České vysoké učení technické v Praze
Fakulta strojní
Ústav přístrojové a řídicí techniky

# Simulace robotických a výrobních systémů

Bakalářská práce

*Josef Koktan*

Bakalářský program: Teoretický základ strojního inženýrství
Bakalářský obor: Bezoborový
Vedoucí práce: Mgr. Ing. Jakub Jura, Ph.D.

Praha, červenec 2021

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Koktan** Jméno: **Josef** Osobní číslo: **472967**

Fakulta/ústav: **Fakulta strojní**

Zadávající katedra/ústav: **Ústav přístrojové a řídící techniky**

Studijní program: **Teoretický základ strojního inženýrství**

Studijní obor: **bez oboru**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Simulace robotických a výrobních systémů**

Název bakalářské práce anglicky:

**Robotic and production systems simulation**

Pokyny pro vypracování:

1) Provést rešerši softwarových možností simulace robotických systémů včetně návazných funkcí (návrh, programování, vizualizace atp.)
2) Analyzovat možnosti využití rešeršovaných SW v automatických systémech dle principů Průmysl 4.0.
3) Navrhnout a otestovat možnosti využití vytvořeného modelu jakožto digitálního dvojčete pro výuku a virtuální zprovoznění
4) Porovnat korporátní a svobodné softwary.

Seznam doporučené literatury:

[1] V-rep, https://www.coppeliarobotics.com/
[2] Robotino, https://wiki.openrobotino.org/
[3] Ciros Studio, https://www.verosim-solutions.com/en/industry/ciros-studio/
[4] Robo DK, https://robodk.com/simulation

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**Ing. Mgr. Jakub Jura, Ph.D., U12110.3**

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

Datum zadání bakalářské práce: **30.04.2021** Termín odevzdání bakalářské práce: **24.08.2021**

Platnost zadání bakalářské práce: _____

| Ing. Mgr. Jakub Jura, Ph.D.<br>podpis vedoucí(ho) práce | podpis vedoucí(ho) ústavu/katedry | prof. Ing. Michael Valášek, DrSc.<br>podpis děkana(ky) |
|---|---|---|

## III. PŘEVZETÍ ZADÁNÍ

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

. _____ _____

Datum převzetí zadání Podpis studenta


© ČVUT v Praze, Design: ČVUT v Praze, VIC

**Vedoucí práce:**
Mgr. Ing. Jakub Jura, Ph.D.
Ústav přístrojové a řídicí techniky
Fakulta strojní
České vysoké učení technické v Praze
Technická 2
160 00 Praha 6
Česká republika

Copyright © Červenec 2021 Josef Koktan

# Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně s tím, že její výsledky mohou být dále použity podle uvážení vedoucího bakalářské práce jako jejího spoluautora. Souhlasím také s případnou publikací výsledků bakalářské práce nebo její podstatné části, pokud budu uveden jako její spoluautor.

V Praze 2021

.....
Josef Koktan

# Abstrakt

Cílem této bakalářské práce je provést rešerši softwarových možností simulace robotických systémů a analýza jejich využití v automatických systémech dle principů Průmyslu 4.0. Dále se bude zabývat tématikou tvorby a virtuálním zprovoznění digitálního dvojčete ve vybraných softwarech. V rámci práce bude také provedeno porovnání jednotlivých softwarů, open source i korporátních.

**Klíčová slova:** Digitální dvojče, simulace, automatizace výroby, Průmysl 4.0, virtuální zprovoznění

# Abstract

The aim of this bachelor thesis is to make research about software options of simulations of robotic systems and analysis of their use in automatic systems by concepts of Industry 4.0. Further focus is on topic of creation and virtual commissioning of digital twin in selected software. As another case of this thesis is done a comparison of individual software, open source as well as corporate.

**Keywords:** Digital twin, simulation, manufacturing automation, Industry 4.0, virtual commissioning

# Poděkování

Děkuji Mgr. Ing. Jakubovi Jurovi, Ph.D., za jeho konzultace, rady a podněty, které mi během vypracovávání této práce poskytl.

Děkuji své rodině, přátelům a přítelkyni za velkou trpělivost a stále trvající podporu během celého studia.

# Seznam obrázků


2.1 Průmyslové revoluce [3] . . . 3
2.2 Horizontální/vertikální integrace[6] . . . 5
2.3 Pyramida automatizace dle normy IEC 62264[9] . . . 6
2.4 Enterprise resource planning [11] . . . 7
2.5 Funkce MES dle MESA 11[12] . . . 9
2.6 Diagram propojení SCADA systému [14] . . . 10
2.7 Schéma propojení digitálního dvojčete s fyzickým systémem[18] . . . 11
2.8 Proces vývoje bez a s virtuálním zprovozněním [19] . . . 12
2.9 Propojení HiL a SiL[20] . . . 13
2.10 Zjednodušené schéma procesu simulace [21] . . . 14

4.1 Uživatelské rozhraní robotického simulátoru V-REP[25] . . . 17
4.2 : Všechny typy objektů, které je možné využít ve V-REP [25] . . . 18
4.3 : Rozhraní pro manipulaci s objekty [25] . . . 19
4.4 : Hierarchie vlastního modelu a modelu z knihovny [25] . . . 21
4.5 : Ukázka dynamické simulace [25] . . . 22
4.6 : Ovládací panel simulace [25] . . . 24
4.7 : Propojení Tip a Target ve stromu robotu Franka[25] . . . 26

5.1 : Ovládací panel simulace [29] . . . 28
5.2 : Model explorer [29] . . . 29
5.3 : Příklad programování pozic robota v Melfa basic IV [29] . . . 29

6.1 : Uživatelské rozhraní [30] . . . 31
6.2 : Seznam tagů aktivních objektů [30] . . . 32
6.3 : Třídění krabic [30] . . . 33
6.4 : Část logického zapojení [30] . . . 34
6.5 : Zvedni a polož [30] . . . 34
6.6 : Logické zapojení úchopu [30] . . . 35

# Výpisy kódu

4.1 Dopravníkový skript . . . . . 25
4.2 Zahajující část skriptu . . . . . 25
4.3 Uchopení předmětu . . . . . 26

# Seznam zkratek

**API** Application Programming Interface. 15, 16, 23, 36

**CAD** Computer aided design. 21, 27, 28, 37

**CNC** Počítačem řízený obráběcí stroj. 1, 28

**CPS** Cyber-Physical system, kyberneticko-fyzikální systém. 3

**DCS** Direct Control System. 5

**ERP** Enterprise Resource Planning/Plánování podnikových zdrojů. x, 4, 6–9

**HiL** Hardware in the loop. vii, x, 12, 13, 37

**HMI** Human Machine Interface, uživatelské rozhraní. 9

**IoT** Internet of Things. 4

**IRL** Industrial Robot Language. 29

**ISA** International Society of Automation. 5

**LPWAN** Low power wide area network, druh bezdrátové sítě. 4

**MES** Manufacturing Execution System. vii, x, 6–9

**MOM** Manufacturing Operations Management. 6, 7

**MPS** Modular Production System. 29

**PLC** Programovatelný logický automat. 3, 5, 9–11, 13, 27–33, 37, 38

**ROS** Robot Operating System. 23, 36

**SCADA** Supervisory Control and Data Acquisition. x, 5, 9

**SiL** Software in the loop. vii, x, 12, 13

# Obsah

**Prohlášení** iv

**Abstrakt** v

**Poděkování** vi

**Seznam obrázků** vii

**Seznam zkratek** ix

**1 Úvod** 1

**2 Průmysl 4.0** 3
2.1 Pyramida automatizace . . . 5
2.1.1 Plánování podnikových zdrojů - ERP . . . 6
2.1.2 Výrobní informační systémy - MES . . . 7
2.1.3 Dispečerské řízení a sběr dat - SCADA . . . 9
2.2 Digitální dvojče . . . 10
2.3 Virtuální zprovoznění . . . 11
2.4 Metody virtuálního zprovoznění . . . 12
2.4.1 Hardware ve zpětné vazbě - HiL . . . 13
2.4.2 Software ve zpětné vazbě - SiL . . . 13
2.5 Simulace . . . 13

**3 Výběr softwaru** 15
3.1 Robotino . . . 15
3.2 Robo DK . . . 15

**4 V-rep** 16
4.1 Uživatelské rozhraní . . . 16
4.2 Objekty . . . 17
4.2.1 Manipulace s objekty . . . 19
4.2.2 Vlastnosti objektů . . . 20
4.2.3 Tvorba vlastního modelu . . . 20
4.3 Dynamický modul . . . 21
4.4 Programování . . . 23
4.4.1 Skripty . . . 23
4.5 Simulace . . . 24
4.6 Vytvořená scéna . . . 25

**5 Ciros Studio** 27
5.1 Uživatelské rozhraní . . . 27
5.2 Modelování . . . 28
5.3 Programování . . . 29

5.4 Simulace . . . . . 30

**6 Factory I/O** **31**
6.1 Uživatelské rozhraní . . . . . 31
6.2 Objekty . . . . . 32
6.3 Řízení . . . . . 32
6.3.1 Třídění krabic . . . . . 33
6.3.2 Zvedni a polož . . . . . 34

**7 Srovnání simulačních softwarů** **36**
7.1 V-rep . . . . . 36
7.2 CIROS Studio . . . . . 37
7.3 Factory I/O . . . . . 38

**8 Závěr** **39**

**Zdroje** **41**

# 1 Úvod

Tato bakalářská práce se zabývá možnostmi virtuálního zprovoznění a simulace robotických systémů. Jejím cílem je provést rešerši softwarů pro vytvoření simulace a zprovoznění virtuálního dvojčete. Dále obě činnosti v daných programech realizovat. Na základě získaných znalostí, z průběhu vytváření rešerše a práce na samotném digitálním dvojčeti, byli porovnány jednotlivé softwary a vybrán ten, který je nejvhodnější pro potřeby simulací výrobních systémů na základě očekávání, která od těchto softwarů máme.

Vývoj nového výrobního systému, výrobní linky obsahující například CNC s řídicím systémem, prochází postupně několika fázemi, od prvotní volby parametrů systému, přes návrh a konstrukci jednotlivých komponent, jejich výrobu, nákup dodatečných dílů, volbu řídicího systému, jeho nastavení, modifikaci a programování rozšířených funkcí, až po montáž. Následuje montáž stroje a poté fáze zprovoznění, při které se ověřuje správnost postupu v předchozích fázích vývoje, probíhá kontrola funkčnosti konstrukce, správné zapojení všech zařízení, provádí se nahrání softwaru, nastavení strojních parametrů řídicího systému, a je nutné otestovat správné naprogramování softwaru a stroj se chová podle očekávání. Během testovací fáze může dojít k poškození stroje například chybným naprogramováním softwaru. Možné problémy v softwaru vedou k opravám a opětovnému testování, což působí průtahy celého vývoje. Zjištěné nedostatky v konstrukci stroje mohou způsobit další odklad termínu dokončení a finanční navýšení projektu.

V dnešním globálně propojeném konkurenčním prostředí jsou na výrobu kladeny stále vyšší nároky. Postup uvedený výše již není dostatečně rychlý, aby výrobce uváděl na trh inovované výrobky nebo byl schopen dodat specializované stroje upravené podle přání zákazníka ve stejném čase jako konkurence. Řešením těchto požadavků je digitalizace výrobních procesů, která s nástupem éry průmyslu 4.0 stále více zaujímá v oblasti průmyslového inženýrství důležitou úlohu. Digitalizace umožňuje mít v rámci firmy lepší kontrolu nad procesy v rámci firmy a zjednodušuje komunikace mezi všemi týmy, které se podílí na chodu společnosti. Jedním z nástrojů digitalizace je také simulace výrobních a nevýrobních procesů. Ta umožňuje transformovat jak jednotlivé linky, tak i celé výrobní systémy do formy modelů a vytvořit takzvané digitální dvojče daného systému. Tato digitální forma má za cíl vizualizaci systémů pro možnost jejich prohlížení v širším kontextu bez nutnosti je procházet osobně nebo například pomocí virtuální reality zaučovat obsluhu jednotlivých výrobních systému pro práci s nimi. V neposlední řadě také urychluje proces projektování těchto systémů, neboť díky simulaci je možno tyto systémy modelovat na počítači, zkoumat jejich funkčnost efektivitu mimo prostředí výroby a samozřejmě také

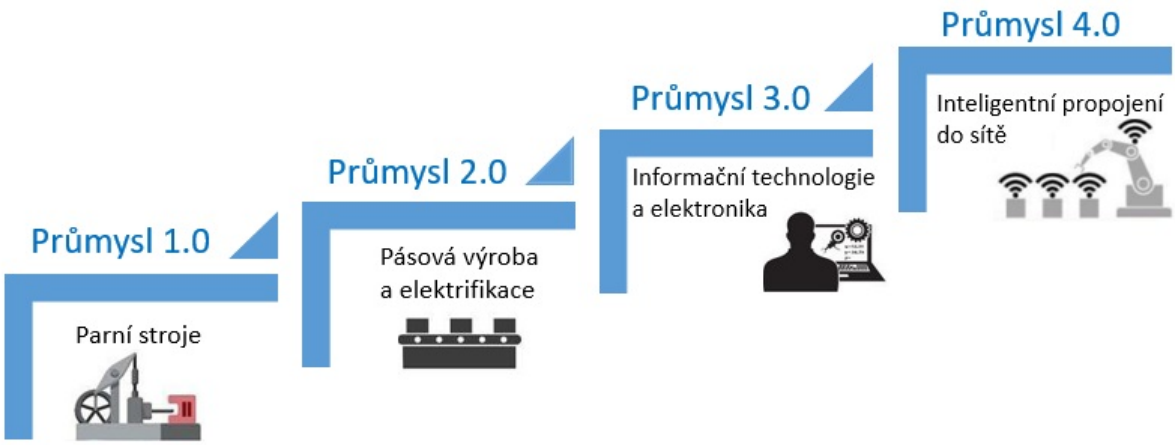
tyto systémy upravovat a optimalizovat bez zásahu do reálných systémů. To pak urychluje tvorbu nových systémů, neboť je možné najít a vyladit všechny chyby ještě předtím, než daná linka stojí fyzicky na svém místě. To samozřejmě vede ke zrychlení projektování a výrazné úspoře vynaložených časových a z toho plynoucích finančních prostředků a také zamezuje nutnosti při úpravách chod systémů omezit, či úplně zastavit, což je samozřejmě nežádoucí jev.[1]

Proto je digitalizace interních procesů pro výrobce v podstatě nutným předpokladem pro zefektivnění výroby, šetření nákladů a tím udržení pozice významného a relevantního dodavatele produktů. Jedním z nástrojů digitalizace je právě i z výše uvedených důvodů proces simulací. Díky možnostem moderní výpočetní techniky dnes na trhu figuruje řada poskytovatelů simulačních softwarů různých kvalit s odlišnými cenami. Je proto nutné při výběru vhodného softwaru postupovat pečlivě a důkladně zvážit všechna pro a proti. Právě z tohoto důvodu vznikla tato práce.

# 2 Průmysl 4.0

Pojem Průmysl 4.0 má označovat čtvrtou průmyslovou revoluci, ovšem spíše než o revoluci jde o postupnou evoluci. Rozvoj v počítačové technice a vývoj programovatelných logických systémů - PLC v minulosti vedl k rozvoji automatizace výroby.

Jednotlivé prostředky automatizace – stroje, PLC či senzory – jsou pak také s těmito systémy propojeny v jednotné síti. Nárůst výpočetního výkonu umožnuje pracovat s obrovským množstvím dat, umožnuje aplikovat prvky umělé inteligence ve formě neuronových sítí čímž vzniká kyberneticko-fyzický systém výroby. Tento a další trendy jsou v současnosti označovány pojmem Průmysl 4.0.[2]




Obrázek 2.1: Průmyslové revoluce [3]

Pan profesor Vladimír Mařík charakterizuje Průmysl 4.0 takto: „Průmysl 4.0 transformuje výrobu ze samotných automatizovaných jednotek na plně integrovaná automatizovaná a průběžně optimalizovaná výrobní prostředí. Vzniknou nové globální sítě založené na propojení výrobních zařízení do kyberneticko-fyzických systémů - CPS (Cyber-Physical Systems).

CPS budou základním stavebním prvkem "inteligentních továren", budou schopny autonomní výměny informací, vyvolání potřebných akcí v reakci na momentální podmínky a vzájemné nezávislé kontroly.“[1]

Průmysl 4.0 zastřešuje řadu dílčích konceptů. Mezi ně patří například:

- **Široká digitalizace** – vytváření digitálních dvojčat jednotlivých výrobků, výrobních strojů i celých výrobních závodů.
- **Big data** – Velká data. Sběr obrovského množství dat z velkého počtu zařízení propojených informační sítí v reálném čase a jejich automatizované zpracování
- **IoT** – internet věcí. Zařízení jsou navzájem propojena prostřednictvím internetu, mohou k tomu být využity například specializované technologie typu LPWAN, které mají velký dosah, nízkou přenosovou kapacitu a jsou energeticky úsporné. Také lze využít mobilní datové sítě.[4]
- **Automatizace** – náhrada lidí nejen v oblasti fyzické práce, ale díky nasazení umělé inteligence i v oblasti kognitivní práce
- Automatizovaná optimalizace výrobních procesů
- Zvyšování a kontrolování kvality výrobků
- Možnost výroby vysoce kustomizovaných výrobků za cenu výrobků vyráběných sériově

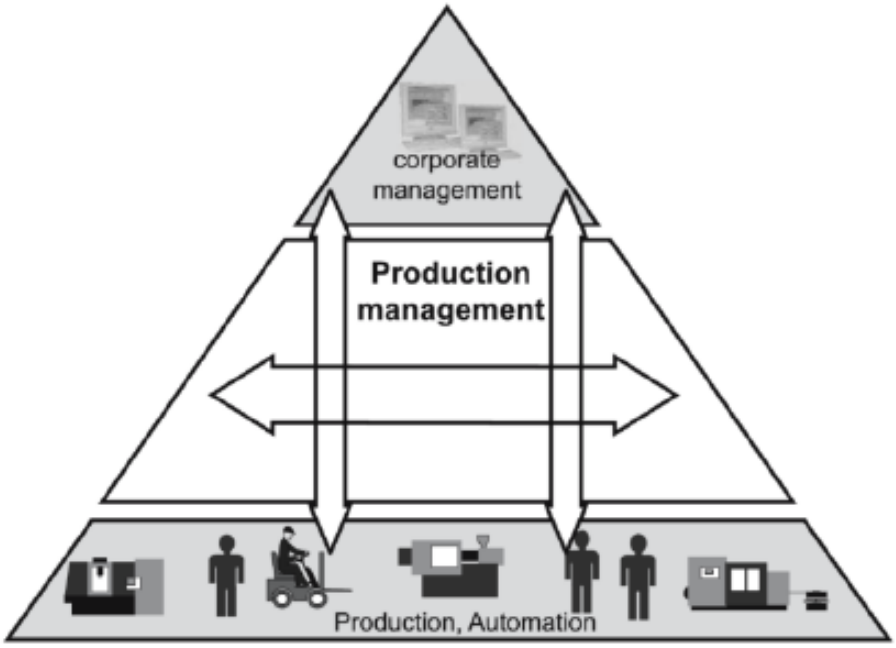
Ztělesněním konceptu je takzvaná "chytrá továrna", která místo centrálního řízení využívá decentralizovanou síť vzájemně komunikujících téměř samostatných strojů.[1] Všechny zúčastněné prvky tohoto cyklu, tedy pracovníci, výrobky, vstupní materiály a systémy, musí být integrovány jako inteligentní, samostatné jednotky v reálném či téměř reálném čase.

Integraci lze rozdělit na 3 hlavní směry, které jsou předpokladem pro fungování celého systému.

**Vertikální integrace** zajišťuje vzájemné propojení všech úrovní výrobní struktury napříč organizací. Probíhá na úrovni výrobního podniku, továrny. Je třeba zajistit propojení úrovně senzorů a aktuátorů skrze veškeré nadřazené vrstvy až po úroveň plánování podnikových zdrojů (ERP systémů). Dále se také počítá s rozvojem modularizace a možností rekonfigurace těchto propojených systémů.[5][6]

„**Horizontální integrace** probíhá na úrovni hodnotových řetězců produktů mezi všemi subjekty, které se na nich podílí. I přestože je životní cyklus výrobku rozdělen mezi vícero výrobních zařízení, potažmo společností, data výrobku jsou navzájem provázaná. Sdílení informací napříč dodavatelským řetězcem umožňuje například snižovat skladové zásoby a dodávat výrobky „just in time“.“ [7]

**Integrace inženýrských procesů** si klade za cíl propojit veškeré inženýrské kroky v etapách životního cyklu výrobku. Digitálně tak má být propojen výzkum a vývoj s plánováním výroby, technologií, samotnou výrobou až po následné poprodejní služby. [5]



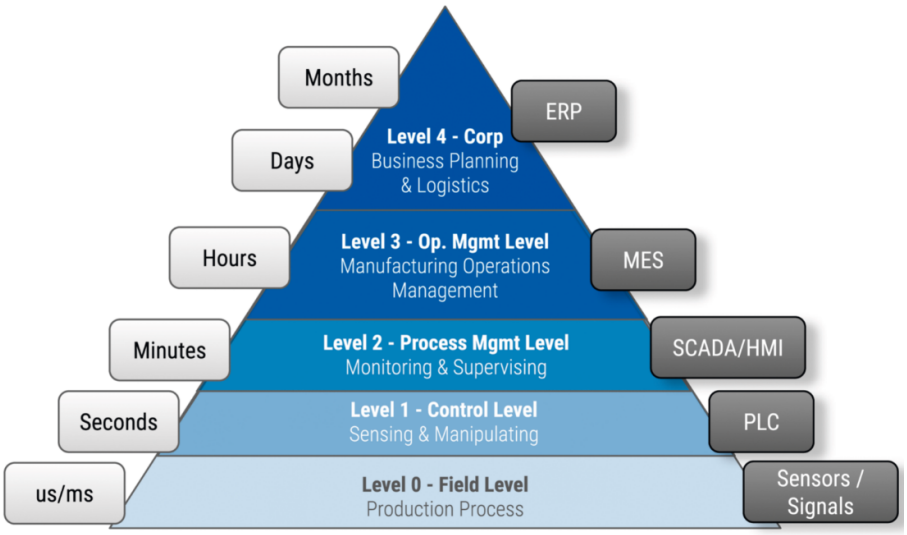

Obrázek 2.2: Horizontální/vertikální integrace[6]

## 2.1 Pyramida automatizace

Klíčovým prvkem průmyslu 4.0 je standardizace. Ačkoliv je výroba ze své podstaty velmi různorodá, lze ji standardizovat a jednotně popsat. To začala vyžadovat již dříve integrace výroby do výrobních a podnikových výpočetních systémů. Nejobsáhlejším souborem používaných norem je systém organizace International Society of Automation (ISA). Dle normy ISA 95 lze hierarchicky rozdělit činnost výrobních podniků do 5 vrstev, které společně tvoří pomyslnou pyramidu výroby.

Nultá úroveň představuje samotný výrobní proces– výrobní stroje, aktuátorů, senzory. Úroveň 1 definuje výrobní prostředky, manipulaci a kontrolu procesu realizované například PLC a DCS. Úroveň 2 monitoruje a dohlíží na výrobní procesy přes SCADA systémy(Supervisory Control and Data Acquisition). Úrovně 0, 1 a 2 tak společně zahrnují samotný výrobní proces a jeho přímé řízení. Dle typu procesu pak lze výrobu rozdělit na spojitou, dávkovou a diskrétní. [8]

Třetí úroveň představuje řízení výrobních operací MOM(Manufacturing Operations Management). Zde jsou definovány činnosti potřebné k uskutečnění fyzického procesu. V této úrovni jsou výrobní informační systémy (MES) předávány výrobní příkazy ke zpracování nižším úrovním. Na čtvrté úrovni stojí plánování celého podniku a logistika realizovaná především systémy ERP. Především funkce třetí úrovně a výměnu informací mezi ERP a MES popisuje norma ISA 95.[9]




Obrázek 2.3: Pyramida automatizace dle normy IEC 62264[9]

Rozdělení na úrovně lze chápat i v souvislosti s rozsahem, kdy 0., 1. a 2. úroveň obsahují samotné stroje, maximálně výrobní buňky, třetí úroveň obsahuje například výrobní závod či linku a čtvrtá úroveň celý podnik, který může mít výrobních závodů několik. Stejně tak rozdělení přímo souvisí s časovou náročností, kdy procesy v ERP mohou probíhat až v řádu dnů, v MOM v řádu hodin a v nižších vrstvách v řádech sekund až milisekund.[7]

### 2.1.1 Plánování podnikových zdrojů - ERP

Cílem ERP systémů je integrovat veškeré funkční oblasti podniku do jednotného informačního systému. Klasický ERP systém tak pokrývá oblasti řízení a kontroly výroby, účetnictví, financí, obchodu, logistiky, marketingu, skladového hospodářství a mnoho dalších. ERP systémy pracují s velkými objemy různorodých dat a primárně slouží pro řízení chodu podniku jako celku. Každá funkční oblast podniku zpravidla využívá vlastní modul či aplikaci uzpůsobenou pro jeho potřeby, tyto moduly jsou však provázány navzájem centrální databází a systém tak umožňuje podniková data integrovat a pracovat s nimi jako s celkem.[10]

Z hlediska výroby jsou nejdůležitější funkce správy objednávek, skladového hospodářství, plánování výroby, správy výrobních dat, kvality, údržby a další. Hlavní smysl ERP systému je jednotná integrace všech podnikových funkcí, která může představovat nevýhodu z hlediska přímého odbavování a řízení výroby na nižších podnikových úrovních.

ERP systém může být moc široce zaměřený na to, aby pokryl specifické požadavky konkrétního výrobního závodu a jeho strojního zařízení. Není primárně určen pro řízení výrobních operací. Nejen z tohoto důvodu jsou zaváděny systémy MES, které propojují podnikovou vrstvu ERP s výrobními procesy.[8]

Obrázek 2.4: Enterprise resource planning [11]

### 2.1.2 Výrobní informační systémy - MES

Kategorii MOM dělíme na řízení operací výroby, skladování, kvality a údržby. Z toho vše nebo část řeší systémy MES v závislosti na konkrétním softwarovém řešení. Je také možnost mít systémy pro správu každé této oblasti oddělené. MES systémy řídí, monitorují a synchronizují exekuci fyzických procesů spojených s výrobou v reálném čase.[6]

Na rozdíl od ERP systémů, se kterými jsou provázané, jsou zaměřeny primárně na výrobní oblast podniku. V ní realizují podněty z podnikové vrstvy a přenášejí zpět informace o výrobě. Celkově mohou MES systémy pokrývat až 11 funkčních oblastí dle organizace MESA zobrazených na Obrázku 2.5.[12]

Z funkčního pohledu tak musí MES splňovat především následující:

- **Správa výrobních zdrojů** Řídí alokaci a sledování prostředků zapojených do výroby, tj. zařízení, nástrojů, materiálu, pracovníků, energie apod. Dále zajišťuje informace o aktuálním stavu a dostupnosti zdroje pro budoucí úkoly.[13]

- **Plánování a rozvrhování výroby**

  Plánování je nezbytným elementem výroby a stejně tak je nedílnou součástí MES systému. K plánování výrovy se dá přistupovat několika způsoby například plánování podle jednotlivých zakázek, anebo plánování podle složitých genetických algoritmů. Na základě některého plánovacího způsobu je vytvořena fronta práce určující práci výrobního zdroje tak aby byla co nejvíce efektivní.[12][13]

- **Řízení a monitoring výroby**

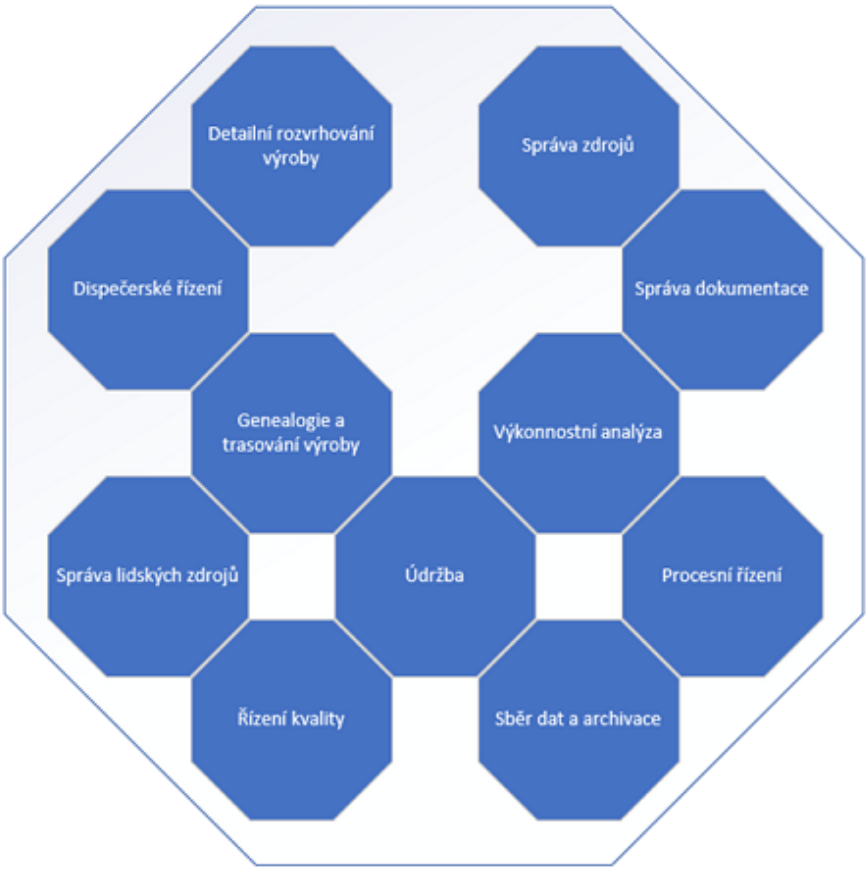
  „Řízení výroby a monitoring výroby je definováno jako souhrn aktivit řídicích tok výroby přiřazováním práce jednotlivým zařízením a osobám, zajišťování potřebného množství surovin a energie, sledování aktuálního stavu výroby, operativní řešení výpadků atd. Dále zajišťuje aktivity, které řídí výrobu specifikovanou v naplánované a uvolněné výrobě (fronta práce). Jestliže je samotné řízení výroby zabezpečeno v řídicím systému, výrobní informační systém zajišťuje kontroly zdrojů a informuje okolní systémy o aktuálním stavu výroby (odvody práce, zabezpečení kontrolních kroků výroby, atd.). Řízení výroby v MES systémech je velmi důležité vzhledem k propojení s ERP systémy a případným online zpřístupněním informací o rozpracované výrobě.“[12]

- **Sběr a úschova dat**

  Obstarává sběr a ukládání procesních a výrobních dat. Množství a typ dat závisí především na složitosti výrobního systému.

- **Výkonnostní analýzy**

  Klíčové výkonnostní ukazatele – KPI vyhodnocují úspěch podniků v jednotlivých oblastech jejich činnosti. Každý podnik má prioritu v jiných ukazatelích na základě firemní strategie. Celková efektivita zařízení (OEE) je asi nejzajímavějším ukazatelem výroby, protože složením více parametrů udává jak moc efektivně se výrobní zařízení využívání .[13]



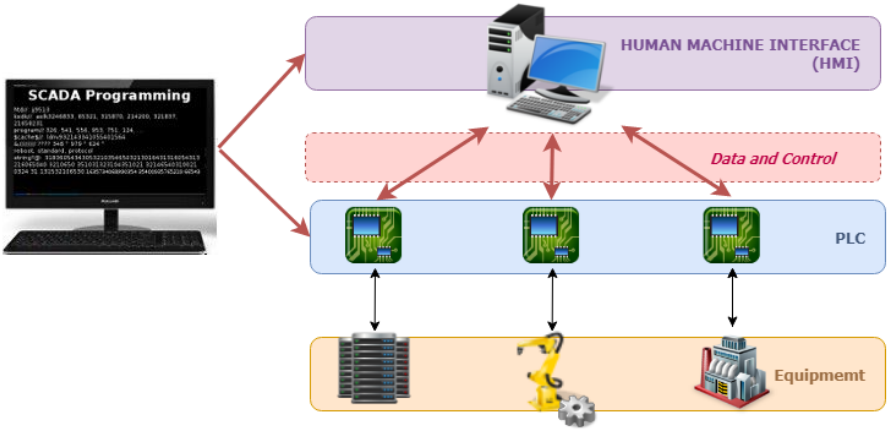

Obrázek 2.5: Funkce MES dle MESA 11[12]

Systémy MES mohou také samy obsahovat či spolupracovat s pokročilými systémy plánování výroby (Adavanced planning and scheduling, APS). Ty na základě dat o objednávkách z ERP systémů a dat o kapacitách z MES systému tvoří optimální plán výroby pro celou společnost i konkrétní výrobní zařízení. Takto vybavený MES systém je schopný zpracovávat data o výrobku, skladové zásoby a objednávku ze systému ERP a odbavit výrobu včetně personálních požadavků, požadavků na kvalitu a finálního vyhodnocení. Komplexita požadavků na systém vůbec nemusí dosahovat takovéto úrovně, existující řešení mohou být složitostí hodně různorodá, aby co nejlépe vyhovovala potřebě.[6][7]

### 2.1.3 Dispečerské řízení a sběr dat - SCADA

SCADA, zkratka z anglického Supervisory Control and Data Aquisition, jsou systémy pro dohled, řízení a sběr dat. SCADA systémy tvoří další propojovací prvek mezi samotným výrobním procesem a operačním řízením. Stojí nad prvky samotného řídícího systému strojů (PLC, HMI). Především řídí průmyslové procesy, monitorují, procesují a sbírají data v reálném čase a přímo komunikují s různými zařízeními zapojenými ve výrobě (to mohou být PLC, senzory, aktuátory, aj.).

Pokud například automaticky řídíme výrobu v celém závodě (typicky procesní průmysl), může být vyžadováno propojení do jednotného systému, který bude zobrazovat klíčová aktuální data, vyhodnocovat je a případně na pokyn pracovníka reagovat změnami požadavků pro připojená zařízení.[7]




Obrázek 2.6: Diagram propojení SCADA systému [14]

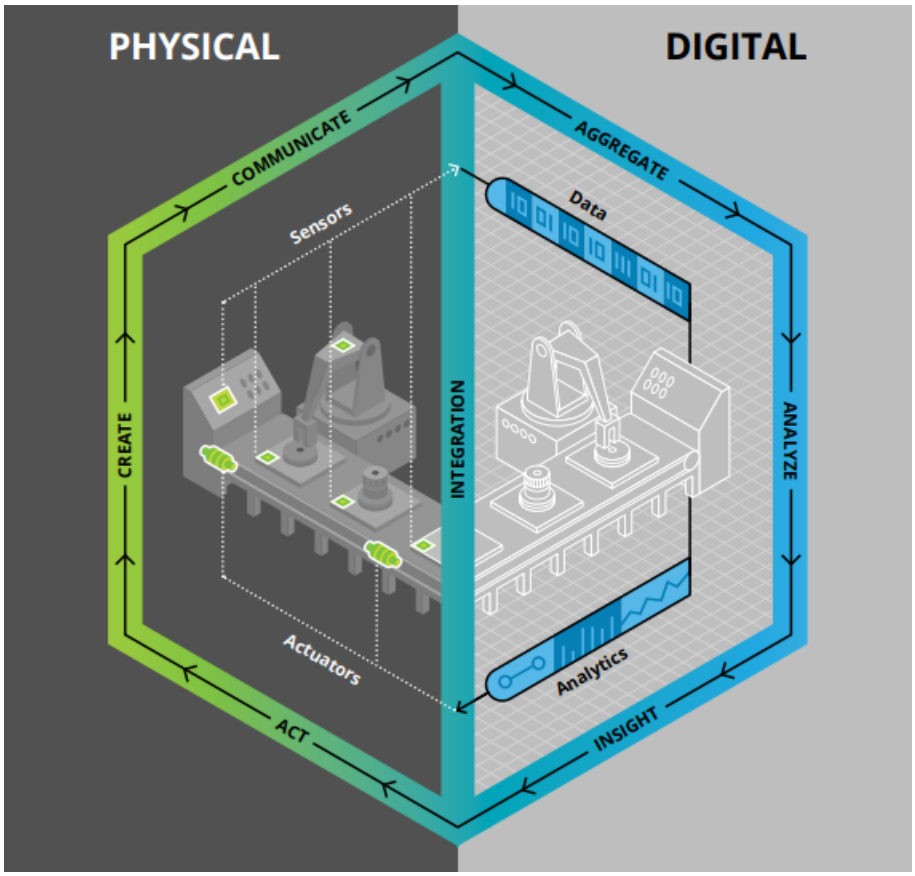
## 2.2 Digitální dvojče

Digitální dvojče je v souvislosti s průmyslem 4.0 velmi frekventovaný pojem, který může zároveň pojmenovávat dosti odlišné věci. Digitálním dvojčetem může být například záznam v databázi, reprezentující výrobek procházející výrobní linkou. Nebo to může být velmi komplikovaný matematický model, reprezentující například přenos tepla a teplotní chování struktury stroje. Důležité je, že digitální dvojče je digitální, existuje pouze jako počítačová reprezentace něčeho fyzického.[15]

Digitální dvojče může reprezentovat různě rozsáhlé objekty, od výrobku, přes jednotlivý stroj až po celou halu nebo dokonce výrobní závod. Využití digitálních dvojčat je široké, mělo by doprovázet výrobek po celou dobu jeho životního cyklu. Může sloužit například k vývoji nového výrobku, jako marketingově-demonstrativní nebo výukový nástroj, k plánování výroby, ke sledování výkonnosti výrobních systémů nebo pro prediktivní údržbu strojů a zařízení.[16]

Digitální dvojče je takový model, který reprezentuje reálnou entitu účelným způsobem. I pro virtuální zprovoznění může být použito digitální dvojče různě složité s ohledem na to, co všechno má být odzkoušeno. Může na něm být pouze testován software. Složitější dvojče obsahující přesný 3D model včetně hmotností a momentů setrvačností může sloužit k dimenzování pohonů a předběžnému hodnocení silových poměrů a dynamických vlastností. Spadá sem i simulace celého procesu montáže a výroby, tedy výrobních časů, možných kolizních stavů na lince, funkčnosti PLC v rámci linky i výroby jako celku, simulace pohybu

pracovních robotů a cobotů, ergonomie pracovišť, materiálových toků, skladovacích procesů apod. Souhrnně lze tedy říci, že digitální dvojče továrny, respektive její výrobní a logistické části, je komplexní celek, který v ideálním případě funguje stejně, jako jeho skutečná předloha.[17][18]




Obrázek 2.7: Schéma propojení digitálního dvojčete s fyzickým systémem[18]

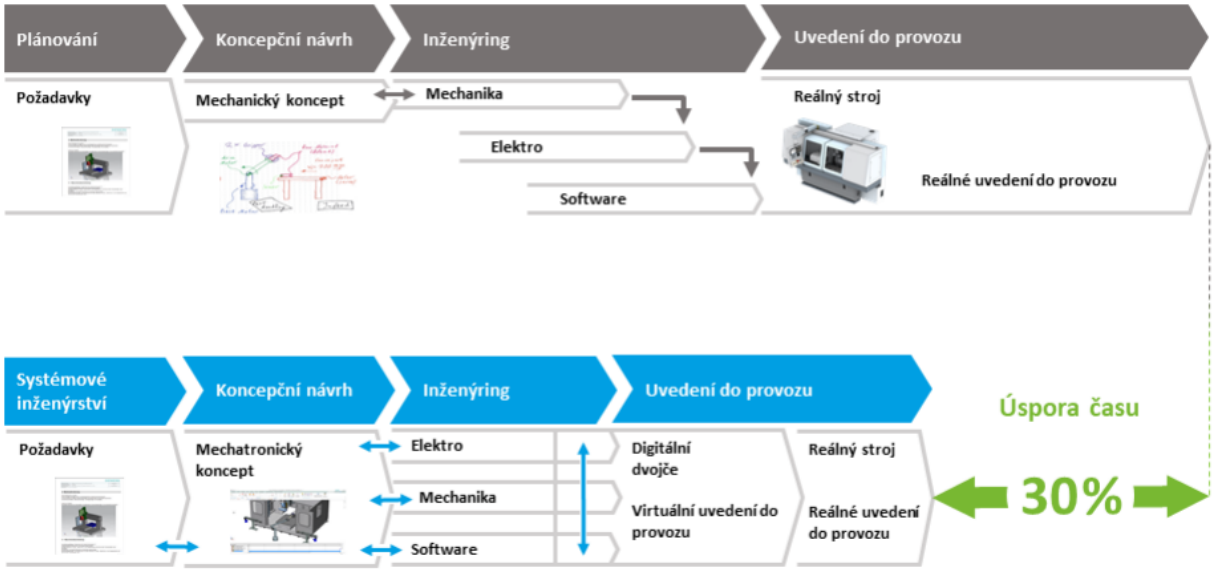
Obrázek 2.7 reprezentuje kybernetico-fyzikální smyčku, která představuje propojení mezi fyzickým systémem a jeho digitálním dvojčetem. Jedná se o další použití i po sestavení fyzického systému. [18]

## 2.3 Virtuální zprovoznění

Virtuální zprovoznění je činnost, při kterém ověřujeme funkčnost PLC programu využitím emulace virtuálního modelu (digitálního dvojčete). Snahou je nahradit část reálného zprovoznění. Snaha tedy je zprovoznit virtuální mechatronický model výrobní linky, který se chová a komunikuje se řízením obdobně jako ten reálný. Mechatronický model se dělí na část mechanickou, která řeší kinematiku a část elektrickou, která simuluje řízení. I přes to, že model bude totožný s reálným, nemusí se chovat stejně. Je nutné ověřit, že je model správně nastaven.[15]

Jak již bylo řečeno v úvodu této práce, proces vývoje nového stroje prochází řadou fází, které na sebe navazují. Virtuální zprovoznění umožňuje zkrátit proces vývoje. K tomuto zkrácení nedochází tím, že by virtuální zprovoznění snížilo pracnost procesu, ale díky němu je možné původně sériovou strukturu vývoje částečně paralelizovat, což je vidět na Obrázku 2.8[19]

Další velkou výhodou virtuálního zprovoznění je možnost odhalovat problémy ve všech fázích vývoje mnohem dříve. Opravy takto včas odhalených problémů bývají mnohem levnější. Virtuální dvojče vytvořené během vývoje může posloužit nejen výrobci, ale i zákazníkovi. Obzvlášť zajímavé využití je jako instruktážní nástroj pro zaučení obsluhy, nebo jako nástroj pro testování NC programů.[1][19]




Obrázek 2.8: Proces vývoje bez a s virtuálním zprovozněním [19]

## 2.4 Metody virtuálního zprovoznění

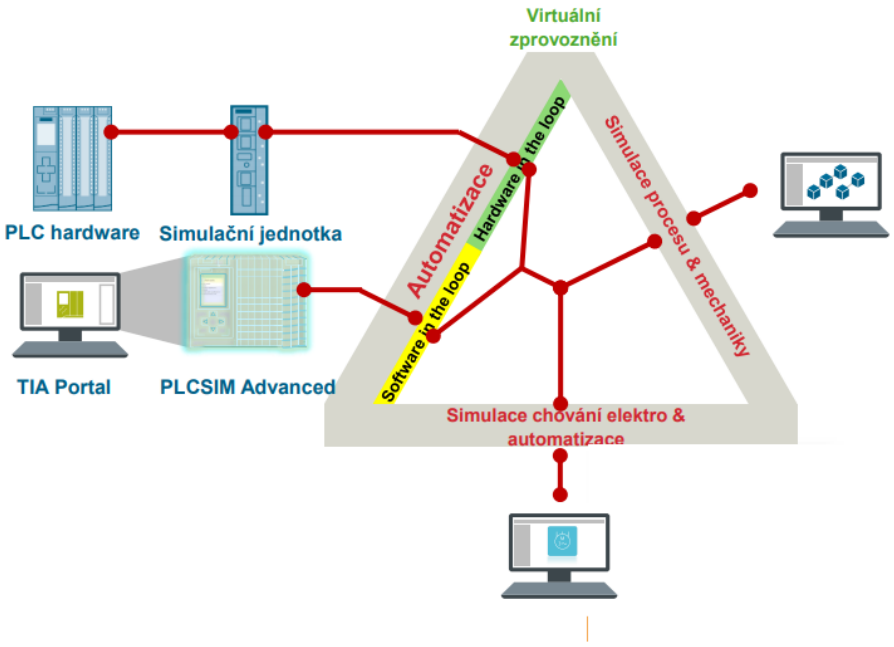
Nejpoužívanějšími způsoby jsou SiL (Software ve zpětné vazbě) a HiL(Hardware ve zpětné vazbě).Obě metody pracují s virtuálním modelem automatizovaného zařízení, na kterém je testován řídící program. Stejně tak je výsledkem obou metod simulace v reálném čase. Virtuální model simuluje všechny procesy a generuje tak signály vstupů a výstupů, které je jsou následně konvertovány do digitální a analogové podoby a přeneseny do řídícího programu.

### 2.4.1 Hardware ve zpětné vazbě - HiL

Při použití technologie HiL(Hardware ve zpětné vazbě) k provádění simulace používáme skutečný procesor, který máme spojený s počítačem. K tomu, aby simulace mohla fungovat je potřeba simulační jednotky jak je vidět na Obrázku 2.9, která komunikuje mezi PLC a počítačem. Tato metoda je vhodná k použití při reálném oživování, kdy nahráváme program na reálné PLC. Změny je dobré nejdříve ověřit na virtuálním modelu a tím se ujistit zda je nově nahraný program bez mechanických kolizí.[16][20]

### 2.4.2 Software ve zpětné vazbě - SiL

Pro provedení virtuálního zprovoznění pomocí SiL(Software ve zpětné vazbě) je potřeba jen software Obrázek 2.9), který funguje jako emulátor a nahradí reálné PLC i simulační jednotku. Tento způsob umožňuje tvorbu programu off-line za postupné validace na virtuálním modelu bez jakéhokoliv potřebného hardwaru.[16] [20]

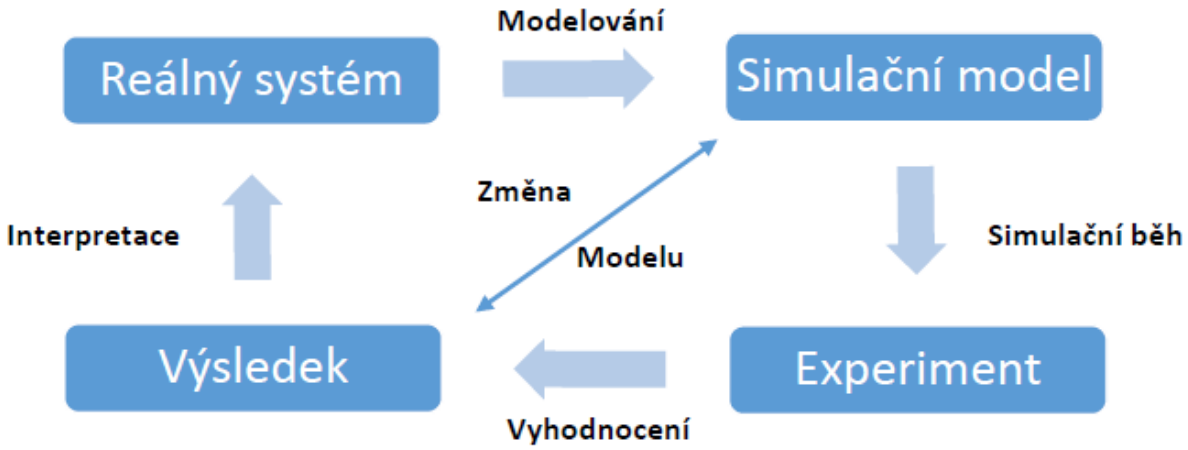



Obrázek 2.9: Propojení HiL a SiL[20]

## 2.5 Simulace

Pod pojmem simulace rozumíme metodu nahrazení zkoumaného systému simulačním modelem a procesy k tomuto nahrazení vedoucí. S tímto modelem jsou následně prováděny experimenty, které mají za cíl pochopit chování zkoumaného systému a v další fázi také ověření funkčnosti změn v tomto systému provedených.[21]

Zjednodušené schéma simulačních procesů ukazuje Zjednodušené schéma procesu simulace




Obrázek 2.10: Zjednodušené schéma procesu simulace [21]

Simulaci lze tedy chápat jako nástroj pro ověření plánovaných změn současných systémů a zavádění systémů nových s možností zkoumat dopady těchto změn ještě před jejich provedením na reálném systému. Širší uplatnění začaly simulační procesy nacházet v devadesátých letech minulého století s rozvojem výpočetní techniky, neboť vzrostla potřeba simulační modely zkoumat ve 2D a 3D provedení v reálném čase pro celý proces, tedy tvorbu modelu, experimentu, jeho sledování a vyhodnocování. Tyto požadavky znamenají práci s velkým množstvím dat a z toho plynoucí potřebu výkonných výpočetních strojů.[22]

# 3 Výběr softwaru

V následujícím seznamu jsou vypsány softwary, které byli vybrány zadáním k vytvoření digitálního dvojčete a provedení virtuálního zprovoznění.

- V-rep
- Robotino
- Ciros Studio
- Robo DK
- Factory I/O

Programy Robotino a Robo DK zde pouze krátce představím, jelikož bylo rozhodnuto, že v nich nebude nevytvářeno digitální dvojče ani nezprovozňováno již hotové.

## 3.1 Robotino

Robotino je simulátor robotických systémů od společnosti Festo Didactic. Robotino nabízí všechna čidla, aktuátory a softwarová rozhraní, která lze očekávat od moderního robotického systému. Snadno použitelné API a knihovna C/C ++ umožňuje transparentní přístup ke všem senzorům a prvkům Robotina.[23]

## 3.2 Robo DK

RoboDK je výkonný simulátor pro průmyslové roboty a jejich programování. Simulační software RoboDK umožňuje vytěžit maximum z vašeho robota. Programování neprobíhá v domácím jazyku robotů, ale aplikace jej převádí do vlastního jazyka.[24]

# 4 V-rep

Virtual Robot Experimentation Platform je open source robotický simulátor vyvíjený švýcarskou společností Coppelia Robotics. Jde o program, jehož integrované vývojové prostředí je založeno na architektuře distribuovaného řízení, kde každý model nebo objekt může být řízen samostatně, a to pomocí embedded scriptů, pluginů či vzdálených API klientů. Tato vlastnost činí z programu V-REP velice univerzální nástroj, který je vhodný pro simulaci většího množství robotů v rámci jedné scény, kde lze analyzovat a jednoduše upravit jejich vzájemnou interakci. Ovladače modelů, tedy controllery samotných robotů, se dají psát v programovacích či skriptovacích jazycích jako jsou C, C++, Python, Java, Lua, Matlab anebo Octave.

Mezi hlavní použití V-REPu patří například vývoj algoritmů, simulacím automatizace, prototypování a verifikaci anebo vzdálenému monitorování. Simulace jdou plně kontrolovat skrze šest programovacích přístupů, které jsou vzájemně kompatibilní. Obsahuje čtyři fyzikální enginy: Bullet Physics, ODE, Newton a Vortex Dynamics, pro simulaci fyziky, dynamické kalkulace a interakci mezi objekty. Dalšími důležitými vlastnostmi programu jsou například detekce kolizí, počítání vzdálenosti mezi objekty, dynamická simulace částic (kupříkladu napodobení proudu vzduchu či vody), senzorů přiblížení anebo snímačů.[25]

## 4.1 Uživatelské rozhraní

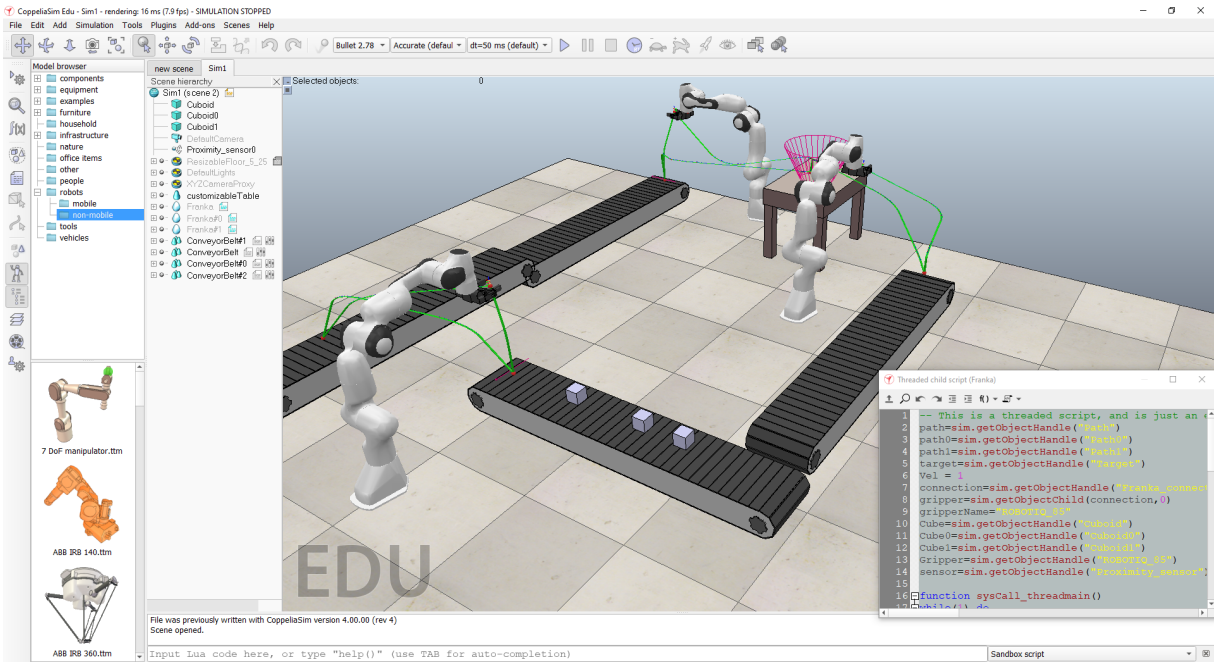
Při zapnutí programu, aplikace spustí výchozí scénu. Jedná se o prázdné pole bez objektů. Uživatel může mít otevřeno více scén současně, nicméně v rámci jednoho spuštění aplikace V-REP se dá pracovat pouze s jednou scénou, ačkoliv jich program dovoluje mít otevřených víc současně.

Aplikační okno se dělí na další elementy. Mezi nejdůležitější z nich, všechny viditelné na Obrázku 4.1 , patří:

- **Panel nástrojů** – seznam nejčastěji používaných funkcí při práci s programem. Většinu z nich lze vyvolat klávesovou zkratkou nebo volbou v menu. Při výchozím nastavení programu se jedná o dva panely, svislý a vodorovný.
- **Prohlížeč modelů** – pomocí kterého můžeme procházet a přenášet do scény vytvořené modely. Ve vrchní polovině se nachází struktura podle složek, ve spodní části

se pak nachází vizualizace modelů vybrané složky. Přenesení náhledu do scény V-REPu se provádí jednoduchým přetáhnutím myši odpovídajícího modelu z knihovny do prostoru.

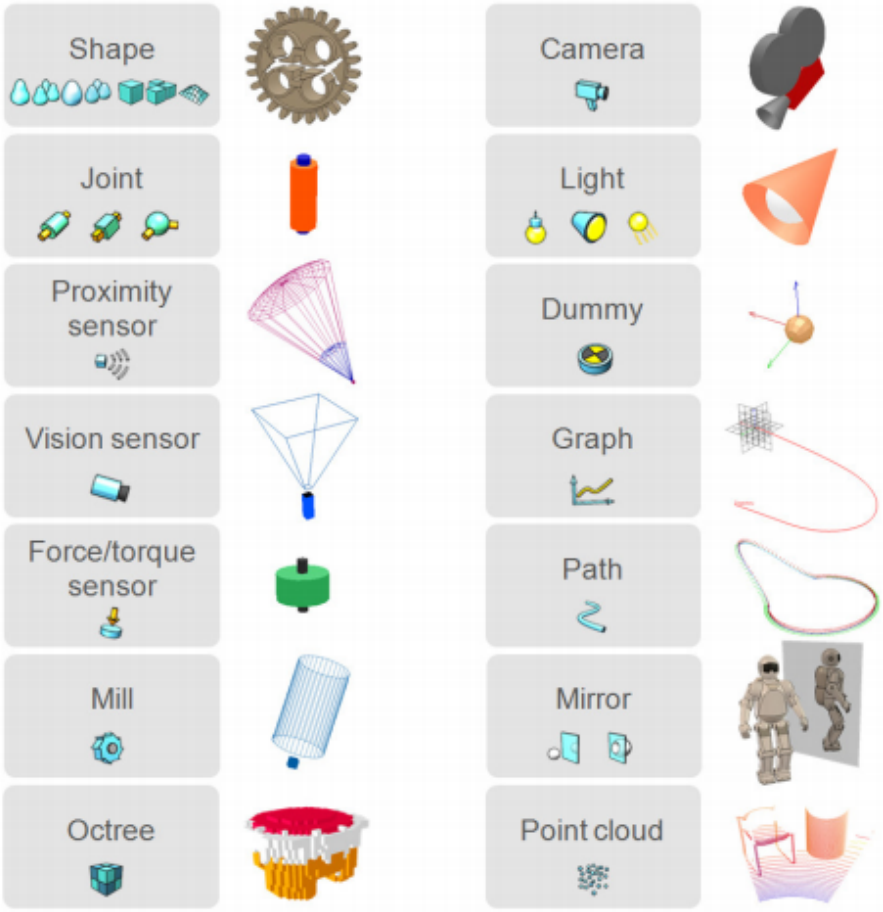
- **Hierarchie scény** – jde o jeden z nejpodstatnějších prvků aplikačního okna k tvorbě a správě modelů scény. Jedná se o základní panel, zobrazující veškerý obsah scény, tedy všechny objekty, vazby a další, které jí tvoří. Ukládání objektů scény do hierarchické struktury vytváří na panelu tzv. strom této struktury. Jednotlivé prvky tohoto stromu můžou být rozvinuty nebo naopak zabaleny. Dvojitým kliknutím na ikonu objektu se vyvolá jeho dialog, obdobná akce na jeho jméno umožní přejmenování. Objekty můžou být přesouvány do jiných objektů pomocí hierarchie scény, čímž se ruší vztah předek-potomek a může se vytvořit vztah nový .




Obrázek 4.1: Uživatelské rozhraní robotického simulátoru V-REP[25]

## 4.2 Objekty

Hlavními prvky v aplikaci, které vytváří scénu jsou objekty scény. Lze je vidět jednak v pohledu scény, ale také v okně její hierarchie, kde jsou znázorněny symbolem specifickým pro daný typ objektu. Jednotlivé typy objektů, včetně jejích náhledu v hierarchii a grafického příkladu jsou znázorněny na Obrázku 4.2




Obrázek 4.2: : Všechny typy objektů, které je možné využít ve V-REP [25]

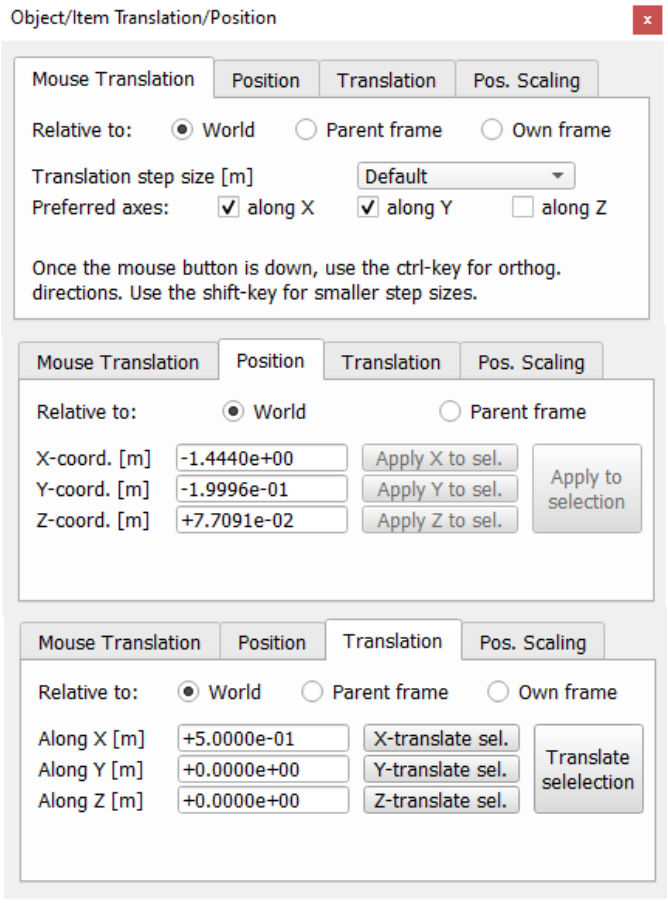
Mezi hlavní objekty patří:

- **Tvary** - Jednoduché tvary je možné vytvořit pomocí zabudovaného nástroje, nebo ty složitější je možné importovat např. ve formátu .dxf nebo .stl, atd. Pro efektivní spolupráci tvaru s dynamickým enginem je nutné tento tvar převést pomocí vestavěného nástroje do konvexního tvaru.
- **Vazby** - K dispozici jsou vazby rotační (revolute), posuvná (prismatic) a sférická (spherical).
- **Kamery** - Slouží pro zobrazení scény, objektů a detailů. Je možně jejich libovolné umístění.
- **Osvětlení** - Je možnosti upravit osvětlení scény.
- **Grafy** - V-REP nabízí široké možnosti zobrazení jednotlivých parametrů pomocí grafů. Od absolutních a relativních vzdálenosti a rychlostí objektů, přes parametry scény a simulace, úhlové rychlosti, atd

- **Dummy** - Dummy má mnoho využití, nejčastěji k propojování objektů s vazbami ve stromové struktuře scény nebo jako vhodně programovatelný objekt.
- **Proximity senzory** - Používají se k měření vzdálenosti mezi objekty v reálném čase.
- **Vizuální senzory** - Umožňují získat obrazovou matici ze scény pro další zpracování.
- **Silové senzory** - Slouží ke spojování statických objektů a také k měření silové interakce mezi objekty.

### 4.2.1 Manipulace s objekty

Se všemi objekty scény lze manipulovat použitím nabídky Object Shift, a poté tažením kurzoru myši, což není příliš přesně, nebo nejlépe změnou parametrů v souřadnicovém systému. Toho lze docílit vzhledem k absolutnímu souřadnicovému systému - Coordinate system, relativnímu souřadnicovému systému nadřazeného objektu nebo relativnímu souřadnicovému systému sebe samého.




Obrázek 4.3: : Rozhraní pro manipulaci s objekty [25]

Je proto vhodné při scéně s větším množství objektů využívat stromové struktury, a určovat mateřské objekty v hierarchii scény, vůči kterým následně umísťovat další objekty. Výhoda tohoto přístupu je, že při manipulaci s mateřským objektem budou zachovány vazby všech podřízených objektů a jejich pozice vůči mateřskému objektu.

Podobným způsobem lze s objekty také rotovat nabídkou Object Rotate. Realizovat to lze zadáním žádaného úhlu rotace ve stupních kolem určité osy. Taktéž je možno rotovat pomocí tažení kurzoru myši. Jiné způsoby manipulace s objekty nejsou možné, důsledkem toho je, že modelování ve V-REP je relativně zdlouhavý proces.

### 4.2.2 Vlastnosti objektů

U některých typů objektů je možnost nastavit speciální vlastnosti, které upravují interakci s jinými objekty nebo jejich chování vůči kalkulačním modulům.

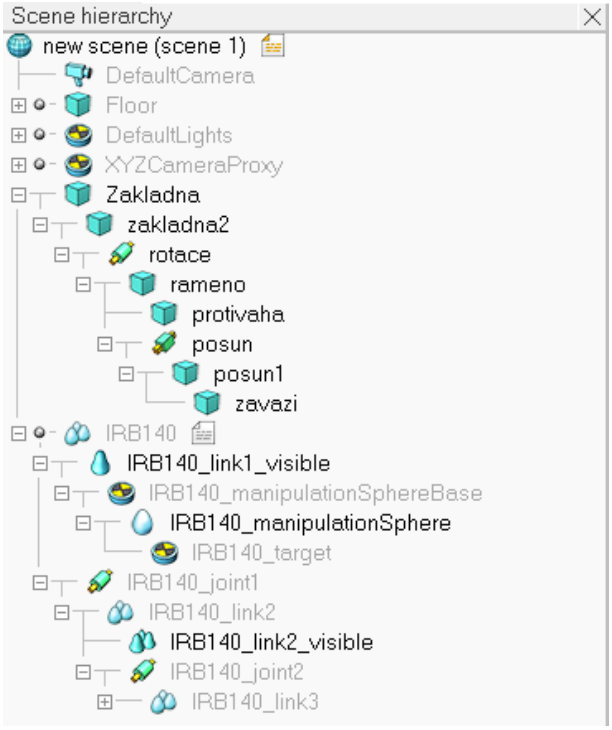
U všech typů objektů je také možnost nastavit vlastnosti v průběhu simulace [25] :

- **Collidable** - nastavení zda objekt může přijít do kolize s jiným objektem či nikoli.
- **Measurable** - toto nastavení určuje, že mezi více měřitelnými objekty může být definována minimální vzdálenost.
- **Detectable** - zde je možnost určit, zda může být daný objekt detekovatelný proximity senzorem, případně jakým typem.
- **Rendable** - zde je možnost určit, zda může být daný objekt detekovatelný vision senzorem.
- **Viewable** - toto nastavení určuje, zda objekt může být zobrazen v dalších pohledech.

### 4.2.3 Tvorba vlastního modelu

Dynamický model je součástí fyzikálního modelu, a proto musíme dodržet jakousi jednoduchost. V simulátoru se modely sestavují hlavně z těles jednodušších tvarů, které jsou vzájemně spojeny pomocí kloubů. Zbytek jsou senzory a objekty, které nejsou pro dynamický model důležité. Při dodržení správného spojení vzniká strom prvků.

Při tvorbě modelu je třeba dát si pozor, aby dynamické části byly připojeny vždy jen na jiné dynamické části. Jelikož mezi dvěma objekty může být definována vždy jen jedna vazba, je občas nutné vytvořit i virtuální díly, které mají za úkol pouze přenést moment. Takovéto skládání složitějšího modelu je velice časově náročné a dosti neefektivní.




Obrázek 4.4: : Hierarchie vlastního modelu a modelu z knihovny [25]

Druhou možností je import CAD modelu například ze Solidworks, ale opět jde hlavně o jednoduché díly či sestavy, jelikož V-REP neumí pracovat se složitými tvary, je vhodné každý díl rozdělit na dvě části, výpočtovou a vizuální.

Ve výpočtové části bude model zjednodušený tak, aby parametry odpovídal původnímu dílu, jak jen to bude možné, ale zároveň, aby co nejméně zatěžoval výpočet. Je tedy nutné experimentálně vyvážit poměr mezi zjednodušením a přesností. V případě složitějších tvarů je možné využít nástroje Vizual Decompozition, který díl zjednoduší do žádaného konvexního tvaru tak, že vytvoří trojúhelníkovou mesh. Nástroj nabízí i změnu parametrů pro dekompozici, z čehož je možné určit potřebnou míru zjednodušení.

Ve vizuální části je naopak snahou, přiblížit se původnímu vzhledu součásti co nejvíce, protože slouží pouze jako vnější skořepina výpočtové části, bez jakékoliv jiné interakce.

## 4.3 Dynamický modul

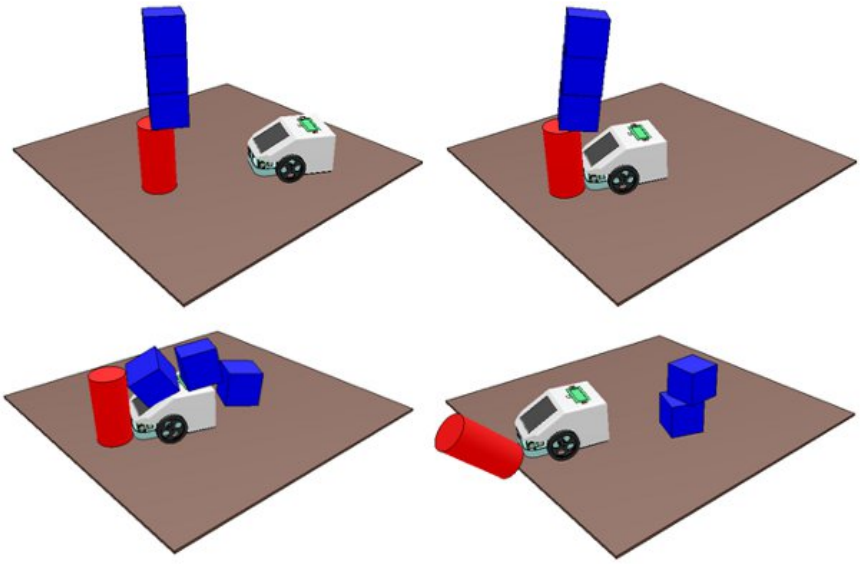
Dynamický modul aplikace V-REP dává na výběr ze čtyř různých fyzikálních enginů. Uživatel si může bohatě vybrat, který z nich nejlépe vyhovuje jeho potřebám i simulaci. Důvodem čtyř fyzikálních enginů je, že simulace fyziky je komplexní úkol, který je možné efektivně řešit různými stupni přesnosti, rychlosti nebo podporou různých funkcí.

1. **Bullet physics library** – open source fyzikální engine, jehož vlastnosti jsou 3D kolize anebo dynamika tuhých těles. Používá se jako herní fyzikální engine nebo ve vizuálních efektech u filmů.[26]

2. **Open Dynamics Engine** – se rovněž řadí mezi open source fyzikální enginy. Jeho hlavními prvky jsou dynamika tuhých těles a kolize. Používá se v mnoha aplikacích a hrách, a stejně jako jeho ekvivalent uvedený výše často zastává roli herního fyzikálního enginu.[27]

3. **Vortex Dynamics** – proprietární (closed source), komerční fyzikální engine poskytující vysoko-věrnostní fyzikální simulaci. Vortex disponuje reálnými parametry, které zrcadlí odpovídající fyzikální jednotky, a jsou použitelné pro veliký počet vlastností či funkcionalit, což z něj činí vysoce realistický engine s vysokou přesností simulované dynamiky.

4. **Newton Dynamics** – multiplatformní life-like fyzikální simulační knihovna, která může být integrována do herních enginů i jiných aplikací a poskytuje ten nejlepší výkon a stabilitu simulace. To jej též činí nástrojem pro jakékoliv simulování fyziky v reálném čase.[28]



Obrázek 4.5: : Ukázka dynamické simulace [25]

Dynamický modul umožňuje simulovat interakce mezi objekty na úrovní blížící se interakcím objektů reálného světa. Dovoluje objektům padat, kolidovat a odrážet se, ale skrze něj můžeme s objekty manipulovat robotickými rameny, simulovat pohyb objektů na pásovém dopravníku či realisticky napodobovat průjezd modelů, například aut, tratí.

## 4.4 Programování

V-REP patří mezi vysoce přizpůsobitelné simulátory a téměř každý krok simulace může být definován uživatelem. To zajišťuje velmi propracované API, které podporuje šest různých programovacích nebo kódovacích přístupů, z nichž každý má své přednosti a hodí se na něco jiného, ale všechny přístupy jsou vzájemně kompatibilní, tedy dají se navzájem kombinovat.[25]

1. **Vložený skript** – Tyto skripty jsou psány pomocí skriptovacího jazyka Lua. Tato metoda umožňuje přizpůsobení konkrétní simulace, simulační scény a do určité míry přizpůsobení samotného simulátoru. Je to nejjednodušší a nejpoužívanější programovací přístup.

2. **Add-on** – tato metoda využívá Lua skripty a umožňuje rychlé přizpůsobení simulátoru. Doplněk se spouští automaticky a běží na pozadí. Doplňky nejsou specifické pro určité simulace nebo modely, nabízejí spíše obecnější funkce simulátoru.

3. **Plugin** – tj. přizpůsobení simulátoru nebo simulace pomocí psaného pluginu pro V-REP. Pluginy se vyvolávají pomocí Lua skriptů, proto jsou pluginy používány ve spojení s vloženými skripty.

4. **Vzdálené API** –tento způsob umožňuje přizpůsobení simulátoru nebo simulace pomocí externí aplikace, která je umístěna na robotu nebo jiném stroji. Tato externí aplikace se připojuje k simulátoru V-REP pomocí příkazů vzdáleného rozhraní API. Díky tomu je možné, aby stejný kód obsažený v robotu, ovládal i jeho virtuální simulaci ve V-REPU.

5. **ROS** – tato metoda přizpůsobuje simulaci pomocí externí aplikace (umístěné na robotu nebo na jiném zařízení), která se připojuje k simulátoru V-REP pomocí ROS (Robot Operating System).

6. **Vlastní klient/server** – simulace je přizpůsobena pomocí klient/server aplikace, která využívá buď skripty nebo pluginy pomocí vlastních komunikačních prostředků (sockets, pipes, atd.).[25]

### 4.4.1 Skripty

Pro psaní skriptů se používá jazyk Lua, jehož pomocí je možné volat funkce pro ovládání simulace a objektů v simulaci. Tento typ skriptu je možné vytvořit ke každému objektu scény, což nabízí možnost vytvoření několika různých skriptů pro jednu scénu.

**Main script** Jedná se o simulační script, který není přidružený a je aktivní pouze za běhu simulace a není spjatý s žádným objektem scény. Při výchozím nastavení obsahuje každá scéna právě jeden tento script, jehož obsahem je základní kód, který zajišťuje běh simulace. Bez tohoto scriptu by simulace nefungovala.

**Child script** Stejně jako main script, i child script patří mezi simulační scripty, nicméně v tomto případě jde o přidružený skript. Aplikace dovoluje mít ve scéně neomezeně takových scriptů, a každý z nich představuje nějakou funkci, zapsanou pomocí programovacího jazyka Lua, která umožňuje zacházení se simulací, jeho programovací prostředí lze vidět na Obrázku 4.1.

Tyto scripty jsou sdružené s objekty ve scéně, a v hierarchii scény jsou označené ikonou bílého scriptu, jehož pomocí lze otevřít skriptový editor s kódem objektu.

## 4.5 Simulace

Simulace je proces, při kterém jsou na všechny objekty ve scéně aplikovány fyzikální zákony a jejich naprogramování. Kombinací těchto dvou procesů způsobí simulované chování objektů na scéně. Uživateli jsou k ovládání simulace na horním panelu přístupné tři základní stavy, a to její běh, pozastavení a zastavení. Další nastavení je rychlost simulace a po jakých časových krocích má být fyzikální model aplikován. To se může projevit na náročnosti výkonu. V případě, že počítač nebude stíhat zpracovávat výpočty v reálném čase, bude docházet k viditelnému zpomalování simulace.

Obrázek 4.6: : Ovládací panel simulace [25]

Interně však simulátor používá další přechodné stavy, při kterých informuje scripty a programy o tom, co bude následovat. Aby se docílilo správného chování scripty či programy by měly vždy reagovat podle aktuální volané funkce, případně podle simulačního stavu.[25]

## 4.6 Vytvořená scéna

Byla vytvořena robotická simulace obsahující tři roboty od výrobce Franka Emika, lze ji vidět na Obrázku 4.1. Předmětem simulace je cyklický pohyb předmětů dokola scény. Pohyb zajišťují jednak dopravníkové pásy a robotické manipulátory, obojí řízené skripty psané v jazyku Lua.

Dopravníkové pásy jsou ovládané pomocí jednoduchého skriptu 4.1. Kde Proximity senzor umístěný na vhodném místě, aby ho mohl uchopit manipulátor, při kontaktu s předmětem změní svou hodnotu a tím zastaví pás do doby než je předmět odstraněn.

```
sensor=sim.getObjectHandle("Proximity_sensor")

local beltVelocity=sim.getScriptSimulationParameter(...)

    if (sim.readProximitySensor(sensor) > 0) then
    beltVelocity=0
    end
```

Výpis kódu 4.1: Dopravníkový skript

Robotické manipulátory mají skript delší. První část tvoří čekací cyklus, kdy manipulátor čeká na dopravení objektu do manipulační zóny jak je vidět ve skriptu 4.2. Cyklus je přerušen stejným signálem od Proximity senzoru, jako je zastaven dopravníkový pás.

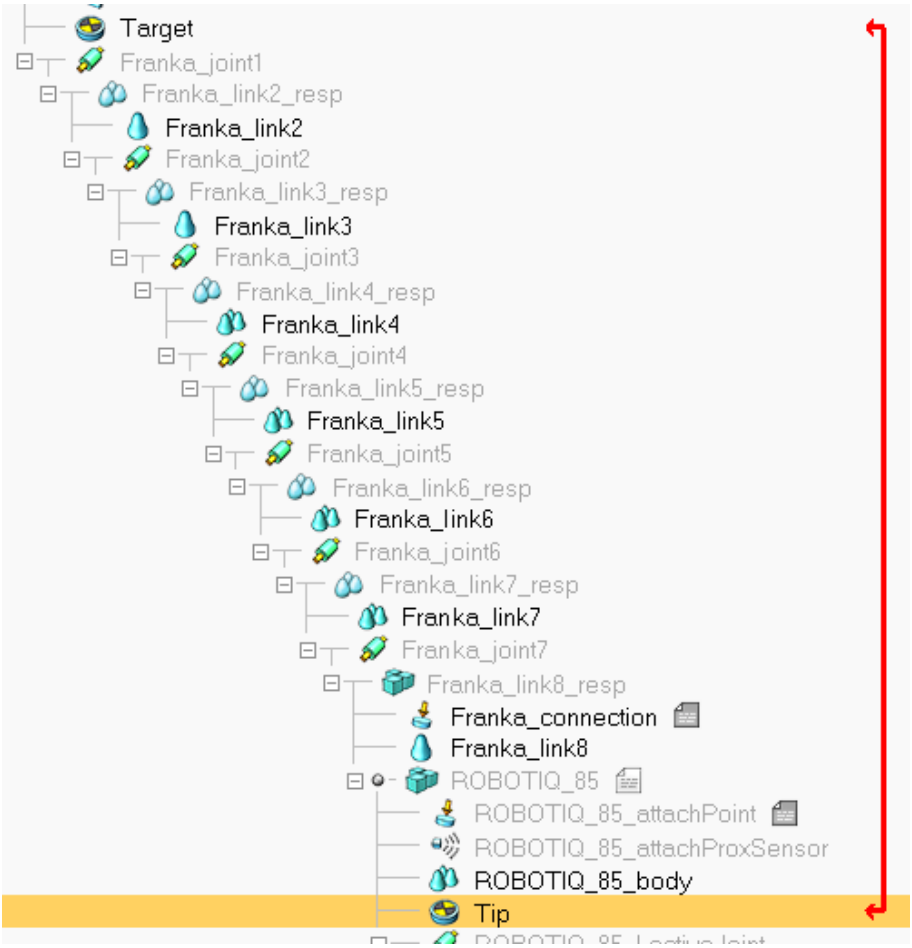
```
while(1) do
    sim.wait(1)

    if(sim.readProximitySensor(sensor) > 0) then
        break
    end

end
```

Výpis kódu 4.2: Zahajující část skriptu

Pro pohyb robota jsem zvolil metodu inverzní kinematiky, kdy se tato vazba vytvoří mezi dvěma Dummy jak je vidět v Obrázku 4.7. Jedna je nazvána Tip, reprezentující špičku manipulátoru, a je v hierarchii umístěna pod Gripper, druhý nazvaný Target je spjat se základnou robota. Po nastavení všech kloubních vazeb v manipulátoru na inverzní kinematiku dostáváme robota, kterého můžeme ovládat v rámci jeho dosahu. Samotný pohyb poté probíhá po předem vyznačených cestách (Path), funkcí *sim.followPath(target,path,3,0,Vel,0.05)* dosáhneme pohybu Dummy Target po vyznačené trase.




Obrázek 4.7: : Propojení Tip a Target ve stromu robotu Franka[25]

Výpis kódu 4.3 zajišťuje uchycení předmětu robotem. Obdobné funkce jsou následně použity pro jeho vypuštění.

```
sim.setObjectParent(Cube,Gripper,true)
sim.setIntegerSignal(gripperName..'_close',1)
sim.setObjectInt32Parameter(Cube,3003,1)
sim.resetDynamicObject(Cube)
```

Výpis kódu 4.3: Uchopení předmětu

# 5 Ciros Studio

CIROS Studio je univerzální software pro 3D simulaci s využitím sahajícím do mnoha směrů. To sahá od využití 3D simulace při tréninku a vzdělávání až k realizaci digitálních továren. Spojuje tedy oblasti plánování, návrhů, elektrických rozvodů, uvedení do provozu, prodeje a marketingu. CIROS Studio dovoluje uživatelům vytvářet rozvržení a procesy, simulace robotických pracovních jednotek, automatizovaných výrobních systémů, testování a vývoj robotických i PLC programů.

Nejčastěji se používá k tvorbě 3D simulací v reálném čase, 3D modelování a programování robota s ověřením kinetické dynamiky mechatronických systémů a případných programových chyb ve virtuálním světě před aplikací do systému a před spuštěním skutečného procesu. Program ověřený v simulaci může být poté přenesen do řídícího systému robotické stanice.

Efektivní přísun modelů je zajištěn kombinací knihovny, obsahující roboty, nástroje, dopravní pásy, podavače a další, a přidávání vlastních modelů. Tím je myšleno volné 3D modelování a import z CAD systémů prostřednictvím standardního formátu STEP.

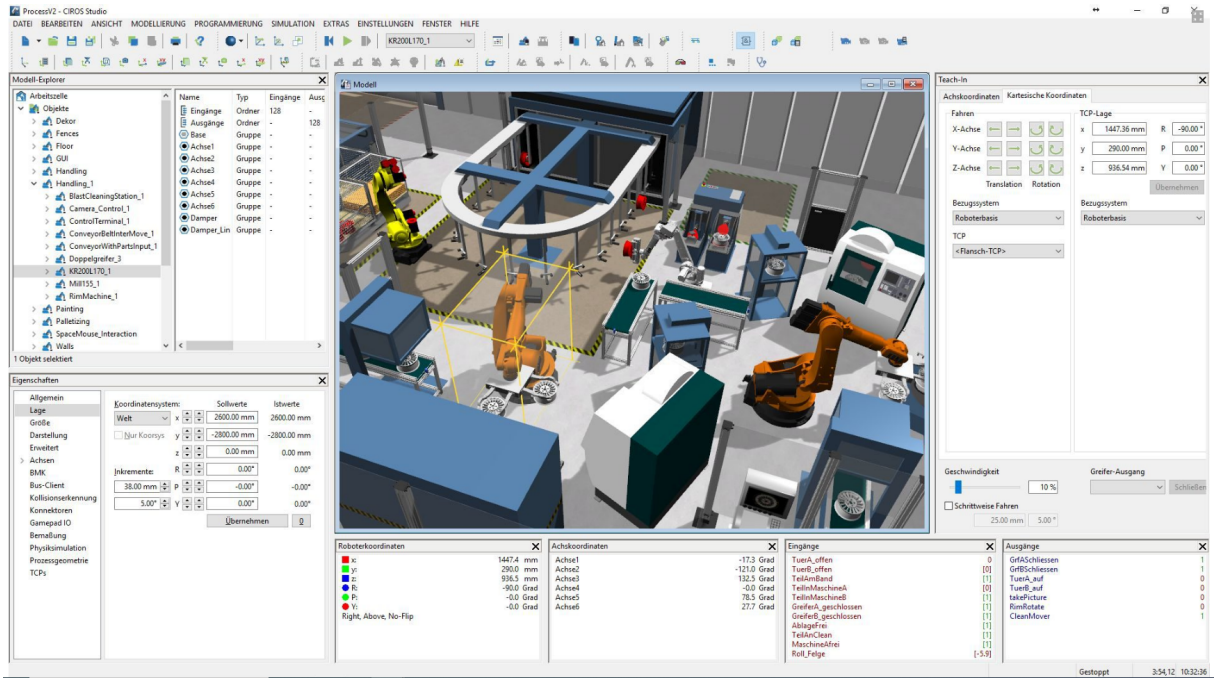
## 5.1 Uživatelské rozhraní

Oproti předešlému V-repu, kterému je CIROS Studio velice podobné, je uživatelské rozhraní o něco složitější. Nejznatelnějším rozdílem je velký počet oken, v kterých uživatel rychle ztrácí přehled. Pro představu předešlí software disponoval v základu třemi aplikačními okny jak je vidět v kapitole 4.1, za to CIROS Studio pracuje téměř neustále s devíti okny najednou.

Nejdůležitější typy pracovních oken jsou :

- **Work cell Window** - zobrazuje grafickou reprezentaci aktuálně vybraného modelu či scény. Není problém otevřít více pohledů, což dovoluje pozorovat jednu scénu z více pohledů současně.
- **Joint Coordinates** - toto okno ukazuje relativní souřadnice jediné vazby robota s odpovídajícími jednotkami, stupně pro rotační vazby a milimetry pro posuvné vazby.
- **World Coordinates** - podává informace o absolutních souřadnicích středového bodu objektu. Kromě souřadnic také informuje o orientaci objektu.

- **Teach-In** - kromě přehledu jednotlivých vazeb robota toto okna dává i možnost skrze vazby robota ovládat.
- **Inputs/Outputs** - okno zobrazující hodnoty inputů a outputů. Hodnota nula je vyobrazena červeně a hodnota 1 zeleně.

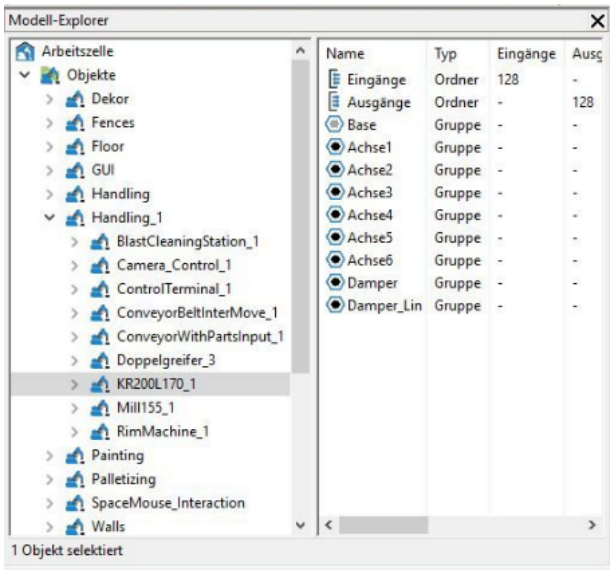


Obrázek 5.1: : Ovládací panel simulace [29]

## 5.2 Modelování

Existují čtyři hlavní nástroje pro snadné modelování či vytváření modelů. Vedle Knihovny modelů a Průzkumníka modelů tu má uživatel ještě možnost poskládat vlastní model z jednotlivých komponent, které lze najít v databázi. Poslední možností je import CAD modelu. Těmito čtyřmi možnostmi se bude tato kapitola zabývat.

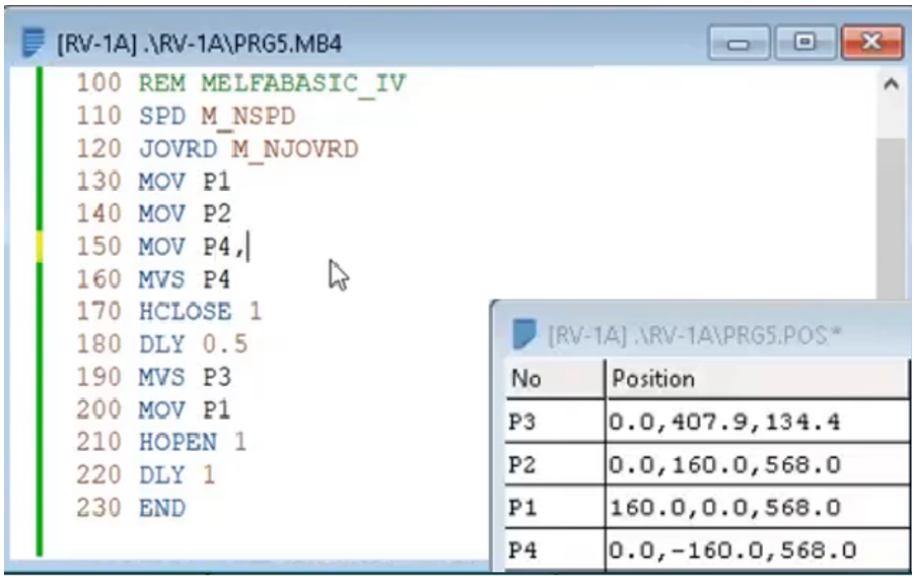
Knihovna CIROS Studia obsahuje širokou škálu modelů různého druhu. Velké množství položek jsou hotové roboty od firem ABB, KUKA, Fanuc a Mitsubishi. Nebo pouze samotné PLC, například od firmy Siemens. Knihovna ale také obsahuje již hotové CNC jednotky, dopravníkové pásy nebo automatické sklady. Dalšími položkami jsou i základní materiály pro vlastní modelování, od jednoduchých tvarů až po senzory.




Obrázek 5.2: : Model explorer [29]

## 5.3 Programování

Programování standartních robotů od velkých firem probíhá v jejich původních programovacích jazycích. Roboti ABB jsou tedy programovány v jazyce RAPID, KUKA roboti v KRL jazyce a roboti Kawasaki ve Vplus. Dalšími podporovanými programovacími jazyky jsou Melfa Basic IV a IRL (Industrial Robot Language).




Obrázek 5.3: : Příklad programování pozic robota v Melfa basic IV [29]

CIROS také nabízí ideální pracovní prostředí pro programování PLC založené na řídících systémech Siemens S7. Součástí je také virtuální učební prostředí pro mechatronický školicí systém MPS(Modular Production System).

## 5.4 Simulace

Moderní technologie PC nám umožňuje vytvářet velice realistické 3D simulace i pro ty nejsložitější automatické systémy bez rizika pro člověka nebo stroj. Simulace běží v reálném čase s možností úpravy rychlosti a se simulačním cyklem až 1 milisekundu. Počet současně simulovaných modelů, např. robotů, I/O kontrolérů nebo PLC je omezeno pouze výkonem použitého počítače. Záznam simulačních dat, například poloh, rychlostí, zrychlení, hodnot I/O a různých údajů o výkonu, se dá exportovat pro další analýzu grafických nebo tabulkových dat.

Aplikace je taky schopná simulovat reálné senzory mnoha typů: indukční a kapacitní přibližovací spínače, optické senzory vzdálenosti, světelné závory, ultrazvukové senzory, kamery pro detekci polohy a orientace, nebo třeba také čtečky čárových kódů.

# 6 Factory I/O

Factory I/O je výukový software zaměřující se na 3D simulace virtuálních továren. Je navržen tak, aby se snadno používal, umožňuje rychle sestavit virtuální továrnu pomocí výběru standardně používaných průmyslových dílů. Factory I/O také obsahuje více než dvacet hotových scén, které jsou připravené k použití, inspirovaných typickým použitím v průmyslu.[30]

Nejběžnějším scénářem je použití Factory I/O jako školicí platformy pro programování PLC, jelikož jsou základním kamenem veškeré automatizace.

## 6.1 Uživatelské rozhraní

Oproti předchozím dvěma softwarům je uživatelské rozhraní Factory I/O velice intuitivní. Je velice jednoduché a uživatelsky přívětivé, bohužel za cenu některých funkcí. Celé aplikaci dominuje jeho hlavní scéna s 3D modelem, která je tady hlavním aplikačním oknem a ostatní okna ho pouze překrývají. Jako každý program podobného typu má Factory I/O v horní části Toolbar, na kterém najdeme podobné funkce jako jsou File, Edit a View, dále se tu nachází ovládání samotné simulace. Velice šikovný způsob pohybu kamery je udělaný pomocní Navigace kamery v levém dolním rohu.

Obrázek 6.1: : Uživatelské rozhraní [30]

## 6.2 Objekty

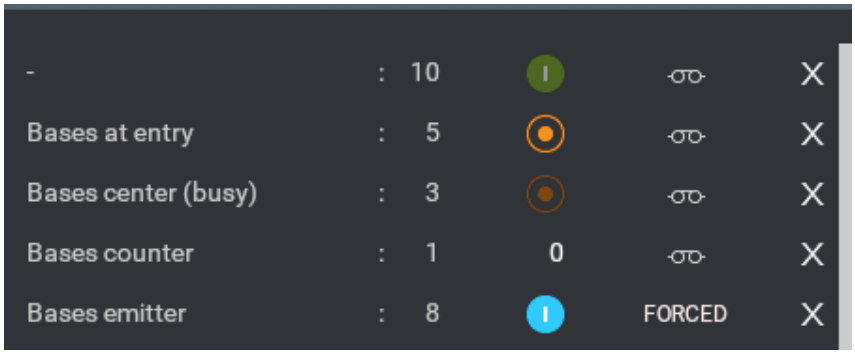
Dalším důležitým prvkem je Palette, což je prakticky knihovna všech dostupných součástek ve Factory I/O. Obsahuje osm kategorií z toho nejdůležitější jsou:

- **Items** - základní předměty s kterými se provádí manipulace. Jde o krabice, bedny, palety a destičky.
- **Load Parts** - kategorie rozdělená na těžké a lehké. Obsahuje různé druhy dopravníkových pásů a posuvných zařízení.
- **Sensors** - obsahuje různé druhy senzorů, od indukčních senzorů, difúzních, až po senzory pohybu.
- **Stations** - jde o kategorii obsahující různé stanice, jedná se o několik strojů jako je například výtah nebo výrobní centrum, bohužel právě ty jsou jedinou robotickou složkou v celém simulátoru.

Další důležitou položkou v knihovně, která nespadá pod žádnou kategorii je Emitter a Remover. Jedná se body kde se objevují a mizí Itemy, reprezentují tedy přísun polotovaru a odběr hotového produktu.

## 6.3 Řízení

Factory I/O funguje ve dvou různých režimech, v režimu úprav a v režimu spuštění. V režimu úprav vytváříte scénu spojením dílů dohromady a v režimu Run ji simulujete v reálném čase. V režimu Run je scéna simulována v reálném čase a lze ji ovládat ručně nebo pomocí externího ovladače (například PLC). Před ovládáním scény pomocí externího ovladače je doporučeno otestovat ji ručně. Tímto způsobem je zajištěno, aby rozložení scény fungovalo dle očekávání.




Obrázek 6.2: : Seznam tagů aktivních objektů [30]

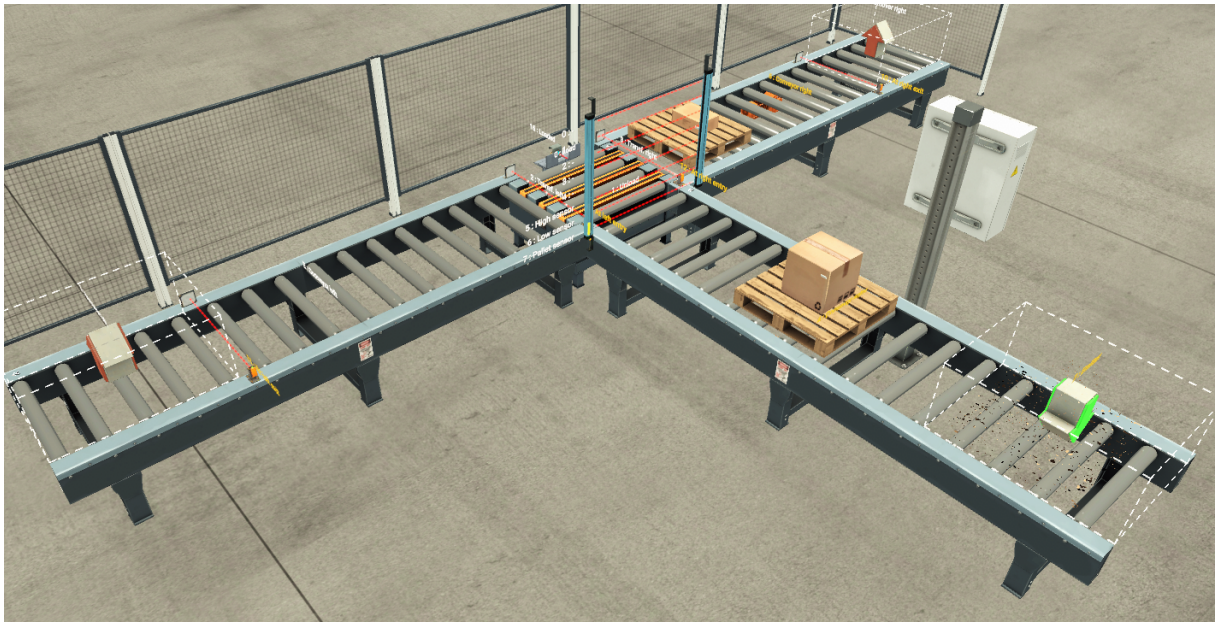
Jakákoli část, která je snímačem nebo akčním členem, má alespoň jeden tag. Tagy mohou být dvou různých typů: tagy snímačů a tagy akčních členů. Mohou obsahovat tři různé datové typy: Boolean pro hodnoty zapnutí/vypnutí, Float pro analogové hodnoty (reálná čísla) a Integer pro konkrétní data. Hodnoty tagů lze kdykoli manuálně vynutit.

I/O Driver je integrovaná funkce zajišťující komunikaci s externím řídícím systémem. Factory obsahuje mnoho I/O kontrolérů, z nichž každý je určen k použití se specifickou technologií. Jedním z nich je také i Control I/O, výkonné Soft PLC určené výhradně pro použití s Factory I/O. Hlavním cílem Control I/O je poskytnout nástroj nezávislý na značce, který se snadno naučí každý kdo začíná ve světě automatizace. S Control I/O vyvíjíte programy s diagramy funkčních bloků pomocí nejběžnějších funkcí dostupných na skutečném PLC.[30]

### 6.3.1 Třídění krabic

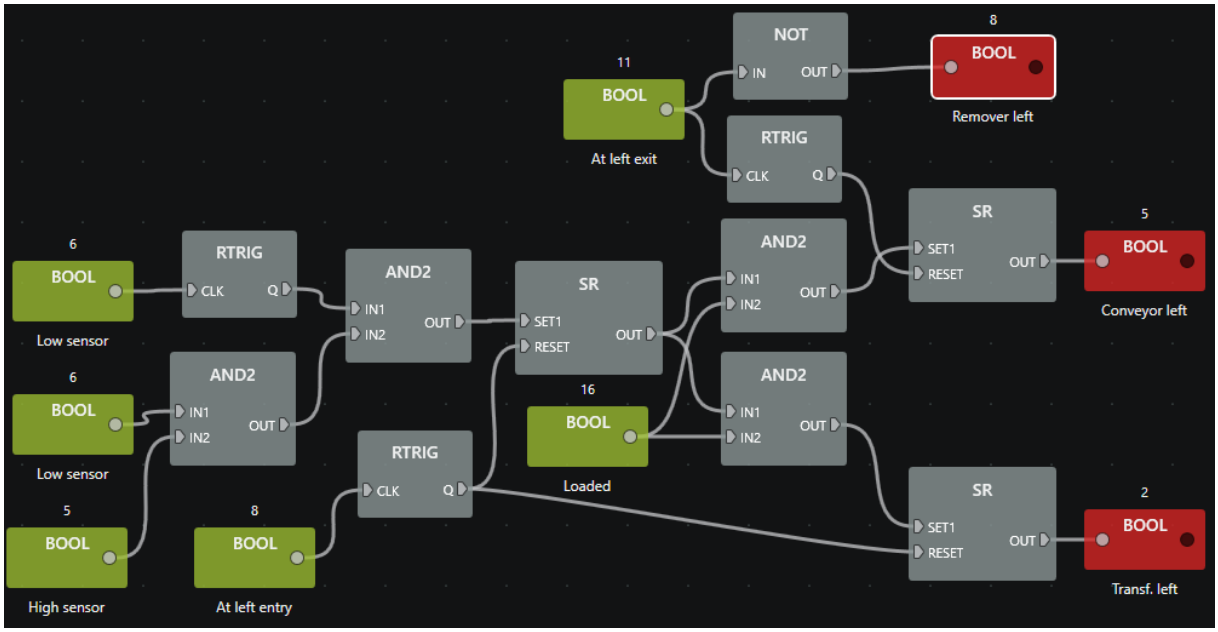
První úlohu kterou jsem simuloval je třídění krabic podle výšky. Jde o jednu z předem vytvořených scén, na které se dobře testuje automatické řízení. Scéna se skládá z jednoho Emitteru, od něho vede dopravní pás k rozdělovači před kterým jsou senzory zjišťující výšku. Na každé straně od rozdělovače je Remover.

Po zapnutí simulace, zmáčkněte tlačítko start na kontrolním panelu, to vyšle první bednu a zbytek je již automatický. V klidu můžete pozorovat, jak se krabice budou samy třídit, velké do leva, malé doprava.



Obrázek 6.3: : Třídění krabic [30]

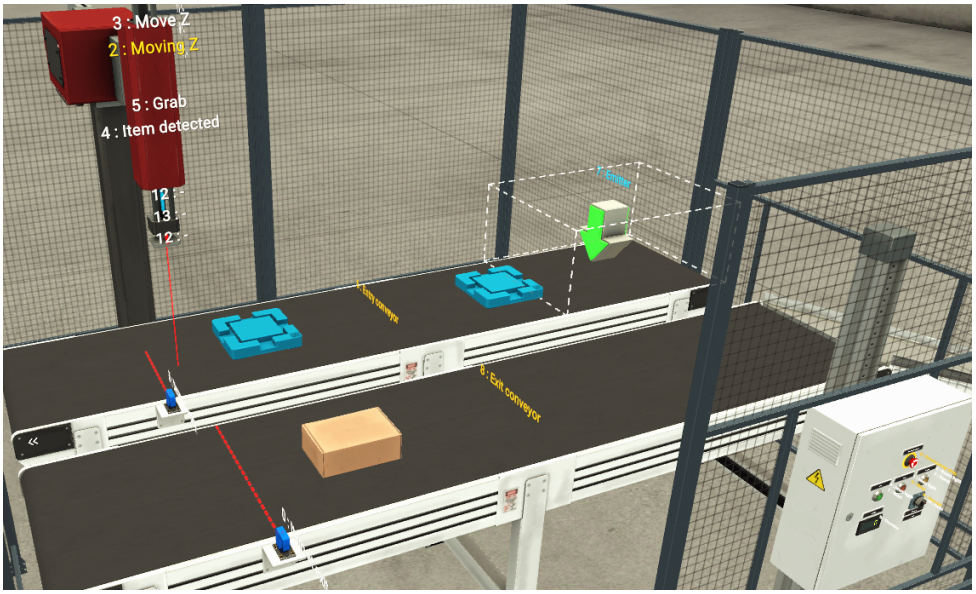
Na Obrázku 6.4 můžeme vidět část logického zapojení třídění v Control I/O. Tato konkrétní část spouští proces odeslání předmětu do leva na základě signálu senzoru výšky.




Obrázek 6.4: : Část logického zapojení [30]

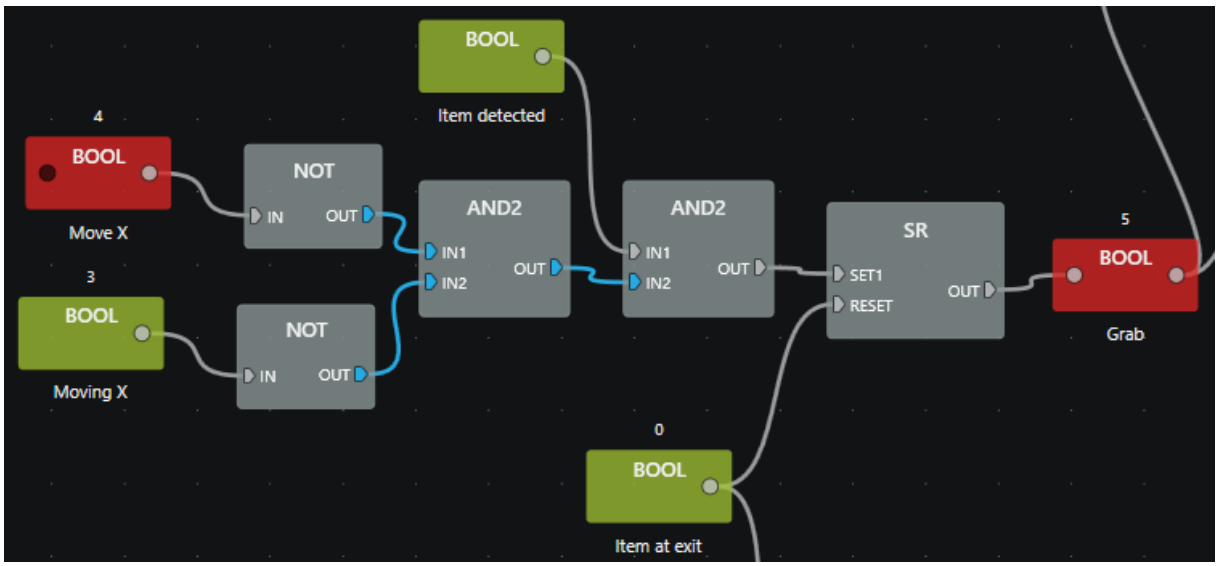
### 6.3.2 Zvedni a polož

Stejně jako u předchozí simulace, jedná se zde o předem vytvořenou scénu. Jedná se o jednoduchou scénu kde manipulátor přesouvá předměty z jednoho pásu na druhý. Simulaci stačí zapnout, vše ostatní by mělo probíhat automaticky.




Obrázek 6.5: : Zvedni a polož [30]

Obrázek 6.6 ukazuje jak funguje logické zapojení úchopu. Jak je vidět, ke spuštění úchopu jsou zapotřebí tři aktivní proměnné, první je aby senzor detekoval objekt, druhé dva se týkají pohybu v ose X. Úchop se resetuje proměnou difúzního snímače na výstupním pásu.




Obrázek 6.6: : Logické zapojení úchopu [30]

# 7 Srovnání simulačních softwarů

V této kapitole budou porovnány softwary z předchozích kapitol na základě vybraných faktorů. Simulační software byl vybrán zadáním, pravděpodobně na základě dostupnosti škole a přístupnosti zkušební verze. Na základě níže zvolených faktorů jsem vyhledal potřebné informace a testoval vybraný software.

1. **Cena/dostupnost**
2. **Uživatelské prostředí**
3. **Funkce**
4. **Dokumentace a Podpora**
5. **Aktualizace**

## 7.1 V-rep

1. **Cena/dostupnost**

   V-rep je svobodným softwarem, nabízí tedy na svých stránkách dvě volné verze a jednu placenou. Jednou z nich je Edu verze, což představuje plnou verzi použitelnou k výukovým potřebám. Druhou verzí je Player, který slouží i ke komerčnímu užitku, nicméně oproti Edu verzi nedovoluje plné možnosti úprav. Třetí placená licence kombinuje dvě předchozí, cena je pouze na vyžádání.

2. **Uživatelské prostředí**

   Prostředí aplikace je jedno z lepších. Pro začátečníka jednoduchý a pohodlný způsob jak se učit s robotickým simulátorem. Hierarchie scény je výborným pomocníkem pro uživatele, který mu dává dodatečnou kontrolu nad celou scénou.

3. **Funkce**

   V-rep poskytuje veškeré funkce, které uživatel potřebuje od robotického simulátoru, nicméně je na aplikaci znát, že jde o svobodný software. Programování lze aplikovat šesti různými způsoby, podporuje řadu programovacích jazyků jako jsou C, C++, Python, Java, Lua, Matlab anebo Octave. Dále obsahuje funkce jako je vzdálené API a řízení přes ROS, které propojují V-rep s externí aplikací umístěné na jiném

zařízení. Kromě vlastní rozsáhlé knihovny modelů, dovoluje výrobu vlastního modelu i import CAD modelů.

4. **Dokumentace a Podpora**

   Na stránkách prodejce je k dispozici anglický manuál k aktuální verzi aplikace 4.2.0. Obsahuje podrobné návody základních i pokročilých funkcí. Pravděpodobně způsobeno tím, že jde o svobodný software, má V-rep tvořivou komunitu, na internetu lze najít řadu video tutoriálů. Prodejce na své stránce také provozuje veřejně přístupné fórum.

5. **Aktualizace**

   S interval půl roku mezi verzemi a poslední verzí 4.2.0 vydanou v dubnu tohoto roku, bych soudil že vývojáři na aplikaci dále pracují.

## 7.2 CIROS Studio

1. **Cena/dostupnost**

   Cena pouze na vyžádání. Dostupné demo je výrazně omezeno jak o většinu funkcí, tak i časově na 30 dní.

2. **Uživatelské prostředí**

   Na první pohled vypadá chaoticky, jeho rozhraní obsahuje velké množství různých oken a ikon s velkým výběrem možností, což pro začátečníka tvoří zmatky. Pro již zkušeného pracovníka v tomto prostředí, jde o velmi kvalitní software.

3. **Funkce**

   Aplikace poskytuje všechny potřebné funkce pro modelování, vytváření digitálního dvojčete i jeho zprovoznění. Z oblasti modelů nabízí rozšířitelnou knihovnu, vytváření vlastních i import CAD modelů. Nabízí možnost připojení virtuálního PLC i zapojení HiL pro testování simulace v reálním čase. Programování probíhá v původních jazycích robotů. Je tedy plně využitelný pro uplatnění principů Průmyslu 4.0.

4. **Dokumentace a Podpora**

   Manuál poskytnutý na stránkách prodejce je bohužel z roku 2008, není tedy příliš aktuální. Některé návody a popisy již neodpovídají novějším verzím programu. Distributor CIROS Studio nabízí na svých stránkách zaškolení a konzultace. V nevelkém množství lze také nalézt tutoriály vytvořené komunitou.

5. **Aktualizace**

   Je nabízena řada rozšíření zaměřených na různé výrobce jako jsou ABB, Kuka a Mitsubishi mají každá svůj balíček přidávající sadu jejich robotů nebo propojení s reálným kontrolérem. jako je například VR package, který podporuje použití VR brýlí.

## 7.3 Factory I/O

1. **Cena/dostupnost**

   Aplikace je nabízená v několika různých typech licence. Začínající na 36 €/rok a nejdražší je za 253 €/rok. Dražší licence většinou přidávají nové virtuální PLC. Nabízen je také Free trial na 30 dní.

2. **Uživatelské prostředí**

   Jednoznačně software s nejvíce intuitivním uživatelským prostředím. Knihovna objektů i řízení jsou přehledné a jednoduché. Aplikace je tedy velice vhodná pro výuku začátečníků.

3. **Funkce**

   Jde především vzdělávací software, chybí mu příliš mnoho funkcí na to, aby byl použitelný v reálném pracovním prostředí. Naprosto postrádá možnost tvorby nebo importu vlastního modelu. Knihovna neobsahuje téměř žádné robotické manipulátory. Nejde tedy o robotický simulátor, ale o vzdělávací platformu automatizačních prvků.

4. **Dokumentace a Podpora**

   Široká nabídka návodů i manuálu na internetu. Obojí je přehledné a k aktuální verzi. Factory I/O na svém webu spravují komunitní fórum. Lze nalézt i větší množství video tutoriálů.

5. **Aktualizace**

   S půl roční frekvencí vydávání verzí se dá usoudit že vývojáři aktivně na produktu pracují.

# 8 Závěr

Hlavním cílem této práce bylo provést rešerši softwarových možností simulace robotických systémů. Analyzovat využití těchto softwarů v automatických systémech dle principů Průmyslu 4.0. Navrhnout a otestovat jejich využití pro vytvoření digitálního dvojčete a jeho zprovoznění. Nakonec byli rešeršované softwary porovnávány podle různých kritérií.

V první, teoretické části bylo provedeno seznámení s principy Průmyslu 4.0, pojmy digitálního dvojčete, virtuálního zprovoznění a dalších s nimi spjatých. Bylo objasněno označení čtvrté průmyslové revoluce, charakterizované koncepty Průmyslu 4.0 a podrobně vysvětlena pyramida automatizace i všechny její části. Od sekce 2.2 se téma posunulo k problematice digitálního dvojčete, virtuálního zprovoznění a jeho metodám. Poslední sekce této kapitoly se věnuje definici simulace samotné.

Druhá část obsahuje samotnou rešerši softwarů. Probíhá zde představování jednotlivých aplikací a jejich vlastností. Snahou bylo určit zda disponují funkcemi, které jsou vhodné pro vytvoření digitálního dvojčete a jestli je možné jeho zprovoznění.

V třetí části, v kapitole 7 bylo provedeno porovnání rešeršovaných softwarů podle pěti parametrů. Snaha byla srovnat je podle důležitosti od prvního do posledního, ale váha jednotlivých kritérií může být individuální. Výsledkem je zhodnocení jednotlivých softwarů, z kterého je patrné, že programy nelze jednoznačně srovnat od nejlepšího k nejhoršímu, ale že každý se hodí na něco jiného.

CIROS Studio se ukázalo jako nejlepší kandidát z funkčního hlediska. Ale naopak je díky svému chaotickému uživatelskému prostředí není nejlepší volbou pro učení začátečníků v oboru.

Factory I/O je přesným opakem, jelikož se jedná především o výukovou platformu a je tedy pro nováčky skvělým začátkem. Ale jeho využití v průmyslu je zpochybnitelné.

V-rep je nakonec slušným kompromisem oproti ostatním dvěma. Jakožto svobodný software sice postrádá některé funkce oproti korporátním, ale stále nabízí široké uplatnění jak ve výrobě tak ve výuce.

# Bibliografie


[1] V. Mařík, *Iniciativa Průmysl 4.0*, 2016. WWW: https://www.mpo.cz/assets/dokumenty/53723/64358/658713/priloha001.pdf.

[2] Siemens, *What is Industry 4.0?* WWW: https://www.plm.automation.siemens.com/global/en/our-story/glossary/industry-4-0/29278.

[3] A. Szydlowská, *Nová průmyslová revoluce – Průmysl 4.0*, 2017. WWW: https://www.datamix.eu/blog/nova-prumyslova-revoluce-prumysl-4-0/.

[4] P. Emilio, *Low-Power WAN (LPWAN) for IoT long-range communication*, 2019. WWW: https://www.ednasia.com/low-power-wan-lpwan-for-iot-long-range-communication/.

[5] H Kagermann, W Wahlster a J Helbig, *Securing the Future of German Manufacturing Industry, Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0*. 2013. WWW: https://www.din.de/blob/76902/e8cac883f42bf28536e7e8165993f1fd/recommendations-for-implementing-industry-4-0-data.pdf.

[6] J. Kletti, *Manufacturing Execution System*. Springer, 2007, ISBN: 978-3-540-49744-8.

[7] Z. Machala, *Příprava a řízení výroby s vertikální integrací systémů*, Diplomová práce, 2019. WWW: https://dspace.cvut.cz/handle/10467/86247.

[8] J. Mantle, *The 5 Layers of the Automation Pyramid and Manufacturing Operations Management*, 2019. WWW: https://www.syspro.com/blog/erp-for-manufacturing/the-5-layers-of-the-automation-pyramid-and-manufacturing-operations-management/.

[9] HiveMQ, *Modernizing the Smart Manufacturing Industry with MQTT*. WWW: https://www.hivemq.com/solutions/manufacturing/modernizing-the-manufacturing-industry/.

[10] L. Laperriere a G. Reinhart, *Encyclopedia of Production Engineering*. Berlin: Springer Berlin Heidelberg, 2014, ISBN: 978-3-642-20616-0.

[11] Keil, *ERP systém*. WWW: https://www.vaclavkeil.cz/erp-system/.

[12] I. ŠTRUBLÍKOVÁ, *MES Systémy ve strojírenství*. WWW: http://mescenter.org/cz/clanky/6-zakladni-funkcionality-mes-systemu.

[13] TULIP, *Manufacturing Execution Systems*. WWW: https://tulip.co/resources/mes-ultimate-guide/#Introduction..

[14] D. Telecom, *How Do SCADA Systems Work?* WWW: https://www.dpstele.com/scada/how-systems-work.php.

[15] AUTOMA, *Funkční model stroje ve virtuálním prostředí usnadní konstrukci a testování*, 2015. WWW: http://automa.cz/Aton/FileRepository/pdf_articles/53524.pdf.

[16] D. Kalina, *Simulace výukových sestav strojů s využitím NX Mechatronics Concept Designer*, Diplomová práce, 2018. WWW: https://dspace.cvut.cz/handle/10467/80534.

[17] V. Staněk, *NÁVRH DIGITÁLNÍHO DVOJČETE CNC OBRÁBĚCÍHO STROJE*, Diplomová práce, 2019. WWW: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=193433.

[18] A. Parrott a L. Warshaw, *Industry 4.0 and the digital twin: Manufacturing meets its match*, 2017. WWW: https://www2.deloitte.com/content/dam/Deloitte/cn/Documents/cip/deloitte-cn-cip-industry-4-0-digital-twin-technology-en-171215.pdf.

[19] FactorioSolutions, *Bulletin Průmyslu 4.0, Digitální dvojčata*, 2018. WWW: https://www.ncp40.cz/files/bulletin-prumyslu4-2018-07.pdf.

[20] M. Kulhánek, *DIGITÁLNÍ DVOJČE*, 2020. WWW: https://www.ncp40.cz/files/skoda-auto-prezentace-digidvojce-121120.pdf.

[21] M. Dlouhý, J. Fábry, M. Kuncová a T. Hladík, *Simulace podnikových procesů*. Brno: ComputerPress, 2007, ISBN: 978-802-5116-494.

[22] L. Volf, *Počítačová simulace v oblasti navrhování a optimalizace výrobních systémů*, 2010. WWW: https://stc.fs.cvut.cz/history/2010/sbornik/papers/pdf/VolfLudek-301296.pdf.

[23] Festo Didactic, *RobotinoWiki*. WWW: https://wiki.openrobotino.org/index.php?title=Main_Page.

[24] RoboDK Inc., *RoboDKl*. WWW: https://robodk.com/.

[25] COPPELIA ROBOTICS, *V-REP User Manual*, 2016. WWW: http://www.coppeliarobotics.com/helpFiles/.

[26] Bullet Real-Time Physics Simulation, *Home of Bullet and PyBullet*, 2019. WWW: https://pybullet.org.

[27] R. Smith, *Open Dynamics Engine*. WWW: https://www.ode.org/.

[28] Newton Dynamics, *About Newton Dynamics*. WWW: http://newtondynamics.com.

[29] VEROSIM SOLUTIONS, *CIROS STUDIO*. WWW: https://www.verosim-solutions.com/en/industry/ciros-studio.

[30] Real Games, *Factory I/O manual*. WWW: https://docs.factoryio.com/.