



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV VÝROBNÍCH STROJŮ, SYSTÉMŮ A ROBOTIKY

INSTITUTE OF PRODUCTION MACHINES, SYSTEMS AND ROBOTICS

VIRTUÁLNÍ ZPROVOZNĚNÍ JEDNOÚČELOVÉ VÝROBNÍ BUŇKY

VIRTUAL COMMISSIONING OF ONE PURPOSE PRODUCTION CELL

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Martin Vala

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Vojtěch Štěpánek, Ph.D.

BRNO 2024

Zadání diplomové práce

| | |
|-------------------|--|
| Ústav: | Ústav výrobních strojů, systémů a robotiky |
| Student: | Bc. Martin Vala |
| Studijní program: | Výrobní stroje, systémy a roboty |
| Studijní obor: | bez specializace |
| Vedoucí práce: | Ing. Vojtěch Štěpánek, Ph.D. |
| Akademický rok: | 2023/24 |

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Virtuální zprovoznění jednoúčelové výrobní buňky

Stručná charakteristika problematiky úkolu:

Cílem práce je virtuálně zprovoznit jednoúčelovou výrobní buňku. Velký důraz je kladen na vytvoření IO signálové struktury odpovídající reálnému dvojčeti a také vytvoření signálového rozhraní pro připojení buňky do větších výrobních celků.

Cíle diplomové práce:

Rešerše dostupných řídicích systémů pro ovládání zvolené technologie.

Rešerše dostupných nástrojů pro virtuální zprovoznění.

Příprava modelu jednoúčelové měřicí stanice pro virtuální zprovoznění v prostředí vhodného CAD systému.

Vytvoření řídicího software.

Virtuální zprovoznění vytvořeného systému.

Závěr a doporučení pro praxi.

Seznam doporučené literatury:

MAREK, Jiří. Konstrukce CNC obráběcích strojů IV.0. Praha: MM publishing, 2018. MM speciál. ISBN 978-80-906310-8-3.

SHIGLEY, Joseph Edward, Charles R MISCHKE a Richard G BUDYNAS, VLK, Miloš, ed. Konstruování strojních součástí. V Brně: VUTIUM, 2010, xxv, 1159 s. Překlady vysokoškolských učebnic. ISBN 978-80-214-2629-0.

Online manuál pro konfiguraci PLC řídicího systému [cit. 3.10.2023]: infosys.beckhoff.com

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2023/24

V Brně, dne

L. S.

doc. Ing. Petr Blecha, Ph.D., FEng.
ředitel ústavu

doc. Ing. Jiří Hlinka, Ph.D.
děkan fakulty

ABSTRAKT

Diplomová práce se zabývá problematikou virtuálního zprovoznění a PLC programování. V první části je uvedena rešerše z oblasti řídicích systémů a programů pro virtuální zprovoznění. V druhé části je pozornost věnována převážně softwaru NX Mechatronics concept designer, kde probíhá prvotní úprava modelu a jeho nastavení pro virtuální zprovoznění. Po vlastním připravení modelu následuje virtuální propojení řídicího systému TwinCAT 3 a dříve zmiňovaného NX MCD přes virtuální OPC server. Poté je pomocí vývojového prostředí TwinCAT 3 vytvořen řídicí kód pro kontrolu jednotlivých operací. Následuje samotné virtuální zprovoznění modelu pro montáž ocelového páskovače.

ABSTRACT

Master thesis deals with the issue of virtual commissioning and PLC programming. In the first part of the thesis a research in the field of control systems and softwares for virtual commissioning is conducted. The second part of the thesis deals mostly with NX Mechatronics concept designer software, where the initial modification of the model and its configuration for virtual commissioning takes place. The actual preparation of the model is followed by the virtual connection of the TwinCAT 3 control system and previously mentioned NX MCD using a virtual OPC server. The control program code is then created using the TwinCAT 3 development environment to control the assembly operations. This is followed by the actual virtual commissioning of single purpose station for assembly of the steel strapping machine.

KLÍČOVÁ SLOVA

Virtuální zprovoznění, řídicí systém, PLC, TwinCAT, Siemens NX, Mechatronics Concept Designer

KEYWORDS

Virtual commissioning, control system, PLC, TwinCAT, Siemens NX, Mechatronics Concept Designer

BIBLIOGRAFICKÁ CITACE

VALA, Martin. *Virtuální zprovoznění jednoúčelové výrobní buňky*. Brno, 2024. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/157828>. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav výrobních strojů, systémů a robotiky. Vedoucí práce Vojtěch Štěpánek.

PODĚKOVÁNÍ

Tímto bych rád poděkoval svému vedoucímu diplomové práce panu Ing. Vojtěchu Štěpánkovi, Ph. D. za cenné rady a materiály, které mi velmi pomohli v orientaci v zadané problematice. Dále bych rád poděkoval své rodině a přátelům, kteří mě na mé prozatímní cestě podporovali.

ČESTNÉ PROHLÁŠ ENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Vojtěcha Štěpánka Ph.D. a s použitím literatury uvedené v seznamu.

V Brně dne 24.5.2024

.....

Bc. Vala Martin

OBSAH

| | | |
|-----------|--|-----------|
| 1 | ÚVOD..... | 15 |
| 2 | MOTIVACE | 17 |
| 3 | PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ | 19 |
| 3.1 | Řídicí systémy | 19 |
| 3.1.1 | Programovatelné řídicí kontroléry | 20 |
| 3.1.2 | Vývojová prostředí | 21 |
| 3.1.3 | Princip činnosti PLC | 23 |
| 3.2 | Virtuální zprovoznění | 24 |
| 3.2.1 | Koncept virtuálního zprovoznění | 24 |
| 3.2.2 | Simulační programy | 25 |
| 3.2.3 | Komunikační protokoly v rámci NX MCD..... | 28 |
| 4 | ANALÝZA PRACOVNÍHO MÍSTA..... | 33 |
| 4.1 | Ocelový páskovač BO-7 Swing | 34 |
| 4.2 | Objekty | 34 |
| 4.3 | Lisovací hlavy | 36 |
| 4.4 | Paletky | 38 |
| 4.5 | Šroubovací pracoviště | 39 |
| 4.6 | Lisovací pracoviště..... | 39 |
| 5 | VIRTUÁLNÍ ZPROVOZNĚNÍ..... | 41 |
| 5.1 | Beckhoff TwinCAT 3 OPC UA Server..... | 41 |
| 5.2 | NX – MCD | 42 |
| 5.2.1 | Základní fyzikální prvky | 43 |
| 5.2.2 | Vazby mezi tělesy | 46 |
| 5.2.3 | Kolizní senzory | 47 |
| 5.2.4 | Akční členy | 48 |
| 5.2.5 | Signály..... | 50 |
| 5.2.6 | Operace..... | 50 |
| 5.2.7 | Mapování signálů | 51 |
| 5.3 | Beckhoff TwinCAT 3..... | 52 |
| 5.3.1 | Definice proměnných | 52 |
| 5.3.2 | Motion | 53 |
| 5.3.3 | Safety – bezpečnost..... | 54 |
| 5.3.4 | Řídicí kód | 54 |
| 5.3.5 | HMI – uživatelské rozhraní..... | 55 |
| 6 | ZHODNOCENÍ A DISKUZE | 59 |
| 7 | ZÁVĚR..... | 61 |
| 8 | SEZNAM POUŽITÝCH ZDROJŮ..... | 63 |
| 9 | SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK..... | 65 |
| 9.1 | Seznam zkratk..... | 65 |
| 9.2 | Seznam tabulek..... | 65 |
| 9.3 | Seznam obrázků | 66 |
| 10 | SEZNAM PŘÍLOH..... | 69 |

1 ÚVOD

Virtuální zprovoznění je nedílnou součástí často skloňovaného termínu průmysl 4.0, který se zaměřuje na digitalizaci a automatizaci výrobních procesů. V kontextu digitalizace se můžeme setkat s pojmy jako jsou internet věcí (IoT), umělá inteligence (AI), smart senzory a virtualizace. S postupným rozvojem průmyslu 4.0 dochází ke zvyšování nároků na spolehlivost, bezpečnost, modulárnost a zkracování doby uvedení výrobních systémů do provozu. S řešením problematiky spolehlivosti a zkracování doby uvedení do provozu přichází možnost nasazení virtuálního zprovoznění.

V současnosti je většina konstrukční činnosti prováděna za pomoci 3D modelovacích softwarů. Právě takto vytvořené modely jsou pevným základem pro následující virtuální zprovoznění. Virtuální zprovoznění jako takové může probíhat současně se samotnou konstrukcí, kdy ji může pozitivně ovlivňovat lokalizováním chyb během návrhu. Mezi jednotlivé výhody patří možnost odsunutí materiálové investice až do stavu úspěšného virtuálního zprovoznění, kde bude garantován úspěch reálného zprovoznění. Další výhodou je odhalení konstrukčních chyb ještě ve virtuálním prostředí, kdy nedochází k fyzickému poškození. Zároveň vlivem dřívějšího zahájení zprovoznění dochází k odladění programových chyb souběžně s finalizací návrhu pracoviště a tím pádem ke zrychlení uvedení do provozu.

Cílem diplomové práce je shrnout současný stav poznání z oblasti dostupných řídicích systémů pro ovládání zadané technologie a nástrojů vhodných pro virtuální zprovoznění. Následně se diplomová práce zabývá úpravou poskytnutého 3D modelu robotizovaného pracoviště. Upravený model je převeden na model konkrétních jednoúčelových pracovišť. Po jeho úpravě dochází k překlopení do stavu vhodného pro virtuální zprovoznění pomocí NX Mechatronics concept designer. Dalším krokem práce je vytvoření řídicího systému ve vývojovém prostředí Beckhoff TwinCAT 3. V neposlední řadě dojde k propojení těchto dvou aplikací a samotnému virtuálnímu zprovoznění.

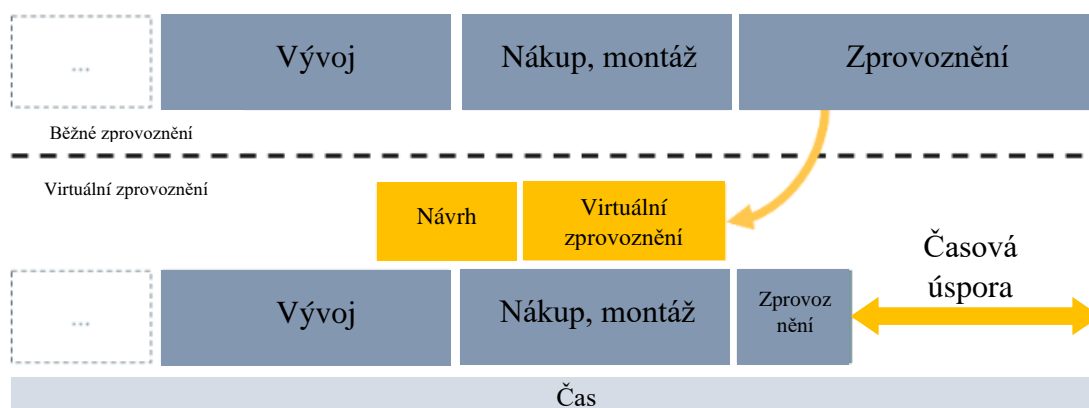
2 MOTIVACE

Motivací pro tuto práci mi byla šance dostat se do kontaktu s nejmodernějším způsobem zprovoznování výrobních systémů. Šance získat vědomosti v oblasti automatizace a PLC programování bez nutnosti materiálové investice do výroby a následně je uplatnit v rámci virtuálního zprovoznění.

Důležitým faktorem volby této práce byla i ukázka virtuálního zprovoznění robota za pomoci NX, TwinCAT 3 a Matlabu, která nám byla prezentována mým současným vedoucím práce panem Ing. Vojtěchem Štěpánkem Ph.D.

3 PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ

Vzhledem k rostoucímu průmyslovému zájmu a zaručení globální konkurenceschopnosti jednotlivých firem stejnoměrně stoupá i zájem o digitalizaci průmyslu. Neodmyslitelnou součástí digitalizace průmyslu se pak stává virtuální zprovoznění jednotlivých výrobních strojů, systémů nebo robotických buněk. Dosavadní průmyslové zkušenosti ukázaly výhodu možnosti použití softwarových aplikací pro usnadnění a zkrácení práce v poli. V ideálním světě by pak zprovoznující pracovník byl schopen přijet na kompletaci s plně odladěným a funkčním systémem a nenásledovalo by zdlouhavé ladění a testování funkčnosti systému. Možnost odladění řídicích systémů už při jeho vývoji, systému tak značně zkrátí doby uvádění do provozu spolu s odhalením dynamických a kinematických nedostatků a umožní změnu případného návrhu ještě před výrobou produktu. Schématickou ukázkou porovnání virtuálního a reálného zprovoznění lze vidět na obr. 1. [1] [2] [21]



Obr. 1 Zjednodušená ukáзка dopadu virtuálního zprovoznění [21] upraveno

3.1 Řídicí systémy

Řídicím systémem rozumíme systém, který řídí chování jiného průmyslového zařízení nebo výrobního systému. Řízení může být zprostředkováno ať už pomocí reléové logiky distribuovaného (řídicího) systému (DCS), mikrokontrolerů nebo například za pomoci programovatelných logických kontrolérů tzn. PLC systémů. Využíváme je například pro automatizované řízení výrobních cyklů, dosahování nebo kontrolovaného dodržování definovaných stavů a vlastností systémů nebo například jako kontrolu definovaného chování zařízení. [1] [2] [8]

Řídicí systémy využíváme v mnoha odvětvích jako je například automobilová výroba, automatizace výroby, řízení dopravy, řízení v energetice, řízení inteligentních domů, řízení 3D tiskáren, obráběcích a tvářecích strojů aj. Konkrétními příklady může být automatizace robotizované výrobní buňky, regulace teploty ve výrobní soustavě nebo budově, automatická kontrola rychlosti posuvových mechanismů, kontrola řízení křížících se dopravních tras, ovládací panel obráběcího centra nebo tvářecího stroje, řízení jednotlivých pohonů obráběcího stroje, přesné řízení polohování servopohonu například jednotlivých os u obráběcích strojů. V případě této práce budou brány v potaz primárně PLC

systemy. Mezi nejznámější výrobce průmyslových PLC patří například firmy Beckhoff, Siemens, Mitsubishi, Allen Bradley nebo B&R Automation viz tab. 1. [1] [2] [8]

Tab. 1 Výrobci jednotlivých PLC systémů [8]

| Výrobce | PLC řady | Vývojové prostředí |
|---------------------------------------|-----------------------------------|---------------------|
| Beckhoff | C60xx Ultra-compact Industrial PC | TwinCAT 3 |
| | C70xx Ultra-compact Industrial PC | |
| | CX5xxx Embedded PC | |
| | CX7xxx Embedded PC | |
| | CX9xxx Embedded PC | |
| Siemens | S7-1200/1500 | TIA Portal |
| | S7-400/300 | |
| Rockwell Automation, Allen Bradley | ControlLogix | Studio 5000 |
| | CompactLogix | |
| | MicroLogix | RSLogix500 |
| | SLC500 | |
| | PLC5 | RSLogix5 |
| Micro800 | | |
| B&R Automation, ABB | X20 | Automation studio 4 |
| | X90 | |

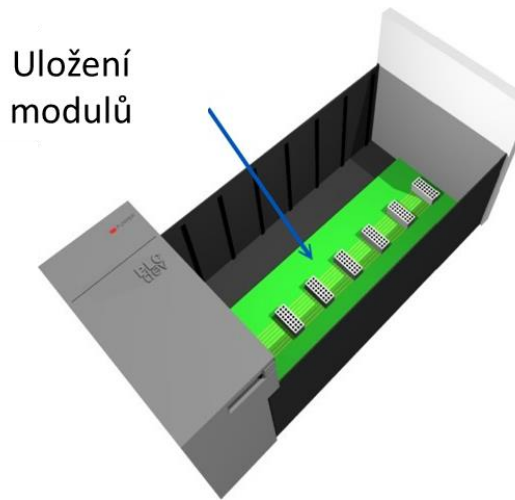
Beckhoff TwinCAT 3 funguje jako balíček nástrojů *Engineering + Runtime*, kdy za pomoci nástroje *Engineering* umožňuje programování a konfiguraci PLC. Nástroj *Runtime* pak slouží pro spuštění kódu v reálném čase nezávisle na operačním systému, což umožňuje funkčnost ostatních programů na PC, jako je například vizualizace pro virtuální zprovoznění. Další výhodou je nemalá zásoba dodatečných funkcí jako jsou HMI, *Measurement*, *Controller*, *Motion Control*, *Connectivity*, nebo například *Vision*. Jednotlivé balíčky výrazně ulehčují práci PLC programátorů za pomoci svých předem programovaných funkčních bloků. V případě *Motion Control* je možné využít knihovnu různých kinematických transformací, jako je například 2D H-bot, či 3D kinematická transformace pro delta robota nebo Stewardovu platformu. Knihovna dále nabízí lineární osy pro CNC systémy či manipulátory. Funkce *Connectivity* umožňuje interakci s jinými systémy. Tato funkce je obzvláště důležitá pro samotné virtuální zprovoznění, protože zaručí komunikaci mezi modelovacím, či simulačním softwarem a PLC kontrolérem. Dalším specifickým balíčkem je *Vision*, který umožní zpracování obrazu, a tedy i strojového vidění v rámci PLC řízení.[4] [7]

V případě výrobce Siemens je výše zmíněný *motion control* zastoupený za pomoci systému *SIMOTION*, který nabízí vysokou škálovatelnost rozsahu výkonu od 1 pohybové osy až po 128 os pro centralizované topologie. Strojové vidění je zastoupeno systémem *SIMATIC Machine Vision*, který stejným způsobem, jako v případě Beckhoff, zpřístupní selektivní pick-and-place manipulační operace, za použití strojového vidění nebo rozeznávání těles.[4] [7]

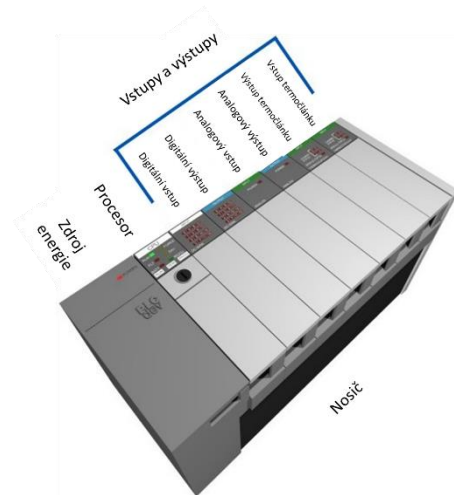
3.1.1 Programovatelné řídicí kontroléry

Programovatelný řídicí kontrolér – PLC je kompaktní, robustní průmyslový počítač cílený pro aplikace v průmyslovém prostředí. Využívá se pro řízení naprogramovaného procesu jednoho či více výrobních systémů nebo strojů z jednoho místa. PLC se skládá z nosiče,

který je připojen ke zdroji energie, na který jsou dále připojeny jednotlivé moduly. Jednotlivými moduly jsou inteligentní část – procesor, část pro analogové či digitální vstupy a výstupy, komunikační modul a vlastní paměť. Některé části lze vidět viz. obr. 2 a obr. 3. [1] [2]

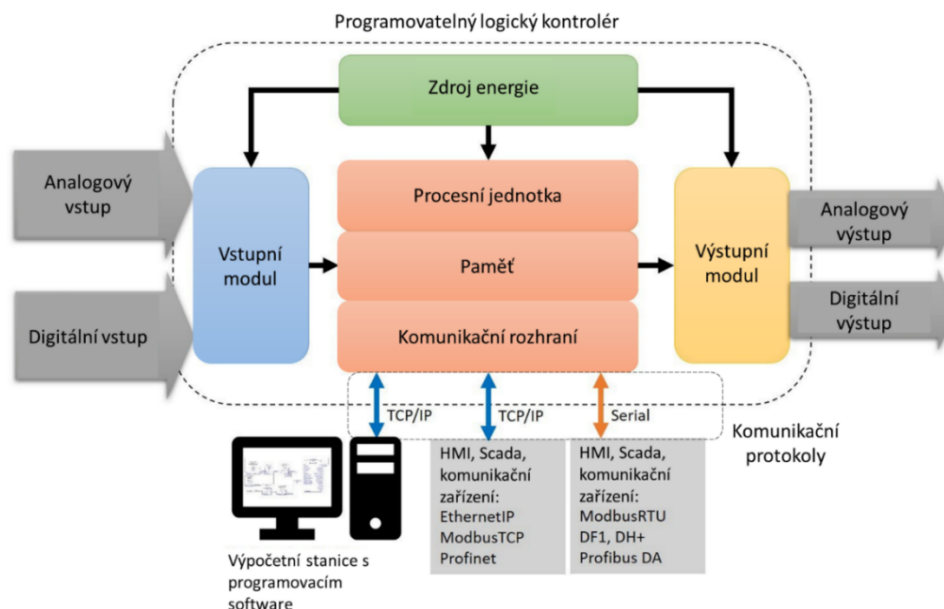


Obr. 2 Ukázka nosiče PLC, upraveno [2]



Obr. 3 Ukázka PLC, upraveno [2]

Obecná funkce PLC sestává z načtení vstupních analogových hodnot jako jsou teplota, tlak, vibrace, nastavení potenciometru nebo digitálních hodnot jako jsou zmáčknutí tlačítka, poloha posuvu, I/O informace od motoru. Následným krokem je uložení a zpracování signálů za pomoci procesní jednotky a jejich komunikace do periférií jako je HMI nebo SCADA. Posledním krokem je vyvození nastavených výstupů podle naprogramovaného chování. Celý proces lze vidět na obr. 4. [1] [2]



Obr. 4 Architektura PLC, upraveno [2]

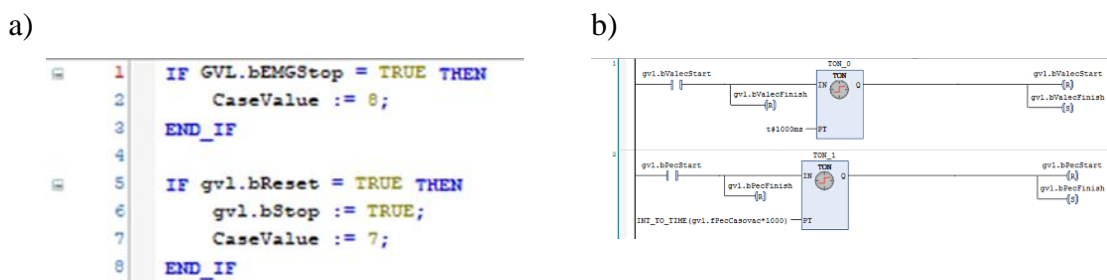
3.1.2 Vývojové prostředí

Nedílnou součástí PLC systémů je samotné vývojové prostředí označované jako IDE (Integrated development environment), které umožňuje psaní samotného řídicího kódu. Obecně vývojové prostředí PLC systémů podléhá orientační normě IEC 61131-3, která

umožňuje psaní řídicího kódu převážně za pomoci pěti různých metod: strukturovaný text, žebříkový diagram, funkční bloková schémata, seznam instrukcí a sekvenční funkční diagram, jak lze vidět na obr. 5 a obr. 6. [3] [5]

Strukturovaný text, viz. obr. 5a, je posloupný zápis symbolických instrukcí podobajících se zápisu pomocí programovacích jazyků jako je Pascal nebo C. Svou strukturou je zmíněnému Pascalu velmi podobný. Výhodným se strukturovaný text stává převážně při práci s daty, řetězci či databázemi nebo pro programování algoritmů. Lze jej použít například pro realizaci digitálních filtrů, linearizaci signálů nebo pro práci s textovými řetězci. Nevýhoda strukturovaného textu je pak oproti jiným metodám v nepřehlednosti toku struktury programu a nutnost znát příkazy a syntaxi zápisu. [3] [5]

Žebříkový diagram vychází z dob reléové logiky, kdy se místo logických obvodů používaly soustavy propojených relé. Svoji strukturou, jak lze vidět na obr. 5b, je žebříkový diagram vhodný pro přehledné zpracování velkého množství logických operací. Jeho výhodou jsou tedy převážně přehledný zápis nebo například možnost rychlého zorientování se v programu i pro běžného pracovníka. Žebříkový diagram není vhodný pro aritmetické operace, či práce s daty. Jeho využití lze najít v robustních, jednoduchých, lehce upravitelných programech, například v automobilovém průmyslu. [3] [5]



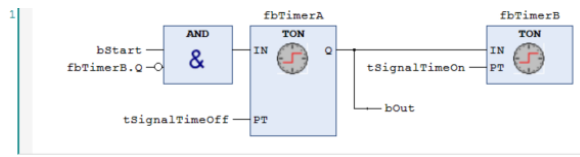
Obr. 5 a) Strukturovaný text ST, b) žebříkový diagram LD

V případě funkčních blokových schémat je kód realizován posloupně propojenými bloky. Tyto bloky realizují různé, předem definované funkce. Jedná se o jednoduchý způsob zápisu kódu, kdy k jednoduchým operacím mezi bloky dochází pomocí tzv. hradel. Termínem hradlo pak rozumíme funkční bloky typu AND či OR. Výhodou zápisu funkčními blokovými schématy je opět přehlednost samotného zápisu, či možnost zpracování velkého množství logických signálů bez ztráty přehlednosti. Ukázka zápisu funkčními blokovými schématy je k dispozici na obr. 6a [3] [5]

Seznam instrukcí, viz obr. 6b, je ve své podstatě nejzákladnějším způsobem psaní řídicího kódu. Jeho výhodou je možnost výkonové optimalizace PLC řízení, kdy programátorovi umožní optimalizaci jak týkající se velikosti programu, či rychlosti jeho vykonávání. [3] [5]

Poslední zmiňovanou metodou je sekvenční funkční diagram. Tento zápis umožňuje psaní kódu prostřednictvím všeobecně známého vývojového diagramu. Zápis je tedy vytvořen za pomoci větvení kódu na základě splnění rozhodovacích podmínek. Tento zápis se využívá převážně jako nadstavba ostatních programů, kdy jsou nejdříve vytvořeny jednotlivé podprogramy, které jsou následně uspořádány do vývojového diagramu. Výhodou tohoto zápisu je přehlednost zápisu, nebo práce s definicí různých stavů programu. Ukázka tohoto zápisu je k vidění na obr. 6c. [3] [5]

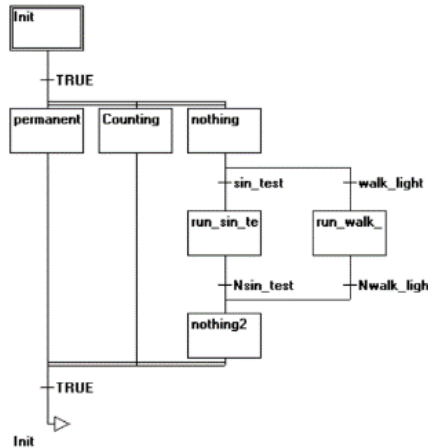
a)



b)

```
LD 17
ST lint (* comment *)
GE 5
JMPC next
LD idword
EQ instruct.sdword
STN test
next:
```

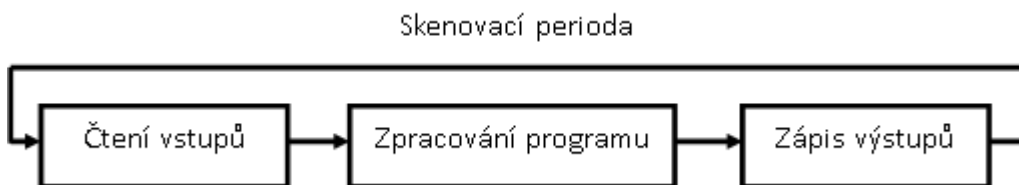
c)



Obr. 6 a) Funkční bloková schémata FBD, b) Seznam instrukcí IL, c) Sekvenční funkční diagram SFC

3.1.3 Princip činnosti PLC

Od programovatelného logického automatu, jak z jeho názvu vyplývá, se očekává, že bude naprogramované činnosti – práce, vykonávat samočinně. Samotná činnost PLC je vykonávána v jednotlivých cyklech, takzvaných skenovacích periodách, kdy dochází k cyklickému vykonávání programové smyčky v určitém časovém intervalu. V rámci skenovací periody dochází k přečtení vstupů, zpracování samotného řídicího programu a následnému zapsání výstupů viz obr. 7. [6]

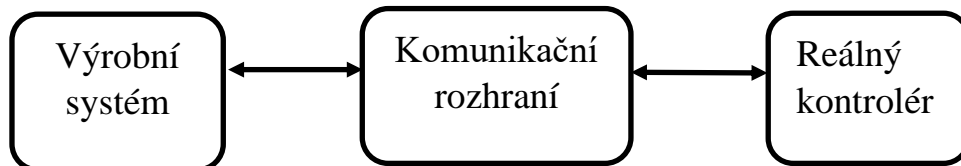


Obr. 7 Skenovací perioda

Mezi jednotlivé úrovně řízení PLC pak patří ruční, přímé či zpětnovazební řízení. Ruční řízení spočívá v interakci člověka se systémem, kdy člověk sleduje aktuální výstupní hodnoty, které kontroluje a na jejíž popud patřičně reaguje. V případě přímého řízení je člověk z části nahrazen řídicím systémem. Metoda spočívá v zadání vstupních a výstupních parametrů a dohlížení na způsob a stav vykonání programu. Poslední úrovní je pak zpětnovazební řízení, kdy systém sám sobě poskytuje informaci o skutečném stavu řízené soustavy. Zpětná vazba je poskytována pomocí senzorů a nástrojů pro diagnostiku. Systém tedy na základě informací zpětné vazby provede akční zásah, kterým řídí chování skutečné soustavy. [6]

3.2 Virtuální zprovoznění

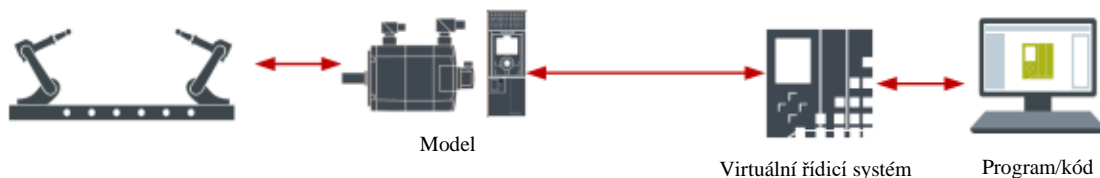
Zprovozněním výrobního systému obecně myslíme jeho uvedení do stavu, kdy bude schopný splňovat svou funkci. Obecně tedy potřebujeme pro zprovoznění výrobní systém, řídicí systém a způsob vzájemné komunikace mezi nimi. Výrobním systémem pak můžeme chápat například robotické svařovací pracoviště nebo montážní linku. Pod řídicím systémem si lze představit kontroléry zmiňované v předchozí kapitole. Vzájemná komunikace je pak zajištěna za pomoci komunikačního rozhraní jako je PROFINET, PROFIBUS nebo například IO-Link. Názorná, schématická ukázka komunikace k vidění na obr. 8.



Obr. 8 Ukázka komunikace výrobního systému

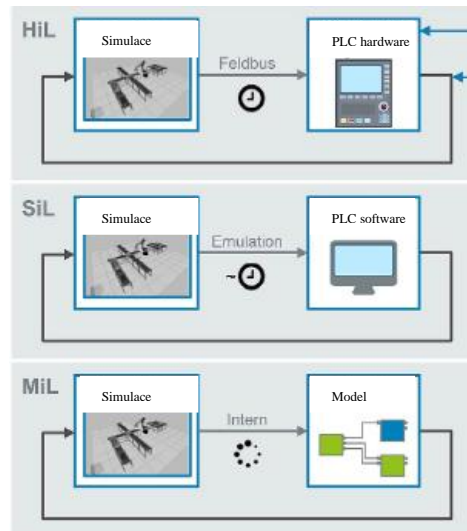
3.2.1 Koncept virtuálního zprovoznění

V koncepci virtuálního zprovoznění pak hovoříme také o 3 různých entitách – virtuální model řízeného výrobního systému, kontrolér (softwarový či hardwarový) a komunikační server. Výrobní systém je simulován CAD virtuálním prostředím, zpracovaného v softwaru umožňující simulaci jednotlivých operací prováděných na pracovišti. Takové softwary jsou například Siemens Process Simulate, ABB Robot Studio nebo NX s nástavbou Mechatronics concept designer aj.



Obr. 9 Koncept virtuálního zprovoznění, upraveno [16]

Kontrolérem se rozumí klasický řídicí systém značky Beckhoff, Siemens aj., ať už v podobě real-time vývojového prostředí TwinCAT 3, TIA Portal, nebo konkrétního fyzického hardware jednotlivých výrobců. Komunikace v případě virtuálního prostředí už neprobíhá pomocí sběrnice, ale přes komunikační server vytvořený přímo na pracovní stanici (PC), či za pomoci přítomného fyzického hardware. Proces virtuálního zprovoznění tedy můžeme dle předchozího popisu rozdělit podle podoby řídicího systému na několik různých metod, jako jsou Model in the loop (MiL), Software in the loop (SiL), Hardware in the loop (HiL), jednotlivé metody jsou vidět na obr. 10. [9] [10]



Obr. 10 HiL, SiL, MiL, upraveno [10]

Prvním krokem metody „model in the loop“, virtuálního zprovoznění je správné nastavení fyzikálních parametrů a aktivních prvků. V tomto kroku zprovozníme všechny parametry a akční prvky tak, aby následná simulace bez řídicího systému odpovídala realitě. Může docházet například k testování dosahu robotů, kolizím, či prvotní zkoušce funkčnosti materiálového toku. V rámci této metody probíhá testování vlastností simulace a kontrola její korelace s realitou. Následuje metoda „software in the loop“, označována zkratkou SiL, kde dochází k připojení kontroléru pomocí výše zmíněného virtuálního serveru. Během tohoto kroku dojde k provázání proměnných signálů v simulaci a otestování funkčnosti softwarového řídicího systému na virtuálním modelu. Poslední metodou je pak připojení hardwarového řídicího systému k virtuálnímu modelu pracoviště a konečná úprava, či kontrola správnosti řídicího systému. Dalo by se tedy říci, že v případě metody „model in the loop“ značené zkratkou MiL, je řídicím systémem samo virtuální prostředí, ve kterém je simulováno. [10]

Metoda „software in the loop“ je pak reprezentována emulací řídicího systému přebírajícího kontrolu nad virtuálním modelem. Z toho vyplývá, že je na PC spuštěn virtuální model, který je zároveň řízen ze stejné či jiné PC stanice pomocí softwarového PLC.[10]

V posledním případě je řídicí systém ovládán přímo za pomoci fyzického hardwaru – „hardware in the loop“. Jedná se tedy o metodu, kdy je virtuální model na PC, ke kterému je připojené hardwarové PLC s nahraným programem. Jednotlivé metody lze vidět na obr. 10. [10]

3.2.2 Simulační programy

Neopominutelnou částí virtuálního zprovoznění je počítačový software zastávající funkci virtuálního simulačního prostředí. Hlavním nárokem na takový program je tedy schopnost fyzikálně interpretovat samotné simulace. Za pomoci tohoto programu je pak vytvořen model pracoviště, kinematické struktury akčních členů, materiálové vlastnosti jednotlivých těles a prvky logického ovládání. Mezi programy vhodné pro virtuální zprovoznění patří NX s nástavbou Mechatronics concept designer, dále MCD, Process simulate nebo Robot studio.


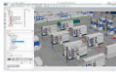





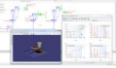




NX Mechatronics concept designer

MCD je nástavba CAD modelovacího software s názvem NX, vyvinuta společností Siemens. MCD, jehož ukázkou lze vidět na obr. 12, umožňuje multidisciplinární přístup k návrhu mechatronických systémů. Za pomoci tohoto programu lze zpracovat návrh systému kombinací mechanických, elektrických a případně softwarových komponent do jednoho celku. Navržený systém je pak možné simulovat a zjistit případné nedostatky v návrhu. Mezi jednotlivé výhody tohoto softwaru patří:

- Ucelené uživatelské rozhraní napříč všemi produkty Siemens
- Možnost importu víceméně všech CAD souborů díky NX bázi
- Multidisciplinární ucelení návrhu (elektrické rozvody, mechanické modely, simulace, signálová struktura)
- Událostně či časově řízený sled operací

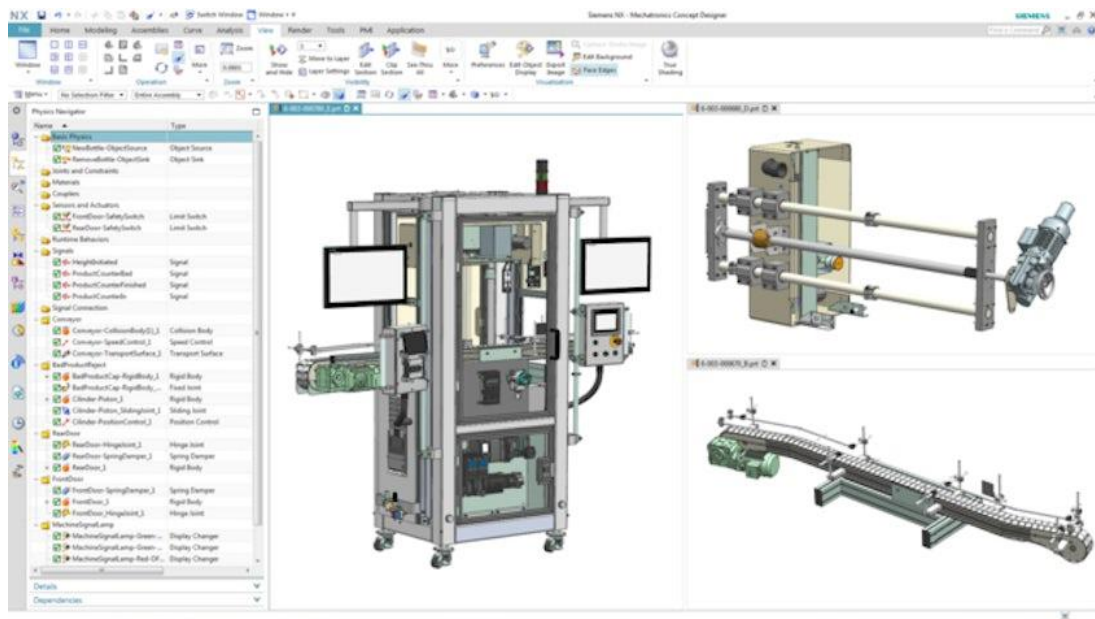
NX MCD je zároveň jedním ze sady simulačních programů Siemens balíčku digitální továrny Tecnomatix, viz. obr. 11, kde je každý program – Plant simulation, Process simulate i NX MCD optimalizován a používán pro své specifické odvětví. Ze zmíněného obrázku lze vidět, že funkcí NX MCD je důvěrná virtuální simulace výrobních strojů. [13]

Další z výhod je možnost propojení s virtuálními servery aplikací třetích stran a umožňuje vytvoření svého vlastního virtuálního serveru. Z toho vyplývá jeho nezávislost na komunikačních softwarech.

| | | | |
|------------------------|---|------------------------------------|---|
| Výrobní linka |  | Tecnomatix Plant Simulation |  |
| Robotická buňka |  | Tecnomatix Process Simulate |  |
| Výrobní stroj |  | NX Mechatronic Concept Designer |  |
| Fyzika těles |  | Simcenter Amesim |  |
| Komunikace mezi tělesy |  | SIMIT |  |
| Automatizace |  | PLCSIM Advanced and WinCC |  |

Obr. 11 Portfolio siemens pro digitální továrnu, upraveno [15]

Základními funkcemi je například definice „tuhého“ tělesa pro vlastní simulaci nebo stanovení „kolizního“ tělesa. Dále obsahuje funkce pro generování součástí tzv. „zdroj objektů“ a funkce pro jejich mazání tzv. „utopení objektů“. Základní vztahy mezi tělesy jsou definovány pomocí vazeb jako jsou pevná, posuvná, rotační, cylindrická, kulová a sledování bodu po trajektorii. Zároveň lze u jednotlivých těles předejít vzájemné kolizi, či tělesa vzájemně transformovat. [13]



Obr. 12 Ukázka NX MCD [13]

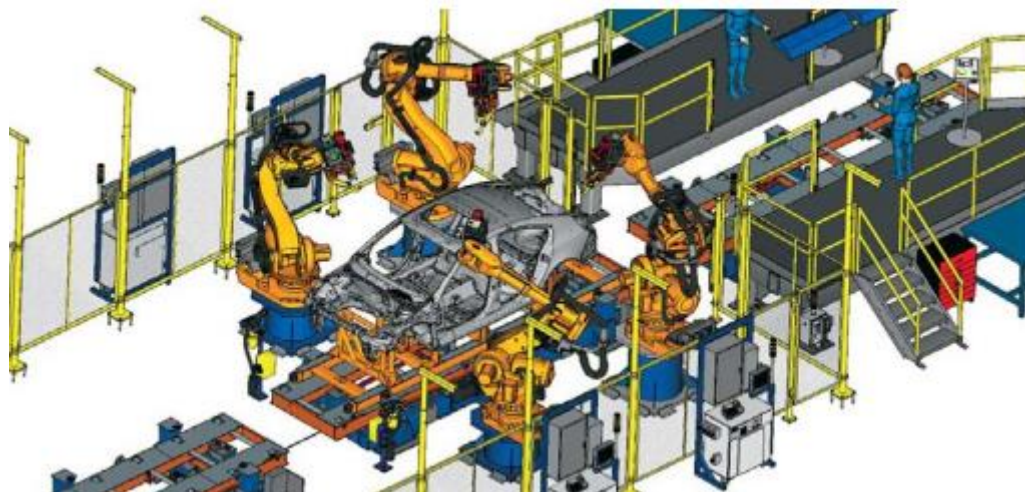
Process simulate

Dalším z balíčků Siemens digitální továrny Tecnomatix je Process simulate, dále jen PS. Tento program umožňuje digitální verifikaci návrhu výrobních procesů ve virtuálním prostředí. Zároveň poskytuje způsob ověření výrobního konceptu jak ve fázi návrhu, tak i v průběhu celého životního cyklu výrobku. Virtuální prostředí PS značně urychluje uvedení nových produktů na trh. Jednou z výhod je schopnost využívat simulovaná data výrobků pro optimalizaci taktu výroby.

PS, viz. obr. 13, umožňuje návrh robotizovaných výrobních buněk spolu s možností sledování celého životního cyklu výrobku. Umožňuje například návrh robotických svařovacích buněk, robotických lakovacích buněk či montážních pracovišť. Do takovýchto pracovišť lze přidávat i různá jednoúčelová pracoviště. V rámci Process simulate existuje i nástavba *PS Assembly*, za pomoci které je vývojářům umožněno optimalizovat montážní sekvenci spolu se zamezením kolizí a nežádoucích jevů. *PS Human* dále umožňuje ověřit návrh pracoviště z hlediska ergonomiky. Vývojář může virtuálně optimalizovat ergonomiku pracoviště pomocí nástrojů pro ověření dosahů atp. Kontrolní hodnoty v rámci PS odpovídají průmyslovým ergonomickým standardům.

Process simulate lze využít pro konfiguraci robotických svařovacích operací, „pick & place“ robotických operací nebo pro virtuální zprovoznění. Samotné virtuální zprovoznění může být uskutečněno pomocí PLC vývojového prostředí TIA Portal spolu s virtuálním OPC serverem. Mezi jednotlivé výhody tohoto softwaru patří: [11]

- Ucelené uživatelské rozhraní napříč všemi produkty Siemens
- Možnost OLP programování robotů
- Událostně či časově řízený sled operací
- Možnost zrychlení simulace
- Možnost off-line plánování robotických tras



Obr. 13 Ukázka Process simulate [11]

Robot studio

RobotStudio, dále RS, od společnosti ABB je nástroj využívaný pro off-line programování, optimalizaci a simulaci jednoho, či více průmyslových robotů spolu s jim přiřazenými periferiemi. Součástí produktu je rozsáhlá knihovna ABB robotů a jejich ovladačů. Do RS je možné importovat i vlastní 3D modely.

RobotStudio umožňuje vývojářům vizualizaci robotických aplikací ve virtuálním prostředí před vlastním uvedením do provozu. Software zároveň disponuje technologií virtuálního kontroléru, zaručující přesné polohování reálného, připojeného robota vůči virtuálnímu. V rámci simulací je možné přidat i funkcionalitu externích os například dopravníků nebo jiných funkčních prvků jako jsou senzory či jiné aktivní prvky. Zároveň je možné si do robota nahrát vlastní program skrze výše zmíněný kontrolér. Pro případné online programování existuje nástavba *RobotStudio Cloud*. Možná využití jsou od vlastního návrhu výrobní buňky, přes samotné programování a testování aplikací, skrze optimalizaci různých procesů, až po vzdělávání v oblasti robotiky a programování. [12]

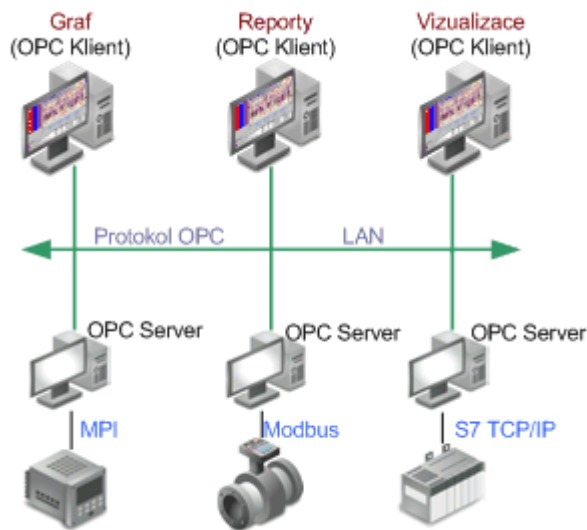
3.2.3 Komunikační protokoly v rámci NX MCD

V minulosti fungovala komunikace mezi jednotlivými zařízeními v systému tak, že musel každý hardware v systému mít vlastní speciální ovladač. Velké množství takto nainstalovaných ovladačů pak způsobovalo vzájemné ovlivňování komunikace a nekompatibilitu s operačním systémem. V případě přidání nového hardwaru bylo nutné upravit celý řídicí systém. V dnešní době dochází k využívání kombinace komunikačních sběrnic a komunikačních protokolů, které softwarově ošetřují způsob komunikace mezi zařízeními.

OPC UA/DA

V případě použití OPC se jediným komunikačním rozhraním stává právě OPC server, přes který putuje veškerá komunikace. Komunikační protokol tohoto serveru funguje na architektuře Klient – Server. Klientem se stává simulační program (NX MCD), který přijímá data od řídicího systému a dochází k využití metody SiL. OPC Server je pak virtuálně vytvořený server na lokálním PC. Protokol OPC tedy převádí data na jeden formát, který jsou schopny číst všechna připojená zařízení. Rozdílem mezi UA (Unified architecture) a DA (Data access) je způsob komunikace, kdy UA využívá průmyslového standardu ethernet pro komunikaci s různými operačními systémy PC a PLC. Na druhou

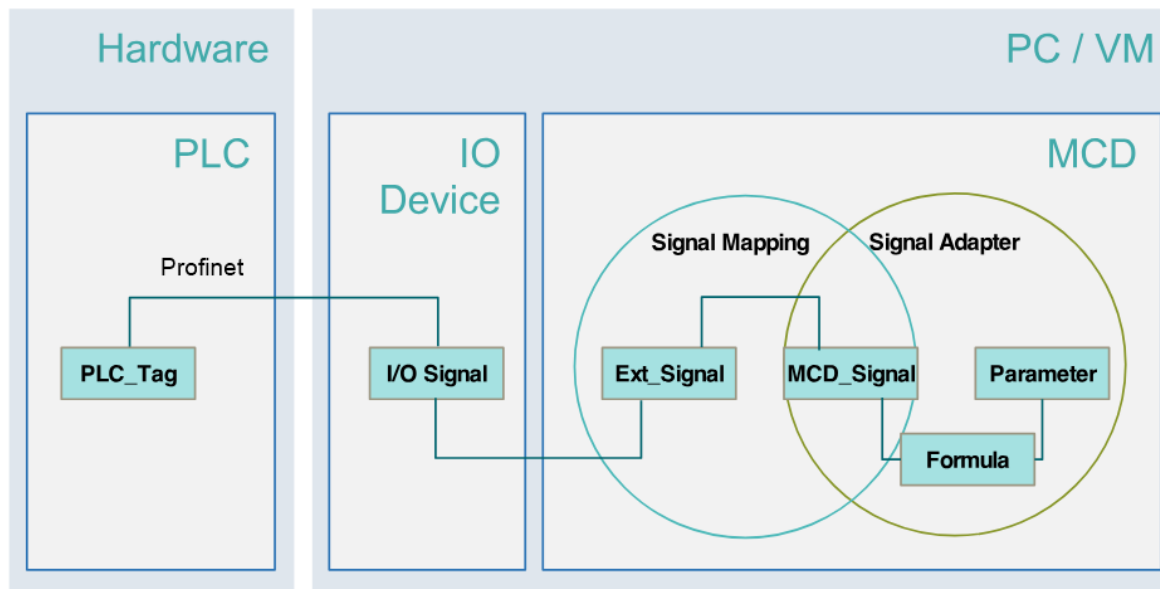
stranu DA využívá standardů COM/DCOM, který podporuje pouze operační systém Windows. Tímto formátem lze komunikovat I/O signály, vizualizaci aj., komunikační strukturu OPC lze vidět na Obr. 14. [16] [18]



Obr. 14 Komunikační struktura OPC [16]

Profinet

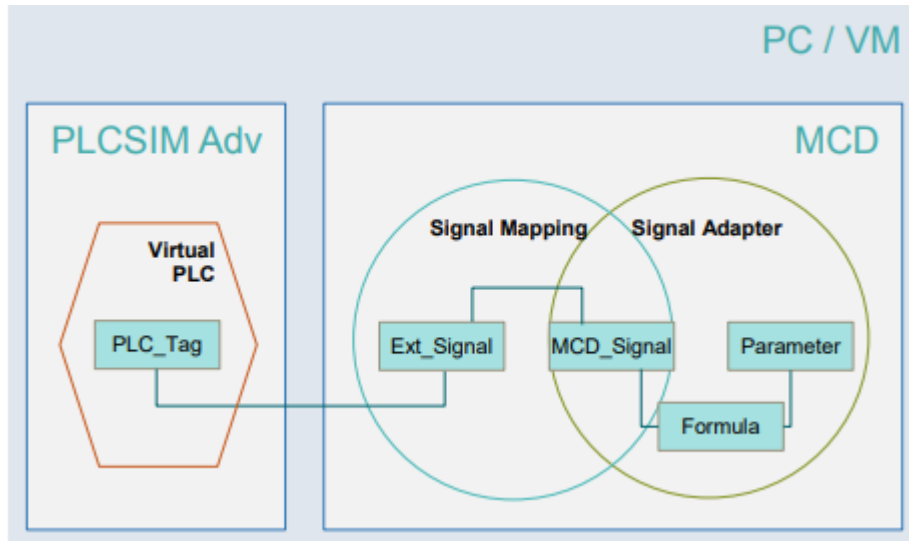
Profinet je ethernetový standard využívaný pro automatizaci. V případě použití komunikace přes Profinet, je nutné využít metody HiL, kdy je nutné připojit hardwarové PLC. Simulační software (NX MCD) běží jako aplikace na PC stanici, ke které je připojeno PLC ukázkou takového připojení lze vidět na obr. 15. [19]



Obr. 15 Komunikační struktura Profinet upraveno [19]

PLCSIM Advanced

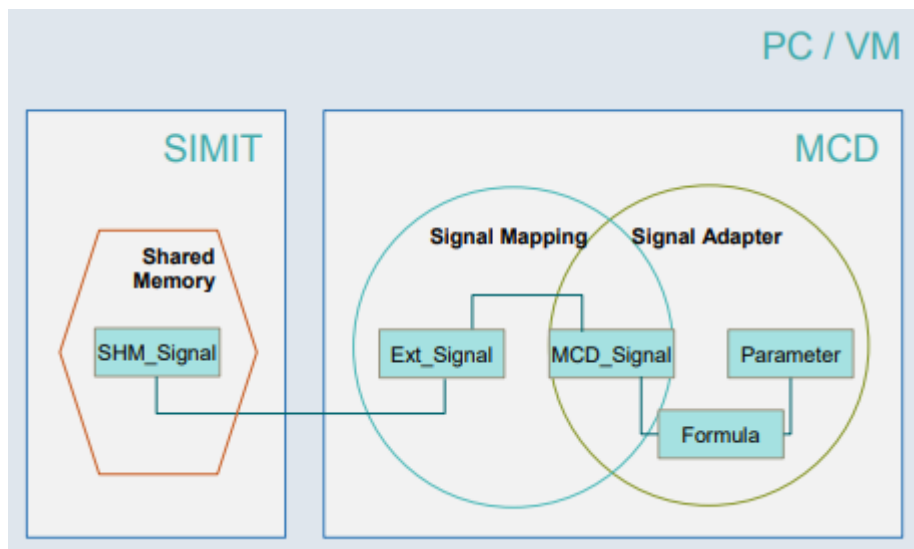
PLCSIM Advanced je aplikace společnosti Siemens, používaná pro virtuální nahrazení hardwarového PLC, která umožňuje pokročilou simulaci chování PLC tohoto výrobce. Zahrnuje komunikaci skrze OPC UA server, zároveň validuje správnost posílaných dat. Jedná se tedy o ucelený balíček k virtuálnímu propojení při virtuálním zprovoznování. Komunikační struktura PLCSIM Advanced je k nahlédnutí na obr. 16.



Obr. 16 Komunikační struktura PLCSIM Advanced [22]

SHM

Principem fungování komunikace SHM (Shared memory access) je vzájemné sdílení paměti mezi PLC a simulačním softwarem. Tato paměť pak funguje jako databáze s přístupovými omezeními. Komunikační struktura je ukázána na obr. 17.



Obr. 17 Komunikační struktura SHM [22]

TCP

Transmission control protocol je komunikační protokol pro sdílení dat mezi jednotlivými vzájemně připojenými prvky. Na rozdíl od UDP je TCP zatěžován dalšími funkcemi, jako je kontrola správnosti dat, korekce chyb nebo samotného příjmu dat. Při detekci chyby

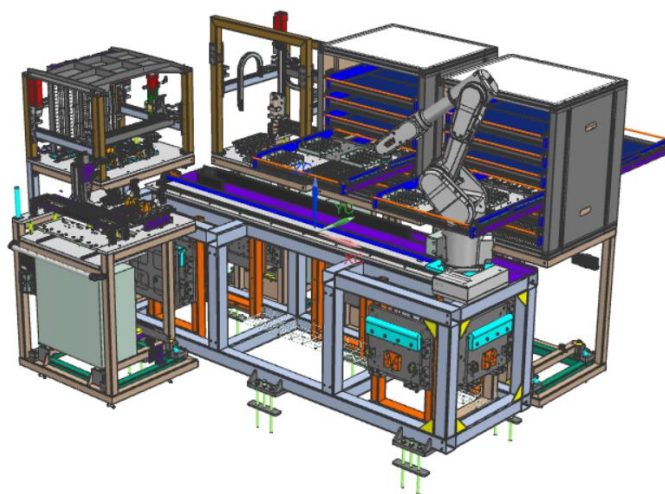
dochází k opětovnému zaslání signálu až do jeho správného přenesení. Z tohoto důvodu je TCP vhodný pro načítání aplikací náchylných na chybovost s nutností jejich kontroly. [20]

UDP

User datagram protocol je stejně jako TCP komunikační protokol pro sdílení dat mezi vzájemně připojenými prvky. V rámci protokolu UDP nedochází k rozsáhlé kontrole správnosti dat či jejich korekci. Z důvodu nenutnosti potvrzení akvizice dat je umožněn nepřetržitý tok dat od vysílače k příjemci. Z tohoto důvodu je UDP v porovnání s TCP rychlejším komunikačním protokolem na úkor správnosti posílaných dat. [20]

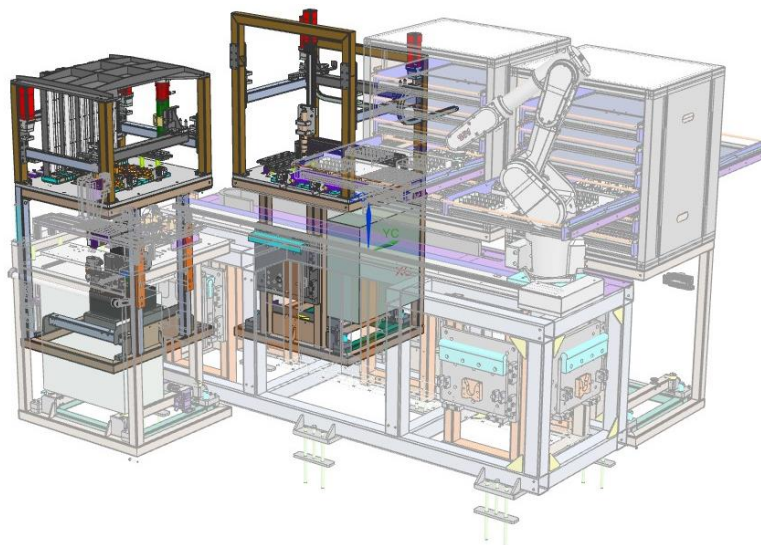
4 ANALÝZA PRACOVISTĚ

V rámci této diplomové práce je virtuálně zprovozněna část robotizovaného pracoviště navrženého firmou Feifer pro částečnou montáž ocelového páskovače, kterou lze vidět na obr. 18. Na pracovišti se dále nachází robot ABB, dopravník sloužící jako další osa robota, skladovací zásobníky a tři pracoviště. První pracoviště slouží pro šroubování pomocí robota. Druhé pracoviště slouží k lisování kluzných pouzder a kolíků. Posledním, třetím, pracovištěm je šroubovací buňka. Hlavní obslužnou entitou pracoviště je robot, který slouží k přenosu pracovních paletek mezi jednotlivými pracovišti.



Obr. 18 Ukázka výrobní buňky

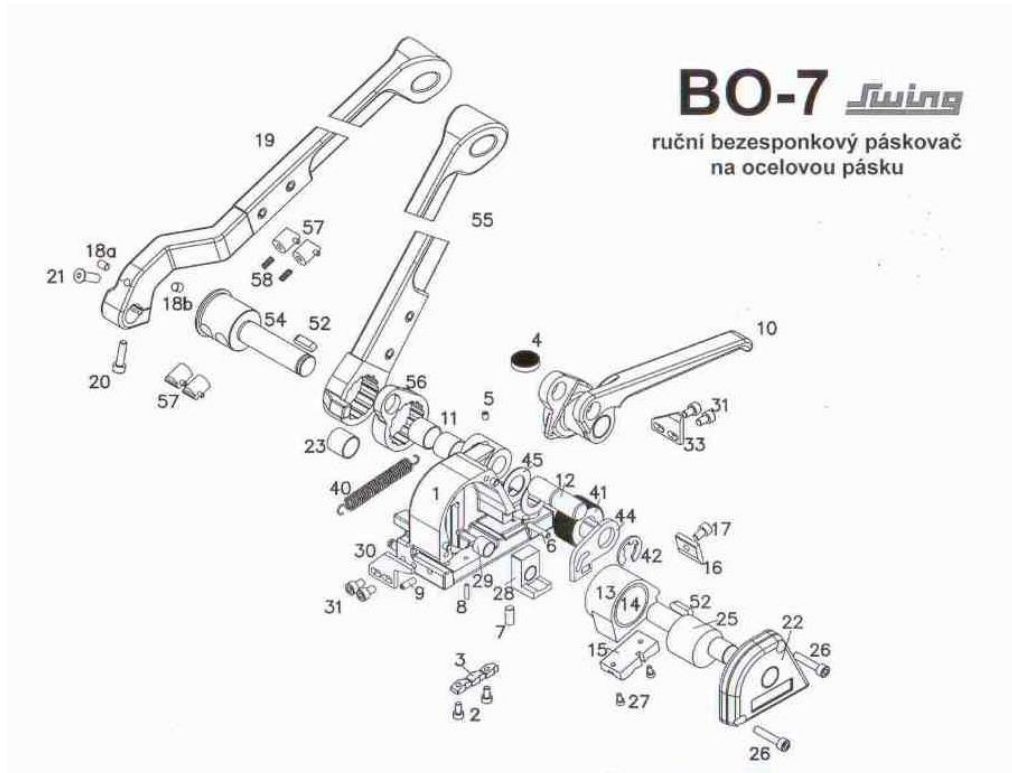
Poslední dvě zmíněná pracoviště lze vidět na obr. 19, z nichž nalevo je lisovací jednoúčelové pracoviště a napravo šroubovací jednoúčelové pracoviště. Tato pracoviště budou mírně upravena pro virtuální zprovoznění a následně zprovozněna.



Obr. 19 Ukázka zprovozňovaných buněk

4.1 Ocelový páskovač BO-7 Swing

Ocelový páskovač BO-7 Swing slouží k páskování ocelovou páskou o šířce 13 až 20 mm a tloušťce 0,4 až 0,8 mm. Páskovač, jehož rozpad lze vidět na obr. 20, funguje na patentovaném principu postupného stříhu zvaného SWING. Výhodou tohoto stříhu je snížení námahy obsluhy. Hlavním polem působnosti páskovače jsou předměty s jednou rovnou plochou jako jsou bedny, ztracená bednění nebo paletované zboží. Další z výhod páskovače je jeho relativně malá hmotnost při celokovové konstrukci. Maximální utahovací síla je 5000 N.



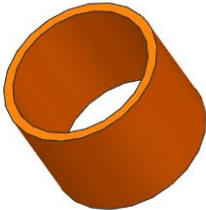
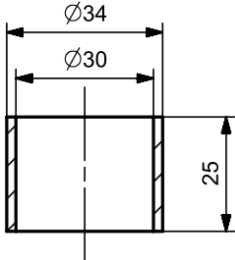
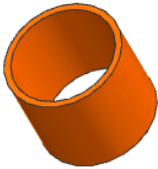
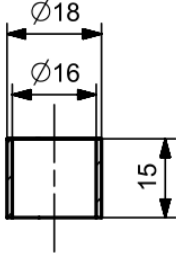
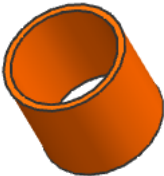
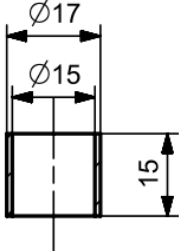

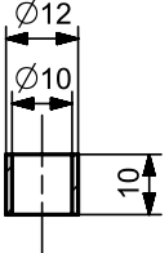
Obr. 20 Schématický obrázek páskovače BO-7 SWING

Na obr. 20 je třeba věnovat pozornost pouze některým dílům. Důležitými prvky jsou díly označeny čísly 1, 13, 22, 28, 30, 31 s 33. Bohužel jednotlivé objekty jsou v této fázi již nalisovány. Nicméně lze na obrázku vidět „těleso“ pod číslicí 1, „ojnici“ pod číslicí 13, „víčko“ pod číslicí 22 a „zvedáček“ pod číslicí 28, které spadají pod lisovací pracoviště, zatímco díly pod číslicemi 30 a 32 jsou střížné pásky a 31 jsou aretační šrouby, spolu s „tělesem“ jsou kompletovány na šroubovacím pracovišti. Jednotlivé operace budou popsány v kapitole 4.4.

4.2 Objekty


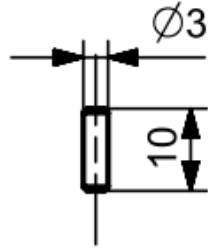

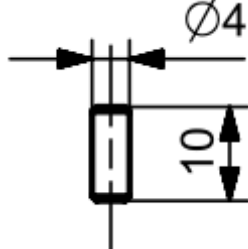

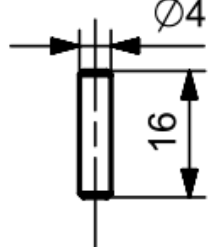

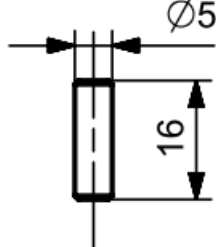

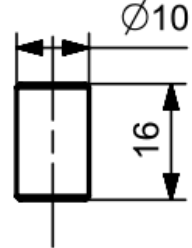
Mezi objekty pro zpracování na jednoúčelovém pracovišti patří kluzná pouzdra čtyř různých průměrů od nejmenšího 10 mm po největší 32 mm. Jejich konkrétní rozměry lze nalézt viz tab. 2.

Tab. 2 Zpracovávaná kluzná pouzdra

| Objekt | Náhled | Rozměry [mm] |
|----------------|---|---|
| Kluzná pouzdra |  |  |
| |  |  |
| |  |  |
| |  |  |

Další lisované objekty jsou zobrazeny viz. tab. 3, kde lze vidět jednotlivé kolíky. Rozměry kolíků jsou od 3 do 10 mm v průměru a v délkách 10 a 16 mm.

Tab. 3 Zpracovávané kolíky

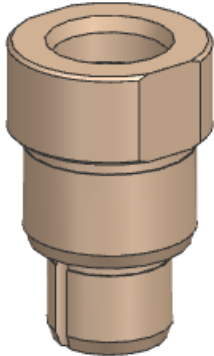
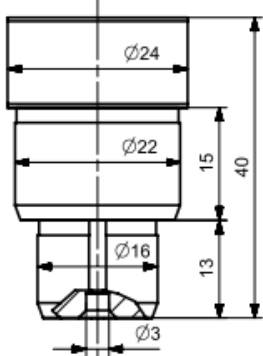
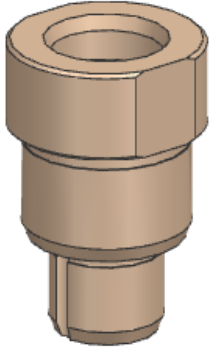
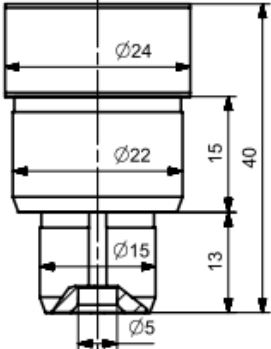
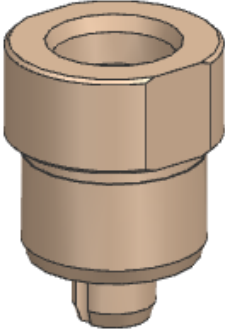
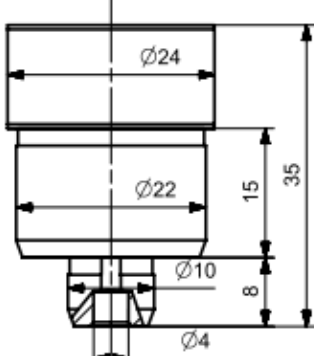
| | | |
|--------|---|--|
| Kolíky |  |  |
| |  |  |
| |  |  |
| |  |  |
| |  |  |

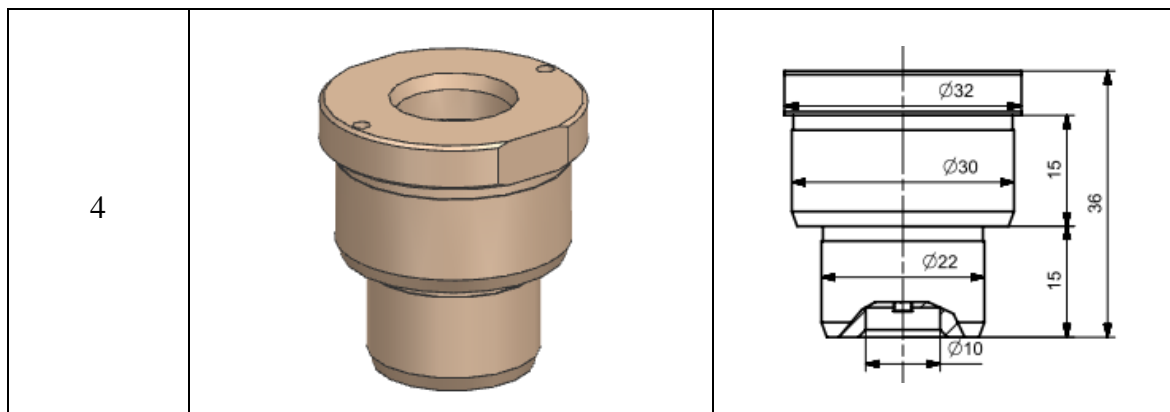
4.3 Lisovací hlavy

Dalším důležitým prvkem pracoviště jsou čtyři lisovací hlavy. Tyto hlavy jsou přichyceny na hydraulický válec a jsou tvarově vyrobeny tak, aby odpovídaly lisovaným součástem. Vnější průměr tedy odpovídá velikosti kluzných pouzder, zatímco vnitřní průměr odpovídá velikosti kolíků. Uvnitř jednotlivých hlav se nachází permanentní magnet, který má za úkol zaručit přítomnost objektu při přepravě a po celou dobu operace. Magnet musí vyvodit

dostatečnou přitažlivou sílu, aby udržel objekt na lisovací hlavě, ale zároveň musí dojít k zanechání objektu na tělesu po lisování. Jednotlivé lisovací hlavy lze vidět na tab. 4.

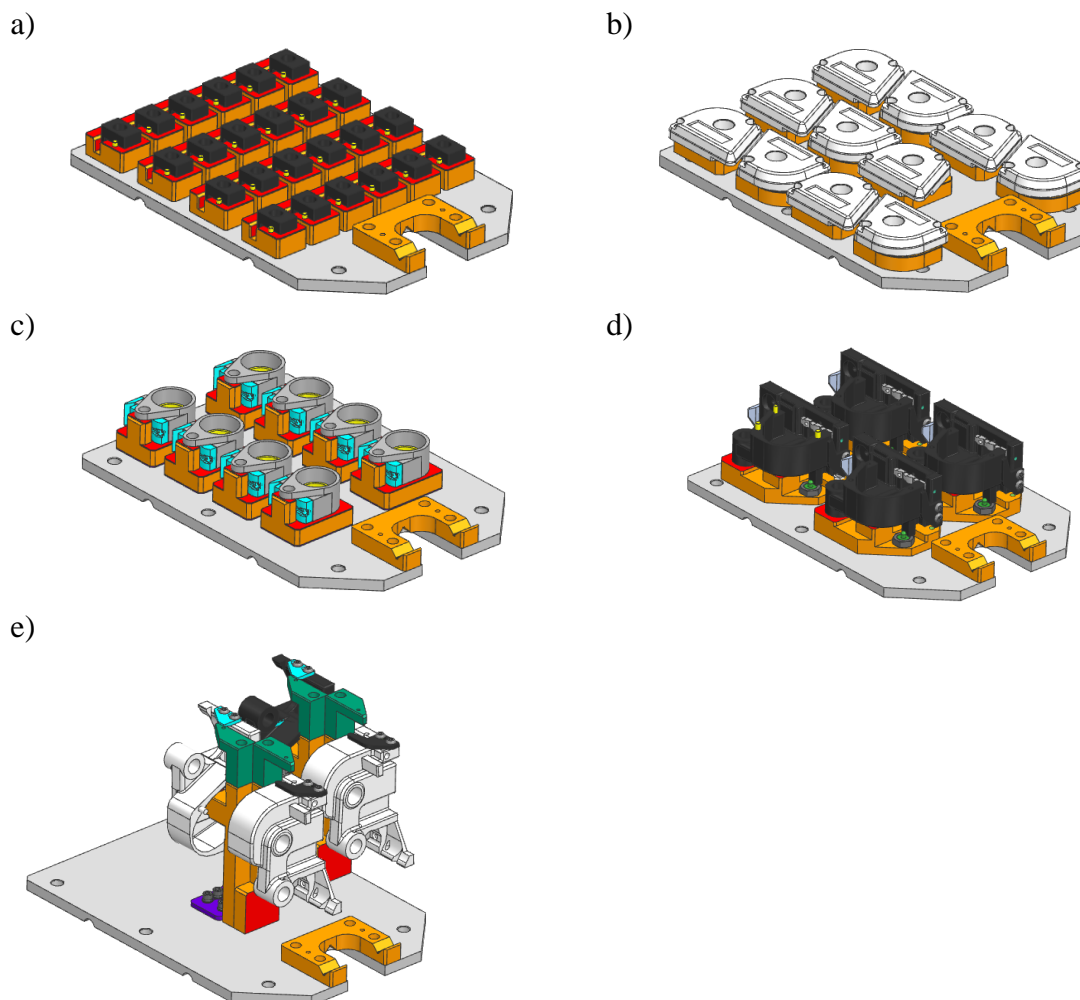
Tab. 4 Jednotlivé lisovací hlavy

| Lisovací hlava | Náhled | Ukázka rozměrů [mm] |
|----------------|---|---|
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |



4.4 Paletky

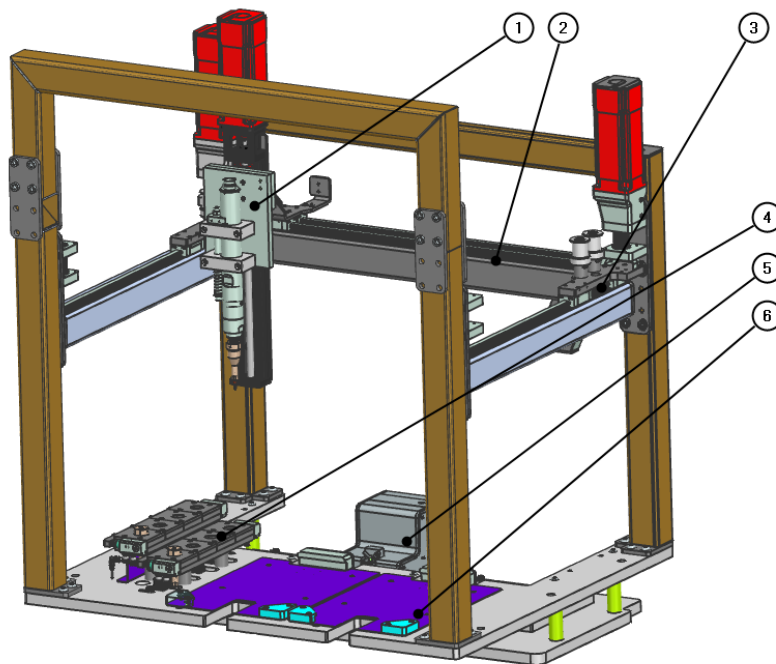
Pro transport jednotlivých těles po pracovišti se využívá pracovních paletek a manipulačního robota. Lisovací pracoviště a šroubovací pracoviště disponují dohromady pěti paletkami. Na jednotlivých paletkách jsou připevněna tělesa typu zvedáček, víko, ojnice a těleso. Jednotlivé paletky lze vidět viz obr. 21.



Obr. 21 Ukázka jednotlivých paletek a) zvedáček, b) víčko, c) ojnice, d) těleso, e) těleso šroubování

4.5 Šroubovací pracoviště

Šroubovací pracoviště disponuje mimo dvou translačních os (2), (3) také třetí pohybovou osou za pomoci kuličkového šroubu (1). Tato osa slouží k translačnímu, axiálnímu pohybu šroubovací hlavy pro operaci šroubování. Zároveň je šroubovací hlava opatřena rotační vazbou pro rotaci, kombinace translačního a rotačního pohybu pak vyvede šroubovací pohyb. Na šroubovacím pracovišti se nachází zásobník šroubovacích hlav (4), podavač šroubů M5 (5) a umístění paletek (6). Principem šroubovacího pracoviště je vyzvednutí šroubu z podavače šroubů a aretaci střížných ploch k tělesu. Manipulační pracovní prostor šroubového pracoviště disponuje pracovním prostorem o rozměrech 420 x 590 x 310 mm. Šroubovací pracoviště lze vidět viz obr. 22.

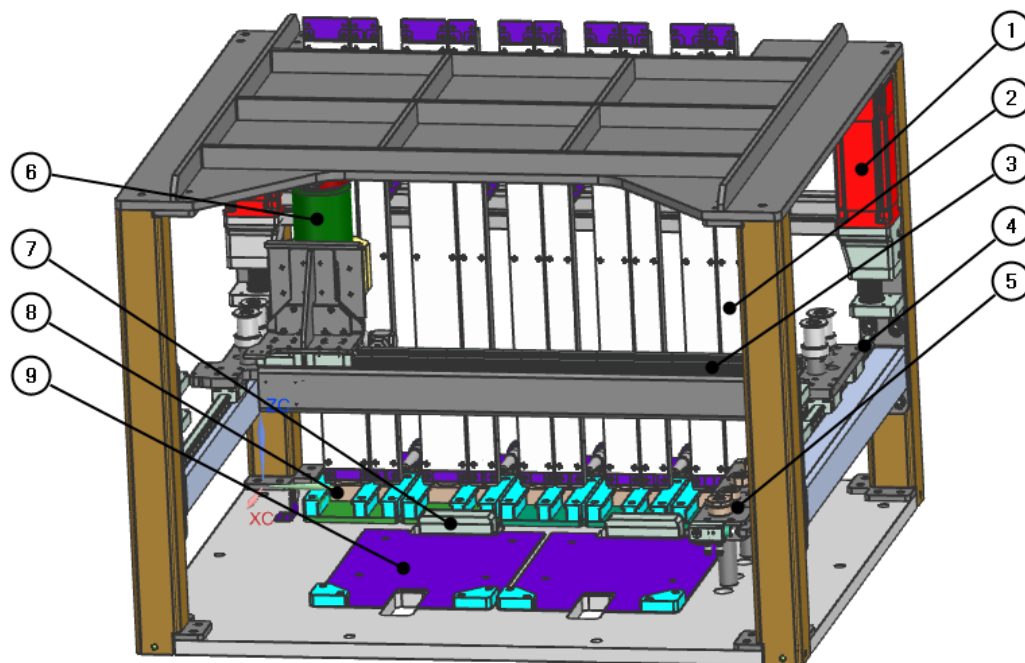


Obr. 22 Ukázka šroubovací stanice 1) Šroubovací Z osa, 2) Osa X, 3) Osa Y, 4) Zásobník šroubovacích hlav, 5) Zásobník šroubů, 6) Umístění paletek

4.6 Lisovací pracoviště

Jak již je zmíněno v předchozí kapitole, jedním z pracovišť je lisovací buňka. Principem této buňky je lisování za pomoci hydraulického válce se servo-proporcionálním ventilem pro řízení jeho vertikálního pohybu. Válcem je v prostoru pracoviště pohybováno za pomoci dvou servomotorem řízených os v jedné rovině. Na pracovišti se dále nachází gravitační zásobník na celkem 9 různých objektů s jejich podavači, zásobník na lisovací hlavy, pojezdy jednotlivých os, plocha pro umístění paletek a jejich upínání. Principem lisovacího pracoviště je přečtení signálu z robota týkající se informace o typu přiložené paletky a následného provedení konfigurovaných lisovacích operací. Jednotlivé akční prvky na pracovišti jsou pneumatické válce pro podávání objektů (8), upnutí lisovacích hlav (5) a samotné paletky (7). Další akční členy jsou senzory, poháněné pohybové osy (3 a 4)

a hydraulický válec pro lisování (6) viz obr. 23. Manipulátor obsluhující toto pracoviště disponuje pracovním prostorem 305 x 590 x 150 mm.



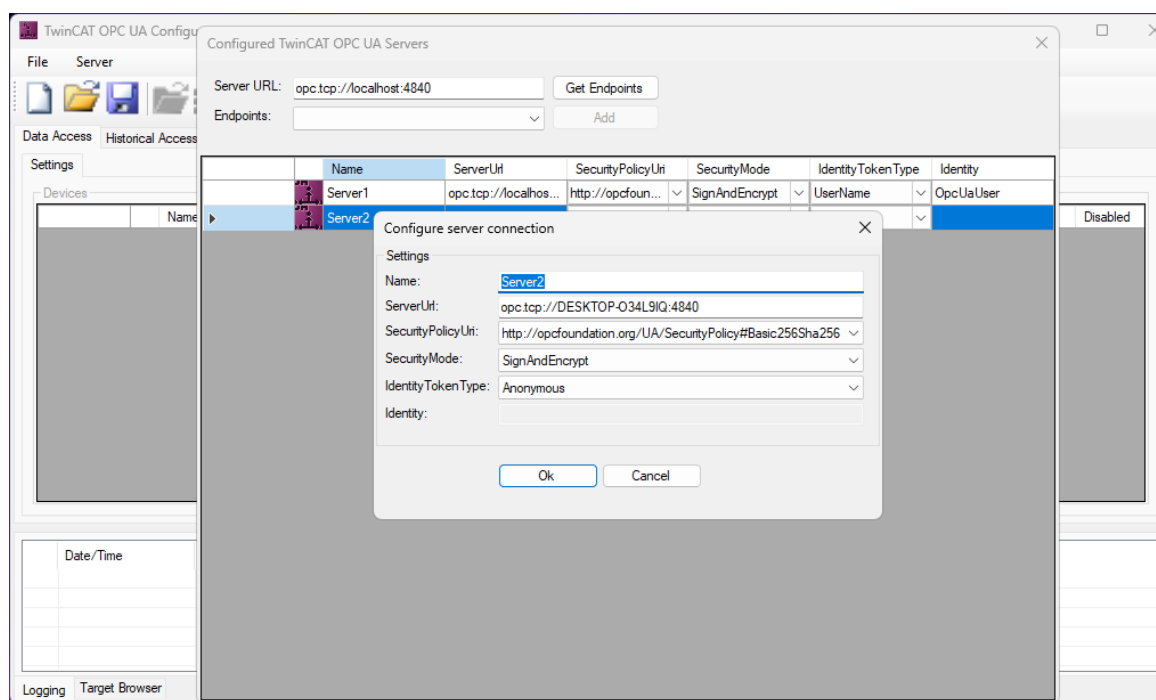
Obr. 23 Ukázka lisovací stanice 1) Servopohon os, 2) Gravitační zásobník, 3) Osa Y, 4) Osa X, 5) Zásobník lisovacích hlav, 6) Hydraulický válec, 7) Uchycení paletek, 8) Podavač objektů, 9) Umístění paletek

5 VIRTUÁLNÍ ZPROVOZNĚNÍ

V této kapitole je vysvětleno vytvoření jednotlivých komponent virtuálně zprovozněného modelu systému spolu s konfigurací simulace. Virtuální zprovoznění v této práci bude zpracováno za pomoci softwaru pro mechatronickou simulaci buňky – NX Mechatronics concept designer. Řídicí systém ovládající akční prvky pracoviště bude vytvořen za pomoci softwaru Beckhoff TwinCAT 3. Virtuálním komunikačním serverem zpracovávajícím signálovou strukturu bude TwinCAT 3 OPC UA Server. V rámci virtuálního zprovoznění bylo zavedeno systematické indexování vazeb a proměnných.

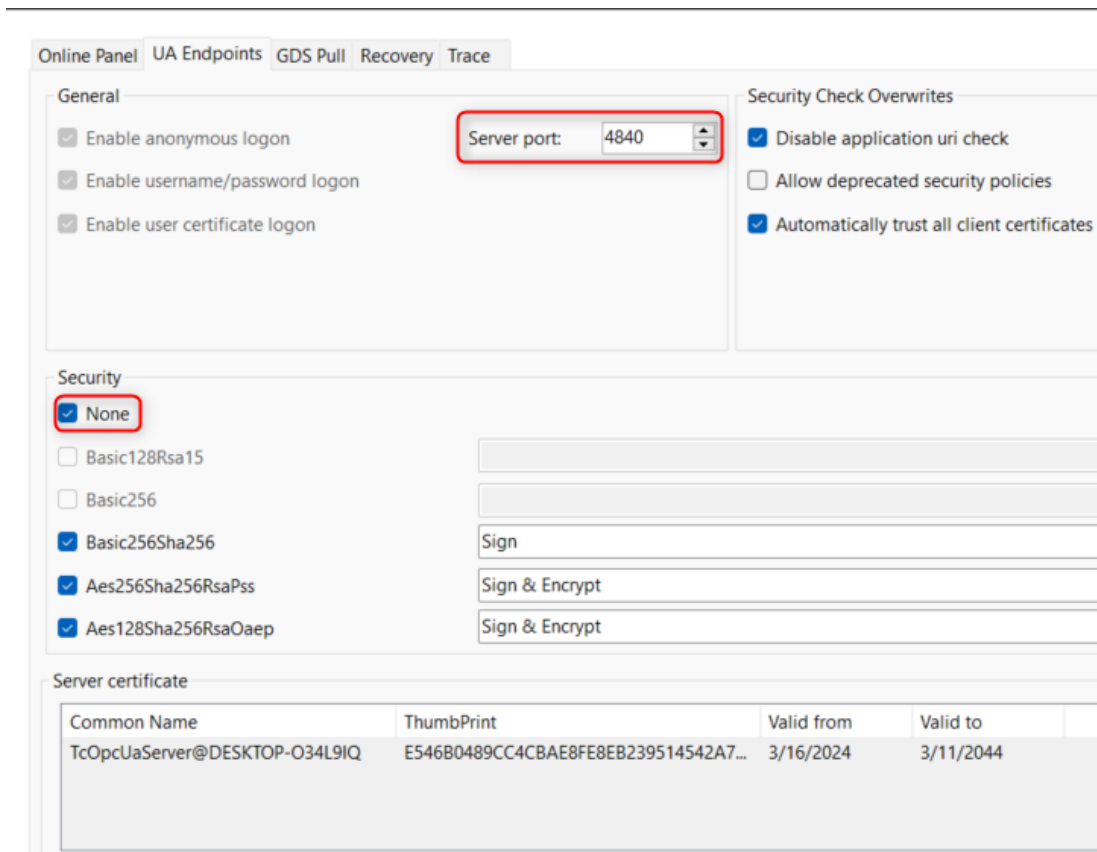
5.1 Beckhoff TwinCAT 3 OPC UA Server

Virtuální server je součástí výše zmíněného „Connectivity“ balíčku od firmy Beckhoff. Pomocí virtuálního serveru je umožněno propojení mezi různými aplikacemi připojenými ke stejnému serveru. OPC UA – Open platform communication unified architecture je standartní komunikační protokol pro výměnu dat mezi výrobními a řídicími systémy. Umožňuje komunikaci mezi PLC a senzory, pohony nebo uživatelským rozhraním. Samotnou konfiguraci OPC serveru lze vidět na obr. 24. Na obrázku lze dále vidět různé parametry jako je způsob šifrování a způsob přihlášení.



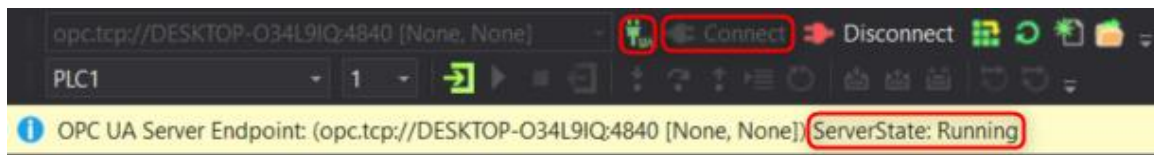
Obr. 24 Konfigurace OPC serveru

Po úspěšném nastavení OPC serveru je ještě nutná jeho aktivace v samotném TwinCAT 3 vývojovém prostředí. V této práci je používáno bezpečnostní nastavení „None“, kvůli nutnosti neustálého přihlašování u jiných nastavení, lze vidět na obr. 25. Kontrola nastavení serverového portu na 4840.



Obr. 25 Nastavení OPC v rámci TwinCAT 3

V případě správného nastavení pak po stisknutí tlačítka „connect“ viz. obr. 26 následného tlačítka pro zjištění stavu OPC serveru vyskakuje lišta hlásící, že server funguje (ServerState: Running).

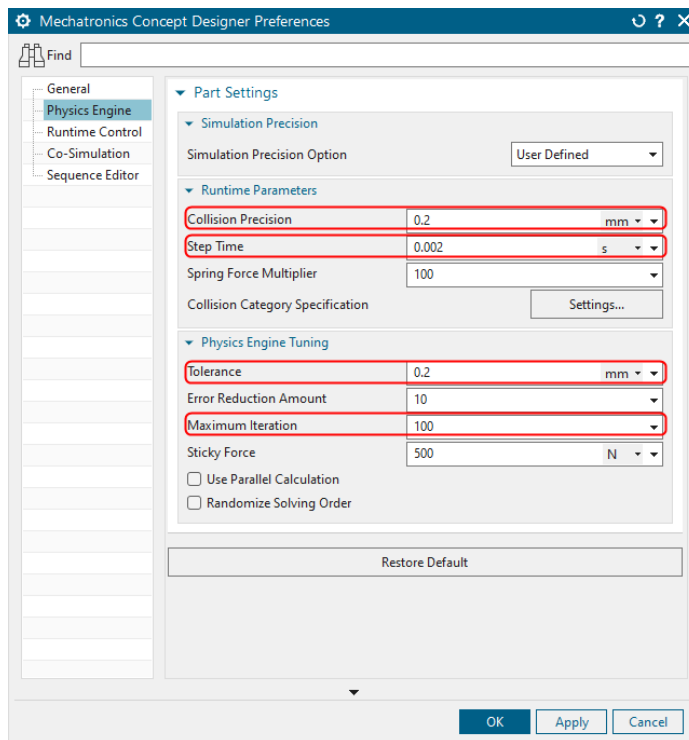


Obr. 26 Správné připojení OPC serveru TwinCAT 3

5.2 NX – MCD

Prvním krokem pro simulaci v prostředí NX MCD je správné porozumění takzvaným preferencím – nastavení samotného simulačního programu. V případě špatného či neoptimálního nastavení simulace může docházet k abnormálnímu zvyšování časového zpoždění simulace (delay time), nebo k úplnému spadnutí simulace. Ke spadnutí simulace může dojít například z důvodu špatné kolize mezi tělesy, která může být způsobena špatným nastavením tolerancí nebo přesností kolize. Jednotlivé nastavitelné parametry v NX MCD lze vidět na obr. 27 spolu s nastavenými hodnotami pro tuto práci a patří mezi ně:

- Kolizní přesnost (Collision precision) [mm]
- Krokování (Step time) [s]
- Tolerance (Tolerance) [mm]
- Počet výpočtů [-]



Obr. 27 Ukázka nastavení NX MCD

Kolizní přesnost, jak název napovídá, určuje nutnou vzdálenost mezi dvěma tělesy, aby byla simulací považována za kolidující. Krokování specifikuje, jaký časový úsek je vypočítán v jednom kroku simulace. Tolerance definuje maximální délkovou odchylku mezi tělesy spojenými pomocí jakékoliv vazby.

Virtuální zprovoznění z pohledu NX MCD spočívá v definici základních fyzikálních prvků, s kterými lze v programu následně pracovat tzn. tuhá a kolizní tělesa. Dalším krokem je definovat kinematické vazby, senzory, akční členy a jednotlivé operace. Poslední částí je definice signálů.

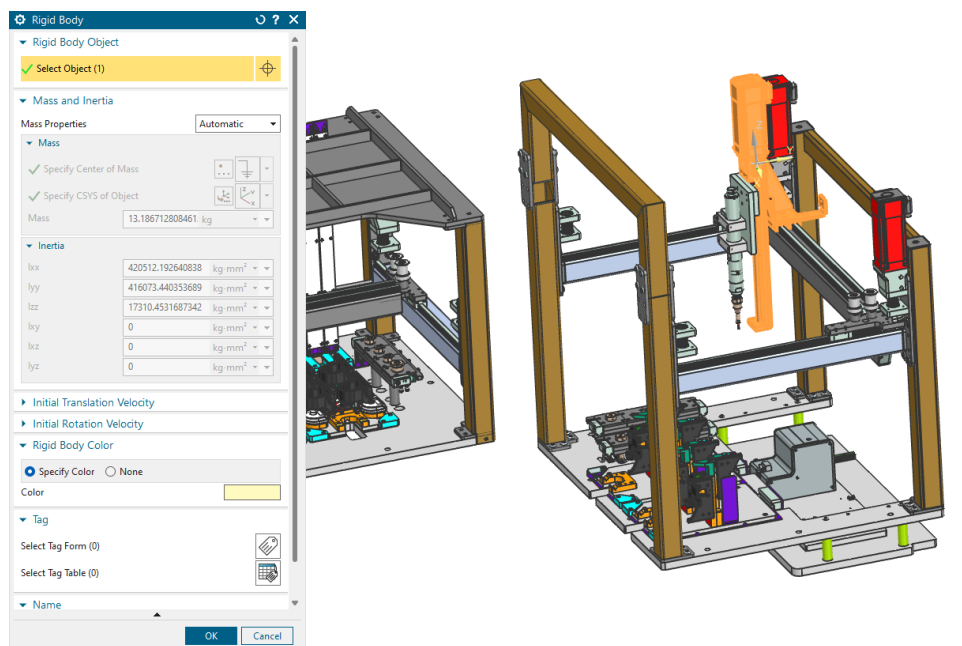
5.2.1 Základní fyzikální prvky

Každý model, respektive těleso, od kterého je očekávána nějaká funkce v rámci běžící simulace, musí být definováno jako rigidní (tuhé) těleso.

Tuhá tělesa

V NX MCD je potřeba definovat tuhá tělesa jako komponenty sestavy, které se mohou volně pohybovat. Jako tuhé těleso lze vybrat více komponent, které se pak budou chovat jako jedno. Tuhá tělesa mají vlastní hmotnost, moment setrvačnosti, počáteční translační či úhlovou rychlost, pozici a orientaci v prostoru. Tato tělesa jsou ovlivňována vnějšími vlivy jako je například základní zrychlení – gravitace. Tělesa, která nejsou definována jako tuhá, pak simulace nebere v potaz a nechává je na pozadí. Mezi taková tělesa patří například oplocení buňky, či její krytování.

Pro vytvoření tuhého tělesa je nutné vybrat všechny komponenty, které mají jako jedno těleso reagovat. Na základě vybraných komponent a jejich materiálových vlastností je automaticky vypočítána váha a moment setrvačnosti tělesa. Zároveň je možné zadat tyto parametry manuálně. Dialogové okno pro definici tuhého tělesa lze vidět na obr. 28.



Obr. 28 Dialog tuhého tělesa – definice X vozíku

Dalšími definovanými tuhými tělesy v simulaci jsou:

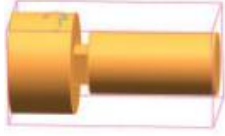
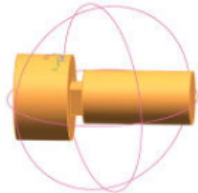
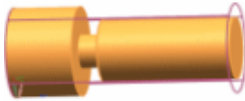

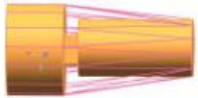

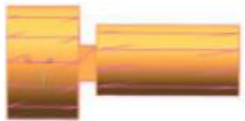
- Jednotlivé lisovací hlavy
- Jednotlivé lisovací objekty
- Šroubovací hlava
- Šrouby
- Vozíky os
- Jednotlivé paletky
- Podavače objektů
- Báze obou buněk
- Zásobníky hlavy

Kolizní tělesa

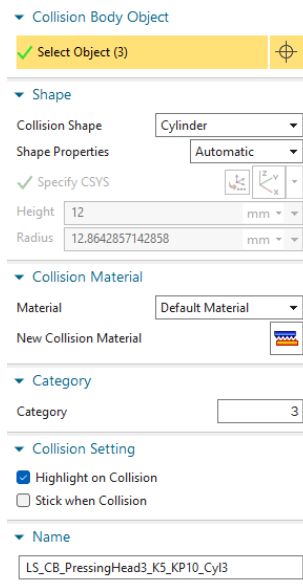
Kolizní tělesa určují chování jednotlivých tuhých těles při vzájemné kolizi. Pro vzájemnou kolizi mezi tělesy je nutné, aby obě tělesa měla definované kolizní parametry. Kolizní tělesa mohou být definovaná jako samostatná anebo spjatá k tuhému tělesu. Kolizní těleso vzniká jako zjednodušený objekt – krychle, koule, válec, kapsle, konvexní nebo za pomoci mřížkování (mesh) tělesa. Jednotlivé možnosti definice kolizního tělesa viz tab. 5. V tabulce jsou popsány jednotlivé popisy kolizních těles, které jsou ohodnoceny podle jejich přesnosti, spolehlivosti a náročnosti na výkon. Geometrickou přesností rozumíme schopnost důkladného kopírování tvaru označeného tělesa. Spolehlivost je schopnost tělesa nezpůsobovat v simulaci chyby typu rozklepávání systému nebo zasekávání. Simulační výkon pak zkoumá jednotlivá tělesa z pohledu výpočetní náročnosti, přičemž je třeba brát každé těleso jako směsici mřížek v prostoru. Elementární tělesa mají mřížek málo s jednoduchou definicí, zatímco složitá tělesa využívají k přesnému popisu tělesa velké množství menších mřížek, které jsou náročnější na výpočetní výkon. Složitým typem je například „mesh“ nebo „konvexní“. Složitější metody využívají síť trojúhelníků nebo

obdélníků snaží se co nejdůvěrněji kopírovat tvar tuhého tělesa. Množství trojúhelníků v této síti je konfigurovatelné v rámci dialogového okna a má značný vliv na plynulý chod simulace. S rostoucím zaplněním sítě roste náročnost simulace na výpočetní techniku. Elementární tělesa na druhou stranu využívají jednoduché tvary, které nedokáží tak dokonale napodobit tuhá tělesa. V případě nutnosti, je možné použít například dvojici elementárních těles pro vytvoření tělesa komplexního. Příkladem lze uvést například použití dvou boxů z tab. 5 pro vytvoření tělesa tvaru L.

Tab. 5 Typy kolizních těles, jejich přesnost, spolehlivost, vliv na simulaci [14]

| Typ | Geometrická přesnost | Spolehlivost | Simulační výkon |
|---|----------------------|--------------|-----------------|
|  Box | Nízká | Vysoká | Vysoký |
|  Koule | Nízká | Vysoká | Vysoký |
|  Válec | Nízká | Vysoká | Vysoký |
|  Kapsle | Nízká | Vysoká | Vysoký |
|  Konvexní | Střední | Vysoká | Střední |
|  Multi-konvexní | Střední | Vysoká | Střední |
|  Mesh | Vysoká | Nízká | Nízká |

V rámci definice kolizního tělesa lze mimo jeho typu nastavit jeho kolizní materiál. Materiál se pak definuje za pomoci tří vlastností – dynamické tření, valivé tření a součinitele restituice. Posledním ale neméně důležitým definovatelným parametrem je kolizní kategorie. Pomocí kolizní kategorie lze stanovit, která tělesa mohou kolidovat s kterými. Základní pravidla kolize jsou, že mezi sebou kolidují tělesa se stejnými kategoriemi, přičemž všechna tělesa kolidují s kategorií 0. Tato pravidla se dále dají upravovat kolizní tabulkou v preferencích simulace. Dialogové okno pro kolizní těleso lze vidět viz obr. 29.

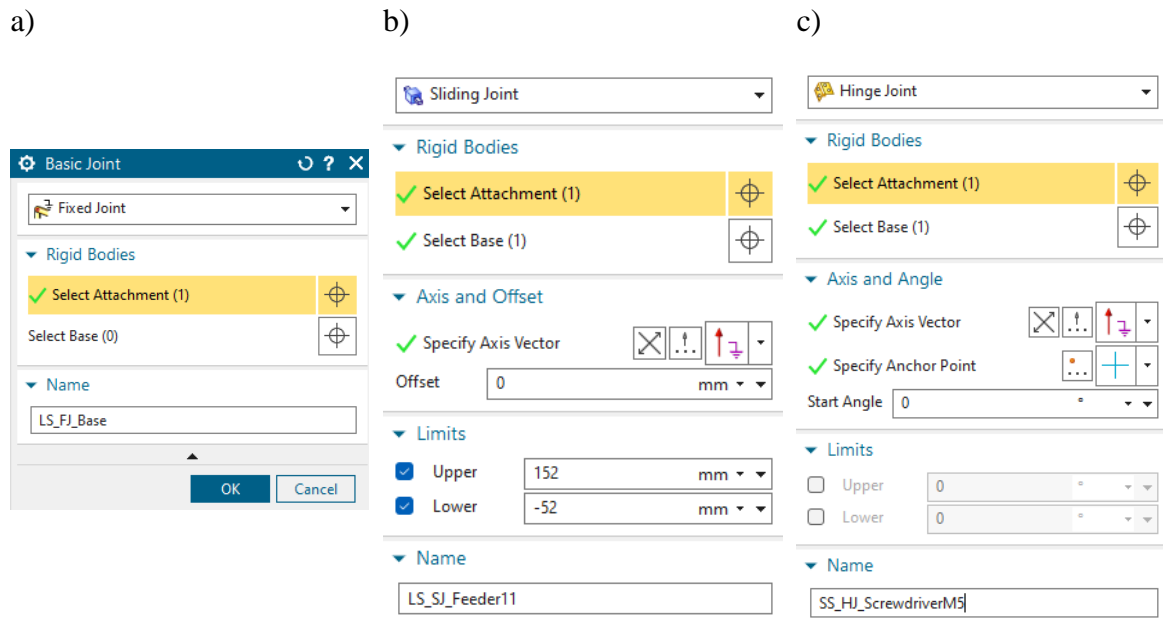


Obr. 29 Dialog kolizního tělesa

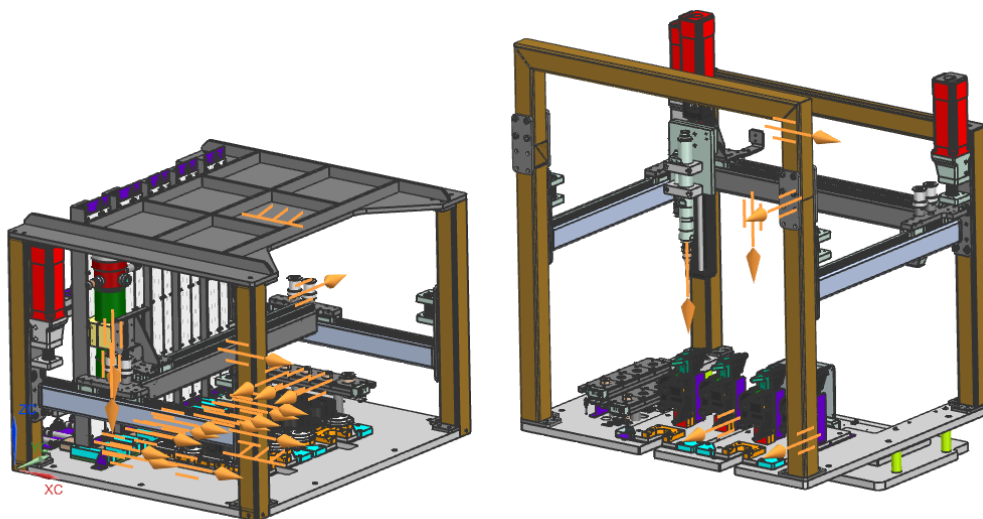
5.2.2 Vazby mezi tělesy

Vazby jednoduše řešeno definují stupně volnosti tuhého tělesa. V této diplomové práci jsou použity fixní (pevné) vazby, posuvné vazby a rotační vazby. Fixní vazby upevňují bázi pracoviště v prostoru a zároveň jsou použity pro transport lisovacích hlav. Translační nebo také posuvné vazby zaručují víceméně všechny pohyby po pracovišti. Jedinou výjimkou je pak kombinace rotační vazby spolu s translační pro vytvoření šroubovacího pohybu na šroubovacím pracovišti. Dialogová okna jednotlivých vazeb lze vidět viz. obr. 30. Translační vazby umožňují pohyb v jedné translační ose, zatímco rotační pouze kolem jedné osy. Pevná vazba zamezuje jakémukoliv pohybu vůči jinému tělesu. Jednotlivé vazby lze vidět na obr. 31. Definovanými vazbami v simulaci jsou:

- Rotační
 - Rotační vazba mezi posuvnou vazbou kuličkového šroubu a šroubovací hlavou
- Translační
 - Pohyb jednotlivých podavačů objektů vůči bázi
 - Pohyb hydraulického válce vůči osovým pojezdům
 - Pohyb upínacích ploch vůči bázi
 - Pohyb osových pojezdů vůči bázi
- Pevné
 - Báze vůči pozadí
 - Kontakty hydraulického válce s lisovacími hlavami



Obr. 30 Dialogová okna jednotlivých vazeb a) fixní, b) translační, c) rotační



Obr. 31 Translační, rotační a fixní vazby v simulaci

5.2.3 Kolizní senzory

Kolizní senzory jsou nedílnou součástí simulací v případě nutnosti zpětné vazby ze simulace, ať už se jedná o přítomnost nebo kolizi. Kolizní senzory fungují podobným způsobem jako kolizní tělesa, přičemž ze samotné kolize vyvozují zpětnou vazbu. Pomocí kolizních senzorů lze zjišťovat přítomnost dílů, vysílat zpětnou vazbu o poloze hydraulických nebo pneumatických válců nebo například spouštět generování objektů. U kolizních senzorů lze definovat kolizní zónu za pomoci stejné metody jako u kolizních těles. Stejným způsobem lze definovat i kolizní kategorii. Dialogové okno pro definici kolizního senzoru typu trigger lze vidět na obr. 32. V rámci dialogového okna lze přepínat kolizní senzory mezi funkcí „Trigger“ a „Switch“. Funkce „Trigger“ zapíše booleovský „true“ na výstup signálu v moment, kdy je kolizní senzor v kontaktu s jiným kolizním

tělesem a „false“ v opačném případě. Na druhou stranu funkce „Switch“ přepisuje mezi hodnotami „true“ a „false“ při každé kolizi. Z toho vyplývá, že první kolize způsobí přepnutí výstupní hodnoty na „true“, následná druhá kolize jej přepne na „false“ a třetí zpět na „true“.

▼ Type

Trigger

▼ Collision Sensor Object

✓ Select Object (1)

▼ Shape

Collision Shape Line

Shape Properties User Defined

✓ Specify CSYS

Length 33 mm

▼ Category

Category 0

▼ Highlight on Collision

Highlight on Collision

▼ Detection Type

System User Both

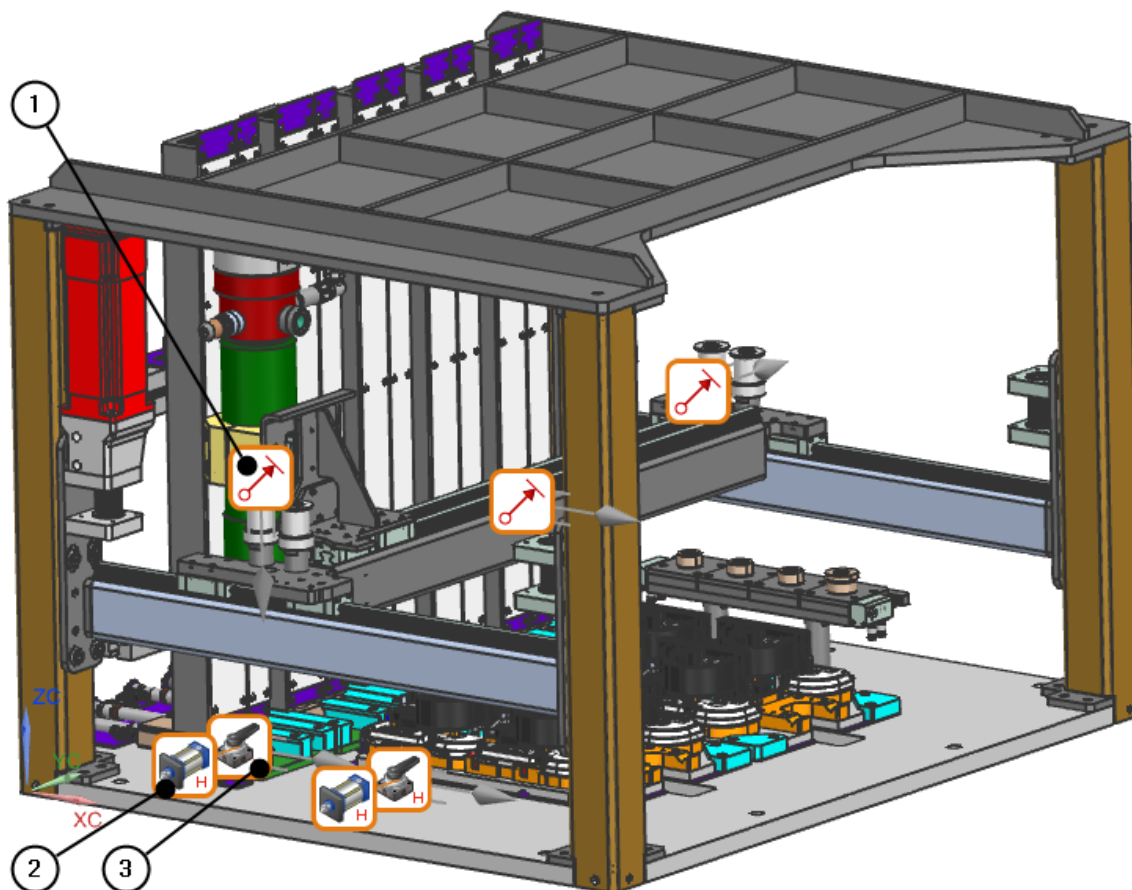
▼ Name

LS_CS_11Presence

Obr. 32 Dialogové okno kolizních senzorů

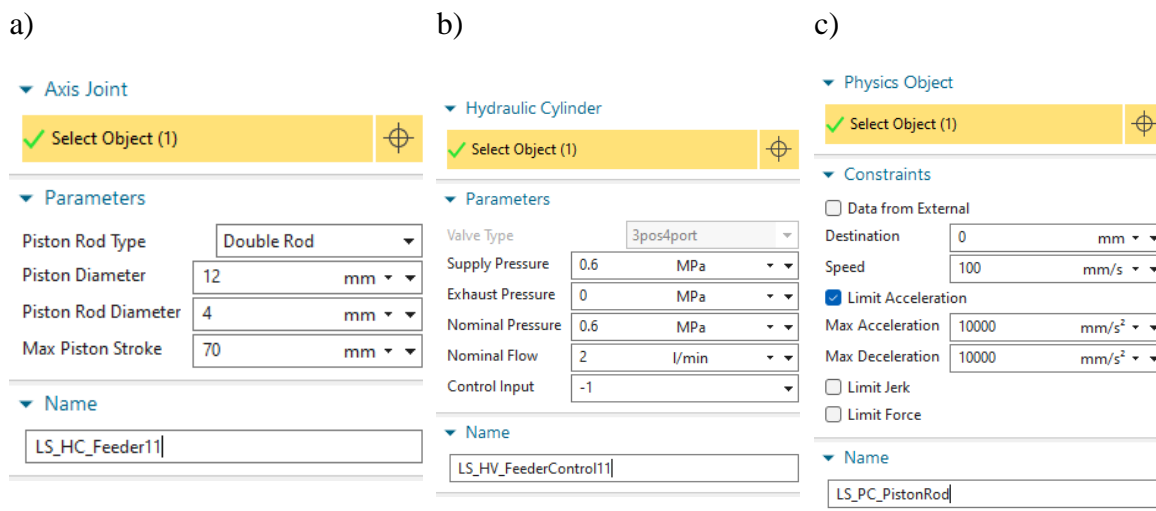
5.2.4 Akční členy

Aby bylo možné s jednotlivými vazbami a tělesy pohybovat, je potřeba definovat akční členy. Jednotlivé akční členy lze definovat podle pozice, rychlosti, síly nebo momentu. Další funkční akční členy jsou pneumatický válec a ventil, hydraulický válec a ventil nebo transportní plocha (dopravník). Pomocí pozice (1) jsou ovládány jednotlivé pojezdy os obou stanic spolu s hydraulickým válcem. Jediným prvkem ovládaným pomocí rychlosti je rotační vazba šroubovací hlavy. Všechny pneumatické prvky, jako jsou jednotlivé podavače a upínače, jsou ovládány za pomoci pneumatických válců (2) a ventilů (3) na obr. 33.



Obr. 33 Ukázka akčních prvků v simulaci

Jednotlivá dialogová okna akčních prvků spolu s jejich definicemi lze vidět na obr. 34. Akční prvky lze definovat primárně za pomoci jejich krajních poloh. V případě pneumatických ventilů pak i za pomoci simulovaného tlaku pro postupné vysunutí válce.



Obr. 34 Dialogová okna jednotlivých akčních prvků a) Pneumatický válec, b) Pneumatický ventil, c) Ovládání pozice

5.2.5 Signály

Signály lze propojit se samotnými parametry jednotlivých prvků, například pozice a tlak u akčních členů. Jejich primární funkcí je přenášet komunikaci a ovládat akční prvky. U jednotlivých signálů lze definovat, jaký parametr budou ovlivňovat nebo číst, zda budou vstupní či výstupní, jaký datový typ budou přenášet a jejich počáteční hodnotu. Dialogové okno pro definici signálů ukazuje obr. 35.

▼ Settings

Connect with Runtime Parameter

✓ Select Physics Object (1)

Parameter Name

IO Type

Data Type

Initial Value

▼ Display Icon

Display Icon

▼ Name

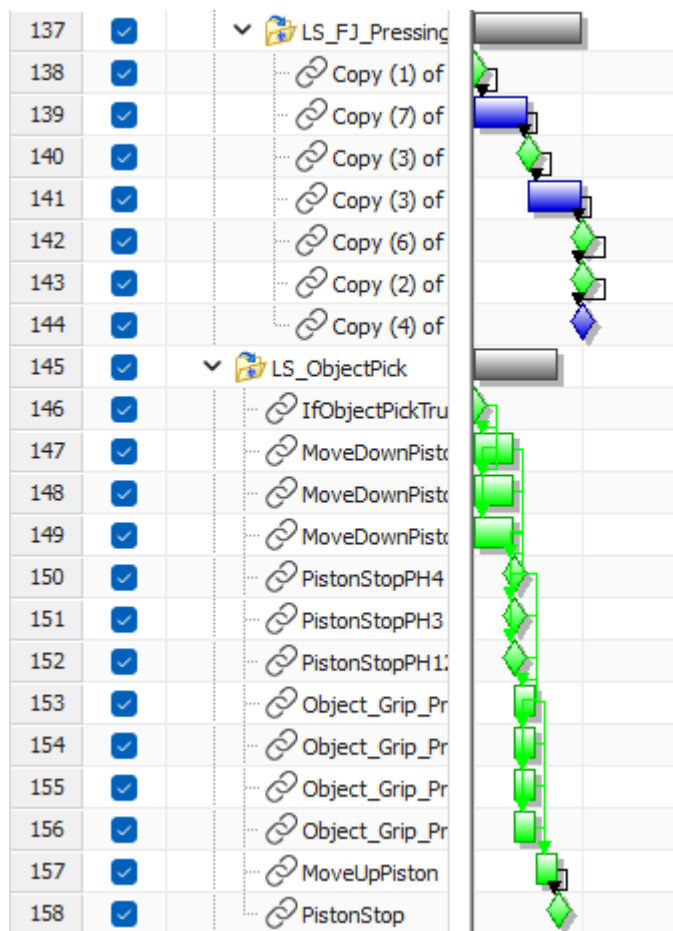
Signal Name

Obr. 35 Dialogové okno signálů

5.2.6 Operace

Ovládání jednotlivých operací, jako je například vysunutí podavače objektů, generování objektů nebo upínání těles, probíhá za pomoci sekvenčního nebo časového ovladače na obr. 36. Zároveň můžeme vidět zelené, podmíněné, operace a modré operace, které se vykonávají bezpodmínečně. Při sepnutí konkrétního signálu, například pro podavač, pak následuje operace vysunutí samotného podavače. Takovýmto způsobem jsou nastaveny různé akce, jako je například zmíněné podání objektů, zvednutí či uvolnění hlavy nebo sepnutí generování objektů nebo paletek. V oblasti „Runtime Parameter“ v dialogovém okně operací je zvolen ovlivňovaný parametr a nastavovaná hodnota. V poli „Condition“ jsou určeny nutné podmínky pro ovlivnění parametru.

a)



b)

Operation
 Physics Object
 ✓ Select Object (1)

Display Icon

Duration
 Time: 0.37 s

Runtime Parameter

| S.. | Name | Oper... | Value | Unit | I... |
|-------------------------------------|--------|---------|----------|------|--------------------------|
| <input checked="" type="checkbox"/> | value | := | 1.000000 | | <input type="checkbox"/> |
| <input type="checkbox"/> | active | := | true | | <input type="checkbox"/> |

Edit Parameter

Condition

| If | Object | Parameter | Oper... | Value |
|--------------------------|-----------------|-----------|---------|-------|
| <input type="checkbox"/> | | | | |
| <input type="checkbox"/> | LS_FJ_Pressi... | active | == | true |
| <input type="checkbox"/> | LS_FJ_Pressi... | active | == | true |

Edit Condition Parameter

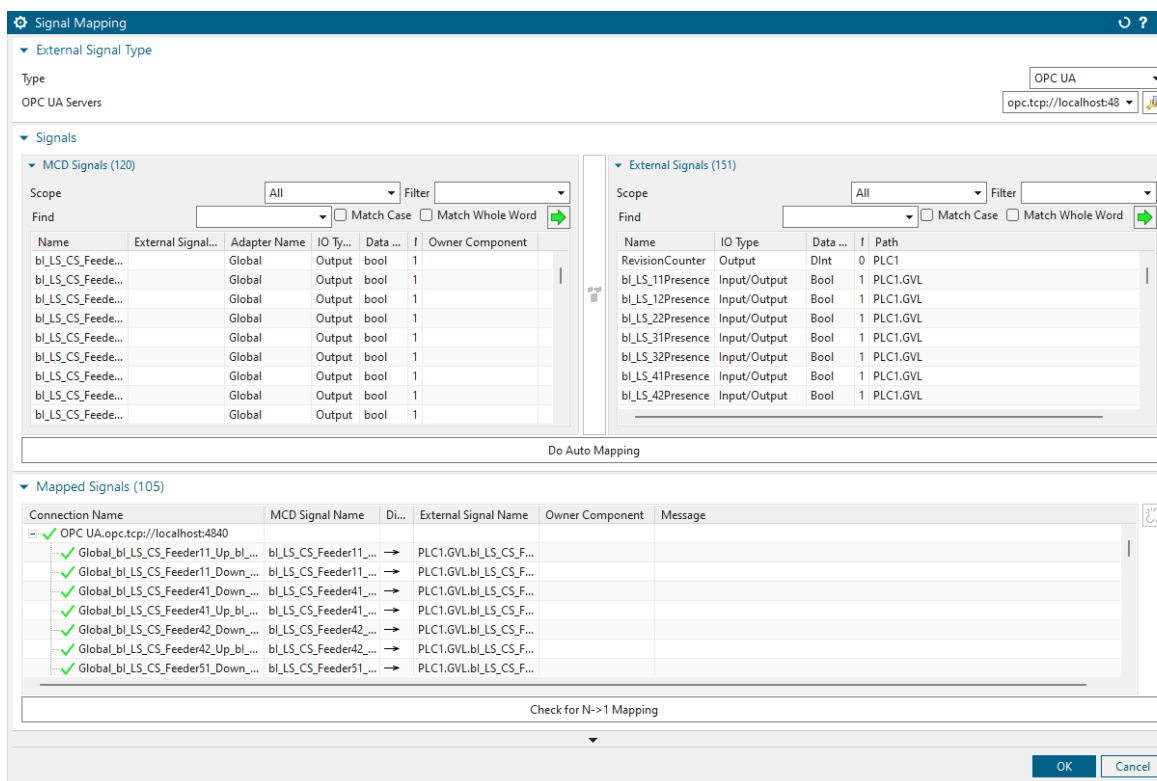
Select Object
 ✓ Select Condition Object (1)

Name
 MoveDownPiston_PressHead12

Obr. 36 Ukázka a) událostně řízené operace a b) dialogové okno operací

5.2.7 Mapování signálů

Po kompletním nastavení mechatronických komponent a nastavení signálů následuje mapování signálů se signály v řídicím systému TwinCAT 3. V dialogovém okně pro signálové mapování je třeba definovat typ připojení. V tomto případě využíváme OPC UA připojení, přičemž server hledáme na lokálním PC, proto vypisujeme adresu localhost:4840. V případě korektního zápisu proměnných (proměnné definované v řídicím systému, musí být na jediné písmeno stejné jako signálové proměnné v MCD) dojde ke kompletní synchronizaci databází a mapování proměnných, jak lze vidět na obr. 37.



Obr. 37 Ukázka mapování signálů

5.3 Beckhoff TwinCAT 3

V následující kapitole byl vytvořen řídicí program pro ovládání zmíněného pracoviště. Program byl vytvořen za pomoci vývojového prostředí TwinCAT 3 od výrobce Beckhoff. K vytvoření samotného programu bylo využito balíčku MOTION pro ovládání jednotlivých os stanic. Byly vytvořeny jednotlivé funkční bloky a programy pro ovládání jednoúčelové stanice. Následně bylo vytvořeno přehledné HMI (Human machine interface) s prvky pro ovládání a safety. V rámci tohoto HMI byla vytvořena i důmyslná obrazovka pro manuální režim umožňující upravovat samotné pojezdy os.

5.3.1 Definice proměnných

Jak již bylo zmíněno ke kompletnímu propojení NX MCD s TwinCAT OPC serverem je nutné, aby byly všechny signály stejně pojmenované. Všechny signálové proměnné použité pro komunikaci se simulačním programem byly definovány pomocí globálního listu proměnných – GVL, jehož ukázkou lze vidět na obr. 38, kde jsou definovány jak vstupní proměnné do lisovací stanice „bI_LS_xxxx“ tak výstupní proměnné „bQ_LS_xxxx“.

```

59  bI_LS_PressingHead1_PickPlaceDone  AT%I^  :  BOOL;
60  bI_LS_PressingHead2_PickPlaceDone  AT%I^  :  BOOL;
61  bI_LS_PressingHead3_PickPlaceDone  AT%I^  :  BOOL;
62  bI_LS_PressingHead4_PickPlaceDone  AT%I^  :  BOOL;
63  bI_LS_PressingHead1_Object_Pick    AT%I^  :  BOOL;
64  bI_LS_PressingHead2_Object_Pick    AT%I^  :  BOOL;
65  bI_LS_PressingHead3_Object_Pick    AT%I^  :  BOOL;
66  bI_LS_PressingHead4_Object_Pick    AT%I^  :  BOOL;
67
68  bI_LS_PressingHeadDetection         AT%I^  :  BOOL;
69
70  fI_LS_PositionZ_Encoder             AT%I^  :  LREAL;
71
72  // HW Outputs
73  //Lisovací
74  bQ_LS_AutoSpawnK3                  AT%Q^  :  BOOL;
75  bQ_LS_AutoSpawnK4x10               AT%Q^  :  BOOL;
76  bQ_LS_AutoSpawnK4x16               AT%Q^  :  BOOL;
77  bQ_LS_AutoSpawnK5                  AT%Q^  :  BOOL;
78  bQ_LS_AutoSpawnK10                 AT%Q^  :  BOOL;
79  bQ_LS_AutoSpawnKP1615              AT%Q^  :  BOOL;
80  bQ_LS_AutoSpawnKP10                AT%Q^  :  BOOL;
81  bQ_LS_AutoSpawnKP15                AT%Q^  :  BOOL;
82  bQ_LS_AutoSpawnKP30                AT%Q^  :  BOOL;
    
```

Obr. 38 Ukázka definice proměnných pro komunikaci s NX MCD

V rámci listu proměnných byly zároveň proměřeny jednotlivé pozice pracoviště, které byly uloženy do proměnlivých polí pro kompatibilnější možnost psaní řídicího kódu. Zároveň je pak možné tyto souřadnice pohodlně využívat v dalších částech programu. Zmínované definice lze vidět na obr. 39.

```

195
196  fOp2Pos1X                          :  ARRAY[1..4] OF ARRAY [1..3] OF LREAL := [ [109, 109, 109 ],
197                                          [176, 176, 176 ],
198                                          [252, 252, 252 ],
199                                          [331, 331, 0 ]];
200
201  fOp2Pos2X                          :  ARRAY[1..4] OF ARRAY [1..3] OF LREAL := [ [109, 109, 109 ],
202                                          [176, 176, 176 ],
203                                          [252, 252, 252 ],
204                                          [331, 331, 0 ]];
205
206  fOp3Pos1X                          :  ARRAY[1..4] OF ARRAY [1..2] OF LREAL := [ [102, 102],
207                                          [162, 162],
208                                          [222, 222],
209                                          [282, 282]];
210
211  fOp3Pos2X                          :  ARRAY[1..4] OF ARRAY [1..2] OF LREAL := [ [102, 102],
212                                          [162, 162],
213                                          [222, 222],
214                                          [282, 282]];
    
```

Obr. 39 Ukázka definice souřadnic

5.3.2 Motion

Jak již bylo zmíněno v teoretické části této práce TwinCAT 3 umožňuje uživateli využít několika jeho přednastavených balíčků. Za pomoci balíčku Motion byly vytvořeny konfigurace jednotlivých pohyblivých os. V případě lisovací stanice tedy pojezd osy X a pojezd osy Y, zatímco u šroubovací stanice je mimo osy X a Y zahrnut také poháněný kuličkový šroub zastávající pojezd osy Z. Na obrázku obr. 40a a lze vidět aktivace samotné osy, kde pomocí řádku „Axis“ definujeme, které osy se pohyb týka, „Enable“

znamená spuštění této osy a „Override“ je potenciometr určující procentuální výkonnost pojezdu osy. Na obrázku obr. 40b lze vidět definici absolutního polohování osy Y. Na řádce „Execute“ určujeme ovládající proměnnou, která spustí absolutní polohování pojezdu osy na souřadnici definovanou pomocí řádku „Position“ při dosažení maximální rychlosti „Velocity“. Podobným způsobem je v práci definována i funkce zastavení os.

| | |
|--|--|
| a) | b) |
| <pre> 1 LS_Axis_X_Activation(2 Axis:= GVL.LS_X_Axis, 3 Enable:= TRUE, 4 Enable_Positive:= TRUE, 5 Enable_Negative:= TRUE, 6 Override:= 100); </pre> | <pre> 18 MoveAbsLS_Y_Axis(19 Axis:= GVL.LS_Y_Axis, 20 Execute:= GVL.bExecute, 21 Position:= GVL.fLSMoveY, 22 Velocity:= GVL.fQ_LS_SpeedY, </pre> |

Obr. 40 Ukázka definice pomocí balíčku Motion a) Aktivace samotné osy X, b) Definice absolutního pohybu osy Y

5.3.3 Safety – bezpečnost

Funkce safety (bezpečnost) v tomto řídicím programu funguje pouze na ukázkou, jelikož bezpečnostní funkce přebírá tzv. safety PLC, které je k tomu uzpůsobené. V reálném PLC je pro bezpečnost využíváno samostatného procesoru, který je programován separátně od vlastního PLC kódu a s kterým by v tomto případě komunikovalo nadřazené PLC buňky. V této práci byla naprogramována podmínka v případě porušení bezpečnostních opatření jako je například otevření dveří u oplocení robotizované buňky nebo pokles tlaku vzduchu pod přijatelnou mez. Hlavním předpokladem je, že bezpečnostní funkci tohoto systému bude mít na starost ovládací systém celé buňky. Stejným způsobem bylo v této práci hleděno na problematiku emergency stop – nouzového zastavení. Ukázka definice těchto bezpečnostních prvků lze vidět na obr. 41, kde lze vidět, že při sepnutí tlačítka emergency stop se stavový automat přehodí do stavu určeného pro nouzové zastavení. Celý programový blok viděný na obrázku je předsunut před stavový automat operací, aby došlo k jeho prioritizaci.

```

IF gvl.b_SafetyDoorsClosed AND gvl.b_SafetyMechanism1 AND gvl.b_SafetyMechanism2 THEN
    gvl.b_Safety := TRUE;
ELSIF gvl.b_EmergencyStop THEN
    GVL.nLSState := 99;
    GVL.nSSState := 99;
ELSE
    GVL.nLSState := 97;
    GVL.nSSState := 97;
END IF

```

Obr. 41 Ukázka definice bezpečnostních funkcí

5.3.4 Řídicí kód

Řídicí kód v této práci je popsán pomocí dílčích programů a funkčních bloků. Celá struktura funguje jako stavový automat, kdy voláním jednotlivých stavů systém prochází operacemi. V samotném kódu tedy vždy nastane definovaný stav, který zavolá specifické operace pro tento stav. Jeden ze stavů lze vidět na obr. 42. Vývojový diagram algoritmu operací lze najít v přílohách práce

```

6      l2:
7      gvl.fQ_LS_PositionZ      := 0;
8      GVL.fLSMoveX            := GVL.fFeederLocationX[9];
9      GVL.fLSMoveY            := GVL.fFeederLocationY[9];
10     GVL.bExecute             := TRUE;
11     IF GVL.LS_X_Axis.NoToPlc.ActPos = GVL.fFeederLocationX[9] AND GVL.LS_Y_Axis.NoToPlc.ActPos = GVL.fFeederLocationY[9] TH
12         GVL.bExecute         := FALSE;
13         gvl.fQ_LS_PositionZ   := GVL.fFeederLocationZ[9];
14         IF gvl.fi_LS_PositionZ_Encoder <= GVL.fFeederLocationZ[9] + 0.1
15     AND gvl.fi_LS_PositionZ_Encoder >= GVL.fFeederLocationZ[9] - 0.1 THEN //Check if location reached
16         gvl.bQ_LS_PressingHead3_Grip := TRUE;
17         IF GVL.bI_LS_PressingHead3_Object_Pick = TRUE THEN
18             gvl.fQ_LS_PositionZ      := 0;
19             nStateOperation          := 1;
20         END_IF
21     END_IF
22 END_IF

```

Obr. 42 Stavový automat – stav 12

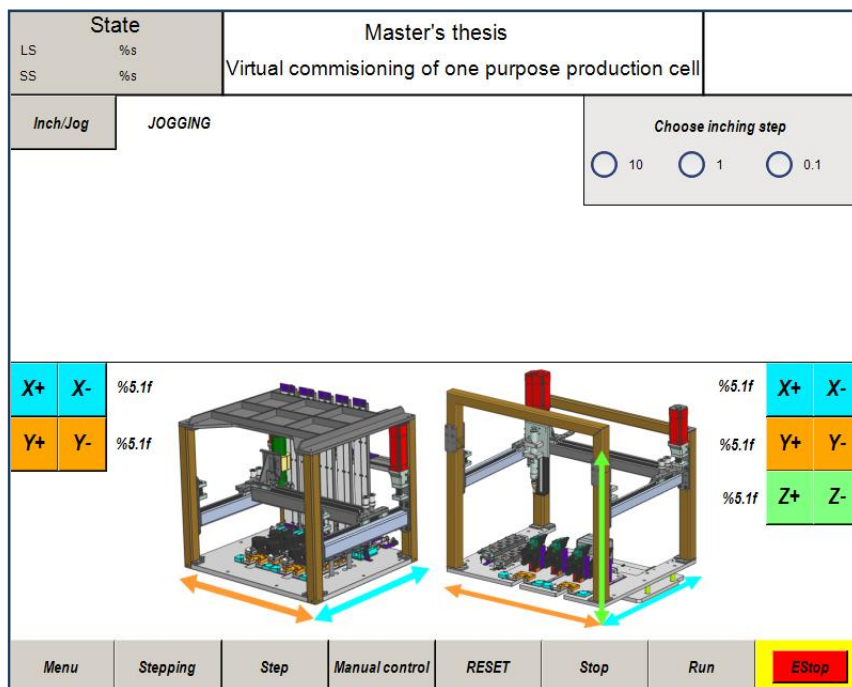
Na obr. 42 lze vidět jeden z mnoha jednoduchých stavů, jedná se o stav vyzvednutí objektu kluzného pouzdra z podavače. Nejdříve je zajištěno, že se manipulátor bude hýbat ve výchozí výšce pomocí příkazu *gvl.fQ_LS_PositionZ := 0;*, následně jsou nastaveny jednotlivé poziční souřadnice aplikováním příkazů na řádcích 8 a 9 na daný podavač a voláním operace *gvl.bExecute := TRUE;*; je spuštěn přesun manipulátoru. Na následujícím řádku manipulátor čeká na příjezd manipulátoru na pozici podavače a následně spustí proces vyzvednutí objektu. Po sjezdu manipulátoru do polohy pro vyzvednutí pak spustí simulované uchopení objektu a zvedne lis, následuje přesun do stavu 1, kde přebírá iniciativu jiný stav. V rámci řídicího kódu jsou vytvořeny funkční bloky pro jednotlivé operace *FB_Operation*, pro přípravu pracoviště *FB_SETUP* a materiálový tok *FB_SPAWN*. Všechny tyto funkční bloky jsou volány z programu *MAIN*. Samotný program zabývající se funkcí bezpečnosti – Safety je řešen zvlášť a jeho funkcí je kontrolovat správnost signálů bezpečnostních prvků. V případě negativního signálu bezpečnostního prvku má pak bezpečnostní funkce za úkol bezpečně zastavit operaci.

5.3.5 HMI – uživatelské rozhraní

Pro ovládání virtuálně zprovozněné buňky bylo vytvořeno základní uživatelské rozhraní, umožňující ovládat základní funkce výrobního systému. Ve vývojovém prostředí TwinCAT není nutné předem stanovit konkrétní displej, ale je vytvořena ovládací plocha se stanovenými parametry jako jsou rozlišení a poměr stran. Pro tuto práci bude uživatelské rozhraní vytvořeno s rozlišením 800x640, a tedy v poměru 4:3.

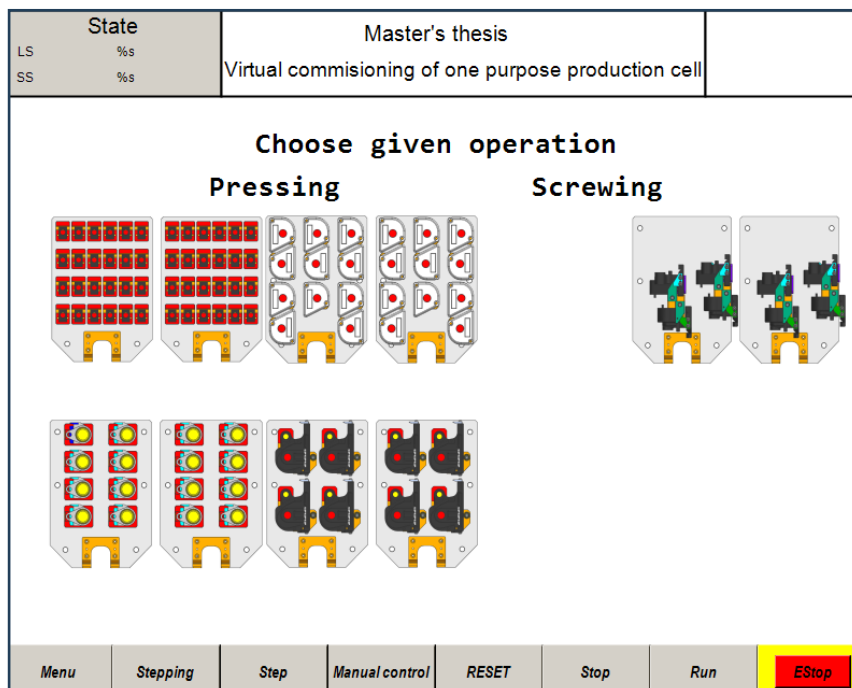
Ovládání této buňky bude naprogramováno na následující režimy: automatické, postupné a ovládání ruční. Automatický režim zaručuje úplné provedení zadané operace při splnění podmínky plynulého chodu – zmáčknuté tlačítko *Run*. Dalším režimem ovládání je postupné, kdy při zamáčknutém tlačítku *Stepping* dochází k čekání na potvrzení tlačítkem *Step*, které posune program na následující krok. Poslední je režim ruční, který umožní pohyb buďto metodou *jogging* nebo *itching*, lze jej vyvolat stisknutím tlačítka *Manual control* a slouží převážně pro najetí a potvrzení správnosti definovaných souřadnic. *Jogging* je plynulý, zrychlovaný pohyb v jedné či více osách, který je prováděn, dokud je zmáčknuté příslušné tlačítko. *Itching* na druhou stranu dovolí uživateli nastavení diference mezi aktuální polohou a následnou, v tomto případě jsou kroky nastaveny na 0.1, 1 a 10 milimetrů. Zmáčknutí tlačítka směru pak zaručí posunutí pojezdu o tuto vzdálenost daných směrem. Manuální obrazovku lze vidět na obr. 43, kde lze vlevo nahoře vidět přepínání mezi metodami *Jogging* a *Inching*, vpravo nahoře nastavení diference, neboli kroku, mezi jednotlivými posuvy. Ovládání je pak zajištěno barevně synchronizovanými tlačítky pro

ovládání os: X+, X-, Y+, Y-, Z+ a Z- spolu s integrovanými obrázky souřadných systémů pracovišť. Ve spodní části se pak nachází obecné menu stejné, či podobné, pro všechny obrazovky. Všechny vytvořené obrazovky lze vidět v příloze 2 této diplomové práce.



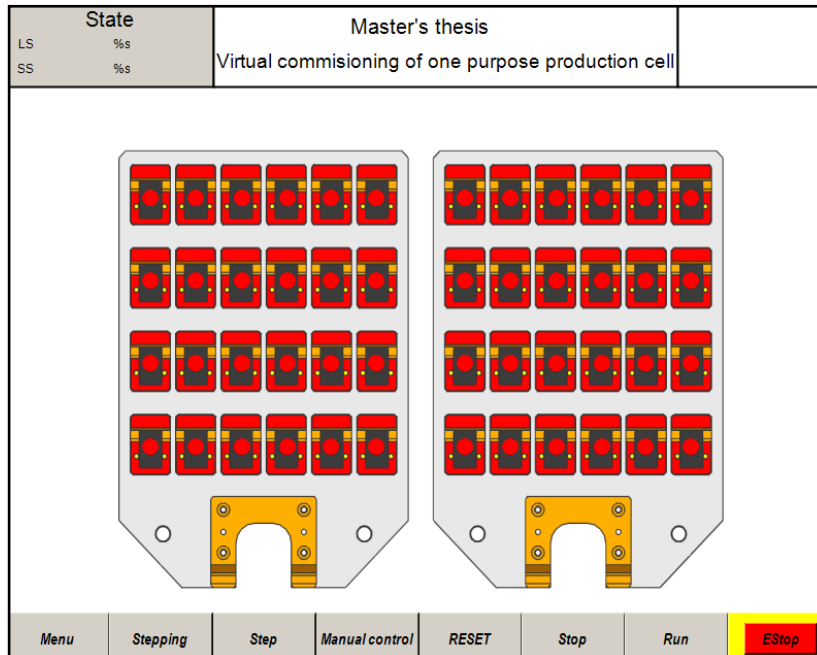
Obr. 43 Ukázka obrazovky – manual

V případě užívání automatického režimu je uživatel přenesen na obrazovku výběru operace. Tato operace funguje jako způsob volby hodnoty signálu, který by byl v reálném provozu používán pro stanovení operační aplikace. Obrazovku lze vidět na obr. 44.



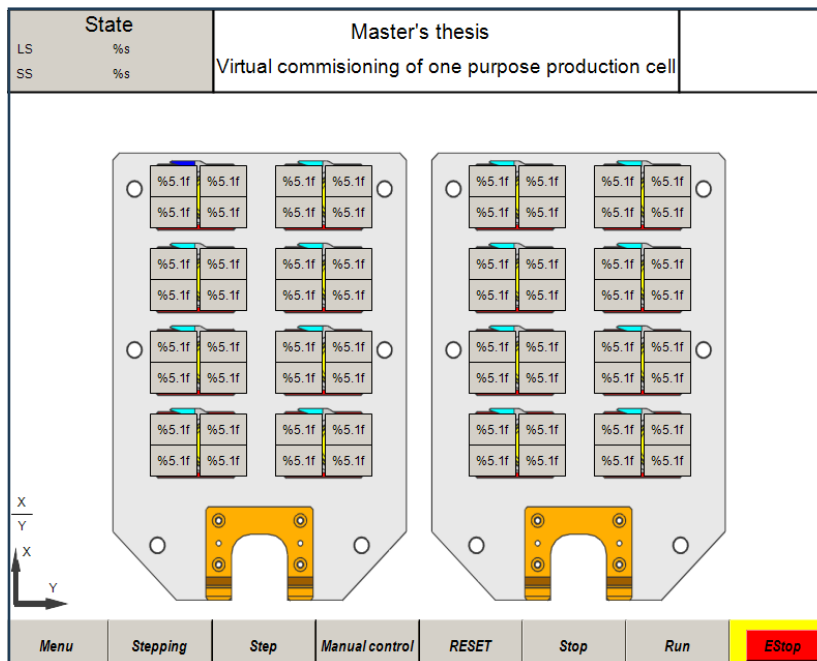
Obr. 44 Ukázka obrazovky pro výběr operace

V případě volby některé z operací ať už lisování nebo šroubování je uživatel přenesen na následující obrazovku na obr. 45. Tato obrazovka slouží pouze k identifikaci hotových operací.



Obr. 45 Ukázka operační obrazovky

Poslední je obrazovka konfigurační, která slouží pro úpravu souřadných hodnot jednotlivých těles. Po kliknutí na některé z šedých polí uprostřed paletky dojde k možnosti zapsání hodnoty od -5 do 5 milimetrů, která bude sloužit pro upřesnění polohy lisovaného nebo šroubovaného dílce.



Obr. 46 Ukázka konfigurační obrazovky

6 ZHODNOCENÍ A DISKUZE

V této diplomové práci bylo virtuálně zprovozněno jednoúčelové stanoviště pro lisování a šroubování. V rámci zprovoznění byl upraven model pracoviště, definován virtuální simulační model pracoviště a vytvořen řídicí program spolu s uživatelským rozhraním.

Virtuální zprovoznění slouží ke kontrole a validaci funkčnosti tohoto pracoviště. Díky této práci lze zkoumat chování virtuálních pracovišť, která v podstatě zastupují jejich reálné entity.

Při vypracování práce bylo překonáno několik obtíží: ladění simulace z pohledu výkonu, kolizí či problematika se samotným modelem. Hlavním problémem byla výpočetní náročnost kladená na simulační počítač. Simulace, i po nespočetném optimalizování, stále nabývala časového zpoždění, které je způsobené převážně velkým množstvím simulovaných pracovních objektů. Problematika kolizí těles velmi souvisí s kolizní přesností, kdy vyšší přesnost vyžaduje větší výpočetní výkon. Zároveň se při snižující přesnosti začínají vyskytovat nepredikovatelné kolizní stavy, které havarují simulaci.

K podobné situaci došlo i v případě upínání lisovacích hlav, které kvůli jejich konvexnímu tvaru v místě úchopu vyžadovaly až enormní množství popisujících elementů. Jejich nadefinování mělo za důsledek téměř okamžitou časovou desynchronizaci mezi řídicím a simulačním programem. Hlavním cílem řešení této problematiky bylo najít správnou rovnováhu mezi výpočetní náročností simulace a její důvěryhodností. Nicméně simulace proběhla kompletně a mimo časového zpoždění bez obtíží, proto se dá říci, že rovnováha byla nalezena.

Vytvoření řídicího programu včetně ovládacího rozhraní proběhlo bez větších obtíží. Důmyslným řešením bylo maticové definování souřadnic jednotlivých pozic, které výrazně ulehčily samotné psaní programu. V rámci této definice byly vytvořeny i konfigurovatelné odchylky pro jednotlivé pozice, které umožňují změnu jejich souřadnic v případě vzniku vychýlení.

Vhodným zlepšením pro samotné pracoviště by bylo přidání koncových snímačů pro referování polohy na obou pracovištích. Případně změna konstrukčního návrhu třetích paletek tak, aby byly všechny lisovací souřadnice uvnitř pracovního prostoru manipulátoru.

Za pomoci těchto zlepšení lze konstatovat smysluplnost aplikace virtuálního zprovoznění v průmyslu. Tato získaná zpětná vazba pro konstrukci zlepšuje samotný návrh pracoviště ještě před jeho uvedením do výroby, jak bylo zmíněno v druhé kapitole této práce.

7 ZÁVĚR

Diplomová práce se zabývala problematikou řídicích systémů a virtuálního zprovoznění výrobních systémů. Nejdříve byly sepsány rešeršní kapitoly zabývající se právě problematikou řídicích systémů, vývojových prostředí a jejich vývojářů. Následně rešerše pokračovala virtuálním zprovozněním, programy řešícími tuto problematiku a komunikačními protokoly v NX MCD.

Praktická část diplomové práce sestává z kombinace tří různých programů: OPC UA server, TwinCAT 3 a NX MCD. Vzájemným propojením došlo k vytvoření virtuálního zprovoznění výrobního celku. Zprovozněnou entitou je v práci jednocelová buňka robotizovaného pracoviště pro montáž ocelového páskovače od společnosti Feifer.

Nejdříve byl v práci věnován čas přípravě samotného modelu pracoviště. V rámci přípravy byly modely lisovacího a šroubovacího pracoviště zjednodušeny pro snížení výpočetní náročnosti, tj. odstranění nepodstatných komponent ku příkladu spojovací materiál. Na šroubovací pracoviště byl přidán model podavače šroubů. Nezprovožňované části pracoviště robotizované buňky byly z vizualizace odstraněny.

Po přechodu do NX MCD je popsán způsob definice virtuálního simulačního modelu pracoviště za pomoci tuhých a kolizních těles. Následuje popis použitých vazeb, tj. rotační, posuvná a pevná, spolu s vysvětlením možností jejich nastavení. Po definici jednotlivých vazeb následuje zprovoznění materiálového toku – přísun kluzných pouzder, kolíků a šroubů. Dalším krokem virtuálního zprovoznění je definice akčních členů jako jsou pneumatické válce, kontroléry pojezdových os či senzorky celého pracoviště. Byla vytvořena signálová struktura ve snaze nejvěrnějšího popisu případné reálné komunikace. Posledním krokem v programu NX MCD bylo uvedení operací v rámci simulace do chodu a jejich provázání na signálovou strukturu.

Dříve popsaná signálová struktura byla přepsána do proměnných v řídicím programu tak, aby došlo k mapování jednotlivých signálů skrze OPC UA server, jenž je nutností pro správné fungování a komunikaci mezi zbývajícími programy. Při přepisování signálů byl kladen důraz na dodržení jejich přesného značení.

Poslední část zahrnovala programování samotného řídicího kódu v prostředí programu TwinCAT 3. Samotný program byl rozdělen na hlavní stavový automat a dílčí funkční bloky. V rámci řídicího programu byla také nakonfigurována softwarově řízená bezpečnost pracoviště včetně tlačítka nouzového zastavení. Celý program lze ovládat pomocí uživatelského rozhraní, kde bylo vytvořeno několik obrazovek, které buďto sledují aktuální dění nebo umožňují konfiguraci jednotlivých pozic.

Virtuálního zprovoznění bylo zdárně dosaženo a veškeré dílčí cíle této práce byly splněny.

8 SEZNAM POUŽITÝCH ZDROJŮ

- [1] *PLC training for beginners*. Online. PLCynergy. 2022. Dostupné z: <https://plcynergy.com/plc-training-for-beginners/>. [cit. 2024-01-05].
- [2] *How PLCs work*. Online. PLCdev. 2005. Dostupné z: <https://www.plcdev.com>. [cit. 2024-01-06].
- [3] *Vše o průmyslu*. Online. CONTROL ENGINEERING ČESKO. Strukturování podle IEC 61131-3: 7 kroků k úspěchu. 2022. Dostupné z: <https://www.vseoprmyslu.cz>. [cit. 2024-01-10].
- [4] *Online manuál pro konfiguraci PLC řídicího systému*. Online. Beckhoff. 2024. Dostupné z: https://infosys.beckhoff.com/index_en.htm. [cit. 2024-01-20].
- [5] *Programovací režimy pro PLC dle IEC 61131-3 (CoDeSys)*. Online. Automatizace.hw.cz. Dostupné z: <https://automatizace.hw.cz/programovaci-rezimy-pro-plc-dle-iec-61131-3-codesys>. [cit. 2024-02-27].
- [6] *Princip činnosti PLC*. Online. Eluc.ikap.cz. Dostupné z: <https://eluc.ikap.cz/verejne/lekce/967>. [cit. 2024-03-02].
- [7] *Totally Integrated Automation Portal – Always ready for tomorrow*. Online. Siemens.com. Dostupné z: <https://www.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html>. [cit. 2024-03-10].
- [8] *PLC Basics – PLC Systems and More!*. Online. Plctable.com. Dostupné z: <https://www.plctable.com/>. [cit. 2024-03-11].
- [9] LIU, Zheng; SUCHOLD, Nico a DIEDRICH, Christian. Virtual Commissioning of Automated Systems. Online. *Automation*. 2012, s. 132-148. ISBN 978-953-51-0685-2. Dostupné z: <https://doi.org/10.5772/45730>. [cit. 2024-04-10].
- [10] *MODEL-IN-THE-LOOP (MIL)*. Online. <https://www.isg-stuttgart.de>. 2024. Dostupné z: <https://www.isg-stuttgart.de/en/solutions/wiktionary/model-in-the-loop-simulation>. [cit. 2024-04-20].
- [11] *Process Simulate*. Online. SIEMENS. Plm.automation.siemens.com. 2011. Dostupné z: https://www.plm.automation.siemens.com/en_gb/Images/7457_tcm642-80351.pdf. [cit. 2024-04-10].
- [12] ABB. *RobotStudio Suite*. Online. New.abb.com. 2024. Dostupné z: <https://new.abb.com/products/robotics/cs/robotstudio>. [cit. 2024-04-22].
- [13] SIEMENS. *NX Mechatronics concept designer*. Online. Plm.sw.siemens.com. 2024. Dostupné z: https://plm.sw.siemens.com/en-US/nx/cad-online/automation/mechatronic-design/?pk_vid=8acf3691f01745f9bd95de986625fa6e1715497738c61288. [cit. 2024-04-24].

- [14] SIEMENS. *Collision body*. Online. Siemens.com. 2020. Dostupné z: <https://docs.sw.siemens.com/en-US/doc/209349590/PL20200507135732916.mechatronics/id1107710>. [cit. 2024-04-26].
- [15] SIEMENS. *Address critical design, engineering and manufacturing challenges with Advanced Machine Engineering*. Online. Siemens.com. 2024. Dostupné z: <https://webinars.sw.siemens.com/en-US/address-manufacturing-challenges-advanced-machine-engineering/>. [cit. 2024-05-01].
- [16] Carbonell I Vilaplana, B. Usage of a mechatronics software for efficient machine development [en línia]. Projecte Final de Màster Oficial, UPC, Escola Tècnica Superior d'Enginyeria Industrial de Barcelona, Departament d'Enginyeria Mecànica, 2020. [Consulta: 22 Maig 2024]. Disponible a: <http://hdl.handle.net/2117/187836>.
- [17] *CO JE OPC? OPC SERVER? OPC KLIENT?* Online. Foxon.cz. 2013. Dostupné z: <https://foxon.cz/blog/prakticka-teorie/159-co-je-opc-opc-server-opc-klient>. [cit. 2024-05-02].
- [18] *Protocols Tailored to Your Application*. Online. Opal-rt.com. 2024. Dostupné z: <https://www.opal-rt.com/software-communication-protocols/>. [cit. 2024-05-02].
- [19] SIEMENS. *NX Mechatronics concept designer - External communication profinet*. Online. Scribd.com. 2024. Dostupné z: <https://www.scribd.com/document/668082299/Case-Profinet>. [cit. 2024-05-04].
- [20] *TCP VS UDP*. Online. Profiber.eu. 2024. Dostupné z: <https://www.profiber.eu/cz/aplikace/tcp-vs-udp/>. [cit. 2024-05-04].
- [21] *What is virtual comissioning?* Online. Visualcomponents.com. 2024. Dostupné z: <https://www.visualcomponents.com/resources/blog/increasing-control-software-quality-with-virtual-commissioning/>. [cit. 2024-05-20].
- [22] SIEMENS. *SIEMENS*. Online. Siemens.com. 2024. Dostupné z: <https://www.siemens.com/global/en.html>. [cit. 2024-05-24].

9 SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK

9.1 Seznam zkratk

| | | |
|----------|--|---------------------------------|
| IoT | Internet of things | Internet věcí |
| AI | Artificial intelligence | Umělá inteligence |
| PLC | Programmable logic controller | Programovatelný logický automat |
| DCS | Distributed control system | Distribuovaný řídicí systém |
| I/O | Input/Output | Vstupy / Výstupy |
| HMI | Human machine interface | Rozhraní člověk – stroj |
| SCADA | Supervisory control and data acquisition | Supervizní řízení a sběr dat |
| IDE | Integrated development environment | Vývojové prostředí |
| CAD | Computer aided design | Počítačově podporovaný návrh |
| MiL | Model in the loop | Model ve smyčce |
| SiL | Software in the loop | Software ve smyčce |
| HiL | Hardware in the loop | Hardware ve smyčce |
| PS | Process simulate | - |
| MCD | Mechatronics concept designer | - |
| OLP | Open license program | - |
| UA | Universal architecture | - |
| DA | Data access | - |
| COM/DCOM | Distributed component object model | - |
| SHM | Shared memory | Sdílená paměť |
| TCP | Transmission control protocol | - |
| UDP | User datagram protocol | - |
| GVL | Global variable list | List globálních proměnných |

9.2 Seznam tabulek

| | | |
|--------|---|----|
| Tab. 1 | Výrobci jednotlivých PLC systémů [8]..... | 20 |
| Tab. 2 | Zpracovávaná kluzná pouzdra..... | 35 |
| Tab. 3 | Zpracovávané kolíky | 36 |
| Tab. 4 | Jednotlivé lisovací hlavy | 37 |
| Tab. 5 | Typy kolizních těles, jejich přesnost, spolehlivost, vliv na simulaci [14]..... | 45 |

9.3 Seznam obrázků

| | | |
|---------|--|----|
| Obr. 1 | Zjednodušená ukázka dopadu virtuálního zprovoznění [21] upraveno | 19 |
| Obr. 2 | Ukázka nosiče PLC, upraveno [2] | 21 |
| Obr. 3 | Ukázka PLC, upraveno [2]..... | 21 |
| Obr. 4 | Architektura PLC, upraveno [2]..... | 21 |
| Obr. 5 | a) Strukturovaný text ST, b) žebříkový diagram LD | 22 |
| Obr. 6 | a) Funkční bloková schémata FBD, b) Seznam instrukcí IL, c) Sekvenční funkční diagram SFC | 23 |
| Obr. 7 | Skenovací perioda | 23 |
| Obr. 8 | Ukázka komunikace výrobního systému | 24 |
| Obr. 9 | Koncept virtuálního zprovoznění, upraveno [16] | 24 |
| Obr. 10 | HiL, SiL, MiL, upraveno [10]..... | 25 |
| Obr. 11 | Portfolio siemens pro digitální továrnu, upraveno [15] | 26 |
| Obr. 12 | Ukázka NX MCD [13] | 27 |
| Obr. 13 | Ukázka Process simulate [11] | 28 |
| Obr. 14 | Komunikační struktura OPC [16] | 29 |
| Obr. 15 | Komunikační struktura Profinet upraveno [19] | 29 |
| Obr. 16 | Komunikační struktura PLCSIM Advanced [22] | 30 |
| Obr. 17 | Komunikační struktura SHM [22] | 30 |
| Obr. 18 | Ukázka výrobní buňky | 33 |
| Obr. 19 | Ukázka zprovozňovaných buněk | 33 |
| Obr. 20 | Schématický obrázek páskovače BO-7 SWING..... | 34 |
| Obr. 21 | Ukázka jednotlivých paletok a) zvedáček, b) víčko, c) ojnice, d) těleso, e) těleso šroubování | 38 |
| Obr. 22 | Ukázka šroubovací stanice 1) Šroubovací Z osa, 2) Osa X, 3) Osa Y, 4) Zásobník šroubovacích hlav, 5) Zásobník šroubů, 6) Umístění paletok..... | 39 |
| Obr. 23 | Ukázka lisovací stanice 1) Servopohon os, 2) Gravitační zásobník, 3) Osa Y, 4) Osa X, 5) Zásobník lisovacích hlav, 6) Hydraulický válec, 7) Uchycení paletok, 8) Podavač objektů, 9) Umístění paletok..... | 40 |
| Obr. 24 | Konfigurace OPC serveru | 41 |
| Obr. 25 | Nastavení OPC v rámci TwinCAT 3 | 42 |
| Obr. 26 | Správné připojení OPC serveru TwinCAT 3 | 42 |
| Obr. 27 | Ukázka nastavení NX MCD | 43 |
| Obr. 28 | Dialog tuhého tělesa – definice X pojezdu | 44 |
| Obr. 29 | Dialog kolizního tělesa..... | 46 |
| Obr. 30 | Dialogová okna jednotlivých vazeb a) fixní, b) translační, c) rotační | 47 |
| Obr. 31 | Translační, rotační a fixní vazby v simulaci | 47 |
| Obr. 32 | Dialogové okno kolizních senzorů..... | 48 |
| Obr. 33 | Ukázka akčních prvků v simulaci | 49 |
| Obr. 34 | Dialogová okna jednotlivých akčních prvků a) Pneumatický válec, b) Pneumatický ventil, c) Ovládání pozice | 49 |
| Obr. 35 | Dialogové okno signálů | 50 |
| Obr. 36 | Ukázka a) událostně řízené operace a b) dialogové okno operací | 51 |
| Obr. 37 | Ukázka mapování signálů | 52 |

| | |
|---|----|
| Obr. 38 Ukázka definice proměnných pro komunikaci s NX MCD | 53 |
| Obr. 39 Ukázka definice souřadnic | 53 |
| Obr. 40 Ukázka definice pomocí balíčku Motion a) Aktivace samotné osy X, b) Definice absolutního pohybu osy Y | 54 |
| Obr. 41 Ukázka definice bezpečnostních funkcí | 54 |
| Obr. 42 Stavový automat – stav 12..... | 55 |
| Obr. 43 Ukázka obrazovky – manual | 56 |
| Obr. 44 Ukázka obrazovky pro výběr operace | 56 |
| Obr. 45 Ukázka operační obrazovky | 57 |
| Obr. 46 Ukázka konfigurační obrazovky | 57 |

10 SEZNAM PŘÍLOH

- Příloha 1 Vývojový diagram posloupnosti operací
- Příloha 2 Ukázka uživatelského rozhraní
- Příloha 3 Video ukázka virtuálního zprovoznění