VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMUNICATIONS

Moderní provozní technologie

Modern operational technologies

HABILITAČNÍ PRÁCE

HABILITATION THESIS

AUTOR PRÁCE: **Ing. Radek Fujdiak, Ph.D.**

AUTHOR

BRNO 2023

ABSTRAKT

Tato práce poskytuje komplexní přehled provozní techniky (OT) a jejích různých součástí a aplikací. Pokrývá podstatu OT, včetně základní terminologie, klíčových komponent, jako jsou procesy, senzory, akční členy, ovladače a rozhraní člověk-stroj, stejně jako komunikační techniky a technologie používané v OT. Práce se také zaměřuje na architekturu OT systémů a konvergenci IT a OT. Obsah je rozdělen do několika kapitol, z nichž každá se zaměřuje na jiný aspekt tématu, včetně případových studií a ukázek OT v různých průmyslových prostředích.

KLÍČOVÁ SLOVA

Provozní technologie (OT); Součásti OT; Procesy; Senzory; Akční členy; Aktuátory; Kontroléry; Rozhraní člověk-stroj (HMI); Komunikační techniky a technologie; IT/OT konvergence; Průmyslové aplikace

ABSTRACT

This work provides a comprehensive overview of operational technology (OT) and its various components and applications. It covers the essence of OT, including basic terminology, key components such as processes, sensors, actuators, controllers and human-machine interfaces, as well as communication techniques and technologies used in OT. The work also delves into the architecture of OT systems and the convergence of IT and OT. The content is divided into several chapters, each focusing on a different aspect of the topic, including case studies and OT demonstrations in various industrial settings.

KEYWORDS

Operational Technology (OT); Components of OT; Processes; Sensors; Actuators; Controllers; Human-Machine Interface (HMI); Communication Techniques and Technologies; IT/OT Convergence; Industrial Applications

FUJDIAK, Radek. *Moderní provozní technologie*. Brno, 2023, 193 s. Habilitační práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací.

PROHLÁŠENÍ

Prohlašuji, že svou habilitační práci na téma „Moderní provozní technologie“ jsem vypracoval samostatně a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené habilitační práce dále prohlašuji, že v souvislosti s vytvořením této habilitační práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkovýcha jsem si plně vědom následků porušení ustanovení §11 a následujícího autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorskýma o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

Podpis autora

Mým nejbližším.

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Rozvoj průmyslu od 18. století, kdy byl poprvé představen tkalcovský stroj, až do současnosti, se stal příkladem neustálého posouvání hranic technologického pokroku. Tyto změny lze rozdělit do pěti průmyslových revolucí: Průmyslová revoluce (1760-1840), Průmyslová revoluce parního stroje (1840-1920), Elektrifikace a automatizace (1920-1960), Konvergence informačních a provozních technologií (1960-2010) a Průmysl 4.0 (od 2010). Průmyslová revoluce (1760-1840) byla charakterizována vynálezem a rozšířením mechanizovaných tkalcovských stavů, které přinesly zrychlení výroby textilních výrobků a změnu způsobu výroby. Průmyslová revoluce parního stroje (1840-1920) se zaměřila na rozšíření parního strojařství, které umožnilo výrobu většího množství produktů za kratší dobu, vedlo k růstu ekonomiky a k zlepšení životního standardu. Elektrifikace a automatizace (1920-1960) byla založena na elektrifikaci průmyslových výrobních linek a automatizaci výrobních procesů, což umožnilo výrobu produktů s vysokou efektivitou a zvýšení produktivity. Konvergence informačních a provozních technologií (1960-2010) přinesla plnou konvergenci informační technologie (Information Technology, IT) a provozní technologie (Operational Technology, OT) a vytvořila integrovanou informační a provozní architekturu, která umožnila nejen výrobu produktů s vysokou efektivitou, ale také řízení výroby a přenos dat v reálném čase. Tyto změny představovaly významný krok vpřed v historii průmyslu a umožnily výrobu produktů s vyšší kvalitou a nižšími náklady. V současnosti se již začíná mluvit o páté průmyslové revoluci, která bude zaměřena na rozvoj robotiky, umělé inteligence, obnovitelných zdrojů, dekarbonizace, deindustrializace, spolupráce mezi člověkem a robotem (kolaborativním robotem), bio-ekonomiky a dalších řešení, která přicházejí do průmyslového ekosystému. Tyto trendy představují budoucnost průmyslu, který bude více efektivní, udržitelný a spojený s informačními technologiemi. Samotný koncept chápání průmyslového světa se tak značným způsobem mění od technologií, až po samotné koncepty chápání výroby, či přístupu k jednotlivým průmyslovým procesům. Výroba se stává více flexibilní a pružnou, což umožňuje výrobcům reagovat na změny v poptávce a zlepšovat své procesy. Tyto změny vedou k vyšší konkurenceschopnosti průmyslových odvětví na světovém trhu a k zlepšení životního standardu pro spotřebitele. A právě zlepšení životního standardu pro spotřebitele je – mimo jiné, jedním z důvodů vzniku této práce, která se zabývá aktuální problematikou průmyslových sítí, resp. konkrétně moderním provozním technologiím, jejich součástem, a to od samotného procesu, přes zařízení, až po infrastrukturu a architekturu. V neposlední řadě nad rámec ucelené teoretické stránky s redefinovanou terminologií přináší tato práce také praktické poznatky z výzkumu a vývoje v rámci reálných průmyslových systémů včetně ukázky jejich realizace od návrhu, přes implementaci až k finalizaci.

1 Podstata a přehled práce

1.1 Motivace práce

V oblasti OT došlo v posledních letech k významným změnám, kdy se prostředí stává stále dynamičtějším a složitějším s konvergencí IT a OT. Digitalizace průmyslu vedla ke značnému pokroku, avšak současně komplikuje pochopení základních principů a technologií, které jsou v této oblasti důležité. Terminologie a pojmy používané k popisu OT a souvisejících technologií mohou být nekonzistentní a matoucí, kvůli kritické povaze mnoha OT systémů je tak nezbytné pečlivé a přesné pochopení daných termínů. V současnosti se rozšiřuje využití digitálních technologií v kritických infrastrukturách, jako jsou energetické, vodní, dopravní a komunikační systémy, což zdůrazňuje nutnost porozumět základním principům OT. Tyto systémy hrají klíčovou roli v našem každodenním životě a vyžadují integraci složitých technologií a procesů, aby byl zajištěn jejich spolehlivý provoz. Proto je nutné, aby vývoj a implementace nových OT technologií a procesů byly provedeny s pečlivostí a ohledem na jejich dopad na kritickou infrastrukturu a širší společnost. S rostoucím významem digitálních technologií a složitostí v oblasti OT se stává tento obor vzrušujícím a důležitým polem pro výzkum a publikování. S rostoucí závislostí na digitálních technologiích je zásadní dosáhnout komplexní porozumění základním principům a technologiím v této oblasti, což může pomoci organizacím lépe se připravit na výzvy a příležitosti, které představují rychle se měnící prostředí a neustálý vývoj OT. Tyto výzvy a příležitosti mohou být komplexní a vyžadují odbornou znalost a schopnost průběžně se vyvíjet a adaptovat se, aby byly tyto problémy dostatečně řešeny. Proto je třeba, aby se organizace a podniky snažily rozvíjet své znalosti a dovednosti v oblasti OT, aby mohly využívat potenciál digitálních technologií a zároveň minimalizovat rizika spojená s jejich využitím. Tato práce tak nabízí jasný a přesný pohled na základní principy a technologie v oblasti OT, a tím poskytuje cenné informace pro organizace, které se snaží rozvíjet své znalosti v této oblasti. V neposlední řadě je motivací nejednotnost v terminologii v rámci OT. Tento problém je způsoben rostoucím rozvojem jednotlivých odvětví v průmyslu, jejich slučováním, rozvojem různých technologií v rámci různých vývojových větví a aplikačních odvětví. Tyto faktory vedou k výskytu nekonzistentností v terminologii a teoretických poznatcích, což může být pro odborníky a laickou veřejnost obtížně srozumitelné. Motivací pro tuto práci je tedy sjednotit terminologii a teoretické poznatky v oblasti OT do jednoho uceleného celku, který bude jasně představovat OT a jejich význam pro společnost. Toto sjednocení terminologie a teoretických poznatků může pomoci odborníkům v této oblasti lépe komunikovat a spolupracovat, stejně jako může pomoci laické veřejnosti lépe pochopit důležitost OT pro naši společnost.

1.2 Cíle práce

Cílem této práce je vymezit a analyzovat současný stav konvergence IT/OT a jejich vliv na průmysl. Růst digitalizace průmyslu v posledních letech přináší nové možnosti, ale také nové výzvy. Tyto změny se projevují také ve změněném prostředí, kde se organizace a podniky snaží přizpůsobit a rozvíjet své OT. Proto je nutné analyzovat a porozumět vlivům těchto změn na implementaci a přijetí OT. Dalším cílem práce je zkoumat nejednotnost terminologie v rámci OT, která vznikla v důsledku rozvoje jednotlivých odvětví a technologií. Tyto cíle vedou k definici a sjednocení terminologie v oboru OT, což pomůže v budoucnu lépe porozumět principům a technologiím v této oblasti. V rámci této práce se také analyzuje role standardů, norem a předpisů při utváření vývoje a implementace OT systémů. Tyto standardy a předpisy zajišťují bezpečný a spolehlivý provoz kritických infrastruktur, a proto je nutné je pečlivě zvážit při vývoji a implementaci nových OT systémů. V neposlední řadě bude tato práce ukazovat možnosti realizace dnešních průmyslových sítí pomocí OT technologií, od návrhu až po finalizaci. Tato ukázka bude demonstrovat, jak lze využít moderních technologií, jako je Průmysl 4.0 a internet věcí (IoT), k realizaci efektivního a spolehlivého provozu průmyslových sítí. Celkově tedy cílem této práce je analyzovat současný stav konvergence IT/OT a její vliv na průmysl, zkoumat výzvy a příležitosti, které přináší digitalizace průmyslu, a definovat prostředí OT v kontextu moderních technologií. Práce také zkoumá nejednotnost terminologie v rámci OT a zabývá se jejím sjednocením, roli standardů, norem a předpisů v utváření vývoje a implementace OT systémů a ukazuje možnosti realizace průmyslových sítí s využitím OT technologií. Tyto cíle jsou zásadní pro komplexní pochopení a rozvoj v oblasti OT a přispějí k lepší přípravě na výzvy a příležitosti, které přicházejí s digitalizací průmyslu. **Stěžejní otázky v rámci cílů práce tak lze shrnout takto:**

1. Jaké výzvy a příležitosti přináší digitalizace průmyslu?
2. Jak měnící se prostředí ovlivňuje přijetí a implementaci OT?
3. Jaká je současná úroveň jednotnosti terminologie používané v rámci OT a jak ji lze standardizovat?
4. Jakou roli hrají normy, předpisy a normy při utváření vývoje a implementace systémů OT?
5. Jaké jsou základní komponenty v rámci OT sítí a jak jsou napojeny na dnešní chápání průmyslových sítí?
6. Jaký je současný stav konvergence IT a OT a jaký je její dopad na průmysl?
7. Jak lze moderní OT technologie, jako je Průmysl 4.0 a IoT, využít k efektivnímu provozu průmyslových sítí?
8. Jaké překážky a řešení představuje implementace OT v průmyslových aplikací?

1.3 Přínos práce

Tato práce přináší řadu užitečných poznatků a přínosů v oblasti konvergence IT/OT. Prvním přínosem je analýza současného stavu konvergence IT/OT a jejího vlivu na průmysl. Tyto informace jsou pro organizace a podniky důležité, aby mohly lépe porozumět změnám, kterým čelí, a přizpůsobit se jim. Dalším přínosem je zkoumání výzev a příležitostí, které přináší digitalizace průmyslu. Tyto informace mohou být užitečné pro organizace při rozhodování o nových projektech a investicích, které mohou přinést významné konkurenční výhody. Tato práce také přináší sjednocení terminologie v oboru OT, což v budoucnu pomůže lepšímu porozumění principům a technologiím v této oblasti a pomohou v budoucnu vyhnout se nejasnostem a nedorozuměním. V rámci této práce se také analyzuje role standardů, norem a předpisů při utváření vývoje a implementace OT systémů. Tyto informace jsou pro organizace důležité pro správné a bezpečné fungování kritických infrastruktur. Tato práce dále představuje možnosti realizace dnešních průmyslových sítí pomocí OT technologií, od návrhu až po finalizaci. Celkově tedy tato práce přináší komplexní pohled na současný stav konvergence IT/OT, její vliv na průmysl, výzvy a příležitosti, které přináší digitalizace, sjednocení terminologie, analýzu role standardů, norem a předpisů a ukázku možností realizace průmyslových sítí pomocí OT technologií.

Návaznost na autorovy publikace. Autorovi publikace jsou v přímé souvislosti s tématikou této práce. Jedná se převážně o návaznost na dlouhodobý výzkum od roku 2017, tedy roku ukončení doktorského studia, v tématech, které jsou mj. základem pro praktickou část této práce. Jedná se tedy převážně o:

- Výzkum, vývoj a testování v oblasti funkčních a přenosových parametrů komunikačních i přenosových technologií v průmyslu (koexistence a interoperabilita [APub41, APub40, APub17, APub15]; výkonnostní, experimentální i funkční analýza, testování a měření [APub36, APub33, APub31, APub47, APub46, APub34, APub19, APub10, APub28]); nové způsoby využití stávajícíh dat [APub49]; blockchainové aplikace [APub14]; modelování a simulace [APub11, APub16, APub4, APub12, APub32]).
- Výzkum a vývoj v oblasti průmyslových testovacích polygonů, architektury a infrastruktury, kyber-fyzických systémů, kybernetických dvojčat a pokročilé virtualizace (komunikační a funkční modely [APub5]; kontejnerizace a virtualizace [APub18, APub1]; testovací a kyber-fyzické polygony [APub3, APub38, APub39]; emulace, modelování a simulace [APub50, APub51]; reálné aplikace [APub30, APub8]).
- Výzkum, vývoj a testování v oblasti kybernetické bezpečnosti a hrozeb v průmyslu (lehká kryptografie [APub6, APub27, APub48]; kybernetické útoky, je-

jich dopad a mitigace [APub45, APub43, APub29, APub52, APub42]; bezpečný návrh a vývojový životní cyklus [APub13, APub9, APub22]; bezpečnostní testování průmyslových sítí [APub7, APub23, APub24, APub35]; detekce anomálií a hrozeb [APub2, APub20, APub26, APub37, APub44, APub25, APub21]).

Návaznost na studijní materiály. Jako první je nutno zmínit autorovi dvě knižní publikace, které přináší v rámci knihy *Budování Cyber Range platformy s technologií cloud computingu* [APed11] zcela nové poznatky v oblastech kybernetických polygonů, edukace i trénování v kybernetické bezpečnosti, a to v návaznosti mj. na průmyslové aplikace. Z pohledu druhé knižní publikace *Counter measure techniques for cryptographic algorithms eliminating power analysis attacks (Extended Version)* jsou to pak převážně metody zajištění bezpečnosti proti fyzickým útokům postranními kanály. Jako další lze zmínit řadu studijních materiálů využívaných v rámci bakalářských kurzů (předměty: BVKS [APed10, APed5], BCZS [APed8], BIOT [APed6], BDAK [APed3, APed2], CZKR [APed4]) i magisterských kurzů (předměty: MKRI [APed1], MPPR [APed7], MVDP [APed9]) na VUT v Brně. Poznatky z těchto materiálů jsou převážně přeneseny do teoretické částí této práce.

Výzkumné projekty a hospodářské smlouvy. Autor práce se aktivně účastnil řady výzkumných projektů interního, národního, evropského či mezinárodního charakteru, stejně tak v případě hospodářských smluv, a to jako řešitel (R), další řešitel či spoluřešitel (S), či člen řešitelského týmu (C):

- V rámci interních projektů se jedná výzkum a vývoj elektronických, informačních a průmyslových komunikačních systémů včetně jejich kybernetické bezpečnosti: FEKT-S-14-2352 (S) [APro4], FEKT-S-17-4184 (S) [APro5], FAST/FEKT-J-16-3344 (S) [APro3], FEKT-S-20-6312 (S) [APro6], FEKT/FIT-J-18-5434 (C) [APro7], FEKT/FIT-J-19-5905 (C) [APro8], a FEKT/FIT-J-19-5906 (C) [APro9].
- Národní projekty se v rámci činnosti autora pohybovaly v rámci oblasti kyberbezpečnostního dohledu nad průmyslovými sítěmi a kritickou infrastrukturou, komunikačními a přenosovými technologiemi pro průmyslové sítě, virtualizace, digitální dvojčata a kyber-fyzické polygony, jednalo se o projekty: VI20172019057 (C) [APro17], FV20487 (S) [APro10], TJ01000381 (S) [APro13], TJ02000332 (R) [APro14], TK02030013 (S) [APro15], VI20192022132 (S) [APro18], FV40366 (S) [APro11], FW01010474 (S) [APro12], a TK03010091 (C) [APro16].
- Evropské a mezinárodní projekty se v rámci autorovy účasti týkaly především průmyslových sítích a problematiky kontradikce parametrů výkonu, pracovní

bezpečnosti a kybernetické bezpečnosti, a to v projetku AQUAS (C)[APro2]. Dále pak další účast na projektu RUGGEDISED (C) [APro1], který se zabýval převážně problematikou chytrých měst a využití moderních digitálních technologií včetně dobré praxe pro integraci těchto poznatků do stávajících měst.

1.4 Struktura práce

Tato práce poskytuje ucelený přehled problematiky moderních OT a jejích různých součástí a aplikací. Obsah je rozdělen do několika kapitol, z nichž každá se zaměřuje na jiný aspekt tématu:

- **Kapitola 1** seznamuje čtenáře s hlavními cíli a motivací práce, včetně její struktury.
- **Kapitola 2** se zaobírá samotnou podstatou OT, včetně základní terminologie a klíčových komponent, jako jsou procesy, senzory, akční členy, ovladače a rozhraní člověk-stroj. Zahrnuje také komunikační techniky a technologie používané v OT, architekturu systémů OT a konvergenci IT a OT.
- **Kapitola 3** se zaměřuje na případové studie a demonstrace OT v různých průmyslových prostředích, jako je průmyslová balicí smyčka, čistička a pivovar. Každá případová studie obsahuje souhrn, seznam použitých komponent, vstupní kritéria a úvahy o návrhu, technické popisy a výsledky testování a ověřování.
- **Kapitola 4** knihu uzavírá shrnutím hlavních zjištění a výsledků práce. Kromě toho jsou zahrnuty také autorovy publikace, výukové materiály a další relevantní odkazy. Pro orientaci je také uveden seznam symbolů a zkratek používaných v celé práci.

Práce tak poskytuje komplexní přehled o OT a jeho různých komponentách a aplikacích a je cenným zdrojem pro každého, kdo chce hlouběji porozumět tomuto složitému a rychle se vyvíjejícímu oboru.

2 Provozní technologie

Provozní technologie, někdy také označované jako operační technologie, jsou zjednodušeně veškerý hardware a software používaný k řízení, monitorování a/nebo udržování fyzických aktiv, procesů nebo událostí. Pro příklad tedy lze říci, že se jedná o obrábění, svařování, lisování, tvarování plastů a mnoho dalších. Tyto technologie se používají (nejen) k výrobě výrobků, které splňují požadavky zákazníka co nejpřesněji, a to co nejefektivněji, k čemuž se právě využívají dnes i moderní technologie pro monitorování, řízení či plánování. To znamená obsáhnutí technologií od fyzických zařízení, tedy např. různých aktuátorů, tedy pohybových zařízení, která dokážou pohybovat nebo ovlivňovat pozici objektů (například při výrobě automobilů mohou být aktuátory použity k ovládání pohybu lisovacích forem nebo k ovládání pohybu svařovacích ramen), až po kontrolní centra, ekonomická či manažerská oddělení.

Tyto technologie jsou páteřním prvkem pro mnoho podniků či firem, jelikož jsou součástí jejich každodenní produkční činnosti [50]. Dnes jsou však OT již součástí téměř všech oblastí lidské činnosti [32], včetně energetiky (naftařství, plynárenství, elektroenergetiky či teplárenství), chemického průmyslu, výrobního a strojírenského průmyslu, vodohospodářství, zpracování odpadu, přepravy, logistiky, potravinářství, zemědělství, zdravotnictví, hutnictví a těžařského průmyslu, ale i spousty dalších oblastí, např. i prosté automatizace budov či domácností. OT tak hrají klíčovou roli v zajišťování tzv. kritické výroby a kritické infrastruktury, jakož i v udržování jejich efektivního fungování. Kritická výroba a kritická infrastruktura jsou dva pojmy, které se týkají klíčových aspektů hospodářství a bezpečnosti jednotlivých zemí. Kritická výroba zahrnuje sektory, jako jsou energie, zdroje, zásobování potravinami a zdravotnictví, které jsou nezbytné pro chod společnosti. Tyto sektory musí být neustále v provozu, aby se zajistilo, že populace má přístup k základním potřebám a službám. Kritická infrastruktura zahrnuje komunikační systémy, dopravní sítě, vodárny a elektrárny, které jsou rovněž klíčové pro fungování společnosti. Tyto systémy musí být chráněny proti vnějším hrozbám, jako jsou přírodní katastrofy, teroristické útoky nebo kybernetické útoky, aby se zajistilo, že budou fungovat neustále a bezpečně.

S pojem kritická výroba se však převážně setkáváme v rámci např. Spojených států amerických. V rámci České republiky (ČR) se využívá hlavně pojmu kritická infrastruktura, definována v ČR dle zákona č. 240/2000 Sb. [52], tedy jako prvek kritické infrastruktury nebo systém prvků kritické infrastruktury, jehož narušení by mělo závažný dopad na bezpečnost státu, zabezpečení základních životních potřeb obyvatelstva, zdraví osob nebo ekonomiku státu. Z definice však tedy kritická výroba v ČR spadá do kritické infrastruktury.

2.1 Základní terminologie

Na začátek je z pohledu terminologie důležité vysvětlit rozdíly v rámci překladu termínu OT v odborných českých textech. Je nutno říci, že aktuálně neexistuje přímá (přesná) definice termínu OT či jeho překladu. Objevují se aktuálně dvě hlavní varianty:

- provozní technologie (zkratkou jako OT, ale někdy také česky jako PT) [14],
- operační technologie (jako přímý překlad, zkratkou jako OT) [37].

Termínově se jedná o ekvivalenty a synonyma. V rámci českého jazyka je však termín *operační technologie* spojen převážně s vyjádřením funkční (provozu-schopná) technologie [17] či s termínem v souvislosti např. se zdravotnickými (chirurgickými) operacemi [7]. Pro ČR lze dále sledovat oficiální směr překladů pro OT z evropských direktiv a nařízení, kde je využíván překlad:

- *Operational technology* jako operační technologie [17], ve smyslu funkční (provozuschopné) technologie, systém či síť.
- *Operational technology* jako provozní technologie [18, 19], ve smyslu programovatelných digitálních systémů nebo zařízení, které interagují s fyzickým prostředím nebo řídí zařízení, která interagují s fyzickým prostředím.

Termín provozní technologie se tak jeví z tohoto pohledu jako vhodnější, díky souladu s platnou legislativou, a bude proto i dále v textu v tomto smyslu používán. Z pohledu zkratky pak je používání OT oproti PT mnohem více zažité, kdy zkratka PT je používána minimálně, a tedy i z tohoto důvodu bude používáno terminologie – provozní technologie (OT).

Termín OT je spjat také s průmyslovými řídicími systémy (ICS, Industrial Control Systems). Termín ICS vznikl jako reakce na rozdílnou terminologii způsobenou různým vývojem jednotlivých odvětví, kdy se v posledních 30 letech hledal termín, který by zahrnoval všechny formy průmyslové automatizace (IA, Industrial Automation) [13]. První návrhy směřovaly do termínu – řídicí systém (Control System), to však bohužel zahrnovalo terminologicky nejen IA, ale také oblasti jako automatizaci budov (Building Automation), či automatizaci domácnosti (Home Automation). Z této problematiky tak vzešel hybridní termín – průmyslový řídicí systém (ICS, Industrial Control System), který byl velmi rychle přijat širokou odbornou veřejností, a který se začal používat jako termín zahrnující veškeré formy IA. ICS je používán mj. i Národním institutem standardů a technologií (NIST, National Institute of Standards and Technologies), např. v rámci publikace NIST SP 800-82 [44].

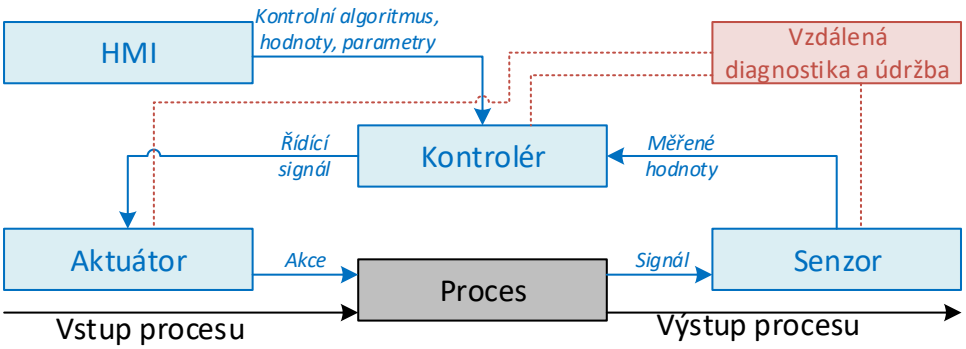
S ohledem k OT, ICS a IA se lze také ještě setkat s označením průmyslová automatizace a řídicí systémy (IACS, Industrial Automation and Control Systems), což je termín představený Mezinárodní společností pro automatizaci (ISA, International Society of Automation). Nutno zmínit, že do roku 2006 byl IACS označo-

ván organizací ISA jako výrobní a průmyslové systémy (M&CS, Manufacturing and Control Systems). ISA označení IACS využívá ve spojitosti s kybernetickou bezpečností v sérii standardů a norem označovaného jako ISA-99 [29] (známého také jako IEC/ANSI/ISA 62433 [41]). Série standardů ISA-99 byla vytvořena v rámci výboru 99 organizace ISA a nesla tedy označení ISA-99. Následně byla série akreditována a publikována Americkým Národním Standardizačním Institutem (ANSI), kde došlo v roce 2010 k přečíslování na ANSI/ISA-62443. Série pak byla převzata i Mezinárodní elektrotechnickou komisí (IEC, International Electrotechnical Commission) jako IEC 62443, což je již označení v oboru široce známé. Z definice standardu IEC 62443-1-1 [28], IACS pokrývá řídicí systémy využívané ve výrobních a zpracovatelských závodech, systémech kontroly životního prostředí, geograficky rozsáhlých systémech (mj. elektřina, plyn a voda), potrubí a petrochemický průmysl, ale i další průmyslová odvětví a aplikace, jako jsou doprava, kde se využívá automatizované nebo dálkově ovládané či monitorované procesy (aktiva). Mnoho odborníků v tomto ohledu kritizuje definici dalšího termínu v podobě IACS, který dále rozděluje komunitu a terminologii, a omezuje tak využití případných užitečných standardů i v jiných odvětvích než jen v průmyslu (např. zdravotnictví) [15]. Právě *průmyslová* v tomto ohledu implikuje na omezení aplikační oblasti pouze pro průmyslovou oblast. Stejně tak z definice uvedené oblasti jako *další průmyslová odvětví*. Nicméně v tomto kontextu je nejspíše myšlen obecný *průmysl*, čemuž by odpovídal i fakt ve využívání termínu IACS vůči informačním technologiím (IT, Information Technology) jako antonymum. Tedy velmi obdobně jako je tomu u zažitých antonym IT a OT. Právě díky tomu je mnohdy OT a IACS zaměňováno či považováno za synonyma, alespoň v některých aplikačních odvětví. To však terminologicky není zcela správně.

OT se odkazují na historický pojem, který zahrnuje široké spektrum technologií, systémů a infrastruktur zaměřených na jakoukoliv činnost v rámci průmyslových procesů, a to jak v oblasti hardwaru, tak i softwaru. Hlavním účelem OT je tedy zajistit spolehlivý provoz těchto průmyslových procesů včetně veškerých jeho funkcionalit. Na druhé straně, IACS je pojem, který byl nově definován v normě IEC 62443, a který se primárně zaměřuje na kybernetickou bezpečnost v automatizačních systémech. IACS tak má mnohem specifičtější zaměření než široký pojem OT a lze ho tedy z principu považovat za podmnožinu OT. ICS je pak často zaměňováno s IACS/OT, ale ve skutečnosti se jedná o specifický typ průmyslového systému, který se zaměřuje pouze na řízení a monitorování průmyslových procesů. Systémy automatizace a řízení budov, které se nezabývají přímo průmyslovými procesy, nejsou zahrnovány do kategorie ICS, ale spadají do širší kategorie OT/IACS. Z tohoto důvodu lze konstatovat, že ICS je podmnožinou OT.

2.2 Hlavní komponenty OT

Přejděme nyní na hlavní komponenty OT systémů. Na obrázku níže (obr. 2.1) můžeme vidět obecnou ukázku OT systému z pohledu jeho komponent a základního logického propojení.



Obr. 2.1: Obecný model s komponentami OT [24].

Jednotlivé části můžeme tedy popsat jako:

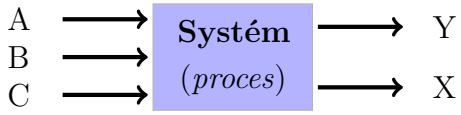
- **Proces** je ve smyslu k OT myšlen jako průmyslový proces, např. výrobní proces, technologický proces nebo proces řízení budov. Tyto procesy se skládají z několika kroků a mají určité cíle, kterých je třeba dosáhnout. Proces se řídí a monitoruje prostřednictvím dalších komponent, jako jsou aktuátory, senzory, kontroléry, rozhraní člověk-stroj (Human Machine Interface, HMI), či blok vzdálené diagnostiky a údržby. Tyto komponenty pomáhají zajistit, aby byl proces řízen správně, a aby se dosáhlo požadovaných výsledků. Může to být například proces v rámci výrobní linky, nebo chladícího systému.
- **Senzor** je zařízení, které měří určité fyzikální veličiny (např. rychlost, teplotu, průtok) a převádí je na signál, který může být snadno interpretován. Tyto signály pak poskytují informace o stavu procesu, který se sleduje. Senzor funguje tak, že reaguje na určitou vstupní veličinu a generuje výstupní signál, který je funkčně související s touto veličinou. Tyto informace se poté přenášejí do kontroléru, který je využije k řízení procesu. Jedná se tedy např. o teploměr, průtokoměr apod.
- **Aktuátor** (akční člen) je zařízení, které slouží k realizaci změny výstupního stavu na základě vstupního signálu ze řídicího systému (přenosu povelu z kontroléru na řízený proces). Tyto signály mohou být buď manuální, nebo automatické. Zjednodušeně aktuátory tedy slouží k pohybu či řízení mechanismu (systému) a je to právě ten mechanismus, jímž řídicí systém ovlivňuje dané

prostředí. Aktuátory pro svoji funkcionalitu využívají zdroj energie (obvykle elektrický proud, hydraulický tlak nebo pneumatický tlak) a přeměňují tuto energii na pohyb. Například v průmyslových procesech se aktuátor může používat k řízení teploty, tlaku nebo polohy. V závislosti na aplikaci se mohou aktuátory lišit v konstrukci, velikosti a typu použité energie. Tyto rozdíly ovlivňují jejich účinnost a účel použití v konkrétní aplikaci. Může se jednat např. o motor nebo ventil.

- **Kontrolér** je zařízení nebo program, který automaticky reguluje řízenou veličinu, resp. slouží k automatizaci a řízení procesu. Může to být jednoduchý mechanický nebo elektronický systém, softwarově založený, jako například ovladač tiskárny, nebo hardwarový, jako např. řídicí systém robota. Kontrolér může být například počítač nebo programovatelný logický kontrolér (Programmable Logic Controller, PLC).
- **HMI** je grafické uživatelské rozhraní, které umožňuje operátorům monitorovat stav procesů, měnit nastavení řízení a ručně přepínat automatické řízení v případě nouze. HMI také umožňuje inženýrovi nebo operátorovi konfigurovat bod nastavení nebo algoritmy a parametry řízení v kontroléru. Zobrazuje také informace o stavu procesu, historické informace, zprávy a další informace pro operátory, administrátory, manažery, partnery a jiné oprávněné uživatele. Operátoři a inženýři používají HMI k monitorování a konfiguraci bodů nastavení, algoritmů řízení, odesílání příkazů a úpravě a určení parametrů v kontroléru. Jednat se může např. o dotykový displej na průmyslovém počítači, fyzické panely s tlačítky a indikačními světly, ale i např. mobilní či webové aplikace.
- **Vzdálená diagnostika a údržba** spojuje činnosti, které umožňují provádět diagnostiku a údržbu systému, a to zvenčí bezpečnostního perimetru tohoto systému, většinou převážně prostřednictvím internetu nebo jiných sítí. Tyto funkce usnadňují správu a kontrolu systému na dálku, což může šetřit čas a zdroje. Navíc umožňují včasnou identifikaci a řešení problémů, což zvyšuje spolehlivost a funkčnost systému. Tyto funkce také umožňují vzdálenou údržbu a opravy bez nutnosti fyzického přístupu, což může být výhodné v situacích, kdy není možné přístup k systému fyzicky zajistit v daný moment. Příklad může být pouhý vzdálený přístup k PLC nebo složitější jako např. systém dispečerského řízení a sběru dat (Supervisory Control and Data Acquisition, SCADA) či distribuovaný řídicí systém (Distributed Control System, DCS).

2.2.1 Proces

Proces je základním prvkem v rámci průmyslu a hraje klíčovou roli v rámci průmyslového systému v moderní společnosti. Proces je tedy série operací, která produkuje požadovaný výstup, jednoduchý příklad můžeme vidět na obr. 2.2 níže.



Obr. 2.2: Příklad jednoduchého systému se třemi vstupy a dvěma výstupy.

Kde A, B, C představují skupinu vstupů a X, Y skupinu výstupů systému resp. procesu. Proces má tedy vždy vstup a výstup, kde výstupy jsou vytvářeny zpracováním vstupů. Vstupní signál pak definuje samotný proces, může ho změnit. nebo může změnit kompletně fungování samotného systému. Z tohoto důvodu se můžeme setkat také s označením vstupu jako *příčiny* (cause), zatímco výstup označujeme jako *efekt* (effect), což je tedy důsledek příčiny. Z pohledu různých typů průmyslových procesů rozlišujeme kontinuální procesy, diskrétní procesy, batch procesy (někdy známé také jako dávkový procesy) a hybridní procesy.

Kontinuální procesy jsou využity obecně tam, kde je materiál kontinuálně vkládán bez přerušení (zpracování ropy, chemické procesy či potravinářství). Lineární diferenciální rovnice prvního řádu se používá k modelování chování systému v průběhu času. Rovnice popisuje rychlost změny stavové proměnné s ohledem na čas a lze ji použít ke studiu chování, např. právě procesu. Kontinuální proces tak může být popsán také lineární diferenciální rovnicí prvního řádu:

$$dx(t)/dt = -kx(t) + u(t), \tag{2.1}$$

kde $x(t)$ je proměnná procesu (teplota, tlak, průtok, apod.), $u(t)$ je řízený vstup, a k je konstanta reprezentující dynamiku procesu. Pokud využijeme přenosovou funkci (Laplaceovu transformaci diferenciální rovnice) reprezentující chování ve frekvenční oblasti, můžeme získat informaci o dynamice procesu v souvislosti s jinými vstupy, což nám pomůže analyzovat stabilitu procesu (systému), časovou odezvu a frekvenční odezvu (i z tohoto důvodu jsou přenosové funkce v průmyslu hojně využívány). Z pohledu kontinuálního procesu je pak, vůči definované lineární diferenciální rovnici, přenosová funkce definována jako:

$$G(s) = K\ /(T_s + 1), \tag{2.2}$$

kde $G(s)$ je přenosová funkce, K je konstanta zesílení, a T_s je časová konstanta procesu. Z pohledu komplexní reprezentace dynamiky procesu, popisu chování procesu

v čase, stability, časové a frekvenční odezvy, je možné využít tzv. stavový model, který je matematickou reprezentací systému, který používá sadu lineárních diferenciálních rovnic prvního řádu k popisu stavových proměnných systému a jejich vztahů ke vstupům a výstupům. Z tohoto pohledu lze tak kontinuální proces popsat jako stavový model tímto způsobem:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.3}$$

kde $x(t)$ je stavový vektor, $u(t)$ je vstupní vektor, $y(t)$ je výstupní vektor, A, B, C, a D jsou matice reprezentující dynamiku procesu a vstupy/výstupy.

Diskrétní procesy jsou typické pro výrobu jednotlivých výrobků nebo sérií výrobků, kde jsou suroviny nebo polotovary zpracovávány v určitých časových intervalech, jako například v automobilovém průmyslu nebo v elektronické výrobě. Velmi obdobně jako kontinuální procesy lze i diskrétní procesy popsat diferenciální rovnicí prvního řádu:

$$x(k+1) = x(k) + Ts \cdot (dx(k)/dt), \tag{2.4}$$

kde $x(k)$ je procesní proměnná s časovým krokem, Ts je vzorkovací čas, a $dx(k)/dt$ je derivace procesní proměnné k časovému kroku. Problém v této definici však nastává hned z podstaty diskrétního procesu, kde proces již není popsán jako kontinuální změna v čase, ale naopak popsán jednotlivými diskrétními kroky. Z tohoto důvodu je mnohem přesnější popsání systému následovně:

$$x(k+1) = x(k) + T \cdot (-kx(k) + u(k)), \tag{2.5}$$

kde $x(k)$ je tedy opět procesní proměnná k časovému kroku, T je velikost časového kroku, $u(k)$ je vstup, a k je konstanta reprezentující dynamiku procesu. Z pohledu

$$G(z) = K\ /(1 - a\ \cdot z^{-1}), \tag{2.6}$$

kde $G(z)$ je přenosová funkce, K je konstanta zesílení, a a je konstanta reprezentující dynamiku procesu. Je nutno říci, že se opět jedná pouze o velmi jednoduchý příklad. V neposlední řadě pak popis diskrétního procesu z pohledu stavového modelu lze odvodit z předchozích bodů následujícím způsobem:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ y(k) &= Cx(k) + Du(k), \end{aligned} \tag{2.7}$$

Batch procesy (sériové procesy) jsou využívány při výrobě většího množství stejného výrobku, kde jsou suroviny nebo polotovary zpracovávány v určitém množství najednou, jako například v farmaceutické výrobě nebo v pekařství. Jedná se o zpravidla o jakýsi podtyp diskrétního procesu. Batch proces může být modelován opět diferenciální rovnicí prvního řádu (jedná se o obecný příklad batch procesu), a to jako:

$$\begin{aligned} dx_1(t)/dt &= -k_1 x_1(t) + u_1(t), \\ dx_2(t)/dt &= -k_2 x_2(t) + u_2(t), \\ &\dots \\ dx_n(t)/dt &= -k_n x_n(t) + u_n(t), \end{aligned} \tag{2.8}$$

kde $x_i(t)$ je procesní proměnná pro sérii i, $u_i(t)$ je vstup pro sérii i, a k_i je konstanta reprezentující dynamiku procesu pro sérii i. Pro získání přenosové funkce, je nutno opět provést Laplaceovu transformaci diferenciální rovnice, kdy tedy dostaneme:

$$\begin{aligned} X_1(s) &= (1/(s+k_1)) \cdot U_1(s), \\ X_2(s) &= (1/(s+k_2)) \cdot U_2(s), \\ &\dots \\ X_n(s) &= (1/(s+k_n)) \cdot U_n(s), \end{aligned} \tag{2.9}$$

kde $X_i(s)$ a $U_i(s)$ jsou transformace $x_i(t)$ a $u_i(t)$, a k_i jsou parametrické konstanty procesu. Nakonec tedy pro získání stavového modelu lze diferenciální rovnici uspořádat do formy matic, resp. jako:

$$\begin{aligned} dx(t)/dt &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.10}$$

kde $x(t)$ je stavový vektor, $u(t)$ je vstupní vektor, a $y(t)$ je výstupní vektor. Matice A, B, C, a D jsou matice, které mohou být odvozeny z diferenciální rovnice.

Job-shop procesy jsou většinou považovány za podmnožinu batch procesů a jsou určené pro jedinečné zakázky, kde jsou vyžadovány specifické technologie a specifické výrobní postupy. Tyto procesy se často vyskytují v průmyslu s vysokou hodnotou dodaného výrobku, jako jsou například lodě nebo letadla. Tyto procesy jsou modelovány za pomocí kombinace diferenciálních rovnic a časově diskrétního modelu, resp. jako:

$$\begin{aligned} dx(t)/dt &= -kx(t) + u(t), \\ x(k+1) &= x(k) + Ts(dx(t)/dt), \end{aligned} \tag{2.11}$$

kde $x(t)$ je procesní proměnná, $u(t)$ je vstup, k je diskrétní časový krok, a T_s je vzorkovací čas. Přenosová funkce pro job-shop proces je získána opět Laplaceovu transformací funkce procesu jako:

$$X(s)/U(s) = 1/(s + k), \tag{2.12}$$

kde $X(s)$ je Laplaceova transformace $x(t)$, $U(s)$ je Laplaceova transformace $u(t)$, s je komplexní frekvenční proměnná, a k je časová konstanta systému. Stavový model pro job-shop proces je získán definicí stavového vektoru a stavové přechodové rovnice:

$$\begin{aligned} dx(t)/dt &= Ax(t) + Bu(t), \\ x(k+1) &= Cx(k) + Du(k), \end{aligned} \tag{2.13}$$

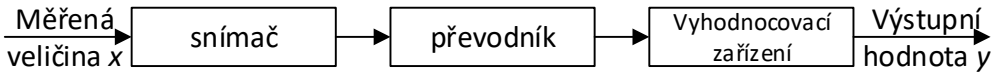
kde $x(t)$ je stavový vektor, $u(t)$ je vstup, A, B, C, a D jsou matice reprezentující dynamiku procesu.

Hybridní procesy kombinují vlastnosti kontinuálních a diskrétních procesů nebo jiných typů procesů. Tyto procesy se často používají k dosažení optimálního výrobního efektu nebo k řešení specifických výrobních potřeb. Hybridní procesy lze modelovat pomocí kombinace těchto modelů nebo pomocí složitějších modelů, které zahrnují více komponent a řídicích strategií. Konkrétní použitý matematický model bude vždy záviset na požadavcích procesu a cílech řídicího systému a nelze již jednoduchým způsobem takový model popsat.

Nakonec je nutno zmínit, že je třeba vždy zvážit specifické požadavky a potřeby řízeného procesu při volbě vhodného typu procesu. Samotný proces je v rámci OT tedy řízen pomocí kontroléru přes aktuátor, který již provádí fyzickou akci dle přijatého elektrického signálu a proces je pak monitorován přes senzor, který dodává kontroléru informace potřebné pro řízení.

2.2.2 Senzor

Senzory jsou zařízení, která detekují změny fyzikálních nebo environmentálních parametrů (velikost či hodnota představuje informační parametr) a převádějí je na elektrické signály, které mohou být zpracovány a analyzovány řídicím systémem. V uvažovaném modelu jsou senzory připojeny k procesu, ovladači a vzdálené diagnostice, což umožňuje takovému systému monitorovat a řídit různé aspekty procesu. Rozdělme si senzor na tři základní prvky: snímač (vstup, měřící zařízení), převodník a vyhodnocovací zařízení (výstup).



Obr. 2.3: Ukázka vnitřní struktury senzoru [31].

Snímací prvek přímo zachycuje data z měřeného prostředí pomocí fyzikálního zákona pro vztažení výstupního signálu k měřené veličině (x). Převodník přebírá výstupní signál ze snímacího prvku a převádí jej do použitelné formy pro další zpracování, typicky vyžadující pomocné napájení. Vyhodnocovací zařízení zpracovává výstupní signál z převodníku a připravuje je pro výstup (y). Tyto tři prvky mohou tvořit jeden konstrukční celek nebo mohou být samostatnými součástmi v závislosti na složitosti senzorového systému. Vztah mezi vstupem (x) a výstupem (y) senzoru lze modelovat pomocí matematické rovnice, která se může lišit v závislosti na typu senzoru a měřeném fyzikálním jevu. Obecnou rovnici pro senzor lze vyjádřit jako:

$$y = f(x), \tag{2.14}$$

kde $f(x)$ je matematická funkce, která představuje vztah mezi vstupem x a výstupem y. V praxi může být funkce $f(x)$ velmi jednoduchá nebo poměrně složitá a může zahrnovat více proměnných a parametrů. Například pro teplotní senzor může být funkce $f(x)$ definována jako lineární vztah mezi vstupní teplotou a výstupním napětím, jako je:

$$y = mx + b, \tag{2.15}$$

kde m je sklon vztahu a b je průsečík. V jiných případech může být funkce f složitější a nelineární, může také zahrnovat více stupňů zpracování a převodů. Například u fotoelektrického senzoru může funkce f zahrnovat konverzi intenzity světla na elektrický signál, který je pak zpracován za účelem vytvoření digitálního výstupu. Konkrétní tvar rovnice bude záviset na konkrétním senzoru a aplikaci. Samotný signál má dvě základní složky – informační parametr (hodnotovou složku) a časový

parametr (časovou složku). Tyto složky mohou být spojité (nekonečné množině hodnot) nebo diskrétní (občas označovány jako nespojité, s konečným počtem hodnot). Signál, který má spojitý čas i hodnoty se nazývá analogový. Signál, který má nespojitý čas a nespojité hodnoty se nazývá diskrétní. Úprava analogové signálu je nutná pro umožnění práce s takovým signálem v digitální podobě, jelikož je nepraktické v rámci digitálního prostředí pracovat s nekonečnou množinou hodnot. K převodu mezi analogovým a diskrétním signálem se využívá dvou základních metod – vzorkování a kvantování. Vzorkování v jednoduchosti zahrnuje odběr spojitého signálu a jeho převedení na signál s diskrétním časem měření hodnoty signálu v diskrétních časových intervalech. Výsledkem je sekvence diskrétních hodnot, z nichž každá představuje amplitudu signálu v odpovídajícím časovém okamžiku. Kvantování v jednoduchosti zahrnuje zaokrouhlování vzorkovaných hodnot na konečný počet kvantizačních úrovní. Tento proces lze považovat za formu komprese, protože spojitý signál je zredukován na konečný počet diskrétních hodnot. Přesnost kvantizačního procesu závisí na počtu použitých kvantizačních úrovní, přičemž větší počet úrovní vede k vyšší úrovni přesnosti. Existuje mnoho různých typů senzorů, které se používají v různých aplikacích. Některé z nejběžnějších typů senzorů jsou: snímače teploty, snímače tlaku, snímače hladiny, snímače rychlosti, snímače polohy, snímače vlhkosti, snímače tekutin, snímače světla, snímače gravitace, akustické snímače, chemické snímače, biologické snímače, snímače vibrací, magnetické snímače, optické snímače, infračervené snímače, snímače pH či senzory z optických vláken. Při používání zmíněných senzorů je pak důležitá také jejich přesnost (těsnost shody mezi naměřenou hodnotou veličiny a pravou hodnotou veličiny měřené veličiny) a preciznost (těsnost shody mezi indikacemi nebo naměřenými hodnotami veličiny získanými opakovanými měřeními na stejném objektu nebo na podobných objektech za specifikovaných podmínek). Přesnost a preciznost jsou dva důležité pojmy v průmyslových řídicích systémech a výrobních procesech. Přesnost (P_{acc}) lze matematicky vyjádřit jako:

$$P_{acc} = \frac{|V_{akt} - V_{mer}|}{V_{akt}} \tag{2.16}$$

kde V_{akt} představuje skutečnou hodnotu a V_{mer} představuje naměřenou hodnotu. Preciznost (P_{pre}) na druhé straně lze matematicky znázornit takto:

$$P_{pre} = \frac{SD}{\bar{V}_{mer}} \tag{2.17}$$

kde SD představuje standardní odchylku naměřených hodnot a $\bar{V}_{mer}$ představuje průměr naměřených hodnot. V kontextu OT je velmi důležité mít přesná a precizní měření ze senzorů. Přesná měření zajišťují, že se pro účely kontroly a rozhodování používají správné hodnoty, zatímco precizní měření zajišťují, že výsledky jsou

konzistentní a spolehlivé. Údržba a kalibrace jsou pak klíčové procesy, které pomáhají zajistit přesnost a preciznost měření senzorů. Pravidelná údržba pomáhá identifikovat a opravit jakékoli problémy, které mohou nastat, jako je opotřebení nebo poškození senzoru. Kalibrace na druhé straně zahrnuje nastavení snímače, aby bylo zajištěno, že měří přesně a precizně. Tyto procesy pomáhají udržovat kvalitu a spolehlivost měření senzorů v průběhu času, což je nezbytné pro efektivní řízení a rozhodování v těchto odvětvích. Z pohledu připojení je pak tedy nutno zmínit napojení senzorů na tři základní části OT modelu – (i) připojení k procesu, (ii) připojení ke kontroléru, a (iii) připojení ke vzdálené diagnostice a údržbě:

- Připojení k procesu, resp. detekce změn (měření), probíhá pak dvěma základními metodami – přímé měření a nepřímé měření. Přímá metoda měření zahrnuje přímou fyzickou interakci mezi senzorem a měřenou veličinou, jako je teplota, tlak nebo intenzita světla. Například teploměr měří teplotu přímým měřením změny teploty na snímacím prvku teploměru a převádí ji na elektrický signál. Nepřímé metody měření zahrnují nepřímou fyzickou interakci mezi senzorem a měřenou veličinou. Například měření posunu pístu v hydraulickém systému pro stanovení tlaku. Poloha pístu se pak používá k odvození tlaku kapaliny v systému, ale tlak se přímo neměří. Místo toho je tedy převedena na elektrický signál pouze poloha pístu.
- Ke kontroléru jsou senzory připojeny prostřednictvím různých komunikačních protokolů, jako jsou digitální protokoly (např. USB, Ethernet, RS-232, RS-485, CAN a Modbus) nebo analogové signály (např. 4-20 mA, 0- 10V a 1-5V). Volba komunikačního protokolu závisí na požadavcích aplikace či senzoru.
- Pro vzdálenou diagnostiku mohou být senzory připojeny ke vzdálenému systému prostřednictvím kabelového nebo bezdrátového připojení, popř. přes cloud v rámci sítě internetu věcí (Internet of Things, IoT), popř. průmyslovému internetu věcí (Industrial Internet of Things, IIoT), a přistupovat k nim vzdáleně prostřednictvím webového rozhraní nebo mobilní aplikace. IoT označuje síť fyzických zařízení (např. právě senzorů), která těmto objektům umožňuje shromažďovat a vyměňovat si data. Průmyslový internet věcí IIoT je podmnožinou internetu věcí, která konkrétně odkazuje na použití technologií IoT v průmyslovém prostředí, jako jsou výrobní závody a energetické sítě. Senzory hrají klíčovou roli v IoT i IIoT tím, že zachycují data z fyzického světa a přenášejí je do sítě ke zpracování a analýze. Pokud propojujeme více senzorů mezi sebou, vytváříme tak tzv. senzorickou síť, např. za účelem sdílení dat senzorů mezi sebou. S růstem IoT a IIoT jsou stále důležitější možnosti vzdálené diagnostiky. Připojením senzorů a dalších zařízení k síti je možné monitorovat a diagnostikovat problémy na dálku, což snižuje potřebu návštěv na místě a umožňuje rychlejší řešení problémů.

2.2.3 Aktuátor

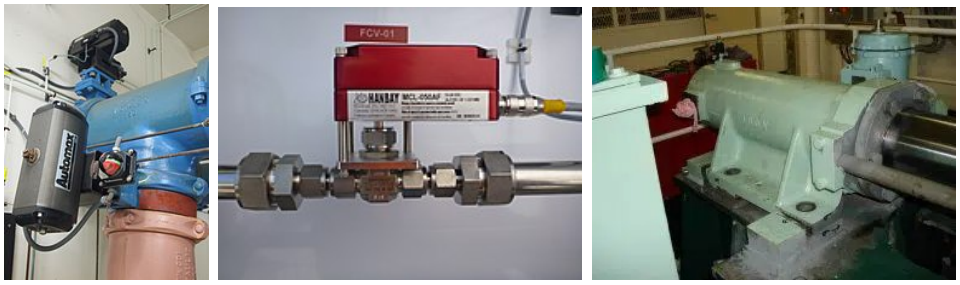
Aktuátory, také nazývané jako akční členy, slouží jako prostředek pro realizaci řídicích akcí v závislosti na výstupech z kontroléru. Je nutno zmínit, že hrají klíčovou roli při zajišťování bezpečného, efektivního a spolehlivého provozu průmyslových procesů, a jsou klíčové pro úspěšné řízení průmyslových procesů, jelikož bez nich by systém nemohl provádět žádné fyzické změny. Fungují na principu přeměny elektrických signálů na fyzické akce a umožňují tak regulovat a řídit průmyslové procesy v reálném čase, rychle reagovat na měnící se podmínky a zajistit, aby výkonnost procesu zůstala ve stanovených mezích. V jednoduchosti si aktuátor lze představit jako fyzické rozhraní mezi kontrolérem a procesem, které umožňuje provádět úpravy procesních proměnných, příkladem může být otevírání nebo zavírání ventilů, spouštění nebo zastavování čerpadel či nastavení úrovně teploty nebo tlaku. Stručně řečeno, hlavní funkcí akčního členu je přijímat elektrické signály z kontroléru a převádět je na fyzické akce, což umožňuje OT manipulovat s procesem a dosahovat požadovaných řídicích cílů. Kontrolér může například vyslat signál do pohonu, aby otevřel ventil, a ovladač v odezvě ventil fyzicky otevře. Aktuátory lze velmi jednoduše popsat pomocí základních lineárních modelů, pro příklad jako:

$$y = kx + b, \tag{2.18}$$

kde y je výstupní veličina (například pozice), x je vstupní signál (například napětí), k je tzv. tuhost a b je offset, nebo v případě nelineárních modelů:

$$y = ax^2 + bx + c, \tag{2.19}$$

kde y je výstupní veličina, x je vstupní signál, a, b a c jsou konstanty, které se určují experimentálně nebo na základě teoretických výpočtů. Tyto modely mohou být upraveny k zohlednění řady specifických vlastností aktuátoru, jako jsou například ztráty, hystereze, šum a další. Z pohledu typu energie, který využívá aktuátor k realizaci fyzické akce, můžeme aktuátory dále dělit na pneumatické, elektrické a hydraulické (a pro úplnost mechanické), viz obrázek níže.



Obr. 2.4: Různé typy aktuátorů, zleva pneumatický, elektrický a hydraulický.

Pneumatické aktuátory využívají tlakového plynu (vzduch) jako zdroj energie. Tyto aktuátory se často používají v průmyslových procesech, kde je nutná rychlá reakce. Pro kontinuální procesy lze popsat pneumatický aktuátor diferenciální rovnicí, uveďme pro příklad rovnici tlumiče:

$$dx/dt + bx = f(t), \tag{2.20}$$

kde x je poloha, b je koeficient tlumení, $f(t)$ je vnější síla a dx/dt je derivace polohy v čase. V diskrétním procesu pak můžeme uvést základní rovnici jako:

$$x(k+1) = x(k) + T \cdot (-bx(k) + u(k)), \tag{2.21}$$

kde $x(k)$ je pozice tlumiče v kroku k, T je velikost kroku, $u(k)$ je vstup, a b je konstanta reprezentující aktuátor.

Elektrické aktuátory využívají elektrickou energii jako zdroj energie. Tyto aktuátory se často používají v moderních průmyslových procesech, kde je potřeba přesného řízení a snadné ovladatelnosti. Pro příklad, uveďme si pro elektrický aktuátor, motor, popis pomocí diferenciální rovnice v rámci kontinuálního procesu:

$$L\frac{di}{dt} + Ri = V \ - Kx \tag{2.22}$$

kde i je proud, L je indukčnost, R je odpor, V je napětí, K je konstanta proporcionality mezi polohou a sílou a x je poloha. V diskrétním procesu pak můžeme uvést základním rovnici jako:

$$x(k+1) = x(k) + T \cdot (-kx(k) + u(k)), \tag{2.23}$$

kde $x(k)$ je pozice elektrického aktuátoru v rámci kroku, T je velikost kroku, $u(k)$ je vstup, a k je konstanta reprezentující dynamiku systému.

Hydraulické aktuátory využívají tlak oleje jako zdroj energie. Tyto aktuátory se často používají v průmyslových procesech, kde je nutná velká síla a účinnost. Z pohledu obecného popisu v rámci kontinuálního systému pro hydraulický aktuátor, píst, můžeme uvažovat opět diferenciální rovnici:

$$A \ \cdot dp/dt + (B \cdot p)/V \ = F/A, \tag{2.24}$$

kde A je plocha válce, B je koeficient tlumení, p je tlak, V je objem válce, F je síla a dp/dt je derivace tlaku v čase. Pokud uvažujeme diskrétní systém, pak lze uvažovat rovnici:

$$x(k+1) = x(k) + T \cdot (-k_h \cdot x(k) + u(k)), \tag{2.25}$$

kde $x(k)$ je pozice pístu v daném kroku, T je velikost kroku, k_h je konstanta reprezentující dynamiku pístu, a $u(k)$ je vstup.

Samozřejmě existují dále také hybridní aktuátory, které kombinují více výše zmíněných typů dohromady, příkladem může být termohydraulický elektrický aktuátor, využívaný např. v systémech horké vody apod. Samozřejmostí je pak tyto druhy aktuátorů rozlišit dále pro úplnost dle směru fyzického pohybu, kdy základně rozlišujeme lineární, rotační či vertikální. Z pohledu připojení k jednotlivým částem OT modelu jsou aktuátory připojeny velmi obdobně jako senzory, tedy:

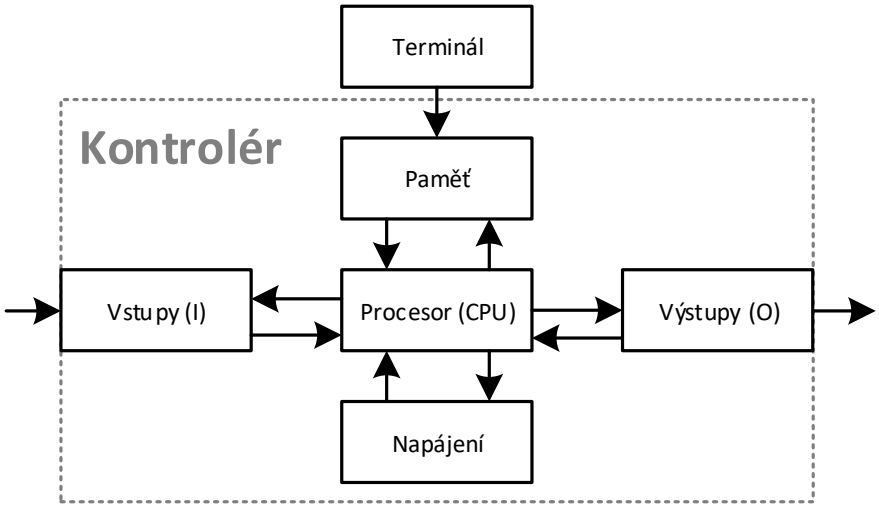
- Připojení ke kontroléru se provádí prostřednictvím různých komunikačních protokolů, jako jsou digitální protokoly nebo analogové signály. Volba komunikačního protokolu opět závisí na požadavcích aplikace a aktuátoru.
- Připojení k procesu je realizováno pomocí elektrického nebo pneumatického signálu, který přenáší informace o požadavcích aplikace na aktuátor. Aktuátor následně ovlivňuje proces pomocí svého pohybu nebo změny stavu.
- Vzdálená diagnostika a údržba se provádí pomocí bezdrátového připojení nebo kabelového připojení ke vzdálenému systému, popř. přes cloud (IoT/IIoT).

Také je nutno zmínit, že je třeba vždy zvážit specifické požadavky a potřeby řízeného procesu při volbě vhodného typu procesu. Samotný proces je v rámci OT modelu tedy řízen pomocí kontroléru přes aktuátor, který již provádí fyzickou akci dle přijatého elektrického signálu a monitorování procesu přes senzor, který dodává kontroléru informace pro řízení.

Při navrhování aktuátoru je zásadní brát v úvahu proces, který je třeba řídit, a požadavky na jeho řízení. To zahrnuje sílu a přesnost ovládání – kolik síly a přesnosti bude aktuátor potřebovat k efektivnímu řízení procesu. Údržba a spolehlivost pohonu jsou také důležité faktory, které je třeba zvážit, příkladem může být servis a závazek údržby a dostupnost náhradních dílů a podpory. Bezpečnost je dalším zásadním aspektem, který je třeba zvážit – zda je aktuátor během používání bezpečný pro lidi a zařízení. Rovněž je třeba brát v úvahu provozní podmínky – zda bude pohon používán v náročných prostředích, jako jsou vysoké teploty nebo tlaky, a zda může za těchto podmínek dobře fungovat. Při výběru nebo používání pohonu je třeba zvážit několik problémů, včetně nákladů, údržby a spolehlivosti, dostupnosti náhradních dílů a podpory, bezpečnosti a provozních podmínek. Některé pohony mohou být drahé na nákup a údržbu, zatímco jiné mohou vyžadovat vyšší úroveň údržby a být méně spolehlivé. Některé mohou mít omezenou dostupnost náhradních dílů nebo podpory, což ztěžuje jejich použití. Další mohou navíc představovat bezpečnostní riziko pro lidi a zařízení během používání a jiné nemusí dobře fungovat v náročných provozních podmínkách. Zvážení všech těchto faktorů je zásadní při výběru správného pohonu pro daný proces, aby byl zajištěn optimální výkon a ovládání.

2.2.4 Kontrolér

Kontrolér je centrální součástí průmyslových systémů a hraje klíčovou roli v automatizaci procesů. Zodpovídá za příjem vstupu ze senzorů a použití informací k ovládání akčních členů. Kontrolér je v podstatě „mozkem“ systému, který rozhoduje na základě aktuálního stavu procesu a provádí akce pro realizaci požadovaného stavu. Celkově je, jak již bylo řečeno, kontrolér kritickou součástí jakékoli průmyslového řídicího systému, protože je odpovědný za zajištění hladkého a efektivního chodu procesu, a za provádění úprav podle potřeby k udržení optimálního výkonu. Jednoduchý příklad a základní schéma pro kontrolér je vidět na obrázku níže.



Obr. 2.5: Ukázka základního schématu kontroléru [35].

Kontrolér se tak tedy skládá z následujících základních součástí:

- **Napájecí zdroj** poskytuje energii potřebnou pro provoz ovladače. V závislosti na konstrukci kontroléru může být napájecí zdroj interní nebo externí.
- **Vstupní modul** je zodpovědný za příjem signálů ze senzorů a dalších vstupních zařízení. Může to být analogový vstupní modul pro příjem analogových signálů nebo digitální vstupní modul pro příjem digitálních signálů.
- **Výstupní modul** je zodpovědný za ovládání akčních členů a dalších výstupních zařízení. Stejně jako vstupní modul, může být v závislosti na typu ovládaného výstupního zařízení analogový nebo digitální.
- **Procesor** je „mozkem“ řadiče. Je zodpovědný za provádění řídicích algoritmů a rozhodování na základě vstupních signálů.

- **Paměť** se používá k ukládání řídicích algoritmů a dalších dat potřebných pro procesor. Může být energeticky závislá (RAM) nebo energeticky nezávislá (ROM nebo flash paměť) v závislosti na požadavcích systému.
- **Uložiště dat** se používá k ukládání historických dat, konfiguračních informací a dalších dat, která je třeba zachovat, i když je kontrolér vypnutý.
- **Firmware** je software, který běží na řídicí jednotce a poskytuje řídicí logiku a další funkce potřebné pro provoz systému.
- **Programovací rozhraní** se používá k programování a konfiguraci kontroléru. Může to být fyzické rozhraní (jako je port USB) nebo vzdálené rozhraní (jako je webové rozhraní) v závislosti na návrhu ovladače.

Vývoj kontrolérů (řídicích jednotek) pro průmyslovou automatizaci se postupem času vyvíjel. Za jeden z starších způsobů ovládání mohou být považovány kontroléry s relé logikou (Relay Logic Controller, RLC). RLC používají elektromechanická relé k řízení procesů, přičemž relé jsou propojena dohromady a tvoří kompletní řídicí systém. Tento typ je však značně omezen z hlediska výpočetního výkonu a paměti, nicméně je jednoduchý a spolehlivý.

Jak můžeme vidět samotné systémy RLC nabízely sice spolehlivost a jednoduchost, nicméně byly značně nevhodné pro složité systémy, které se v rámci historie každým rokem rozšiřovaly. Z tohoto důvodu byl vytvořen nový přístup, a to přes programovatelné logické kontroléry (Programmable Logic Controllers, PLC), které byly představeny jako pokročilejší a flexibilnější alternativa k RLC. PLC postupně nahrazovaly elektromechanická relé, což umožnilo složitější řídicí algoritmy a větší výpočetní výkon. PLC také přidaly možnost ukládat a vyvolávat programy, což usnadňovalo jejich úpravy a aktualizace. PLC jsou dnes široce používány v automatizaci průmyslových procesů, jako jsou montážní linky, roboty a další průmyslové stroje. PLC jsou navrženy tak, aby zvládaly různé vstupy a výstupy, včetně digitálních a analogových signálů, a mohly řídit řadu průmyslových procesů. Obvykle zahrnují vstupní/výstupní (I/O) moduly, centrální procesorovou jednotku (CPU), paměť a programovací rozhraní. PLC lze programovat pomocí různých programovacích jazyků, například pomocí grafická liniová logika (Ladder Logic, LD) nebo za pomocí strukturovaný text (Structured Text, STX). Obvykle se však programují pomocí softwarové aplikace běžící na počítači.

Zároveň postupně vznikaly požadavky (převážně v rámci energetiky, ale také v jiných průmyslových oblastech) na vzdálená řízení, což umožnilo vznik tzv. vzdálené koncové jednotky (Remote Terminal Unit, RTU), která byla vyvinuta jako způsob vzdáleného řízení procesů (ne nutně, nicméně často i ve venkovním prostředí). RTU se původně používaly v telekomunikacích a veřejných službách, ale od té doby byly přijaty i v jiných průmyslových odvětvích. RTU se obvykle používají k monitorování a ovládání vzdálených zařízení, jako jsou potrubí nebo větrné turbíny. Z tohoto

pohledu je nutné ještě zmínit modulární terminálové jednotky (Modular Terminal Unit, MTU), což jsou jednotky, které jsou navrženy pro použití v aplikacích, kde je vyžadováno více řídicích funkcí. Jednotky MTU se skládají z modulů, které lze snadno přidat nebo odebrat, což umožňuje větší flexibilitu a škálovatelnost. V rámci kontextu s RTU se používají pro jejich řízení a ve většině případů jsou před samotným kontrolním centrem. RTU/MTU obvykle zahrnují obdobné části jako PLC, tedy vstupní/výstupní (I/O) moduly, centrální procesorovou jednotku (CPU), paměť a programovací rozhraní. Ve srovnání s PLC pak jsou obvykle rozměrově menší, mají nižší spotřebu energie a jednodušší design, díky čemuž jsou vhodné pro použití ve vzdálených místech, kde je omezený výkon, a prostor je na prvním místě. Většinou však také nemají stejnou úroveň výpočetního výkonu, kapacity paměti a rozšiřitelnosti jako PLC, které jsou určeny pro složitější řídicí aplikace. V neposlední řadě mají RTU i MTU velmi často bezdrátové rozhraní (pro příjem a odesílání dat), které dovoluje komunikovat či připojit zařízení i tam, kde není plná komunikační infrastruktura.

Omezení RLC (nízká úroveň programovatelnosti a neschopnost udržovat či přenášet velké množství dat) a PLC (složité a nákladné) vyústila ve vytvoření tzv. programovatelných řídicích relé (Programmable Controller Relay, PCR), občas také pouze jako chytré relé nebo programovatelné relé. PCR byly navrženy tedy tak, aby nabízely cenově výhodné řešení pro jednoduché řídicí aplikace, které vyžadovaly pokročilejší programovací schopnosti než RLC, a aby byly menší a cenově dostupnější než PLC, ale přesto nabízely možnost programovat složité řídicí sekvence. Zavedení PCR umožnilo automatizaci širšího spektra průmyslových aplikací a poskytlo pro tyto aplikace nákladově efektivnější řešení. PCR většinou poskytují základní (jednodušší) řídicí funkce, jako je přepínání, časování a počítání. PCR tak nabízí kombinaci výhod RLC (jednoduchost a cenová dostupnost) a PLC (programovatelnost). PCR má však i některá omezení. Za prvé, má omezený výpočetní výkon ve srovnání s PLC. To znamená, že je méně vhodný pro složité řízení, které vyžaduje velký výpočetní výkon. Dalším omezením je omezený počet vstupů a výstupů, které může PCR zpracovat. To může být problematické v aplikacích, kde je třeba zpracovat velké množství dat a reagovat na ně v reálném čase. A konečně, PCR může být omezenější, pokud jde o možnosti programování, ve srovnání s PLC, což může ztížit návrh a údržbu (a to zejména u složitějších aplikací).

2.2.5 Rozhraní člověk-stroj

Pro začátek si definujme základní oblast zaměřenou na technologický design z pohledu potřeb člověka. Jako první je nutné představit termín uživatelské rozhraní (User Interface, UI). UI odkazuje na vizuální a interaktivní prvky produktu nebo služby, se kterými uživatel interaguje, jako jsou tlačítka, ikony a text. Cílem návrhu UI je, aby interakce mezi uživatelem a produktem/službou byla co nejintuitivnější a nejefektivnější. Rozhraní lze v tomto smyslu pak chápat ve své podstatě jakékoliv rozhraní mezi uživatelem a jakýmkoliv softwarem či hardwarem. Nad rámec UI je nutno brát také v úvahu celkový dojem z produktu, a to jak splňuje potřeby a očekávání uživatele. Z tohoto pohledu pak definujeme tzv. uživatelskou zkušenost (User Experience, UX), zahrnující i UI. UX si klade za cíl v rámci designu zohledňovat potřeby a cíle uživatele a poskytovat mu bezproblémový a příjemný zážitek z používání takového designu. Pokud bychom se posunuli o úroveň výše, definujeme pak již tzv. zákaznickou zkušenost (Customer Experience, CX), tedy celkovou zkušenost, kterou má zákazník se společností a jejími produkty/službami, včetně UX a dalších interakcí se společností, jako je zákaznický servis a marketing. Cílem je tedy vytvořit pozitivní, trvalý vztah mezi zákazníkem a společností. V neposlední řadě pak končíme vlastní návrh v rámci zkušenosti se značkou (Brand Experience, BX), kde již zahrnujeme všechny interakce, které má osoba se značkou, včetně CX a dalších kontaktních bodů, jako je reklama, vztahy s veřejností a firemní identita. Cílem pak je tedy vytvořit konzistentní a nezapomenutelný zážitek napříč všemi kontaktními body a vybudování silného uznání značky a loajality. Ve své podstatě se jedná u UI, UX, CX a BX o jednotlivé nadřazené části designu [21], zahrnující do sebe i socio-ekonomické aspekty, čímž tzv. tvarují finální vlastnosti produktu, např. pomocí zvyšování uživatelské přívětivosti, apod. Platí tedy, že:

$$UI \subset UX \subset CX \subset BX \tag{2.26}$$

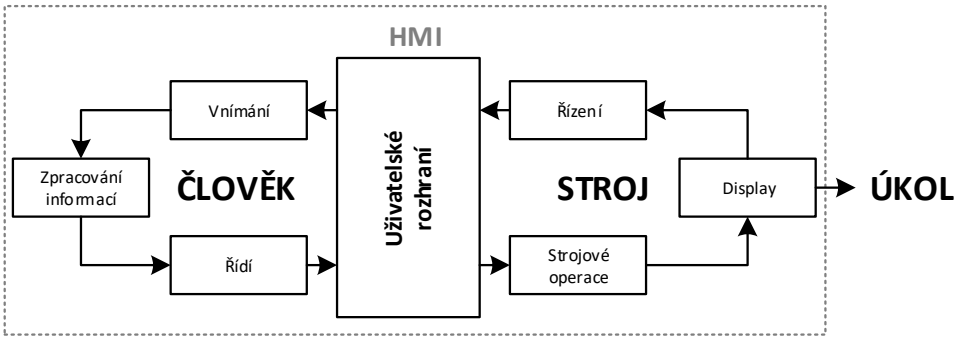
Z pohledu OT se však budeme primárně zaměřovat na pohled UI. Zde existuje celá řada vrstev UI, jako znakové uživatelské rozhraní (Character User Interface, CUI), textové uživatelské rozhraní (Textual User Interface, TUI) či rozhraní příkazového řádku (Command Line User Interface, CLI), aj. Jak názvy napovídají, tak jednotlivé rozhraní vycházejí ze vstupů. Dále také velká část vrstev vychází např. z jednotlivých smyslů: hmatové UI, zrakové UI, sluchové UI, čichové UI, balanční UI, či chuťové UI. Pokud existuje kombinace daných smyslů, pak se jedná o tzv. kompozitní UI (Composite User Interface, CUI), příkladem může být grafické uživatelské rozhraní (Graphical User Interface, GUI), které často kombinuje hmat (klávesnice či dotykovou obrazovku) a zrak (vizuální resp. grafická podoba zobrazována na obrazovce). Dále existují i další UI rozhraní, kde za zmínku zcela jistě

stojí UI virtuální reality (Virtual Reality Interface, VRI), UI rozšířené reality (Augmented Reality Interface, ARI), UI využívající hologramu (Hologram User Interface, HUI), ale i spousty dalších. Z pohledu OT se však budeme věnovat převážně tzv. rozhraní člověk-stroj (Human-Machine Interface, HMI), což je dle definice rozhraní v jakékoliv formě používané člověkem přímo k interakci se strojem (např. ve formě řízení). Záměnu definice člověka (HMI) za uživatele (UI) můžeme považovat za náhodu i záměr současně, jelikož se zde nejedná o obecného uživatele, ale skutečně o člověka a nikoliv např. uživatele, kterým je zvíře (jednoduchý příklad, viz Kanzi, opice schopné ovládat tablet [42], viz obr. 2.6).

Obr. 2.6: Vlevo ukázka nezcela běžného uživatele v rámci UI a vpravo ukázka běžného uživatele-člověka v rámci HMI [42, 26].

Terminologie týkající se interakce člověk-stroj se postupem času vyvíjela, aby se stala inkluzivnější a přesnější. Nejstarší termín, HCI (Human Computer Interface), odrážel dominantní zaměření na počítačovou technologii a ignoroval roli člověka v interakci. To se změnilo se zavedením CHI (Computer Human Interface), který uznal důležitost jak člověka, tak počítače v interakci. Tento termín byl však stále omezený tím, že plně nezohledňoval vliv pohlaví a dalších faktorů diverzity. Byl zaveden termín MMI (Man Machine Interface), ale čelil kritice za to, že je genderově zaujatý a exkluzivní. Nakonec se jako správný jazyk objevil termín HMI (Human Machine Interface), který zahrnuje mužské i ženské operátory, a uznává důležitost lidského prvku v interakci, tak jako na obr. níže. HMI tak označuje tedy rozhraní nebo vrstvu rozhraní mezi lidskými operátory a stroji, systémy nebo zařízeními, které ovládají nebo s nimi komunikují. HMI poskytuje operátorům (lidem) vizuální a intuitivní prostředky pro monitorování, ovládání a interakci se základními systémy. Toho lze dosáhnout různými prostředky, jako jsou grafické displeje, dotykové obrazovky, tlačítka, spínače a další vstupní/výstupní zařízení. Cílem návrhu HMI je, aby interakce mezi člověkem a strojem byla co nejúčinnější, bezpečná a uživatelsky přívětivá. Ná-

vrh HMI by měl brát v úvahu potřeby a omezení lidského operátora, včetně úvah, jako je viditelnost, použitelnost a bezpečnost. HMI by také mělo být přizpůsobitelné a konfigurovatelné, aby vyhovovalo potřebám různých uživatelů, aplikací a prostředí. Vývoj HMI vedl k vývoji pokročilých systémů, které dnes již zahrnují pokročilé technologie, jako je strojové učení, počítačové vidění a zpracování přirozeného jazyka, aby se dále zlepšila interakce člověk-stroj.



Obr. 2.7: Schématické zobrazení interakce mezi člověkem a strojem (HMI) [9].

V kontextu OT nás však zajímá nejvíce rozhraní HMI, které je však dnes velmi často, nicméně chybně, zaměňováno s grafickým uživatelským rozhraním (GUI, Graphical User Interface). Obě tyto rozhraní spadají pod obecnou skupinu UI, nicméně jak je vidět z popisu výše, tak se nejedná o synonyma. HMI může být tvořeno různým způsobem v rámci průmyslu a jedním z nich je pomocí GUI, nicméně dalšími příklady může být např. s terminálem operátorského rozhraní (Operator Interface Terminal, OIT). Rozdíl mezi HMI s OIT a HMI s GUI[1] můžeme názorně vidět na obr. níže.

Přejděme nyní k základním částem systému HMI, mezi které tedy patří (viz obrázek níže):

- **Software** je hlavní komponenta, která poskytuje operátorovi vizuální rozhraní pro interakci se strojem. Obvykle se skládá z několika modulů, jako je grafický modul, reportovací modul, konfigurační modul a síťový modul.
- **Operační systém** spravuje vstupní a výstupní (I/O) komponenty HMI a poskytuje platformu pro běh softwaru. Pokud není k dispozici žádný operační systém, musel by se systém HMI při plnění svých funkcí spoléhat na vyhrazenou hardwarovou platformu. Tato hardwarová platforma by musela zahrnovat dostatečný výpočetní výkon a paměť pro spuštění softwaru HMI a také nezbytné vstupní/výstupní (I/O) porty pro připojení k monitorovanému nebo řízenému

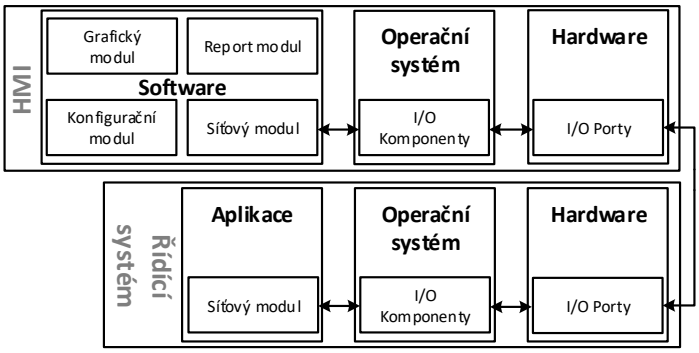
[1]HMI od firmy MapleSystems OIT3160-B00 a HMI5070B.

Obr. 2.8: Vlevo ukázka HMI bez GUI s OIT a vpravo ukázka HMI s GUI.

stroji. Hardwarová platforma by také mohla zahrnovat další komponenty, jako jsou grafické procesory, dotykové obrazovky a komunikační rozhraní pro podporu uživatelského rozhraní HMI a konektivitu s jinými zařízeními. V tomto případě by byl software HMI navržen tak, aby běžel přímo na hardwaru, bez potřeby operačního systému. Tento přístup může být prospěšný v některých průmyslových aplikacích, kde je kritický výkon a spolehlivost v reálném čase. Může však mj. omezit flexibilitu a škálovatelnost systému HMI a také zvýšit jeho složitost a náklady.

- **Hardware** poskytuje fyzické rozhraní mezi HMI a strojem (např. kontrolér, ale také senzor, aktuátor, apod.). Obvykle zahrnuje I/O porty, které se připojují k řídicímu systému stroje a poskytují prostředky pro přenos dat mezi těmito dvěma systémy.

Spojení mezi HMI a řídicím systémem stroje se obvykle vytváří prostřednictvím hardwarových I/O portů. Software HMI je zodpovědný za prezentaci dat a informací operátorovi, zatímco řídicí systém (v tomto kontextu bez ohledu jaký) je zodpovědný za řízení procesů stroje. HMI poskytuje operátorovi pohodlné rozhraní pro interakci se strojem, sledování jeho výkonu a provádění úprav podle potřeby.



Obr. 2.9: Jednotlivé bloky HMI [51].

2.2.6 Vzdálená diagnostika a údržba

Vzdálená diagnostika a údržba označují schopnost vzdáleně monitorovat a řídit různé aspekty průmyslových systémů, včetně schopnosti diagnostikovat a opravovat problémy, provádět změny a aktualizace softwaru a provádět běžné úkoly údržby. To se provádí s cílem minimalizovat prostoje a zajistit, aby průmyslové systémy fungovaly na své optimální úrovni účinnosti. V průmyslových řídicích systémech je vzdálená diagnostika a údržba často dosahována pomocí technologií vzdálené komunikace. Ty umožňují vzdálený přístup k řídicím systémům, které lze připojit k podnikové síti, internetu nebo cloudu. To umožňuje vzdálené monitorování a ovládání systémů, stejně jako možnost přenášet data a provádět diagnostické a údržbové úkoly ze vzdáleného místa. Jednou z klíčových výhod vzdálené diagnostiky a údržby je schopnost zkrátit prostoje, protože problémy lze diagnostikovat a opravit dříve, než povedou k úplnému selhání systému. To může také snížit potřebu údržby a oprav na místě, které mohou být časově náročné a drahé. Kromě toho může vzdálená diagnostika a údržba pomoci zlepšit celkovou účinnost a spolehlivost průmyslových systémů, a také minimalizovat riziko lidské chyby během údržby a oprav. Z hlediska implementace lze vzdálenou diagnostiku a údržbu integrovat do řídicích systémů pomocí specializovaných softwarových a hardwarových nástrojů, jako jsou nástroje pro vzdálený přístup, nástroje pro správu sítě a systémy vzdáleného monitorování. Tyto nástroje lze použít k monitorování stavu systémů, diagnostice problémů a provádění údržby a oprav na dálku, bez nutnosti zásahu na místě. Vzdálená diagnostika a údržba se týká schopnosti monitorovat, diagnostikovat a provádět údržbu řídicího systému ze vzdáleného místa. Mezi klíčové součásti systému vzdálené diagnostiky a údržby patří:

- **Vzdálený přístup** resp. možnost přístupu k řídicímu systému ze vzdáleného místa, obvykle prostřednictvím zabezpečeného připojení pomocí notebooku, tabletu nebo chytrého telefonu.
- **Monitorování** využívá senzorů a dalších monitorovacích zařízení pro sběr dat o výkonu a stavu řídicího systému. Tato data lze poté analyzovat a použít k identifikaci potenciálních problémů a zlepšení výkonu systému.
- **Diagnostika** je použití softwarových nástrojů k diagnostice problémů s řídicím systémem, jako je identifikace vadných součástí, zjišťování softwarových chyb nebo určení příčiny problémů s komunikací.
- **Údržba** se vztahuje k použití vzdáleného přístupu k provádění úkolů údržby na řídicím systému, jako je aktualizace softwaru, konfigurace zařízení nebo výměna vadných součástí.
- **Reporting** zahrnuje generování reportů na základě dat shromážděných monitorovacími a diagnostickými nástroji, které poskytují pochopení stavu systému.

2.3 Komunikační techniky a technologie v OT

2.3.1 Komunikační techniky

Většina průmyslových komunikačních protokolů není přímo založena ani na sedmivrstvém modelu ISO/OSI, ani na čtyřvrstvém modelu TCP/IP. Místo toho mají často svůj vlastní komunikační model a architekturu protokolu navrženou speciálně pro potřeby průmyslových automatizačních a řídicích systémů. Zatímco koncepty a principy modelů ISO/OSI a TCP/IP mohou být v těchto protokolech přítomny, jsou obvykle implementovány přizpůsobenějším a optimalizovaným způsobem, který bere v úvahu specifické požadavky průmyslových sítí, jako jsou data v reálném čase, přenos, determinismus a spolehlivost. Například mnoho průmyslových komunikačních protokolů používá zjednodušenou nebo upravenou verzi vrstvy datového spojení a fyzické vrstvy z modelu ISO/OSI a nemusí zahrnovat vyšší vrstvy, jako jsou vrstvy relace, prezentace a aplikace. To pomáhá snížit režii a zajistit rychlou a efektivní komunikaci v řídicích systémech v reálném čase. Na obrázku níže můžeme vidět, jak jsou některé průmyslové protokoly vsazeny v rámci modelů ISO/OSI či TCP/IP.

<table>
<tr><th colspan="2">OSI Model</th><th>Protokoly:</th><th colspan="2">TCP/IP Model</th></tr>
<tr><td rowspan="3">Data</td><td>Vrstva 7
Aplikační vrstva</td><td>Modbus, DeviceNet,
Ethernet/IP, ...</td><td rowspan="3">Vrstva 4
Aplikační vrstva</td><td rowspan="3">Data</td></tr>
<tr><td>Vrstva 6
Prezenční vrstva</td><td>Kompresní algoritmy</td></tr>
<tr><td>Vrstva 5
Relační vrstva</td><td>NFS, SQL, SMB, RPC,
P2P, SCP, SDP, SIP, ...</td></tr>
<tr><td>Segmenty</td><td>Vrstva 4
Transportní vrstva</td><td>TCP / UDP</td><td>Vrstva 4
Transportní vrstva</td><td>Segmenty</td></tr>
<tr><td>Pakety</td><td>Vrstva 3
Síťová vrstva</td><td>IP (IPv4, IPv6, ARP,
IGMP, ICMP, ...)</td><td>Vrstva 3
Síťová vrstva</td><td>Pakety</td></tr>
<tr><td>Vrstva 2
Linková vrstva</td><td>Vrstva 2
Linková vrstva</td><td>Ethernet, ...</td><td rowspan="2">Vrstva 1
Síťové rozhraní</td><td rowspan="2">Bity a rámce</td></tr>
<tr><td>Vrstva 1
Fyzická vrstva</td><td>Vrstva 1
Fyzická vrstva</td><td>RS-232, UTP kabely
(CAT5, CAT6), ...</td></tr>
</table>

Obr. 2.10: Ukázka zjednodušeného pohledu na protokoly v rámci průmyslu v kontextu modelů ISO/OSI a TCP/IP [45].

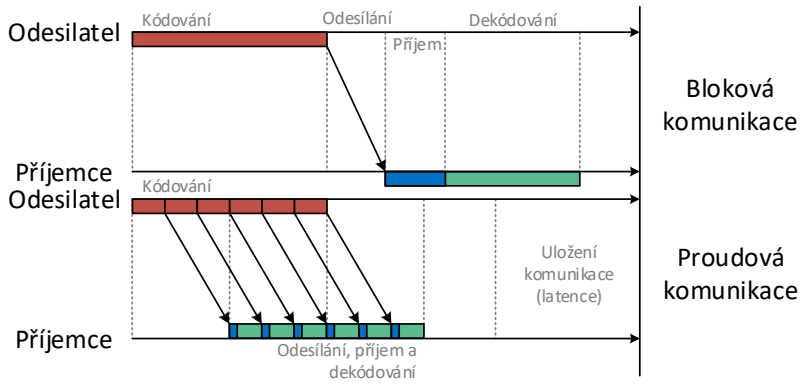
Nicméně situace v rámci průmyslových protokolů nelze vnímat takto zjednodušeně, jednotlivé vrstvy jsou skutečně mnohdy značně odlišné od běžných protokolů v rámci ISO/OSI či TCP/IP. To platí nejen v rámci poslední vrstvy, ale v rámci celé struktury v rámci komunikačního modelu, viz příklad uvedený na obrázku níže pro vybrané (nejběžnější) průmyslové komunikační protokoly, o kterých budeme mluvit blíže také dále (Modbus RTU, Modbus TCP, DeviceNet, Ethernet/IP, EtherCAT a Profinet).

<table>
<tr><th>OSI Model</th><th>Modbus RTU</th><th>Modbus TCP</th><th>DeviceNet</th><th>Ethernet/IP</th><th>TCP/IP Model</th></tr>
<tr><td>Vrstva 7
Aplikační vrstva</td><td>Modbus
Aplikační vrstva</td><td>Modbus
Aplikační vrstva</td><td>CIP
Aplikační vrstva</td><td>CIP
Aplikační vrstva</td><td rowspan="3">Vrstva 4
Aplikační vrstva</td></tr>
<tr><td>Vrstva 6
Prezenční vrstva</td><td rowspan="3"></td><td rowspan="3"></td><td>CIP
Data management</td><td>CIP
Data management</td></tr>
<tr><td>Vrstva 5
Relační vrstva</td><td>CIP routování
Management připojení</td><td>CIP routování
Management připojení</td></tr>
<tr><td>Vrstva 4
Transportní vrstva</td><td>TCP</td><td rowspan="2">DeviceNet
Transportní vrstva</td><td>TCP / UDP</td><td>Vrstva 4
Transportní vrstva</td></tr>
<tr><td>Vrstva 3
Síťová vrstva</td><td>IP</td><td>IP</td><td>Vrstva 3
Síťová vrstva</td></tr>
<tr><td>Vrstva 2
Linková vrstva</td><td>Master / Slave</td><td>Ethernet 802.3
MAC/LLC</td><td>CAN
CSMA/NBA</td><td>Ethernet 802.3
MAC/LLC</td><td rowspan="2">Vrstva 1
Síťové rozhraní</td></tr>
<tr><td>Vrstva 1
Fyzická vrstva</td><td>RS232 / RS-485</td><td>Ethernet
Fyzická vrstva</td><td>DeviceNet
Fyzická vrstva</td><td>Ethernet
Fyzická vrstva</td></tr>
</table>

Obr. 2.11: Ukázka podrobného zobrazení vrstev ISO/OSI a TCP/IP vybraných průmyslových protokolů [30].

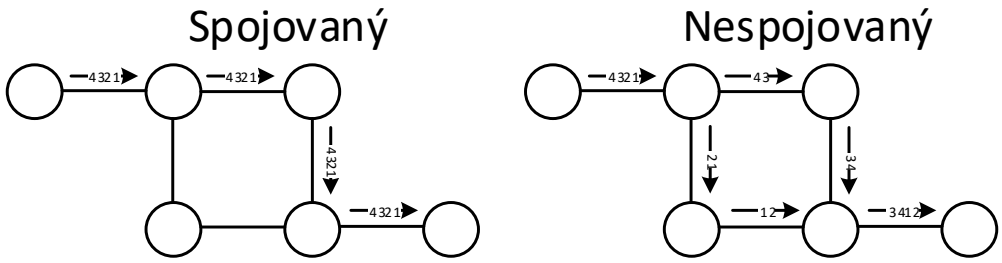
V kontextu průmyslových sítí se pro přenos dat a komunikaci používají různé techniky. Tyto techniky lze široce rozdělit do různých typů na základě režimu přenosu dat, konektivity, spolehlivosti a duplexního režimu. Pojďme se na každou z těchto kategorií podívat podrobněji.

Proudový a blokový přenos. Při proudovém přenosu jsou data přenášena nepřetržitě v toku bez jakékoli konkrétní hranice nebo struktury, podobně jako nepřetržitý proud vody. V průmyslu lze uvažovat pro proudový přenos protokoly Audio Video Bridging (AVB), Time-Sensitive Networking (TSN), User Datagram Protocol (UDP) či Real-time Transport Protocol (RTP). Na druhou stranu při přenosu bloků jsou data rozdělena do samostatných bloků, z nichž každý má přesně definovanou hranici, strukturu a velikost. Průmyslové protokoly jako Profinet, Modbus TCP/RTU, CANopen, EtherNet/IP, ale i spousty dalších využívají k přenosu dat blokový přenos.



Obr. 2.12: Ukázka (zleva): (i) blokový přenos, a (ii) proudový přenos [36].

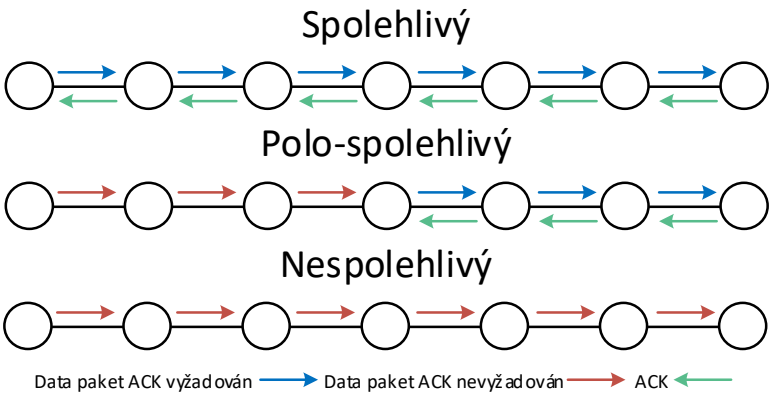
Spojovaný a nespojovaný přenos. Spojovaný přenos (Connection-Oriented, CO), známý také jako přenos orientovaný na připojení, vyžaduje vytvoření vyhrazeného spojení mezi odesílatelem a příjemcem, než dojde k přenosu dat. Tím je zajištěno, že data jsou doručena v pořadí, v jakém byla odeslána, a že se při přepravě neztratí. Příkladem spojového přenosu je protokol TCP (Transmission Control Protocol), ale také např. Modbus TCP. Na druhou stranu nespojovaný přenos (CL, Connectionless) nevyžaduje vyhrazené připojení a data jsou přenášena jako nezávislé pakety. Díky tomu je přenos rychlejší, ale také se zvyšuje pravděpodobnost ztráty nebo poškození dat při přenosu. Příkladem nespojového přenosu je protokol UDP (User Datagram Protocol), ICMP (Internet Control Message Protocol) či ARP (Address Resolution Protocol).



Obr. 2.13: Ukázka (zleva): (i) spojovaný přenos, a (ii) nespojovaný přenos [38].

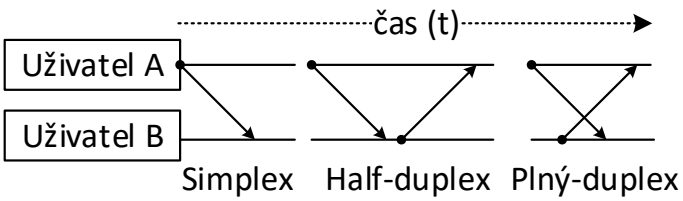
Spolehlivý a nespolehlivý přenos. Spolehlivý přenos zajišťuje přesné a úplné doručení dat od odesílatele k příjemci. Aby toho bylo dosaženo, spolehlivé přenosové protokoly často používají techniky, jako je kontrola chyb, řízení toku a opakované přenosy ztracených nebo poškozených dat. Příkladem spolehlivého přenosu v průmyslových sítích může být protokol Modbus TCP. Tento protokol využívá kontrolu integrity a řízení přenosu, aby zajistil spolehlivost přenosu dat v rámci sítě. Dalším příkladem může být protokol PROFINET, který také nabízí vysokou spolehlivost přenosu dat v průmyslových aplikacích. Na druhou stranu nespolehlivý přenos nezaručuje, že data budou doručena přesně nebo úplně. Je rychlejší, ale není vhodný pro kritické aplikace, kde je důležitá přesnost dat. Nespolehlivý přenos je typický pro některé protokoly, jako například UDP (User Datagram Protocol). Tento protokol posílá data bez jakýchkoliv záruk, že se dostanou do cíle, nebo že budou doručena v pořádku. Další příklady protokolů pro nespolehlivý přenos mohou být například protokoly pro přenos hlasu, jako je RTP, nebo protokoly pro zprostředkování zábavy,

jako je například protokol pro streamování videa RTSP (Real Time Streaming Protocol). Existuje také varianta polospolehlivé komunikace, což je druh komunikace, který poskytuje rovnováhu mezi spolehlivou a nespolehlivou komunikací. Poskytuje určitou úroveň spolehlivosti, ale ne tolik, jako spolehlivá komunikace. Úroveň spolehlivosti je určena požadavky aplikace. Při polospolehlivé komunikaci může dojít ke ztrátě některých dat, ale celková komunikace je stále považována za úspěšnou. Tento typ komunikace se často používá v průmyslových sítích, kde kritičnost dat není vysoká, ale přesto je vyžadována určitá úroveň spolehlivosti. Například v průmyslové automatizaci může být polospolehlivá komunikace použita pro účely monitorování, kde ztráta některých datových paketů nemusí mít významný dopad na celkový proces, ale přesto je nutné informace přenášet přesně. Některé protokoly, které podporují polospolehlivou komunikaci v průmyslových sítích, zahrnují User Datagram Protocol (UDP) s řízením toku a mechanismy opravy chyb.



Obr. 2.14: Ukázka (z vrchu dolů): (i) spolehlivý přenos, (ii) polo-spolehlivý přenos, a (iii) nespolehlivý přenos [16].

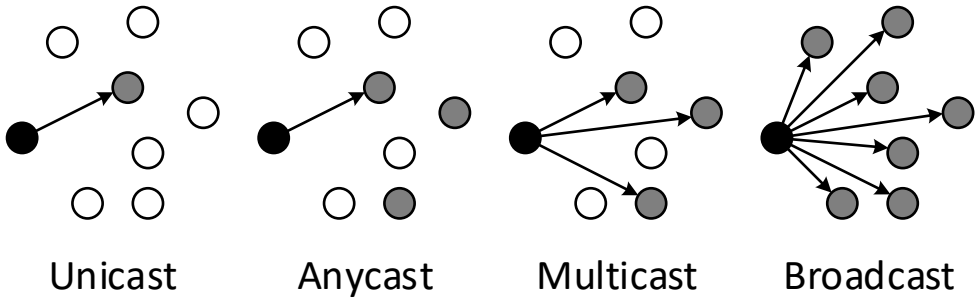
Simplex, Half-Duplex a plný-Duplex. Simplexní přenos označuje přenos dat pouze jedním směrem, od odesílatele k příjemci. Simplexní (jednosměrnou) komunikaci často využívají protokoly jako např. Modbus RTU, DNP3 či HART. Half-duplexní přenos umožňuje přenos dat oběma směry, ale současně pouze jedním směrem. Pro příklad můžeme uvést protokol CAN, který je často využíván jako half-duplex. Plně duplexní přenos umožňuje přenos dat v obou směrech současně. Průmyslové protokoly jako Profinet, EtherNet/IP a Modbus TCP využívají plně duplexní přenos. Rozdíl mezi těmito typy přenosu názorně, viz obrázek níže.



Obr. 2.15: Ukázka (zleva): (i) simplex, (ii) half-duplex, a (iii) plný duplex.

Unicast, Anycast, Broadcast a Multicast. Unicast označuje přenos dat od jednoho odesílatele k jednomu příjemci. Příkladem mohou být protokoly Remote Procedure Call (RPC) nebo Simple Mail Transfer Protocol (SMTP). Modbus TCP pak používá unicast komunikaci např. tam, kde jsou data odesílána z jednoho zdroje do jednoho cíle. Anycast je typ unicastu, kde jsou data odesílána do konkrétního uzlu vybraného ze skupiny uzlů na základě určitých kritérií. Příkladem může být Domain Name System (DNS). Anycast se pak běžně nepoužívá přímo v rámci průmyslových protokolů. Broadcast označuje přenos dat od jednoho odesílatele do všech uzlů v síti. Příkladem může být protokol Internet Protocol (IP) nebo Address Resolution Protocol (ARP). Některé průmyslové protokoly pak, jako je např. CANopen, používají broadcast komunikaci, kdy jsou data odesílána z jednoho zdroje do všech zařízení v síti. Multicast označuje přenos dat od jednoho odesílatele do určité skupiny uzlů v síti. Příkladem může být protokol Internet Group Management Protocol (IGMP) nebo Routing Information Protocol (RIP). Multicast se příliš často v průmyslových protokolech nepoužívá. Pro úplnost ještě možná nutno zmínit tzv. incast komunikaci, což je typ komunikace, kde velký počet příjemců současně požaduje data od jednoho odesílatele, což způsobuje, že odesílatel je zahlcen a není schopen včas zpracovat příchozí požadavky. Incast komunikace se často vyskytuje ve velkých datových centrech, prostředích cloud computingu a vysoce výkonných počítačových systémech. Například, když velký počet uživatelů přistupuje současně k oblíbené webové stránce, server, který je hostitelem webové stránky, může být přetížen a nemůže včas reagovat na všechny příchozí požadavky. To může u některých uživatelů vést k pomalému načítání stránky nebo dokonce k vypršení časového limitu. Ke zmírnění účinků incast komunikace mohou správci sítě použít techniky vyvažování zátěže k distribuci příchozích požadavků na více serverů nebo použít algoritmy řízení přetížení k regulaci toku dat. Incast komunikace není v průmyslovém prostředí příliš využívána, protože se jedná především o problém, který se vyskytuje ve vysoce výkonných počítačových sítích a sítích datových center. V takových sítích může incast vést k zahlcení sítě a snížení výkonu v důsledku velkého objemu dat přenášených

od/do jednoho cíle. To může způsobit kolize sítě a ztrátu datových paketů. V průmyslovém prostředí se obvykle upřednostňují spolehlivější a robustnější komunikační protokoly pro zajištění hladkého fungování řídicích a automatizačních systémů.



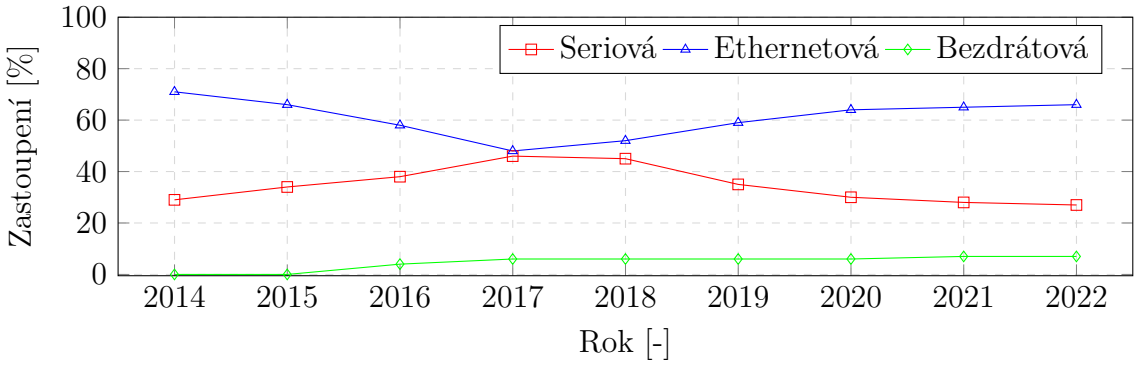
Obr. 2.16: Ukázka (zleva): (i) unicast, (ii) anycast, (iii) multicast, a (iv) broadcast [25].

Multi-drop. Multi-drop komunikace je typ komunikace, ve které je více zařízení připojeno k jedné komunikační lince, známé také jako multi-drop sběrnice. Při tomto typu komunikace sdílejí všechna zařízení stejnou komunikační linku a jsou schopna přijímat i vysílat data. Multi-drop komunikace se často používá v průmyslových prostředích, kde je potřeba, aby více zařízení komunikovalo mezi sebou navzájem nebo s centrálním ovladačem. Běžným příkladem multi-drop komunikace v průmyslovém prostředí je použití sběrnice RS-485. V tomto případě je více zařízení připojeno k jedné sběrnici RS-485 a mohou spolu komunikovat. Nabízí výhodu nákladové efektivity, protože jednu komunikační linku lze použít pro připojení více zařízení, čímž se snižuje potřeba samostatných komunikačních linek pro každé zařízení. Má však i svá omezení, jako je riziko kolizí dat a nutnost adresovacích mechanismů, které zajistí, že data budou odeslána na správné zařízení.

Závěrem lze říci, že tyto různé techniky používané pro přenos dat a komunikaci hrají v průmyslových sítích zásadní roli. Výběr techniky závisí na konkrétních požadavcích aplikace, jako je přesnost dat, rychlost a velikost sítě. Například protokol jako Profinet využívá blokový přenos, spojově orientovaný přenos, spolehlivý přenos, plně duplexní přenos a přenos unicast k zajištění přesného a efektivního přenosu dat v aplikacích průmyslové automatizace.

2.3.2 Komunikační a přenosové technologie

Jako první je nutno stanovit si médium pro přenosové technologie. V rámci OT sítí se jedná převážně o tři základní typy: sériová komunikace, ethernetová komunikace a bezdrátová komunikace. Optická komunikace je převážně pak v rámci WAN a GAN sítí, které pro zjednodušení nebudeme v tomto případě uvažovat. **Sériová komunikace** je jednoduchý a starší typ komunikačního protokolu, který používá jednu sériovou linku pro přenos dat. Tyto linky se často používají v průmyslových aplikacích k propojení jednotlivých zařízení a automatizačních systémů. Sériové komunikace mohou využívat různé protokoly, jako například RS-232, RS-485 nebo USB. Tyto protokoly se liší v rámci přenosového rozsahu, rychlosti přenosu a dalších specifických funkcí. **Ethernetová komunikace** je v současnosti nejrozšířenější typ komunikačního protokolu, který se používá v průmyslových aplikacích. Tyto sítě využívají jednoduchých a robustních protokolů, jako je například TCP/IP, které umožňují efektivní komunikaci mezi zařízeními a automatizačními systémy. Ethernetové sítě mohou být propojeny pomocí kabelů, jako jsou například ethernetové kabely kategorie CAT5 či CAT6. **Bezdrátová komunikace** je typ komunikačního protokolu, který umožňuje propojení zařízení a automatizačních systémů bez nutnosti použití kabelů. Tyto sítě využívají různých frekvenčních pásem a protokolů, jako je například Wi-Fi či Zigbee. Bezdrátové sítě se často používají v průmyslových aplikacích k propojení zařízení v obtížně přístupných nebo mobilních prostředích. Z pohledu zastoupení těchto technologií v rámci průmyslových sítí se můžeme podívat na obrázek níže, kde vidíme postupný ústup sériové komunikace a přechod převážně na ethernetové typy sítí (včetně postupného a pozvolného zvyšování zastoupení bezdrátového média), zdroj statistických dat je společnost Hardware meet Software Network[2], resp. její každoroční statistiky o zastoupení jednotlivých typů médií.

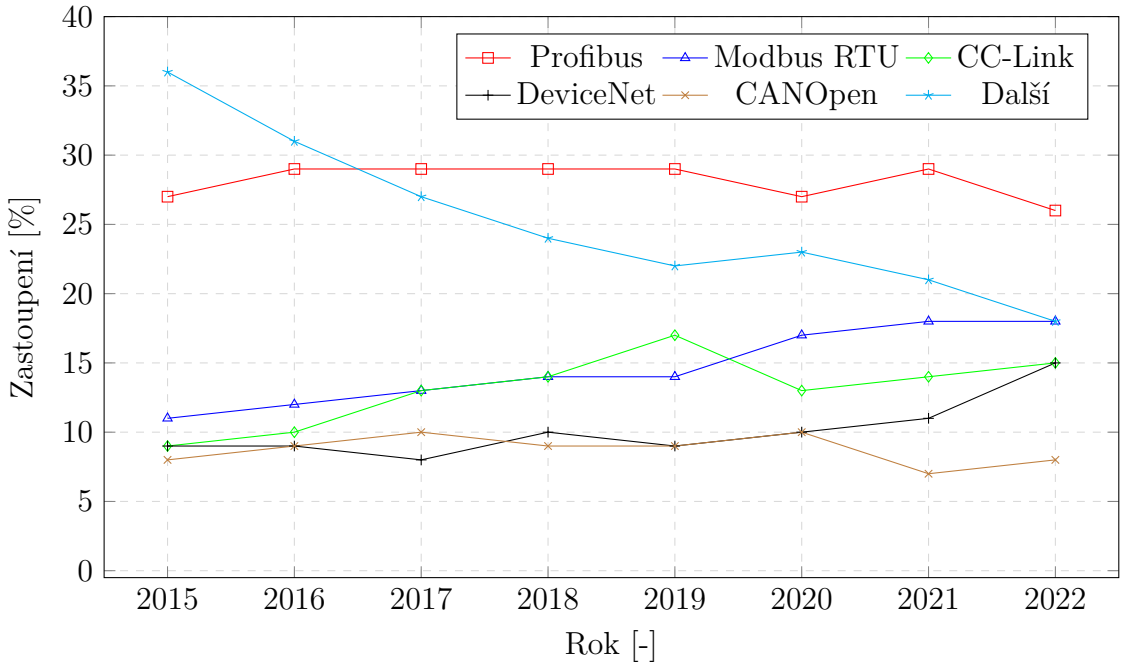


Obr. 2.17: Vývoj v rámci zastoupení jednotlivých typů médií v průmyslu.

[2]https://www.hms-networks.com/

A. Sériová komunikace

V rámci sériové komunikace, přejděme nejdříve k sériovým přenosovým technologiím (zjednodušeně je můžeme i vnímat jako fyzickou vrstvu komunikačních sériových technologií). Základními technologiemi jsou RS-232/422/485. RS-232 je standard pro sériový přenos dat a je široce používán pro komunikaci na krátkou vzdálenost mezi počítačovými systémy a periferními zařízeními. RS-232 podporuje plně duplexní i half-duplexní komunikaci a běžně se používá pro komunikaci Point-to-Point (PtP). Maximální vzdálenost pro RS-232 je obvykle kolem 15 metrů s maximální rychlostí přenosu dat kolem 20 kbps. RS-422 je diferenční sériový komunikační standard, který se běžně používá na delší vzdálenosti a pro multi-drop komunikaci. RS-422 podporuje plně duplexní komunikaci a běžně se používá pro vícebodovou komunikaci. Maximální vzdálenost pro RS-422 je kolem 4000 metrů s maximální rychlostí přenosu dat 10 Mbps. RS-485 je také diferenciální sériový komunikační standard, který podporuje multi-drop komunikaci a běžně se používá na delší vzdálenosti. RS-485 podporuje plně duplexní i half-duplexní komunikaci a lze jej použít pro vícebodovou komunikaci. Maximální vzdálenost pro RS-485 je kolem 4000 metrů s maximální rychlostí přenosu dat 10 Mbps. Z pohledu sériových komunikačních protokolů můžeme vidět zastoupení jednotlivých protokolů na obrázku níže (zdroj Hardware meet Software Network[3]).



Obr. 2.18: Vývoj v rámci zastoupení sériových komunikačních protokolů.

[3]https://www.hms-networks.com/

Profibus [23] je široce používaný komunikační protokol v oblasti průmyslové automatizace. Je to otevřený, digitální komunikační standard pro systémy řízení procesů a automatizace. Profibus poskytuje rychlou a spolehlivou komunikaci mezi provozními zařízeními, jako jsou senzory, akční členy a řídicí jednotky. Profibus je založen na specifikaci fyzické vrstvy RS-485. Komunikační rychlost Profibusu se pohybuje od 9,6 kbps do 12 Mbps, díky čemuž je vhodný pro různé aplikace, od jednoduchého řízení procesů až po složité automatizační systémy. Profibus je široce používán v řadě průmyslových odvětví, včetně petrochemie, úpravy vody, výroby energie a výroby. Je to nákladově efektivní, flexibilní a spolehlivé komunikační řešení pro mnoho průmyslových aplikací a poskytuje standardizovaný přístup ke komunikaci mezi různými zařízeními od různých výrobců.

Modbus RTU [40] je populární průmyslový komunikační protokol používaný v automatizačních systémech a systémech řízení procesů. Používá se k navázání komunikace mezi různými zařízeními, jako jsou programovatelné logické automaty (PLC), počítače a další průmyslová zařízení. Modbus RTU pracuje v konfiguraci master-slave, kde jedno zařízení funguje jako master a ostatní zařízení fungují jako slave. Master odešle požadavky na data a podřízené jednotky odpoví požadovanými informacemi. Modbus RTU používá jednoduchou binární reprezentaci dat a cyklickou kontrolu redundance (CRC) pro kontrolu chyb, takže je efektivní a spolehlivý pro komunikaci v náročných průmyslových prostředích. Běžně se implementuje pomocí komunikačních médií RS-232 či RS-485 v závislosti na konkrétních požadavcích aplikace, jako je komunikační vzdálenost, rychlost a odolnost proti elektromagnetickému rušení. Jeho maximální rychlost se pohybuje kolem 115 kbps. Modbus RTU je široce používán v mnoha průmyslových aplikacích, včetně řízení procesů, automatizace budov a systémů obnovitelné energie. Jeho popularita je způsobena snadností použití, jednoduchou a flexibilní komunikační strukturou a širokou kompatibilitou s širokou škálou zařízení a systémů.

CC-link [33] je průmyslový komunikační protokol, který se běžně používá pro komunikaci mezi průmyslovými řídicími systémy a zařízeními. Jedná se o systém fieldbus, což znamená, že se používá pro řízení v reálném čase a výměnu dat v distribuovaném řídicím systému. CC-Link je založen na architektuře master-slave, kde master zařízení řídí komunikaci mezi zařízeními a slave zařízení reagují na požadavky master. CC-Link je známý svou vysokorychlostní komunikací a velkým počtem podporovaných zařízení. Podporuje komunikační rychlosti až 10 Mbps, díky čemuž je vhodný pro použití v náročných průmyslových aplikacích, které vyžadují rychlou komunikaci. CC-Link také podporuje multi-drop komunikaci, která umožňuje připojení

více podřízených zařízení ke stejné komunikační lince. To z něj dělá efektivní a nákladově příznivé řešení pro rozsáhlá nasazení. Pokud jde o fyzickou vrstvu, CC-Link lze implementovat pomocí měděného kabelu. Měděný kabel se obvykle používá pro kratší vzdálenosti. CC-Link je široce používán v různých průmyslových odvětvích, včetně automobilového, petrochemického, potravinářského a nápojového průmyslu a dalších výrobních prostředích. Je uznáván jako mezinárodní standard a je podporován velkým počtem výrobců zařízení, což usnadňuje nalezení kompatibilních zařízení pro použití v systému CC-Link.

DeviceNet [8] je průmyslový komunikační protokol, který je běžně implementován pomocí RS-485, i když jej lze nasadit také pomocí RS-232. Volba mezi RS-232 a RS-485 bude záviset na specifických požadavcích aplikace, jako je komunikační vzdálenost a odolnost vůči elektromagnetickému rušení. DeviceNet je průmyslový komunikační protokol, který se používá pro připojení řídicích zařízení, jako jsou senzory, akční členy a programovatelné řídicí jednotky v řídicím systému. Byl vyvinut organizací Open DeviceNet Vendor Association (ODVA) a je založen na protokolu Controller Area Network (CAN). DeviceNet využívá jednoduchý, cenově výhodný a flexibilní způsob připojení průmyslových zařízení, což z něj činí ideální řešení pro širokou škálu průmyslových aplikací. Protokol poskytuje rychlou, deterministickou komunikaci a podporuje komunikaci peer-to-peer i multi-drop. Používá odolný kabelážní systém průmyslové kvality a podporuje maximální délku kabelu 500 metrů. Má typickou rychlost 1–4 Mbps. Protokol také podporuje diagnostiku sítě a diagnostiku pro jednotlivá zařízení, což usnadňuje odstraňování problémů a údržbu sítě. DeviceNet podporuje řadu profilů zařízení, včetně digitálního vstupu/výstupu, analogového vstupu/výstupu, řízení pohybu a dalších. Tato flexibilita z něj dělá oblíbenou volbu pro průmyslové automatizační systémy a je široce používán v průmyslových odvětvích, jako je automobilový průmysl, potraviny a nápoje a plasty.

CAN [11] je komunikační protokol založený na zprávách, který byl navržen speciálně pro automobilové a průmyslové aplikace. Byl vyvinut v 80. letech 20. století společností Bosch a nyní se používá v široké řadě aplikací, včetně automobilového, leteckého, lékařského vybavení a průmyslové automatizace. Protokol CAN využívá komunikační systém založený na zprávách s více mastery, kde může více zařízení odesílat a přijímat zprávy současně. Jednou z klíčových vlastností CAN je jeho schopnost zvládnout vysokorychlostní komunikaci s rychlostmi od 10 kbps do 1 Mbps. To platí tedy pro základní fyzickou vrstvu CANBus. CAN využívá také metodu přenosu diferenciálního signálu, která umožňuje robustní komunikaci i v přítomnosti elektrického šumu a jiných typů rušení. Protokol využívá mechanismus detekce kolizí a arbitráže, aby bylo zajištěno, že je přenášena vždy pouze jedna zpráva a že

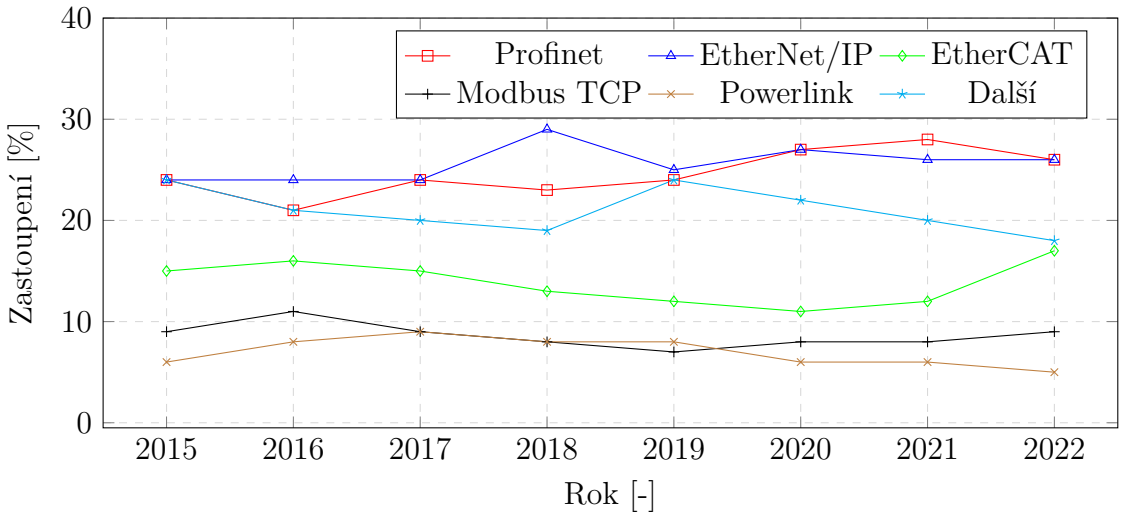
zprávy jsou přenášeny ve správném pořadí. Protokol CAN je implementován pomocí 2-drátové sběrnice, která umožňuje nákladově efektivní a jednoduchou elektroinstalační infrastrukturu. Podporuje také multi-drop komunikaci, což znamená, že na stejnou komunikační linku lze připojit více zařízení. Díky tomu je ideálním řešením pro aplikace, kde mezi sebou potřebuje komunikovat více zařízení, například právě v průmyslovém řídicím systému.

Další sériové protokoly [46, 27, 34, 10] se vyskytují v průmyslových sítích již v řádu nižších jednotkách procent. K těmto protokolům patří ASI, Sercos I/II, ControlNET, INTERBUS. Zbytek protokolů je zanedbatelný. ASI je průmyslový komunikační protokol obvykle implementovaný pomocí 2-drátové sběrnice. ControlNET je průmyslový komunikační protokol běžně implementovaný pomocí kabelu pomocí kroucené dvojlinky. INTERBUS je průmyslový komunikační protokol typicky implementovaný pomocí RS-485.

B. Ethernetová komunikace

Ethernet je typ komunikačního protokolu místní sítě (LAN), který používá sdílený kabel k přenosu dat mezi více zařízeními. Ethernet byl poprvé standardizován v 80. letech 20. století standardem IEEE 802.3 a od té doby se stal dominantní LAN technologií s rostoucím počtem průmyslových aplikací. Ethernetové sítě lze konstruovat pomocí kroucených párů kabelů, optických vláken nebo telefonního drátu. Tyto sítě podporují přenosové rychlosti dat v rozmezí od 1 Mbps do 400 Gbps. Jednou z hlavních výhod Ethernetu oproti jiným typům průmyslových sítí, jako jsou sériové přenosové protokoly, je jeho vysoká rychlost a schopnost podporovat velké množství zařízení v jedné síti. Ethernet také podporuje half-duplexní i plně duplexní komunikaci, což umožňuje současný přenos dat v obou směrech v jedné síti. Další výhodou Ethernetu je jeho široké využití a podpora ze strany různých výrobců hardwaru a softwaru, což usnadňuje nalezení kompatibilních komponent a integraci stávajících systémů do větší sítě. To může pomoci snížit náklady spojené s budováním a údržbou průmyslové sítě. Použití Ethernetu v průmyslových sítích má však některé nevýhody. Jednou z nich je, že ethernetové sítě mohou být zranitelné vůči rušení z jiných elektronických zařízení, což může vést k chybám přenosu a zpomalení. Kromě toho mohou být ethernetové sítě složitější na nastavení a údržbu ve srovnání se sériovými přenosovými protokoly, zejména ve větších průmyslových prostředích. Celkově vzrůstající obliba Ethernetu v průmyslových sítích je způsobena jeho vysokou rychlostí, schopností podporovat velké množství zařízení a širokou podporou ze strany prodejců hardwaru a softwaru. Je však důležité zvážit možné nevýhody a pečlivě navrhnout a udržovat sítě Ethernet, aby byla zajištěna

spolehlivá a efektivní průmyslová komunikace. Na obrázku níže jsou zobrazeni hlavní zástupci v rámci průmyslových sítí z oblasti komunikačních technologií využívající Ethernet jako přenosovou technologii (zdroj Hardware meet Software Network[4]).



Obr. 2.19: Vývoj v rámci zastoupení ethernetových komunikačních protokolů.

Profinet [22] je jedním z nejrozšířenějších ethernetových komunikačních protokolů používaných v průmyslu. Je to síťový protokol postavený na standardech IEEE 802.3 pro komunikaci mezi jednotlivými průmyslovými zařízeními, jako jsou automatizační systémy, programovatelné automaty, senzory a akční prvky. Profinet je součástí architektury Industry 4.0 a umožňuje real-time komunikaci mezi zařízeními, což umožňuje efektivní řízení průmyslových procesů. Protokol Profinet nabízí vysokou rychlost a efektivitu komunikace, když umožňuje přenos velkých objemů dat v reálném čase. Profinet podporuje rychlosti od 100 Mbps až do 10 Gbps. Tyto rychlosti závisí na konkrétní implementaci a vybavení sítě. Je důležité si uvědomit, že výběr správné rychlosti závisí na specifických potřebách a požadavcích na přenosový profil sítě, jako je například frekvence a množství dat, které se přenášejí. Obecně platí, že vyšší rychlosti poskytují lepší výkon a spolehlivost, ale také vyžadují více prostředků a nákladů na implementaci a údržbu. Má také schopnost řízení více jednotek v síti, což umožňuje jednodušší správu a údržbu systému. Profinet také podporuje redundantní sítě, což zajišťuje vysokou dostupnost systému a zabraňuje výpadkům v případě poruchy. Výhody použití protokolu Profinet zahrnují možnost real-time komunikace, vysokou rychlost a efektivitu, podporu redundantních sítí, jednoduchou správu a údržbu, a schopnost řízení více jednotek v síti. Nevýhody zahrnují

[4]https://www.hms-networks.com/

vyšší náklady na implementaci oproti jiným ethernetovým protokolům a vyšší nároky na technické znalosti pro správu a údržbu systému. V závislosti na specifických potřebách průmyslového prostředí může být použití protokolu Profinet výhodné pro zlepšení efektivity a spolehlivosti průmyslových procesů.

EtherNet/IP [12], označován také zkráceně jako ENIP, je jeden z protokolů z rodiny Common Industrial Protocol (CIP), který se používá pro komunikaci mezi průmyslovými zařízeními a řídicími systémy. EtherNet/IP využívá Ethernetových sítí k přenosu dat. EtherNet/IP umožňuje real-time komunikaci mezi průmyslovými zařízeními, jako jsou automatizační systémy, stroje a senzory, a řídicími systémy, jako jsou programovatelné automaty (PLC) a řídicí jednotky. Tento protokol umožňuje zařízením spolupracovat při řízení a ovládání průmyslových procesů. EtherNet/IP podporuje různé rychlosti přenosu dat, včetně 10 Mbps a 100 Mbps. Stejně jako ostatní Ethernetové protokoly, EtherNet/IP podporuje plný-duplex (obousměrný) přenos dat. Výhodou EtherNet/IP je, že umožňuje snadnou integraci s existujícími Ethernetovými sítěmi a širokou dostupnost komponent a zařízení, které tento protokol podporují. Navíc EtherNet/IP nabízí vysokou propustnost dat a škálovatelnost, což umožňuje rozšiřovat sítě v průběhu času. Nevýhodou EtherNet/IP je, že může být náročnější na správu a údržbu oproti jiným industriálním komunikačním protokolům. Kromě toho mohou být někdy potřebné speciální bezpečnostní opatření, aby bylo zajištěno bezpečné a spolehlivé používání protokolu v průmyslových sítích.

Modbus TCP [47] je síťový protokol, který je založen na přenosu dat pomocí TCP/IP sítí. Je to v podstatě modifikace klasického Modbus RTU protokolu, který byl původně navržen pro použití s sériovými sítěmi, a který byl adaptován pro použití s TCP/IP sítěmi. Modbus TCP je výkonný a spolehlivý protokol, který umožňuje zařízením v síti komunikovat mezi sebou. Modbus TCP podporuje širokou škálu rychlostí, včetně 10 Mbps a 100 Mbps. Výhodou Modbus TCP je, že umožňuje více zařízením komunikovat se stejným zdrojem dat najednou, což znamená, že mohou být použity v širší škále aplikací než sériové sítě. Modbus TCP také umožňuje vzdálenou správu a konfiguraci zařízení pomocí síťových nástrojů, což umožňuje snadnou správu a monitoring sítě. Významnou nevýhodou Modbus TCP je vyšší latence oproti sériovým sítím, což může být pro některé aplikace kritické. Také může být náročnější na výkon, což může být problémem pro některá zařízení s nižšími výpočetními zdroji. Celkově lze říci, že Modbus TCP je spolehlivý a široce používaný protokol pro komunikaci v průmyslových sítích. Jeho podpora komunikace více zařízení se stejným zdrojem dat, a podpora vzdálené správy a konfigurace jsou velkými výhodami, které ho činí vhodným pro širokou škálu průmyslových aplikací.

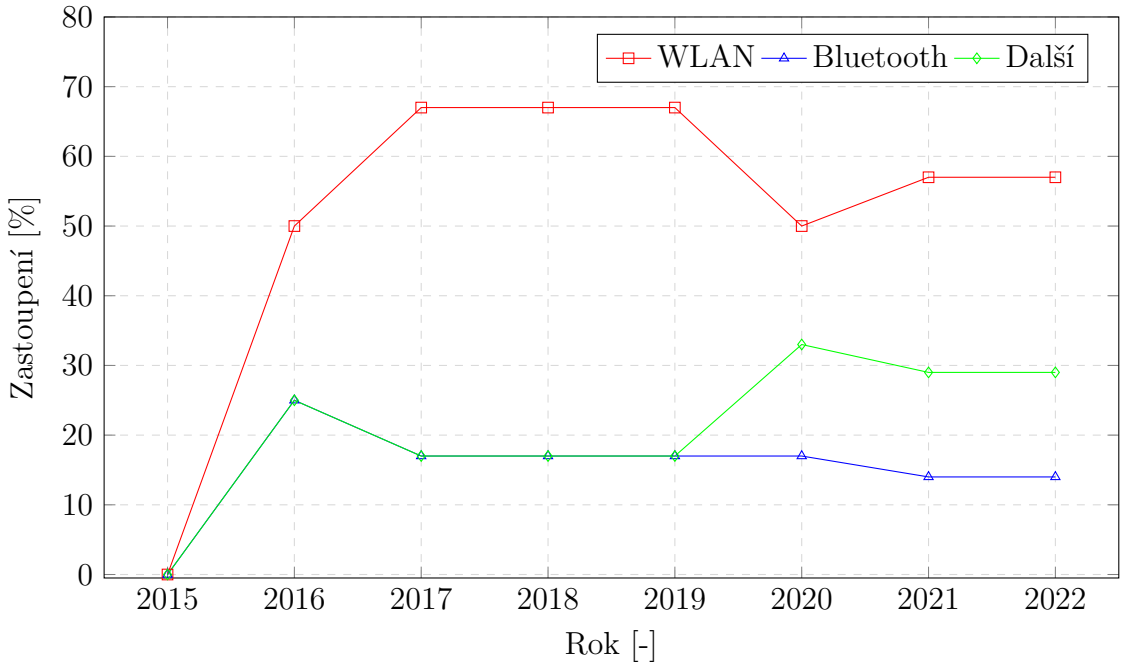
Powerlink [6] (PROFINET real-time communication protocol) je protokol průmyslového Ethernetu v reálném čase vyvinutý rakouskou společností B&R Automation. Je navržen pro vysokorychlostní komunikaci v aplikacích průmyslové automatizace a poskytuje nízkou latenci, vysoký determinismus a vysokou spolehlivost. Protokol je navržen pro provoz ve standardních ethernetových sítích a podporuje datové rychlosti až 100 Mbps. Powerlink poskytuje komunikační kanál v reálném čase pro řízení pohybu, I/O komunikaci a další časově kritické aplikace. Podporuje různé komunikační služby, včetně cyklické výměny dat, acyklické výměny dat a deterministických služeb. Jednou z klíčových výhod Powerlinku je jeho schopnost poskytovat komunikaci v reálném čase s nízkou latencí kolem 100 mikrosekund. Díky tomu je vhodný pro aplikace, jako je řízení pohybu a robotika, kde je nezbytná rychlá a přesná komunikace. Kromě toho Powerlink podporuje komunikaci master-slave i peer-to-peer, díky čemuž je flexibilní pro širokou škálu aplikací průmyslové automatizace. Další výhodou Powerlinku je jeho vysoký determinismus, který zajišťuje, že komunikace je předvídatelná a konzistentní. Toho je dosaženo použitím vyhrazených komunikačních kanálů, zaručené šířky pásma a plánování založeného na prioritách. Stručně řečeno, Powerlink je ethernetový protokol v reálném čase navržený pro aplikace průmyslové automatizace, který poskytuje nízkou latenci, vysoký determinismus a vysokou spolehlivost. Díky podpoře různých komunikačních služeb a flexibilním komunikačním modelům je vhodný pro širokou škálu aplikací.

Další ethernetové protokoly [39, 27] se vyskytují v průmyslových sítí již v řádu nižších jednotkách procent. Převažují zde protokoly CC-LINK and FIELD Sercos III. CC-LINK je široce používán v Japonsku a Asii, a nabízí vysokou rychlost a spolehlivost pro řízení a správu průmyslových zařízení. Sercos III je navržen pro řízení pohybu a řízení výrobních procesů a nabízí široké možnosti konfigurace a diagnostiky. Oba protokoly podporují vysokou rychlost a efektivitu přenosu dat a jsou široce používány v různých průmyslových odvětvích, jako jsou automobilový průmysl, výroba a potravinářský průmysl. Ostatní protokoly jsou již svým zastoupením zanedbatelné.

C. Bezdrátová komunikace

Bezdrátová komunikace se stává stále populárnější v průmyslových prostředích kvůli rostoucí poptávce po flexibilitě a mobilitě ve výrobních procesech. Pokud jde o rychlost, bezdrátová komunikace se v průmyslových prostředích může pohybovat od několika kbps až po stovky Mbps, v závislosti na použité technologii. Dosah bezdrátové komunikace v průmyslu je obvykle omezen na několik set metrů, ale lze jej rozšířit pomocí opakovačů nebo jiných technologií pro rozšíření dosahu. Frekvenční pásma

používaná průmyslovými bezdrátovými technologiemi jsou obvykle bez licence, což znamená, že jsou sdílená s jinými zařízeními a mohou být rušena. Rušení může ovlivnit spolehlivost a výkon bezdrátové komunikace v průmyslových prostředích a při výběru bezdrátové technologie je důležité pečlivě zvážit provozní prostředí. Ve srovnání se sériovou a ethernetovou komunikací nabízí bezdrátová komunikace v průmyslových prostředích několik výhod, jako je flexibilita, mobilita a snadná instalace. Má však také několik nevýhod, jako je právě omezený dosah, náchylnost k rušení a potenciální bezpečnostní rizika. Je důležité pečlivě zvážit specifické požadavky průmyslového prostředí a zvolit vhodnou komunikační technologii, která těmto potřebám vyhovuje. Na obrázku níže vidíme zastoupení jednotlivých bezdrátových přenosových technologií v rámci průmyslových sítí (zdroj Hardware meet Software Network[5]),



Obr. 2.20: Vývoj v rámci zastoupení bezdrátových komunikačních protokolů.

WLAN [48] (Wireless Local Area Network, WLAN) využívá rádiové vlny pro bezdrátovou komunikaci v určitém rozsahu, obvykle až do vzdálenosti několika set stop. V průmyslovém prostředí se WLAN používá pro různé aplikace, včetně automatizace, řízení a monitorování. Nejčastěji používaným protokolem WLAN pro průmyslové aplikace je IEEE 802.11, který zahrnuje standardy pro různá frekvenční pásma a datové rychlosti, včetně 2,4 GHz a 5 GHz. WLAN nabízí několik výhod v průmyslovém prostředí, včetně snadné instalace a flexibility, protože nevyžaduje fyzickou

[5]https://www.hms-networks.com/

kabeláž. Umožňuje také mobilitu, protože zařízení lze připojit bezdrátově, aniž by byla fyzicky připojena k síti. Používání WLAN v průmyslovém prostředí má však také některé nevýhody, jako je potenciální rušení jinými bezdrátovými zařízeními a omezený dosah ve srovnání s drátovými sítěmi. Kromě toho mohou průmyslové sítě WLAN vyžadovat speciální konfigurace a bezpečnostní opatření k zajištění spolehlivosti a bezpečnosti v drsném průmyslovém prostředí. Maximální rychlost WLAN v průmyslovém prostředí závisí na konkrétním používaném standardu a frekvenčním pásmu, stejně jako na počtu zařízení připojených k síti a na přítomnosti rušení. Obecně se maximální rychlosti pro průmyslové sítě WLAN pohybují od 11 Mbps do 600 Mbps. Celkově je WLAN oblíbenou volbou pro průmyslovou komunikaci díky snadnému použití a flexibilitě, ale je důležité pečlivě zvážit specifické požadavky průmyslového prostředí a správně nakonfigurovat a zabezpečit síť pro spolehlivou a bezpečnou komunikaci.

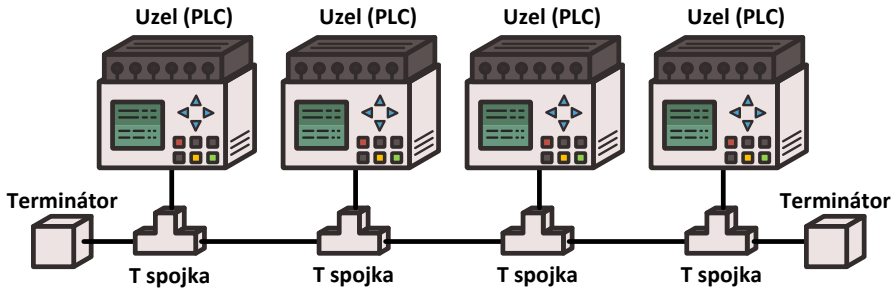
Bluetooth [20] je standard bezdrátové komunikace, který umožňuje výměnu dat mezi zařízeními na krátké vzdálenosti. Poprvé byl představen v roce 1994 a od té doby prošel několika revizemi a vylepšeními, včetně technologie nízkoenergetické verze Bluetooth (Bluetooth Low Energy, BLE), která je navržena pro aplikace s nízkou spotřebou a nízkou rychlostí přenosu dat. Bluetooth pracuje v pásmu 2,4 GHz ISM a využívá techniku frekvenčního přeskakování rozprostřeného spektra (Frequency Hopping Spread Spectrum, FHSS) ke snížení rušení s jinými zařízeními. Technologie podporuje datové rychlosti 1 Mbps a poskytuje dosah až 100 metrů, ačkoli skutečný dosah může být ovlivněn překážkami, jako jsou zdi a jiné konstrukce. V průmyslových sítích se Bluetooth často používá pro komunikaci mezi stroji (M2M), kde jej lze použít k připojení senzorů, akčních členů a dalších zařízení k řídicímu systému. BLE lze například použít pro bezdrátové monitorování stavu, prediktivní údržbu a pro přenos procesních dat ze senzorů v reálném čase. Jednou z hlavních výhod Bluetooth v průmyslových sítích je jeho nízká spotřeba energie, díky čemuž je vhodný pro zařízení napájená bateriemi a pro aplikace s omezeným výkonem. Kromě toho je Bluetooth široce podporován a lze jej snadno integrovat do různých zařízení a systémů. Používání Bluetooth v průmyslových sítích má však některé nevýhody. Technologie je například určena pro komunikaci na krátkou vzdálenost, což může omezit její použití ve větších průmyslových objektech. Kromě toho je Bluetooth citlivý na rušení z jiných zařízení, což může ovlivnit spolehlivost komunikace. Celkově mohou být Bluetooth a BLE užitečnými nástroji v průmyslových sítích, zejména pro aplikace, které vyžadují komunikaci s nízkou spotřebou energie a nízkou přenosovou rychlostí.

Další bezdrátové protokoly [43, 49] v průmyslové síti se neomezují pouze na WLAN a Bluetooth (i přesto, že ty představují majoritu). Existuje několik dalších technologií, které jsou speciálně navrženy tak, aby vyhovovaly jedinečným požadavkům průmyslového prostředí. Tyto technologie zahrnují senzorové sítě, IoT, IIoT a sub-GHz řešení, jako jsou LoRaWAN, Sigfox a NB-IoT. Každá z těchto technologií má své silné stránky a omezení, takže jsou vhodné pro konkrétní případy použití. Senzorové sítě se používají pro monitorování a sběr dat z různých senzorů v průmyslovém prostředí. Tyto sítě obvykle pracují v nízkofrekvenčních pásmech, jako je pásmo ISM pod 1 GHz nebo 2,4 GHz, a mají omezený dosah přibližně 1 km až 10 km. Rychlost přenosu dat se u těchto sítí pohybuje od několika kbps do několika stovek kbps a latence může být od milisekund do několika sekund. Z hlediska chybovosti je u těchto sítí obecně nízká, ale může se zvýšit v případě rušení z jiných zdrojů. Technologie IoT a IIoT se používají pro připojení velkého množství zařízení v průmyslovém prostředí a pro umožnění komunikace mezi stroji. Tyto technologie pracují ve frekvenčních pásmech pod 1 GHz, 2,4 GHz a 5 GHz a mají dosah přibližně 100 m až 1 km. Přenosová rychlost se u těchto sítí pohybuje od několika kbps do několika Mbps a latence může být od milisekund do několika sekund. Chybovost u těchto sítí je obecně nízká, ale může se zvýšit v případě rušení z jiných zdrojů. Sub-GHz řešení, jako jsou LoRaWAN, Sigfox a NB-IoT, se používají pro nízkoenergetickou komunikaci na dlouhé vzdálenosti v průmyslových prostředích. Tyto technologie pracují ve frekvenčním pásmu pod 1 GHz a mají dosah přibližně 10 km až 50 km. Rychlost přenosu dat se u těchto sítí pohybuje od několika kbps do několika stovek kbps a latence může být od milisekund do několika sekund. Chybovost u těchto sítí je obecně nízká, ale může se zvýšit v případě rušení z jiných zdrojů. Ve srovnání se sériovou a ethernetovou komunikací nabízejí bezdrátové technologie větší flexibilitu a mobilitu. Lze je snadno nasadit a překonfigurovat, díky čemuž jsou ideální pro aplikace, kde nasazení kabelů není možné, nebo je příliš nákladné. Bezdrátové technologie však mohou být rušeny jinými zdroji a jejich výkon může být ovlivněn faktory prostředí, jako je teplota, vlhkost a vzdálenost. Bezdrátové technologie mohou navíc vyžadovat více energie k provozu než jejich kabelové protějšky, což může být problémem u zařízení napájených bateriemi. Závěrem lze říci, že bezdrátové technologie, jako jsou senzorové sítě, IoT, IIoT a sub-GHz řešení, hrají důležitou roli v průmyslové komunikaci, poskytují větší flexibilitu a mobilitu, a také umožňují komunikaci mezi stroji. Tyto technologie se používají ve specifických případech, kdy není možné zavést kabely, nebo kde je vyžadována komunikace na dlouhé vzdálenosti s nízkou spotřebou energie.

2.3.3 Topologie sítě

V průmyslových sítích hrají fyzické i logické topologie důležitou roli v celkovém návrhu a funkčnosti sítě. Fyzická topologie se týká fyzického uspořádání sítě a způsobu propojení zařízení. To zahrnuje typ použitého kabelu (jako je měděný nebo optický kabel), typ použitého konektoru a uspořádání zařízení v síti (jako je hvězdicová nebo kruhová konfigurace). Logická topologie se týká způsobu přenosu dat po síti, nezávisle na fyzickém uspořádání. To zahrnuje používané komunikační protokoly, jako je Ethernet, TCP/IP nebo Modbus, a způsob přenosu dat z jednoho zařízení do druhého. Při návrhu průmyslových sítí je třeba pečlivě zvážit jak fyzické, tak logické topologie, protože ovlivňují celkový výkon, spolehlivost a bezpečnost sítě. Z pohledu logické topologie je nutno dále zmínit tři základní techniky, které dnes dopomáhají s realizací logické topologie nad rámec samotných protokolů, a tedy i oddělit skutečnou fyzickou topologii od té logické: Virtualizace síťových funkcí (Network Function Virtualization, NFV), softwarově definované sítě (Software-Defined Networking, SDN) a Cloud computing. Všechny tyto technologie souvisejí s topologií průmyslové sítě, protože mění způsob, jakým jsou sítě navrhovány, nasazovány a spravovány. Cílem těchto technologií je zjednodušit síťové operace a učinit je flexibilnějšími a škálovatelnějšími. NFV umožňuje síťovým funkcím, jako jsou brány firewall, směrovače a nástroje pro vyrovnávání zátěže, běžet na virtuálních počítačích namísto na fyzických zařízeních. To umožňuje nasadit a spravovat síťové funkce jako software namísto vyhrazeného hardwaru, což může zvýšit efektivitu a snížit náklady. SDN na druhé straně odděluje řídicí rovinu od datové roviny v síti. Řídicí rovina, která je zodpovědná za rozhodování o síťovém provozu, běží na centralizovaném řadiči, zatímco datová rovina, která se stará o předávání paketů, běží na přepínačích. Toto oddělení odpovědností umožňuje automatizovat mnoho síťových operací, jako je dopravní inženýrství a prosazování bezpečnostní politiky, což může zjednodušit správu sítě a snížit prostoje. Cloud computing je mezitím model poskytování služeb IT, ve kterém jsou zdroje poskytovány přes internet na základě platby za použití. To může zahrnovat infrastrukturu jako službu (IaaS), platformu jako službu (PaaS), software jako službu (SaaS) a zařízení jako službu (DaaS). Díky využití cloud computingu mohou průmyslové sítě získat přístup ke škálovatelným výpočetním zdrojům na vyžádání, aniž by musely investovat a udržovat vlastní infrastrukturu. Z pohledu architektury topologie pak rozdělujeme v průmyslových sítích nejčastěji topologii: sběrnicovou, hvězdicovou, kruhovou, stromovou, propojenou (MESH) a hybridní.

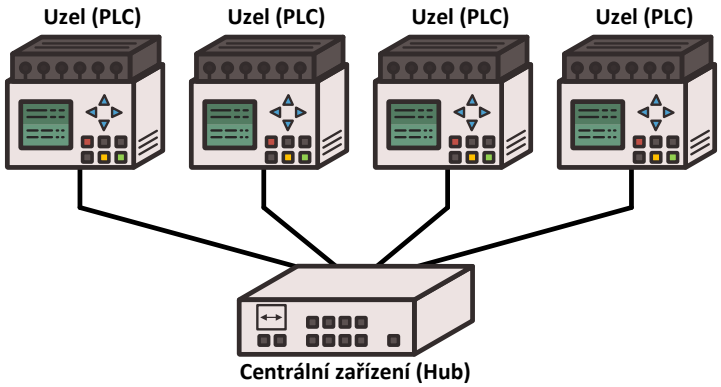
Sběrnicová topologie (Bus Topology) je složena z hlavního vedení, nazývaného sběrnice, kde fyzickým přenosovým médiem je zde nejčastěji koaxiální kabel či síťový kabel RJ-45. Ukázka sběrnicové topologie je zobrazena na obr. 2.21. Jednotlivé uzly jsou připojeny ke sběrnici pomocí tzv. T spojky. Při komunikaci je vyslán signál z uzlu na obě strany a putuje až na samotný konec sběrnice. Z tohoto důvodu musí být sběrnice vždy na obou koncích zakončena terminátorem, který zajišťuje pohlcení signálu, tak, aby nedocházelo k odrazu zpět do sítě, což by zvyšovalo chybovost sítě, způsobovalo kolize, a snižovalo celkový výkon sítě. Komunikace probíhá pouze v half-duplexním módu, kde hlavně v rámci fyzické topologie mají všechny uzly stejnou prioritu s využitím protokolů přístupu k médiu (Medium Access Control, MAC).



Obr. 2.21: Ukázka sběrnicové topologie sítě.

Sběrnicová topologie představuje velmi snadné připojení jednotlivých uzlů do sítě (i přesto, že vyžaduje specializovaná zařízení jako terminátor či T spojku). Výhodou je snadné pochopení topologie a její přehlednost. Oproti ostatním topologiím vyžaduje na sestavení logicky menší délku kabelu (např. oproti hvězdicové topologii). Díky těmto faktům jsou tak značně redukovány i vstupní kapitálové náklady (Capital Expenditures, CAPEX) a vybudování této sítě je tak levné. Značnou nevýhodou sítě je pak celkem samozřejmá závislost na sběrnici, tedy poškození sběrnice vyvolá okamžitý výpadek celé sítě. To znesnadňuje diagnostiku sítě a lokalizaci poruchy. Provozní náklady (Operational Expenditures, OPEX) této sítě se tak mohou s časem značně navyšovat. Z této závislosti pak vychází také omezení délky vedení i maximální propustnosti sítě, kde každé další zařízení či zvýšení provozu snižuje výkonnost celé sítě, což v konečném důsledku limituje i počet zařízení na jednu síť. Tato topologie je tak vhodnější spíše pro menší lokální sítě. Z pohledu bezpečnosti tedy tato topologie vytváří tzv. jediný bod selhání (Single Point of Failure, SPOF) v podobě sběrnice. Dále je nutno si uvědomit, že veškerý provoz v síti je sdílen všemi připojenými uzly. Navíc jakékoliv nakažení jednoho uzlu škodlivým softwarem (Malicious Software, Malware) ovlivní i všechny další uzly v síti.

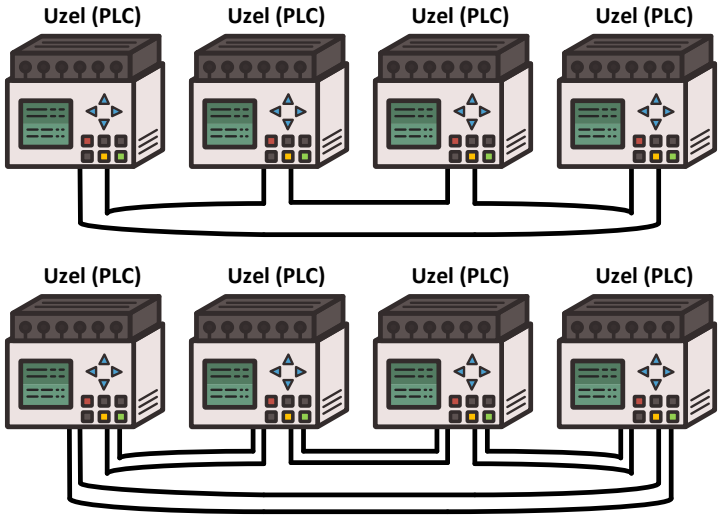
Hvězdicová topologie (Star Topology) představuje PtP zapojení každého uzlu přímo do právě jednoho centrálního zařízení (rozbočovač, přepínač, router či jiné) a nepřímo ke každému dalšímu zařízení. Ukázka hvězdicové sítě je zobrazena na obr. 2.22. Celá síť je tedy řízena pouze z jednoho místa – centrálního zařízení. Toto připojení je velmi oblíbené v bezdrátových sítích, nicméně setkat se s ním můžeme samozřejmě i v rámci drátových. Fyzickým přenosovým médiem je tak vzduch, ale i koaxiální kabel, síťový kabel RJ-45, optické vlákno, aj. Oproti ostatním topologiím (např. kruhová topologie) vyžaduje na sestavení méně kabeláže (ale stále více než sběrnicová topologie), ale např. oproti sběrnicové topologie jsou CAPEX náklady vyšší, a to už jen díky nutnosti zakoupení centrálního zařízení. Menší nároky na kabeláž a existence centrálního řízení pak umožňují snadnou správu a nastavení sítě. Díky tomu je i jednoduchá samotná diagnostika chyb v síti, čemuž přispívá i aditivní vrstva ochrany tvořena nezávislosti chodu sítě na jednotlivých uzlech. Připojení dalších uzlů probíhá za chodu bez nutnosti vypínat síť jako např. u sběrnicové či kruhové topologie.



Obr. 2.22: Ukázka hvězdicové topologie sítě.

Nicméně stejně, jako v případě sběrnicové topologie, chyba centrálního zařízení zde vyvolá okamžitý výpadek sítě. Z tohoto důvodu jsou v rámci hvězdicové topologie kladeny velké nároky na centrální zařízení, a to nejen z pohledu spolehlivosti, ale také výkonnosti, kdy výkon celé sítě odpovídá výkonu centrálního uzlu. To vytváří následně nároky z dlouhodobého hlediska na náklady typu OPEX. Je logické, že se zvyšujícím se provozem i počtem uzlů rostou společně i nároky na centrální uzel. V neposlední řadě je nutno podotknout, že centrálním bodem prochází veškerá komunikace v síti, což představuje samozřejmě očividné bezpečnostní riziko, mj. i zde je přítomen SPOF v podobě tedy centrálního bodu či nakažení centrálního bodu malwarem znamená případě přímé ohrožení všech ostatních bodů v síti.

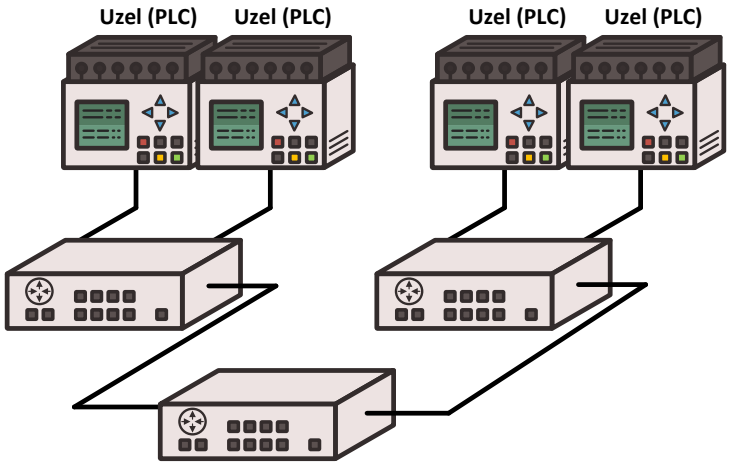
Kruhová topologie (Ring Topology) je charakterizována tím, že jednotlivé uzly v síti jsou vzájemně propojeny v kruhovém uspořádání. Tyto uzly tvoří spojnici pro přenos dat mezi nimi, kdy data putují po kruhu, tedy od jednoho uzlu k druhému, až se dostanou zpět na svůj zdroj. Ukázka kruhové topologie je zobrazena na obr. 2.23.



Obr. 2.23: Ukázka kruhové (nahoře) a zdvojené kruhové topologie (dole).

Fyzickým přenosovým médiem může být například koaxiální kabel, síťový kabel nebo optické vlákno. Výhodou kruhové topologie je, že při přerušení jednoho spojení stále existuje cesta pro přenos dat. Dále navíc při správném fungování protokolů dochází ke zvýšení spolehlivosti a odolnosti sítě. V případě potřeby může být kruhová topologie snadno rozšířena přidáním dalšího uzlu do kruhu. Nevýhodou je vysoká náročnost na správu a diagnostiku sítě. V případě selhání jednoho uzlu může dojít k výpadku celé sítě. Navíc, stejně jako u sběrnicové a hvězdicové topologie, kruhová topologie představuje jediný bod selhání (SPOF). Toto je pravda však jenom částečně, protože v případě kruhové topologie existuje vždy ještě druhá cesta pro přenos dat, což znamená, že pokud dojde k selhání jednoho zařízení, data mohou putovat jinou trasou. Toto zvyšuje spolehlivost sítě a snižuje pravděpodobnost výpadku celé sítě. Avšak je nutné si uvědomit, že selhání jednoho zařízení může vést k redukci celkového výkonu sítě. Proto se v praxi často používá topologie dvojitého kruhu, kde existují dva samostatné kruhy, takže pokud dojde k selhání jednoho z nich, mohou data putovat druhou trasou a zajistit tak vysokou dostupnost sítě. V případě topologie dvojitého kruhu (Dual Ring Topology) se tedy jedná o kombinaci dvou oddělených kruhových topologií, kdy pokud jeden kruh selže, druhý může pokračovat v přenosu dat. Tato topologie tak zvyšuje spolehlivost a odolnost sítě. Nicméně, stejně jako u jednoduché kruhové topologie, i zde existuje SPOF a nároky na správu a diagnostiku sítě jsou vysoké.

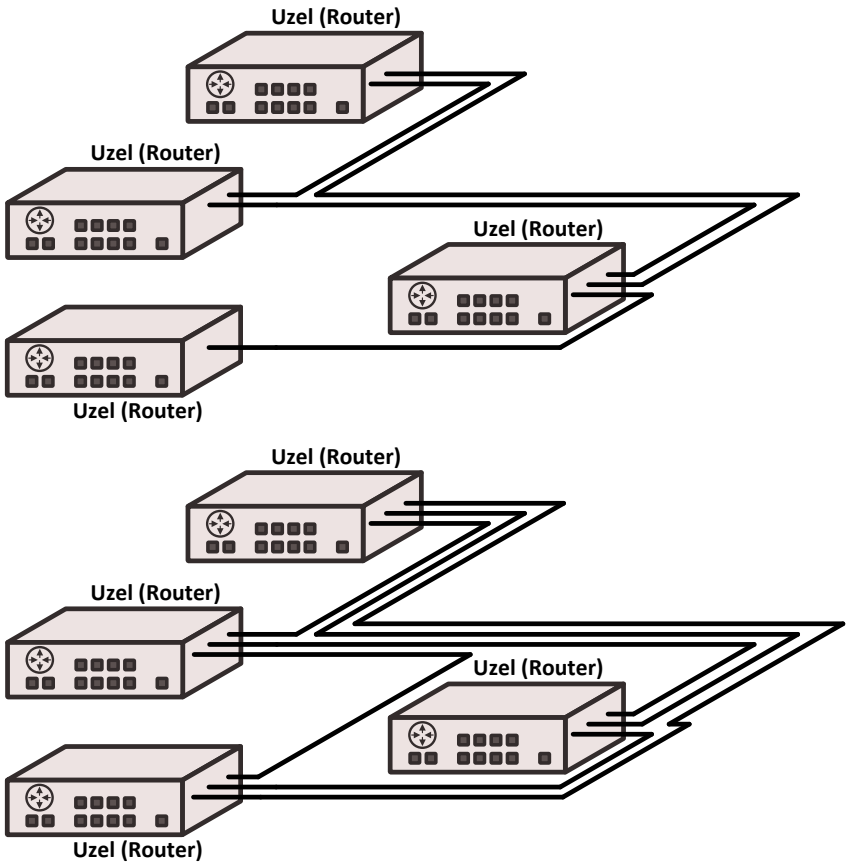
Stromová topologie (Tree Topology) je vícevrstvá topologie, kde jsou jednotlivé uzly a segmenty řazeny hierarchicky. Hlavním bodem sítě je tzv. kořenový uzel, od kterého se větví další uzly a segmenty, tvořící tak větve stromu. Tyto větve pak mohou být dále rozděleny na další podvětve. Ukázka stromové topologie je zobrazena na obr. 2.24.



Obr. 2.24: Ukázka stromové topologie sítě.

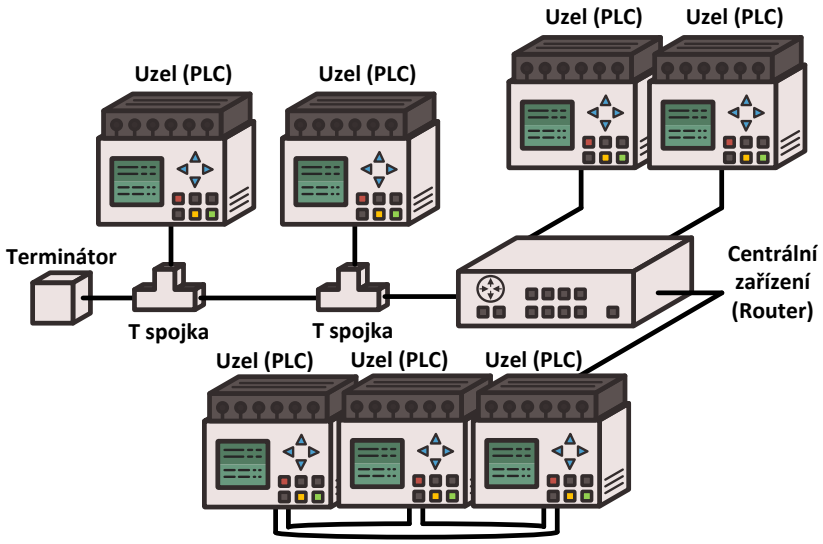
Stromová topologie se používá hlavně pro velké sítě, kde je nutné oddělit jednotlivé segmenty a uzly do jednotlivých vrstev, čímž se zvyšuje přehlednost sítě a snižuje náročnost správy. Tento typ topologie umožňuje v případě selhání jednoho segmentu či uzlu, aby byl provoz přerušen pouze na daném místě a nikoliv na celé síti. Výhodou stromové topologie je možnost rozšíření sítě bez nutnosti jejího přerušení, jednoduchost správy a diagnostiky, stejně jako i možnost použití různých typů fyzických přenosových médií. Nicméně, oproti jiným topologiím, vyžaduje stromová topologie více kabeláže, a také vyšší náklady na CAPEX. Navíc, v případě selhání kořenového uzlu, může dojít k výpadku celé sítě. Pro bezpečnost sítě je nutné také zvýšit pozornost na možnost útoku na kořenový uzel.

Propojená topologie (partial MESH) představuje kombinaci více topologií do jedné. Tyto topologie mohou být například hvězdicové, kruhové nebo sběrnicové. Tento typ topologie lze považovat za hybridní řešení, které kombinuje výhody různých topologií. V tomto případě existuje několik centrálních bodů, které jsou spojeny v rámci hvězdicového zapojení, zatímco uzly jsou připojeny do kruhového zapojení. Toto řešení umožňuje vyšší redundanci a spolehlivost v případě selhání jednoho z uzlů, nicméně vyžaduje vyšší nároky na správu a údržbu sítě. Plně propojená (Full MESH) topologie představuje nejkomplexnější formu sítě, kde každý uzel je přímo připojen ke všem ostatním uzlům. Tyto spoje mohou být přímé nebo nepřímé (přes jiný uzel). Toto řešení umožňuje nejvyšší redundanci a spolehlivost, nicméně vyžaduje nejvyšší nároky na správu a údržbu sítě a také nejvyšší náklady na sestavení sítě. Tyto topologie se používají hlavně v kritických aplikacích, kde je nutná maximální spolehlivost a redundance. Ukázka tohoto typu topologie, viz obrázek níže.



Obr. 2.25: Ukázka propojené topologie (výše) a plně propojené topologie (níže).

Hybridní topologie (Hybrid Topology) představuje směs různých typů topologií, které se vzájemně kombinují. Tyto topologie spojují výhody jednotlivých typů a nabízejí tak vyšší flexibilitu a spolehlivost. Tyto topologie se často používají v rozsáhlejších sítích, kde je potřeba se vyhnout nevýhodám jednotlivých typů topologií. V hybridní topologii mohou být například části sítě vytvořeny jako hvězdicová topologie a jiné části jako kruhová. Tyto části mohou být pak propojeny dohromady jako sběrnicová topologie. Tento typ topologie má tak výhody, jako je snadné připojení jednotlivých uzlů, vysokou spolehlivost a flexibilita. Hybridní topologie také umožňuje škálovat síť s rostoucími požadavky. Výhodou hybridní topologie je, že lze využít jednotlivé výhody jednotlivých topologií a vytvořit tak vysoce spolehlivou a flexibilní síť. Nevýhodou této topologie je náročnost jejího nastavení a složitá správná konfigurace, zejména v případě větších sítí. Navíc může být vyšší náročnost na diagnostiku a opravu v případě poruchy. Ukázka tohoto typu topologie, viz obrázek níže.



Obr. 2.26: Ukázka hybridní topologie sítě.

2.3.4 Typy sítí dle velikosti a geografického rozsahu

Je důležité rozlišovat sítě v rámci jejich rozsahu a geografického rozložení v průmyslových aplikacích, protože každý typ sítě má své specifické vlastnosti a výhody/nevýhody v závislosti na svém rozsahu a geografickém rozložení. Rozlišování těchto sítí v průmyslových aplikacích umožňuje správně volit síť pro konkrétní účely a zajistit, že bude mít dostatečný výkon a bezpečnost pro danou aplikaci. Tyto sítě také ovlivňují, jak budou zařízení komunikovat, a jak bude možné provádět vzdálenou diagnostiku a údržbu. Výběr správné sítě je tedy klíčový pro úspěšnou implementaci průmyslových aplikací. Dnes již existuje značné množství typů sítí dle rozlohy, hlavními zástupci jsou:

Nano sítě ($\approx$ stovky nm, okolí nanozařízení) jsou sítě, které se vyskytují v okolí nanozařízení a mohou být použity k řízení a komunikaci mezi různými mikro- a nanosystémy. Tyto sítě jsou vyznačeny velmi nízkými rychlostmi přenosu a omezeným dosahovým rozsahem kolem 100 nm. Výhody těchto sítí jsou, že jsou velmi malé a mohou být integrovány do malých zařízení, jako jsou mikročipy, nanosenzory a podobně. Tyto sítě také umožňují řízení a monitorování různých mikro- a nanosystémů, což je užitečné v průmyslových aplikacích, jako jsou například automatické řízení výrobních linek. Nevýhody nano sítí jsou, že jsou velmi omezené v dosahovém rozsahu a přenosové rychlosti, což může být limitující pro některé průmyslové aplikace, které vyžadují vyšší rychlost a dosah. Tyto sítě také vyžadují velmi specializované zařízení a vysoké náklady na vývoj a integraci do průmyslových aplikací. V závěru lze říci, že nano sítě jsou vhodné pro specifické průmyslové aplikace, jako je řízení a monitorování mikro- a nanosystémů, ale mohou být omezené pro větší průmyslové aplikace, kde je potřeba vyšší rychlost a větší dosah.

Komunikace v blízkém poli (Near Field Communication, NFC) ($\approx$ desítky cm, okolí zařízení) je typ sítě, která se používá k blízkému komunikačnímu styku mezi zařízeními v okolí zařízení, tj. v rozmezí asi 10 cm. Tyto sítě se často používají k přenosu dat mezi mobilními telefony, platebními kartami, inteligentními kartami a dalšími zařízeními. V průmyslových aplikacích se NFC sítě mohou používat k rychlému a snadnému přenosu dat mezi zařízeními, například k automatickému spárování zařízení, nebo k rychlému přenosu informací o stavu. Mezi klady NFC sítí patří snadné použití, malá velikost a nízké nároky na energii, což z nich činí vhodnou volbu pro mnoho malých zařízení. Nevýhody zahrnují malý dosah a malou rychlost přenosu dat, což může být omezující pro některé aplikace. Navíc, protože NFC sítě vyžadují blízký kontakt mezi zařízeními, mohou být náchylnější k rušení a jiným typům interference.

Síť v blízkosti těla (Body Area Network, BAN) ($\approx$ jednotky m, okolí těla) jsou sítě, která se vyskytují v okolí lidského těla a používají se k přenosu dat mezi zařízeními, která jsou připojena k tělu, jako jsou například chytré hodinky, fitness náramky nebo sluchátka. Tyto sítě se používají k přenosu malých objemů dat a informací, jako jsou například údaje o zdravotním stavu, nebo kaloriích spálených při cvičení. V rámci průmyslových aplikací mohou BAN sítě sloužit k monitorování zdravotního stavu pracovníků, kteří jsou vystaveni zvýšenému riziku, jako jsou například zaměstnanci v rizikových oborech, jakými jsou například hornictví, těžba a další. Tyto sítě také mohou být použity k monitorování a sledování pracovního výkonu a efektivity, což může pomoci optimalizovat výrobní procesy a zlepšit bezpečnost práce. Mezi hlavní výhody BAN sítí patří jejich malá velikost a nízká spotřeba energie, což umožňuje jejich integraci do malých zařízení, která jsou nositelná přímo na těle. Tyto sítě také poskytují vysokou míru flexibility a mobility, což umožňuje uživatelům přenášet data mezi zařízeními bez ohledu na to, kde se nacházejí. Nevýhody BAN sítí zahrnují nízkou šířku pásma, což může vést k nízké rychlosti přenosu dat a omezenému množství dat, které lze přenést. Tyto sítě také mohou být náchylné k rušení ze strany jiných zařízení, což může vést ke ztrátě dat či jiným problémům.

Osobní síť (Personal Area Network, PAN) ($\approx$ <10 m, pracovní místo) je typ sítě, který se obvykle používá k propojení zařízení v blízkosti jednoho uživatele, jako jsou například chytré telefony, tablety, počítače a další zařízení. Tyto sítě se často používají k propojení zařízení v rámci kancelářských prostor, nebo k propojení zařízení v domácnostech. V rámci průmyslových aplikací mohou být PAN sítě použity ke komunikaci mezi zařízeními používanými v průmyslových prostředích, jako jsou například pracovní stanice, mobilní terminály, atd. Tyto sítě mohou být také použity k propojení zařízení v rámci průmyslových budov, například k propojení pracovních stanic s řídicími systémy. Výhody použití PAN sítí v rámci průmyslových aplikací zahrnují snadnou implementaci a nízké náklady. Tyto sítě také poskytují vysokou flexibilitu a možnost připojit velké množství zařízení. Nevýhodou může být omezený dosah sítě, což může být problém při komunikaci mezi zařízeními vzdálenými větší vzdáleností. Také mohou být někdy k dispozici pouze nízkorychlostní sítě, což může omezovat rychlost přenosu dat.

Domácí síť (Home Area Network, HAN) ($\approx$ desítky m, domácnost) se často používá pro komunikaci mezi zařízeními v domácnosti, jako jsou počítače, telefony, televize, hračky a chytrá domácí zařízení. Tyto sítě poskytují vysokou rychlost přenosu dat pro komunikaci mezi zařízeními v domácnosti. V rámci průmyslových aplikací mohou být HAN sítě použity pro spojení mezi průmyslovými zařízeními v kancelářích nebo výrobních budovách, kde se požaduje vysoká rychlost přenosu dat. Tyto

sítě mohou být použity pro vzdálený přístup k průmyslovým zařízením z kanceláře či z jiného místa. Avšak vzhledem k tomu, že HAN sítě nejsou navrženy s ohledem na bezpečnost, nemohou být použity pro kritické průmyslové aplikace, jako je např. řízení toku energie v elektrárnách nebo řízení výrobního procesu.

Lokální síť (Local Area Network, LAN) ($\approx$ stovky m, budova) je typ sítě, který slouží k propojení počítačů a dalších zařízení v jedné lokalitě, jako je budova, kancelář nebo škola. Tyto sítě se často používají ke sdílení informací a zdrojů, jako jsou tiskárny, soubory a internetové připojení. LAN sítě se často implementují pomocí technologií jako je Ethernet, Wi-Fi nebo Bluetooth. V rámci průmyslových aplikací mohou být LAN sítě použity k propojení průmyslových automatů a kontrolérů v rámci jedné výrobní haly nebo budovy. Tyto sítě mohou umožnit automatickou komunikaci mezi automaty a kontroléry, což může vést ke zlepšení efektivity a produktivity výroby. LAN sítě také umožňují vzdálenou správu a diagnostiku průmyslových zařízení. Nevýhodou LAN sítí může být jejich omezený dosah, který se vztahuje pouze na určitou lokalitu, jako je budova nebo výrobní hala.

Kampus síť (Campus Area Network, CAN) ($\approx$ jednotky km, kampus) jsou sítě, které se obvykle používají pro propojení různých budov na stejném areálu, jako je například univerzita nebo výrobní kampus. Tyto sítě se často skládají z lokálních sítí LAN v jednotlivých budovách a vysokorychlostního spojení mezi těmito budovami. V rámci průmyslových aplikací mohou být CAN sítě použity pro propojení různých výrobních zón nebo provozů na stejném areálu. Tyto sítě umožňují sdílet informace, jako jsou například informace o stavu výroby, produkční data a systémové zprávy. To může pomoci zlepšit efektivitu a produktivitu výrobního procesu. Nevýhodou CAN sítí může být jejich vysoká náročnost na správu a údržbu, zejména v případě, že se jedná o rozlehlý kampus s mnoha budovami. Tyto sítě také mohou být náchylné k poruchám, což může mít vliv na spolehlivost a dostupnost průmyslových aplikací. Přesto mohou být CAN sítě velmi užitečné v rámci průmyslových aplikací, pokud jsou správně nastaveny a spravovány. Tyto sítě mohou poskytnout vysokou rychlost a spolehlivost pro přenos dat mezi jednotlivými budovami na stejném areálu, což může přispět k lepší koordinaci a řízení průmyslových procesů.

Metropolitní síť (Metropolitan Area Network, MAN) ($\approx$ desítky km, město) jsou určeny pro komunikaci mezi různými lokálními sítěmi a sítěmi většího rozsahu, jako jsou rozlehlé sítě (WAN). Tyto sítě se často používají v průmyslových aplikacích pro přenos dat mezi různými odděleními v jednom městě, nebo mezi městy v jedné oblasti. Výhody metropolitních sítí zahrnují vyšší propustnost a lepší kapacitu pro přenos dat než lokální sítě, stejně jako možnost vzdáleného monitorování

a řízení zdrojů. Tyto sítě také umožňují lepší spolupráci mezi různými odděleními v rámci jednoho podniku. Nevýhody těchto sítí mohou zahrnovat vyšší náklady na implementaci a správu, stejně jako větší závislost na externích dodavatelích síťových služeb. V rámci průmyslových aplikací mohou metropolitní sítě být použity pro komunikaci mezi různými průmyslovými zařízeními, jako jsou například továrny, sklady a logistická centra, kde se přenášejí velké objemy dat. Tyto sítě mohou být také využity pro komunikaci se vzdálenými pracovišti, jako jsou například pobočky, nebo filiálky, kde se provádí výroba nebo služby pro zákazníky.

Rozlehlé sítě (Wide Area Network, WAN) ($\approx$ stovky km, kampus) jsou určeny pro široké geografické oblasti a poskytují připojení mezi sítěmi v různých místech. Tyto sítě mohou být použity pro spojení mezi budovami v jednom městě, mezi městy v jednom státě, mezi státy na jednom kontinentu, a dokonce i mezi kontinenty. WAN sítě jsou často používány pro připojení mezi vzdálenými kancelářemi nebo pro vzdálený přístup k datům a aplikacím. V rámci průmyslových aplikací, WAN sítě mohou být použity pro řízení vzdálených zařízení nebo pro vzdálený přístup k datům a informacím o výrobním procesu. Tyto sítě mohou být také použity pro vzdálené monitorování a řízení bezpečnostního výkonu a zabezpečení. Nevýhodou WAN sítí může být vyšší latence a nižší rychlost přenosu dat v důsledku větší vzdálenosti, což může mít vliv na rychlost a spolehlivost řízení průmyslových procesů. Kromě toho mohou být náklady na implementaci a správu WAN sítí vyšší v porovnání s jinými typy sítí.

Globální sítě (Global Area Network, GAN) jsou sítě, které pokrývají oblasti za hranice jednoho města, státu nebo dokonce kontinentu. Tyto sítě se často používají k propojení vzdálených lokalit, jako jsou mezinárodní kanceláře, výrobní závody a kampusy. V rámci průmyslových aplikací jsou GAN/IAN sítě často používány ke sledování a řízení vzdálených zařízení a výrobních procesů. Tyto sítě umožňují spojení mezi centrálními kontrolními systémy a vzdálenými zařízeními a mohou být využity k vzdálené diagnostice a údržbě. Tyto sítě také poskytují možnosti pro vzdálené monitorování a řízení dodávek energie, vodního hospodářství a dalších průmyslových procesů. Mezi hlavní výhody GAN/IAN sítí patří schopnost propojit vzdálené lokality, což umožňuje centralizované řízení a sledování výrobních procesů. Tyto sítě také umožňují vzdálenou diagnostiku a údržbu, což snižuje náklady a zvyšuje efektivitu. Mezi nevýhody patří vyšší náklady na implementaci a údržbu, jakož i vyšší citlivost na poruchy a bezpečnostní rizika, jako jsou útoky na síť. Tyto sítě také vyžadují vysokou úroveň zabezpečení, aby se zajistilo, že data nemohou být ztracena nebo zneužita.

Sítě blízkého okolí (Near-me Area Network, NAN) jsou speciálním případem sítí v blízkém okolí. Tyto sítě se používají k propojení zařízení v okolí, jako jsou například senzory, inteligentní domácí zařízení a další. NAN sítě zahrnují zařízení v rozsahu několika metrů. Mezi výhody těchto sítí patří nízké náklady na provoz a instalaci, možnost propojení velkého počtu zařízení v blízkém okolí, široké spektrum možností použití v oblasti IoT a inteligentních domácností. Mezi nevýhody těchto sítí pak patří omezená komunikační vzdálenost, omezený přenosový poměr a rychlost přenosu dat a vysoká citlivost na rušení z jiných zdrojů. V rámci průmyslových aplikací jsou NAN sítě využívány především v oblasti inteligentních domácností, kde propojují různá zařízení, jako jsou například chytré termostaty, inteligentní žárovky a další. NAN sítě jsou také často využívány pro automatizaci průmyslových procesů, jako jsou například automatické řízení skladů, monitorování, a řízení výrobních linek a další.

Rádiová přístupová síť (Radio Access Network, RAN) je další speciální typ sítě, který se používá k připojení mobilních zařízení, jako jsou telefony, tablet a další přenosná zařízení, k telekomunikačním sítím. Tyto sítě používají rádiové vlny k přenosu dat mezi zařízením a sítí. RAN sítě jsou klíčovou součástí mobilních sítí a umožňují bezdrátový přístup k internetu všude tam, kde je signál dostupný. V rámci průmyslových aplikací se RAN sítě mohou používat k připojení mobilních pracovních zařízení, jako jsou mobilní telefony a tablety, k průmyslovým sítím. Tyto sítě mohou být také využívány pro přístup k vzdáleným systémům řízení a monitorování, jako jsou systémy dispečerského řízení a sběru dat (Supervisory Control And Data Acquisition, SCADA). RAN sítě však mohou být také náchylné k rušení a narušení, což je nutné vzít v úvahu při jejich použití v průmyslových aplikacích. Výhody RAN sítí zahrnují schopnost připojit se k internetu kdekoliv na cestách, možnost připojení k vzdáleným systémům řízení a monitorování a snadnost použití pro uživatele. Nevýhody zahrnují náchylnost k rušení a narušení a nutnost přístupu k signálu, aby bylo možné použít tuto síť. Je nutno zmínit, že tento typ sítí v sobě zahrnuje mnohdy bezdrátové varianty již zmíněných sítí, např. bezdrátová PAN (Wireless PAN, WPAN), bezdrátová BAN (Wireless BAN, WBAN), bezdrátová LAN (Wireless LAN, WLAN), bezdrátová MAN (Wireless MAN, WMAN), bezdrátová WAN (Wireless WAN, WWAN), ale také sítě dle aplikace jako mobilní Ad-Hoc sítě (Mobile Ad-Hoc Network, MANET), bezdrátové senzorické sítě (Wireless Sensor Network, WSN) či bezdrátové sítě průmyslové automatizace (Wireless Industrial Automation Network, WIAN).

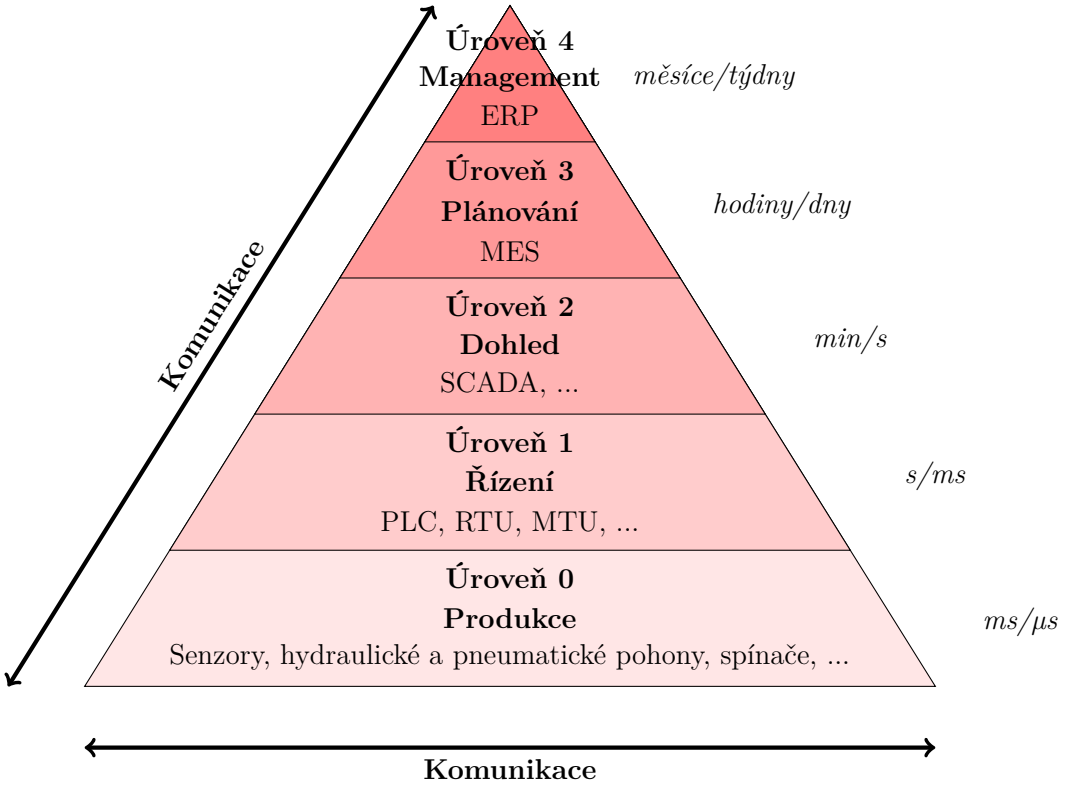
2.4 Architektura OT

V rámci OT, architektura označuje celkový návrh a strukturu systémů používaných k řízení a monitorování fyzických procesů. Efektivní OT architektura je zásadní pro zajištění spolehlivého a bezpečného provozu průmyslových procesů a zároveň podporuje obchodní cíle, jako je efektivita, produktivita a bezpečnost. Architektura OT zahrnuje integraci různých komponent, včetně hardwaru, softwaru a komunikačních systémů, stejně jako organizační a provozní procesy, které tyto komponenty podporují. Architektura musí být navržena tak, aby splňovala specifické potřeby a požadavky organizace s přihlédnutím k faktorům, jako je velikost a složitost systému, typ řízených procesů a požadovaná úroveň zabezpečení a bezpečnosti. Aktuálně neexistuje žádný univerzální přístup k architektuře OT, který by se dal popsat jako tzv. "one-fits-all"řešení. Architektura, která je nejvhodnější pro konkrétní organizaci, bude záviset na řadě faktorů, včetně konkrétních řízených procesů, velikosti a složitosti systému, regulačních požadavků a požadavků na dodržování předpisů a obchodních cílů organizace. Jedním z přístupů k architektuře OT, který si získal široké přijetí, je použití vrstvené architektury, která může poskytnout rámec pro organizaci různých součástí systému OT. Tento přístup obvykle zahrnuje rozdělení systému do několika vrstev, z nichž každá má specifickou sadu funkcí a odpovědností. Přesný počet a složení vrstev se může lišit v závislosti na konkrétních potřebách organizace, ale některé běžné vrstvy v architektuře OT mohou zahrnovat následující:

- **Fyzická vrstva**, která zahrnuje senzory, akční členy a další fyzické komponenty, které se používají k řízení a monitorování fyzických procesů.
- **Řídicí vrstva**, která zahrnuje řadiče a další zařízení, která se používají ke správě fyzických komponent a zajišťují, že fungují v rámci specifikovaných parametrů.
- **Dohledová vrstva**, která zahrnuje rozhraní člověk-stroj (HMI) a další softwarové nástroje používané k monitorování a řízení systému a také k poskytování dat a analýz nadřízenému managementu.
- **Podniková vrstva**, která zahrnuje obchodní systémy a procesy, které se používají k řízení organizace jako celku, a může zahrnovat funkce, jako je řízení zásob, logistika a řízení dodavatelského řetězce.

Tento vrstvený přístup může poskytnout užitečný rámec pro organizaci různých součástí OT systému a zajištění jejich efektivní spolupráce. Je však důležité mít na paměti, že přesné složení a struktura každé vrstvy bude záviset na konkrétních potřebách a požadavcích organizace a může být nutné ji odpovídajícím způsobem upravit. I přesto, že tedy neexistuje jeden model, který by byl vhodný pro všechny případy, byl výše uvedený vrstvený model tedy přijat širokou škálou odborníků, protože z něj lze čerpat jak hierarchické složení, tak síťové složení atd. Z tohoto

důvodu vznikly tzv. referenční modely, kde jedním z nich je např. model popsaný v rámci standardu ANSI/ISA-95. Jedná se o pyramidové schéma jednotlivých úrovní automatizace, které je zobrazeno na obr. 2.27. V rámci jednotlivých úrovní probíhá jak horizontální, tak i vertikální obousměrná komunikace. V rámci obrázku jsou také zobrazeny jednotlivé příklady zařízení i účelu jednotlivých úrovní, kde v neposlední řadě můžeme také vidět časovou náročnost procesů v jednotlivých úrovních.



Obr. 2.27: Pyramidové referenční schéma automatizačních úrovní dle ANSI/ISA-95.

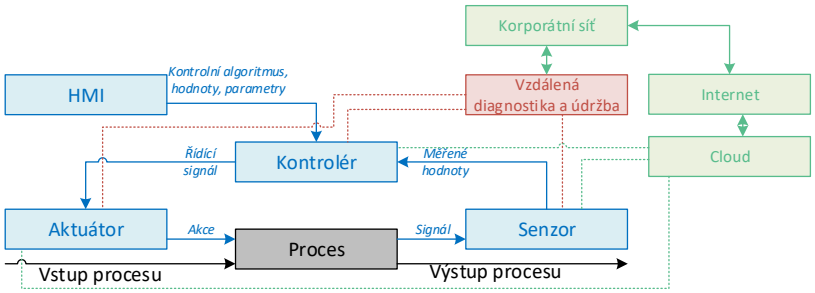
Referenční model je tak rozdělen do pěti úrovní:

- **Úroveň 4 Management** (Management level). Jedná se o úroveň, kde jsou integrovány systémy plánování podnikových zdrojů (ERP, Enterprise Resource Planning). Na této úrovni shromažďuje vysoký management data, analyzuje je a reaguje na vzniklé potřeby společnosti. Jedná se obvykle o sadu různých počítačových softwarových aplikací, které dovolují vnitřní pohledu do fungování společnosti do nejmenších detailů a dovoluje tedy sledování od výroby po prodej, nákup, financování, výplaty, efektivitu, produktivitu a další. Tato úroveň tak dovoluje vrcholovému managementu transparentnost chodu společnosti.
- **Úroveň 3 Plánování** (Planning level). Jedná se o úroveň, která integruje systémy jako řízení specializované expedice (Just-In-Sequence, JIS – Just-In-Time, JIT – Transportation Management Systems, TMS), řízení intralogistiky (Warehouse Management System, WMS), řízení výroby (Manufacturing

Execution System, MES), řízení kvality (Quality Management System, QMS), řízení údržby (Enterprise Asset Management, EAM) a pokročilé plánování (Advanced Planning and Scheduling, APS) či řízení provozu (MOM, Manufacturing Operation Management). Jedná se o úroveň monitorující celý výrobní proces v dané společnosti od surovin až po hotový produkt. To umožňuje managementu přijímat rozhodnutí na základě těchto informací jako např. upravovat stavy objednávek surovin či přepravení plány. Vše tedy na základě skutečných dat získaných z úrovní níže.

- **Úroveň 2 Dohled** (Supervisory level). Tato úroveň zahrnuje systémy vzdáleného i lokálního dohledu, příkladem může být systémy pro SCADA či distribuované řídicí systémy (Distributed Control System, DCS), které jsou určeny k centrálnímu řízení více systému z jednoho bodu, může se např. jednat o dohledové centrum nad přenosovou sítí zahrnující jednotlivé trafostanice, může se jednat o jednotlivé čističky vody apod. Je nutno zde rozlišovat HMI, které je spíše na lokální úrovni níže. Tato úroveň skutečně slouží pro centrální řízení více systémů dohromady, tak aby mohl být dán větší kontext jednotlivým procesům.
- **Úroveň 1 – Řízení** (Control level). Jedná se o úroveň zahrnující lokální HMI, PLC, MTU, RTU a další zařízení, které mají za účel řídit jednotlivé fyzické prvky úrovně 0, tedy zařízení vykonávající již skutečné fyzické úkony.
- **Úroveň 0 – Produkce** (Field level). Tato nejnižší úroveň obsahuje jednotlivé prvky, které již vykonávají fyzické úkony na základě signálů řízení z vyšších úrovní. Jedná se o akční členy, senzory, motory, hydraulické a pneumatické jednotky, různé spínače a další.

Tento model již částečné zahrnuje i IT systémy, kde pro úpravu např. našeho představeného modelu OT bychom mohli tento model jednoduše upravit jako na obr. níže, který by pak lépe odpovídal aktuální situaci.



Obr. 2.28: Aktualizované schéma OT modelu o IT součásti.

S IIoT/IoT a cloudovými službami se architektura značným způsobem rozrůstá

a síťové se mění značným způsobem. Architektura nicméně již řadu let stále zůstává stejná a model vrstev se používá dodnes. Z tohoto pohledu zavádění digitálních a IT systému do OT přináší tzv. IT/OT konvergenci. Konvergence IT/OT se týká sloučení systémů IT a OT v průmyslovém a výrobním prostředí. Historicky byly IT a OT systémy vyvíjeny a spravovány odděleně, s různými týmy, technologiemi a cíli. V posledních letech je však konvergence těchto systémů stále důležitější, protože společnosti se snaží optimalizovat své operace, zlepšit efektivitu a snížit náklady. Historii konvergence IT/OT lze vysledovat až do počátků výpočetní techniky, kdy byly k řízení průmyslových procesů poprvé použity sálové počítače. Až v 80. a 90. letech se však termín „průmyslová automatizace“ začal široce používat, protože byly vyvinuty nové technologie, jako jsou programovatelné logické automaty (PLC) a distribuované řídicí systémy (DCS) pro řízení průmyslových procesů. Během této doby zůstaly IT a OT systémy do značné míry oddělené, přičemž IT systémy se zaměřovaly na obchodní procesy a správu dat a OT systémy se zaměřovaly na řízení a automatizaci fyzických procesů. S tím, jak se však internet a další digitální technologie staly více rozšířenými, si společnosti začaly uvědomovat potenciální výhody integrace těchto systémů a začal se objevovat koncept konvergence IT/OT. Na počátku 21. století vznik průmyslového internetu věcí (IIoT) dále urychlil konvergenci IT a OT systémů. IIoT se týká použití zařízení a senzorů připojených k internetu ke sběru a analýze dat z průmyslových procesů a použití těchto dat ke zlepšení efektivity, snížení nákladů a optimalizaci provozu. To vyžadovalo užší spolupráci mezi IT a OT týmy a také vývoj nových technologií a standardů na podporu této konvergence. V dnešní době je konvergence IT a OT systémů stále důležitější pro společnosti, které chtějí zůstat konkurenceschopné a efektivní v rychle se měnícím podnikatelském prostředí. Společnosti investují do nových technologií, jako je edge computing, umělá inteligence a pokročilá analytika, aby podpořily tuto konvergenci, a vyvíjejí nové organizační struktury a procesy na podporu spolupráce mezi IT a OT týmy. Celkově konvergence IT a OT systémů představuje významnou transformaci v průmyslovém a výrobním prostředí a má potenciál odemknout významné výhody z hlediska efektivity, produktivity a snížení nákladů.

3 Případové studie a demonstrace

Tato část ukazuje praktickou aplikaci teorií a konceptů diskutovaných v předchozích částech. Hlavním cílem je ukázat, jak se komponenty OT sítí, komunikační techniky a technologie a architektura OT sítí spojují ve scénářích reálného světa. Sekce je rozdělena do tří podsekcí, z nichž každá představuje demonstraci navrženou v laboratoři pro konkrétní odvětví:

- První sekce zdůrazňuje použití OT v případu balicí výrobní linky a ukazuje integraci různých komponent, jako jsou procesy, senzory, akční členy, ovladače, HMI a vzdálená diagnostika, aby se vytvořil účinný a efektivnější výrobní proces.
- Druhá sekce se zaměřuje na aplikaci OT v městské čistírně vody, přičemž zdůrazňuje význam komunikačních a přenosových technologií, topologií a referenčních modelů pro zajištění efektivního a bezpečného provozu procesu úpravy vody.
- Třetí sekce představuje ukázku použití OT v pivovaru, ukazuje integraci IT/OT v chemickém/potravinářském průmyslu. Ukázka zdůrazňuje důležitost konvergence IT/OT a roli referenčních modelů při vytváření efektivního výrobního procesu.

Každá sekce je rozdělena na shrnutí, použité komponenty, vstupní kritéria (včetně předpokladů, vývoje a návrhu), technický popis, testování a verifikace. Celkově tato kapitola poskytuje komplexní a praktický přístup k pochopení aplikace OT ve scénářích reálného světa. Je nutno zmínit, že tyto demonstrátory byly vytvořeny v rámci výzkumné činnosti pro projekt reg. č. FV40366 [APro11] (Datový monitoring pro zvýšení spolehlivosti procesů chytrých továren), podpořený Ministerstvem průmyslu a obchodu České republiky. Jedná se o komplexní průmyslový testovací polygon, který umožňuje edukaci, testování a výzkum nových moderních průmyslových řešení, a to jak v rámci řešení otázek interoperability, tak i otázek bezpečnosti či potřeb generování datových sad pro umělou inteligenci, které dnes představují hlavní překážku v budování efektivních algoritmů. Samotný polygon byl dále také komercializován, a to prostřednictvím firmy GreyCortex s.r.o. a firmy Vodafone Czech Republic a.s. Polygony i jeho data byly využity v rámci prestižních publikací ve vlastních impaktovaných časopisech a mezinárodních konferencích, ale také ve studentských soutěžních příspěvcích (vytvořených pod vedením autora). Výuka v rámci vytvořených prostředí pak navazuje pomocí kybernetické arény (BUTCA) přes výukové scénáře (v oboru Informační bezpečnosti) a dále také prostřednictvím bakalářských, diplomových a doktorských prací. Bližší popis pro přínos těchto výsledků byl již představen v předchozích kapitolách, viz mj. příslušná kap. *Přínos práce*.

3.1 Příklad I: Průmyslová balicí smyčka

3.1.1 Shrnutí

Bezpečnostní incidenty jsou v rámci průmyslových zařízení stále více časté. To je způsobeno zejména propojením IT (Information Technology) a OT (Operational Technology) infrastruktur. Toto propojení přispělo nejen k snazší dosažitelnosti těchto sítí a možnému propojení k jiným OT infrastrukturám skrze IT sítě, ale vystavilo také OT sítě útokům vyskytujícím se v IT sítích. Aby bylo možné zajistit vyšší úroveň bezpečnosti, je nutné kombinovat nejnovější technologie a přístupy. K umožnění tohoto vývoje je však zapotřebí dat, na základě kterých bude tento výzkum proveden. Pro tyto účely bylo vytvořeno testovací prostředí (testbed) Balicí smyčky. Vytvořená balicí smyčka slouží k vytváření datových sad. Hlavní výhodou vytvořené smyčky je velmi blízké přiblížení vytvořeného pracoviště reálnému prostředí. Další výhodou je vytvoření softwarových verzí, které jsou z pohledu generovaných dat totožné s těmi fyzickými.

Z provedeného experimentálního testování vyplývá, že tento způsob přístupu k průmyslovým pracovištím, kde vznikají i jejich softwarové verze je velmi efektivní z pohledu generování rozmanitých datových sad. Nedochází totiž *jen* k přenosu nevýznamných hodnot, které se snaží simulovat reálný průmyslový provoz. Naopak z důvodu založení softwarové verze na základě znalostí získaných z fyzického pracoviště umožňuje generovat totožná data, která lze považovat za průmyslový provoz s přidanou hodnotou případného reálného využití.

Díky existenci fyzického testovacího pracoviště je možné provádět vývoj a výzkum pomocí různých experimentálních testování a sledování dopadů testování nejen na reálných zařízeních. Lze tak vhodně testovat nejen efektivitu detekčních mechanismů, ale i pozorovat dopad na jednotlivých fyzických zařízeních. Z důvodu vytvoření softwarových dvojčat je možné nejen vytvářet komplikovanější zapojení, ale i testovat dopad testů na větším množství zařízení bez nutnosti využití fyzických prvků.

Účely testovacího prostředí:

- Dlouhodobý sběr dat ze standardního i nestandardního provozu.
- Vytvoření datových sad.
- Průzkum vektorů útoků.
- Simulace kybernetických útoků a anomálií.
- Simulace anomálií na fyzických zařízeních.
- Optimalizace výsledných řešení.

3.1.2 Použité komponenty

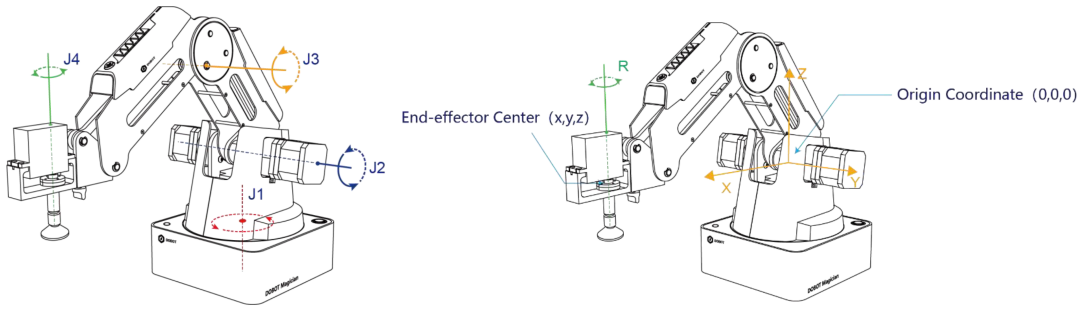
K realizaci navrženého pracoviště byly vybrány zařízení Dobot Magician. Výběr byl proveden na základě široké škály komponent, pomocí kterých lze pracoviště vhodně doplnit o další funkce. Zároveň je možné provádět jejich ovládání skrze Python knihovnu pydobot [3, 5].

Vybrané komponenty:

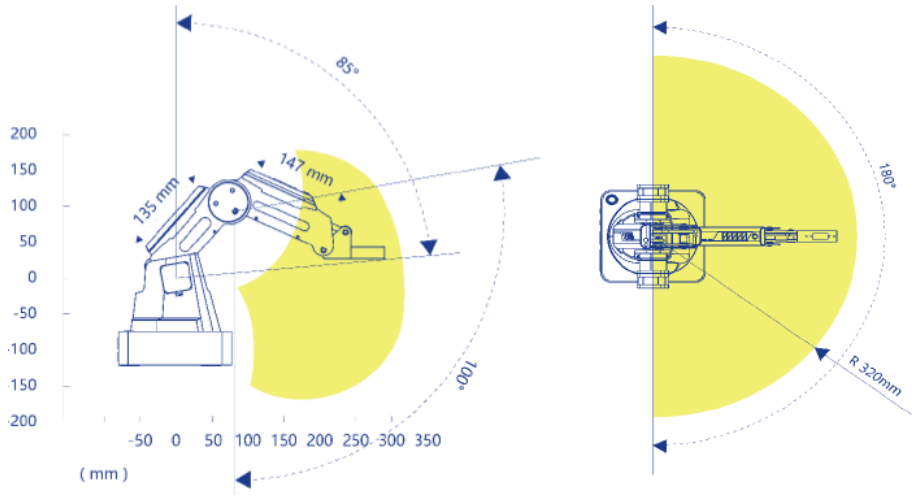
- 3x Robotická paže Dobot Magician
- RGB senzor
- Dopravníkový pás
- Kolejnice k pojezdu robotické paže
- 5x Pěnová kostka (červená, modrá, zelená a žlutá barva)
- Plastový úložný box + víko
- Plastová krabička + bublinková fólie
- Stůl 160 x 75 cm

Robotická paže

Dobot Magician je vybaven čtyřmi motory, viz obr 3.1, které ovládají jeho pohyb a definují hlavici (hlavní prvek k uchopení předmětů) v kartézském souřadnicovém systému, viz obr 3.1. Motory *J1*, *J2* a *J3* zajišťují pohyb hlavice, motor *J4* zajišťuje rotaci uchopeného předmětu pomocí hlavice. Dobot je však ovládán skrze kartézské souřadnice a pohyby motorů jsou automaticky dopočítány pro větší uživatelskou přívětivost. Výchozí ovládání je tak prováděno prostřednictvím čtyř souřadnic, a to x, y, z pro definování pozice hlavice a r pro natočení hlavice. V případě využití kolejnice pro paži 1 musela být přidána další souřadnice l pro udání pozice paže ve vztahu ke kolejnici. Dobota definuje široký pracovní prostor, v rámci kterého je schopen práce, viz obrázek 3.2.



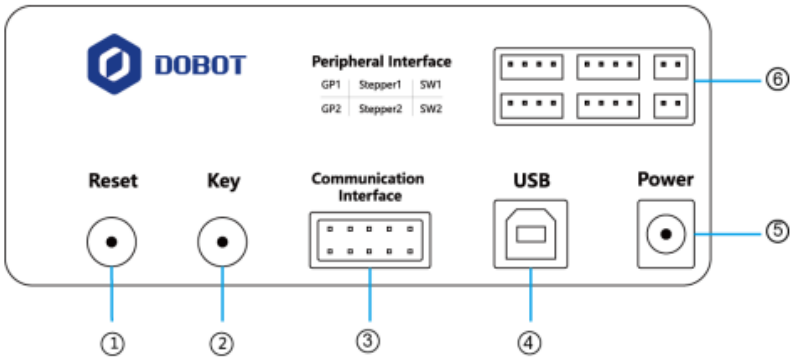
Obr. 3.1: Robotická paže – vlevo umístění motorů, vpravo definování kartézského souřadnicového systému [2].



Obr. 3.2: Robotická paže – pracovní prostor [2].

Rozhraní – základna

Dobot je vybaven několika komunikačními rozhraními na zadní straně základny a na předloktí paže. K dobotu je možné připojit modul pro komunikaci pomocí technologie Bluetooth nebo pomocí Wi-Fi. Lze připojit periferie jako jsou například vakuová pumpa, senzor, pohyblivý pás atd. Zadní stranu základny dobota s rozhraními lze vidět na obrázku 3.3. Popis jednotlivých rozhraní a tlačítek lze vyčíst v tabulce 3.1 a detailnější popis rozhraní pro periferie v tabulce 3.2.



Obr. 3.3: Robotická paže – základna [2].

Tab. 3.1: Robotická paže, rozhraní – základna.

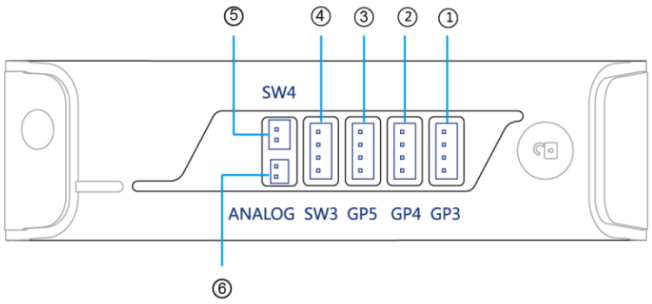
Číslo	Název	Funkce	Popis funkce
1	Reset tlačítko	Restartování programu	Během resetování svítí žlutá kontrolka. Reset trvá 5 sekund a po restartu se kontrolka rozsvítí zeleně (vše v pořádku).
2	Key tlačíko	Tlačítko funkčního klíče	Spuštění offline programu (krátké stisknutí) nastavení výchozí pozice (*home* proces, stisknutí déle než 2 sekundy).
3	Komunikační rozhraní	Bezdrátová komunikace	Pro připojení modulů pro komunikaci přes Bluetooth nebo Wi-Fi.
4	USB rozhraní	Připojení USB	Pro připojení pomocí USB k počítači.
5	Napájení	Napájení	rozhraní pro připojení napájení.
6	Rozhraní	Rozhraní pro periferie	Rozhraní pro připojení periferií jako vakuová pumpa, senzor atd.

Tab. 3.2: Robotická paže, rozhraní pro periferie.

Název	Funkce
SW1	Výkonové rozhraní vzduchového čerpadla; výstup 12V regulovatelného výkonu
SW2	Výstupní 12V regulovatelného výkonu
Stepper 1	Uživatelsky definované rozhraní pro krokování; rozhraní extruderu (režim 3D tisku)
Stepper 2	Uživatelsky definované rozhraní stepperu
GP1	Signální rozhraní vzduchového čerpadla; rozhraní snímače barev; rozhraní infračerveného senzoru; uživatelsky definované obecné rozhraní
GP2	Uživatelem definované obecné rozhraní

Rozhraní – paže

Na předloktí paže dobota (mezi motorem J3 a J4) se nachází další komunikační rozhraní. Jednotlivá rozhraní lze vidět na obr 3.4. Popis jednotlivých rozhraní lze vidět v tabulce 3.3.



Obr. 3.4: Robotická paže – předloktí [2].

Tab. 3.3: Robotická paže, rozhraní – robotická paže.

Číslo	**Název**	**Funkce**
1	GP3	Rozhraní koncového efektoru; servo rozhraní osy R; uživatelsky definované obecné rozhraní
2	GP4	Rozhraní automatického vyrovnávání, uživatelsky definované obecné rozhraní
3	GP5	Signální rozhraní laserového gravírování; uživatelsky definované obecné rozhraní
4	SW3	Rozhraní Hot end (režim 3D tisku); Výstupní 12V regulovatelného výkonu
5	SW4	Rozhraní ventilátoru (režim 3D tisku); Výkonové rozhraní laserového gravírování; Výstupní 12V regulovatelného výkonu
6	ANALOG	Termistorové rozhraní (režim 3D tisku)

Zobrazované barvy na základně

Základna obsahuje indikační LED. Tato LED umožňuje zobrazení celkem čtyř barev (červená, zelená, modrá a žlutá). Jednotlivá barva indikuje definovaný stav robotické paže, bližší popis je zobrazen v rámci tabulky 3.4.

Tab. 3.4: Jednotlivé barvy zobrazované základnou robotické paže.

Barva	Význam
Červená	paže se nemůže dostat na stanovené pozice – pozice je mimo pracovní prostor paže nebo hrozí poškození
Zelená	paže pracuje ve stanoveném pracovním prostoru
Modrá	pokud bliká modré světélko, dochází k nastavování ruky do pozice home – proces *home*
Žlutá	stav po spuštění před nastavením správného kartézského systému a pracovního prostoru paže

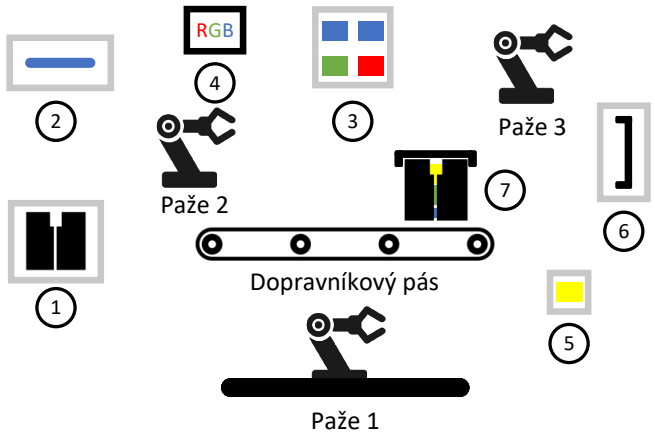
3.1.3 Vstupní kritéria, předpoklady, vývoj a návrh

Návrh experimentálního pracoviště, balicí proces

Experimentální pracoviště je navrženo tak, že obsahuje 7 přesně definovaných bodů/sektorů. Vizualizace pracoviště je zobrazena na obrázku 3.5. Pracoviště je obsluhováno třemi robotickými pažemi. Paže 1 je umístěna na kolejnici a je tak možný její pohyb po pracovišti. Paže 2 je vybavena RGB senzorem umožňující rozlišit barvu předmětu nacházejícího se nad senzorem. Paže 3 pohybuje dopravníkovým pásem. Navržený balicí proces nejprve pracuje s objektem *1* – box pro ukládání materiálů. Box je následně přenesen na dopravníkový pás pomocí paže 1. Do boxu, pomocí paže 2, je vložen ochranný materiál (*2*) ochraňující uložený materiál v boxu. V tomto kroku je spuštěn dopravníkový pás, který přemístí box blíže k sektoru *3*, kde je uložen materiál pro vložení do boxu. Za pomoci paže 2 je vybrán materiál ze sektoru *3*, ověřena jeho barva v sektoru *4* a v případě splnění definované podmínky na barvu materiálu je vložen do boxu. V případě nesplnění podmínky je materiál vrácen na původní pozici a je zvolen jiný. Po vložení materiálu do boxu opět dochází k posunu dopravníkového pásu blíže sektoru *5*. Tento sektor obsluhuje paži 1, která vloží další vybraný materiál ze sektoru *5* do boxu. Balicí proces zakončuje paže 3, která

provede uzavření boxu pomocí uchopení víka (sektor *6*), jeho rotaci a uložení na box. Tímto je balicí proces u konce a na stanovišti *7* se nachází box s definovaným materiálem připraven k dalšímu zpracování.

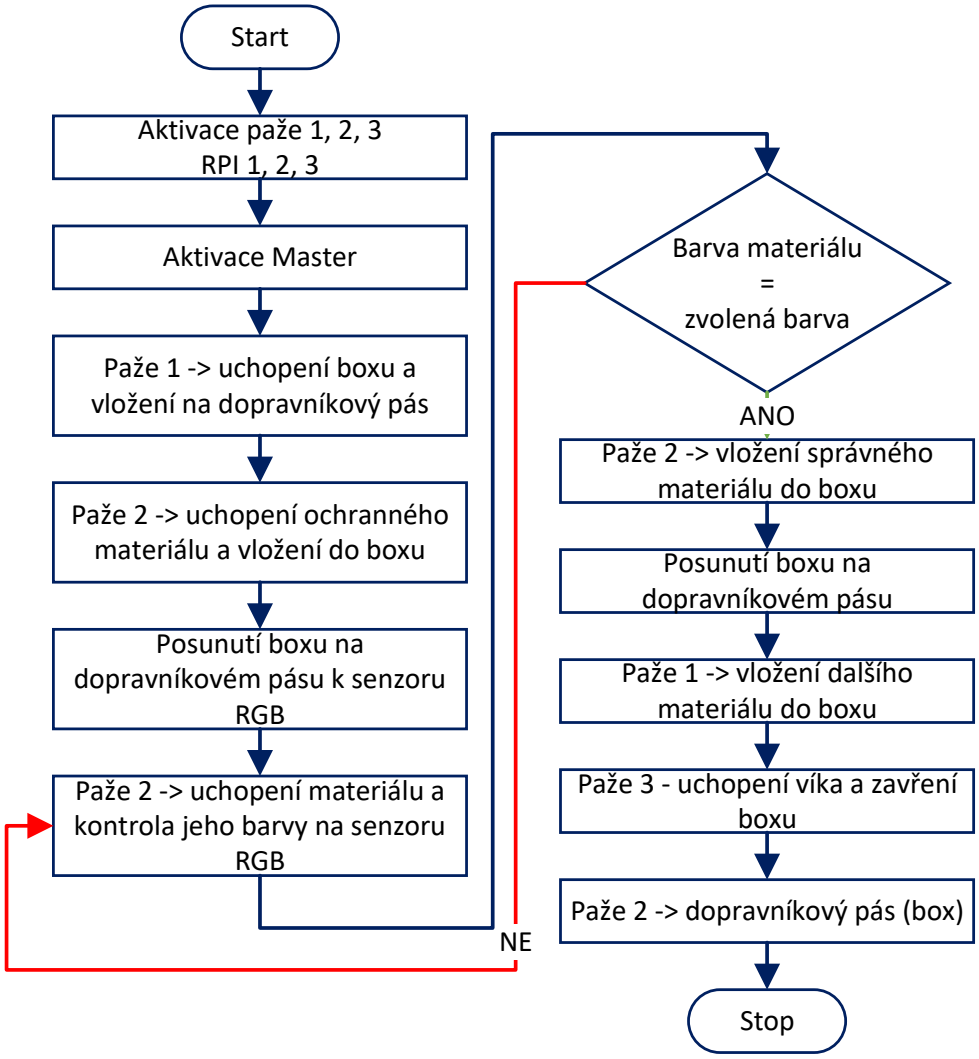
- Robotické paže:
  - Robotická paže 1 – upevněna na kolejnici
  - Robotická paže 2 – zajištění obsluhy paže a RGB senzoru
  - Robotická paže 3 – zajištění obsluhy paže a dopravníkového pásu
- Sektory:
  1. Box pro uložení materiálu
  2. Ochranný materiál
  3. Uložení materiálu pro následné vložení do boxu (sklad)
  4. Identifikace barvy materiálu
  5. Uložení materiálu pro následné vložení do boxu (sklad) 2
  6. Uložení víka pro uzavření boxu
  7. Finální pozice výsledného produktu k dalšímu zpracování



Obr. 3.5: Návrh experimentálního pracoviště.

Návrh jednotlivých kroků balicího procesu

Na obrázku 3.6 je zobrazen sled jednotlivých kroků, které jsou provedeny pro úspěšné dokončení balicího procesu. Jednotlivý postup je přesně definován s výjimkou výběru materiálu, který je volen náhodně a dochází k následné detekci barvy materiálu. Jednotlivé kroky však není možné paralelizovat, protože jsou na sobě závislé z pohledu *předávání* materiálu v různém stavu zpracování. Snaha o paralelizaci by způsobovala nejen nadbytečné zatížení robotické paže (např. již uchopený ochranný materiál), ale přinesla by jen minimální urychlení celého procesu. Dále by však hrozilo možné mechanické poškození při pohybu robotických paží.



Obr. 3.6: Návrh jednotlivých kroků balicího procesu.

Master-slave

Komunikace je vždy řízena nadřazenou master stanicí, která definuje operace, které budou následně vykonávány. Master stanice k tomu využívá operace čtení a zápisu hodnot slave stanic. Z důvodu využití průmyslového protokolu Modbus/TCP je využíváno různých typů adres na slave zařízení k řízení činnosti robotické paže. Master stanice zasílá koordinační souřadnice, které definují umístění robotické paže, operace (např. aktivování RGB senzoru a určení barvy předmětu nacházející se nad RGB senzorem). Pomocí těchto operací je také řízen dopravníkový pás (směr otáčení a vzdálenost, kterou má pás urazit), popřípadě umístění robotické paže ve vztahu ke kolejnici.

Slave-Robotická paže

Slave stanice využívá pro komunikaci s dobotem knihovnu pydobot. Z důvodu připojení dalších typů zařízení, jako jsou RGB senzor, dopravníkový pás, paže umístěná na kolejnici, bylo třeba knihovnu rozšířit a doplnit o další funkce. K samotnému ovládání tak jsou využívány funkce:

- vyčtení pozice z paže,
- definování nové pozice paže,
- aktivování/deaktivování sání,
- aktivování/deaktivování RGB senzoru,
- určení RGB složek předmětu umístěného nad RGB senzorem,
- pohyb dopravníkového pásu,
- pohyb paže 1 po kolejnici.

Samotné vykonávání příkazů je prováděno skrze instalovanou knihovnu pydobot umožňující řízení robotické paže a připojených zařízení k robotické paži. Samotná paže je připojena k Raspberry Pi (RPi) skrze USB kabel. Další obslužná zařízení jsou připojena k robotické paži, která zajišťuje vykonávání příkazů. Veškeré funkce vykonávané robotickou paží mohou být prováděny nezávazně na sobě, nebo s návazností. Je tak možné definovat více příkazů ve sledu a robotické paže je vykonává odděleně (po skončení jednoho úkonu zahájí následující), popřípadě jsou prováděny simultánně. Za ryze konfigurační lze označit funkci home, která slouží k vyresetování souřadnicového systému, to je provedeno mechanickým otočením robotické paže (otočení je provedeno samovolně skrze programový kód po vyvolání funkce home). Po tomto vyresetování dojde ke správnému nastavení robotické paže a po vyresetování je paže nastavena do výchozí pozice. Ke správnému chodu je doporučeno tuto funkci využívat vždy při prvotním spuštění hardwarových robotických paží.

Definované požadavky na vytvářený testbed

Tabulka 3.5 zobrazuje jednotlivé požadavky, které byly stanoveny před začátkem vývoje průmyslového testbedu. Aby bylo možné jednotlivé požadavky zajistit, je nutné brát v úvahu jejich nasazení již při návrhu a vývoji průmyslové smyčky. Mezi nejvíce kritické body je možné zařadit požadavky na zajištění dostatečné přesnosti a opakovatelnosti při zachování stabilního chodu. Je tak nutné brát v úvahu jednotlivé okolní vlivy, které mohou způsobit změnu běhu programu oproti ostatním, jako je například odlišný čas potřebný pro resetování souřadnicového systému robotické paže atp.

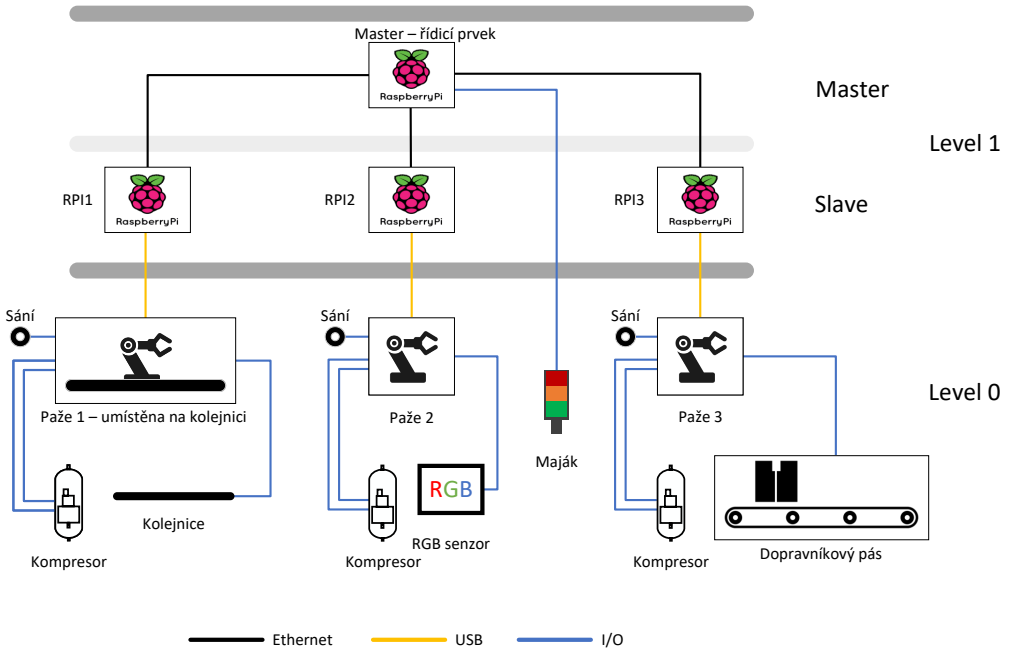
Tab. 3.5: Jednotlivé požadavky na vytvářený testbed.

Požadavek na testbed
Využití průmyslového protokolu
Implementace šifrované verze protokolu
Zajištění stabilního chodu
Zajištění přesnosti a opakovatelnosti
Vytvoření virtualizované verze
Možnost grafické vizualizace
Zajištění snadné správy, modifikovatelnosti

3.1.4 Technický popis

Logické zapojení

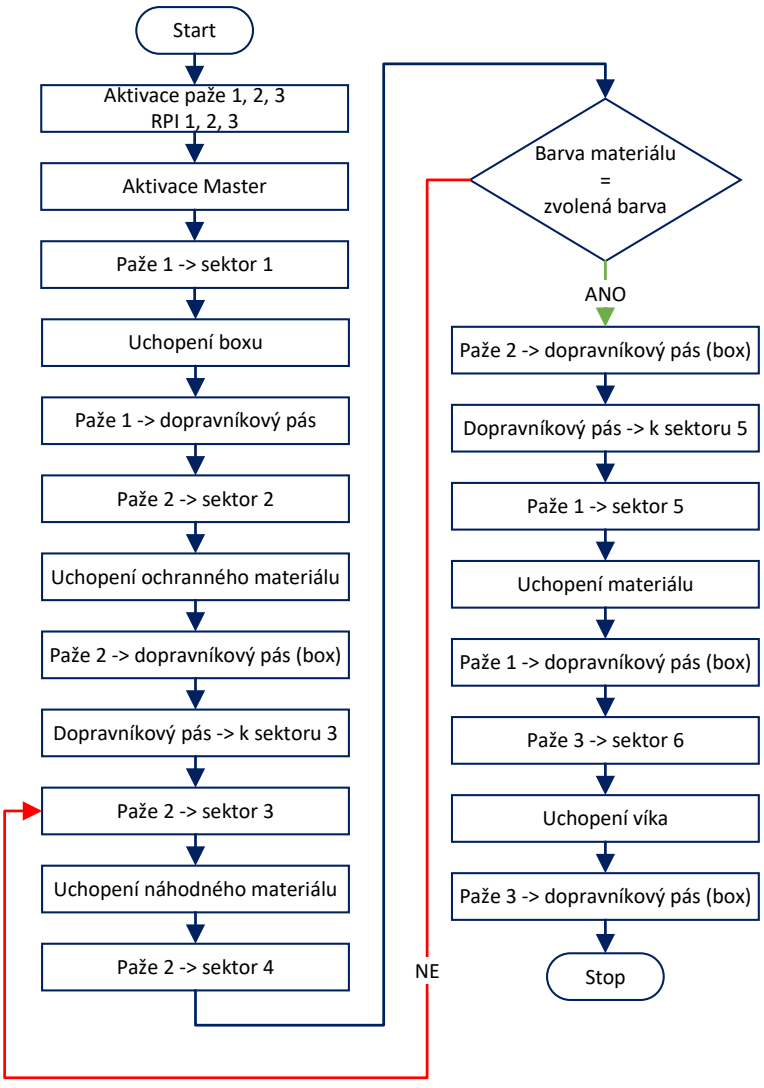
Balicí smyčka je sestavena na experimentálním pracovišti sestávající se ze tří zařízení Dobot Magician s využitím průmyslového protokolu Modbus/TCP. Pracoviště má za cíl demonstrovat balicí linku využívající dopravníkový pás k předávání *výrobku* na další určenou polohu, kde bude následně zpracován (obsloužen) další robotickou paží (rukou). Logické sestavení experimentálního pracoviště je vyobrazeno na obrázku 3.7. Síťové zapojení lze rozdělit dle Purdue modelu na dva levely. Level 0 obsahuje samotné aktivní prvky ovládané skrze zařízení na vyšší úrovni (jejich chování je plně řízeno skrze slave, resp. master stanici), kromě signalizačního majáku, který je přímo řízen master stanicí. Tento level obsahuje robotické paže a další připojené komponenty, jako RGB senzor a dopravníkový pás. Level 1 lze dále rozdělit na master a slave zařízení, kde komunikace je plně řízena master zařízením a slave zařízení pouze vykonává a monitoruje vykonávané činnosti. Z důvodu simulace PLC (Programmable Logic Controller) bylo využito zařízení RPi. Jednotlivé RPi zařízení komunikují pomocí průmyslového protokolu Modbus/TCP, ke kterému byla využita volně dostupná knihovna Pymodbus [4].



Obr. 3.7: Logická architektura balicí smyčky.

Logický cyklus

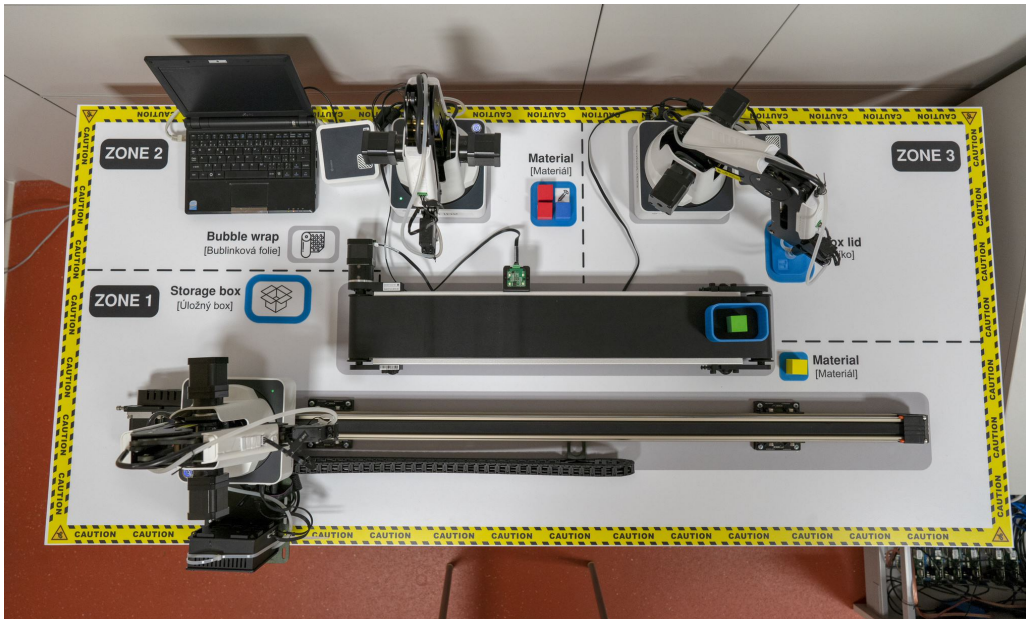
Na obrázku 3.8 je zobrazen jeden balicí cyklus na vytvořeném experimentálním pracovišti. Cyklus je zahájen aktivací slave stanice a samotných robotických paží. Následně master stanice navazuje spojení a řídí jednotlivé operace. Paže 1 je umístěna nad sektor *1*, je uchopen box a přemístěn na dopravníkový pás. Paže 2 uchopí ochranný materiál ze sektoru *2* do boxu. Následně je spuštěn dopravníkový pás a box přesunut blíže k sektoru *3*. Tento sektor je také obsluhován paží 2, ta náhodně uchopí jeden z materiálů uloženém na definovaných pozicích v sektoru *3*. Tento materiál přemístí nad sektor *4*, kde je identifikována barva materiálu. V případě shody s definovanou barvou je materiál přenesen do boxu. Pokud nedojde ke shodě, je materiál vrácen na původní pozici v sektoru *3* a následně je zvolen jiný materiál k otestování barvy. Box je následně přesunut pomocí dopravníkového pásu blíže k sektoru *5*, kde je za pomoci paže 1 přenesen materiál ze sektoru *5* do boxu. Cyklus je dokončen pomocí paže 3, která převezme materiál ze sektoru *6* a přemístí jej na box. Tento stav je finální fáze celého procesu a samotný box by mohl být dále zpracováván/přemísťován. V tomto kroku je však spuštěn inverzní proces a dojde k odebrání veškerého materiálu v boxu na původní pozice ve vymezených sektorech.



Obr. 3.8: Postup operací během jednoho balicího cyklu.

Fyzické experimentální pracoviště

Experimentální pracoviště je složeno ze tří robotických paží. Každá robotická paže má umístěné porty na předloktí a na své základně (jak je zmíněno výše). Každá robotická paže má připojen kompresor v portech *GP1* a *SW1*. Ovládání rotace *přísavky* umístěné na konci robotické paže je připojeno do portu *GP3-port1* na předloktí paže. Robotická paže *1* je dále připevněna ke kolejnici, k tomu jsou využity porty *Communication port* a *Stepper 2*. Robotická paže *2* má připojen RGB senzor do portu *GP2* a robotická paže *3* má připojen dopravníkový pás do portu *GP2*. Fotodokumentace na obrázku 3.9.

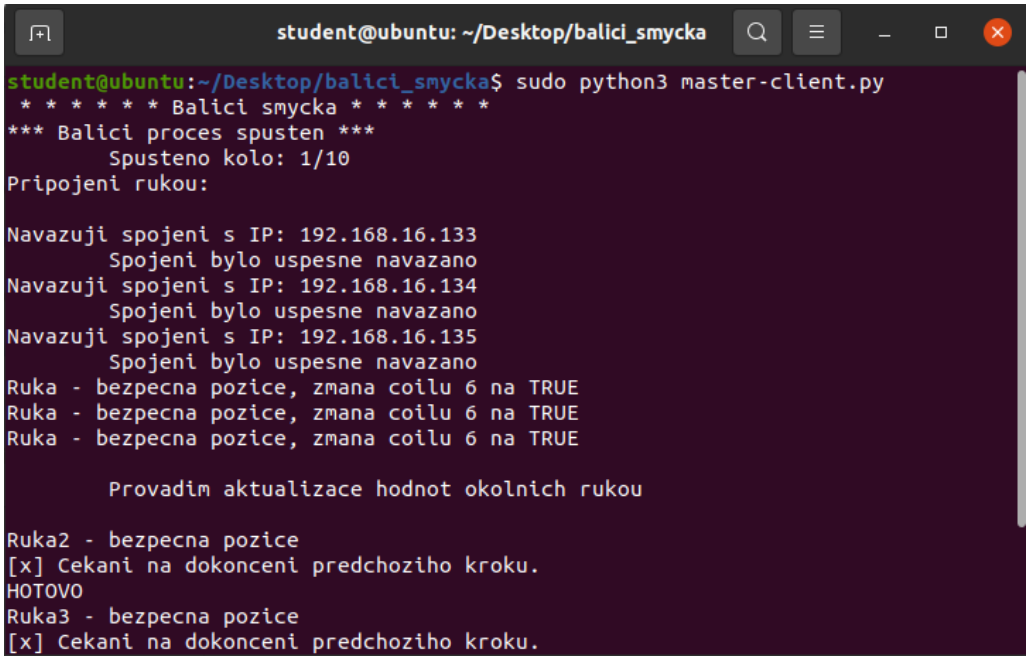


Obr. 3.9: Fotodokumentace experimentálního pracoviště.

Virtualizovaná verze

Na základě vytvořeného pracoviště byla dosavadní implementace na fyzických zařízeních rozšířena také o virtualizovanou verzi pracoviště. Z pohledu komunikace mezi jednotlivými prvky sítě nedošlo k žádné změně. Jsou zasílány/vykonávány totožně příkazy jako v případě fyzického pracoviště, jediná změna nastala v nezaslání příkazu k vykonání pohybové/logické akce na fyzické zařízení. Z pohledu implementace je to řešeno pomocí definování proměnné v konfiguračním souboru robotické paže, které je následně využito pro spuštění kódu. Pomocí virtuální verze robotické paže je možné dále rozšiřovat a libovolně nahrazovat, popřípadě doplňovat existující strukturu sítě o další prvky s cílem realizace komplikovanější infrastruktury, k vytvoření kritických stavů v sítí, úpravy komunikačních parametrů (např. latence) apod. Tato virtuální

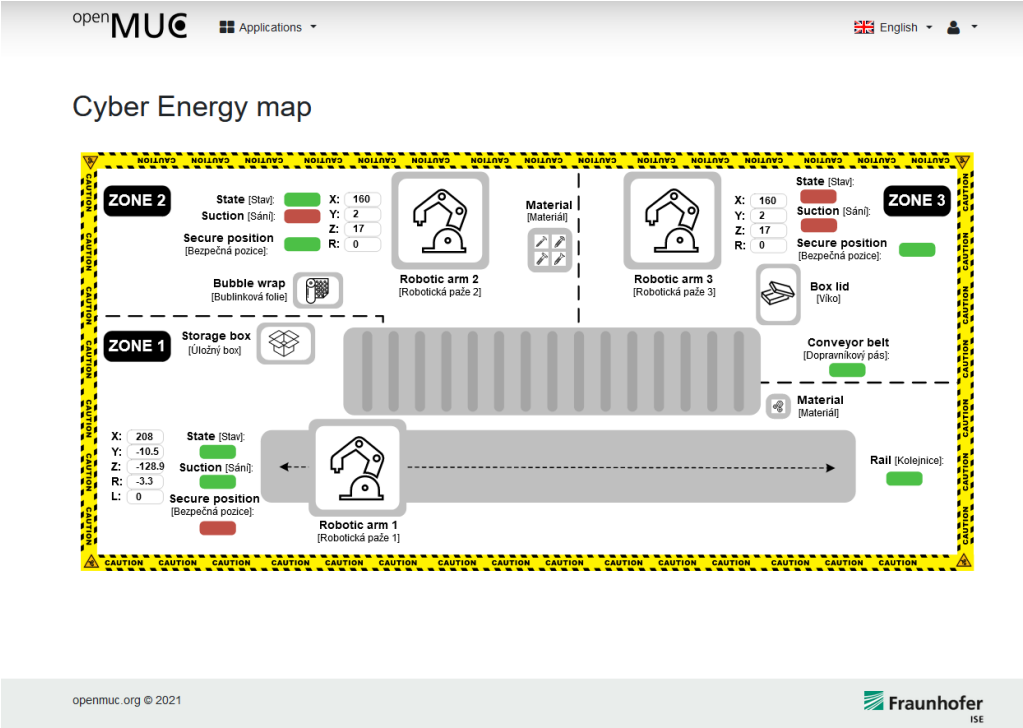
verze také umožňuje provádění dalšího výzkumu z pohledu srovnávání komunikačních vzorů, detekce virtualizovaných pracovišť, simulace kódu před jeho vykonáním apod. Na obrázku 3.10, je zobrazen programový výstup kódu po jeho spuštění. Nejprve dochází k navázání TCP spojení s jednotlivými zařízeními a následně je zahájen celý cyklus. Dokončení jednotlivých kroků je kontrolováno totožným způsobem jako v případě fyzických zařízení.



Obr. 3.10: Programový výstup virtuální verze robotických paží.

Vizualizace

Z důvodu využívání průmyslové komunikace je možné nejen zpracovávat komunikaci v reálném čase, ale také vyčítat jednotlivé paměťové bloky na slave stanici. Zde jsou uloženy jednotlivé parametry, které jsou vyhodnocovány a vykonány slave stanicí. Vyčítání a zobrazování je prováděno pomocí monitorovacího a řídicího softwaru OpenMuc [1]. Obrázek 3.11 zobrazuje vizualizaci skrze tento software. U každé robotické paže jsou zobrazeny její pozice (X, Y, Z, R, L), stav, sání a dosažená pozice. Graficky vizualizován je také stav dopravníkového pásu a kolejnice. Tato provedená vizualizace dále přibližuje vytvořený testbed reálnému nasazení. Je tak možné jej prezentovat, jako případný HMI, který umožňuje nejen zobrazování pozic jednotlivých zařízení, ale také umožňuje zaslání příkazů na jednotlivé stanice.



Obr. 3.11: Vizualizce robotických paží.

Obnovitelnost a přenositelnost

Aby bylo možné předcházet ztrátě dat, resp. zajistit obnovitelnost a přenositelnost testbedu, byl využit NAS server. V rámci tohoto serveru je spuštěn software na pro účely verzování a správy kódu. Tento software je tak možné využít nejen v případě ztráty kódu na jednotlivých zařízeních (RPi), ale také k jednoduchému rozšiřování na další stanice (robotické paže). Pomocí tohoto softwaru je možné provádět případné změny/vylepšení dosavadního kódu na jedné stanici a změny se následně mohou promítnout na jednotlivé stanice bez nutnosti provádění rozsáhlých změn v jednotlivých stanicích. Z důvodu zajištěného verzování je také možný návrat k předchozí konfiguraci.

3.1.5 Testování a verifikace

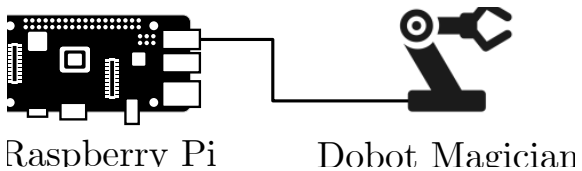
Tato kapitola se zaobírá popisem zabezpečení balicí smyčky, možnostmi generování dat pomocí robotických paží, určením obousměrného zpoždění v rámci vytvořeného polygonu a bylo také identifikované úzké hrdlo v rámci komunikace. Tyto jednotlivé části byly součásti prováděného výzkumu a vývoje. Cílem tohoto testování je určit vlastnosti vytvořeného průmyslového testbedu, které mohou být následně využity pro účely navazujícího testování a výzkumu.

Stanovení předpokladů na jednotlivé testy

V tabulce 3.6 jsou vyneseny předpokládané hodnoty, které by měly být v rámci prováděného testování ověřeny. Jednotlivé testovací scénáře jsou zaměřeny na kontrolu provedeného zabezpečení v rámci provedeného výzkumu testovacího prostředí. Další scénář je zaměřen na testování schopnosti testbedu generovat data (ty mohou být dále použita k dalšímu zpracování, například pomocí technik strojového učení). Mezi očekávanými hodnotami je přibližně 1 kB při provedení jednoho balicího cyklu. Mezi další scénáře je zařazeno testování doby RTT (Round Trip Time). Z pohledu testování je vždy využíváno zapojení, viz obrázek 3.12, dochází tedy k přímému připojení ovládacího stroje RPi a ovládaného prvku (Dobot Magician). Další scénář je zaměřen na dopad parametru čekání (parametr wait), který je definován v rámci knihovny dobot. Tento parametr poskytuje základní zabezpečení z pohledu prováděných operací na robotické paži. Zajišťuje časové pozdržení dalších požadovaných operací po robotické paži, než dojde k vykonání předchozího požadavku/příkazu. V základním předpokladu je urychlení procesu cca o 1 v rámci jednoho definovaného kola. S tímto parametrem je spojen také scénář, kdy je detekován dopad deaktivovaného parametru čekání na šířku pásma. Očekávaný rozdíl mezi šířkou pásma s aktivovaným parametrem čekání a bez něj je cca 0,4 Mb/s.

Tab. 3.6: Předpoklady výsledků testování.

Prvek testování, ověření	Požadavek/předpoklad
Bezpečnost	Implementace zabezpečeného protokolu
Schopnost generování dat, srovnání dopadů použitého šifrování	1 kB/jeden cyklus
Testování doby RTT	0,2 s
Parametr čekání – meření rozdílu	1 s (jedno kolo)
Určení úzkého hrdla, dopad na šířku pásma při deaktivovaném parametru čekání	0,4 Mb/s



Obr. 3.12: Blokové schéma zapojení při využití pouze jedné robotické paže.

Testovací scénář – bezpečnost

Tento scénář je zaměřen na implementaci zabezpečení v rámci průmyslového protokolu Modbus/TCP. Cílem je zajištění důvěrnosti přenášených dat, ale i jejich integrity. Dále je třeba zajistit bezpečnost testbedu z pohledu programového návrhu. Bezpečnost (zejména fyzickou) je možné zajistit skrze programový kód. Jednotlivé předpoklady tak jsou:

- Zajištění důvěrnosti a integrity v rámci přenosu.
- Vhodný programový návrh k zajištění fyzické bezpečnosti.

Z pohledu implementovaného protokolu Průmyslový protokol Modbus/TCP, stejně jako řada dalších průmyslových protokolů, neposkytuje v základní verzi žádné zabezpečení. Tyto průmyslové protokoly byly navrženy pro práci v oddělených a zabezpečených systémech. V případě propojení informačních (IT) a provozních (OT) technologií je tak nutné využívat dodatečné zabezpečení formou IDS (Intrusion Detection System), nebo IPS (Intrusion Prevention System), popřípadě využít zabezpečenou verzi průmyslového protokolu. V případě průmyslového protokolu Modbus/TCP je možné využít Modbus Security. Tato verze průmyslového protokolu provádí tunelování Modbus/TCP skrze TLS (Transport Layer Security). Pomocí této verze dochází k mírnému zpoždění z důvodu manipulace skrze TLS, je tím však dosaženo zabezpečení komunikace. Zejména je tím zajištěna vzájemná autentizace jednotlivých stanic prostřednictvím certifikátů, zajištění autorizace skrze TLS a šifrování datového provozu. Na základě předložených šifrovacích schémat klientem server vybírá šifrovací schéma, které bude následně využito pro šifrování komunikace pomocí symetrické kryptografie (např. TLS_AES_256_GCM_SHA384). Tato verze tak provádí ochranu proti velkému množství útoků, které není možné (bez využití dalších mechanismů) detekovat/předcházet v případě nezabezpečené varianty průmyslového protokolu. Vytvořené experimentální pracoviště má implementované obě dvě verze průmyslového protokolu. Mezi verzemi je možné se přepínat a generovat tak data pomocí protokolu Modbus/TCP i pomocí Modbus/TLS (Modbus/TCP Security).

Z pohledu programového návrhu Aby nedošlo k fyzickému poškození robotické paže, je využita definovaná adresa v každé robotické paži, která značí bezpečné uložení paže. Před každou sekcí pohybu dané robotické paže je provedena kontrola této adresy/hodnoty v ostatních robotických pažích. Pokud tato hodnota není rovna jejich bezpečnému uložení (nebylo dosaženo *bezpečné* pozice) není dovolen pohyb ostatních robotických paží. Kontrola tohoto bloku je možná z důvodu implementované aktualizace hodnot (vzájemné komunikaci robotických paží) jednotlivých

paměťových bloků okolních paží. Při využívání těchto hodnot okolních paží může dojít k libovolnému rozšíření podmínek pohybu/podporovaných akcí v závislosti na zvolené implementaci/fyzickém uložení jednotlivých robotických paží.

Dosažené výsledky V rámci tohoto scénáře byla zajištěna bezpečnost přenášených dat za pomoci využití zabezpečeného průmyslového protokolu. Dochází tak ke vhodné implementaci zabezpečení bez nežádoucího nadbytečného zvýšení vytížení linky z důvodu nevhodně navrženému postupu zabezpečení. Fyzická bezpečnost byla zajištěna pomocí vhodného programového návrhu, kde byly definovány bezpečné pozice v rámci průmyslového testbedu, aby nedocházelo k fyzickému poškození robotických paží z důvodu závady na zařízení, nevhodnému postupu atd.

Testování – generování dat

Tento scénář je zaměřen zejména na generování dat s využitím vytvořeného testbedu. Tyto data mohou dále sloužit pro navazující výzkum a vývoj. Data mohou sloužit jako vstup do dalších aplikací, může docházet k vytváření specifických případů atp. Jedná se tak o jeden z nejvíce významných výstupů celého testbedu. Mezi základní předpoklady pro generování dat lze zahrnout:

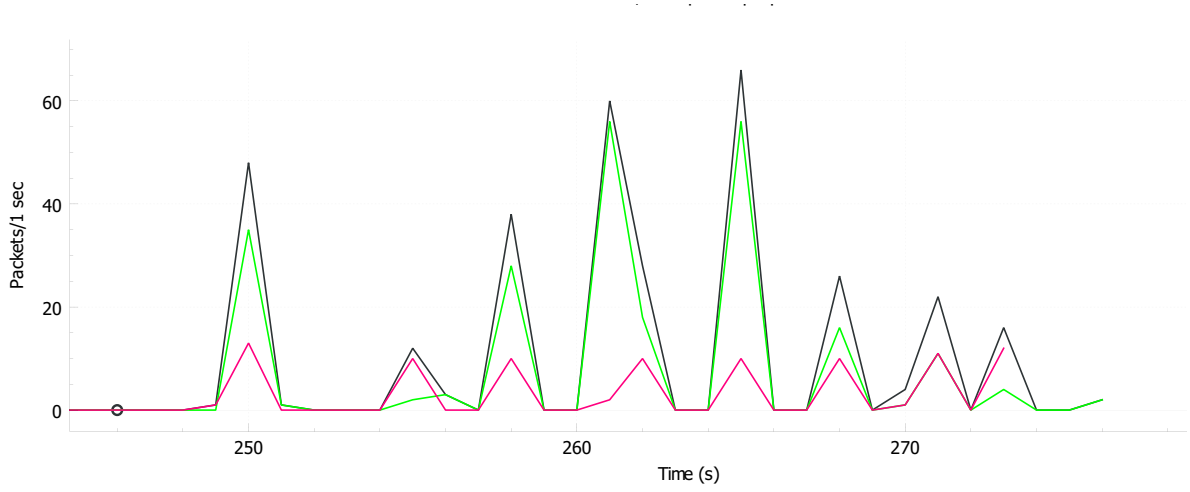
- Definovat schopnost generování dat z pohledu celkového objemu dat na jednu robotickou paži.
- Definovat využití jednotlivých registrů, resp. využití jednotlivých kódů funkce.
- Definovat nárůst způsobený vzájemnou komunikací robotických paží.
- Definovat rozdíl z pohledu objemu dat při použitém zabezpečení.

Před provedenou optimalizací Na základě provedeného testování založeného na datové komunikaci robotické paže *1*, vyplynuly výsledky, viz tabulka 3.7. Tabulka srovnává různé možnosti nastavení robotické paže z pohledu prodlevy mezi jednotlivými dotazy na slave stanici a obnovovací frekvencí slave stanice nutné k vyhodnocení dat slave stanice. Jedna robotická paže na základě provedení pouhých 10/11 operací provádí komunikaci s rychlostí přenosu cca 7,5 kb/s. V průměru je možné vygenerovat, se zmíněným nastavením, přibližně 967 kB/s (pouze s jednou robotickou paží). V případě využití všech tří paží přibližně 3 MB/s. Generování dat není z pohledu experimentálního pracoviště žádným způsobem omezeno z důvodu neustálého běhu ve smyčce a je tak možné generovat libovolně velké datové objemy. Na základě interního nastavení jednotlivých master a slave stanic je možné komunikaci generovat s vyšší, či nižší intenzitou. V rámci běhu jedné smyčky je v závislosti na použitém interním nastavení, v případě použití tří paží, vygenerováno přibližně 3 MB/s. Po dokončení jednoho cyklu je vygenerován přibližně 900 MB dat.

Tab. 3.7: Výsledky datového provozu – Paže 1.

Měření	Home	Master – prodleva čtení [s]	Slave – obnovovací frekvence [s]	Doba trvání [s]	Počet operací	Počet query	Počet response	FC3 – query	FC3 – resp	FC6 – query	FC6 – resp	Celkový počet zpráv	Velikost souboru [B]	Rychlost [paketů/s]	Rychlost [b/s]	Průměrná velikost paketu [B]
1	Ne	0,01	0,5	27,00	10	162	161	117	116	45	45	323	25390	12,0	7504	78
2	Ano	0,01	0,5	87,60	11	181	181	136	136	45	45	362	28412	4,2	2594	78
3	Ne	0,10	0,5	27,20	10	201	200	157	156	44	44	401	31434	14,9	9255	78
4	Ano	0,10	0,5	86,35	11	210	210	165	165	45	45	420	32907	4,8	2950	78
5	Ne	0,01	1,0	30,22	10	261	260	217	216	44	44	521	40734	17,4	10000	78
6	Ano	0,01	1,0	89,57	11	205	204	160	159	45	45	409	32055	4,6	2863	78
7	Ne	0,10	1,0	29,94	10	82	82	38	38	44	44	164	13066	5,6	3491	78
8	Ano	0,10	1,0	90,48	11	85	84	40	39	45	45	169	13455	1,9	1189	78

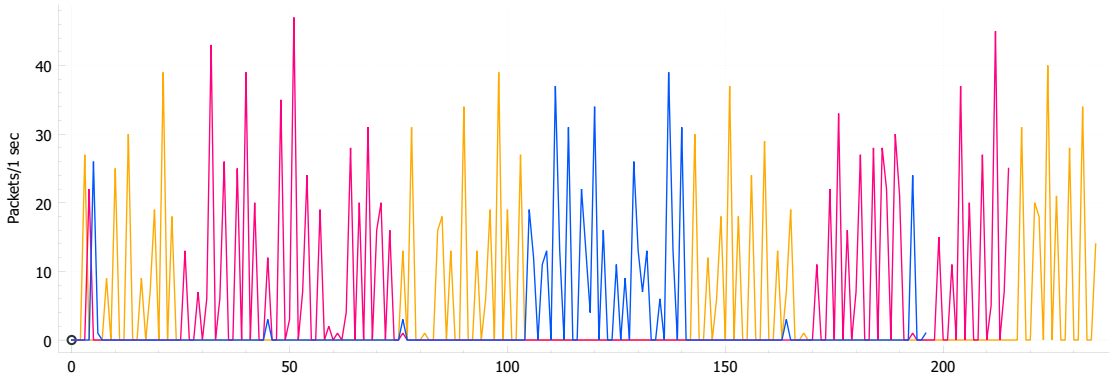
Obrázek 3.13 zobrazuje počet vygenerovaných paketů za sekundu v případě použití prvního měření. Černou barvou je vyznačena veškerá Modbus/TCP komunikace, zelenou barvou kód funkce 3 (function code) a červenou barvou kód funkce 6. Kód funkce 3 (Read Holding Registers) slouží k přenášení číselných hodnot a kód funkce 6 (Write Holding Register) slouží pro přenášení binárních hodnot (typicky pro indikaci prováděného stavu).



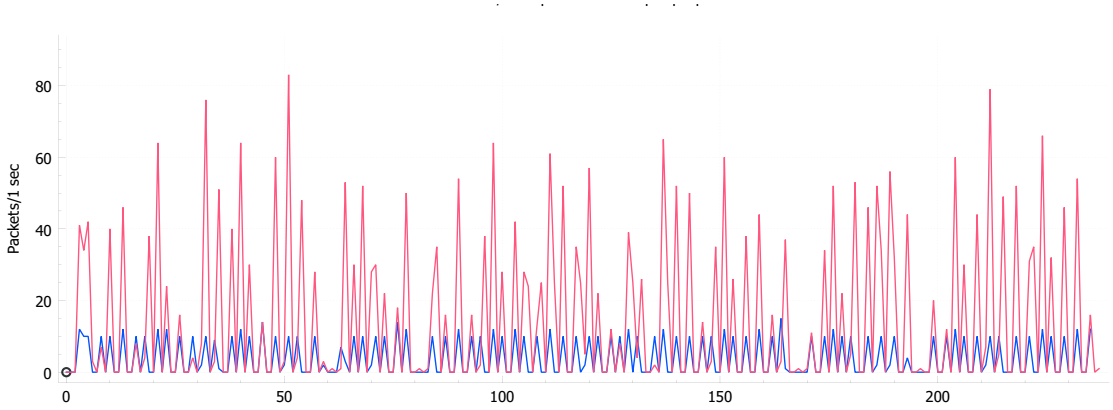
Obr. 3.13: Vytížení přenosového média z pohledu prvního testu.

Po provedené optimalizaci Po provedené optimalizaci datového toku došlo k redukci nadbytečného datového provozu, k využívání průmyslové datové komunikace

efektivnějším způsobem, a také k většímu přiblížení reálnému provozu. Komunikace byla zrychlena a zefektivněna z důvodu razantního snížení počtu neobsloužených dotazů na stav zařízení (nebyly zpracovány z důvodu přílišného zatížení slave stanice a takové zprávy nesou zanedbatelné množství informace). Po provedené optimalizaci je pracovní cyklus proveden v průměrném času 3,9 minuty. Během této doby dochází k vygenerování přibližně 350 kB dat pouze protokolu Modbus/TCP. S průměrným vytížením linky 11 kb/s, průměrnou velikostí paketu 78 B a průměrném celkovém počtu 4500 paketů (cca 19 paketů/s). Celkový počet provedených operací při použití tří robotických paží je přibližně 108 v závislosti na prvotním výběru kostky pro oskenování barvy a následném uložení do boxu. Velikost datového provozu může být dále navyšována zvýšením počtu provedených operací, a také s využitím virtualizovaných verzí robotických paží. Obrázek 3.14 zobrazuje sled operací prováděných rozdílnými pažemi, paže *1* je zobrazena oranžově, paže *2* růžově a paže *3* modře. Obrázek 3.15 zobrazuje rozložení použitých kódů funkce, kde kód funkce *3* je zobrazen růžově a kód funkce *6* je zobrazen modře.

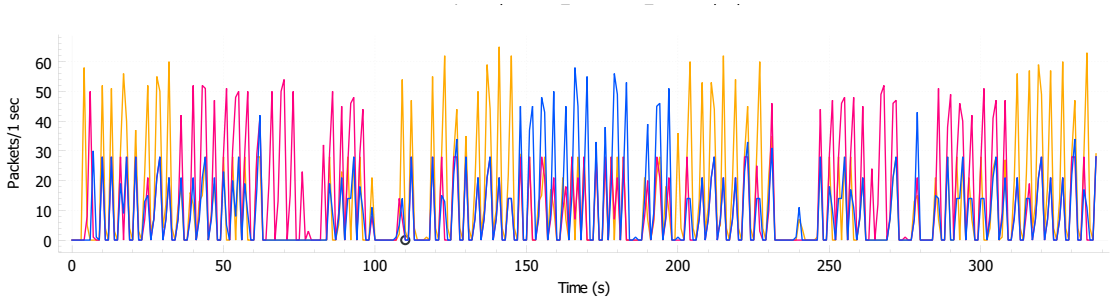


Obr. 3.14: Vytížení přenosového média z pohledu jednotlivých robotických paží.

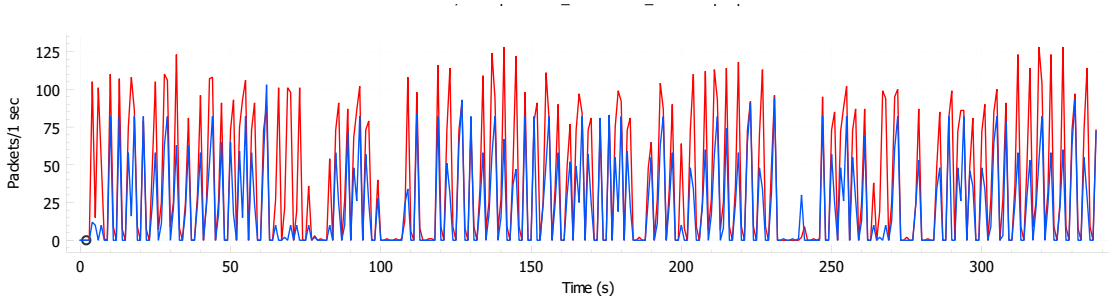


Obr. 3.15: Vytížení přenosového média z pohledu rozdílných kódů funkce.

Po implementaci vzájemné komunikace robotických paží Po provedené implementaci vzájemné komunikace mezi robotickými pažemi došlo ke zvýšení datového provozu. Toto rozšíření bylo implementováno do již optimalizovaného řešení. Po provedení vzájemné komunikace došlo k prodloužení trvání doby jednoho balicího cyklu na hodnotu přibližně 5,5 minuty. Během této doby bylo vygenerováno 23187 paketů, kdy během sekundy je vygenerováno přibližně 43 000 bitů. Datový provoz dosahuje celkové velikosti 1,8 MB. Stále bylo využito pouze fyzických robotických paží, bez použití virtualizovaných verzí. Počet provedených cyklů i jednotlivé operace lze dále upravovat. Během komunikace bylo vygenerováno 13223 paketů náležící kódu funkce *3* a 30301 paketů pro kód funkce *6*. Obrázek 3.16 zobrazuje sled operací prováděných rozdílnými pažemi, paže *1* je zobrazena oranžově, paže *2* růžově a paže *3* modře v případě využité vzájemné komunikace robotických paží. Obrázek 3.17 zobrazuje rozložení použitých kódů funkce, kde kód funkce *3* je zobrazen růžově a kód funkce *6* je zobrazen modře.



Obr. 3.16: Vytížení přenosového média z pohledu jednotlivých robotických paží se vzájemnou komunikací.



Obr. 3.17: Vytížení přenosového média z pohledu rozdílných kódů funkce se vzájemnou komunikací.

Srovnání fyzické a virtualizované verze Tabulka 3.8 srovnává fyzickou verzi experimentálního pracoviště s virtuální variantou. Z pohledu množství vygenerovaného

provozu generuje fyzická verze přibližně o 1 MB více (uvažováno jedno kolo/cyklus). To je způsobeno urychlením prováděné akce, nedochází tak k generování dotazů kontrolující dokončení operace. Tyto dotazy jsou kritické z pohledu kontroly chování robotických paží a nesou jen velmi nízkou informační hodnotu. Z tohoto důvodu dosahuje virtuální verze nižších hodnot vygenerovaných paketů oproti fyzické verzi (z pohledu request paketů o cca 5500 paketů). Při pohledu na počet vygenerovaných hodnot vzhledem k běhu programu dosahuje virtuální verze výrazně vyššího generovaného datového toku za 1 s (více než trojnásobek). V případě virtuální verze se tak jedná o efektivnější způsob generování dat.

Tab. 3.8: Srovnání fyzické a virtualizované verze.

Verze	Fyzická	Virtuální
Počet rob. paží	3	3
Doba běhu cyklu [s]	3985	1244
Velikost vygenerovaného provozu [kB]	19623	18777
Počet paketů celkem	252687	241778
Počet paketů za 1 s	63,41	194,36
Počet request paketů	126385	120889
Počet response paktů	126302	120889
Počet FC 3	149872	140442
Počet FC 6	79458	78036
Průměrný datový tok [b/s] (tři paže)	39000	120000
Master – prodleva mezi dotazy [ms]	10	10
Slave – obnovovací frekvence [s]	1	1

Srovnání šifrované a nešifrované verze Experimentální pracoviště podporuje využití šifrované i nešifrované verze protokolu Modbus. Z důvodu téměř totožného datového provozu je možné tyto data srovnávat a využívat k dalším účelům. V tomto případě byl dohodnut jako symetrický protokol AES, přesněji šifrovací schéma TLS_AES_256_GCM_SHA384. Tabulka 3.9 srovnává jednotlivé verze z pohledu vygenerovaných hodnot. Šifrovaná verze protokolu generuje přibližně o 37 % více dat, než nešifrovaná verze, jeden paket odpovídající šifrované verzi je také o cca 15,06 B větší než v případě využití nešifrované verze protokolu. Nešifrovaná verze protokolu tak v případě využití třech robotických paží přibližně 5445 B/s a šifrovaná verze protokolu 8930 B/s.

Tab. 3.9: Srovnání šifrované a nešifrované verze protokolu.

		Nešifrovaná verze	Šifrovaná verze
	Port	502	802
	Počet generovaných kol	10	10
	Aktivovaná aktualizace	Ano	Ano
	Délka trvání [s]	3957.102	5719.893
	Zaznamenáno paketů	277520	550968
	Zaznamenáno dat [B]	21548847	51078978
	Průměrný počet zaslaných B/s	5445	8930
Paže 1	Zaznamenáno paketů	95956	201078
	Zaznamenáno dat [B]	7450401	18638036
	Průměrný počet zaslaných B/s	1882	3258
Paže 2	Zaznamenáno paketů	102342	207038
	Zaznamenáno dat [B]	7945737	19189571
	Průměrný počet zaslaných B/s	2008	3355
Paže 3	Zaznamenáno paketů	79222	142852
	Zaznamenáno dat [B]	6152709	13251371
	Průměrný počet zaslaných B/s	1554	2316

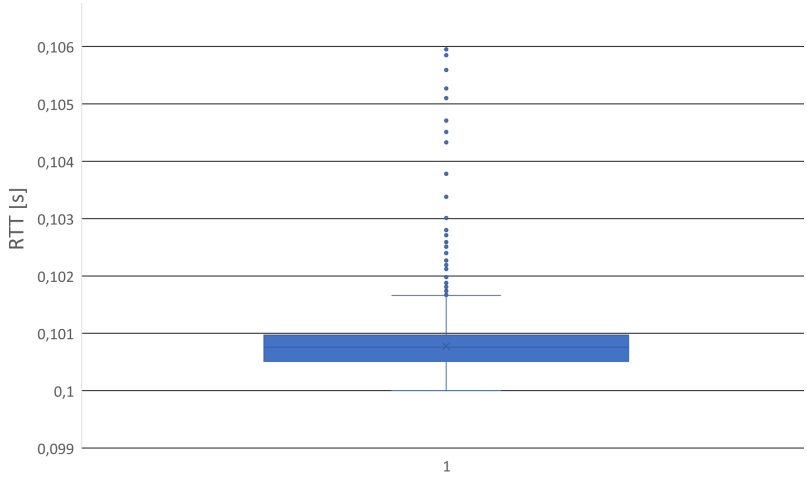
Dosažené výsledky V rámci toto testování došlo k identifikaci objemu v rámci kterého je možné provádět generování dat. V případě nešifrované verze vytvořený testbed umožňuje generovat datový obsah o objemu cca 2,2 MB a v případě šifrované verze (Modbus/TCP Security) je možné generovat objem cca 5,1 MB při využití tří robotických paží. Z provedeného testování se jeví provedený vývoj průmyslového testbedu jako efektivní se schopností generovat rozmanitá data. Mezi velký přínos lze zařadit využití virtualizované verze jež umožňuje provoz i bez nutnosti fyzického připojení hardwarového zařízení. Lze tak i testovat scénáře při kterých hrozí možnost fyzického poškození.

Testování – identifikace RTT v rámci balicí smyčky

Tento scénář je zaměřen na identifikaci RTT v rámci balicí smyčky. Za předpokládanou hodnotu lze označit 0,2 s.

- Identifikovat RTT v rámci balicí smyčky.

Testování V rámci testování balicí smyčky bylo testováno také obousměrné zpoždění. Obrázek 3.18 zobrazuje rozložení hodnot RTT delay získaného z experimentálního testování. Robotická paže Dobot je připojena skrze USB port do Raspberry Pi (jak již bylo zmíněno výše). Na základě zasílaných zpráv a získaných potvrzení na tyto zprávy bylo vypočítány hodnoty RTT. Výpočet je založen na zaslaných 18 881 dotazů, resp. odpovědí. Průměrná hodnota je rovna 0,1008 s. Minimální hodnota 0,1 s, maximální hodnota 0,1017 s, medián 0,1008 s. První kvartil (Q1) začíná na hodnotě 0,1005 a třetí kvartil (Q3) dosahuje hodnoty 0,1010 s. Maximální hodnotu přesahuje 97 hodnot, jejichž průměrná hodnota je rovna 0,1023 s.



Obr. 3.18: Zobrazení Round Trip Time delay, komunikace skrze USB rozhraní.

Výsledky Z provedeného testování vyplynulo, že RTT z pohledu mediánové hodnoty je přibližně 0,1 s na jedno kolo.

Testování – dopady aktivovaného/deaktivovaného parametru čekání

Tento scénář je zaměřen na identifikaci dopadů aktivovaného, popř. deaktivovaného parametru čekání (parametr wait) definovaného v rámci knihovny pydobot.

- Identifikovat dopady deaktivace parametru wait.
- Identifikovat časové dopady deaktivace parametru wait.

Testování V rámci testování robotické paže byl vytvořen experimentální scénář sestávající z 6 definovaných operací, které budou v cyklech robotickou paží vykonávány. Scénář je složen z operací:

1. posun paže o −10 cm na ose x (vlevo),
2. posun paže o 10 cm na ose z (nahoru),

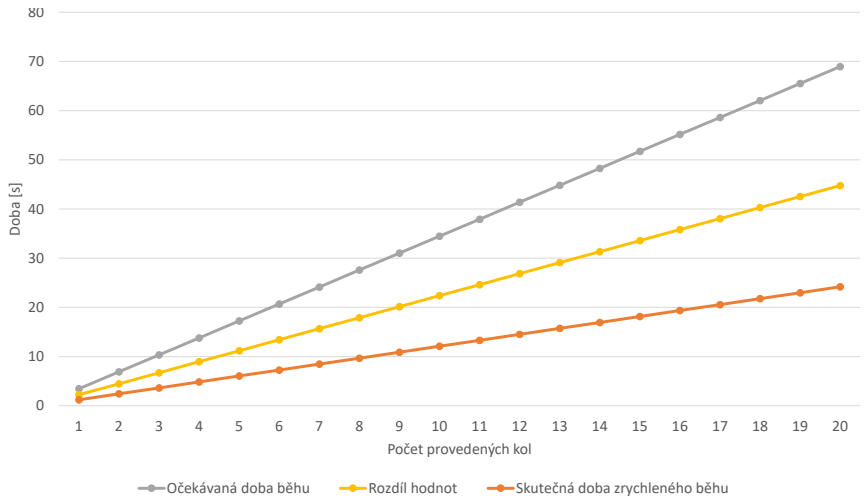
3. posun paže o 20 cm na ose x (vpravo),
4. aktivace sání,
5. posun paže o $-10cm$ na ose z (dolů),
6. deaktivace sání.

Tento experimentální scénář byl rozdělen do dvou fází. První fáze je prováděno v *bezpečném* režimu, s aktivovaným parametrem *wait*. Tento parametr zajišťuje vykonání dalšího příkazu až poté, co byl předchozí příkaz úspěšně dokončen. Druhý scénář má tento parametr deaktivovaný s cílem otestování chování robotické paže v případě zahlcení operacemi. Navržený experimentální scénář v první fázi (aktivovaný parametr *wait*) má průměrné hodnoty jednotlivých operací rovny:

1. posun paže o -10 cm na ose x (vlevo), 0,8552 s,
2. posun paže o 10 cm na ose z (nahoru), 0,7044 s,
3. posun paže o 20 cm na ose x (vpravo), 1,0066 s,
4. aktivace/deaktivace sání, 0,1765 s,
5. posun paže o -10 cm na ose z (dolů), 0,7049 s.

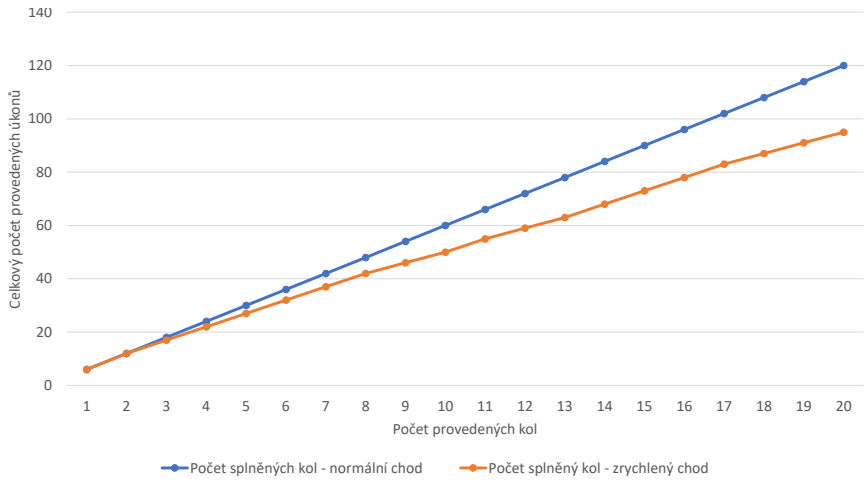
Celková doba běhu jednoho takového cyklu (kola) je v průměrné hodnotě rovna době 3,4476 s. V případě druhé fáze (deaktivovaný parametr *wait*) je průměrná hodnota jednoho kola rovna 1,4475 s a medián roven 1,2089 s (z celkového počtu 451 kol).

Obrázek 3.19 slouží ke srovnání těchto dvou fází v grafické podobě. Šedá křivka zobrazuje časové nároky na provedení jednotlivých kol, kde časové nároky jsou lineárního charakteru. Oranžová křivka zobrazuje časové nároky na jedno kolo. Žlutá křivka zobrazuje rozdíl těchto hodnot. V případě deaktivovaného parametru *wait* jsou časové nároky na kolo odlišné, ale tento časový údaj je nutné interpretovat tak, že jsou vygenerovány příkazy k řízení robotické paže (6 příkazů) a následně je kolo ukončeno (operace se ukládají do paměti robotické paže). Tento parametr má umožňovat provádění operací simultánně vzájemně nezávislých operací (např. posun po kolejnici a natáčení robotické paže do jiné pozice). Ve vytvořeném testování tento parametr způsobuje *hromadění* operací v paměti, protože nelze tyto operace provádět simultánně (v případě zahájení přesunu paže do jiné polohy je nutné její dokončení, aby bylo možné zahájit novou operaci).



Obr. 3.19: Srovnání normálního a zrychleného běhu -– časové hledisko.

Obrázek 3.20 zobrazuje klesající počet vykonaných operací s narůstajícím počtem provedených kol (oranžová křivka) s normálním během (modrá křivka). Pokud je parametr *wait* deaktivován, jsou jednotlivé operace zasílány do robotické paže, kde jsou uloženy do interní paměti ke zpracování bez ohledu na momentální stav (proces) robotické paže. Jak dochází k plnění paměti robotické paže dochází postupně k *vynechávání* operací, které mají být vykonány. Modrá křivka zobrazuje lineární průběh, kde se každé kolo stává z přesného počtu šesti kroků. V případě oranžové křivky, počet kroků v každém kroku s dobou běhu zmenšuje. Dochází tak k *anomáliím* v chodu paže. Definované *čtvercové* chování v experimentálním chování postupně přechází v trojúhelník a následně jen úsečky. Dochází také k celkovému zpomalení chodu. Přibližně při třetím kole dochází k anomáliím. Ukončení třetího kola v případě deaktivovaného parametru je provedeno po 3,6332 s od startu – je tak vykonáno 6 příkazů z celkových celkových 18, které odpovídají ukončenému třetímu kolu.



Obr. 3.20: Srovnání normálního a zrychleného běhu -– hledisko počtu vykonaných operací.

Výsledky V rámci tohoto scénáře byly identifikovány dopady způsobené deaktivací parametru wait. V případě deaktivace tohoto parametru je způsoben rozdíl cca 1,2 s na jedno definované koho skládající se ze čtyř pohybu, aktivace a deaktivace sání.

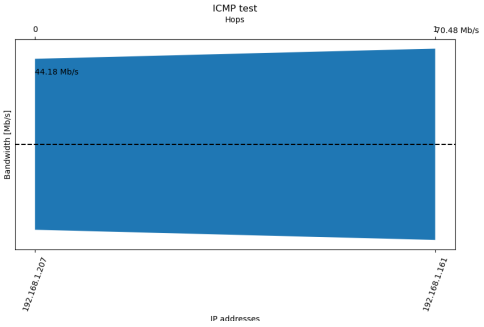
Testování – určení úzkého hrdla a dopadu aktivace paží na šířku pásma

V rámci tohoto scénáře je nutné identifikovat úzké hrdlo v rámci vytvořené komunikace. Z důvodu využití jedné master stanice a tří slave stanic je provedeno měření vždy z jednoho zdroje.

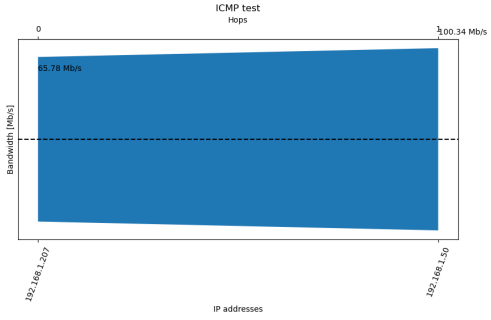
- Identifikovat úzké hrdlo v rámci navrženého systému.
- Identifikovat změnu šířky pásma při spuštěném procesu balicí smyčky.

Testování Na experimentálním pracovišti byla provedena identifikace úzkého hrdla mezi jednotlivými robotickými pažemi pomocí vytvořeného nástroje. Vytvořený nástroj detekuje šířku pásma z pohledu přenosových protokolů TCP, UDP a ICMP. Pracoviště je složeno z jedné master stanice (192.168.1.207) a tří robotických paží (192.168.1.161; 192.168.1.50; 192.168.1.66). U robotické paže *1* byla detekována šířka pásma 70,48 Mb/s (viz obrázek 3.21), u robotické paže *2* 100,34 Mb/s (viz obrázek 3.22) a u robotické paže *3* 120,82 Mb/s (viz obrázek 3.23). Tyto hodnoty byly založeny na přenosu jedné zprávy o objemu 74 B z master stanice do jednotlivých robotických paží. Pro evaluaci byl použit nástroj iperf. Výsledky provedeného testování jsou zobrazeny v tabulce 3.10. Testování bylo provedeno ve dvou scénářích s totožným nastavením pro aktivované a deaktivované robotické paže (paže prováděly činnost během testování). Testování vždy probíhalo 10 s s nastavenou šířkou

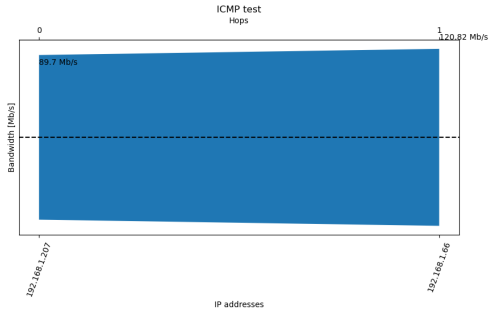
TCP okna na hodnotu 43,8 kB, resp. 128 kB na master stanici, resp. robotické paži. Z provedeného testování je zřejmé, že využíváním pásma dochází k zmenšení hodnoty pro ostatní síťový provoz o cca 0,2 Mb/s u jedné robotické paže. Na straně master stanice dochází ke snížení cca 0,1–0,5 Mb/s v závislosti na vytížení sítě. V případě, že by byly brány v potaz nejvyšší naměřené hodnoty, je možné paralelně obsluhovat až 200 robotických paží v případě 100 Mb linky, kde by byly všechny robotické paže aktivní v jeden časový okamžik. Z pohledu zahlcení běžným provozem tak může docházet až v případě využití vyššího počtu, jak 200 robotických paží (fyzických/virtuálních) po jedné lince.



Obr. 3.21: Šířka pásma – robotická paže 1.



Obr. 3.22: Šířka pásma – robotická paže 2.



Obr. 3.23: Šířka pásma – robotická paže 3.

Tab. 3.10: Dopad robotických paží na šířku pásma.

Výsledky Na základě provedeného testování bylo určeno úzké hrdlo téměř totožné v celém systému, a to zejména z důvodu využití totožných zařízení. Dále při aktivované komunikaci/aktivaci balicí smyčky došlo ke snížení dostupné šířky pásma o cca 0,2 Mb/s.

Výsledky provedeného testování vytvořeného testbedu

Jednotlivé dosažené výsledky z celkového testování jsou zobrazeny v rámci tabulky 3.11. Tabulka zobrazuje hodnoty, které byly předpokládány před začátkem testování. Z tabulky však vyplývá, že zejména v oblasti schopnosti generování dat byl základní požadavek překonán. Za největší přínos tohoto testbedu je možné označit generování rozmanitých dat založených na skutečném procesu získaném na reálných hardwarových zařízeních. Díky vzniku virtualizovaných verzí je možné nejen provádět rozšíření zapojení, ale i provádět navazující výzkum čistě na virtualizovaných verzích. To by však nebylo možné dosáhnout bez základního využití hardwarových prvků. Mezi další navazující výzkum a vývoj lze například označit zpracování vygenerovaných dat pomocí technik strojového učení a neuronových sítí pro účely detekce anomálií, vytváření komunikačních vzorů, vytváření otisků zařízení atp.

Tab. 3.11: Srovnání výsledků testování.

Prvek testování, ověření	Požadavek/předpoklad	Otestováno
Bezpečnost – kontrola	Implementace zabezpečeného protokolu	Implementace zabezpečeného protokolu + bezpečnostních funkcí v rámci kódu
Schopnost generování dat, srovnání dopadů použitého šifrování	1 MB/jeden cyklus	2,2 MB (nešifrovaná verze, 3 paže) 5,1 MB (šifrovaná verze, 3 paže)
Testování doby RTT	0,2 s	0,1008 s (medián)
Parametr čekání – meření rozdílu	1 s (jedno kolo)	1,2 s (jedno kolo, medián)
Určení úzkého hrdla, dopad na šířku pásma při deaktivovaném parametru čekání	0,4 Mb/s	0,2 Mb/s

Kontrola požadavků na vytvořený testbed

Tabulka 3.12 provádí kontrolu dodržení jednotlivých požadavků na průmyslovou balicí smyčku před zahájením vývoje. Veškeré parametry byly dodrženy. V rámci sloupce poznámky je uvedeno jakým způsobem byl daný požadavek splněn.

Tab. 3.12: Jednotlivé implementované požadavky na vytvořený testbed.

Požadavek na testbed	**Zajištěno**	**Poznámka**
Využití průmyslového protokolu	Ano	Modbus/TCP
Implementace šifrované verze protokolu	Ano	Modbus/TCP Security
Zajištění stabilního chodu	Ano	Využití parametru čekání z pohledu (dobot, skript)
Zajištění přesnosti a opakovatelnosti	Ano	Volbou robotické paže
Vytvoření virtualizované verze	Ano	Bez nutnosti fyzického připojení paží
Možnost grafické vizualizace	Ano	OpenMuc
Zajištění snadné správy, modifikovatelnosti	Ano	Pomocí konfiguračním skriptů

3.2 Příklad II – Čisticka

3.2.1 Shrnutí

Testovací prostředí čističky odpadních vod (ČOV) spadá do oblasti průmyslových řídicích systémů, které jsou řízeny pomocí automatizačních zařízení jako jsou například programovatelné logické automaty (Programmable Logic Controller – PLC). ČOV přesněji spadají do oblasti vodohospodářství. V dnešní době se v této oblasti, jako ve většině ostatních průmyslových oblastech, přechází ke vzdálené komunikaci, která má mnoho výhod jako například vzdálený odečet dat, monitorování stavu vodohospodářských zařízení, a nebo také možnost ovládání jednotlivých řešení. Se vzdáleným přístupem většinou pomocí systémů dispečerského řízení a sběru dat (Supervisory Control And Data Acquisition – SCADA) souvisí ale také bezpečnostní hrozby a to především vzhledem k současné konvergenci IT a OT. V současnosti je důležité věnovat velkou pozornost právě kybernetickým hrozbám a tedy především kybernetickým útokům, které mohou mít dopad například na bezpečnost osob, finance, ekologii a také know-how firmy. Proto je důležité včasně předcházet takovýmto útokům. K tomu jsou v současnosti nejvhodnější systémy detekce anomálií, které mohou pracovat na základě algoritmů strojového učení. Tímto směrem je vhodné také vést hlavní směry výzkumu, tedy především v oblastech metod strojového učení pro detekci anomálií. Vzhledem k problematické přístupnosti k datům, pro trénování detekčních metod je nutné taková data generovat. K takovému generování dat jsou právě ideální testovací prostředí (testbedy) simulující takovýto provoz. Díky tomu je možné demonstrovat na takových testbedech také nestandardní provoz – tedy například provoz ve kterém se objeví kybernetické útoky. Vzhledem k velkým dopadům nemohou být takové útoky generovány v reálných provozech a tedy právě k takovému sběru dat jsou testbedy ideální. Díky tomu je možné vytvářet velké datasety s velkým množstvím různých útoků.

Tento testbed se snaží přiblížit reálné ČOV, založené na technologii Sequencing Batch Reactor (SBR) to je označení pro nádrž, ve které dochází zároveň k biologickému čištění pomocí provzdušňování, a k separaci vzniklého kalu od vyčištěné vody. Testovací prostředí bylo navrženo dle reálné čistírny odpadních vod, která pracuje v malé obci. Hlavním cílem stavby tohoto testbedu byla možnost sběru procesních dat a síťových dat a především také, jak bylo výše zmíněno generování nestandardního provozu pro vytváření datasetů. Zvolili jsme čistírnu odpadních vod, protože stále více čistíren (pro domácnosti, firmy, obce nebo větší oblasti) je řízeno a monitorováno na dálku pomocí systémů SCADA. Tento testbed tedy slouží k výzkumu kybernetické bezpečnosti a možných vektorů útoků v této oblasti. Primárním cílem je pro nás sběr procesních dat, výzkum kybernetické bezpečnosti a výzkum v oblasti

metod strojového učení pro detekci anomálií v průmyslových sítích.

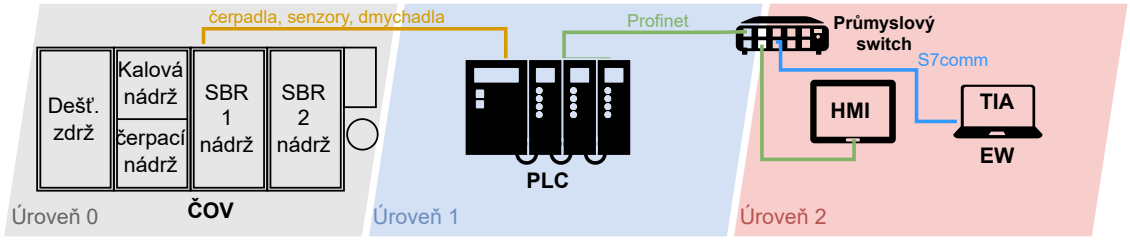
Účely testbedu:

- Dlouhodobý sběr procesních i komunikačních dat z testbedu.
- Sběr dat z nestandardního provozu.
- Průzkum možných vektorů útoků.
- Demonstrace anomálií na fyzických zařízeních.
- Testování detekčních nástrojů.
- Výzkum v oblasti detekčních metod anomálií v průmyslových sítích.
- Zlepšení kybernetické bezpečnosti v ostrém provozu.
- Optimalizace reálných řešení.

3.2.2 Použité komponenty

Fyzický testbed

Na obrázku 3.24 je uvedeno obecné schéma testbedu ČOV, které je rozděleno dle průmyslové pyramidy (purdue modelu ANSI/ISA-99). Na úrovni 0 je fyzická ČOV. Fungují zde jednotlivá čerpadla nezbytná pro provoz čistírny, následují snímače hladiny, snímače kalu, snímače čerpání, snímače dešťové vody a dmychadla. Na úrovni 1 pracuje PLC od společnosti Siemens, které řídí celý proces čištění odpadních vod. Na vrstvě 2 pracuje inženýrské pracoviště, z něhož lze upravovat program ČOV nebo dohlížet na její provoz. Na této vrstvě je také HMI, které umožňuje obsluze ovládat jednotlivé procesy ČOV a je určeno také pro vizualizaci současného stavu. Jednotlivé komponenty propojuje průmyslový přepínač.



Obr. 3.24: Obecný diagram ČOV testbedu.

Jednotlivé hardwarové a softwarové komponenty jsou popsané v tabulce 3.13. Dále budou komponenty popsány podrobněji.

Komponenty

Nádrže Celý model čistírny odpadních vod se skládá z pěti nádrží, které jsou zkonstruované z plexiskla o tloušťce 5 mm. Dále je zde recipient který není na viditelném

Tab. 3.13: Jednotlivé komponenty ČOV.

Název hardwaru	**Popis**
Siemens s7-1200	PLC
Siemens KTP700	HMI
Siemens CSM1277	Průmyslový switch
Dmychadla	Dmychadla
Bezkontaktní snímače hladiny kapalin	Senzory
Snímače vertikální hladiny vody	Senzory
Plovákové snímače	Senzory
Vodní čerpadla	Senzory
Inženýrská stanice (PC)	Řízení provozu

Software	
TIA portal v15.1	Programování
	Dohled

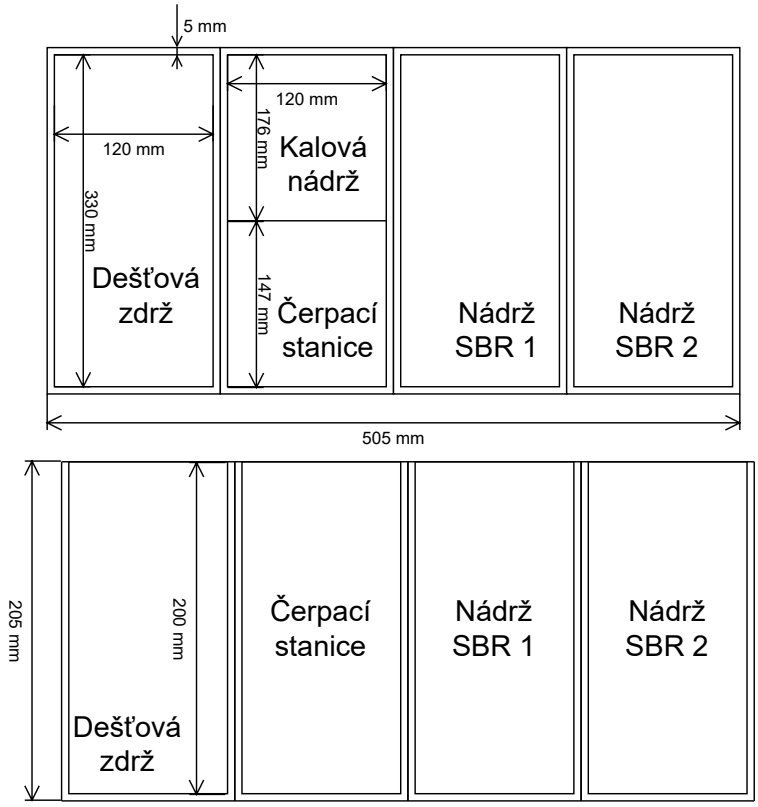
Protokoly
Profinet
S7comm

místě, ale je skrytý v konstrukci stolu. Tento recipient simuluje plastová nádoba. Rozměry použitých nádrží jsou uvedeny v tabulce 3.14.

Tab. 3.14: Rozměry nádrží.

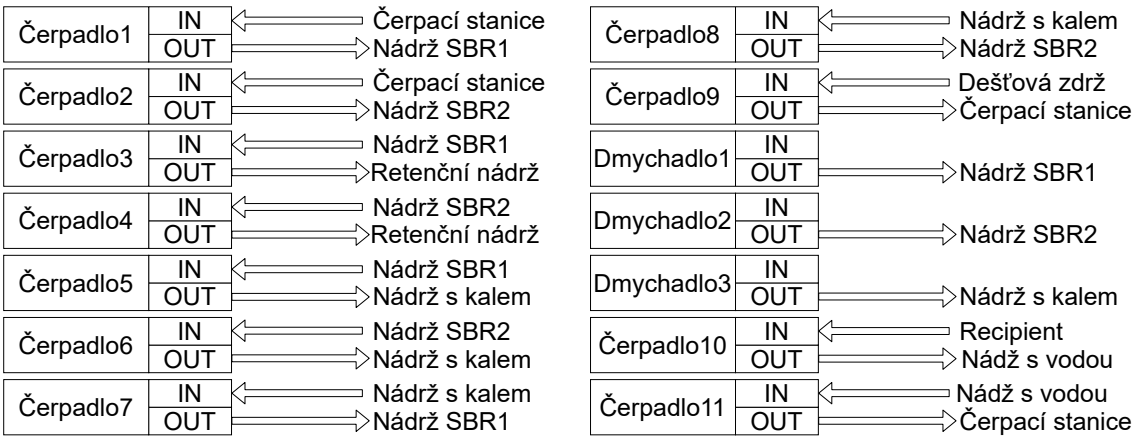
Nádrž	**Rozměry nádrže (H×Š×V) [cm]**	**Objem [l]**
Retenční dešťová nádrž	33×12×19,75	7,8
Čerpací nádrž	14,75×12×19,75	3,5
Kalová nádrž	17,75×12×19,75	4,2
SBR #1	33×12×19,75	7,8
SBR #2	33×12×19,75	7,8

Na obrázku 3.25, lze vidět půdorys testovacího prostředí ČOV a také podélný řez. Testbed byl vytvořen v poměru 1:12 k reálné ČOV.

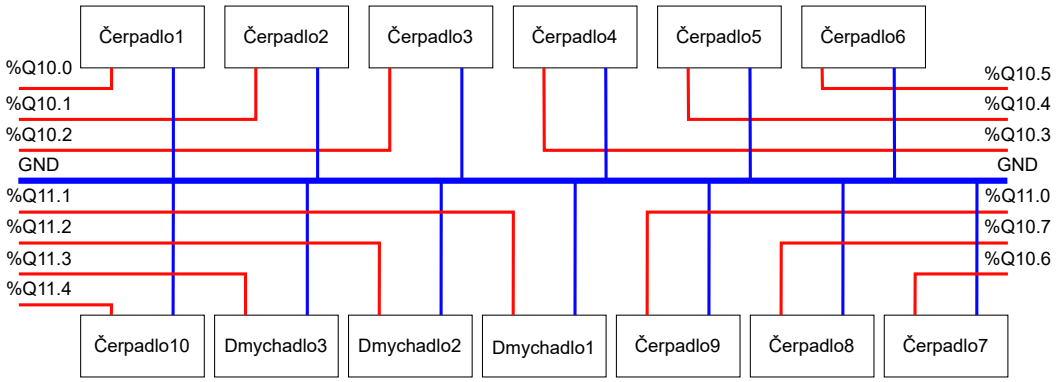


Obr. 3.25: Půdorys testovacího prostředí ČOV.

Čerpadla a dmychadla Pro testbed byla zvolena vodní čerpadla DC 6-12V R385, jsou to neponorná čerpadla, které jsou schopny pracovat i bez vody, tedy dokáží pumpovat i vzduch, díky tomu je lze použít i jako dmychadla. Při použití není nutno dodržovat polaritu (směr čerpání se otočením polarity nezmění). Je nutné pouze použít čistou vodu aby se čerpadla a také hadičky nezanesly. Parametry čerpadel jsou následující: napájecí napětí těchto čerpadel je 12 V, proud 300–700 mA, průtok až 2 l/min, maximální dosah sání je až 1,2 m. V reálných čistírnách bývá ve skutečnosti menší množství čerpadel, která se v čerpání střídají (např. pouze 2) a směr kapaliny se řídí pomocí elektromagnetických ventilů. Tyto ventily nebylo možno využít u našeho testbedu z toho důvodu, že ventily jsou vyráběny s průměry výstupů a vstupů větších než je možno použít u modelu. Proto musel být použit větší počet čerpadel a dmychadel než v reálných situacích. Aby bylo možné čerpat vodu z jednotlivých nádrží nebo použít dmychadla, byly použity silikonové hadičky s vnitřním průměrem 4 mm a vnějším průměrem 6 mm. Jednotlivé propojení hadiček a nádrží lze vidět na obrázku 3.26. Dále pak na obrázku 3.27 lze vidět zapojení čerpadel a dmychadel, společná zem (GND) je označena modře, napětí je označeno červeně a označuje připojení na jednotlivé výstupy PLC.



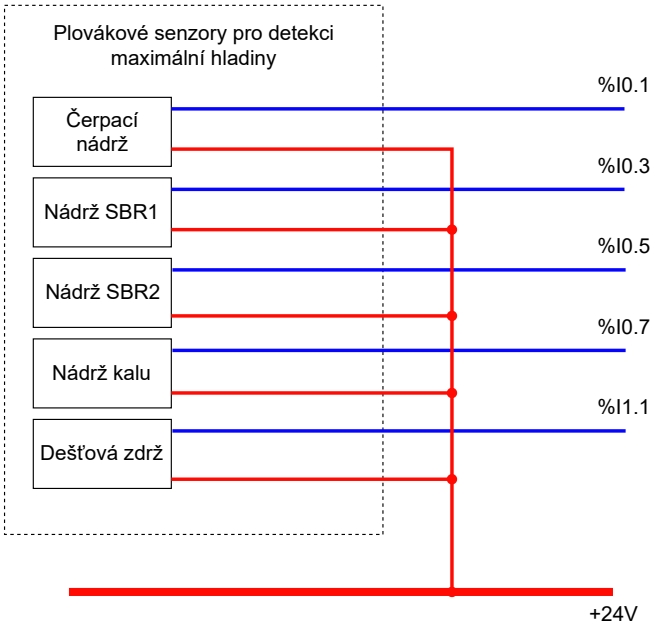
Obr. 3.26: Propojení čerpadel a nádrží.



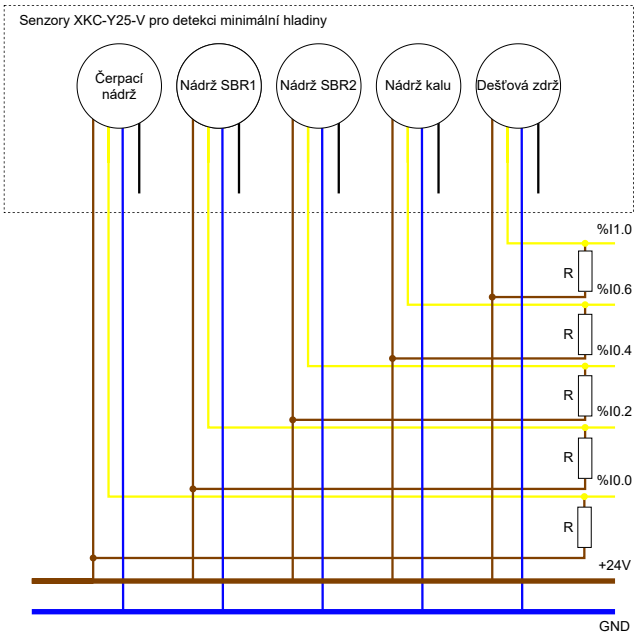
Obr. 3.27: Zapojení čerpadel a dmychadel.

Senzory Pro snímání hladiny v testbedu jsou použity dva typy senzorů, a to plovákové senzory a bezkontaktní senzory. Plovákové senzory mají tyto parametry: typ snímače je NO/NC; maximální spínané napětí je 230 VAC, 230VDC; maximální spínaný výkon je 50 W; maximální spínaný proud je 500 mA; maximální trvalý proud je 1 A. Tyto senzory jsou v testbedu použity ke snímání maximální hladiny v nádržích. Na obrázku 3.28 lze vidět zapojení plovákových senzorů, lze také vidět ke kterému vstupnímu portu PLC jsou jednotlivé senzory připojeny. Senzory jsou napájeny pomocí 24 V. Bezkontaktní senzory mají následující parametry: typ snímače je NO/NC; vstupní napětí je 5–24 V; vstupní proud je 5 mA; výstupní proud je 1–100 mA; doba odezvy je 500 ms. Tyto senzory slouží k detekci minimální hladiny v nádržích. Kvůli rozdílné logice PLC, která pracuje s režimem PNP a senzorů, které pracují s režimem NPN, bylo upraveno zapojení tak, aby senzory pracovali

také v logice PNP a to tak, že byl do zapojení přidán pull-up rezistor (2,7 kΩ). Na obrázku 3.29 lze vidět zapojení těchto senzorů.



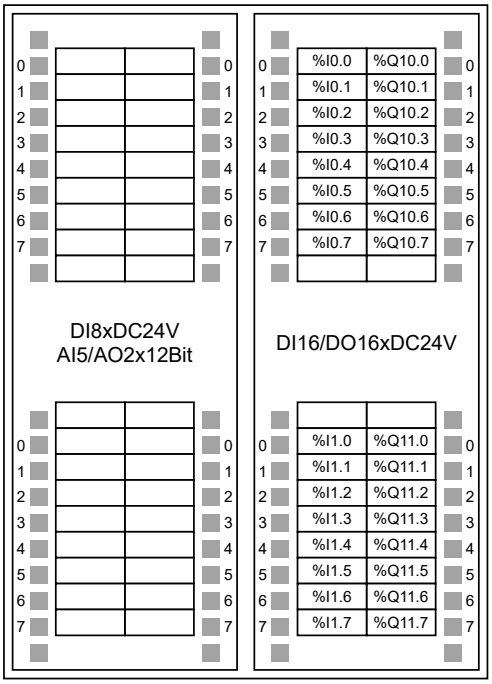
Obr. 3.28: Plovákové senzory – zapojení.



Obr. 3.29: Bezkontaktní senzory – zapojení.

Zdroj Jako zdroj pro napájení všech částí byl zvolen stabilizovaný zdroj napětí SITOP PSU100C od firmy Siemens (6EP1332-5BA10). Zdroj je připojen ke standardnímu napětí o velikosti 230 V. Výstup zdroje je napětí 24 V. Tímto zdrojem jsou tedy napájeny všechny součásti tohoto testbedu.

PLC Pro testbed bylo využito PLC Siemens S7-300 CPU314C-2 PN/DP. Na obrázku 3.30 je zobrazena adresace využitých portů PLC pro tento testbed.



Obr. 3.30: PLC – vstupní a výstupní porty.

HMI Pro ovládání jako rozhraní člověk-stroj a možnosti vizualizace bylo použito HMI od firmy Siemens KTP-700 Basic. Pro toto HMI bylo nakonfigurováno několik obrazovek, kde jsou jednotlivé procesní prvky ČOV. Díky tomu lze na HMI tyto prvky sledovat a ovládat (senzory, čerpadla apod.). Pro tento testbed bylo vytvořeno šest obrazovek. Tyto obrazovky budou ukázány v dalších sekcích.

Souhrn komponent

Jednotlivé komponenty ze kterých je složena čistírna odpadních vod jsou uvedeny v tabulce 3.15. V této tabulce jsou komponenty označeny i dle adresace PLC.

Tab. 3.15: Komponenty čistírny odpadních vod.

Označení	Název zařízení	Popis
PLC	Siemens S7-300	PLC sloužící pro řízení procesu
HMI	Siemens KTP700	HMI sloužící jako rozhraní člověk stroj
Čerpadlo #1 (%Q10.0)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z čerpací nádrže do nádrže SBR #1
Čerpadlo #2 (%Q10.1)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z čerpací nádrže do nádrže SBR #2
Čerpadlo #3 (%Q10.2)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z SBR #1 nádrže do retenční nádrže
Čerpadlo #4 (%Q10.2)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z SBR #2 nádrže do retenční nádrže
Čerpadlo #5 (%Q10.4)	Vodní čerpadlo DC 6-12V R385	Přečerpává kal z SBR nádrže #1 do kalové nádrže
Čerpadlo #6 (%Q10.4)	Vodní čerpadlo DC 6-12V R385	Přečerpává kal z SBR nádrže #2 do kalové nádrže
Čerpadlo #7 (%Q10.6)	Vodní čerpadlo DC 6-12V R385	Přečerpávání kalu z kalové nádrže do nádrže SBR #1
Čerpadlo #8 (%Q10.7)	Vodní čerpadlo DC 6-12V R385	Přečerpávání kalu z kalové nádrže do nádrže SBR #2
Čerpadlo #9 (%Q11.0)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z retenční nádrže do čerpací nádrže
Čerpadlo #10 (%Q11.4)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z retenční nádrže do nádrže s vodou
Čerpadlo #11 (%Q11.5)	Vodní čerpadlo DC 6-12V R385	Přečerpává vodu z nádrže s vodou do čerpací stanice
Plovákový senzor hladiny (%I0.1)	Plovákový senzor vodní hladiny	Detekce pro zastavení přítoku vody z retenční nádrže
Plovákový senzor hladiny (%I0.3)	Plovákový senzor vodní hladiny	Spouští aeraci a sedimentaci v nádrži SBR #1
Plovákový senzor hladiny (%I0.5)	Plovákový senzor vodní hladiny	Spouští aeraci a sedimentaci v nádrži SBR #2
Plovákový senzor hladiny (%I0.7)	Plovákový senzor vodní hladiny	Kontrola maximální hladiny
Plovákový senzor hladiny (%I1.1)	Plovákový senzor vodní hladiny	Kontrola maximální hladiny
Optický senzor hladiny (%I0.0)	Bezkontaktní čidlo XKC-Y25-V	Při detekci vody spouští čerpadlo #1
Optický senzor hladiny (%I0.2)	Bezkontaktní čidlo XKC-Y25-V	Kontrola minimální hladiny
Optický senzor hladiny (%I0.4)	Bezkontaktní čidlo h XKC-Y25-V	Kontrola minimální hladiny
Optický senzor hladiny (%I0.6)	Bezkontaktní čidlo XKC-Y25-V	Kontrola miniální hladiny
Optický senzor hladiny (%I1.0)	Bezkontaktní čidlo XKC-Y25-V	Při detekci vody spouští čerpadlo #9
Dmychadlo #1 (%Q11.1)	Vodní čerpadlo DC 6-12V R385	Provzdušňování (aerace) nádrže SBR #1
Dmychadlo #2 (%Q11.2)	Vodní čerpadlo DC 6-12V R385	Provzdušňování (aerace) nádrže SBR #2
Dmychadlo #3 (%Q11.3)	Vodní čerpadlo DC 6-12V R385	Provzdušňování (aerace) kalové nádrže

3.2.3 Vstupní kritéria, předpoklady, vývoj a návrh

Požadavky a předpoklady

Hlavním požadavkem je možnost sběru dat průmyslové síťové komunikace. Vzhledem k tomu je cíleno na fyzické testovací prostředí (testbed), který by takové generování dat mohl simulovat. Díky takovému generování dat je možné vytvářet detekční modely anomálií. Proto je důležité mít možnost testovat i nestandardní (anomální) provoz, což by v reálném prostředí nebylo možné. Z tohoto důvodu je třeba vytváření testovacích prostředí. Hlavní požadavky byly tedy:

- Vytvoření fyzického prostředí simulujícího ČOV.
- Komunikace musí obsahovat průmyslový komunikační protokol.
- Umístění fyzických komponent HMI a PLC.
- Možnost grafické vizualizace.
- V testovacím prostředí je možné testovat nestandardní provoz (různé vektory kybernetických útoků).
- Pomocí testovacího prostředí je možný dlouhodobý sběr dat standardní a nestandardní komunikace.
- Vytvořit virtualizované řešení.
- Na základě vygenerovaných dat je možné vytvářet datasety pro detekční nástroje anomálií.

Návrh testovacího prostředí ČOV

Popis obecné funkcionality testovacího prostředí ČOV Nejdříve je na začátku je pomocí tlačítka nebo na HMI pomocí přepínače spuštěn přítok jako simulace přítoku vody. Dojde tak k přečerpávání z vodního přítoku (nádrž ve stole) do čerpací nádrže, popřípadě může být ručně plněna dešťová zdrž pro simulaci deště. Pokud je čistička v provozu a v čerpací stanici se nachází voda, je tato voda přečerpávána do nádrže SBR1, přičemž při plnění SBR nádrží jsou zároveň tyto nádrže provzdušňovány. Pokud je nádrž SBR1 naplněná, dochází k aeraci nebo sedimentaci anebo k odčerpávání kalu a vyčištěné vody, je plněna nádrž SBR2. Po naplnění SBR nádrže dojde k cyklu aerace a sedimentace, přičemž počet cyklů a jednotlivé doby těchto procesů lze nastavit pomocí HMI. Po ukončení cyklu aerace a sedimentace dochází k odčerpávání kalu ze dna SBR nádrže do nádrže s kalem, ve které dochází při plnění k provzdušňování kalu. S odčerpáváním kalu zároveň dochází k odčerpání vyčištěné vody do dešťové zdrže (kvůli zachování cyklu). Nádrže s mikrosítem a nádrže představující recipient není využito z důvodu možnosti chodu čističky v cyklech, jelikož je nepraktické simulovat bez kapacitní recipient a téměř neustálý přívod vody na ČOV. Po odčerpání kalu a vyčištěné vody je SBR nádrž připravena k dalšímu naplnění z čerpací stanice. Pokud se v dešťové zdrži nachází voda, je přečerpávána do čerpací stanice. Při přeplnění čerpací stanice je voda vracena zpět do dešťové zdrže pomocí přepadu. Přepad je situován i mezi nádrží s kalem a čerpací stanicí, kde dochází k odtoku kalu při naplnění nádrže s kalem. Poslední přepad je situován z dešťové zdrže do vodní nádrže, která je uschována ve stole, tento přepad slouží k prevenci proti přeplnění čističky. Obsluha čističky krom spuštění, pozastavení chodu čističky, kontroly stavu ČOV a nastavování určitých parametrů může čističku pomocí HMI vypustit, kdy dojde k zastavení chodu čističky a voda z nádrže s kalem a čerpací stanice je čerpána do nádrže SBR1. Z SBR nádrží je voda čerpána do dešťové zdrže a odtud dochází k přečerpávání do vodní nádrže pod stolem. K odčerpávání z jednotlivých nádrží dochází ještě nastavenou dobu po poklesu hladiny vody na takovou hladinu, kdy optické senzory nedokáží tuto hladinu detekovat, aby došlo k odčerpání velké většiny vody z nádrží. Tato operace projde ve čtyřech cyklech a poté je čistička vypuštěna celkově a cykly začínají znovu.

Další části procesu, která je samostatně spustitelná, je vypouštění ČOV. To znamená, že při tomto procesu se vypouští z jednotlivých nádrží obsah do recipientu, který je uschován v rámci stolu s čističkou. Proces je samostatný a nezávislý na samotném cyklu.

Komunikace Komunikace probíhá pomocí průmyslového protokol S7comm. Jedná se o proprietární protokol firmy Siemens. Tento protokol byl zvolen vzhledem k vel-

kému zastoupení na evropském trhu. Dále také vzhledem k tomu, že samotný protokol není široce prozkoumán a je třeba mu věnovat velkou pozornost z pohledu kybernetické bezpečnosti. S tímto protokolem souvisí také nejrozšířenější knihovna pro tento protokol s názvem SNAP7, pomocí této knihovny lze komunikovat se zařízeními, které podporují komunikaci pomocí protokolu S7comm. Tato knihovna je velice vhodnou k tvorbě simulačních virtualizovaných testovacích prostředí.

Komponenty V rámci testovacího prostředí je cíleno především na fyzická zařízení od firmy Siemens, a to zejména na HMI a PLC nebo na fyzické senzory, jako jsou čerpadla, plovákové senzory a optické senzory. Dále je cíleno také na možnost komunikace se SCADA systémy. V tomto případě je možné komunikovat se SCADA řešením, které bylo vytvořeno na základě platformy OpenMUC. S tímto řešením je možno komunikovat pomocí protokolu S7comm.

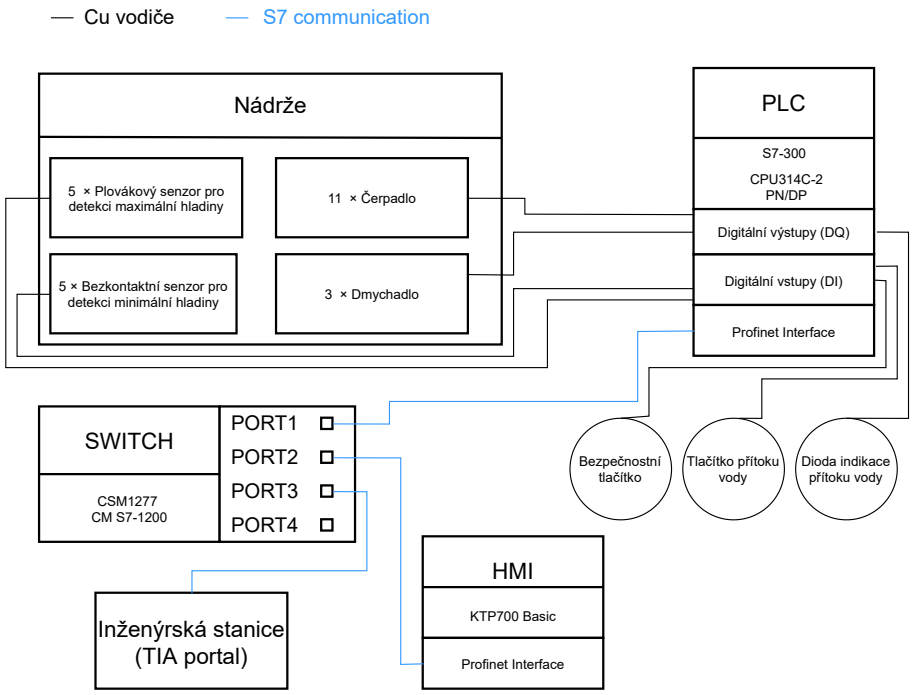
3.2.4 Technický popis

Na obrázku 3.31 lze vidět vzhled testovacího prostředí.

Obr. 3.31: Testovací prostředí ČOV.

Blokové schéma

Na obrázku 3.32 lze vidět blokové schéma komunikace testbedu ČOV. Hlavním řídicím prvkem celé komunikace je PLC S7-300 (6ES7314-6EH04-0AB0) od firmy Siemens, které řídí provoz celého testbedu. Jako digitální vstupy jsou zde 5x plovákové senzory pro detekci maximální hladiny, 5x bezkontaktní senzory pro detekci minimální hladiny, Bezpečnostní tlačítko a tlačítko přítoku vody. Jako digitální výstupy jsou zde 11x čerpadla, 3x dmychadla a dioda indikace přítoku vody. Dále PLC komunikuje s HMI a inženýrskou stanicí, na které běží TIA portal pomocí rozhraní profinet a přímo pomocí proprietárního protokolu firmy Siemenes S7comm. Pro komunikaci v rámci SCADA systému OpenMUC je použit taktéž protokol S7comm.



Obr. 3.32: Blokové schéma testbedu ČOV.

Konstrukce

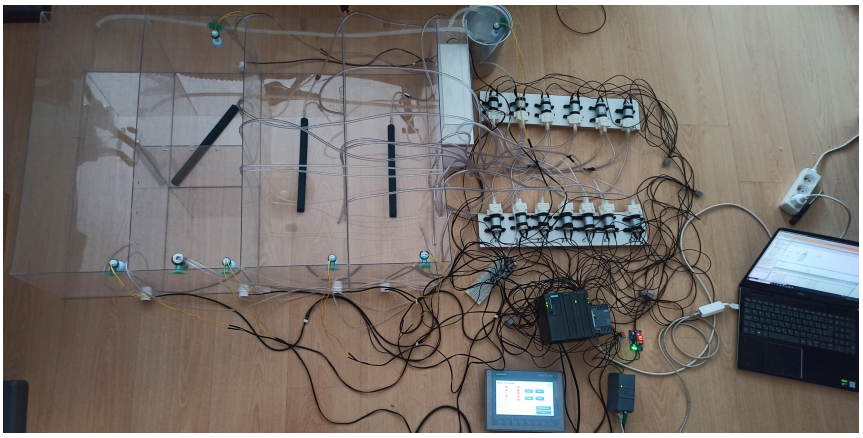
Při vytváření testbedu byly zkonstruovány dvě verze nádrží pro tento testbed. V textu budou následně popsány nedostatky první verze a důvody proč byla vytvořena verze druhá.

Konstrukční nedostatky první verze Pro první verzi byly použity pro nádrž plexiskla (plexisklové desky vyrobené z polystyrolu), které mají tloušťku 2 mm. Jednotlivé části bylo potřeba slepit dohromady speciálním lepidlem (je nutné dbát na

pevnost a vodotěsnost). Nejdříve bylo testováno několik komerčně dostupných lepidel (vteřinová lepidla, tavná pistole, chemoprén), požadované vlastnosti na pevnost a vodotěsnost splňovalo pouze lepení pomocí tavné pistole, ale vzhledem k nepřijatelnému estetickému řešení nemohla být zvolena ani tato možnost. Vzhledem k tomu bylo použito jednosložkové polymerové transparentní lepidlo, které je přímo určeno k lepení plexiskel (bylo použito lepidlo Acrifix 116). Lepení muselo probíhat postupně (lepidlo potřebuje k úplnému vytvrzení alespoň 24 hodin). V tabulce 3.16 lze vidět rozměry (kde H – hloubka, Š – šířka, V – výška) a objem nádrží této první verze. Po zkonstruování bylo bohužel zjištěno, že rozměry nádrží jsou příliš velké a tím pádem tloušťka plexiskla je nedostatečná a nádrže nebyly schopny udržet takový tlak vody. Proto musela být zvolena menší varianta se silnějšími plexiskly. Zkonstruovaná první verze čističky lze vidět na obrázku 3.33.

Tab. 3.16: Rozměry nádrží první verze.

Nádrž	**Rozměry nádrže (H×Š×V) [cm]**	**Objem [l]**
Retenční dešťová nádrž	66×24,5×30	48,5
Čerpací nádrž	30×24,5×30	22,0
Kalová nádrž	36×24,5×30	26,5
SBR #1	66×24,5×30	48,5
SBR #2	66×24,5×30	48,5



Obr. 3.33: První verze nádrží ČOV.

Optimalizace konstrukčního návrhu

Následně byl vytvořen nový model čistírny odpadních vod který se skládá z pěti nádrží, které jsou zkonstruované z plexiskla o tloušťce 5 mm. Dále je zde nádrž s vodou simulující přítok, která není na viditelném místě, ale je skrytá v konstrukci stolu.

Jedná se o 30 litrovou plastovou nádobu. Jednotlivé desky byly slepeny opět pomocí lepidla Acrifix 116 a díky tomu byly vytvořeny nádrže pro čistírnu. Pro ujištění aby byly spoje nepropustné byly vnitřní hrany pokryty transparentním silikonem (Pattex Perfect Sanitary). Rozměry použitých nádrží jsou uvedeny v tabulce 3.17.

Tab. 3.17: Rozměry nádrží finální verze.

Nádrž	**Rozměry nádrže (H×Š×V) [cm]**	**Objem [l]**
Retenční dešťová nádrž	33×12×19,75	7,8
Čerpací nádrž	14,75×12×19,75	3,5
Kalová nádrž	17,75×12×19,75	4,2
SBR #1	33×12×19,75	7,8
SBR #2	33×12×19,75	7,8

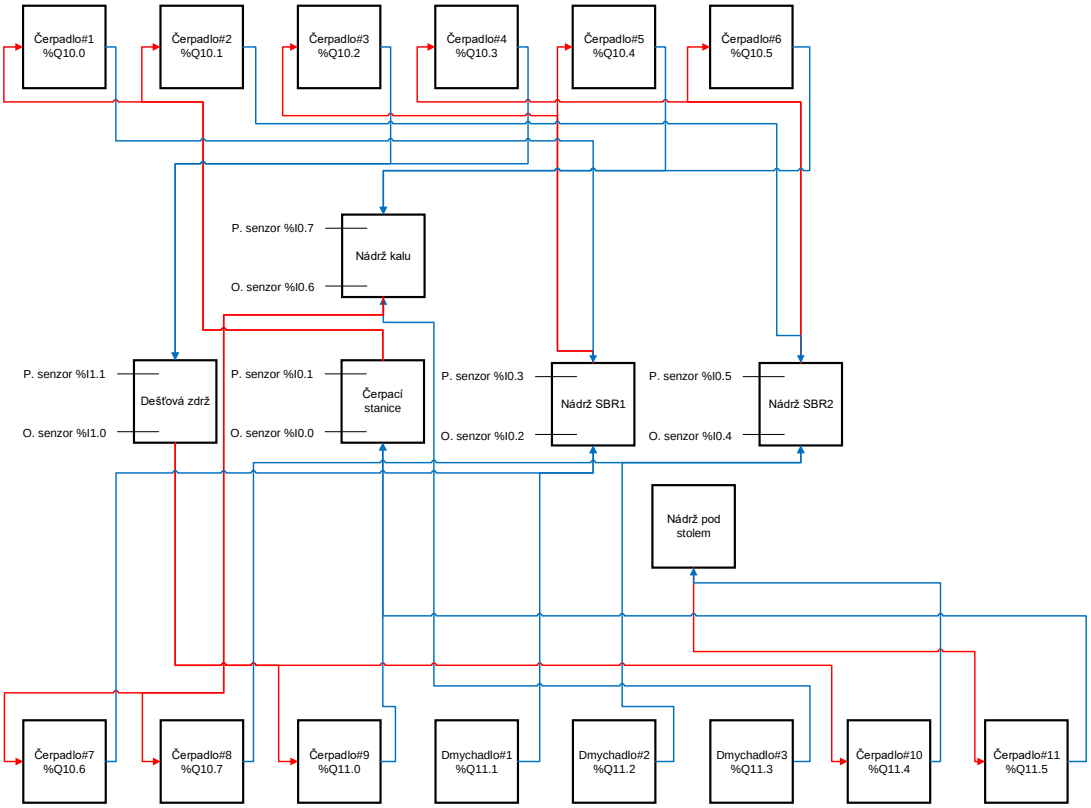
Optimalizace procesu Z procesního hlediska se v tesbedu nachází pět plovákových senzorů pro detekci maximální hladiny, jedenáct čerpadel , tři dmychadla a pět bezkontaktních senzorů. Poté byly pro lepší funkcionalitu přidány ještě dvě tlačítka, čerpadlo a indikační dioda. Jedno tlačítko pro zapnutí přítoku vody a indikační dioda indikující tento přítok a také čerpadlo které umožňuje simulování čerpání přítokové vody. Nakonec bylo přidáno také bezpečnostní tlačítko, které vypne celý testbed.

Popis logického procesu ČOV

Následně je na obrázku 3.34 znázorněno procesní schéma ČOV. Poté následuje detailní popis celého procesu. Po spuštění PLC a programu pomocí HMI je nutné zapnout automatické napouštění pomocí Čerpadla#11 (%Q11.5), které čerpá vodu z nádrže, schované v pracovním stole a simulující přítok vody do ČOV, do čerpací nádrže. Napouštění ČOV se spouští pomocí tlačítka (%I1.2) nebo pomocí přepínače na HMI na stránce *Stav čerpadel*. Voda je čerpána do čerpací stanice, pokud je tlačítko nebo přepínač sepnut a dokud není sepnut plovákový senzor(%I0.1) pro detekci maximální hladiny vody v čerpací stanici. Po sepnutí plovákového senzoru je čerpání vody do čerpací stanice zastaveno, a opětovně spuštěno po odčerpání vody z čerpací stanice, ale pouze pokud je napouštění ČOV spuštěno, tím je zaručen dostatečný přítok vody na ČOV. Pokud je v čerpací stanici dostatek vody pro detekci minimální hladiny vody optickým senzorem (%I0.0), je spuštěno napouštění nádrže SBR1 pomocí Čerpadla#1 (%Q10.0). S napouštěním nádrže SBR1 je spuštěno i provzdušňování této nádrže pomocí dmychadla#1 (%Q11.1). Po napuštění nádrže SBR1, a tím i sepnutí plovákového senzoru (%I0.2) pro detekci maximální hladiny vody v nádrži SBR1, je spuštěn určitý počet cyklů aerace a sedimentace, kdy se v časových intervalech střídá provzdušňování nádrže dmychadlem#1 a sedimentace, kdy se v nádrži nic neděje a dochází k usazování kalu na dno nádrže.

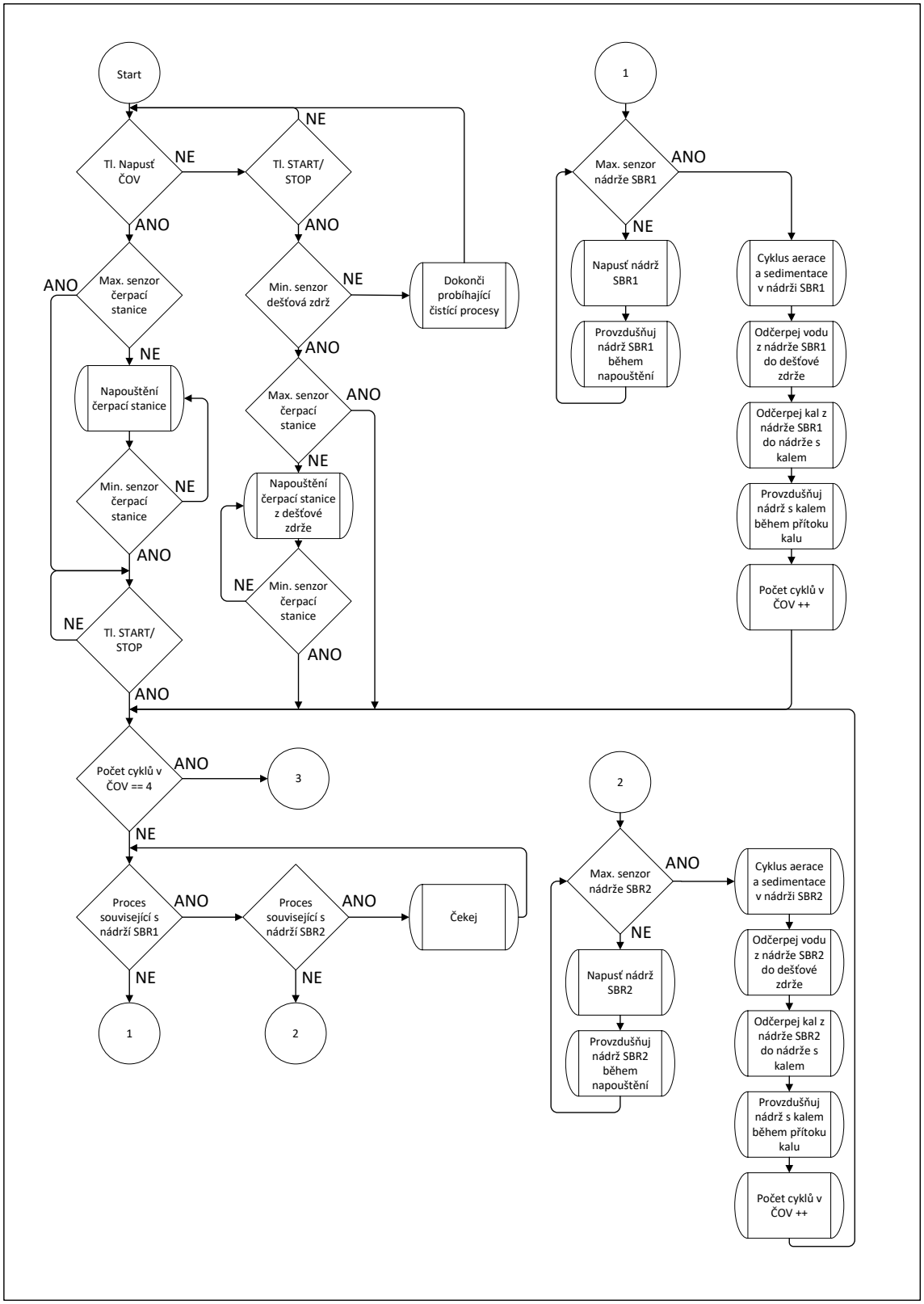
Počet cyklů lze nastavit pomocí HMI v rozsahu 1–5 cyklů. Doba sedimentace/aerace lze také nastavit pomocí HMI v rozsahu 10–1800s. Po provedení všech cyklů je z nádrže SBR1 odčerpána vyčištěná voda do dešťové zdrže, která simuluje i nádrž s mikrosítem (recipient). Čerpání z nádrže SBR1 do dešťové zdrže je provedeno pomocí čerpadla#3 (%Q10.2), které čerpá vodu po určitou dobu, která lze nastavit pomocí HMI v rozsahu 30–600s. Po přečerpání vyčištěné vody je spuštěno čerpání kalu z nádrže SBR1 do nádrže s kalem pomocí čerpadla#5 (%Q10.4). Doba čerpání je opět nastavitelná pomocí HMI v rozsahu 30-300s. Zároveň s čerpáním kalu je spuštěna aerace kalové nádrže pomocí dmychadla#3 (%Q11.3), kdy doba chodu dmychadla lze nastavit skrze HMI v rozsahu 10-1800s. Po přečerpání kalu z nádrže SBR1 je jeden cyklus čištění ukončen a je napouštěna nádrž SBR2 z čerpací stanice pomocí čerpadla#2 (%Q10.1), dokud není sepnut plovákový senzor (%I0.5) pro detekci maximální hladiny vody v nádrži SBR2. Zároveň s napouštěním je nádrž provzdušňována pomocí dmychadla#2 (%Q11.2). Jelikož se v dešťové zdrži nachází z prvního cyklu čištění voda a je jí dostatek pro detekci hladiny pomocí optického senzoru (%I1.0), je pomocí čerpadla#9 (%Q11.0) voda z dešťové zdrže přečerpávána do čerpací stanice, dokud optický senzor (%I1.0) detekuje hladinu nebo dokud není sepnut plovákový senzor(%I0.1) pro detekci maximální hladiny vody v čerpací stanici, čímž je zajištěna cirkulace vody v rámci čističky. Po napuštění nádrže SBR2 a sepnutí plovákového senzoru (%I0.5) je spuštěn určitý počet cyklů aerace a sedimentace, kdy se v časových intervalech střídá provzdušňování nádrže dmychadlem#2 (%Q11.2) a sedimentace, kdy se v nádrži nic neděje a dochází k usazování kalu na dno nádrže. Počet cyklů a doba aerace je stejná jako u provzdušňování nádrže SBR1. Po provedení všech cyklů je z nádrže SBR2 odčerpána vyčištěná voda do dešťové zdrže pomocí čerpadla#4 (Q10.3). Po přečerpání vyčištěné vody je spuštěno čerpání kalu z nádrže SBR2 do nádrže s kalem pomocí čerpadla#6 (%Q10.5). Doba čerpání je nastavitelná pomocí HMI, a je stejná jako u čerpadla#5 (%Q10.4). Zároveň s čerpáním kalu je spuštěna aerace kalové nádrže pomocí dmychadla#3 (%Q11.3). Po přečerpání kalu z nádrže SBR2 je druhý cyklus čištění ukončen a je napouštěna nádrž SBR1. Takhle celkem proběhnou 4 cykly a dojde k pozastavení napouštění ČOV a čistících procesů v čističce, zároveň dojde k vypouštění ČOV. Při vypouštění ČOV je z čerpací stanice plněna nádrž SBR1/SBR2, dle hladiny vody v nádržích, pomocí Čerpadla#11 (%Q11.5)/čerpadla#2 (%Q10.1). Zároveň je nádrž SBR1/SBR2 plněna vodou z nádrže s kalem pomocí čerpadla#7 (%Q10.6)/čerpadla#8 (%Q10.7). Z nádrže SBR1/SBR2 je voda přečerpávána do dešťové zdrže pomocí čerpadla#3 (%Q10.2)/čerpadla#4 (%Q10.3). Z dešťové zdrže je voda čerpána do nádrže ve stole pomocí čerpadla#10 (%Q11.4). Tato čerpadla čerpají, dokud optické senzory minimální hladiny vody pro čerpací stanici (%I0.0), nádrž s kalem (%I0.6), SBR1 nádrž (%I0.2), SBR2 nádrž (%I0.4) a dešťovou zdrž (%I1.0) detekují hladinu vody. Pokud

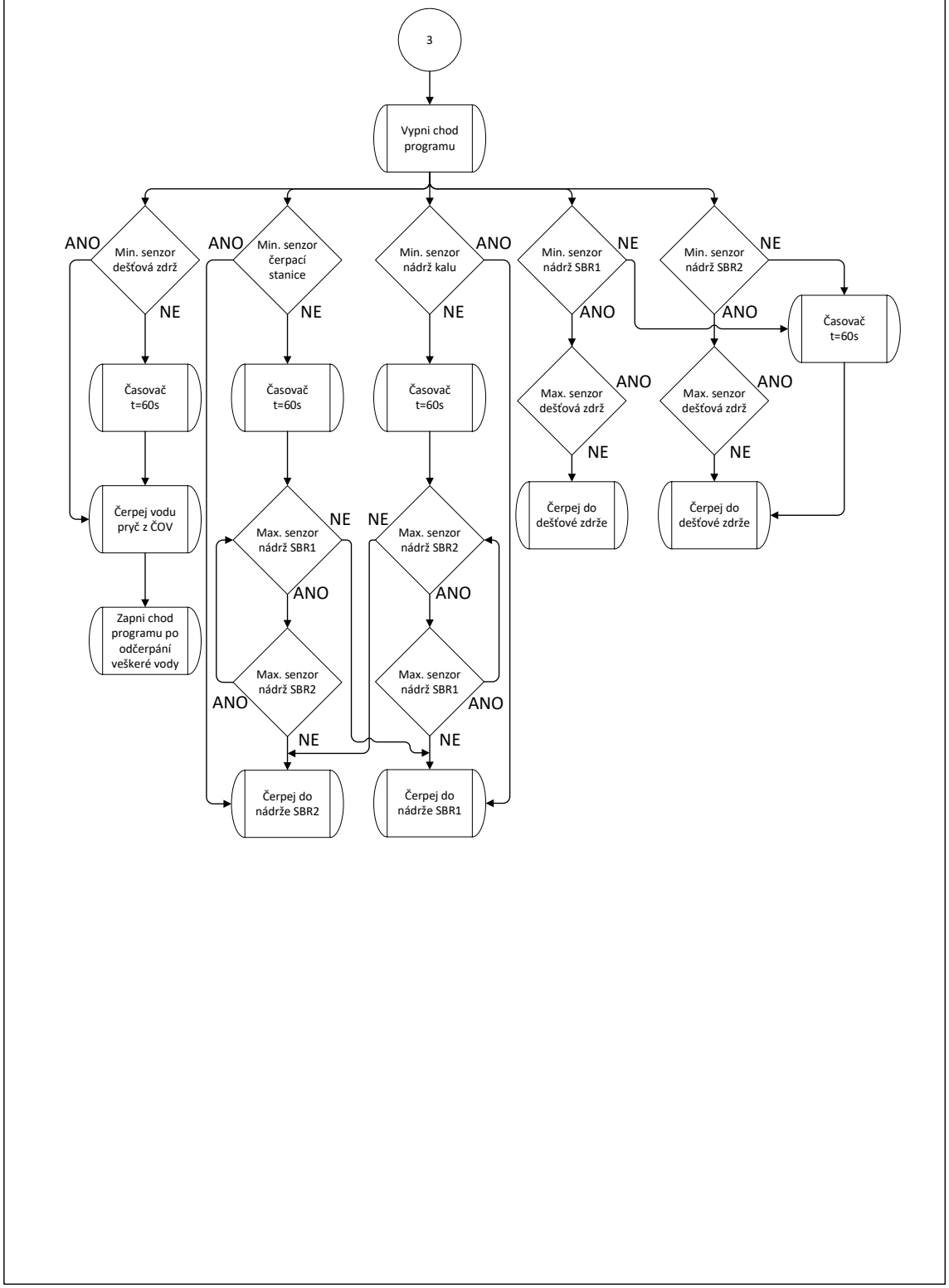
optické senzory nedetekují minimální hladinu vody, odčerpávají jednotlivá čerpadla v jednotlivých nádržích ještě 60s, aby došlo k odčerpání i zbytkové vody. Pokud je napouštění zapnuto, tak po odčerpání vody z ČOV dojde opět k napouštění čističky a opakování čtyř čistících cyklů.



Obr. 3.34: Procesní schéma navrhované čističky odpadních vod.

Diagram procesu ČOV Na následujících dvou diagramech lze vidět detailní rozbor celého procesu ČOV.


Start
Tl. Napusť ČOV
NE
ANO
Tl. START/ STOP
NE
ANO
Max. senzor čerpací stanice
ANO
NE
Min. senzor dešťová zdrž
NE
ANO
Dokonči probíhající čistící procesy
Napouštění čerpací stanice
Max. senzor čerpací stanice
ANO
NE
Min. senzor čerpací stanice
NE
ANO
Napouštění čerpací stanice z dešťové zdrže
NE
Min. senzor čerpací stanice
ANO
Tl. START/ STOP
NE
ANO
Počet cyklů v ČOV == 4
ANO
3
NE
Proces související s nádrží SBR1
ANO
Proces související s nádrží SBR2
ANO
Čekej
NE
NE
1
2
1
Max. senzor nádrže SBR1
ANO
NE
Napusť nádrž SBR1
Provzdušňuj nádrž SBR1 během napouštění
Cyklus aerace a sedimentace v nádrži SBR1
Odčerpej vodu z nádrže SBR1 do dešťové zdrže
Odčerpej kal z nádrže SBR1 do nádrže s kalem
Provzdušňuj nádrž s kalem během přítoku kalu
Počet cyklů v ČOV ++
2
Max. senzor nádrže SBR2
ANO
NE
Napusť nádrž SBR2
Provzdušňuj nádrž SBR2 během napouštění
Cyklus aerace a sedimentace v nádrži SBR2
Odčerpej vodu z nádrže SBR2 do dešťové zdrže
Odčerpej kal z nádrže SBR2 do nádrže s kalem
Provzdušňuj nádrž s kalem během přítoku kalu
Počet cyklů v ČOV ++

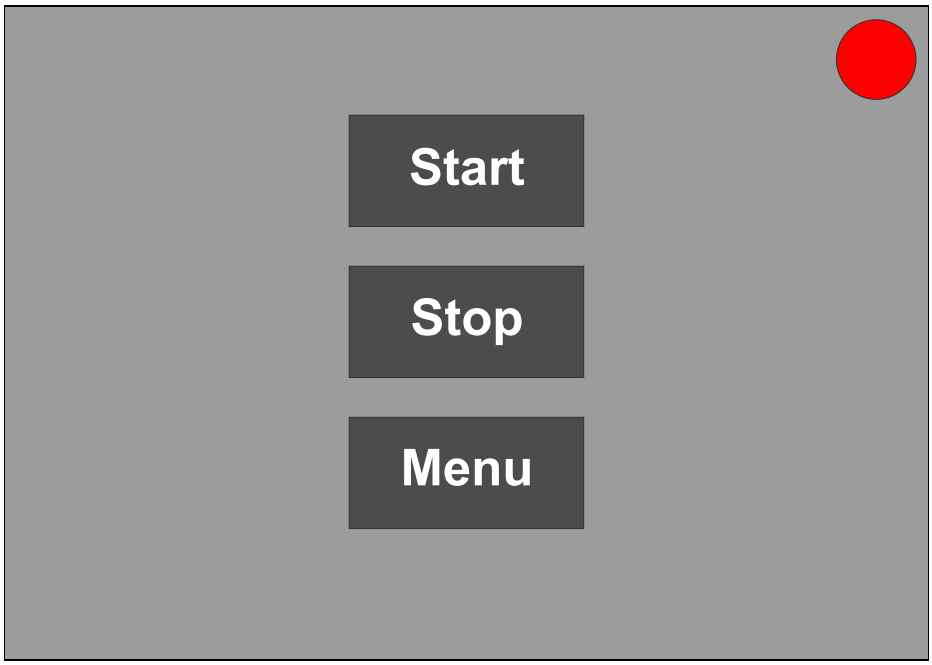

3
Vypni chod programu
Min. senzor dešťová zdrž
ANO
NE
Časovač t=60s
Čerpej vodu pryč z ČOV
Zapni chod programu po odčerpání veškeré vody
Min. senzor čerpací stanice
ANO
NE
Časovač t=60s
Max. senzor nádrž SBR1
NE
ANO
Max. senzor nádrž SBR2
ANO
NE
Čerpej do nádrže SBR2
Min. senzor nádrž kalu
ANO
NE
Časovač t=60s
Max. senzor nádrž SBR2
NE
ANO
Max. senzor nádrž SBR1
ANO
NE
Čerpej do nádrže SBR1
Min. senzor nádrž SBR1
NE
ANO
Max. senzor dešťová zdrž
ANO
NE
Čerpej do dešťové zdrže
Min. senzor nádrž SBR2
NE
ANO
Časovač t=60s
Max. senzor dešťová zdrž
ANO
NE
Čerpej do dešťové zdrže

Možnosti vizualizace

V rámci testovacího prostředí jsou dvě možnosti vizualizace. Prvním z nich je fyzické rozhraní člověk-stroj tedy HMI od firmy Siemens KTP-700 Basic. Pro toto HMI bylo nakonfigurováno několik obrazovek, kde jsou vizualizovány jednotlivé procesní prvky ČOV. Díky tomu lze na HMI tyto prvky sledovat a ovládat (senzory, čerpadla apod.). Pro tento testbed bylo vytvořeno šest obrazovek.

Druhou možností je využití softwaru SCADA softwaru OpenMUC, který byl vytvořen speciálně pro polygon na VUT a jsou do něj zapojeny jednotlivá testovací prostředí. Grafická stránka je naprosto totožná s vizualizací na fyzickém HMI a tedy v rámci SCADA řešení je taktéž šest obrazovek, které jsou následně popsány.

Úvodní obrazovka Jedná se o obrazovku, která se zobrazí při spuštění HMI a PLC a lze ji vidět na obrázku 3.35 . Tato obrazovka obsahuje signalizační kruh v pravém rohu, který slouží k signalizaci chodu programu (červená – program zastaven, zelená – program spuštěn). Dále jsou zde tři možnosti tlačítek. Prvním je tlačítko Start – spouští chod programu. Druhé tlačítko je Stop, které slouží k zastavení programu. Pokud je zmáčknuto tlačítko Stop a poté opětovně stisknuto tlačítko Start, tak čistička pokračuje opět v činnosti tam, kde byla zastavena. Posledním tlačítkem je tlačítko Menu, kterým lze přejít na novou obrazovku pro konfiguraci procesů a sledování jednotlivých parametrů.

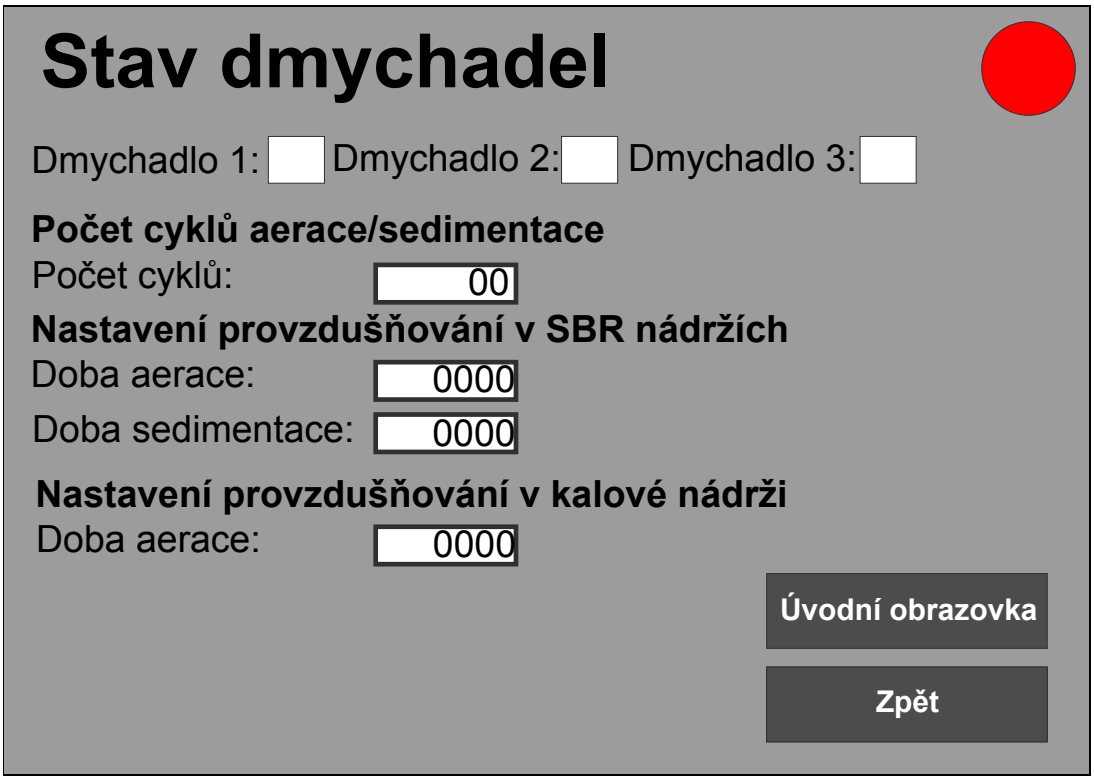


Obr. 3.35: Úvodní obrazovka.

Menu Obrazovku *Menu* kterou lze vidět na obrázku 3.36 je rozcestníkem pro nastavování a sledování jednotlivých parametrů. Je zde pět možností pro přepnutí na další obrazovky. Jsou to tlačítka: Stav dmychadel, Stav čerpadel, Stav nádrží, Vypustit čističku a tlačítko Zpět (pro návrat na úvodní obrazovku).

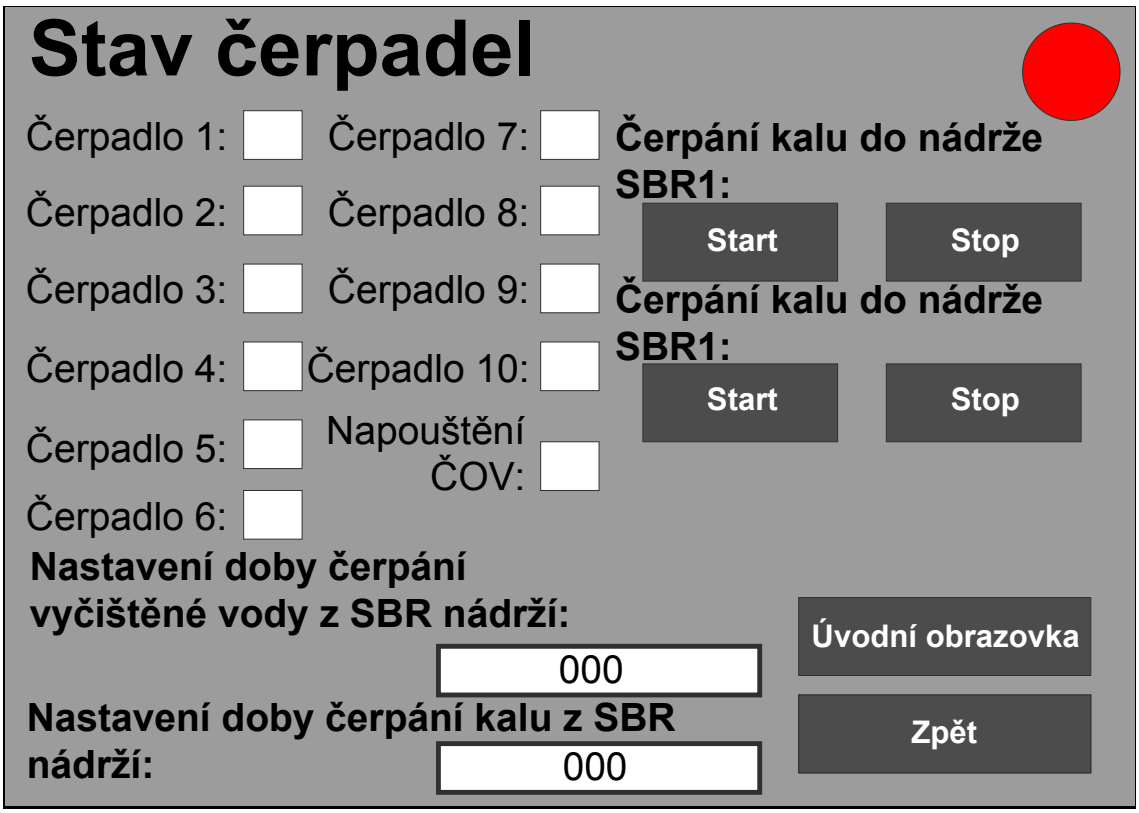
Obr. 3.36: Obrazovka menu.

Stav dmychadel Obrazovku *Stav dmychadel* lze vidět na obr. 3.37. Na obrazovce lze vidět indikace chodu dmychadel – nečinnost dmychadla je značena červenou barvou, aktivita dmychadla je označena zeleně. Na této obrazovce lze nastavovat také samotná dmychadla. Lze nastavit počet cyklů aerace a sedimentace dmychadel v celočíselném rozmezí 1–5 cyklů. Dále lze nastavit také dobu aerace a také dobu sedimentace. Zde se nastavuje čas v sekundách v rozmezí 10 s až 1800 s. Tyto parametry jsou pouze pro dmychadla v SBR nádržích (#1 a #2). Pro kalovou nádrž probíhá pouze aerace kterou lze také nastavit. Nakonec jsou zde dvě tlačítka návrat na úvodní obrazovku a tlačítko zpět díky kterému se lze vrátit na obrazovku Menu.



Obr. 3.37: Obrazovka stavu dmychadel.

Stav čerpadel Obrazovku *Stav čerpadel* lze vidět na obrázku 3.38. Na obrazovce lze vidět indikaci chodu všech čerpadel – nečinnost čerpadla je značena červeně, aktivita čerpadla je značena zeleně. Dále jsou zde tlačítka pro čerpání kalu do nádrží SBR #1 a SBR #2, kdy je možné čerpat pouze do jedné nádrže v jednu chvíli nelze mít sepnuté obě čerpadla v jednu chvíli. Při zapnutí druhé čerpadla se čerpadlo, které je v provozu vypne a zapne se druhé čerpadlo. Dále je zde možné nastavit dobu čerpání vody z SBR nádrží (nastavuje se v sekundách, a lze nastavit hodnotu v rozmezí 10 s –600 s). Nakonec lze také nastavit dobu čerpání kalu z SBR nádrží (hodnota musí být v rozmezí 10 s – 300 s). U reálných aplikací čistíren se nezadává čas na čerpání ale čerpá se dle senzorů hladiny (tedy hodnoty jsou nastavovány v metrech dle hladiny vody). Nakonec jsou zde dvě tlačítka návrat na úvodní obrazovku a tlačítko zpět díky kterému se lze vrátit na obrazovku Menu.



Obr. 3.38: Obrazovka stavu čerpadel.

Stav nádrží Obrazovku *Stav nádrží* lze vidět na obrázku 3.39. Zde lze vidět všech 5 nádrží a stav senzorů v těchto nádržích. Tedy lze se podívat na indikátory jestli je minimální nebo maximální hladina v nádrži. Prázdná nádrže má indikátor minimální hladiny zbarven zeleně a maximální zbarven červeně. Pokud hladina překročí výšku minimálního senzory indikátor se zbarví červeně. Pokud je dosažen senzor maximální hladiny je indikátor maximální hodnoty zbarven zeleně. Nakonec jsou zde dvě tlačítka návrat na úvodní obrazovku a tlačítko zpět díky kterému se lze vrátit na obrazovku Menu.

Obr. 3.39: Obrazovka stavu nádrží.

Vypuštění ČOV Obrazovku *Vypuštění* lze vidět na obrázku 3.40. Na obrazovce lze vidět stav vody při vypouštění v jednotlivých nádržích, lze zde pozorovat maximální a minimální hladinu v jednotlivých nádržích. Je zde také indikace chodu čerpadla. Červená znamená, že čerpadlo je nečinné a zelená znamená že čerpadlo je v provozu. Voda je čerpána z recipientu (dešťové zdrže) zpátky do nádrže s vodou, která se nachází pod stolem.

Obr. 3.40: Obrazovka Vypuštění ČOV.

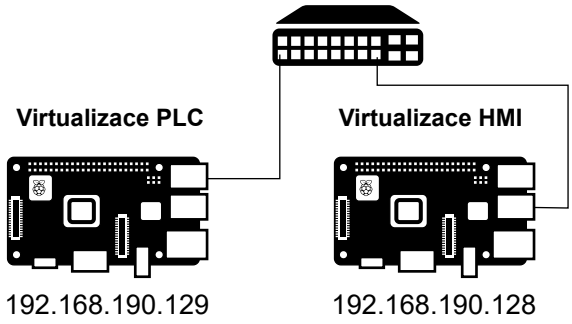
Funkcionality testbedu

V rámci testbedu je možné detekovat maximální a minimální hladiny v ČOV. Dále je možné využít přečerpávání z nádrží a využít aerace a sedimentace. Je možné simulovat přítok vody a odtok vyčištěné vody. Celý testbed pracuje v cyklu samostatně, ale je možné do něj zasahovat a upravovat provoz tak jako v reálných čističkách. V rámci procesu je simulován přítok vody do čerpací stanice, dále přítok vody do dešťové zdrže a přečerpávání do čerpací stanice. Je zde simulován také proces aerace a sedimentace, tedy čištění vody pomocí dvou nádrží SBR. Celkově může být testbed ovládán pomocí HMI kde lze vypustit celou ČOV, popřípadě měnit parametry aerace a sedimentace nebo sledovat vizualizované hladiny vody.

Virtualizovaná verze

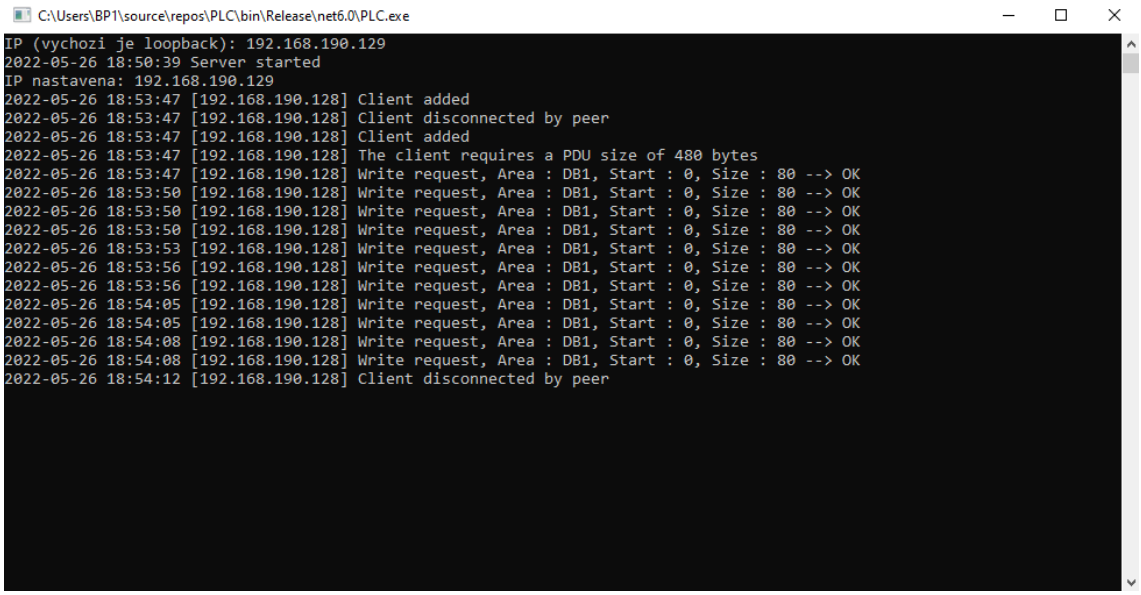
V rámci možnosti více simulací byla také vytvořena virtualizovaná verze simulující proces ČOV. Tato virtualizace byla realizována pomocí knihovny sharp7 jež je od stejných autorů jako knihovna SNAP7 s tím rozdílem, že tato knihovna je určena

pro jazyk C#. Spuštění této simulace je možné jak v rámci operačního systému Windows, tak v rámci operačního systému Linux. Komponenty jsou simulovány dvěma virtuálními stanicemi. První je virtualizované PLC a druhá stanice je virtualizované HMI. Aby bylo možné tyto virtuální stroje použít na operačním systému Linux a tedy udělat z nich emulační zařízení pracující na Raspberry pi bylo nutné v rámci Linuxu na Raspberry pi nainstalovat software Mono, který umožňuje spouštět aplikace v programovacím jazyce C#. Místo rozšíření knihovny snap7 snap7.dll muselo být použito rozšíření libsnap7.so. Schéma emulace virtualizované verze lze vidět na obrázku 3.41. Celý simulační proces se snaží co nejvíce přiblížit právě předchozímu fyzickému testbedu ČOV a tedy procesní běh je stejný jako u fyzického testbedu. Z toho důvodu zde již není popsán.



Obr. 3.41: Schéma virtualizované verze.

PLC Program PLC řídí celou komunikaci, oznamuje události na které reaguje. Události typu připojení, odpojení klienta. Dále pak událost o velikosti PDU, zapisování do datových bloků nebo čtení z těchto datových bloků. PLC je vizualizováno pouze v rámci příkazové řádky, tedy nemá vlastní grafické rozhraní (jelikož reálná PLC také nemají grafická rozhraní). Na obrázku 3.42 lze vidět běh programu simulující PLC. V rámci komunikace lze vidět připojení klienta a velikost PDU. Lze také vidět akci zápisu informací do datového bloku.



Obr. 3.42: Běh programu virtualizovaného PLC.

HMI Program HMI již má vlastní grafické rozhraní (jak je u HMI běžné). Toto grafické rozhraní lze vidět na obrázku 3.43. Pomocí tohoto rozhraní lze komunikovat s PLC a měnit příkazy v rámci procesu ČOV. Je možné nastavit IP adresu PLC na které se má HMI připojit. Dále se zadají parametry doba plnění nádrží, počet cyklů aerace a sedimentace, provzdušňování v kalové nádrží a SBR nádržích. Je také možné nastavit dobu čerpání. Po spuštění začne proces stejně jak ve fyzické ČOV.

Obr. 3.43: Ukázka virtualizovaného HMI.

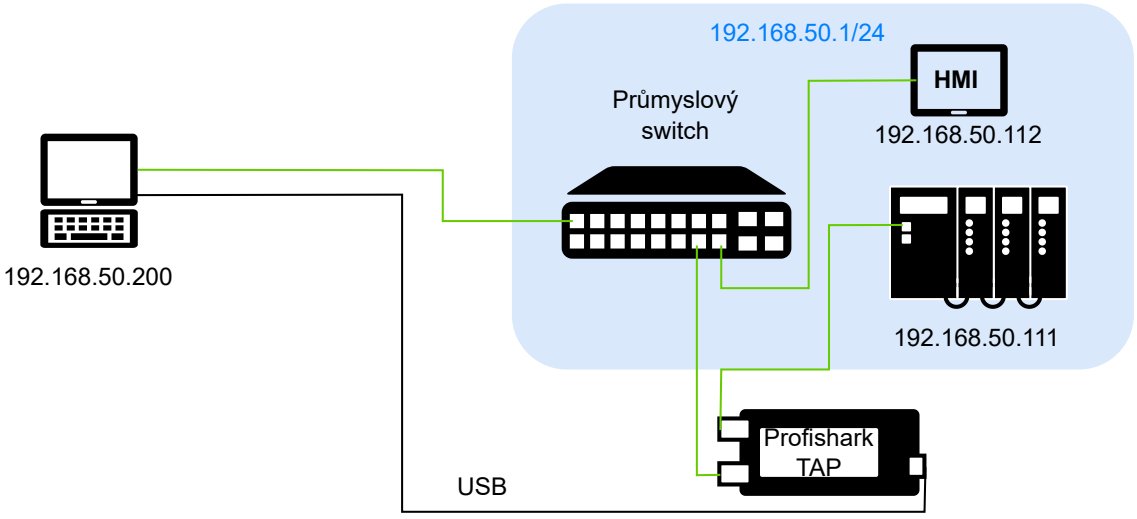
3.2.5 Testování a verifikace

Testovací protokol: Generování dat standardního provozu

Předpoklady Cílem generování dat standardního procesu je kvantifikovat možnosti sběru dat při standardním procesu testovacího prostředí. Jednotlivé měřené cíle byly následující:

- Změřit standardní dobu cyklu jednoho procesu ČOV.
- Množství vygenerovaných paketů za tento jeden cyklus.
- Množství paketů pro protokol S7comm.
- Počet S7comm paketů za sekundu.
- Informace o průměrných velikostech zpráv protokolu S7comm.
- Změřit možnosti generování pro speciální proces vypouštění.

Na obrázku 3.44 lze vidět schéma zapojení pracoviště při zachytávání standardního provozu testovacího prostředí ČOV.

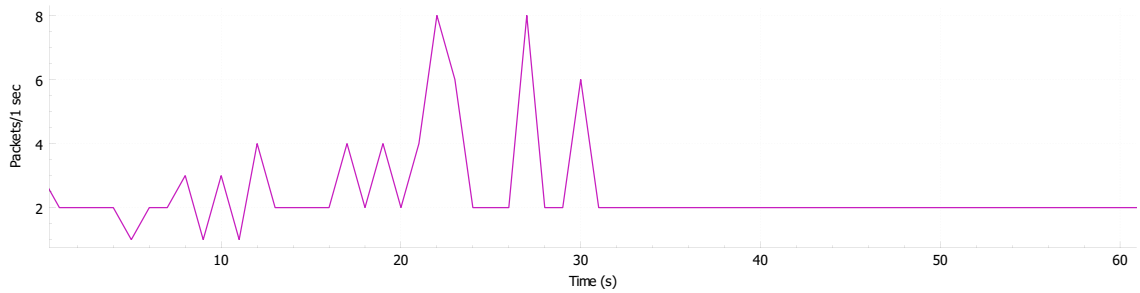


Obr. 3.44: Schéma pracoviště při zachytávání standardního provozu.

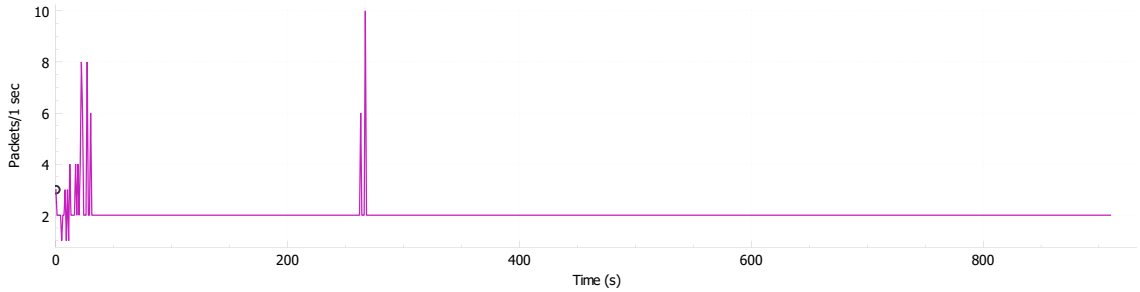
Testování Generování standardního provozu je důležité pro možnost zaznamenání standardního provozu ČOV bez anomálií. Díky tomuto standardnímu provozu lze identifikovat změny v rámci časových řad přijatých a odeslaných paketů. Pomocí analyzátoru komunikace byl zachycen jeden cyklus procesu ČOV. To znamená, že se čtyřikrát zopakuje čistící cyklus a dojde k odčerpání vody z čistírny.

Jeden cyklus testbedu při čištění vody trvá při základním nastavení (které lze změnit) 15 minut. Za tento cyklus je celkově vygenerováno 4737 paketů z toho 1862 paketů protokolu S7comm (39,3 %). Je tedy možno vygenerovat v rámci celého cyklu 5,2 paketů za sekundu a pouze pro S7comm 2,0 paketů za sekundu. Jsme tedy

schopni generovat 790 bajtů za sekundu (6322 bitů/s) v rámci celého cyklu a pouze v rámci S7comm jsme schopni generovat 381 bajtů za sekundu (3055 bitů/s). Jednotlivé S7 zprávy mají v průměru 187 Bajtů. Na obrázku 3.45 lze vidět provoz paketů S7comm během jedné minuty největšího provozu. Na dalším obrázku 3.46 lze vidět to stejné při provozu celého cyklu (15 minut).



Obr. 3.45: Vyobrazení přenosů paketů během minuty provozu – pouze komunikace S7comm.



Obr. 3.46: Přenos paketů během celého cyklu provozu (15min) – pouze komunikace S7.

Dále je možno generovat další data z procesu pouze vypouštění ČOV. Jedná se o speciální proces, který není zahrnut do běžného provozu. Tento proces je prováděn jen velmi zřídka, ale lze jej také počítat do standardního provozu. Proto je třeba mít data i ze standardního procesu vypouštění pro možnost zahrnutí detekčních metod i do tohoto procesu. Tento proces je závislý na aktuálním množství vody v ČOV, ale při nejnižší hladině, proces trvá 11 minut. Za tuto dobu je celkově vygenerováno 334 kB dat. Celkově je vygenerováno 2223 paketů z toho 1382 S7comm paketů. Průměrně je při tomto procesu generováno za sekundu 3,3 paketů nebo pokud se zaměříme pouze na S7comm pakety tak 2 pakety za sekundu. Průměrná rychlost je 443 bajtů za sekundu pro celý proces a pouze pro protokol S7comm je to v průměru 319 bajtů za sekundu.

Souhrn dosažených výsledků V tabulce 3.18 lze vidět souhrn hodnot při standardním provozu procesů ČOV, který vychází z části testování.

Tab. 3.18: Souhrn hodnot při standardním provozu procesů ČOV.

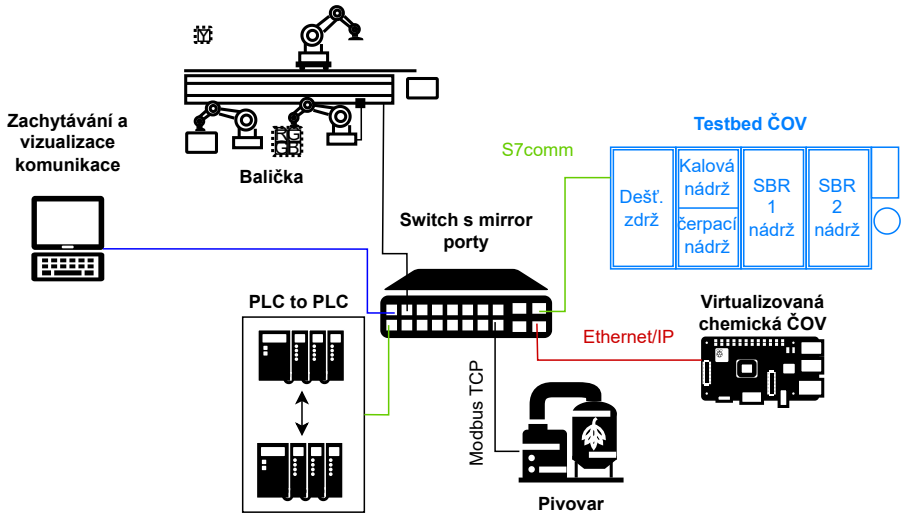
Standardní proces ČOV	
Doba cyklu	15 minut
Celkový počet vygenerovaných paketů	4737
Počet S7comm paketů	1862
Počet S7comm paketů za sekundu	2
Průmer S7comm zprávy	187 Bajtů
Proces vypouštění	
Doba cyklu	11 minut
Celkový počet vygenerovaných paketů	2223
Počet S7comm paketů	1382
Počet S7comm paketů za sekundu	2
Průmer S7comm zprávy	187 Bajtů

Testovací protokol: Bezpečnostní cvičení

Předpoklady Cílem bezpečnostního cvičení je sběr dat standardního a nestandardního provozu pro možnosti testování modelů strojového učení pro detekci anomálií v průmyslových sítích. Testování je zaměřeno na generování dat v rámci celého polygonu, to znamená několik testovacích prostředí průmyslových sítí v jedné síti. V rámci tohoto testování bylo zapojeno i testovací prostředí ČOV na kterém probíhaly také bezpečnostní cvičení. Komunikace s tímto testovacím prostředím byla realizována pomocí protokolu S7comm. Testovací den se rozdělí na tři části, a to na část přípravy, část normálního provozu a část testování různých anomálií. Ve fázi příprav jsou do sítě připojovány a konfigurovány jednotlivá průmyslová zařízení. Ve fázi normálního provozu pracují jednotlivé scénáře v běžném provozu a opakují jednotlivé cykly. V části testování anomálií jsou na zařízení vytvářeny různé útoky. Předpoklady pro toto testování byly následující:

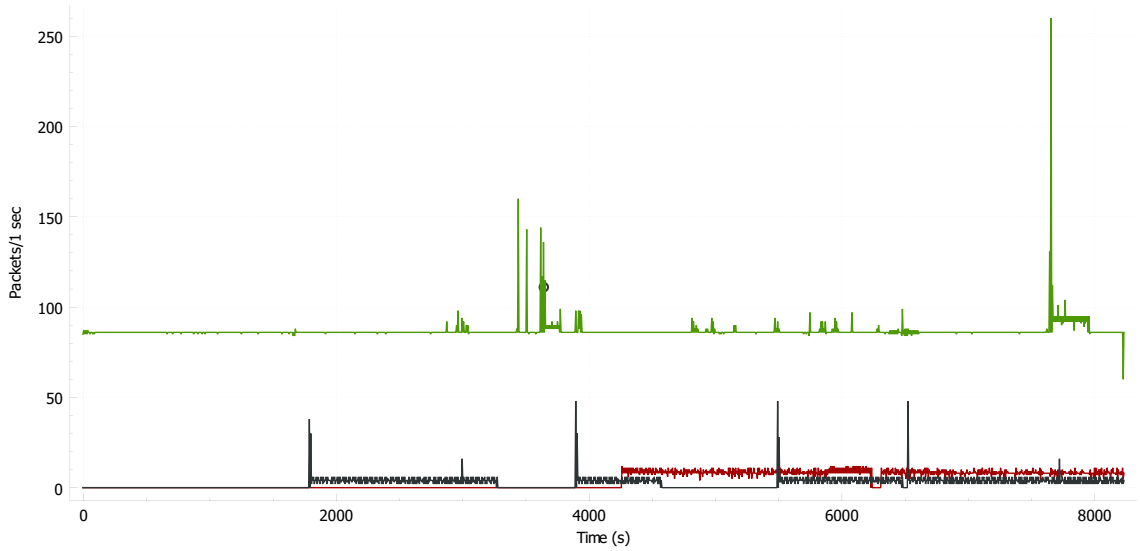
- Definovat objem dat, čas a počet paketů při fázi příprav.
- Definovat objem dat, čas a počet paketů při fázi normální provoz.
- Definovat objem dat, čas a počet paketů při fázi testování anomálií.
- Zaměřit se na informace o množství vygenerovaných paketů pro protokoly S7comm, Modbus-TCP a Ethernet/IP.

Na obrázku 3.47 lze vidět zapojení jednotlivých testovacích prostředí pro bezpečnostní cvičení. Testovací prostředí ČOV je zde vyznačeno modrou barvou.



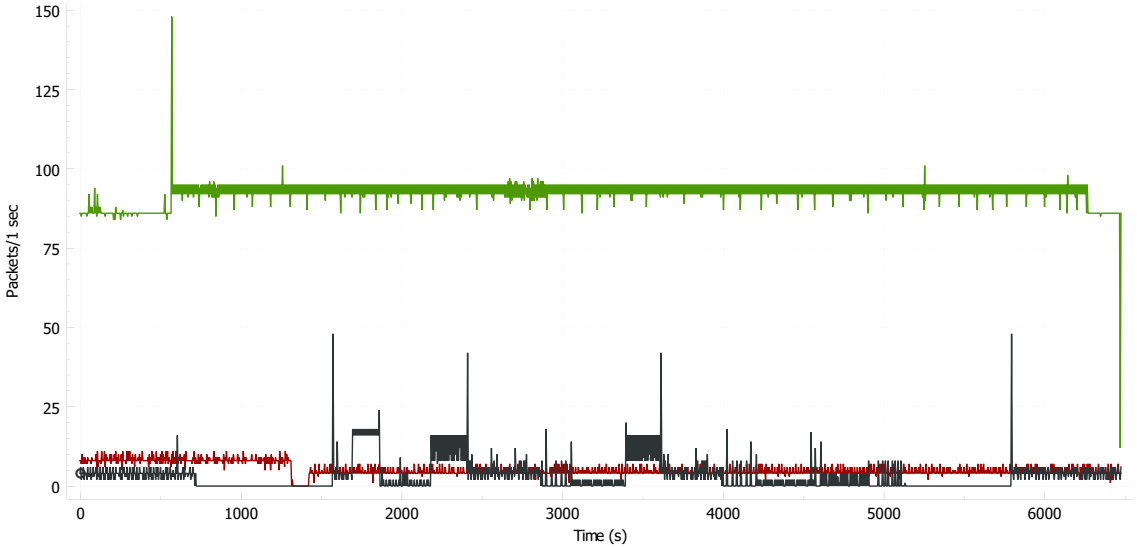
Obr. 3.47: Schéma bezpečnostního cvičení – ČOV vyznačena modrou barvou.

Fáze příprav Z této fáze máme datové soubory (pcapy) o objemu 1058 MB celá fáze trvala 2 hodiny a 17 minut. Kdy bylo zachyceno celkově 3 054 855 paketů. Průměrně bylo vygenerováno 371.4 paketů za sekundu a 119 000 bajtů za sekundu. Z toho bylo pro protokol S7comm celkově 23,3 % paketů tedy 711 339 paketů. Průměrně bylo těchto paketů generováno 86,5 za sekundu a průměrný objem toku za sekundu bylo 7567 bajtů. Pro protokol Modbus TCP bylo celkově 0,6 % paketů tedy 19 647 paketů Průměrně byly tyto pakety generovány 3,1 za sekundu a průměrný objem toku za sekundu bylo 236 bajtů. Pro protokol Ethernet/IP bylo celkově 1,1 % paketů tedy 32 237 paketů. Průměrně bylo těchto paketů generováno 8,1 za sekundu a průměrný objem toku za sekundu bylo 920 bajtů. Na obrázku 3.48 lze vidět průběh paketů jednotlivých protokolů během celého cyklu.



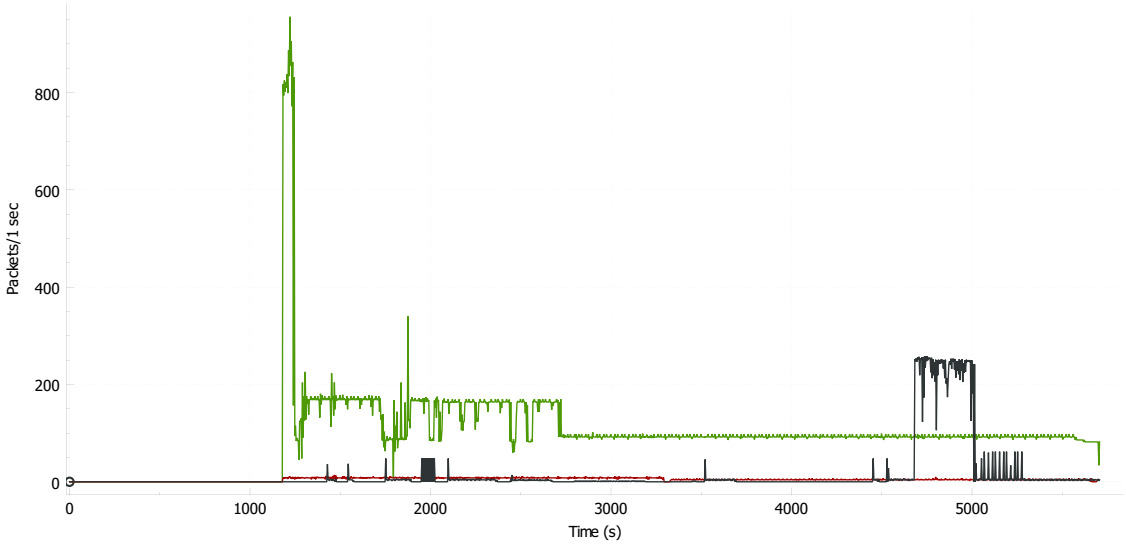
Obr. 3.48: Přenos paketů během celého cyklu provozu. Zeleně jsou označeny pakety S7comm, červeně jsou označeny pakety Ethernet/IP a černě jsou označeny pakety Modbus-TCP.

Normální provoz Z této fáze máme datové soubory (pcapy) o objemu 348 MB celá fáze trvala 1 hodinu a 47 minut. Kdy bylo zachyceno celkově 2 644 787 paketů. Průměrně bylo vygenerováno 408,8 paketů za sekundu a 44 000 bajtů za sekundu. Z toho bylo pro protokol S7comm celkově 22,5 % paketů tedy 594 658 paketů. Průměrně bylo těchto paketů generováno 91.9 za sekundu a průměrný objem toku za sekundu bylo 8402 bajtů. Pro protokol Modbus TCP bylo celkově 0,7 % paketů tedy 19 470 paketů Průměrně byly tyto pakety generovány 3 za sekundu a průměrný objem toku za sekundu bylo 233 bajtů. Pro protokol Ethernet/IP bylo celkově 1,2 % paketů tedy 32 129 paketů. Průměrně bylo těchto paketů generováno 5 za sekundu a průměrný objem toku za sekundu byl 562 bajtů. Na obrázku 3.49 lze vidět průběh paketů jednotlivých protokolů během celého cyklu.



Obr. 3.49: Vyobrazení přenosů paketů během celého cyklu provozu. Zeleně jsou označeny pakety S7comm, červeně jsou označeny pakety Ethernet/IP a černě jsou označeny pakety Modbus.

Anomálie Z této fáze máme datové soubory (pcapy) o objemu 1000 MB celá fáze trvala 1 hodinu a 35 minut. Bylo zachyceno celkově 5 159 155 paketů. Průměrně bylo vygenerováno 904,3 paketů za sekundu a 160 000 bajtů za sekundu. V této fázi byly testovány také DoS útoky proto větší objem. Z toho bylo pro protokol S7comm celkově 15,1 % paketů tedy 781 281 paketů. Průměrně bylo těchto paketů generováno 136,9 za sekundu a průměrný objem toku za sekundu bylo 22 000 bajtů. Pro protokol Modbus TCP bylo celkově 1,7 % paketů tedy 89 344 paketů. Průměrně bylo těchto paketů generováno 16,8 za sekundu a průměrný objem toku za sekundu bylo 1 287 bajtů. Pro protokol Ethernet/IP bylo celkově 0,7 % paketů tedy 34 812 paketů. Průměrně bylo těchto paketů generováno 6,3 za sekundu a průměrný objem toku za sekundu bylo 717 bajtů. Na obrázku 3.50 lze vidět průběh paketů jednotlivých protokolů během celého cyklu.



Obr. 3.50: Vyobrazení přenosů paketů během celého cyklu provozu. Zeleně jsou označeny pakety S7comm, červeně jsou označeny pakety Ethernet/IP a černě jsou označeny pakety Modbus.

Souhrn dosažených výsledků V tabulce 3.19 lze vidět souhrn hodnot bezpečnostního cvičení při všech třech fázích.

Tab. 3.19: Souhrn hodnot bezpečnostního cvičení.

Fáze příprav	
Doba fáze příprav	137 minut
Objem dat	1058 MB
Počet paketů	3 054 855
Průměrný počet paketů za sekundu	371,4
Počet S7comm paketů/ za sekundu generováno	711 339 / 86,5
Počet Modbus TCP paketů/ za sekundu generováno	19 647 / 3,1
Počet Ethernt/IP paketů/ za sekundu generováno	32 237 / 8,1
Normální provoz	
Doba fáze normálního provozu	107 minut
Objem dat	348 MB
Počet paketů	2 644 787
Průměrný počet paketů za sekundu	408,8
Počet S7comm paketů/ za sekundu generováno	594 658 / 91,9
Počet Modbus TCP paketů/ za sekundu generováno	19 470 / 3
Počet Ethernt/IP paketů/ za sekundu generováno	32 129 / 5
Testování anomálií	
Doba fáze testování anomálií	95 minut
Objem dat	1000 MB
Počet paketů	5 159 155
Průměrný počet paketů za sekundu	904,3
Počet S7comm paketů/ za sekundu generováno	781 281 / 136,9
Počet Modbus TCP paketů/ za sekundu generováno	89 344 / 16,8
Počet Ethernt/IP paketů/ za sekundu generováno	34 812 / 6,3

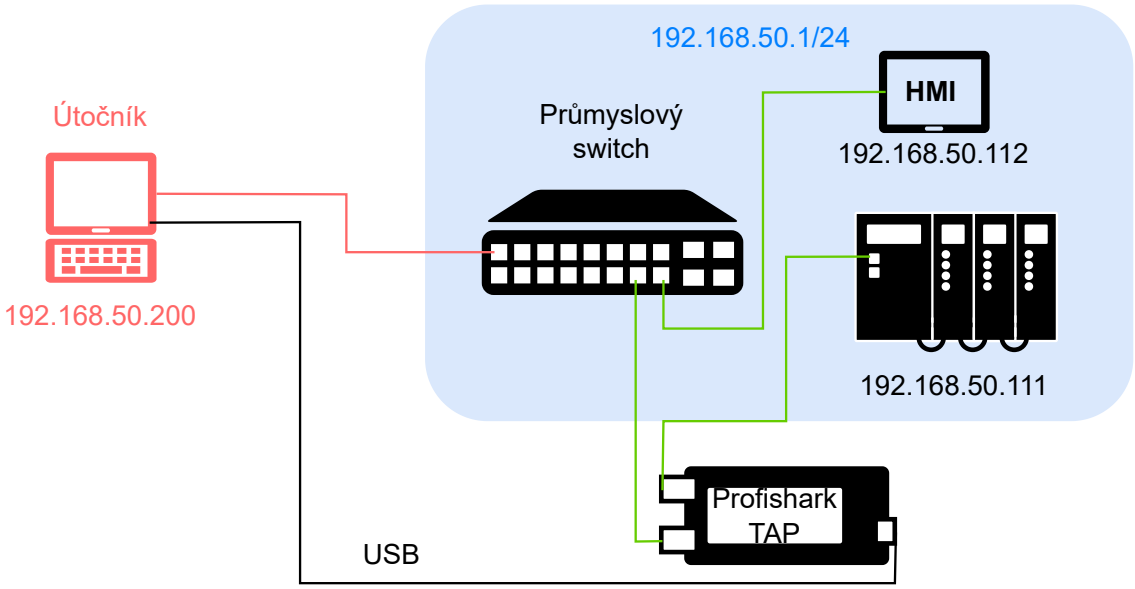
Testovací protokol: Testování útoků na ostrém provozu

Předpoklady Cílem tohoto testování je demonstrace útoků na testovací prostředí pro možnost sběru dat z nestandardního (anomálního) provozu, díky tomu je možné

vygenerovat datasety pro výzkum v rámci metod strojového učení pro detekci anomálií v průmyslových sítích. V rámci testbedu byl vytvářen nestandardní (anomální) provoz, který se zaměřoval na kybernetické útoky. Díky tomu je možné generovat vyvážené datasety a mít tak vhodný poměr mezi standardním a nestandardním provozem pro následné detekce anomálních provozů. V rámci testování bylo provedeno několik scénářů s útoky, ty lze rozdělit do následujících kategorií:

- Vkládání chyb do běhu programu (injekční útoky):
  - Útoky pomocí skriptů.
  - Útoky pomocí externích aplikací.
- Získávání informací a skenování:
  - Získávání informací o zařízení pomocí externích aplikací.
  - Získání programu PLC.
  - Skenování v průmyslové síti a dopad skenování.
- DoS útoky:
  - DoS útoky na průmyslová zařízení.

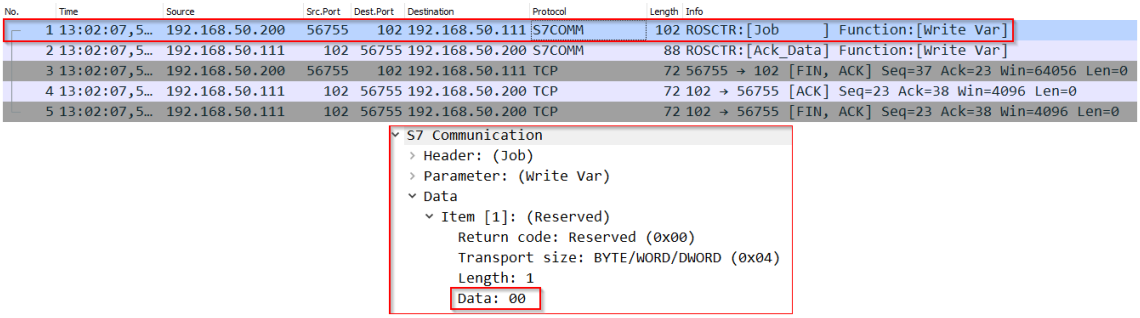
Vkládání chyb do běhu programu (injekční útoky) V rámci této kategorie útoků byly testovány dva různé přístupy možností změn běhu programu vkládáním informací přímo do kódu. Prvním z nich bylo využití vlastních skriptů fungujících na základě Snap7 knihovny. Jako další byl zvolen jeden zástupce externích softwarů pracujících v režimu klient. Oba scénáře byly realizované se stejným nastavením sítě a se stejným útočným počítačem viz obrázek 3.51. Na obrázku je modře zvýrazněna běžná průmyslová síť na které jsou zařízení na IP adresách v rozsahu 192.168.50.1–255. Pro možnost zjišťování dopadů a monitorování komunikace byl před PLC Siemens S7-300 připojen Profishark TAP, který umožňuje vytvářet pcap soubory a monitorovat tak provoz, který přichází a odchází z PLC. V rámci průmyslové sítě byl v tomto případě monitorován pouze provoz na ČOV. V této síti se nachází průmyslový switch dále PLC na IP adrese 192.168.50.11 a také HMi na IP adrese 192.168.50.112. Nakonec pro útok byl zvolen klasický notebook s operačním systémem Windows 10. Tento počítač se nachází na IP adrese 192.168.50.200. K průmyslové síti je možné přistoupit buď fyzicky, nebo přes VPN, tedy validní jsou oba scénáře, především tedy možnost vzdáleného přístupu. Následně budou popsány výše zmíněné dva přístupy útoku.



Obr. 3.51: Rozvržení sítě pro útok (injection code).

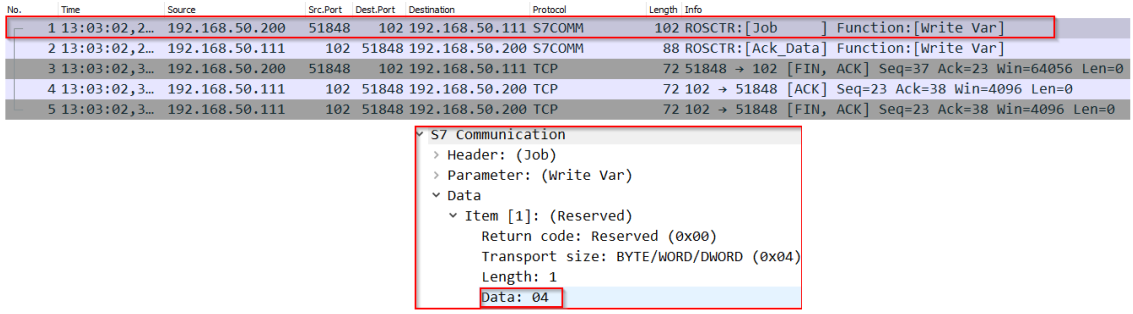
Útok pomocí skriptu V rámci tohoto útoku byl napsán skript v jazyce Python u kterého byla využita knihovna SNAP7, která umožňuje komunikaci s PLC od firmy Siemens. Tedy přesněji umožňuje komunikaci pomocí protokolu S7comm, který je proprietárním protokolem firmy Siemens. Pomocí importované knihovny SNAP7, se nejdříve připojí zařízení k PLC pomocí rozhraní PG a parametrů 0,2 které jsou standardní při základním nastavení a použití jedno PLC Siemens S7-300. Následně je zjištěn stav PLC a poté je zde vyčtení parametrů PLC pro zjištění současného stavu v databázi. Jedná se o databázi 11 ve které je hned na prvním bajtu uložena informace o běhu programu v hexadecimálním tvaru. Pokud program běží znamená to že je v paměti hexadecimální číslo 04. Ve skriptu je připravena falešná hodnota reprezentující hodnotu 00, to znamená zastavení běhu programu. Pokud je tato hodnota zaslána na zařízení běh programu se okamžitě zastaví. Následně je možné opětovně program spustit pomocí proměnné program_on. To znamená, že tímto útokem lze převzít plnou kontrolu nad během PLC, lze přepisovat jakékoli hodnoty v jiných databázích.

Následuje ukázka komunikace, která byla zachycena analyzátorem v síti. Nejdříve tedy byla zaslána data na zastavení programu ČOV při běžícím procesu. Komunikaci lze vidět na obrázku 3.52. Nejdříve lze vidět zaslanou zprávu z přepsáním hodnot na PLC a následné potvrzení. Žádost o přepsání dat na 00 lze vidět také na obrázku. Důležité parametry jsou vyznačeny červeně.



Obr. 3.52: Ukázka útoku – vypnutí programu falešnými daty.

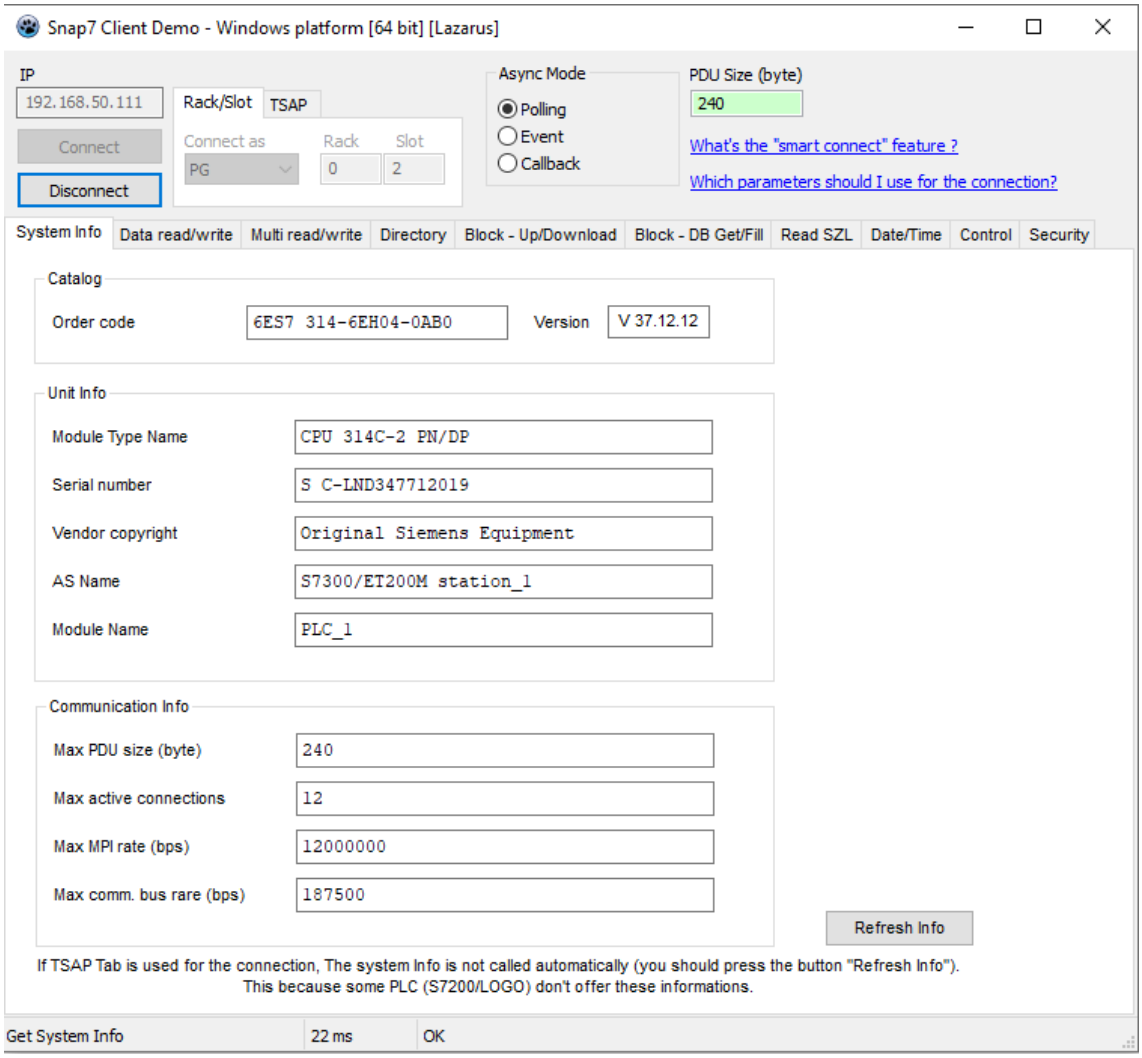
Poté byl po nějaké době program opětovně spuštěn pomocí stejného skriptu, akorát byla hodnota opět přepsána na 04. To znamená spuštění procesu ČOV. Ukázka je na obrázku 3.53.



Obr. 3.53: Ukázka útoku – zapnutí programu.

Útok pomocí externí aplikace Možnosti změny parametrů v běhu programu byly testovány i pomocí aplikace, kterou je možné se připojit na zařízení, aplikace nese název clientdemo a jedná se o simulaci klientské stanice od tvůrců knihovny SNAP7. S touto aplikací lze také demonstrovat zastavení cyklu procesu čističky, ale v rámci tohoto testování byla zvolena k přepsání hodnota počtu cyklů aerace. Standardně je počet cyklů aerace nastaven na 2. Při tomto útoku demonstrujeme možnost změny cyklů aerace na vyšší počet a také na přetečení pole pro volbu tohoto počtu aerace, jelikož je možné na HMI volit možnosti pouze v rozsahu 1–5 ne více ani méně. Při změně se v procesu cykly mění okamžitě například při nastavení hodnoty 5 začne místo 2 cyklů opakovat 5 cyklů, tedy lze díky tomuto útoku demonstrovat manipulaci s funkcionalitami čističky. Nejdříve lze popsat samotnou aplikaci clientdemo, tu lze vidět na obrázku 3.54. V rámci této aplikace se lze připojit jakožto client na jakékoli PLC disponující komunikací pomocí protokolu S7comm. Po připojení k PLC lze vyčíst informace jako výrobní číslo, verze zařízení, typ modulu, sériové číslo, název a další informace. Lze také vyčítat a zapisovat hodnoty a další. Nejdříve byly tedy

vyčteny informace o hodnotách v databázi při procesu. Ukázka vyčtených bajtů když byl na HMI nastaven počet cyklů na 1 lze vidět na obrázku 3.55, červeně je zde označeno místo v databázi kde lze hodnotu přepisovat.



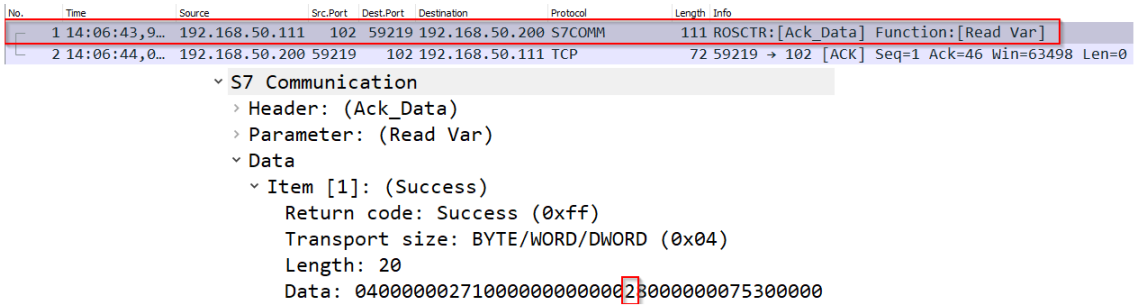
Obr. 3.54: Ukázka rozhraní programu clientdemo.

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0000	$04	$00	$00	$00	$27	$10	$00	$00	$00	$00	$00	$01	$00	$00	$00	$00
0010	$75	$30	$00	$00	$00	$00	$00	$00	$00	$00	$00	$00	$00	$00	$00	$00

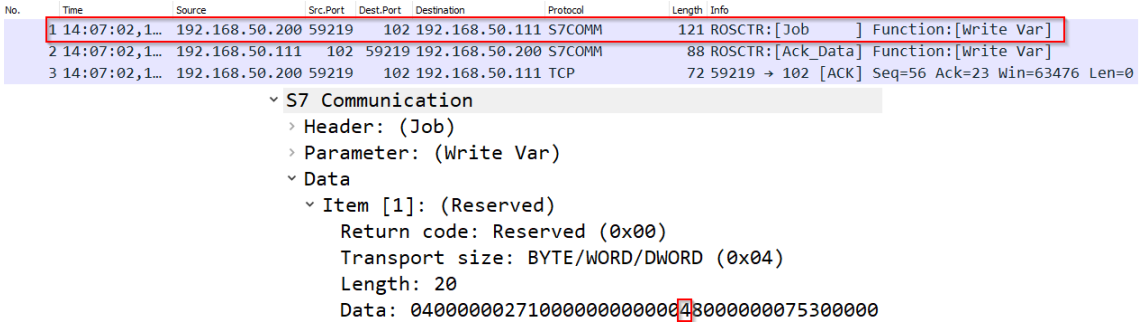
Obr. 3.55: Počet cyklů nastaven na 1.

Postup byl takový, že proces byl uveden do původního stavu a byl v základu nastaven na počet cyklů aerace 2. Pomocí programu bylo vyčteno z databáze 11 několik bajtů dat, aby byl zjištěn současný stav cyklů aerace. To lze vidět na obrázku 3.56, stav byl nastaven tedy na 2 (na obrázku vyznačeno červeně v posloupnosti dat S7).

Dále byl pomocí programu zaslán požadavek na zápis hodnoty na počet cyklů 4 a tento požadavek byl od PLC přijat. Při procesu se změnila počet aerace z dvou na 4 tedy aerace probíhala místo dvou čtyřikrát. Tyto požadavky lze vidět na obrázku 3.57.

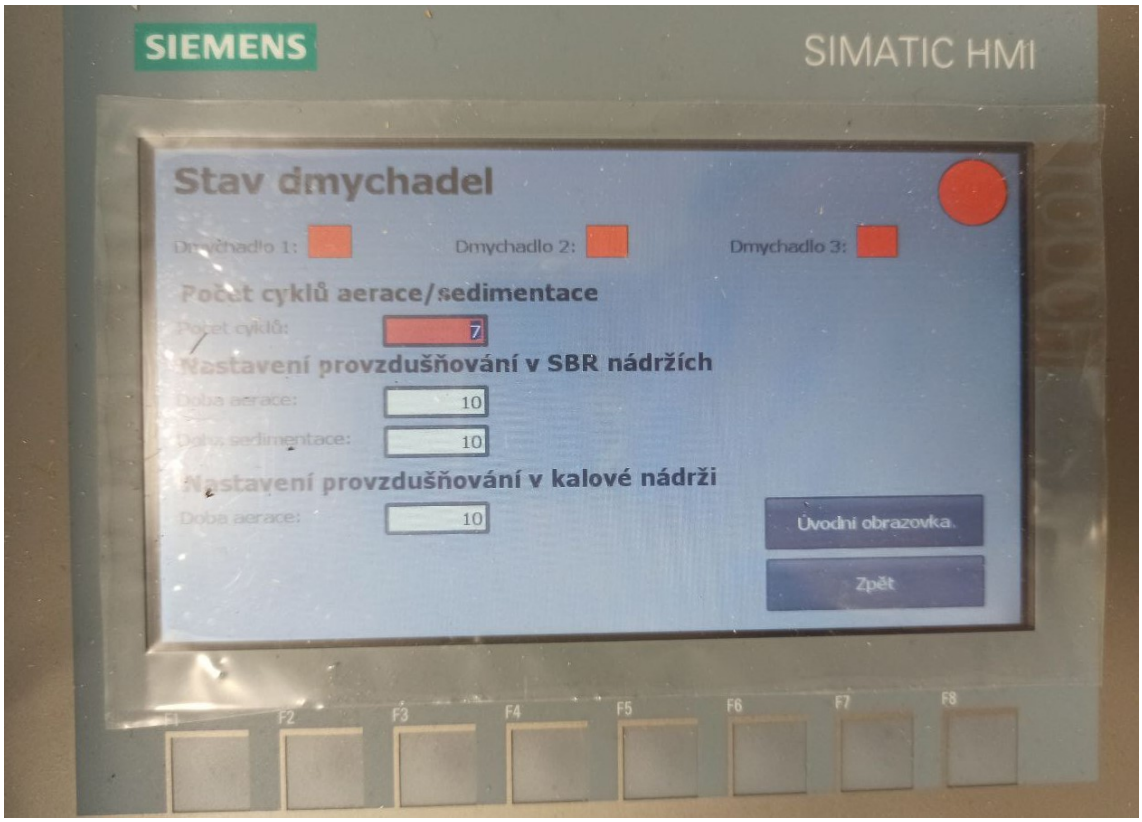


Obr. 3.56: Čtení počtu cyklů ČOV.

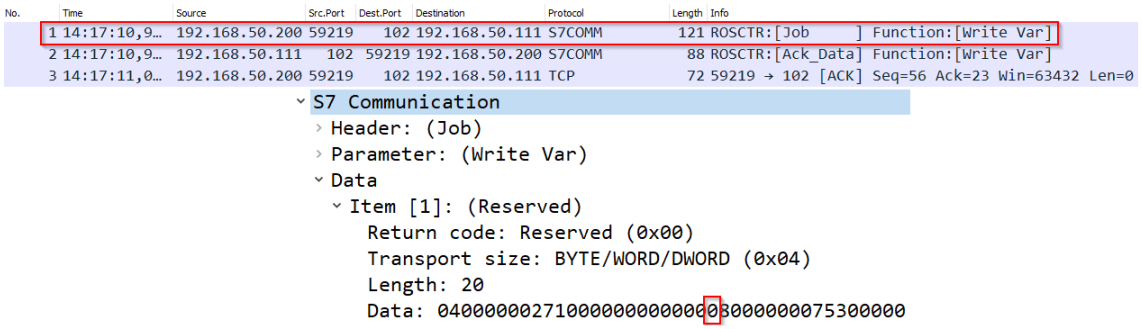


Obr. 3.57: Změna počtu cyklů aerace ze 2 na 4.

Posledním krokem byla demonstrace přetečení stavu. V rámci možnosti nastavení hodnot lze na HMI nastavit pouze cykly v rozmezí 1–5 více nebo méně nelze nastavit. Přepsáním hodnoty v registrech však je možné změnit hodnotu na vyšší například 7 nebo na 0. Při takové změně dojde k chybě na HMI a informativní oblast s hodnotou začne svítit červeně viz obrázek 3.58 kdy byla hodnota změněna na 7 tedy více cyklů než 5. Komunikaci lze vidět na obrázku 3.59 tady byla demonstrována změna komunikace na 0. PLC obě hodnoty přijalo.

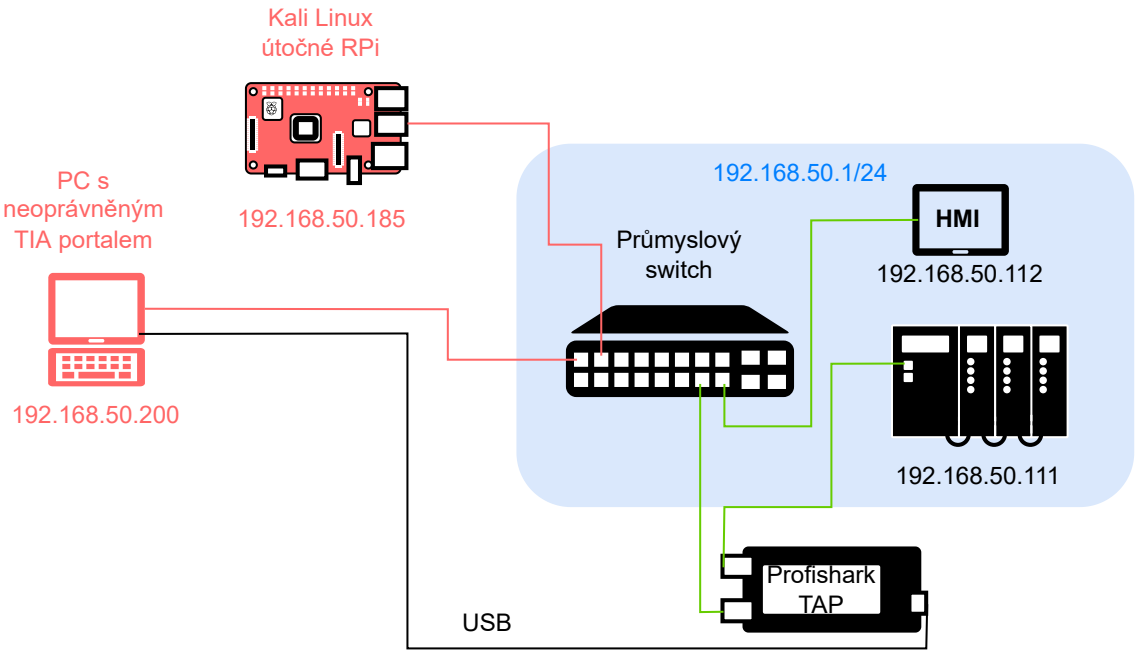


Obr. 3.58: Přetečení počtu cyklů hodnota 7.



Obr. 3.59: Přetečení počtu cyklů hodnota 0.

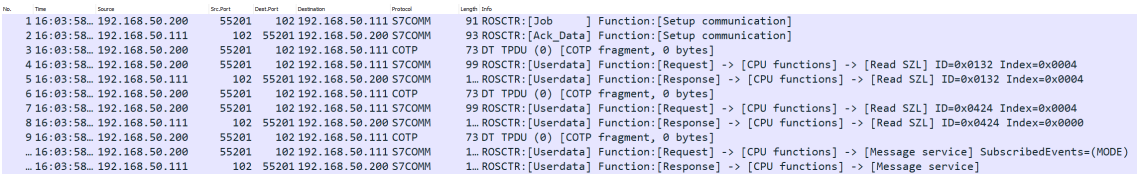
Získávání informací a skenování V rámci této části testování nestandardního chování v průmyslové síti bylo demonstrováno co lze o zařízeních zjistit pomocí externích aplikací, především jak lze využívat TIA portal k získávání informací, a to například k získání programu samotného PLC. Nakonec byly také testovány dopady skenování na průmyslovou síť a možnosti skenování. Schéma sítě je lze vidět na obrázku 3.60. Do scénáře bylo přidáno Raspberry pi, které má operační systém Kali Linux. Toto Raspberry pi bylo do sítě přidáno kvůli možnosti použití aktivního skeneru Nmap. Na útočném počítači s operačním systémem Windows na IP adres 192.168.50.200 byl zprovozněn neoprávněný TIA portál pro možnosti vyčítání informací z PLC s7-300.



Obr. 3.60: Schéma scénářů pro získávání informací a skenování.

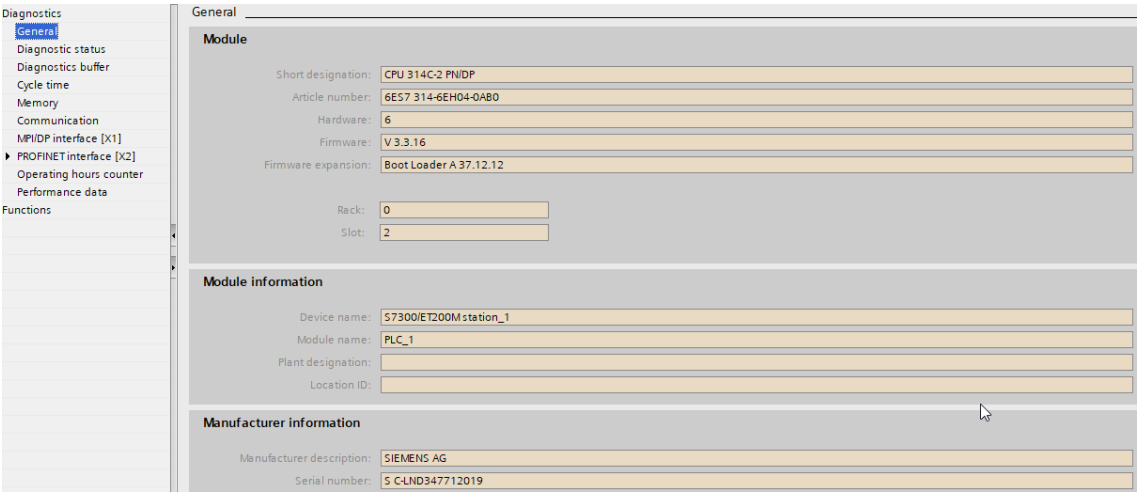
Získávání informací pomocí externích aplikací V rámci získávání informací o průmyslových zařízeních bylo testováno několik nástrojů, prvním bylo testování možností metasploitu, který byl ale užitečný pouze pro získání informací jako sériový model PLC a verze firmwaru, ale i tyto informace mohou být použity k dalším útokům. Pro získání validních informací o zařízení pro následující možné útoky je samotný TIA portal, který je hlavním softwarem pro programování Siemens PLC. Postupovali jsme tedy tak, že na útočné PC s Windows 10 jsme nainstalovali TIA portal v15.1, který je odlišný s TIA portalem, se kterým se PLC u ČOV běžně dorozumívá (ten je na verzi 15, a jedná se o jinou licenci). S tímto neoprávněným TIA portálem se bylo možné pomocí online diagnostiky připojit na PLC S7-300 a vyčíst z něj poměrně zajímavé informace. Už pouze tento zásah do sítě by měl být brán

jako nestandardní chování jelikož může jít o sběr informací právě nějakého útočníka. Ukázka ustanovení této online diagnostiky zachycená pomocí profisharku lze vidět na obrázku 3.61.



Obr. 3.61: Zachycení online diagnostiky a ukázka ustanovení.

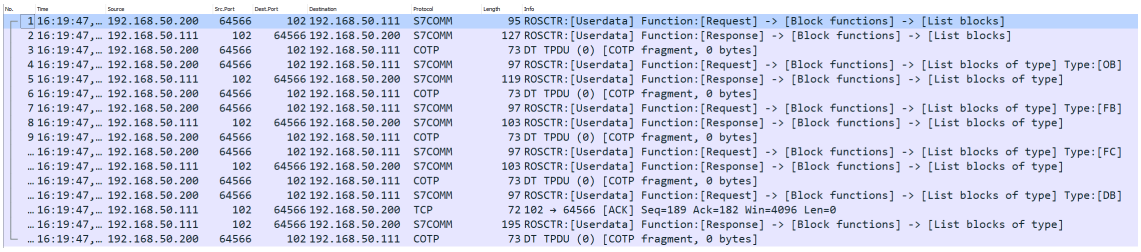
Dále je zde ukázka informací, které byly vyčteny na neoprávněném TIA portalu. Tu lze vidět na obrázku 3.62. Zde je možné vidět verzi firmwaru, výrobní číslo, v rámci jakého racku a slotu je PLC nastaveno – tato informace je velmi důležitá když se útočník chce na PLC připojit například pomocí předem zmíněného skriptu, je vždy třeba nastavit tyto parametry v rámci skriptu. V rámci levého panelu lze vidět další možnosti na které se lze podívat v online diagnostice.



Obr. 3.62: Ukázka online diagnostiky v TIA portalu.

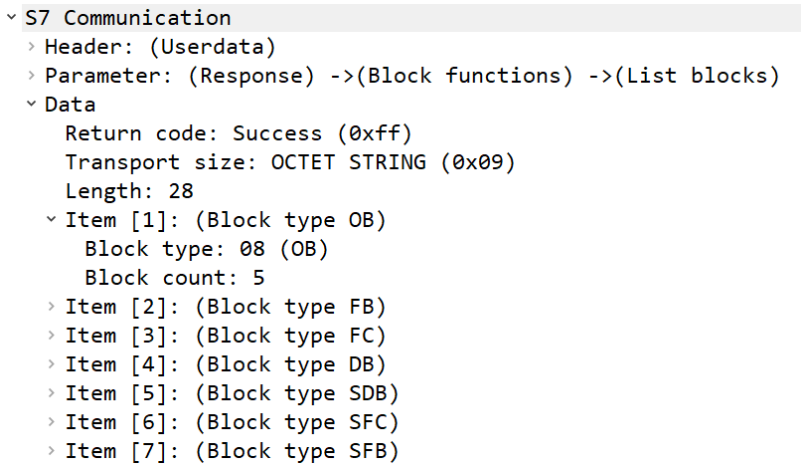
Získání programu z PLC Software TIA Portal také umožňuje získat program z PLC. Tedy tohle může být využito například konkurenční firmou, pokud bude chtít znát řešení určité firmy, může takto získat informace o tom jak je dané PLC naprogramované. V rámci tohoto scénáře byl použit opět TIA Portal verze 15.1 tedy neoprávněný TIA Portal na útočném PC. V rámci něj je potřeba přejít opět do online diagnostiky, vyhledat zařízení a připojit se na něj. Poté je třeba vytvořit nový projekt a označit projekt, pak jen stačí pomocí kolonky online kde lze nalézt tlačítko upload device as a new station a díky tomu lze vyčíst program. Toto vyčtení programu

bylo ve scénáři úspěšné a bylo možno tak vidět celý program pro ČOV testbed. Celá tato komunikace byla zachycena pomoci profishark TAPu a lze tento proces dobře vidět na paketech v rámci komunikace útočného PC (192.168.50.200) a PLC (192.168.50.111). Na obrázku 3.63 lze vidět prvních pár paketů při začátku vyčítání informací o programu. Proti diagnostice kdy bylo vyčítáno [CPU Functions], jsou nyní vyčítány funkce bloků [Block functions]. Pouze pro ukázku na obrázku 3.64. lze vidět data jednoho z paketů kdy se jedná o odpověď na dotaz o počtu a typech bloků v programu (List blocks).



No.	Time	Source	Src.Port	Dest.Port	Destination	Protocol	Length	Info
1	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	S7COMM	95	ROSCTR:[Userdata] Function:[Request] -> [Block functions] -> [List blocks]
2	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	S7COMM	127	ROSCTR:[Userdata] Function:[Response] -> [Block functions] -> [List blocks]
3	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	COTP	73	DT TPDU (0) [COTP fragment, 0 bytes]
4	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	S7COMM	97	ROSCTR:[Userdata] Function:[Request] -> [Block functions] -> [List blocks of type] Type:[OB]
5	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	S7COMM	119	ROSCTR:[Userdata] Function:[Response] -> [Block functions] -> [List blocks of type]
6	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	COTP	73	DT TPDU (0) [COTP fragment, 0 bytes]
7	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	S7COMM	97	ROSCTR:[Userdata] Function:[Request] -> [Block functions] -> [List blocks of type] Type:[FB]
8	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	S7COMM	103	ROSCTR:[Userdata] Function:[Response] -> [Block functions] -> [List blocks of type]
9	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	COTP	73	DT TPDU (0) [COTP fragment, 0 bytes]
…	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	S7COMM	97	ROSCTR:[Userdata] Function:[Request] -> [Block functions] -> [List blocks of type] Type:[FC]
…	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	S7COMM	103	ROSCTR:[Userdata] Function:[Response] -> [Block functions] -> [List blocks of type]
…	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	COTP	73	DT TPDU (0) [COTP fragment, 0 bytes]
…	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	S7COMM	97	ROSCTR:[Userdata] Function:[Request] -> [Block functions] -> [List blocks of type] Type:[DB]
…	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	TCP	72	102 → 64566 [ACK] Seq=189 Ack=182 Win=4096 Len=0
…	16:19:47,…	192.168.50.111	102	64566	192.168.50.200	S7COMM	195	ROSCTR:[Userdata] Function:[Response] -> [Block functions] -> [List blocks of type]
…	16:19:47,…	192.168.50.200	64566	102	192.168.50.111	COTP	73	DT TPDU (0) [COTP fragment, 0 bytes]

Obr. 3.63: Prvních několik paketů při vyčítání programu z PLC.



```
S7 Communication
  Header: (Userdata)
  Parameter: (Response) ->(Block functions) ->(List blocks)
  Data
    Return code: Success (0xff)
    Transport size: OCTET STRING (0x09)
    Length: 28
    Item [1]: (Block type OB)
      Block type: 08 (OB)
      Block count: 5
    Item [2]: (Block type FB)
    Item [3]: (Block type FC)
    Item [4]: (Block type DB)
    Item [5]: (Block type SDB)
    Item [6]: (Block type SFC)
    Item [7]: (Block type SFB)
```

Obr. 3.64: Ukázka paketu s informacemi o blocích v programu.

Skenování v průmyslových sítích a dopady tohoto skenování Na testovacím prostředí byl také testován scénář využití aktivního skenování, a to jmenovitě softwaru Nmap. Aktivní skenování má významný dopad na běh průmyslových sítí. Tato aktivní skenování mohou vyřadit komunikace celé sítě, nebo způsobit zpoždění v komunikaci mezi jednotlivými zařízeními, což může vést k velkému dopadu na výsledný proces. Například v rámci procesu ČOV může toto skenování zpomalit PLC, a tím rozhodit proces čištění, tato zpoždění mohou vést až k fyzickým škodám. Problematiku aktivního skenování je potřeba brát v potaz také vzhledem k její jednoduchosti,

kdy výpadek komunikace může způsobit i nepoučený uživatel, který má přístup do sítě, a aktivní skenování spustí. V rámci skenování bylo využito Raspberry Pi jako útočné zařízení, jak bylo vyznačeno ve schématu. Na tomto zařízení je zprovozněn operační systém Linux na kterém lze spustit aktivní skenovací nástroj Nmap. První skenování bylo zaměřeno na informace o tom zdali je na IP adrese 192.168.50.111 nějaké aktivní zařízení. Příkaz i výsledek skenování lze vidět na obrázku 3.65. Na výsledku skenování lze vidět aktivní zařízení jeho MAC adresu a také že se jedná o zařízení od firmy Siemens. Další skenování bylo pro zjištění otevřených portů, toto skenování lze vidět na obrázku 3.66. Z výsledku skenování lze vidět, že je otevřený port 102 a je na něm aktivní proces iso-tsap – z čehož útočník může usoudit, že se jedná o komunikaci pomocí protokolu S7comm, která komunikuje právě na portu 102 a že se tedy jedná o komunikaci s PLC nebo HMI popřípadě jiným průmyslovým zařízením, které dokáže komunikovat pomocí protokolu S7comm. Jako třetí bylo použito opět skenování portů ale se zaměřením na získání více informací, toto skenování má často velký dopad na průmyslovou síť a občas dochází k výpadkům komunikace PLC. Výsledky skenování lze vidět na obrázku 3.67.

```
utko@utko-desktop:~$ sudo nmap -sP 192.168.50.111
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 15:28 EDT
Nmap scan report for 192.168.50.111
Host is up (0.00037s latency).
MAC Address: AC:64:17:7F:48:4D (Siemens AG)
Nmap done: 1 IP address (1 host up) scanned in 13.23 seconds
```

Obr. 3.65: První skenování – kontrola aktivního zařízení.

```
utko@utko-desktop:~$ sudo nmap -sS -p- 192.168.50.111
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 15:31 EDT
Nmap scan report for 192.168.50.111
Host is up (0.046s latency).
Not shown: 65534 closed ports
PORT    STATE SERVICE
102/tcp open  iso-tsap
MAC Address: AC:64:17:7F:48:4D (Siemens AG)

Nmap done: 1 IP address (1 host up) scanned in 45.05 seconds
```

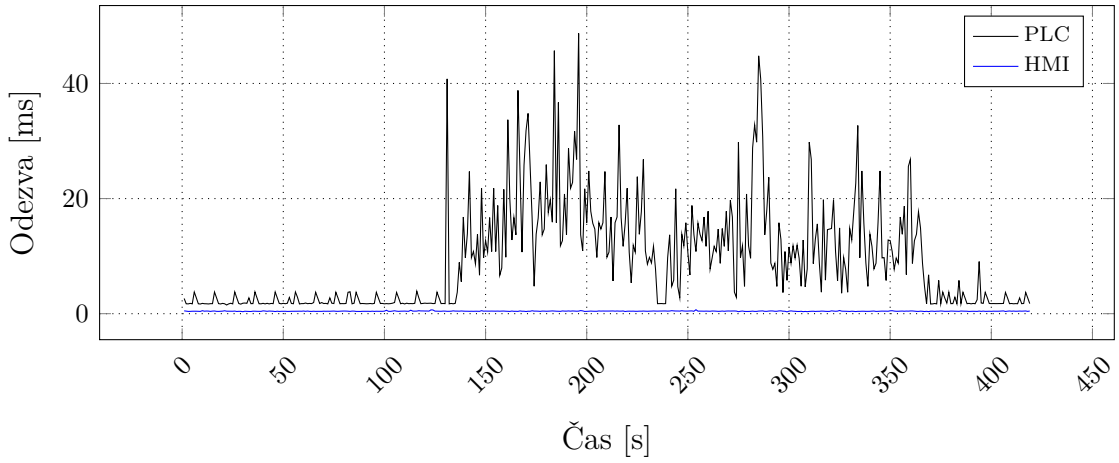
Obr. 3.66: Druhé skenování – kontrola otevřených portů.

Jak již bylo zmíněno aktivní skenování má velký dopad na průmyslová zařízení a je třeba nepoužívat aktivní skenování a vyhýbat se mu pokud je to možné. Na obrázku 3.68 lze vidět dopad skenování otevřených portů na PLC, kdy zpoždění je na průmyslové zařízení poměrně vysoké dosahuje až 40 ms, a to má často významný vliv na průmyslová zařízení.

```
utko@utko-desktop:~$ sudo nmap -sS -p- 192.168.50.111
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 15:31 EDT
Nmap scan report for 192.168.50.111
Host is up (0.046s latency).
Not shown: 65534 closed ports
PORT    STATE SERVICE
102/tcp open  iso-tsap
MAC Address: AC:64:17:7F:48:4D (Siemens AG)

Nmap done: 1 IP address (1 host up) scanned in 45.05 seconds
```

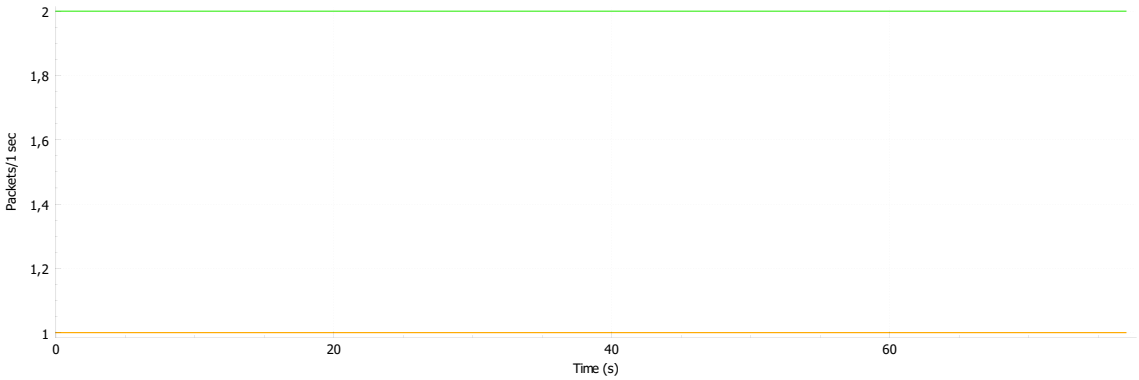
Obr. 3.67: Třetí skenování – kontrola otevřených portů a obdržení více informací.



Obr. 3.68: Vliv skenování otevřených portů na PLC.

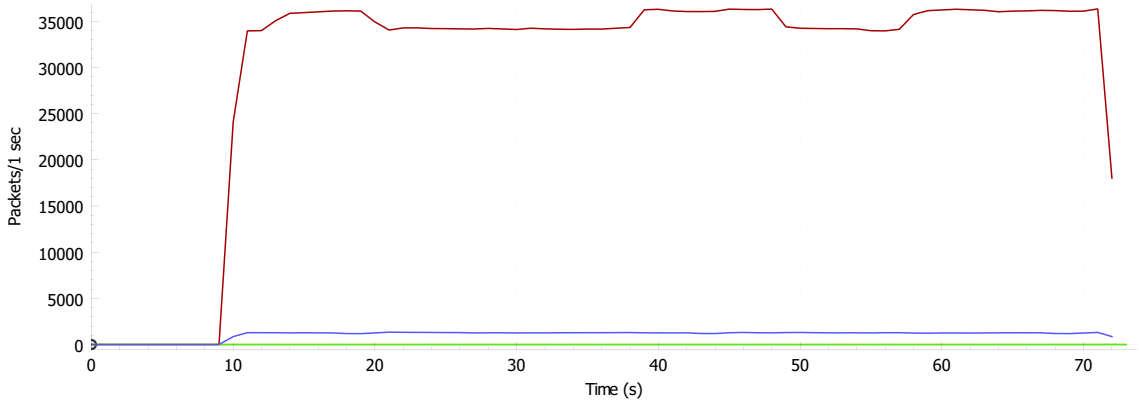
DoS – útoky V rámci testbedu byly také testovány DoS útoky, které jsou jedněmi z nejčastějších útoků, vzhledem k jejich poměrně snadné realizaci. tyto útoky mají za cíl odepřít služby na zařízení to znamená rozhození celého procesu a nemožnost komunikace s dalšími zařízeními v dané síti. Tedy je znemožněna komunikace například mezi PLC a HMI. V rámci scénáře testování bylo opět vycházeno z předchozího schématu. K DoS útokům bylo opět použito zařízení Raspberry Pi s operačním systémem Kali Linux. K útoku bylo použito programu hping3 a příkazu *sudo hping3 -V -c 200000 -d 150 -S -p 120 —flood 192.168.50.111* na PLC a *sudo hping3 -V -c 200000 -d 150 -S -p 120 —flood 192.168.50.112* na HMI. V rámci testovacího scénáře byla sledována komunikace mezi PLC HMI a jaký mají tyto DoS útoky vliv na tuto komunikaci. Nejdříve je zde ukázka minutového provozu kdy je spuštěn pouze program, ale není započat ještě cyklus ČOV (ČOV čeká na přítok vody) jedná se o stav bez DoS útoku. Tento průběh lze vidět na obrázku 3.69. Zelená zde značí zdrojové zařízení HMI které posílá zprávy na cílové zařízení PLC. PLC je zde značeno jako zdroj žlutou, a jedná se o komunikaci kterou zasílá na HMI. Z obrázku lze

vidět, že komunikace není nijak zahlcená, z HMI jsou zasílány v průměru 2 pakety za sekundu na PLC. Z PLC na HMI je posílán v průměru 1 paket za sekundu.



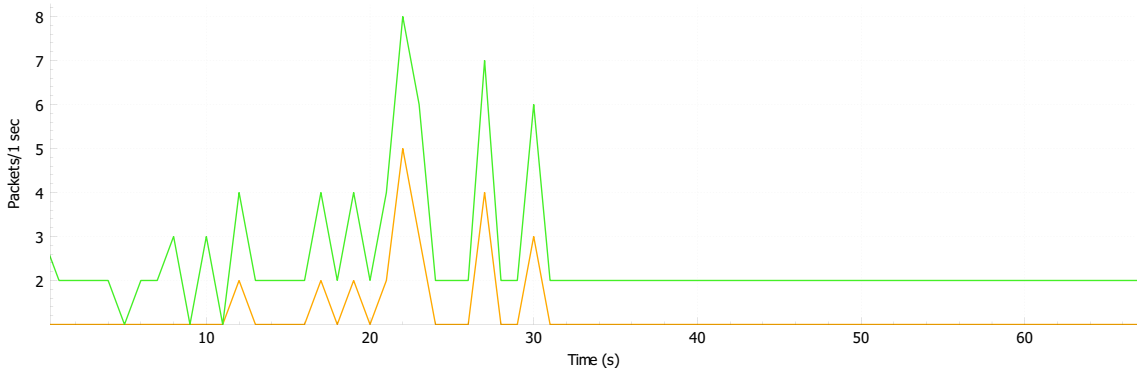
Obr. 3.69: Provoz v komunikaci mezi PLC a HMI během jedné minuty bez DoS útoku.

Dále je testován stejný typ provozu, to znamená pouze spuštěný program ČOV, který ale není v procesu čeká na přítok vody. Zde je již spuštěn DoS útok ze zařízení Raspberry pi. Tento útok lze vidět na obrázku 3.70 a jde opět o provoz veden zhruba jednu minutu. DoS útok byl realizována na PLC a jde tedy vidět, že PLC bylo velmi zahlceno oproti standardnímu provozu. Červeně je zde vyznačena komunikace z útočného zařízení na PLC. Modře je vyznačena komunikace z PLC na útočné zařízení. Zeleně je vyznačena komunikace kde je zdroj HMI a cíl PLC. Při nejvyšším zatížení zvládlo PLC odpovědět rychlostí 1241 paketů za sekundu. HMI nebylo schopno téměř komunikovat s PLC jelikož PLC bylo maximálně vytíženo a tak nebylo možno posílat příkazy pomocí HMI na PLC. Lze také vidět že komunikace žlutá (zdrojem je PLC a cílem HMI) je také na nule a není možné komunikovat ani z PLC na HMI.



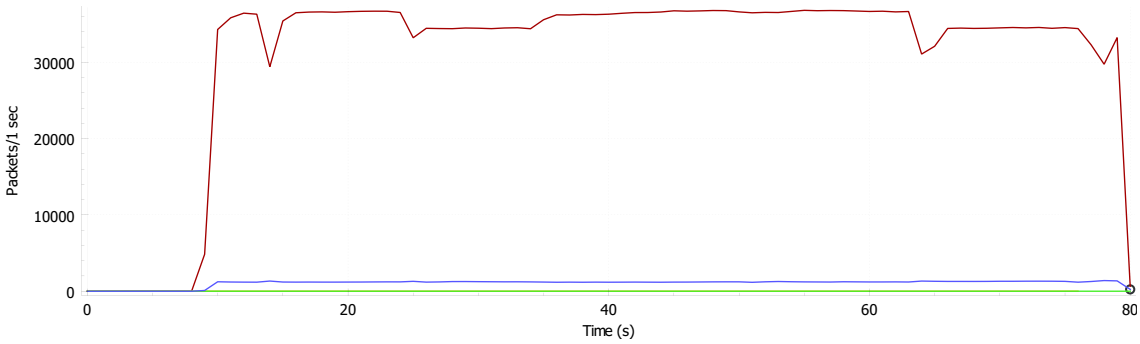
Obr. 3.70: Provoz v komunikaci mezi PLC a HMI během jedné minuty s DoS útokem.

Na obrázku 3.71 lze vidět komunikaci ČOV v procesu a bez DoS útoku, jedná se opět o vyjmutí zhruba jedné minuty z komunikace pro demonstraci. Opět je zelenou vyznačena komunikace zdrojového zařízení HMI. a žlutou zdrojového zařízení PLC. Při procesu je již komunikace častější, především při ustanovení. Bývá zasíláno z HMI až 8 paketů za sekundu. z PLC na HMI pak zhruba 5 paketů za sekundu.

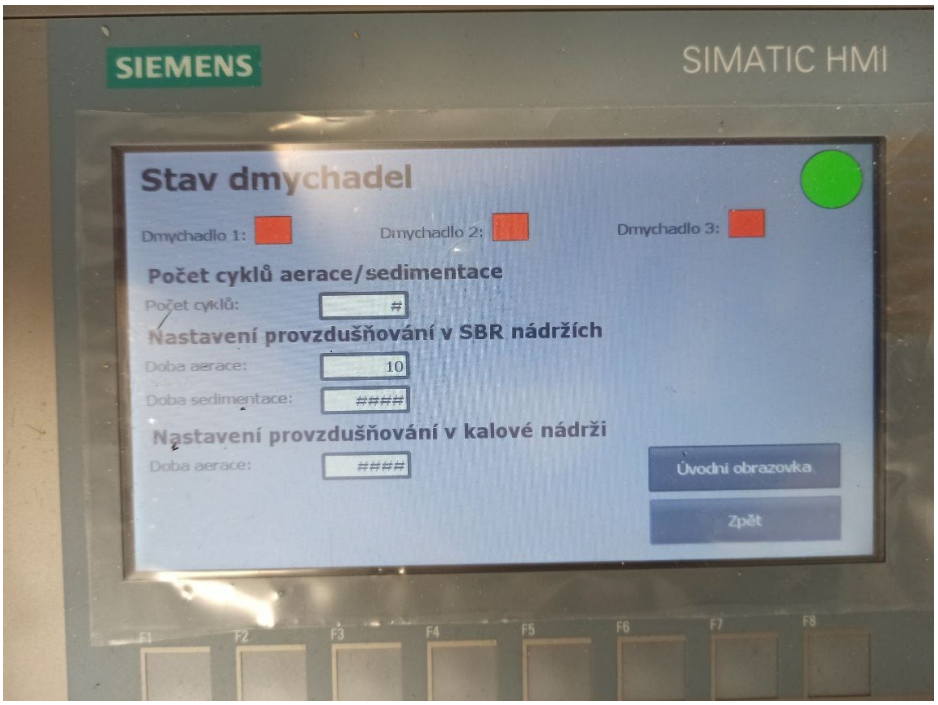


Obr. 3.71: Provoz v komunikaci mezi PLC a HMI během jedné minuty bez DoS útoku ve standardním procesu ČOV.

Dále byl realizován DoS útok na PLC při standardním procesu ČOV. Výsledky jsou vidět na obrázku 3.72 a jsou stejné jako u prvního útoku s tím že maximum paketů které PLC dokázalo odpovědět bylo 1355 za sekundu. Při tomto útoku zase nebylo možné komunikovat mezi PLC a HMI, začali také výpadky na HMI. Tedy tento útok měl vysoký dopad na komunikaci v průmyslové síti a na proces ČOV. V rámci scénářů byl také veden DoS útok přímo na HMI. Kdy v procesu útoku nebylo možné ovládat samotné HMI a také docházelo k chybám na HMI ve vyčítaných informativních boxech tyto chyby lze vidět na obrázku 3.73.



Obr. 3.72: Provoz v komunikaci mezi PLC a HMI během jedné minuty s DoS útokem ve standardním procesu ČOV.



Obr. 3.73: Ukázka chyb na HMI při DoS útoku.

Souhrn dosažených výsledků V tabulce 3.20 lze vidět testované typy nestandardního provozu pro generování dat.

Tab. 3.20: Testované typy nestandardního provozu pro generování dat.

Typ anomálie	**Otestováno**
Útoky pomocí skriptů	Ano
Útoky pomocí externích aplikací	Ano
Získávání informací o zařízení pomocí externích aplikací	Ano
Získání programu PLC	Ano
Skenování v průmyslové síti a dopad skenování	Ano
DoS útoky na průmyslová zařízení	Ano

Kontrola požadavků na testovací prostředí

V tabulce 3.21 je souhrn splněných požadavků pro vývoj testovacího prostředí ČOV. Všechny požadavky pro dané řešení byly splněny.

Tab. 3.21: Požadavky na testovací prostředí.

Požadavky na testovací prostředí	**Zajištěno**	**Poznámka**
Vytvoření fyzického prostředí simulujícího ČOV	Ano	
Komunikace musí obsahovat průmyslový komunikační protokol	Ano	S7comm protokol
Možnost grafické vizualizace	Ano	HMI, OpenMUC (SCADA)
Umístění fyzických komponent HMI a PLC	Ano	HW od firmy Siemens
Možnost testování nestandardního provozu	Ano	Otestovány různé vektory útoků
Virtualizované řešení	Ano	S7comm protokol
Vytváření datasetů	Ano	
Dlouhodobý sběr standardní a nestandardní komunikace	Ano	

3.3 Příklad III – Pivovar

3.3.1 Shrnutí

Za účelem výzkumné činnosti bylo realizováno emulační vývojové prostředí (testbed) pro simulace a testování procesu v potravinářském průmyslu je založeno na konceptu automatizované procesu výroby piva. Koncept se v určitých částech odlišuje od klasických velkých výroben, kde jsou jednotlivé procesy rozloženy do více nádob pro lepší manipulaci s velkými objemy. Vzhledem k limitaci místa na ústavu telekomunikací VUT byl zvolen princip dvou nádob, mezi kterými se kapalina bude v průběhu procesu přečerpávat. Výhodou procesu vaření piva je jeho komplexnost a modulárnost, ale zároveň poměrně jednoduchá možnost analýzy a vektorizace celého procesu. Komplexnost zajišťují různorodé analogové senzory, které generují data teploty, průtoku kapaliny nebo váhy, doplněné binárními ventily a přepínači. Modulárnost je v možnosti úpravy jednotlivých kroků při výrobě piva, jelikož existuje celá řada typů piv, kde základní proces je stejný, ale liší se v nastavení různých parametrů jako jsou teploty, hmotnosti délka dílčího procesu atd. Vektorizaci komunikačních dat zjednodušuje daný recept na typ piva, neměnný základní proces a znalost použitých senzorů, ventilů a přepínačů. Proces výroby piva se skládá z několika dílčích částí – čištění, vaření, kvašení a zrání. Poslední dvě části (kvašení a zrání) jsou časově náročné (dny až týdny), kdy z pohledu automatizace dochází jen hlídání teploty a případného CO_2. Oproti tomu čištění a vaření jsou poměrně komplexní procesy, kdy dochází ke spínání různých prvků a hlídání pomocí senzorů a koncových členů. Čištění je krátký jednoduchý proces, kde jsou však využity skoro všechny komponenty, jelikož je nutné provést proplach přes všechny cesty. Vaření je podstatně komplexnější a časově náročnější proces, který je dále rozdělen na dílčí podprocesy. Z pohledu komunikace je celý proces řízen pomocí centrální PLC jednotky Siemens S7-1500, na kterou jsou připojeny již zmíněné senzory, ventily, přepínače. PLC je přes průmyslový Ethernet a přepínač propojen s HMI jednotkou, ze které lze spouštět/vypínat, nastavovat nebo kontrolovat automatizovaný proces. Dále je součástí částí infrastruktury dohledové centrum SCADA, které mimo funkce HMI jednotky i funkce jako je sběr dat, nebo ukládání dat a logů. Jak již popis napovídá, hlavním cílem realizace výzkumného testbedu, je simulace datové komunikace blížící se reálné situaci z průmyslového potravinářského odvětví. V rámci české republiky je pivovarnictví velmi rozšířené odvětví, kde velké zastoupení má i automatizace, a vzdálené ovládání. S tím samozřejmě vzrůstá i náchylnost systémů na možné kybernetické útoky, s čím je spjat i další cíl realizace. Tím je možnost simulace kybernetických útoků, jejich analýza a realizace protiopatření.

Obr. 3.74: Testbed Pivovar.

3.3.2 Použité komponenty

V níže uvedené tabulce 3.22 jsou popsána fyzická zařízení s jejich parametry a označením v rámci pivovaru.

Tab. 3.22: Seznam zařízení.

Označení	Popis	Fyzické zařízení
VO1	nádoba na zásobu vodu pro celý proces vaření	200l válcová nerezová nádoba o rozměrech 980mm x 530mm
CR1, CR2	čerpadlo na vodu (CR1) a rmutování (CR2)	CR1 – TC CP15-1.5, CR2 – MP-15RM
VN1, VN2	nádoba na proces rmutování a chmelovaru	Válcová nerezová nádoba o rozměrech 500mm x 500mm a objemu 100l
PR1, PR2	průtokoměry pro sledování množství napuštěné vody	průtokoměr YF-S201
VV1, VV2	NC ventily pro ovládání průtoku vody	El.mag. ventil VPCS22014-15DC24V, -20-130 °C, DD 24 V, NC, 0-10Bar
OD1	Odsávání pro rmutování a chmelovar	Odsávací ventilátor TT-125 s aktivním filtrem
MR1	motor pro zdvih při rmutování a chmelovaru	ZY1016Z – DC 12 V, 200 W, 350 ot./min, převodový poměr 1:9,8
KC1-KC2	koncové spínače pro limitaci stavu nádob během vaření	ME-8108, 1x NC, 1x NO, AC380V/6A, AC250V/10A, IP65
KC3-KC5	Vertikální snímač hladiny pro limitaci stavu kapalin během vaření	0 °C až 120 °C, max. tlak 1,6 MPa, 220V/0,5A
TC1-TC3	senzory pro sledování teploty během procesu vaření	senzor PT100, -20 až +450 °C
VS1	váhový senzor pro rmutovací/chmelovar nádobu	GUANG CE YZC-1B, 200kg ±0,02%, DC 10-15V
ML1	míchadlo pro chmelovar	míchací nerezová lopatková hlavice
CL1	chlazení po chmelovaru	CW-3000, 10 l/min, 50 W/°C
KN1	kvasná nádoba	100l válcová nerezová nádoba o rozměrech 680 mm x 455 mm
VP1-VP7	nc ventil pro ovládání průtoku pivního extraktu během procesu vaření	El.mag. ventil SMS1MF13E4D16, -20 až 120 °C, DC 24 V, NC, 0-10Bar
OH1-OH10	Tištěné topná tělesa pro rmutovací/chmelovar nádobu	230V/1000W
RS1	Siemens SCALANCE XB005	Sítový směrovač pro datovou komunikaci
PLC1	Siemens S7-1500 CPU 1512C-1 PN	Programovatelný logický kontrolér pro automatizované ovládání pivovaru
HMI1	Siemens SCALANCE KTP700	Zobrazovací a ovládací jednotka pivovaru.
SRV1	SCADA OpenMUC	SCADA server pro dohled, záznam a vzdálené ovládání pivovaru.
RPi1	Raspbery Pi zero W	Jednodeskový počítač komunikující protokole Modbus pro přenos měřených teplotních hodnot.

TC CP15-1.5 Cirkulační čerpadlo na pitnou vodu z napájení 230 V a maximálním příkonem 28 W. Provozní tlak kapaliny je až 10 bar s teplotním intervalem od 2 až do 95 °C. Maximální průtok kapaliny je 12 l/min s výtlakem až 1,3 metrů.

MP-15RM Magnetické nerezové čerpadlo na potravinářské kapaliny s napájení 230 V a maximálním příkonem 25 W. Pracovní teplota protékající kapaliny může být až 100 °C s krátkodobými výkyvy až do 120 °C. Maximální průtok kapaliny je 19 l/min s výtlakem až 4,3 metrů.

YF-201 Průtokoměr vody s možností měření průtoku od 1 až do 30 l/min pomocí Hallovy sondy, kde se v závislosti na průtoku mění frekvence na výstupu čím vyšší je frekvence tím vyšší je průtok kapaliny. Výstupní napětí je 5 V TTL, charakteristikou 450 pulzů na 1 litr protékající kapaliny a přesností 2%. Zařízení se je možno napájet napětím od 5 do 18 V DC. Provozní teplota okolí i protékajícího média je v rozsahu -25 až 80 °C. Připojení je možné pomocí ventilu G1/2".

SMS1MF13E4D16, VPCS22014-15DC24V Solenoidové ventily pro automatizační techniku plynných a kapalných médii v průmyslovém odvětví. Ventily jsou v poloze zavřeno (NC) bez přívodu napájení. Napájecí napětí ventilů je DC 24 V. Tělo ventilu, které je v kontaktu s médiem je z nerezové oceli AISI 304 s těsnění z nitril-kaučuk EPDM, a plastovou ochranou elektrických části IP65. Ventil je možno používat s pracovním tlakem od 0 MPa až po 1 MPa. Při používání by okolní teplota neměla přesáhnout interval -10 až 50 °C a teplota média by měla být v rozsahu -20 až 120 °C. Připojení ventilu je pomocí závitu G1/2"s průtokem až 3600 l/h.

TT-125 Odtahový/odsávací ventilátor s průchozím průměrem 125 mm a kompatibilitou se standardním komponenty vzduchotechniky o stejném průměru. Průtok je při plném výkonu až 280 000 l/h s maximální teplotou média 60 °C. Ventilátor je napájen AC 230 V a chráněn IPX4 proti stříkající vodě.

CW-3000 Průmyslové termolýzní chladící zařízení pro udržení teploty vody na nastavenou úroveň. Cirkulace vody je pomocí čerpadla s průtokem 15 l/min a zásobníkem na 9 litrů. Chladící výkon zařízení je 50 W/°C. Napájecí napětí je 230 V s výkonem přibližně 100 W.

KC1, KC2 (ME-8108) Nastavitelný průmyslový koncový spínač s voděodolným pouzdrem.

Tištěná topná tělesa Tištěná topná tělesa slouží pro kontaktní ohřev rovných ploch. Tělesa disponují rychlým náběhem teplot, snadnou montáží a možností vysokého zatížení. Napájecí napětí těles je 230 V a maximální výkon jednoho tělesa je 1000 W.

GUANG CE YZC-161 Tenzometrický senzor pro měření zatížení do 50 kg s přesností 0,02 %. Zařízení je možné bezpečně přetížit až do 120 % s deformačním přetížením až 150 %. Zařízení je možno napájet napětím od 10 do 15 V. Pracovní teplota je v rozsahu -35 až 80 °C. Přístroj splňuje krytí IP65, což zajišťuje ochranu proti tryskající vodě.

Rasbperry Pi Zero W Jednodeskový počítač Raspberry Pi Zero W (RPi Zero) je zmenšená verze populárních minipočítačů Raspberry Pi, které primárně slouží pro výukové a vývojové účely. V posledních letech se značně rozšířili do průmyslového odvětví díky své modularitě. Hlavní výhodou oproti běžnému minipočítači jsou vstupní a výstupní GPIO piny, na které je možné připojit různé senzory a ovládací prvky a tím např. automatizovat vybraný proces. Softwarová část RPi umožňuje chod různým typům operačních systémů, kde mezi nejběžnější patří Raspbian, který je odvozenou verzí od známého linuxového operačního systému Debian. V rámci testbedu jsou na GPIO piny RPi Zero připojena dva senzory teploty DS18B20 a pomocí protokolu Modbus jsou snímané hodnoty teploty přenášeny do PLC1 jednotky.

DS18B20 Senzor DS18B20 snímá teplotu v rozsahu 55 až 125 °C, přičemž v rozsahu -10 až 85 °C má garantovanou přesnost ±0,5 °C. Nevýhodou tohoto senzoru je poměrně dlouhá doba (cca 0,7 sekund) převodu snímané teploty na digitální 12-bitové slovo, které je dále posíláno pomocí OneWire sběrnice. Výhodou je, že senzor je samotně použitelným pro velkou řadu zařízení a nepotřebuje žádný převodník jako např. odporové senzory teploty. Každý DS18B20 má jedinečný 64bitový sériový kód, který umožňuje více DS18B20 fungovat na stejné 1-drátové sběrnici.

Siemens Simatic S7-1500 Programově logický automat (PLC1) od společnosti Siemens s CPU 1512C-1 PN. 5x Analogový vstup, 2x Analogový výstup, 32 Digitálních vstupů, 32 Digitálních výstupů. Jedná se o centrální jednotku pro řízení celého testbedu. Signálová komunikace mezi koncovými prvky (ventily, čerpadla atd.) a PLC1 probíhá pomocí zmíněných vstupů a výstupů. Síťová komunikace je realizována pomocí ethernetového rozhraní a protokolu Siemens S7comm a Modbus TCP.

Siemens Simatic HMI KTP700 Průmyslová zobrazovací a ovládací jednotka KTP700 (HMI1) od společnosti Siemens slouží k ovládání celého testbedu. Síťová komunikace probíhá pomocí ethernetového rozhraní a protokolu Siemens S7comm mezi HMI1 a PLC1.

Siemens Scalace XB005 Průmyslový ethernetový přepínač s přenosovou rychlostí až 100 Mbit/s.

3.3.3 Vstupní kritéria, předpoklady, vývoj a návrh

Vstupní kritéria a předpoklady

Detekce a analýza komunikačních dat je z velké míry závislá na zaznamenaných datech reálné komunikace. Struktura protokolu, jeho komunikační schéma a vzor komunikujícího zařízení je možné získat z dokumentace a virtuální simulací obrazu zařízení. Reálná komunikace je však ovlivněna dalšími faktory, které jsou buď těžko předvídatelné, nebo složitě simulovatelné. Také samotná simulace nestandardních stavů (útoky, nedostupnost zařízení, výpadky energie atd.) je v průmyslové datové komunikaci prakticky nerealizovatelné, jelikož by znamenala snížení produktivity, a tedy i finanční ztrátu pro podnik.

Z tohoto důvodu jsou realizovány vývojová testovací prostředí (testbed) blížící se reálným scénářům, na kterých jsou tyto nestandardní stavy simulovány. V tomto dokumentu popsaný fyzický emulátor pivovaru je jedním z nich. V testbedu není cílem simulovat celý proces velkých pivovarů, ale různých stavů, které se při výrobě vyskytují. Díky automatizaci pomocí programovatelných logických automatů (PLC) je možné tyto stavy zpracovávat a přenášet pomocí síťových protokolů do ovládacích jednotek. Analýza této komunikace umožňuje vytvářet vzory, které slouží klasifikaci standardních i nestandardních stavů a procesů. Z výše uvedeného textu lze vyvodit kritéria a předpoklady, které by dané testovací prostředí mělo splňovat, aby adekvátně simulovalo reálné provoz. Předpoklady pro fyzickou realizaci testovacího prostředí.

- Automatizační řídicí jednotka (**PLC**) komunikující průmyslovým protokolem.
  - V rámci průmyslu jsou jedněmi z nejvyužívanějšími aplikační protokoly – **ModbusTCP**, **Profinet**, **OPC UA**, **EthernetIP** a **S7comm**.
- Další síťové prvky vyskytující se průmyslové komunikaci.
  - Pro lokální ovládání zařízení a vizualizaci se v průmyslu využívají rozhraní člověk-stroj (**HMI** – Human Machine Interface), které je k PLC připojeno buď přes sériovou komunikaci nebo pomocí jednoho z výše uvedených síťových protokolů. Pro vzdálený přístup a dohled slouží rozhraní

SCADA (Supervisory Control And Data Acquisition), které je stejně jako HMI připojeno pomocí jednoho ze síťových protokolů.

- Rozhraní a stanice pro záznam a analýzu síťového provozu.
- Fyzické prvky jako jsou přepínače, ventily, senzory atd. nacházející se v reálném procesu pro výrobu piva.
- Úložiště pro dlouhodobý sběr dat.

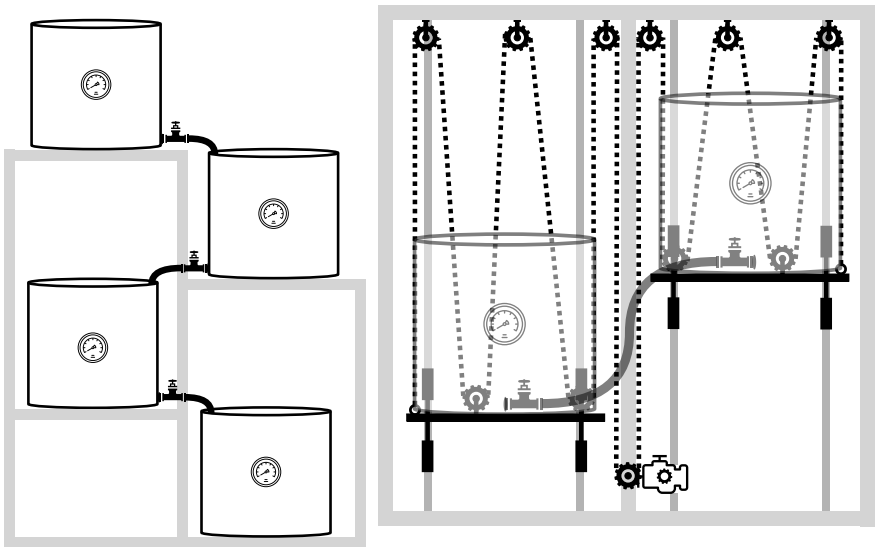
Požadavky a předpoklady na datovou část testbedu:

- Záznam standardní i nestandardní komunikace.
- Analýza komunikace a klasifikace standardní komunikace i nestandardní.

Návrh konstrukce

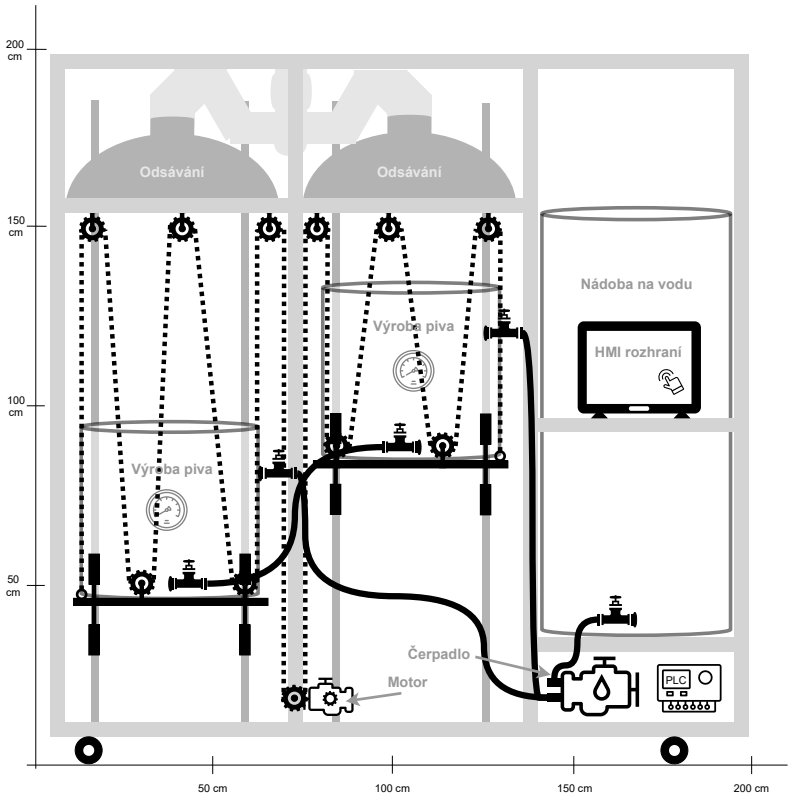
Při vývoji testbedu Pivovar byl využit koncept spádového mechanismu. Ten se využívá v malých (mikro, nano) nebo domácích pivovarech. Během procesu výroby piva je použita řada nádob pro jednotlivé procesy výroby. Mezi těmito nádobami je daný produkt procesu přečerpáván většinou pomocí čerpadel, na které jsou kladeny vysoké nároky ať už pohledu hygienického, potravinářského nebo technologického. Z tohoto důvodu jsou čerpadla finančně nákladná. U malých a domácích pivovarů se většinou využívá gravitačních sil pro přečerpání mezi nádobami, které se umisťují v kaskádovém uspořádání, kde celý proces začíná v nejvyšší nádobě a postupně se přepouští do nižších. Tento typ spádového pivovaru má nevýhodu, že na každou část procesu je potřeba další nádoba, což v např. v části rmutování stylem dekokce (produkce se několikrát dělí, zpracovává odděleně, a následně znovu slučuje) značně navyšuje počet nádob. Princip běžného spádového systému je zobrazen na obrázku 3.75 v levé části. Druhou variantou je využití kladkového systému s motorem, který by měnil vzájemnou polohu nádob vůči sobě a tím umožňoval přepouštění. Celkově tak dojde k úspoře počtu nádob. Nevýhodou je komplexnější systém. Princip kladkového spádového systému je zobrazen na obrázku 3.75 v pravé části.

Pro samotný návrh testbedu byl zvolen komplexnější systém kladkového spádového pivovaru. V návrhu se předpokládalo využití tří nádob, kde dvě budou sloužit pro výrobu celého procesu a třetí bude zásobníkem vody, aby byla zajištěna větší soběstačnost. Celý návrh je zobrazen na obrázku 3.76. V horní části je systém, který bude zajišťovat odsávání par a filtraci pachů uvolňujících se při vaření piva. Dále je v návrhu čerpadlo na napouštění vody do nádob na výrobu piva. I zde mohlo být využito spádového systému, ale čerpadla na čistou vodu jsou řádově levnější než potravinářská kalová čerpadla pro vysoké teploty, které by se muselo použít pro přečerpání mezi nádobami pro výrobu piva. Také samotné napouštění vody by se tím komplikovalo. Objem nádoby na vodu musí být vzhledem k objemu vařeného piva minimálně 2x větší, aby voda stačila na celý proces. K tomu by byl potřeba



Obr. 3.75: Princip spádového systému.

dražší a výkonnější motor, a komplexnější systém zdvihu, který by ještě prodlužoval již tak poměrně dlouhý proces vaření. Ovládání celého procesu bude zajišťovat PLC jednotka z HMI rozhraním pro kontrolu dohled na výrobou.

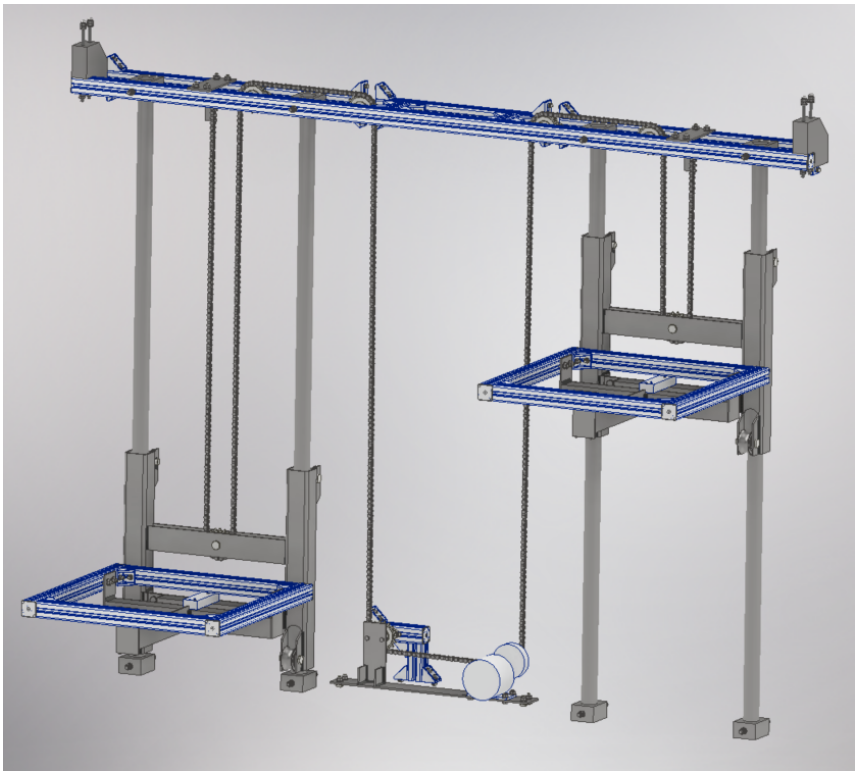


Obr. 3.76: Finální návrh fyzického testbedu pivovar.

Navržená konstrukce zobrazená na obrázku 3.77, je složena ze dvou hlavních částí: zdvižný mechanismus a nosná konstrukce. Důraz byl kladen na nosnost mechanismu, kterou musí konstrukce unést. Celá konstrukce pivovaru je složena z hliníkových profilů, plechů a jinak tvarovaných ocelových profilů.

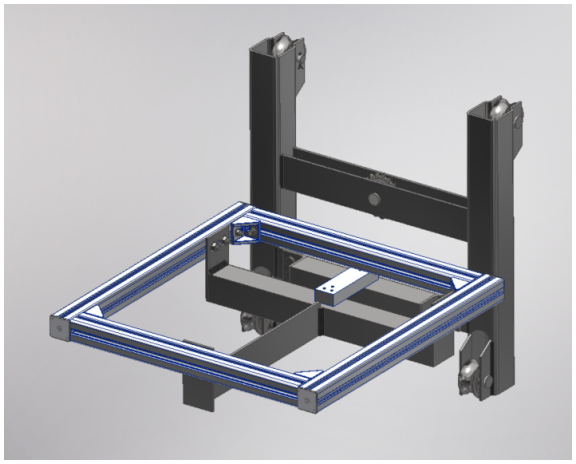
Obr. 3.77: Kompletní sestava.

Zdvižný mechanismus Cílem návrhu bylo vytvoření mechanismu, který by byl schopen pohybovat nádobami vždy v opačném směru pohybu jedné části vůči druhé. Dalším požadavkem bylo umístění celého zdvižného mechanismu do spodní části rámu s tím, že mechanismus nesměl zasahovat nad varné nádoby, kde budou umístěny další přístroje pro chod pivovaru. Poslední požadavek byl zaměřen na nosnost samotných pojezdů, která byla dimenzována na dvě stolitrové nádoby. Koncept zdvižného mechanismu byl založen na kladkovém mechanismu, kdy za pomoci kladek se protichůdně zvedají obě nádoby. Pohyb nádob je zajištěn motorem, ve spodní části rámu, díky efektu přerozdělování délky řetězu. Oba konce řetězu jsou přes tenzometry připevněny k rámu. Z výše uvedených kritérií, byl sestaven finální návrh pro realizaci, viz obrázek 3.78. Návrh je složen z několika částí: zdvižné plošiny, držáku pohonu, pomocné konstrukce, pojezdů a řetězového systému.



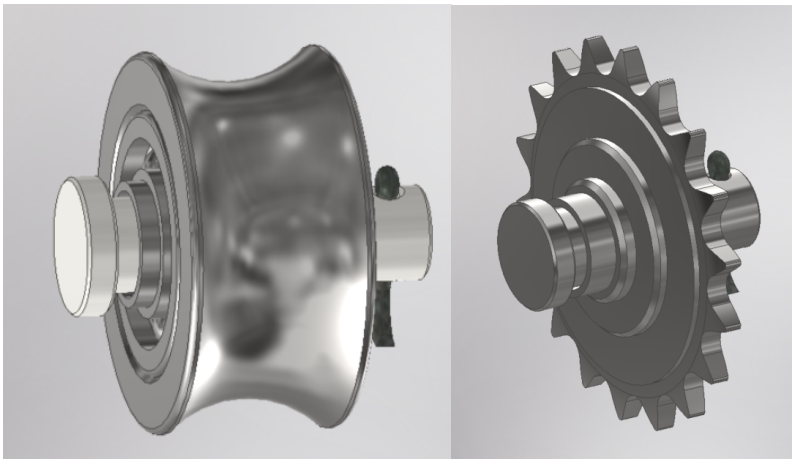
Obr. 3.78: Kompletní návrh zdvižného mechanismu.

Zdvižná plošina Zdvižná plošina tvoří hlavní část zdvižného mechanismu, která slouží pro umístění nádob a jejímu posunu (viz obrázek 3.79). Je složena z několika hlavních částí a pojízdných komponentů. Hlavní komponenty sestavy tvoří tři svařence: hlavní úchytný svařenec, spojovací úchyt a rozpěrku. Sestava je doplněna hliníkovým rámečkem, ze kterého je vyrobena hlavní konstrukce, na které je ustanovena jedna z nádob.



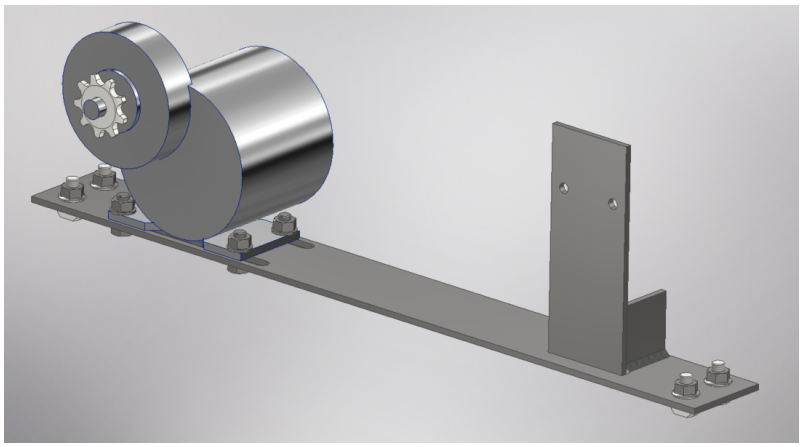
Obr. 3.79: Zdvižná plošina.

Do souboru pojízdných komponentů se řadí soustavy rolen (obrázek 3.80) a řetězová kladka (obrázek 3.80). Každá ze soustav rolen je složená z čepu, na které se nachází ložisko vsunuté do těla rolny. V soustavě se dále nachází rozpěrný kroužek stabilizující polohu čepu, a samotný čep je proti mechanickému výsunu zajištěn závlačkou. Řetězová kladka slouží k zajištění pohybu plošiny za pomoci řetězu. Soustava řetězové kladky je složena z čepu, rozpěrných kroužků, napínacího ozubeného kola a závlačky.



Obr. 3.80: Pojízdné komponenty.

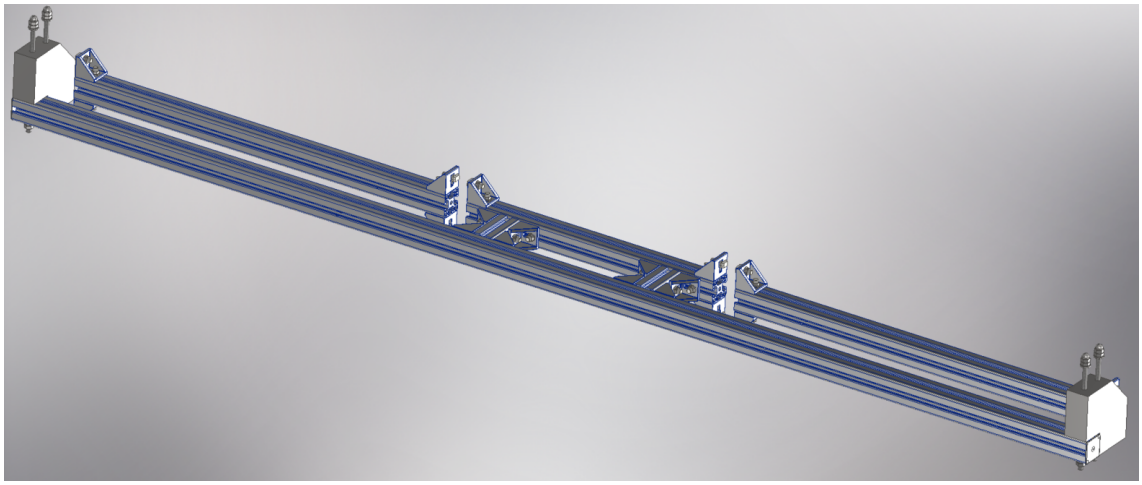
Držák pohonu Sestava je určena pro úchyt motoru nesoucí jedno ozubené kolo k nosné konstrukci. Je složena ze svařence a pohonného motoru, viz obrázek 3.81. Svařenec zajišťuje fixní polohu v konstrukci, jak pro motor, tak i pro samotný řetězový systém. Motor pro posun plošin byl zvolen stejnosměrný elektromotor s příkonem 12 V, výkonem 200 W a převodovkou pro pomalejší chod soustavy.



Obr. 3.81: Držák pohonu.

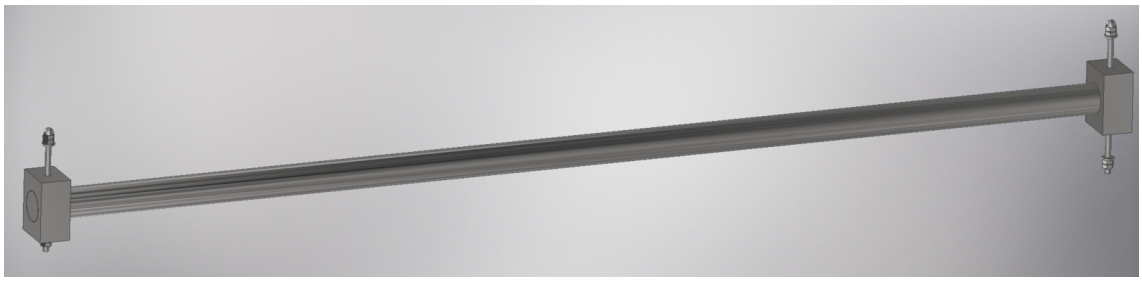
Pomocná konstrukce

Nosná konstrukce nedovolovala svými rozměry splnit požadavky pro uchycení pojezdového systému. Z tohoto důvodu byla vytvořena pomocná konstrukce, která byla přidána k rámu nosné konstrukce. Ta zajišťuje uchycení pojezdů, soustav ozubených kol, úchytů řetězů a přerozdělení zátěže konstrukce, viz obrázek 3.82. Je složena z hliníkových profilů a dvou kovových bloků. Jednotlivé profily slouží, jak pro uchycení pojezdů, tak pro uchycení řetězového systému. Uchycení k rámu je provedeno šroubovými spoji v kovových blocích a za pomoci speciálních rohových spojek.



Obr. 3.82: Pomocná konstrukce.

Pojezdy Soustava slouží ke stanovení rozmezí pohybu plošin v konstrukci a ustálení možných bočních rázů. Soustava je složena z kruhové tyče, která je na koncích uchycena do bloků, které tuto soustavu skrze dlouhé šrouby upevňují k rámu konstrukce (viz obrázek 3.83).

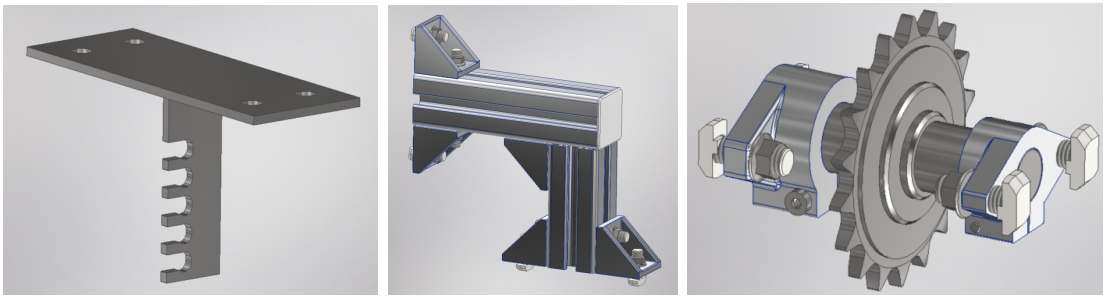


Obr. 3.83: Pojezd.

Řetězový systém Systém slouží k pohybu zdvižných plošin, které jsou taženy po pojezdech, které zajišťují stabilitu veškerých pohyblivých součástí. Systém je složen z řetězu, svařence a pojízdných komponent. Celý řetězový systém je zobrazen na obrázku 3.84).

Obr. 3.84: Řetězový systém.

Vytvořené svařence slouží k ukotvení konce řetězu, viz obrázek 3.85. Složeny jsou ze dvou plechů speciálního tvaru, který odpovídá přesným polohám čepů řetězu. Ve spodní části systému se nachází pomocné rameno (viz obrázek 3.85) z hliníkových profilů, která napomáhá k uložení jedné z pěti soustav řetězových kol. Jedna soustava řetězového kola (viz obrázek 3.85) se skládá ze dvou úchytů ke konstrukci, malé hřídele, rozpěrného kroužku a napínacího řetězového kola.



Obr. 3.85: Součásti řetězového systému, zleva – držák řetězu, pomocné rameno a soustava řetězového kola.

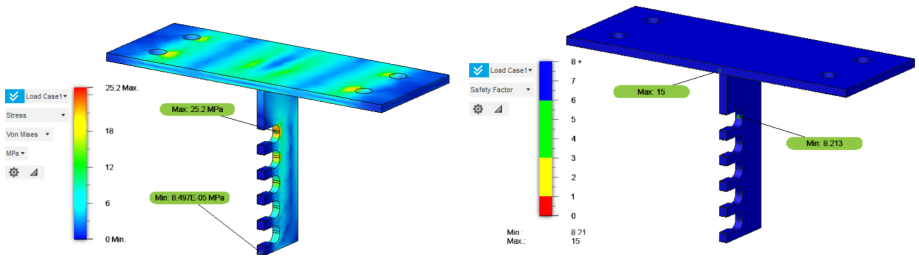
Zátěžové testování návrhu

Celá sestava byla vytvořena ve vývojovém prostředí Autodesk Inventor a otestována programem Fusion. Mechanismus byl navržen na maximální nosnost měřících senzoru hmotnosti (2 000 N). Mezi nejvíce namáhané součásti patřili nosné součástky, konkrétně hřídele soustav ozubených kol, pojezdové plošiny, držáky řetězu a pomocná konstrukce. Výsledky zátěžového testování jsou zobrazeny v tabulce 3.23.

Tab. 3.23: Výsledky zátěžových testů.

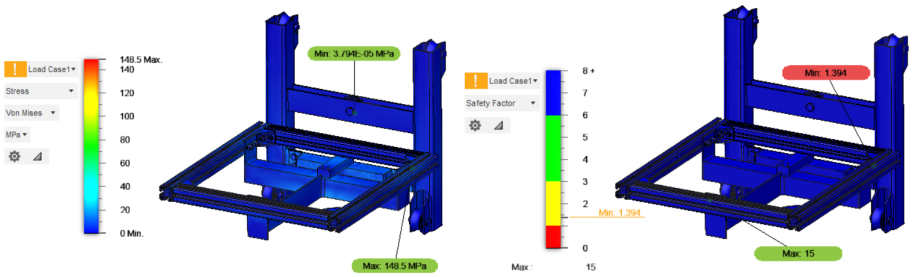
Sestava	Zátěž [N]	Maximální napětí [MPa]	Koeficient bezpečnosti
Držák řetězu	1 250	25,20	8,213
Zdvižná plošina	2 500	148,50	1,394
Soustava řetězového kola	1 250	15,03	13,770
Pomocná konstrukce	5 000	56,43	3,688

Držák řetězu Součást byla otestována s vyhovujícím výsledkem. Výsledné hodnoty simulace jsou zobrazeny na obrázku 3.86. Při maximálním možném zatížení 1 250 N dosáhl koeficient bezpečnosti nejnižší hodnoty 8,213 (obrázek 3.86). Koeficient odpovídá i zatížení v zaoblení nejvyššího výřezu plechu, které dosahuje svého maxima 25,2 MPa (obrázek 3.86).



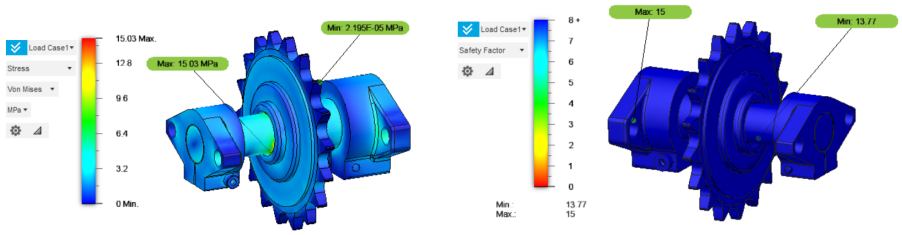
Obr. 3.86: Výsledek simulace držáku řetězu – zleva statické namáhání a koeficient bezpečnosti.

Zdvižná plošina Součást byla otestována s vyhovujícím výsledkem, viz obrázek 3.87. Únosnost plošiny je dána nosností měřícího senzoru (tenzometr), jehož maximální únosnost činí 2 000 N. Hmotnost zdvižné plošiny, která odpovídá součtu hmotnosti nosné konstrukce, nádoby a příslušenství, činí 50 kg (500 N). Pro zatížení 2 500 N, které odpovídá maximálnímu možnému zatížení, dosáhl koeficient bezpečnosti nejnižší hodnoty 1,394 (obrázek 3.87). Koeficient odpovídá i namáhání šroubového spoje v daném místě, které dosahuje svého maxima 148,5 MPa (obrázek 3.87).



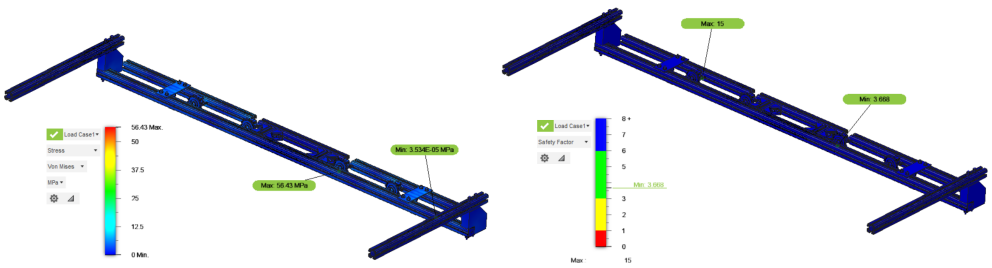
Obr. 3.87: Simulace plošiny – zleva statické namáhání a koeficient bezpečnosti.

Soustava řetězového kola Součást byla otestována s vyhovujícím výsledkem, viz obrázek 3.88. Test byl proveden při maximální možné tahové síle působící na řetězové kolo, tedy 1 250 N. Toto zatížení působí oboustranně na obou koncích. Koeficient bezpečnosti při maximální možné zátěži dosáhl nejnižší hodnoty 13,77 (obrázek 3.88). Koeficient odpovídá i namáhání v tlaku v daném místě, které dosahuje svého maxima 15,03 MPa (obrázek 3.88).



Obr. 3.88: Simulace řetězového kola – zleva statické namáhání a koeficient bezpečnosti.

Pomocná konstrukce Součást byla otestována s vyhovujícím výsledkem, viz obrázek 3.89. Součást byla testována pro maximální nosnost 5 000 N. Koeficient bezpečnosti dosáhl nejnižší hodnoty 3,668 (obrázek 3.89). Koeficient odpovídá i namáhání tlaku daného místa, které dosahuje maxima 56,43 MPa (obrázek 3.89).

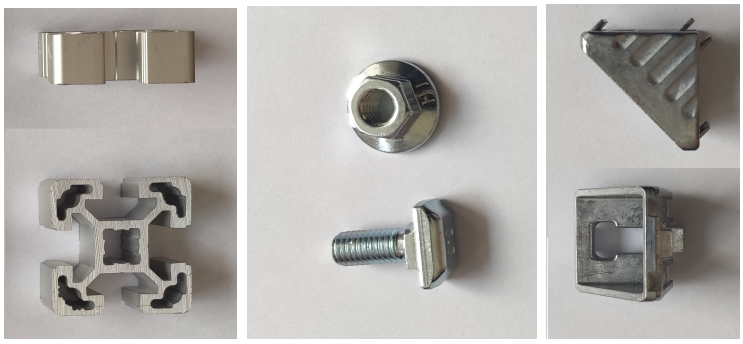


Obr. 3.89: Simulace konstrukce – zleva statické namáhání a koeficient bezpečnosti.

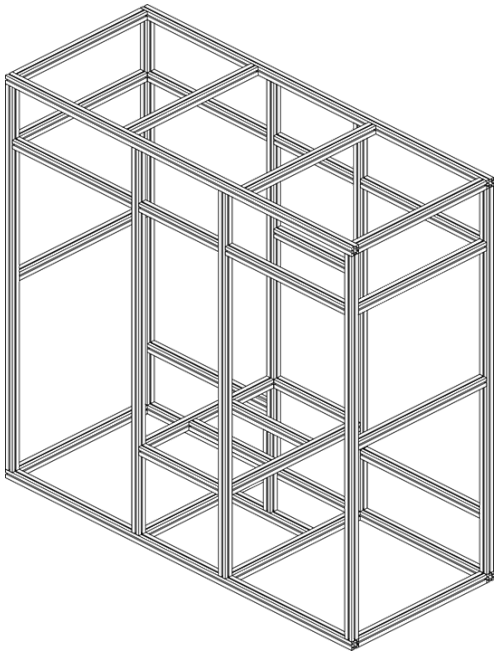
Realizace konstrukce

Realizace konstrukce včetně pohybového mechanismu byla rozdělena do několika částí: realizace nosné konstrukce, nákup a úprava materiálu pro zdvižný mechanismus a finální kompletace.

Realizace nosné konstrukce Základní konstrukce bez zdvižného mechanismu je postavena z hliníkových čtvercových profilu o průměru 40 mm x 40 mm, který je uveden na obrázku 3.90, spojovacího materiálu (viz obrázek 3.90) a spojovacích rohů (viz obrázek 3.90). Celá konstrukce je uvedena na obrázku 3.91 a na obrázku 3.92.



Obr. 3.90: Konstrukční materiál – zleva hliníkový profil, spojovací materiál a roh.



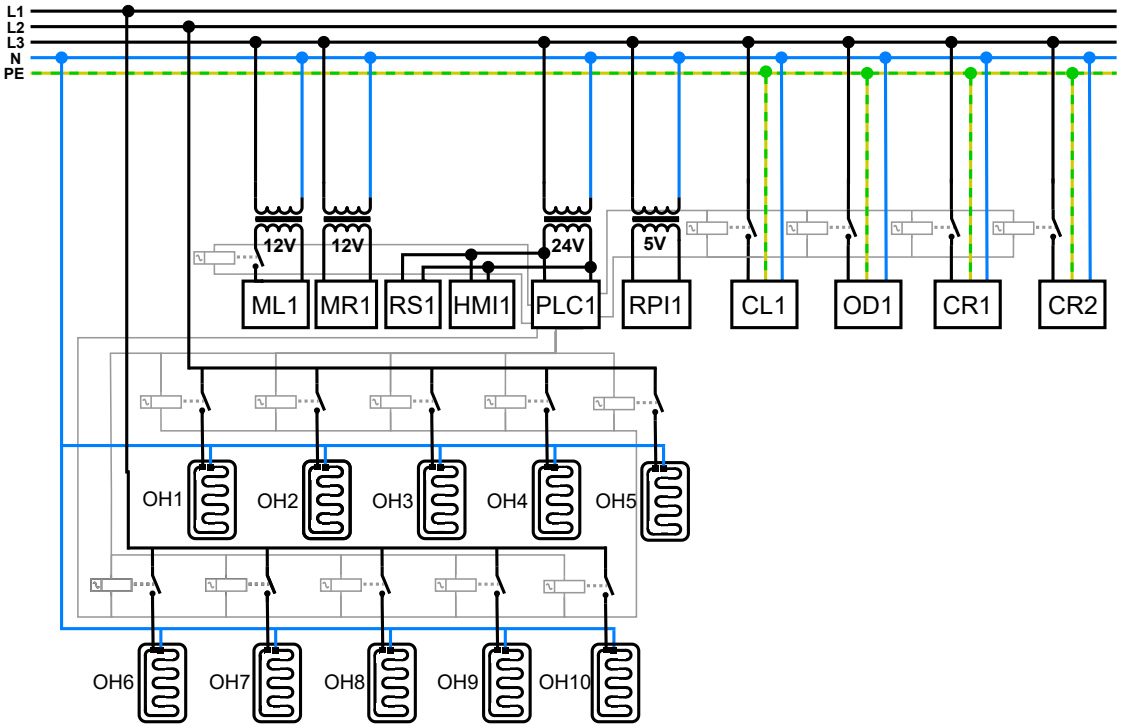
Obr. 3.91: Realizovaná konstrukce pivovaru z hliníkových profilů.

Obr. 3.92: Ukázka výsledné sestavy.

3.3.4 Technický popis

Elektrické schéma napájení

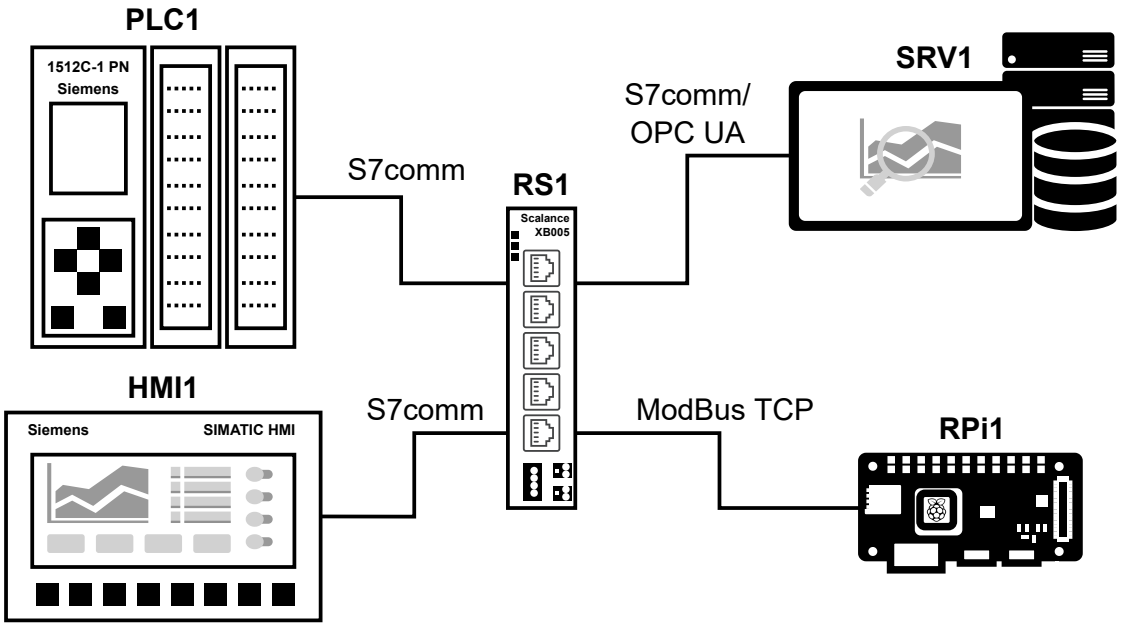
Napájení vývojového testbedu zajišťuje třífázová přípojka 400V/16A. Fáze L1 a L2 slouží pro připojení tištěných topných desek (OH1-OH10), které ohřívají nádoby VN1 a VN2. Poslední fáze L3 napájí zbylá zařízení na přímo nebo přes některý z transformátorů. Některé prvky mají také přeřazený relé modul řízený z PLC jednotky, aby je bylo možné spínat v požadované fázi procesu výroby.



Obr. 3.93: Elektrické schéma napájení prvků v testbedu.

Komunikační schéma

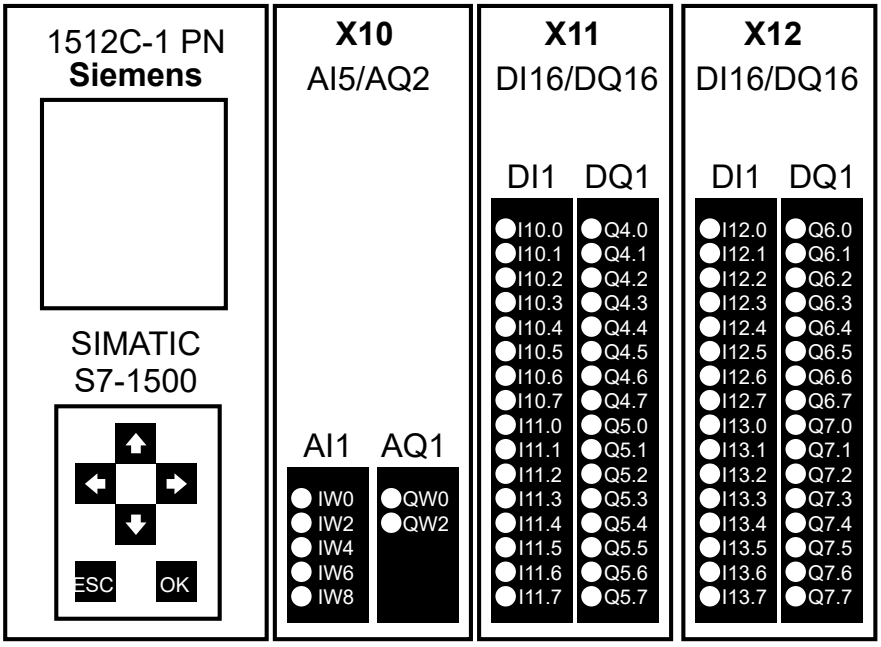
Komunikace v rámci testbedu je rozdělena na datovou a signálovou. Datová komunikace probíhá mezi zařízení PLC, HMI1, SRV1 a RPi1 propojené pomocí průmyslového Ethernetu přes síťový přepínač RS1 (viz obrázek 3.94). Data jsou přenášena pomocí tří hlavních protokolů. První protokol S7comm zajišťuje výměnu dat mezi zařízeními PLC1, HMI1 a SCADA (SRV1) OpenMUC. Tento protokol je nativně používán u průmyslových zařízení společnosti Siemens, pod kterou spadají i obě zařízení. Druhým protokolem je Modbus TCP, který je implementován do jednodeskového počítače Raspberry Pi Zero W (RPi1). Na tento počítač jsou připojeny dva senzory teploty TC1 a TC2. Pomocí protokolu Modbus jsou následně přenášeny do PLC1 a SVR1. Třetí protokol OPC UA slouží pro komunikaci se vzdáleným dohledovým centrem SCADA OpenMUC. Tento protokol byl zvolen pro jeho podpory ze strany PLC1, poměrně velkému zastoupení na trhu a modulárnosti z pohledu objektové strukturalizace dat. Signálová komunikace je zprostředkována mezi koncovými prvky (senzory, ventily, přepínači) a vstupně výstupním rozhraním PLC1 jednotky. Jejichž bližší popis je uveden v následující podkapitole 3.3.4.



Obr. 3.94: Obecné komunikační schéma.

Řídicí jednotka (PLC1)

Řídicí jednotka celého testbedu je realizovaná pomocí Siemens SIMATIC S7-1500, která je zobrazena na obrázku 3.95 se všemi vstupy a výstupy. Jednotka má 5 analogových vstupů (X10-AI1), 2 analogové výstupy (X10-AQ1), 32 digitálních vstupů (X11-DI1, X12-DI1) a 32 digitálních výstupů (X11-DQ1, X12-DQ1). Na tyto vstupy jsou pak připojeny prvky – senzory, ventily, přepínače uvedené níže. U každého je uvedena adresa v rámci PLC1, která odpovídá adresám na obrázku 3.95 a celý souhrn je uveden v tabulce 3.24.



Obr. 3.95: Vstupy a výstupy řídicí jednotky Siemens SIMATIC S7-1500.

Tab. 3.24: I/O adresy v rámci S7-1500 s připojenými zařízeními.

Digitální výstupy	
PLC adresa	**Zařízení**
Q4.0	OH1
Q4.1	OH2
Q4.2	OH3
Q4.3	OH4
Q4.4	OH5
Q4.5	OH6
Q4.6	OH7
Q4.7	OH8
Q5.0	OH9
Q5.1	OH10
Q5.2	VV1
Q5.3	VV2
Q5.4	VP1

Digitální výstupy	
PLC adresa	**Zařízení**
Q5.5	VP2
Q5.6	VP3
Q6.7	VP4
Q6.0	VP5
Q6.1	VP6
Q6.2	VP7
Q6.3	CR1
Q6.4	CR2
Q6.5	MR1_R
Q6.6	MR1_L
Q6.7	OD1
Q7.0	ML1
Q7.1	CL1

Analogové vstupy	
PLC adresa	**Zařízení**
IW0	-
IW2	-
IW4	VS1
IW6	VS2
IW8	-

Digitální vstupy	
PLC adresa	**Zařízení**
I10.0	KC1
I10.1	KC2
I10.2	PR1
I10.3	-
I10.4	-

VS1, VS2 (GUANG CE YZC-161) Tenzometrické senzory jsou pasivní elektrotechnické součástky, které se používají k měření mechanického napětí na povrchu tělesa. Změnou délky tenzometru dochází ke změně odporu. V případě zapojení čtyř tenzorů do Wheatstonova můstků je změna odporu převedena na napětí. Analogové napětí přímo připojeno na vstupy IW4 (VS1) a IW6 (VS2), kde jsou zpracovány PLC1 jednotkou.

DS18B20 (Alternativa k PT100) Senzor DS18B20 je samostatný snímač teploty, který je možné připojit na širokou škálu zařízení. V rámci komunikace je senzor připojen na GPIO piny Raspberry Pi, kde jsou snímaná digitální data zpracovávána python programem a pomocí ethernetového rozhraní a protokolu Modbus přeposílána do PLC1 S7-1500.

PR1 (YF-201) Senzorová část průtokoměru YF-201 je založena na Hallově jevu, který se používá pro měření magnetických polí. Uvnitř průtokoměru je umístěna turbína, kterou otáčí protékající voda. Na lopatce turbíny je umístěn magnet, který je při průchodu snímán pomocí Hallovy sondy. Samotné měření průtoku je detekováním pulzů, které jsou přenášeny datovým pinem, přičemž charakteristika pulzů je 450 pulzů na litr kapaliny. Datový pin je připojen na vstupní rozhraní I10.2 PLC1 jednotky. Napájení průtokoměru zajišťuje transformátor 12 V.

OH1-OH10 Tištěná topná tělesa jsou napájena střídavým napětím 230 V. Pro každé těleso (OH1-OH10) je spínání provedeno přes relé jednotku, která je ovládána z výstupní rozhraní Q4.0 až Q5.1 PLC1 jednotky.

VP1, VP7 (SMS1MF13E4D16), VV1, VV2 (VPCS22014-15DC24V) Ventily VV1, VV2 a VP1 až VP7 jsou NC ventily, kdy přívodem napětí 24 V jsou přepnuty do otevřeného stavu. Ventily jsou připojeny na digitální výstupní Q5.2 až Q6.2 rozhraní PLC1. Oba typy ventilů jsou napájeny ze spínaného zdroje DC 24 V, který je společný s PLC1 jednotkou.

CR1, CR2 Čerpadla jsou napájena napětím 230 V a jejich ovládání je provedeno přes relé jednotku, která je iniciována z výstupního rozhraní PLC1 jednotky. CR1 je připojeno na pin Q6.3 a CR2 na pin Q6.4.

MR1_R, MR1_L Motor pro zdvihu pivovaru je napájen 12 V a řízen přes kontrol, který určuje směr otáčení motoru. Přepínač směru je vyveden na PLC1 výstupní rozhraní Q6.5 pro otáčení motoru vpravo a Q6.6 pro otáčení motoru vlevo.

OD1 (TT-125) Odsávací jednotka TT-125 je lopatkový ventilátor s napájecím napětím 230 V. Jednotka je napojena přímo na přívod elektrického napájení přes relé jednotku, která je ovládána z výstupní rozhraní Q6.7 PLC1 jednotky.

ML1 Míchadlo je napájeno 12V transformátorem, který je přes relé kontakt připojen na výstupní rozhraní Q7.2 PLC1 jednotky.

CL1 (CW-3000) Chladící jednotka CW-3000 je napájena střídavým napětím 230 V. Ovládání je provedeno přes relé jednotku, která je iniciována z výstupní rozhraní Q7.1 PLC1 jednotky.

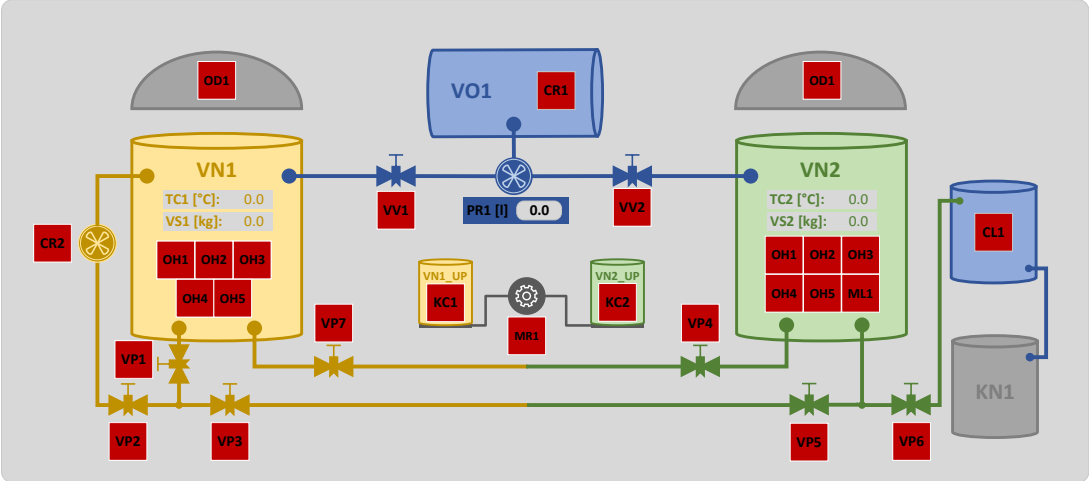
Vizualizace rozhraní SCADA OpenMUC (SRV1)

Vizualizace grafického rozhraní dohledového centra OpenMUC SCADA (SRV1) je realizováno dvěma obrazovkami. První slouží jako úvodní obrazovka (viz obrázek 3.96), kde je možné spouštět a ukončovat celý proces.

Obr. 3.96: Grafický panel ve SCADA OpenMUC – Úvodní obrazovka.

Druhý panel vizualizuje veškeré prvky v rámci testbedu – senzory, přepínače a ventily připojené na fyzickou část pivovaru. Komunikaci dohledového centra a řídicí jednotky PLC1 zajišťuje protokol OPC UA. Zobrazení, které je viditelné operátorovi je uvedeno na obrázku 3.97. Hlavními prvky jsou nádoby VN1, VN2, které slouží k samotnému procesu vaření, doplněné nádobou na vodu VO1. Na tyto nádoby jsou připojeny čerpadla CR1 (čerpání vody do VN1 a VN2), CR2 (přečerpávání ve VN1) a ventily VV1, VV2 (napouštění vody) a VP1 až VP7 (kontrola průtoku mezi VN1 a VN2). Jelikož se jedná o kladkový spádový pivovar, tak motor MR1 zajišťuje

změnu poloh nádob s koncovými spínači KC1 a KC2, které signalizují hraniční stav VN1 a VN2. V každé nádobě jsou uvedena snímaná data ze senzorů (TC1, TC2 – teplota, VS1, VS2 – váha), stav topných těles (OH1-OH10), koncový spínač proti přetečení vody (KC3, KC4, KC5) a časomíra (TVN1, TVN2) pro měření cyklů při procesu vaření. Posledními prvky jsou průtokoměr (PR1) pro měření přečerpané vody ze zásobníku VO1, odsávání nečistot a vlhkosti při vařené (OD1), chlazení po ukončení vaření (CL1) a kvasná nádoba (KN1) pro uložení vařeného produktu.



Obr. 3.97: Grafický panel ve SCADA OpenMUC – Hlavní obrazovka.

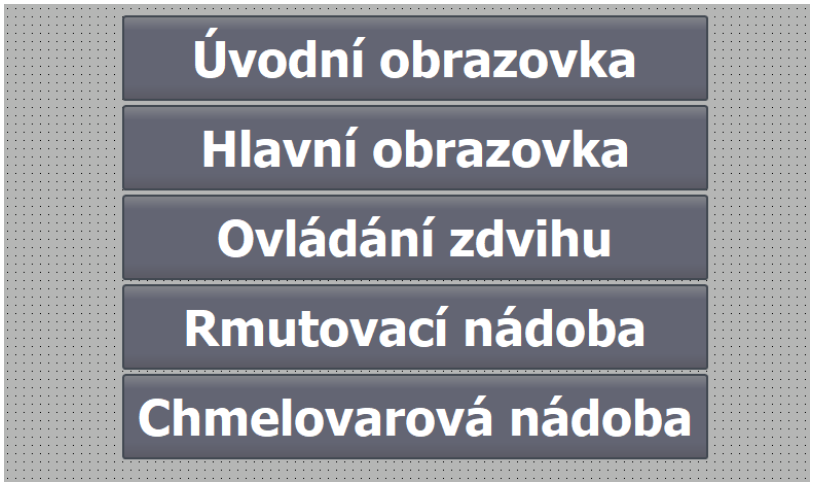
Vizualizace grafického HMI panelu (HMI1)

Ovládací rozhraní člověk-stroj (HMI) pro testbed Pivovar zajišťuje vizualizační jednotka SIMATIC KTP-700 BASIC od společnosti Siemens. Komunikaci mezi HMI1 jednotkou a PLC1 zajišťuje ethernetový kabel a protokol S7comm. Oproti dohledovému centru je grafické rozhraní u HMI jednotky rozděleno na více částí/obrazovek pro jednodušší a detailnější ovládání celého mechanismu.

Úvodní obrazovka První obrazovka, která je zobrazena po spuštění testbedu hlavním vypínačem. Obrazovka obsahuje tlačítka START a STOP pro spuštění, tlačítko menu pro přechod mezi dalšími obrazovkami a čas od spuštěni procesu vaření piva. Vyobrazení je uvedeno na obrázku 3.98.

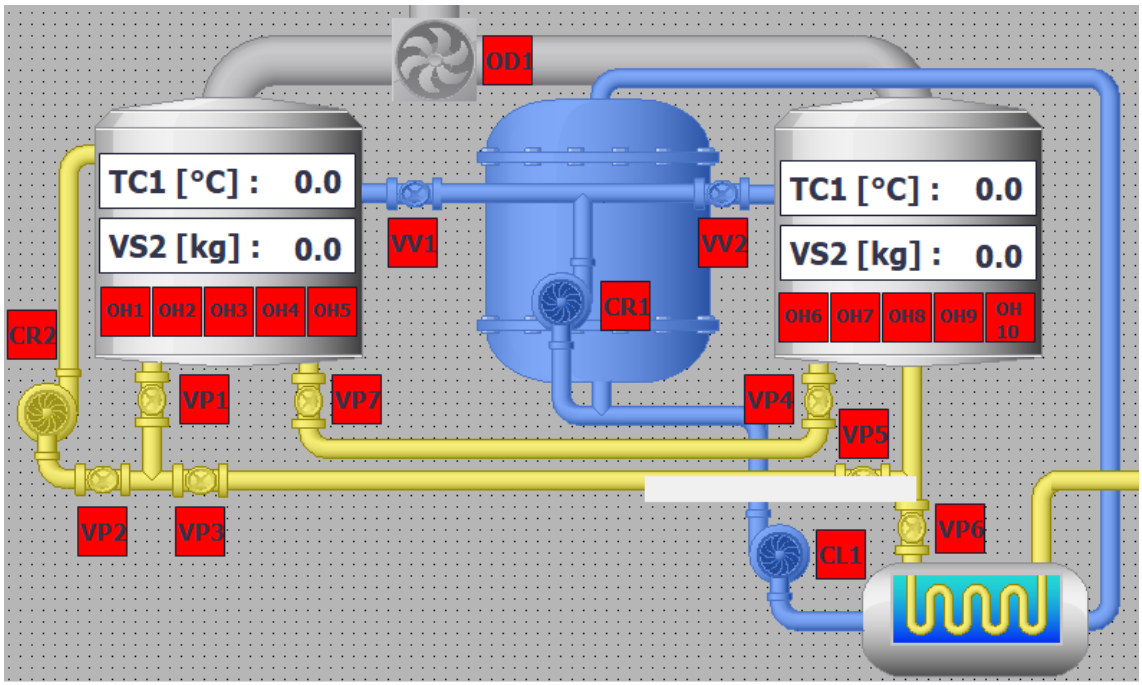
Obr. 3.98: Grafický panel HMI jednotky část – Úvodní obrazovka.

Menu Jak již bylo uvedeno, tak tato část grafického rozhraní, slouží jako rozcestník mezi jednotlivými obrazovkami (viz obrázek 3.99). Jsou zde čtyři možnosti (Hlavní obrazovka, Rmutovací nádoba, Chmelovarová nádoba, Zdvihový pohon) pro detailnější ovládání pivovaru a poslední možnost k návratu na úvodní obrazovku.



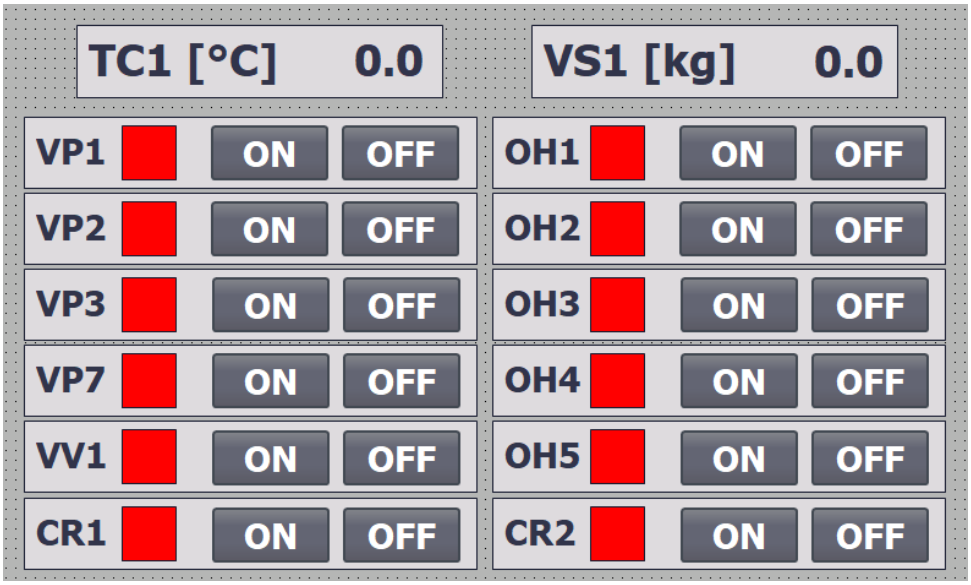
Obr. 3.99: Grafický panel HMI jednotky část – Menu.

Hlavní obrazovka Jak již název napovídá, tak jde o hlavní obrazovku, která slouží jako primární dohledová jednotka. Na této obrazovce jsou zobrazeny stavy všech fyzických prvků, které jsou připojeny do testbedu. Zde je tedy možné kontrolovat průběh celého procesu a zjišťovat případné poruchy.



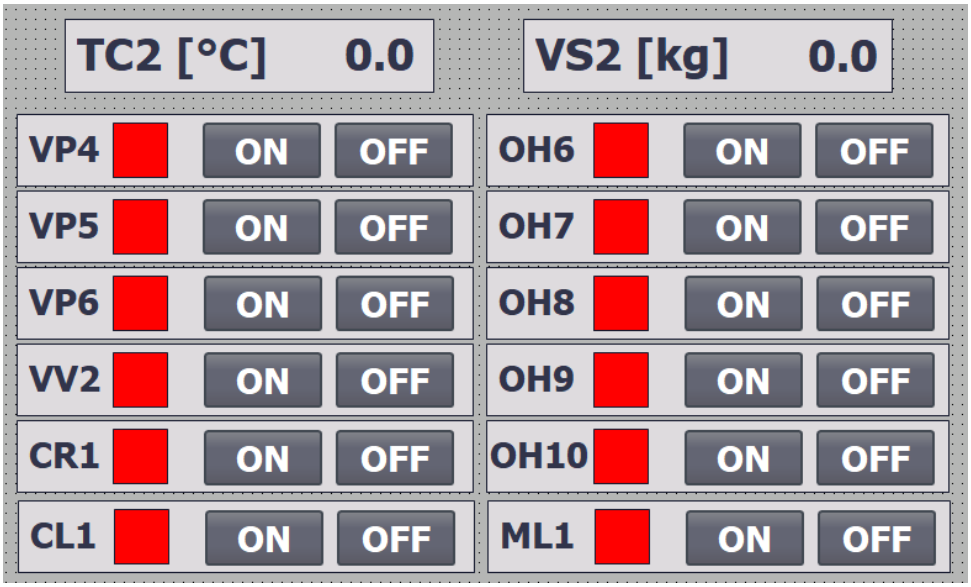
Obr. 3.100: Grafický panel HMI jednotky část – Hlavní obrazovka.

Rmutovací nádoba Podrobnější náhled oproti Hlavní obrazovce, kde jsou zobrazeny přehledněji stavy pro prvky napojené na Rmutovací nádobu. Je zde také možnost ručně přepínat všechny fyzické prvky.



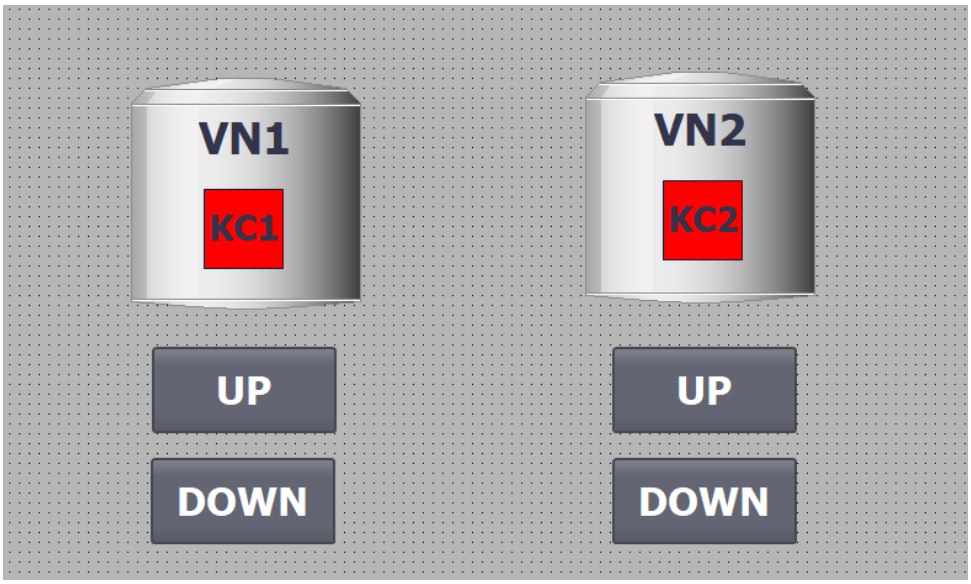
Obr. 3.101: Grafický panel HMI jednotky část – Rmutovací nádoba.

Chmelovarová nádoba Stejně jako u Rmutovací obrazovky se jedná o detailnější náhled na fyzické prvky připojené na nádobu Chmelovar. Je zde také možnost ručně přepínat všechny fyzické prvky.



Obr. 3.102: Grafický panel HMI jednotky část – Chmelovarová nádoba.

Zdvihový systém Poslední obrazovka HMI panelu, kde je možné ručně ovládat zdvih jednotlivých nádob.



Obr. 3.103: Grafický panel HMI jednotky část – Zdvihový systém.

Proces výroby

Celý proces vaření piva se skládá ze tří hlavních procesů – vaření, kvašení a zrání. Proces kvašení a zrání jsou dlouhodobé procesy, při kterých se prakticky kontroluje jen teplota a případně množství CO_2. Nejkomplexnější je proces vaření, kdy se musí dodržet stanovené postupy, jinak mohou ve výsledném pivu vznikat nežádoucí chutě a vůně. Celý proces vaření je popsán v následujících podkapitolách.

Čištění – Proplach Před samotným zahájím procesu vaření piva probíhá proplach cest pro odstranění případných nečistot po předchozím vařením nebo dezinfekcí. Nejprve je pomocí motoru MR1 nastavena nádoba VN2 do horní polohy, než hlásí koncový spínač KC2 pozici sepnuto. Dále jsou ventily VV1 a VV2 přepnuty do otevřené polohy a je spuštěno čerpadlo CR1, čímž je čerpána voda do obou nádob. Po dobu čerpání vody je pomocí VS1 a VS2 kontrolována hmotnost obou nádob, dokud každá z nádob nevrací nárůst o 10 kg. Následně je vypnuto čerpadlo CR1 a uzavření ventilů VV1 a VV2. Dalším krokem je ohřev vody v obou nádobách na dezinfekční teplotu, která by měla dosáhnout 70 °C. Vodou však budeme potřebovat vypláchnout veškeré cesty, kde bude docházet výměně tepla mezi vodou a zbylými prvky, které budou mít nižší teplotu. Z toho důvodu je teplota vody nastavena na 90 °C. U obou nádob jsou spuštěny topná tělesa (5. krok) dokud senzor teploty TC1 respektive TC2 nehlásí dosaženou teplotu. Následně probíhá proplach cest přes ventily VP2, VP3, VP5 a čerpadlo CR2, kdy je všechna voda přečerpána do nádoby VN1. Dále je proplach cest přes ventil VP1, kdy ventil VP2 je přepnuty do otevřené polohy a CR2 je spuštěno na 5 sekund (T_VN1=5s), poté jsou VP1 a VP2 přepnut do uzavřené polohy. Dále probíhá proplach přes ventil VP4, VP6 a VP7. Po celou otevření ventilů kontroluje senzor VS1 hmotnost v nádobě, dokud není dosaženo hmotnosti 0 kg, poté jsou všechny tři ventily uzavřeny. Tímto krokem končí proces proplachování. Přehledně pak, viz tabulka níže.

Tab. 3.25: Proces proplachu.

Krok	Proces	Popis
1.	MR1=VN2_UP→KC2=ON	Nastavení nádoby VN2 do horní polohy.
2.	VV1,VV2=ON	Přepnutí ventilů VV1 a VV2 do otevřené polohy.
3.	CR1=ON	Spuštění čerpadla CR1.
4.	VS1, VS2=10kg→CR1 ,VV1, VV2=OFF	Kontrola napuštěné vody do VN1 a VN2 s následným uzavřením ventilů VV1, VV2 a čerpadla CR1.
5.	OH1, OH2, OH6, OH7=ON	Spuštění topných těles pro ohřev VN1 (OH1, OH2) a VN2 (OH6, OH7)
6.	TC1, TC2=90 °C→OH1, OH2, OH6, OH7=OFF	Kontrola teploty pro VN1 a VN2 s následným uzavřením OH1, OH2, OH6 a OH7 po dosažení teploty 90 °C.
7.	VP2, VP3, VP5, CR2=ON→VS2=0kg	Přepnutí ventilů VP2, VP3 a VP5 do otevřené polohy do doby než hmotnostní senzor VS2 vrací hodnotu 0kg.
8.	VP1, VP2=ON	Přepnutí ventilů VP1 a VP2 do otevřené polohy.
9.	CR2=ON→5s	Spuštění čerpadla CR2 na dobu 5 sekund.
10.	VP1, VP2=OFF	Přepnutí ventilu VP2 do uzavřené polohy.
11.	MR1=VN1_UP→KC1=ON	Nastavení nádoby VN1 do horní polohy.
12.	VP4, VP6, VP7=ON→VS1=0kg	Přepnutí ventilů VP4 a VP6 do otevřené polohy do doby než hmotnostní senzor VS1 vrací hodnotu 0kg.

Vaření – Vystírka a zapářka V počáteční fázi vaření piva dochází k tzv. vystírání a zapářce. Vystírání je sloučení sladového šrotu s vodou. Vystírání se provádí studené (cca 20 °C) nebo teplé (35-38 °C) podle kvality použitého sladu. V dnešní době dodávané sladové šroty jsou velmi kvalitní, takže není potřeba provádět studené vystírání. Před začátkem vystírání je potřeba naplnění předem dané množství sladového šrotu podle typu a množství výsledného piva. Slad určuje mnoho faktorů jako je barva, stupňovitost, vůni, chuť atd. Nedá se tedy přímo stanovit pevné množství sladu, ale záleží na daném receptu. Nicméně, pokud bude předpokládat nejběžnější typy, dá se vycházet z tabulky 3.26, která kolik je potřeba množství vody na 1 kg sladu, abychom dostali požadovanou hustotu rmutu (viz kapitola 3.3.4).

Tab. 3.26: Poměr vody a sladu u nejběžnějších typů piva.

Typ piva	**Ležák světlý**	**Ležák tmavý**	**Ale**
voda/slad [l/kg]	3,5-4	3-3,5	2,5-3,5

Pro popis a demonstraci scénáře bude uvažován typ Ale s 15 kg sladového šrotu a průměrnou hodnotu 3,0 z intervalu zobrazeného v tabulce 3.26. Do nádoby VN1 je tedy nasypáno 15 kg sladu, kde tuto hodnotu hlídá a zobrazuje hmotnostní senzor VS1. Do druhé nádoby VN2 je napuštěno požadované množství vody 45 l, která je ohřáta na teplotu 42 °C. Teplota vody je mírně větší, než je udávána pro teplé vystírání z důvodu tepelné výměny vody při průchodu z jedné nádoby do druhé, a potom i v samotné nádobě VN1. Dále je přečerpána voda z nádoby VN2 do VN1 pomocí ventilů VP2, VP3 a VP5 a čerpadla CR2. Následuje zapářka, kdy je teplota vody se sladem navýšena na 52 °C, kde je udržována po dobu 20 minut (7. a 8. krok). Úroveň teplot jsou při procesu výroby piva velmi důležité, jelikož při nich dochází k různým procesů v samotném sladu. Během vystírky dochází ke zvýšení objemu škrobových zrn a přípravě na další procesy. V zapářce jsou potom podpořeny proteolytické enzymy a probíhá degradace škrobových zrn. Viz tabulka níže.

Tab. 3.27: Proces vystírání a zapářka.

Krok	**Proces**	**Popis**
1.	MR1=VN2_UP→KC2=ON	Nastavení nádoby VN2 do maximální polohy s kontrolním koncovým spínačem KC2.
2.	VS1=15kg	Kontrola nasypávání množství sladu do nádoby VN1.
3.	VV2, CR1=ON→VS2=45kg	Spuštění čerpadla a přepnutí ventilu VV2 do otevření polohy pro napouštění 45l vody do nádoby VN2.
4.	OH6, OH7= ON→TC2=42 °C	Spuštění topných těles OH6 a OH7 do dobu, než voda dosáhne teploty 42 °C.
5.	OD1=ON	Zapnutí odsávání par z vaření.
6.	VP2, VP3, VP5, CR2=ON→VS2=0kg	Přepnutí ventilů VP2, VP3, VP5 do otevřené polohy a spuštění čerpadla CR2 pro přečerpání vody do nádoby VN1 po dobu, než VS2 vrátí hodnotu 0 kg.
7.	OH1, OH2=ON→TC1=52 °C	Spuštění topných těles OH1 a OH2 do doby, než je dosaženo teploty 52 °C.
8.	T_VN1=20min∼IF=TC1<50-54 °C>THAN=OH1, OH2<ON-OFF>	Udržování teploty v nádobě VN1 po dobu 20 min v rozsahu 50-54 °C pomocí topných těles OH1 a OH2.

Vaření – Rmutování Po dokončení zapářky probíhá rmutování, kdy dochází ke štěpení dlouhých cukerných řetězců škrobu. Rmutování se dělí na dva základní přístupy – dekokce a infuze. Dekokce je většinou používána u spodně kvašených piv, např. typu ležák. Infuze je potom běžnější u svrchně kvašených piv (např. Ale).

Nicméně, jak již bylo řečeno, tak na prvním místě je samotný recept, takže je možné např. použít infuzní přístup pro spodně kvašená piva. Dekokce má tu výhodu, že při ní dochází k vyšší výtěžnosti melanoidních a karamelových látek a pivo je ve výsledku tzv. plnější. Nevýhodou je složitější proces a nutnost mít minimálně dvě nádoby, jelikož během procesu je potřeba rmut rozdělit, a s oběma částmi pracovat odděleně. U obou přístupů se se rmutování dále dělí na dle teploty na nižší cukrotvornou teplotu (60-65 °C), vyšší cukrotvornou teplotu (70-73 °C) a odrmutovací teplotu (76-78 °C).

Infuze Infuze je jednodušší přístup rmutování s nižší výtěžností melanoidních a karamelových látek ze sladu. Celý proces probíhá v jedné nádobě, kde se pomocí ohřevu a čerpadla a nebo míchadla dosahuje potřebných cukrotvorných teplot. V tabulce 3.28 je uveden celý postup infuze. Jako první krok je příprava vyslazovací vody, která bude použita po dokončení celého procesu. Vyslazovací voda slouží k doplnění výsledného rmutu na požadované množství mladiny, a následně i piva. V rámci fáze vystírka a zapářka bylo použito 45 litrů vody na 15 kg sladu, jelikož určitý objem vody nám zůstane ve sladovém šrotu (cca 1 l/1 kg sladu), tak do finální chmelovaru může počítat pouze s 30 litry rmutu. Pokud chceme mít finální produkt o objemu 60 litrů, budeme muset použít 35 litrů vyslazovací vody. Pět litrů je připočítáno na chmelovar, kde dochází k cca 8-10 % odpářky vody. Vyslazovací voda je napuštěna do nádoby VN2, kde je následně spuštěn ohřev, a teploměr TC2 hlídá teplotu, až do hodnoty 80 °C. Poté je teplota udržována až do konce rmutvání, kde bude jako poslední část použita k vyslazování. Samotné rmutování probíhá v nádobě VN1, kde je po zapářce teplota díla cca 52 °C navýšena pomocí tepelných těles OH1 a OH2 na 62 °C a pomocí čerpadla CR2 je zajištěno promíchávání díla pro větší výtěžnost a zabránění připalování. Během celého procesu při navyšování teplot by nárůst neměl být moc rychlý (cca 1 °C/min), aby nedošlo k teplotnímu stresu enzymů, které by mohly při příliš rychlém ohřevu denaturovat a přestat řádně fungovat. Také při rychlému ohřevu by mohlo docházek k velkému zahřívání dna nádoby a následnému připalování. Po dosažení teploty 62 °C je zařazena prodleva 30 minut, kdy je teplota udržována. Stejný proces probíhá při vyšší cukrotvorné teplotě. Dílo je ohřáto na teplotu 72 °C po dobu 20 min. V poslední fázi dormutování je dílo zahřáto na teplotu 76 °C, kde je opět udržováno po dobu 20 minut.

Tab. 3.28: Proces rmutování – infuze.

Krok	Proces	Popis
1.	VV2, CR1=ON→VS2=35kg	Spuštění čerpadla CR1 a přepnuti ventilu VV2 do otevření polohy pro napouštění 35l vody do nádoby VN2.
2.	OH6=ON //TC2=80 °C→OH6=OFF//	Spuštění ohřevu OH6 pro nádobu VN2
3.	//IF=TC2<79-80 °C>→OH6=<ON-OFF//	Průběžná kontrola teploty ve VN2 pomocí TC2 a případný ohřev pomocí OH6.
4.	VP1, VP2, CR2=ON	Přepnutí ventilů VP1, VP2 do otevřené polohy a spuštění čerpadla CR2.
5.	OH1, OH2→TC1=62 °C	Nastavení cílové teploty kapaliny v nádobě VN1 na 62 °C.
6.	T_VN1=30min∼IF=TC1<60-64 °C>THAN=OH1, OH2=<ON-OFF>	Udržování teploty v nádobě VN1 v intervalu 60-64 °C po dobu 30 min pomocí OH1, OH2 a TC1.
7.	OH1, OH2→TC1=72 °C	Nastavení cílové teploty kapaliny v nádobě VN1 na 72 °C.
8.	T_VN1=20min∼IF=TC1<70-74 °C>THAN=OH1, OH2=<ON-OFF>	Udržování teploty v nádobě VN1 v intervalu 70-74 °C po dobu 20 min pomocí OH1, OH2 a TC1.
9.	OH1, OH2→TC1=76 °C	Nastavení cílové teploty kapaliny v nádobě VN1 na 76 °C.
10.	T_VN1=10min∼IF=TC1<75-77 °C>THAN=OH1, OH2=<ON-OFF>	Udržování teploty v nádobě VN1 v intervalu 75-77 °C po dobu 10 min pomocí OH1, OH2 a TC1.
11.	VP1, VP2, CR2=OFF	Přepnutí ventilů VP1, VP2 do uzavřené polohy a vypnutí čerpadla CR2.

Vaření – Scezování a vyslazování Po dokončení fáze rmutování následuje scezování. V této fázi dochází oddělení sladiny od sladového mláta (tzv. předek) a následného vytěžení zbytku cukrů a dalších extraktů ze sladového mláta. Nejprve je vyslazovací voda o teplotě 80 °C přečerpána přes čerpadlo CR2 do nádoby VN1, aby následně celé dílo (sladina) byla přečerpána do nádob VN2 přes ventil VP7 a VP4. Viz tabulka níže.

Tab. 3.29: Proces scezování.

Krok	Proces	Popis
1.	IF=TC2<79-81 °C>THAN=OH6<ONOFF>	Kontrola teploty vody ve VN2 a případné ohřev na požadovanou teplotu. 79-81 °C.
2.	MR1=VN2_UP→KC2=ON	Přesunutí nádoby VN2 do horní polohy pomocí motoru MR1.
3.	VP2, VP3, VP5=ON	Přepnutí ventilů VP2, VP3, VP4 a VP5 do otevřené polohy.
4.	CR2=ON→VS2=0kg	Zapnutí čerpadla CR2 po dobu, než hmotnostní senzor VS2 vrátí hodnotu 0 kg.
5.	VP2=OFF	Přepnutí ventilů VP2, VP3 a VP4 do uzavřené polohy po dokončení čerpání.
6.	MR1=VN1_UP→KC2=ON	Přesunutí nádoby VN1 do horní polohy pomocí motoru MR1.
7.	VP7, VP4=ON→VS1=0kg	Přepnutí ventilů VP7 a VP4 do otevřené polohy po dobu, než hmotnostní senzor VS1 vrátí hodnotu 0 kg.

Vaření – Chmelovar Poslední částí vaření je chmelovar, kdy se sladina v nádob VN2 přivede k varu, který je udržován po dobu mezi 60 až 90 minut. Během tohoto procesu se v daných časech přidává chmel. Po dokončení je provedeno primární chlazení, kdy se pomocí míchadla teplotní výměny kapaliny a okolí sníží teplota na cca 60 °C. Míchadlo má také druhou funkci, kdy se provede tzv. whirpool, tedy využití odstředivé síly, kdy se zbytky nežádoucích látek usadí ve středu nádoby a nejsou přeneseny do procesu chlazení. Viz tabulka níže.

Tab. 3.30: Proces chmelovaru.

Krok	Proces	Popis
1.	OH6, OH7, OH8, OH9, OH10=ON	Spuštění topných těles OH6, OH7, OH8, OH9, OH10.
2.	//T_VN2=90min//	Nastavení časovače na 90 min pro průběh chmelovaru.
3.	DC1=90min	Přidání první dávky chmele na začátku chmelovaru.
4.	DC1=45min	Přidání druhé dávky chmele v polovině chmelovaru.
5.	DC1=20min	Přidání třetí dávky chmele 70 minutě chmelovaru.
6.	T_VN2=0min→OH6, OH7, OH8, OH9, OH10=OFF	Vypnutí topných těles OH6-OH10 po ukončení časovače.
7.	ML1=ON→TC2=60 °C	Spuštění míchadla ML1 po dobu, než senzoru TC2 vrátí hodnotu 60 °C.
8.	OD1=OFF	Vypnutí odsávání OD1 po ukončení chmelovaru.
8.	T_VN2=10min	Prodleva pro usazení nežádoucích látek produkovaných při chmelovaru.

Vaření – Chlazení

Po procesu vaření následuje chlazení mladiny. Chlazení mladiny na správnou teplotu je klíčové pro následný proces kvašení. Pivní kvasinky jsou velmi náchylné na změnu teploty, kdy v případě příliš nízké teploty by se mohlo samotné kvašení zastavit, a nebo při příliš velké by vznikly nežádoucí chutě a vůně v pivu. Proces začíná spuštěním chlazení CL1. Následně j otevřen ventil VP6 a kapalina je přečerpávána do doby, než váhový senzor VS2 vrací hodnotu 0 kg. Viz tabulka níže.

Tab. 3.31: Proces chlazení.

Krok	Proces	Popis
1.	CL1=ON	Spuštění chlazení CL1.
2.	VP6= ON	Přepnutí ventilů VP4 a VP6 do otevřeného stavu pro chlazení mladiny.
3.	VS2=0kg→ Cl1, VP6=OFF	Přepnutí ventilů VP4 a VP6 do uzavřeného stavu po dosažení stavu 0kg na VS2.

Specifikace testbedu

Testbed simulující proces výroby piva byl navržen a realizován jako komplexní simulátor průmyslové procesu s řadou senzorů, přepínačů a ventilů. Všechny tyto prvky zajišťují širokou škálu dat, které slouží jako vstupy pro průmyslové komunikační protokoly. V rámci komunikace jsou zastoupeny protokoly Modbus TCP, S7comm a OPC UA.

Protokoly

Modbus TCP Jeden z nejstarších, nejjednodušších, ale i nejpoužívanějších protokolů v současné době. V rámci komunikace testbedu je implementován pomocí knihovny pymodbus na minipočítači Raspberry Pi 3B+ (RPI1), který zpracovává analogové a digitální signály ze senzorů. Data ze senzorů jsou dále přeposílána pomocí protokolu Modbus do PLC1 jednotky S7-1500, která je hlavní řídicím členem celého testbedu.

S7comm Proprietární protokol S7comm od společnosti Siemens, který je uzpůsoben pro komunikaci mezi průmyslovými zařízeními. Protokol byl zvolen vzhledem k velkému podílu zařízeních Siemens SIMATIC na trhu. Díky tomu je vhodný jako kandidát z pohledu kybernetické bezpečnosti. S7comm má také rozšířenou podporu v podobě knihovny SNAP7 s implementací pro různé programovací jazyky, čím usnadňuje realizaci jak virtualizovaných zařízeních, tak i možnost jednoduché simulace kybernetických hrozeb.

OPC UA Komunikační protokol OPC UA je vzhledem ke svojí modularitě jedním z nejpoužívanějších protokolů mezi PLC jednotkami a dohledovými centry SCADA. V rámci testbedu zajišťuje komunikaci mezi řídicí PLC1 jednotkou a softwarovým SCADA serverem OpenMUC.

Funkcionality testbedu

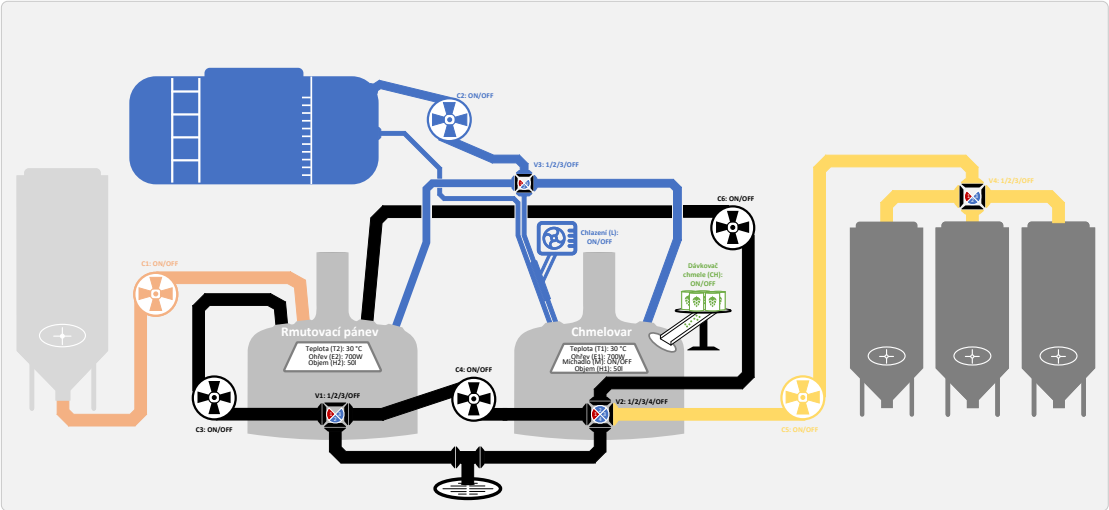
Jak již bylo zmíněno, fyzická realizace testbedu umožňuje simulaci různých fyzických prvků. Mezi ně patří binární stavové relé a elektromagnetické ventily, které mají v otevřeném stavu kontinuální průběh kladné logické hodnoty. Dalšími stavovými prvky jsou koncové spínače, které jsou v klidovém stavu přenáší logickou kladnou hodnotu. Dále jsou obsaženy analogové senzory teploty a hmotnosti, které nepřetržitě snímají a posílají hodnoty do řídicí jednotky, kde jsou pomocí A/D převodníku převedena na desetinná čísla. Procesně je testbed automatizován od počátečního napouštění vody (mimo sypání sladu, který probíhá ručně) až po finální přečerpání do kvasné nádoby. Z pohledu dílčích procesu je v testbedu realizováno:

- Napouštění a kontrola hmotnosti kapaliny,
- kontrolovaný ohřev kapaliny pomocí topných jednotek a senzoru teploty,
- změna polohy tělesa s koncovým spínačem,
- přečerpávání kapaliny pomocí čerpadel a propustných ventilů.

Virtualizovaná verze

Fyzické testbedy mají z pohledu reálných prvků a tedy i hodnot jasnou výhodu oproti virtuálním verzím. Fyzické prvky jsou však i jejich částečnou nevýhodou, kdy úprava nebo rozšíření sebou nese finanční i časové náročnost. Virtuální verze jsou v tomto ohledu podstatně jednodušší. Z tohoto důvodu byla vytvořena virtuální verze pivovaru, která však není totožná s fyzickou verzí. Pro komunikaci byl zvolen starší, ale stále využívaný protokol Modbus TCP, který je primárně implementován v operačním systému Raspbian na řídicí jednotce Raspberry Pi 3B+. Celý program je v implementován v programovacím jazyce Python 3.9 a realizován tak, aby byl jednoduše přenositelný, nebo duplikovatelný. Grafické rozhraní virtualizované verze, které je dostupné z dohledového centra OpenMUC, je zobrazeno na obrázku 3.104. Virtualizovaná verze je mimo grafické rozhraní spustitelná i v terminálové verzi, kde je řídicí stanice (místo OpenMUC) realizována pomocí dalšího programu simulujícího stranu klienta. Do samotného procesu který byl popsán v kapitole 3.3.4 bylo přidáno přisypávání sladu pomocí čerpadla C1. Spádové přečerpávání mezi nádobami varného procesu a finální přečerpání do kvasné nádoby bylo nahrazeno třemi čerpadly C4, C5 a C6. A poslední změnou je možnost přesnějšího nastavení topných

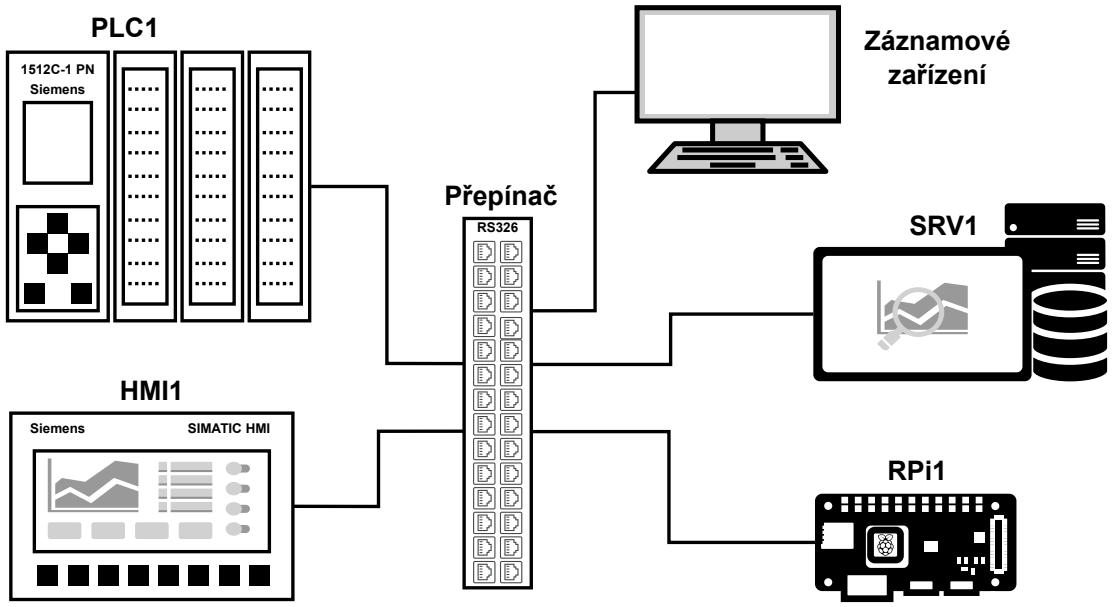
jednotek, kde virtualizovaná verze není omezena fyzickými topnými tělesy. Zbylá část procesu byla zachována, jak je popsáno v kapitole 3.3.4.



Obr. 3.104: Virtualizovaná verze testbedu – Pivovar.

3.3.5 Testování a verifikace

Jedním z klíčových požadavků při návrhu testovacího prostředí Pivovar bylo, aby prostředí mohlo sloužit pro generování, záznam a analýzu síťového provozu průmyslových protokolů. V této kapitole je provedeno testování za účelem analýzy zaznamenané síťové komunikace pro scénáře – standardní provoz procesu vaření piva, nestandardní provoz vyvolaný z důvodu nefunkčnosti některého z prvků nacházejících se v testbedu. Na obrázku 3.105 se nachází schéma testovacího prostředí pro níže uvedené scénáře (Testovací protokoly). Prostředí obsahoval stanice PLC1, HMI1, SRV1 a RPi1, které byly připojeny do přepínače Mikrotik CRS326. Na tomto přepínači bylo nastaveno zrcadlení portů tak, aby komunikace mezi výše uvedenými stanicemi byla přeposlána na počítač, který je ve schématu označen jako Záznamové zařízení.



Obr. 3.105: Schéma testovacího prostředí pro analýzu komunikace.

Testovací protokol 1: generování, záznam a analýza standardního provozu

Prvním z testovaných scénářů bylo generovaní, následný záznam a finální analýza standardního provozu z řídicí stanice PLC1, HMI1, RPi1 a SRV1, který představoval jeden cyklus vaření piva. Standardní provoz je chápán jako proces nebo cyklus, který probíhá dle očekávaných kritérií bez anomálií. Stanovením a klasifikací standardního provozu můžeme detekovat změny komunikaci, a rychleji tak reagovat na případné poruchy a útoky. Pro analýzu standardního provozu byl zvolen jeden cyklus vaření, tedy od počátečního napouštění vody, až po finální přečerpání do kvasné nádoby. Součástí cyklu byla, jak lokální zobrazovací jednotka HMI, tak i dohledové centrum OpenMUC, kam se periodicky každou sekundu posílají hodnoty všech prvků procesu. Celý proces vaření při definovaných objemem z kapitoly 3.3.4 trvá více než 6 hodin, kde nejdelší části jsou během ohřevu a chlazení kapaliny.

Během celého procesu bylo zaznamenáno 200 568 paketů, kde 36,8 % bylo protokolu S7comm, 18,4 % Modbus TCP a zbytek byla komunikace síťových protokolu TCP, UDP atd. Po zaokrouhlení jde o 9,28 paketů za sekundu. Z toho je 3,42 paketů pro protokol S7comm a 1,71 pro Modbus. Průměrně mají zprávy velikost 187 bajtů pro S7comm a 173 bajtů pro Modbus TCP. Při standardním provozu testbed generuje přibližné 923 bajtů za sekundu komunikace S7comm a Modbus.

Testovací protokol 2: generování, záznam a analýza nestandardního provozu

V druhém z testovaných scénářů bylo provedeno generování, záznam a následná analýza síťového provozu ze stanice PLC1, HMI1, RPi1 a SRV1, během jednoho cyklu vaření piva, kde byly zavedeny výpadky různých členů.

Nestandardní provoz je chápán jako časová intervalem ohraničená událost vyskytující se ve standardním provozu. Nestandardní provoz může být způsoben legitimní, nebo nelegitimní událostí. Legitimní událost je způsobena bez úmyslného záměru k poškození některého z prvků daného systému. Může se jednat např. o výpadek zařízení z důsledku chybné aktualizace, konce životnosti nebo neodborné manipulace. Oproti tomu nelegitimní událost je způsobena úmyslně s cílem poškodit dané zařízení, systém, společnost atd. V případě zpětné analýzy těchto událostí je v mnohých případech těžké od sebe odlišit, jestli je jednalo o legitimní nebo nelegitimní událost.

Zde jsou uvedeny realizované legitimní výpadky, které byly simulovány v testovacím prostředí Pivovar, během jednoho procesu vždy po dobu 60 sekund.

Výpadek PLC1 stanice Centrální řídicí jednotka, která obsluhuje celý proces vaření piva. Na tuto jednotku je připojena většina prvků v podobě ventilů, čerpadel, relé atd. Při odpojení/výpadku stanice došlo k přerušení síťové komunikace a odepnutí všech aktivovaných prvků. Výpadek byl aktivován na začátku celého procesu při fázi čištění, kdy docházelo k přečerpání z nádoby VN2 do VN1 (7. krok). NC ventily, které byly v otevřeném stavu, se vrátily po ztrátě napájení do uzavřené polohy a přestala se přečerpávat kapalina. Z pohledu komunikace bylo ukončeno spojení z HMI1, SRV1 a RPi1 jednotkami. Pouze SRV1 a HMI1 se pokoušeli o opětovné navázání spojení s PLC1. Stanice RPi1 neiniciovala žádnou komunikaci pomocí průmyslového protokolu Modbus. Po znovuspuštění PLC1 stanice musel být spuštěn celý cyklus od začátku, jelikož stanice si nepamatuje poslední stav, ve kterém se nacházela při přerušení.

Výpadek RPi1 stanice Druhý z testovaných výpadků byl na stanici RPi1, která zajišťuje odečet a přenos hodnot z teplotních senzorů. Odečty teplot na obou nádobách jsou klíčové v případě ve téměř v celém procesu. Z tohoto důvodu byl tento výpadek simulován ve dvou situacích.

První situace byla během fáze Čištění, kdy docházelo k vypouštění kapaliny, a tedy stanice PLC1 z pohledu probíhající fáze programu nekontrolovala hodnotu teploty. Z pohledu komunikace bylo ukončeno TCP spojení mezi PLC1 a RPi1. Následně bylo ze strany PLC1 provedeny pokusy o navázání spojení nového TCP spojení.

Druhá situace byla realizována ve fázi Rmutování, kdy je kontrola teploty stěžejní pro správný průběh celé fáze. Z pohledu komunikace výpadek probíhal totožně jako v první případě. TCP spojení bylo ukončeno a ze strany PLC1 byly prováděny pokusy o znovu navázání. Stěžejní zde bylo chování PLC1, bez možnosti čtení teplotní hodnot ze stanice RPi1 a tím případné narušení celé fáze Rmutování. Po analýze bylo zjištěno, že PLC1 sice pokoušela navázat znovu spojení, ale zároveň si v paměti udržovala poslední odečtenou hodnotu z RPi1. Z pohledu krátkodobého výpadku (v řádu jednotek minut) by měla nefunkčnost RPi1 stanice minimální dopad. Celý proces by byl sice prodloužen, ale na výsledný produkt to neovlivnilo. Oproti tomu v případě dlouhodobého výpadku by následky mohly mít až destruktivní charakter. Pokud by docházelo k neustálému ohřevu kapaliny nebo chlazení, podle toho v jaké fázi by se proces nacházel, mohlo by dojít k poškození prvků v systému nebo samotného vařeného produktu. Například ve fázi Rmutování mají nastavené rmutovací teploty výrazný vliv na výsledný produkt. Stejně tak při fázi Vaření by mohlo dojít k vyššímu odparu nebo až připálení, což zase mělo dopad na výsledný produkt.

Výpadek HMI1 stanice Grafická zobrazovací jednotka pro lokální ovládání a kontrolu celého procesu komunikuje pomocí protokolu S7comm s řídicí jednotkou PLC1. Po startu legitimního proces vaření není HMI1 panel nezbytný pro dokončení procesu. Pokud není nutný zásah operátor do procesu, tak panel slouží jen jako zobrazovací jednotka. V případě výpadku, který byl realizován během fáze Rmutování i fáze Vaření, by nemělo dojít, mimo pokles komunikace mezi PLC1 a HMI1, k narušení procesu. Tento předpoklad by potvrzen během obou realizovaných výpadků. Z pohledu komunikace bylo při obou výpadcích ukončeno TCP spojení mezi PLC1 a HMI1 jednotkami. Jelikož je TPC spojení inicializováno ze strany HMI1 jednotky, nedocházelo ani k pokusům o navázání nového TCP spojení.

Výpadek SRV1 stanice Dohledové a řídicí centrum SCADA OpenMUC slouží pro vzdálenou vizualizací průběhu procesu vaření piva. OpenMUC umožňuje vzdálené spuštění nebo ukončení a následný dohled nad procesem. Výpadek stanice byl realizován ve fázi Rmutování a Vaření. Z pohledu komunikace je navázáno TCP spojení mezi PLC1 a HMI1 stanicí. Při výpadku bylo toto spojení ukončeno a jelikož je inicializováno ze strany SRV1, tak se v komunikaci nenacházeli pokusy o opětovné navázání.

Hodnocení legitimních výpadku Z testování legitimních výpadků bylo zaznamenáno a analyzováno chování stanic PLC1, SRV1, HMI1 a RPi1. Z pohledu vlivu na proces vaření jsou kritické stanice PLC1 a RPi1, které přímo ovládají některé z koncových prvků. HMI1 a SRV1 jsou klíčové pro spuštění procesu a pro jeho případné

ukončení, ale během vaření má jejich případný výpadek minimální dopad na proces. Z pohledu komunikace nevnáší výpadky téměř žádnou nadbytečnou komunikaci např. z pohledu opakovaných dotazů (mimo výpadek RPi1).

Kontrola požadavků na vytvořený testbed

V tabulce 3.32 je provedeno shrnutí stanovených požadavků a jejich realizace v rámci testbedu Pivovar dle kapitoly 3.3.3. Veškeré stanovené kritéria a požadavky byly splněny.

Tab. 3.32: Tabulka požadavků a jejich realizace v testovacím prostředí.

Požadavek	Zajištěno	**Realizace v rámci testbedu**
Průmyslový protokol	Ano	S7comm, Modbus TCP.
PLC	Ano	Automatizační jednotka Siemens SIMATIC S7-1500.
HMI	Ano	Rozhraní člověk-stroj Siemens SIMATIC KTP-700.
SCADA	Ano	Open-source softwarové verze OpenMUC.
Datové úložiště	Ano	QNAP NAS s více než 15 TB úložného prostoru.
Fyzické prvky	Ano	El.mag. ventily, relé , senzory – teploty váhy, průtoku.
Záznamové zařízení	Ano	Síťová sonda Profishark 1G, VUT síťová sonda.
Analýza a simulace	Ano	Stolní a rackové počítače se softwarem jako je Kali linux, Debian, Tenzorflow, Wireshark, TIA portál atd.
Virtualizovaná verze	Ano	Virtualizovaná verze s protokolem Modbus.

4 Závěr

Tato práce představila aktuální výzvy a příležitosti, které přináší digitalizace průmyslu i nové trendy jako Průmysl 4.0, IoT, IIoT a další. Jeden z hlavních přínosů práce je ucelená terminologie, která napomáhá k chápaní vývoji celého OT odvětví, spojování jednotlivých odvětví včetně jejich slučování a oddělování, jako např. v případě kybernetické bezpečnosti. Byly také představeny základní normy, předpisy a standardy, které se tykají návrhu a vývoje v prostředí OT. V rámci práce byly dále vysvětleny základní komponenty OT včetně názorných příkladů z praxe včetně jejich možné simulace a matematického popisu. Byly přiblíženy jednotlivé modely komponent i jejich vzájemná návaznost a propojenost. Ucelený přehled tak jasným způsobem ukazuje složitost těchto systémů a nutnost precizního návrhu i samotné implementace. V neposlední řadě tato práce přibližuje výzkum a vývoj v rámci návrhu, implementace, testování, optimalizace i finalizace tří vybraných ukázek - případových studií (průmyslová balicí smyčka, čistička a pivovar). Tyto ukázky názorným způsobem popisují postup od vzniku myšlenky, návrhu funkčních a užitných parametrů až po jejich naplnění. Tímto práce uzavírá komplexní pohledu na dnešní OT svět. Závěrem jsou již uvedeny stručně odpovědi na stanovené stěžejní otázky:

- **Jaké výzvy a příležitosti přináší digitalizace průmyslu?**
  - Digitalizace přináší příležitosti, jako je zvýšení efektivity, produktivity a flexibility ve výrobním procesu. Umožňuje také využití pokročilých technologií, jako je umělá inteligence, analytika velkých dat a internet věcí (IoT). Přináší však také výzvy, jako jsou rizika kybernetické bezpečnosti, potřeba nových dovedností a náklady na implementaci nových technologií. V rámci práce byly všechny kladné i záporné stránky digitalizace blíže vysvětleny.
- **Jak měnící se prostředí ovlivňuje přijetí a implementaci OT?**
  - Měnící se prostředí, jako jsou nové předpisy, technologický pokrok a ekonomické faktory, mohou ovlivnit přijetí a implementaci OT. Společnosti mohou čelit problémům při zavádění nových technologií kvůli starším systémům, nedostatku odborných znalostí nebo odporu vůči změnám. Práce představila aktuální situaci v rámci dnešních OT.
- **Jaká je současná úroveň jednotnosti terminologie používané v rámci OT a jak ji lze standardizovat?**
  - Současná úroveň jednotnosti v terminologii používané v OT se může lišit v závislosti na odvětví a společnosti. Standardizace lze dosáhnout použitím mezinárodních norem a osvědčených postupů, jakož i iniciativ specifických pro odvětví za účelem vytvoření společné terminologie. Nicméně aktuálně chybí celistvá terminologie a značná část odvětví používá a je zvyklá na vlastní metodiku. Práce přiblížila hlavní rozdíly včetně možného ucelení terminologie.

- **Jakou roli hrají normy, předpisy a normy při utváření vývoje a implementace systémů OT?**
  - Normy, předpisy a normy hrají důležitou roli při utváření vývoje a implementace systémů OT. Poskytují pokyny pro návrh, provoz a údržbu systémů OT a také zajišťují interoperabilitu a shodu s právními a bezpečnostními požadavky. Práce představila hlavní standardy, normy a další legislativní nařízení v rámci kontextu s OT.
- **Jaké jsou základní komponenty v rámci OT sítí a jak jsou napojeny na dnešní chápání průmyslových sítí?**
  - Mezi základní komponenty OT sítí patří senzory, akční členy, ovladače a komunikační protokoly. Tyto komponenty jsou vzájemně propojeny a tvoří síť, která umožňuje sběr, zpracování a přenos dat. Dnešní chápání průmyslových sítí zahrnuje pojmy jako interoperabilita, kybernetická bezpečnost a používání standardních komunikačních protokolů. Všechny základní komponenty včetně architektury OT jsou v práci představeny.
- **Jaký je současný stav konvergence IT a OT a jaký je její dopad na průmysl?**
  - Konvergence mezi IT a OT je rostoucí trend v tomto odvětví, kde se hranice mezi těmito dvěma doménami stále více stírají. Tento trend umožňuje větší integraci mezi obchodními systémy a produkčními systémy, stejně jako použití pokročilých technologií ke zlepšení efektivity a produktivity. V rámci práce je konvergence diskutována s názornou ukázkou jednotlivých překážek i přínosů.
- **Jak lze moderní OT technologie, jako je Průmysl 4.0 a IoT, využít k efektivnímu provozu průmyslových sítí?**
  - Moderní OT technologie lze použít k monitorování a řízení průmyslových sítí v reálném čase, sběru a analýze dat a optimalizaci výrobních procesů. Průmysl 4.0 a IoT umožňují využití pokročilých technologií, jako je prediktivní údržba, vzdálené monitorování a strojové učení, ke zlepšení efektivity a snížení prostojů. Práce přibližuje moderní trendy v OT a dává jim kontext v rámci současných průmyslových sítí.
- **Jaké překážky a řešení představuje implementace OT v průmyslových aplikací?**
  - Překážky implementace OT v průmyslových aplikacích zahrnují náklady na implementaci, starší systémy, rizika kybernetické bezpečnosti a potřebu nových dovedností. Mezi řešení těchto překážek patří vytvoření jasného obchodního případu, upřednostnění kybernetické bezpečnosti, investice do školení a vzdělávání a spolupráce se zkušenými partnery. Práce představuje tři vybrané případové studie a ukazuje možnosti od návrhu, přes implementaci až po finalizaci průmyslové aplikace.

Autorovy publikace

[APub1] Benedikt, J.; Vrtal, M.; Fujdiak, R.; aj.: Virtualization platform for urban infrastructure. In *2022 22nd International Scientific Conference on Electric Power Engineering (EPE)*, IEEE, 2022, s. 1–5.

[APub2] Blazek, P.; Fujdiak, R.; Hodon, M.; aj.: Communication Anomaly Detection in Cyber-physical Systems. In *SEIA'2019 Conference Proceedings*, Lulu. com, 2020, str. 311.

[APub3] Blazek, P.; Fujdiak, R.; Mlynek, P.; aj.: Development of cyber-physical security testbed based on IEC 61850 architecture. *Elektronika ir Elektrotechnika*, ročník 25, č. 5, 2019: s. 82–87.

[APub4] Fujdiak, R.; Blazek, P.; Apvrille, L.; aj.: Modeling the trade-off between security and performance to support the product life cycle. In *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, IEEE, 2019, s. 1–6.

[APub5] Fujdiak, R.; Blazek, P.; Chmelar, P.; aj.: Communication Model of Smart Substation for Cyber-Detection Systems. In *International Conference on Computer Networks*, Springer, 2019, s. 256–271.

[APub6] Fujdiak, R.; Blazek, P.; Mikhaylov, K.; aj.: On track of sigfox confidentiality with end-to-end encryption. In *Proceedings of the 13th International Conference on Availability, Reliability and Security*, 2018, s. 1–6.

[APub7] Fujdiak, R.; Blazek, P.; Mlynek, P.; aj.: Developing Battery of Vulnerability Tests for Industrial Control Systems. In *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, IEEE, 2019, s. 1–5.

[APub8] Fujdiak, R.; Mikhaylov, K.; Pospisil, J.; aj.: Insights into the issue of deploying a private LoRaWAN. *Sensors*, ročník 22, č. 5, 2022: str. 2042.

[APub9] Fujdiak, R.; Mikhaylov, K.; Stusek, M.; aj.: Security in low-power wide-area networks: State-of-the-art and development toward the 5G. In *LPWAN Technologies for IoT and M2M Applications*, Elsevier, 2020, s. 373–396.

[APub10] Fujdiak, R.; Mlynek, P.; Blazek, P.; aj.: Seeking the relation between performance and security in modern systems: metrics and measures. In *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2018, s. 1–5.

[APub11] Fujdiak, R.; Mlynek, P.; Malina, L.; aj.: Development of IQRF technology: Analysis, simulations and experimental measurements. *Elektronika ir Elektrotechnika*, ročník 25, č. 2, 2019: s. 72–79.

[APub12] Fujdiak, R.; Mlynek, P.; Misurec, J.; aj.: Simulated coverage estimation of single gateway LoRaWAN network. In *2018 25th International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, 2018, s. 1–4.

[APub13] Fujdiak, R.; Mlynek, P.; Mrnustik, P.; aj.: Managing the secure software development. In *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, IEEE, 2019, s. 1–4.

[APub14] Fujdiak, R.; Mlynek, P.; Slacik, J.; aj.: Investigating the Suitability of Blockchain for Smart Grid. In *2019 20th International Scientific Conference on Electric Power Engineering (EPE)*, IEEE, 2019, s. 1–6.

[APub15] Fujdiak, R.; Orgon, M.; Hallon, J.; aj.: Radiation of an Electromagnetic Field from the Power Line Communication Adapters. In *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2018, s. 1–4.

[APub16] Fujdiak, R.; Pokorny, J.; Zobal, L.; aj.: Security and Performance Tradeoffs for Data Distribution Service in Flying Ad-Hoc Networks. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2019, s. 1–5.

[APub17] Fujdiak, R.; Slacik, J.; Orgon, M.; aj.: Investigation of power line communication and wi-fi co-existence in smart home. In *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2018, s. 1–4.

[APub18] Fujdiak, R.; Uher, V.; Mlynek, P.; aj.: IP Traffic Generator Using Container Virtualization Technology. In *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2018, s. 1–6.

[APub19] Gadala, M.; Strigini, L.; Fujdiak, R.: Authentication for Operators of Critical Medical Devices: A Contribution to Analysis of Design Tradeoffs. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, ACM, 2022.

[APub20] Holasova, E.; Fujdiak, R.: Deep Neural Networks for Industrial Protocol Recognition and Cipher Suite Used. In *2022 IEEE International Carnahan Conference on Security Technology (ICCST)*, IEEE, 2022, s. 1–7.

[APub21] Holasova, E.; Fujdiak, R.; Kuchar, K.: Specific Anomaly Detection Method in Wireless Communication Networks. In *2020 4th Cyber Security in Networking Conference (CSNet)*, IEEE, 2020, s. 1–3.

[APub22] Holasova, E.; Kuchar, K.; Fujdiak, R.; aj.: Security modules for securing industrial networks. In *2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)*, IEEE, 2021, s. 1125–1132.

[APub23] Ilgner, P.; Fujdiak, R.: Fuzzing Framework for IEC 60870-5-104 Protocol. In *Proceedings of the 5th International Conference on Computer Science and Software Engineering*, 2022, s. 190–194.

[APub24] Ilgner, P.; Fujdiak, R.: Fuzzing ICS Protocols: Modbus Fuzzer Framework. In *2022 IEEE International Carnahan Conference on Security Technology (ICCST)*, IEEE, 2022, s. 1–6.

[APub25] Kuchar, K.; Fujdiak, R.; Blazek, P.; aj.: Simplified Method for Fast and Efficient Incident Detection in Industrial Networks. In *2020 4th Cyber Security in Networking Conference (CSNet)*, IEEE, 2020, s. 1–3.

[APub26] Kuchar, K.; Holasova, E.; Fujdiak, R.; aj.: Incident Detection System for Industrial Networks. In *Big Data Privacy and Security in Smart Cities*, Springer, 2022, s. 83–102.

[APub27] Malina, L.; Srivastava, G.; Dzurenda, P.; aj.: A secure publish/subscribe protocol for internet of things. In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, 2019, s. 1–10.

[APub28] Masek, P.; Younesian, E.; Bahna, M.; aj.: Performance Analysis of Different LoRaWAN Frequency Bands for mMTC Scenarios. In *2022 45th International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2022, s. 417–420.

[APub29] Mikhaylov, K.; Fujdiak, R.; Pouttu, A.; aj.: Energy attack in LoRaWAN: experimental validation. In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, 2019, s. 1–6.

[APub30] Mikhaylov, K.; Stusek, M.; Masek, P.; aj.: Communication performance of a real-life wide-area low-power network based on Sigfox technology. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, IEEE, 2020, s. 1–6.

[APub31] Mikhaylov, K.; Stusek, M.; Masek, P.; aj.: On the Performance of Multi-Gateway LoRaWAN Deployments: An Experimental Study. In *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2020, s. 1–6.

[APub32] Mlynek, P.; Misurec, J.; Silhavy, P.; aj.: Simulation of achievable data rates of broadband power line communication for smart metering. *Applied Sciences*, ročník 9, č. 8, 2019: str. 1527.

[APub33] Mlynek, P.; Misurec, J.; Toman, P.; aj.: Performance testing and methodology for evaluation of power line communication. *Elektronika ir Elektrotechnika*, ročník 24, č. 3, 2018: s. 88–95.

[APub34] Mlynek, P.; Slacik, J.; Fujdiak, R.: Experimental Measurements of Communication Technologies for Mesh Distribution Networks of Low Voltage. In *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2018, s. 1–5.

[APub35] Pokorny, J.; Fujdiak, R.; Kovanda, M.; aj.: Traffic Analysis of IEEE 802.11 on Physical Layer by using Software Defined Radio. In *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2020, s. 78–81.

[APub36] Pospisil, J.; Fujdiak, R.; Mikhaylov, K.: Investigation of the performance of TDoA-based localization over LoRaWAN in theory and practice. *Sensors*, ročník 20, č. 19, 2020: str. 5464.

[APub37] Pospisil, O.; Blazek, P.; Fujdiak, R.; aj.: Active Scanning in the Industrial Control Systems. In *2021 International Symposium on Computer Science and Intelligent Controls (ISCSIC)*, IEEE, 2021, s. 227–232.

[APub38] Pospisil, O.; Blazek, P.; Kuchar, K.; aj.: Application perspective on cybersecurity testbed for industrial control systems. *Sensors*, ročník 21, č. 23, 2021: str. 8119.

[APub39] Pospisil, O.; Fujdiak, R.; Mikhaylov, K.; aj.: Testbed for lorawan security: Design and validation through man-in-the-middle attacks study. *Applied Sciences*, ročník 11, č. 16, 2021: str. 7642.

[APub40] Potisk, L.; Hallon, J.; Orgon, M.; aj.: Electromagnetic compatibility of PLC adapters for in-home/domestic networks. *Journal of Electrical Engineering*, ročník 69, č. 1, 2018: s. 79–84.

[APub41] Róka, R.; Fujdiak, R.; Holasova, E.; aj.: Protection Schemes in HPON Networks Based on the PWFBA Algorithm. *Sensors*, ročník 22, č. 24, 2022: str. 9885.

[APub42] Ruotsalainen, H.; Shen, G.; Zhang, J.; aj.: LoRaWAN Physical Layer-Based Attacks and Countermeasures, A Review. *Sensors*, ročník 22, č. 9, 2022: str. 3127.

[APub43] Sikora, M.; Fujdiak, R.; Kuchar, K.; aj.: Generator of slow denial-of-service cyber attacks. *Sensors*, ročník 21, č. 16, 2021: str. 5473.

[APub44] Sikora, M.; Fujdiak, R.; Misurec, J.: Analysis and detection of application-independent slow Denial of Service cyber attacks. In *2021 IEEE International Conference on Intelligence and Security Informatics (ISI)*, IEEE, 2021, s. 1–6.

[APub45] Sikora, M.; Krivulcik, A.; Fujdiak, R.; aj.: Design of Advanced Slow Denial of Service Attack Generator. In *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2020, s. 99–104.

[APub46] Slacik, J.; Mlynek, P.; Fujdiak, R.: Broadband Power-line Devices Comparison and HomePlug AV2 Experimental Measurement. In *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2018, s. 1–4.

[APub47] Slacik, J.; Mlynek, P.; Fujdiak, R.; aj.: Capabilities and Visions of Broadband Power-Line in Smart Grids Applications. In *2019 20th International Scientific Conference on Electric Power Engineering (EPE)*, IEEE, 2019, s. 1–5.

[APub48] Tin, P. T.; Vozriak, M.; Fujdiak, R.; aj.: Secrecy Performances of Wireless Relay Systems Affected by Hardware Impairments. In *2019 PhotonIcs & Electromagnetics Research Symposium-Spring (PIERS-Spring)*, IEEE, 2019, s. 1522–1529.

[APub49] Voznak, M.; Hendrych, J.; Orcik, J.; aj.: Population Mobility Data Retrieval from Wireless Cellular Networks. In *Proceedings of the 2019 Progress in Electromagnetics Research Symposium (PIERS-Rome)*, ročník 2019, 2019, ISBN 978-4-88552-316-8, s. 1–9.

[APub50] Vrtal, M.; Benedikt, J.; Fujdiak, R.; aj.: Investigating the Possibilities for Simulation of the Interconnected Electric Power and Communication Infrastructures. *Processes*, ročník 10, č. 12, 2022: str. 2504.

[APub51] Vrtal, M.; Benedikt, J.; Topolánek, D.; aj.: Power grid and data network simulator. In *2022 22nd International Scientific Conference on Electric Power Engineering (EPE)*, IEEE, 2022, s. 1–4.

[APub52] Zobal, L.; Kolář, D.; Fujdiak, R.: Current State of Honeypots and Deception Strategies in Cybersecurity. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2019, s. 1–9.

Autorovy pedagogické materiály

[APed1] Fujdiak, R.: Základy kryptografie a kryptografických algoritmů. 2015, (*skripta pro předmět MKRI*).

[APed2] Fujdiak, R.: Datová komunikace: Internet of Things IoT a Sigfox. 2017, (*laboratorní úloha pro předmět BDAK*).

[APed3] Fujdiak, R.: Datová komunikace: Technologie LoRaWAN. 2017, (*laboratorní úloha pro předmět BDAK*).

[APed4] Fujdiak, R.: Foundations of Cryptography: Random Numbers. 2018, (*přednáška pro předmět CZKR*).

[APed5] Fujdiak, R.: Vysokorychlostní komunikační systémy: LPWA Komunikační technologie - LoRaWAN. 2019, (*přednáška pro předmět BVKS a KVKS*).

[APed6] Fujdiak, R.: Vysokorychlostní komunikační systémy: LPWA Komunikační technologie - Sigfox. 2019, (*přednáška pro předmět BVKS a KVKS*).

[APed7] Fujdiak, R.; Jiří, M.: Počítačem podporovaná řešení inženýrských problémů (P1–P13). 2014, (*audiovizuální tvorba pro předmět MPPR*).

[APed8] Fujdiak, R.; Jiří, M.: Číslicové zpracování signálů (P1–P12). 2014, (*audiovizuální tvorba pro předmět BCZS*).

[APed9] Fujdiak, R.; Jiří, M.: Vyšší techniky datových přenosů: Technologie LPWAN a jejich místo mezi ostatními bezdrátovými technologiemi. 2016, (*přednáška pro předmět MVDP a LVDP*).

[APed10] Mašek, P.; Štůsek, M.; Fujdiak, R.; aj.: Komunikační systémy pro IoT. 2019, (*skripta pro předmět BVKS a KVKS*).

[APed11] Stodůlka, T.; Fujdiak, R.: *Budování Cyber Range platformy s technologií cloud computingu*. Vysoké učení technické v Brně, 2022, ISBN 9788021460645.

Autorova účast na projektech

[APro1] 731198: Rotterdam, Umea and Glasgow: Generating Exemplar Districts In Sustainable Energy Deployment (RUGGEDISED). 01.11.2016–31.10.2021, H2020-EU (*člen řešitelského týmu*).

[APro2] 737475: Aggregated Quality Assurance for Systems (AQUAS). 01.05.2017–30.06.2020, H2020-EU (*člen řešitelského týmu*).

[APro3] FAST/FEKT-J-16-3344: Agregační brána pro zabezpečený přenos dat z okamžitých měření fyzikálních veličin. 01.01.2016–31.12.2016, FEKT - interní (*spoluřešitel*).

[APro4] FEKT-S-14-2352: Výzkum elektronických komunikačních a informačních systémů. 01.01.2014–31.12.2016, FEKT - interní (*člen řešitelského týmu*).

[APro5] FEKT-S-17-4184: Výzkum informačních a komunikačních systémů a jejich bezpečnost. 01.01.2017–31.12.2019, FEKT - interní (*člen řešitelského týmu*).

[APro6] FEKT-S-20-6312: Výzkum elektronických komunikačních a informačních a systémů a jejich využití pro zabezpečení kritických infrastruktur. 01.01.2020–31.12.2022, FEKT - interní (*člen řešitelského týmu*).

[APro7] FEKT/FIT-J-18-5434: Výzkum efektivních kryptografických metod pro zvýšení bezpečnosti nastupujících komunikačních technologií v oblasti Internetu věcí. 01.01.2018–31.12.2018, FEKT - interní (*člen řešitelského týmu*).

[APro8] FEKT/FIT-J-19-5905: Experimentální prostředí pro výzkum, evaluaci a testování distribuovaných datových systémů. 01.01.2019–31.12.2019, FEKT - interní (*člen řešitelského týmu*).

[APro9] FEKT/FIT-J-19-5905: Pokročilé metody hluboké inspekce v aplikační vrstvě pro obranu proti dnešním hrozbám. 01.01.2019–31.12.2019, FEKT - interní (*člen řešitelského týmu*).

[APro10] FV20487: Inteligentní řešení pro zvýšení efektivity a automatizace pracovního procesu pro implementaci konceptu Průmysl 4.0. 01.09.2017–31.12.2019, MPO (*spoluřešitel*).

[APro11] FV40366: Datový monitoring pro zvýšení spolehlivosti procesů chytrých továren. 01.08.2019–31.12.2021, MPO (*spoluřešitel*).

[APro12] FW01010474: Analýza, detekce a mitigace hrozeb dostupnosti síťových služeb. 01.04.2020–31.12.2022, TA ČR (*spoluřešitel*).

[APro13] TJ01000381: Pokročilé behaviorální modely aplikační vrstvy pro efektivní analýzu provozu v podnikových sítích. 01.01.2018–30.06.2019, TA ČR (*spoluřešitel*).

[APro14] TJ02000332: Pokročilé metody monitorování provozu bezdrátových sítí. 01.06.2019–31.05.2021, TA ČR (*hlavní řešitel*).

[APro15] TK02030013: Kyber-fyzikální dvojče městské infrastruktury zítřka. 01.07.2019–30.06.2023, TA ČR (*spoluřešitel*).

[APro16] TK03010091: Dopady kybernetické bezpečnosti na regulované oblasti smart meteringu. 01.10.2020–31.12.2021, TA ČR (*člen řešitelského týmu*).

[APro17] VI20172019057: Monitoring a analýza komunikace pro bezpečnostní dohled kritických energetických infrastruktur. 01.01.2017–31.12.2019, MV ČR (*spoluřešitel*).

[APro18] VI20192022132: Kybernetická aréna pro výzkum, testování a edukaci v oblasti kyberbezpečnosti. 01.07.2019–30.06.2022, MV ČR (*spoluřešitel*).

Ostatní reference

[1] OpenMUC.
URL `https://www.openmuc.org/`

[2] Dobot Magician User Guide. 2019.
URL `https://afrel.co.jp/pdf/product/dobot-magician-user-guideV1.7.0_English.pdf`

[3] Pydobot 1.3.2. 2021.
URL `https://pypi.org/project/pydobot/`

[4] Pymodbus 2.5.3. 2021.
URL `https://pypi.org/project/pymodbus/`

[5] Python library for Dobot Magician. 2021.
URL `https://github.com/luismesas/pydobot`

[6] Barrios-Aviles, J.; Rosado-Munoz, A.; Iakymchuk, T.; aj.: Powerlink and ethernet/ip comparison as robust industrial ethernet protocols. *IFAC-PapersOnLine*, ročník 50, č. 1, 2017: s. 363–368.

[7] BBraun: Operační technologie.

[8] Biegacki, S.; VanGompel, D.: The application of DeviceNet in process control. *ISA transactions*, ročník 35, č. 2, 1996: s. 169–176.

[9] Bligård, L.-O.: *Predicting mismatches in user–artefact interaction. Development of an analytical methodology to support design work*. Chalmers University of Technology, 2012.

[10] Bothamley, K.; Rodgerson, J.: Emerging Ethernet Protocols. *IEEE article*, 2001.

[11] Bozdal, M.; Samie, M.; Jennions, I.: A survey on can bus protocol: Attacks, challenges, and potential solutions. In *2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, IEEE, 2018, s. 201–205.

[12] Brooks, P.: Ethernet/IP-industrial protocol. In *ETFA 2001. 8th International Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 01TH8597)*, ročník 2, IEEE, 2001, s. 505–514.

[13] Byres, E.: SCADA Security Basics: SCADA vs. ICS Terminology. *Tofino Security*, ročník 5, 2012.

[14] Control Engineering Česko: Propojení dat IT a OT. 2020.

[15] Cosman, E.: ICS, IACS, SCADA And So On: Do The Abbreviations Matter? 2021, (Industry trends category).

[16] Costa, D. G.; Guedes, L. A.: A discrete wavelet transform (DWT)-based energy-efficient selective retransmission mechanism for wireless image sensor networks. *Journal of Sensor and Actuator Networks*, ročník 1, č. 1, 2012: s. 3–35.

[17] Council of the European Union: Proposal for a Regulation of the European Parliament and of the Council laying down measures for a high common level of cybersecurity at the institutions, bodies, offices and agencies of the Union - General approach. 2022, (ST 14128 2022 INIT).

[18] Council of the European Union: Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on horizontal cybersecurity requirements for products with digital elements and amending Regulation (EU) 2019/1020. 2022, (COM/2022/454 final).

[19] Council of the European Union: Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on horizontal cybersecurity requirements for products with digital elements and amending Regulation (EU) 2019/1020. 2022, (ST 12429 2022 INIT).

[20] Darroudi, S. M.; Gomez, C.: Bluetooth low energy mesh networks: A survey. *Sensors*, ročník 17, č. 7, 2017: str. 1467.

[21] Desai, R.: *Integration of User Experience (UX), Customer Experience (CX) and Brand Experience (BX) with B2B and B2C Models*. IEEE, 2017.

[22] Dias, A. L.; Sestito, G. S.; Turcato, A. C.; aj.: Panorama, challenges and opportunities in PROFINET protocol research. In *2018 13th IEEE International Conference on Industry Applications (INDUSCON)*, IEEE, 2018, s. 186–193.

[23] Drahoš, P.; Bélai, I.: The PROFIBUS protocol observation. *IFAC Proceedings Volumes*, ročník 45, č. 11, 2012: s. 258–263.

[24] Emake, E. D.; Adeyanju, I. A.; Uzedhe, G. O.: Industrial Control Systems (ICS): Cyber attacks & Security Optimization. *International Journal of Computer Engineering and Information Technology*, ročník 12, č. 5, 2020: s. 31–41.

[25] Ghaffoori, S.: Unicast Multicast Broadcast Anycast and Incast Traffic Types. *OrhanErgun*, 2021.

[26] Gomolka, Z.; Zeslawska, E.; Twarog, B.; aj.: Use of a DNN in Recording and Analysis of Operator Attention in Advanced HMI Systems. *Applied Sciences*, ročník 12, č. 22, 2022: str. 11431.

[27] He, Q.-F.; Zeng, Q.-J.; Tang, X.-M.: Research and implement on industry control networks based on embedded SERCOS-III protocol. In *2011 International Conference on Electronics, Communications and Control (ICECC)*, IEEE, 2011, s. 3868–3872.

[28] IEC: Industrial communication networks – Network and system security – Part 1-1: Terminology, concepts and models. 2009.

[29] ISA, A.: ISA-99. 00. 01-2007 Security for Industrial Automation and Control Systems Part 1 Terminology, Concepts, and Models. *International Society for Automation*, ročník 10, 2007.

[30] Johnson, G.: Determinism in industrial ethernet: A technology overview-part 2. *Process Technology*, 2009.

[31] KADLEC, K.; KMINEK, M.: Měřící technika. *Vysoka skola chemicko-technologicka. Praze*, 2005.

[32] Lara, P.; Sánchez, M.; Villalobos, J.: Enterprise modeling and Operational Technologies (OT) application in the oil and gas industry. *Journal of Industrial Information Integration*, 2020: str. 100160.

[33] Li, J.-Q.; Yu, F. R.; Deng, G.; aj.: Industrial internet: A survey on the enabling technologies, applications, and challenges. *IEEE Communications Surveys & Tutorials*, ročník 19, č. 3, 2017: s. 1504–1526.

[34] Lian, F.-L.; Moyne, J. R.; Tilbury, D. M.: Ethernet, ControlNet, and DeviceNet. *IEEE Control Systems Magazine*, 2001.

[35] Mandal, R.; Maity, T.; Chaulya, S.; aj.: Automation of underground coal mines using PLC. *Journal of Mines, Metals and Fuels*, 2016: s. 174–180.

[36] Meessen, C.: More on stream versus block oriented protocol. *Distributed Information System*, 2007.

[37] Michael, F.: Nejzranitelnějším článkem Průmyslu 4.0 jsou lidé. Rozhovor s Tomášem Froňkem (SIEMENS) o rizikách automatizace a digitalizace. 2021.

[38] Neha, T.: Connection Oriented and Connectionless Services. *Binary Terms*, 2019.

[39] Noguchi, S.; Suzuki, K.; Chino, S.; aj.: FDT technology for CC-link network. In *SICE Annual Conference 2011*, IEEE, 2011, s. 1560–1565.

[40] Patel, D. N.; Somani, B.: A Review on Implementation of MODBUS Communication Protocol and its Applications. *International Journal of Electronics Engineering Research. ISSN*, 2017: s. 0975–6450.

[41] Phinney, T.: IEC 62443: Industrial network and system security. *Last accessed July*, ročník 29, 2013.

[42] Schweller, K.: Apes with apps. *IEEE Spectrum*, ročník 49, č. 7, 2012: s. 38–45.

[43] Seferagić, A.; Famaey, J.; De Poorter, E.; aj.: Survey on wireless technology trade-offs for the industrial internet of things. *Sensors*, ročník 20, č. 2, 2020: str. 488.

[44] Stouffer, K.; Lightman, S.; Pillitteri, V.; aj.: Guide to industrial control systems (ics) security–nist special publication (sp) 800-82 revision 2. *NIST, Tech. Rep*, 2015.

[45] Sundararajan, A.; Chavan, A.; Saleem, D.; aj.: A survey of protocol-level challenges and solutions for distributed energy resource cyber-physical security. *Energies*, ročník 11, č. 9, 2018: str. 2360.

[46] Švéda, M.; Vrba, R.: ASI Interconnectivity. *IFAC Proceedings Volumes*, ročník 30, č. 7, 1997: s. 569–574.

[47] Swales, A.; aj.: Open modbus/tcp specification. *Schneider Electric*, ročník 29, 1999: s. 3–19.

[48] Wang, Q.; Liu, X.; Chen, W.; aj.: Building robust wireless LAN for industrial control with the DSSS-CDMA cell phone network paradigm. *IEEE Transactions on Mobile Computing*, ročník 6, č. 6, 2007: s. 706–719.

[49] Willig, A.; Matheus, K.; Wolisz, A.: Wireless technology in industrial networks. *Proceedings of the IEEE*, ročník 93, č. 6, 2005: s. 1130–1151.

[50] Willis, M. J.; Tham, M. T.: Advanced process control. *Department of Chemical and Process Engineering, University of Newcastle Upon Tyne, UK*, 1994.

[51] Zhang, P.: *Advanced industrial control technology*. William Andrew, 2010.

[52] Česká Republika: Zákon č. 240/2000 Sb.: Zákon o krizovém řízení a o změně některých zákonů (krizový zákon). *Sbírka zákonů České republiky*, 2000.

Seznam symbolů, veličin a zkratek

ANSI	Národní Standardizační Institut
APS	Pokročilé plánování
ARI	Rozhraní rozšířené reality
BAN	Síť v blízkosti těla
BX	Zkušenost se značkou
CAN	Kampus síť
CAPEX	Kapitálové náklady
CLI	Rozhraní příkazového řádku
CPS	Kontinuální procesní řízení
CUI	Kompozitní uživatelské rozhraní
CX	Uživatelská zkušenost
DCS	Systém distribuovaného řízení
DMZ	Demilitarizovaná zóna
DPC	Diskrétní procesní řízení
EAM	Řízení údržby
ERP	Plánování podnikových zdrojů
FMI	Rozhraní žena-stroj
GAN	Globální síť
GUI	Grafické uživatelské rozhraní
HAN	Domácí síť
HCI	Rozhraní člověk-počítač
HMI	Rozhraní člověk-stroj
HSI	Rozhraní člověk-systém
HUI	Uživatelské rozhraní využívající hologramu
HVI	Rozhraní člověk-vozidlo
HW	Hardware
IA	Průmyslová automatizace
IACS	Průmyslová automatizace a řídící systémy
IAN	Internetová síť
ICS	Průmyslové řídící systémy
IEC	Mezinárodní elektrotechnická komise
IIoT	Průmyslový Internet věcí
IoT	Internet věcí
ISA	Mezinárodní společnost pro automatizaci
ISO	Mezinárodní organizace pro normalizaci
IT	Informační technologie
JIS	Metody "Ve správném pořadí"

JIT	Metoda "Právě včas"
LAN	Lokální síť
M&CS	Výrobní a řídící systémy
MAC	Řízení přístupu k médiu
MAN	Metropolitní síť
Malware	Škodlivý software
MES	Řízení výroby
MMI	Rozhraní muž-stroj
MOM	Řízení provozu
MRP	
MTU	Centrální řídící jednotka
NFC	Komunikace v blízkém okolí
NFV	Virtualizace síťových funkcí
NIST	Národní institut standardů a technologií
OIT	Terminál operátorského rozhraní
OPEX	Provozní náklady
OT	Provozní technologie
PAN	Osobní síť
PLC	Programovatelný logický automat
PtP	Bod-bod
QMS	Řízení kvality
RTU	Vzdálená telemetrická jednotka
SCADA	Dispečerské řízení a sběr dat
SDN	Softwarově definované sítě
SPOF	Jediný bod selhání
SW	Software
TCP	Přenosový řídící protokol
TMS	Systém pro plánování a řízení přepravních zakázek
TUI	Uživatelské textové rozhraní
UDP	Uživatelský datagramový protokol
UI	Uživatelské rozhraní
UX	Uživatelská zkušenost
VRI	Rozhraní virtuální reality
WAN	Rozlehlá síť
WMI	Rozhraní žena-stroj
WMS	Řízení intralogistiky