu dat pomocí point to point spojením

## 2.2 Simatic IPC panely

Portfolio panelových počítačů SIMATIC je optimální volbou pro vizualizační úlohy přímo na stroji nebo v provozu. Zařízení typu "vše v jednom" Panel PC spojují průmyslový počítač s operátorským panelem, a to s vysokou účinností. Jsou robustní, výkonné a mají kvalitní displej. Různé rodiny zařízení pokrývají širokou škálu požadavků v oblasti automatizace výroby a procesů. [12]

### 2.2.1 Hardwarová Architektura

#### Procesorová architektura

Simatic IPC Panely jsou vybaveny výkonnými procesory, které zajišťují rychlý a spolehlivý výkon pro průmyslové aplikace. Tyto procesory jsou optimalizovány pro běh náročných operačních systémů a řídicích aplikací. V závislosti na náročnosti aplikace si uživatel může vybrat od Intel Atom až po Intel Xeon.

#### Paměťové systémy

IPC Panely mají integrované paměťové systémy s dostatečnou kapacitou pro ukládání operačních systémů, aplikací a uživatelských dat. Různé modely mohou nabízet různé úrovně paměti podle potřeb konkrétní aplikace.



## **Rozhraní a konektivita**

IPC Panely mají širokou škálu rozhraní pro připojení k perifériím a externím zařízením, včetně Ethernetu, USB, sériových portů a dalších. To umožňuje flexibilní a snadné propojení s dalšími zařízeními v průmyslovém prostředí.

## **Displej a ovládací prvky**

Každý IPC Panel má integrovaný dotykový displej a uživatelské ovládací prvky, které umožňují interakci s operačním systémem a řídicími aplikacemi. Displeje mají vysoké rozlišení a jsou odolné vůči prachu, vlhkosti a mechanickému poškození. Nabídky úhlopříček obrazovky jsou vysoce škálovatelné v rozmezí 4 až 24 palců.

## **Robustní konstrukce**

IPC Panely jsou navrženy pro průmyslové prostředí a mají robustní konstrukci, která zajišťuje odolnost vůči vibracím, teplotním extrémům a jiným nepříznivým podmínkám provozu v průmyslových zařízeních.

## **Rozšiřitelnost a modularita**

Některé modely IPC Panelů jsou navrženy s možností rozšíření pomocí rozšiřujících modulů nebo karet, což umožňuje přizpůsobení konkrétním potřebám aplikace a snadnou aktualizaci nebo vylepšení hardwaru. [11]

### **2.2.2 HMI software**

S rozhraním SIMATIC HMI lze implementovat rozsáhlé řešení průmyslových vizualizací, které kladou velmi vysoké požadavky na řízení a monitorování. Jedná se o efektivní nástroj při plánování a provádění inovativních řídicích opatření s jasným a efektivním zobrazením, a to přímo u zařízení, nebo pro sledování výrobních linek a složitých zařízení.

Softwarové systémy SIMATIC WinCC je možné nasadit od jednoduchých aplikací pro jednotlivé stroje s omezenými funkcemi až po komplexní SCADA řešení pro celé průmyslové závody, poskytují díky integraci do TIA Portalu vysokou účinnost a nabízejí škálovatelný a flexibilně rozšiřitelný rozsah funkcí pro široké spektrum aplikací. [9]

### 2.2.3 IPC 477 E Pro

Tento panel je využívám ve svařovnách Škoda-auto, které byly postavena na VASS standardu verze 6. Jedná se o výkoný, bezúdržbový, odolný vestavěný panelový počítač s univerzální konfigurací. Panel neobsahuje žádné rotační části (nemá ventilátory ani pevný plotnový disk). Vysoká bezpečnost je zajištěna díky operačnímu systému. Panel je od výrobce dodáván ready-to-use a má v sobě předinstalovaný software. Panel má vestavěnou retentivní paměť NV-RAM a lze ji použít se softwarovým řadičem. Verze pro oproti standardní obsahuje multi touch screen a konstrukce je plně uzavřená s certifikací IP65. [3] [2]

#### Parametry :

- Procesory: Intel Celeron, Core i3, Core i5, Xeon E3
- Disk: SSD 128/240/480 GB
- Ram 4/8/16 GB DDR4 s volitelnou ECC
- Retentivní paměť: 512 kbyte, 128 KB lze uložit do vyrovnávací paměti, volitelná
- 1x PCIe a 1x slot pro CFAST kartu
- 3x RJ45 port 1000 Mbps
- 4x USB 3.0
- Volitelný 2x COM port (RS 232/422/485), nastavitelný v BIOS
- Napájení 24V DC 65W



Obrázek 2.2: IPC 477E 19” [4]

## 2.3 Profinet

Profinet je průmyslová komunikační sběrnice navržená pro řídicí systémy v průmyslové automatizaci. Tento standard je postaven na technologii průmyslového Ethernetu.

Profinet využívá Ethernet jako svou přenosovou vrstvu, jak je specifikováno v normě IEEE 802.3. Jako součást síťové infrastruktury se používají síťové přepínače, také nazývané switche (značeny také jako komunikační mosty - bridges), které jsou definovány v normě IEEE 802.1D. Přepínač pracuje na úrovni druhé vrstvy komunikačního modelu ISO/OSI a pro koncová zařízení v síti je transparentní. To znamená, že každý přepínač přenáší zprávy z vstupního portu na výstupní port a koncová zařízení v síti nemají povědomí o existenci přepínačů. U funkce přepínače se předpokládá, že celá síťová architektura je založena výhradně na přepínačích.[5]

### Oblast použití

Profinet umožňuje integraci stávajících průmyslových sběrnicevých systémů, jako je například Profibus, bez potřeby modifikace a změny existujících zařízení. Tím je zajištěna ochrana současných investic zákazníka. V rámci komunikačního systému Profinet může zákazník využívat jedinou komunikační sběrnici pro všechny úrovně řízení. To přispívá ke snížení nákladů na skladování, instalaci, údržbu a školení personálu.

Kromě úspory nákladů poskytuje jednotná komunikační struktura založená na Ethernetu také široké možnosti pro technologické inovace. Například parametrizaci, diagnostiku a vizualizaci je možné provádět pomocí webových služeb. Díky dálkovému přístupu přes Ethernet lze nabídnout nový způsob poskytování servisu a celosvětové technické podpory, což přispívá k vyšší provozní spolehlivosti technologických zařízení. [7]

### Typy zařízení

**Systém Profinet se skládá z následujících zařízení :**

- IO-Controlleru, který řídí automatizační úlohu
- IO-Device, provozního zařízení, které je monitorováno a řízeno IO-Controllerem. IO-Device může být složeno z několika modulů a podmodulů
- IO-Supervisor, což je software obvykle běžící na PC, který slouží k nastavení parametrů a diagnostice jednotlivých IO-zařízení.

[6]

## 2.4 Tia Portal

Totally Integrated Automation Portal (TIA Portal) spojuje širokou škálu inženýrských nástrojů do jediné uživatelsky přívětivé platformy. Nabízí podporu pro digitalizaci pracovních postupů od digitálního návrhu přes integrovaný inženýring až po transparentní procesy a analytickou práci s daty v průběhu všech fází projektu.

### Flexibilní simulace automatizace

Využití simulace celé pracovní stanice, zahrnující PLC a HMI, a její vyhodnocení na základě virtuálního modelu výrazně urychluje proces uvedení do provozu díky PLCSIM Advanced, což je digitální replikou PLC.

### Spojení do jednoho prostředí

V TIA Portalu jsou integrovány všechny klíčové automatizační komponenty, což umožňuje programování široké škály prvků od PLC, periférií a pohonů až po HMI, bezpečnostní prvky, řízení polohy, pohybu a dokonce i správu energetických zdrojů.

[16]

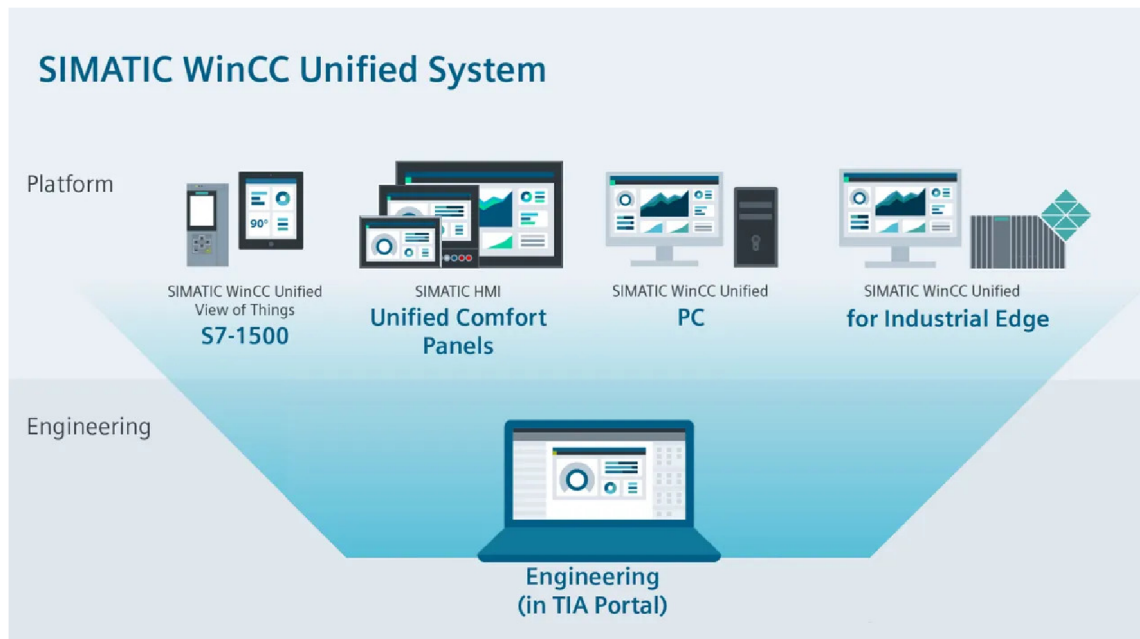
### Volby TIA Portalu :

- Simatic S7-PLCSIM Advanced
  - Komplexní simulace funkcí kontroléru včetně interakce s virtuálními modely zařízení a strojů
- TIA Portal cloud Connector
  - Umožňuje centrální správu inženýrského softwaru na serveru
- TIA Portal Teamcenter Gateway
  - Zabezpečení projektů v rámci celého závodu pomocí ukládání projektů TIA portálu v systému teamcenter
- Simatic target 1500S pro SIMULINK
  - Doplněk pro Simulink, který z modelu Simulinku vygeneruje spustitelný objekt pro procesor podporující ODK
- SIMATIC ODK 1500S
  - Podporuje vývoj funkcí systému Windows a reálného času a umožňuje integraci vysokoúrovňových jazyků
- SIMATIC STEP 7 Safety
  - Umožňuje bezproblémovou integraci bezpečnosti strojů do automatizačního systému SIMATIC.

- SIMATIC Visualization Architect SiVArc
  - Implementace standardizovaných vizualizací HMI s automatickým generováním
- SINAMICS Drive Control Chart (DCC)
  - Konfigurace a rozšíření funkcí zařízení pomocí volně dostupných řídicích, matematických a logických bloků
- TIA Portal Test Suite
  - Automatizované generování a ověřování programových pokynů jako generování a provádění aplikačních testů pomocí SIMATIC S7-PLCSIM Advanced
- SIMATIC OPC UA
  - Otevřený komunikační standard pro komunikační koncepty Průmyslu 4.0
- SIMATIC Energy Suite/S7 Energy Efficiency Monitor
  - Energetická transparentnost výroby díky automatické tvorbě a intuitivní konfiguraci
- SIMATIC ProDiag
  - Diagnostika chyb v aplikacích
- WinCC/WebUX
  - Diagnostika chyb v aplikacích [17]

## 2.5 WinCC

WinCC (TIA Portal) je software pro všechny aplikace HMI, od nejjednodušších provozních řešení se základními panely až po aplikace SCADA na víceuživatelských systémech na bázi PC. Inženýrský software SIMATIC WinCC (TIA Portal) je k dispozici ve verzích WinCC Basic, WinCC Comfort, WinCC Advanced a WinCC Professional.



Obrázek 2.3: WinCC Unified System [19]

### Generování

Zrychlení vývoje řešení je umožněno automatickým generováním vizualizací HMI pomocí SIMATIC Visualization Architect (SiVArc). Rychlá a jednoduchá diagnostika strojů a pracovišť je umožněna pomocí SIMATIC ProDiag, efektivní implementace energy managementu je zajištěna a programy jsou automaticky vytvářeny pomocí softwarových generátorů, které jsou připojeny prostřednictvím rozhraní TIA Portal Openness.

### Diagnostika

Při projektování diagnostiky strojů a zařízení je prostřednictvím SIMATIC ProDiag šetřeno programování v CPU a podporováno odstraňování problémů přímo na HMI. Zprávy s informacemi o typu monitorování, místě a příčině poruchy i o jejím odstranění jsou obdrženy, a možná porucha může být identifikována a odvrácena včas. [19]

## 2.6 Programovací standardy

Programovací standardy představují klíčový prvek v procesu vývoje softwaru, zajišťující jednotný a konzistentní přístup k psaní, organizaci a dokumentaci kódu. Tyto standardy definují pravidla a doporučení, která programátoři následují při vývoji softwaru, aby se zlepšila jeho čitelnost, udržitelnost a znovupoužitelnost.

Pokud jsou standardy kódování správně definovány a implementovány, uživatelský kód může být snadno orientován vývojáři, dokonce i těmi, kteří právě nastoupili do týmu. Za ideálních okolností je žádoucí, aby zdrojový kód vypadal tak, jako by byl psán, laděn a udržován jediným vývojářem.

### Programovací konvence

Programovací konvence jsou pokyny pro konkrétní programovací jazyk, které zahrnují doporučení pro udržení čistoty kódu. Tato doporučení se primárně soustředí na osvědčené postupy v oblasti stylu programování a organizace, které pomáhají udržovat konzistenci, ale zároveň umožňují určitou míru flexibility.

Různé programovací jazyky a komunity mohou mít své specifické konvence, ale zde je několik běžných kódovacích konvencí, které jsou často používány:

- **Formátování:** V celé databázi kódu dodržujte jednotné formátování.
- **Odsazení:** Pro lepší čitelnost se používají jednotné odsazení (obvykle 2 nebo 4 mezery).
- **Pojmenovací konvence:** Používají se popisné názvy pro proměnné, funkce, třídy atd. (např. `JmenoPromenne`, `jmeno_promenne`).
- **Závorky:** Použití důsledných umístění závorek a závorek.
- **Komentáře:** Jasné a výstižné komentáře ke složitým částem kódu.
- **Bílý prostor:** Efektivní používání bílých znaků zvyšuje čitelnost kódu. Vyhněte se nadměrnému nebo zbytečnému množství bílých znaků.
- **Délka řádku kódu:** Omezte délku řádků, aby se zlepšila čitelnost. Mnoho stylistických příruček doporučuje, aby řádky nepřesahovaly 80 nebo 120 znaků.

[1]

## 3 VASS standard

VASS standard slouží jako návod při vytváření programových bloků, nebo jako konfigurační nástroj pro již hotové bloky na zadanou specifickou úlohu.

Popisuje požadavky z pohledu koncernu Volkswagen na automatizační systém, software, pospup konfigurace SIMATIC TIA STEP 7 a WinCC. Podklady k VASS standartu byly čerpány v literatuře[18].

### Základní body :

- Zajištění kvalitního celkového výsledku
- Zajištění vysoké úrovně dostupnosti a flexibility celého systému
- Minimalizace nutných investičních, provozních a rozšiřovacích nákladů
- Optimalizace pracovní náplně a podmínek pro skupinu uživatelů
- Modularizace celého komplexního řídicího systému na snadno vyměnitelné jednotky
- Zjednodušení a standardizace rozhraní mezi jednotlivými moduly

### Cílová skupina

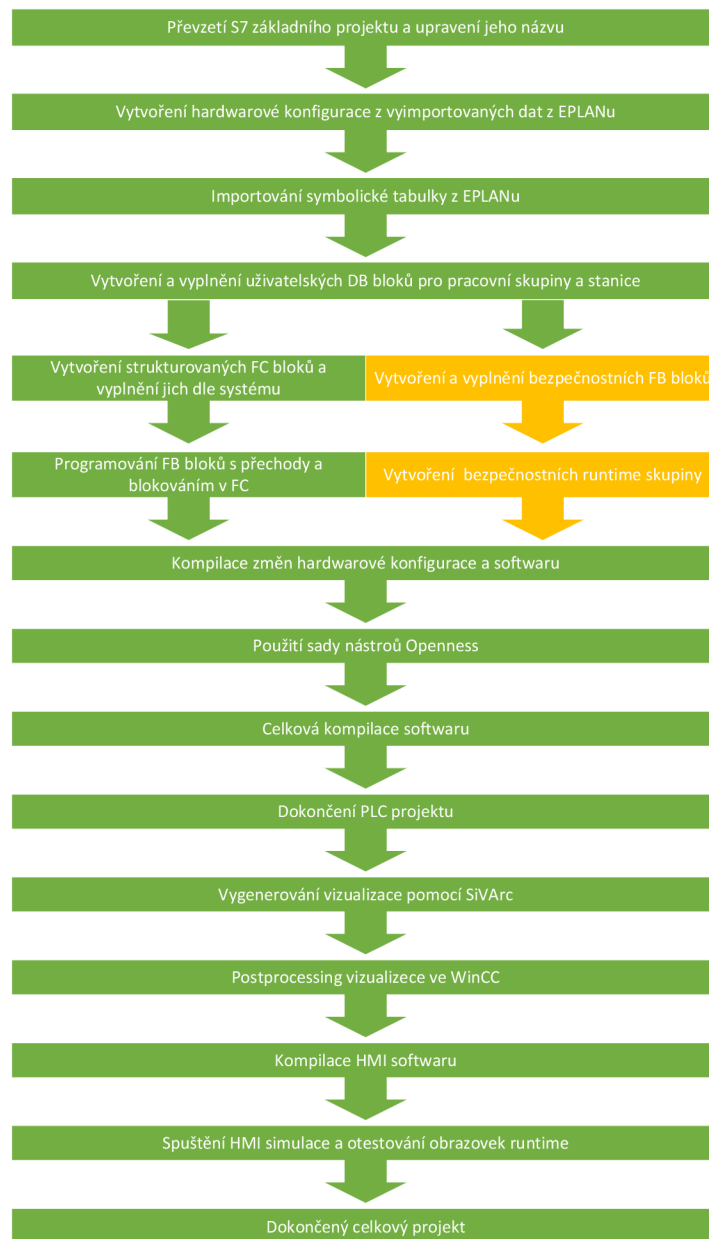
Příručka VASS standartu je určena všem provozovatelům systémů, projektovým manažerům, projektovým inženýrům, údržbářům a servisním pracovníkům, kteří potřebují vytvářet nebo upravovat projekty v kontextu normy VASS. Poskytuje řešení pro realizaci automatizačních projektů pomocí SIMATIC TIA STEP 7 a WinCC.

### Potřebné softwarové nástroje :

- SIMATIC STEP 7 Professional + Safety Advanced V17 UPD5
- SIMATIC WinCC Advanced V17 UPD5
- SIMATIC WinCC Panel Images V17 UPD5
- STEP 7 PLCSIM V17 UPD5
- SIMATIC Visualization Architect V17 UPD5
- TIA Openness V17 UPD5
- SIMATIC WinCC Runtime Advanced 8192 PowerTags V17 UPD5
- SIMATIC ProDiag for WinCC Runtime Advanced
- SINAMICS Startdrive V17
- SINAMICS DCC



## 3.1 Vytvoření projektu

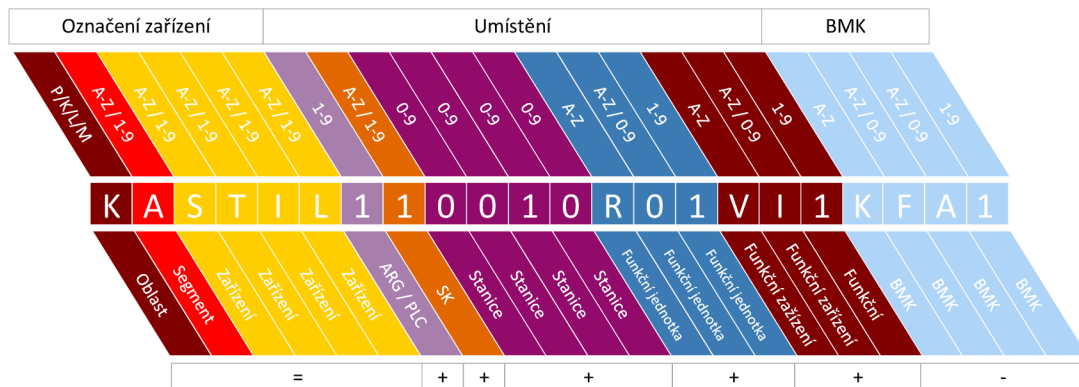


Obrázek 3.1: Diagram vytvoření projektu

## 3.2 Profinet

### Struktura názvů zařízení PROFINET

Základem pro vytvoření PROFINET hardwarové konfigurace je důležité pojmenování profinet zařízení. Za tímto účelem byla vytvořena konvence pro pojmenovávání. Každé profinet zařízení obdrží 22 znakové profinet jméno, které vychází z EPLAN dokumentace.



Obrázek 3.2: Struktura názvů zařízení PROFINET

### Import AutomationML z EPLANu

Ve fázi návrhu hardwaru je obvyklé, že v dokumentech schémat zapojení jsou definovány součásti, nastavení rozvaděčů s přiřazením modulů do slotů a topologie sítě s propojením portů široké škály zařízení. Rozhraní AML mezi EPLAN Electric P8 a TIA umožňuje obousměrnou výměnu těchto dat. Jako pomůcku při vytváření konfigurace hardwaru lze data ze systému EPLAN exportovat ve formátu AML a importovat do portálu TIA.

## 3.3 VASS definice funkcí a funkčních bloků

FB a FC jsou základními jednotkami programování v systémech PLC. FC jsou podprogramy pro strukturalizaci programu. Mohou být s nebo bez parametrů a pracují s globálními i lokálními proměnnými. Lokální proměnné však nejsou ukládány do dalšího bloku. FB jsou parametrizovatelné části programu. Paměť proměnných mají uloženou v pevně přiřazeném datovém bloku.

Ve standardním softwaru VASS se rozlišují typy FC a FB, které jsou popsány níže.

## Strukturované FC

Jedná se o bloky, které slouží ke strukturování uživatelského programu. Obecně to jsou instance pro řízení akčních členů. Tyto bloky slouží typicky k volání modulových FB. Strukturované funkce (FC) se volají z OB1.

## Modulové FC

Bloky pro ovládání funkčních modulů, které vykonávají určitou funkci. Modulové FC (od FC150) jsou obecně integrovány do strukturovaných FC z knihoven.

## Strukturované FB

Tyto bloky slouží ke strukturování uživatelského programu. Využívají se pro řízení systémových procesů pomocí sekvencí (jazyk GRAPH).

## Modulové FB

Bloky ovládající funkční moduly akčních členů a jednotek nebo jiné FB bloky, které provádějí specifickou funkci (např. počítání výrobního cyklu, komunikace s robotem, systémová hlášení). Modulové FB jsou zpravidla integrovány jako jednotlivé instance ve struktuře FC z knihoven a začínají na hodnotě 100 (FB100).

## 3.4 Pravidla pro generování vizualizace

Vzhledem k tomu, že vizualizace je ve velké míře generována automaticky na základě analýzy projektu S7 pomocí nástroje SiVArc, je nutné dodržovat určitá pravidla při vytváření projektu.

- **Symbolické pojmenování strukturovaných FC:** Symbolické názvy strukturovaných FC se zadávají prostřednictvím vlastností objektu.
- **Názvy instancí a deklarační komentářů modulových FB:** Název instance volání jedné instance se definuje při integraci modulů FB do uživatelského programu.
- **Propojení tagů na vstupech modulových FB:** Pro některá přiřazení ve vizualizaci jsou důležité značky nakonfigurované na rozhraní příslušných modulů FB ("FB\_No halt", "FB\_Meld\_4" a "FB\_Schutzkreis").
- **Úvod k zobrazení textu na čelním panelu:** Ve STEP 7 je modulové FB voláno ve strukturovaném FC. Každému modulovému FB je přiřazena instance DB. V TIA Portalu je třeba ve vlastnostech bloku zadat symbolický název. Komentář se musí zadat ve vlastnostech bloku v části Informace.

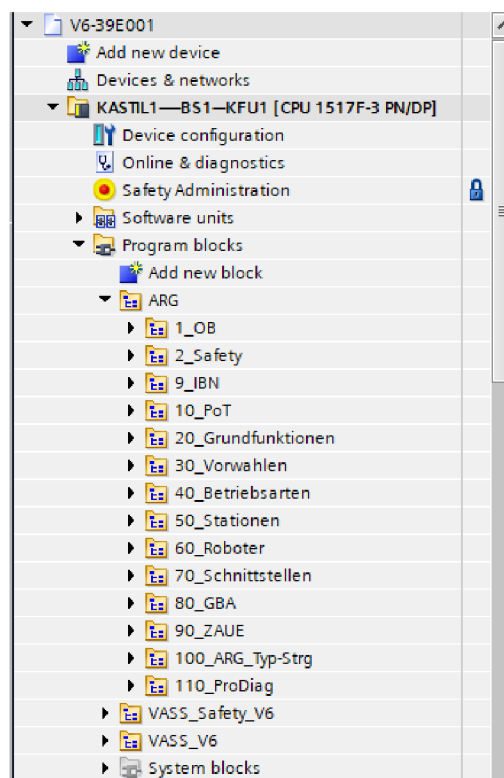
Pro příklad "FB\_Status" :

- Symbolické jméno : 110010V01\_FM#FB\_Status\_DB
- Informace - komentář : Overall

## 3.5 Struktura projektu TIA Portal

### Strom projektu

Následující strom s příslušnými složkami je zadám v základním projektu a používá se ve všech projektech. Bloky ve složkách v ukázkovém obrázku 3.3 odkazují na reference a mohou se lišit v závislosti na systému. Označení jednotlivých složek je pevně ukotveno.



Obrázek 3.3: Strom projektu

### OB1

Všechny bloky s cyklickým voláním jsou uloženy v této složce. Tyto bloky slouží jako rozhraní mezi systémem a uživatelským programem. Systémový program CPU volá organizační bloky při jednoznačně definovaných událostech, jako je restart, časové přerušení nebo chyba systému. Hlavní program je obsažen v OB1, ostatní bloky mají přidělena čísla podle spouštěcích událostí. Číslo OB určuje prioritu volání (nižší číslo - vyšší priorita).

### Safety

Do této složky se ukládají programy z bezpečnostní oblasti.

## **PoT**

Všechny bloky "výroba bez dílu" se ukládají zde.

## **Grundfunktionen**

Složka pro bloky z oblasti základních funkcí vyšší úrovně (obsahuje funkce nadřazené celé lince, například diagnostiku, chyby, ovládání mobilních panelů).

## **Vorwahlen**

Složka pro bloky z oblasti předvolby vyšší úrovně (obsahuje funkce nadřazené celé lince, například zapnutí ovládacího napětí, stlačeného vzduchu).

## **Betriebsarten**

Ve složce jsou uloženy všechny bloky pro provozní režimy (obsahuje například vyhodnocení tlačítka nouzového zastavení a spínačů pohonů).

## **Stationen**

Do této složky spadají bloky stanic (programové sekvence), konkrétně bloky FC20 až FC99 (například bloky ovládající otočné stoly, vzduchové terminály).

## **Roboter**

Složka pro bloky robotů, jedná se o bloky FC100 až FC119.

## **Schnittstellen**

Do této složky patří bloky rozhraní pro výměnu dat s okolními systémy FC130 až FC139.

## **GBA**

Do této složky patří bloky pro velkoplošné obrazovky.

## **ZAUE**

Bloky centrálního monitorování systému. Odesílají data do nadřazeného systému.

## **ARG\_Typ\_Strg**

Blok pro správu typů, jedná se o blok FC140.

## **ProDiag**

Všechny bloky se systémovými alarmy.

## 3.6 SiVArc

SiVArc (SIMATIC Visualization Architect) je doplňkový balíček ve WinCC pro TIA Portal. Pomocí SiVArc lze generovat vizualizace pro jedno či více HMI zařízení z programových bloků a pomocí uživatelsky vytvořených šablon obrazovek.

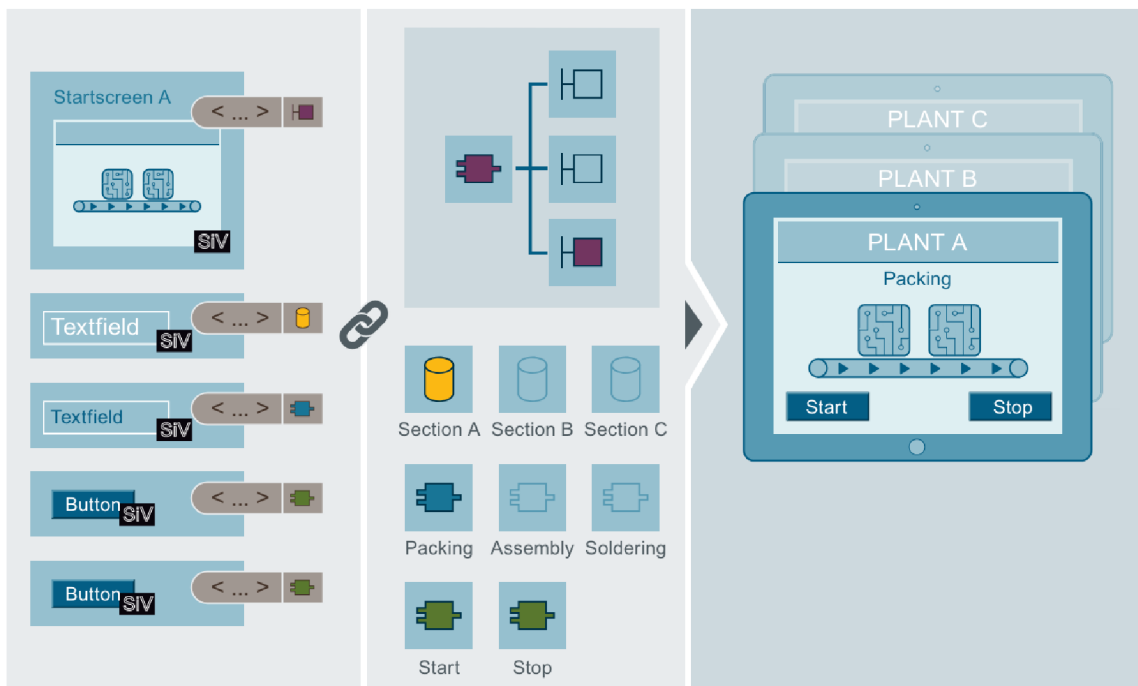
Pro správné umístění a segmentaci generovaných bloků slouží pravidla pro generování.

Nástrojem SiVArc je možné generovat následující objekty HMI z dat řídicí jednotky:

- Obrazovky, faceplaty, výběry zobrazovacích a ovládacích prvků
- Externí tagy
- HMI text listy

Bez nutnosti odkazu na řídicí program je možné pomocí SiVArc vygenerovat výběr objektů z knihovny projektu WinCC a použít je v projektu WinCC jako instance.

SiVArc může simultánně vytvářet vizualizace pro více HMI zařízení, více PLC a proxy zařízení. Během tvorby vizualizace s pomocí nástroje SiVArc je možné pokračovat v práci s TIA Portal v další instanci. Díky nástroji SiVArc a funkci "TIA Portal Multiuser" je také možné zapojit do práce na projektu SiVArc různé uživatele.[15]



Obrázek 3.4: SiVArc [15]

### 3.6.1 Přípravné kroky

#### Označení názvů HMI zařízení pro více operátorských stanic

Příklad značení při kompletní generaci vizualizace na 2 operátorských stanicích :

- Kxyyyyz—IA1—KFP1#1 (pro operátorskou stanicí 1)
- Kxyyyyz—IA2—KFP1#2 (pro operátorskou stanicí 2)

Čísla na konci názvu reprezentují číslo operátorské stanice a používají se v případě, když se generuje celá generace vizualizace.

Příklad značení při částečné generaci vizualizace dělené mezi 2 operátorské stanice :

- Kxyyyyz—IA1—KFP1#1#T
- Kxyyyyz—IA2—KFP1#2#T

#### Rozložení

Rozložení stanic má odlišnou strukturu. Ve VASS standardu existují různá uspořádání umístění objektů na obrazovce pro "Předvolby", "Stanice", "Válečkový dopravník", "Akumulační dopravník", "Věžový buffer", "Správa typů", "Framer" a "FT" (dopravníkové technologie).

Pro správné umístění objektů na obrazovce závisí na záznamu v poli položky "Structure FC" části "Plug-ins".

Při použití více operátorských stanic se názvy operátorských stanic zkracují, například "AIT1", aby bylo možné umístit HMI objekty na příslušnou operátorskou stanicí.

#### Postup při použití různých rozložení stanic :

1. Otevření struktury FC
2. Vstup do záložky "Plug-ins"
3. Zadání požadovaného textu (např. "Vorwahlen", "Betriebsarten", "Roboter", "RobSafe", "Schnittstellen", "GBA", "ZAU" a pod.) v záložce SiVArc - "Tag definitions"
4. Pro umístění detailních stránek, například bloku pohonů (Antrieb) se kromě tagu definice je třeba zadat také název přidruženého čelního štítu (např. SEW\_Modulo) do pole příslušné sítě.
5. Umístění provozních režimů pro příslušnou operátorskou stanicí (pouze dopravníkové systémy). Je třeba vytvořit odpovídající tag, který identifikuje HMI zařízení (zkratka pro operátorskou stanicí) a musí být dodatečně zadán v příslušném řádku.
6. Uložení a kompilace projektu

### 3.6.2 Generace SiVArc

Během generování vizualizace se generují obrazovky, objekty HMI a externí tagy. Obrazovky HMI a objekty obrazovek lze generovat pouze v případě, když jsou pravidla SiVArc zahrnuta v projektu. Pokud nejsou definována žádná pravidla, SiVArc generuje pouze externí tagy.

Pokud projekt obsahuje několik HMI zařízení nebo připojených PLC, SiVArc generuje vizualizaci pro HMI zařízení od připojených PLC, která jsou vybrána v editorech pravidel.

Dialog pro výběr stanice se zobrazí při prvním spuštění generování v projektu. Zde se vybere příslušné zařízení pro generování.

Generování vizualizace probíhá postupně, zařízení po zařízení. Pokud zařízení nelze vygenerovat, SiVArc pokračuje v generování dalšího zařízení. Pokud uživatel generování zruší, vizualizace s již kompletně vygenerovanými zařízeními zůstane zachována.

Pro každou dodatečnou generaci vizualizace s výběrem stanice je důležité všimnout si následujícího :

- Pokud zařízení není nadále vybráno pro další generaci v dialogovém okně, vygenerované objekty a ruční změny zůstanou v projektu zachovány.
- K odstranění vygenerovaných objektů PLC, které již nejsou vybrány pro další generaci, je nutné odstranit propojení mezi PLC a HMI zařízením.
- Pokud dojde k odstranění PLC, ke kterému byla již vygenerovaná vizualizace, všechny vygenerované SiVArc objekty spojené s tímto PLC budou při dalším generování rovněž odstraněny.
- Pokud je blokové volání v uživatelském programu smazáno a generování SiVArc je spuštěno znovu, SiVArc objekty, které byly vygenerovány pro tento blok, budou také smazány.
- Projekt by neměl obsahovat žádné "snímky skutečných hodnot" datových bloků. To může vést k chybě při generování. (opraveno ve verzi TIA V16 UPD2)

Dialog pro výběr stanice se zobrazí vždy při prvním generování SiVArc v projektu.

Tento dialog se při dalším generování nezobrazí. SiVArc pak generuje stejná zařízení HMI a PLC jako při předchozím generování.

Pro změnu nastavení je nutné vybrat projekt nebo zařízení ve stromu projektů a následně v kontextové nabídce Runtime vybrat možnost "Generování vizualizace" a "Generování s výběrem stanice".

Aby byla zajištěna konzistence generovaných dat, v případě shodných názvů se nově generované objekty vytváří vždy pod názvem, který má být nově vygenerován.



### 3.6.3 Generovací matice

Generovací matici lze použít pro následné změny při přiřazování generovaných objektů. Umožňuje rozdělit vygenerované prvky (obrazovky a faceplaty) mezi další operátorské stanice nebo obrazovky.

Vygenerované obrazovky a objekty obrazovek pro HMI zařízení nebo typové zařízení jsou zobrazeny v okně editoru "Generation matrix" po každém generování.

Dále je možné upravit přiřazení následujícím způsobem:

- Vygenerováním objektu obrazovky v jiné obrazovce
- Vygenerováním obrazovky v jiném HMI zařízení

## 4 Vytvoření diagnostického nástroje

Pro testování byla sestavena stanice z plc Simatic 1517F, switche Scalance x202-2p a decentrální periferie ET 200SP. Stanici bylo potřeba postavit hlavně, protože simulátor nebyl schopen dodat všechny potřebné funkce k práci.



Obrázek 4.1: Testovací soustava

### 4.1 Analýza bloků v TIA Portalu

Při hledání dat důležitých pro rozšiřitelnost stávajících výrobních linek bylo hledání směřováno k instrukčním blokům v TIA Portalu a VASS blokům. V TIA Portalu se nejvhodnější bloky nacházely v záleže diagnostické bloky. U VASS bloků hledání začalo ve funkci "Grundfunktionen".

## 4.1.1 Diagnostické bloky

Diagnostics	
RD_SINFO	Read current OB start information
RT_INFO	Read runtime statistics
LED	Read LED status
Get_IM_Data	Reading identification and maintenance data
Get_Name	Reading the name of a module
GetStationInfo	Read information of an IO device
GetChecksum	Read out checksum
GetSMCInfo	Read out information about the memory card
GetClockStatus	Read out status of the CPU clock
DeviceStates	Read module status information in an IO system
ModuleStates	Read module status information of a module
GEN_DIAG	Generate diagnostic information
GET_DIAG	Read diagnostic information

Obrázek 4.2: Knihovna diagnostických bloků

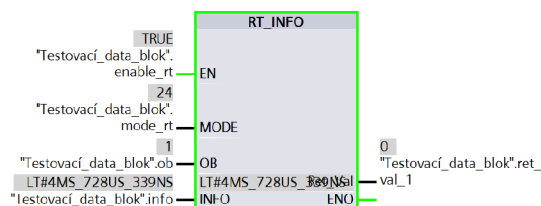
V této kapitole je dán bližší popis k výše uvedeným blokům, které jsou relevantní pro řešení práce.

### RT\_INFO

Tato instrukce slouží ke generování statistik o době běhu konkrétního organizačního bloku, komunikace nebo uživatelského programu. Pro výběr parametru slouží vstup mode.

Konkrétně lze sledovat aktuální, minimální a maximální dobu běhu konkrétního OB, průměrnou hodnotu v procentech celkového času běhu využitého OB s vyšší prioritou, průměrnou hodnotu v procentech celkového času běhu využitou komunikací, nejkratší/nejdelší doba cyklu od posledního startupu, aktuální/předchozí doba cyklu.

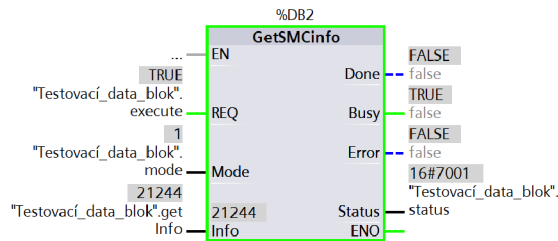
Na obrázku 4.3 je zapojen blok RT\_INFO v módu pro sledování nejnižší doby cyklu.



Obrázek 4.3: Blok RT\_INFO

### GetSMCInfo

Instrukce slouží k získání informací o vložené paměťové kartě v PLC. Selektor parametru je vstup MODE. Z bloku lze vyčíst velikost karty a její obsazenost.



Obrázek 4.4: Blok GetSMCinfo

## 4.1.2 VASS bloky

Ze standardizovaných bloků lze vyčíst počet účastníků sítě, připojených i udávaných podle HW konfigurace. Tyto informace se nacházejí v bloku FB Steuerung. Z bloku FB Diagnose SNMP lze vyčíst jednotlivé IP adresy aktivních zařízení. Níže je uveden úsek kódu, který je využit v práci a vychází z bloku FB steuerung.

SPNS#FB_Steuerung_DB				
	Name	Data type	Start value	Monitor value
66	iPN_Soll	int	0	2
67	iPN_ist	int	0	2

Obrázek 4.5: Počet účastníků sítě

```

1 #RetVal_konfiguriert := DeviceStates(LADDR := "Local~PROFINET_IO-
2 System", MODE := 1, STATE := #State_konfiguriert);
3 #RetVal_stoerung := DeviceStates(LADDR := "Local~PROFINET_IO-System",
4 MODE := 2, STATE := #State_stoerung);
5 #RetVal_deaktiviert := DeviceStates(LADDR := "Local~PROFINET_IO-
6 System", MODE := 3, STATE := "DB_ARG".State_deaktiviert);
7 #RetVal_vorhanden := DeviceStates(LADDR := "Local~PROFINET_IO-System"
8 ,MODE := 4, STATE := #State_vorhanden);
9 #RetVal_wartung := DeviceStates(LADDR := "Local~PROFINET_IO-System",
10 MODE := 5, STATE := #State_wartung);
11 IF (#RetVal_konfiguriert = 0) AND (#RetVal_deaktiviert = 0) AND
12 (#RetVal_vorhanden = 0) THEN
13     #HMI_UDT.PN_ist := 0;
14     #iPN_Soll := 0;
15     #iPN_Abg := 0;
16     FOR #iZaehler := 1 TO 512 DO
17         IF #State_vorhanden[#iZaehler] THEN
18             #HMI_UDT.PN_ist := #HMI_UDT.PN_ist + 1;
19         END_IF;
20         IF "DB_ARG".State_deaktiviert[#iZaehler] THEN
21             #iPN_Abg := #iPN_Abg + 1;
22         END_IF;
23         IF #State_konfiguriert[#iZaehler] THEN
24             #iPN_Soll := #iPN_Soll + 1;
25         END_IF;
26         "Profinet_TNDeaktiv_DB".TNDeaktiv[#iZaehler] :=
27         #State_vorhanden[#iZaehler];
28     END_FOR;
29 END_IF;

```

### 4.1.3 Stahování dat vytíženosti paměti

Ze strany zadavatele práce byl požadavek na zobrazení vytížení paměti CPU, protože při integracích výrobních linek se občas narazilo na problém překročení code work memory, tedy paměti programu. Tento problém vzniká z důvodu obsazení velkého množství programových bloků.

Po dlouhém bádání všemy možnými směry v nápovědě TIA Portalu, knihovnách, Siemens support nebyl nalezen blok, který by přímo dokázal vyčítat vytížení paměti pro SIMATIC S7-1500. Byla nalezena pouze zmínka na diagnostický blok RDSYSST, který byl obsažen v předchozím vývojovém prostředí STEP 7 a tuto predispozici měl, ale bohužel nebyl implementován do TIA Portalu.

Způsoby, jak vyčíst vytížení paměti, jsou dva a to pomocí online diagnostiky v TIA Portalu, nebo za pomoci vytvoření web serveru.

#### Řešení problému

Při řešení bylo přistupováno k problému z druhé strany od známých dat. Respektive vyčítat data z web serveru a posílat je nazpátek do PLC.

Byl tedy vytvořen již zmiňovaný web server a VB skript v TIA Portalu, který poběží na vizualizačním panelu kde bude stahovat zdrojový kód web serveru, ze kterého vybere požadované data a zapíše jejich hodnoty pomocí HMI tagů do datového bloku PLC.

Listing 4.1: Ukázka zdrojového kódu pro stahování zdrojového kódu

```
1
2 Dim site
3 site = "http://172.21.1.10/Portal/Portal.mwsl?PriNav=Online&SecNav=
4 Memory"
5 Dim temp_path
6 temp_path = "D:\temp.txt"
7
8 Dim req
9 Set req = CreateObject("MSXML2.XMLHTTP.6.0")
10 req.open "POST", site, False
11 req.setRequestHeader "Content-Type", "application/x-www-form-urlencoded"
12 req.send
13
14 Dim stream
15 Set stream = CreateObject("ADODB.Stream")
16 stream.Open
17 stream.Type = 2 'text
18 stream.Position = 0
19 stream.Charset = "utf-8"
20 stream.WriteText req.responseText
21 stream.SaveToFile temp_path, 2
22 stream.Close
```

## 4.2 Vytvoření VASS modulu

### 4.2.1 Práce v základním VASS projektu

V základním projektu VASS musí být použita všechna požadovaná rozšíření, aby byla zajištěna plná funkčnost systému VASS.

#### Vložení GSDML souborů do hardwarové konfigurace

GSDML soubory dodává výrobce zařízení, nebo je lze stáhnout ze servisní stránky. Pro jejich přidání je tlačítko v záložce nastavení.

#### Nastavení zařízení v hardwarové konfiguraci

Zařízení je nutné nalézt v katalogu a přetáhnout ho do obrazovky konfigurace. Zařízením je potřeba v nastavení přiřadit IP adresu, jméno, číslo, dále v případě možnosti i vstupní/výstupní adresy.

#### Práce v TIA knihovně

Před vytvořením bloku do nového modulu je nutné nastavit následující specifikace:

- Číslo bloku
- Jméno bloku
- Datová struktura jména (podle potřeby)
- Verze bloku

Číslovací systém pro verze bloku je následující: Verze x.y.z

- x: hlavní verze (je inkrementován standardizátorem v případě, že dojde k úplné změny verze v důsledku rozšíření funkce)
- y: podverze (je inkrementován standardizátorem v případě změn, které byly provedeny v rámci opravy chyby)
- z: index změny (je inkrementován v případě změn kódu na stránkách)

### 4.2.2 Vytvoření FB modulu v jazyce SCL

#### Přidání nového bloku

V navigaci projektu vyhledejte odpovídající složku v části "Program blocks", podsložce "VASS\_V6" a pravým tlačítkem myši vyberte "Add new block".

Ve vyskakovacím okně přiřadte názvu předponu "FB\_". Programovací jazyk zvolte SCL. Číslo bloku přiřadte manuálně dle následující tabulky 4.1 .

Standardní rozsah	Obecné	5100...5199
	Robot	5200...5249
	Robot procesy	5250...5299
	Strojní procesy	5300...5349
	Pohonová technologie	5350...5399
	Svařovací technologie	5400...5449
	Interní dopravníková zařízení	5450...5499
	Technologie	5500...5549
	Typ řízení	5550...5599
	ZAU	5600...5619
	PDD/PDE	5620...5639
	Sekvenční bloky	5640...5649
	Alarmové systémy	5650...5659
	Bloky nástrojů	5660...5669
Interní dopravníková technika	5700...5749	
Montáže	5750...5799	

Tabulka 4.1: Číslování FB bloků

### Vlastnosti bloku

Záložka "Information": Název bloku bez předpony "FB\_" by měl být zapsán do okna "User-defined ID". Jako autor se uvádí tvůrce nebo společnost. V záložce "Attributes" je třeba zvolit parametr "Optimized block access".

### Změna modulové komponenty na knihovní typ

Jednoduše přetáhneme blok do adresáře knihovny v záložce na pravé straně.

### Vytvoření tagů pro vizualizaci

Vytvořte HMI-UDT s požadovanými tagami. Název UDT je zapsán tak, že "HMI-UDT" je pevný a následuje ho název FB oddělený znakem "\_". V našem případě tedy "*HMI-UDT\_Systemanalyse*".

SiVArc rozpozná pouze řetězec "HMI-UDT". Nerozlišuje malá a velká písmena.

### Identifikátor v SCL

Identifikátory mohou být například názvy konstant, značek a FB s maximální délkou 24 znaků. Diakritika, vyhrazené znaky a slova nejsou v identifikátorech povoleny. První znak v identifikátorech musí být písmeno nebo podtržítka. Velikost písmen v textu se nerozlišuje.

Identifikátor označující datový typ musí být umístěn před názvem tagu.

Identifikátory:

- x: Byte
- b: Byte
- w: Word
- dw: Double word
  
- i: Integer
- di: Double integer
- r: Real
  
- p: Pointer
  
- ar: Array
- st: Structure : Time

HMI-UDT_Systemanalyse							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	iPN_ist	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	iPN_frei	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	ar_IpFree	Array[0..256] of Bool		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	ar_IpMantis	Array[0..2] of UInt		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	tLongests	LTime	LT#Ons	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	tShortests	LTime	LT#Ons	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	tCurrent	LTime	LT#Ons	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	diSMC_free	UDInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	diCode_mem	UDInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	diWork_mem	UDInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	diRetentive_mem	UDInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	tCycle_max	LTime	LT#Ons	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Obrázek 4.6: Vytvoření data type pro obrazovku a FB

### 4.2.3 Základní příkazy pro řízení programu

Stručná ukázka popisující funkce v programu.

Listing 4.2: Přřazení hodnot

```
1 #KKPxxR := #xSF_R;
```

Listing 4.3: Cyklus for

```
1 (/* Counting loop*/)
2 FOR #i := 0 TO 100 DO
3 #arrVariable[#i] := FALSE; (/* Statement section FOR */)
4 END_FOR;
```



```

1  (* Counting loop with step width */)
2  FOR #i := 0 TO 100 BY 2 DO
3  #arrVariable[#i] := FALSE; (* Statement section FOR */)
4  END_FOR;

```

Listing 4.4: Příkaz case

```

1  (* Create multiway branch */)
2  CASE #j OF
3  1: (* Statement section case 1 */)
4  ;
5  2..4: (* Statement section case 2 to 4 */)
6  ;
7  ELSE (* Statement section ELSE */)
8  ;
9  END_CASE;

```

Listing 4.5: Příkaz if

```

1  (* Execute condition */)
2  IF #Cfg.%X0 THEN (* Condition */)
3  #xStatement := #SFxxR; (* Statement section IF */)
4  END_IF;

```

```

1  (* Branch condition */)
2  IF #Cfg.%X1 THEN (* Condition */)
3  #xStatement := TRUE; (* Statement section IF */)
4  ELSE
5  #xStatement := FALSE; (* Statement section ELSE */)
6  END_IF;

```

```

1  (* Branch condition multiple times */)
2  IF #Cfg.%X2 THEN (* Condition */)
3  #xStatement := TRUE; (* Statement section IF */)
4  ELSIF #Cfg.%X3 THEN (* Condition */)
5  #xStatement := FALSE; (* Statement section ELSIF */)
6  ELSE (* Statement section ELSE */)
7  #xStatement := FALSE;
8  END_IF;

```

Zakázané příkazy:

- While ... DO ...
- Repeat ... Until ...
- Goto
- Return

Tyto příkazy se obecně v programování nepoužívají, protože jejich použitím v programu může dojít k zacyklení.

## 4.2.4 Vytvoření diagnostického bloku

V diagnostickém nástroji jsou použity jak bloky standardní knihovny TIA Portal, tak bloky VASS standardu.

Ze standardních bloků TIA knihovny jsou využívány bloky RT\_Info a GetSMCInfo (popsané v kapitole 3.1.1). RT\_Info je pomocí parametru MODE nastaveno tak, aby získávalo aktuální, nejdelší, nejkratší a maximální povolený čas cyklu PLC. Volání bloku GetSMCInfo je nastaveno tak, aby vracelo vytížení SD karty. Tato hodnota odpovídá hodnotě load memory, protože CPU při detekování vložené karty nahrává load data do karty namísto do interního úložiště.

Listing 4.6: Aplikační kód RT\_info

```
1 REGION (/**/)
2   #RetValueRT_1 := RT_INFO(MODE := 23, OB := 1, INFO :=
3   #HMI_UDT.tLongests);
4   #RetValueRT_2 := RT_INFO(MODE := 24, OB := 1, INFO :=
5   #HMI_UDT.tShortests);
6   #RetValueRT_3 := RT_INFO(MODE := 25, OB := 1, INFO :=
7   #HMI_UDT.tCurrent);
8   #RetValueRT_3 := RT_INFO(MODE := 30, OB := 1, INFO :=
9   #HMI_UDT.tCycle_max);
10 END_REGION
```

Z VASS bloků jsou využity zmíněné FB Steuerung a FB Diagnose SNMP z kapitoly 3.1.2 a upravené pro získání požadovaných dat.

Z bloku FB Steuerung je použit a upraven úsek kódu pro zjištění počtu připojených účastníků na profinet síti. Jeho použitý úsek je uveden ve výše zmíněné kapitole.

Z druhého bloku, FB Diagnose SNMP, je využit data blok, který udává pole aktivních účastníků sítě. Byl vytvořen algoritmus, který s tímto polem pracuje a vytváří nové, které udává informaci o volných IP adresách. V tomto algoritmu byly zohledněny IP adresy, které nesmí být obsazené a jsou vynechány. Jako další výstup, doplněním k poli volných adres, je celkový počet právě těch to adres.

Poslední částí programu je příprava vybraných proměnných pro odesílání do nadřazené sítě (ZAU). Do bloku k tomu předurčenému je možné přiřadit proměnné typu word nebo integer. Bylo tedy nutné konvertovat proměnné do správného typu.

Listing 4.7: Aplikační kód konvertace

```
1 REGION (/**/)
2   #Pom1:= ROUND(LTIME_TO_DINT(#HMI_UDT.tLongests)/1000000) ;
3   #ZAU.tLongests := DINT_TO_INT(#Pom1);
4   #Pom2 := ROUND(LTIME_TO_DINT(#HMI_UDT.tCycle_max) / 1000000);
5   #ZAU.tCycle_max := DINT_TO_INT(#Pom2);
6   #ZAU.iPN_frei:= #HMI_UDT.iPN_frei;
7   #ZAU.diCode_mem:=UDINT_TO_INT(#HMI_UDT.diCode_mem);
8 END_REGION
```

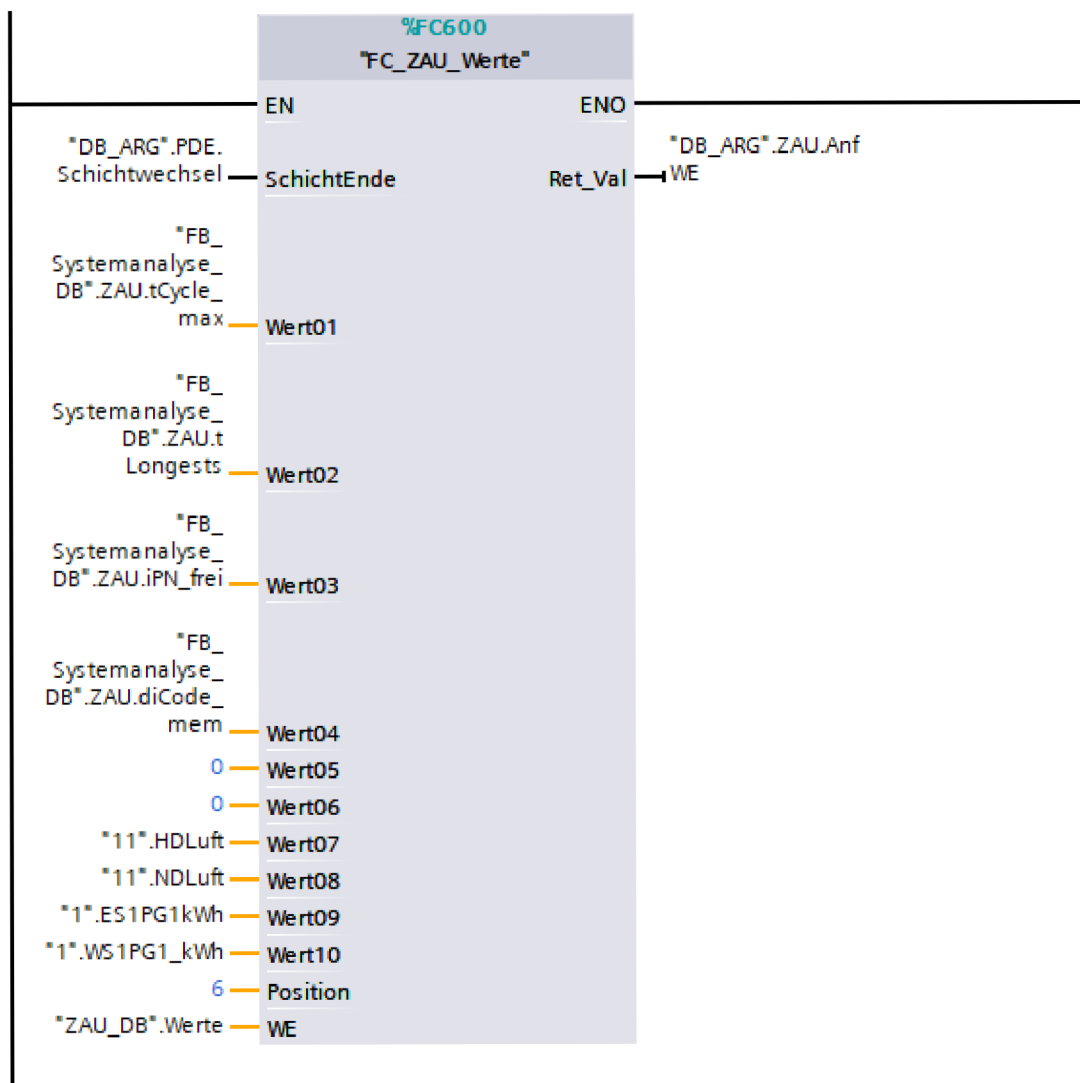
U proměnných tLongests a Tcycle\_max bylo navíc potřeba zkrácení čísla, protože docházelo k přetečení proměnné a tím pádem znehodnocení informace.

## 4.2.5 ZAU

”ZAU” je funkční blok, který obsahuje všechny bloky centrálního monitorování systému. Obsahuje předpřipravené bloky pro odeslání dat do nadřazené sítě. Bloky jsou tři a odlišují se ve velikosti proměnných které lze odeslat. Prvnímu bloku lze přiřadit proměnné typu velikosti bool, druhému word a třetímu double word. Lze přiřazovat proměnné velikostně identické, např. u wordu lze odeslat integer a u double wordu např. double integer.

V postupu práce je jako další krok přiřadit data bloku v požadovaném formátu. Příprava dat byla zmíněna v minulé kapitole. Datům bylo nutné přiřadit pozice, které byly odsouhlaseny oddělením IT.

Byly vybrány nejdůležitější parametry k odeslání: maximální (limitní) doba cyklu, nejdelší doba cyklu, volné ip adresy a obsazenost paměti kódu PLC (code memory).



Obrázek 4.7: Přiřazení proměnných v ZAU

## 4.3 Práce v základním projektu WinCC

### 4.3.1 Vytvoření faceplate

Faceplate je vizuální prvek, který slouží k zobrazování a ovládání určitých zařízení nebo procesů v uživatelském rozhraní HMI. Jedná se o předdefinovaný grafický prvek obsahující různé ovládací prvky, informace a grafiku, které umožňují uživatelům snadnou interakci s monitorovanými objekty. Před vytvořením faceplatu do nového modulu je nutné provést následující specifikace:

- Faceplate jméno a případně i název HMI-UDT
- Jméno bloku

Číslovací systém pro verze bloku je následující: Verze x.y.z

- x: hlavní verze (je inkrementován standardizátorem v případě, že dojde k úplné změně verze v důsledku rozšíření funkce)
- y: podverze (je inkrementován standardizátorem v případě změn, které byly provedeny v rámci opravy chyby)
- z: index změny (je inkrementován v případě změn kódu na stránkách)

#### Nastavení HMI-UDT

Pro faceplate je vytvořen datový typ "HMI-UDT". Struktura odpovídá vizualizaci rozhraní deklarovanému v modulu FB ve statické oblasti DB. Viz. obr. 4.6

#### Povolené barvy

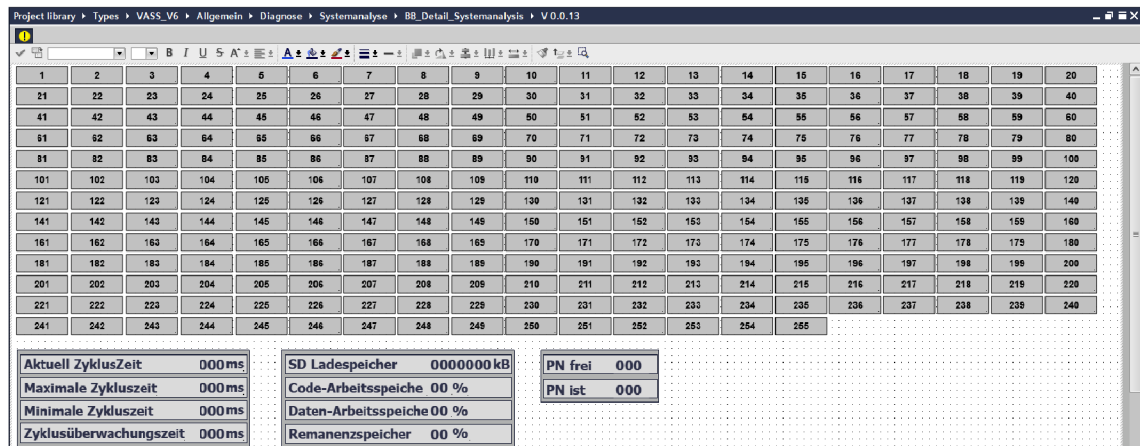
Tabulka povolených barev ve WinCC slouží k definování barevných možností, které mohou být použity pro vizualizaci dat v uživatelském rozhraní. Tento nástroj umožňuje uživatelům specifikovat a omezit barevné varianty, které mohou být aplikovány na různé prvky v jejich projektu, což pomáhá zajistit konzistenci a estetiku v celém systému vizualizace.

Barva	Hodnota R	Hodnota G	Hodnota B
Červená	255	0	0
Žlutá	255	255	0
Světle žlutá	255	255	153
Modrá	0	0	255
Světle modrá	60	140	255
Zelená	0	255	0
Světle šedá	192	192	192
Tmavě šedá	128	128	128
Indigově modrá	24	28	49

Tabulka 4.2: Povolené barvy

## Vložení objektů na obrazovku

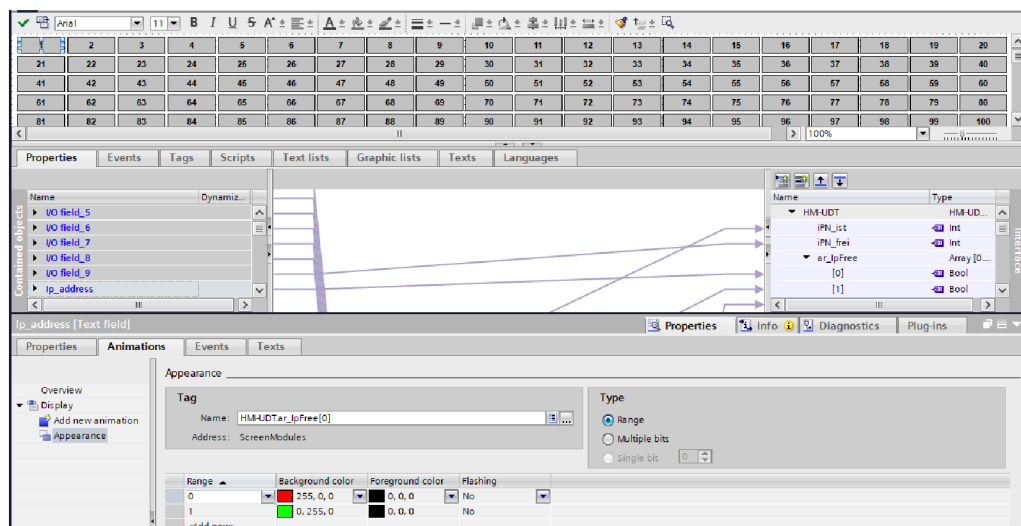
Obrazovka obsahuje bloky, které vizuálně identifikují obsazené IP adresy a textová pole zobrazující počet obsazených a volných ip adres, hodnoty vytížení jednotky PLC a její paměti. Tento přehledný layout umožňuje uživatelům okamžitě identifikovat obsazené adresy v síti a monitorovat vytížení PLC pro efektivní správu a diagnostiku systému.



Obrázek 4.8: Vytvoření obrazovky s objekty

## Přirazení animací

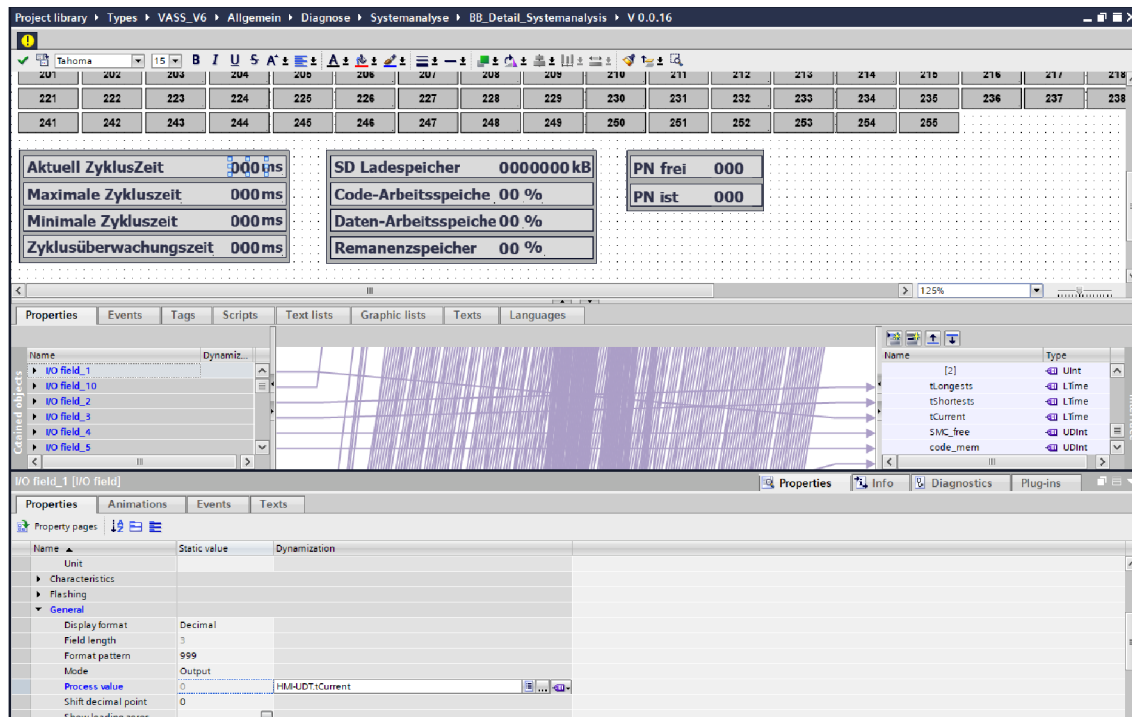
V záložce animations přiřadíme proměnným objektům vlastnosti. V našem případě se mění barva pozadí textového pole, která je závislá na hodnotě bitu pole "ar\_IpFree". Červené pole značí obsazenou ip adresu a zelená volnou.



Obrázek 4.9: Přirazení animací

## Přiřazení proměnných

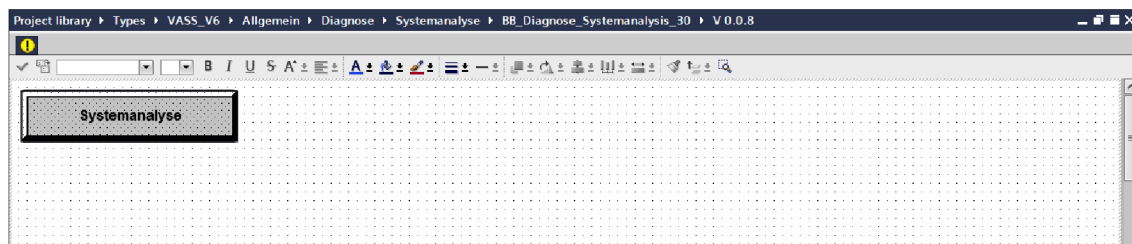
Pro přiřazení procesních hodnot k vstupně-výstupním polím se využívá speciálně vytvořený datový typ nazvaný "HMI-UDT". Tento datový typ umožňuje strukturované a efektivní zpracování procesních dat v uživatelském rozhraní.



Obrázek 4.10: Přiřazení proměnných

### 4.3.2 Vytvoření tlačítka

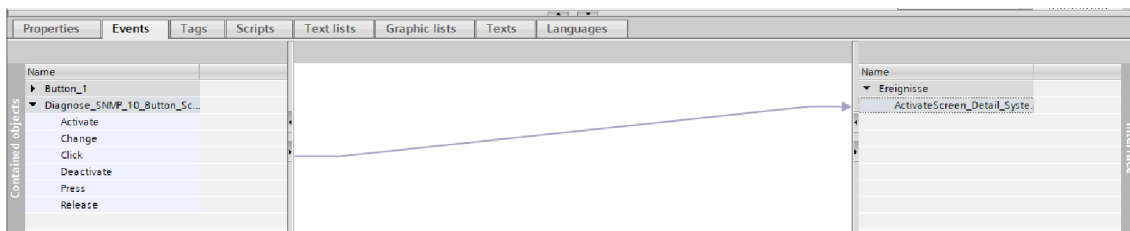
Tlačítko slouží pro otevření okna s obrazovkou, která byla vytvořena v předchozím kroku.



Obrázek 4.11: Vytvoření tlačítka

## Přiřazení obrazovky tlačítku

V dalším kroku je třeba přiřadit event pro otevření obrazovky. Tlačítku přiřadíme event typu "click", tedy provedení akce po kliknutí na prvek.

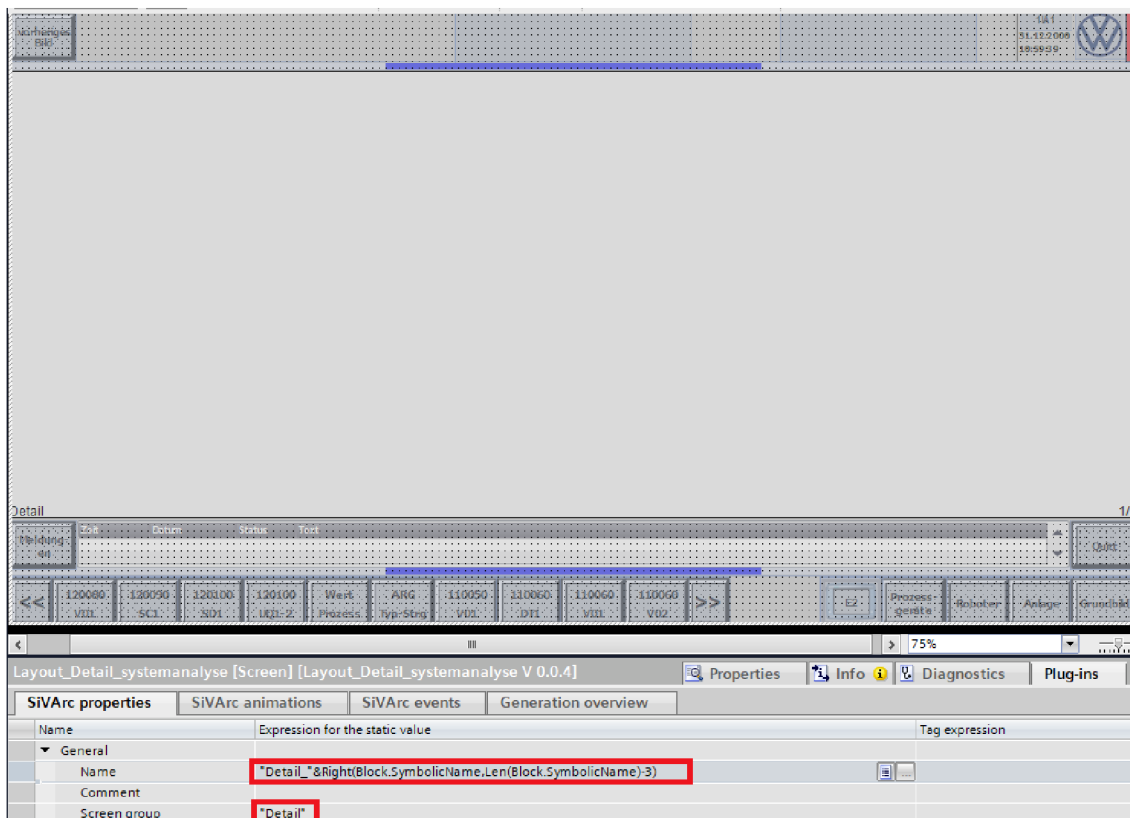


Obrázek 4.12: Přiřazení obrazovky tlačítku

## 4.3.3 Layouty

### Vytvoření layoutu pro obrazovku

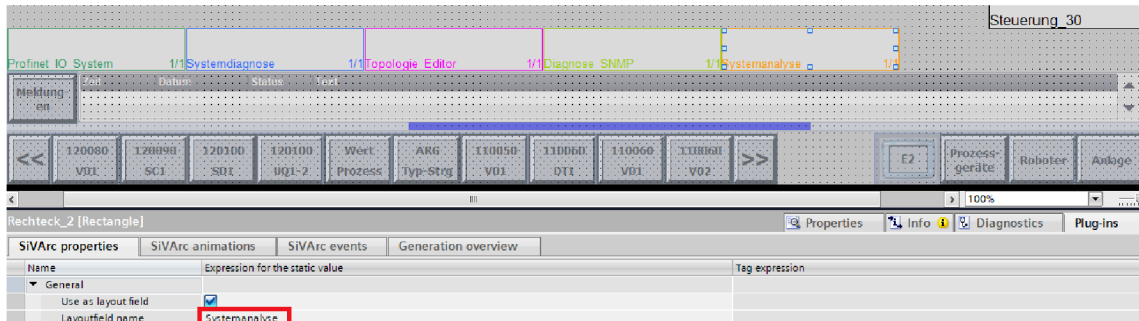
Layout slouží jako podklad pro obrazovku. Zde je třeba nastavit název, který bude přiřazen vygenerované obrazovce a dále skupinu obrazovky, respektive segment pro vložení obrazovky.



Obrázek 4.13: Layout 1

## Upravení layoutu pro tlačítko

Tlačítko je nutné vložit na některou z obrazovek. Vybrána byla nejvhodnější, tedy obrazovku "Diagnose". Na této obrazovce upravíme layout a přidáme místo pro naše tlačítko.

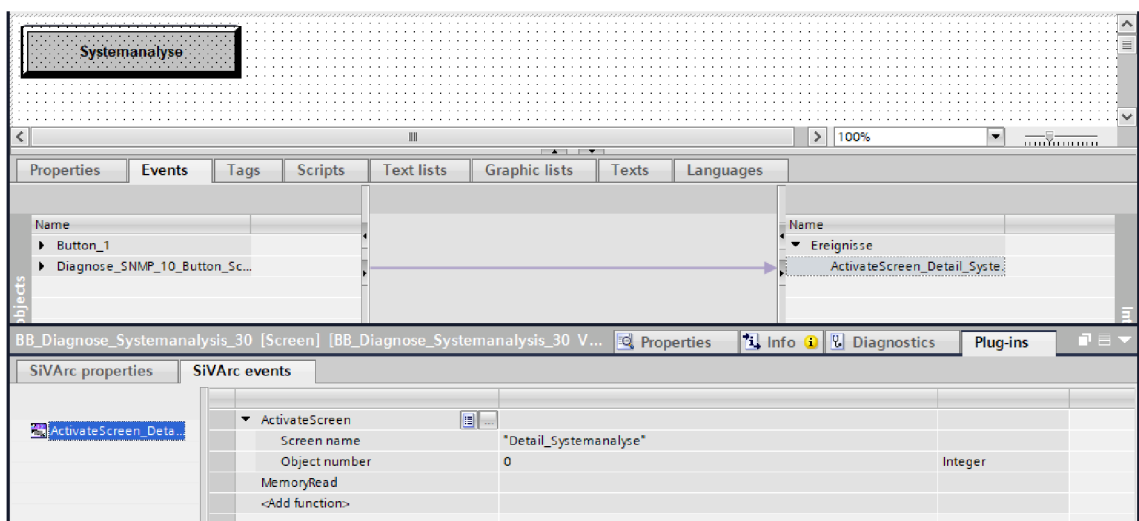


Obrázek 4.14: Layout 2

### 4.3.4 Přřazení makra

V kapitole 4.1 byla zmíněna nemožnost dosažení na požadovaná data přímo v PLC, proto bylo vytvořeno VB makro, které stahuje data z web serveru, filtruje požadovaná data a ty uloží pomocí tagů do proměnných v databloku FB Systemanalyse. Makro se spouští na určitý podmět, je ho tedy potřeba vybrat. Jako spoušť je nejvhodnější akce zmáčknutí tlačítka pro otevření obrazovky.

Toto pravidlo se přidá ve faceplatu tlačítka v záložce plug-ins, SiVArc events a přidáním funkce ("<Add function>") vyjede roletka s možnými akcemi, kde zvolíme makro "MemoryRead".



Obrázek 4.15: Přřazení makra



### 4.3.5 Pravidla generování

V editoru "Screen rules" se definují pravidla obrazovky, podle kterých jsou SiVArc faceplaty generovány v obrazovkách pro různé technologie. Pravidlo se skládá z následujících bodů:

- **Název:** Jedinečný název pravidla obrazovky
- **Programový blok:** Modul FB, který je v libovolném čase uživatelském programu.
- **Objekt na obrazovce:** Generování předlohy nebo faceplatu. Předloha musí být uložena v knihovně projektu.
- **Obrazovka (layout):** Generování šablony obrazovky, na které se generuje faceplate. Šablona generování musí být uložena v knihovně projektu.
- **Pole polohy:** Polohovací pole je součástí polohovacího schématu obrazovky. Políčko pro polohování slouží k zadání polohy generované faceplatu.
- **Podmínky (volitelné):** Jedná se o SiVArc výraz, který se vyhodnotí při zpracování pravidla dané obrazovky. Pokud není zadána žádná podmínka, pravidlo generování obrazovky se vždy provede. Podmínka platí souhrnně pro skupinu pravidel. Podmínka může být upřesněna pro jednotlivá pravidla ve skupině pravidel obrazovek.
- **Komentář (volitelné):** Individuální komentář pro pravidlo obrazovky

#### Vytvoření nového pravidla

Nyní jsou všechny potřebné podklady k dispozici pro generování hotového výstupu a zbývá ještě přidat pravidla pro generování.

Postup:

- Pravidlo obrazovky se vytvoří dvojklikem na "<Create new rule>"
- Přiřazení názvu pravidla. Název je dán dle syntaxe "PLC\_FB\_název FB\_název HMI faceplate layoutu\_číslo stránky". Číslo stránky u jedné obrazovky se nepíše.
- Výběr, pro které PLC se má pravidlo generování použít
- Zvolení programového bloku (modul FB), pro který se vygeneruje obrazovka.
- V části "Screen object" se vybírá šablona generování zobrazovacího a ovládacího prvku.
- V sloupci "Master copy" se vybírá šablona generování obrazovky, ve které se objekt generuje.

- Pokud pro šablonu generování existuje schéma umístění, vyberte oblast umístění v části "Layout field". V případě kdy oblast umístění není zadána, bude vygenerovaný objekt faceplate umístěn na obrazovce podle polohovacího schématu SiVArc.
- V části "HMI device" se vybírá zařízení HMI, pro které má pravidlo obrazovky platit. Pomocí ikon na panelu nástrojů lze zobrazit nebo skrýt typy HMI zařízení. Pokud není zvoleno žádné HMI zařízení, pravidlo obrazovky se použije pro všechna zařízení HMI, která jsou připojena k vybrané řídicí jednotce.

Pokud má být pravidlo provedeno pouze pro objekty nebo programové bloky (FB moduly), které splňují určitou podmínku, naprogramujte příslušný výraz v části "Condition" pomocí skriptování SiVArc.

### Vytvoření pravidla pro diagnostický nástroj

Postup popsán výše. Při vkládání programového bloku je dobré zvolit blok z knihovny (nápoředa nabízí i z projektu). Důvod je jednoduchý a to kopírování pravidel pro různé projekty. Při zvolení bloku z projektu by se u každého projektu museli pravidla předělat, takto ne.

Pravidla byli nutné přidat 2. Jednu pro vytvoření obrazovky s faceplatem a druhou pro vytvoření tlačítka na požadované obrazovce. Viz obr.4.15

	Name	Rule trigger	Screen object	Master copy/Screen type	Layout field
1	▶ VASS_V6_HMI_19Z_Global				
2	▶ VASS_V6_HMI_19Z				
3	▼ Diagnose				
4	▶ PLC_FB_systemanalyse	FB_Systemanalyse	BB_Detail_Systemanalysis	Layout_Detail_systemanalyse	Detail
5	▶ Button_systemanalyse	FB_Systemanalyse	BB_Diagnose_Systemanalysis_30	Layout_030_SPS+PN_19Z	Systemanalyse
6	<create new rule>				
7	<create new rule>				

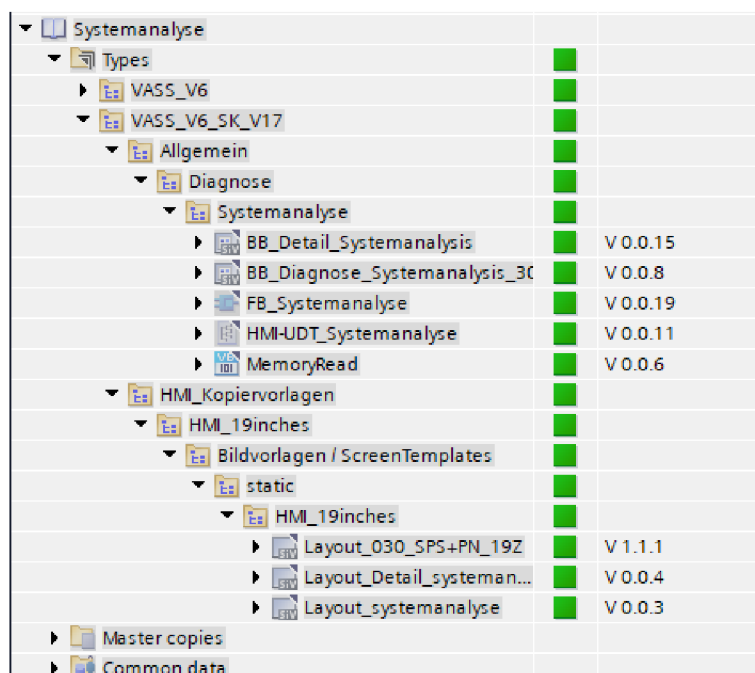
Obrázek 4.16: Pravidla generování

## 4.4 Zprovoznění na výrobní lince

Po dlouhém testování a řešení vzniklých problémů bylo dalším krokem nahrání programu do PLC na výrobní lince. Nahrání muselo předcházet dohodnutí se s výrobou a údržbou provozu. Požadavek byl schválen s termínem nahrání v neděli, kdy není výrobní provoz a v případě poruchy by nenastali žádné škody.

Před cestou na halu bylo nutné vytvořit knihovnu (obr. 4.17), která bude obsahovat všechny potřebné bloky. Bloky je nutno utřídit do složek hierarchicky tak, aby se správně importovali do knihovny projektu na správné místo.

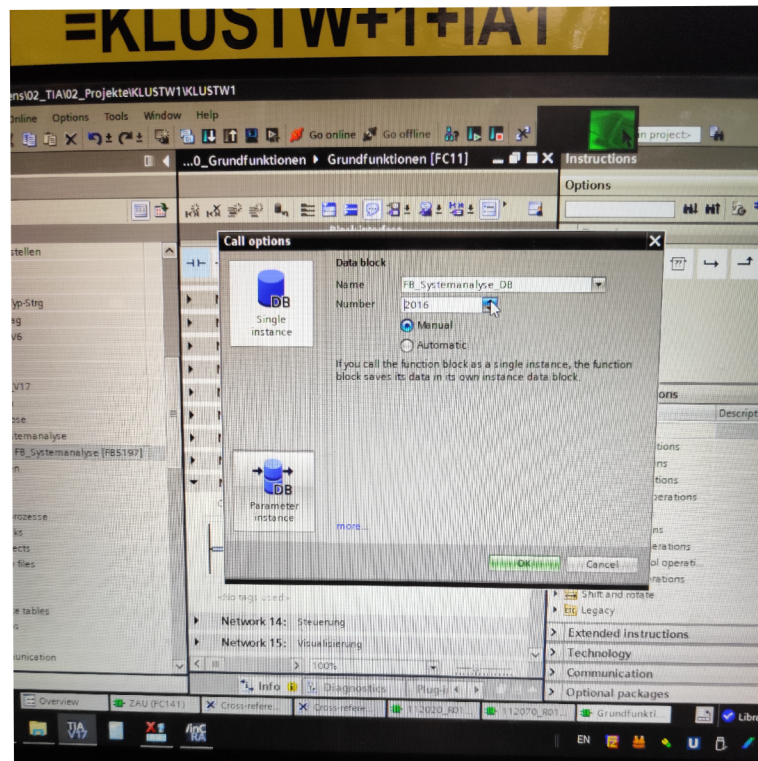
Knihovna obsahuje faceplate obrazovky, faceplate tlačítka, layout pro faceplate obrazovky, upravený layout obrazovky pro vkládání tlačítka, funkční blok, plc data type a VB makro.



Obrázek 4.17: Knihovna systemanalyse

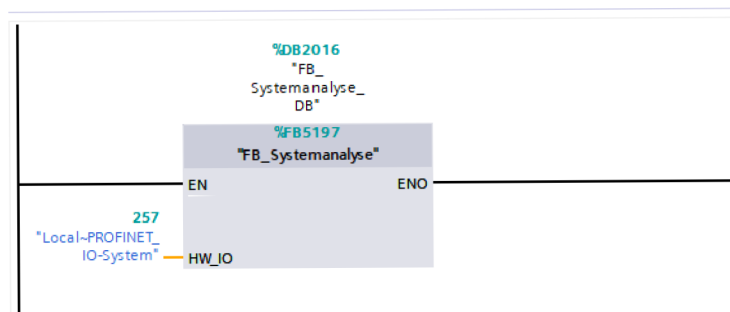
Vytvořenou knihovnu zbývá importovat do zvoleného projektu. Stačí kliknout na knihovnu pravým tlačítkem myši zvolit "update types" a "project". Nyní se importovali všechny věci z vytvořené knihovny do knihovny projektu.

Dalším krokem bylo nahrání programového bloku "FB\_Systemanalyse" do PLC. Blok byl nahrán do FC bloku "Grundfunktionen". Při vkládání bylo nutné dodržet číslování instančního data bloku dle VASS standardu. Zvoleno bylo nejmenší volné číslo začínající na pozici 2000. Viz obr. 4.18



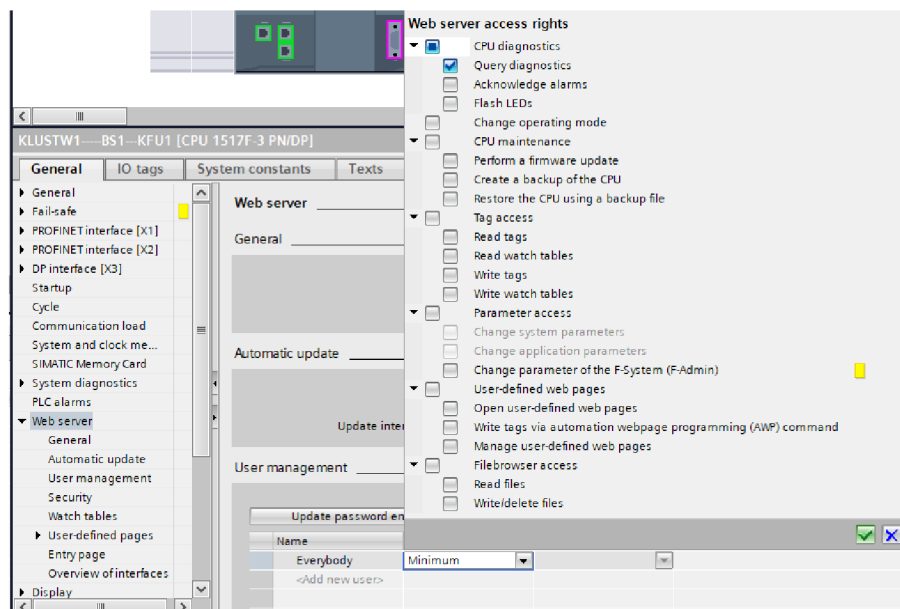
Obrázek 4.18: Nahrání bloku FB<sub>systemanalyse</sub>

K bloku je nutné přiřadit proměnou "HW\_IO". Jedná se o identifikátor profinet zařízení, konkrétně PLC a je důležitá pro správný chod programu.



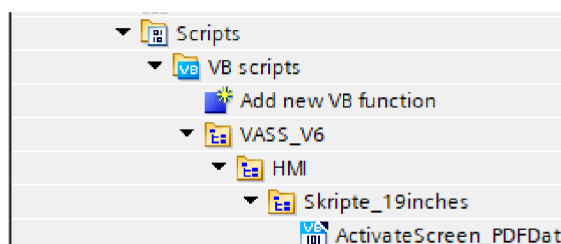
Obrázek 4.19: Nastavení bloku FB<sub>systemanalyse</sub>

Dalším krokem bylo nutné zapnout web server v PLC. Je to důležité pro chod vytvořeného makra, protože požadovaná data stahuje právě z web serveru. Web server se aktivuje ve složce PLC, záložce "Device configuration" a zde se nachází záložka "Web server". Zde se zaškrtně pole "Activate web server on this module". Následně je nutné nastavit přístupové práva. Pro funkčnost makra stačí povolit pouze "Query diagnostics" (viz obr 4.20). Heslo necháme prázdné, k porušení bezpečnosti nedojde, tento blok je pouze ke čtení.



Obrázek 4.20: Web server

Následně se nesmí zapomenout nahrát makro do programu. Na to se v průběhu testování párkrát zapomnělo a musela se spustit znovu generace. Makro stačí přetáhnout do složky pro VB skripty viz obr. 4.24 .



Obrázek 4.21: Cesta k VB skriptům

Dále se vytváří pravidla pro generování. Popis vytváření pravidel byl popsán v kapitole 4.3.4 a samotné pravidla jsou vidět na obrázku 4.15 . Před generací SiVArc je potřeba provést kompilaci PLC programu, protože WinCC porovnává bezpečnostní známku PLC.

Po generaci SiVArC bylo potřeba přiřadit HMI tag k obrazovce, ačkoliv by se měl vytvořit automaticky.



Obrázek 4.22: Vytvoření HMI tagu

Ještě je potřeba provést kompilaci vizualizace a restartovat WinCC.



Obrázek 4.23: Vizualizace na panelu

Na obrázku 4.23 můžete vidět diagnostický nástroj nahraný na panel výrobní linky. Barevné pole označuje barevně obsazenost ip adres. Volné adresy jsou zelené. Červené adresy jsou obsazeny zařízeními a obsahují i adresy, které jsou zakázány používat. Dále lze vidět časy cyklu a vytížení paměti PLC.

## 5 Závěr

Tato diplomová práce vznikla za účelem vytvořit diagnostický nástroj, který by ulehčoval analýzu stávajících zařízení před jeho rozšířením.

Práce byla úspěšně vyřešena a splňuje všechny požadavky, které byly vloženy do zadání.

Postup práce, ačkoli to na první pohled nemusí být zřejmé, nebyl vůbec triviální a obsahoval mnoho překážek, které bylo nutné vyřešit.

Prvním krokem při řešení práce byla analýza diagnostických bloků v systémové knihovně TIA Portalu a bloků vytvořených dle VASS standardu, které jsou využívány v zařízeních koncernu Volkswagen.

Analýza standardních bloků v TIA Portal knihovně neuspokojila všechny požadavky. Užitečné byly pouze bloky, které byly schopny zjistit časy cyklů PLC a vytížení vložené SD karty. Hledání dalších možných diagnostických nástrojů bylo rozšířeno o vyhledávání na internetu a kontaktování i technické podpory Siemens. Výsledek pátrání nepřinesl nic užitečného, byla nalezena pouze zmínka o diagnostickém bloku "RDSYSST", který byl implementován STEP 7 Classic V5 a umožňoval získat mnoho informací o PLC, kde jedním z nich byla vytíženost pracovní paměti. Bohužel tento blok nebyl implementován do TIA Portálu.

Při pátrání v základním VASS projektu jsem byl nasměrován mým kolegou a konzultantem Tomášem Snětivým na potenciálně slibné bloky. Výsledek hledání dovedl k blokům FB Steuerung a FB Diagnose SNMP. Z bloku FB Steuerung jsme schopni vyčíst aktuální počet aktivních a odpojených účastníků sítě profinet. Blok FB Diagnose SNMP obsahuje i pole konkrétních účastníků, respektive obsazenost IP adres. Stále ale nebyly nalezeny všechny potřebné informace, protože ze strany zaměstnavatele byl požadavek na získání informace o vytížení pracovní paměti.

Možností, jak získat tato data, bylo stahování informací z webového serveru. V první testovací fázi byl vytvořen VBS skript, který stahoval webovou stránku web serveru, následně ji zpracoval a uložil požadovaná data do textového souboru, který se načel pomocí ve VB skriptu v IPC panelu. Tento krok byl během fáze testování zdokonalen a stahování dat bylo přímo implementováno do VB skriptu. Jednalo se tedy o vylepšení, u kterého odpadla nutnost běhu skriptu na pozadí IPC panelu a odpadla možnost nestability či blokování antivirovou ochranou.

Práce pokračovala v další fázi projektu vytvořením obrazovek v SiVArc. Jednalo se o učení za pochodu formou pokus omyl, protože SiVArc není moc známá aplikace a nenachází se k ní moc materiálů.

Ve fázi, kdy bylo vše hotovo a otestováno, přišli přípravy na nahrání do výrobní linky. Byla stažena záloha z vybrané linky a zprovozněna na testovací stanici. Vše proběhlo v pořádku, ale při generování SiVArc obrazovek se vyskytla chyba. Chybové hlášení sdělovalo, že chybí v projektu VB skript "MemoryRead" (vytvořené makro), ačkoli byl obsažen v projektu. Po dlouhých hodinách hledání problému a zkoušení různých verzí knihoven a záloh, chybu se nedařilo vyřešit. Chyba byla vyřešena až vytvořením nového layoutu pro faceplate obrazovky. Chyba vznikla nejspíše při vytváření nové knihovny.

Nahrání na výrobní lince již proběhlo bez problémů.

Nejvíce na této práci si cením zobrazení informace o vytížení pracovní paměti, protože tato funkce není programově přístupná.



## Použitá literatura

- [1] *Coding Standards: What Are They and Why Are They Important?* 2023. Dostupné také z: <https://blog.codacy.com/coding-standards>.
- [2] *Data sheet IPC 477E*. -. Dostupné také z: <https://viola.cz/produkt/stazeni-souboru?id=34787%5C&productId=164849%5C&productSlug=siemens-6av7241-3db44-0fa2-simatic-ipc-477e-19-touch-1366-x-768-with-front-usb-4-usb>.
- [3] *HMI systems katalog*. -. Dostupné také z: [https://cache.industry.siemens.com/dl/files/146/109744146/att\\_1145917/v1/simatic-st80-stpc-complete-english-2023.pdf](https://cache.industry.siemens.com/dl/files/146/109744146/att_1145917/v1/simatic-st80-stpc-complete-english-2023.pdf).
- [4] *IPC 477E*. -. Dostupné také z: <https://mall.industry.siemens.com/mall/cs/cz/Catalog/Product/6AV72415DA070DS2>.
- [5] *Principy komunikace a diagnostika sítě Profinet*. 2013. Dostupné také z: [https://www.automata.cz/aton/filerepository/pdf\\_articles/10377.pdf](https://www.automata.cz/aton/filerepository/pdf_articles/10377.pdf).
- [6] *Profinet*. -. Dostupné také z: <https://www.wikiwand.com/en/Profinet>.
- [7] *Profinet – standard pro průmyslový Ethernet v automatizaci*. -. Dostupné také z: [https://www.automata.cz/cz/casopis-clanky/profinet-standard-pro-prumyslovy-ethernet-v-automatizaci-2005\\_04\\_30419\\_715/](https://www.automata.cz/cz/casopis-clanky/profinet-standard-pro-prumyslovy-ethernet-v-automatizaci-2005_04_30419_715/).
- [8] *Řídicí systém Simatic S7-1500*. 2013. Dostupné také z: <https://www.elektroprumysl.cz/automatizace/ridici-system-simatic-s7-1500-je-zamerena-vykonnost-a-efektivitu>.
- [9] *Siemens Simatic HMI*. -. Dostupné také z: <https://www.siemens.com/global/en/products/automation/simatic-hmi.html>.
- [10] *SIMATIC CPU 1517F-3 PN/DP*. [B.r.]. Dostupné také z: <https://mall.industry.siemens.com/mall/cs/cz/Catalog/Product/6ES7517-3FP00-0AB0>.
- [11] *Simatic IPC katalog*. -. Dostupné také z: <https://assets.new.siemens.com/siemens/assets/api/uuid:c2d2a8a7-be2e-433d-a4d6-e87fd6f306cd/dffa-b10299-04-7600-simatic-ipc-16s-brochure-poster-online.pdf>.
- [12] *Simatic panel IPC*. -. Dostupné také z: <https://www.siemens.com/global/en/products/automation/pc-based/simatic-panel-pc.html>.
- [13] *SIMATIC S7-1500*. -. Dostupné také z: <https://www.siemens.com/cz/cs/products/automation/systems/industrial/plc/simatic-s7-1500.html>.

- [14] *SIMATIC S7-1500 signal modules.* -. Dostupné také z: <https://www.siemens.com/global/en/products/automation/systems/industrial/plc/simatic-s7-1500/signal-modules.html>.
- [15] *SIMATIC Visualization Architect.* -. Dostupné také z: [https://cache.industry.siemens.com/dl/files/281/109742281/att\\_899541/v1/TIAPortalSivarcenUS\\_en-US.pdf](https://cache.industry.siemens.com/dl/files/281/109742281/att_899541/v1/TIAPortalSivarcenUS_en-US.pdf).
- [16] *TIA – plně integrovaná automatizace napříč celým podnikem.* [B.r.]. Dostupné také z: <https://www.siemens.com/cz/cs/products/automation/industry-software/automatizacni-software/tia-portal.html>.
- [17] *TIA Portal katalog.* -. Dostupné také z: <https://assets.new.siemens.com/siemens/assets/api/uuid:0c66eeeb-b67d-4c98-aa0d-290f82d5c0d2/7801-09-tia-p-ipdf-en-181029-1.pdf>.
- [18] *VASS V6 Standard Standardization Guideline TIA.* 2022.
- [19] *Visualize future.* -. Dostupné také z: <https://www.siemens.com/global/en/products/automation/simatic-hmi/wincc-unified.html>.