



**FAKULTA
STROJNÍ
ČVUT V PRAZE**

Ústav přístrojové a řídicí techniky

**Návrh a realizace systému pro sledování
efektivnosti zařízení**

**Design and implementation of a system for
efficiency of equipment monitoring**

DIPLOMOVÁ PRÁCE

2020

Bc. Vladimír CHMELÍK

Studijní program: N2301 STROJNÍ INŽENÝRSTVÍ
Studijní obor: 2301T034 Přístrojová a řídicí technika
Vedoucí práce: Ing. Jaroslav Bušek, Ph.D.



ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Chmelík** Jméno: **Vladimír** Osobní číslo: **458482**
 Fakulta/ústav: **Fakulta strojní**
 Zadávací katedra/ústav: **Ústav přístrojové a řídicí techniky**
 Studijní program: **Strojní inženýrství**
 Studijní obor: **Přístrojová a řídicí technika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Návrh a realizace systému pro sledování efektivity zařízení

Název diplomové práce anglicky:

Design and implementation of a system for efficiency of equipment monitoring

Pokyny pro vypracování:

1. Vypracujte rešerše potřebné k řešení systému sledování efektivity vybraného zařízení.
2. Proveďte rozbor zvolené technologie z hlediska sledování efektivity vybraného zařízení.
3. Navrhněte a realizujte jednoduchý systém pro sledování efektivity zařízení.
4. Proveďte zhodnocení dosažených výsledků.

Seznam doporučené literatury:

- [1] SINGH, Ranteshwar, et al. Overall Equipment Effectiveness (OEE) calculation-Automation through hardware & software development. *Procedia Engineering*, 2013, 51: 579-584.
 [2] AHIRE, Chandrajit P.; RELKAR, Anand S. Correlating failure mode effect analysis (FMEA) & overall equipment effectiveness (OEE). *Procedia Engineering*, 2012, 38: 3482-3486.
 [3] DAL, Bulent; TUGWELL, Phil; GREATBANKS, Richard. Overall equipment effectiveness as a measure of operational improvement—a practical analysis. *International Journal of Operations & Production Management*, 2000.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Jaroslav Bušek, Ph.D., U12110.3

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

Datum zadání diplomové práce: **30.04.2020**

Termín odevzdání diplomové práce: **27.08.2020**

Platnost zadání diplomové práce: _____

Ing. Jaroslav Bušek, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

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

III. PŘEVZETÍ ZADÁNÍ

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

10.5.2020
Datum převzetí zadání


Podpis studenta

Abstrakt

Diplomová práce je zaměřena na metody získávání a zpracování dat o provozu výrobního podniku. V teoretické části je rozebrán výběr vhodné SQL pro ukládání dat, dále pak popis komunikace OPC UA a její praktické využití a v poslední části popis statistických parametrů pro hodnocení výkonnosti podniku.

V praktické části je popsán program vytvořený v programu Automation Studio pro sběr a vyhodnocení dat. Dále pak struktura připojení senzorů na I/O ostrůvky a praktické vyhodnocení nasbíraných dat.

Klíčová slova

SQL databáze, PLC, OPC UA, OEE, Automation Studio, SCADA, sběr dat

Abstract

The diploma thesis is focused on methods of obtaining and processing data on the operation of a manufacturing company. The theoretical part discusses the selection of suitable SQL for data storage, then a description of OPC UA communication and its practical use and in the last part a description of statistical parameters for evaluating the performance of the company.

The practical part describes the program created in the Automation Studio program for data collection and evaluation. Furthermore, the structure of the connection of sensors to the I / O islands and the practical evaluation of the collected data.

Key words

SQL database, PLC, OPC UA, OEE, Automation Studio, SCADA, data collection

Prohlášení

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

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

.....
podpis

V Praze dne 20.8.2020

Vladimír Chmelík

Poděkování

Ráb bych poděkoval vedoucímu mé diplomové práce Ing. Jaroslavu Buškovi, Ph.D. za pomoc, rady a vedení v průběhu diplomové práce. Dále bych rád poděkoval firmě Plast Form Service i.m. za možnost provést praktickou část práce s jejich podporou a technickým vybavením. Na závěr bych rád poděkoval mojí rodině za podporu během studia.

Obsah

ÚVOD	1
1 ZÁKLADNÍ STRUKTURA SYSTÉMU	2
2 UKLÁDÁNÍ DAT	4
2.1 MICROSOFT SQL SERVER.....	5
2.1.1 Licence.....	5
2.1.2 Výhody a nevýhody	6
2.2 MYSQL	6
2.2.1 Licence.....	7
2.2.2 Výhody a nevýhody	7
2.3 MARIADB.....	7
2.3.1 Licence.....	7
2.3.2 Výhody a nevýhody	8
2.4 KONEČNÝ VÝBĚR DATABÁZE.....	8
3 KOMUNIKACE S PLC	9
3.1 ZÁKLADNÍ POPIS.....	9
3.2 POPIS KOMUNIKACE A ZABEZPEČENÍ.....	10
3.3 VYUŽITÍ V PRŮMYSLU	10
4 UKAZATELE ÚSPĚŠNOSTI A EFEKTIVITY VÝROBNÍHO PODNIKU	12
4.1 KLÍČOVÉ UKAZATELE VÝSLEDKŮ (KRI)	12
4.2 UKAZATELE VÝKONOSTI (PI)	12
4.3 KLÍČOVÉ UKAZATELE VÝKONOSTI (KPI)	13
4.3.1 Celková efektivita zařízení (OEE).....	13
4.3.2 Výtěžnost prvního průchodu (FPY)	15
4.4 APLIKACE KOEFICIENTŮ VE FIRMĚ PLAST FORM SERVICE IM S.R.O.....	15
5 POPIS ŘEŠENÍ PRO SNÍMÁNÍ PROVOZU	16
5.1 STRUKTURA ŘEŠENÍ SBĚRU DAT	16
5.1.1 Ukládání dat.....	18
5.1.2 Vizualizace.....	20
5.1.3 Hlavní funkce programu.....	20
5.2 POPIS PROGRAMU.....	21
5.2.1 Cyklus stroje	21
5.2.2 Měření OEE	22
5.2.3 Ukládání do databáze	23
5.2.4 Zasílání e-mailového upozornění a alarmy	28
5.3 POPIS JEDNOTLIVÝCH ČÁSTÍ VIZUALIZACE.....	30
5.3.1 Přehled výroby	30
5.3.2 Parametry výroby stroje.....	32
5.3.3 Historie cyklů.....	34
5.3.4 Výběr dílu	34
5.3.5 Údržba.....	35
5.4 UŽIVATELSKÉ PŘÍSTUPY.....	35
5.4.1 Přístup bez přihlášení	35
5.4.2 Kontrola kvality	36
5.4.3 Mistr směny.....	36
5.4.4 Hlavní technolog	36

5.4.5	Administrátor	36
5.4.6	Údržba.....	36
6	PRÁCE A VYUŽITÍ SQL DATABÁZE	38
6.1	STRUKTURA DATABÁZE	38
7	APLIKACE KOMUNIKACE OPC UA	40
8	ANALÝZA A VYUŽITÍ ZÍSKANÝCH DAT	41
9	PLÁNY NA ROZŠÍŘENÍ.....	43
10	ZÁVĚR	45
11	SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ	46
12	SEZNAM OBRÁZKŮ	48
13	SEZNAM TABULEK.....	49
14	SEZNAM GRAFŮ	50

Úvod

S nástupem nových technologií v oblasti IT a automatizace začíná hrát velkou roli kontrola výrobního procesu. V moderních podnicích je kladen důraz především na efektivitu a výkonost. Ovšem aby bylo možné výrobní proces zefektivnit, je zapotřebí vidět rozdíl mezi současným stavem a stavem po zavedení různých opatření a vylepšení. Za tímto účelem byla vytvořena celá řada koeficientů, který mají za úkol změřit a porovnat jednotlivé výrobní procesy. K jejich výpočtu jsou ovšem zapotřebí data z výroby. Získávání dat, jejich strukturování a následné vyhodnocování postupem času získává na důležitosti.

Cílem diplomové práce bylo vytvoření jednoduché metody, jak získávat a vyhodnocovat data z výroby. Diplomová práce byla vytvářena ve spolupráci s firmou Plast Form Service i.m. s.r.o. V této firmě byly následně aplikovány a ozkoušeny všechny praktické výstupy, které tato práce zahrnuje. Diplomová práce se opírá o výsledky předmětu Projekt 2, kde byla vypracována ekonomická rozvaha. Ta měla za cíl porovnat různé možnosti zavedení systému pro sběr dat ve firmě. Vedení firmy se rozhodlo pro vytvoření vlastního programu.

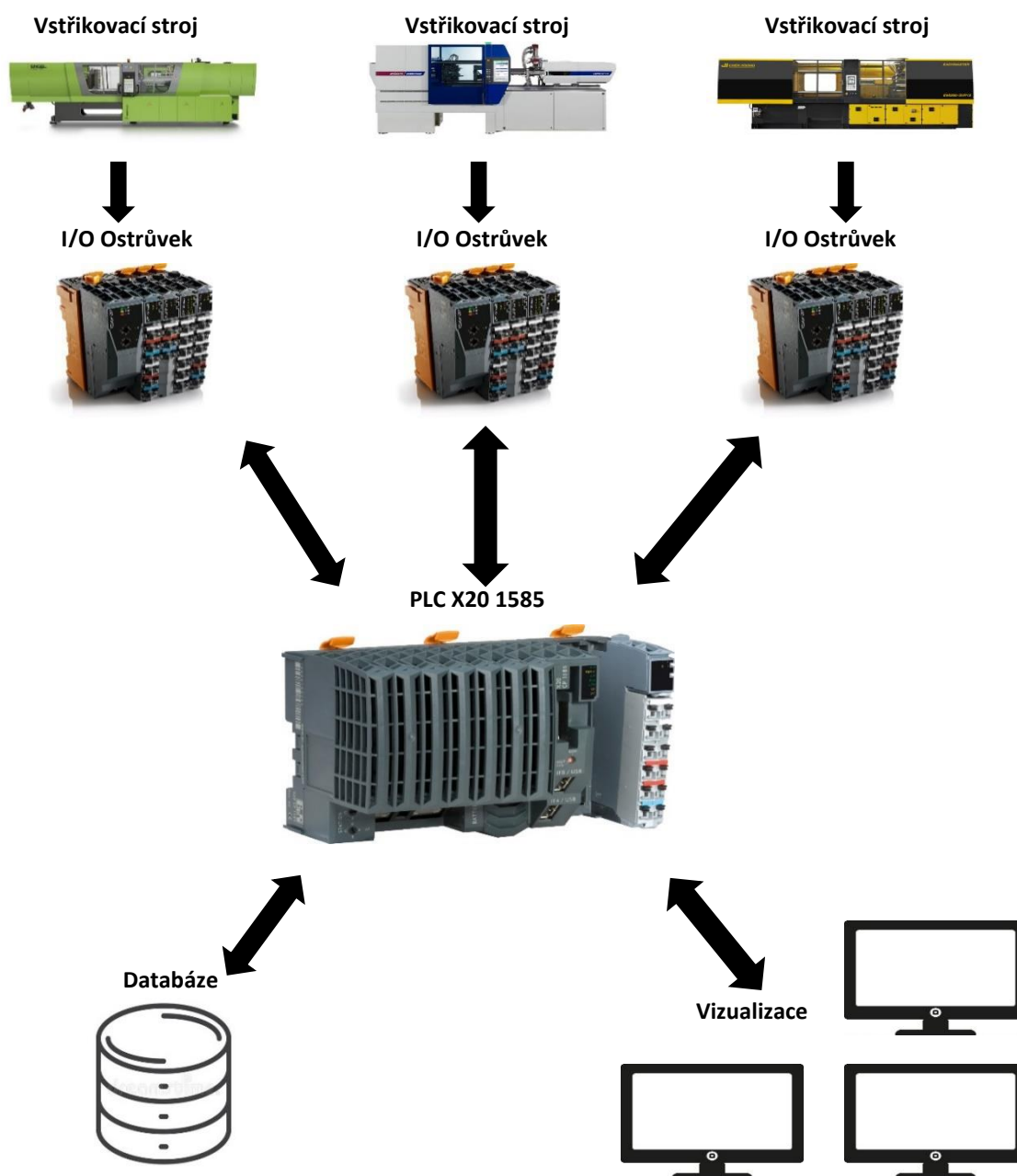
Hlavními cíli diplomové práce bylo vytvoření systému pro sběr dat jak po stránce hardwarové, tak softwarové. Nalezení metody vyhodnocení efektivity výroby a usnadnění práce s kontrolou celého procesu pomocí vizualizace a alarmových kontrol.

Firma Plast Form Service i.m. s.r.o. disponovala na začátku projektu 23 stroji, které by měly být v budoucnu napojeny na systém pro sběr dat. Diplomová práce se zaměřovala na připojení prvních dvou strojů a ozkoušení celého konceptu řešení sběru dat. K dispozici firma poskytla jeden server pro běh databáze. Tento server byl pracovníkem firmy v oboru IT připraven z pohledu zálohování a kontroly běhu serveru. Dále bylo uvolněno několik PC pro vizualizaci. V rámci předchozí práce na projektech byla vytvořena první datová síť pro komunikaci s I/O ostrůvky a PLC, které bylo připojeno do firemní sítě.

Systém v první fázi, na kterou se zaměřuje tato diplomová práce, měl plnit kontrolní roli. Díky tomu bylo možné na vizualizaci pracovat bez toho, aniž by výpadek či práce na programu závisela na výrobě. V budoucnu je ovšem počítáno s větším propojením s celým provozem. V rámci diplomové práce bude stručně popsán způsob propojení systému s pracovníky firmy.

1 Základní struktura systému

Celou strukturu systému je možné rozdělit na 4 části. PLC, vizualizaci, databázi a připojení na stroje. Mezi každou částí bylo zapotřebí vytvořit komunikační kanál. V rámci Projektu 2 byla vypracována finanční rozvaha dostupných možností na trhu. Mezi variantami bylo jak objednání celého systému od dodavatele, tak cenový odhad varianty provedení systému vytvořením v rámci firmy. Vedení firmy se rozhodlo vytvořit si systém vlastní a díky finančním a uživatelským výhodám využít produkty firmy B+R automatizace, spol. s r.o. [1]. Díky těmto produktům byla pokryta velká část zamýšlené struktury systému. Po softwarové stránce bylo využito programu Automation studio, kde bylo možné vytvořit vizualizaci a hlavní program pro PLC. Pomocí I/O ostrůvků byly propojeny senzory jednotlivých strojů s PLC, obojí zakoupené od firmy B+R automatizace, spol. s r.o.



Obrázek 1.1 Struktura systému pro sběr dat

Pro databázi bylo zvoleno oddělené řešení, jelikož program Automation studio neměl implementovanou SQL databázi. K napojení signálů ze strojů se uvažovalo o dvou variantách. V první o osazení stroje vlastními senzory pro určení cyklu a druhou variantou bylo využití stávajících senzorů stroje a vyvedení druhého signálu na vstup našeho I/O ostrůvku. Jelikož bylo možné signály stroje dohledat a využít, byla zvolena druhá varianta [2].

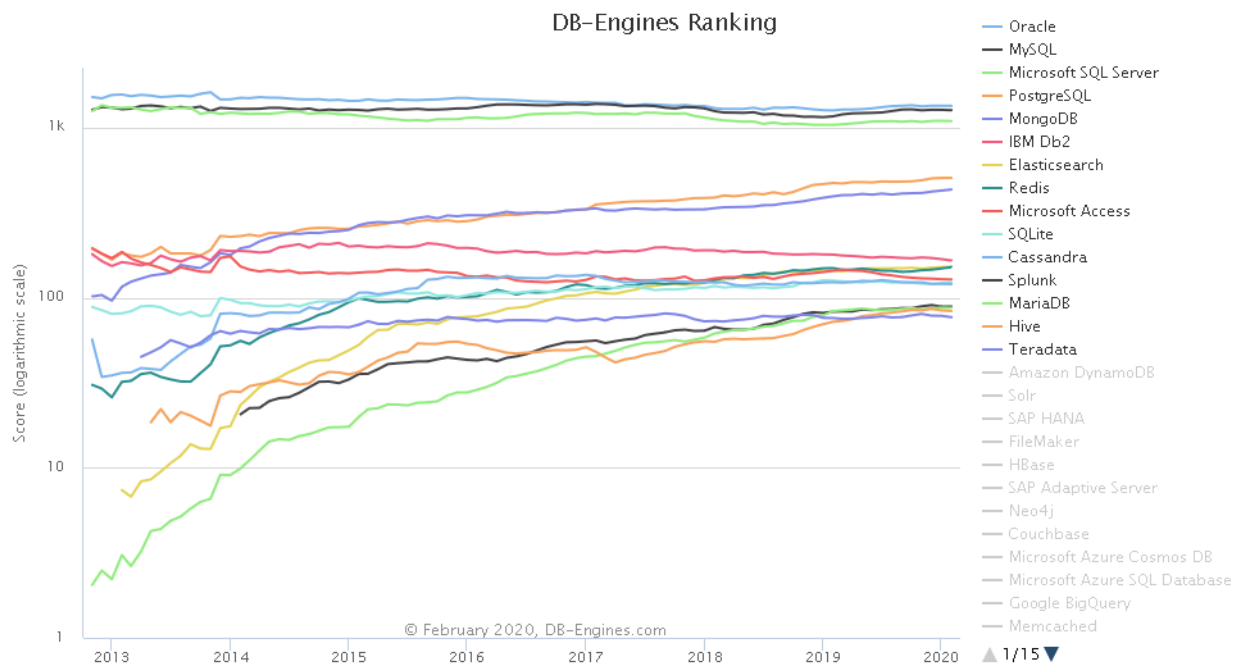
2 Ukládání dat

Pro ukládání dat byla zvolena SQL databáze. SQL – Structured Query Language (Strukturovaný dotazovací jazyk). Jedná se o nástroj pro manipulaci, správu a organizaci dat. Databáze využívající tento programovací jazyk se využívají především k vytváření strukturovaného zápisu dat v přehledných tabulkách. Tento jazyk je charakterizovaný především využíváním tzn. dotazů, pomocí kterých můžeme do databáze ukládat data a následně je i získávat podle zadaných kritérií. SQL jazyk nepatří mezi plnohodnotné programovací jazyky, jelikož např. neobsahuje řídicí programové konstrukce, které jsou běžné v ostatních plnohodnotných programovacích jazycích typu C, ST, Java atd [3].

V dnešní době je již databází na trhu celá řada. V grafu 2.1 je možné vidět srovnání 15 nejlépe hodnocených databází dnešní doby a jejich postupný nárůst popularity. Bodování jednotlivých databází v grafu bylo vytvořeno pomocí několika faktorů [4].

1. Počet nalezených výsledků největších světových vyhledávačů (např. Google)
2. Obecný zájem o danou databázi měřenou pomocí četnosti dotazů na vyhledávání
3. Frekvenci odborných diskusí v rámci internetu
4. Počet pracovních nabídek, ve které je databáze uvedena
5. Počet životopisů, ve kterých je daná databáze uvedena
6. Počet zmínek na sociálních sítích

Pro účely diplomové práce se bylo potřeba omezit na tři základní databáze. Důvodem tomu byly nástroje, které umožňovali rychlejší a snadnější práci při programování. Program Automation Studio, který byl v rámci předchozích Projektů vybrán pro vytváření programu, sice měl možnost pracovat s větším množstvím databází, než zde budou uvedeny, ovšem za cenu složitější práce s vytvářenou aplikací. V rámci programu Automation Studio bylo možné od firmy B+R zakoupit takzvané Mapp komponenty. Jednalo se o funkční bloky, či ucelené programy, které uživatel mohl využívat a konfigurovat dle své potřeby. Jednou z těchto Mapp komponent byla MpData, která umožňovala jednoduchou práci s databází. Tato Mapp komponenta podporovala 3 databáze. Microsoft SQL server, MySQL a MariaDB.



Graf 2.1 Oblíbenost 15 nejlépe hodnocených databázových systémů a jejich trend v posledních 7 letech [5]

2.1 Microsoft SQL server

Microsoft SQL server byl třetí nejlépe hodnocenou databází světa. Bylo možné jej nalézt v mnoha velkých aplikacích. Databázový systém byl využíván v drtivé většině případů s technologiemi firmy Microsoft. Příkladem průmyslového využití mohl být SCADA systém WinCC, který k ukládání výrobních dat využíval právě Microsoft SQL server [6]. V nejnovější verzi 2019 bylo možné nalézt několik nových funkcí. Do této verze byla přidána možnost správy dat i z jiných typů databází. Dále zde byly aplikovány nové metody správy dat založených na umělé inteligenci pro zpracování velkých objemů dat [7].

2.1.1 Licence

Firma Microsoft nabízela několik edicí Microsoft serverů, které se lišili jak cenou, tak svými možnostmi. Pro účely diplomové práce bylo uvažováno o edicích Express, Standart a Enterprise.

Edice Express byla bezplatná verze Microsoft SQL serveru. Jednalo se o řešení pro malé aplikace. Poskytovala kompletní databázové nástroje pro práci s daty přes jazyk T-SQL. Jelikož se jednalo o neplacenou verzi, měla svá omezení. Databáze mohla mít pouze omezenou velikost. Edice Express byla omezena na 10 GB dat. Pro práci s daty bylo využívání procesoru omezeno na 1 GB operační paměti. Při instalaci byla omezena na jeden procesor, nebo čtyři jádra. V souladu s licenčním

omezením se tato aplikace nesměla využívat v síťových aplikacích [8] a taktéž bylo omezeno využívání některých funkcí, například fulltextového vyhledávání [9].

Edice Standart byla již placenou verzí. Bylo možné zakoupit buď doživotní licenci v ceně 3 586 dolarů, nebo roční licenci za 1 418 dolarů [10]. Jednalo se o plnohodnotné řešení. Kompletní databázové nástroje, plnohodnotná podpora pro datové integrace (SSIS), analytics services (SSAS) a management dat. Tato edice umožňovala ukládat 524 PB dat a pracovat s 128 MB operační paměti. Instalace byla omezena na 4 procesory nebo 24 jader. Obsahovala balík mnoha dalších funkcí pro správu a vyhledávání dat [9].

Edice Enterprise byla nejvyšší možnou verzí, kterou společnost Microsoft nabízela. Cena za doživotní licenci byla 13 586 dolarů, nebo roční předplatné za 5 434 dolarů ročně. [10] Tato verze se lišila oproti edici Standart hlavně množstvím funkcí. Rozšířené možnosti disaster plánů, availability, security, paralelní zpracování dotazů atp. Tato edice neměla omezený počet procesorů, jader, ani omezení operační paměti [9].

2.1.2 Výhody a nevýhody

Mezi hlavní výhody patřila možnost vysokého stupně integrity s dalšími programy od firmy Microsoft. [11] To přinášelo možnost jednodušší práce s tímto systémem. Dále zde byla možnost s využitím služby Azure vytvořit celou databázi v cloudu [9].

Mezi nevýhody patřila jednak cena u placených verzí, nebo omezení velikosti edice Express [9]. Dále zde byla otázka bezpečnosti dat. Jelikož byla možnost, že by se v budoucnosti mohla databáze využívat i k ukládání citlivých údajů, bylo zapotřebí zohlednit i toto hledisko. Zde právě díky integraci s dalšími prvky byla možnost, že v rámci konkrétní firmy mohou být tato data ohrožena. Tato nevýhoda se vztahovala především na verzi Express, která měla omezené možnosti ochrany dat [11].

2.2 MySQL

MySQL patřila mezi nejoblíbenější SQL databáze dnešní doby. V kombinaci s Linuxem, Apachem programovacím jazykem PHP tvořila řadu svobodného softwaru LAMP. Byla navržena a optimalizována pro webové aplikace. MySQL používaly velké internetové firmy jako například Facebook, Twitter nebo Youtube. Ve svých počátcích byla databáze tvořena jako výkonná aplikace na úkor funkcí. To se ovšem začalo v poslední době měnit a verze 5.0 přinesla řadu nových funkcí [12].

Nejnovější verze nesla označení 8.0. U této verze byl vylepšen výkon, InnoDB byla znatelně rychlejší jak při zápisu, tak při čtení. Zlepšena byla také bezpečnost výchozí autentizace a přibyla vylepšení z OpenSSL. Nově byla také rozšířena syntaxe JSON [13]. Mezi další zajímavé funkce patřila rozšířená možnost výpočtů v rámci databáze [14].

V dnešní době patřila MySQL firmě Oracle Comparison, která roku 2009 koupila firmu Sun, která databázi do této doby vyvíjela. To přineslo jistý otazník, jak bude vypadat budoucnost této databáze [15].

2.2.1 Licence

MySQL byla možná k dostání ve dvou základních variantách. Bezplatná open source verze MySQL Community a placené verze MySQL Standart a Enterprise.

Verze Community byla po výkonové stránce zcela bez omezení. Licenční práva také umožňovala využití v rámci sítě. Pokud tedy nebude tendence prodat výsledný program třetí straně, nebyl z hlediska licence žádný problém [16]. Výhody v podobě placené verze byly hlavně v oblasti jednotlivých vylepšujících funkcí. Mezi nejvýznamnější vylepšení placených verzí byla hlavně oblast technické podpory a snazšího upgradu softwaru na novější verze. Verze Standard byla prodávána za 2 000 dolarů a Enterprise za 5 000 dolarů [17].

2.2.2 Výhody a nevýhody

Systém byl založen na open source licenci, jehož používání není v základní verzi nijak omezeno. Díky otevřenému kódu ho bylo možné provozovat téměř na každém operačním systému. Jednalo se o prověřený systém, který byl vždy považován za robustní a jednoduché řešení. Díky tomu neměla databáze velké nároky na výpočetní výkon [11].

Mezi nevýhody patřila především otázka budoucnosti. Firma Oracle Comparison postupně omezovala open sourcovou licenci [11]. Některé nevýhody pramenily z výhod. Jelikož se jednalo o jednoduchou koncepci databáze, postrádala některé rozšířené funkce [18].

2.3 MariaDB

MariaDB vznikla jako reakce na odkoupení MySQL firmou Oracle Comparison. Jednalo se takřka o totožný produkt jako MySQL. Mezi vývojáře MariaDB patřili i původní programátoři MySQL. Hlavní původní myšlenkou bylo vytvořit takovou databázi, která by byla určitou jistotou pro uživatele MySQL. Přejít na MariaDB z MySQL byl možný velmi snadno. MariaDB měla totiž stejnou strukturu, a tudíž bylo možné pouze přehrát data do nové verze [19]. Na databázi přešla roku 2013 například databáze Wikipedie [20]. Vzrůstající popularitu je možné pozorovat i na grafu 2.1

2.3.1 Licence

MariaDB bylo možné získat v plné licenci zcela zdarma. Oproti MySQL neměla dvojí licencování a bylo možné ji využít v plném rozsahu zcela zdarma bez jakéhokoliv omezení [21].

2.3.2 Výhody a nevýhody

Výhody a nevýhody ve funkčnosti této databáze byly podobné jako u MySQL. Jednalo se o jednoduchou a robustní databázi. Výhodou bylo možnost získání plné verze zcela zdarma. Další výhodou bylo, že v případě zvolení databáze MySQL a zjištění případných problémů, bylo možno jednoduše přejít na databázi MariaDB [19].

Nevýhodou oproti MySQL byla absence případné technické podpory [19].

2.4 Konečný výběr databáze

Konečný výběr databáze se odvíjel především od požadavků firmy Plast Form Service i.m. s.r.o. Jelikož se jednalo o malou firmu, nebylo potřeba kupovat žádnou licenční verzi, jelikož uvažovaná databáze nepředpokládala žádné složité struktury, a tudíž nebylo potřeba se příliš ohlížet na specializované funkce. Hlavním kritériem tedy byla cena. Další kritérium byla rychlost systému a objem uložených dat. Systém musel být dimenzován na aktuální požadavky firmy. Firma vlastnila 23 strojů. Průměrný výrobní cyklus stroje se pohyboval okolo jedné minuty. Což nám dává minimálně 23 dotazů každou minutu při maximálním běhu firmy, a navíc další doprovodné dotazy na databázi, jako například vyčítání dat z databáze, či ukládání dalších parametrů.

Po tomto zhodnocení se přestalo přemýšlet o MS SQL. Produkty od firmy Microsoft mají sice spoustu výhod, hlavně s ohledem na konektivitu, ale cena placených a omezení neplacených verzí byla taková, že se produkt firmy Microsoft pro naši aplikaci nevyplatil. V případě, neosvědčení jiného řešení, zde byla možnost využít neplacenou verzi a omezení 10 GB uložených dat vyřešit občasným exportem dat mimo databázi. To ovšem zatím nebylo zapotřebí.

Konečný výběr nakonec připadl na databázi MySQL ve verzi Community. Důvodů k výběru této databáze bylo několik. Hlavní slovo v tomto rozhodnutí mělo vedení firmy, která se rozhodla využít ověřený produkt od velké firmy. V současnosti se z hlediska požadavků jednalo o dostatečné řešení a v případě potřeby bylo možné bez větší práce přejít na databázi MariaDB. A v neposlední řadě hrála pro skutečnost, že firma B+R automatizace, spol. s r.o. se podle manuálu zaměřovala především na databázi MySQL.

Později již po uskutečnění výběru databáze vydala firma B+R automatizace, spol. s r.o. aktualizace programu Automation studio, ve které se nástroj Mapp Database rozšířila i o podporu databáze PostgreSQL. Jelikož se v dané době již používala databáze MySQL bez větších obtíží, nebyl zde důvod uvažovat o této variantě.

3 Komunikace s PLC

Jedním z hlavních problémů při vytváření systému pro sběr dat bylo vyřešení otázky komunikace, jelikož se jednalo o vyčítání dat z různých výrobních strojů. Tento problém bylo možno částečně obejít díky napojení I/O ostrůvku na jednotlivé stroje a tím získat základní informace pomocí hardwaru od jednoho výrobce. Zakoupený hardware od firmy B+R automatizace, spol. s r.o. využíval pro svojí komunikaci mezi jednotlivými vstupy a PLC dva komunikační protokoly. Pro komunikaci se samotným I/O ostrůvkem využívalo zakoupené PLC X20CP1585 komunikační protokol Powerlink [22]. Komunikace mezi jednotlivými moduly I/O ostrůvku probíhala pomocí X2X sběrnice [23].

Důležitější otázkou však bylo, jak vyřešit komunikaci z PLC do vizualizace. Tato otázka byla rozšířena i o možnost, jak komunikovat s několika stroji, které již nějaký standart v síťové komunikaci měly. Bylo nutné zjistit, jaké komunikační protokoly PLC pro tento typ komunikace podporuje. Po konzultaci s výrobcem PLC bylo rozhodnuto využít komunikační protokol OPC UA, který PLC X20CP1585 podporuje.

3.1 Základní popis

Zkratka OPC UA (z anglického Open Platform Communications – Unified Architecture), se dá přeložit jako otevřená komunikační platforma (OPC) a sjednocená architektura (UA) [24]. Jedná se o komunikační protokol vycházející z protokolu OPC Classic. Tento protokol vyvinula společnost OPC Foundation za spolupráce s dalšími firmami působícími v oboru automatizace a robotizace [25]. Jedním z těchto partnerů OPC Foundation byla i firma B+R automatizace, spol. s r.o. [26].

Komunikační standard OPC UA byl vyvinut především z důvodu možnosti připojení hardwaru od různých výrobců. Dalším důležitou funkcí bylo vytvoření univerzálního komunikačního protokolu i pro další vrstvy systému, jako například komunikace se SCADA HMI [25]. Oproti původnímu protokolu OPC Classic se liší například tím, že OPC UA je možné provozovat na různých operačních systémech (nejen na platformách Microsoft) [24]. Na rozdíl od OPC Classic, který odděleně definuje přístup k procesním datům, historickým datům a alarmům, OPC UA nově využívá jednoho komunikačního přístupu a ke všem těmto datům přistupuje stejně a je tudíž tzv. samopopisný [27].

Hlavními požadavky na nový standard byly:

1. Nezávislost na platformě
2. Škálovatelnost
3. Použití jednoho i více vláken

4. Bezpečnost založená na nových standardech
5. Nastavitelný timeout pro každou službu
6. Rozdělení objemných datagramů na menší části [27]

3.2 Popis komunikace a zabezpečení

Protokol OPC UA podporuje dva typy komunikace. Binární komunikaci pomocí TCP/IP a komunikaci pomocí webové služby HTTP protokolu typu Klient – Server. Jako síť je možné využít již stávající metalickou, nebo Wi-Fi síť. Přenos dat probíhá pomocí tzv. zpráv mezi klienty a servery. Každé zařízení je popsáno vlastní třídou. Třída obsahuje jméno zařízení, použití, které definuje omezení zařízení, datový typ a popis, jenž obsahuje popis samotného zařízení a jeho účel. Pro adresaci se využívá jméno zařízení a k datům se přistupuje pomocí objektů [28].

Komunikace probíhá tak, že server vyšle na síť požadavek se jménem zařízení, ID bloku, metodu a zda chce číst nebo zapisovat. Pomocí jména zařízení se vybere konkrétní zařízení, pro které je zpráva určena. To obsahuje několik bloků. Konkrétní blok je vybrán pomocí ID. Do vybraného bloku je následně zaslán požadavek, zda se bude číst, nebo zapisovat. Nakonec metoda určí, co se bude s daty dít [28].

OPC UA využívá 4 úrovně zabezpečení. Jedná se o omezení toho, kdo s jakými daty bude moci nakládat, nebo je sledovat. První úroveň se nazývá Anonymní. V případě, že jsme nastavili pro určitou hodnotu toto zabezpečení, má možnost danou proměnnou pozorovat každý, kdo má na své straně platný certifikát. Jedná se o nejnižší zabezpečení, které je nastaveno automaticky. Druhá úroveň tzv. serverová autentizace využívá k ověření jméno a heslo. To v důsledku znamená, že z jakéhokoliv zařízení je možné se na server připojit pomocí kombinace jména a hesla. Klient ovšem neověřuje, zdali je server důvěryhodný. Jedná se o nejběžnější způsob zabezpečení. Třetí úroveň zabezpečení se nazývá autentizace pomocí certifikátu. Tato úroveň zajišťuje, že se na server může připojit pouze zařízení, které má platný certifikát. Tyto certifikáty (stejně jako jména a hesla v druhé úrovni) přiděluje administrátor a může jejich platnost i zrušit. V této úrovni stále nedochází k ověření důvěryhodnosti serveru ze strany klienta. K tomuto ověření dochází až ve čtvrté nejvyšší úrovni zabezpečení tzv. oboustranné autentizaci. To vyžaduje nastavení jak ze strany serveru, tak klienta [24].

3.3 Využití v průmyslu

Díky své struktuře se v průmyslu využívá všude tam, kde je zapotřebí propojit více různých zařízení. Jelikož v dnešní době je internetové připojení takřka v každé průmyslové budově, je jeho rozšíření velice jednoduché. V největší míře se protokol využívá v systémech částečné automatizace. Všude tam, kde je zapotřebí hlídat energie, ovládat teplotu, či doručovat kritická data. Dále je možné ho

využít všude tam, kde je zapotřebí určitý pojem o čase jednotlivých procesů. Příkladem může být počítání cyklů, či zaznamenávání události v určitém čase [28].

V naší konkrétní aplikaci bylo výše zmíněného protokolu využito pro propojení dat do vizualizace. Zde se PLC chovalo jako server. PLC X20CP1585 mohlo zároveň fungovat i jako klient. Této skutečnosti by se dalo využít, kdyby pro získávání dat z novějších strojů nebylo využíváno I/O ostrůvků, ale přímé komunikace. Tato varianta by byla možná pouze u novějších strojů a pro připojení by bylo nutné zakoupit na stroje komunikační karty.

4 Ukazatele úspěšnosti a efektivity výrobního podniku

Velké množství dat sebou přineslo problém, jak daná data efektivně vyhodnotit. Jelikož se jednalo o poměrně velké množství získávaných dat, bylo užitečné pro data vypovídající o výkonnosti a efektivitě výroby vytvořit koeficient, který by co nejlépe odrážel aktuální stav výroby a zároveň by byl porovnatelný i s výrobou budoucí. To by bylo možné využít pro zhodnocení úspěšnosti možných organizačních a technologických zásahů ve firmě. Ve sféře výroby existovalo těchto ukazatelů celá řada. Každý měl svá specifika a na výrobu pohlížel z různých úhlů, jak po stránce rychlosti a efektivity, tak finančního výtěžku. Finanční stránka bývá u většiny podniků zmapována nejlépe. Jedná se o mapování poměru výdaje vs. zisky. Tento náhled ovšem neumožňoval snímání konkrétního optimalizačního řešení, ale posuzoval stav firmy jako celku [29]. Pro snímání výrobních procesů bylo možné využít parametrů jako například klíčové ukazatele výsledku, ukazatele výkonnosti či klíčové ukazatele výkonnosti, které ve své práci popsali David Parmenter [30].

4.1 Klíčové ukazatele výsledků (KRI)

Tento soubor měřítek se využívá především ke zpětnému vyhodnocení výsledku. Nejedná se o parametry, které by bylo možné získávat v průběhu výroby. Z jejich hodnot je možné porovnat, zda se v uplynulé době dosáhlo zhoršení nebo zlepšení. Nevýhodou těchto parametrů je, že je sice možné porovnat jednotlivé měsíce nebo kvartály, ovšem pomocí koeficientů nezjistíme, kde se konkrétně může podnik zlepšit. Jedná se především o ukazatele typu: spokojenost zákazníků, spokojenost zaměstnanců, poměr nakoupeného materiálu a ceny, četnost škod v materiálu atd. Problematika těchto měřítek mnohdy je, že daná měřítka musíme kombinovat, abychom dostali relevantní výsledky [29].

4.2 Ukazatele výkonnosti (PI)

Jedná se o hlediska, která nějakým způsobem vypovídají o budoucím vývoji ve firmě. Oproti KRI jsou pouze nefinančního rázu a jsou měřeny výrazně častěji (nebo i průběžně). Tyto parametry vypovídají o výkonnosti procesů ve výrobě [30]. Bude uvedeno několik příkladů.

Počet zpuštěných zakázek ve výrobě. Tento parametr dává informace o tom, kolik zakázek je v provozu aktuálně rozděláno. Pokud je množství aktuálně zpracovávaných zakázek příliš vysoké, nemá smysl rozjíždět zakázky další. Pokud je ovšem rozdělaných zakázek málo, může dojít k tomu, že čas lidí nebude efektivně využit a výroba „vyhladoví“. Úkolem je držet množství zakázek na optimálním počtu [30].

Průběžná doba procesu. Doba, která uplyne mezi zavedením zakázky do výrobního procesu až po jejich výstup [30].

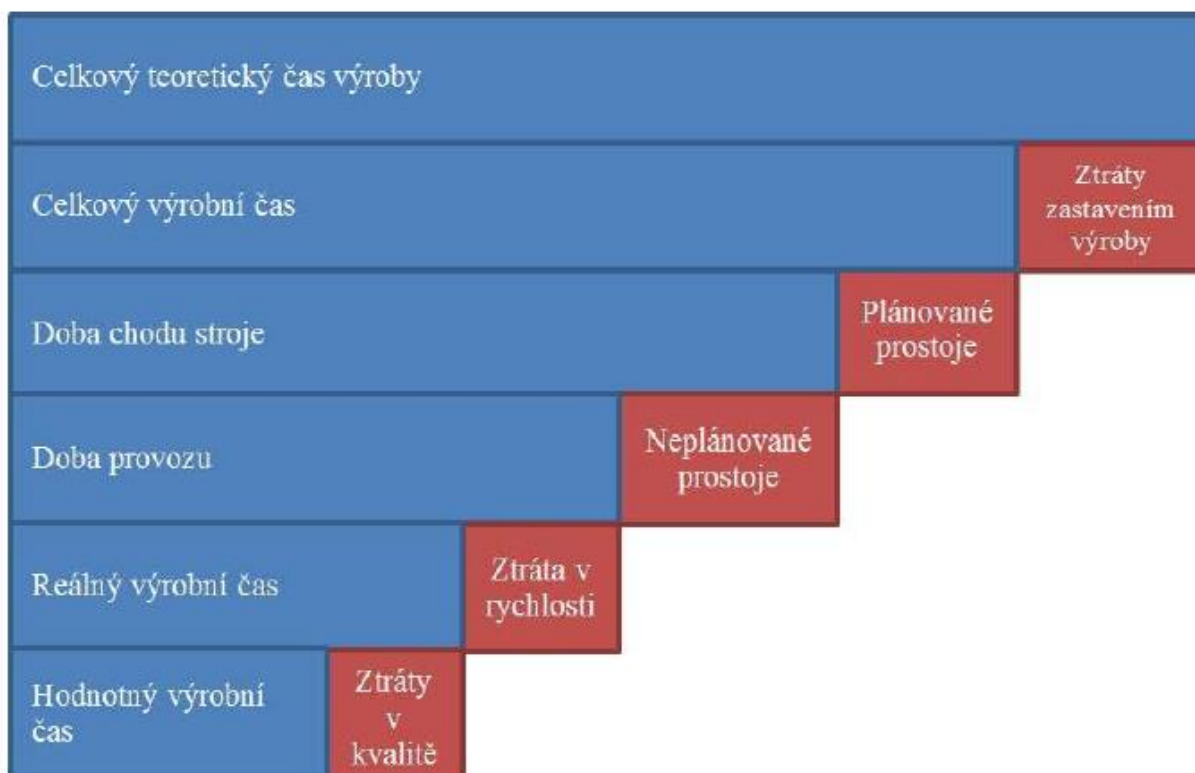
Efektivní využití doby procesu. Čas, který je skutečně potřeba k vyrobení zakázky ku času teoretickému [30].

4.3 Klíčové ukazatele výkonosti (KPI)

Klíčové ukazatele výkonnosti jsou podmnožinou ukazatelů výkonnosti (PI), které jsou pro výrobu kritické. Parametrů KPI je celá řada a autoři prací na toto téma málokdy uvádějí parametr, který by byl podle nich nejdůležitější [30]. Důvodem je především to, že každý podnik je specifický a nedá se paušalizovat nejlepší řešení volby parametru. KPI jsou ze své podstaty nefinančního rázu, tudíž jsou pro naše účely nejzajímavější. Mezi velmi používané parametry patří tzv. OEE, FPY a využití kapacit. V praxi se jich ovšem používá celá řada, jako například zpoždění doby dodávek, využití pracoviště a mnoho dalších [30].

4.3.1 Celková efektivita zařízení (OEE)

Overall Equipment Efficiency neboli celková efektivita zařízení je ukazatel efektivity využití výrobního stroje a udává se v procentech. Jedná se o parametr, který je možné sledovat online a do jeho hodnoty zasahuje několik faktorů. Konkrétně se jedná o tzv. výkon, kvalitu a dostupnost. Výsledný výpočet nám říká, jak efektivně se daří stroj využívat a kde je možné výrobu zefektivnit. Z výsledku je možné odvodit poměr času, kterého je potřeba k vyrobení kusu zasláného k zákazníkovi v poměru ku času teoretickému. Vliv na výslednou hodnotu OEE je možné vyjádřit pomocí grafu 4.1 [31].



Graf 4.1 Faktory ovlivňující hodnotu OEE [32]

Do parametru dostupnosti stroje se promítají ztráty časové z hlediska zastavování a přerušování výroby. Jedná se o čas teoreticky využitelný pro výrobu a čas, kdy stroj reálně vyráběl. Do tohoto času se projeví všechny plánované i neplánované prostoje. Plánované prostoje souvisí s údržbou nebo přípravou stroje pro budoucí výrobu, neplánované prostoje souvisí s poruchami [31].

Parametr výkonu je dán poměrem mezi teoretickým výrobním časem na jeden vyrobený díl a časem reálným [31]. Tento parametr je ovlivňován především obsluhou stroje. Jestliže v našem případě byl cyklus plánovaný na 60 s a výsledný čas výroby byl při ručním vyndávání dílů ze stroje delší, docházelo ke snížení tohoto parametru.

Posledním parametrem OEE je kvalita. V tomto parametru je zahrnut poměr všech vyrobených kusů k počtu bez vady vyrobených kusů [31].

Celkový výsledek se následně vypočítá podle rovnice 1 [30]. V praxi je vhodné sledovat nejen celkový parametr OEE, ale také jeho dílčí části. Z těch je lépe patrné, v kterém úseku výrobního procesu je potřeba výrobu zefektivnit. Jestliže klesne parametr dostupnosti, je potřeba zhodnotit, zda je chyba na straně údržby, která nebyla schopna udržovat stroj v takovém stavu, aby byl stroj výroby schopný nebo na straně plánování výroby, která nedokázala navrhnout efektivní využití stroje. Parametr výkonu hodnotí především obsluhu stroje. Při nízkém výkonu je vhodné zhodnotit, zdali obsluha stroje pracuje dle požadavků, nebo zda není nutné navrhnout efektivnější způsob práce na pracovišti. Nízký koeficient kvality může indikovat problém v nastavení stroje nebo špatnou průběžnou kontrolu a opožděné zjištění nedostatečné kvality výroby.

$$\frac{\text{Délka směny} - \text{Přestávky} - \text{Prostoj}}{\text{Délka směny} - \text{Přestávky}} \times \frac{\text{Celkové množství výrobků}}{\text{Délka směny} - \text{Přestávky} - \text{Prostoj}} \times \frac{\text{Počet bezchybných výrobků}}{\text{Celkové množství výrobků}} \times 100 \quad (1)$$

Tento parametr je oblíbený především v MES systémech¹, které mají za úkol pomoci s plánováním a snímáním výroby. Jako příklad je možné zmínit systémy MES Pharis od firmy Unis [33] nebo Comes OEE od firmy Compas [34]. Dle vyjádření zástupce firmy Unis, se kterým byl veden rozhovor při obchodním jednání na téma koeficientu OEE, je teoreticky možné dosáhnout hodnoty OEE až 80 %. V praxi dle jeho zkušeností podniky s nejlepším hodnocením dosahovaly hodnoty OEE 60 %.

¹ z anglického označení Manufacturing Execution Systems, které se volně překládá do češtiny jako Výrobní informační systémy

4.3.2 Výtěžnost prvního průchodu (FPY)

Tento parametru udává kvalitu kontroly jakosti. Parametr vyjadřuje množství špatných kusů, které se dostanou přes první kontrolní bod [30]. V případě firmy Plast Form Service IM s.r.o. se v největší míře jedná o průchod přes obsluhu stroje. U velkého množství kusů byla předepsána 100% vizuální kontrola. Ta by teoreticky měla odstranit veškeré zmetkové díly. U tohoto parametru je důležitá prvotní definice prošlého zmetku a bezchybně vyrobeného kusu [35]. V některých případech je FPY možné chápat i jako ekvivalent koeficientu kvality vypočítávaného v rámci OEE.

4.4 Aplikace koeficientů ve firmě Plast Form Service IM s.r.o.

V rámci firmy Plast Form Service IM s.r.o. by bylo možné sledovat většinu z uvedených parametrů. Pokud se ovšem omezíme na parametry, které by bylo možné získávat v rámci snímání dat ze stroje, se ukázalo jako nejhodnější měřit parametr OEE. Z hlediska jednotlivých parametrů bylo nejjednodušší sledovat parametry výkonu a dostupnosti. Tyto parametry byly odvozeny přímo z chodu stroje a jejich měření bylo závislé pouze na nastavení jediného parametru, a to požadovaného času cyklu. Co se týče parametru kvality, bylo potřeba většího zásahu uživatele. Varianty se nabízely dvě. První variantou bylo nalezení způsobu, kterým by obsluha přímo oznamovala vyrobení špatného kusu. Druhou variantou bylo vytvoření povinnosti zápisu údaje o počtu zmetků do papírového protokolu. Počet zmetků byl i před zavedením snímání výroby zaznamenáván, tudíž se jednalo pouze o přepis hodnot do databáze a nebylo zapotřebí organizační změny.

Ve prospěch měření OEE hovořil i fakt, že v rámci programu Automation Studio bylo možné využít jednu z Mapp komponent, která sloužila k usnadnění programování právě tohoto parametru. V zásadě bylo zapotřebí pouze definovat vstupní parametry a vše ostatní již bylo vypočítáváno v rámci Mapp komponenty.

Dalším parametrem, který přicházel v úvahu byl FPY. Tento parametr se sice nevztahoval k výrobě jako takové, ale měl velký dopad na chod kontroly celého procesu. V úvahu přicházela kontrola především těch dílů, které se ve výrobě pohybovaly (například od stroje na montáž). Tento parametr by mohl být kontrolován pomocí sledování počtu kusů, které nebyly původně označeny za zmetky, ale následně byly zachyceny na jiném pracovišti v rámci firmy. V tuto chvíli ovšem nebylo měření tohoto parametru zásadní, jelikož vedení firmy rozhodlo o napojení co největšího množství strojů na úkor rozšiřování funkcí systému pro sběr dat z výroby.

5 Popis řešení pro snímání provozu

V této kapitole bude popsána samotná struktura celého řešení jak po hardwarové, softwarové tak organizační stránce (v rámci pracovní náplně jednotlivých pracovníků firmy). Celý systém byl testován na dvou výrobních strojích. Oba stroje byly vyrobeny firmou Battenfeld. Jednalo se o stroj s označením 01-4500 (interní očíslování stroje – uzavírací síla v kN) se systémem Unilog 4000 a stroj 12-1000 se systémem Unilog B2. Důvodů pro výběr těchto strojů bylo několik a hlavním z nich byla skutečnost, že se jednalo o dva kritické stroje, co se množství zakázek týče. Ne každá forma lze umístit na každý stroj z důvodu různých technologických nároků. Neméně důležitým důvodem bylo, že systém Unilog 4000 a Unilog B2 měly ve firmě největší zastoupení. Systém sice nebyl důležitý z hlediska komunikace, ale v každé řadě systémů se napříč různými modely strojů dodržovaly podobné principy rozvodu signálů po stroji, které bylo potřeba využít.

5.1 Struktura řešení sběru dat

Úvodem bude nastíněna základní koncepce celého sběru dat. V dalších kapitolách bude podrobněji rozepsáno konkrétní řešení celého systému.

Celý systém pro sběr dat se dal rozdělit do několika samostatných částí, o kterých se dalo v rámci mezí uvažovat samostatně a bylo možné je v průběhu jeho vytváření vyměnit. První důležitou částí bylo dovedení signálů ze strojů do jednoho místa. Součástí této diplomové práce bylo i vytvoření programu, který by sloužil pro výpočet jednotlivých operací. Další částí práce byla i databáze pro ukládání dat jak výrobních, tak informačních (informace o dílech, strojích...). Poslední částí bylo vytvoření vizualizace a komunikace mezi vizualizací a řídicím programem.

Jelikož byla možnost využít pro vizualizaci a snímání a vyhodnocení dat jeden programový nástroj od firmy B+R automatizace, spol. s r.o. s názvem Automation Studio, bylo této možnosti využito. To usnadnilo propojení těchto dvou částí a podstatně zrychlilo vývoj celého řešení.

Veškeré prvky byly dimenzovány tak, aby bylo možné systém rozšířit do celé firmy. Konečným cílem celého projektu bylo připojení všech 23 strojů a vytvoření 3 větších vizualizačních pracovišť (konkrétně na stanoviště pro technologa, mistry a kontrolu kvality). Dále se v konceptu počítalo s vizualizací na velké obrazovce v největší výrobní hale. Konečná podoba systému by měla zahrnovat i vizualizaci na každém výrobním pracovišti, pro zadávání parametrů přímo pracovníkem u stroje. V první fázi, na kterou byla napsána tato práce, počítá s připojením dvou výše zmíněných strojů a vizualizací pro technologa a kontrolu kvality (toto stanoviště bylo dočasně sdíleno i s mistry směny).

Základní rozvržení celého konceptu pro sběr dat je možné vidět na obrázku 1.1

Hardwarová část byla řešena modulárním systémem pod názvem X20 od firmy B+R automatizace, spol. s r.o. Hlavní výpočetní komponentou bylo PLC X20CP1585. Toto PLC sloužilo pro zpracování veškerých snímaných dat. Dále pak sloužilo pro chod vizualizace a komunikaci s databází. Toto PLC bylo koncipováno tak, aby v budoucnu mohlo zajistit dostatečný výpočetní výkon pro sběr dat z celé firmy.

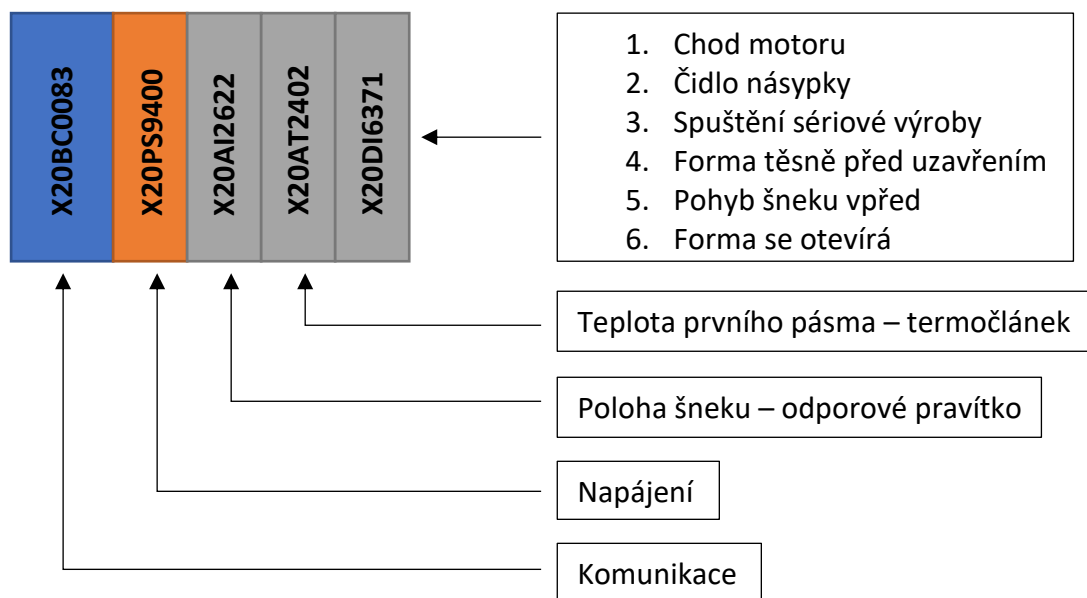
K oběma strojům byly připojeny I/O ostrůvky. Každý ostrůvek obsahoval několik vstupních modulů. Prvním modulem byla karta X20DI3671, která umožňovala snímání 6 digitálních vstupů. Na tyto vstupy byly přivedeny signály:

1. **Chod Motoru** – Tento signál indikoval provoz motoru pro rozvod hydraulické kapaliny ve stroji. V případě, že stroj zaznamenal problém při výrobě, automaticky vypnul motor. Tím se dalo indikovat neplánované zastavení výroby.
2. **Čidlo násypky** – Signál byl napojen na čidlo, které bylo ke stroji přidáno. Toto čidlo mělo za úkol snímat spodní hladinu materiálu v zásobníku a bylo ke stroji přidáno z důvodu rizika nedostatku materiálu v násypce. To by mělo za následek dlouhé zdržení při opětovném najíždění stroje.
3. **Spuštění sériové výroby** – signál byl napojen na tlačítko, které bylo instalováno na ovládací panel stroje. Bylo zde přidáno z důvodu oddělení přípravného provozu při najíždění výroby a samotné sériové výroby.
4. **Forma těsně před uzavřením** – Jednalo se o signál, který stroj vydává těsně před dovřením formy do tlaku. V rámci stroje se jednalo o bezpečnostní prvek sloužící pro ochranu formy před špatným uzavřením.
5. **Pohybu šneku vpřed** – Během výroby docházelo ve stroji k pohybu šneku vpřed v rámci procesu plastikace. Po celou dobu pohybu vpřed byl signál aktivní. V okamžiku zastavení pohybu se signál přepnul do logické nuly
6. **Forma se otevírá** – Jednalo se o signál, který byl aktivní po dobu otevírání formy.

Signály 4, 5 a 6 sloužily pro definici jednoho výrobního cyklu. Jestliže stroj prošel všemi popsánymi fázemi, bylo možné předpokládat vyrobení jednoho dílu. Veškeré tyto signály byly snímány z karet stroje, ze kterých bylo možné získat signály o hodnotě 24 V reprezentující logickou 1.

Další kartou v rámci ostrůvku byla X20AI2262, která obsahovala dva analogové vstupy. Využíván byl pouze jeden vstup, a to pro snímání signálu polohy odporového pravítka. Toto pravítko bylo napojeno na polohu šneku a snímalo tzv. polštáře. Zde pravítko snímalo množství materiálu zbývajících v prostoru před šnekem po dokončení vstřiku do formy [36]. Tato hodnota byla důležitá jednak z kontroly stability výroby, tak pro kontrolu dojezdu šneku do nejzazší hranice (tedy na nulu). Tento stav mohl nastat z mnoha různých důvodů (špatného nastavení stroje, špatného materiálu atp.).

Poslední karta X20AT2402 byla využita pro měření teploty termočláneku. Tento signál byl vyveden ze senzoru teploty na tzv. prvním pásnu (topný pás umístěný těsně za vstřikovací tryskou na komoře stroje). Tento senzor se nacházel nejbližší k vyráběnému dílu a s jeho pomocí se dala odvozovat konečná teplota taveniny vstupující do stroje. Tato hodnota byla sledována hlavně za účelem snímání stability procesu.



Obrázek 5.1 Schéma I/O ostrůvku se zapojením senzorů

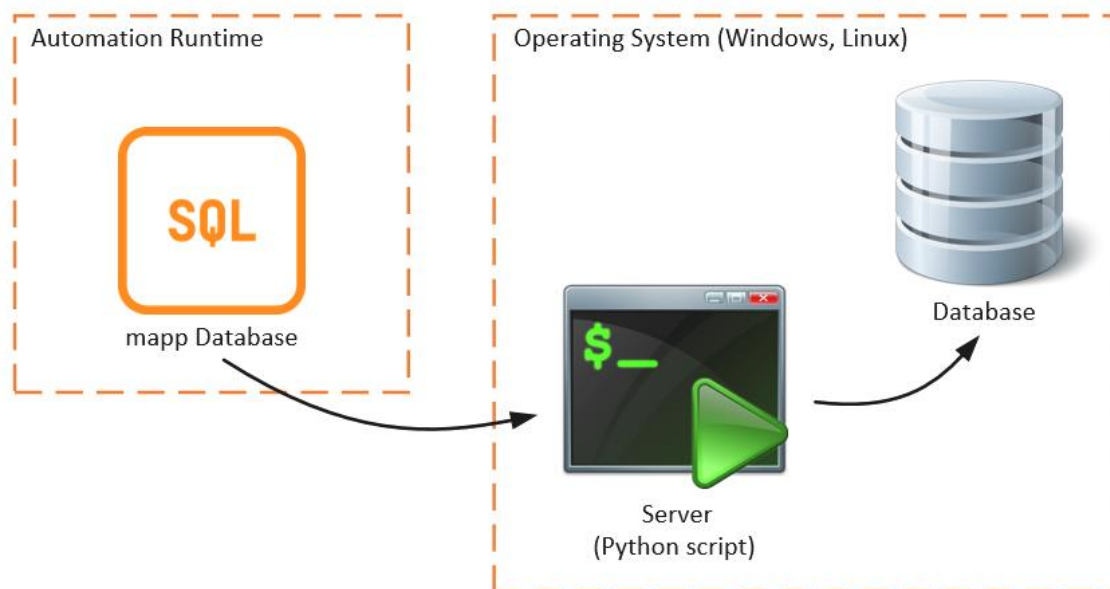
Veškerá napojení byla ještě před samotným připojením ke stroji otestována mimo stroj na snímačích stejného typu, jako se nacházely ve strojích. Hlavním důvodem bylo ověření, zda připojení dalšího odběrného místa signálu nebude ovlivňovat signál jdoucí do stroje.

K I/O ostrůvkům byl přiveden datový ethernetový kabel RJ45. Dle doporučení výrobce musel mít kabel stínění minimálně odpovídající standardu CAT – 5 s připojením na kostru I/O ostrůvků a PLC, a to z důvodu potencionálně velkého rušení od motorů jednotlivých strojů. Propojení strojů do jedné sítě bylo vyřešeno pomocí modulů I/O ostrůvku X20BC0083, které umožňovaly sériové zapojení strojů. Tímto způsobem ovšem nebude možné vytvořit paralelní zapojení více větví. V budoucnu se tento rozvod bude řešit pomocí Hubu 0AC808.9-1, který umožňoval rozvětvení na 7 samostatných větví. Tím se navíc ušetří velké množství kabelu. Do každé větve bylo plánováno zapojit maximálně 5 za sebou jdoucích strojů ve vzdálenosti maximálně 80 m mezi každým strojem, aby bylo i s rezervou dodrženo doporučení výrobce. Toto doporučení bylo odvozeno od požadavku dodržení přednastavené doby odpovědi každého modulu do 25 μ s. Každý Hub a každý modul X20BC0083 způsoboval drobné zpoždění. V případě potřeby ovšem bylo možné tuto dobu prodloužit na delší dobu. Teoreticky bylo možné do celé sestavy adresovat až 256 různých zařízení, což bohatě zajišťuje dostatečný počet) pro budoucí rozvoj systému. Délka byla omezena hlavně z důvodu rušení signálu. Komunikace po této lince probíhala pomocí protokolu Powerlink.

5.1.1 Ukládání dat

Data bylo možné ukládat dvěma různými způsoby. Prvním způsobem bylo ukládání na CompactFlash², která disponovala pamětí 4 GB. Do této paměti se ukládaly především záznamy o alarmech. Na této paměti je též uložen celý program. Do budoucna bylo plánováno využít tuto paměť i pro ukládání jednotlivých výrobních dokumentů pro podporu bezpapírového provozu. Tuto paměť bylo možno v případě potřeby rozšířit.

Druhým způsobem bylo využití SQL databáze. Jak již bylo zmíněno, v programu Automation studio bylo možné využít Mapp komponenty, díky kterým bylo možné jednoduše s databází pracovat. V aktuální chvíli využívá program mezi databází a PLC ještě mezistupeň v podobě Pythonovského skriptu. Tento skript bylo možné snadno nakonfigurovat a využít k připojení k různým druhům databází. Bylo zapotřebí zadat typ používané databáze a následně port PLC a databáze.



Obrázek 5.2 Princip připojení k databázi

Bohužel s touto komponentou byly v průběhu programování největší potíže. Jelikož se jednalo o poměrně nový produkt, obsahoval spoustu chyb v kódu. Největší potíže dělaly hlavně dvě konkrétní chyby. První problém byl v kontrole spojení. Mapp komponenta byla vybavena diagnostickými prvky pro zjištění chyby komunikace se skriptem. Bohužel tomu tak nebylo. To se projevovalo hlavně při restartu počítače, kdy se skript vypnul. Po restartu bylo v počítači s databází nastaveno, aby se skript spustil automaticky. Pro obnovení spojení ovšem bylo nutné restartovat Mapp komponentu. Zde se ovšem objevil druhý zásadní problém. Pokud se PLC pokusilo při přerušení spojení odeslat dotaz do databáze a následně po neúspěšném pokusu došlo k restartu Mapp komponenty, celé PLC spadlo do servisního módu a přestalo vykonávat svoji funkci. Veškeré tyto problémy byly zaslány na technickou podporu a bylo přislíbeno jejich vyřešení.

```

D:\MySQL\python\python.exe
i_ve_smene', 'vyorbni_mod', 'prekroceni_casu') VALUES ('0.000000', '12', '1000', '2020-4-24 12:40:7.00', '171', '-0.611526', '290', '0', '1', '0')
192.168.1.62 - - [24/Apr/2020 12:40:22] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 12:40:22] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `zaznam_vyroby` (`id_zaznam`, `id_stroj`, `id_dily`, `cas_odeslani`, `cas_cyklu`, `polstar`, `teplota`, `porad
i_ve_smene', 'vyorbni_mod', 'prekroceni_casu') VALUES ('0.000000', '12', '1000', '2020-4-24 12:43:0.00', '172', '-0.555763', '290', '0', '1', '0')
192.168.1.62 - - [24/Apr/2020 12:43:14] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 12:43:15] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `zaznam_vyroby` (`id_zaznam`, `id_stroj`, `id_dily`, `cas_odeslani`, `cas_cyklu`, `polstar`, `teplota`, `porad
i_ve_smene', 'vyorbni_mod', 'prekroceni_casu') VALUES ('0.000000', '12', '1000', '2020-4-24 12:45:51.00', '171', '-0.555763', '290', '0', '1', '0')
192.168.1.62 - - [24/Apr/2020 12:46:06] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 12:46:06] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `zaznam_vyroby` (`id_zaznam`, `id_stroj`, `id_dily`, `cas_odeslani`, `cas_cyklu`, `polstar`, `teplota`, `porad
i_ve_smene', 'vyorbni_mod', 'prekroceni_casu') VALUES ('0.000000', '1', '1000', '2020-4-24 12:46:11.00', '0', '0.167585', '298', '0', '1', '0')
192.168.1.62 - - [24/Apr/2020 12:46:26] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 12:46:26] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `zaznam_vyroby` (`id_zaznam`, `id_stroj`, `id_dily`, `cas_odeslani`, `cas_cyklu`, `polstar`, `teplota`, `porad
i_ve_smene', 'vyorbni_mod', 'prekroceni_casu') VALUES ('0.000000', '12', '1000', '2020-4-24 12:48:43.00', '171', '-0.555763', '290', '0', '1', '0')
192.168.1.62 - - [24/Apr/2020 12:48:57] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 12:48:58] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `cas_spojeni` (`cas_spojeni`) VALUES ('2020-4-24 13:37:31.00')
192.168.1.62 - - [24/Apr/2020 13:37:45] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 13:37:46] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `statistika` (`id_stroj`, `smena`, `datum`, `dostupnost`, `vykon`, `kvalita`, `oe`) VALUES ('12', '1', '2020-4-
24 0:0:0.00', '1.000000', '0.000000', '0.000000', '0.000000')
192.168.1.62 - - [24/Apr/2020 13:59:05] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 13:59:06] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `statistika` (`id_stroj`, `smena`, `datum`, `dostupnost`, `vykon`, `kvalita`, `oe`) VALUES ('1', '1', '2020-4-
24 0:0:0.00', '1.000000', '0.000000', '0.000000', '0.000000')
192.168.1.62 - - [24/Apr/2020 13:59:10] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 13:59:11] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `cas_spojeni` (`cas_spojeni`) VALUES ('2020-4-24 14:37:31.00')
192.168.1.62 - - [24/Apr/2020 14:37:45] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 14:37:46] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `cas_spojeni` (`cas_spojeni`) VALUES ('2020-4-24 15:37:31.00')
192.168.1.62 - - [24/Apr/2020 15:37:45] "POST / HTTP/1.1" 200 -
192.168.1.62 - - [24/Apr/2020 15:37:46] "POST / HTTP/1.1" 200 -
query will be executed: INSERT INTO `cas_spojeni` (`cas_spojeni`) VALUES ('2020-4-24 16:37:31.00')
192.168.1.62 - - [24/Apr/2020 16:37:46] "POST / HTTP/1.1" 200 -

```

Obrázek 5.3 Běh skriptu v Pythonu

Tyto problémy byly dočasně řešeny tak, že v okamžiku problému byl zaslán upozorňující email na tuto událost, aby bylo možné problém vyřešit manuálně některým pracovníkem. V tuto chvíli sice nebyla databáze funkční, ovšem bylo možné alespoň udržet v provozu veškeré alarmy, které hlídaly výrobu.

5.1.2 Vizualizace

Výhodou využití programu Automation studio bylo, že softwarové prostředí pro tvorbu vizualizace a prostředí pro tvorbu programu byly zakomponovány do jednoho vývojového prostředí. To umožnilo snazší vytváření celého systému. Pro tvorbu vizualizace se využívala komponenta Mapp View, která byla k zakoupení spolu s jednotlivými uživatelskými přístupy. Vizualizace byla postavena na JS, HTML 5 a CSS. Samotnou vizualizaci šlo následně spustit na jakémkoliv prohlížeči v rámci sítě.

Následně se ovšem ukázalo, že v některých prohlížečích nefunguje vizualizace zcela správně, a proto bylo nutné využívat pouze prohlížeče Chrome, na kterém byla vizualizace výrobcem testována.

Komunikace byla řešena pomocí protokolu OPC UA, který byl v rámci Automation studia též zakomponován, a tudíž byla práce velice jednoduchá.

V první fázi byly pořízeny licence pro 3 uživatele. Jedna licence sloužila pro hlavního technologa, druhá pro mistra společně s kontrolou kvality a třetí byla sdílelo vedení firmy a testování vývoje systému.

Pro tvorbu vizualizace bylo možné využít grafických editorů, které usnadňovaly práci při vytváření podoby vizualizace. V době vytváření programu bylo ovšem nutné některé věci v rámci vizualizace ošetřovat přímo v samotném kódu. Například se jednalo o tvorbu akcí a událostí na jednotlivá tlačítka, či celková konfigurace vizualizace (využívané obrazovky, jméno vizualizace atd.).

5.1.3 Hlavní funkce programu

Program byl koncipován tak, aby v první řadě informoval o aktuálním provozu a chybách v provozu strojů. To bylo realizováno pomocí vizualizace provozu v podobě tabulky a e-mailových upozornění na alarmové stavy (například zastavení stroje při sériové výrobě, docházející materiál atp.).

Druhou funkcí bylo využití zaznamenávání dat, díky kterým bylo možné zjistit problémy v historii výroby. K tomu sloužil výpis z historie cyklů z databáze a grafy základních hodnot, jako byla teplota, čas cyklu a polštář.

Posledními výstupními údaji byly statistické záznamy o chodu stroje, které se ukládaly do databáze a bylo si je možné prohlédnout v rámci vizualizace.

5.2 Popis programu

Kód pro vykonávání programu bylo možné rozdělit na několik samostatných částí. To mělo několik důvodů a následných výhod. PLC disponovalo možností rozdělit kód do 8 samostatných tříd, kde bylo možné nastavit dva důležité parametry. Prvním parametrem byla perioda, se kterou se program opakoval. Ta musela být celočíselným násobkem frekvence procesoru. To umožnilo rozdělit program na části, kde bylo potřeba vykonávat danou sekvenci často a na části, které probíhaly jen jednou za čas. Dalším parametrem byla tolerance překročení. Pokud v rámci provádění třídy uplynul čas, ve kterém bylo možné program vykonat, a jestliže bylo nutné obsloužit program s vyšší důležitostí, došlo k přerušení stávajícího programu. Ten pokračoval dále v okamžiku, kdy na něj zbyl

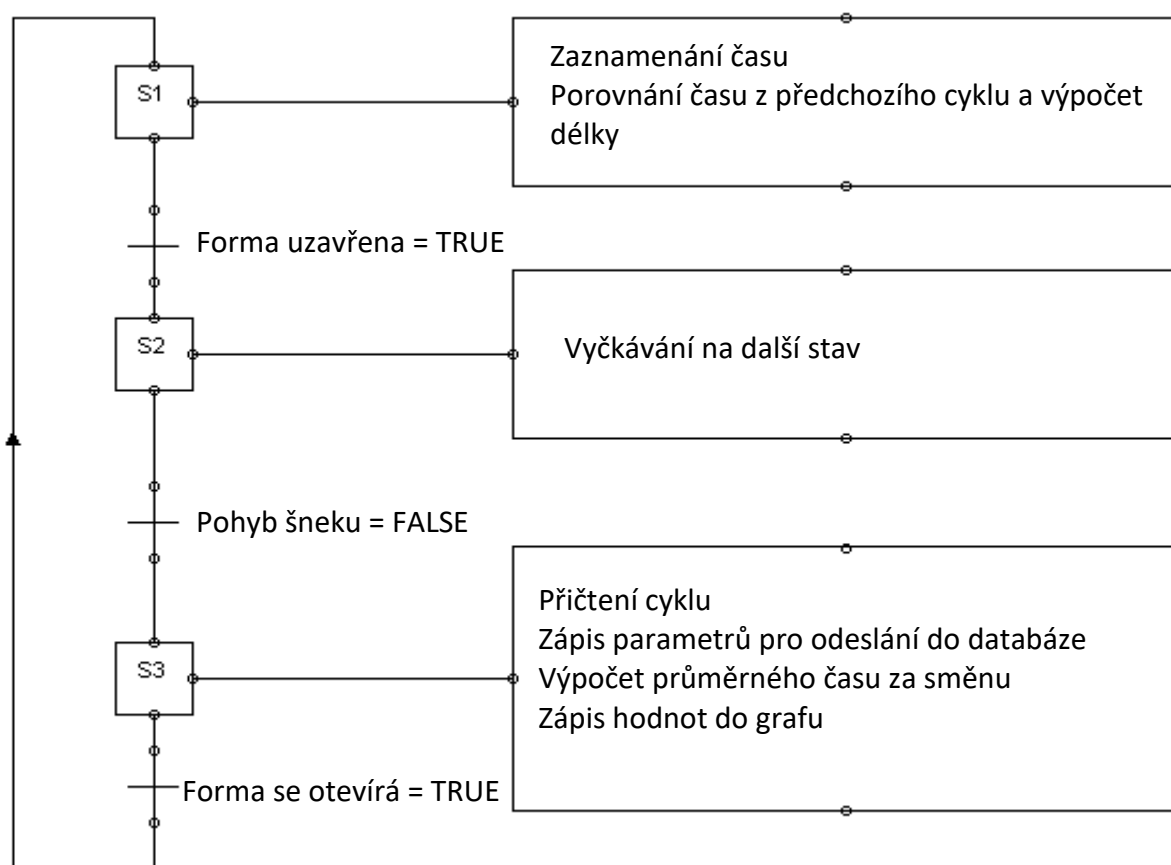
čas. Tolerance udávala, jaká maximální doba mohla uplynout od začátku vykonávání konkrétní třídy. Díky tomu bylo možné optimalizovat vykonávání programu.

Pro programování bylo využito principu stavového stroje, díky kterému šlo program vytvořit přehledněji. Celý program a proměnné byly připraveny k tomu, aby bylo možné napojit snadno další stroje.

V rámci programu byly vytvořeny 4 hlavní části, které budou detailněji popsány. Jednalo se o počítání cyklů stroje, měření statistického údaje OEE, ukládání do databáze a zaslání e-mailového upozornění.

5.2.1 Cyklus stroje

Počítání cyklů bylo řešeno pomocí stavového stroje (obrázek 5.4), který přepínal své stavy na základě fyzického pohybu stroje. Stavy byly rozdělena na Forma se uzavřela (S1), Vstřikování materiálu (S2) a Forma se otevírá (S3).



Obrázek 5.4 Znáznornění stavového stroje Cyklu pomocí SFC diagramu

V rámci S1 byla vypočítávána délka cyklu. Ten byl definován od zavření formy do následného zavření formy. Výpočet probíhal pomocí porovnání času z předchozího průchodu tímto stavem a

časem aktuálním. To s sebou neslo problém, že čas cyklu byl zjištěn až při začátku vykonávání dalšího cyklu, proto se to muselo v programu ošetřit. V případě dlouhé prodlevy byla do databáze zapsána hodnota času 0. V případě potřeby bylo možné zjistit čas pomocí časového rozdílu zápisu jednoho a druhého údaje. Důvodem bylo omezení na straně hardwaru, kde bylo nutné pracovat s dostupnými signály. Výsledná nepřesnost zaznamenávání dat byla zanedbatelná. Veškeré tyto procesy byly provedeny při přechodu do dalšího stavu, který byl inicializován pomocí zapnutí signálu forma uzavřena.

Ve stavu S2 bylo pouze vyčkáváno na ukončení vstřikovacího procesu. Do dalšího stavu se dostalo pomocí deaktivace signálu pohybu šneku vpřed. Hlavní význam tohoto stavu bylo, že v okamžiku, kdy nedošlo k tomuto pohybu šneku, a tudíž i průchodu tímto stavem, nedošlo k vyrobení dílu.

Ve stavu S3 byl v první fázi navýšen počet sériových cyklů. To se stalo pouze za předpokladu, že bylo aktivní tlačítko sériové výroby. Po případném opomenutí zapnutí tohoto tlačítka bylo po určitém množství kusů bez sepnutého tlačítka zasláno e-mailové upozornění.

Dále zde byla připravena strukturovaná proměnná pro odeslání dat do databáze. Zapisováno bylo pořadí cyklu v rámci směny, čas odeslání, označení dílu a technologické parametry. Dále zde byl vykonán zápis hodnot pro výpočet průměrného času na díl za směnu. Sice by bylo jednodušší využít data o času cyklu databáze, ale z důvodů výše zmíněných problémů bylo potřeba tento údaj od databáze oddělit. Nakonec byla provedena aktualizace grafu ve vizualizaci. Stejně jako ve stavu jedna byly tyto úkony provedeny v rámci přechodu do dalšího stavu. Ten byl inicializován signálem Forma se otevírá.

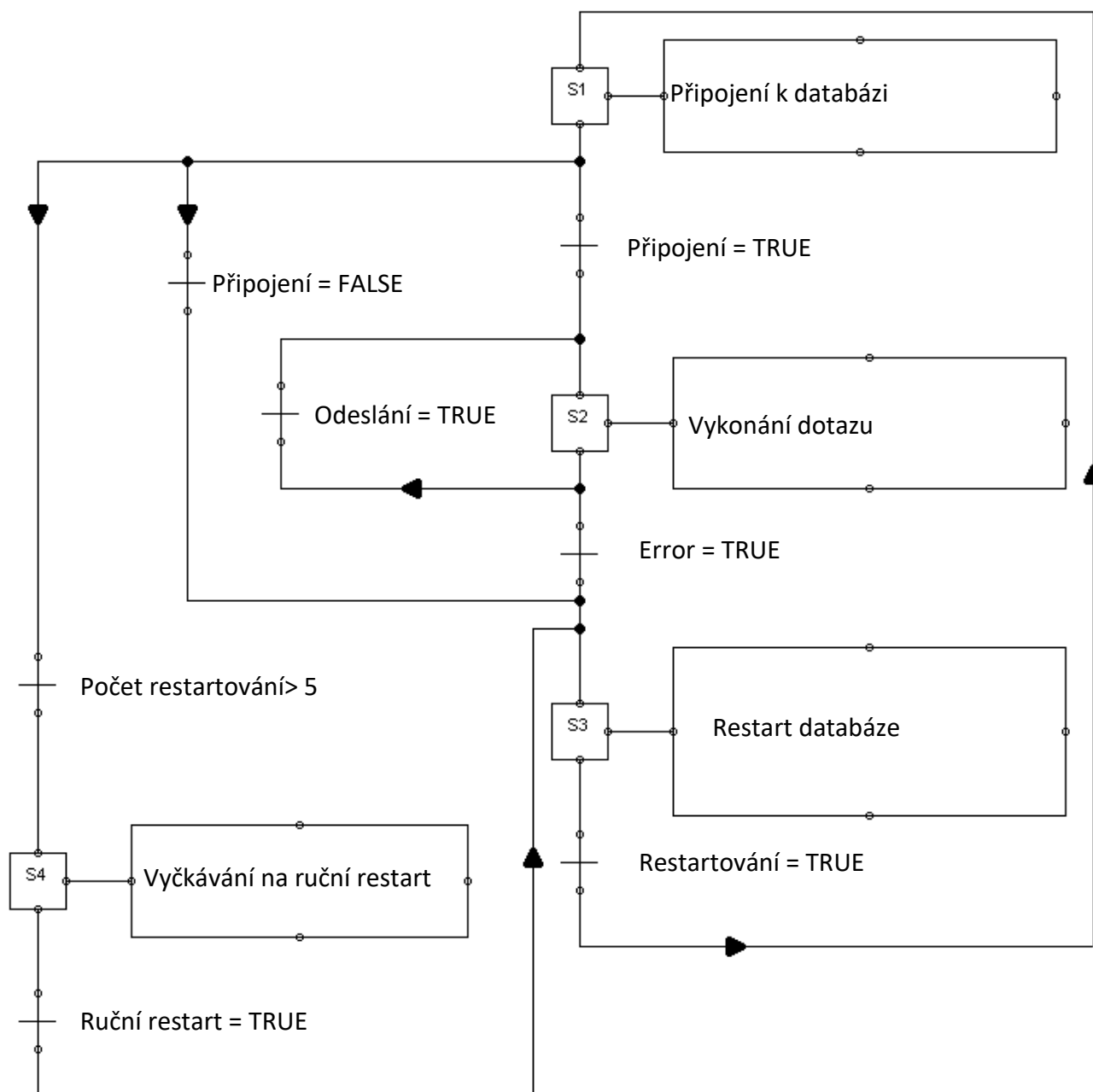
5.2.2 Měření OEE

K měření statistického údaje OEE byla využita Mapp komponenta s názvem Mapp OEE. K této Mapp komponentě byly přivedeny potřebné informace (počet cyklů, zastavení stroje atd.). Následně bylo zapotřebí provést konfiguraci údajů o časech směn a pauz pro správný výpočet. Koeficient OEE se měřil a vypočítával kontinuálně s tím, že při každém dokončení směny byla hodnota OEE zapsána do databáze a vynulována. V rámci výpočtu výkonosti stroje byly vypočítávány další dva parametry, které byly pojmenovány jako OEE24h a OEE5d. Jednalo se o průměrnou hodnotu OEE na stroji v posledních 24 hodinách a 5 dnech. To umožňovalo vytvořit si obrázek nejenom o aktuální směně, ale i o celkovém využití stroje. V budoucnu při napojení většího počtu strojů bude zjišťována i celková hodnota na všech snímaných strojích.

5.2.3 Ukládání do databáze

Celá práce s databází byla vytvořena na principu stavového stroje (viz. obrázek 5.5) a byla zde využita Mapp komponenta Mapp Database. Tento stavový stroj má velké množství stavů, které jde ovšem rozložit do několika kategorií. První kategorií (S1) je připojování k SQL databázi. Druhá

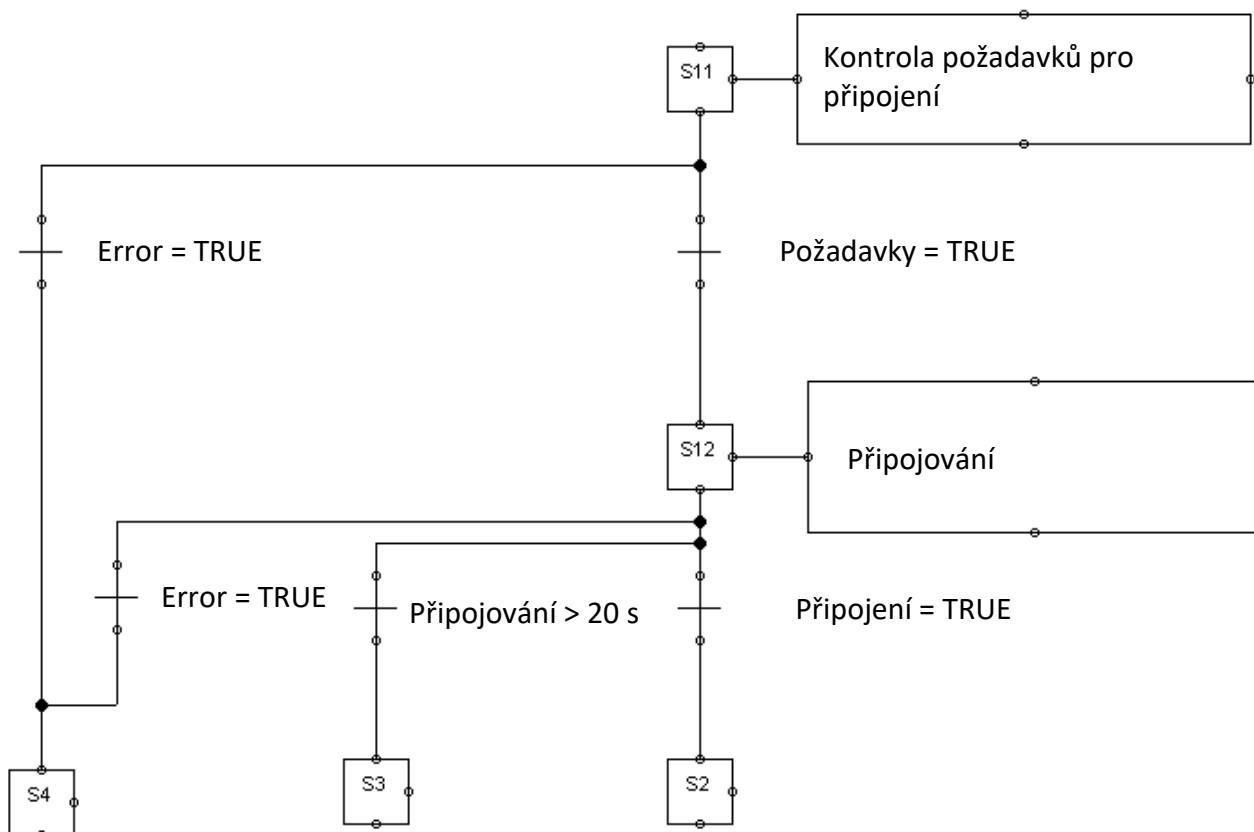
kategorie je vyčkávání na příkaz spuštění dotazu a jeho samotné provedení (S2). Třetí částí je restart databáze (S3). Poslední část je vyčkávání při poruše (S4). Stavový stroj byl programován tak, aby bylo možné program automaticky restartovat při vzniku problému, to ovšem díky výše zmíněným problémům s mapp komponentou nebylo možné. Zde bude popsána především zamýšlená varianta s popisem aktuálních opatření.



Obrázek 5.5 Znárodnění stavového stroje Databáze pomocí SFC diagramu

Při připojování databáze (S1) (viz obrázek 5.6) prochází stavový stroj těmito stavy. Ve stavu S11 dochází při splnění podmínek pro umožnění připojení k vyslání povelu k začátku připojování. Tím se dostane do stavu S12, kde se vyčkává, než samotné připojení proběhne. Pokud je vše úspěšně

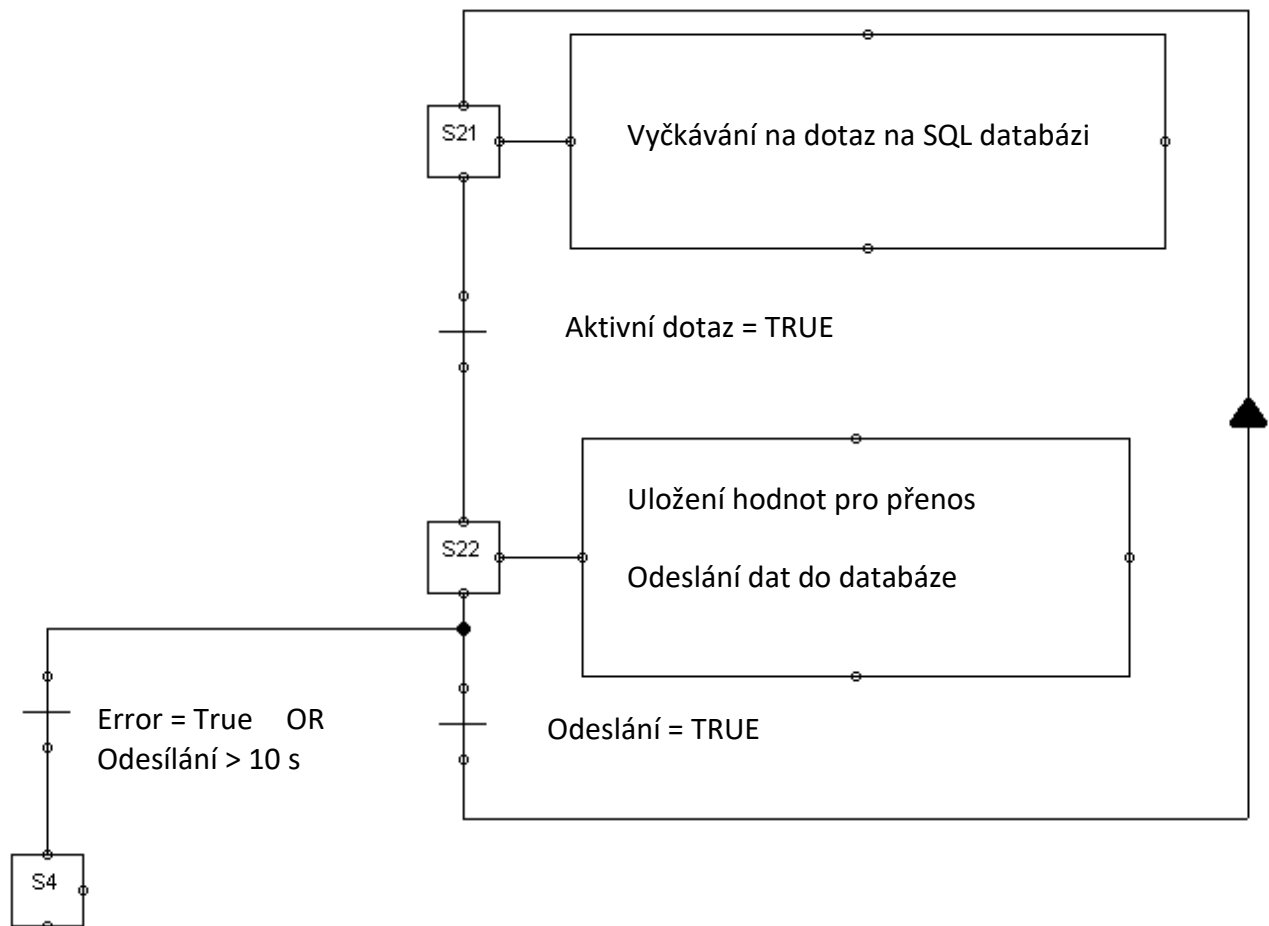
připojeno, vejde stavový stroj do stavů S2. V případě komplikací mohou nastat dvě další varianty. První je, že se v rámci stavu S12 nepodaří k databázi připojit během 20 sekund. V tom případě dojde k restartu databáze ve stavu S3. Zde byl restart v případě prvního připojování bezpečný, proto mohl proběhnout automaticky. Druhá varianta byla, že by se při větších problémech přesunula databáze



Obrázek 5.6 Podrobnější diagram přechodů stavu S1 pomocí SFC diagramu

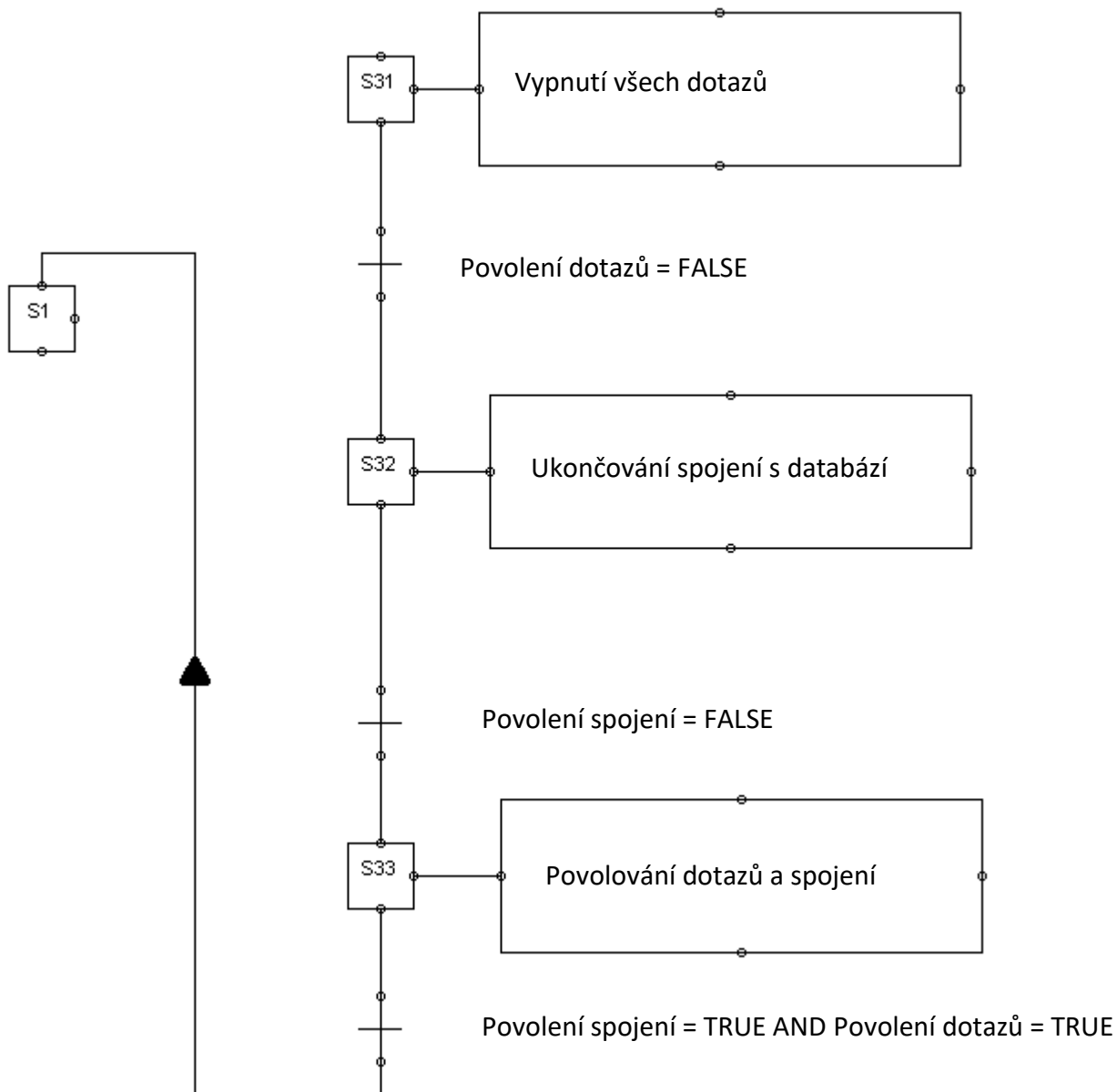
do stavu S4, kde by vyčkávala na ruční restart. Do tohoto stavu se bylo možné dostat buď při aktivaci chybové hlášky od Mapp komponenty, nebo při 5 neúspěšných pokusech o připojení.

V rámci stavu S2 (viz. obrázek 5.7) dochází ve stavu S21 k vyčkávání na povel k uložení (např. uložení vykonaného cyklu), nebo načtení informací (např. databáze dílů) z databáze. V rámci celkové struktury programu zde byl vytvořen i zásobník, který umožňuje uložit jednotlivé hodnoty pro zápis a ukládat je do databáze postupně. V rámci stavu S22 až N (stavech sloužících k vykonání jednotlivých dotazů) dochází k překopírování uložených hodnot do odesílané struktury. Ze stavu S22 se do stavu S21 dostane po potvrzení úspěšného odeslání či přijetí dat. Stav S22 se může ještě dále větvit v závislosti na povaze dotazu. Ze stavu S22 je možné se dostat do stavu S3. To nastane v okamžiku, když dotaz není vykonán do 10 sekund. V rámci omezení v důsledku chyby v kódu Mapp komponenty bylo nutné automaticky přejít do stavu S4 a vyčkat na ruční restart. Totéž se stane i při zaznamenání chybové hlášky ze strany Mapp komponenty.



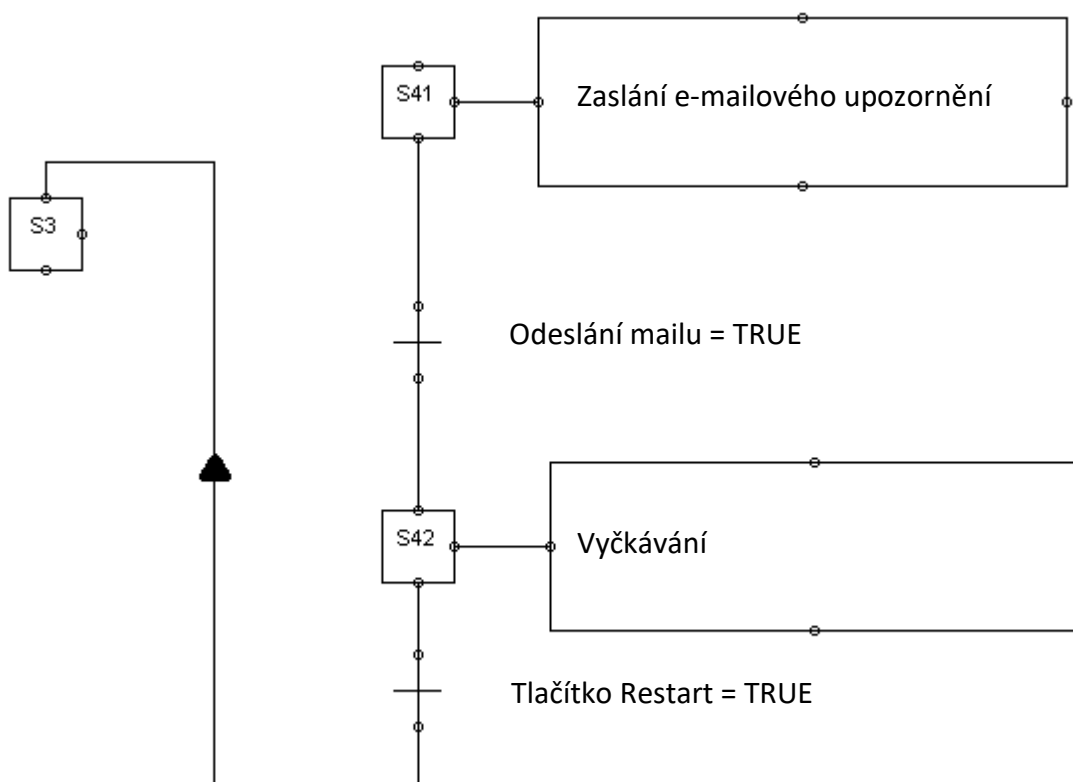
Obrázek 5.7 Podrobnější diagram přechodů stavu S2 pomocí SFC diagramu

Stav S3 zajišťuje restart databáze (viz. obrázek 5.8). V prvním kroku S31 dojde k vypnutí všech dotazů. Tento krok vychází z principu programování Mapp Database. Tato komponenta obsahuje několik ucelených funkčních bloků. Jedním z nich je funkční blok pro připojování, o kterém již zde bylo mluveno. Pro každý vykonávaný dotaz se používá další, samostatně fungující funkční blok. Parametry funkčního bloku se konfiguruje v první řadě v samotném programu. Další konfigurace se dělá v konfiguračním souboru, kde se vytváří dotaz pomocí jazyka SQL. Pro kompletní restart je tudíž zapotřebí restartovat nejen samotné připojení, ale i všechny dotazy. Za předpokladu, že by nebyly problémy s kódem Mapp Database, by bylo možné tento krok udělat jinak (vypnout vždy samostatný dotaz). Ale toto řešení se ukázalo jako efektivní a přehlednější. V kroku S32 dojde k povelu restartu databáze. Do stavu S33 se stavový stroj dostane v okamžiku, kdy je potvrzené odpojení od databáze. Ve stavu S33 dojde k procesu povolování využívání funkčních bloků jednotlivých dotazů a přejde se do stavu S1, kdy se začne obnovovat spojení.



Obrázek 5.8 Podrobnější diagram přechodů stavu S3 pomocí SFC diagramu

Stav S4 nastává v okamžiku, kdy program nedokáže sám obnovit spojení a je zapotřebí zásahu zvenčí. V rámci stavu S41 (viz. obrázek 5.9) dojde k zaslání informace o chybě formou e-mailu a ve stavu S42 vyčkává na restart. Při stisknutí tlačítka ve vizualizaci od pověřené osoby dojde k přechodu na stav S3, kdy dojde k restartu a následnému připojení.



Obrázek 5.9 Podrobnější diagram přechodů stavu S4 pomocí SFC diagramu

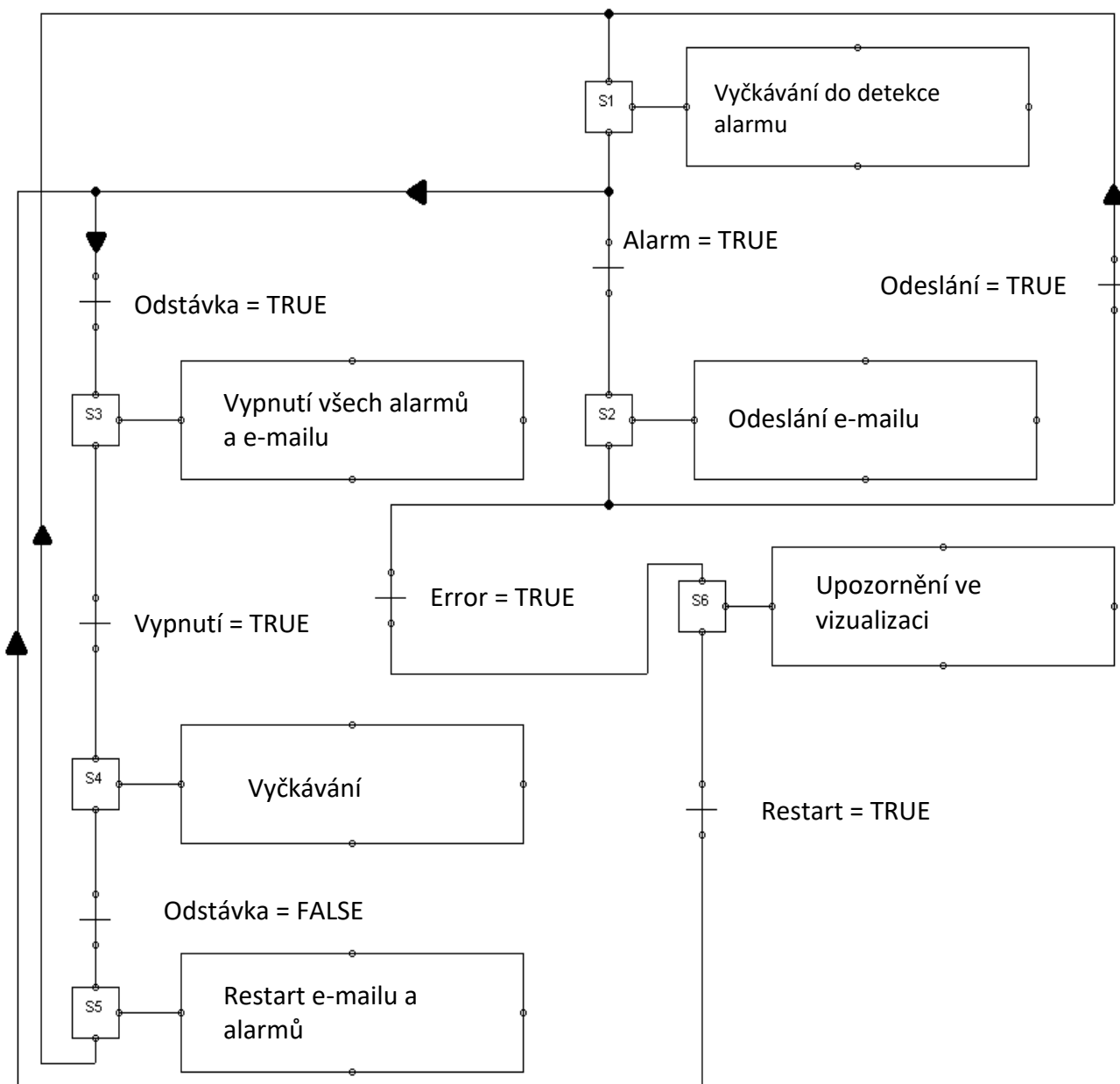
5.2.4 Zaslání e-mailového upozornění a alarmy

Tato část programu kombinovala stavový stroj (viz. obrázek 5.10), Mapp komponenty Mapp AlarmX a Mapp Alarm History. V rámci Mapp AlarmX bylo možné definovat alarmy, které využívaly proměnné z celého programu. Zde bylo možné nadefinovat několik možných typů alarmu. Např. překročení hodnoty, velký nárůst hodnot atd. Každému alarmu bylo možné přiřadit číslo, jméno a úroveň.

Stavový stroj e-mailu obsahoval stavy Vyčkávání a detekce alarmu (S1), Odeslání a kontrola odeslání e-mailu (S2), Vypnutí alarmů (S3), Vyčkávání ve vypnutém stavu (S4), Restart alarmů (S5) a Error e-mailu (S6).

Ve stavu S1 stavový stroj vyčkával do vyvolání okamžiku, kdy se spustí Mapp AlarmX. V tu chvíli dojde k detekci chybové hlášky. Podle chybové hlášky je sestavena hlavička posílaného e-mailu. Ta se skládá z čísla stroje a názvu alarmu. Dále se vybere e-mailová adresa. Jelikož byl problém jednoduše vytvořit způsob, který by e-mail odeslal více lidem najednou, bylo vytvořeno několik skupin e-mailových adres, které se následně rozeslaly všem označeným lidem. V rámci sestavování hlavičky byl alarm deaktivován. Každý e-mailové upozornění bylo možné vypnout v rámci vizualizace. Pokud by alarm nebyl povolen ve vizualizaci, stavový stroj by se opět vrátil do stavu S1 s deaktivovanou chybovou hláškou. V případě, že alarm byl povolen, přešel stavový stroj do stavu S2. Pokud byla detekována ve stavu S1 proměnná „Odstávka e-mailu“, přešel stavový stroj do stavu S3.

Ve stavu S2 došlo k povelu odeslání e-mailu. V případě potvrzení odeslání došlo k zápisu alarmu do Mapp Alarm History a přechodu do stavu S1. Pokud by byla detekována chyba odeslání, přešel by mail do stavu S6.

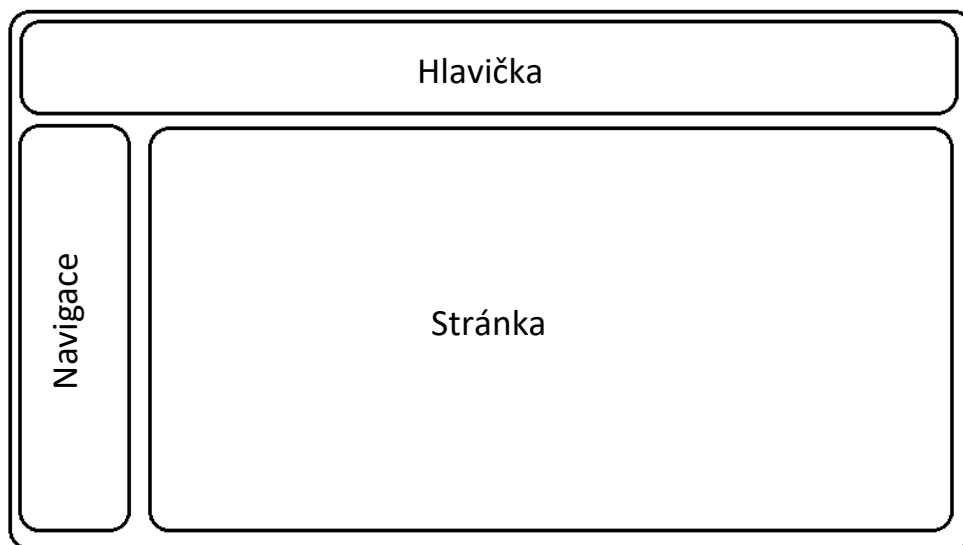


Obrázek 5.10 Znárodnění stavového stroje e-mailového upozornění a alarmů pomocí SFC diagramu

Ve stavu S3 došlo k vypnutí Mapp AlarmX. Po potvrzení vypnutí přešel stavový stroj do stavu S4, kde vyčkával na „Vypnutí signálu“ a „Odstávka e-mailu“. V okamžiku, kdy byl signál deaktivován, došlo k přechodu do stavu S5. Zde byl proveden restart Mapp AlarmX a po potvrzení této akce se stavový stroj vrátil do stavu S1. V případě, že došlo k problémům během odesílání e-mailu a přechodu do stavu S6, bylo nutné ručně restartovat e-mail. Následně pak došlo k přechodu do stavu S3, aby po restartu e-mailu nedošlo k hromadnému odeslání veškerých alarmů, které se v době poruchy e-mailu staly.

5.3 Popis jednotlivých částí vizualizace

Vizualizace měla za úkol v první řadě zobrazit aktuální stav výroby. Dále pak umožnit nadefinování současné výroby a zjistit historická data o výrobě. Okno bylo rozděleno do třech oblastí (viz. obrázek 5.11). Hlavička, navigace a aktuální stránka. V hlavičce byly například informace o stavu databáze (z důvodu výše popsaných problémů bylo dobré mít stav databáze stále na očích) a přihlašovací okénko, které se zobrazilo po rozkliknutí ikony. V navigaci byly zobrazeny veškeré stránky, které měl uživatel k nahlédnutí. Informace a použití jednotlivých stránek vizualizace zde budou podrobněji popsány.



Obrázek 5.11 Rozložení vizualizačního okna

5.3.1 Přehled výroby

Tato stránka byla načtena jako první po zapnutí databáze. Jednalo se o znázornění běhu všech připojených strojů. Každý stroj se mohl nacházet v jednom ze 4 režimů, které byly barevně a pomocí ikony znázorněny na stránce. V případě, že stroj byl vypnutý, byl řádek označen červeně. Během přípravy stroje a najíždění výroby se řádek zbarvil žlutě. Za běhu stroje byl řádek zelený. V případě, že se stroj zastavil během provozu, došlo k rozblíknání řádku a odeslání varovného e-mailu.

V každém řádku, který reprezentoval jednotlivé stroje, byl souhrn nejdůležitějších informací. Jako první se zde nacházela informace o formě, která se nacházela na stroji. Dále zde byly informace o parametrech aktuálního cyklu (Pořadí dílu v aktuální směně, čas cyklu, teplota prvního pásma a polštář). Tyto hodnoty platily pro poslední cyklus, který stroj vykonal. Tyto informace byly důležité z hlediska kontinuity výroby. Dále zde byla ikona, která zobrazovala stav materiálu v násypce. Jednalo se především o výstražnou informaci. V okamžiku, kdy materiál klesl pod danou úroveň, bylo nutné

materiál buď okamžitě doplnit, nebo stroj zastavit. V případě, že by výroba pokračovala dále a za běhu by materiál došel, bylo by nutné stroj zdlouhavě připravovat na další najetí výroby. Tuto hodnotu bylo ale potřeba umístit do krajní polohy z důvodu využívání různých materiálů. Některý materiál se totiž plní pouze v malém množství, jelikož by jinak docházelo k jeho vlnutí.

Další informace se vztahovaly k předchozí směně. Jednalo se o průměrný čas a počet vyrobených kusů za předchozí směnu. To umožňovalo si udělat rychlý přehled o aktuální rychlosti výroby. Poslední dvě informace se vztahovaly k času aktuálního cyklu. Byly zde zobrazeny dva časové údaje. Prvním byl požadovaný čas cyklu. Jednalo se o čas, kolem kterého se měla výroba pohybovat. Druhým časem byl maximální povolený čas. To znázorňovalo časovou hranici, kterou by výroba neměla překročit. Za určitého počtu překročených cyklů došlo k odeslání varovného e-mailu.

Tento náhled umožňoval na jednom místě zjistit celkový stav výroby v celé firmě. V budoucnu zde budou zobrazeny veškeré připojené stroje.

Číslo stroje	Vyráběný díl	Stav stroje	Počet cyklů za směnu	Akt. čas cyklu (s)	Akt. polštář	Akt. teplota 1. pásma	Akt. stav násypky	Prům. cyklus předchozí sm.	Počet cyklů předchozí sm.	Požadovaný cyklus	Maximální čas cyklu (s)
01-4500	Hubice.	■	76	146	33.2	104	✓	146.7	107	150	160
03-2100	Pouzdro SEMI ODG, S1, P bez potisu, zális...	▶	120	120	33.5	284	✓	0.0	0	130	140
04-2100	Monoblok VERA modrý-výztuž.	✘	191	105	8.1	214	✓	104.4	211	105	115
11-1000	Rám P10 COMPACT Černý - ROVNANÉ.	✘	118	152	44.5	302	⚠	153.3	113	155	165
12-1000	Rám P07 HUNGARY Černý + červená tečka.	▶	136	160	16.5	295	✓	159.6	120	160	170
13-0800	Hřbet P10 FULLSIZE -S- Černý.	■	243	45	41.5	31	⚠	53.2	436	50	60
14-2700	Kryt rámu vnitřní L.	✘	215	53	2.5	276	⚠	0.0	1	60	70
16-2700	Filter Inner Basic ECE WGN OBD RH	■	0	0	0.0	17	⚠	0.0	0	200	210
17-1600	Pouzdro spouštědla SEMI Černé, zálisek 15...	▶	245	85	14.0	274	✓	91.2	240	90	100
23-1000	Těsnění Respirátor	■	0	35	12.0	14	⚠	35.8	243	35	40

Obrázek 5.12 Vizualizace – Přehled výroby

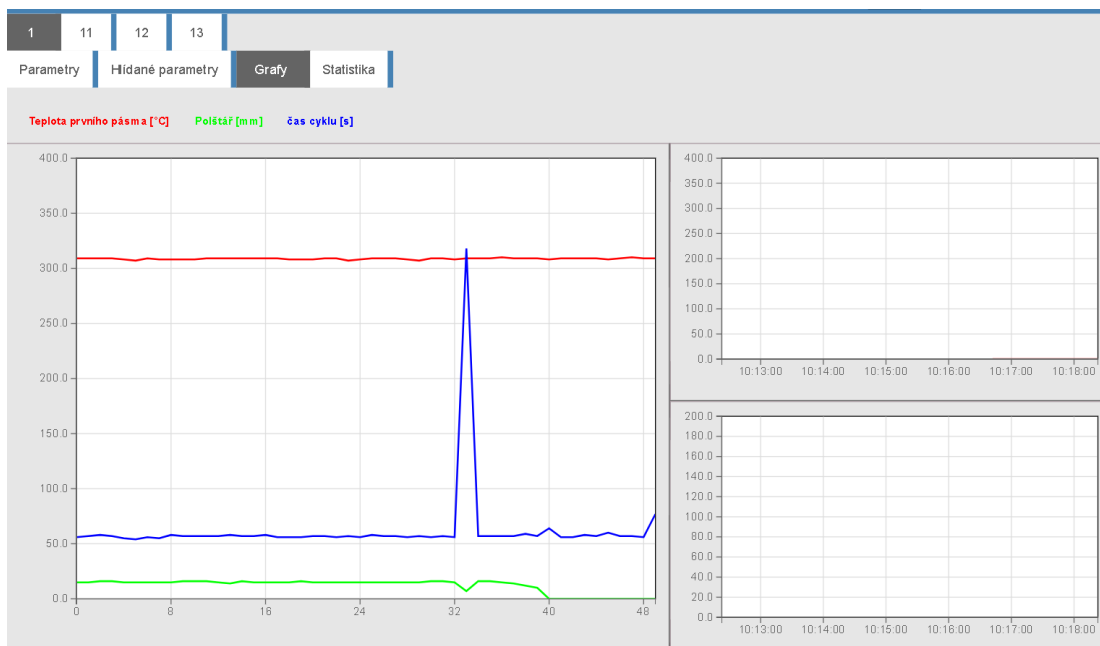
5.3.2 Parametry výroby stroje

Zatímco *Přehled výroby* sloužil k vytvoření komplexní představy o celé výrobě, na stránce *Stroje* byly zobrazeny veškeré dostupné informace o jednotlivých stojích v několika záložkách. Na první záložce bylo možné zjistit a v případě oprávnění i nastavit, veškeré hlídané limity výroby. Jednalo se o parametry času, teploty a polštáře. Bylo zde možné nastavit jak horní, tak i dolní hranici. U teploty a polštáře byly zobrazeny i OK hodnoty, aby bylo možné stanovit základní bod při nastavování výroby. Veškeré tyto hodnoty bylo možné načíst a uložit do databáze. Dále se zde nacházela počítadla směny a série. Zatímco počítadlo směny se nulovalo každých 8 hodin, počítadlo série se nulovalo pouze při změně dílu, nebo po stisknutí tlačítka. Toto počítadlo bylo vytvořeno pro přehled o celkovém počtu vyrobených kusů od zahájení výroby daného kusu. Na další záložce bylo možné vypnout nebo zapnout e-mailová upozornění na jednotlivé alarmy. Dále zde bylo možné nastavit dva číselné parametry pro odeslání e-mailu o překročení povoleného času. Jelikož nebylo žádoucí, aby se odesílal e-mail o každém překročení povoleného času, bylo možné nastavit jednak počet po sobě jdoucích překročených cyklů a následně i počet překročených cyklů během jedné hodiny. To umožňovalo zajistit jistou toleranci pro různě časově závislé díly.

Parametr	Hodnota
Jméno dílu	Hůbice
Kód	1060001
Zákazník	KEMPER spol.s.r.o.
Nastavení požadovaného OK cyklu (s)	150
Počet kusů vyrobených v aktuální směně	0
Nastavení OK hodnoty	0.0
Nastavení maximálního času (s) - Upozornění emailem	160
Průměrný čas předchozí směny	0.0
Horní OK hranice	0.0
E-mailové upozornění na alarmy	<input checked="" type="checkbox"/>
počet kusů od změny dílu nebo vynulování	848
Dolní OK hranice	0.0

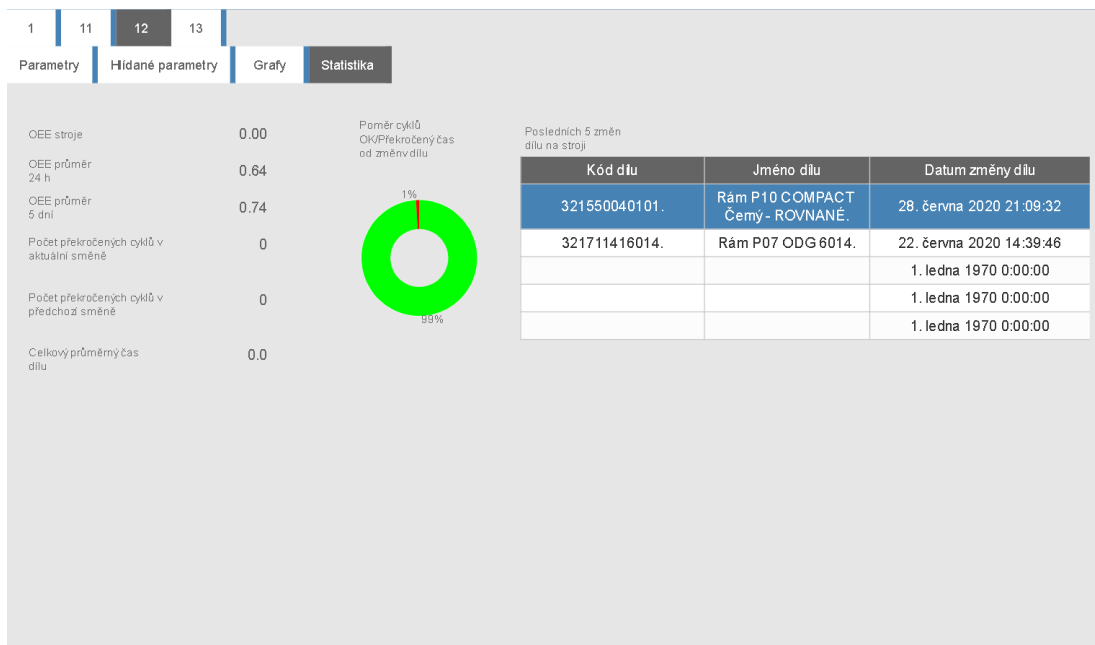
Obrázek 5.13 Vizualizace – Parametry

Další záložka byla věnována třem grafům. Na prvním grafu byly zobrazeny hodnoty teploty, polštáře a času u posledních 50 vyrobených kusů. Na dalších dvou grafech bylo možné sledovat aktuální hodnoty teploty a polohy šneku (polštáře).



Obrázek 5.15 Vizualizace – Grafy

Na poslední záložce bylo zobrazeno několik statistických údajů. Jednalo se o 3 výše zmíněné parametry OEE (aktuální, za 24 h a za 5 dní) a OEE z předchozí směny. Dále zde bylo možné vidět graficky a procentuálně zobrazený poměr mezi kusy vyrobených v požadovaném čase a kusech, které povolený čas překročily. Poslední informací zde byla tabulka, kde bylo zapsáno posledních 5 vyráběných dílů a času, kdy byl daný díl změněn.



Obrázek 5.14 Vizualizace – Statistika

5.3.3 Historie cyklů

Další částí vizualizace byla stránka Historie cyklů. Zde bylo možné poslat dotaz do databáze a pomocí tabulky vyobrazit historii vyráběných cyklů. Bylo zde možné vidět čas cyklu, teplotu, polštář, o jaký díl se jednalo, v kolik hodin byl díl vyroben a zda byla překročena maximální doba cyklu. Na jeden dotaz bylo zobrazeno vždy maximálně 100 cyklů. Pro možnost prohlédnutí starších cyklů byla vytvořena možnost listování.

Stroj:		kód	Datum/čas konkrétního cyklu	Pořadí cyklu	Čas cyklu (s)	Polštář (mm)	Teplota 1. pásma	Max. čas cyklu překročen
01-4500		00202218-06-0000.	1. července 2020 7:54:38	115	77	0.3	309.0	1.0
Výpis posledních 100 cyklů		00202218-06-0000.	1. července 2020 7:52:23	113	57	0.3	310.0	0.0
0		00202218-06-0000.	1. července 2020 7:51:26	112	57	0.3	309.0	0.0
>>		00202218-06-0000.	1. července 2020 7:50:28	111	60	0.3	308.0	0.0
		00202218-06-0000.	1. července 2020 7:49:28	110	57	0.3	309.0	0.0
		00202218-06-0000.	1. července 2020 7:48:31	109	58	0.3	309.0	0.0
		00202218-06-0000.	1. července 2020 7:47:33	108	56	0.3	309.0	0.0
		00202218-06-0000.	1. července 2020 7:46:36	107	56	0.4	309.0	0.0
		00202218-06-0000.	1. července 2020 7:45:40	106	64	10.2	308.0	0.0
		00202218-06-0000.	1. července 2020 7:44:36	105	57	12.2	309.0	0.0
		00202218-06-0000.	1. července 2020 7:43:38	104	59	14.0	309.0	0.0
		00202218-06-0000.	1. července 2020 7:42:39	103	57	14.6	309.0	0.0
		00202218-06-0000.	1. července 2020 7:41:42	102	57	15.6	310.0	0.0
		00202218-06-0000.	1. července 2020 7:40:45	101	57	15.6	309.0	0.0
		00202218-06-0000.	1. července 2020 7:39:47	100	57	7.3	309.0	0.0
		00202218-06-0000.	1. července 2020 7:38:50	99	318	15.2	309.0	1.0
		00202218-06-0000.	1. července 2020 7:33:31	98	56	15.7	308.0	0.0
		00202218-06-0000.	1. července 2020 7:32:35	97	57	15.6	309.0	0.0
		00202218-06-0000.	1. července 2020 7:31:38	96	56	15.1	309.0	0.0
		00202218-06-0000.	1. července 2020 7:30:42	95	57	14.8	307.0	0.0

Obrázek 5.16 Vizualizace – Historie cyklů

5.3.4 Výběr dílu

Na této záložce bylo možné vyhledat díl v databázi. Po výběru konkrétního dílu se veškeré informace uložily do paměti PLC a v okamžiku vyrobení dílu bylo jasně dáno, o který díl se jednalo. Pro usnadnění vyhledávání byly vytvořeny dva pomocné vyhledávací prvky. Prvním byl výběr skladu, do kterého se díly posílají zákazníkovi. Druhou možností upřesnění výběru bylo textové pole, do kterého bylo možné napsat část názvu nebo kódu dílu. V databázi se následně našla shoda obou kritérií. Po výběru dílu bylo zapotřebí vybrat stroj a následně uložit změny.

V rámci *Výběru dílu* bylo možné načíst i parametry výroby. V první fázi byly v databázi uloženy informace o požadované a maximální délce cyklu. Tabulka byla připravena pro budoucí zadání informací o požadované teplotě a polštáři.

Jméno dílu	Kód	OK cyklus	Max cyklus	OK teplota	Max teplota	Min teplota	OK polštář	Max polštář	Min polštář
Tlačítka LS/PS/LH/LS OBU5050.	TX000000338,339,340,341	64.0	74.0	0.0	0.0	0.0	0.0	0.0	0.0
Nosič tlačítka OBU5050/OBU5051.	TX000000342.	24.0	38.0	0.0	0.0	0.0	0.0	0.0	0.0
Tlačítko LH - šipka dolů OBU5051.	TX000000467.	34.0	44.0	0.0	0.0	0.0	0.0	0.0	0.0
Tlačítko LS - křížek OBU5051.	TX000000468.	34.0	44.0	0.0	0.0	0.0	0.0	0.0	0.0
Tlačítko PH - šipka nahoru OBU5051.	TX000000469.	34.0	44.0	0.0	0.0	0.0	0.0	0.0	0.0
Tlačítko PS - enter OBU5051.	TX000000470.	34.0	44.0	0.0	0.0	0.0	0.0	0.0	0.0

Obrázek 5.17 Vizualizace – Změna dílu

5.3.5 Údržba

Tato jednoduchá vizualizace měla za úkol pomoci pracovníkům údržby při napojování dalších strojů a kontrole funkčnosti jednotlivých senzorů. Na stránce bylo možné zadat číslo stroje. Následně se zobrazily veškeré hodnoty čidel, které jsou na I/O ostrůvek připojeny.

5.4 Uživatelské přístupy

V rámci systému bylo vytvořeno několik uživatelských přístupů. V této části bude krátce shrnuto práva a povinnosti jednotlivých zainteresovaných pracovníků, kteří se systémem pro sběr dat přicházel do kontaktu. Omezení jednotlivých práv a přístupů bylo vytvořeno na úrovni vizualizace a komunikace OPC UA. Pomocí jednotlivých nástrojů bylo rozlišeno, zda uživatel se svým přístupovým právem může jednotlivé prvky vidět a následně i používat. Pro přehlednější rozlišení, zda má uživatel možnost prvek pouze vidět, nebo i použít, bylo využito změny odstínu prvku. Například čísla, která se nemohla měnit, měla šedivý odstín.

5.4.1 Přístup bez přihlášení

Uživatel, který nebyl přihlášen, měl k dispozici *Přehled výroby a Historii cyklů*. To znamenalo, že bez přihlášení nebylo možné nic upravovat ani ukládat. Tyto hodnoty se zobrazovaly i všem ostatním uživatelským přístupům.

5.4.2 Kontrola kvality

Po přihlášení osoby s rolí kontrola kvality byla navíc zpřístupněna stránka *Parametry strojů a Vyhledávání dílu*. Pracovník s touto pozicí má ve firmě za úkol uvolňování a kontrolu výroby. Pro tento účel mu bylo umožněno měnit díly na stroji v okamžiku, kdy byla výroba uvolněna. Tento pracovník měl za úkol potvrdit uvolnění do výroby tím, že stiskl tlačítko pro sériovou výrobu. Jelikož se jednalo pouze o kontrolní pozici, která neměla bližší povědomí o samotné technologii, nebylo s tímto přihlášením do vizualizace možné měnit hodnoty parametrů na stroji. V případě potřeby ovšem bylo možné vypnout veškerá e-mailová upozornění, nebo jejich dílčí kontroly jednotlivých alarmů.

5.4.3 Mistr směny

Uživatel s těmito právy měl stejně jako kontrola kvality možnost měnit díly na stroji. Oproti kontrole měl ovšem též možnost změnit časové hodnoty výroby, jelikož se tato hodnota může měnit se změnou technologie nebo materiálu. S tímto uživatelským přístupem nebylo ovšem možné vypnout kontrolní hlášení na e-mail. Tento způsob provedení práv měl za následek vzájemnou kontrolu mezi mistrem směny a kontrolou kvality. Obě tyto pozice byly přítomny na všech 3 směnách.

5.4.4 Hlavní technolog

Pracovník v pozici hlavního technologa měl za úkol v rámci firmy nastavovat parametry na stroji hlavně na začátku nové výroby. Následně pak povinnost nastavování a najíždění stroje bylo přesunuto pod kompetenci mistrů směny. Z podstaty této pozice měla osoba s tímto uživatelským přístupem možnosti stejné, jako kontrola kvality a mistr směny. K těmto právům bylo navíc možné nově zvolené hodnoty uložit do databáze. Pokud by změnil mistr směny například hodnotu maximálního povoleného času cyklu, projevila by se tato změna při aktuální výrobě. Při novém načtení z databáze se ovšem tato hodnota vrátila na původní hodnotu. Na rozdíl od tohoto mohl hlavní technolog uložit hodnoty natrvalo.

5.4.5 Administrátor

Tento uživatel měl přístup ke všem prvkům ve vizualizaci. V rámci tohoto přístupu byla zpřístupněna dodatečná stránka *Diagnostika*. Na této stránce se mohl dozvědět například stav databáze a měl možnost pomocí tlačítka provést její restart. Dále mohl stejným způsobem pracovat i s částí programu obsluhující odesílání e-mailů a částí ukládající historii alarmů.

5.4.6 Údržba

Pro pracovníky údržby byly po přihlášení k dispozici stánky, na které nebylo potřeba žádné přihlášení, a stránka *Údržba*, kde bylo možné sledovat hodnoty jednotlivých senzorů. V této fázi bylo

vedením rozhodnuto, že tito pracovníci nebudou mít přístup k ovládní výměny dílu na stroji, byť právě oni byli zodpovědní za výměnu forem na stroji.

	Bez přihlášení	Kontrola kvality	Mistr směny	Hlavní technolog	Administrátor	Údržba
Přehled výroby	X	X	X	X	X	X
Historie cyklů	X	X	X	X	X	X
Parametry strojů		X	X	X	X	
Vyhledávání dílu		X	X	X	X	
Změna dílu		X	X	X	X	
Změna parametrů			X	X	X	
Vypnutí e-mailu		X		X	X	
Ukládání parametrů do databáze				X	X	
Diagnostika					X	
Údržba					X	X

Tabulka 5.1 Přehled uživatelských práv

6 Práce a využití SQL databáze

SQL databáze byla využita k ukládání výrobních dat, tabulky dílů a strojů, parametrů výroby a statistických údajů. V rešerši na téma ukládání dat byla vybrána databáze My SQL, která splňovala požadovaná kritéria. Databáze byla nainstalovaná na samostatný server, který byl umístěn do firemní sítě. Pro práci s databází byla využita komponenta Mapp Database. Zde byly nakonfigurovány veškeré dotazy, které byly volány v programu PLC. Pro práci s databází bylo zapotřebí vytvořit struktury, jejíž proměnné měly stejné názvy, jako sloupce v jednotlivých tabulkách. Do těchto struktur byla následně načtena data, která byla poslána z databáze.

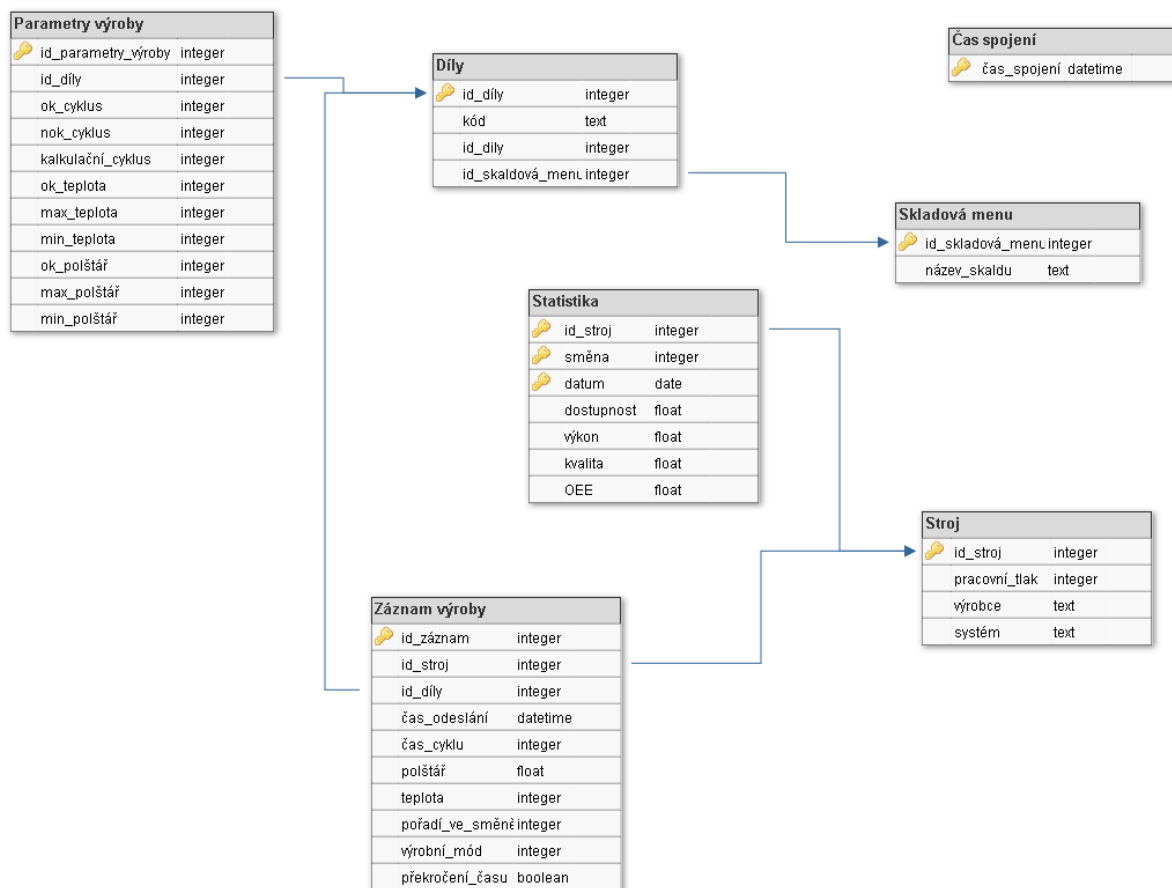
Hlavními úkoly databáze bylo ukládání dat a jejich následné filtrování na základě zvolených parametrů. Do databáze byly ukládány veškeré informace o každém zdvihu. Na základě těchto údajů bylo možné například spočítat, kolik se daného dílu vyrobilo za dané období, nebo jaký byl průměrný čas výroby různých dílů. Pro získávání těchto informací byl využit databázový klient HeidiSQL, ve kterém bylo možné formulovat dotazy a prohlížet jednotlivá data. Důvodem pro filtrování dat v rámci HeidiSQL a ne vizualizace bylo, že vedení firmy v první fázi neumělo definovat, jaká data ho konkrétně budou zajímat. To by se ale mělo do budoucna změnit.

6.1 Struktura databáze

Struktura databáze je graficky znázorněna na obrázku 6.1. Do databáze bylo zapotřebí v první fázi nahrát aktuální data o vyráběných dílech. V tabulce dílů byly uloženy informace o kódu dílu, jeho názvu a skladu, do kterého se díl posílá. Na informace o skladech byla vytvořena další tabulka s názvy a informacemi o všech skladech a pomocí ID spojena s jednotlivými díly. Ke každému dílu byly v samostatné tabulce přiřazeny parametry výroby. V této fázi by bylo možné tyto informace převést na jednu tabulku, jelikož aktuální stav vypadal tak, že ke každému dílu byl přiřazen jeden řádek s výrobními parametry svázané pomocí ID. To by se ovšem změnilo, pokud by se přidala i informace o materiálu a výsledek by byl takový, že k jednomu dílu bude přiřazeno více řádků parametrů v závislosti na aktuálním materiálu. Proto bylo lepší tabulku rozdělit rovnou.

Dalšími uloženými parametry byly informace o strojích. V těchto tabulkách byla data, která se příliš neměnila (až na občasné přidání nového dílu, či změny parametrů). Jednalo se o základní data, která sloužila především k definici konkrétní výroby.

V tabulkách parametry výroby, Statistika a Čas spojení docházelo k pravidelnému posílání nových dat v rámci běžného provozu výroby. Tabulka Parametry výroby sloužila k zapisování jednotlivých cyklů a pomocí ID byla svázaná s konkrétním dílem a strojem. Do této tabulky přicházela data i v jednotkách sekund. Tabulka Statistika se měnila každých 8 hodin při změně směny. Tyto informace sloužily pro zhodnocení efektivity výroby. Poslední tabulkou byl čas spojení, který sloužil pouze pro udržení otevřeného spojení s databází. Do této tabulky se každou hodinu uložil aktuální čas.



Obrázek 6.1 Struktura databáze

7 Aplikace komunikace OPC UA

Jelikož bylo možné pro vizualizaci a program PLC využít jednoho integrovaného programu, bylo možné pracovat s OPC UA velice jednoduše. Aplikace Automation Studio měla pro tento účel integrované jednoduché nástroje pro správu přenosu dat i uživatelských přístupů a rolí.

Jako první bylo zapotřebí definovat a vytvořit role, které byly popsány v kapitole 5.4. V prvním kroku při konfiguraci OPC UA byla nadefinována základní práva u jednotlivých rolí. Tyto defaultní hodnoty bylo následně možné pozměnit u každé jednotlivé hodnoty.

Dalším krokem bylo označení veškerých proměnných, které se pro komunikaci budou používat. Jednotlivé hodnoty byly nakonfigurovány a byla jim přiřazena práva přístupu uživatelů a v případě potřeby i jednotky. Tam, kde to bylo potřeba, byly nastaveny limity (například maximální možná nastavitelná hodnota času cyklu). U OPC UA byla výhoda, že bylo možné jednoduše definovat různé rozpětí hodnot pro různé role uživatelů.

Posledním krokem bylo propojení jednotlivých komponent ve vizualizaci s jednotlivými hodnotami. Zde bylo možno definovat, zdali se jednalo o komunikaci obousměrnou, nebo jednosměrnou. Po provedení těchto kroků si program sám vytvořil komunikační kanál. K vizualizaci bylo následně možné pomocí IP adresy PLC přistoupit z jakéhokoliv zařízení v rámci firemní sítě.

V této konfiguraci plnila vizualizace při komunikaci přes OPC UA roli klienta a PLC roli serveru. V případě, že by se stroj napojoval k PLC pomocí přímé komunikace bez využití prostředníka v podobě I/O ostrůvku a byl by stroj vybaven kartou pro tuto komunikaci, mohlo PLC plnit i roli klienta.

8 Analýza a využití získaných dat

V době, kdy byl zaveden systém do provozu, dopadla na firmu Plast Form Service i.m. s.r.o. koronavirová krize. To mělo za následek velké utlumení výroby, jelikož ve firmě v té době činil podíl výroby pro automobil cca 50 %, který byl zasažen nejvíce. Tento sektor byl na dlouhé období utlumen natolik, že firma mohla být v provozu jen na část týdne a vždy bylo v provozu pouze malé množství strojů. Jeden z důsledků bylo, že napojené stroje byly dlouhou dobu mimo provoz, a tudíž nemohly sbírat data. Největší množství dat bylo nasbíráno ve 30. týdnu roku 2020. Jednalo se o týden před celozávodním volnem. Zde bylo možné pozorovat běžný provoz připojených strojů. Pro příklad analýzy parametru OEE byl vybrán stroj 12-1000 od firmy Battenfeld. Na tomto stroji byly vyráběny v průběhu týdne 2 různé díly. Jednalo se o rámy na dva typy pistolí vyráběné pro firmu Česká zbrojovka a.s. Stroj měl potenciál vyrábět celý týden. První díl měl požadovaný čas na cyklus 50 s a druhý díl 160 s. Během týdne došlo k výměně formy.

Díky zápisu informací do databáze bylo zjištěno, že průměrný čas prvního dílu byl 50,86 s a druhého 160,2 s. Jedná se o aritmetický průměr všech cyklů, které v průběhu týdne proběhly. Na konci každé směny byly do databáze uloženy údaje o výkonu, dostupnosti a OEE stroje. Jak již bylo zmíněno (viz kapitola 4.4) v aktuální době nebyl řešen zápis počtu NOK dílů pomocí systému, ale pomocí papírových zápisů. Počty vyrobených dílů a počty NOK a OK dílů je možné vidět v tabulce 8.1

id_stroj	Směna	Datum	Vyrobena	NOK	OK
12	1	20.07.2020	517	32	485
12	2	20.07.2020	507	17	490
12	3	20.07.2020	421	28	393
12	1	21.07.2020	166	5	161
12	2	21.07.2020	6	0	6
12	3	21.07.2020	94	15	79
12	1	22.07.2020	124	12	112
12	2	22.07.2020	179	4	175
12	3	22.07.2020	138	24	114
12	1	23.07.2020	170	16	154
12	2	23.07.2020	0	0	0
12	3	23.07.2020	122	12	110
12	1	24.07.2020	87	7	80
12	2	24.07.2020	0	0	0
12	3	24.07.2020	0	0	0

Tabulka 8.1 Počet vyrobených kusů

Z těchto údajů byla dopočítána hodnota kvality. Veškeré údaje byly vloženy do tabulky 8.2 a správná hodnota OEE byla vypočítána pomocí rovnice 1.

id_stroj	Směna	Datum	Dostupnost	Výkon	Kvalita	OEE
12	1	20.07.2020	0,997479	0,91965	0,938104	0,86056
12	2	20.07.2020	0,872185	0,96939	0,966469	0,81714
12	3	20.07.2020	0,760802	0,99528	0,933492	0,70685
12	1	21.07.2020	0,999986	1,0635	0,96988	1,03145
12	2	21.07.2020	0,018335	1,20346	1	0,02207
12	3	21.07.2020	0,540893	0,99548	0,840426	0,45253
12	1	22.07.2020	0,734918	0,98508	0,903226	0,65389
12	2	22.07.2020	0,9894	1,20306	0,977654	1,16371
12	3	22.07.2020	0,895674	1,11257	0,826087	0,8232
12	1	23.07.2020	0,965797	1,04576	0,905882	0,91493
12	2	23.07.2020	0	0	0	0
12	3	23.07.2020	0,753581	0,99025	0,901639	0,67283
12	1	24.07.2020	0,318711	1,19303	0,91954	0,34964
12	2	24.07.2020	0	0	0	0
12	3	24.07.2020	0	0	0	0

Tabulka 8.2 Hodnoty OEE

V tabulce 8.3 je zobrazena hodnota OEE pro jednotlivé směny a celková hodnota OEE, která charakterizuje míru využití stroje během týdne.

Směna 1	0,7620949
Směna 2	0,400583
Směna 3	0,5310801
Celkem	0,564586

Tabulka 8.3 Hodnoty OEE pro jednotlivé směny a celková hodnota OEE v rámci týdne

Z naměřených dat lze usoudit, že nejlepší směna byla od 6:00 do 14:00. Jedním z důvodů také je, že během první směny byl stroj vždy alespoň chvíli v provozu. V ostatních směnách došlo k zastavení stroje na celou směnu. Celková hodnota OEE udává, že potenciál stroje byl v průběhu týdne využit přibližně na 56 %. Dalším zajímavým údajem je, že hodnota výkonu dosahovala v některých směnách hodnoty vyšší, než 1. To znamená, že během směny byl průměrný čas cyklu nižší, než bylo požadováno.

9 Plány na rozšíření

Na počátku psaní praktické části diplomové práce byly připojeny na systém dva stroje. Při dokončení této práce již bylo napojeno strojů deset. V blízké době je předpoklad, že se připojí i zbylé stroje. V této fázi je aktuálně v plánu vyčkat a zjistit, co systém ve skutečnosti firmě přináší jak po stránce produktivity, tak financí. Celý systém je vytvořen tak, aby ho bylo možné do budoucna dále rozšiřovat. Plány na rozšíření je možné rozdělit do 3 částí. Software, hardware a integrace do podnikové informační struktury.

U softwaru byl předpoklad, že další rozšíření by se týkalo především zvýšení informačního významu pro potřeby údržby. Ve firmě byly vedeny záznamy o stavu forem pomocí papírové dokumentace. Zde bylo zapotřebí ručně evidovat počet provedených cyklů na každé formě a počet výrobních hodin. Tyto informace bylo možné vyhodnotit z aktuálního sběru dat. Jedinou překážkou byla nedokončená databáze o stavu forem ze strany firmy. Ovšem pro efektivní využití těchto informací by bylo zapotřebí též vytvoření jasného plánu údržby jednotlivých forem. Další, co se v rámci údržby nabízelo a bylo možné jednoduše aplikovat, byla kontrola počtu zdvihů stroje a plánování jejich údržby.

Další možností integrace systému by byla implementace evidence rozdělaných zakázek. Firma fungovala na papírové výrobní příkazy, ze kterých následně byly přepsány informace do počítače. Tento mezikrok by bylo možné jednoduše odstranit. Posledním plánovaným vylepšením bylo vytváření pravidelných reportů o evidenci výroby.

Po stránce hardwaru bylo možné rozšířit systém několika směry. Prvním bylo napojení nejen strojů, ale i dalších zařízení. Konkrétně se jednalo o kontrolu množství materiálu v sušičkách granulátu. To by umožnilo upozornit na docházející předpřipravený materiál, který se následně sype do stroje. Mezi dalšími plány bylo umístit ke každému stroji dotykový panel. Zde by bylo možné zobrazovat například dokumenty o balení dílů, nebo soupis častých vad dílu. Vše by bylo k dispozici obsluze. Dalším přínosem by bylo, že obsluha stroje by mohla okamžitě zaznamenávat informace o NOK dílu. Díky těmto informacím by se mohly lépe hledat a analyzovat chyby v toku materiálu a kontrole výroby. Další velké rozšíření by se mohlo ubírat cestou kontroly montáže dílů a skladu.

Posledním velkým plánem bylo přeorganizování struktury informačních systémů. V aktuální podobě byl systém absolutně oddělen od podnikového ERP systému. To sebou přinášelo několik úskalí. Jedním z největších se ukázalo vedení dvojí databáze. To by se ve velké míře projevilo až při nárůstu počtu funkcí systému. Zatím bylo zapotřebí pouze duplikovat informace o nových dílech a zákaznících, což neprobíhalo často. Další myšlenkou bylo spojit databázi podnikového ERP systému a systému pro sběr dat. Pro tento účel by bylo možné vytvořit tří vrstvou architekturu, kdy by se mezi aplikaci a databázi vložila vrstva aplikačního serveru, který by zprostředkoval a upravoval informace z databáze. To by mělo za následek zjednodušení práce s daty a větší variabilitu při výměně

jednotlivých druhů softwarů. Dále by bylo možné využít více menších aplikací, které by si data vzájemně vyměňovala. Například již zmíněný skladový systém.

10 Závěr

V rámci diplomové práce byl navrhnout a vytvořen systém, který ve firmě Plast Form Service i.m. s.r.o. sbíral data o výrobě ze vstřikovacích strojů. Na tento systém bylo v červenci roku 2020 napojeno a odzkoušeno deset strojů. Provoz těchto strojů byl zobrazen v kanceláři hlavního technologa firmy. Díky vizualizaci bylo možné nastavit parametry výroby, jako například délku cyklu, či různá alarmová upozornění. Pro práci s vizualizací byl vytvořen systém rolí, které byly přiřazeny jednotlivým uživatelům. Každá role měla ve vizualizaci svá práva. Veškeré alarmy byly odesílány jako upozornění na e-mail příslušným pracovníkům. Vizualizace komunikovala s PLC pomocí OPC UA.

Veškeré hodnoty o výrobě byly ukládány do databáze MySQL. V databázi byly uloženy parametry výroby a technologické hodnoty u každého provedeného cyklu. Dále byly do databáze ukládány statistické údaje, ze kterých následně probíhala analýza efektivnosti využití strojů.

Výsledný systém měl za následek lepší možnost dohledu nad výrobou. Tento efekt se projevil na zvýšení stability výroby a zlepšení možností hodnocení jednotlivých zaměstnanců a směn. Podle vyjádření vedoucího pracovníka firmy Plast Form Service i.m. s.r.o. se u připojených strojů téměř okamžitě zvýšila rychlost výroby a tím se ušetřila půl hodina výroby na každé směně. Za jeden výrobní týden tak byla získána jedna výrobní směna navíc.

Při analýze získaných dat bylo zjištěno, že výroba by mohla být ještě mnohem efektivnější. Částečně byl tento fakt ovlivněn koronavirovou krizí, která měla za následek snížení výroby. V rámci 30. týdne roku 2020 bylo ovšem možné pozorovat a zhodnotit provoz jednoho stroje, který byl na systém napojen. Zde byl vypočítán týdenní parametr OEE o hodnotě 0,565 po zaokrouhlení. Pokud budeme uvažovat v praxi maximální možnou hodnotu OEE 0,8 vyplývá nám, že stroj by v rámci týdne mohl být využit o necelých 30 % efektivněji.

11 Seznam literatury a informačních zdrojů

- [1] CHMELÍK, Bc. Vladimír. *Připojení PLC k vstřikovacímu stroji*. Praha, 2019.
- [2] CHMELÍK, Bc. Vladimír. *Finanční kalkulace projektu sběru dat pro firmu PFS Lysá nad Labem*. Praha, 2019.
- [3] SMUTNÝ, Lubomír. *Základy SQL. Sylaby, elektronické učebnice* [online]. Ostrava: VŠB – Technická univerzita Ostrava, 2006 [cit. 2020-02-15]. Dostupné z: <http://books.fs.vsb.cz/SQLReference/Sadovski/SQL-PRVN.HTM>
- [4] Ranking definition. *Db-engines* [online]. Vídeň: solid IT, c 2020 [cit. 2020-08-04]. Dostupné z: https://db-engines.com/en/ranking_definition
- [5] Ranking trend. *Db-engines* [online]. Vídeň: solid IT, c 2020 [cit. 2020-08-04]. Dostupné z: https://db-engines.com/en/ranking_trend
- [6] PLACHÝ, Ing. Ladislav. Optimalizace výrobních postupů pomocí SCADA WinCC. *Automa*. 2017, **2017**(10), 14-15.
- [7] Nové funkce systému SQL Server 2019. *Microsoft* [online]. Praha: Microsoft, c 2020 [cit. 2020-08-04]. Dostupné z: <https://www.microsoft.com/cs-cz/sql-server/sql-server-2019-features#ft1>
- [8] SQL Server Express je zdarma, ale *Michal Zobec: Blog // ZOBEC Consulting* [online]. Praha: ZOBEC Consulting, 2015 [cit. 2020-08-04]. Dostupné z: <https://www.michalzobec.cz/sql-server-express-je-zdarma-ale-4121>
- [9] ZEDNÍČEK, Ing. Jan. Instalace SQL Server 2016 – Plánování, HW a SW požadavky. *Biportal.cz* [online]. Praha: biportal, 2017 [cit. 2020-08-04]. Dostupné z: <https://biportal.cz/instalace-sql-server/>
- [10] Jak licencovat SQL Server. *Microsoft* [online]. Praha: Microsoft, c 2020 [cit. 2020-08-04]. Dostupné z: https://www.microsoft.com/cs-cz/sql-server/sql-server-2019-pricing#CP_StickyNav_1
- [11] PETŘÍK, Tomáš. *Volba databázového systému na základě požadavků projektu IS/ICT*. Praha, 2008. Bakalářská Práce. Vysoká škola ekonomická v Praze, Fakulta informatiky a statistiky.
- [12] FIALA, Adam. *Porovnání vybraných databázových systémů*. Praha, 2017. Bakalářská práce. Vysoká škola ekonomická v Praze, Fakulta informatiky a statistiky.
- [13] FIKAR, Jan. Vyšlo MySQL 8.0, mělo by být rychlejší a bezpečnější. *Root.cz* [online]. Praha: Internet Info, c 1998-2020, 20. 4. 2018 [cit. 2020-08-04]. Dostupné z: <https://www.root.cz/zpravicky/vyslo-mysql-8-0-melo-by-byt-rychlejsi-a-bezpecnejsi/>
- [14] What Is New in MySQL 8.0. *MySQL.com* [online]. USA: Oracle Corporation, c 2020 [cit. 2020-08-04]. Dostupné z: <https://dev.mysql.com/doc/refman/8.0/en/mysql-nutshell.html>
- [15] Sun nakonec kupuje firma Oracle: co bude s MySQL? *Root.cz* [online]. Praha: Internet Info, c 1998-2020 [cit. 2020-08-04]. Dostupné z: <https://www.root.cz/clanky/sun-nakonec-kupuje-firma-oracle-co-bude-s-mysql/>
- [16] Databáze zdarma pro komerční použití. *Itcek.cz* [online]. Praha: Fyzická osoba zapsaná v Živnostenském rejstříku – Jitka Vachtová, 2015, 15. 11. 2019 [cit. 2020-08-04]. Dostupné z: <https://www.itcek.cz/clanky/clanek/248>
- [17] MySQL Products. *MySQL.com* [online]. USA: Oracle Corporation, c 2020 [cit. 2020-08-04]. Dostupné z: <https://www.mysql.com/products/>
- [18] Nevýhody MySQL. *Garth.cz* [online]. Praha: Garth, 2007 [cit. 2020-08-04]. Dostupné z: <http://www.garth.cz/uvod-do-mysql/nevyhody-mysql/>
- [19] Srovnání – kdy je lepší MySQL a kdy MariaDB? *Computerworld: Deník pro IT profesionály* [online]. Praha: Internet Info DG, a.s, 2005, 16.08.2013 [cit. 2020-08-04]. Dostupné z: <https://computerworld.cz/software/srovnani-kdy-je-lepsi-mysql-a-kdy-mariadb-50258>

- [20] FELDMAN, Asher. *Wikipedia Adopts MariaDB. Diff* [online]. San Francisco: wikimedia foundation, 2001 [cit. 2020-08-04]. Dostupné z: <https://diff.wikimedia.org/2013/04/22/wikipedia-adopts-mariadb/>
- [21] MySQL vs. MariaDB. *VSHosting KnowledgeBase* [online]. Deutschland: phpMyFAQ Team, c 2001–2016, 23.4.2020 [cit. 2020-08-04]. Dostupné z: <https://support.vshosting.cz/content/2/19/cs/mysql-vs-mariadb.html>
- [22] POWERLINK – Ethernet TCP/IP. *Br-automation* [online]. Brno: B&R Headquarters: Brno, c 2020 [cit. 2020-08-04]. Dostupné z: <https://www.br-automation.com/cs/produkty/networks-and-fieldbus-modules/powerlink-ethernet-tcpip/>
- [23] Produkty B&R – automatizace, která má vtip. *Automa*. 2005, **2005**(3), 13-14.
- [24] VRÁTNÝ, Timotej. *Využití standardu OPC UA v průmyslové komunikaci*. Praha, 2019. Bakalářská práce. České vysoké učení technické v Praze – Fakulta strojní. Vedoucí práce Ing. Mgr. Jura Jakub Ph.D.
- [25] What is OPC? *OPC Foundation* [online]. USA: HIREAWIZ, c 2020 [cit. 2020-08-04]. Dostupné z: <https://opcfoundation.org/about/what-is-opc/>
- [26] OPC UA. *Br-automation* [online]. Brno: B&R Headquarters: Brno, c 2020 [cit. 2020-08-04]. Dostupné z: <https://www.br-automation.com/cs/technologie/opc-ua/>
- [27] Komunikace přes rozhraní OPC UA. *Promotic* [online]. Ostrava: MICROSYS, c 2020 [cit. 2020-08-04]. Dostupné z: <https://www.promotic.eu/cz/pmdoc/Subsystems/Comm/OPC/OPCUA.htm>
- [28] PECHÁČEK, Jakub. *Komunikační protokoly pro chytré sítě*. Liberec, 2019. Bakalářská práce. Technická universita v Liberci – Fakulta mechatroniky, informatiky a mezioborových studií. Vedoucí práce Ing. Jan Kraus, Ph.D.
- [29] KUBIŠTOVÁ, Ing. Jana. *Analýza výrobních ukazatelů v podniku a jejich vliv na včasné dodání objednávek*. Praha, 2019. Diplomová práce. České vysoké učení technické v Praze – Fakulta elektrotechnická. Vedoucí práce doc. Ing. Július Bemš, Ph.D.
- [30] METELKOVÁ, Bc. Lenka. *Projekt tvorby interních KPI výrobní částispolečnosti Meopta – optika, s.r.o.* Zlín, 2013. Diplomová práce. Univerzita Tomáše Bati ve Zlíně – Fakulta managementu a ekonomiky. Vedoucí práce Ing. Daniel Remeš, Ph.D.
- [31] KRÁLOVÁ, Lenka. *Výkonnost výrobních procesů ve strojírenské firmě*. Jihlava, 2013. Bakalářská práce. Vysoká škola polytechnická Jihlava – Katedra ekonomických studií. Vedoucí práce Ing. Petr Tyráček, Ph.D., MBA.
- [32] TOPIČOVÁ, Markéta. *Štíhlá výroba a SMED ve výrobní firmě*. Jihlava, 2013. Bakalářská práce. Vysoká škola polytechnická Jihlava. Vedoucí práce Ing. Petr Tyráček, Ph.D., MBA.
- [33] OEE – hodnocení efektivity výrobních zařízení. *MES Pharis* [online]. Brno: UNIS, c 2020 [cit. 2020-08-05]. Dostupné z: <https://www.pharis.cz/cs/klicove-oblasti-kpi>
- [34] Comes OEE [online]. Žďár nad Sázavou: Comes OEE, c 2020 [cit. 2020-08-05]. Dostupné z: <https://www.oee.cz/>
- [35] FPY (First Pass Yield). *ManagementMania.com* [online]. Wilmington: ManagementMania, c 2011-2016 [cit. 2020-08-05]. Dostupné z: <https://managementmania.com/cs/fpy-first-pass-yield>
- [36] ZEMAN, Lubomír. *Vstříkávání plastů: teorie a praxe*. Praha: Grada Publishing, 2018. ISBN 978-80-271-0614-1.

12 Seznam obrázků

OBRÁZEK 5.1 SCHÉMA I/O OSTRŮVKU SE ZAPOJENÍM SENZORŮ	18
OBRÁZEK 5.2 PRINCIP PŘIHOJENÍ K DATABÁZI.....	19
OBRÁZEK 5.3 BĚH SKRIPTU V PYTHONU.....	20
OBRÁZEK 5.4 ZNÁZORNĚNÍ STAVOVÉHO STROJE CYKLU POMOCÍ SFC DIAGRAMU.....	22
OBRÁZEK 5.5 ZNÁZORNĚNÍ STAVOVÉHO STROJE DATABÁZE POMOCÍ SFC DIAGRAMU	24
OBRÁZEK 5.6 PODROBNĚJŠÍ DIAGRAM PŘECHODŮ STAVU S1 POMOCÍ SFC DIAGRAMU	25
OBRÁZEK 5.7 PODROBNĚJŠÍ DIAGRAM PŘECHODŮ STAVU S2 POMOCÍ SFC DIAGRAMU	26
OBRÁZEK 5.8 PODROBNĚJŠÍ DIAGRAM PŘECHODŮ STAVU S3 POMOCÍ SFC DIAGRAMU	27
OBRÁZEK 5.9 PODROBNĚJŠÍ DIAGRAM PŘECHODŮ STAVU S4 POMOCÍ SFC DIAGRAMU	28
OBRÁZEK 5.10 ZNÁZORNĚNÍ STAVOVÉHO STROJE E-MAILOVÉHO UPOZORNĚNÍ A ALARMŮ POMOCÍ SFC DIAGRAMU	29
OBRÁZEK 5.11 ROZLOŽENÍ VIZUALIZAČNÍHO OKNA	30
OBRÁZEK 5.12 VIZUALIZACE – PŘEHLED VÝROBY.....	31
OBRÁZEK 5.13 VIZUALIZACE – PARAMETRY	32
OBRÁZEK 5.15 VIZUALIZACE – STATISTIKA	33
OBRÁZEK 5.14 VIZUALIZACE – GRAFY	33
OBRÁZEK 5.16 VIZUALIZACE – HISTORIE CYKLŮ	34
OBRÁZEK 5.17 VIZUALIZACE – ZMĚNA DÍLU	35
OBRÁZEK 6.1 STRUKTURA DATABÁZE	39

13 Seznam tabulek

TABULKA 5.1 PŘEHLED UŽIVATELSKÝCH PRÁV	37
TABULKA 8.1 POČET VYROBENÝCH KUSŮ	41
TABULKA 8.2 HODNOTY OEE.....	42
TABULKA 8.3 HODNOTY OEE PRO JEDNOTLIVÉ SMĚNY A CELKOVÁ HODNOTA OEE V RÁMCI TÝDNE	42

14 Seznam grafů

GRAF 2.1 OBLÍBENOST 15 NEJLÉPE HODNOCENÝCH DATABÁZOVÝCH SYSTÉMŮ A JEJICH TREND V POSLEDNÍCH 7 LETECH [5]	5
GRAF 4.1 FAKTORY OVLIVŇUJÍCÍ HODNOTU OEE [32]	13