

VŠB – Technická univerzita Ostrava

Fakulta strojní

Katedra robotiky

Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert

Design of the Robotized Workcell Using
the Simulation Software RobotExpert

Student:

Bc. Tomáš Krejčí

Vedoucí bakalářské práce:

Ing. Tomáš Kot, Ph.D.

Ostrava 2019

VŠB - Technická univerzita Ostrava
Fakulta strojní
Katedra robotiky

Zadání diplomové práce

Student: **Bc. Tomáš Krejčí**

Studijní program: N2301 Strojní inženýrství

Studijní obor: 2301T013 Robotika

Téma: **Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert**
Design of the Robotized Workcell Using the Simulation Software RobotExpert

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Proveďte analýzu možností vytváření robotizovaných simulačních pracovišť v prostředí simulačního systému RobotExpert.
2. Proveďte analýzu dostupných knihoven robotů a periferií a možností doplňování těchto knihoven.
3. Proveďte analýzu vazeb systému RobotExpert na ostatní softwarové komponenty systému Tecnomatix a zhodnoťte možnosti využití stávajících CAD systémů Creo a Solidworks pro tvorbu mechanismů v systému RobotExpert.
4. Vytvořte na vhodném příkladu robotizovaného pracoviště manuál systému RobotExpert.

Seznam doporučené odborné literatury:


1. *RobotExpert Release Notes*. [online] Dostupné z https://docs.plm.automation.siemens.com/tdoc/tecnomatix/13.1/RBX#uid:index_xid1135295
2. *RobotExpert Reference Manual*. Dostupné z https://docs.plm.automation.siemens.com/tdoc/tecnomatix/13.1/RBX#uid:index_xid1015930
3. *RobotExpert Getting Started Videos*. Dostupné z https://www.youtube.com/playlist?list=PL1m1vu8_quoCkCBRbz6OhfiRy0sL66jvW

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí diplomové práce: **Ing. Tomáš Kot, Ph.D.**

Datum zadání: 21.12.2018

Datum odevzdání: 20.05.2019


prof. Dr. Ing. Petr Novák
vedoucí katedry




prof. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě dne 20. května 2019



.....
Podpis studenta

Prohlašuji, že:

- jsem si vědom, že na tuto moji závěrečnou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. Zákon o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (dále jen Autorský zákon), zejména § 35 (Užití díla v rámci občanských či náboženských obřadů nebo v rámci úředních akcí pořádaných orgány veřejné správy, v rámci školních představení a užití díla školního) a § 60 (Školní dílo),

- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo užit tuto závěrečnou diplomovou práci nekomerčně ke své vnitřní potřebě (§ 35 odst. 3 Autorského zákona),

- bude-li požadováno, jeden výtisk této diplomové práce bude uložen u vedoucího práce,

- s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užit dílo v rozsahu § 12 odst. 4 Autorského zákona,

- užit toto své dílo, nebo poskytnout licenci k jejímu využití, mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše),

- beru na vědomí, že - podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů - že tato diplomová práce bude před obhajobou zveřejněna na pracovišti vedoucího práce, a v elektronické podobě uložena a po obhajobě zveřejněna v Ústřední knihovně VŠB-TUO, a to bez ohledu na výsledek její obhajoby.

V Ostravě dne 20. května 2019



Podpis autora práce

Jméno a příjmení autora práce:

Bc. Tomáš Krejčí

Adresa trvalého pobytu autora práce:

Velká Strana 163, Dubicko 789 72

ANOTACE DIPLOMOVÉ PRÁCE

KREJČÍ, T. *Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert*: diplomová práce. Ostrava: VŠB – Technická univerzita Ostrava, Fakulta Strojní, Katedra robotiky, 2019, 122 s. Vedoucí práce: Kot, T.

Diplomová práce se zabývá metodikou tvorby robotizovaných pracovišť v simulačním prostředí softwaru RobotExpert z produktové řady Tecnomatix Siemens PLM. V úvodu jsou naznačeny aktuální požadavky trhu na právě probíhající čtvrtou průmyslovou revoluci označovanou jako Průmysl 4.0. Následně je vysvětleno, proč je RobotExpert jedním z důležitých nástrojů rozvíjejícího se průmyslu. V práci jsou popsány pracovní postupy a vysvětleny důležité funkce softwaru. Součástí práce jsou taktéž video-postupy pro nejčastěji využívané robotické aplikace. Z práce se tak stává vhodný studijní materiál pro všechny zájemce o simulace a off-line programování průmyslových robotů. Díky velké podobnosti systému lze získané poznatky uplatnit nejen pro práci se softwarem RobotExpert, ale také pro software Process Simulate a obdobné produkty z řady Tecnomatix.

ANNOTATION OF DIPLOMA THESIS

KREJČÍ, T. *Design of the Robotized Workcell Using the Simulation Software RobotExpert*: Diploma Thesis. Ostrava: VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Robotics, 2019, 122 p. Thesis head: Kot, T.

This thesis focuses on the methodology of creating robotized workplaces in the simulation interface of RobotExpert software from the Tecnomatix product line by Siemens PLM. The introduction outlines current market requests for the ongoing fourth industrial revolution called Industry 4.0. Subsequently, it explains the crucial role of RobotExpert in the emerging industry. The core of the thesis consists of detailed description of working procedures and explains important software features. The work also includes video-procedures for the most commonly used robotic applications. Thus, the thesis becomes suitable study material for all those who are interested in simulations and off-line programming of industrial robots. Thanks to great similarity between systems it is possible to apply gained knowledge not only in work with RobotExpert software but also for similar products from Tecnomatix series.

Seznam použitých termínů

ARC anglické označení pro obloukové svařování.

CAD (z angličtiny computer-aided design) je počítačová podpora projektování.

Cobot (z anglického collaborative robot) je robot, který je navržen pro přímou fyzickou interakci s člověkem.

Cojt je formát souboru se všemi informacemi pro práci v softwaru RobotExpert.

DCM je (anglická zkratka pro dynamic control model) dynamicky řízený model.

Digitální dvojče je komplexní virtuální model stroje simulující jeho operace, efektivnost, model, nástroje, spotřebu materiálu, servisní činnosti apod.

Digitální podnik je komplexní virtuální model podniku simulující celou síť procesů, jejich metod, modelů, nástrojů, servisních činností apod.

Efaktor je výkonným subsystémem robotu.

GTAC (zkratka pro global technical acces center) je technická podpora produktu firmy Siemens.

International Federation of Robotics (zkratka **IFR**) je mezinárodní robotická federace.

Joint (česky kloub) je označení které se používá v robotice pro kloubové spojení s jedním nebo více stupni volnosti pohybu.

Jt je formát souboru modelu, který je součástí Cojt souboru.

Off-line programování (zkratka OLP) je způsob programování robotů ve virtuálním prostředí.

OM je zkratka pro objekt manipulace.

Pick and Place je (anglický výraz pro seber a umístí) proces přemísťování objektů.

PLM Tecnomatix je portfolio softwarů zajišťující digitalizaci celého podniku od firmy Siemens.

Prismatický joint je typ kloubu umožňující translační pohyb.

Process Simulate (zkratka **PS**) je nástroj pro simulaci a off-line programování robotických buněk a člověka.

RCS modul je virtuální řídicí jednotka robotu (kontrolér), který ví, jak se vyrovnat se vstupy a výstupy RRS.

RobotExpert (zkratka **RE**) je nástroj pro simulaci a off-line programování robotických buněk.

RRS je mezinárodní norma specifikující komunikační algoritmy a protokoly pro každého z robotů zastoupených v simulačním prostředí.

Souřadný systém (zkratka **SS**) umožňuje jednoznačně popsat polohu bodu v prostoru pomocí čísel.

TCPF (anglická zkratka pro tool center point frame), jedná se koncový bod robotu.

Wobj (anglicky work object) je definovaný prostor ke kterému jsou vztaženy souřadnice vykonávaného procesu.

XML je zkratka z anglického eXtensible Markup Language, rozšiřitelný značkovací jazyk.

Seznam obrázků

Obrázek 1.1 – Typické příklady průmyslových robotů [1]	14
Obrázek 1.2 – Osy robotu [1]	14
Obrázek 1.3 – Světový vývoj robotů v průmyslu [2]	15
Obrázek 1.4 – Čtyři průmyslové revoluce [3]	15
Obrázek 1.5 – IoT [4]	16
Obrázek 1.6 – Digitální řešení pro životní cyklus stroje [5]	16
Obrázek 1.7 - Teachpendant	17
Obrázek 1.8 - Představení Tecnomatix [6]	19
Obrázek 1.9 – Modularita Tecnomatix [7]	19
Obrázek 1.10 - Úvodní obrazovka používaného softwaru V.13.1.2	22
Obrázek 1.11 - Pracovní prostředí RE	23
Obrázek 2.1 - Přístup k oknu Customize	24
Obrázek 2.2 - Lišta nástrojů	25
Obrázek 2.3 - Pick Intent	26
Obrázek 2.4 - Pick Level	26
Obrázek 2.5 – Navigation Cube	26
Obrázek 2.6 – Strom objektů	28
Obrázek 2.7 - Zobrazit/Skrýt	28
Obrázek 2.8 - Seznam skupin	28
Obrázek 2.9 - Data v Excelu	28
Obrázek 2.10 - Screen Layout	29
Obrázek 2.11 - Layout Manager	29
Obrázek 2.12 - Nastavení cesty knihovny	30
Obrázek 2.13 - Knihovna robotů a periférií	30
Obrázek 2.14 - GTAC podpora	31
Obrázek 2.15 - Vkládání komponent	31
Obrázek 2.16 - Stažení robotu od výrobce	32
Obrázek 2.17 - Převod formátu	32
Obrázek 2.18 - Výběr souboru pro převod	33
Obrázek 2.19 - Typ převáděného souboru	33
Obrázek 2.20 - Potvrzení převodu	34
Obrázek 2.21 - Kontrola úspěšného převodu	34
Obrázek 2.22 - Knihovna s roboty	34
Obrázek 2.23 - Převod CAD modelů	35
Obrázek 2.24 - Definice součásti	35
Obrázek 2.25 - Vkládání složitých souborů	36

Obrázek 2.26 - Využití textur [9].....	36
Obrázek 2.27 - PMI data [9].....	36
Obrázek 2.28 - Funkce Attach.....	37
Obrázek 2.29 - Nastavení jednotek.....	38
Obrázek 2.30 - Modelování součástí.....	38
Obrázek 2.31 - Tvorba krychle.....	39
Obrázek 2.32 - Umístění vytvořeného modelu.....	39
Obrázek 2.33 - Sjednocení.....	40
Obrázek 2.34 - Rozdíl.....	41
Obrázek 2.35 - Průnik.....	41
Obrázek 2.36 - Měřítka součástí.....	42
Obrázek 2.37 - Měřítka z bodu do bodu.....	42
Obrázek 2.38 - Tvorba skici.....	43
Obrázek 2.39 - Vytažení skici.....	43
Obrázek 2.40 - Skica z kontury.....	44
Obrázek 2.41 - Změna barvy součástí.....	44
Obrázek 2.42 - Duplikace objektů.....	45
Obrázek 2.43 - Úpravy objektu.....	46
Obrázek 2.44 - Placement Manipulator.....	47
Obrázek 2.45 - Sledování kolize mezi objekty.....	48
Obrázek 2.46 – Nastavení kolizní dovjice.....	48
Obrázek 2.47 – PM.....	49
Obrázek 2.48 – Relocate.....	49
Obrázek 2.49 - Poznámky, rozměry.....	50
Obrázek 2.50 - Ukázka poznámky.....	50
Obrázek 2.51 - Snapshot Editor.....	52
Obrázek 2.52 - Markup editor.....	52
Obrázek 3.1 - Polohovadlo ABB [10].....	53
Obrázek 3.2 - Převedení formátu polohovadla.....	53
Obrázek 3.3 – Uvolnění polohovadla pro úpravy.....	54
Obrázek 3.4 - Kinematický editor.....	54
Obrázek 3.5 - Kinematika (Base).....	55
Obrázek 3.6 - Vytvoření Linků.....	55
Obrázek 3.7 - Joint Properties.....	56
Obrázek 3.8 - Součást Base.....	56
Obrázek 3.9 - Bod ve středu příruby.....	57
Obrázek 3.10 - Osa pohybu.....	57
Obrázek 3.11 - Konstantní Joint.....	58

Obrázek 3.12 – Kinematika příruby	58
Obrázek 3.13 - Joint Jog	59
Obrázek 3.14 - Pozice polohovadla.....	59
Obrázek 3.15 - Joint Jog pozice.....	60
Obrázek 3.16 - Efektor (závislost pohybu).....	61
Obrázek 3.17 - Závislost Following	61
Obrázek 3.18 - Logicko-matematická závislost	62
Obrázek 3.19 - Create Crank	63
Obrázek 3.20 – Výběr kloubů.....	64
Obrázek 3.21 - Spojení táhel s klouby.....	64
Obrázek 3.22 - Vytvořený paralelogram.....	65
Obrázek 3.23 - Pohyb paralelogramu.....	65
Obrázek 3.24 - Skrytí čar ve stromu objektů	66
Obrázek 3.25 - Joint Jog Robotu.....	67
Obrázek 3.26 - Robot Jog	67
Obrázek 3.27 - Robot s polohovadlem	68
Obrázek 3.28 - Robot Properties.....	68
Obrázek 3.29 - Přidání externího Jointu	69
Obrázek 3.30 - Ovládání externích os	69
Obrázek 3.31 - Kinematika robotu.....	70
Obrázek 3.32 - Kinematika TTT	70
Obrázek 3.33 - Create Toolframe.....	71
Obrázek 3.34 - Nastavení Baseframe	71
Obrázek 3.35 - Efektor	72
Obrázek 3.36 - Mount Tool	72
Obrázek 3.37 - Výběr nástroje	73
Obrázek 3.38 - Nový koncový bod robotu	73
Obrázek 3.39 - Definice nástroje.....	75
Obrázek 3.40 - Ověření správnosti připojení efektoru	75
Obrázek 4.1 - Robot s dráhou pohybu	77
Obrázek 4.2 - Tvorba pozic.....	77
Obrázek 4.3 - Path Editor s pozicemi.....	78
Obrázek 4.4 - Úprava dráhy	78
Obrázek 4.5 - Zobrazení kolizí	79
Obrázek 4.6 - Sequence Editor	80
Obrázek 4.7 - Ověření dosahu robotu	81
Obrázek 4.8 - Smart Place.....	82
Obrázek 4.9 - TCP Tracker	83

Obrázek 4.10 - Vykreslování pohybu [9]	83
Obrázek 4.11 - Sledování vytížení robotu	84
Obrázek 4.12 - Tvorba animace	85
Obrázek 4.13 Zastavení nahrávání	85
Obrázek 5.1 - Pick and Place	87
Obrázek 5.2 - Strom operací	88
Obrázek 5.3 - Dosah robotu	88
Obrázek 5.4 - Continous Process Generator - Arc	89
Obrázek 5.5 - Continous Process Generator - Coverage pattern	90
Obrázek 5.6 - Projekt Arc Seam	90
Obrázek 5.7 - Nastavení hořáku	91
Obrázek 5.8 - Externí osy	91
Obrázek 5.9 - Project Continous Mfg	92
Obrázek 5.10 - Hromadná úprava pozic	93
Obrázek 5.11 - Nová pozice kontinuálního procesu	93
Obrázek 5.12 - Dopravník s objektem manipulace	95
Obrázek 5.13 - Object Flow Operation	95
Obrázek 6.1 - Teach Pendant	96
Obrázek 6.2 - Motion Parameters	96
Obrázek 6.3 - OLP příkazy	96
Obrázek 6.4 - Nastavení kontroléru [11]	98
Obrázek 6.5 - Robot Setup [11]	99
Obrázek 6.6 - Controllers Settings [11]	99
Obrázek 6.7 - Path Editor změny [11]	100
Obrázek 6.8 - Customize Columns	100
Obrázek 6.9 Robotic Program Inventory	101
Obrázek 7.1 - TTT manipulátor	102
Obrázek 7.2 - Buňka Fix	106
Obrázek 7.3 - Skica pro dráhu robotu	108
Obrázek 7.4 - Struktura programu	109
Obrázek 7.5 - Buňka drát	110
Obrázek 7.6 - Buňka Pick and Place	114
Obrázek 7.7 - Kinematika efektoru	115
Obrázek 7.8 - Rozmístění kostek	117

Obsah

Seznam použitých termínů.....	6
Seznam obrázků.....	7
Úvod.....	13
1 Průmyslová robotika.....	14
1.1 Průmyslové roboty	14
1.2 Programování robotů	17
1.3 Tecnomatix – představení	19
1.4 RobotExpert	22
2 Základní úkony.....	24
2.1 Orientace pomocí tlačítek myši	24
2.2 Základní příkazy.....	25
2.3 Základní typy objektů	27
2.4 Rozvržení pracovního prostředí	29
2.5 Vkládání součástí.....	30
2.6 Modelování	38
2.7 Umístění součástí	47
2.8 Tvorba poznámek, označení, rozměrů a obrázků.....	50
2.9 Snapshots a Markups funkce	52
3 Kinematika robotů a dalších zařízení.....	53
3.1 Tvorba kinematiky	53
3.2 Rozpohybování robotu	67
3.3 Efektor	72
4 Simulace	76
4.1 Skok nebo přesun robotu na pozici	76
4.2 Dráha pohybu	77
4.3 Detekce kolize.....	79
4.4 Sequence Editor	80
4.5 Ověření dosahu robotu	81
4.6 Zaznamenání trajektorie TCP.....	83
4.7 Monitorování průběhu vlastností robotu	84
4.8 Tvorba animace	85
5 Robotické aplikace.....	87
5.1 Pick and Place	87
5.2 Continous Process Generator (svařování, lakování)	89
5.3 Continous Feature Operation (ostatní kontinuální operace)	91
5.4 Dopravník	95
6 Off-line programování.....	96
6.1 Teach Pendant (pokročilé robotické příkazy).....	96
6.2 Virtuální kontrolér robotu	97
6.3 Optimalizace programu před nahráním do reálného robotu.....	98
6.4 Přenos programu z RobotExpert do robotu a obráceně	101

7	Praktické příklady	102
7.1	Vlastní zařízení TTT	102
7.2	Buňka Fix.....	106
7.3	Buňka Drát.....	110
7.4	Buňka Kostky (Pick and Place)	114
8	Závěr	118
9	Seznam použité literatury.....	120
10	Seznam příloh	122

Úvod

Tato diplomová práce se zabývá porozuměním simulačního softwaru RobotExpert a vysvětlení jeho funkcí. Nejdříve jsou popisovány jednotlivé funkce zvlášť a závěrem jsou použity v komplexních praktických úlohách. Tyto úlohy jsou doprovázeny i videi s kompletním postupem tvorby simulačních robotizovaných pracovišť.

RobotExpert je software od společnosti Siemens PLM a slouží pro simulaci a off-line programování robotizovaných pracovišť. Program je uzpůsobený pro práci s roboty různých značek. Siemens poskytuje i obdobný software Process Simulate, který slouží spíše pro komplexnější složená pracoviště. Má téměř totožné pracovní prostředí i postupy jako RobotExpert, až na možnost využití více funkcí. Z toho důvodu se tato práce stává vhodným počátečním studijním materiálem pro všechny zájemce pracující v kterémkoliv z těchto softwarů.

S rozvojem průmyslu 4.0 dochází k nárůstu robotiky a potřeby nasazení těchto softwarů v průmyslu. Je tedy potřeba zdokonalovat nejen tyto programy, ale i samotné studijní materiály. Vydavatelé těchto softwarů většinou zveřejní pouze vysvětlení některých základních postupů, případně nabízí možnost využívat implementované nápovědy, které ovšem ne vždy odpovídají používané verzi softwaru a jsou psány v angličtině. Další možností pro firmy, které investovaly do pořízení těchto programů je vyslat své zaměstnance na drahé série školení.

Troufám si říct, že tato práce je jednou z mála rozsáhlých návodů napsaných v českém jazyce. Od čtenáře se však očekává alespoň základní znalost angličtiny, jelikož software samotný je v anglickém jazyce. Při čtení této práce je vhodné otevřít si i samotný RobotExpert a jednotlivé funkce si vždy odzkoušet. V případě, že by se vám některé funkce zdály málo detailně popsány, nebojte se využívat i oficiální nápovědu implementovanou do programu.

1 Průmyslová robotika

Počátky průmyslové robotiky sahají až do 60. let 20. století, kdy byl podán první patent na průmyslový robot (obrázek 1.1). Do robotizace už dnes musí investovat každá technologická firma, jinak bude pohlcena svojí konkurencí. V dnešní době poptávka po robotizaci roste exponenciálně a není se čemu divit. Vždyť robot je schopen pracovat nepřetržitě a bezchybně i v nebezpečném prostředí. Na trhu práce je momentálně nedostatek pracovní síly a jedinou možností, jak tento nedostatek kompenzovat je nasazení robotů.

1.1 Průmyslové roboty

Jsou automaticky ovládané, reprogramovatelné víceúčelové manipulátory, programovatelné alespoň ve třech osách. Mohou být pevné, nebo mobilní a jsou určeny k nasazení v průmyslu.



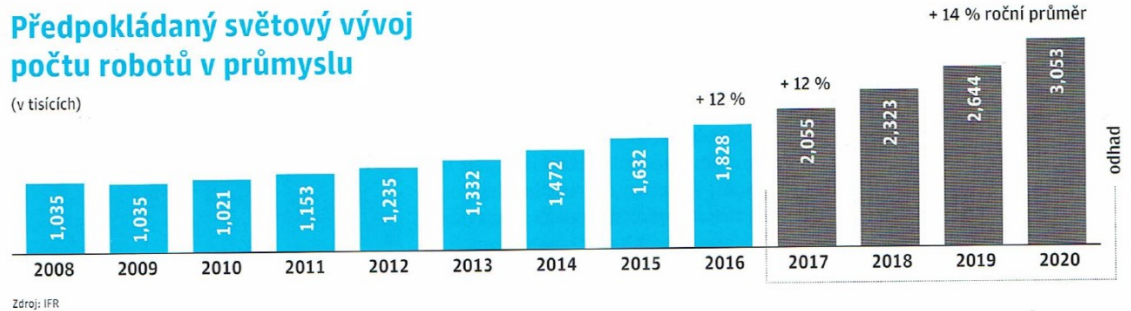
Obrázek 1.1 – Typické příklady průmyslových robotů [1]



Obrázek 1.2 – Osy robotu [1]

1.1.1 Vývoj robotiky v průmyslu

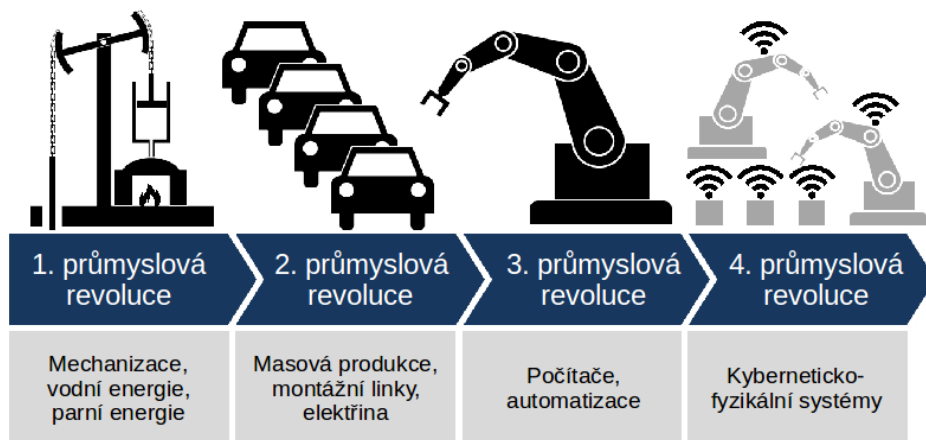
Průmyslové roboty jsou stále dostupnější a zvládají i složitější operace. To jsou další důvody, jež mají za následek masivní nárůst počtu robotů ve výrobě. Samozřejmě tomu napomáhá i nedostatek zaměstnanců ve firmách. Mezinárodní robotická federace (IFR) předpokládá v roce 2020 téměř dvojnásobný počet robotů oproti roku 2016 ve světovém průmyslu. Je tedy potřeba zdokonalovat nástroje a rekvalifikovat zaměstnance pro práci s roboty.



Obrázek 1.3 – Světový vývoj robotů v průmyslu [2]

1.1.2 Průmysl 4.0

Toto označení nese novodobá inovace v průmyslu. Je považována za čtvrtou průmyslovou revoluci. V minulosti ji předcházeli pára, později elektřina a pak počítačové technologie. Nyní to jsou kyberneticko-fyzikální systémy.



Obrázek 1.4 – Čtyři průmyslové revoluce [3]

Podstatou Průmyslu 4.0 je digitalizace, rozšiřování internetu, rozvoj chytrých technologií. Průmysl potřebuje zkrátit čas uvedení produktu na trh bez ztráty kvality. Tomu by mělo pomoci řešení s názvem Digitální podnik. To je takový podnik, který staví veškeré systémy na digitálních technologiích.

S tím souvisí i nový trend internet věcí (anglicky Internet of Things, zkratka IoT), což je systém, ve kterém jsou stroje řízeny na dálku a taktéž jsou schopny posílat důležité informace zpět. To vše je umožněno, díky vybavení všech objektů vloženými čipy, senzory a softwaru. Nejedná se však o použití pouze v průmyslu, ale tento trend zasahuje i do domácností a spotřebičů. Příkladem může být lednička, která sama rozpozná, že dochází mléko a následně upozorní majitele nebo ho dokonce i sama objedná.



Obrázek 1.5 – IoT [4]

Digitální dvojče v digitální továrně bude doprovázet každé strojní zařízení od prvního nápadu až po jeho recyklaci. S virtuálním dvojníkem lze vyladit veškeré chyby zařízení bez rizika i časové a finanční ztráty. Je tedy vytvořeno dvojče, které umožňuje simulaci provozu a toku výroby. Pomocí cloudu lze provádět hodnocení jako jsou počty vyrobených položek, analýza prostojů, poruchovost a energetické údaje. Zařízení taktéž je schopno upozorňovat, případně i domluvit potřebné servisní úkony a prohlídky.



Obrázek 1.6 – Digitální řešení pro životní cyklus stroje [5]

1.2 Programování robotů

Průmyslové roboty lze programovat dvěma způsoby. Jedná se on-line, nebo off-line programování. Softwary jako je RobotExpert umožňují provádět off-line programování. Při on-line nebo taky přímém programování, se lze setkat s naváděním robotu do žádané polohy přes ovládací panel nebo uchopením posledního článku robotu a jeho následného přesunu.

1.2.1 On-line programování

Je typické tím, že je obsluha fyzicky na pracovišti a navádí robot do žádané polohy pomocí přenosného ovládacího panelu (používá se taky označení techpendant, smartpad, flexpendant, apod. - liší se dle výrobce), který je kabelem propojen s řídicím systémem robotu. Může tedy nastavovat údaje jako je způsob, rychlost pohybu (zde je bezpečnostně omezena rychlost při pohybu v ručním režimu) a jiné logické funkce.



Obrázek 1.7 - Teachpendant

V manuálním režimu se používají dva druhy pohybu. Prvním pohybem je pohyb kartézský v souřadnicích X,Y,Z,A,B,C, kde A,B,C je rotační pohyb kolem os X,Y,Z. Druhý druh pohybu je Joint mód, u kterého dochází k natáčení jednotlivých os robotu A1-A6.

Obecně jsou čtyři způsoby, jakými může robot projet zadanou dráhu (vztaženo k nástroji):

PTP (Point to Point) – robot se přesune z počátečního do koncového bodu po nejrychlejší možné dráze, kterou si propočítal.

LINE (lineární pohyb) – robot se pohybuje po úsečce.

CIRC (pohyb po kružnici) – robot se pohybuje po kruhové dráze, která je určena třemi body (počáteční, cílový a pomocný bod)

SPLINE – křivková interpolace pohybu

Druhý způsob online programování je přímé navádění obsluhou tak, že obsluha uchopí nástroj nebo koncový bod robotu a navádí ho do požadované polohy. Robot je schopen si pamatovat koncové body, ale i dráhu, po které se přesouval. Toho lze využívat u aplikací jako je lakování, svařování a podobné. Nejčastěji se tento způsob využívá při nasazování kolaborativních robotů, které je možné provozovat bez bezpečnostního oplocení. Poslední dobou jsou tyto roboty (coboty) na vzestupu.

1.2.2 Off-line programování

Nebo taky nepřímé programování je založena na práci se softwarem pro 3D vizualizaci robotizovaného pracoviště ve virtuálním prostředí. Program většinou obsahuje knihovnu s roboty. Samotné pracoviště lze v omezené míře modelovat přímo v programu, ale častěji dochází k nahrání již vytvořeného 3D modelu a pouhého přiřazení zvoleného robotu. Lze tedy simulovat pohyby robotů, polohovadel a veškeré signály například ze snímačů. Simulace je tvořena definováním koncových bodů nebo dráhy robotu. Lze tak ověřit i případné kolize a jednoduše vyměnit model robotu bez toho, aby firma musela investovat do jiného modelu. Po dokončení procesu v softwaru lze nahrát vytvořený program přímo do fyzického robotu, zde dochází už pouze k mírné úpravě přímo na pracovišti. Velkou výhodou tohoto způsobu je, že není potřeba, aby byl robot fyzicky k dispozici. Nemusí se tedy zastavovat výroba kvůli přeprogramování. Taktéž lze v těchto programech jednoduše tvořit simulace, které jsou oblíbeným výstupem práce pro zákazníka.

Příklady softwarů pro off-line programování:

- **Kuka Sim Pro** (pouze pro roboty KUKA)
- **RobotStudio** (pouze pro roboty ABB)
- **Roboguide** (pouze pro roboty FANUC)
- **Tecnomatix - Process Simulate**
- **Tecnomatix - RobotExpert**
- **Octopuz**
- **SprutCAM Robot**
- **RoboDK**

1.3 Tecnomatix – představení

Tecnomatix firmy Siemens PLM software (obrázek 1.8) je sada pro komplexní řešení digitální výroby. Klade si za cíl umožnit zákazníkovi chytřejší rozhodování, zvyšovat produktivity, snižovat náklady a zvýšit návratnost investic. S nově nastupujícím Průmyslem 4.0 jsou kladeny nároky na digitalizaci výroby. Tento software nabízí nové možnosti zvýšení produktivity. Lze vytvořit kompletní model pracoviště a následně simulovat a optimalizovat výrobu.



Obrázek 1.8 - Představení Tecnomatix [6]

Software Tecnomatix je modulární, to znamená, že lze implementovat pouze moduly, které zákazník potřebuje (obrázek 1.9).



Obrázek 1.9 – Modularita Tecnomatix [7]

Planning

Umožňuje navrhnout digitální model továrny a definovat výrobní postupy a procesy.

Simulation

Před spuštěním výroby simuluje komplexně použití robotů, montáž, logistiku a ergonomičnost.

Production

Fyzická digitalizace výroby. Propojení technologické přípravy s výrobou a umožnění rychlé reakce na změny.

Modul Robotizace

V oblasti off-line programování robotů ze sady Tecnomatix si lze vybrat ze dvou programů, hlavní rozdíl je ten, že **Process Simulate** obsahuje nástroje pro analýzy, simulací člověka, optimalizaci ergonomie atp. Zatímco **RobotExpert** využijí zejména menší firmy, které tuto nástavbu nepotřebují, a pracují s čistě prvky pro off-line programování a simulace. Můžete se stále setkat i se softwarem Robocad, ten je dnes už ovšem považován spíše za předchůdce RobotExpert.

Process Simulate Robotics (PS)

Tecnomatix Process Simulate je integrovaným a komplexním prostředím pro simulaci a optimalizaci výrobních procesů, které umožňuje simulovat funkci jednotlivých pracovišť i celých linek ve 3D prostředí. Je podporována simulace a off-line programování průmyslových robotů v jejich nativním jazyce. Process Simulate obsahuje nástroje pro analýzu kolizí, návrh montážních operací a automatické hledání bezkolizní trajektorie robotu či montovaných dílců. Process Simulate také umožňuje simulování práce člověka, optimalizaci pracovišť z hlediska ergonomie, testování náročnosti jednotlivých lidských činností ve výrobě a přípravu montážních postupů. [8]

RobotExpert (RE)

Tecnomatix RobotExpert je kompaktním softwarovým nástrojem určeným k intuitivnímu off-line programování průmyslových robotů v 3D prostředí. RobotExpert nativně podporuje řídicí jednotky pro roboty řady výrobců, např. ABB, Fanuc, IGM, Kawasaki, Kuka, Staubli, Yaskawa/Motoman. Pomocí RobotExpert lze navrhovat a programovat robotické buňky pro širokou škálu technologických operací, například manipulaci, montáž, odjehlování, lakování, leštění, nanášení lepicí pasty, obloukové svařování, frézování atd. [8]

Rozdíly mezi RobotExpert a Process Simulate

Na první pohled jsou výše zmíněné softwary podobné. Process Simulate je prakticky stejný software rozšířený pouze o funkce, které menší firmy nevyužijí. To je důvod, proč má PS vyšší pořizovací náklady.

Obecně lze říci, že RobotExpert postačí na robotickou buňku, zatímco Process Simulate umožňuje simulovat celé výrobní haly. Studie vytvořené v RobotExpert a Process Simulate nejsou kompatibilní.

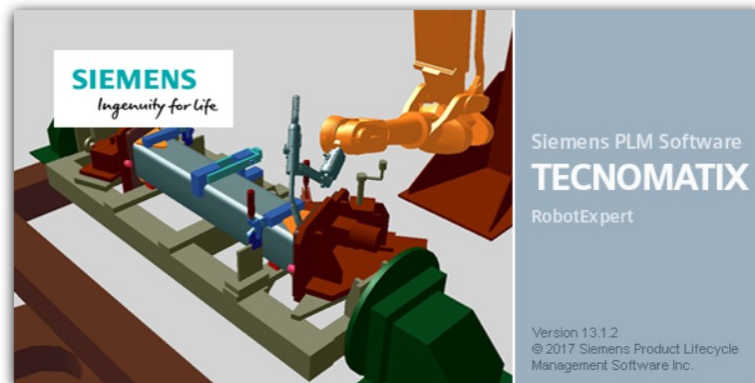
Mezi nejdůležitější funkce, které RobotExpert (oproti PS) nepodporuje, lze zařadit:

- bodové svařování,
- senzory a logické bloky,
- propojení s PLC zařízením,
- simulace řízená událostmi,
- načtení mračen bodů,
- simulace člověka, ergonomie,
- virtuální realita.

V této diplomové práci pracuji se softwarem RobotExpert verze 13.1.2.

1.4 RobotExpert

Základní popis softwaru je zmíněn v předchozí kapitole. V této kapitole následuje výpis základních funkcí a výhod. [9]



Obrázek 1.10 - Úvodní obrazovka používaného softwaru V.13.1.2

Mezi základní funkce softwaru RobotExpert patří:

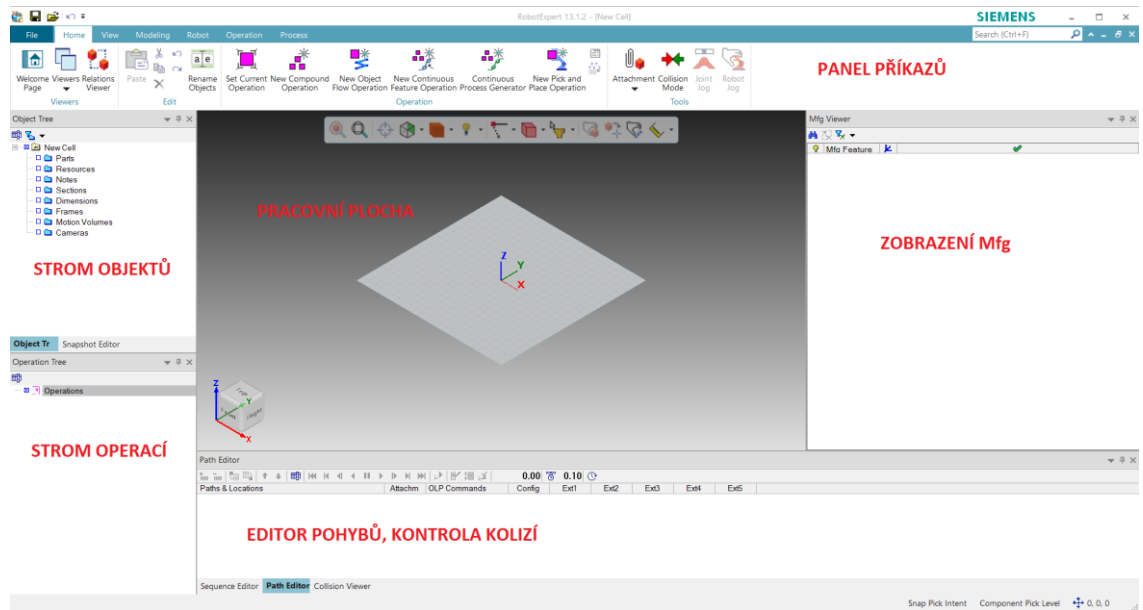
- 3D modelování kinematiky nástrojů, periférií a robotů,
- podpora robotů z široké škály prodejců,
- tvorba a export robotických procesů a drah, včetně logiky,
- detekce kolize,
- výpis operací procesu pomocí Ganttova grafu,
- interakce s kontrolérem robotu,
- off-line programování robotů,
- výpočet přesného času cyklu pomocí realistických simulací robotů (RRS),
- vysoce přizpůsobené rozhraní a funkčnost pro obsluhu,
- možnost doplnit knihovnu robotů nahráním staženého modelu od výrobce,
- podpora průmyslových robotů ABB, Epson, Fanuc, IGM, Kawasaki, Kuka, NC, Staubli, Yaskava, UR, Panasonic, Closs, Denso.

Výhody použití softwaru RobotExpert

- virtuální optimalizace robotických procesů,
- optimalizace času cyklu,
- standardizované programování robotů,
- zkrácené prostoje v případě změny nebo zavedení nového výrobku,
- minimalizuje nebezpečí poranění při zavádění nového zařízení a nákladné škody,
- snadná příprava speciálních syntaxe programů robotu,
- lze nahrát upravený program zpět z robotu do počítače a lze tak udržovat aktuálnost,
- možnost použití 3D brýlí – podpora stereoskopického promítání obrazu.

Pracovní prostředí

Pracovní prostředí programu se základním popisem znázorňuje obrázek 1.11.



Obrázek 1.11 - Pracovní prostředí RE

HW požadavky [9]

Doporučená konfigurace systému

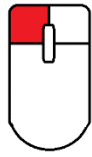
- operační systém Windows 7 pro optimální výkon a uživatelské prostředí,
- 16 GB RAM,
- rozlišení obrazovky: 1280 x 1024 nebo vyšší, širokoúhlý formát pro optimální výkon.

Minimální konfigurace systému

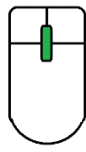
- procesor 64bit (x64),
- operační systémy Windows 10, Windows 8.1, Windows 7,
- 8 GB paměti RAM,
- rozlišení obrazovky: 1280 x 1024,
- pro instalaci je potřeba 3 GB místa na disku.

2 Základní úkony

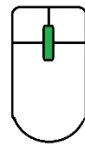
2.1 Orientace pomocí tlačítek myši



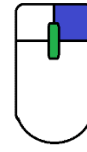
Výběr



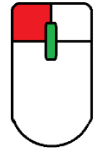
Rotace
(Zmáčknutí)



Přiblížení
(Rolování)



Posunutí



Přiblížení

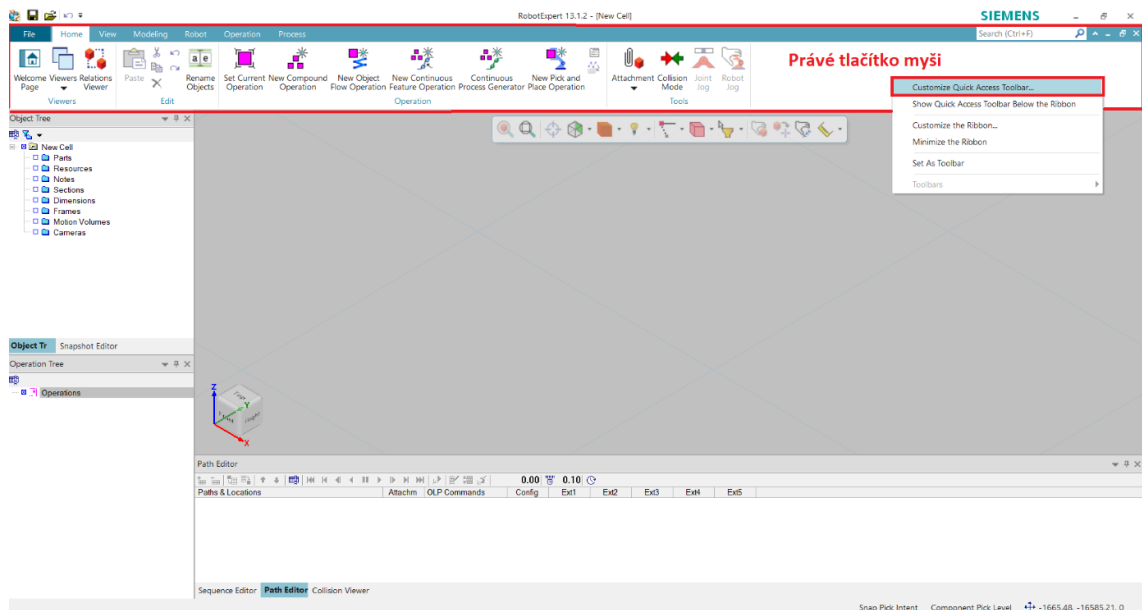
Lze použít i defaultně nastavené kombinace:

SHIFT + KOLEČKO = Posunutí

CTRL + KOLEČKO (zmáčknutí) = Přiblížení

ALT + LEVÉ TLAČÍTKO = Přiblíží vybranou oblast

Pokud si ovšem chcete nastavit své vlastní zkratky pro snadnější ovládání nebo změnit lištu s nástroji, lze tak učinit stisknutím **pravého tlačítka myši v oblasti pracovní lišty**, poté zvolte **Customize Quick Access Toolbar** (obrázek 2.1).



Obrázek 2.1 - Přístup k oknu Customize











Otevře se okno, kde zvolíte, zdali chcete přidávat klávesové zkratky, způsob chování myši, měnit horní lištu nebo panel nástrojů. Pokud je pro vás defaultní nastavení směru otáčení matoucí, doporučuji ho změnit podle zvyklostí ze softwaru PTC Creo Parametric. Otevřete **File** → **Options** → **Graphic Viewer** → **Navigation Settings** a zaškrtněte pouze možnost **Rotate object (Vis method)**.

2.2 Základní příkazy

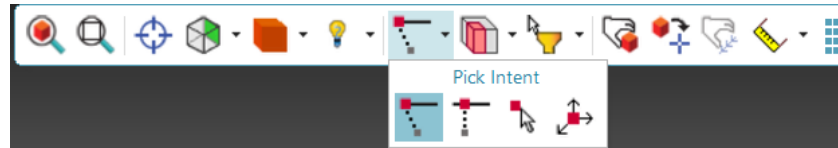
Představení základních nástrojů pro manipulaci v grafickém okně „Graphic Viewer“ obrázek 2.2.







Obrázek 2.2 - Lišta nástrojů

	Zoom to Selection	Přiblíží pohled na označenou komponentu.
	Zoom to Fit	Přiblíží nebo oddálí pohled tak, aby byl obraz vycentrovaný vzhledem k obrazovce.
	View Center	Podle Vámi označeného bodu vytvoří centrum zobrazení, kolem kterého bude rotovat celkový pohled. Model tedy bude vystředěn vzhledem k obrazovce tak, aby byl tento bod uprostřed.
	View Points	Zobrazí model v pohledech nárys, půdorys, bokorys a dalších.
	View Style	Vybere styl zobrazení modelu. V nabídce je stínovaný, zvýraznění hran, drátové zobrazení, skeleton.
	Display	Zobrazí nebo skryje součásti.
	Placement	Přesune objekt do nové pozice přetažením myší nebo zadáním vzdálenosti.
	Relocate	Přemístí objekt z jednoho souřadného systému do druhého.
	Measurements	Měření vzdáleností a úhlů. Podobné jako Create Dimension (Modeling → Note) , s tím rozdílem, že po použití Create Dimension kóta sama nezmizí.
	Selection Filter	Filtr výběru může být použit pro vybrání objektů podle jejich typu.

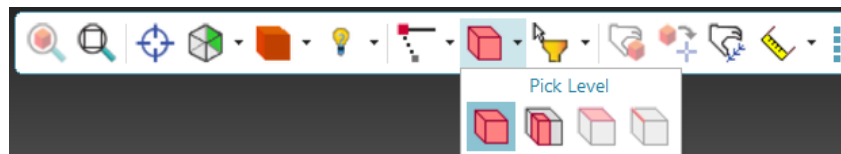
Pick Intent – Vybere přesný bod na objektu (obrázek 2.3).







Obrázek 2.3 - Pick Intent



- 
Snap Vybere vrchol, střed hrany nebo střed plochy, podle toho, který bod je nejbližší.
- 
On Edge Vybere bod na hraně, která je nejbližší bodu, kam jste kliknuli myší.
- 
Where Picked Vybere bod přesně v místě, kde jste kliknuli myší.
- 
Self Origin Vybere souřadný systém součásti.

Pick level – Definuje, zdali se má označit celek nebo část zvolené komponenty (obrázek 2.4).

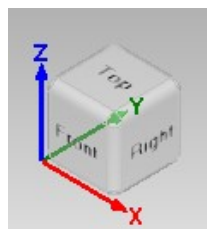


Obrázek 2.4 - Pick Level

- 
Component Označí celý díl/sestavu.
- 
Entity Označí jen vybranou část sestavy.
- 
Surface/Face Označí zvolenou plochu.
- 
Edge Označí hrany.

Poslední dvě možnosti   jsou dostupné pouze při použití určitých příkazů, jako je například Project Arc Seam nebo při užití různých měřících příkazů.

Navigation Cube lze taktéž využít pro zobrazení modelu v požadovaném pohledu. Nachází se v levém spodním rohu pracovního okna (obrázek 2.5).



Obrázek 2.5 – Navigation Cube

2.3 Základní typy objektů

RobotExpert používá pro práci čtyři základní typy objektů. Jsou to **Parts** (součásti), **Operations** (operace), **Resources** (zdroje) a **Manufacturing features** (výrobní funkce).

1. **Parts** jsou součásti, které tvoří výrobek. Strom součástí znázorňuje vzájemné hierarchické propojení jednotlivých dílů.



2. **Operations** jsou akce prováděné za účelem tvorby výrobku, respektive pohybu a simulace procesu. Strom operací je seřazen podle pořadí, ve kterém jsou provedeny.



3. **Resources** definují typ používaného zařízení, např. nástroj, robot, polohovadlo, zařízení a další. Strom zdrojů popisuje seznam použitých zdrojů a jejich pořadí podle doby umístění.

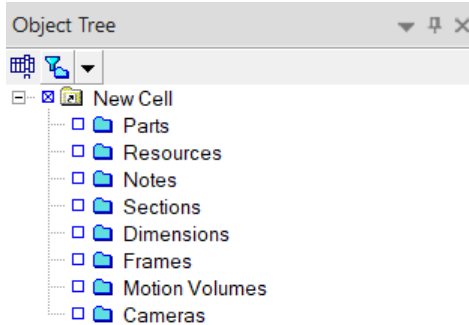


4. **Manufacturing features (Mfg)** jsou výrobní funkce používané pro definování vztahů, mezi součástmi nebo zdroji. Příkladem výrobního prvku je například křivka definující dráhu robotu po obrysu součásti pro operace jako je obloukové svařování, lakování, broušení atd.

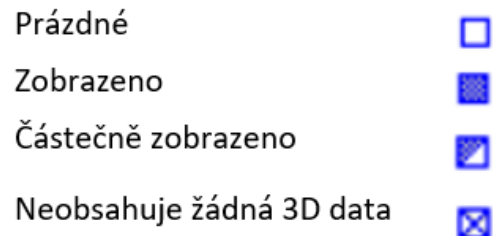


RobotExpert používá objektově orientované prostředí, to znamená, že nejdříve musíte ve stromu objektů zvolit objekty se kterými chcete pracovat, až poté budou k dispozici požadované možnosti.

Object Tree (strom objektů) se defaultně nachází vlevo a zobrazuje všechna používaná zařízení, součásti, kóty, poznámky apod. Význam zbarvení čtverečku u každé součásti vysvětluje obrázek 2.7. Pro **skrytí konkrétní součásti** stačí jen kliknout myší na příslušný čtvereček. Přejmenování jakékoliv součásti provedete klávesou **F2**.



Obrázek 2.6 – Strom objektů



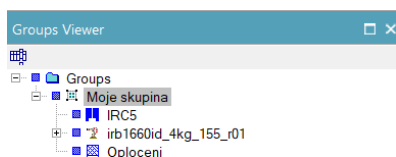
Obrázek 2.7 - Zobrazit/Skrýt

Tvorba skupin součástí může být užitečná funkce, potřebujete-li například vytvořit kusovník součástí nebo opakovaně posouvat více prvků současně.

Zvolte **View** → **Screen Layout** → **Viewers** → **Group viewer**, zobrazí se okno s aktuálně používanými skupinami. Pro tvorbu nových skupin, držte klávesu CTRL a označte ve stromu objektů prvky, které mají vytvořit skupinu. Následně klikněte pravým tlačítkem myši a zvolte **Create Group**, zobrazí se okno, ve kterém je stále ještě možné přiřadit nebo odebrat prvky sestavy. V tomto kroku můžete přiřadit skupině vlastní název.

Pro práci se skupinami musí být požadovaná skupina označena myší. Nyní lze se skupinou pracovat jako s jednou součástí, lze ji posouvat, otáčet apod. Stále je však možné pracovat s jednotlivými prvky nezávisle na skupině. Tato skupina je uložena pouze v buňce a nemá vliv na původní strukturu stromu součástí.

Skupiny lze používat pro hromadné skrytí/zobrazení součástí, vytváření tokových operací, vytváření kolizních párů, export do aplikace Excel například pro tvorbu kusovníků. Stačí pravým tlačítkem myši kliknout na skupinu (v tomto případě „Moje skupina“) a zvolit **Export Groups to Excel** (obrázek 2.8 a obrázek 2.9).





Obrázek 2.8 - Seznam skupin

	A	B	C	D
1	#	Object	Level	Group Hierarchy
2	1	Moje skupina	1	Moje skupina
3	2	Oploceni	2	Moje skupina
4	3	irb1660id_4kg_155_r01	2	Moje skupina
5	4	IRC5	2	Moje skupina

Obrázek 2.9 - Data v Excelu

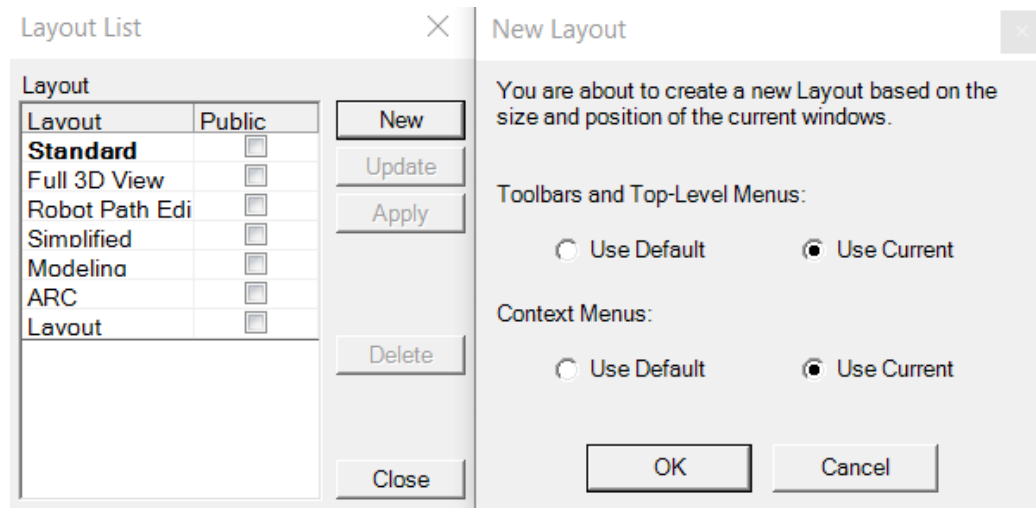
2.4 Rozvržení pracovního prostředí

Zobrazit nebo skrýt podlahu lze pomocí ikony vlevo   , ikona na pravé straně umožňuje změnit rozměry podlahy. Pro změnu pracovního prostředí zvolte záložku **View** → **Screen Layout** → **Layout Manager** (obrázek 2.10).



Obrázek 2.10 - Screen Layout

Defaultně je nastavené pracovní prostředí Standard. To lze změnit rozkliknutím šipky a zvolením již předdefinovaných rozvržení obrazovky. Pomocí Layout Manager lze tato rozvržení upravovat nebo vytvářet nové (obrázek 2.11).



Obrázek 2.11 - Layout Manager

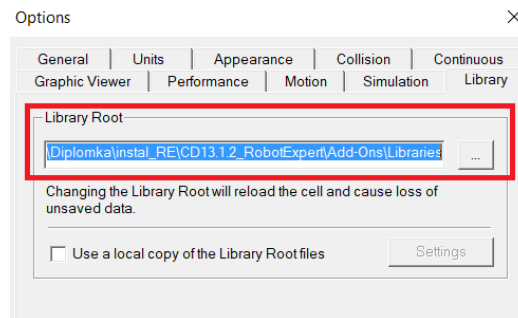
Po zvolení **New** vyberte **Use Current**, po potvrzení se v Layout Listu zobrazí nový, právě vytvořený Layout. Jeho rozvržení je stejné, jaké aktuálně máte, chcete-li to změnit otevřete nebo pozavírejte příslušná okna, vraťte se do Layout Manageru a u požadovaného Layoutu stiskněte **Update**. Jeho název lze změnit po označení a zmáčknutí klávesy **F2**.

2.5 Vkládání součástí

V softwaru RobotExpert rozlišujeme vkládání součástí z knihovny a vkládání (import) modelů cizích formátů, které je nejdříve potřeba převést na srozumitelný formát.

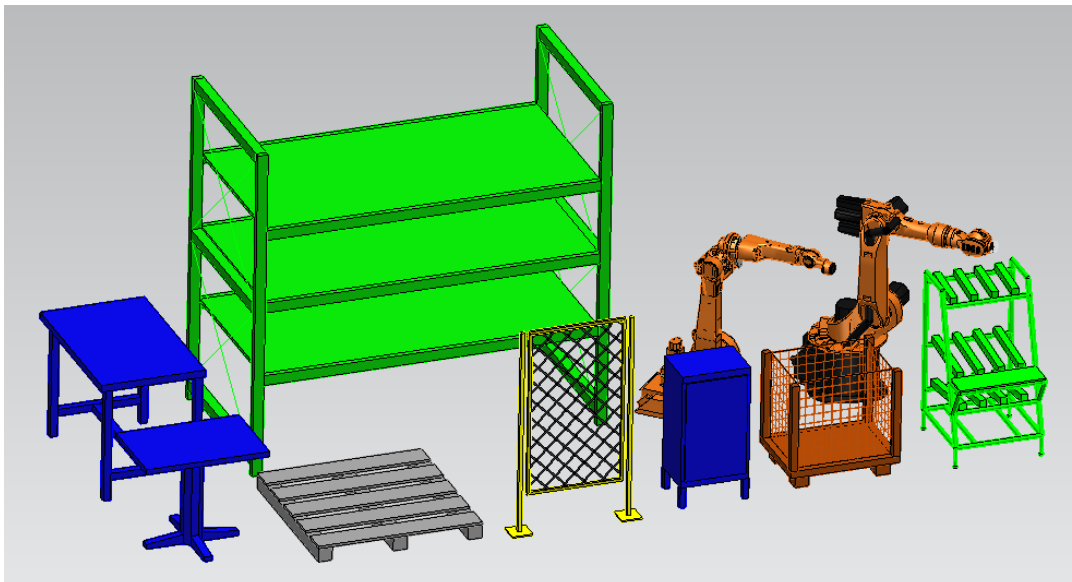
2.5.1 Nastavení knihovny

Otevřete **File** → **Options** → **Library** a nastavte si cestu do vaší knihovny (obrázek 2.12). Máte-li plnou licenci, najdete knihovnu v instalačním souboru **CD13.1.2_RobotExpert** → **Add-Ons** → **Libraries**. Později budeme všechny součásti ukládat a nahrávat právě do této knihovny.



Obrázek 2.12 - Nastavení cesty knihovny

Primárně se v knihovně nachází jen základní periférie jako jsou skříňky, regály, bedny, stoly, palety a oplocení různých velikostí. Naleznete zde pouze roboty značek ABB a KUKA, přičemž v nabídce dohromady není více než 35 robotů. Je tedy vhodné knihovnu doplnit o další modely, ať už vlastní nebo stažené. Typické zástupce ze všech komponent knihovny znázorňuje obrázek 2.13.

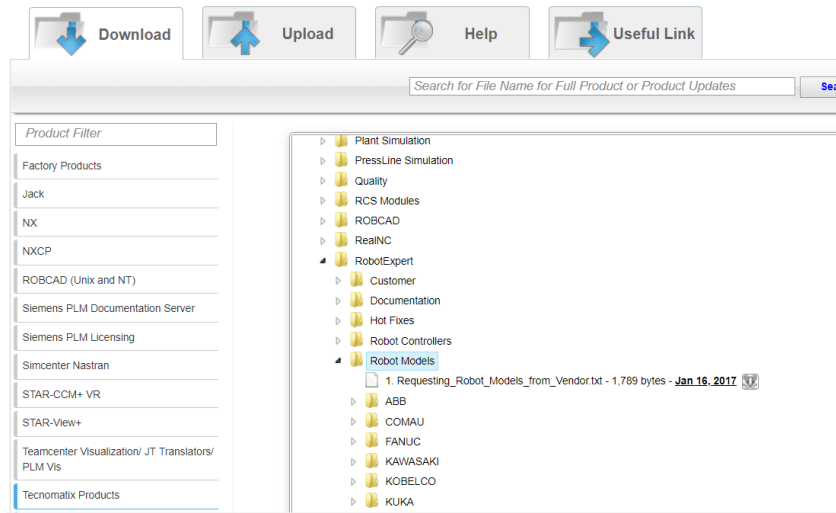


Obrázek 2.13 - Knihovna robotů a periférií

2.5.2 Možnosti doplňování knihovny

Knihovnu lze doplňovat vlastními modely, jednoduše zkopírováním do složky vaší knihovny (případně převodem formátu viz kapitola 2.5.4). Velké množství modelů robotů a kontrolérů, které stačí pouze stáhnout a vložit do složky knihovny naleznete na stránkách siemensu GTAC (obrázek 2.14). Nachází se zde i plné verze jimi dodávaných softwarů, patche, dokumentace apod. Pro přístup se zaregistrujte.

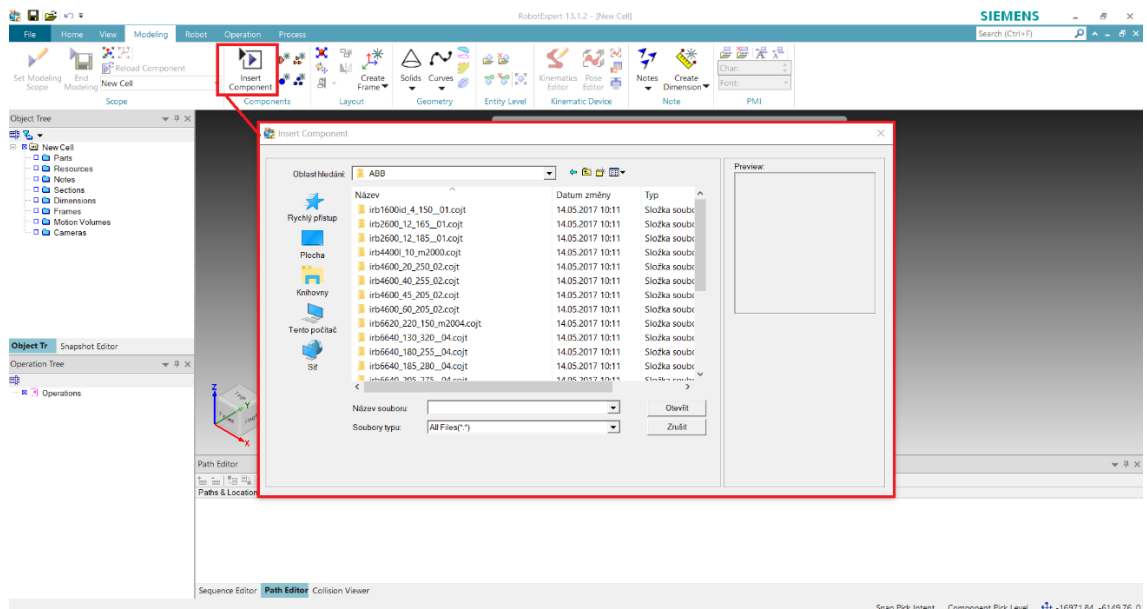
Odkaz na GTAC: <https://download.industrysoftware.automation.siemens.com/>



Obrázek 2.14 - GTAC podpora

2.5.3 Vložení robotu s kinematickými vazbami

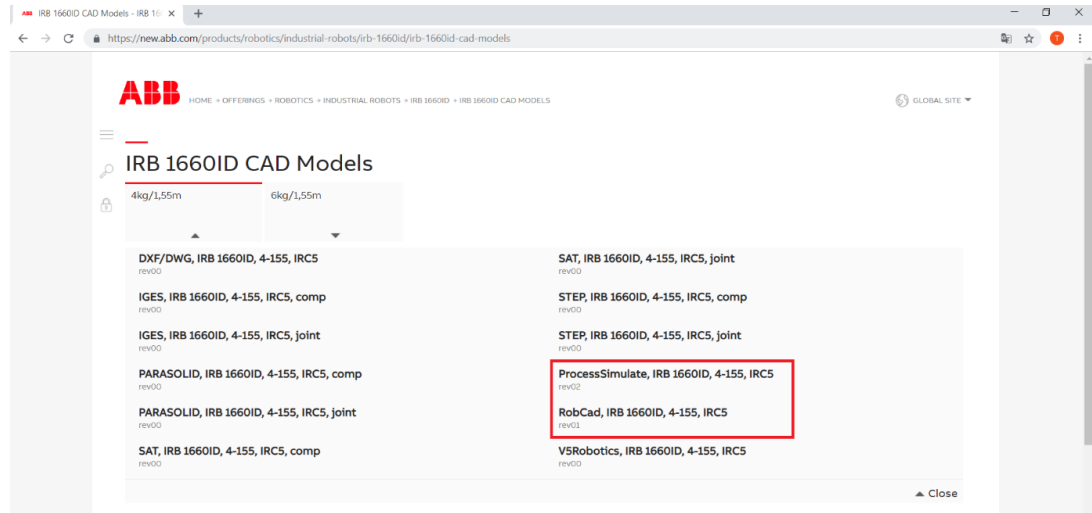
Pro vkládání robotů nebo jiných komponent z knihovny rozklikněte kartu **Modeling** → **Components** → **Insert Component** viz obrázek 2.15.



Obrázek 2.15 - Vkládání komponent

Zobrazí se okno s obsahem poslední používané složky. Pro vkládání součásti je nutné, aby měl soubor příponu **.cojt**. Součást se vloží automaticky do světového souřadného systému, souřadnice [0, 0, 0].

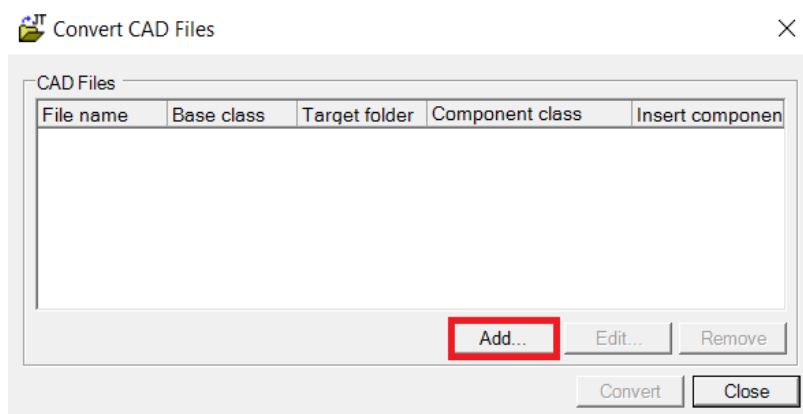
Pokud se ale v knihovně nenachází robot, který potřebujete použít, je nutné tento typ stáhnout ze stránek výrobce robotu. Může se stát, že v nabízených modelech nebude soubor pro **RobotExpert (přípona .cojt)**. Je ovšem možné převést formát ze softwarů **RobCad (přípona .co)** a **Process Simulate (taktéž přípona .cojt ovšem po pouhém vložení, není kompatibilní se softwarem RobotExpert)**. Máte-li možnost volby, doporučuji stahovat soubory pro Process Simulate (obrázek 2.16), průběh převodu se zdá být značně rychlejší. Po převedení už ale fungují oba typy souborů stejně.



Obrázek 2.16 - Stažení robotu od výrobce

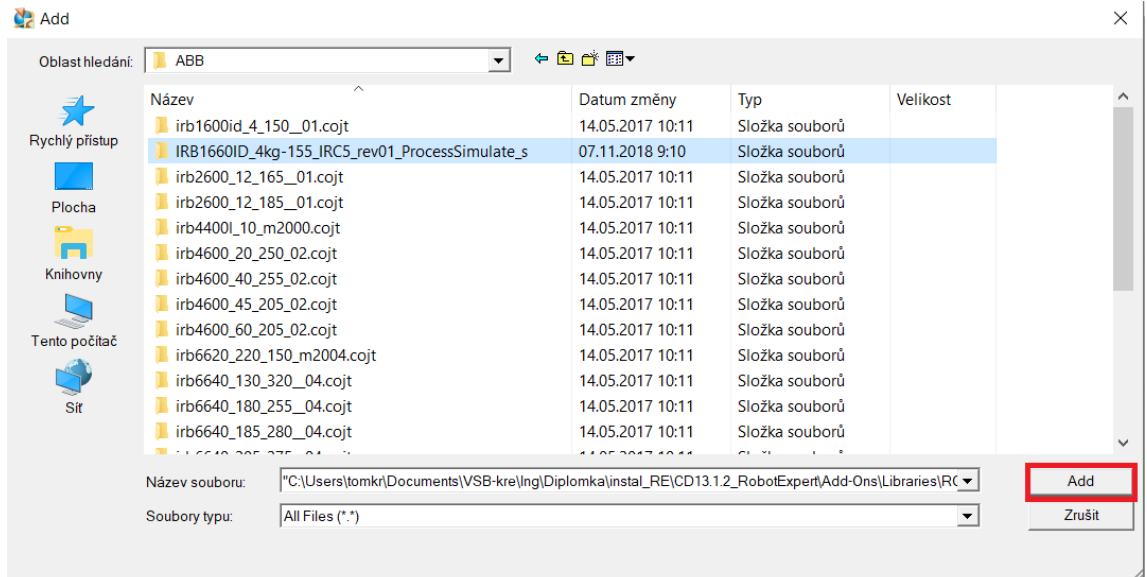
2.5.4 Převod formátu robotů a dalších zařízení

Podporované formáty převodu jsou NX, JT, Solid Edge, Solid Works, Pro/Engineer, CATIA, Parasolid, STEP, IGES a Tecnomatix CO/COJT. Po stažení daného souboru se vraťte zpět do RE a zvolte **File** → **Import/Export** → **Convert CAD Files** (obrázek 2.17).



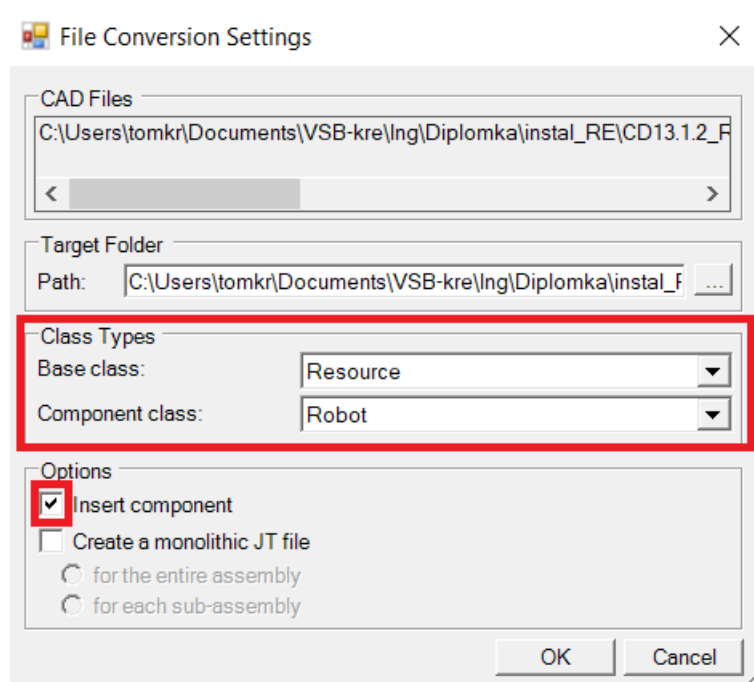
Obrázek 2.17 - Převod formátu

Zvolíme **Add...** otevře se okno, ve kterém zvolíme cestu ke staženému souboru a vložíme opět stisknutím **Add** (obrázek 2.18).



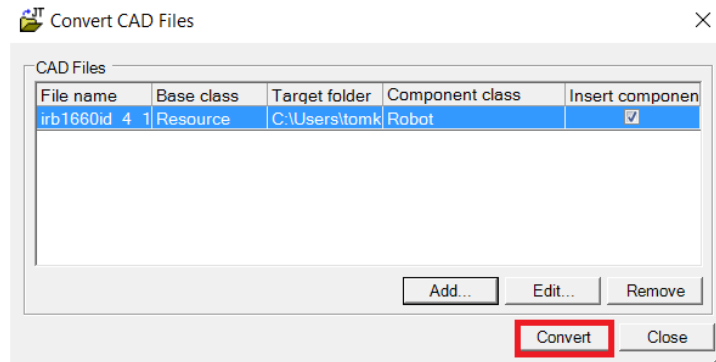
Obrázek 2.18 - Výběr souboru pro převod

Zobrazí se okno **File Conversion Settings** (obrázek 2.19). Zde vybereme **Base class** → **Resource** a jako **Component class** → **Robot**. Nastavíme si cestu, kam se nový soubor uloží (**Target Folder** → **Path**). Vždy zvolte cestu do vaší knihovny, podle toho, jak je nastaven **Library Root**. Chceme-li s modelem ihned pracovat, zaškrtneme políčko **Insert component** a potvrdíme **OK**. Model bude automaticky vložen do nulových souřadnic světového souřadného systému.



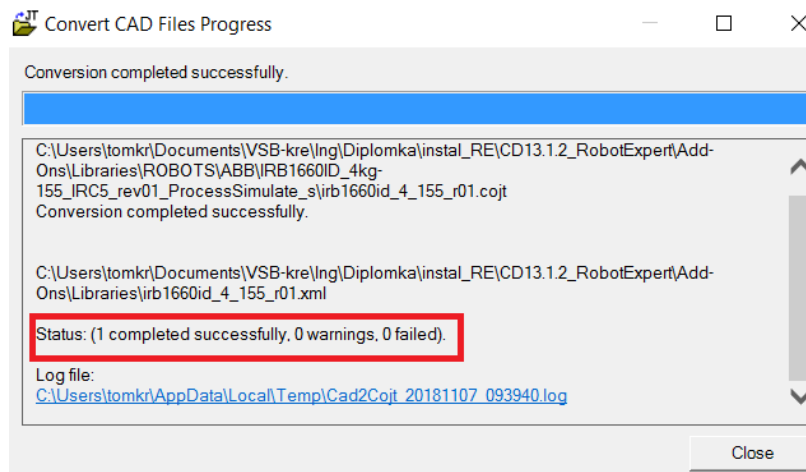
Obrázek 2.19 - Typ převáděného souboru

Lze přidat více komponent a konvertovat formát současně. Zvolte **Convert** (obrázek 2.20).



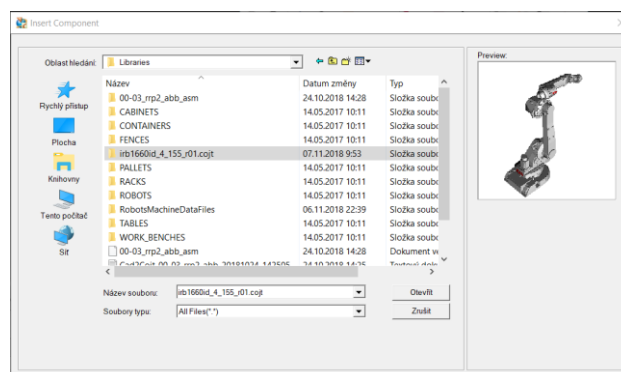
Obrázek 2.20 - Potvrzení převodu

Zobrazí se výpis, ze kterého je zřejmé, jestli bylo konvertování úspěšné. S takto vloženým robotem je možné bez dalšího nastavování hýbat s jeho koncovým bodem. Jednotlivé osy mají omezený rozsah pohybu, daný výrobcem. Rozsah lze libovolně měnit, viz kapitola 3.1.



Obrázek 2.21 - Kontrola úspěšného převodu

Po úspěšném konvertování se ve vámi zvolené složce vytvoří překonvertovaný soubor se správnou příponou. V tomto případě je cesta nastavena na ROBOTS → ABB, odkud ji lze příště používat opakovaně přímo z knihovny **Modeling** → **Components** → **Insert Component** (obrázek 2.22).

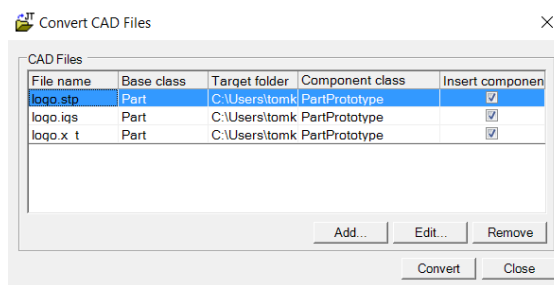


Obrázek 2.22 - Knihovna s roboty

2.5.5 Převod formátu vymoделované součásti / sestavy

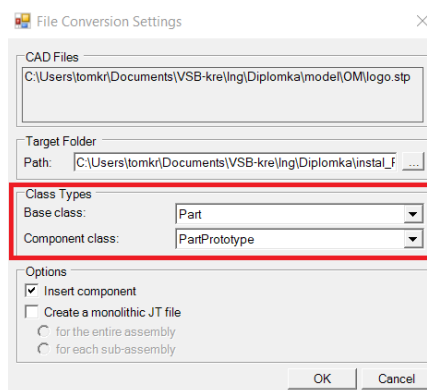
Pro vložení Vámi vytvořeného modelu nebo sestavy je nutné převést model do podporovaného formátu. Takto uložený model lze konvertovat na formát srozumitelný pro RobotExpert (.cojt). Podporované nativní formáty převodu CAD systému jsou NX, JT, Solid Edge, Solid Works, Pro/Engineer, CATIA, a ostatní formáty Parasolid, STEP, IGES a Tecnomatix CO/COJT. Pokud je Váš CAD modelář novější než používaná verze softwaru RobotExpert, nebude převod z nativního formátu fungovat a musíte použít STEP, IGES či Parasolid.

Upozorňuji, že barvy, které jste přiřadili ve Vašem modeláři, zůstanou zachovány i pro RobotExpert pouze při převodu nativního souboru. Pro tzv. mrtvý model – STEP a další ztratí model veškeré barvy a je potřeba mu přiřadit barvy nové přímo v RE, pokud chcete něco víc než jen šedý model pracoviště. Výjimkou je novější STEP AP214, který je schopen zachovat informaci o barvách. Konvertování probíhá stejně jako v předchozím případě **File** → **Import/Export** → **Convert CAD Files** (obrázek 2.23).



Obrázek 2.23 - Převod CAD modelů

Tentokrát je potřeba zvolit **Base class** → **Part** a **Component class** → **PartPrototype** (obrázek 2.24).



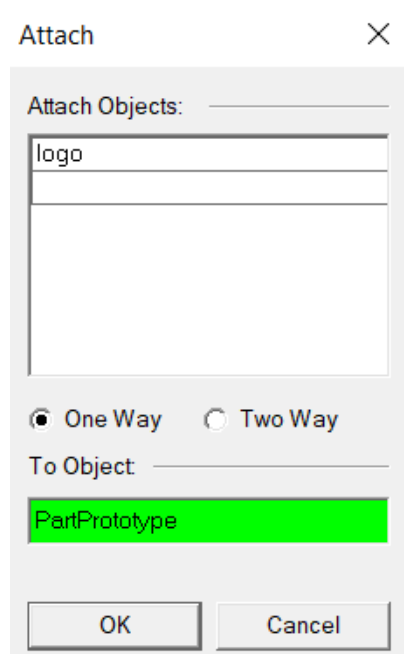
Obrázek 2.24 - Definice součásti

Pokud vkládáte složité sestavy, které by mohli být svojí velikostí náročné pro software, je doporučeno nahrávat soubor s označeným políčkem **Create a monolithic JT file** (obrázek 2.25). Takto nahraná sestava, bude vložena jako jedna součást (u volby **for the entire assembly**) a ve stromu objektů nebude rozepsaný každý díl zvlášť. Případně lze zobrazit pouze podsestavy zvolením **for each sub-assembly**.

2.5.6 Funkce Attach (připojení k rodičovskému objektu)

Propojení dvou součástí pomocí funkce **Attach**, kterou naleznete v záložce **Home** → **Tools** → **Attachment**, má za následek to, že pokud budete s takto propojenými součástmi hýbat, jejich poloha vůči sobě zůstane zachována, respektive budou se hýbat spolu. Funkce **Detach** naopak závislost mezi objekty rozpjí.

V okně **Attach Objects** vybíráte, ke kterému objektu se prvek připojí v okno **To Object** volíte rodičovskou součást, ke které bude prvek připojen. Pokud necháte zaškrtnuté okénko **One Way**, bude ono propojení fungovat jen jedním směrem, tzn., že se budou objekty pohybovat spolu, pouze pokud bude hýbáno s rodičovskou součástí. Při zaškrtnutí **Two Way**, tedy závislost v obou směrech, je jedno, který prvek bude přesouván, vždy se pohybují spolu (obrázek 2.28).

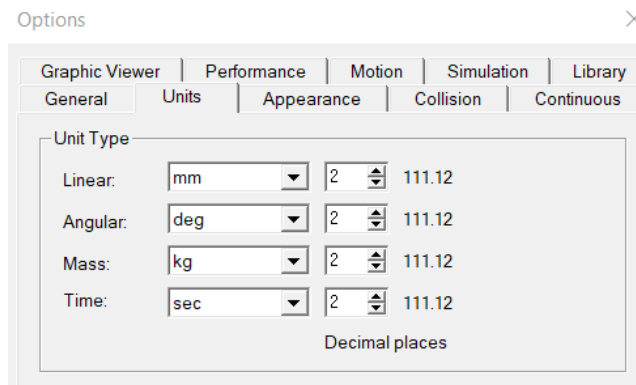


Obrázek 2.28 - Funkce Attach



2.6 Modelování

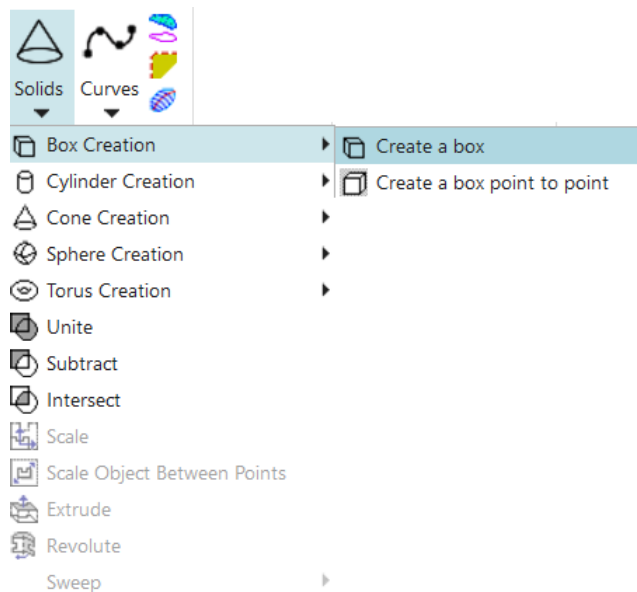
Než začnete modelovat, ujistěte se, že máte nastavené správné jednotky (obrázek 2.29).

File → **Options** → **Units**



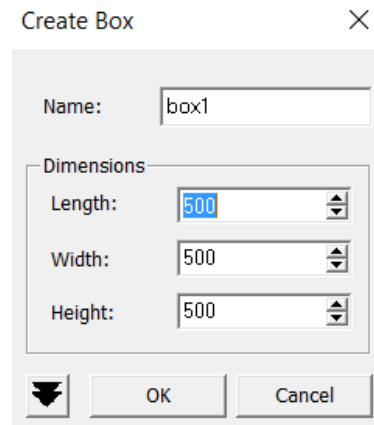
Obrázek 2.29 - Nastavení jednotek

Vyberte **Modeling** → **Components** → **Create New Part**  (případně **New Resource** ). Poté v části **Geometry** → **Solids** si vyberte požadovaný tvar (obrázek 2.30). Jako příklad nyní vytvoříme krychli. Zvolte **Create a box**.





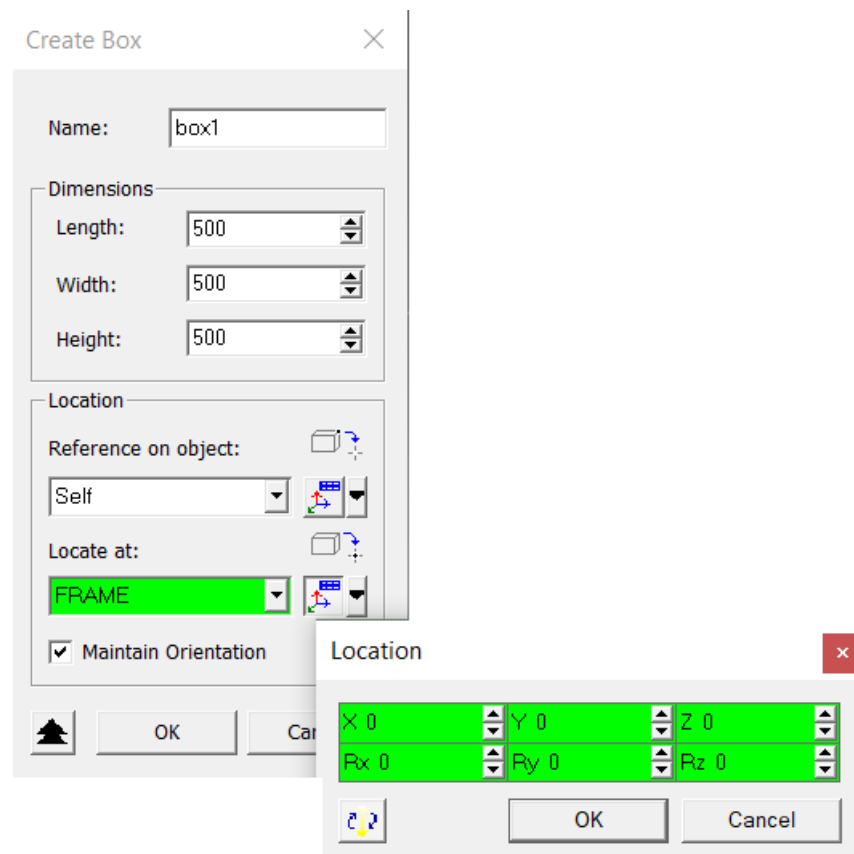
Obrázek 2.30 - Modelování součástí

V dalším kroku jsme vyzváni k zadání rozměrů a pojmenování součásti (obrázek 2.31).



Obrázek 2.31 - Tvorba krychle


Můžeme potvrdit **OK** nebo rozkliknout šipku  a nadefinovat umístění. Po zvolení **Create Frame of Reference**  budete vyzváni k zadání souřadnic (obrázek 2.32).



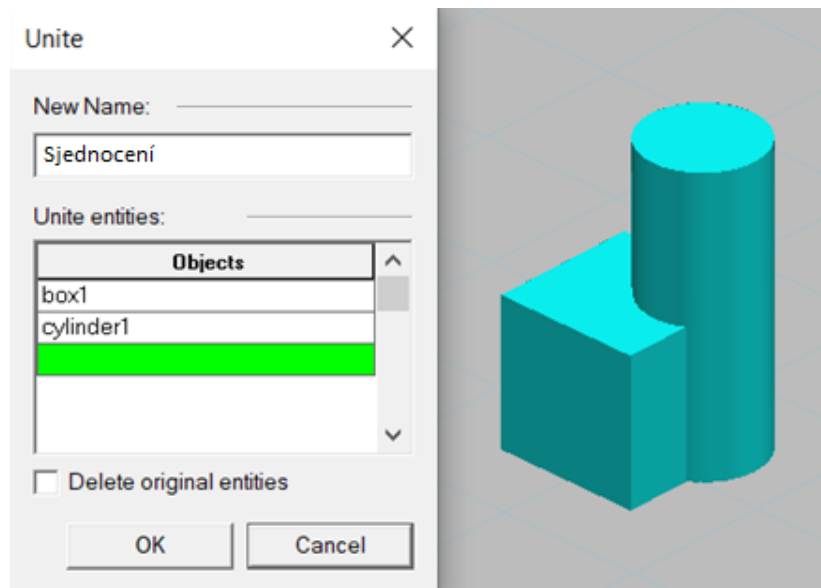
Obrázek 2.32 - Umístění vytvořeného modelu

V položce **Solids** lze také tělesa slučovat, vytvořit jejich průniky a rozdíly.

2.6.1 Unite (Sjednocení)

Modeling → **Geometry** → **Solids** → **Unite**  **Unite**

Pro názornost jsem vytvořil válec a umístil ho tak, aby procházel krychlí. Pro sjednocení součástí stačí pouze označit jednotlivé geometrické součásti a potvrdit OK. Pokud nezaškrtneme okénko **Delete original entities**, vytvoří se kopie již sjednocené součásti ve stejných souřadnicích jako originální entity. Všimněte si, že po vytvoření se v objektu stromu objevil sjednocený díl s názvem **Sjednocení** (název, který jsme součásti přiřadili), viz obrázek 2.33.

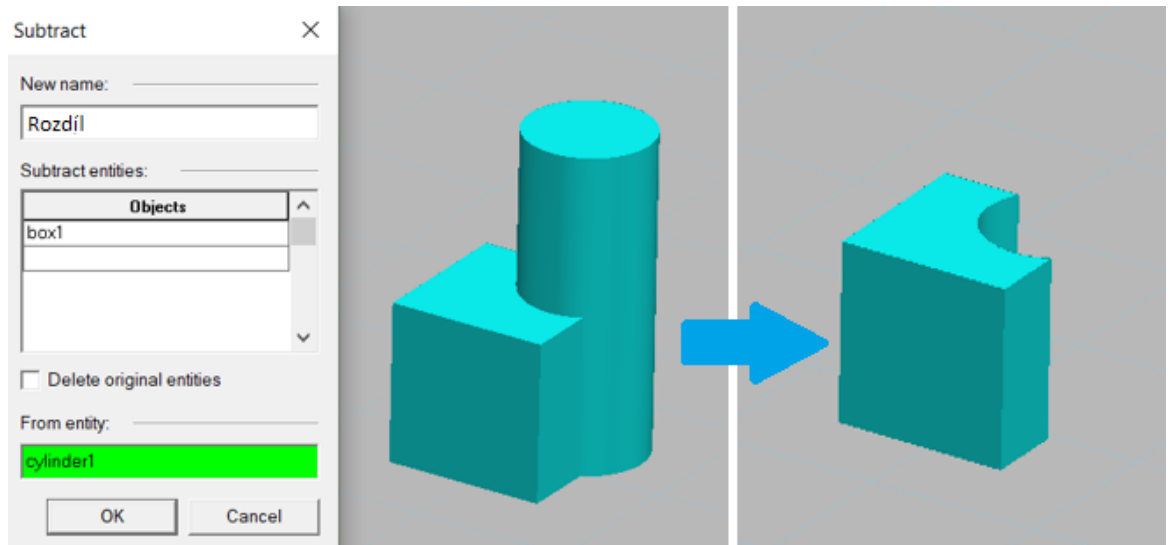


Obrázek 2.33 - Sjednocení

2.6.2 Subtract (Rozdíl)

Modeling → Geometry → Solids → Subtract  Subtract

Nejdříve v okně **Subtract entities** vybereme **objekty, které se odebírají**. Potom v okně **From entity** se vybírá **objekt, ze kterého se odebere** (obrázek 2.34).

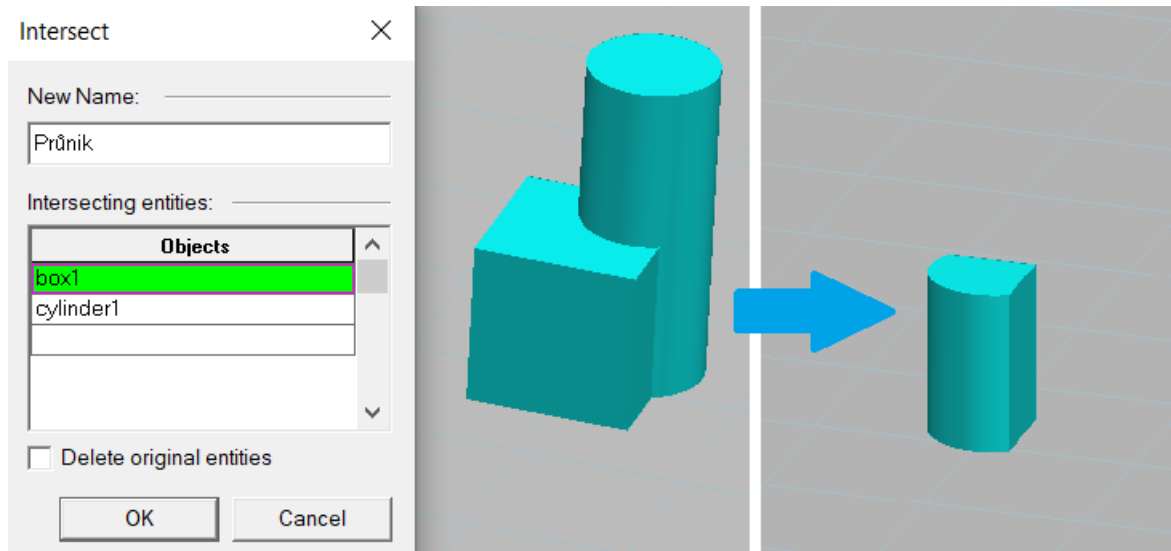


Obrázek 2.34 - Rozdíl

2.6.3 Intersect (Průnik)

Modeling → Geometry → Solids → Intersect  Intersect

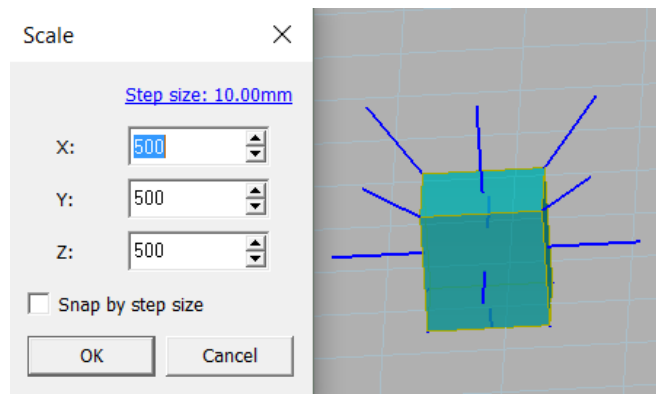
Vybírají se pouze objekty, které provedou průnik (obrázek 2.35).



Obrázek 2.35 - Průnik

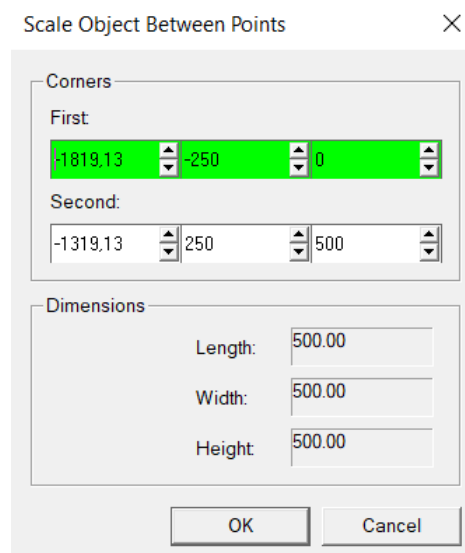
2.6.4 Změna rozměrů objektu

Lze provést po označení objektu v záložce **Modeling** → **Geometry** → **Solids** → **Scale** (případně **Scale Object Between Points**). U možnosti **Scale** stačí přepsat rozměry v daných osách. Taktéž je možné rozměry měnit ručně tažením modrých čar na objektu (obrázek 2.36).



Obrázek 2.36 - Měřítko součásti

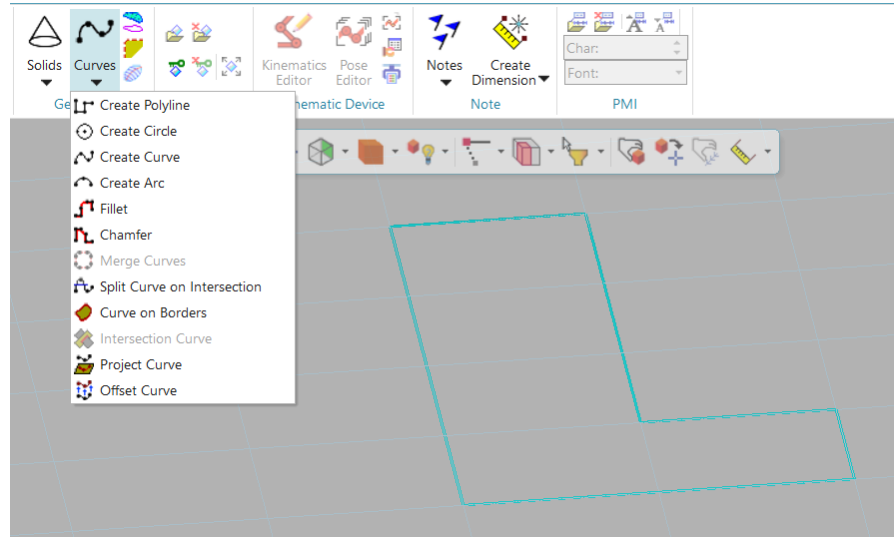
U možnosti **Scale Object Between Points** se zadávají souřadnice protějších rohů (obrázek 2.37).



Obrázek 2.37 - Měřítko z bodu do bodu

2.6.5 Tvorba vlastní skici

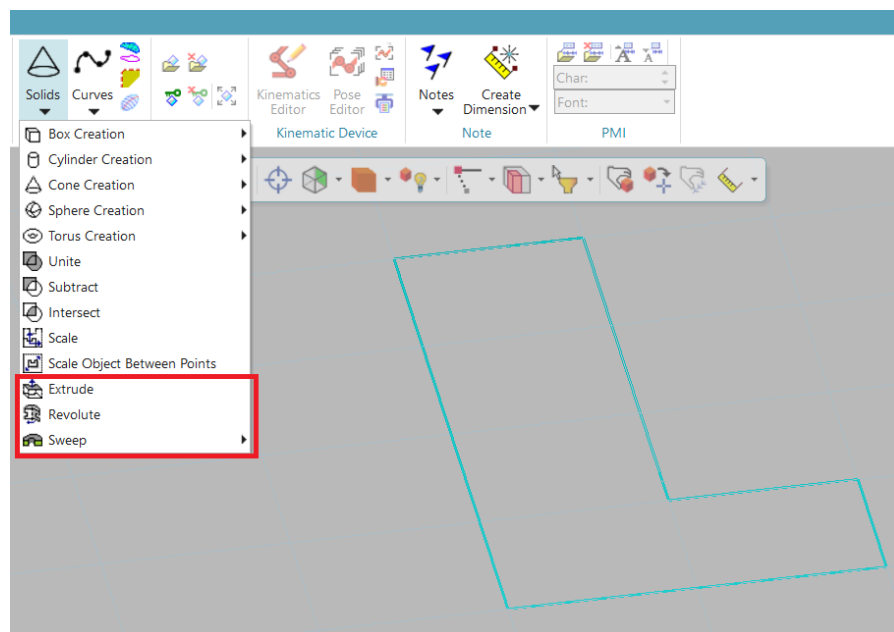
V případě, že typické tvary v záložce Solids našim potřebám nestačí, je zde možnost vytvořit vlastní skicu. Nabídka **Modeling** → **Geometry** → **Curves** obsahuje vše, na co jste zvyklí z CAD modelářů, včetně sražení hran, tvorba rádiusů, apod. Celý proces tvorby skici je celkem intuitivní a není potřeba dalšího vysvětlování (obrázek 2.38).



Obrázek 2.38 - Tvorba skici


2.6.6 Vytažení skici

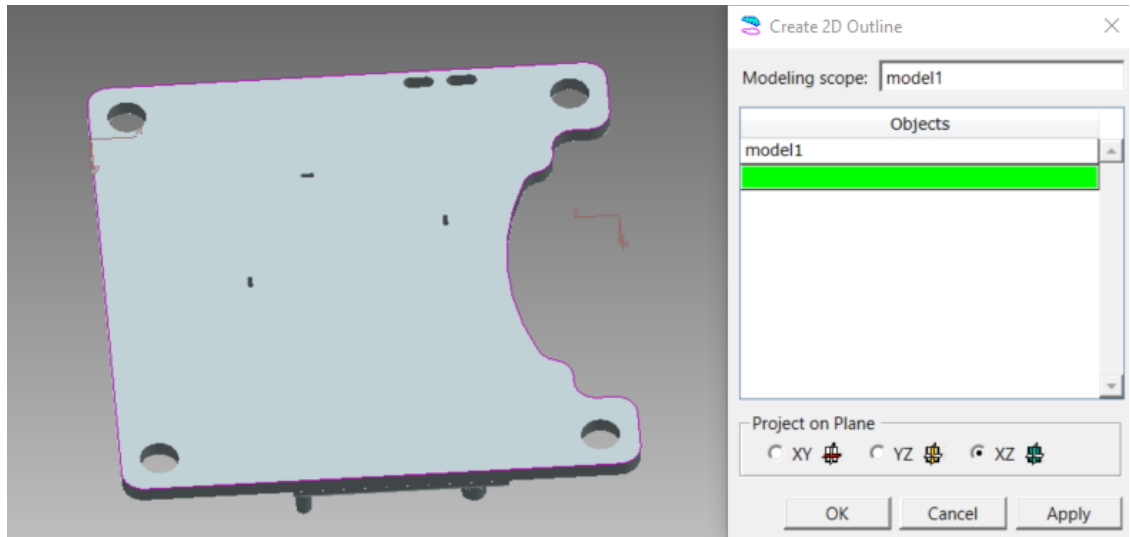
Provede se v záložce **Modeling** → **Geometry** → **Solids**, kde máme na výběr Vytažení (**Extrude**), rotaci (**Revolute**) nebo vytažení profilu po skice (**Sweep**). Viz obrázek 2.39.



Obrázek 2.39 - Vytažení skici

2.6.7 Tvorba 2D kontury z modelu

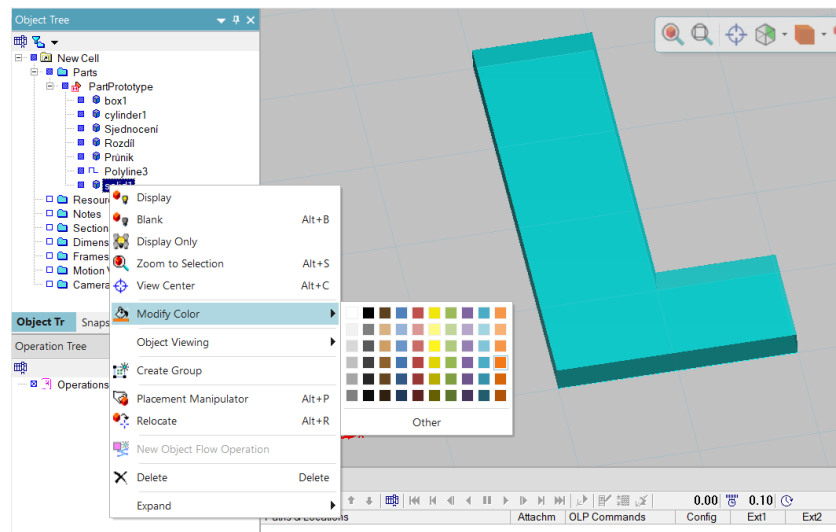
Konturu lze vytvořit pomocí **Modeling** → **Geometry** → **Create 2D Outline** . Aby byl nástroj dostupný, je nutné požadovaný prvek uvolnit k úpravám přes **Set Modeling Scope**. V okně vyberte model a rovinu, ve které se 2D obrys promítne (obrázek 2.40). Po potvrzení **Apply**, se vytvoří fialový obrys a nový prvek ve stromu objektů se kterým lze dále pracovat.



Obrázek 2.40 - Skica z kontury

2.6.8 Změna barvy objektu


Změnu barvy provedeme kliknutím **pravým tlačítkem myši** na vytvořený objekt ve stromu modelů a vybráním kolony **Modify Color**. Zde už zbývá jen vybrat požadovanou barvu (obrázek 2.41).

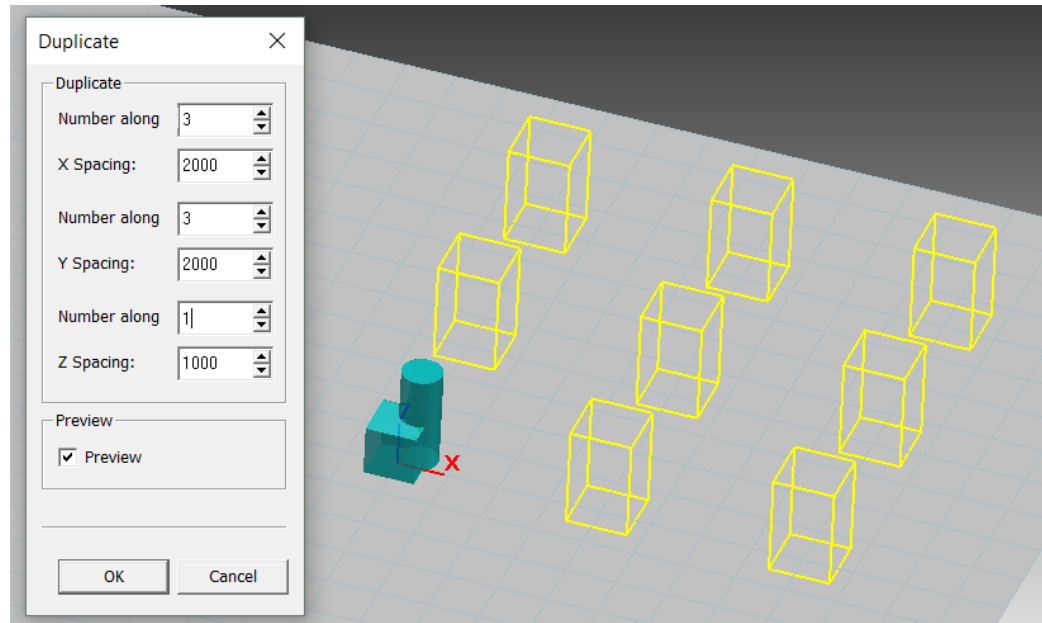


Obrázek 2.41 - Změna barvy součásti

2.6.9 Zkopírování objektů

Modeling → Layout → Duplicate Objects



Označte součást pro kopírování a klikněte na ikonu **Duplicate Objects** . Zobrazí se tabulka, ve které je potřeba nastavit kolikrát bude objekt v dané ose vložen (Number along). Pro každou osu se také musí nastavit velikost mezery, mezi jednotlivými objekty (Spacing). Při zaškrtnutém okénku **Preview** se bude zobrazovat aktuální náhled rozložení (obrázek 2.42).

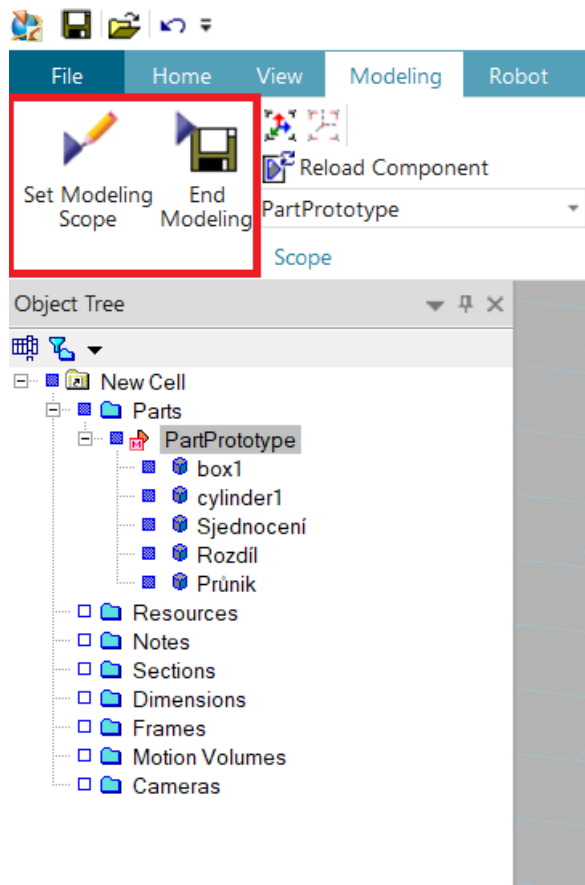


Obrázek 2.42 - Duplikace objektů

2.6.10 Ukončení modelování

Modeling → Scope → End Modeling


Aby ikony **End Modeling** a **Set Modeling Scope** nebyly zašedlé, je potřeba ve stromu objektů označit **PartPrototype** (obrázek 2.43). Jakmile dokončíte modelování, stisknete ikonu **End Modeling**, dojde k ukončení úprav a objekt se uloží do knihovny. Všimněte si, že ve stromu objektů se změní ikona z  na . Případné úpravy objektů, jsou možné po stisknutí **Set Modeling Scope**. Pokud chcete vrátit změny do původního stavu, zvolte **Reload Component** (nelze udělat po uložení do knihovny, tzn. po zvolení End Modeling). Tato funkce nemá vliv na změnu názvu a přiřazené operace, vrátí pouze původní 3D geometrii.




Obrázek 2.43 - Úpravy objektu


2.7 Umístění součástí

Funkce **Placement Manipulator** může být používána současně s funkcí **Relocate** pro stejný objekt.


Funkce **Restore Object Initial Position** , kterou naleznete v záložce **Modeling** → **Layout** lze použít pro navrácení polohy objektu do původní pozice vztažené k jeho rodičovskému objektu, když byla pracovní buňka nahrazena (případně uložena).

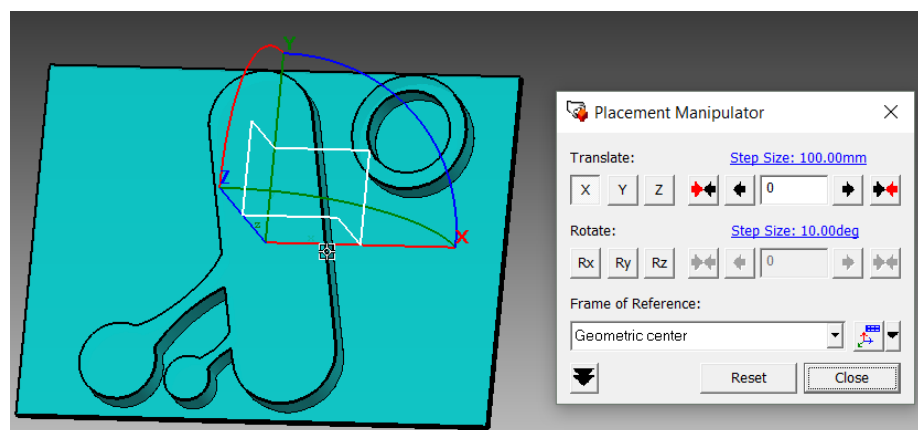
Existují tři možnosti, jak měnit pozici jednotlivých součástí v pracovní buňce.

1. **Fast Placement**  nachází se v záložce **Modeling** → **Layout**. Slouží pro přesun objektů pouze v osách X a Y tažením myši. Je možné používat ho současně pro více vybraných objektů.




2. **Placement Manipulator**  nachází se na pracovní liště (viz kapitola 2.2), případně je dostupný pod klávesovou zkratkou ALT+P.

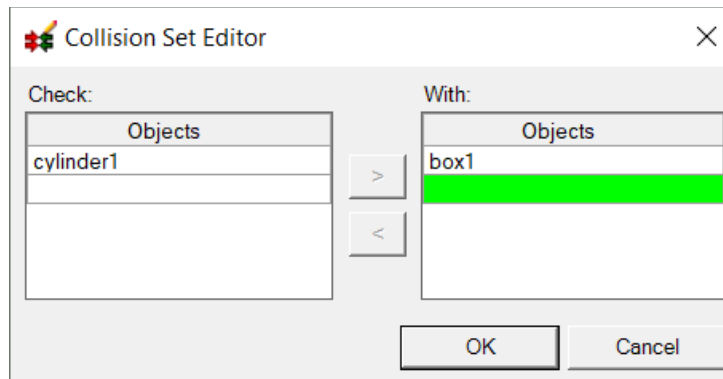
Přemisťování tímto nástrojem je možné tažením myši jednotlivých os (X-červená, Y-zelená, Z-modrá) a jejich rotací (Rx,Ry,Rz) nebo rovin (vyznačeny bíle). Viz obrázek 2.44.

Další možností je používat šipky , kterými se objekt posune o zvolený krok (v tomto případě 100mm) v kladném nebo záporném směru. Pro přesný posun lze zadávat konkrétní hodnotu o jakou se má součást ve vybrané ose posunout nebo rotovat. Objekt se bude posouvat vůči referenčnímu souřadnému systému → **Frame of Reference**.




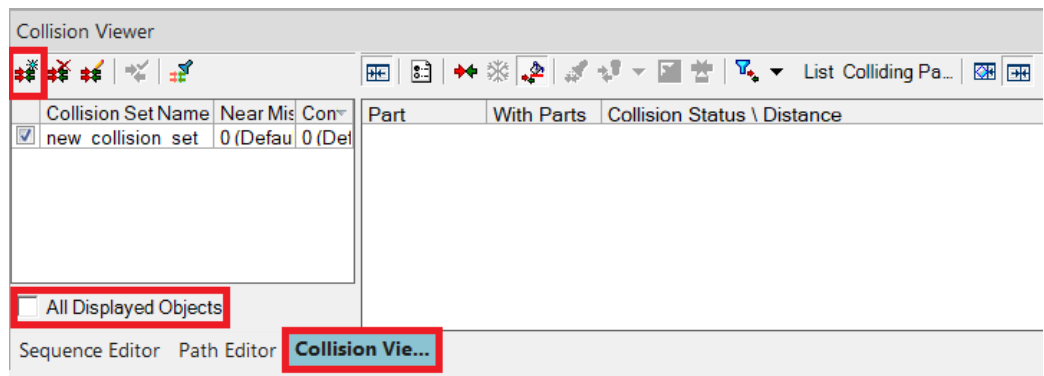
Obrázek 2.44 - Placement Manipulator

Pokud zvolíte šipky **Move Negative To Collision**  nebo **Move Positive To Collision**  objekt se bude pohybovat ve zvoleném směru, dokud není detekována kolize s dalším objektem. Detekci mezi jednotlivými objekty je nutné nejdříve nakonfigurovat. Konfiguraci provede pomocí okna **Collision Viewer**, který se defaultně nachází ve spodní části obrazovky. Zde zvolte **New Collision Set**  pro nastavení kolize mezi zvolenými objekty (obrázek 2.45).




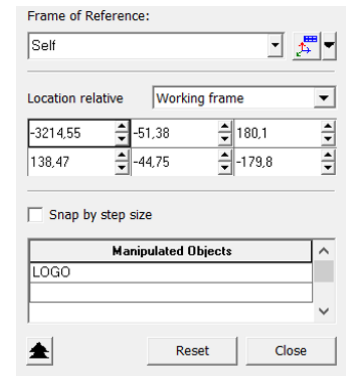
Obrázek 2.45 - Sledování kolize mezi objekty

Pokud chcete detekovat všechny objekty v sestavě, zaškrtněte políčko **All Displayed Objects** (obrázek 2.46). Poté, už můžete používat šipky pro pohyb do kolize  v okně **Placement Manipulator**. Objekty však na sebe nedolehnut úplně, ale zůstává mezi nimi bezpečná vzdálenost. Tato vzdálenost je ovlivněna nastavenou velikostí kroku. Více o detekci kolize naleznete v kapitole 4.3.




Obrázek 2.46 – Nastavení kolizní dovjice

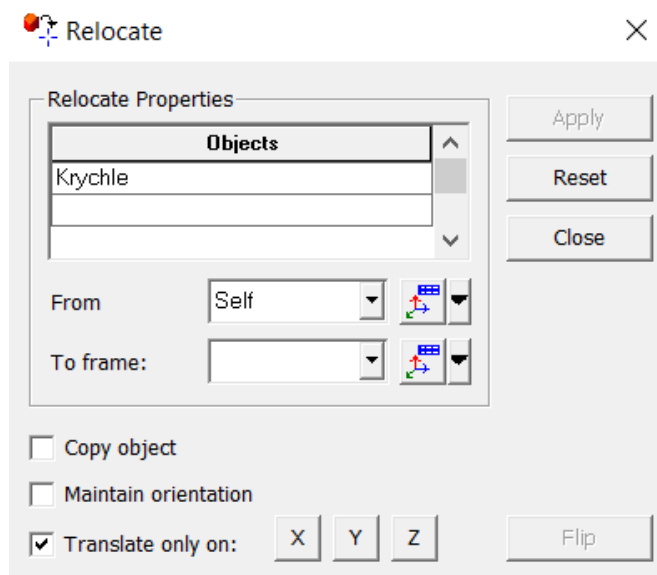
Po rozbalení šipky  v okně **Placement Manipulator** (obrázek 2.47) se zobrazí další možnosti. Zobrazí se poloha pozice vztažená k vybranému souřadnému systému. Při zaškrtnutém políčku **Snap by step size** se při tažení osy pomocí myši objekt přesouvá skokově o nastavený krok.



Obrázek 2.47 – PM

3. **Relocate**  nachází se na pracovní liště (viz kapitola 2.2), případně je dostupný pod klávesovou zkratkou ALT+R.

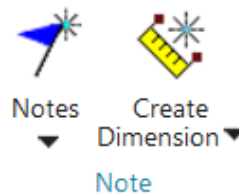
Pomocí tohoto nástroje můžete **přesouvat objekty mezi dvěma souřadnými systémy**. Stačí zvolit ze kterého souřadného systému (dále SS) **From frame** se má objekt přesunout do následujícího souřadného systému **To frame**. Uvědomte si však, že orientace os původního a požadovaného souřadného systému sjednotí svoji orientaci, takže se původní objekt přetočí do orientace požadovaného SS. Chcete-li přesunout objekt pouze v jedné nebo dvou osách, zaškrtněte políčko **Translate only on** a zvolte požadované osy. Pokud však potřebujete zachovat orientaci původního SS, zaškrtněte políčko **Maintain orientation**. Poslední možností je zkopírovat objekt s původního SS do požadovaného, to provedeme zaškrtnutím políčka **Copy object**. Po provedení kteréhokoliv příkazu zvolíme **Apply**, pro navrácení do předchozího stavu zvolíme **Reset**. Viz obrázek 2.48.



Obrázek 2.48 – Relocate

2.8 Tvorba poznámek, označení, rozměrů a obrázků

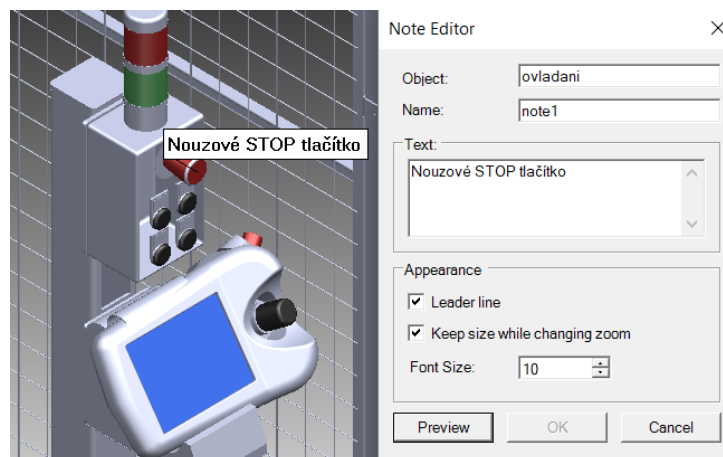
Poznámky i rozměry se nachází ve stromu objektů, kde je lze jednoduše editovat, či zobrazit/skrýt.






Obrázek 2.49 - Poznámky, rozměry

2.8.1 Poznámky

V záložce **Modeling** → **Note** → **Notes** lze přidávat poznámky přímo do obrazu. Zvolte **Create note**, zobrazí se tabulka, kde do okna **Object** přiřadíte objekt, ke kterému se poznámka vztahuje. Napíšete požadovaný text, který se má zobrazovat. Zaškrtnuté políčko **Leader line** znamená, že se bude zobrazovat vodící čára poznámky. Políčko pod ním **Keep size while changing zoom** zachová velikost textu při oddalování nebo přibližování modelu.



Obrázek 2.50 - Ukázka poznámky

- Při rozkliknutí šipky **Notes** zde naleznete tři možnosti pro tvorbu poznámek:
- Create Note**  Vytvoří vlastní poznámku – viz výše.
 - Object Notes**  Defaultně přiřadí poznámku jejíž text je název zvoleného objektu.
 - Location Notes**  Poznámka obsahuje název, pozici a orientaci vybraného bodu objektu vztahující se k pracovní oblasti. Při přemístění nebo přejmenování se automaticky aktualizuje.

Při kliknutí pravým tlačítkem myši na vytvořenou poznámku se zobrazí možnosti:



Zde je možné editovat, skrýt/zobrazit, aktualizovat poznámku a změnit její barvu.

2.8.2 Rozměry

V záložce **Modeling** → **Note** → **Create Dimension** lze přidávat trvalé kóty přímo do obrazu. Při přemísťování objektů se rozměry automaticky aktualizují (to platí i při pohybu těchto objektů během simulace).

Point-to-point Dimension 	➤ Rozměr mezi dvěma body
Minimal Distance Dimension 	➤ Nejkratší vzdálenost mezi prvky
Linear Dimension 	➤ Lineární vzdálenost mezi prvky
Angular Dimension 	➤ Měření úhlu mezi prvky
X-Axis Dimension 	➤ Vzdálenost v ose X mezi objekty
Y-Axis Dimension 	➤ Vzdálenost v ose Y mezi objekty
Z-Axis Dimension 	➤ Vzdálenost v ose Z mezi objekty
Normal to Source Curve Dimension 	➤ Měření vzdálenosti ke zdroji křivky
Normal to Target Curve Dimension 	➤ Měření vzdálenosti k bodu křivky
Curve Length Dimension 	➤ Měření délky křivky

2.8.3 Export obrázků

V záložce **File** → **Import/Export** → **Export Images**  vytvoříte obrázek aktuálního pohledu. Budete vyzváni k zadání názvu a cílové složky pro uložení. Dostupný formát obrázku je bmp, jpg, gif, tif.

2.9 Snapshots a Markups funkce

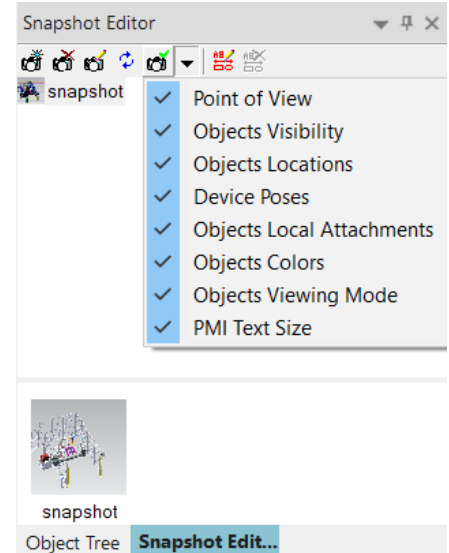
Jedná se o užitečné funkce, zejména při vytváření sestav nebo při jejich budoucích modifikacích. Slouží k ukládání scén a popisků.

2.9.1 Snapshot Editor

View → **Viewers** → **Snapshot Editor**

Funkce snapshots vytvoří snímek aktuální scény. Později lze scénu resetovat do původně zachycené. Toho lze využít například při tvorbě nové verze sestavy. Představte si, že potřebujete odzkoušet nějaké modifikace na sestavě. Nejdříve si vytvoříte snapshot snímek a poté pracujete dále. Pokud zjistíte, že nové změny nejsou ideální a chcete se vrátit ke staré verzi, jednoduše stačí aktivovat uložený snapshot a vytvořená scéna se vrátí do původního stavu. Samozřejmě si můžete ukládat kolik snapshotů chcete.

První čtyři ikony umožňují vytvářet, odstranit, editovat nebo aktualizovat snapshot. Čtvrtou ikonou „Apply Snapshot“ lze aktivovat dříve uložený snapshot. Pod šipkou se nachází nabídka atributů, které se aplikují při aktivaci staršího snapshotu (obrázek 2.51).

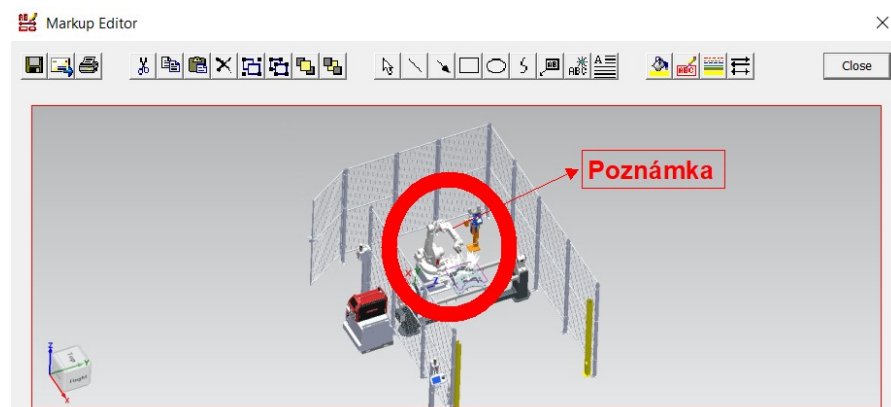


Obrázek 2.51 - Snapshot Editor

2.9.2 Markup Editor

Operation → **Documentation** → **Markup Editor** (lze jej otevřít i přes Snapshot Editor)

Jedná se o nástroj, kterým lze přidávat poznámky, odkazy a různobarevná zvýraznění a následně jej uložit jako obrázek ve formátu .BMP nebo .JPG. Lze ho využívat například pro vytvoření popisu pracoviště nebo tvorbu poznámek pro kolegy, kteří budou přebírat vaši práci. Pokud otevřete Markup Editor přes dříve zmíněný Snapshot editor, je možné se k poznámkám vracet i později pomocí uložených snapshotů.



Obrázek 2.52 - Markup editor

3 Kinematika robotů a dalších zařízení

3.1 Tvorba kinematiky

Pokud potřebujete ve své simulaci používat jakékoliv pohyblivé zařízení, ke kterému máte jen CAD data, musíte vytvořit kinematické vazby a omezení ve kterých se může pohybovat.

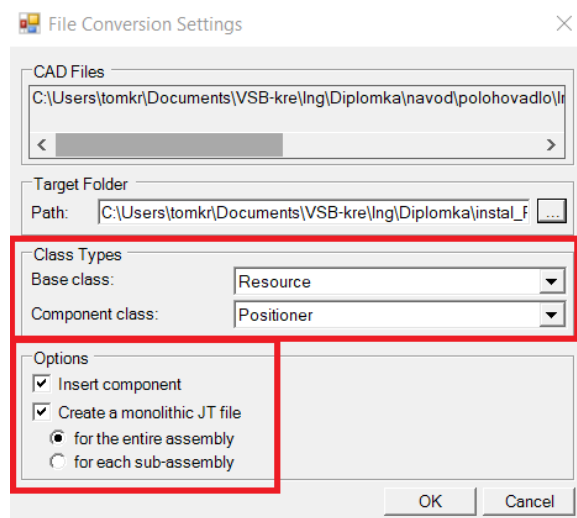
Pro ukázkou, jsem stáhl **STEP** soubor jednoduchého polohovadla, ze stránek firmy ABB. Konkrétně se jedná o typ **IRBP A A250 (ver. D1000)** obrázek 3.1. **Příloha H** obsahuje CAD model polohovadla.

IRBP A
Industrial robot positioner



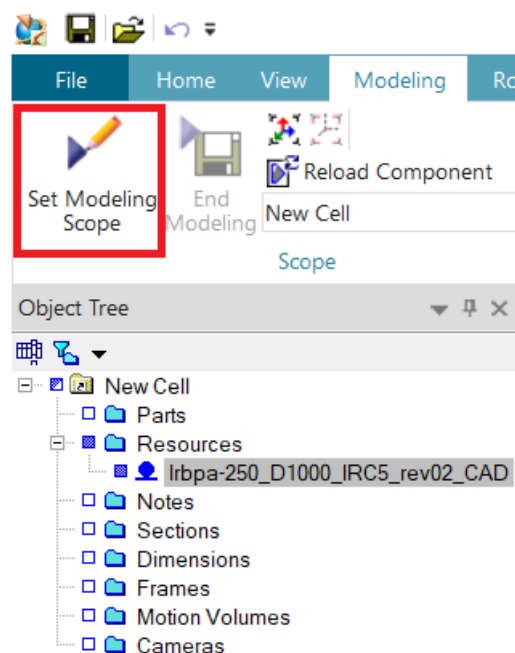
Obrázek 3.1 - Polohovadlo ABB [10]

Nejdříve provedeme převod STEP formátu na COJT. Podrobněji je převod rozepsán v kapitole 2.5.4. Při převodu však nezapomeňte zvolit třídu komponenty jako **Positioner**. Taktéž je vhodné zaškrtnout políčko **Create a monolithic JT file**, ušetří nám to dalších pár kroků. Doporučuji si změnit cestu pro uložení, do Vaší knihovny (**Target Folder**).




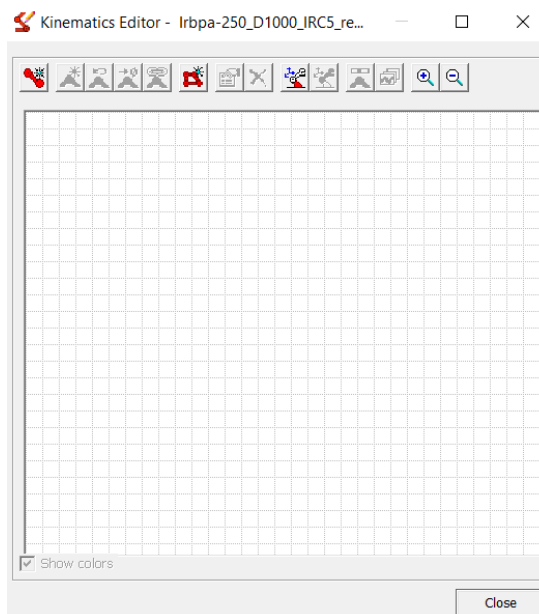
Obrázek 3.2 - Převod formátu polohovadla

Po nahrání **označte polohovadlo ve stromu objektů** a klikněte na **Set Modeling Scope**, čímž uvolníte objekt pro úpravy.



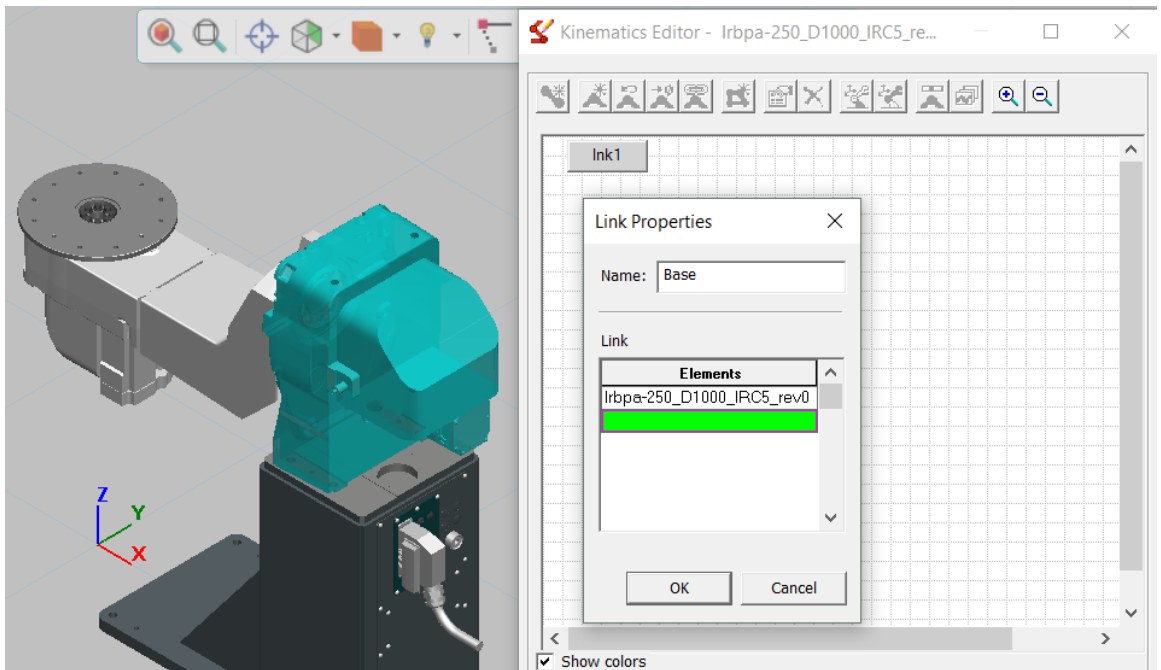
Obrázek 3.3 – Uvolnění polohovadla pro úpravy

V dalším kroku je potřeba taktéž mít označené polohovadlo ve stromu objektů, až poté je možné zvolit **Kinematics Editor (Modeling → Kinematic Device)**. Zobrazí se následující okno (obrázek 3.4), ve kterém zvolíme hned první ikonu  (**Create Link**).



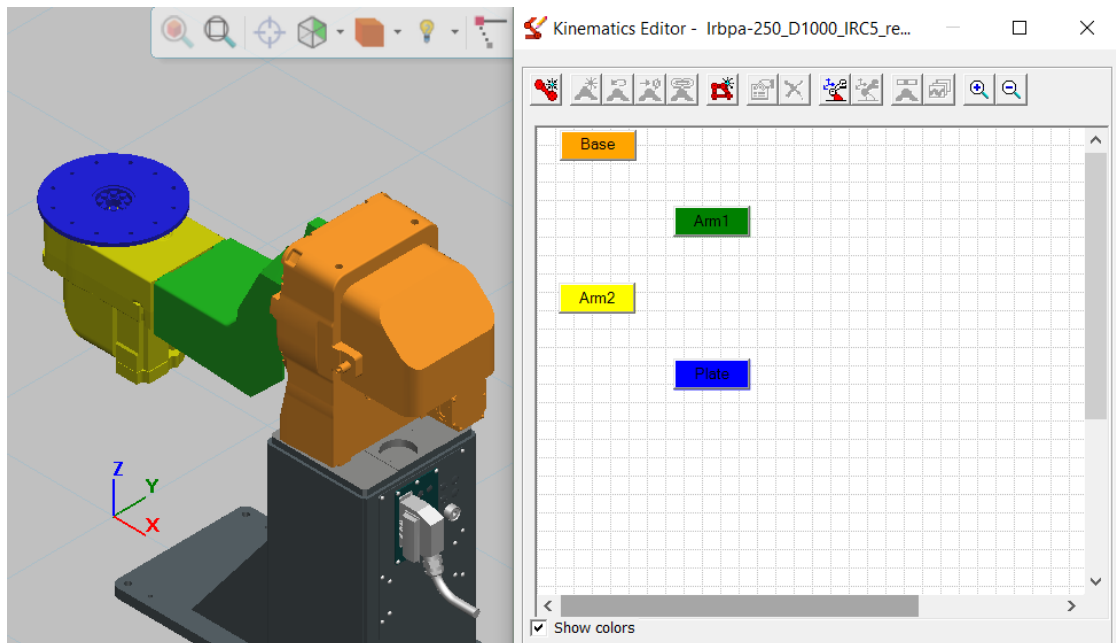
Obrázek 3.4 - Kinematický editor

Link si pojmenujeme jako Base a označíme první část (obrázek 3.5). Potvrdíme OK. Pozor, link nemůže mít stejný název jako součást.




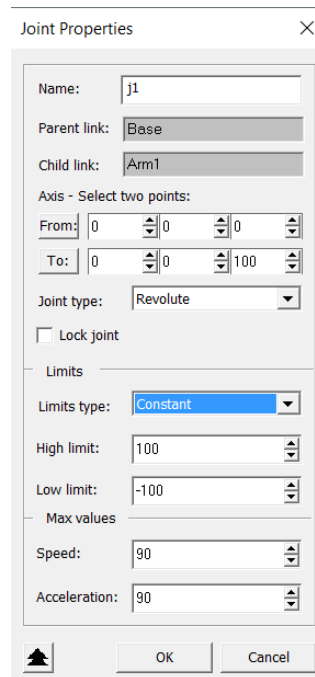
Obrázek 3.5 - Kinematika (Base)

Stejný postup aplikujeme i pro další části. Správně označený díl nebude zašedlý, ale bude mu přiřazena barva (obrázek 3.6). V tomto případě jsou vytvořeny čtyři linky a to **Base**, **Arm1**, **Arm2** a **Plate**.



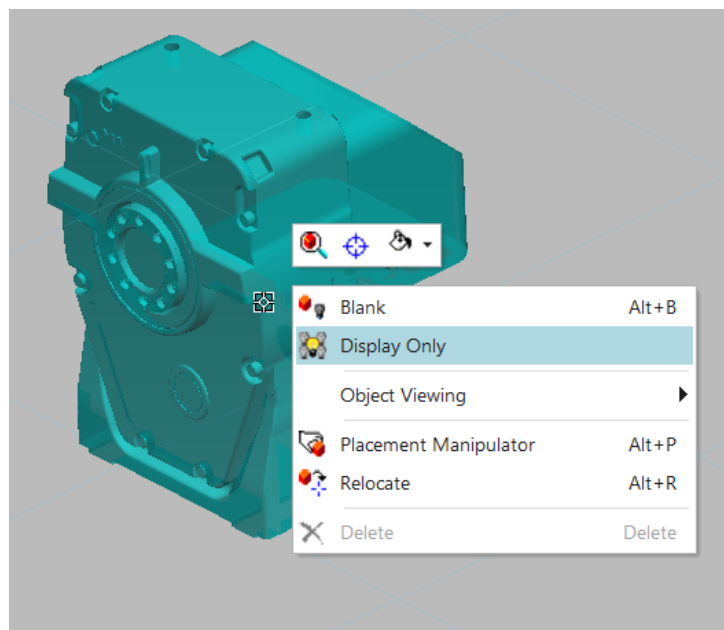
Obrázek 3.6 - Vytvoření Linků

Držíme klávesu CTRL a označíme linky **Base** a **Arm1**. Poté lze zvolit hned druhou ikonu  **Create Joint**, která nám umožní nastavit kinematickou vazbu, mezi zvolenými prvky. V tomto okně (obrázek 3.7) lze nastavit typ vazby, v tomto případě zvolíme rotační **Revolute**. Dále je možné nastavit omezení pohybů a rychlostí.




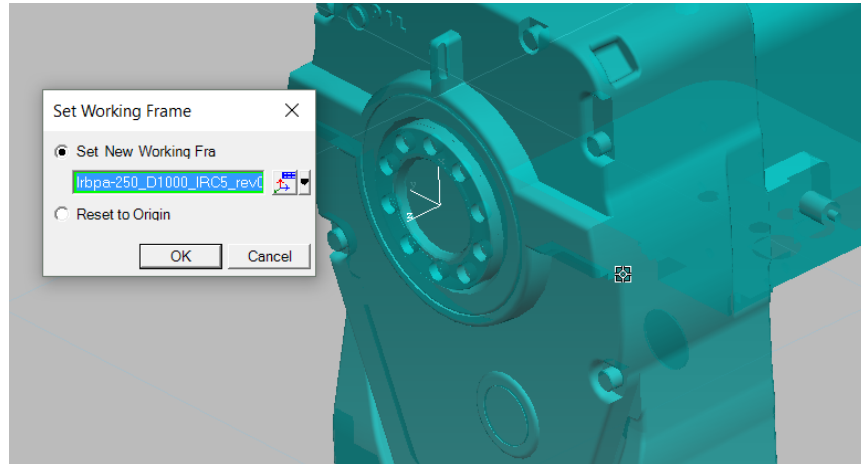
Obrázek 3.7 - Joint Properties

Ještě, než okno potvrdíme, je nutné označit dva body, mezi kterými pohyb probíhá. Pro jednoduchost klikněte pravým tlačítkem na rodičovskou část, kterou jsme si pojmenovali jako Base a vyberte **Display Only**.



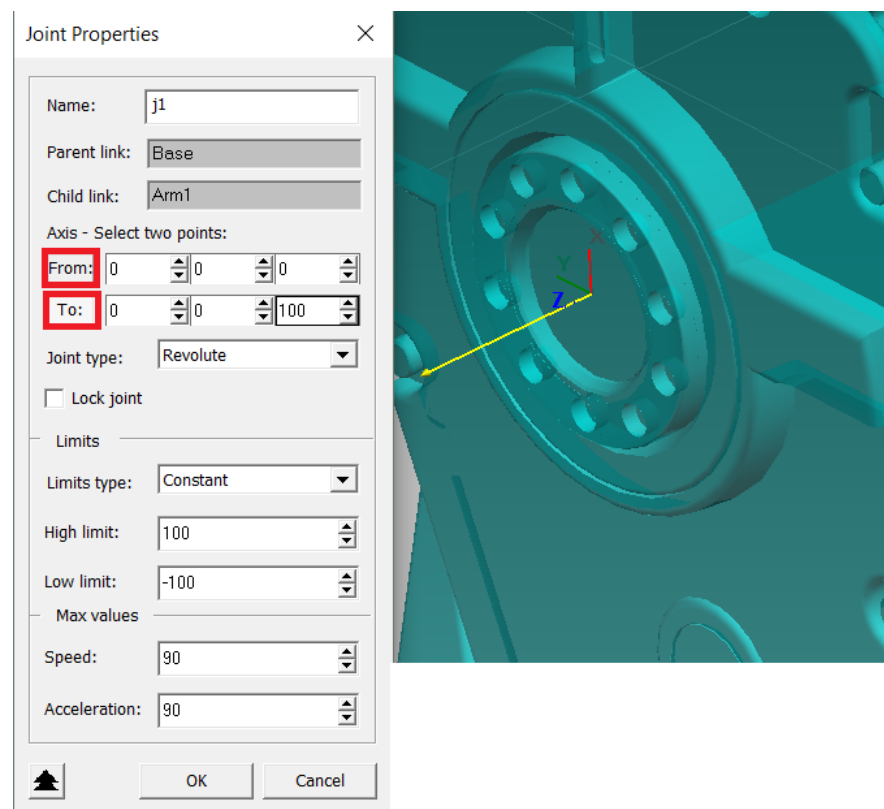
Obrázek 3.8 - Součást Base

Zobrazí se pouze vybraná část, na které nyní vytvoříme souřadný systém pomocí ikony  **Set working frame (Modeling → Scope → Set working frame)**. Zvolíme bod ve středu příruby a potvrdíme (obrázek 3.9).



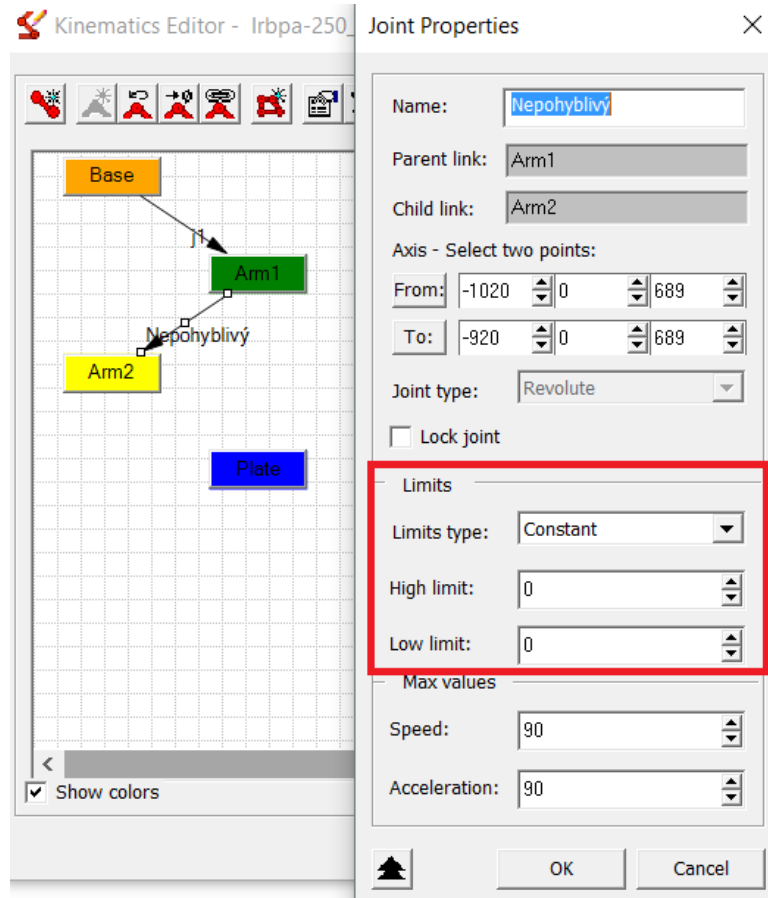
Obrázek 3.9 - Bod ve středu příruby

V nastavení Jointu nejdříve klikneme na **From** a vybereme vytvořený souřadný systém. Potom, uděláme to stejné s druhým řádkem kliknutím na položku **To**. Tím nastavujeme body, mezi kterými bude pohyb probíhat. U rotace je to ten jeden a samý bod. Vyskočí upozornění, že není možné v obou případech vybírat stejný bod, proto osu Z přepíšeme, například na hodnotu 100 (obrázek 3.10). Nyní můžeme potvrdit stisknutím **OK**.



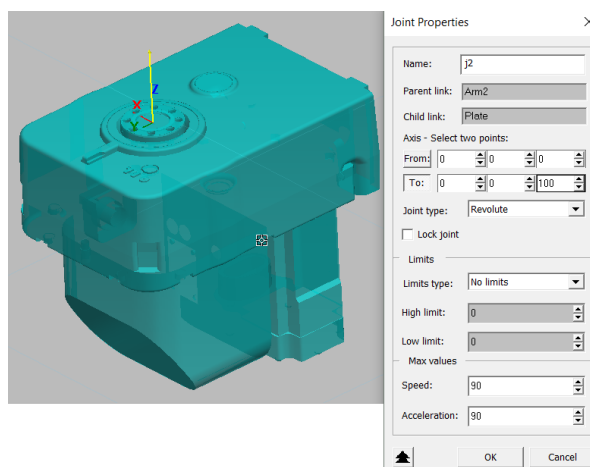
Obrázek 3.10 - Osa pohybu

Nyní vytvoříme stejným způsobem **Joint** mezi **Arm1** a **Arm2**. Tento Joint nastavujeme, protože chceme, aby se Arm2 pohybovalo spolu s Arm1, jelikož je stáhnutý model rozdělen na dvě části. Tentokrát není potřeba volit body pohybu. Důležité je pouze vybrat konstantní omezení pohybu (**Limits Type: Constant**) a jako limity necháme nuly (obrázek 3.11).




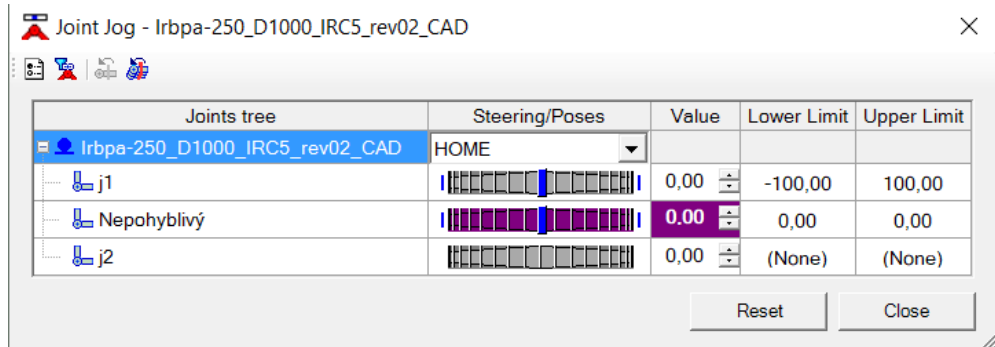
Obrázek 3.11 - Konstantní Joint

Zbývá pouze vytvořit poslední **Joint** mezi **Arm2** a **Plate**. Vytvoříme tedy opět souřadný systém na rodičovské součásti Arm2. Nastavujeme stejným způsobem jako v prvním případě. Tentokrát však nastavíme omezení pohybu bez limitů (**Limits Type: No limits**) obrázek 3.12.



Obrázek 3.12 – Kinematika příruby

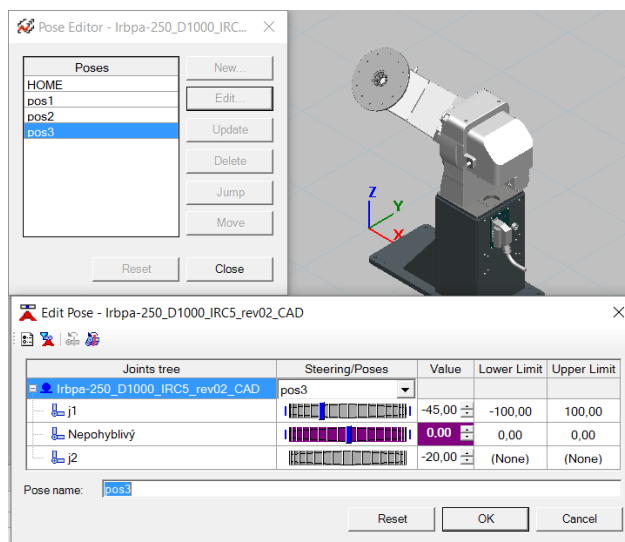
Funkčnost nastavených vazeb ověříme zvolením ikony  **Open Joint Jog**, která se nachází v Kinematicém Editoru. Při tažení posuvníku jednotlivých Jointů se dynamicky mění poloha modelu (obrázek 3.13). Návrat do původní pozice provedeme stisknutím tlačítka **Reset**.



Obrázek 3.13 - Joint Jog

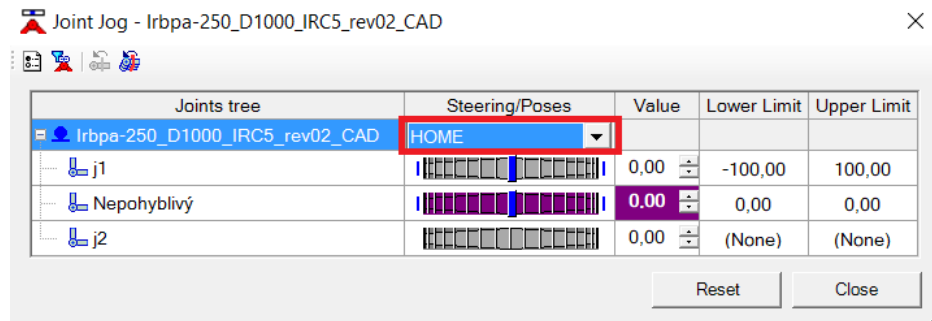
Polohovadlo si uložíme ikonou „ukončením úprav“, která se nachází **Modeling** → **Scope** → **End Modeling**. Cestu pro uložení, jsme si nastavili už při převodu formátu. Můžete se setkat s aplikací, ve které se bude polohovadlo opakovaně vracet do stejné pozice a vypisovat pokaždé stejné souřadnice by bylo zbytečně zdlouhavé. Práci si můžeme ušetřit uložení pozic pomocí **Pose Editor**, který se nachází v záložce **Modeling** → **Kinematic Device** → **Pose Editor**. Opět je nutné mít označené polohovadlo ve stromu objektů.

Otevře se **Pose Editor**. Po pravé straně lze vytvářet nové polohy nebo je editovat, smazat, atd. Přesunout polohovadlo do uložené polohy je možné dvojklikem na název uložené pozice nebo pouhým označením a tlačítkem **Jump** nebo **Move**. Viz obrázek 3.14 .



Obrázek 3.14 - Pozice polohovadla

Pro pohyb polohovadla nemusíme vždy otevírat editor kinematiky. Při sestavování programu budete používat ikonu **Joint Jog**, která se nachází v záložce **Robot** → **Tool and Device** → **Joint Jog**. Otevře se známe okno. Pro volbu uložených pozic lze rozkliknout nabídku, kterou červeně znázorňuje obrázek 3.15.



Obrázek 3.15 - Joint Jog pozice

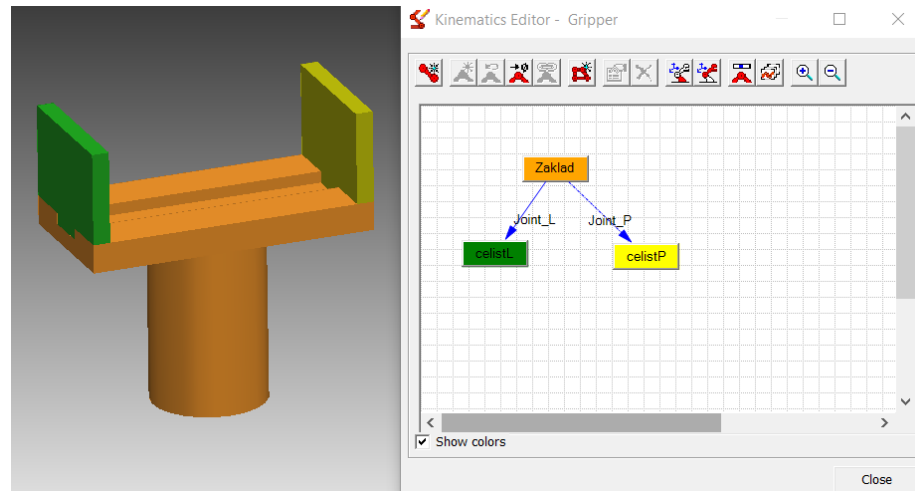
3.1.1 Kinematické závislosti

Někdy je potřeba vytvořit závislost pohybu mezi součástmi mechanismu. Existují tři typy závislostí, se kterými můžeme pracovat. Jsou to:


- **Following** – závislost pro následování pohybu rodičovského jointu.
- **Joint Function** – závislost mezi jointy vyjádřena pomocí logických a matematických funkcí.
- **Coupling** – neboli spojování, může být použito pouze pro složené zařízení (součást obsahující odkazy na další dílčí komponenty). Používá se k definování závislosti jointů z jedné komponenty na druhou.

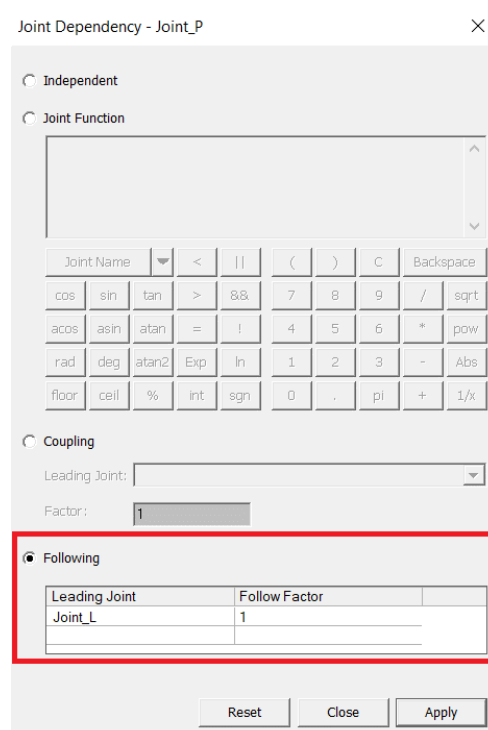
3.1.1.1 Follow – následuj

Závislost mezi jointy využijeme například při tvorbě efektoru pro Pick and Place operace. V prostředí RE jsem si vytvořil zjednodušený nástroj, který tvoří tři části - základ, levá a pravá čelist (obrázek 3.16).



Obrázek 3.16 - Efektor (závislost pohybu)

Jointy vytvořím tak jak jsme si ukazovali již dříve, zadáme typ pohybu, směr, limity, rychlosti. Nyní si označím joint, který bude tvořit závislost k druhému jointu. V tomto případě vybírám Joint_P, který spojuje základ a pravou čelist. Zvolte položku **Joint Dependency** , zobrazí se následující okno (obrázek 3.17).

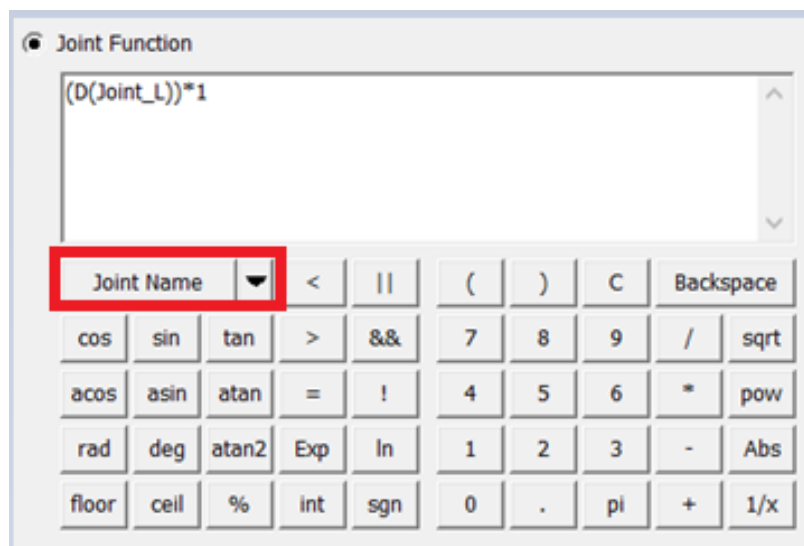


Obrázek 3.17 - Závislost Following

Vybereme závislost typu **Following** (následuj) a přiřadíme **Leading Joint** (tj. joint, který bude Joint_P následovat), v tomto případě Joint_L. Zbývá už jen zadat **Follow Factor**, přiřazení čísla 1 znamená, že se bude pohybovat o stejnou vzdálenost, jako rodičovský joint. Takže při změně na číslo 2, by se Joint_P pohyboval dvakrát rychleji, tudíž by urazil dvakrát větší vzdálenost. Důležité je zadat správné znaménko, protože tím určujeme směr pohybu. Nyní je vše nastaveno a můžeme ověřit správnost pohybu pomocí **Joint Jog**.

3.1.1.2 Joint Function – logicko-matematické vyjádření


Stejný výsledek závislosti, jen pomocí funkcí dostaneme zadáním $(D(\text{Joint_L})) * 1$. To můžeme buď napsat ručně nebo zvolit okénko **Joint Name** a vybrat požadovaný joint. Následně dopsat vynásobení jedničkou (obrázek 3.18). Jednička má v tady stejnou úlohu jako Follow Factor v předchozím případě.



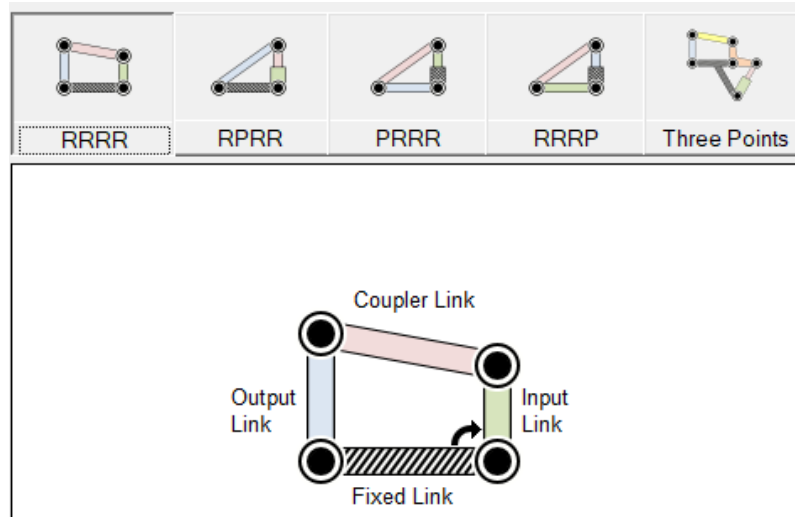
Obrázek 3.18 - Logicko-matematická závislost

Používáme-li tuto funkci, uvidíme v **Joint Jog** pouze lištu s Joint_L, nikoliv oba vytvořené jointy, jak tomu bylo při závislosti typu Follow. **Příloha F** obsahuje další logické funkce a definice.


3.1.2 Kinematické struktury (Paralelogram)

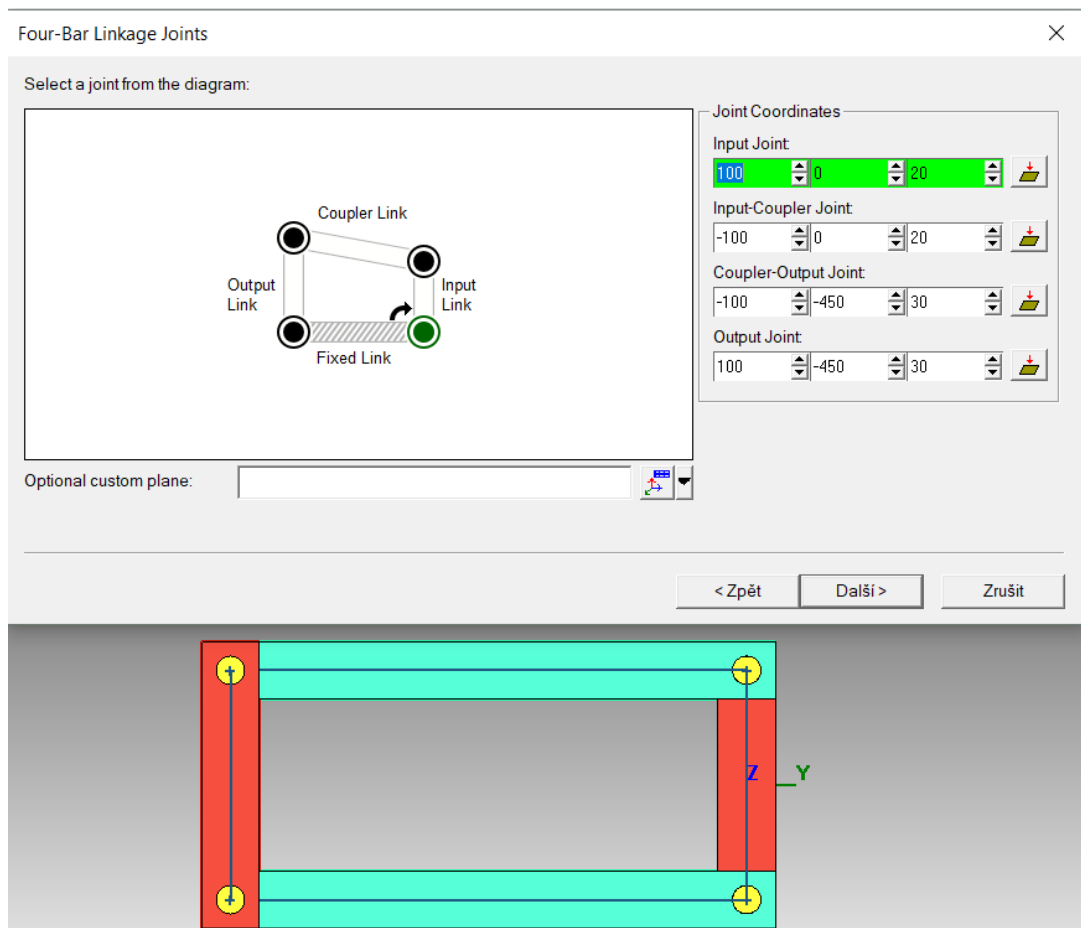
V této kapitole se budeme bavit o funkci **Create Crank** , která se používá k definování kinematických struktur, sestávajících z alespoň jednoho nezávislého jointu a více závislých jointů spojené v kinematické smyčce.

Před použitím musíte mít pouze připravený model, kterému chcete přiřadit kinematiku. Zbývá už jen označit součást ve stromu objektů a zvolit **Modeling** → **Scope** → **Set Modeling Scope**. Tím se nám zpřístupní funkce Create Crank nacházející se v **Modeling** → **Kinematic Device** → **Kinematics Editor** → **Create Crank**. Práce s touto funkcí je velice intuitivní, je to takový průvodce, který nás provede každým kliknutím k definování správného mechanismu. Na začátku jsme vyzváni k vybrání typu našeho mechanismu. Písmena R a P jsou druhy pohybu jointů jdoucí po sobě **R = rotace a P = prismatický**. První mechanismus RRRR je tedy tvořen jointy se čtyřmi rotacemi, je to typický mechanismus paralelogramu, který si zde blíže popíšeme (obrázek 3.19).



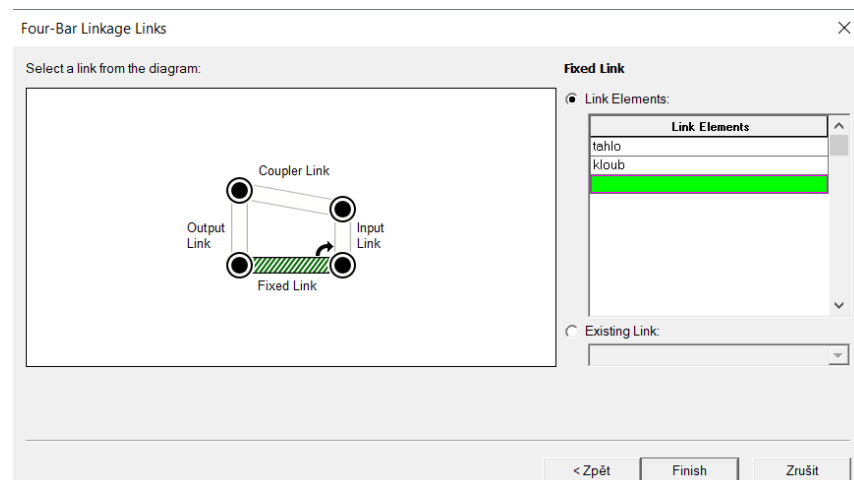
Obrázek 3.19 - Create Crank

Zvolíme **RRRR** mechanismus a tlačítkem **Další** přejdeme na následující krok. Zvolíme osy kloubů, podle obrázku při použití výběru **Snap Pick Intent**  se nám automaticky nabídne výběr středu kloubu, poté co myší najedeme poblíž středu. Pro někoho možná jednodušší způsob bude vytvořit si předem souřadné systémy do středu kloubu (**Modeling** → **Layout** → **Create Frame**) a v tomto kroku je označit (obrázek 3.20).



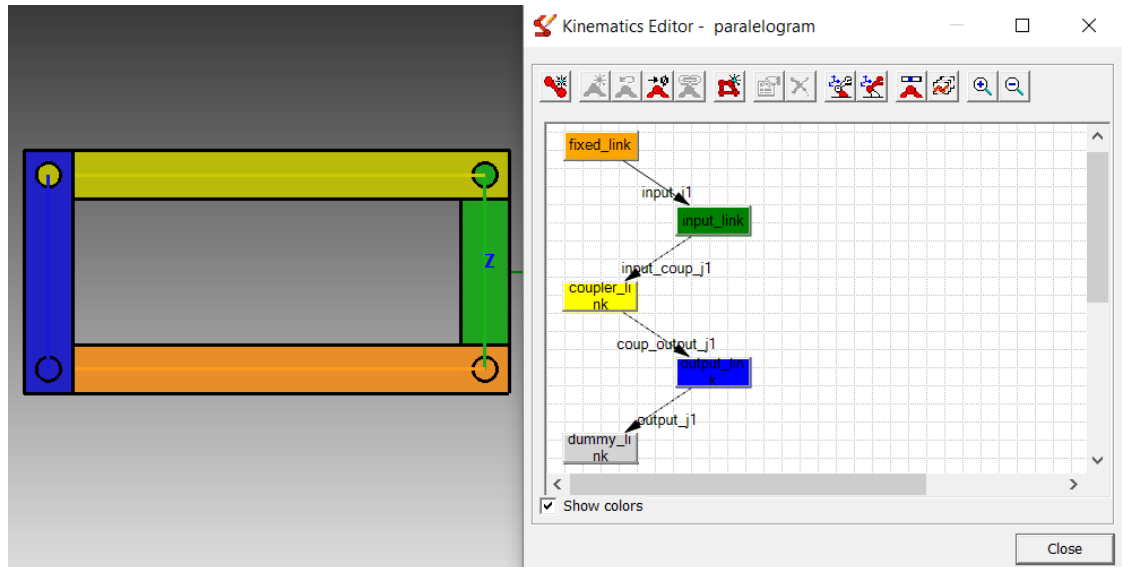
Obrázek 3.20 – Výběr kloubů

Přesuneme se k dalšímu kroku (obrázek 3.21). Zde jsme vyzváni k označení příslušných součástí, které budou tvořit **Fixed Link** (pevná vazba). Vybral jsem spodní táhlo a kloub vpravo dole. V tomto okně ještě zůstaneme a na schématickém obrázku klikneme myší na táhlo s nápisem **Input Link** (vstupní vazba). Přiřadíme mu odpovídající součásti modelu (červené táhlo a kloub vpravo nahoře). Stejně postupujeme i pro definování **Coupler Link** (spojovací vazba) a **Output Link** (výstupní vazba). Jakmile máme hotovo, potvrdíme tlačítkem **Finish**.



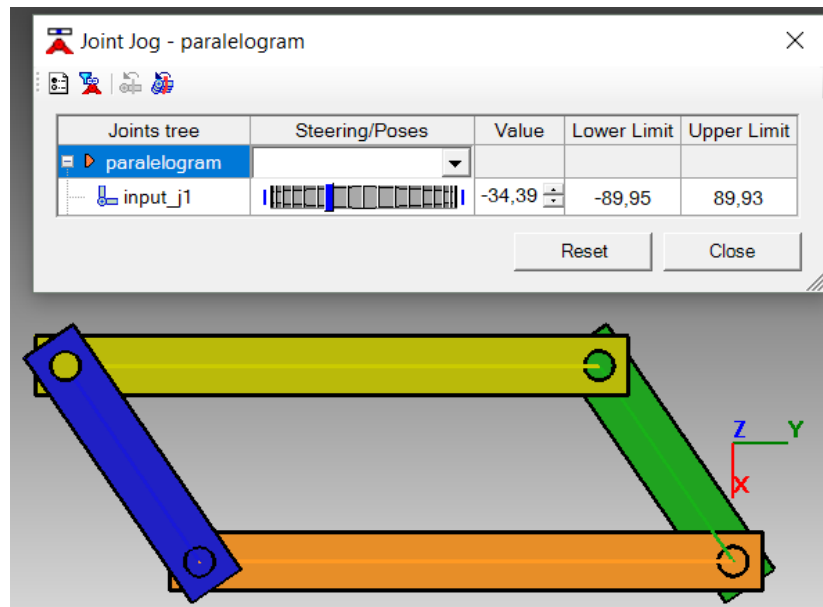
Obrázek 3.21 - Spojení táhel s klouby

V tomto kroku je už definice kinematiky hotova, což vidíme v okně **Kinematics Editor**. Pro zajímavost si můžete jednotlivé jointy a jejich závislost (Joint Dependency) otevřít a podívat se, jak byly automaticky definovány (obrázek 3.22). Všimněte si, že byl vytvořen **dummy_link**, bez přiřazené geometrie. To je fiktivní spojení k dokončení smyčky kinematické struktury (náhrada za pevné spojení).



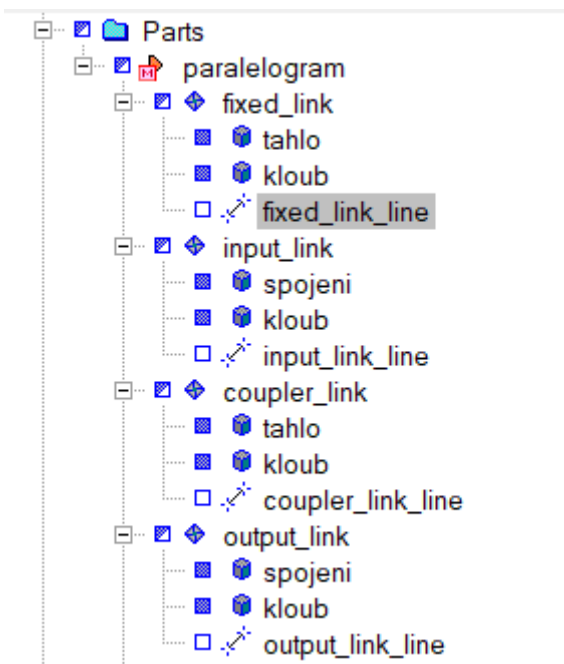
Obrázek 3.22 - Vytvořený paralelogram

Pro kontrolu si ještě otevřete **Joint Jog** a ověřte, zdali je pohyb správný (obrázek 3.23). Pokud vám automaticky vytvořené **limity** nevyhovují, můžete je upravit v **Kinematics Editor** změnou prvního jointu s názvem **input_J1**, ostatní jointy nemají přiřazené limity, protože jsou závislé na jointu prvním. Stejným způsobem změňte i jejich **rychlost**.



Obrázek 3.23 - Pohyb paralelogramu

Všimněte si, že v modelu zůstali vidět čáry propojující klouby. Jejich skrytí jednoduše provedte ve stromu objektů. Jedná se o vytvořený `fixed_link_line`, `input_link_line`, `coupler_link_line` a `output_link_line` (obrázek 3.24). Až po tomto kroku uložte mechanismus s vytvořenou kinematikou do knihovny potvrzením **End Modeling** (Modeling → Scope).



Obrázek 3.24 - Skrytí čar ve stromu objektů

Takto vytvořený kinematický mechanismus můžete rozšířit o další součásti jednoduše vytvořením nového linku a jointů v **Kinematics Editor**. To se hodí, pokud je například paralelogram součástí složitějšího mechanismu.

3.2 Rozpohybování robotu

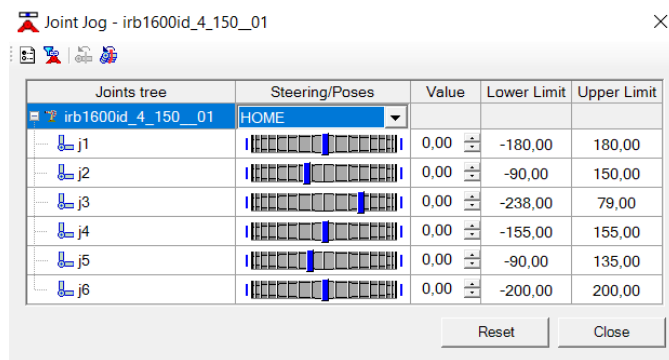
V této kapitole si ukážeme, jaké jsou možnosti pro pohyb s koncovým bodem robotu a jak připojit k robotu sedmou osu. Taktéž se dozvíte, co je potřeba udělat, aby váš vlastní model robotu byl chápán jako robot a mohl tak využívat všechny funkce softwaru.

3.2.1 Polohování robotu

Pohyb koncovým bodem robotu je umožněn dvěma způsoby. Jedná se o **Joint Jog** a **Robot Jog**. Aby nebyly tyto možnosti zašedlé, je potřeba nejdříve označit robot, se kterým se bude pohybovat.

Joint Jog (Robot → Toll and Device → Joint Jog)

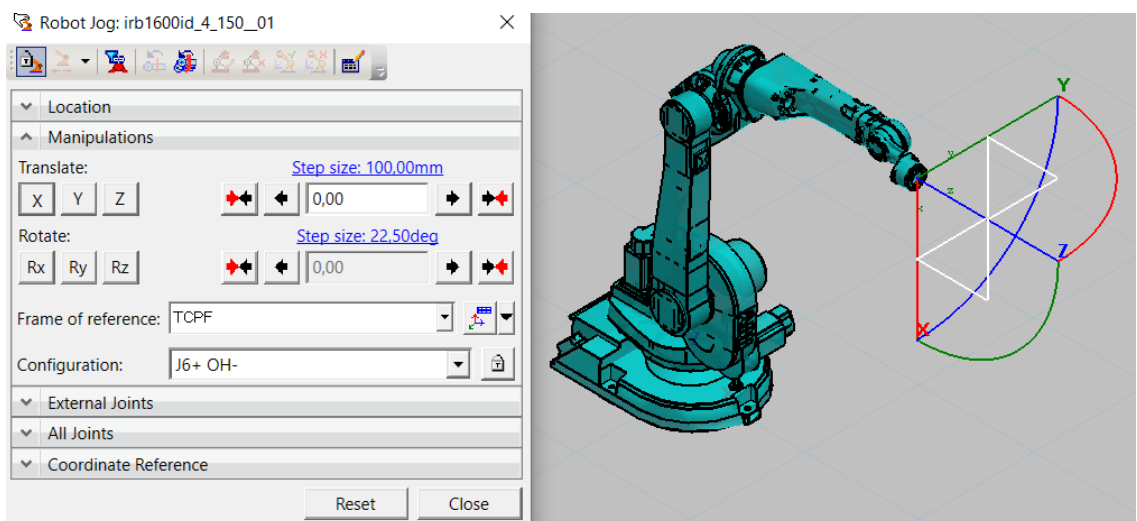
Je funkce, se kterou jsme se již seznámili při nastavování polohovadla (obrázek 3.25). Robot se zde ovládá tažením posuvníku, přepisováním hodnot (**Value**) nebo zvolením námi předem definovaných pozic (**Poses**). Když se posuvník dostane na hraniční hodnotu, změní se jeho barva na fialovou.



Obrázek 3.25 - Joint Jog Robotu

Robot Jog (Robot → Reach → Robot Jog)

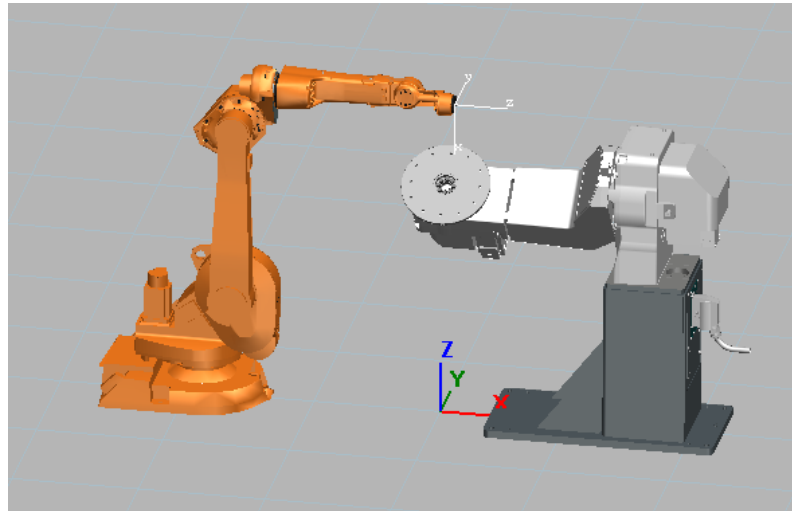
Umožňuje přesunování koncového bodu robotu, tažením jednotlivých os pomocí myši. Případně tlačítka šipek lze posouvat robot o zvolený krok u vybrané osy. Viz obrázek 3.26.



Obrázek 3.26 - Robot Jog

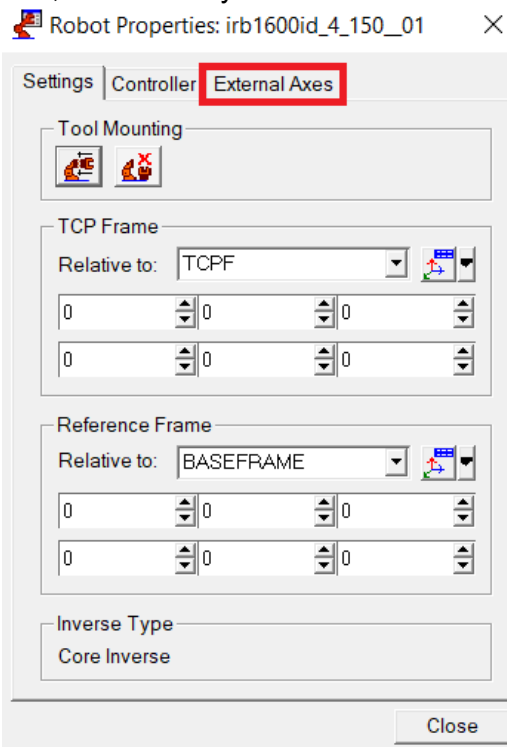
3.2.2 Propojení Jointů externích kinematických zařízení

V této kapitole si ukážeme, jak propojit jointy polohovacích zařízení s jointy závislého robotu. Pro ukázkou si vložíme polohovadlo **IRBPA-250_D1000**, které jsme nastavovali v kapitole 3.1 a robot **IRB1600ID_4_150** (obrázek 3.27).



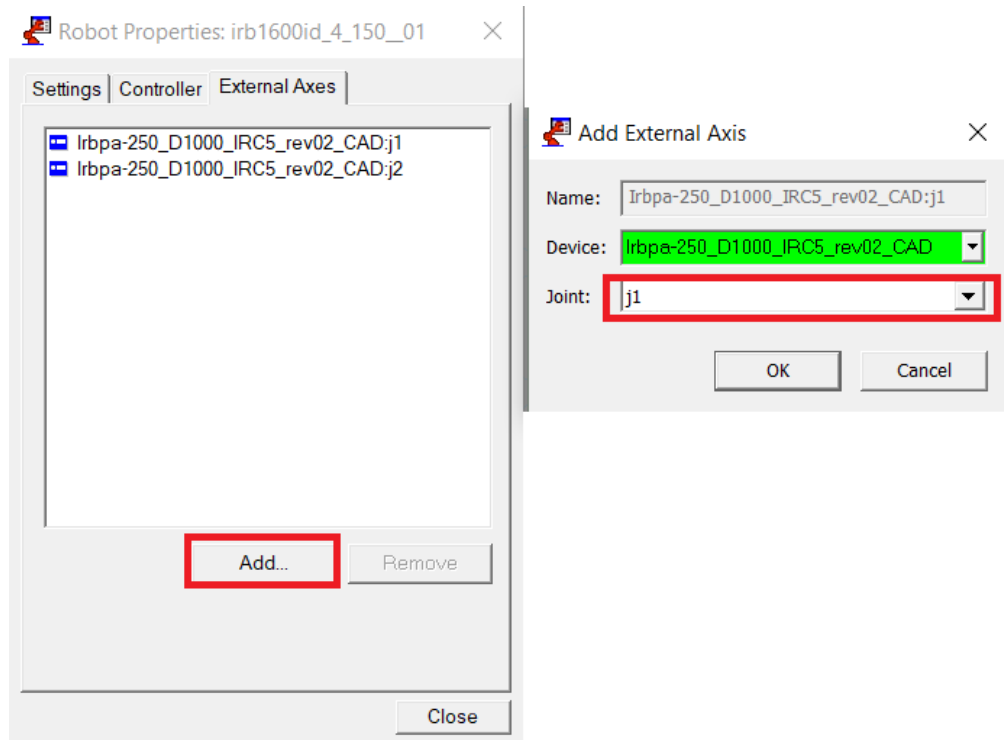
Obrázek 3.27 - Robot s polohovadlem

Ve stromu objektů označíme robot a v záložce **Robot** → **Setup** → zvolíme ikonu **Robot Properties**. Dále se otevře okno, ve kterém vybereme **External Axes** (obrázek 3.28).



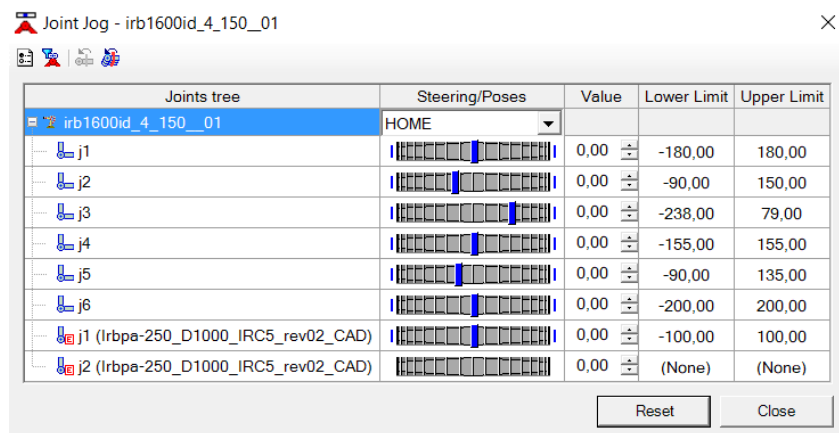
Obrázek 3.28 - Robot Properties

Tlačítkem **Add...** přidáme jointy **j1** a **j2** (obrázek 3.29).



Obrázek 3.29 - Přidání externího Jointu

Nyní lze ovládat robot a jeho externí osy, respektive připojená zařízení pomocí **Robot** → **Tool and Device** → **Joint Jog**. Červené E zobrazující se u jointů znamená, že se jedná o externí osy robotu (obrázek 3.30).

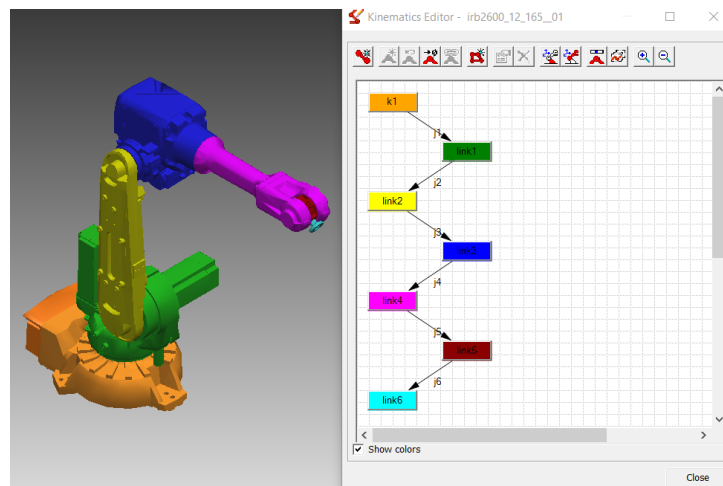


Obrázek 3.30 - Ovládání externích os

3.2.3 Inverzní kinematika robotu

Úloha inverzní kinematiky je proces, při němž je známa poloha koncového bodu TCPF (nástroje nebo konce robotického ramene). Při přesunu koncového bodu na novou pozici je potřeba dopočítat úhly natočení jednotlivých kloubů robotu.

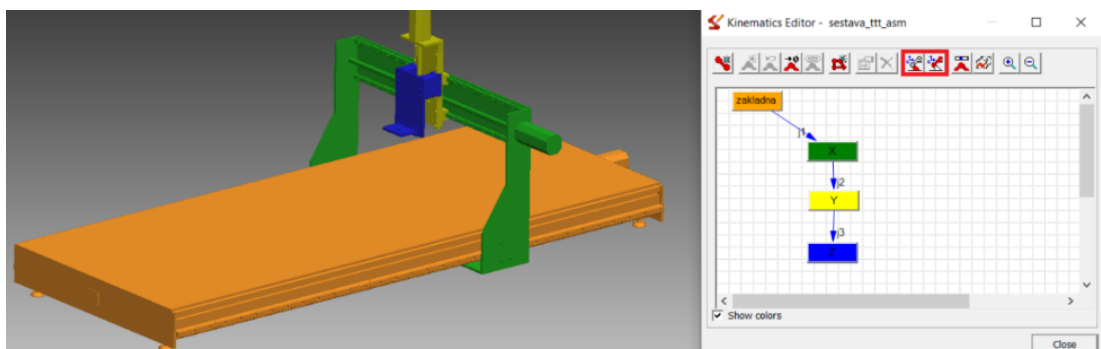
Obecně je tato úloha v RE jednoduchá, stačí přidat **TOOLFRAME** na konec kinematického řetězce robotu. Tímto krokem je taktéž přidána **TCPF**. Pro lepší pochopení, jak má vypadat kinematický řetězec robotu, si vložte jakýkoliv robot z knihovny, uvolněte jej pro modelování (**Set Modeling Scope**) a zobrazte si **Kinematics Editor** (obrázek 3.31) a prohlédněte si jeho prvky ve stromu objektů (Object Tree).





Obrázek 3.31 - Kinematika robotu

Pro 3 až 6 (v některých případech i pro 7) jointů používá RE tzv. **core inverse (základní inverzní řešitel)** pro řešení inverzní kinematiky. Pro více než 6 os je používán **approximate inverse** (přibližný inverzní řešitel). Který způsob řešení je použit u daného robotu zjistíte v zobrazení **Robot** → **Setup** → **Robot Properties** daného robotu.

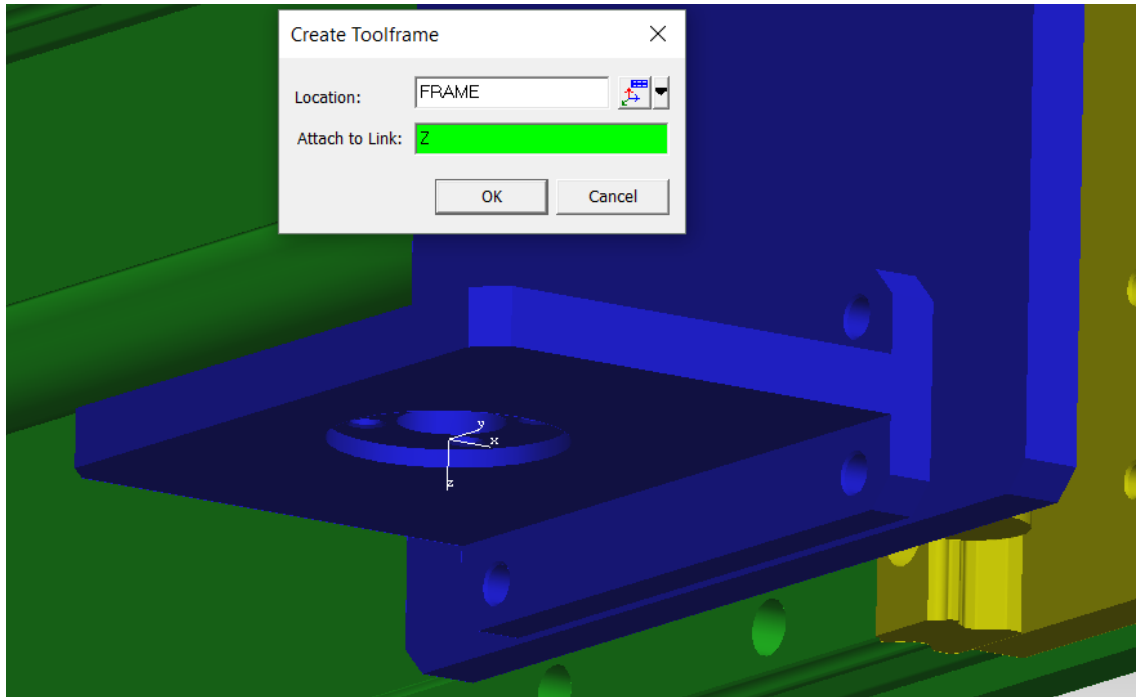
Jako příklad si uvedeme definování inverzní kinematiky pro 3-osý manipulátor (TTT – 3x translace). Práce s tímto modelem je dále rozebíraná v praktických ukázkách viz kapitola 7.1. **Příloha E** obsahuje CAD model manipulátoru. Jeho kinematickou strukturu znázorňuje obrázek 3.32. Označíme robot ve stromu objektů a uvolníme pro modelování (Set Modeling Scope), následně vybereme Kinematics Editor.



Obrázek 3.32 - Kinematika TTT

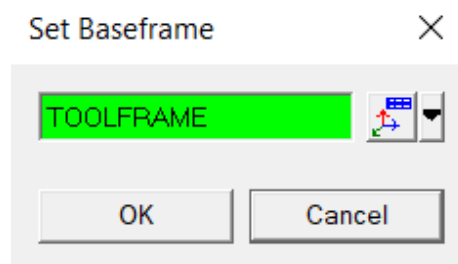
V kinematickém editoru budeme používat ikony **Set BaseFrame**  a **Create ToolFrame**  červeně znázorňuje obrázek 3.32.

Jako prvním nastavím **Create ToolFrame** pro vytvoření TCPF bodu. V kolonce **Location** zvolím bod nebo předem vytvořený souřadný systém a upravím jeho orientaci. Druhá kolonka **Attach to Link** je její rodičovský prvek sestavy (v tomto případě modrá součást) obrázek 3.33.



Obrázek 3.33 - Create Toolframe

V druhém kroku definuji **BaseFrame**, tedy bod, ke kterému se automaticky připojí nástroj. V tomto případě je ToolFrame totožná s BaseFrame (obrázek 3.34). Po vložení nástroje bude TCPF mechanismu umístěn na nástroji. Definování vlastního nástroje si blíže probereme v kapitole 3.3.2.



Obrázek 3.34 - Nastavení Baseframe

To, že byla inverzní kinematika správně nastavena si můžete ověřit například zobrazením funkce **Robot Jog** (Robot → Reach). Pokud nemá zařízení definovanou inverzní kinematiku, tak je tato funkce nedostupná.

3.3 Efektor

3.3.1 Připojení nástroje/efektoru

Jako příklad uvádím připojení svařovacího hořáku. Tento nástroj je součástí ukázkového příkladu, který je součástí instalačního souboru. Cesta pro nalezení je **CD13.1.2_RobotExpert** → **Add-Ons** → **ExampleCell** → **Getting Started** → **ArcWeld_Library** → **libraries** → **resource** → **weldtorches**.

Zde naleznete složku **Robacta5000_36_S_type.cojt**. Tuto složku si musíte zkopírovat a umístit do Vaší knihovny, protože vkládat soubory pomocí **Modeling** → **Components** → **Insert Component** je možné jen z vašeho Library Rootu, který jsme nastavovali v kapitole 2.5.1.

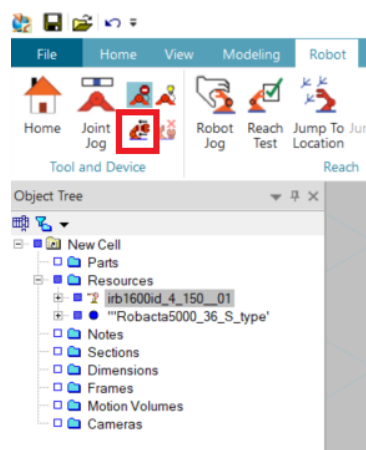
Z knihovny taktéž vložíme robot, na který hořák budeme připojovat.

Například **irb1600id_4_150_01** od firmy ABB. Ve stromu objektů

označíme vložený robot a v záložce **Robot** → **Tool and Device** vybereme ikonu **Mount Tool** (obrázek 3.36).




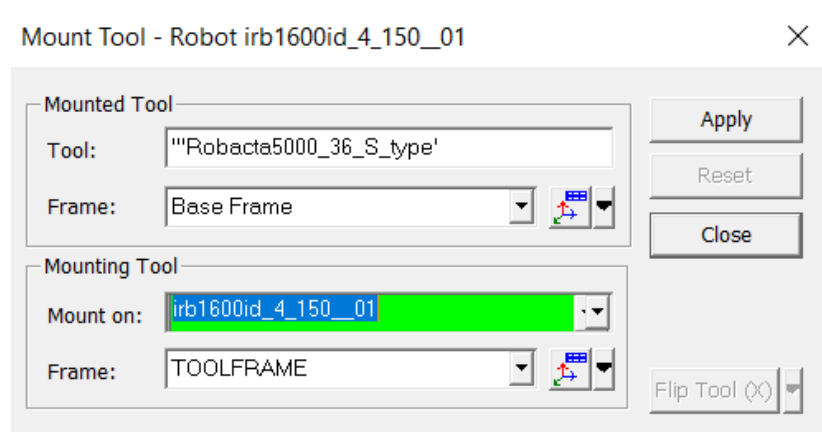
Obrázek 3.35 - Efektor



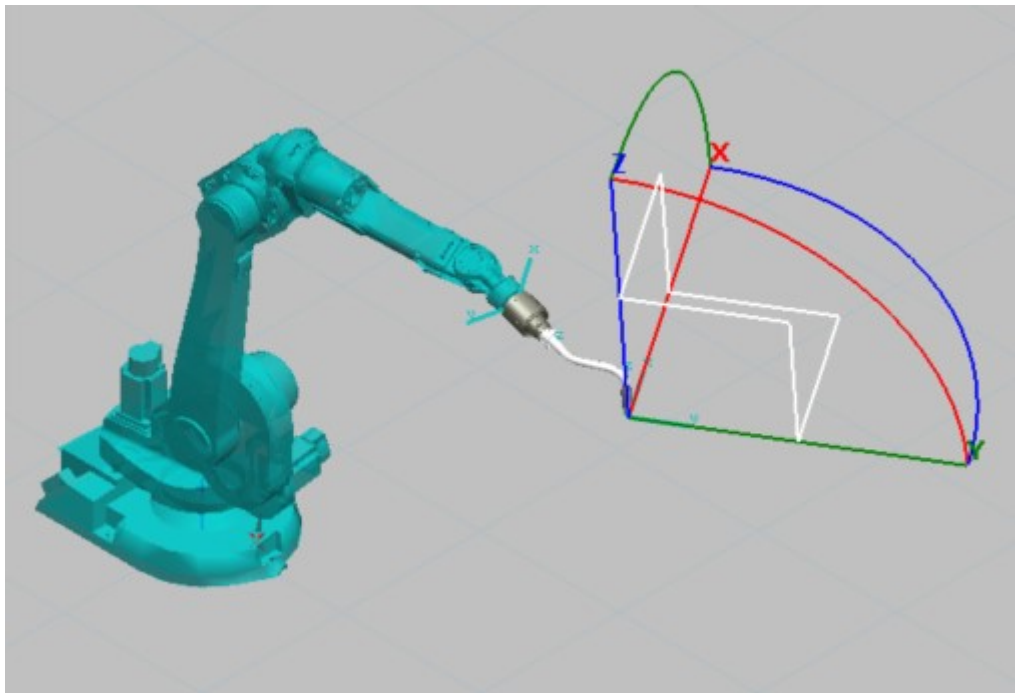
Obrázek 3.36 - Mount Tool

V záložce **Mount Tool** vybíráme nástroj, například označením ve stromu objektů (obrázek 3.37). Souřadný systém, podle kterého se připojí nástroj k souřadnému systému robotu se vybírá v obou případech automaticky. Tlačítkem **Flip Tool** můžeme potočit nástroj podle vybrané osy. Potvrdíme tlačítkem **Apply**. Nyní je nástroj připojen k robotu a lze si vyzkoušet (pomocí **Robot** → **Reach** → **Robot Jog**), že robot následuje koncový souřadný systém nástroje a ne interface robotu (obrázek 3.38).

Naopak rozpojení nástroje a robotu provedeme ikonkou  UnMount Tool (ve stromu objektů musí být označen nástroj).



Obrázek 3.37 - Výběr nástroje



Obrázek 3.38 - Nový koncový bod robotu

3.3.2 Definice nástroje/efektoru

Připojit nástroj k robotu již umíme, ale pokud vytváříte vlastní mechanismus, je potřeba jej vhodně definovat v RE. Nástrojem se rozumí objekt, který může být připojen k robotu a vykonávat daný úkol.

Typy nástrojů:

Gripper → Uchopení objektu mezi čelisti, používá se pro Pick and Place operace.

Paint Gun → Lakovací pistole.

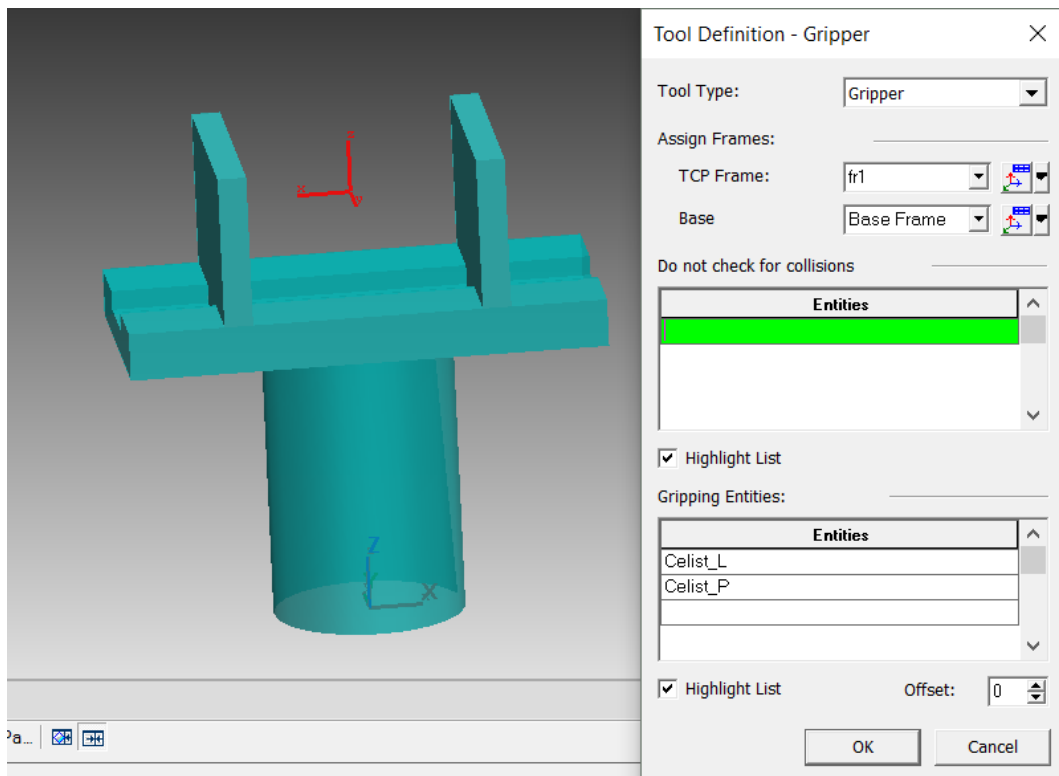
Gun → Vše ostatní (obloukové svařování, laserové řezání, frézování, leštění, vodní paprsek).

Servo Gun → Bodové svařování.

Pneumatic Servo Gun → Pneumaticky řízené.

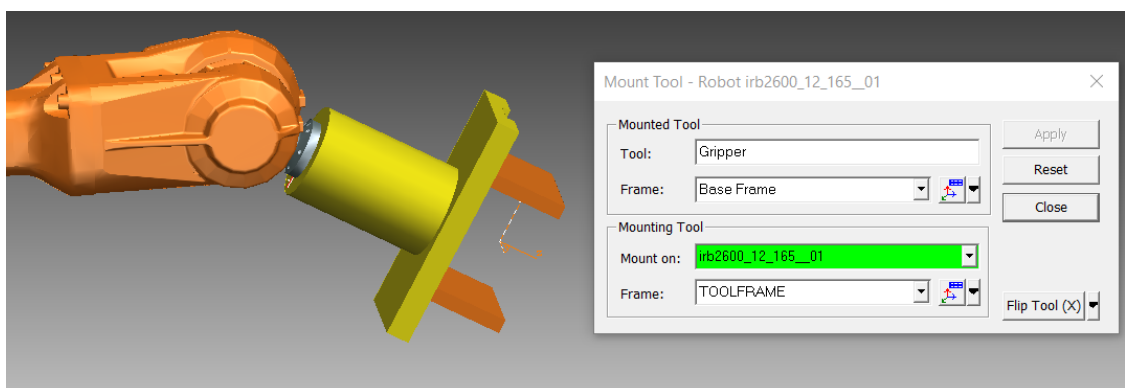
Definování modelu jako nástroje, je pro uvedené typy obdobné, proto si jako příklad uvedeme jen **Gripper** pro zjednodušenou součást vymodelovanou v RE. Kinematiku tohoto efektoru jsme již nastavovali v kapitole 3.1.1.1.

Vložte si dříve vytvořený model a povolte ho pro úpravy zvolením **Modeling** → **Scope** → **Set Modeling Scope**. Ve stromu objektů si součást označíme a přejdeme k samotné definici **Modeling** → **Kinematics Device** → **Tool Definition**. Zobrazí se okno (obrázek 3.39) a jako **Tool Type** vybereme **Gripper**. Dále **TCP Frame** je bod nástroje, podle kterého se bude robot polohovat (poslední bod kinematického řetězce robotu). Jako TCP volím předem vytvořený bod **fr1**. **Base** je naopak místo pro připojení nástroje k robotu. V kolonce **Do not check for collision** zvolíme ty prvky součásti, které nebudou detekovány při kolizi. V posledním okně **Gripping Entities** zvolíme čelisti. Uchopení je provedeno zjištěním kolize, mezi **Gripping Entities** definovanými pro nástroj a jakýmkoliv fyzickým objektem. Hodnota **Offset** definuje vzdálenost, ve které dochází k detekci kolize.



Obrázek 3.39 - Definice nástroje

Potvrdíme OK a efektor uložíme ukončením modelování (**Modeling** → **Scope** → **End Modeling**). V tomto kroku je dobré si ověřit, že byly souřadnice TCP a Base vhodně zvoleny. Ověření provedeme přimontováním gripperu na jakýkoliv robot z knihovny (obrázek 3.40). Musíme mít označený robot ve stromu objektů a zvolit **Robot** → **Tool and Device** → **Mount Tool** jak bylo podrobněji ukázáno v předchozí kapitole.



Obrázek 3.40 - Ověření správnosti připojení efektoru

4 Simulace

Simulací pracoviště lze odhalit kolize pohybujících se součástí, ověřit dosah robotu, tvořit animace operací a další. Pro odladění programu je simulace důležitá zejména pro ověření času vytvářené operace a její následná modifikace. V této kapitole si přiblížíme, jak postupovat při tvorbě dráhy robotu a jakými nástroji lze odhalit její nedokonalosti a provést následnou úpravu. Konkrétní operace si ukážeme až v kapitole 5.

4.1 Skok nebo přesun robotu na pozici



Jump To
Location

Jump to Location (Robot → Reach)

Označený robot skočí do zvolené pozice.



Move to
Location

Move to Location (Robot → Play)

Nejdříve označte pozici a poté zvolte toto tlačítko. Robot se přesune na požadovanou pozici ze své původní pozice.

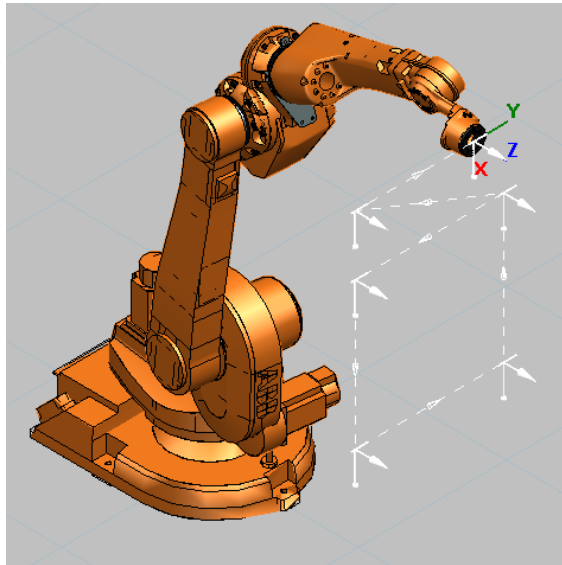


Jump Assigned
Robot

Jump Assigned Robot (Robot → Reach)


Nejdříve označte pozici a poté zvolte toto tlačítko. Robot skočí na vybranou pozici.

4.2 Dráha pohybu



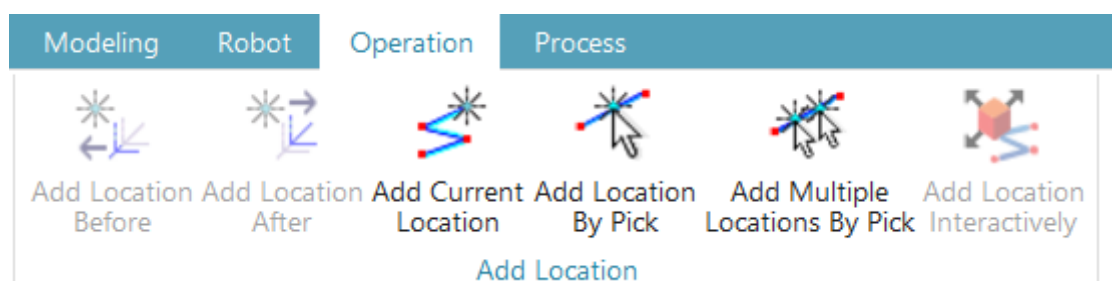
Obrázek 4.1 - Robot s dráhou pohybu

Uložení pozic polohy robotu a simulace jeho pohybu se provádí pomocí **Operations** → **Create Operation** → **New Operation**. Zde máme na výběr několik běžných aplikací, jako například **Pick and Place** (uchopení a položení objektu) a další.

Jako příklad si uvedeme **New Generic Robotic Operation** (všeobecnou robotickou operaci). Po zvolení se tato operace vytvořila ve stromu operací (**Operation Tree**). Přidáme vytvořenou operaci do **Path Editoru** ikonou **Add Operations to Editor** .

4.2.1 Vytvoření lokací

Robotem můžeme pohybovat již známým způsobem, tedy použitím **Joint Jog** nebo **Robot Jog** nacházející se v záložce **Robot**. Tyto pozice do stromu operací přidáváme v záložce **Operation** → **Add Location**, kde máme na výběr několik možností (obrázek 4.2). **Add Current Location** přidá aktuální pozici. Výběrem **Location By Pick** lze vybrat předem vytvořené body, mezi kterými se bude robot pohybovat. Případně přidání pozice před/za (**Add Location After/Before**).

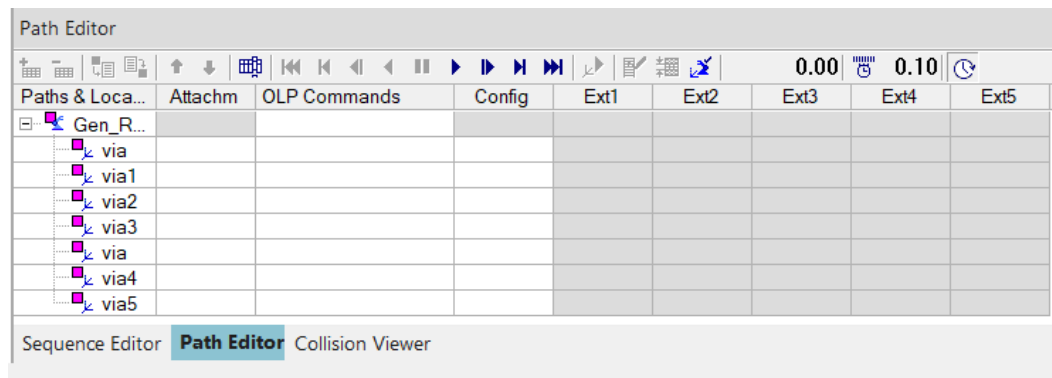


Obrázek 4.2 - Tvorba pozic

V **Path Editoru** lze měnit pořadí pozic, které robot projíždí, případně je i kopírovat.

Spouštění a resetování simulace se provádí pomocí tlačítek .

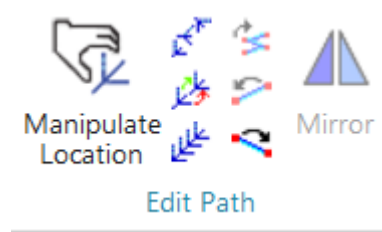
Změna rychlosti simulace  .



Obrázek 4.3 - Path Editor s pozicemi

4.2.2 Úprava lokací

Pro úpravu lokací je možné používat již známé nástroje **Placement Manipulator** a **Relocate**. Další nástroje pro úpravu lokací se nachází v záložce **Operation** → **Edith Path**.



Obrázek 4.4 - Úprava dráhy

Najdete zde nástroje pro zrcadlení lokací, úpravu jejich orientace nebo i otočení směru pořadí vytvořené cesty (obrázek 4.4). Osobně používám nejčastěji funkci **Copy Location Orientation**, která orientuje pozice podle vybrané referenční pozice.

4.3 Detekce kolize




V případě, že dojde ke kolizi, barva objektu se změní na červenou. Je možné zvolit, zda se simulace při kolizi zastaví, případně jestli bude přehrán zvuk v kolizním stavu. Zobrazení kolizi naleznete v záložce **View** → **Viewers** → **Collision Viewer** a defaultně se nachází dole vedle Path Editoru.

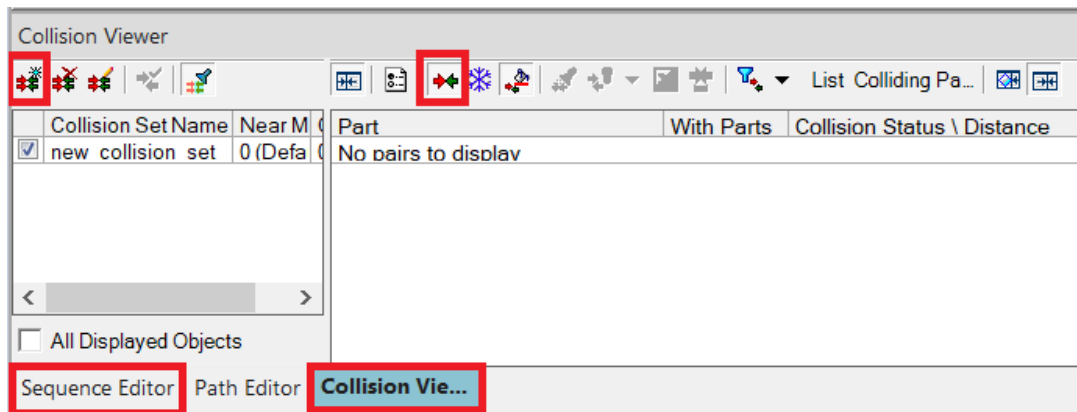
Existují dva typy kolizí:

1. Dynamická – dochází k nim během simulace nebo při umísťování objektů
2. Statická – dochází k nim bez pohybu objektů

Rozlišujeme 3 úrovně kolize, které lze rozlišit zbarvením kolizních objektů:

- Žlutá - objekty, které mají mezi sebou předem určenou mezeru
- Červená - objekty se dotýkají (oranžová, pokud je aktivována detekce chyb)
- Červená - objekty sebou pronikají

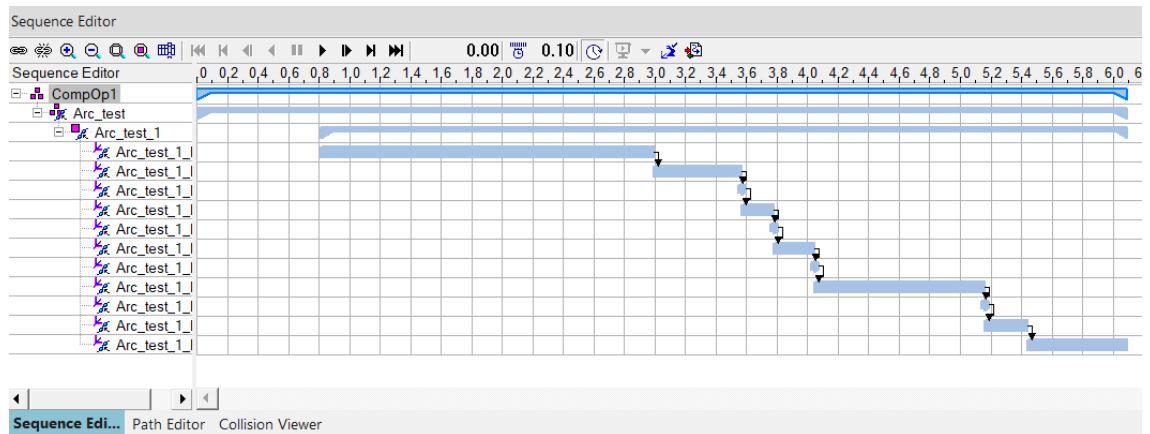
Vzdálenosti, které budou detekovány, zastavení simulace při kolizi a další detaily se nastavují v záložce **File** → **Options** → **Collision** (obrázek 4.5). Pro aktivaci kolize je nutné nastavit, které objekty mají být mezi sebou sledovány **New Collision Set**  a následně aktivovat kolizní mód **Collision Mode On/Off** . Dodatečně lze vygenerovat zprávu o dynamické kolizi v Excelu. V záložce **Sequence Editor** vybráním nástroje **Dynamic Collision Report** .



Obrázek 4.5 - Zobrazení kolizí

4.4 Sequence Editor

Nástroj pro editaci pořadí prováděných operací a přidávání různých událostí. Nachází se vedle Path Editoru, případně jej lze zobrazit pomocí **View** → **Viewers** → **Sequence Editor**. Pro přidání operace do sequence editoru musíte nejdříve označit danou operaci ve stromu operací a zvolit možnost **Set Current Location**, která se nachází v záložce **Operation** → **Create Operation** nebo pod pravým tlačítkem myši.





Obrázek 4.6 - Sequence Editor

Po přidání operace do editoru je možné myší přesunovat pořadí jednotlivých operací (obrázek 4.6). Než se pustíte do úprav, doporučuji si vytvořit nový snapshot (viz kapitola 2.9.1), protože po přesunu operace není možné vrátit se o krok zpět nebo resetovat průběh do počáteční polohy.

Za zmínku stojí i možnost přidat do operací různé události pomocí funkce **New Event (Operation** → **Events)**. Zde jsou možnosti pro přidání události jako jsou například Attach/Detach, zmizení, přiblížení obrazu, zapnutí sledování kolizí, průhlednost apod. Další možnosti jako je přidání doby čekání a posílání signálu apod. si ukážeme v kapitole 6.1.

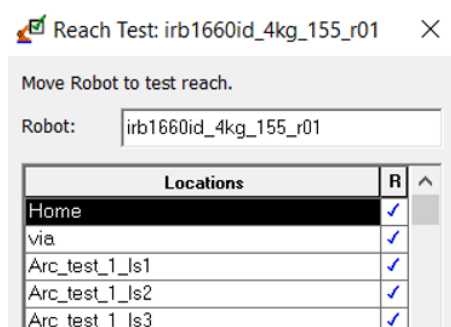
4.5 Ověření dosahu robotu

Dosah robotu do požadovaných pozic lze ověřit dvěma způsoby.

- Použitím funkce:
1. **Reach Test** 
 2. **Smart Place** 






Reach test (Robot → Reach)

Po otevření stačí vybrat robot a pozice které budou ověřovány. Skok robotu do konkrétní pozice lze provést dvojklikem myši na příslušný řádek. Během ověřování dosahu můžete přesouvat robot a sledovat, jak se bude měnit jeho dosah do jednotlivých bodů (obrázek 4.7).



Obrázek 4.7 - Ověření dosahu robotu

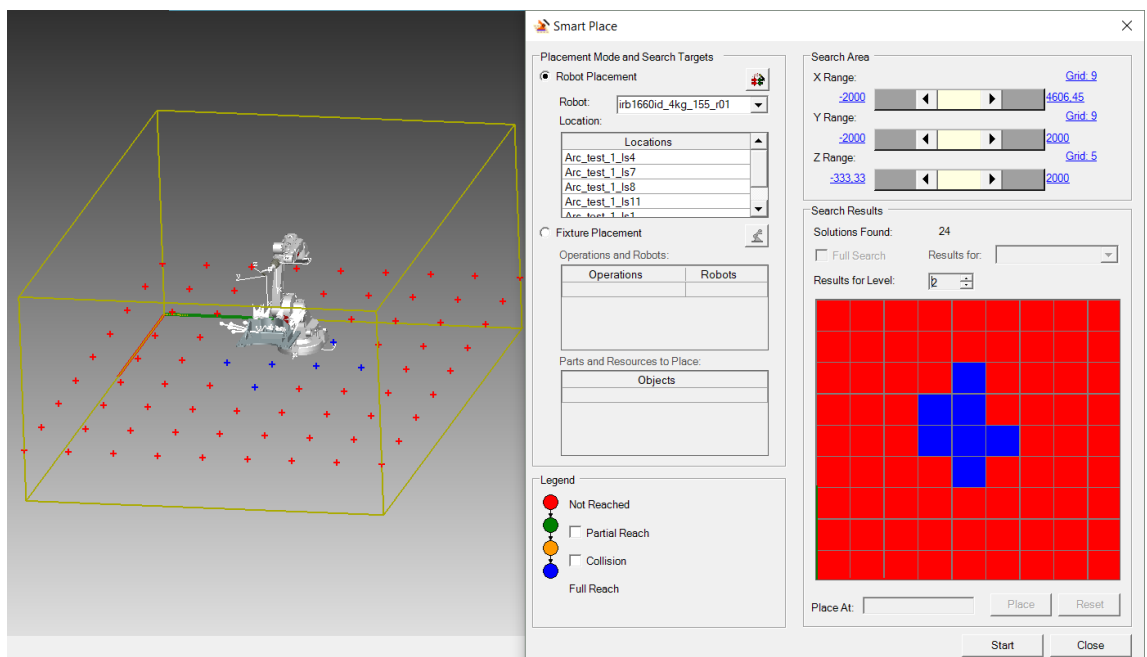
Ve sloupci R (reach = dosah) se můžete setkat s následujícími znaky:

- ✓ Robot dosáhne do zvoleného bodu.
-  Robot dosáhne této pozice, ale musí otáčet TCPF tak, aby odpovídal TCPF umístění.
-  Robot dosáhne na místo v rámci svých fyzických mezí, ale mimo jeho pracovní limity.
-  Robot dosáhne na místo v rámci svých fyzických mezí, ale mimo jeho pracovní limity, ale musí otáčet TCPF, aby odpovídal TCPF umístění.
-  Robot dosáhne požadovaného místa, ale nachází se mimo své fyzické hranice.
-  Robot dosáhne požadovaného místa, ale nachází se mimo své fyzické hranice, ale musí otáčet TCPF tak, aby odpovídal TCPF umístění.
- ✗ Robot nemůže dosáhnout požadovaného bodu.

Smart Place (Robot → Reach)


Tato funkce vypočítá možná umístění robotu vůči bodům, na které má dosáhnout (obrázek 4.8). Opět je potřeba nejdříve zvolit posuzovaný robot a pozice. V oblasti **Search Area** vytvoříte oblast, ve které bude vyhledáváno nejlepší umístění robotu. Taktéž v oblasti **Legend** můžete přiřadit vyhledávání míst, ve kterých dochází k částečnému dosahu nebo kolizi. Spusťte **Start** a sledujte, jak se okno vybarví. Přemístění robotu do modře vyznačené pozice (s plným dosahem) lze provést dvojklikem na danou oblast nebo označením místa a tlačítkem **Place**, návrat provedete tlačítkem **Reset**. Lze prověřit i řešení v jiném výšce robotu změnou hodnoty **Results for Level**.

Při použití druhé možnosti **Fixture Placement** je postup stejný, jen dochází k přemísťování předmětu s pozicemi procesu, zatímco robot zůstává na svém místě.

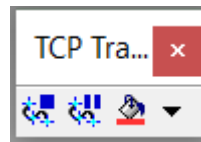


Obrázek 4.8 - Smart Place

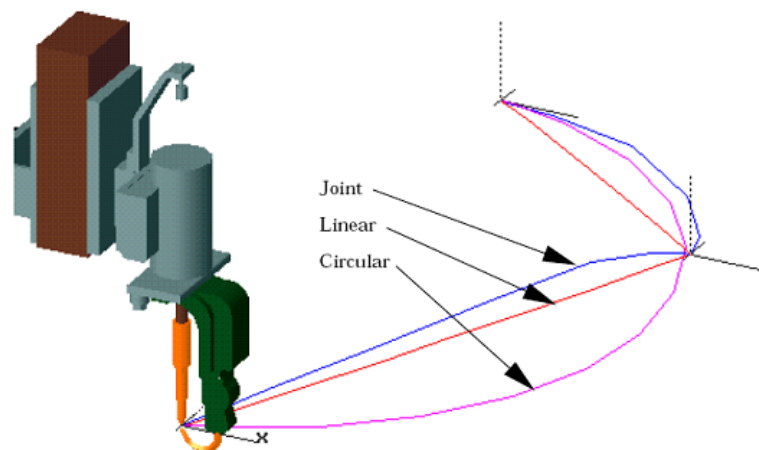
4.6 Zaznamenání trajektorie TCP

Příkaz **TCP Tracker**  (**Robot** → **Analysis**) umožňuje zaznamenávat trajektorii koncového bodu robotu (TCPF) jako křivku (obrázek 4.10). Stopa se zobrazuje jako modelovaný objekt ve stromu objektů. Stopa se tvoří pro jakýkoliv pohyb robotu, ať už v průběhu simulace tak i při použití funkce Robot Jog.


Při spuštění této funkce se zobrazí okno, kde lze změnit barvu stopy nebo pozastavit její vykreslování (obrázek 4.9).



Obrázek 4.9 - TCP Tracker



Obrázek 4.10 - Vykreslování pohybu [9]

Užitečné může být využití nástroje pro měření křivky **Curve Length** . Tak lze jednoduše zjistit ujetou vzdálenost TCPF robotu.

Pro vytvoření efektu kreslení jen v určitém čase nebo místě lze využít OLP příkaz, který se nachází v Teach Pendantu. To budeme blíže rozebírat v kapitole 6.1. Ukázka použití je také rozpracovaná v praktickém příkladu v kapitole 7.2.

4.7 Monitorování průběhu vlastností robotu

Robot → Play → Robot Viewer

Užitečný nástroj pro sledování rychlostí, zrychlení jednotlivých kloubů a spotřeby proudu robotu. Nejdříve v horní části okna přiřadíte sledovaný robot a potom už jen posuňte Jointy robotu do sledované pozice nebo spusťte simulaci a nechte si vykreslit graf sledující rychlosti, zrychlení a proudy v čase.

Joint Monitor

-Grafické znázornění velikosti natočení jednotlivých jointů v každém okamžiku simulace.

Joint Status

-Zobrazuje aktuální hodnotu pro každý joint robotu (včetně externích jointů) a jejich hodnoty v čase. Případně lze zobrazit i procentuální vyjádření natočení.

TCPF Speed monitor

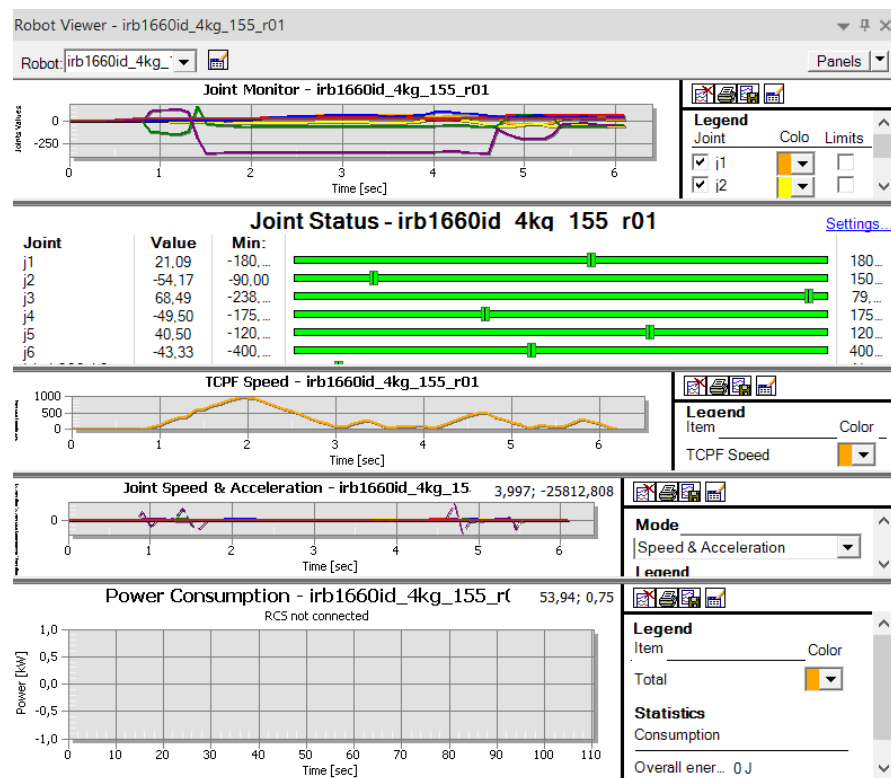
-Grafické znázornění koncového bodu robotu (TCPF).

Joint Speed and Acceleration

-Zobrazuje grafy rychlosti a zrychlení jednotlivých jointů v čase.

Power Consumption

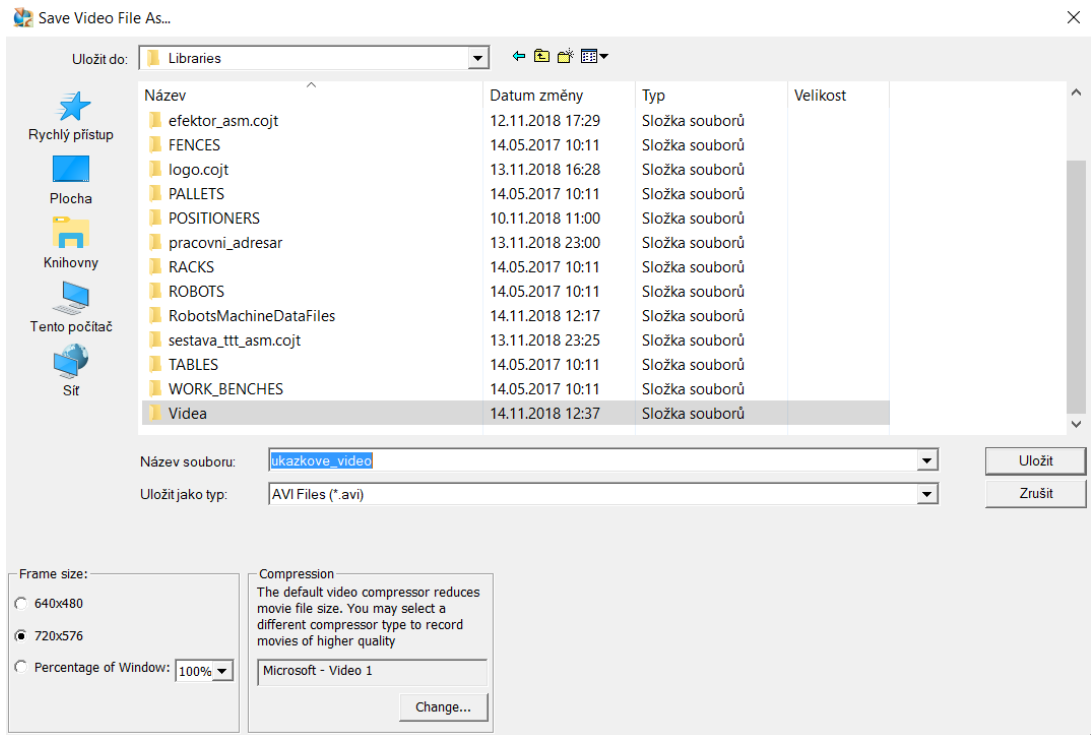
-Grafické znázornění aktuální spotřeby energie robotu v jednotkách kW (kilowatty) → musí být přiřazen RSC modul kontroléru. O RCS si řekneme více v kapitole 6.2.



Obrázek 4.11 - Sledování vytížení robotu

4.8 Tvorba animace

Ikona pro nahrávání se nachází v záložce **Operation** → **Documentation** → **AVI Recorder**. Po zvolení je potřeba nastavit cestu, kam se video bude ukládat a kvalitu, s jakou bude video nahráno (obrázek 4.12). Na kvalitu má vliv i způsob komprese souboru. Před nahráváním doporučuji si vaši práci uložit!



Obrázek 4.12 - Tvorba animace

Spuštění nahrávání započne po startu simulace, klasickým způsobem v **Path Editoru**. Pro ukončení nebo pozastavení nahrávání se provádí pomocí okna **AVI**, které se otevřelo již od začátku nahrávání (obrázek 4.13).



Obrázek 4.13 Zastavení nahrávání

Vytvořené video je ve formátu AVI a nachází se v místě vámi zadané cesty.





Pokud vám kvalita videa nestačí, je vždy možné použít externí software pro nahrávání plochy vašeho počítače. Ve windows 10 je tato funkce již obsažena. Nejjednodušší je zpřístupnění této funkce pomocí klávesové zkratky WIN+G nebo vyhledejte v nastavení počítače tzv. herní panel. Pro ořezání zobrazovaného obrazu nahraného videa už musíte použít jakýkoliv volně stažitelný software. Dobrou zkušenost mám se softwarem Xmedia Recode.

Používáte-li jiný operační systém nebo vám z nějakého důvodu tento postup nevyhovuje, můžete využít i jiné programy pro nahrávání videa. Při tvorbě video postupů, které jsou v příloze jsem používal software OBS Studio.

Pohybující se kamera

Pro tvorbu pohybujících se záběrů rozklikněte kartu **View** → **Camera**. Tato funkce slouží i k zobrazení více oken s různými úhly záběru při použití více kamer.

V záložce **Camera** se nachází tyto 4 možnosti:

- **Create Camera** 
Vytvoří kameru ve vybrané pozici.
- **Open Camera Viewer** 
Otevře nové okno s pohledem vybrané kamery.
- **Align Camera With Current View** 
Naorientuje kameru dle aktuálního zobrazení
- **Add Current View as Camera Location** 
Přidá pozici aktuální polohy kamery do dráhy pohybu.

V prvním kroku vytvořte kameru (zobrazí se ve stromu objektů). Označte ji a přejděte do záložky **Operation** → **Create Operation** → **New operation** a zvolte **New Object Flow Operation**. Zde musíte přiřadit počáteční a koncový bod cesty pohybu nebo použít již vytvořenou dráhu. Po potvrzení se ve stromu operací zobrazí právě vytvořená operace. Pokud ji označíte, zpřístupní se vám poslední funkce (**Add Current View as Camera Location**) v záložce kamera, kterou můžete přidávat další pozice do dráhy pohybu. Pohled z kamery si zobrazíte pomocí funkce **Open Camera Viewer**.

Vytvořenou operaci vložte pod složenou operaci (**New Compound Operation**) i s dalšími operacemi, které mají běžet současně. Tuto složenou operaci poté vložte do **Sequence editoru** a upravte vůči sobě jejich průběh. Práce s editorem byla vysvětlena v kapitole 4.4.

5 Robotické aplikace

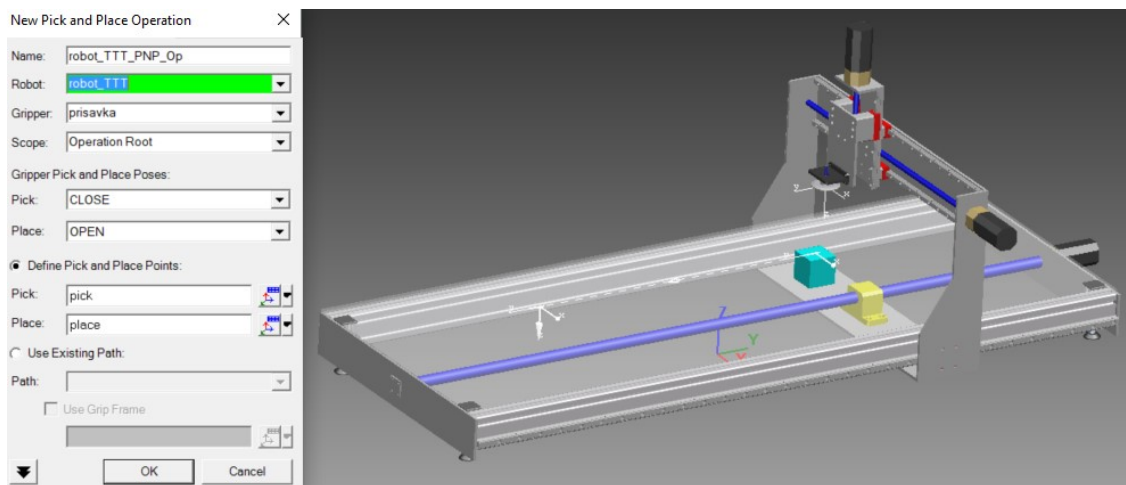
Robot je univerzální zařízení používané pro různorodé aplikace. Při tvorbě těchto aplikací musíme k tvorbě programu přistupovat individuálně, a to je důvod, proč je v RE tolik druhů robotických operací. Postupně si je zde všechny popíšeme.

Obecně by se dalo říci, že můžeme aplikace rozdělit do těchto kategorií:

- Manipulace s materiálem (montáž, paletizace, přenos dílů)
- Svařování (svařování obloukem „ARC“, bodové)
- Lakování
- Ostatní (lepení, frézování, laserové řezání, čištění, vodní paprsek, apod)

5.1 Pick and Place

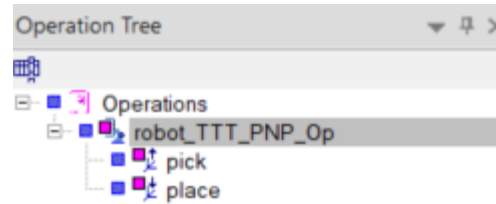
Jako první krok vyberte robot z knihovny a spojte ho s nástrojem (Mount Tool). Poté přejděte k výběru operace **Operation** → **Create Operation** → **New Operation** → **New Pick and Place Operation**. Nechcete-li z nějaké důvodu používat operaci **New Pick and Place Operation** pro přemístění objektu, můžete použít i obecnou robotickou operaci (**New Generic Robotic Operation**) a uchopování objektu přiřadit přes OLP příkazy (viz kapitola 6.1).



Obrázek 5.1 - Pick and Place

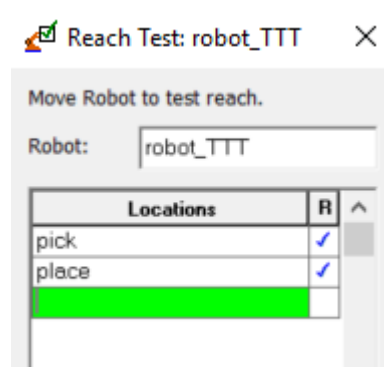
Zobrazí se okno, ve kterém je potřeba přiřadit robot, který bude aplikaci vykonávat. Nástroj, který uchopí předmět. Kolonku **Scope** (můžeme ponechat původní) definuje prostor, ve kterém se bude pohyb provádět. V části **Gripper Pick and Place Poses** přiřadíme polohy, které jsme nastavili při definování nástroje (viz kapitola 3). V tomto případě používám přísavku, takže výsledek nemusí být hned zřejmý, ale pokud by zde byl použit dvou-čelistový efektor, byl by tento krok definicí, kdy se mají čelisti zavřít/rozevřít. V poslední části přiřadíme souřadnice místa, kde bude předmět uchopen (**Pick**) a kde položen (**Place**). Potvrdíme OK.

Po tomto kroku se ve stromu operací (**Operation Tree**) vytvoří Pick and Place operace (obrázek 5.2). Nyní tam jsou pouze dva body s názvem „pick“ a „place“. Body můžete kopírovat, mazat a přidávat. Přidání nového bodu provedete s označenou operací ve stromu objektů v záložce **Operation** → **Add Location**.



Obrázek 5.2 - Strom operací

V dalším kroku doporučuji provést ověření dosahu robotu mezi jednotlivými body. Zvolte **Robot** → **Reach** → **Reach Test** a zobrazí se tabulka, kde stačí přiřadit používaný robot a ze stromu operací vybrat požadované body.



Obrázek 5.3 - Dosah robotu


Pokud je vše v pořádku, zobrazí se modrá fajfka. Zobrazuje-li se červený křížek nebo jiný znak přejděte do kapitoly 4.5, zde zjistíte význam jednotlivých chyb a můžete provést nápravu. Důležité je si uvědomit, že orientace TCPF (robotu nebo nástroje) se promítne do vytvořeného bodu. To znamená, že orientace os bude po najetí do bodu stejná. Používáte-li univerzální šestiosý robot, uvidíte jeho natočení v simulaci. V tomto případě používám pouze tříosý translační robot a natočení jednotlivých os nelze provést. Musíte tedy body vkládat zodpovědně, případně je můžete ještě editovat (viz kapitola 4.2.2).

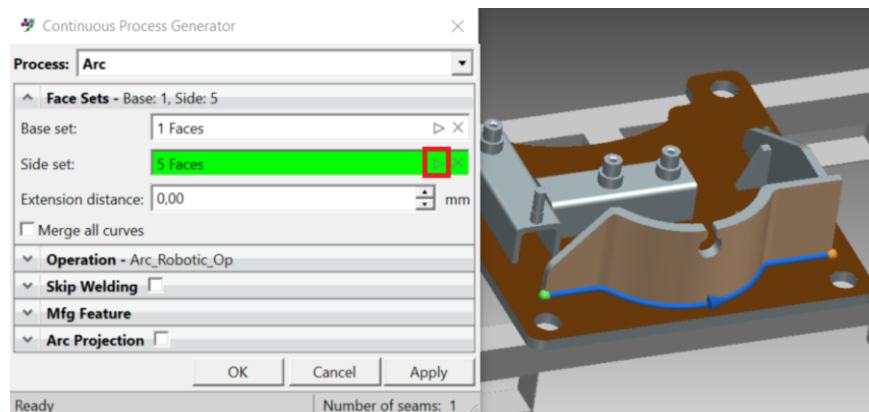
V posledním kroku už můžete přidat operaci do **Path Editoru**  a spustit simulaci.

5.2 Continuous Process Generator (svařování, lakování)

Continuous feature operation (česky nepřetržitá operace) musí být vždy spojena s jedním nebo více díly, které jsou buď přímo nebo nepřímo přidružené ke scéně. Jak už bylo řečeno, spadají sem všechny aplikace, které pracují kontinuálně jako například lakování, lepení, čištění, atd. Jako příklad si uvedeme obloukové svařování a lakování.

Obloukové svařování

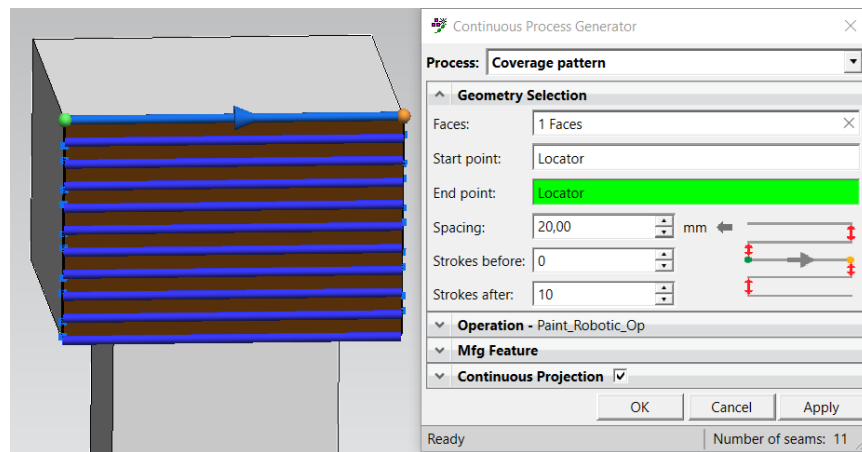
Otevřete si **Operation** → **Create Operation** → **New Operation** → **Continuous Process Generator**, zobrazí se okno (obrázek 5.4), ve kterém zvolíme **Base set** (podstavu dílce) a **Side Set** (stěnu prvku). Jako **Process** musí být přiřazený **Arc**. Svar se automaticky vygeneruje v místě spojení těchto prvků. Při výběru Side Set klikněte na jakoukoliv část stěny, a pro vytvoření svaru po celé délce klikněte na symbol  (**Tangent faces**). Dodatečně lze ještě změnit název operace v záložce **Operation** nebo místo uložení buňky v záložce **Mfg Feature**.



Obrázek 5.4 - Continuous Process Generator - Arc

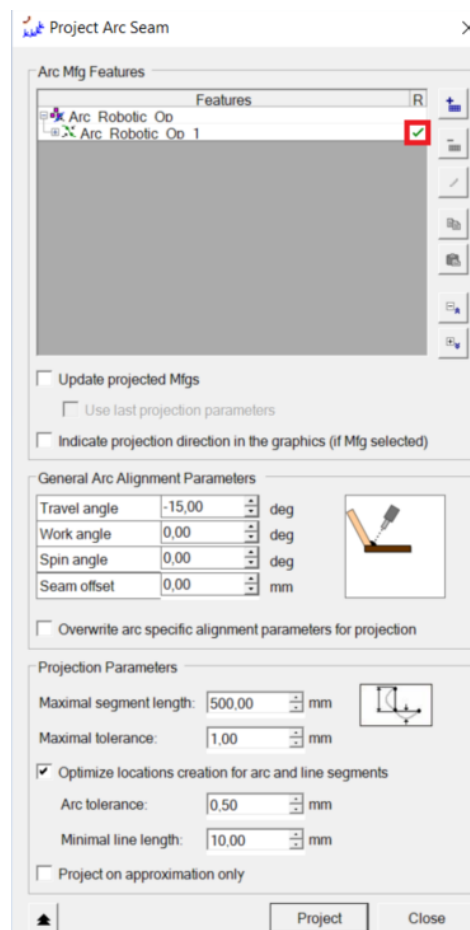
Lakování

Otevřete si **Operation** → **Create Operation** → **New Operation** → **Continuous Process Generator**, zobrazí se okno (obrázek 5.5), ve kterém nejdříve přiřadíme **Process** jako **Coverage pattern**. Jako **Faces** zvolíme ty plochy, na kterých bude proces probíhat. Následuje výběr počátečního a koncového bodu řídicí úsečky (Start/End point). Zbývá už jen nastavit velikost mezery mezi jednotlivými úsečkami a vybrat kolikrát se promítnou nad nebo pod řídicí úsečkou. Dodatečně lze ještě změnit název operace v záložce **Operation** nebo místo uložení buňky v záložce **Mfg Feature**. RobotExpert v základu neumožňuje simulovat pokrytí barvou, ale pomocí **Teach Pendantu** a záložky **Paint** můžete aktivovat lakování mezi body. Práce s Teach Pendantem je blíže rozebírána v kapitole 6.1. Spuštění procesu se v simulaci projeví jako zobrazení kužele, který znázorňuje rozprašování barvy. Velikost kužele se odvíjí od nastaveného nástroje, který je potřeba definovat jako **Paint Gun** a v kolonce **Tip Frame** zvolit počátek rozstříku, zatímco **TCP Frame** je jeho konec. Software Process Simulate má pro lakování více užitečných nástrojů, ale i tam je jejich použití podmíněno použitím doplňující licence.



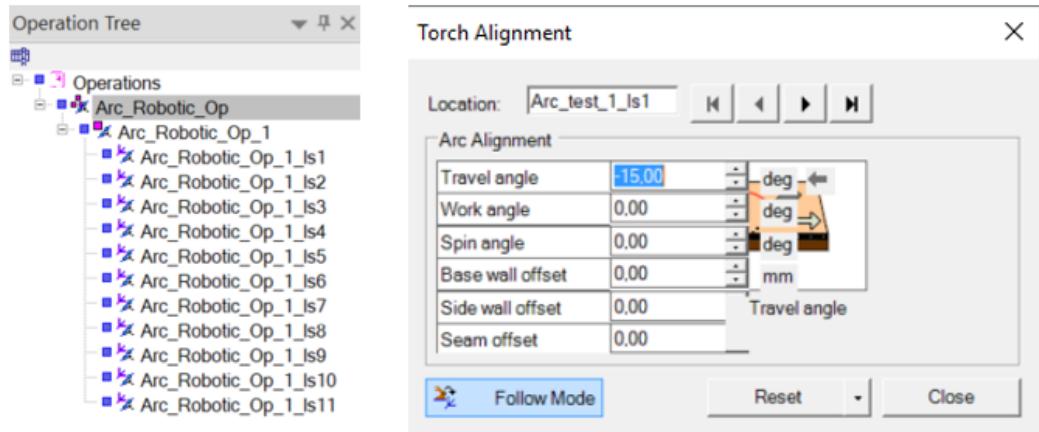
Obrázek 5.5 - Continuous Process Generator - Coverage pattern

Při těchto aplikacích je dobré zobrazit si **Mfg Viewer**, který zobrazuje procesní prvky (svary apod.) při operacích jako je svařování, lepení a podobné. Nyní vytvoříme dráhu robotu. Zvolte **Process** → **Arc** → **Project ARC Seam**. Zde můžete nastavit parametry jako je pracovní úhel, natočení hořáku, posun svaru a další (obrázek 5.6). Po kliknutí na políčko **Project** se zobrazí zelená fajfka u názvu operace v okně **Arc Mfg Features**, poté můžete okno zavřít **Close**.




Obrázek 5.6 - Projekt Arc Seam

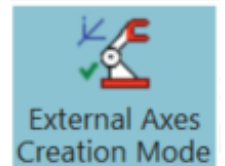
Tím jsme vytvořili dráhu robotu, kterou můžeme dodatečně upravovat ve stromu operací. Chcete-li upravovat naklonění svařovacího hořáku, lze tak dodatečně učinit v záložce **Process** → **Arc** → **Torch alignment**. Mezi jednotlivými body se lze přesouvat šipkami v horní části okna. Aby se souběžně zobrazil robot s hořákem v daném bodu, je potřeba mít zapnutý **Follow Mode** (obrázek 5.7).



Obrázek 5.7 - Nastavení hořáku

Operaci vložíme do **Path Editoru** , kde můžeme případně přidat další pozice robotu, jako například začátek a konec (domácí pozice robotu).


Aby simulace neignorovala polohu sedmé (externí) osy, v tomto případě polohovadla, je potřeba mít zapnutý **External Axes Creation Mode (Robot** → **OLP**). Ikona obrázek 5.8.



Obrázek 5.8 - Externí osy

5.3 Continuous Feature Operation (ostatní kontinuální operace)

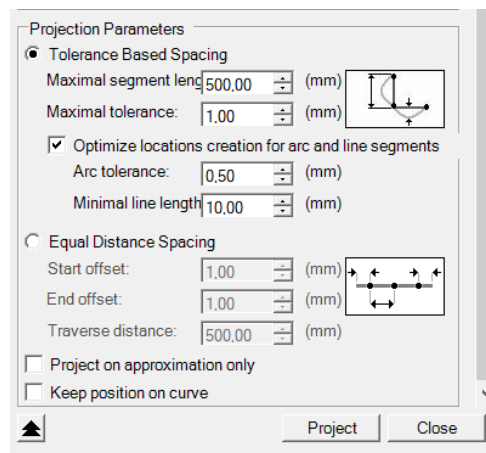
Tvorba dráhy z křivky

Potřebujete-li vytvořit proces při kterém robot bude následovat vámi vytvořenou křivku, lze tak učinit pomocí funkce **Create Continuous Mfg from Curve**  (**Process** → **Continuous**). Funkce je ukázána v praktické ukázce v kapitole 7.2.

Project Continuous Mfg (**Process** → **Continuous**)

Při nastavování parametrů této funkce postupujeme podobně jako při tvorbě dráhy pro obloukové svařování popsané v předchozí kapitole. Funkci lze vybrat až po vytvoření jakékoliv Continuous operace.

- **Tolerance Based Spacing** → rozteč na základě zadané tolerance a vzdáleností
- **Equal Distance Spacing** → systém rozmístí polohy rovnoměrně, dle zadaných parametrů
- **Project on approximation only** → projektování na přesné geometrii, pouze JT soubory s přesnou geometrií
- **Keep position on curve** → udržuje polohu na křivce, při zaškrtnutí této volby robot kopíruje křivku i pokud není ve stejné rovině jako součást



Obrázek 5.9 - Project Continuous Mfg

Location Manipulator

Již zmíněno v kapitole 4.2.2. Zde ji ale rozepisují podrobněji, protože velmi usnadňuje práci s kontinuální operací.

Nachází se na liště nástrojů v grafickém okně pracovního prostředí. Používá se pro přesun polohy pozic. Tato funkce je užitečná, když chcete přesouvat několik pozic dohromady, i když jsou různého typu. Při zapnutém **Follow Mode** robot skočí do vybrané pozice.

V horním levém rohu se nachází následující možnosti:

- **Reset Absolute Location** 

Resetuje absolutní umístění vybraných pozic


- **Snap to Maximum Allowed** 

Pokud pozice překročí svoji maximální hodnotu, funkce je nastaví na maximální povolenou hodnotu.




- **Flip Locations** 

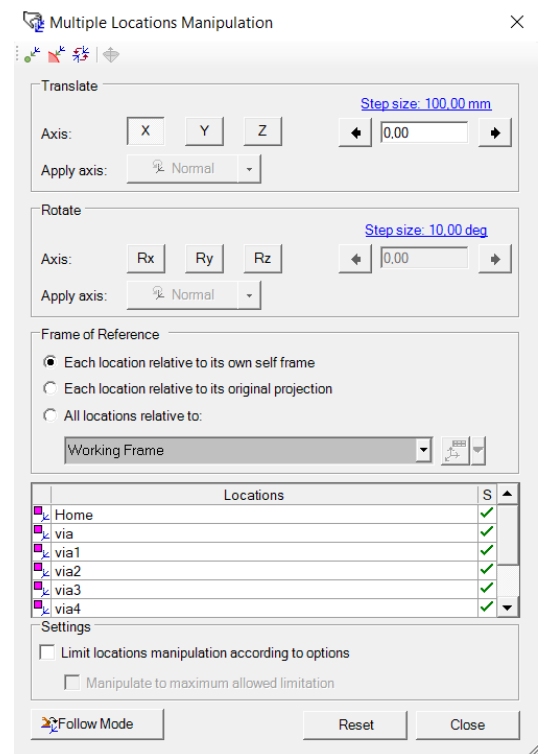
Otočí pozici kolem své osy o 180°

- **Show Location Limits in Graphic Viewer**

Ikona kužele  bude zobrazována v grafickém prohlížeči. Představuje povolené odchylky pro pozice.

U pozic se můžete setkat s těmito případy:

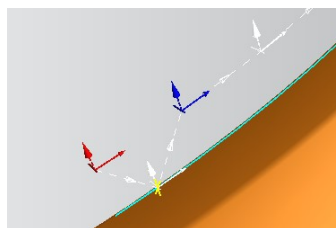
-  - Systém přesunul pozici dle pokynu
-  - Pozice mimo dosah
-  - Systém přesunul pozici co nejdále, ale systémové omezení brání k plnému vykonání pokynu



Obrázek 5.10 - Hromadná úprava pozic

Insert location Inside Seam (Process → Continuous)

Slouží k vložení nové pozice mezi již vygenerovanou dráhu (obrázek 5.11). Použitelný pouze pro kontinuální procesy, kdy je tvořen „šev“ například pro ARC proces svařování. Zachová parametry procesu.






Obrázek 5.11 - Nová pozice kontinuálního procesu

Split Seam Operation (Process → Continuous)

Pomocí tohoto příkazu rozdělíte operaci (typu seam – šev) na dvě nezávislé operace. Pokud například dojde k problému s dosažitelností robotu, můžete operaci rozdělit na dvě a druhou část přiřadit jinému robotu. Je možné rozdělit pouze operaci obsahující 3 a více pozic.

Vodní paprsek – Water Jet (Process)

RobotExpert může simulovat nepřetržité operace pohybující se podél dané křivky. Například při řezání vodním paprskem, je na robotu namontovaná tryska s vodním paprskem řezajícím specifikovanou část podél definované křivky (tzv. makro cesta). Během nepřetržitého provozu lze tak zapnout/vypnout proudění vody nebo i aplikaci barev při lakování.

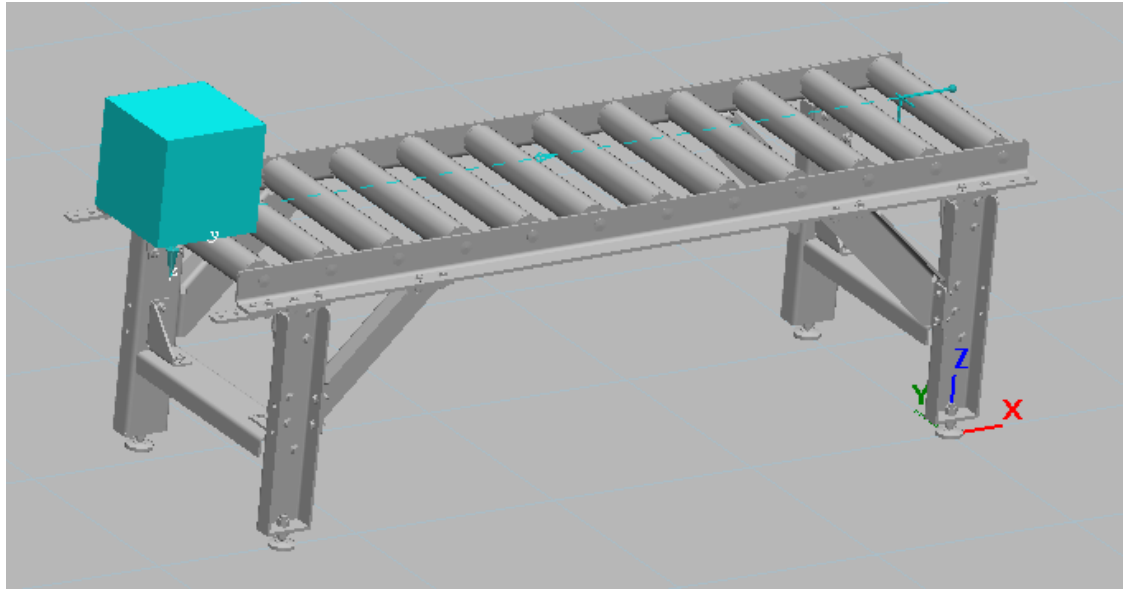
- **Create Macro**  –Vytvoří obdélníkové, kruhové makro nebo makro z jediné pozice nepřetržité operace
- **Edit Macro**  –Vymazání nebo změna makra přidruženého k nepřetržitému provozu
- **Display Macros**  – Zobrazí nebo skryje makra pro vybranou operaci

Vizualizace makro projekce je podporována pouze pro části s přesnou geometrií. Nepřetržitá makra vyžadují vlastní xml informace s konfigurací maker. Pokud má zvolená operace přiřazený robot s kontrolérem, který podporuje typy pohybu, je typ pohybu automaticky nastaven a zobrazen v **Process Type**, který se nachází v **Path editoru**. V opačném případě zde musíte zvolit správný **Process type** (dostupné hodnoty jsou získány z xml souboru). Pokud se to nepodaří, systém vyhodnotí umístění makra jako běžnou pozici.

V současné době jsou kontinuální makra podporované pouze pro roboty ABB, které používají vlastní kontroléry.

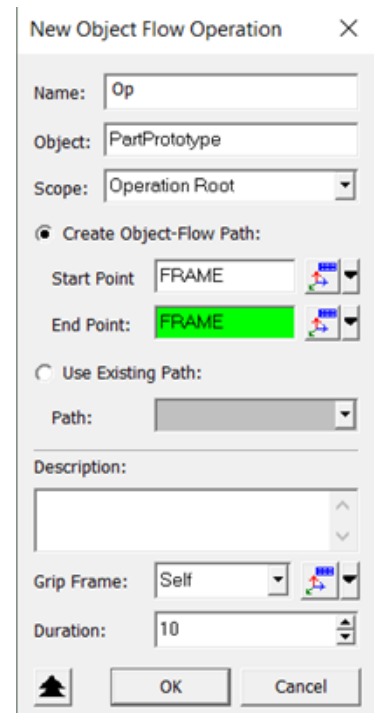
5.4 Dopravník

V této kapitole si ukážeme, jak vygenerovat pohyb objektu po dopravníku. Bohužel RobotExpert nemá funkci pro definici dopravníku jako je tomu například v softwaru Process Simulate, ale i tak existuje způsob jak si „dopravník“ vytvořit. Nejdříve si vložte nebo vytvořte objekt, který se bude pohybovat a také dopravník, na kterém budete vytvářet cestu pohybu (obrázek 5.12).



Obrázek 5.12 - Dopravník s objektem manipulace

Otevřeme si **Operation** → **Create Operation** → **New Operation** → **New Object Flow Operation**. Otevře se okno, ve kterém nastavíme položku **Object = Objekt, který se bude pohybovat**, dále nastavíme odkud kam se má objekt pohybovat. To lze učinit dvěma způsoby. Prvním je **Create Object-Flow Path**, kde je potřeba nastavit **Start Point** a **End Point**, pokud vám nevyhovují body na vašem dopravníku, vytvořte si souřadné systémy podle potřeby. Druhým způsobem je použít již vytvořenou dráhu (**Use Existing Path**), pokud jste již vytvářeli dráhu v rámci jiné operace. V kolonce **Grip Frame** vybírejte, bod objektu, který se bude pohybovat po dráze. Položka **Duration** nastavuje dobu trvání pohybu objektu po dopravníku (obrázek 5.13). Po potvrzení se ve vašem stromu operací (**Operation Tree**) vytvoří nová operace.



Obrázek 5.13 - Object Flow Operation

6 Off-line programování

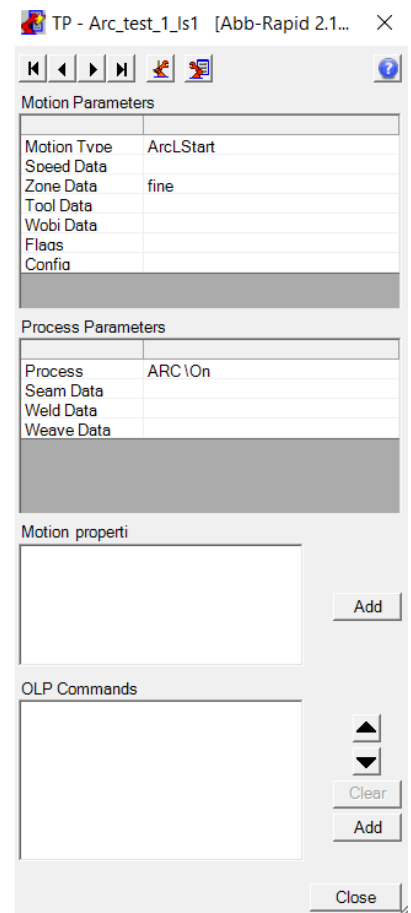
Používá se zkratka OLP a jedná se o proces navrhování, modifikace a testování robotického programu v externím systému, bez potřeby skutečného robotu. Jakmile je program vytvořen, lze jej stáhnout z externího systému a nahrát do vlastního řídicího systému robotu. Tento proces je možné i obrátit a stáhnout program z robotu do externího systému a provádět jeho úpravy.

6.1 Teach Pendant (pokročilé robotické příkazy)

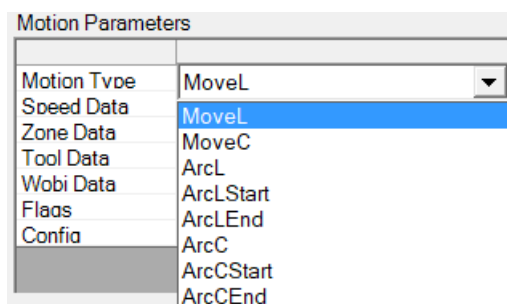
Ve stromu objektů nebo v path editoru klikněte pravým tlačítkem myši na pozici, které chcete přiřadit nová data a zvolte **Teach Pendant**. Nebo přes záložku **Robot** → **OLP**.

V Teach Pendantu (Obrázek 1.7obrázek 6.1) lze nastavovat pro každou pozici informace o rychlosti, zóně, druhu pohybu jakým se má robot do bodu dostat a další (viz Motion Parameters obrázek 6.2).

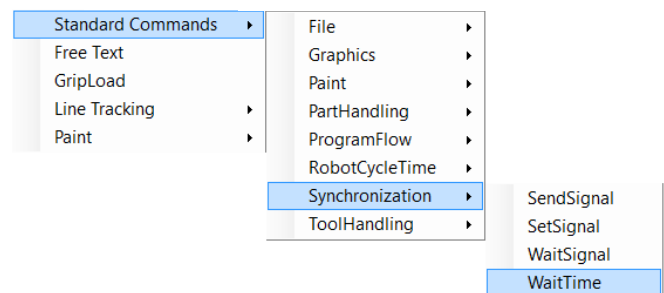
Důležitou funkcí jsou i tzv. **OLP příkazy**, které se nachází ve spodní části funkce Teach Pendant. Přes tlačítko **Add** se dostaneme do nabídky, kde lze nalézt příkazy jako je WaitTime, signály, uchopování objektů, aktivace lakování, zmizení objektů, kreslení, výměna nástrojů a mnoho dalších (obrázek 6.3).



Obrázek 6.1 - Teach Pendant



Obrázek 6.2 - Motion Parameters



Obrázek 6.3 - OLP příkazy

6.2 Virtuální kontrolér robotu

Jak získat příslušný model robotu a převést jej do naší knihovny jsme už probrali 2.5.3. Důležité je však vědět, že složka robotu s příponou COJT obsahuje i soubor **motionparameters.e**. Jedná se o textový soubor obsahující informace o pohybových parametrech robotu, jako je rychlost, zrychlení/zpomalení, zóna, konfigurace a časy rozjezdů a ustálení. Ujistěte se, že je tento soubor obsažen ve složce robotu. Pokud vás zajímá, jak taktový textový soubor vypadá, můžete si jej otevřít a prohlížet například v poznámkovém bloku z vaší knihovny.

V softwaru RobotExpert je několik rozdílných typů řízení kontroléru:

- **DCM** (známé také jako MOP) – Dynamicky řízený model
- **DCM** (s RRS) – Realistická robotická simulace

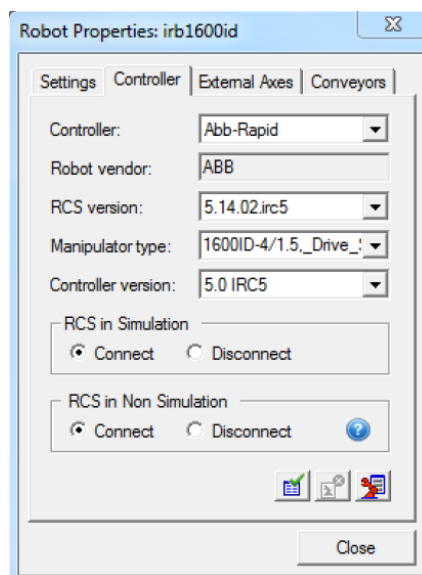
RRS je mezinárodní norma specifikující komunikační algoritmy a protokoly pro každého robota zastoupených v simulačním prostředí. V podstatě se jedná o rozhraní, které RobotExpert používá pro přístup k virtuálnímu kontroléru robotu (RCS).

RCS modul je virtuální řídicí jednotka robotu (kontrolér), který ví, jak se vyrovnat se vstupy a výstupy RRS. Je vyvinutý výrobcí robotů tak, aby se choval stejně jako skutečný řídicí systém. Na RCS modul pro každou značku robotů zvlášť je potřeba mít zakoupenou licenci.

Kontrolér robotu může být nastaven na záložce **Robot** → **Setup** → **Robot Properties** → **Controller**. Jak kontrolér nastavit pro úspěšné stažení programu si blíže rozebereme v kapitole 6.3 a 6.4.

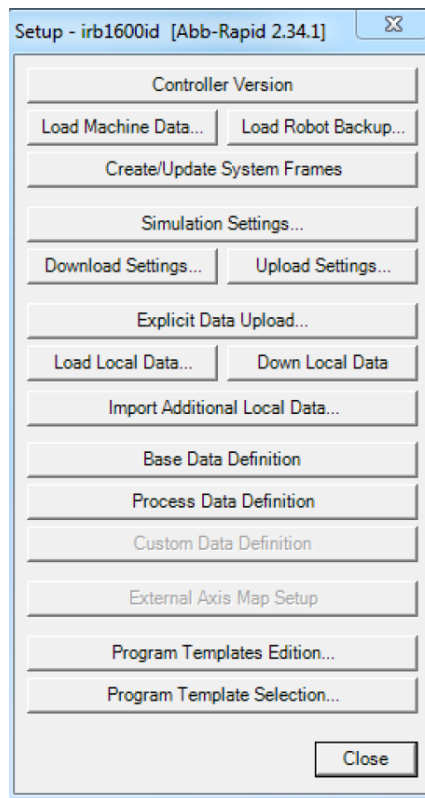
6.3 Optimalizace programu před nahráním do reálného robotu

Prvním krokem je přiřazení kontroléru k robotu. Budu zde pracovat s kontrolérem od ABB, ale postup je totožný i pro ostatní výrobce až na nějaké názvy a rozdílné parametry. Označte si robot ve stromu objektů a vyberte funkci **Robot Properties (Robot → Setup)**. Zde přejděte do záložky **Controller** (obrázek 6.4). Vyplňte příslušná pole a zaškrtněte u obou RCS možností **Connect**. Pokud nemáte nainstalované RCS moduly (musíte mít i licenci pro jejich použití), všechna pole s tím související budou zašedlá. Následně ve spodní části v okně **Robot Properties** zvolte pole **Validate**, čímž připojíte RCS a ověříte jeho správnost propojení mezi klientem a serverem. V případě úspěšného propojení se objeví hláška „**Controller settings are valid. RCS was initialized**“. Lze pokračovat i bez RCS modulu, avšak je potřeba počítat s tím, že pokud budete přenášet program bez RCS modulu, nebude reálný pohyb odpovídat simulaci.



Obrázek 6.4 - Nastavení kontroléru [11]

Přejděte do funkce **Robot Setup (Robot → Setup)** obrázek 6.5. V této funkci lze nastavit všechny potřebné parametry pro definování pohybů a akcí robotu. Každý kontrolér má jinak definované parametry (již od výrobce).



Obrázek 6.5 - Robot Setup [11]

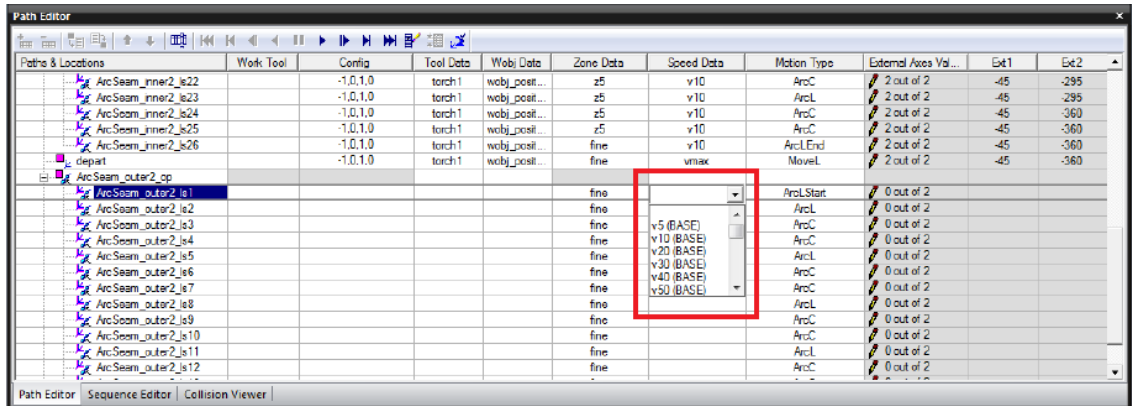
- Load Machine Data** → nahraje data z RCS modulu (rychlost, data nástroje, a další)
- Simulation Settings** → mimo jiné se zde zapíná RRS (viz kapitola 6.2)
- Base Data Definition** → lze upravovat jednotlivé základní informace, jako je rychlost, zóna, wobj, nástroj apod.
- Process Data Definition** → definování dat pro jednotlivé procesy (např. svařování)
- Weld Comp. Settings** → nastavení offsetu v ose Z (pro svařování)

V dalším kroku nastavíme kontrolér **Controller Settings (Robot → Setup)** (obrázek 6.6). Zde přiřadíte ke každému robotu správný kontrolér, manipulátor a RCS nastavení. V záložce **Connect to RCS** zaškrtněte všechna pole. Opět zde lze ověřit správnost propojení tlačítkem **Validate**. Nastavení je stejné jako dříve, přes funkci Robot Properties, s tím rozdílem, že zde můžete nastavovat kontroléry pro více robotů současně.

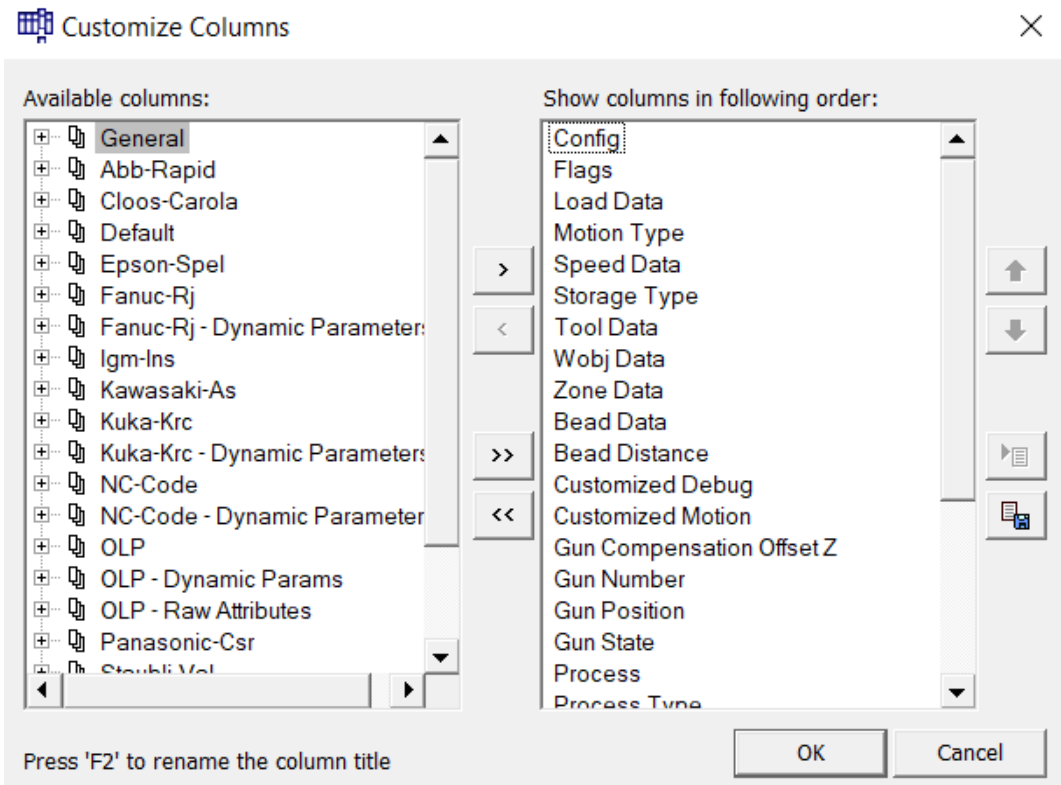
Robot Name	Controller	Vendor	RCS Version	Manipulator Type	Controller Version	Connect to RCS		RCS Status
						Simulation	Non Simulation	
irb1600id	Abb-Rapid	ABB	5.07.01.irc5	2400/10WJ_Inverted_Type_B_RC1_N	5.0 IRC5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-----
irb6620_220_150_m2004	Abb-Rapid	ABB	5.14.02.irc5	IRB_6620-150/2.2_Drive_System_04	5.0 IRC5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-----

Obrázek 6.6 - Controllers Settings [11]

V **Path Editoru** můžete měnit rychlosti, zóny a další data pro každou pozici zvlášť (obrázek 6.7). Pro zobrazení těchto dat vyberte v editoru tlačítko **Customize Columns** a zde si přiřadíte jaké informace chcete v editoru zobrazovat. Například podle druhu kontroléru (obrázek 6.8).



Obrázek 6.7 - Path Editor změny [11]



Obrázek 6.8 - Customize Columns

Po provedených úpravách doporučuji aktualizovat parametry v path editoru a spustit znovu simulaci a zkontrolovat kolize. Aktualizaci jednotlivých parametrů lze udělat ručně, přepsáním parametrů u každého řádku, nebo využitím funkce **Auto Teach**. Tato funkce je jednou z funkcí, která se nachází na liště path editoru. Jejím použitím aktualizujete automaticky veškeré parametry v průběhu běžící simulace. Po tomto kroku již můžete přejít ke stáhnutí vytvořeného programu do robotu.

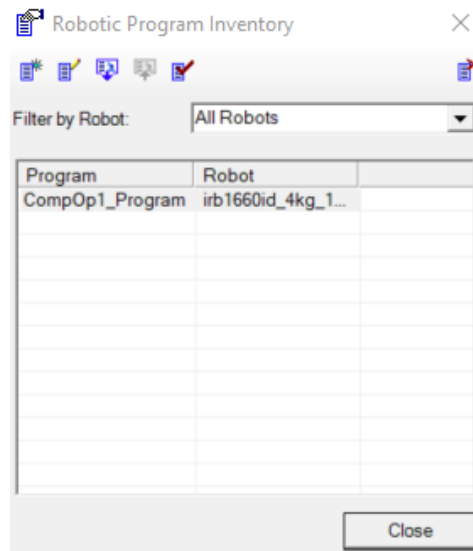
6.4 Přenos programu z RobotExpert do robotu a obráceně

Robot → **Program** → **Robotic Program Inventory** (obrázek 6.9)

Zapamatujte si následující:

Downloading → stažení programu z RE do robotu

Uploading → nahrání programu z robotu do RE



Obrázek 6.9 Robotic Program Inventory

Stažení programu z RE do robotu

Označte si robot ve stromu objektů a otevřete **Robotic Program Inventory**. Nachází se zde několik možností, vyberte první **Create New Program**. Zde je potřeba přiřadit robot a zvolit si název programu. Po potvrzení se vám v inventáři objeví nový program. Tento program lze editovat v **Path Editoru**. Pro stažení programu zvolte možnost **Download to Robot** a vyberte cílovou složku pro uložení.

Pro stažení složené operace (Compound Operation) využijte nástroj **Create program for Compound Operation (Robot → Program)**.

Nahrání programu z robotu do RE

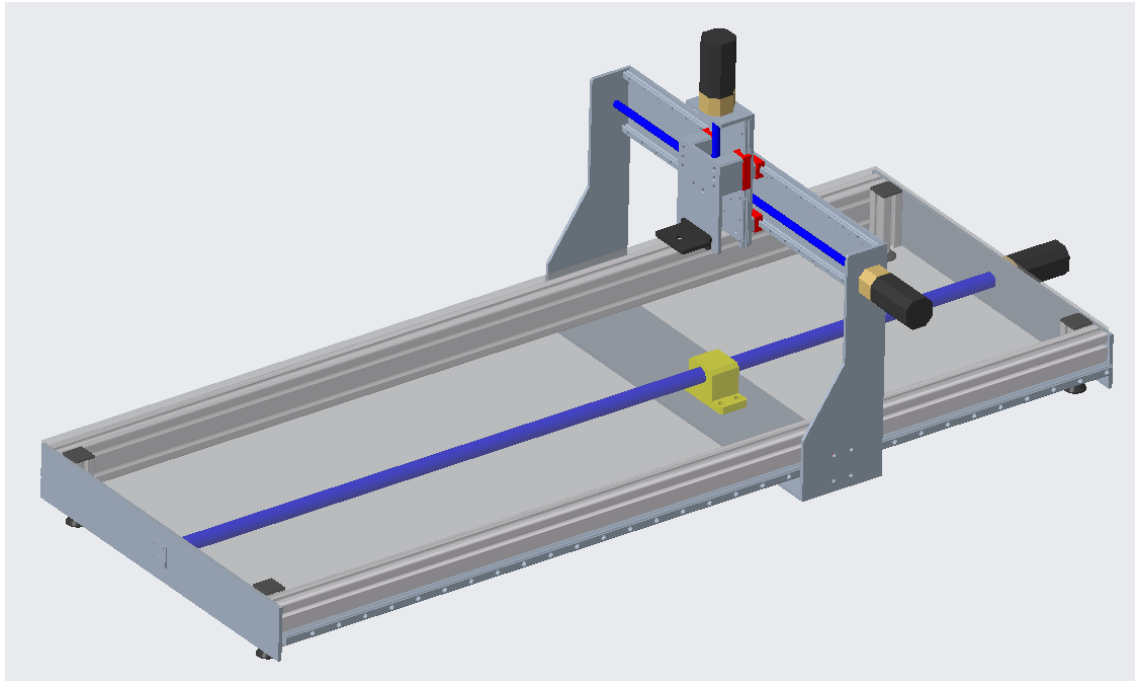
Postup je obdobný. Otevřete si **Robotic Program Inventory**. Vyberte robot, pro který se bude stahovat program. Zvolte **Upload Program**. Nahráný program se objeví v inventáři a ve stromu operací.

7 Praktické příklady

V této kapitole bude vysvětlena tvorba simulací kompletních pracovišť. Každý příklad doprovází i video s ukázaným postupem.

7.1 Vlastní zařízení TTT

Příloha E obsahuje video postup a CAD model manipulátoru se třemi stupni volnosti. Jedná se o tři translace (TTT) viz obrázek 7.1.



Obrázek 7.1 - TTT manipulátor

1. Vložení modelu

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **TTT_CAD.stp**
- Class Types: Base class → **Resource**, Component class → **Robot**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

2. Tvorba kinematiky

- **Set Modeling Scope** (Modeling)
- **Kinematics Editor** (Modeling → Kinematic Device)
- **Create Link** (Vytvořte čtyři Joints, pro každou podsestavu modelu jednu)
- Při tvorbě Linků je pojmenujte jako **Base, X, Y, Z**

- Vytvořte **Joint j1** mezi linkem **Base a X**
- Typ → **Prismatic** v ose pohybu, Limity → **Constant (700; -1500)**
- Vytvořte **Joint j2** mezi linkem **X a Y**
- Typ → **Prismatic** v ose pohybu, Limity → **Constant (400; -400)**
- Vytvořte **Joint j3** mezi linkem **Y a Z**
- Typ → **Prismatic** v ose pohybu, Limity → **Constant (180; -80)**
- Ověřte správnost pomocí Joint Jog
- **End Modeling** (Modeling) = uložení do knihovny

3. Tvorba inverzní kinematiky (Definice TCPF robotu)

- **Set Modeling Scope** (Modeling)
- **Kinematics Editor** (Modeling → Kinematic Device)
- **Set Baseframe** → Zvolte bod na součásti Priruba_Z souřadnice (0;300;507,7;180;0;0)
- **Create Toolframe** → **Location** stejný jako u Baseframe
- **Attach to Link** → podsestava **Z**
- **End Modeling** (Modeling) = uložení do knihovny
- Ověřte správnost funkcí Robot Jog

4. Tvorba OM

- **Create New Part** (Modeling → Components)
- **Create Box** (Modeling → Geometry → Solids → Box Creation)
- Název součásti = **OM**, rozměry **150x150x150**, Locate at **TTT_CAD (-10;0;195;0;0;0)**
- **End Modeling** (Modeling) = uložení do knihovny, název **OM**

5. Vložení modelu nástroje

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **TTT_efektor.stp**
- Class Types: Base class → **Resource**, Component class → **Gripper**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

6. Definice nástroje

- **Set Modeling Scope** (Modeling) pro součást TTT_efektor
- **Tool Definition** (Modeling → Kinematic Device)
- **Typ zvolíme Gripper**
- **TCP Frame** → koncový bod robotu
- **Base** → bod pro připojení na interface robotu
(orientace osy **Z** stejná jako TCPF robotu)
- **Gripping Entities** → zvolte spodní plochu přísavky
- **End Modeling** (Modeling) = uložení do knihovny

7. Připojení nástroje na robot

- Označte robot
- **Mount Tool** (Robot → Tool and Device)
- V kolonce Tool vyberte nástroj TTT_efektor
- Apply
- Při nesprávném připojení se vraťte k definici nástroje a upravte orientaci **Base**
Pro opětovné připojení nástroje doporučuji použít funkci **Reload Component**
(Modeling → Scope) a až poté znovu připojit nástroj k robotu (Mount Tool)
- Ověřte připojení nástroje pomocí **Robot Jog** (TCPF je nyní umístěn na nástroji)

8. Pick and Place aplikace

- **New Pick and Place Operation** (Operation → Create Operation → New Operation)
- Zvolte **Robot** TTT, zvolte **Gripper** TTT_efektor
- **Pick** → Close, **Place** → Open
- **Pick** (střed vrchní plochy **OM**) **souřadnice (-10;0;345;0;0;0)**
- **Place** (souřadnice stejné posunuté jen v ose Y o 700 mm)
souřadnice (-10;700;345;0;0;0)
- **Add Current Location** (Operation → Add Location) = přidání domácí pozice
- Domácí pozici přejmenujte jako Home (klávesa F2)
- Vytvořte pozici s názvem **pick_100** o stejných souřadnicích jako pozice pick jen pozvednuta v ose Z o 100 mm
- Vytvořte pozici s názvem **place_100** o stejných souřadnicích jako pozice place jen pozvednuta v ose Z o 100 mm
- Proved'te test dosahu robotu → **Reach Test** (Robot → Reach)
Jednotlivé pozice musí mít stejnou orientaci os X a Y jako TCPF robotu.
Robot je schopen pouze translačních pohybů!

- Vložte vytvořenou operaci **TTT_PNP_Op** do **Path Editoru**
- V **Path Editoru** poskládejte pozice následovně:
 - Home
 - Pick_100
 - Pick
 - Pick_100
 - Place_100
 - Place
 - Place_100
 - Home
- Spusťte simulaci (Play Simulation Forward)

9. Tvorba animace

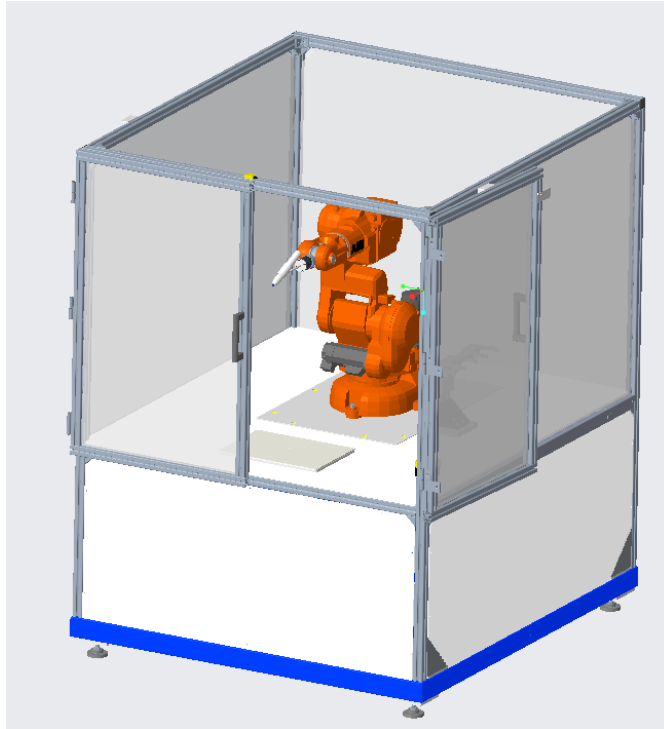
- **AVI Recorder** (Operation → Documentation)
- Zvolte název, rozlišení videa a cestu pro uložení a potvrďte OK
- Spusťte simulaci
- Po proběhlé simulaci zvolte **Stop Recording**, video se uloží

10. Uložte vytvořenou práci

- **Save Cell (CTRL+S)**

7.2 Buňka Fix

Jedná se o buňku s robotem, jehož efektor tvoří fix s tlačnou pružinou. Na podložce je magnety připevněn papír, na který robot kreslí. Dráha robotu je vytvořena podle křivek skici. **Příloha B** obsahuje video postup a CAD model (obrázek 7.2). Autorem CAD modelu je Ing. Michal Vocetka.



Obrázek 7.2 - Buňka Fix

1. Vložení modelu buňky

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **bunka_fix.stp**
- Class Types: Base class → **Resource**, Component class → **Pedestal**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

2. Vložení robotu IRB140

- **Insert Component** (Modeling→Components) = vložit z knihovny
- Pokud v knihovně chybí, stáhní ze stránek ABB → **robot IRB140_m2004c**
- Přemístí robot na podstavec podle děr pro šrouby (**Relocate, Placement Manipulator**)

3. Vložení modelu nástroje

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **efektor_fix.stp**
- Class Types: Base class → **Resource**, Component class → **Gun**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

4. Definice nástroje

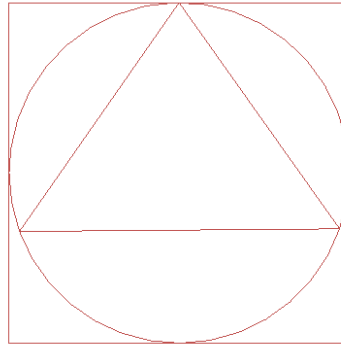
- **Set Modeling Scope** (Modeling) pro součást **efektor_fix**
- **Tool Definition** (Modeling → Kinematic Device)
- **Typ zvolíme Gun**
- **TCP Frame** → koncový bod robotu
Při umísťování vezměte v úvahu, že Fix má v sobě pružinu, která vyvíjí tlak na papír. Špička fixu je tvarově náročná, je tedy dobré vytvořit si **Frame** a označit ho jako **TCPF**.
- **Base** → bod pro připojení na interface robotu
(orientace os stejná jako TCPF robotu)
- **End Modeling** (Modeling) = uložení do knihovny

5. Připojení nástroje na robot

- Označte robot
- **Mount Tool** (Robot → Tool and Device)
- V kolonce Tool vyberte nástroj **efektor_fix**
- Apply
- Při nesprávném připojení se vraťte k definici nástroje a upravte orientaci **Base**
Pro opětovné připojení nástroje doporučuji použít funkci **Reload Component** (Modeling → Scope) a až poté znovu připojit nástroj k robotu (Mount Tool)
- Ověřte připojení nástroje pomocí **Robot Jog** (TCPF je nyní umístěn na nástroji)

6. Tvorba křivky pro dráhu

- **Create New Part** (Modeling → Components)
- **Create Box** (Solids → Geometry), název → papír, rozměry 420x297x5 mm
- **Create Circle (Curves)** vytvořte kruh poloměru 130 mm umístěný na střed papíru
- **Create Polyline (Curves)** vytvořte čtverec o straně 260 mm (umístěný na střed papíru)
- **Create Polyline (Curves)** vytvořte trojúhelník umístěný do kruhu dle libosti (viz obrázek 7.3).

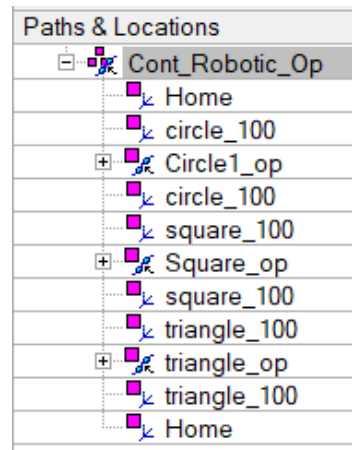


Obrázek 7.3 - Skica pro dráhu robotu

- Označte křivky a zvolte **Create Continuous Mfgs from Curves** (Process → Continuous)
- Zvolte všechny křivky a jako Assign to part vyberte tvořený díl → PartPrototype
- Zobrazte si **Mfg Viewer**, uvidíte zde vytvořené Mfg z vašich křivek
- **End Modeling** (Modeling) = uložení do knihovny

7. Tvorba dráhy z křivky

- **New Compound Operation (Operation → Create Operation)**
- **New Continuous Feature Operation**
- Zvolte **Robot** irb140_m2004c_r1, zvolte **Tool** efektor_fix
- Přiřaďte **Continuous Mfgs** → Circle, Square a Triangle z Mfg Vieweru
- **Project Continuous Mfg** (Process → Continuous)
- Parametry zvolte vhodně vzhledem k rozměrům křivky a označte **Keep position on curve**
- Vytvořte si domácí pozici (**Add Current Location**)
- Vložte vytvořenou operaci **CompOp** do **Path Editoru** (Add operations to Editor)
- **Mezi jednotlivými křivkami vytvořte přesun efektoru nad papírem**
- **Strukturu programu doporučuji volit následovně (viz obrázek 7.4), hodnota 100 za názvem znamená posunutí v ose Z o 100 mm (nad papírem)**



Obrázek 7.4 - Struktura programu

- V případě potřeby upravte orientaci pozic tak ať se efektor nepřetáčí do kolize s buňkou, jedna z možností, jak upravit orientaci je pomocí nástrojů v sekci **Operation** → **Edith Path**, nebo nástrojem **Single or Multiple Locations Manipulation**, který naleznete na pracovní liště
- Spusťte simulaci (Play Simulation Forward)

8. Efekt kreslení

- Nástroj **Tool Centre Point Tracker** (Robot → Analysis) tvoří čáru projeté dráhy robotu
- Funkci ale nespustíme přímo, přidáme ji přes OLP příkaz tak aby se vykreslovala pouze při zvolených pohybech
- V **Path Editoru** klikněte pravým tlačítkem myši na bod **Circle_Is1** a zvolte **Teach Pendant**
- Vyberte **Add** → Standart Commands → Graphics → **TCPTracker**
- Zvolte barvu a přiřaďte **Start TCPF Tracker**
- V bodě **Circle_Is5** postupujte stejným způsobem a přiřaďte k němu **Pause TCPF Tracker**
- Stejně tak pro čtverec, pro bod **Square_Is1** přiřaďte **Resume TCPF Tracker**
- Pro **Square_Is5** opět **Pause TCPF Tracker**
- Stejně tak pro Trojúhelník, **Triangle_Is1** přiřaďte **Resume TCPF Tracker**
- Pro **Triangle_Is4** přiřaďte **Stop TCPF Tracker**
- Spusťte simulaci
- Vykreslená čára se ukládá do stromu objektů
- Po skončení zvolte **End Modeling** pro uložení čáry do knihovny

9. Tvorba animace

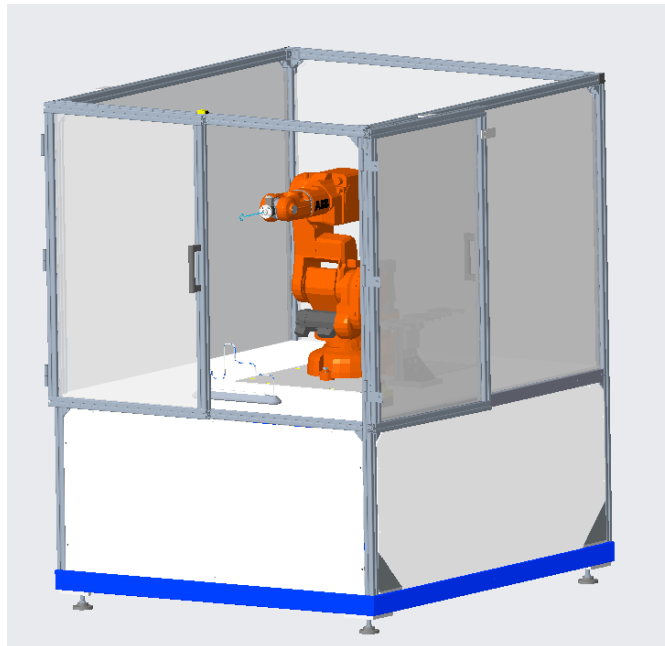
- **AVI Recorder** (Operation → Documentation)
- Zvolte název, rozlišení videa a cestu pro uložení a potvrďte OK
- Spusťte simulaci
- Po proběhlé simulaci zvolte **Stop Recording**, video se uloží

10. Uložte vytvořenou práci

- **Save Cell (CTRL+S)**

7.3 Buňka Drát

Jedná se o buňku s robotem, jehož efektor tvoří z drátu vytvořené očko. Očko efektoru robot navleče na zahýbaný drát a kopíruje jeho tvary tak, aby se okraje oka nedotýkala drátu. Dráha robotu je vytvořena podle vygenerované křivky drátu. **Příloha A** obsahuje video postup a CAD model (obrázek 7.5). Autorem CAD modelu je Ing. Michal Vocetka.



Obrázek 7.5 - Buňka drát

1. Vložení modelu buňky

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **bunka_drat.stp**
- Class Types: Base class → **Resource**, Component class → **Pedestal**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

2. Vložení robotu IRB140

- **Insert Component** (Modeling→Components) = vložit z knihovny
- Pokud v knihovně chybí, stáhní ze stránek ABB → **robot IRB140_m2004c**
- Přemístí robot na podstavec podle děr pro šrouby (**Relocate, Placement Manipulator**)

3. Vložení modelu nástroje

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **efektor_ocko.stp**
- Class Types: Base class → **Resource**, Component class → **Gun**
- Options: **Insert Component** (ANO),
Create a monolithic JT file (ANO - entire assembly)
- Convert

4. Definice nástroje

- **Set Modeling Scope** (Modeling) pro součást **efektor_ocko**
- **Tool Definition** (Modeling → Kinematic Device)
- **Typ zvolíme Gun**
- **TCP Frame** → koncový bod robotu (střed očka)
- **Base** → bod pro připojení na interface robotu (**orientace os** stejná jako TCPF robotu)
- **End Modeling** (Modeling) = uložení do knihovny

5. Připojení nástroje na robot

- Označte robot
- **Mount Tool** (Robot → Tool and Device)
- V kolonce **Tool** vyberte nástroj **efektor_ocko**
- Apply
- Při nesprávném připojení se vraťte k definici nástroje a upravte orientaci **Base**
Pro opětovné připojení nástroje doporučuji použít funkci **Reload Component** (Modeling → Scope) a až poté znovu připojit nástroj k robotu (Mount Tool)
- Ověřte připojení nástroje pomocí **Robot Jog** (TCPF je nyní umístěn na nástroji)

6. Tvorba 2D křivky z modelu pro dráhu

- **Create New Part** (Modeling → Components)
- **Set Modeling Scope** (Modeling) pro vloženou buňku
- Najedťte na konec ve stromu objektů, naleznete zde součást **00_sestava**
- Součást vyjměte (**CTRL+X**) a vložte do vytvořeného PartPrototype (**CTRL+V**)
- Drát je nyní samostatný a bude se s ním lépe pracovat
- **Create 2D Outline** (Modeling → Geometry)
- Křivku vytvořte z drátu (součást wire)
- Vytvořenou křivku umístěte do osy drátu
- Označte křivky a zvolte **Create Continous Mfgs from Curves** (Process → Continuous)
- Zobrazte si **Mfg Viewer**, uvidíte zde vytvořené Mfg z vašich křivek
- **End Modeling** (Modeling) = uložení do knihovny

7. Tvorba dráhy z křivky

- **New Compound Operation** (Operation → Create Operation)
- **New Continous Feature Operation**
- Zvolte **Robot** irb140_m2004c_r1, zvolte **Tool** efektor_ocko
- Přiřaďte **Continous Mfgs** z Mfg Vieweru
- **Project Continous Mfg** (Process → Continuous)
- Parametry zvolte vhodně vzhledem k rozměrům křivky a označte **Keep position on curve**
- Vytvořte si domácí pozici (**Add Current Location**)
- Vložte vytvořenou operaci **CompOp** do **Path Editoru** (Add operations to Editor)
- Dráhu upravte tak, aby odpovídala orientaci TCP efektoru
- Doplněte body domácí pozice a příjezd očka k drátu
- Vytvořte kopii všech bodů
- Pořadí zkopírovaných bodů otočíte funkcí **Reverse Operation** (Operation → Edith Path)
- Spusťte simulaci (Play Simulation Forward)

8. Kontrola kolize

- Otevřete **Collision Viewer** (nachází se vedle Path Editoru)
- Zvolte **Collision Set Editor**
- Budeme kontrolovat kolizi mezi efektořem a drátem
- Zapněte **Collision Mode On/Off**
- V záložce **Collision Options** lze zvolit, jak se simulace zachová v případě kolize
- Nastavte si, aby se při kolizi simulace zastavila
- Spusťte simulaci
- Kolizní body upravte

9. Tvorba animace

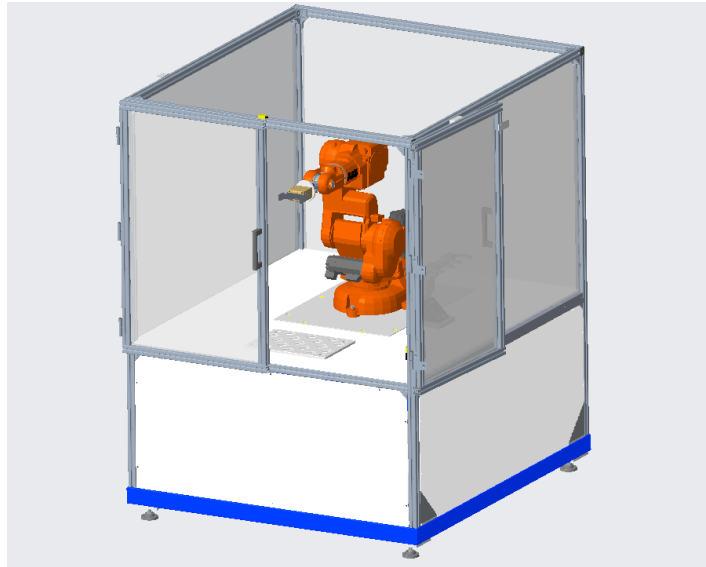
- **AVI Recorder** (Operation → Documentation)
- Zvolte název, rozlišení videa a cestu pro uložení a potvrďte OK
- Spusťte simulaci
- Po proběhlé simulaci zvolte **Stop Recording**, video se uloží

10. Uložte vytvořenou práci

- **Save Cell (CTRL+S)**

7.4 Buňka Kostky (Pick and Place)

Jedná se o buňku s robotem, aktivní část efektoru tvoří dvě čelisti. Robot přemísťuje kostky v zásobníku. **Příloha C** obsahuje video postup a CAD model (obrázek 7.6). Autorem CAD modelu je Ing. Michal Vocetka.



Obrázek 7.6 - Buňka Pick and Place

1. Vložení modelu buňky

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **bunka_kostky.stp**
- Class Types: Base class → **Resource**, Component class → **Pedestal**
- Options: **Insert Component (ANO)**
Create a monolithic JT file (ANO - entire assembly)
- Convert

2. Vložení robotu IRB140

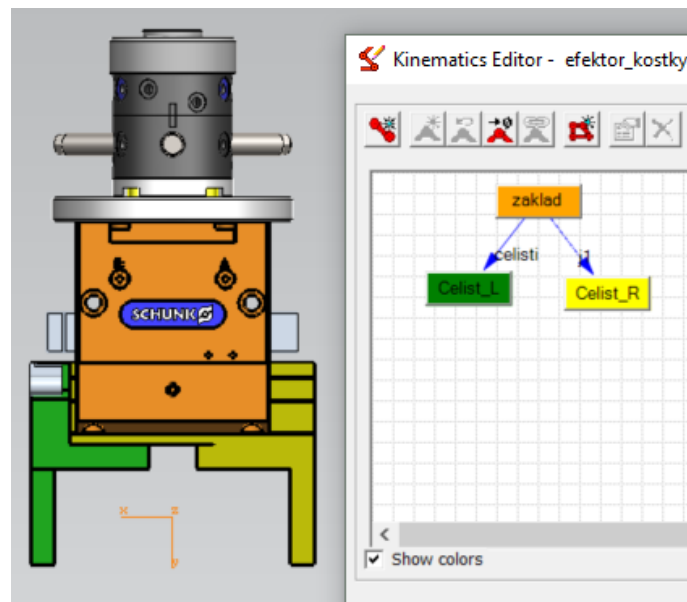
- **Insert Component** (Modeling→Components) = vložit z knihovny
- Pokud v knihovně chybí, stáhní ze stránek ABB → **robot IRB140_m2004c**
- Přemístí robot na podstavec podle děr pro šrouby
(Relocate, Placement Manipulator)

3. Vložení modelu nástroje

- **Convert CAD Files** (File → Import/Export)
- Add → soubor **efektor_kostky.stp**
- Class Types: Base class → **Resource**, Component class → **Gripper**
- Options: **Insert Component** (ANO)
Create a monolithic JT file (ANO - entire assembly)
- Convert

4. Vytvořte kinematiku pro čelisti efektor

- **Set Modeling Scope** (Modeling) pro součást **efektor_kostky**
- **Kinematics Editor** (Modeling → Kinematic Device)
- Vytvořte linky a jointy (viz obrázek 7.7)
- Joint „celisti“ bude prismatický s omezením pohybu od 0 do 49 mm
- Joint „j1“ bude prismatický bude následovat joint „celisti“
- Vytvořte závislost pro j1 (**Joint Dependency** typ Following nebo Joint Function)
- Pro **Joint Function** zadejte **(D(celisti))*1**
- Ověřte správnost nastaveného pohybu (Joint Jog)
- Vytvořte polohy pro uchopování kostek **Pose Editor** (Modeling→Kinematic Device)
- Poloha s názvem **CLOSE** bude mít hodnotu **34 mm**
- Poloha s názvem **OPEN** bude mít hodnotu **20 mm**
- **End Modeling** (Modeling) = uložení do knihovny



Obrázek 7.7 - Kinematika efektoru

5. Definice nástroje

- **Set Modeling Scope** (Modeling) pro součást **efektor_kostky**
- **Tool Definition** (Modeling → Kinematic Device)
- **Typ zvolíme Gripper**
- **TCP Frame** → koncový bod robotu (mezi čelistmi, 20 mm od horní plochy)
- **Base** → bod pro připojení na interface robotu
(orientace os stejná jako TCPF robotu)
- **End Modeling** (Modeling) = uložení do knihovny

6. Připojení nástroje na robot

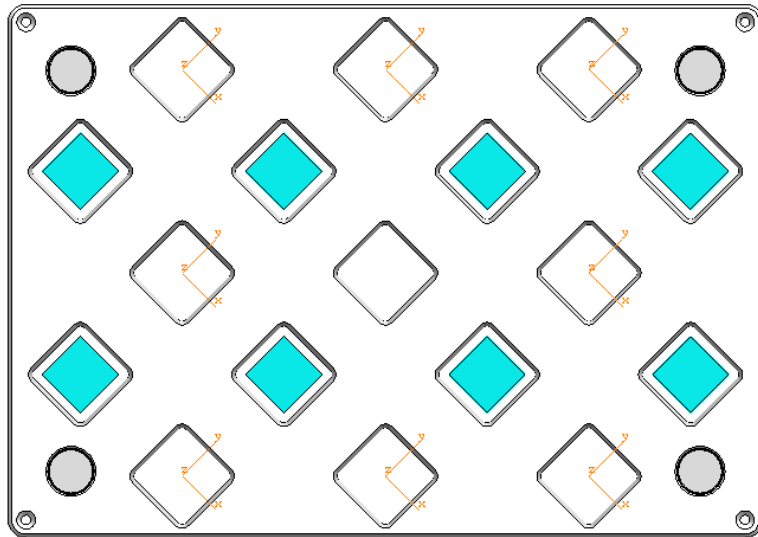
- Označte robot
- **Mount Tool** (Robot → Tool and Device)
- V kolonce **Tool** vyberte nástroj **efektor_kostky**
- Apply
- Při nesprávném připojení se vraťte k definici nástroje a upravte orientaci **Base**
Pro opětovné připojení nástroje doporučuji použít funkci **Reload Component**
(Modeling → Scope) a až poté znovu připojit nástroj k robotu (Mount Tool)
- Ověřte připojení nástroje pomocí **Robot Jog** (TCPF je nyní umístěn na nástroji)

7. Vymodelujte objekt manipulace – kostky

- **Create Box** (Modeling → Solids)
- **Rozměry** 30x30x30mm
- Kostku přemístěte do připraveného zásobníku
- **End Modeling** (Modeling) = uložení do knihovny
- Duplikujte kostku do dvou řad, v každé řadě 4
- **Duplicate** (Modeling → Layout) s krokem ose X a Y vždy o 113,4 mm
- Rozmístění kostek znázorňuje obrázek 7.8

8. Připravte si body pro tvorbu dráhy

- Vytvořte souřadné systémy (body) do středu otvorů pro odkládání kostek
- Rozmístění bodů znázorňuje obrázek 7.8
- Body i kostky, vhodně pojmenujte, pro přehlednější tvorbu programu



Obrázek 7.8 - Rozmístění kostek

9. Tvorba Pick and Place

- **New Compound Operation (Operation → Create Operation)**
- **New Pick and Place Operation**
- Zvolte **Robot** irb140_m2004c_r1, zvolte **Tool** efektor_kostky
- Pro Pick a Place kolonku přiřadte odpovídající **OPEN/CLOSE** polohy
- Definujte body pro **Pick** (na kostce) a **Place** (vytvořený souřadný systém)
- Tvorbu Pick and Place Operace opakujte pro všech 8 kostek
- Body z operací vyjměte a vložte bod jednu jedinou pick and place operaci
- Vytvořte body nad pick i place pozice pro lineární nájezd robotu
- Pozice v path editoru vhodně uspořádejte
- Spusťte simulaci (Play Simulation Forward)

10. Tvorba animace

- **AVI Recorder (Operation → Documentation)**
- Zvolte název, rozlišení videa a cestu pro uložení a potvrďte OK
- Spusťte simulaci
- Po proběhlé simulaci zvolte **Stop Recording**, video se uloží

11. Uložte vytvořenou práci

- **Save Cell (CTRL+S)**

8 Závěr

V rámci této práce vzniklo několik praktických úloh, kde jsou rozebírány konkrétní robotické operace. Tyto úlohy jsou vždy doprovázeny video-postupem práce. Jednotlivé funkce jsou popisovány tak, aby je pochopil i člověk, který nemá žádné zkušenosti s podobnými programy. Vezměte v úvahu i fakt, že RobotExpert je komplexní program a během jednoho školního roku není možné detailně popsat všechny funkce a jejich návaznosti. Věřím však tomu, že jsou v práci všechny důležité činnosti dostatečně vysvětleny a že po jejím přečtení člověk nabyde správných návyků pro práci se softwarem a případné nedostatky si již zvládne dohledat sám.

Se softwarem se po krátkém navyknutí pracuje poměrně příjemně. Je však znát absence některých funkcí, jako je například přímá podpora pro práci s dopravníky, bodové svařování a další logické bloky a podpůrné činnosti. Věřím tomu, že většina firem i navzdory vyšší ceně investuje spíše do softwaru Process Simulate kvůli jeho možnosti práce s komplexnějšími modely. Pro uživatele je taktéž náročné zorientovat se v doplňujících licencích, které potřebuje pro konkrétní operace. I přes zmíněné nedostatky jsem toho názoru, že není špatné se nejdříve naučit pracovat se softwarem RobotExpert a následně přejít na rozšířenější Process Simulate.

V rámci práce jsem také nahrál jednoduchý program do reálného robotu. V době psaní práce nebyla k dispozici licence na RCS modul. Bez tohoto modulu nelze simulovat přesný pohyb a rychlosti robotu. Porovnání simulace s robotem je ukázáno na videu viz **příloha D**.

Za ukázkový výstup práce považuji video, na kterém je simulace zjednodušeného centra robotiky, které je součástí **přílohy I**. Video ukazuje, čeho je schopen dosáhnout student po pečlivém přečtení této práce za předpokladu, že si popisované funkce souběžně prakticky zkouší.

Poděkování

Za odborné vedení, ochotu, lidský přístup a čas, který mi věnoval při zpracovávání diplomové práce děkuji svému vedoucímu Ing. Tomáši Kotovi, Ph.D. Zároveň děkuji Ing. Michalu Vocetkovi za pomoc při odzkoušení programu v reálném robotu a Ing. Danielu Huczalovi za vyjednávání licencí a podpůrných materiálů. Nakonec bych rád poděkoval své partnerce MUDr. Veronice Machalové za projevenou trpělivost a výpomoc s korekcí textu a rodině za podporu v celém průběhu mého studia.

Diplomová práce prezentuje znalosti nabyté v rámci řešení projektu: „Digitální dvojčata robotických systémů a jejich verifikace“, reg. č.: „SP2019/69“ financovaných ze státního rozpočtu České Republiky, na kterém jsem se v průběhu magisterského studia podílel.

9 Seznam použité literatury

- [1] KRYS V. *PaP RTP-Úvodní přednáška*. Moodle VŠB – TU Ostrava [online]. Dostupné z: https://lms.vsb.cz/pluginfile.php/729779/mod_resource/content/1/P_%C3%9Avod%20do%20PaP RTP.pdf
- [2] *IFR: Počet robotů vzroste do roku 2020 na dvojnásobek*. Robotic journal: svět robotiky a automatizace v průmyslu. Praha: Tech Media Publishing s.r.o, 2018. č.02/2018, s.8, ISSN 2533-4425.
- [3] *Průmysl 4.0* – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Pr%C5%AFmysl_4.0
- [4] *Internet of Things (1. část) - Všichni hovoří o IoT, ale co to vlastně je?* | SOS electronic. [online]. SOS electronic s.r.o. 1991 [cit. 15.05.2019]. Dostupné z: <https://www.soselectronic.cz/articles/no-name/internet-of-things-1-cast-vsichni-hovori-o-iot-ale-co-to-vlastne-je-2034>
- [5] *Siemens na MSV představí digitální řešení pro celý životní cyklus stroje* | Tiskové centrum SIEMENS. 302 Found [online]. Siemens, s.r.o. 1996 [cit. 15.05.2019]. Dostupné z: <https://www.siemens.cz/press/siemens-na-msv-predstavi-digitalni-reseni-pro-cely-zivotni-cyklus-stroje>
- [6] *Tecnomatix brožura*. [online]. [cit. 15.05.2019]. Dostupné z: http://www.itscz.net/doc/brozury_file/tecnomatix-brozura-cz-31.pdf
- [7] *Tecnomatix, Longterm Technology Services*. [online]. 2019 Longterm Technology Services. [cit. 15.05.2019]. Dostupné z: <https://longtermtec.com/home-tecnomatix/>
- [8] *Svařování roboty - AXIOM TECH s.r.o.* [online]. AXIOM TECH s.r.o. [cit. 15.05.2019]. Dostupné z: <https://www.axiomtech.cz/25302-svarovani-roboty>
- [9] *Siemens PLM Software - Learning Advantage*. [online]. 2016 Siemens Product Lifecycle Management Software Inc. [cit. 15.05.2019]. Dostupné z: <https://training.plm.automation.siemens.com/mytraining/index.cfm>

- [10] *IRBP A - ABB Robotics - Workpiece Positioners*. ABB Group - Leading digital technologies for industry [online]. 2019 ABB [cit. 15.05.2019]. Dostupné z: <https://new.abb.com/products/robotics/application-equipment-and-accessories/workpiece-positioners/irbp-a>
- [11] *Robot Expert Quick Start Guide*. Scribd - Read books, audiobooks, and more [online]. 2019 Scribd Inc. [cit. 15.05.2019]. Dostupné z: <https://www.scribd.com/doc/294300095/Robot-Expert-Quick-Start-Guide>
- [12] *GTAC: Siemens PLM Software*. [online]. Siemens AG 2002 [cit. 15.05.2019]. Dostupné z: <https://download.industrysoftware.automation.siemens.com/#download>
- [13] *Siemens Documentation*. [online]. Dostupné z: https://docs.plm.automation.siemens.com/tdoc/tecnomatix/13.1.2/RBX#uid:index_xid1015930
- [14] *Národní iniciativa Průmysl 4.0*. BusinessInfo.cz - Oficiální portál pro podnikání a export [online]. 1997 [cit. 15.05.2019]. Dostupné z: <https://www.businessinfo.cz/cs/clanky/narodni-iniciativa-prumysl-40-71386.html#!&chapter=1>
- [15] *RobotExpert Getting Started*. YouTube [online]. Dostupné z: https://www.youtube.com/playlist?list=PL1m1vu8_quoCkCBRbz6OhfiRy0sL66jvW
- [16] *Siemens PLM Community*. [online]. 2019 Siemens Product Lifecycle Management Software Inc [cit. 15.05.2019]. Dostupné z: <https://community.plm.automation.siemens.com/>

10 Seznam příloh

Uvedené přílohy odpovídají elektronické podobě na přiloženém DVD.

Příloha A Buňka - drát

bunka_drat.STP
efektor_ocko.STP
video-postup_drat.MP4

Příloha B Buňka - fix

bunka_fix.STP
efektor_fix.STP
video-postup_fix.MP4

Příloha C Buňka – kostky

bunka_kostky.STP
efektor_kostky.STP
video-postup_kostky.MP4

Příloha D Přenos programu

RE-program.MOD
Odzkouseni-programu.MP4

Příloha E TTT – vlastní zařízení

TTT.STP
TTT_efektor.STP
video-postup_TTT.MP4

Příloha F Příkazy pro logicko-matematické vyjádření kinematiky

prikazy_kinematiky.PDF

Příloha G Tvorba kinematiky paralelogramu

Paralelogram.STEP

Příloha H Polohovadlo ABB

Irbpa-250_D1000_IRC5_rev02_CAD.STP

Příloha I Simulace UCR

UCR.MP4

Příloha J Odkazy na videa

Odkazy-videoa.PDF