



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

NÁVRH, SIMULACE A ŘÍZENÍ ROBOTICKÉHO PRACOVNÍŠTĚ PRO BODOVÉ A KONTINUÁLNÍ SVAŘOVÁNÍ

DESIGN, SIMULATION AND CONTROL OF ROBOTIC CELL FOR SPOT AND CONTINUAL WELDING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jozef Holíček

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Václav Kaczmarczyk, Ph.D.

BRNO 2017



Diplomová práce

magisterský navazující studijní obor **Kybernetika, automatizace a měření**
Ústav automatizace a měřicí techniky

Student: Bc. Jozef Holíček

ID: 154734

Ročník: 2

Akademický rok: 2016/17

NÁZEV TÉMATU:

Návrh, simulace a řízení robotického pracoviště pro bodové a kontinuální svařování

POKyny PRO VYPRACOVÁNÍ:

- 1) Vytvořte model linky sestávající ze dvou robotů s vzájemně se překrývajícími drahami ramen.
- 2) Nastudujte, popište a realizujte plánování drah těchto robotů, předcházení kolizím při statické i dynamické simulaci.
- 3) Nastudujte a popište další nástroje, které aplikace nabízí pro modelování a simulaci stacionárních robotů, zejména s ohledem na kontinuální a bodové svařování.
- 4) Vytvořte aplikaci pro PLC, která bude simulované robotické pracoviště řídit.
- 5) Proveďte ověření správné funkce simulovaného modelu a aplikace.

DOPORUČENÁ LITERATURA:

Firemní dokumentace AXIOMTECH pro Tecnomatix Process Simulate

Termín zadání: 6.2.2017

Termín odevzdání: 15.5.2017

Vedoucí práce: Ing. Václav Kaczmarczyk, Ph.D.

Konzultant:

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Táto práca sa zaoberá modelovaním a simuláciou výrobného procesu, ktorý pozostáva z niekoľkých stacionárnych robotov. Tento model výrobného procesu je vytvorený v programe Tecnomatix Process Simulate.

V prvej kapitole je popísaný princíp digitálnej továrne a stručný úvod do virtuálnej výroby.

V druhej kapitole je podrobne popísaná práca s programom Tecnomatix Process Simulate a postup potrebný pre návrh výrobnéj linky kontinuálneho a bodového zvarovania.

V tretej kapitole je popísaný riadiaci program PLC SIMATIC 300 a vizualizačný program.

KLÚČOVÉ SLOVÁ

Tecnomatix, Process Simulate, Digitálna výroba, Stacionárny robot, PLC, Simulácia procesu zvarovania

ABSTRACT

This thesis deals with modeling and simulation of production process, which consists of a few stationary robots. This model of production process is made in program Tecnomatix Process Simulate.

Principle of digital manufacturing and virtual production is described in the first chapter.

The second chapter describes how to work with program Tecnomatix Process Simulate and how to make a design of production process of continuous and spot welding.

The controlling program of PLC SIMATIC 300 and the program for process visualization are described in the last chapter.

KEYWORDS

Tecnomatix, Process Simulate, Digital manufacturing, Stationary robot, PLC, Simulation of welding process

HOLÍČEK, Jozef. *Návrh, simulace a řízení robotického pracoviště pro bodové a kontinuální svařování*. Brno, 2017, 65 s. Diplomová práca. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedúci práce: Ing. Václav Kaczmarczyk, Ph.D.

VYHLÁSENIE

Vyhlasujem, že som svoju diplomovú prácu na tému „Návrh, simulace a řízení robotického pracoviště pro bodové a kontinuální svařování“ vypracoval samostatne pod vedením vedúceho diplomovej práce, s využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práva tretích osôb a taktiež som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných či majetkových. Som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúceho autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka Českej republiky č. 40/2009 Sb.

V Brne

.....

podpis autora

POĎAKOVANIE

Rád by som poďakoval vedúcemu diplomovej práce Ing. Václavovi Kaczmarczykovi, Ph.D. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci. Taktiež by som chcel poďakovať spoločnosti AXIOM TECH s.r.o. za poskytnutie licenčných súborov k programu Tecnomatix, ktoré mi veľmi pomohli v mojej práci.

V Brne

.....

podpis autora

OBSAH

Úvod	9
1 Digitálna továreň	10
2 Tecnomatix Process Simulate	12
2.1 O programe Tecnomatix Process Simulate	12
2.2 Vloženie robota	14
2.3 Modelovanie nových komponentov	16
2.3.1 Návrh stola pre kontinuálne zváranie	16
2.3.2 Návrh stola pre bodové zváranie	20
2.3.3 Konvertovanie dokumentu JT na formát COJT	21
2.3.4 Nastavenie kinematiky zváracieho stola	23
2.3.5 Súčiastka určená na kontinuálne zváranie	29
2.3.6 Súčiastka určená na bodové zváranie	29
2.3.7 Úprava vákuového uchopovacieho nástroja	30
2.4 Simulácia zvárania	32
2.4.1 Kontinuálne zváranie	32
2.4.2 Bodové zváranie	38
2.5 Virtuálne riadenie aplikácie	44
3 Riadenie modelu linky pomocou PLC	50
3.1 OPC Server	50
3.1.1 VIPA OPC-Editor	51
3.2 Konfigurácia PLC	54
3.2.1 Riadiaci program	55
3.3 Vizualizácia simulačného procesu	58
4 Záver	61
Literatúra	63
Zoznam symbolov, veličín a skratiek	64
A Obsah priloženého CD	65

ZOZNAM OBRÁZKOV

1.1	Priemyselné roboty.[6]	10
2.1	Portfólio programu Tecnomatix.[4]	12
2.2	Okno príkazu Controller Settings.	14
2.3	Okno príkazu Mount Tool.	15
2.4	Manipulátor od firmy FÖRSTER.[5]	16
2.5	Base - základná konštrukcia zvaracieho stola.	17
2.6	Arm - rotujúce rameno zvaracieho stola.	17
2.7	Plate - rotujúca pracovná doska zvaracieho stola.	18
2.8	Navrhnutý zvarací stôl pre kontinuálne zvaranie.	18
2.9	Odporúčené nastavenia pre uloženie dokumentu JT.	19
2.10	Navrhnutý zvarací stôl pre bodové zvaranie.	20
2.11	Príkaz na konvertovanie JT dokumentov.	21
2.12	Nastavenie pre konvertovanie JT dokumentov.	22
2.13	Spustenie modelovacieho režimu zvaracieho stola.	23
2.14	Kinematics Editor pre definovanie skupín odkazov.	24
2.15	Kinematics Editor a Joint Properties pre definovanie spojení.	25
2.16	Nastavenie osí pre určenie rotačného pohybu.	26
2.17	Ukážka rotačného pohybu na zvaracom stole.	27
2.18	Definovanie externých osí robota.	28
2.19	Definovanie kinematiky stola pre bodové zvaranie.	28
2.20	Časti prípravku určeného na kontinuálne zvaranie.	29
2.21	Časti prípravku určeného na bodové zvaranie.	29
2.22	Vákuový nástroj určený na uchopovanie.	30
2.23	Nastavenie uchopovacieho nástroja pre robota.	31
2.24	Generátor kontinuálnych procesov.	32
2.25	Generovanie projekcie pre proces kontinuálneho zvarania.	33
2.26	Nastavovanie hodnôt pre externé osi robota pri zvaracom procese.	34
2.27	Nastavovanie natočenia robota pri procese kontinuálneho zvarania.	35
2.28	Program kontinuálneho zvarania.	36
2.29	Pridanie OLP Commands.	37
2.30	Prelínanie dráh oboch robotov pri procese kontinuálneho zvarania.	38
2.31	Nastavenie doby pre bodové zvaranie.	38
2.32	Definovanie bodu určujúceho miesto bodového zvarania.	39
2.33	Generovanie projekcie bodov určených pre bodové zvaranie.	40
2.34	Vytvorenie novej operácie bodového zvarania.	41
2.35	Zmena natočenia hlavy robota pre bodové zvaranie.	41
2.36	Program bodového zvarania.	42

2.37	Prelínanie dráh oboch robotov pri procese bodového zvárania.	43
2.38	Funkcia New Object Flow Operation.	44
2.39	Material Flow Viewer.	45
2.40	Sequence Editor v Line Simulation Mode.	46
2.41	Riadiace signály robota.	47
2.42	Priradovanie operácií ku programu robota.	47
2.43	Nastavenie komunikácie OPC Server v programe Tecnomatix.	48
2.44	Signály využívané na riadenie simulácie.	49
3.1	Bloková schéma komunikácie.	50
3.2	Vytvorenie siete v programe OPC-Editor.	51
3.3	Nastavenie PLC stanice v programe OPC-Editor.	52
3.4	Online test OPC Servera.	53
3.5	HW konfigurácia riadiaceho PLC.	54
3.6	Tabuľka použitých premenných.	55
3.7	Vývojový diagram funkcie FC3 - prepínanie medzi krokmi programu.	57
3.8	Definovanie komunikácie v programe WinCC flexible Advanced.	58
3.9	Definovanie premenných v programe WinCC flexible Advanced.	59
3.10	Tvorba vizualizačného programu v prostredí WinCC flexible Advanced.	59
3.11	Runtime vizualizačného programu.	60
4.1	Zhotovené pracoviská kontinuálneho a bodového zvárania.	61

ÚVOD

Cielom tejto diplomovej práce je oboznámenie sa s modelovaním a simulačným riadením stacionárnych robotov v systéme Tecnomatix Process Simulate.

Výsledkom tejto práce bude návod, pomocou ktorého bude možné zostrojiť simuláciu kontinuálneho alebo bodového zvaracieho procesu. V oboch vytvorených simulačných pracoviskách by sa mali dráhy programovaných robotov prekrývať, pričom by nemalo dochádzať ku kolíznym stavom medzi robotmi a predmetmi v ich dosahových vzdialenostiach.

V práci budú popísané všetky nástroje a príkazy programu Tecnomatix Process Simulate, ktoré sú potrebné pre vyhotovenie zvaracích operácií.

Jeden zo simulačných procesov bude riadený prostredníctvom PLC (Programmable Logic Controller) SIMATIC rady 300. Na komunikáciu medzi PLC a riadiacim počítačom so simulačným programom sa bude využívať rozhranie OPC (Object Linking and Embedding for Process Control) servera.

Posledným krokom bude overenie správnej funkcie simulovaných modelov a aplikácií.

V **kapitole č. 1** je priblížený pojem *Digitálna továreň*. V tejto časti sa práca zaoberá hlavnými cieľmi a výhodami, ktoré virtuálna výroba prináša.

V **kapitole č. 2** je podrobne popísaná práca s vývojovým prostredím Tecnomatix Process Simulate pri zostrojovaní simulácie zvarania. V **časti č. 2.2** je popísaný spôsob vkladania robota, s ktorým budeme v programe pracovať. V **časti č. 2.3** sa práca zaoberá postupom modelovania a “oživovania” nových komponentov využívaných v simulácií. Taktiež sa tu nachádza popis súčiastok určených na zvaranie a úprava pracovného nástroja robota. **Časť č. 2.4.1** popisuje spôsob vytvorenia simulácie kontinuálneho zvarania a v ďalšej **časti č. 2.4.2** je popísaný spôsob vytvorenia simulácie bodového zvarania. V **časti č. 2.5** sa nachádza návod popisujúci nastavenie simulácie kontinuálneho zvarania pre možnosť riadenia pomocou PLC.

V **kapitole č. 3** sa práca venuje nastaveniu komunikácie a zhotoveniu riadiaceho programu. V **časti č. 3.1** je popísaný postup vytvorenia OPC servera. V **časti č. 3.2** sa nachádza konfigurácia riadiaceho PLC a popis jeho programu slúžiaceho na ovládanie simulácie kontinuálneho zvarania. V **časti č. 3.3** je popísaný vizualizačný program na overenie funkcie riadiaceho programu.

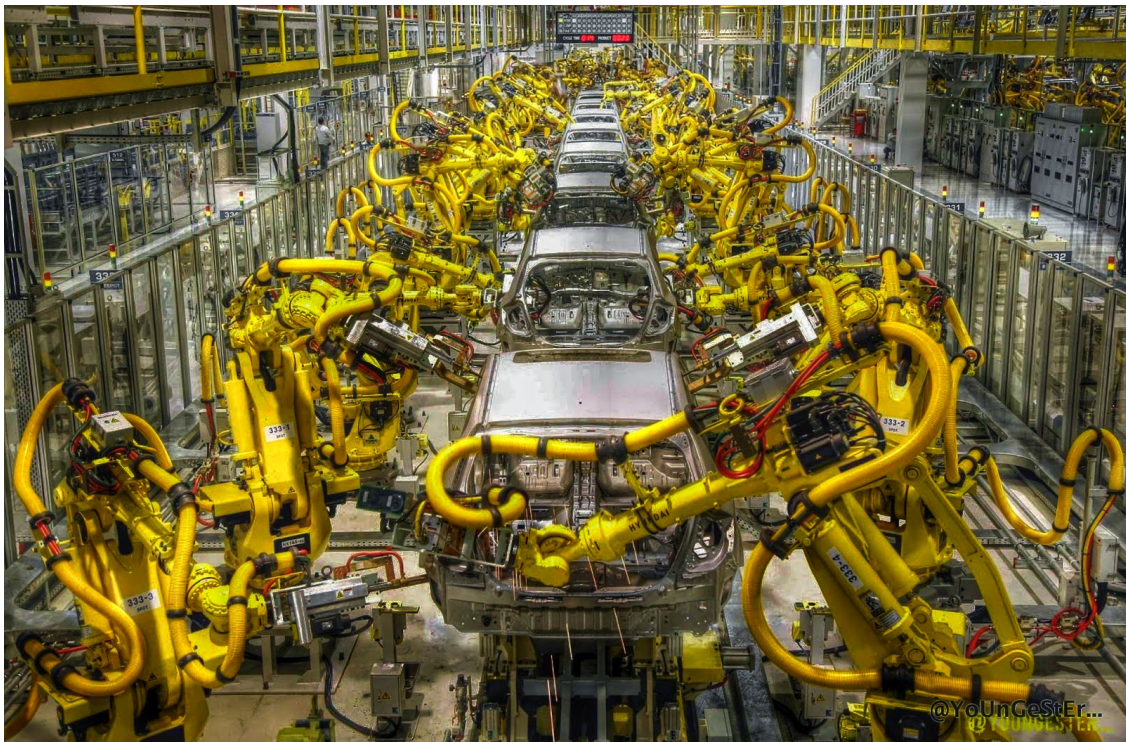
V **kapitole č. 4** je súhrn výsledkov tejto diplomovej práce.

1 DIGITÁLNA TOVÁREŇ

V poslednej dobe sa čoraz viac uvádza do povedomia pojem *Digitálna továreň*. Tento výraz vo všeobecnosti pokrýva veľmi širokú oblasť úkonov spojenú s prípravou nového výrobného procesu vo virtuálnej podobe. Oblasť, v ktorej sa využíva daný systém je veľmi rozsiahla. Začína samotným návrhom jednotlivých komponentov využívaných v procese, pokračuje simuláciou pracovných úkonov vo výrobe a končí spracovaním údajov podniku pre jeho zefektívnenie a skvalitnenie.

Jedna z mnoho definícií digitálnej továrne:

Digitálna továreň je zastrešujúci pojem pre rozsiahlu sieť digitálnych metód, modelov a nástrojov (spoločne so simuláciou a 3D-vizualizáciou), ktoré sú integrované v rámci priebežného dátového managementu. Cieľom je komplexné a systémové plánovanie, projektovanie, overovanie a priebežné zlepšovanie všetkých dôležitých štruktúr, procesov a zdrojov reálnej továrne v súvislosti s jej výrobkami.[1]



Obr. 1.1: Priemyselné roboty.[6]

Takže, digitálna továreň je zobrazenie reálneho projektu podľa požiadaviek zákazníka. Základnou výhodou tohto prístupu k návrhu výrobného procesu je minimalizovanie výskytu rizika vzniknutého na navrhovanom projekte. To sa dá realizovať napríklad včasným zistením chyby v návrhu výrobného procesu, v skoršej fáze plánovania alebo geometrickým overením montážnej sekvencie. Vďaka týmto riešeniam

sme schopní zamedziť či už menším alebo rozsiahlejším zmenám vo výrobe ešte pred jej zahájením a znížiť, poprípade úplne eliminovať značné finančné a časové straty.

Medzi ďalšie výhody patrí znižovanie počtu fyzických prototypov a skrátenie času potrebného na zahájenie nového výrobného procesu. Skrátenie doby spúšťania nového projektu je možné vďaka presnému digitálnemu návrhu výrobných liniek, čím sa zabezpečí paralelná spolupráca zložiek zapojených do vývoja pracoviska. Snahou je teda zhotoviť výsledný výrobný proces vo verzii, ktorá je plne odladená a neobsahuje skryté nežiadúce chyby.

S digitálnou výrobou sa taktiež spája pojem **Virtuálna výroba**. Virtuálna výroba má využitie vo viacerých odvetviach priemyslu. Vďaka digitálnemu návrhu je možné prepojenie výrobných a konštruktérskej úrovne podniku. Týmto prepojením sa zabezpečí včasná spätná väzba v prípade, ak výrobní inžinieri odhalia chybu na navrhovanom komponente z hľadiska vyrobiteľnosti daného výrobku.

Tieto výhody sú spojené nie len s výrobou jednotlivých komponentov, ale aj s návrhom a výrobou celých podnikových hál. Pomocou virtuálneho 3D návrhu je možné simulovanie a hľadanie najvhodnejšej varianty výrobného procesu. Tým sa zvyšuje produktivita a kvalita výroby podniku.[2]

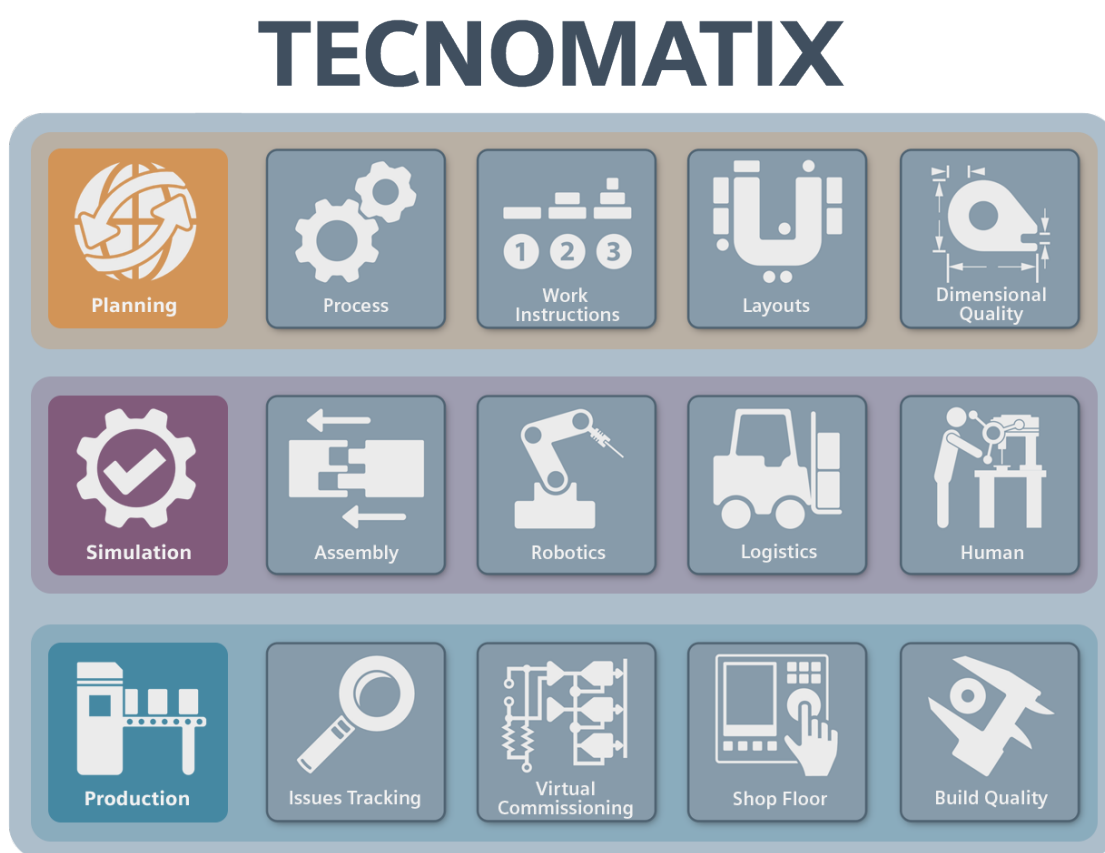
Ďalšou výhodou je možnosť testovania riadiaceho programu už za realizácie výrobných liniek a nie až po jej dokončení. Tým je možné realizovať včasný redizajn výrobných liniek a zamedziť tak časovým a finančným stratám. Cieľom virtuálnej výroby je taktiež minimalizovanie fyzických testov riadiaceho programu výrobných liniek, čím sa skráti čas jej nábehu.

2 TECNOMATIX PROCESS SIMULATE

2.1 O programe Tecnomatix Process Simulate

Tecnomatix Process Simulate je program vyvinutý spoločnosťou Siemens. Základným cieľom tohto programu je tvorba digitálnej továrne. Tecnomatix Process Simulate umožňuje navrhovať, analyzovať, simulovať a predovšetkým optimalizovať výrobné procesy. Výsledkom je teda virtualizovaná výroba, ktorú je možné doladovať a pri následnej implementácii je možné hľadanie chýb a ich včasné odstránenie.

Portfólio programu Tecnomatix je graficky znázornené na obr. 2.1. Zahŕňa viacero aplikácií a je veľmi rozsiahle.



Obr. 2.1: Portfólio programu Tecnomatix.[4]

Nástroj Process Simulate umožňuje predchádzať statickým a dynamickým kolíziám, radenie operácií do správneho poradia pre zefektívnenie montážneho procesu,

návrh montážnej linky a výsledného pracoviska pre zlepšenie ergonómiky a zvýšenia bezpečnosti na pracovisku, či robotizovať proces a zlepšovať spoluprácu medzi človekom a strojom.

Tým sa dostávame k možnostiam programu Tecnomatix Process Simulate, ktorým sa táto práca venuje z veľkej časti, a teda k simulácii statických priemyselných robotov.

Je možné simulovať a overovať viacero robotických operácií ako lakovanie, bodové zvarovanie, kontinuálne zvarovanie, prípadne rôznu manipuláciu s materiálom. Dajú sa simulovať statické a dynamické dosahy robotov, zisťujú sa kolízie medzi robotmi a okolitým prostredím, hľadajú sa optimálne nástroje, s ktorými roboty vykonávajú operácie, optimalizuje sa dosah robotov a ich celkové rozmiestnenie v priestore.

Program Tecnomatix Process Simulate umožňuje taktiež off-line programovanie robotov. Týmto spôsobom je možné zásadne urýchliť nábeh novej výroby, pretože ešte pred fyzickou montážou zhotovíme program, ktorý je následne možné na mieste nahráť do riadiacej jednotky daného robota. Po nahratí sa overí jeho funkčnosť, poprípade sa program doladí drobnými zmenami, čím sa zásadne urýchluje proces programovania robotov oproti konvenčnému ručnému programovaniu priamo na mieste. Týmto spôsobom sa taktiež dajú realizovať zmeny vo výrobnom procese, ktorý už je v chode a po simulácii je možné tento program robota implementovať priamo do reálnej výroby.

2.2 Vloženie robota

Celá zostava robota pozostáva z viacerých častí. Prvou časťou je samotný robot. Druhou časťou je konzola, na ktorú je robot umiestnený. Tretou časťou je nástroj, ktorý sa robotovi umiestňuje na koniec ramena a s týmto nástrojom robot vykonáva určité operácie. Väčšinu 3D modelov komponentov robota používam z knižníc, ktoré nám poskytla firma AXIOM TECH s.r.o.. Je však možné vložiť model robota od ľubovoľného výrobcu, poprípade si užívateľ môže model vytvoriť samostatne.

Použité knižnice pre model robota určeného na kontinuálne zváranie:

- Robot - ABB - irb1600id_4_150__01.cojt
- Konzola - Risers - ma386245s_f01_0019.cojt
- Nástroj - Weld gun - Robacta5000_36_S_type.cojt

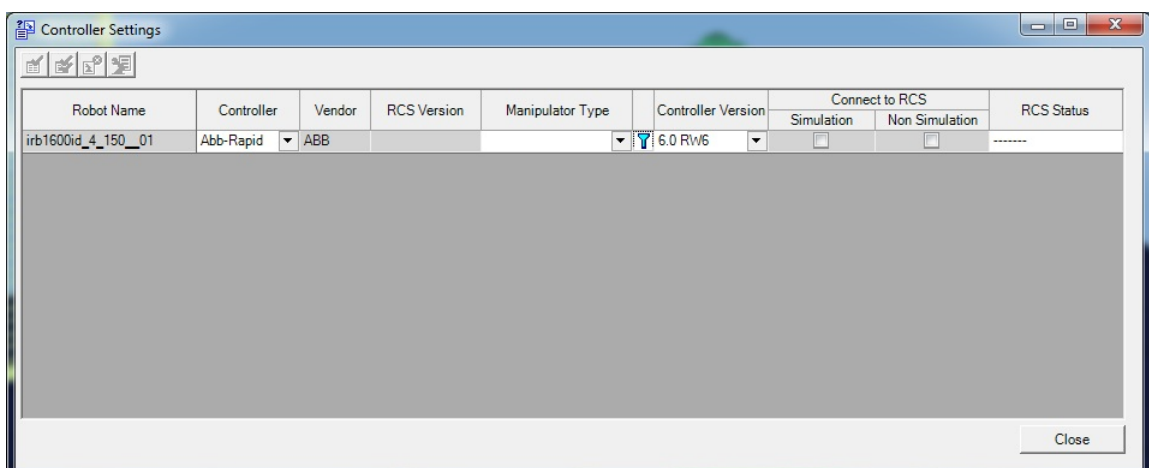
Použité knižnice pre model robota určeného na bodové zváranie:

- Robot - ABB - irb2600_12_165__01.cojt
- Konzola - Risers - ma386245s_f01_0019.cojt
- Nástroj - Spotweld gun - sa_81429b*.cojt

Použité knižnice pre model robota určeného na manipuláciu materiálu:

- Robot - KUKA - kr100pa.cojt
- Konzola - Risers - ma386245s_f01_0019.cojt
- Nástroj 1 - Uchopovací nástroj - box_gripper.cojt
- Nástroj 2 - Uchopovací nástroj - GRIPPER_VACUUM_xholic01.cojt

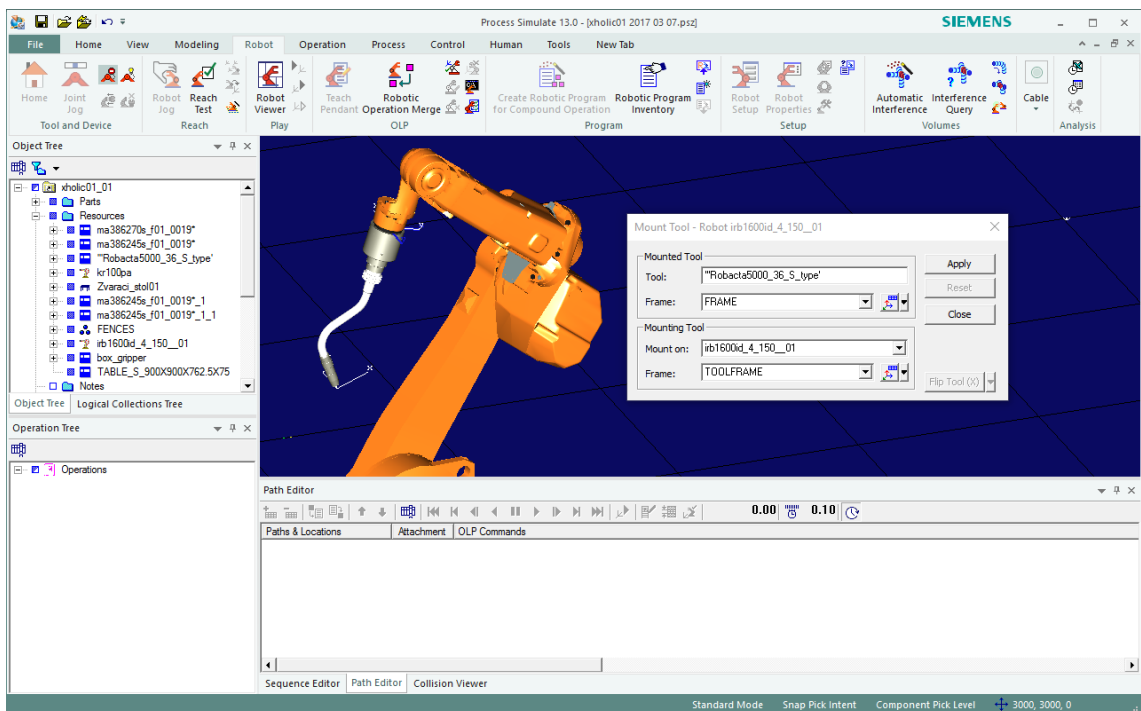
Po vložení všetkých 3D modelov do projektu cez príkaz **Insert Component** si robota umiestnime na podstavnú konzolu. Našou prvou úlohou je nastavenie ovládača daného robota. Toto nastavenie je možné pomocou príkazu **Controller Settings**, ktorého okno sa nachádza na obr. 2.2. Toto nastavenie nám umožňuje



Obr. 2.2: Okno príkazu Controller Settings.

vo výsledku pomocou programu Tecnomatix Process Simulate vygenerovať riadiaci program pre daného robota, podľa nášho simulačného programu. V našom prípade je potrebné definovať dva parametre. Do poľa **Controller** vyberieme z ponuky, v ktorej sa nachádzajú všetky aktuálne nainštalované ovládače v systéme, požadovaný ovládač pre nášho robota. V poli **Controller Version** vyberieme verziu ovládača nášho robota.

Posledný krok je upevnenie nástroja k hlave robota. Po označení robota v Graphic Viewer alebo v Object Tree sa pomocou príkazu **Mount Tool** dostaneme do okna, v ktorom sa definuje nástroj pre robotov. Toto okno je znázornené na obr. 2.3. Do poľa **Tool**, kliknutím ľavého tlačidla myši na objekt vyberieme nástroj pre robota a rovnakým spôsobom určíme v poli **Mount on** robota, na ktorý sa má nástroj upevniť. Pomocou ikony **Create Frame of Reference** je možné upraviť orientáciu nástroja v hlave robota do požadovanej polohy. Tlačidlom **Apply** potvrdíme uchopenie nástroja k robotu.



Obr. 2.3: Okno príkazu Mount Tool.

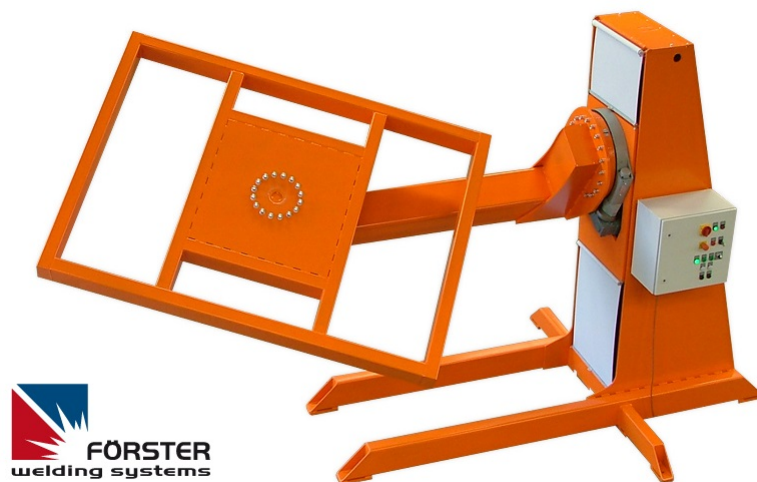
2.3 Modelovanie nových komponentov

Pri návrhu simulovaného procesu môžeme naraziť na dva základné problémy. Prvým je použitie zariadenia, na ktoré výrobca neposkytuje 3D model a druhým je použitie nášho vlastného zariadenia. V oboch prípadoch je potrebné 3D model vyhotoviť, a s týmto modelom budeme následne pracovať v simulačnom programe.

Pre vytváranie CAD (Computer Aided Design) súborov, ktoré zahrňujú konštrukčný návrh súčiastok daného komponentu som využíval program Solid Edge. Solid Edge je grafický softvérový nástroj, v ktorom je možné navrhovať súčiastky v 3D. Základnou výhodou tohto programu je možnosť bezplatnej verzie, ktorú užívateľ nevyužíva na komerčné účely a možnosť uloženia vymodelovaných súčiastok vo formáte JT. Tento formát je následne možné rozbaľiť a ďalej používať v programe Tecnomatix Process Simulate.

2.3.1 Návrh stola pre kontinuálne zváranie

Pri návrhu zváracieho stola, ktorý využívam v simulácii kontinuálneho zvárania, som sa nechal inšpirovať manipulátorom od firmy FÖRSTER. Manipulátor sa nachádza na obr. 2.4. V praxi sa často používajú podobné manipulátory, ktoré otáčajú zváranú súčiastku, a tým šetria čas operácie. Manipulátor pozostáva z troch častí a má dve rotujúce osi. V riadiacej jednotke robota je možné tieto osi nastaviť ako externé osi a priamo ich ovládať.

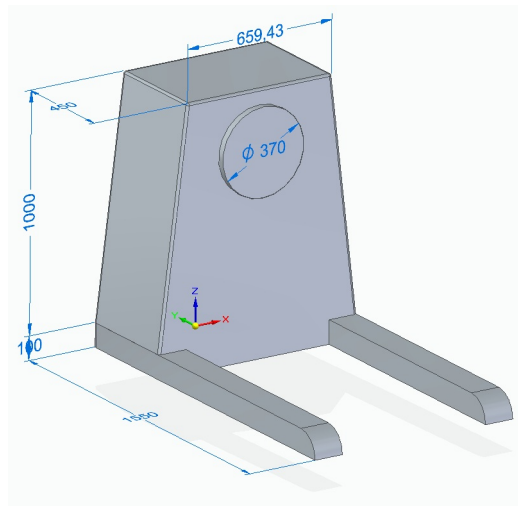


Obr. 2.4: Manipulátor od firmy FÖRSTER.[5]

Ako prvé bolo potrebné vymodelovať jednotlivé časti, z ktorých sa zvärací stôl skladá:

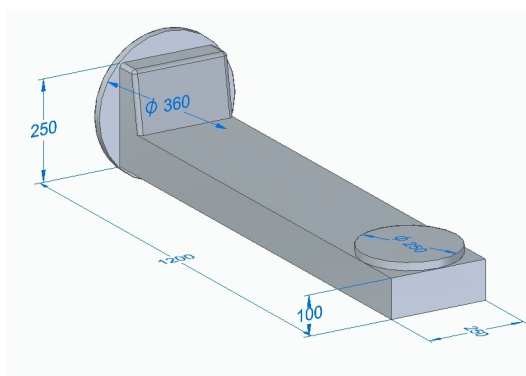
- Base - základná konštrukcia
- Arm - rotujúce rameno
- Plate - rotujúca pracovná doska

Base



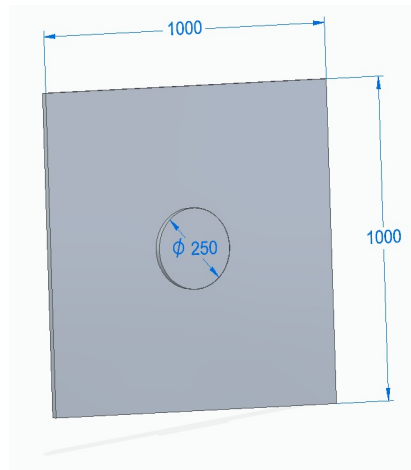
Obr. 2.5: Base - základná konštrukcia zväracieho stola.

Arm



Obr. 2.6: Arm - rotujúce rameno zväracieho stola.

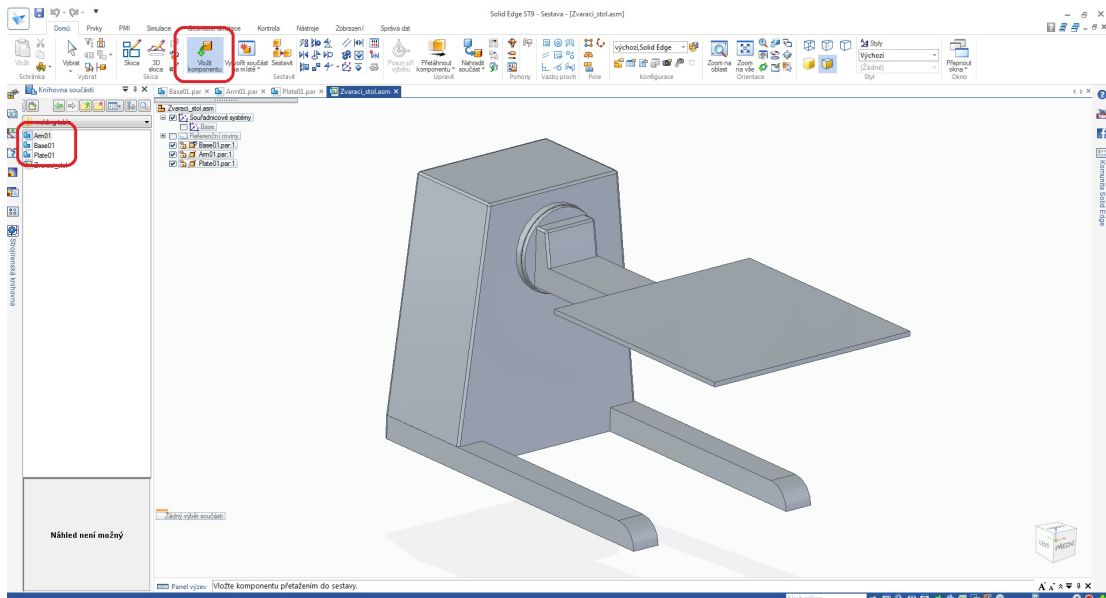
Plate



Obr. 2.7: Plate - rotujúca pracovná doska zváracieho stola.

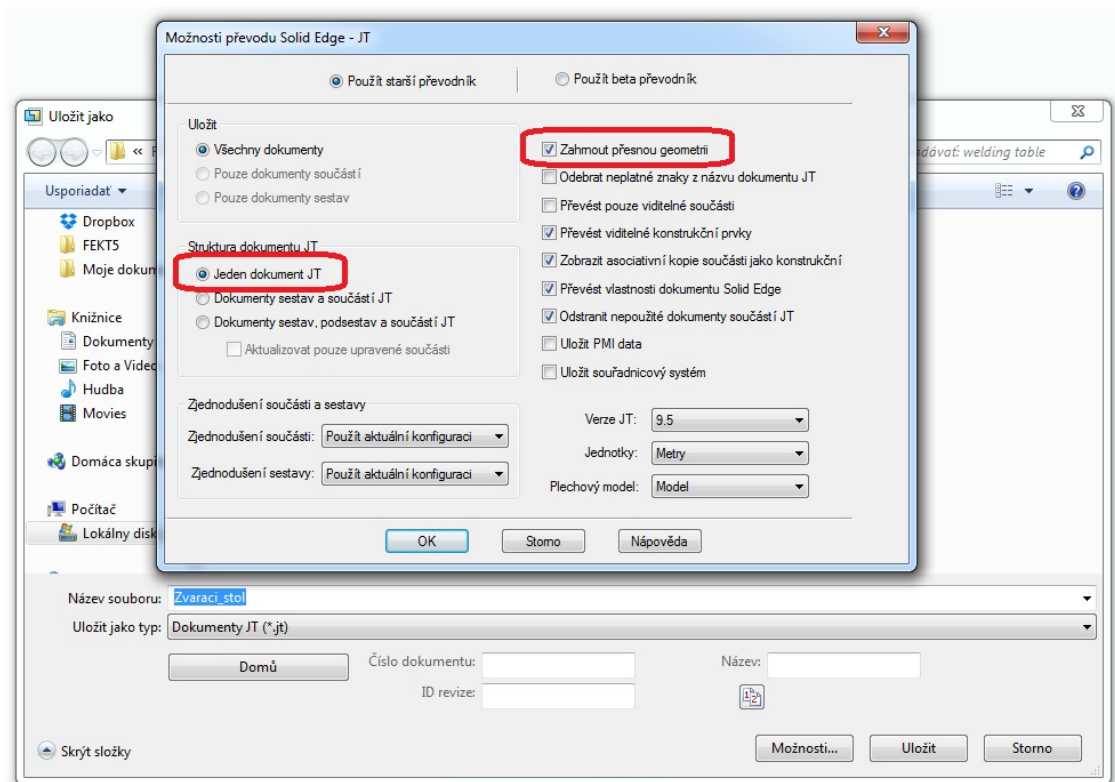
Zvárací stôl pre kontinuálne zváranie

Po vymodelovaní všetkých troch komponentov, z ktorých pozostáva navrhnutý zvärací stôl, je ďalšou úlohou ich uloženie do požadovanej zostavy. Pomocou príkazu **Vložiť komponentu**, vyberieme všetky tri časti a postupne ich rozmiestnime do požadovaného tvaru.



Obr. 2.8: Navrhnutý zvärací stôl pre kontinuálne zváranie.

Ako posledné je potrebné uložiť vymodelovanú zostavu v dokumente JT, aby s ňou bolo možné pracovať v programe Tecnomatix Process Simulate. Na obr. 2.9 sú zobrazené doporučené nastavenia pri ukladaní dokumentu JT. Je dôležité, aby sa výsledný súbor uložil ako jeden dokument, pretože inak by ho nebolo možné v programe Tecnomatix Process Simulate načítať, a taktiež je dôležité do súboru zahrnúť presnú geometriu súčiastky. Tieto nastavenia doporučujem pre všetky súčiastky, s ktorými sa bude ďalej pracovať v programe Tecnomatix Process Simulate.

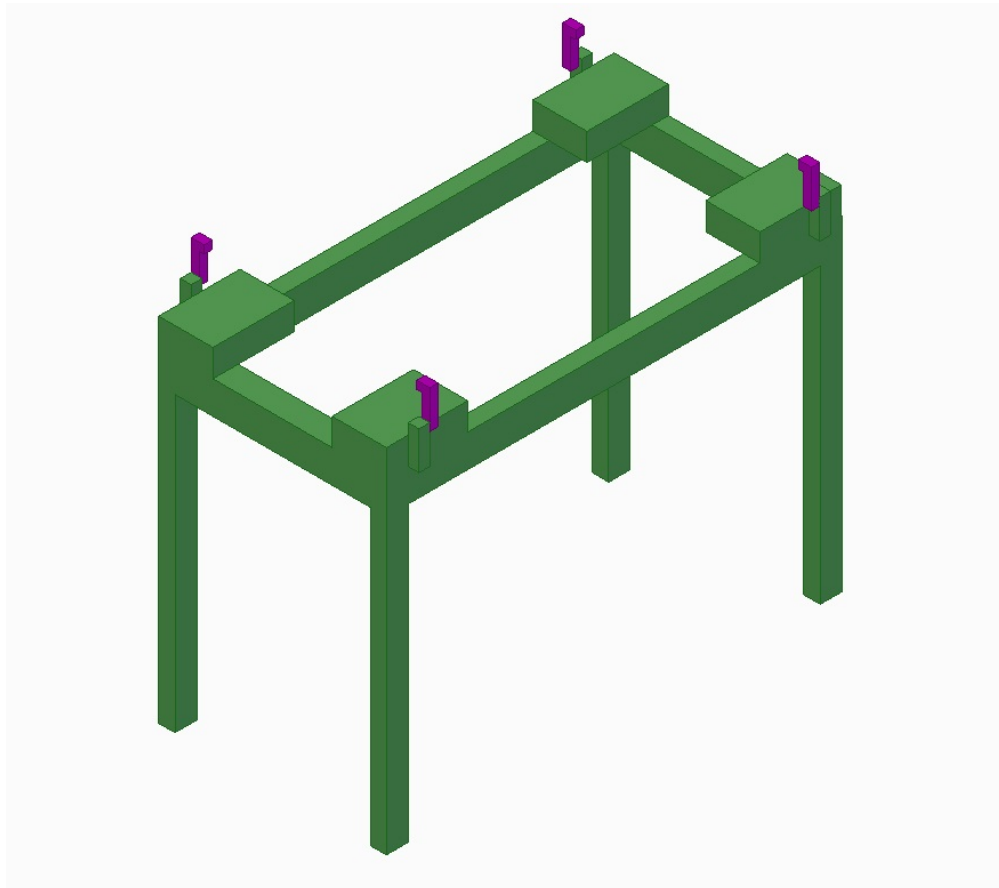


Obr. 2.9: Odporučené nastavenia pre uloženie dokumentu JT.

2.3.2 Návrh stola pre bodové zváranie

Pre proces bodového zvárania, bolo taktiež potrebné vymodelovať uchopovacie zariadenie slúžiace na fixovanie materiálu určeného na zváranie. Môj vymodelovaný stôl pozostáva z klasickej štvornohej konštrukcie a štyroch kolíkov určených na priťahovanie dvoch materiálov k sebe.

Zvárací stôl pre bodové zváranie



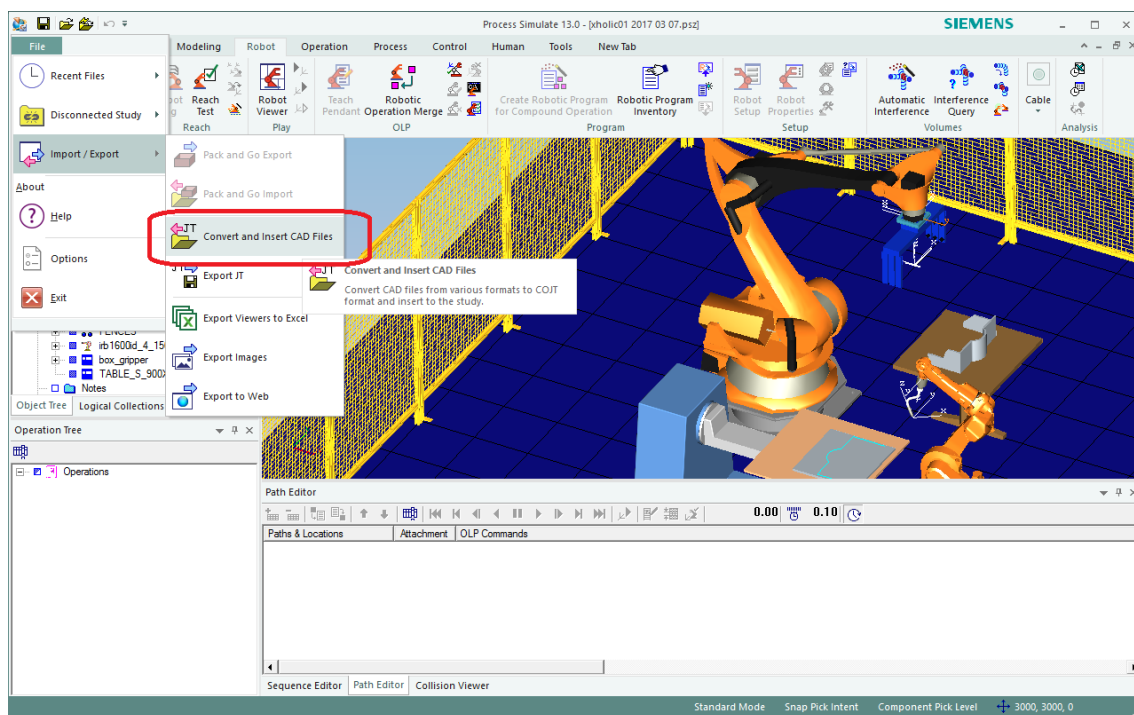
Obr. 2.10: Navrhnutý zvárací stôl pre bodové zváranie.

Vymodelovaný stôl pre bodové zváranie je taktiež potrebné uložiť vo formáte JT, aby bolo možné s ním v programe Tecnomatix Process Simulate ďalej pracovať. Nastavenie východzieho formátu bolo popísané v predchádzajúcej časti tejto práce a je znázornené na obr. 2.9.

2.3.3 Konvertovanie dokumentu JT na formát COJT

V ďalšej časti si ukážeme ako konvertovať vymodelované zariadenia tak, aby sme ich mohli následne “oživiť” a zadať im požadované kinematické vlastnosti.

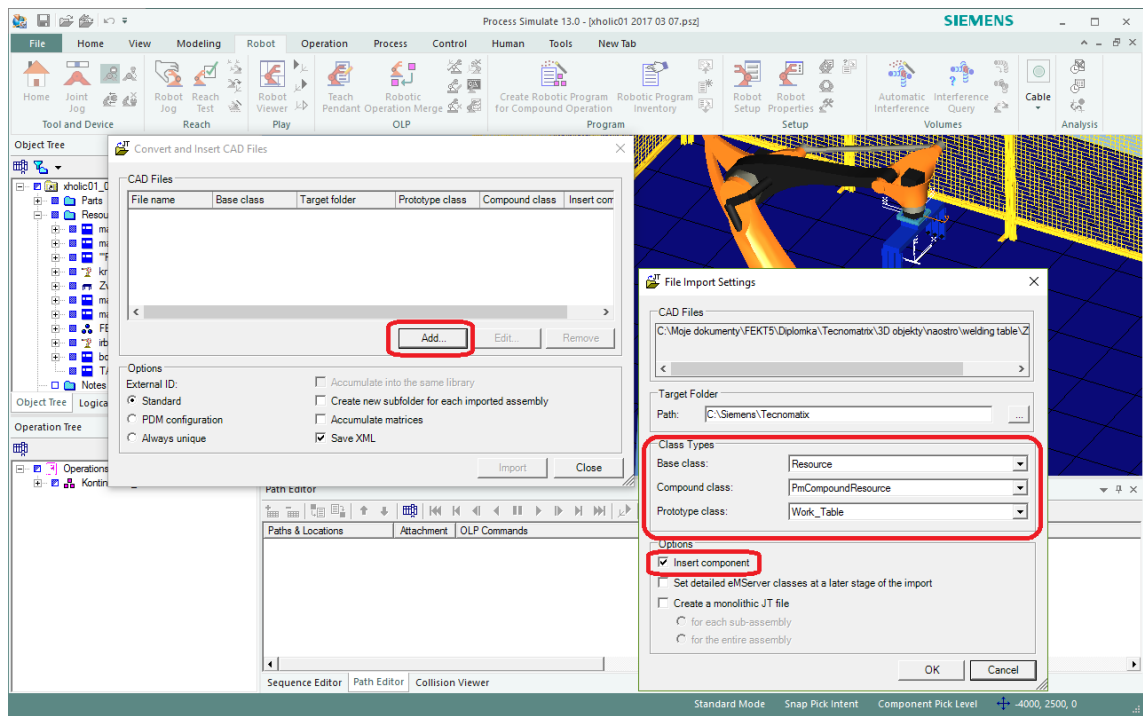
Pre konvertovanie v programe Tecnomatix Process Simulate využívame príkaz *Convert and Insert CAD Files*, pomocou ktorého je možné konvertovať vymodelované JT dokumenty do formátu COJT, s ktorým vie Tecnomatix Process Simulate pracovať.



Obr. 2.11: Príkaz na konvertovanie JT dokumentov.

Vyberieme dokument JT, v ktorom sa nachádza naša vymodelovaná zostava. Zadefinujeme triedu, do ktorej súčiastka patrí. Pre navrhovaný zvarač stôl sú nastavenia zobrazené na obr. 2.12.

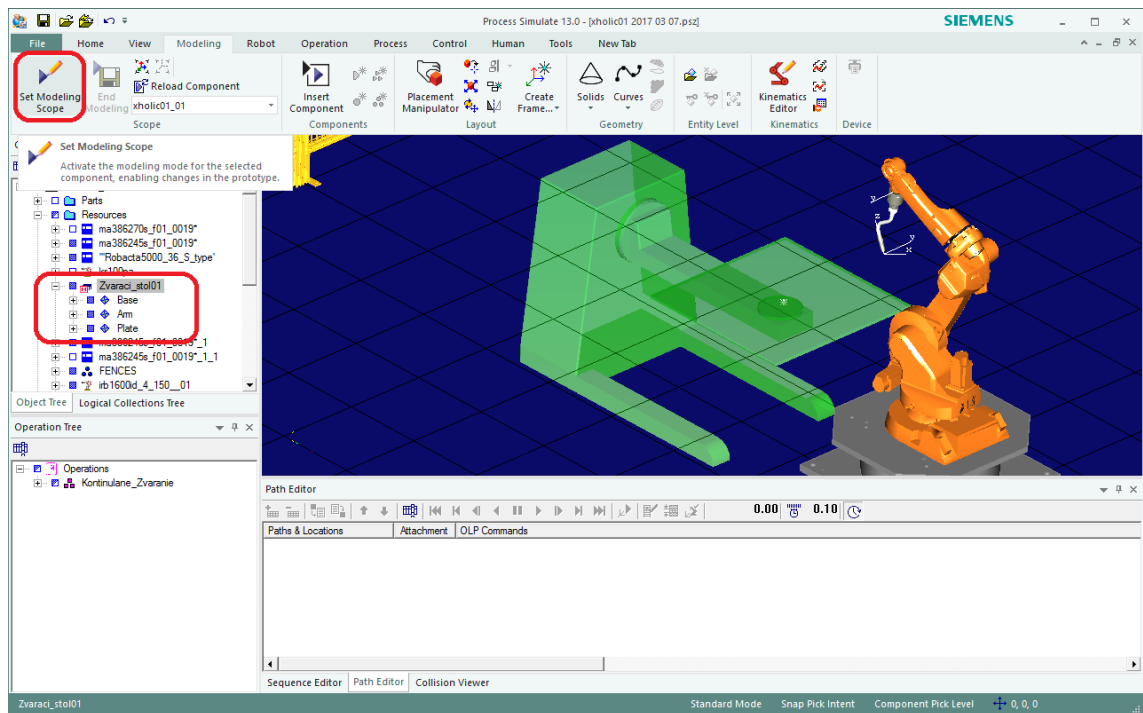
V prípade, ak chceme súčiastku vložiť do projektu ihneď po konvertovaní zvolíme možnosť ‘Insert component’. V záložke Modeling je možné kedykoľvek vkladať už konvertované súbory pomocou príkazu *Insert component*.



Obr. 2.12: Nastavenie pre konvertovanie JT dokumentov.

2.3.4 Nastavenie kinematiky zváracieho stola

Po umiestnení zváracieho stola do projektu je čas na nastavenie kinematiky. Označíme si zvárací stôl v Object Tree a umožníme modelovanie súčiastky pomocou príkazu **Set Modeling Scope** v záložke Modeling. Na obr. 2.13 môžeme pozorovať, že v Object Tree sa nám zobrazí malý symbol červeného M hned vedľa súčiastky, ktorú budeme upravovať, a taktiež sa nám zobrazia všetky tri časti, s ktorých zvárací stôl pozostáva. V tomto bode je možné pomocou príkazu **Modify Color** v záložke View zmeniť farbu jednotlivých častí stola pre lepšie zobrazenie.

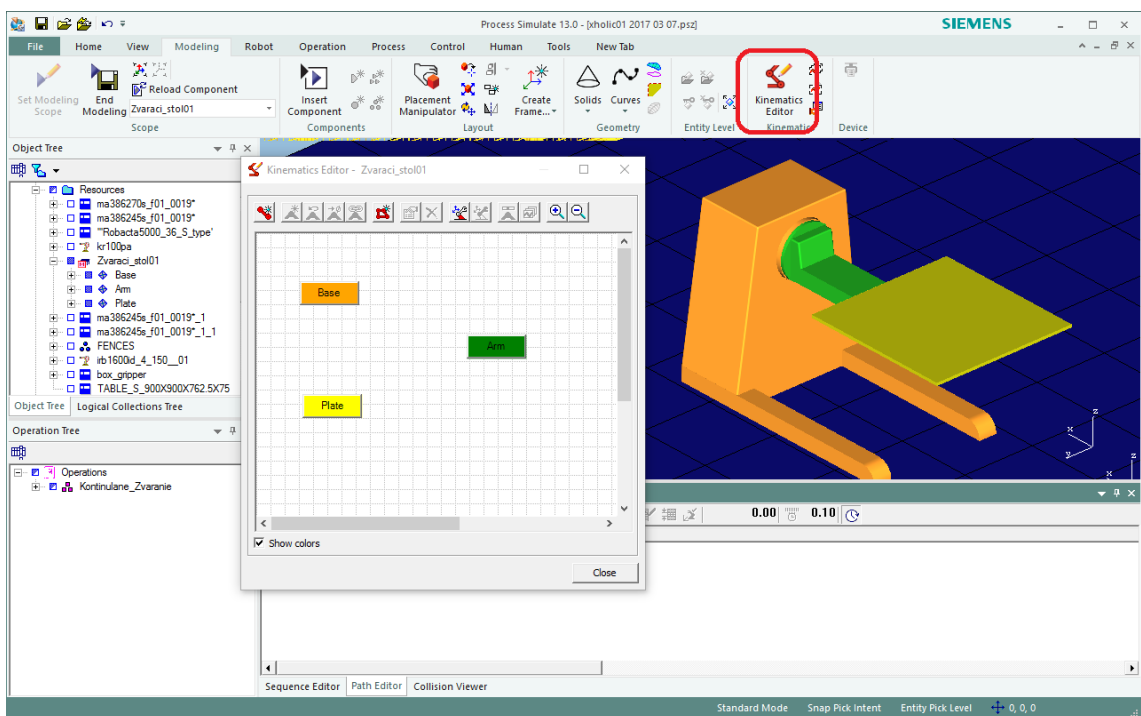


Obr. 2.13: Spustenie modelovacieho režimu zváracieho stola.

Po označení zváracieho stola v Object tree sa pomocou príkazu **Kinematics Editor** presunieme do editora, v ktorom je možné upravovať kinematiku zariadenia. Editor je zobrazený na obr. 2.14.

V prvom kroku k definovaniu kinematiky objektu využijeme príkaz **Create Link**. Pomocou tohto príkazu si zadefinujeme všetky objekty do určitých skupín so spoločnou kinematikou. Pokiaľ chceme, aby sme kinematiku zadefinovali pre zariadenie správne, musíme dbať na to, aby boli všetky komponenty, ktoré sa majú pohybovať spoločne, obsiahnuté v jednom odkaze. Napríklad, pokiaľ by sme chceli, aby sa spoločne s otočným ramenom pohybovali aj skrutky a matice, je potrebné ich umiestniť do jedného spoločného odkazu.

Časti zariadenia so spoločnou kinematikou sa delia do odkazov nasledujúcim spôsobom. Príkazom **Create Link** sa presunieme do okna Link Properties, v ktorom sa definujú vlastnosti odkazu. Do poľa s označením Name zadáme názov nášho prvého odkazu. Do poľa Link Elements, postupne ľavým tlačidlom myši vyberieme z Graphic Viewer všetky objekty zariadenia, ktoré budú patriť do prvého odkazu. Teda vyberieme všetky časti zariadenia, pre ktoré chceme mať spoločnú kinematiku. Objekty je možné vyberať v Graphic Viewer alebo v Object Tree. Po vložení všetkých častí potvrdíme voľbu tlačidlom OK. Rovnakým spôsobom si postupne zadefinujeme aj ostatné skupiny objektov celého zariadenia.



Obr. 2.14: Kinematics Editor pre definovanie skupín odkazov.

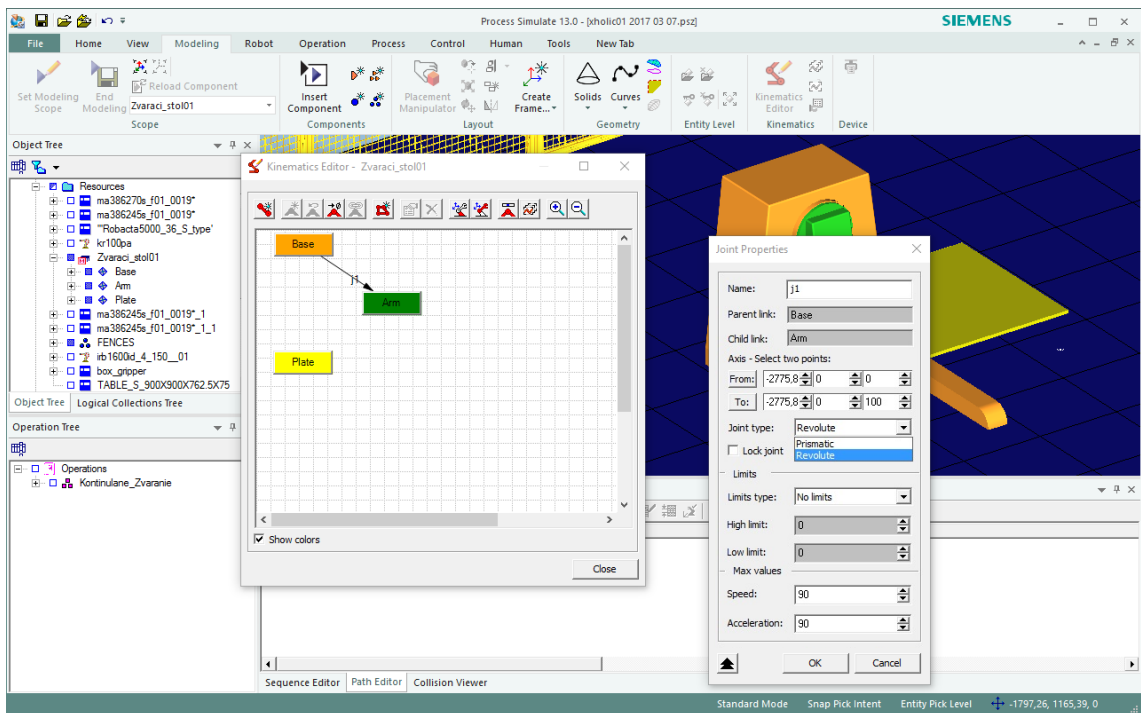
Druhým krokom ku nastaveniu kinematiky zariadenia je definovanie spojov. Spoj nám vytvára prepojenie medzi dvoma odkazmi. Prvým odkazom je Parent link a druhým odkazom je Child link. Pomocou týchto dvoch druhov odkazov je možné určiť hierarchiu spojení. To znamená, že pohyb Parent linku nám ovplyvní pohyb všetkých odkazov, ktoré sa nachádzajú v hierarchii pod jeho Child link.

K dispozícii sú dva rôzne typy spojení, ktoré môžeme definovať v programe Tecnomatix Process Simulate:

- Revolving joint - rotácia objektu okolo osi
- Prismatic joint - lineárny posun pozdĺž osi

Spoj je možné vytvoriť dvoma spôsobmi. Prvá možnosť je pretiahnutie čiary

lavým tlačidlom myši smerom od Parent link do Child link. Druhá možnosť ako vytvoriť nové spojenie, je označenie Parent linku a Child linku pomocou klávesy <Ctrl> a príkazom **Create Joint** vytvoríme nové spojenie, pričom prvý zvolený odkaz je typu Parent link a druhý zvolený odkaz je typu Child Link. Na obr. 2.15 je zobrazené nové spojenie medzi odkazom Base a Arm. Smer šípky nám graficky znázorňuje, že odkaz Base je Parent link a odkaz Arm je Child link. Na obr. 2.15 sa taktiež nachádza okno Joint Properties, ktoré slúži na nastavenie vlastností spojenia.



Obr. 2.15: Kinematics Editor a Joint Properties pre definovanie spojení.

Do poľa s označením Name, zadáme názov vytvoreného spojenia. V poliach Parent link a Child link, môžeme pozorovať nami zvolené časti zariadenia, medzi ktorými sa pohyb definuje.

V poli Joint Type volíme druh pohybu, podľa našich požiadavok. Pre zvärací stôl sú všetky pohyby rotujúce, takže zvolíme možnosť Revolving joint.

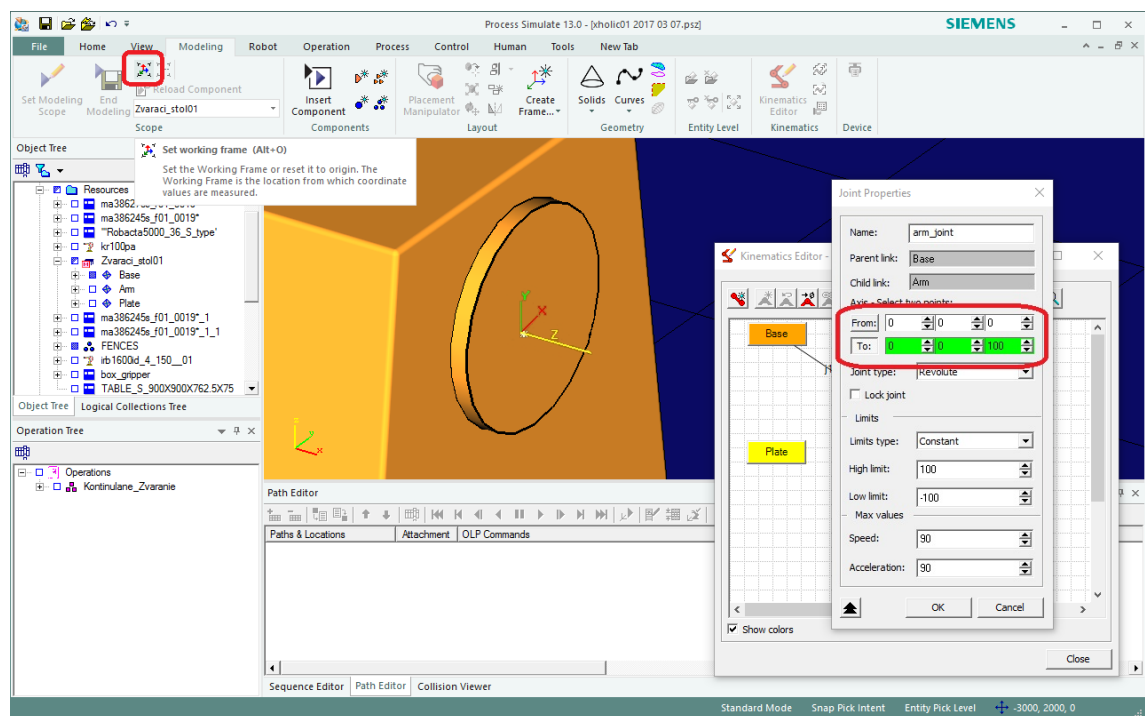
Ďalším parametrom, ktorý je možné meniť, sú limitné hranice pre pohyb. Máme na výber medzi konštantnými hodnotami, premennými hodnotami a hodnotami bez obmedzenia. V prípade zväracieho stola som stanovil limitné hodnoty pre otočné rameno v rozsahu 100 a -100.

Taktiež je možné meniť parametre pre maximálnu rýchlosť a maximálnu akceleráciu pohybu.

Najdôležitejším parametrom je však určenie osi, v ktorej sa daný pohyb bude

vykonávať. Táto os sa vymedzuje pomocou dvoch bodov. Najjednoduchší spôsob je použitie príkazu **Set working frame**, pomocou ktorého premiestnime Working Frame do požadovaného miesta zväracieho stola na miesto, v ktorom chceme, aby sa os nachádzala. V časti Axis do poľa **From** zadáme nulové súradnice a do poľa **To** zadáme rovnaké súradnice ale s menšou zmenou. Zmeníme hodnotu súradnice osi, v ktorej chceme, aby sa rotujúci pohyb vykonával. Postup je znázornený na obr. 2.16.

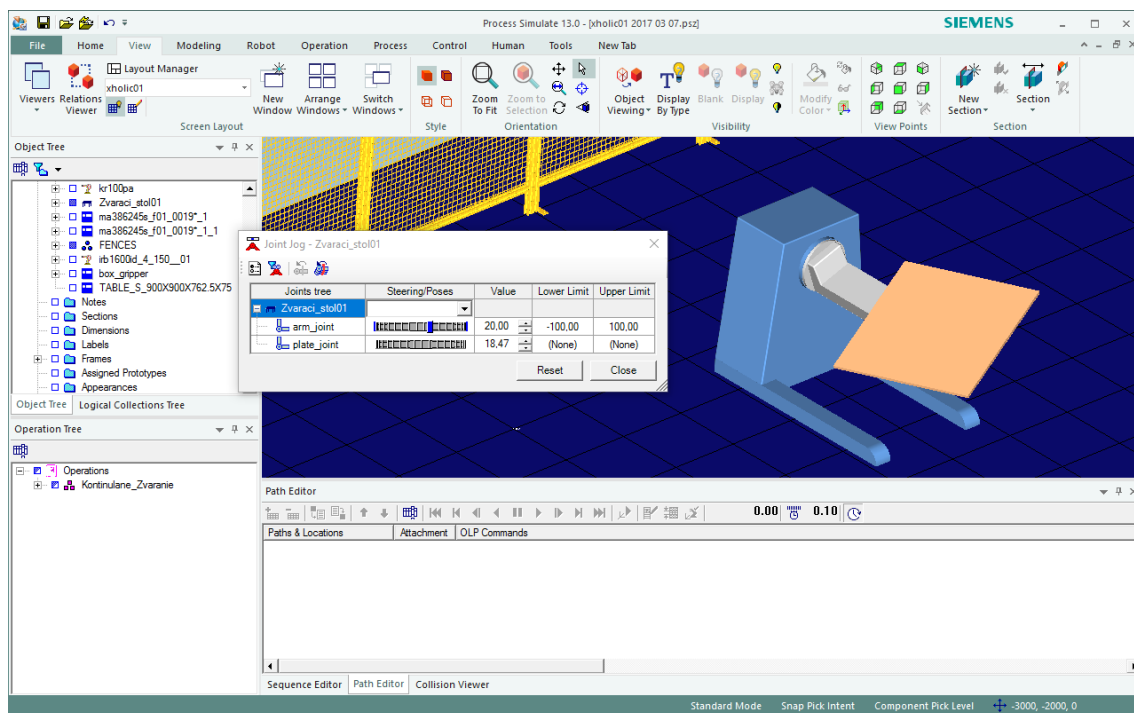
Rovnakým spôsobom zadefinujeme aj všetky ostatné spojenia medzi odkazmi. Pokiaľ máme úspešne zadefinovať všetky spojenia, tak sa nám podarilo úspešne “oživiť” naše vymodelované zariadenie.



Obr. 2.16: Nastavenie osí pre určenie rotačného pohybu.

Pomocou príkazu **End Modeling** ukončíme možnosť modelovať zariadenie a vedľa daného objektu v Object Tree, nám zmizne malý symbol červeného M, ktoré indikovalo možnosť jeho upravovania. Pomocou príkazu **Joint Jog** môžeme manipulovať zo zariadením. Ukážka manipulácie so zväracím stolom je znázornená na obr. 2.17. Vo výsledku má zvärací stôl dve rotačné osi, pričom jedna z nich, má limitné parametre v hodnotách od 100 do -100 a druhá nemá nastavené limitné parametre.

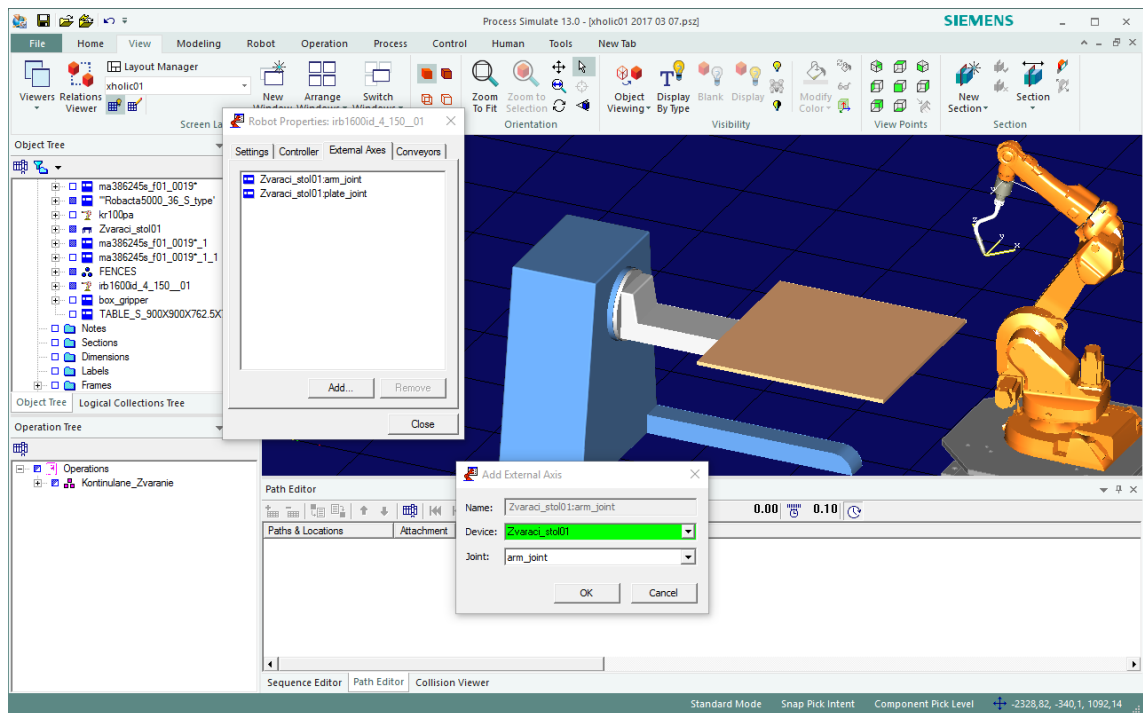
Taktiež je možné zadefinovať vytvorené osi manipulátora ako externé osi zväracieho robota, a týmto spôsobom zlepšiť spoluprácu medzi robotom a zväracím stolom. Pomocou príkazu **Robot Properties** sa dostaneme do okna, v ktorom je



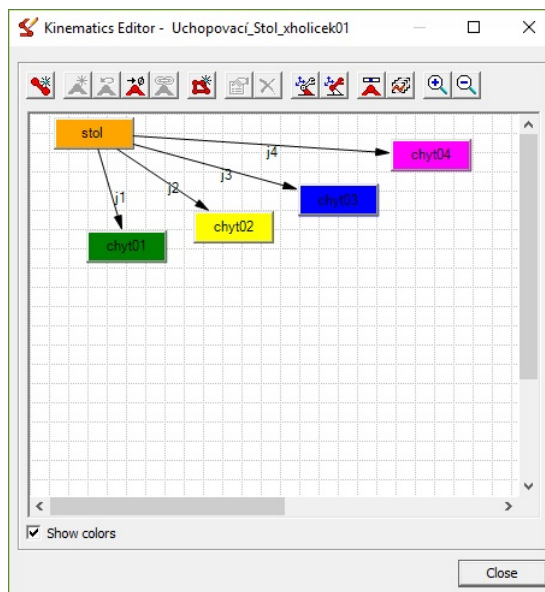
Obr. 2.17: Ukážka rotačného pohybu na zväracom stole.

možné nastavovať vlastnosti robota. V záložke External Axes sa pomocou tlačidla Add pridávajú externé osi pre robota. Postupne si definujeme obe osi zväracieho stola tak, ako je znázornené na obr. 2.18. Teraz je možné ovládať rotujúce pohyby zväracieho stola, pomocou robota funkciou **Joint Jog**.

Podobne ako sme teraz nastavovali kinetiku zväracieho stola určeného na kontinuálne zváranie, nastavíme kinetiku aj pre stôl určený na fixovanie materiálu pre bodové zváranie. Rozdiel bude samozrejme v kinematike zariadenia. Celá konštrukcia stola bude statická, avšak je potrebné definovať rozsah pohybu, pre štyri prítlačné kolíky určené na fixovanie materiálu. Definovanie štyroch pohybov pre zvärací stôl je znázornené na obr. 2.19. Rozdiel medzi kolíkmi je v tom, že kolíky na pravej strane stola, majú pohyb definovaný v rozmedzí hodnôt 0 až 90. A kolíky na ľavej strane stola, majú pohyb definovaný v rozmedzí -90 až 0. To je spôsobené tým, že majú kolíky na protilahlých stranách stola definovaný rotačný pohyb do opačných strán.



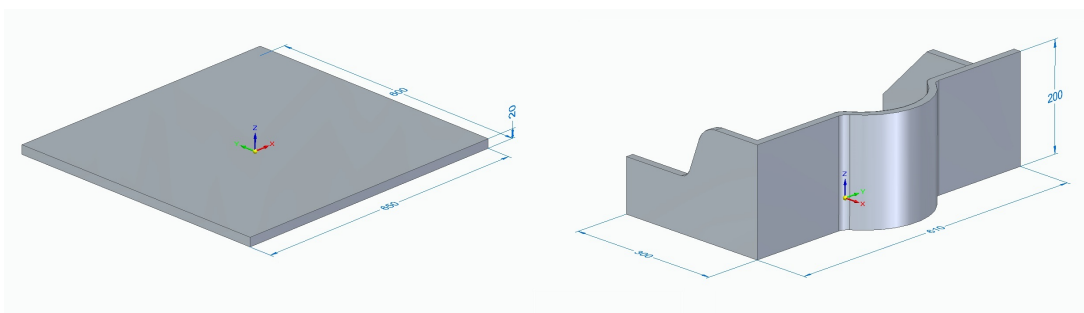
Obr. 2.18: Definovanie externých ôsí robota.



Obr. 2.19: Definovanie kinematiky stola pre bodové zváranie.

2.3.5 Súčiastka určená na kontinuálne zváranie

Prípravok určený na kontinuálne zváranie sa skladá z dvoch častí. Podobne ako sa modelovali CAD súbory komponentov zvaracieho stola v programe Solid Edge, tak sa modelovali aj časti prípravku určeného na zváranie.



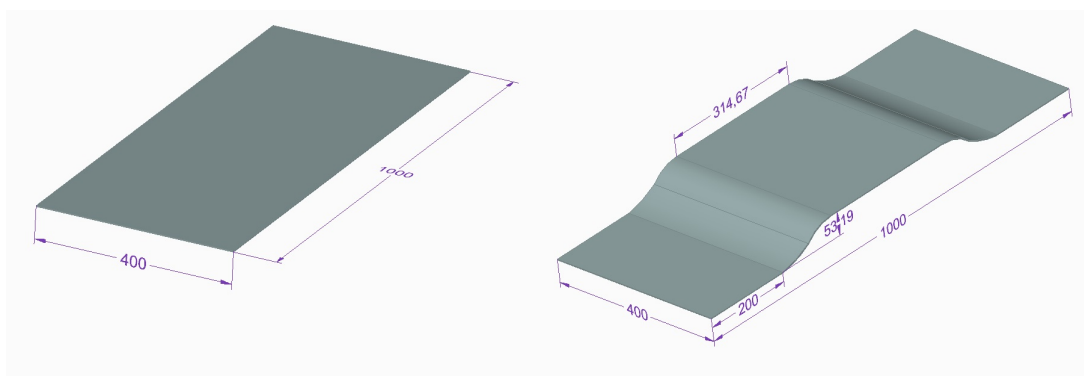
Obr. 2.20: Časti prípravku určeného na kontinuálne zváranie.

Dokumenty JT oboch vymodelovaných častí prípravku určeného na zváranie konvertujeme rovnakým spôsobom, ako JT dokument zvaracieho stola popisovaný v časti 2.3.3. Rozdiel je iba v tom, že časti zvaracieho stola sme definovali ako **Resource**, ale časti prípravku definujeme ako **Part**.

Po vložení oboch častí do projektu umiestnime podstavu prípravku na rotujúcu dosku zvaracieho stola a pomocou príkazu **Attachment** v záložke Tools je možné ju k rotujúcej doske stola “prilepiť”. To znamená, že sa budú pochybovať spoločne.

2.3.6 Súčiastka určená na bodové zváranie

Prípravok určený pre proces bodového zvárania sa skladá z dvoch plechov o hrúbke 2 mm. Tieto časti, je taktiež potrebné skonvertovať do formátu COJT.

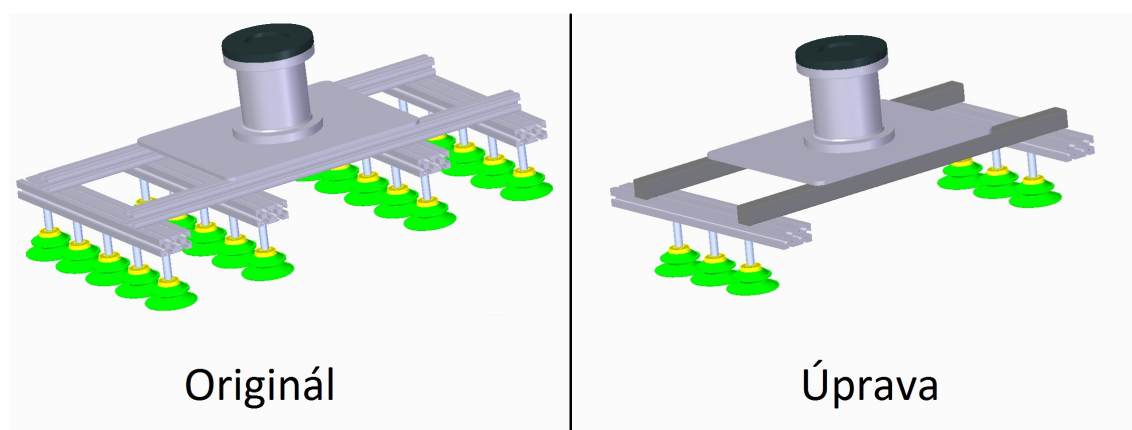


Obr. 2.21: Časti prípravku určeného na bodové zváranie.

2.3.7 Úprava vákuového uchopovacieho nástroja

Na manipuláciu s plechovými časťami, podobnými ako budem využívať pri simulácií bodového zvarovania, sa v praxi využívajú nástroje s vákuovými prísavkami. Vákuový nástroj som celý nezhotovoval sám, ale iba upravoval tak, aby rozmerovo vyhovoval vytvorenej simulácii. Vymodelovanú súčiastku som stiahol z webového servera www.grabcad.com a túto súčiastku na server nahral užívateľ ccrdgb. [8]

Na obr. 2.22 je znázornený originálny a upravený nástroj. Odstránil som dve rady vákuových prísaviek a upravil som ich rozloženie tak, aby bolo možné manipulovať so zahnutým plechom v simulácií bodového zvarovania. Tieto úpravy som vykonával v programe Solid Edge a výsledný nástroj som uložil ako dokument JT.



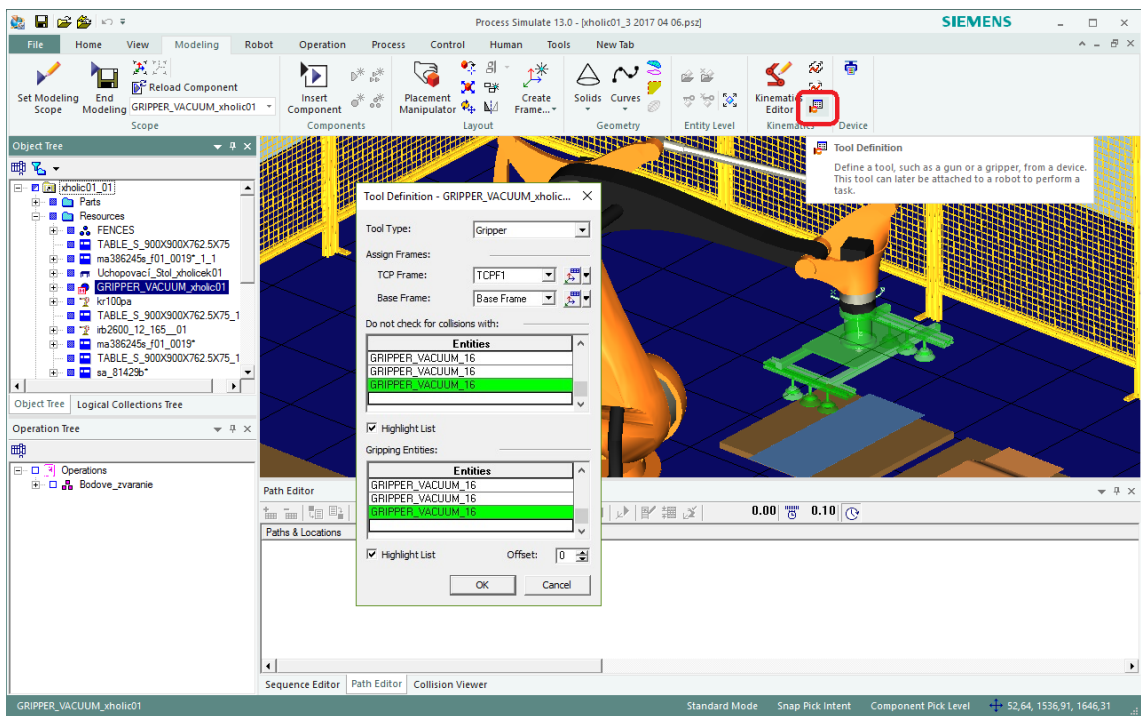
Obr. 2.22: Vákuový nástroj určený na uchopovanie.

Po vložení modelu nástroja do programu Tecnomatix je potrebné definovať jeho presnú pracovnú náplň. V prípade vákuových prísaviek sa jedná o uchopovaciu činnosť.

Definovať nástroj robota je možné pomocou príkazu **Tool Definition** v záložke Modeling. Tento príkaz sa pre nástroj sprístupní po umožnení modelovacích funkcií. Označíme nástroj v Object Tree alebo v Graphic Viewer a pomocou príkazu Set Modeling Scope povolíme modelovanie a upravovanie nástroja. V Object Tree sa nám vedľa komponentu, ktorý upravujeme zobrazí symbol červeného M. To znamená, že nástroj je možné upravovať a meniť jeho vlastnosti.

Na obr. 2.23 je znázornené okno funkcie Tool Definition, v ktorom sa nastavujú parametre nástrojov určených pre robotov. V časti *Tool Type* vyberieme pre prípad vákuových prísaviek možnosť Gripper. Táto vlastnosť slúži na uchopovanie materiálu v simulácií. V časti *Assign Frames* definujeme dva body nástroja. Bod TCP Frame nám určuje miesto, v ktorom sa má nástroj upevňovať k robotovi a bod Base Frame slúži na nasmerovanie nástroja do presných pozícií. V prípade vákuového nástroja

som Base Frame volil v spodnej časti nástroja medzi prísavkami. Do časti *Do not check for collisions with* postupne vyberieme všetky objekty nástroja, s ktorými nebude možné evidovať kolízie v simulácií. V tomto prípade je potrebné vybrať všetky prísavky nástroja, pretože tie sa v reálnom procese deformujú pri pritlačení k prenášanej súčiastke, keďže sú vyrobené z gumy. Vybrané objekty sa sfarbia na žlté. Do časti *Gripping Entities* sa vyberajú objekty, pri ktorých má simulácia prípravok určený na presun uchopiť. V prípade vákuového nástroja sa vyberú všetky prísavky. To znamená, že pokiaľ sa nejaký objekt dostane v simulácií do kontaktu s prísavkou a my budeme od robota vyžadovať uchopenie objektu, tak sa daný predmet ukotví k nástroju. Vybrané objekty sa znižujú modrou farbou. Pre ukončenie potvrdíme tlačidlom OK a príkazom End Modeling ukončíme úpravy na nástroji. Teraz je možné nástroj upevniť k robotovi pomocou príkazu **Mount Tool** a simulovať s ním operácie vykonávané s robotmi.



Obr. 2.23: Nastavenie uchopovacieho nástroja pre robota.

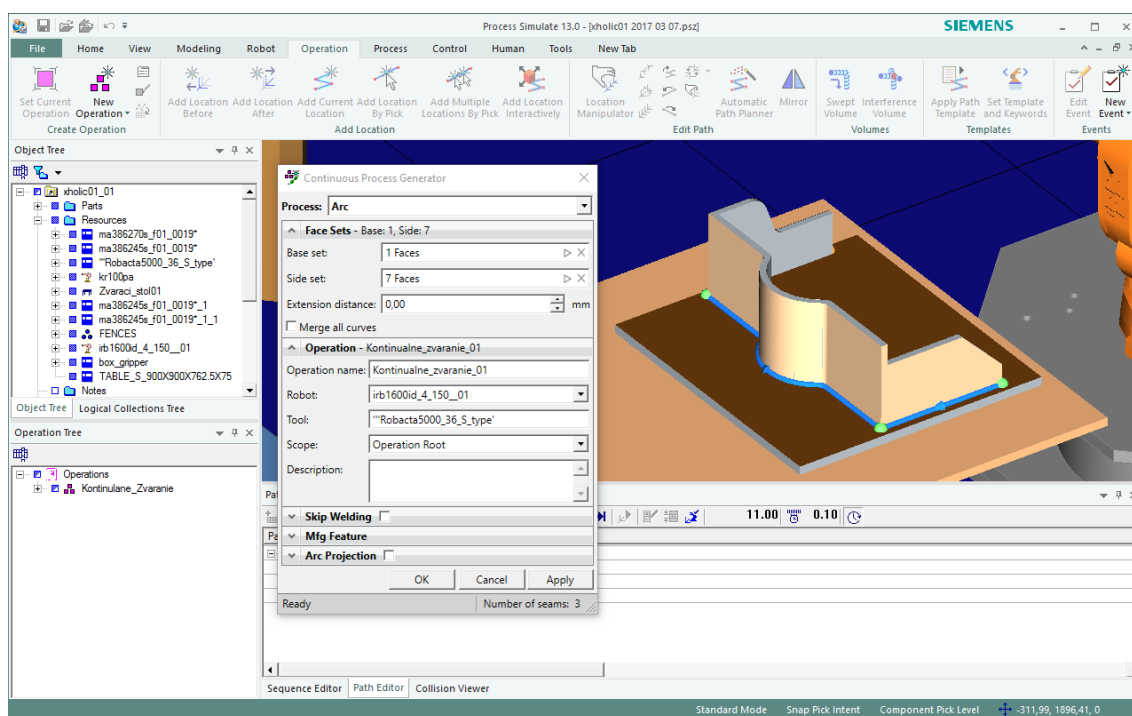
2.4 Simulácia zvárania

Zváranie je technologický proces, pri ktorom sa dva tavitelné materiáli spájajú pomocou lokálneho roztavenia, zaliatia a následného ochladenia. V tejto práci sa venujem simulácii kontinuálneho a bodového zvárania realizovaného priemyselnými robotmi. Všetky potrebné pohyby robotov budú definované v **Standard Mode**, avšak virtuálne riadenie je spracované v **Line Simulation Mode**.

2.4.1 Kontinuálne zváranie

Pre simuláciu kontinuálneho zvárania slúži v programe Tecnomatix Process Simulate príkaz **Continuous Process Generator**. Pri definovaní pohybu máme na výber dva druhy procesov, a to **Arc** proces a **Coverage pattern** proces.

V prípade Arc procesu sa vyberie Base plocha a Side plocha, pričom je možné do každého poľa vložiť viacej plôch. Po vybratí sa Base plochy zafarbujú na tmavohnedo a Side plochy na bledohnedo. Modrá čiara, ktorá sa po výbere plôch objaví nám signalizuje, že výber plôch prebehol v poriadku a znázorňuje nám krivku generovaného pohybu. Spoločne s modrou čiarou sa objaví aj modrá šípka, ktorá označuje smer generovaného pohybu. Smer sa dá zmeniť kliknutím ľavého tlačidla myši na šípku. Tento typ procesu je vhodný pre zváranie dvoch rôznych častí.

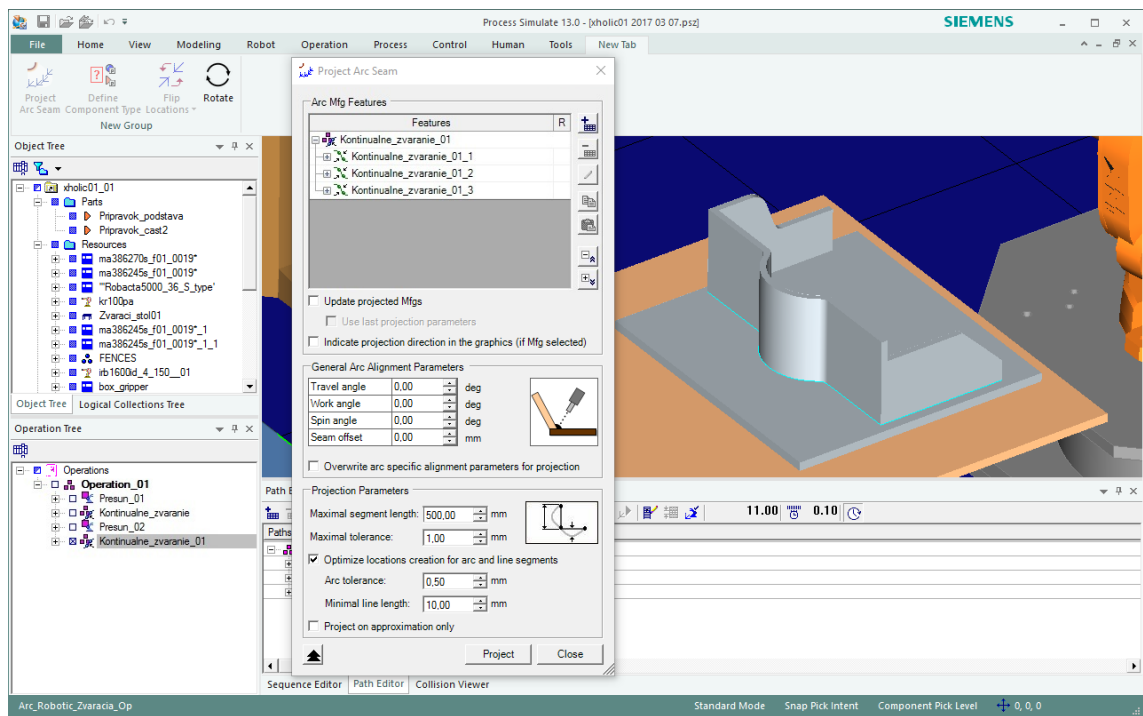


Obr. 2.24: Generátor kontinuálnych procesov.

V prípade Coverage pattern procesu je potrebné definovať Base plochu a dva body, ktoré určujú odkiaľ a kam má kontinuálny pohyb prechádzať. Tento typ procesu je vhodný pre simulovanie procesu lakovania väčších povrchov.

Ako môžete vidieť na obr. 2.24 pre simulovanie kontinuálneho zvarovania využívam Arc proces. Ďalej, je možné zadať názov operácie, robota a nástroj, s ktorým sa bude operácia vykonávať. Funkcia *Merge all curves* slúži na spojenie všetkých vygenerovaných kontinuálnych procesov a spojí ich do jedného. V tomto prípade funkciu pre spojenie pohybov nevyžívam, aby nedochádzalo k prípadným kolíziám pri prechode zvaracieho nástroja medzi dvoma stranami polotovaru.

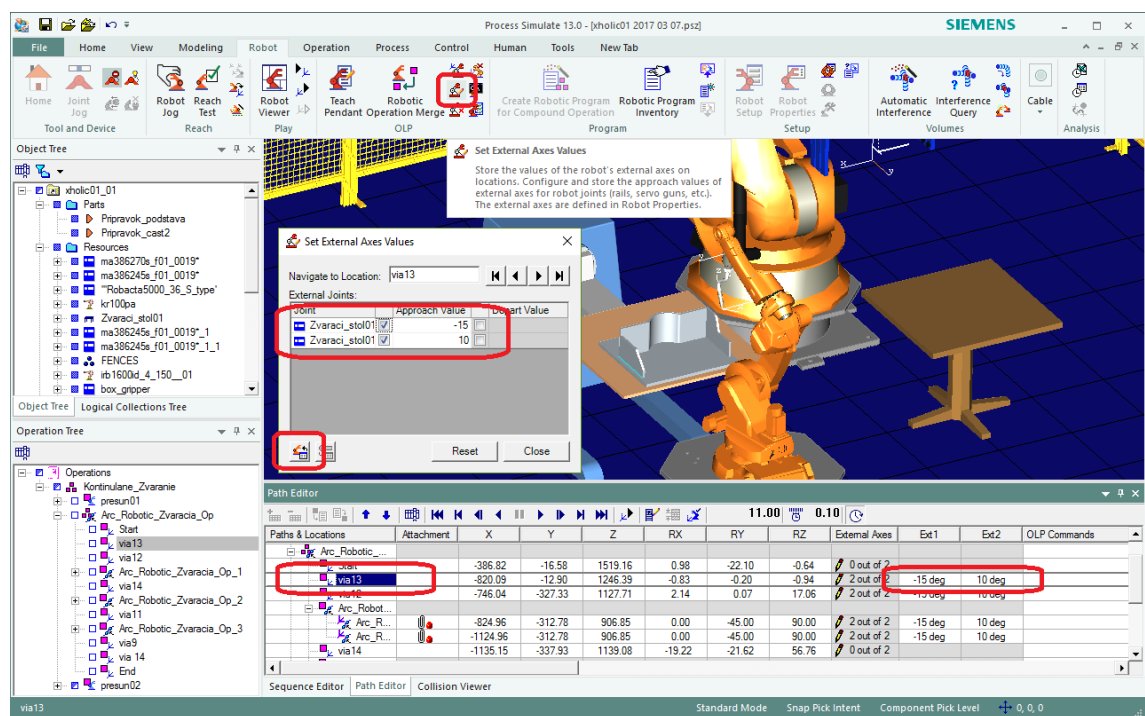
Po vytvorení kontinuálneho procesu je ďalšou úlohou vygenerovanie projekcie, ktorá vytvorí kroky a pridelí im súradnicové body pre náš kontinuálny proces. Toto generovanie projekcie je možné pomocou príkazu *Project Arc Seam* a ten je znázornený na obr. 2.25.



Obr. 2.25: Generovanie projekcie pre proces kontinuálneho zvarovania.

Takto vygenerovanú projekciu môžeme presunúť do časti Path Editor a spustiť simuláciu. V tomto okamžiku však nie je proces presný a je možné, že dochádza ku kolíziám medzi objektami používanými v aplikácii. Preto je potrebné postupne nastaviť externé osi zvaracieho stola a natočenie zvaracej hlavy robota do požadovaných polôh tak, aby nedochádzalo ku kolíziám medzi robotom a rotujúcim zvaracím stolom.

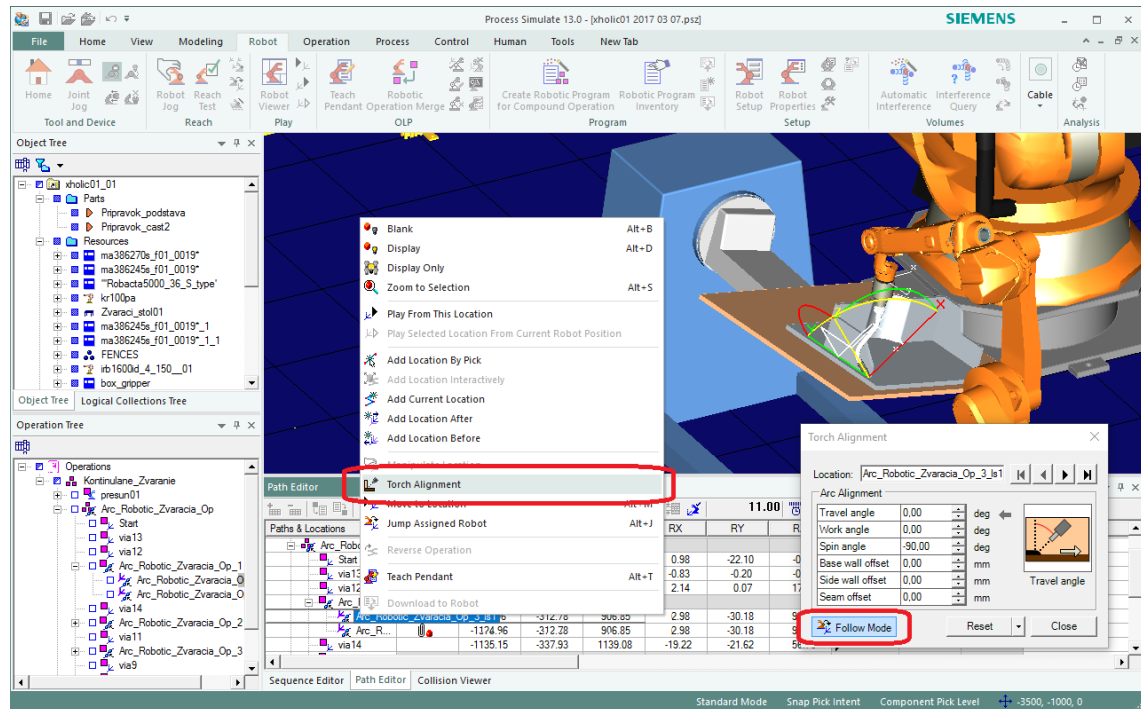
Nastavenie polôh zvracieho stola prevedieme pomocou funkcie **Set External Axes Values**, ktorá sa nachádza v záložke Robot. V časti External Joints si vyberieme z ponuky definovaných externých osí tie, ktoré budeme pri manipulácii používať. V prípade aplikácie kontinuálneho zvarania, budeme využívať obe osi manipulačného stola. Uhol natočenia sa definuje v časti Approach Value. Pre lepšie znázornenie procesu je vhodné zapnutie funkcie Follow Mode. Pomocou tejto funkcie vieme presne určiť, v akej polohe sa nachádza robot, voči rotujúcemu manipulatoru a vieme taktiež určiť, či robot dočiahne na požadované miesto zvaru. Postupne nastavíme externé osi pre celý proces kontinuálneho zvarania. Medzi definovanými polohami procesu je možné prepínať pomocou šípok v pravej hornej časti okna. Funkcia je znázornená na obr. 2.26.



Obr. 2.26: Nastavovanie hodnôt pre externé osi robota pri zváracom procese.

Ďalšou funkciou, ktorú budeme využívať je funkcia **Torch Alignment**. Pomocou tejto funkcie je možné nastavovať uhol natočenia zvaracej hlavy robota. Pravým tlačidlom myši klikneme na prvý bod kontinuálneho procesu v ponuke Path Editor a vyberieme z ponuky možnosť Torch Alignment. Funkcia sa ovláda a nastavuje podobne ako funkcia Set External Axes Value v predchádzajúcom kroku. Zapnutím funkcie Follow Mode, môžeme sledovať presnú polohu rotujúceho zvracieho stola a nástroja robota. V časti Arc Alignment sa nastavujú uhly natočenia zvaracej hlavy v rôznych osiach. Opäť postupne nastavíme uhly natočenia nástroja robota, pre celý

proces kontinuálneho zvarovania. V prípade simulácie kontinuálneho zvarovania som naráčal iba jednu os nástroja, konkrétne *Spin angle* na hodnotu -90. V prípade, ak sa na zvarovanie používa iný nástroj, môžu sa nastavenia líšiť. Funkcia je znázornená na obr. 2.27.



Obr. 2.27: Nastavovanie natočenia robota pri procese kontinuálneho zvarovania.

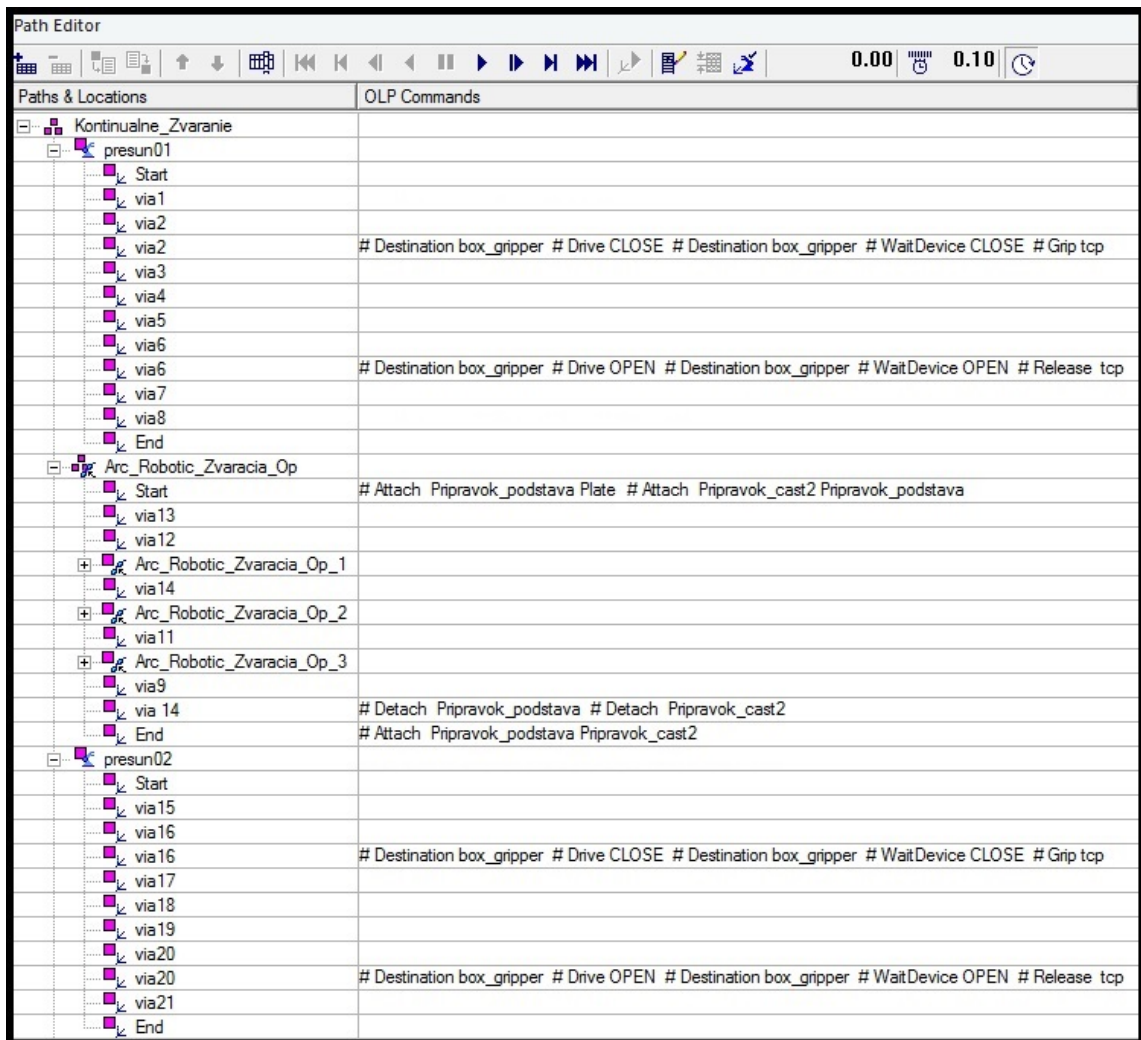
Overenie správneho nastavenia celého simulačného procesu je možné pomocou funkcie **Collision Mode**, ktorá sa nachádza v časti Collision Viewer. Túto funkciu je možné spustiť v layoute **Standard**. V časti Collision Set sa definujú objekty, medzi ktorými sa majú kontrolovať kolízne stavy. Funkcia sa zapína/vypína pomocou ikony Collision Mode On/Off. Po spustení simulácie v prípade kolízie objekty sčervenejú a v časti Collision Options je možné nastaviť, aby sa simulácia v danom mieste zastavila. Týmto spôsobom je možné určiť miesta v simulovanom programe, v ktorých dochádza ku kolíznym stavom a ďalej ich riešiť.

V prípade, ak po správnom nastavení natočenia zvaracieho stola a nástroja robota ďalej dochádza ku kolíziám, je možné využiť funkciu **Auto Teach**. Tieto kolízie sú zapríčinené otáčaním častí určených k zvarovaniu a robot nevie presne nájsť miesta zvaru. Na začiatok vymažeme staré polohy robota pomocou funkcie **Clear Teach Location**, ktorá sa nachádza v záložke Robot. Pre povolenie funkcie musí byť v časti Path Editor označený proces kontinuálneho zvarovania. Po vymazaní naučených polôh je čas na naučenie nových. Pomocou ikony v záložke Robot vyberieme funkciu **Set**

Robots for Auto Teach a vyberieme robota pre kontinuálne zváranie. Zavrieme okno a spustíme simuláciu od začiatku. V prípade, ak celý proces prebehol úspešne bez nežiadúcich kolízií, opäť vyberieme funkciu **Set Robots for Auto Teach** a zmažeme z ponuky vybraného robota.

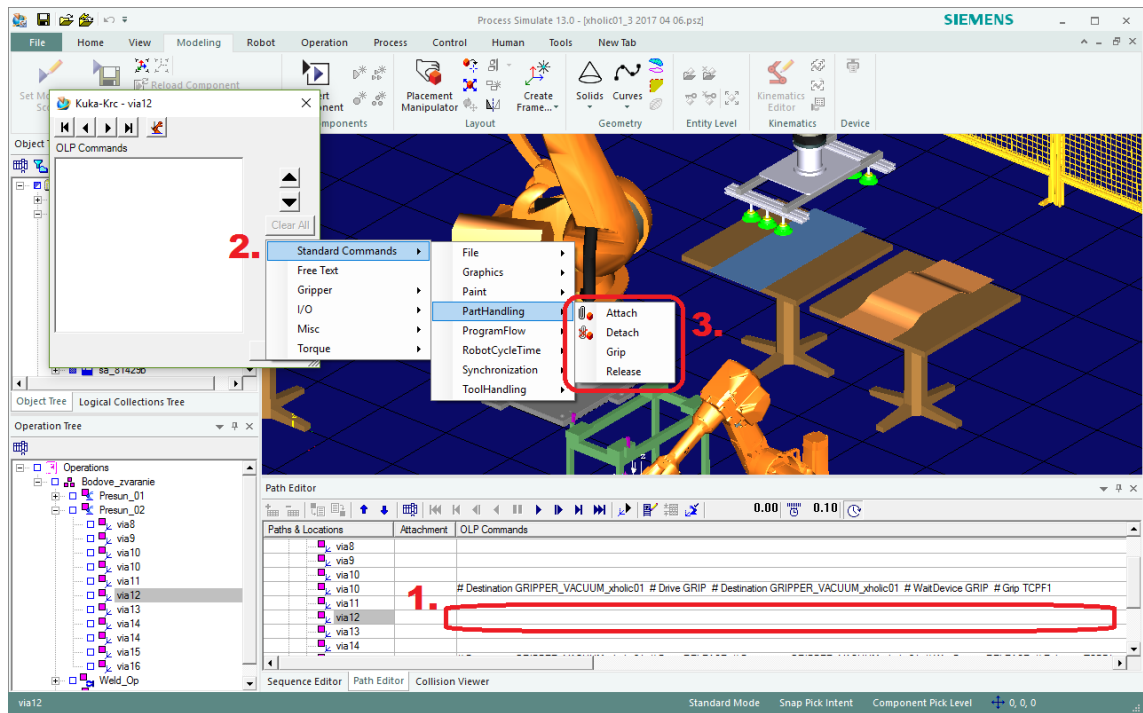
V tejto chvíli je robot úspešne naučený na celý proces kontinuálneho zvárania. Pokiaľ simulácia prebieha aj naďalej s kolíziami, môžeme sa pokúsiť napraviť to opätovným nastavením uhla natočenia zváracieho stola a natočenia zváraciej hlavice robota.

Posledným krokom je definovanie operácií presunov súčiastky určenej na zváranie. Celý program je znázornený na obr. 2.28. V príkazoch **OLP Commands** sú zadefinované všetky operácie uchopenia (Attach), odpojenia (Detach), prípadne otvárania a zatvárania gripovacieho nástroja určeného na prenášanie súčiastok.



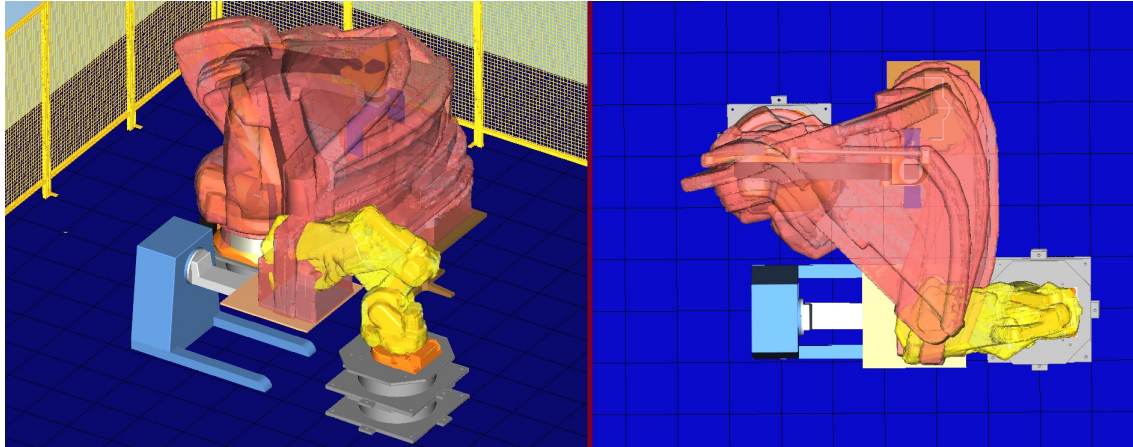
Obr. 2.28: Program kontinuálneho zvárania.

Príkazy je možné pridať do **OLP Commands**, pomocou kliknutia ľavým tlačidlom myši na požadované voľné okno príslušnej operácie v časti OPL Commands. Následne je možné pridávanie príkazov pre operáciu pomocou tlačidla Add. Celý postup voľby príkazu je znázornený na obr. 2.29. Príkaz Grip slúži na zatvorenie manipulačného nástroja a príkaz Release slúži na otvorenie manipulačného nástroja.



Obr. 2.29: Pridanie OLP Commands.

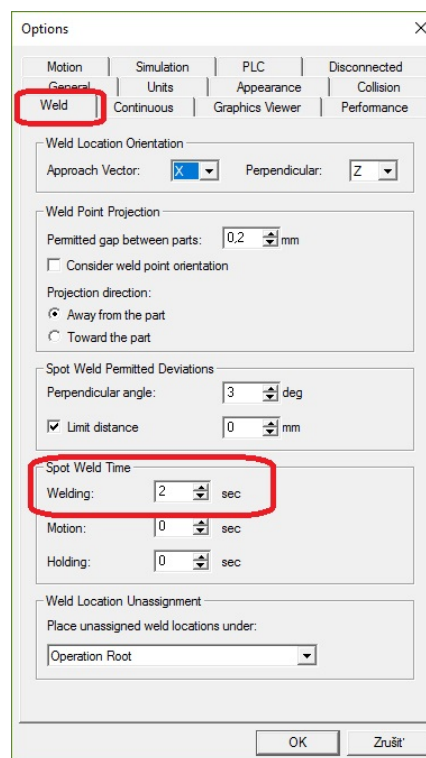
Na obr. 2.30 je znázornené prelínanie dráh spolupracujúcich robotov pri procese kontinuálneho zvarovania. Priestorové vykreslenie bolo uskutočnené pomocou funkcie **Swept Volume**, ktorá sa nachádza v záložke Operation. Do tejto inštrukcie nie je možné vkladať grafické objekty simulácie, ale pre vykreslenie sa vyberajú konkrétne operácie a program Tecnomatix už sám vypočíta priestor, v ktorom sa roboty pre danú operáciu pohybujú. Plochy znázorňujúce pohyb jednotlivých robotov sú farebne odlišené pre určenie miest, v ktorých sa dráhy robotov prekrývajú.



Obr. 2.30: Prelínanie dráh oboch robotov pri procese kontinuálneho zvárania.

2.4.2 Bodové zváranie

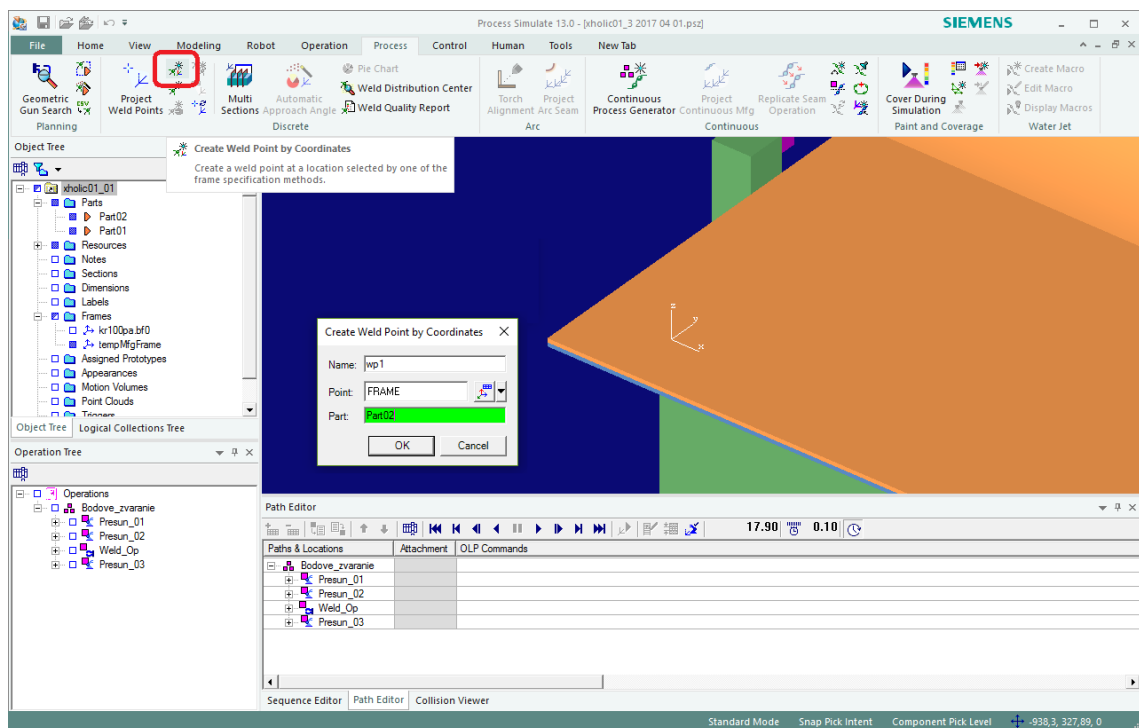
Pre simuláciu bodového zvárania je potrebné presne stanoviť body, v ktorých sa má zvar vykonať. Na definovanie týchto bodov budeme využívať sekvenciu príkazov programu Tecnomatix Process Simulate.



Obr. 2.31: Nastavenie doby pre bodové zváranie.

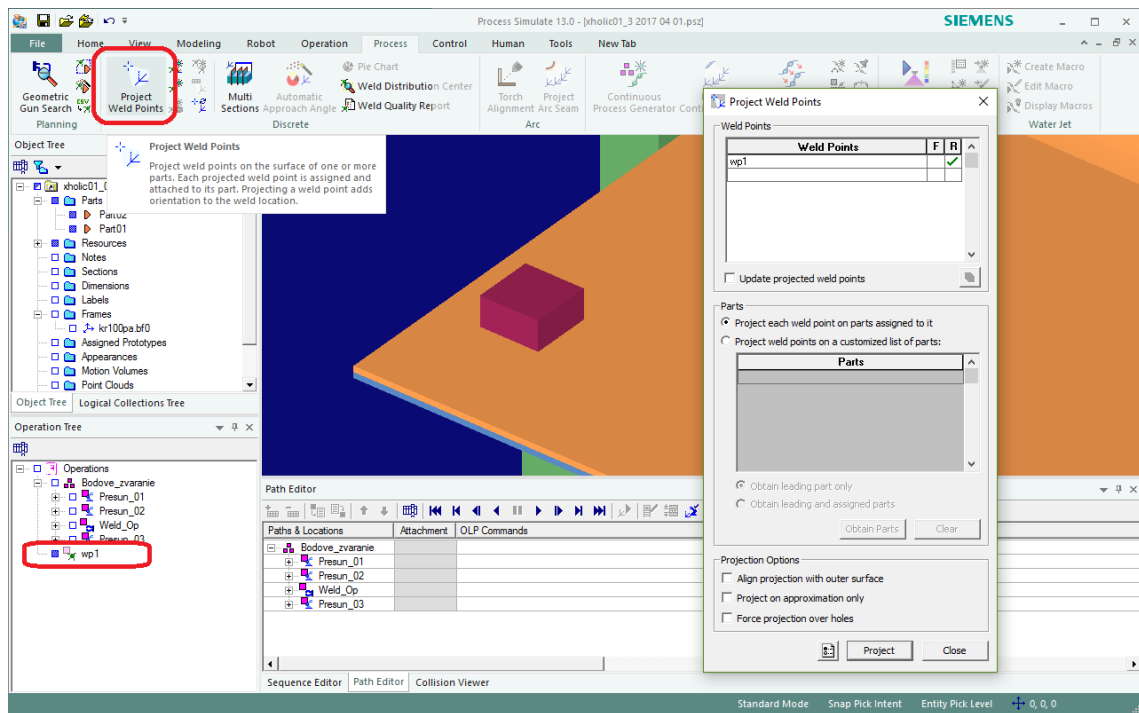
Na začiatok je potrebné nastaviť východzie vlastnosti pre operáciu bodového zvárania a predovšetkým čas, ktorý určuje dobu zváracej operácie. Tento čas je možné editovať v hlavných nastavenia programu Tecnomatix v záložke Weld.

Prvým krokom vytvorenia operácie bodového zvárania je súradnicové umiestnenie bodov na prípravku určeného na zváranie. Tvorba nového bodu je znázornená na obr. 2.32 a realizuje sa pomocou inštrukcie **Create Weld Point by Coordinates**, ktorá sa nachádza v záložke Process. V tejto inštrukcii je potrebné definovať názov nového bodu, miesto v ktorom sa má nový bod vytvoriť a súčiastku, na ktorej bude zvar realizovaný. Po potvrdení tlačidlom OK, sa na vybranom mieste zobrazí fialová kocka a v Operation Tree sa zobrazí operácia pod názvom nového bodu. Postupne sa zdefinujú všetky požadované miesta zvarov. V prípade tejto simulácie je určených celkom šesť bodov, tri na každej strane súčiastky slúžiacej na zváranie.



Obr. 2.32: Definovanie bodu určujúceho miesto bodového zvárania.

Všetky zadané body si označíme v Operation Tree a pomocou funkcie **Project Weld Points** v záložke Process z nich vytvoríme operácie s definovanými súradnicami, ktoré je možné v simulácii využívať na ovládanie robota k miestu zvaru. Funkcia je znázornená na obr. 2.33. V časti *Weld Points* sú zobrazené všetky body, ktoré sme vybrali. Toto nastavenie je možné upravovať pridaním alebo odobraním konkrétneho bodu. Po stlačení tlačidla Project program Tecnomatix automaticky zdefinuje súradnice operácii určujúcej miesto bodového zvaru. Pokiaľ všetko prebehlo

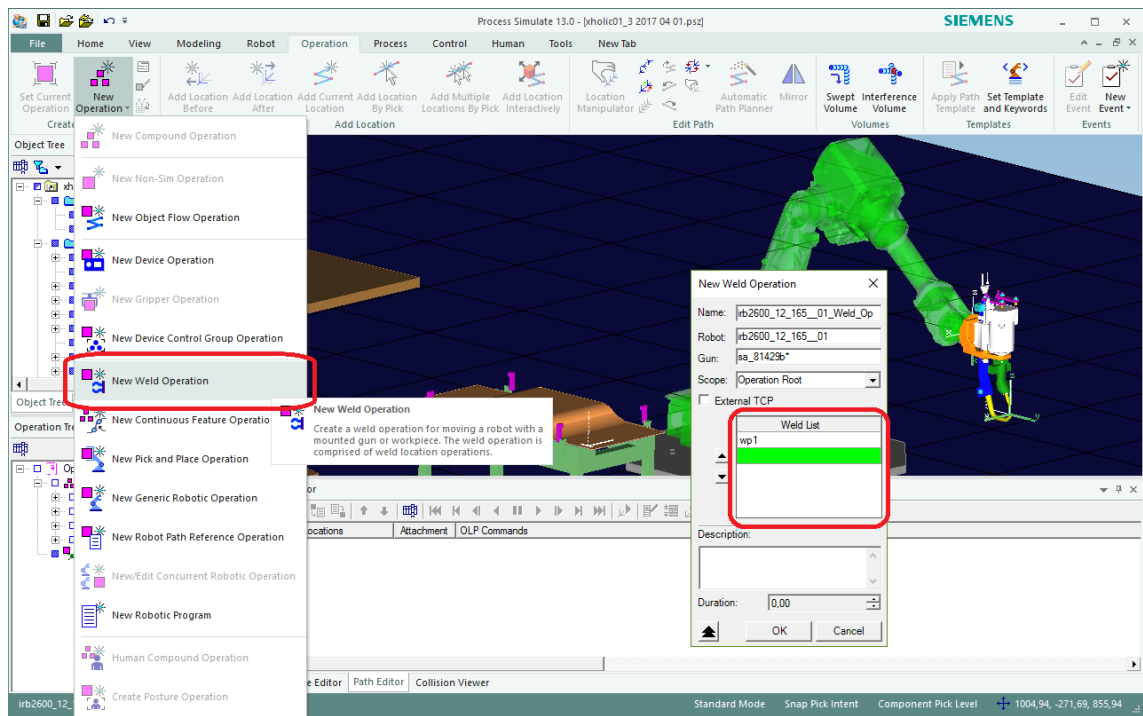


Obr. 2.33: Generovanie projekcie bodov určených pre bodové zvaranie.

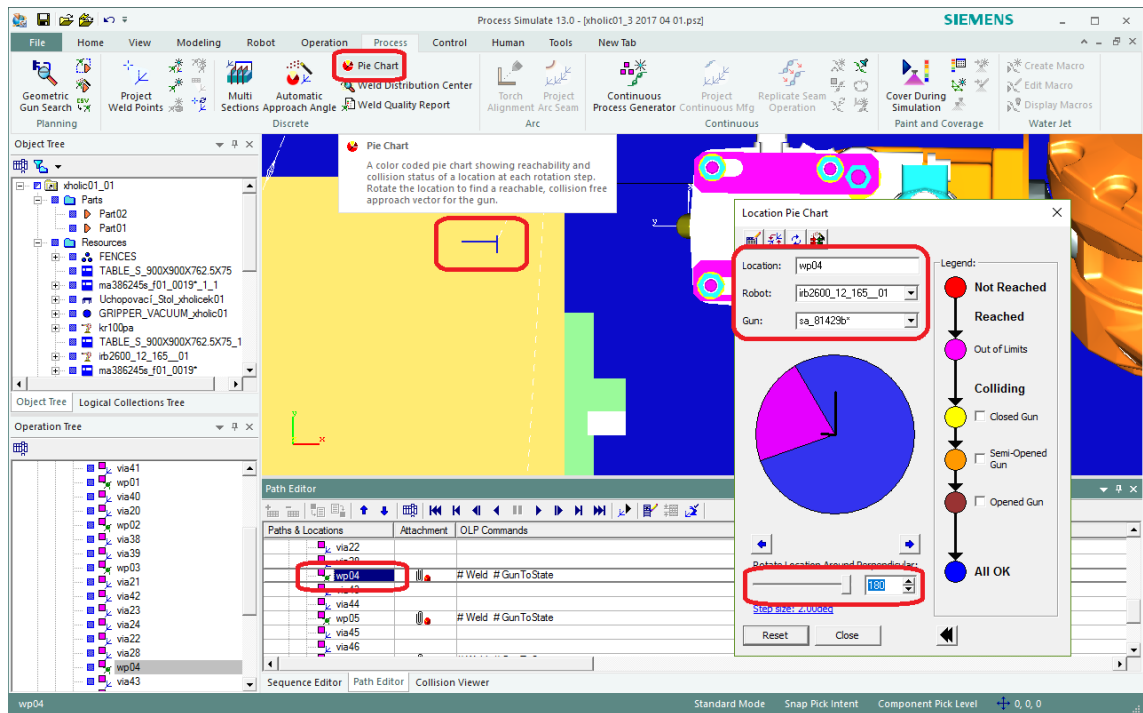
v poriadku, tak sa vedľa konkrétneho bodu v časti označenej písmenom R zobrazí zelená fajka a v časti Operation Tree sa zmení ikona operácie. Túto operáciu je možné teraz využívať v simulácii.

Ďalším krokom je vytvorenie novej operácie bodového zvarania pomocou príkazu **New Weld Operation** v záložke Operation. Príkaz je znázornený na obr. 2.34. V tomto príkaze je potrebné okrem názvu operácie, robota a nástroja, definovať aj miesta, v ktorých sa má operácia zvarania vykonávať. Tieto miesta sa definujú v časti Weld List a z Operation Tree sem vyberieme všetky body, ktoré sme vytvárali v predchádzajúcich krokoch.

Po potvrdení tlačidlom OK sa vytvorí nová sekvenciu operácií v Operation Tree, ktorú budeme ďalej upravovať tak, aby nedochádzalo ku kolíziám medzi robotom a ostatnými objektami simulácie. Stav, pri ktorom nedochádza ku kolíziám docielime pridávaním pozícií robota na požadované miesta a správnym natakáním lokácií miest určených na zvaranie tak, aby sme zabezpečili k týmto miestam robotovi prístup zo správnej strany. Natakávanie lokácií je možné pomocou funkcie **Pie Chart** v záložke Process. Funkcia je znázornená na obr. 2.35. V prípade tejto simulácie bolo potrebné upravovať natočenie iba pre tri lokácie, konkrétne o hodnotu 180°. Hodnota natočenia sa nastavuje v časti **Rotate Location Around Perpendicular**. S meniacou hodnotou natočenia, by sa mala natakáť aj čierna značka v koláčovom



Obr. 2.34: Vytvorenie novej operácie bodového zvarovania.



Obr. 2.35: Zmena natočenia hlavy robota pre bodové zvarovanie.

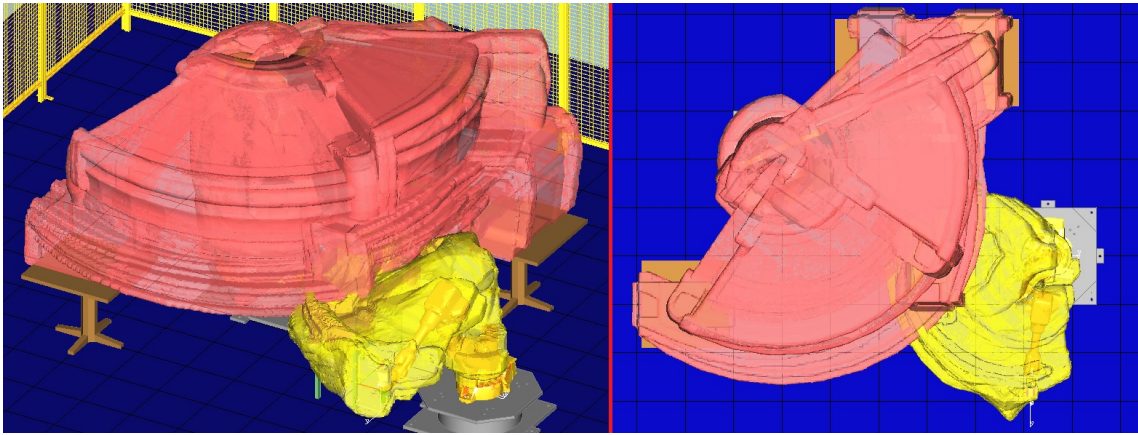
grafe funkcie. Pri natáčaní je potrebné pohybovať sa v modrej časti grafu. Fialová časť grafu znázorňuje natočenie, v ktorom už je lokácia mimo dosah a nie je možné túto operáciu s robotom vykonať.

Celý program s pozíciami a OLP príkazmi, pre simuláciu bodového zvarovania je znázornená na obr. 2.36.

Paths & Locations	Attachment	OLP Commands
Bodove_zvaranie		
Presun_01		
via 1		
via 2		
via		# Destination GRIPPER_VACUUM_xholic01 # Drive GRIP # Destination GRIPPER_VACUUM_xholic01 # WaitDevice GRIP # Grp TCPF1
via 3		
via 4		
via 5		# Destination GRIPPER_VACUUM_xholic01 # Drive RELEASE # Destination GRIPPER_VACUUM_xholic01 # WaitDevice RELEASE # Release TCPF1
via 6		
via 7		
Presun_02		
via 8		
via 9		
via 10		# Destination GRIPPER_VACUUM_xholic01 # Drive GRIP # Destination GRIPPER_VACUUM_xholic01 # WaitDevice GRIP # Grp TCPF1
via 11		
via 12		
via 13		
via 14		# Destination GRIPPER_VACUUM_xholic01 # Drive RELEASE # Destination GRIPPER_VACUUM_xholic01 # WaitDevice RELEASE # Release TCPF1
via 15		
via 16		
Weld_Op		
via 17		
via 25		
via 18		
via 19		
via 41		
wp01	🔥	# Weld # GunToState
via 40		
via 20		
wp02	🔥	# Weld # GunToState
via 38		
via 39		
wp03	🔥	# Weld # GunToState
via 21		
via 42		
via 23		
via 24		
via 22		
via 28		
wp04	🔥	# Weld # GunToState
via 43		
via 44		
wp05	🔥	# Weld # GunToState
via 45		
via 46		
wp06	🔥	# Weld # GunToState
via 47		
via 48		# Attach Part01 Part02
via 26		
via 27		
Presun_03		
via 30		
via 31		
via 32		# Destination GRIPPER_VACUUM_xholic01 # Drive GRIP # Destination GRIPPER_VACUUM_xholic01 # WaitDevice GRIP # Grp TCPF1
via 33		
via 34		
via 35		
via 35		# Destination GRIPPER_VACUUM_xholic01 # Drive RELEASE # Destination GRIPPER_VACUUM_xholic01 # WaitDevice RELEASE # Release TCPF1
via 36		
via 37		

Obr. 2.36: Program bodového zvarovania.

Podobne ako pri procese kontinuálneho zvarania, aj pri procese simulácie bodového zvarania som určil pomocou príkazu *Swept Volume* prekrytie dráh robotov. Toto prekryvanie dráh robotov pri simulácii je znázornené na obr. 2.37.

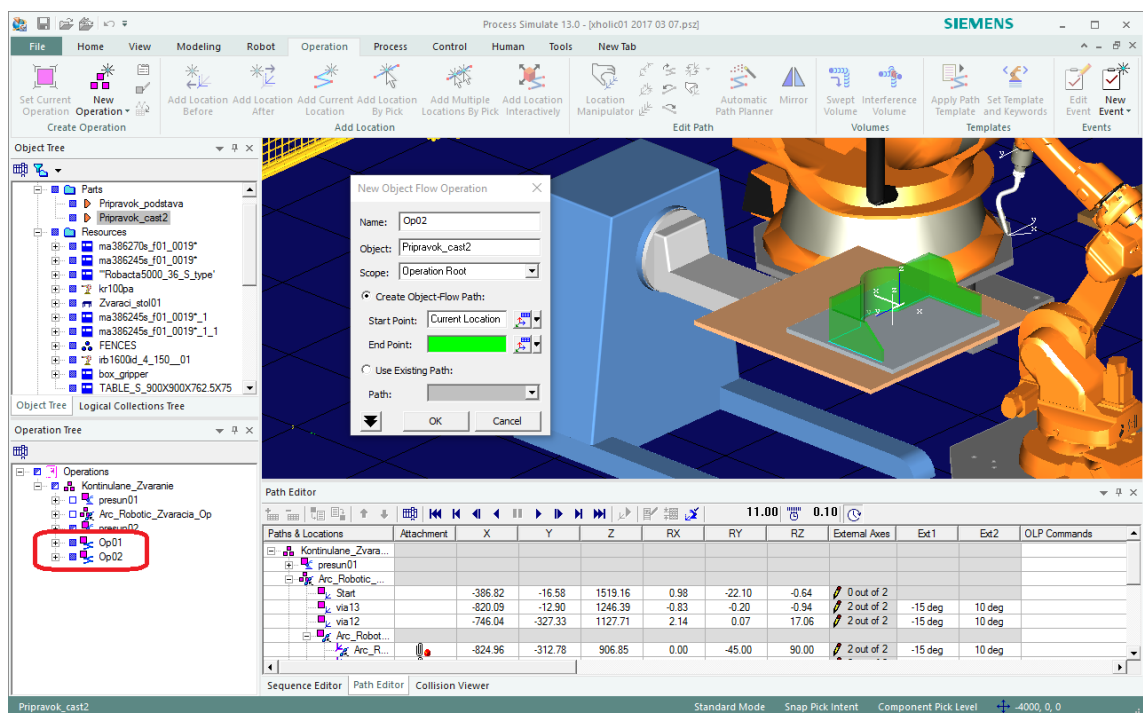


Obr. 2.37: Preĺňanie dráh oboch robotov pri procese bodového zvarania.

2.5 Virtuálne riadenie aplikácie

Ako som už spomínal na začiatku časti 2.4, virtuálne riadenie procesov sa odohráva v **Line Simulation Mode**. Pre oba procesy zvarovania je postup nastavovania komunikácie zhodný, a preto ďalej budem pracovať iba so simuláciou kontinuálneho zvarovania. V tomto móde však nie je možné pracovať s časťami **Parts**, preto bude našou prvou úlohou vytvorenie častí **Appearances**.

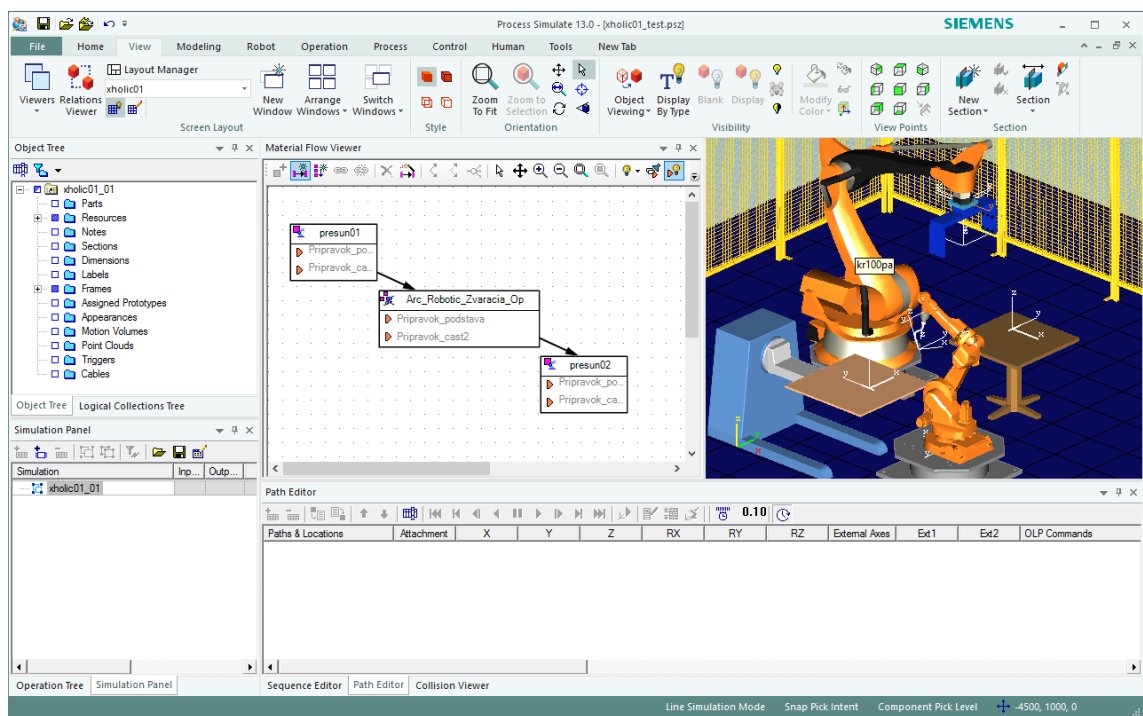
V Graphic Viewer alebo v Object Tree zvolíme súčiastku, z ktorej budeme chcieť vytvoriť časť Appearances. V záložke Operation vyberieme možnosť New Operation a pomocou funkcie **New Object Flow Operation** vytvoríme prázdnu operáciu pre požadovanú súčiastku. Na obr. 2.38 je znázornené okno funkcie. Je potrebné vyplniť iba časť name, object a scope. Rovnakú operáciu vytvoríme pre všetky súčiastky, ktoré budeme chcieť v simulácii ďalej používať. V prípade kontinuálneho zvarovania sú to dve časti, z ktorých pozostáva výrobok určený na zvaranie. Nové operácie sa musia nachádzať v podskupine KontinuálneZvaranie v časti **Operations Tree**, v inom prípade by nebolo možné prepnúť simuláciu do **Line Simulation Mode**.



Obr. 2.38: Funkcia New Object Flow Operation.

V záložke Home, v časti Study, sa pomocou ikony prepne do **Line Simulation Mode**. Po úspešnom prepnutí módu simulácie, prepne v záložke View simuláciu do **Advanced Simulation**. V časti Operation Tree nájdeme naše vytvorené operácie, z ktorých vytvoríme nové objekty Appearances. Pravým tlačidlom

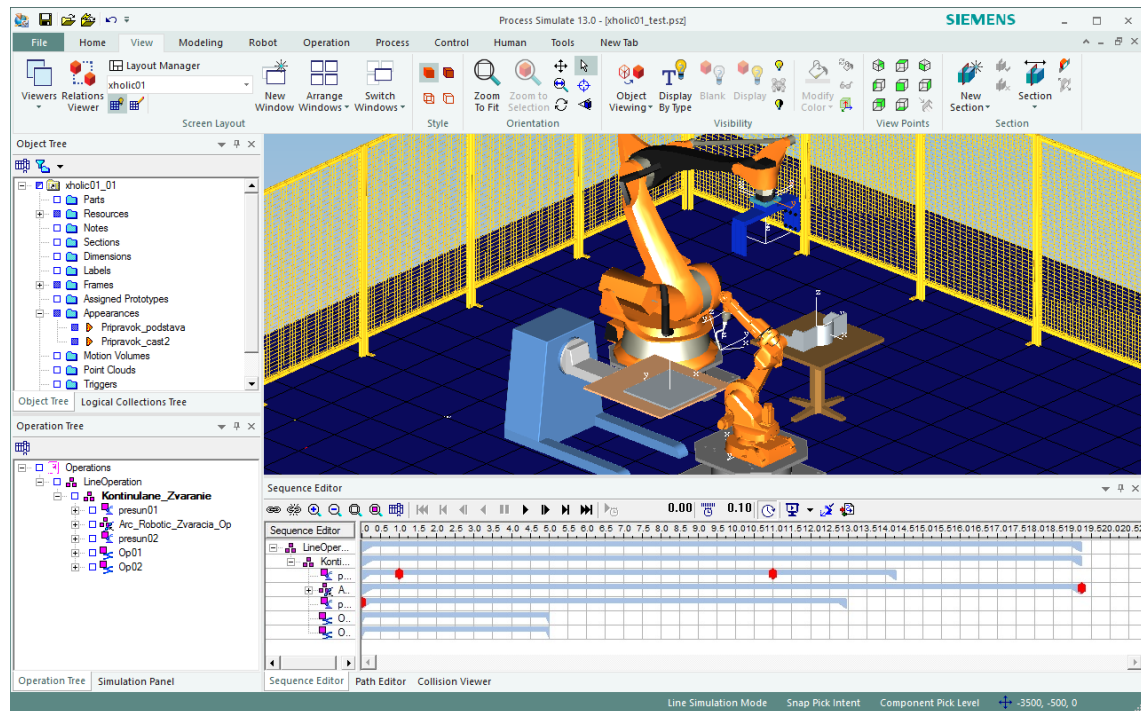
myši klikneme na obe operácie postupne a zvolíme možnosť **Generate Appearances**. Operáciu generovania Appearances je nutné vykonať iba raz. Pokiaľ nastavíme všetky ďalšie body správne, budú sa Appearances generovať samostatne, pri každom spustení simulácie. Nové objekty by sa mali okamžite objaviť v časti Object Tree v Appearances. Ďalšou úlohou je nastaviť operáciu, v ktorej sa má súčiastka automaticky generovať, nastaviť operácie, v ktorých sa má so súčiastkou pracovať a nastaviť operáciu, v ktorej má súčiastka zaniknúť. Toto všetko sa definuje v časti **Material Flow Viewer**. Do tohoto zobrazenia sa dostaneme v záložke View v časti Viewers. Príkazom **Generate Material Flow Links** sa vyberú všetky operácie, v ktorých sa bude so súčiastkou pracovať. Na obr. 2.39 je zobrazený celý cyklus toku objektov. V prvej operácii presun01 sa materiál generuje, v druhej operácii zvarovania sa s materiálom pracuje a v poslednej operácii presun02 materiál zaniká. Tok materiálu medzi operáciami znázorňujú čierne šípky.



Obr. 2.39: Material Flow Viewer.

V tomto okamihu je definované tvorba a zánik súčiastok Appearances, s ktorými je možné pracovať v **Line Simulation Mode**. Posledným krokom je pridanie príkazu **Attach** a **Deattach** pre vytvorené súčiastky, rovnako, ako tomu bolo pri statickej simulácii v **Standard Mode**. Najľahšia cesta je v Sequence Editor pomocou pravého kliknutia myši na danú operáciu v čase, kedy chceme akciu vykonať vybrať príkaz **Attach Event / Deattach Event**. Postupne pridáme k operáciám všetky príkazy, podobne ako v OLP Commands na obr. 2.28 v statickej simulácii. Na

obr. 2.40 je znázornený nový sled operácií, pričom červené kosoštvorce znázorňujú operácie Attach a Deattach.



Obr. 2.40: Sequence Editor v Line Simulation Mode.

Pokiaľ sme vykonali všetky potrebné nastavenia správne, tak je simuláciu procesu zvárania pripravená pre **Line Simulation Mode**.

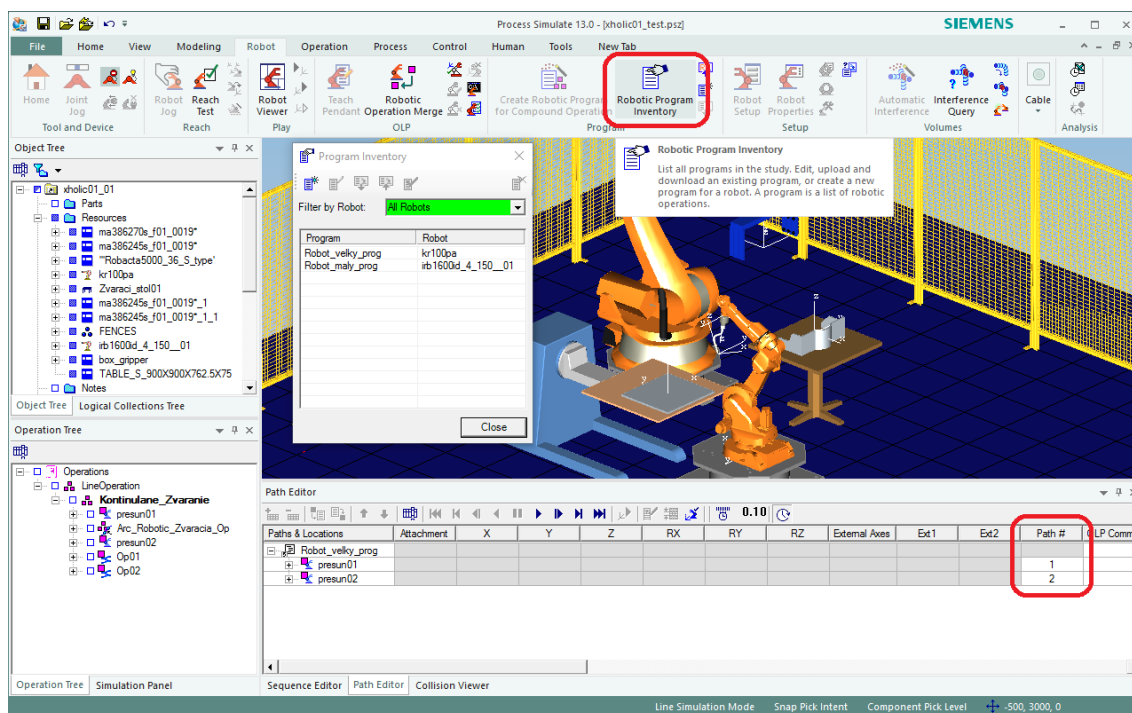
Pre riadenie simulácie je potrebné najskôr vygenerovať riadiace signály, pomocou ktorých sa budú roboti ovládať prostredníctvom PLC. Základné signály je možné vygenerovať pomocou funkcie **Robot Signals**. K tejto funkcii sa dostaneme kliknutím pravého tlačidla myši na určitého robota, pričom funkcia sa nachádza v časti **Robot Signals and Modules**. Vo funkcii Robot Signal je možné vytvárať signály jednotlivu alebo je možné vygenerovať všetky východzie signály robota naraz pomocou tlačidla **Create Default Signals**. Na obr. 2.41 sú znázornené všetky východzie signály, pomocou ktorých je možné robota ovládať. Signály je potrebné vytvoriť pre oba roboty zvlášť.

Ďalším krokom je vytvorenie programov pre oba roboty. Do týchto programov vložíme operácie, ktoré budú roboti vykonávať. Programy sa tvoria pomocou funkcie **Robotic Program Inventory**, ktorá sa nachádza v záložke Robot. Tlačidlom Create New Program sa vytvára nový program robota, v ktorom je potrebné zadať názov programu robota, pre ktorého program vytvárame a popri prípade komentár k programu. Tlačidlom Open in Path Editor sa program otvára v Path Editore,

PLC Signal	Signal Name	I/O	Signal Function	HW T...	Address	External Connection	Comments
Robot_velky_startProgram	startProgram	Q	Starting Program	BOOL	No Address	xholic01	
Robot_velky_programNum	programNumber	Q	Program Number	BYTE	No Address	xholic01	
Robot_velky_emergenStop	emergencyStop	Q	Program Emergency Stop	BOOL	No Address	xholic01	
Robot_velky_programPause	programPause	Q	Program Pause	BOOL	No Address	xholic01	
Robot_velky_programEnded	programEnded	I	Ending Program	BOOL	No Address	xholic01	
Robot_velky_mirrorProNum	mirrorProgramNumber	I	Mirror Program Number	BYTE	No Address	xholic01	
Robot_velky_errorProNum	errorProgramNumber	I	Error Program Number	BOOL	No Address	xholic01	
Robot_velky_robotReady	robotReady	I	Robot Ready	BOOL	No Address	xholic01	
Robot_velky_at_HOME	HOME	I	Pose Signal	BOOL	No Address	xholic01	

Obr. 2.41: Riadiace signály robota.

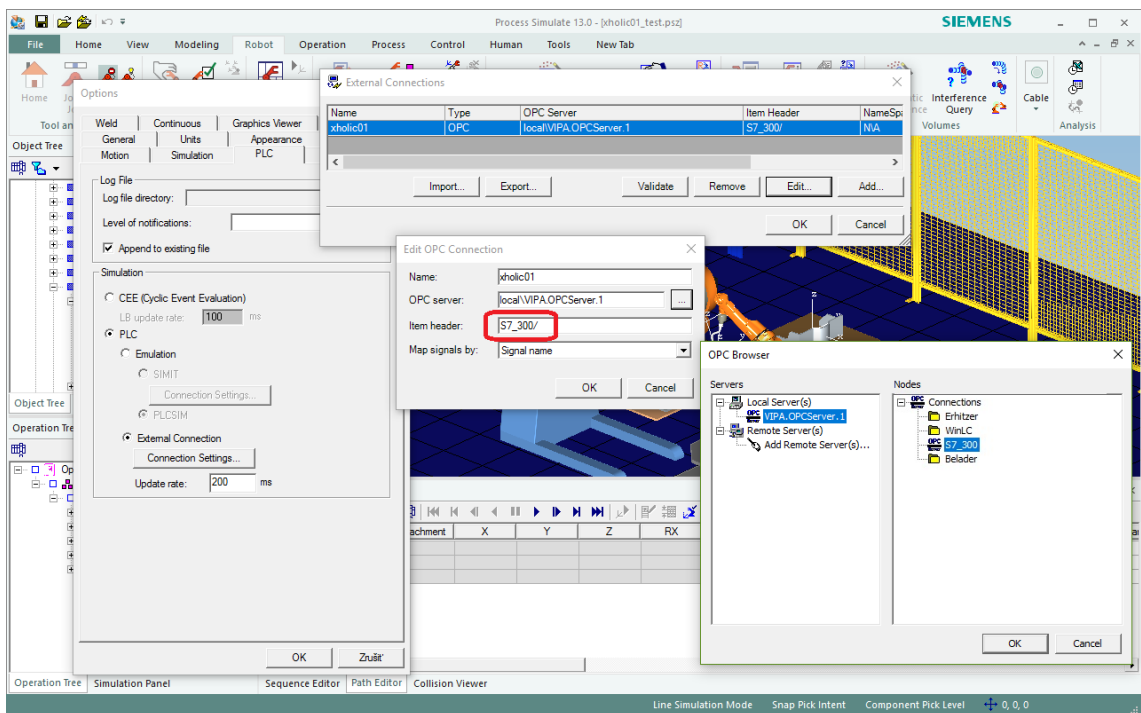
do ktorého pomocou myši pretiahneme požadované operácie robota, ktoré chceme prostredníctvom novovytvoreného programu ovládať. Týmto operáciám je potrebné určiť **Path**, aby bolo možné operácie spúšťať samostatne. Nastavenia sú znázornené na obr. 2.42. Po vytvorení všetkých riadiacich programov pre oba roboty a pridelení Path čísiel postupne nastavíme naše programy ako východzie pomocou tlačidla Set as Default Program. Východzie nastavené programy sa znázornia tučným písmom.



Obr. 2.42: Priradovanie operácií ku programu robota.

V prípade, ak chceme simuláciu riadiť prostredníctvom PLC, musíme v nastaveniach programu Tecnomatix Process Simulate nastaviť rozhranie, prostredníctvom

ktorého budeme komunikovať. Táto práca využíva na komunikáciu OPC server, ktorého nastavenie a spustenie je popísané **časti č. 3.1.1**. Nastavenie komunikácie v programe Tecnomatix sa realizuje v **Options** v záložke PLC. V prípade, ak na simuláciu nechceme využívať signály z PLC volíme v časti Simulation voľbu CEE (Cyclic Event Evaluation). V prípade, ak chceme využívať na riadenie externé signály vyberieme možnosť PLC - External Connection. Toto externé spojenie je však potrebné zdefinovať pomocou funkcie **Connection Settings**. Ďalej pomocou tlačidla Add pridáme nové OPC Connection. Pri vytvorení nového spojenia je potrebné určiť názov spojenia, podľa ktorého budeme v programe Tecnomatix Process Simulate pristupovať. Ďalej určiť OPC server z ponuky lokálnych serverov. Server musí byť spustený a správne nakonfigurovaný. A posledná časť, ktorú je nutné nastaviť je v časti Item header za názov riadiacej stanice pripísať znak „/“, pretože inak by pri spustení simulácie došlo k chybe. Nastavenie OPC spojenia je znázornené na obr. 2.43.



Obr. 2.43: Nastavenie komunikácie OPC Server v programe Tecnomatix.

Vytvorené signály je možné zobrazit, prípadne upravovať, v časti Signal Viewer. Tu sa definuje aj to, či budú signály simulované v programe Tecnomatix Process Simulate, alebo sa bude na riadenie simulácie pre konkrétny signál využívať PLC a typ komunikácie, ktorý sme definovali v predchádzajúcom odseku. Pokiaľ chceme signály riadiť prostredníctvom PLC, zaškrtneme možnosť PLC Connection pre daný signál.

Tento signál, však musí byť vytvorený v OPC serveri, v inom prípade by simulácia pri spustení hlásila chybu. V časti External Connection sa volí zadaný OPC server, ktorý chceme na riadenie využívať. Tento názov by sa mal zhodovať s názvom OPC servera, ktorý sme definovali v **Options** a jeho nastavenie je znázornené na predchádzajúcom obr. 2.43.

Signal Name	Memory	Type	Address	IEC Format	PLC Connection	External Connection	Resource	Comment
Robot_velky_emergenStop	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_programPause	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_programEnded	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_mirrorProNum	<input type="checkbox"/>	BYTE	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_errorProNum	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_robotReady	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_velky_at_HOME	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	kr100pa	
Robot_maly_startProgram	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_programNum	<input type="checkbox"/>	BYTE	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_emergenStop	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_programPause	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_programEnded	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_mirrorProNum	<input type="checkbox"/>	BYTE	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_errorProNum	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_robotReady	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	
Robot_maly_at_HOME	<input type="checkbox"/>	BOOL	No Address	No Address	<input checked="" type="checkbox"/>	xholic01	irb1600id_4	

Obr. 2.44: Signály využívané na riadenie simulácie.

Na obr. 2.44 je znázornený Signal Viewer spoločne so Simulation Panel, v ktorom je možné jednotlivé premenné simulovať, pokiaľ na riadenie daného signálu nevyužívame PLC. Pokiaľ na riadenie PLC využívame, je možné tu hodnotu premennej sledovať a kontrolovať, či sa simulácia správa podľa nastavených hodnôt.

Riadiaci program PLC využívaný na ovládanie simulácie kontinuálneho zvráťania je popísaný v časti č. 3.2.1.

3 RIADENIE MODELU LINKY POMOCOUCO PLC

Poslednou úlohou je vytvorenie riadiaceho programu pre ovládanie simulácie kontinuálneho zvarovania. Riadiaci program je zhotovený vo vývojovom prostredí SIMATIC Step7 a je určený pre PLC od firmy SIEMENS. V mojej aplikácii využívam PLC rady 300 a programovací jazyk STL (Statement List). Komunikačné rozhranie medzi riadiacim programom v PLC a simulačnými aplikáciami v Tecnomatix Process Simulate tvorí OPC server. Na vytvorenie OPC Servera používam program od spoločnosti VIPA OPC-Editor.

Konfigurácia OPC Servera je podrobne popísaná v časti č. 3.1.1 a tvorba riadiaceho programu je popísaná v časti č. 3.2. Navyiac v časti č. 3.3 je popísaný jednoduchý vizualizačný program vytvorený v prostredí WinCC flexible Advanced, slúžiaci na signalizáciu stavov a ovládanie simulácie.

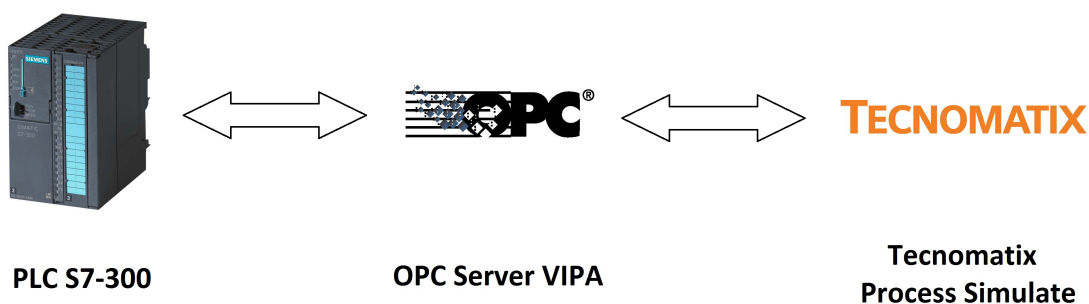
3.1 OPC Server

OPC Server je softvérový komunikačný štandard vytvorený pre priemyselnú automatizáciu, v ktorej zabezpečuje výmenu dát medzi softvérovými aplikáciami a hardvérovými zariadeniami. Dôvodom vzniku tohto štandardu bolo zlepšenie predávania dát medzi rôznymi aplikáciami na rôznych úrovniach podnikového systému. Ďalším dôvodom bolo umožnenie komunikovať medzi zariadeniami od rôznych výrobcov.

Tento typ komunikácie využíva architektúru **Klient-Server**, pričom server poskytuje dáta a klient spracováva požadované dáta, s ktorými vykonáva operácie. OPC Server môže komunikovať s viacerými OPC Klientmi naraz a naopak, OPC Klient dokáže spracovávať dáta od viacerých OPC Serverov.

Najväčšou výhodou OPC Servera je, že sa jedná o otvorený štandard, čo umožňuje viacerým výrobcom prispôbovať ich zariadenia pre tento spôsob komunikácie.

Na obr. 3.1 je znázornená schéma komunikácie využívaná v tejto diplomovej práci.



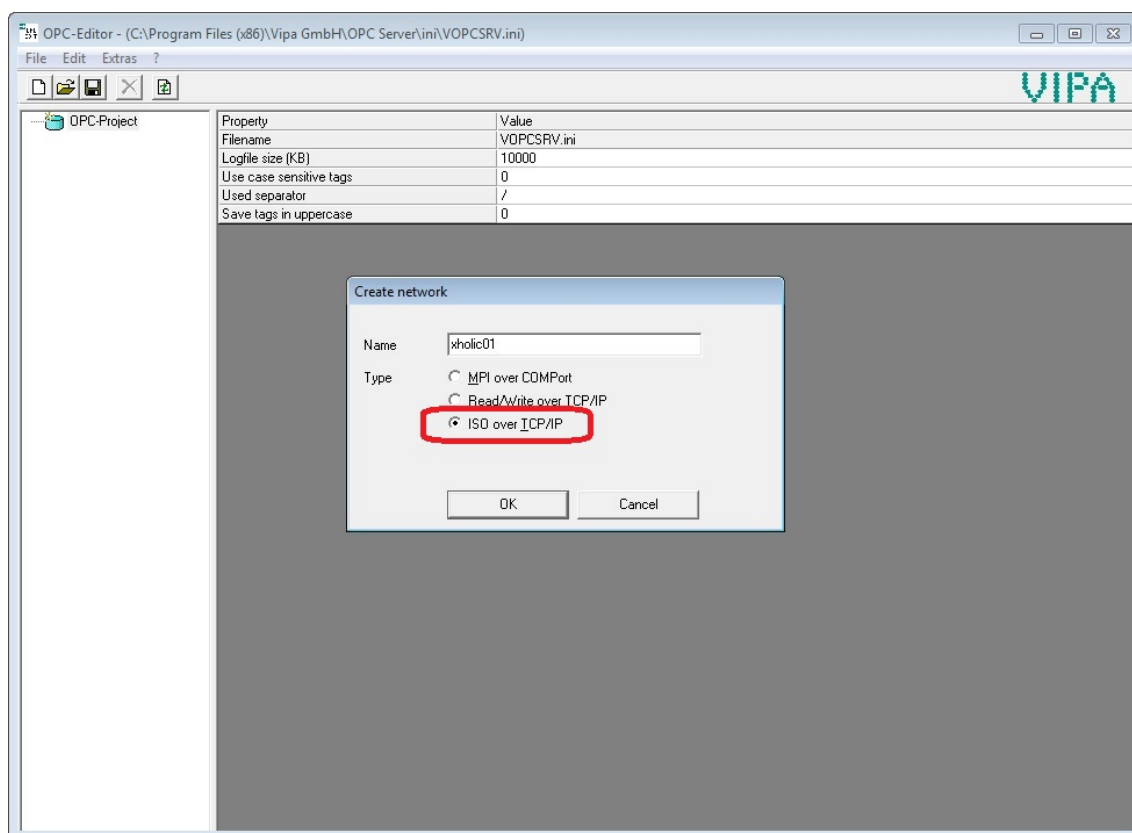
Obr. 3.1: Bloková schéma komunikácie.

3.1.1 VIPA OPC-Editor

V nasledujúcej časti je popísaný postup správneho nastavenia a spustenia OPC Servera od spoločnosti VIPA.

V prvom rade je potrebné spúšťať program OPC-Editor ako správca, v inom prípade sa nám server nepodarí sprevádzkovať. Po vytvorení nového projektu v záložke File je ďalšou úlohou pridanie siete, napríklad pomocou kliknutia pravým tlačidlom myši na novovytvorený projekt.

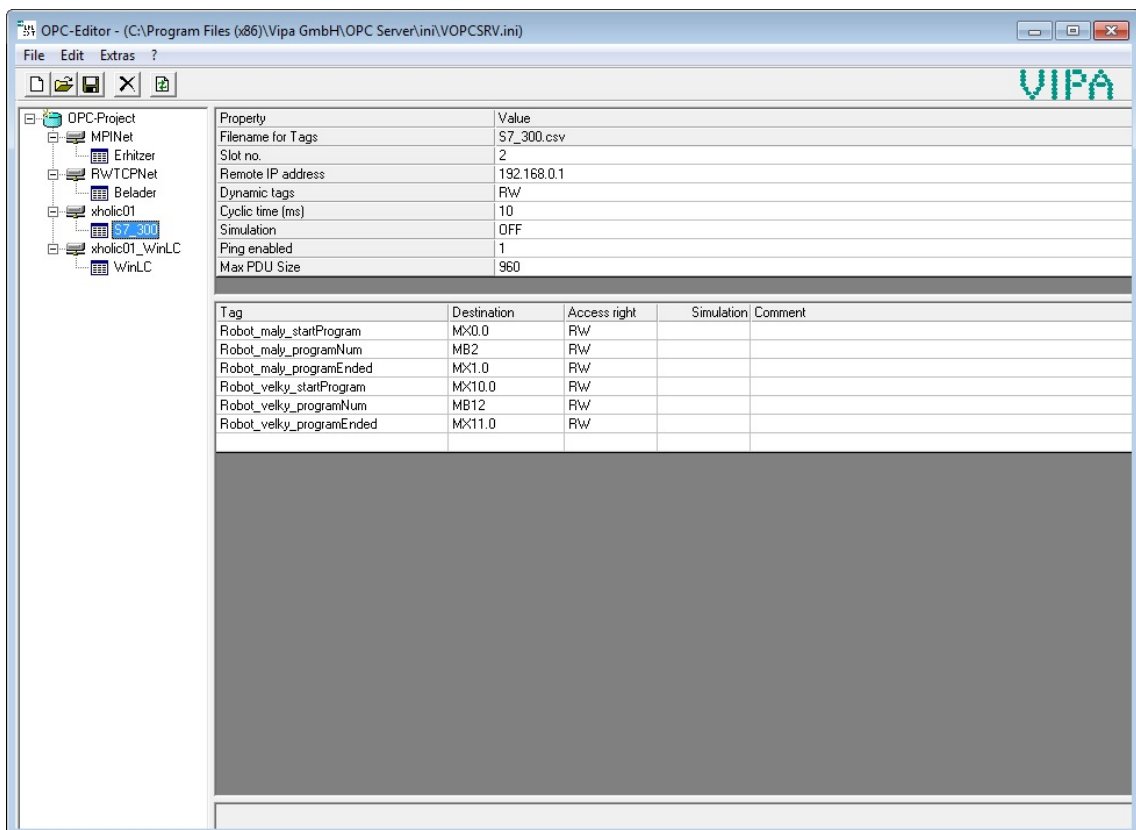
Nastavenie siete je znázornené na obr. 3.2. Ako prvé sa volí názov siete a následne typ komunikácie po sieti. V aplikácii využívam typ **ISO over TCP/IP**. Po vytvorení siete určíme pevnú IP (Internet Protocol) adresu počítača a túto adresu vpíšeme aj do kolónky **Local IP address**, v časti pre nastavenie siete.



Obr. 3.2: Vytvorenie siete v programe OPC-Editor.

Po vytvorení siete je ďalším krokom pridanie PLC stanice, pomocou ktorej budeme vytvorené aplikácie riadiť. Názov riadiacej stanice volíme rovnaký ako názov PLC v programe SIMATIC Step7. V prípade, ak server komunikuje s viacerými pracoviskami tým zabezpečíme, že pri neočakávanej poruche na serveri určíme rýchlejšie miesto a príčinu poruchy.

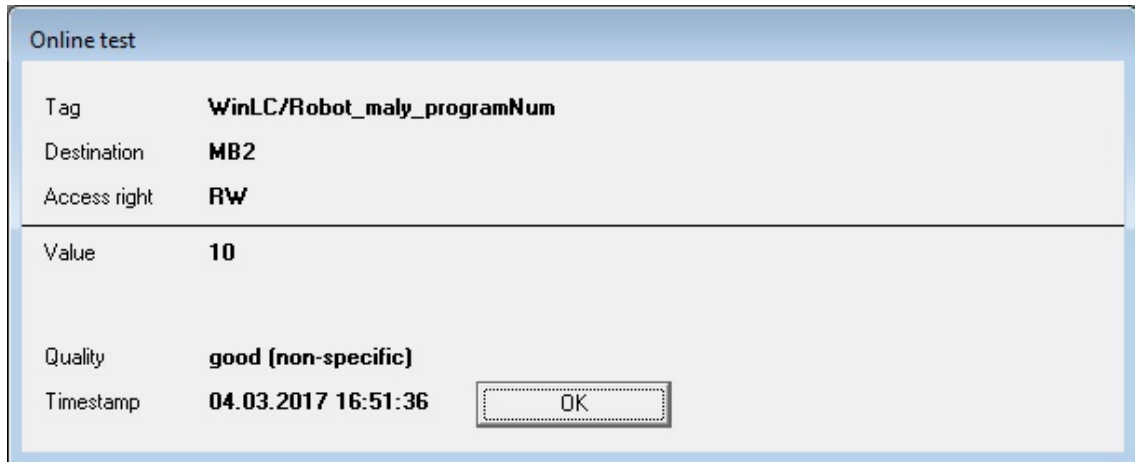
Na nasledujúcom obr. 3.3 je zobrazené nastavenie pridanej PLC stanice. Do časti **Tag** sa definujú premenné, ktoré sa v komunikácii budú využívať. Opäť je potrebné, aby mali premenné rovnaký názov ako premenné v programe SIMATIC Step7. V časti **Destination** sa určuje ich adresa z riadiaceho PLC. Je možné pracovať so vstupnými, výstupnými a aj merkerovými hodnotami. V časti **Access right** sa definuje právo na vykonávanie akcií s premennou. V mojom prípade sú všetky premenné nastavené ako **RW** čo znamená, že sú určené na čítanie aj na zápis. Zoznam s definovanými premennými je možné upravovať aj v externom programe. Tento zoznam sa nachádza v súbore, ktorý je uvedený v kolónke **Filename for Tags** a má koncovku csv. V časti **Slot no.** je definované číslo slotu v zostave stanice, na ktorom sa nachádza procesor riadiaceho PLC. Toto číslo sa opäť musí zhodovať s číslom slotu v hardvérovej konfigurácii programu SIMATIC STEP7. Posledný parameter, ktorý je potrebné upraviť je v kolónke **Remote IP address**. Na toto miesto je potrebné vložiť IP adresu riadiaceho PLC.



Obr. 3.3: Nastavenie PLC stanice v programe OPC-Editor.

Po definovaní všetkých údajov je potrebné nastavenia uložiť a to najlepšie do východzej zložky. Posledným krokom je už iba spustenie servera. To je možné napríklad pomocou ikony **Reconfigure OPC Server**, ktorá je umiestnená v ľavej

hornej časti. Po spustení servera môžeme overiť jeho funkčnosť pomocou príkazu **Check online value**, ktorý je možné zvoliť po kliknutí pravým tlačidlom myši na ľubovoľnú premennú. Avšak v prípade testovania je potrebné, aby bol užívateľ pripojený pomocou ethernetového spojenia na PLC stanicu, v ktorej je nahraný program s premennými, ktoré majú definované rovnaké symbolické názvy ako premenné v OPC serveri.



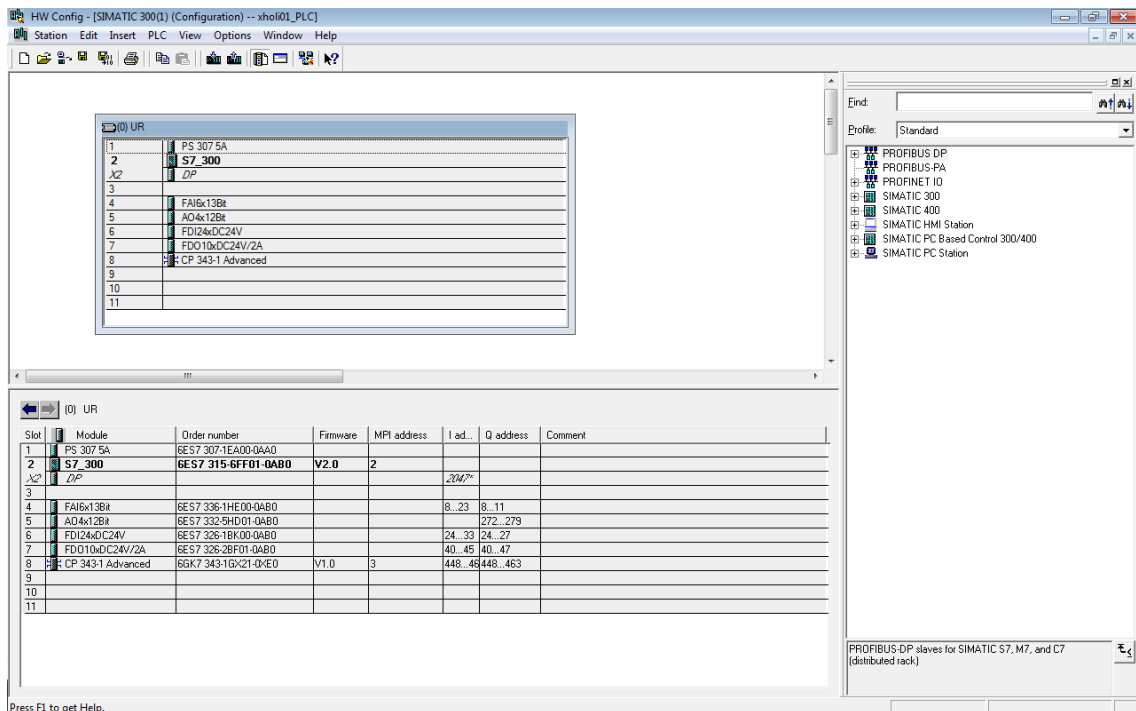
Obr. 3.4: Online test OPC Servera.

V prípade, ak sa nepodarí server spustiť alebo výsledok online testu premenných nedopadol úspešne, je možné pomocou **Log files** v záložke Extras prísť na chybu a pokúsiť sa ju vyriešiť.

3.2 Konfigurácia PLC

Ako som spomínal v úvode tejto práce, riadiaci automat slúžiacim na ovládanie simulačného procesu bude PLC od firmy SIEMENS. Na programovanie PLC od tohto renomovaného výrobcu je možné využiť program SIMATIC Step7. V tejto časti si popíšeme spôsob konfigurácie riadiaceho PLC tak, aby bola zabezpečená jeho správna funkčnosť.

V mojej aplikácii na riadenie využívam PLC SIMATIC rady 300, ktoré sa nachádza v laboratóriách ústavu automatizácie a meracej techniky. Hardvérová konfigurácia je znázornená na obr. 3.5. Môžeme vidieť, že zostava riadiaceho automatu pozostáva z viacerých vstupných a výstupných kariet. Tieto karty v mojej práci nevyužívam, no pre správny chod PLC je potrebné presne zadefinovať všetky karty zostavy. Na nahratie programu z počítača využívam komunikáciu MPI (Message Passing Interface). Túto komunikáciu je potrebné definovať v karte, ktorá sa nachádza na slotě č. 2, teda karta samotného procesora PLC. V Slotě č. 8 sa nachádza komunikačná karta CP 343-1 Advanced, v ktorej je potrebné definovať ethernetovú komunikáciu. Použitá IP adresa, by sa mala zhodovať s adresou zadávanou do OPC servera.



Obr. 3.5: HW konfigurácia riadiaceho PLC.

Ďalším krokom je definovanie názvov premenných, ktoré budeme v programe využívať, aby sme docielili prehľadnosť výsledného riadiaceho programu. Všetky

premenné, ktoré využívam sú typu merker, teda vnútorné premenné. Ich prehľad sa nachádza na obr. 3.6.

	Address	Symbol	Display format	Status value	Modify value
1	MB 0	"Robot_maly_programNum"	DEC	0	
2	M 2.0	"Robot_maly_startProgram"	BOOL	false	
3	M 2.1	"Robot_maly_emergenStop"	BOOL	false	
4	M 2.2	"Robot_maly_programPause"	BOOL	false	
5	M 2.3	"Robot_maly_programEnded"	BOOL	false	
6	M 2.4	"Robot_maly_errorProNum"	BOOL	false	
7	M 2.5	"Robot_maly_robotReady"	BOOL	false	
8	M 2.6	"Robot_maly_at_HOME"	BOOL	false	
9	MB 4	"Robot_maly_mirrorProNum"	DEC	0	
10					
11	MB 10	"Robot_velky_programNum"	DEC	1	
12	M 12.0	"Robot_velky_startProgram"	BOOL	true	
13	M 12.1	"Robot_velky_emergenStop"	BOOL	false	
14	M 12.2	"Robot_velky_programPause"	BOOL	false	
15	M 12.3	"Robot_velky_programEnded"	BOOL	false	
16	M 12.4	"Robot_velky_errorProNum"	BOOL	false	
17	M 12.5	"Robot_velky_robotReady"	BOOL	false	
18	M 12.6	"Robot_velky_at_HOME"	BOOL	false	
19	MB 14	"Robot_velky_mirrorProNum"	DEC	1	
20					
21	MB 20	"Krok"	DEC	1	
22	M 21.0	"Start"	BOOL	true	
23	M 21.1	"Stop"	BOOL	false	
24	M 21.2	"Pause"	BOOL	false	
25	M 21.3	"Restart"	BOOL	false	
26					
27	M 22.0	"Pause_status"	BOOL	false	
28	M 22.1	"Run_cmd_PLC"	BOOL	false	
29	M 22.2	"Run_status"	BOOL	true	
30					

Obr. 3.6: Tabuľka použitých premenných.

3.2.1 Riadiaci program

Program, ktorým riadim simuláciu procesu, využíva na riadenie signály popísané v predchádzajúcej časti. Tento riadiaci program pozostáva z viacerých blokov:

- OB1 - cyklické volanie funkčných blokov FC1, FC2 a FC3
- OB100 - volanie funkčného bloku FC100 pri zapnutí napájania automatu
- FC1 - hlavný program definujúci činnosť programu pri jednotlivých krokoch
- FC2 - program zabezpečujúci zastavenie a opätovné spustenie pri povelu Pauza
- FC3 - program zabezpečujúci sekvenciu prepínania medzi krokmi programu
- FC100 - program zabezpečujúci inicializáciu pri spustení automatu

V nasledujúcej časti je popísaná základná funkcia jednotlivých FC blokov riadiaceho programu.

FC1 - hlavný program

- Krok 0 - inicializácia merkov do počiatočného stavu
- Krok 11 - spustenie operácie veľkého robota - Presun01
- Krok 1 - medzikrok operácie Presun01
- Krok 12 - spustenie operácie malého robota - Operácie zvárania
- Krok 2 - medzikrok operácie zvárania
- Krok 13 - spustenie operácie veľkého robota - Presun02
- Krok 3 - medzikrok operácie Presun02
- Krok 4 - nastavenie programu do počiatočného stavu po ukončení celého cyklu simulácie

Pri každom spustení operácie v krokoch 11, 12 a 13 sa spúšťa robota, s ktorým sa má daná operácia vykonávať a nastavuje sa číslo operácie, ktorú chceme, aby robot vykonával. Toto číslo operácie, by sa malo zhodovať s číslom **Path**, ktoré sme určili v programe Tecnomatix pri nastavovaní dynamickej simulácie. Definovanie Path je znázornené na obr. 2.42 v časti 2.5.

FC2 - pauza simulácie

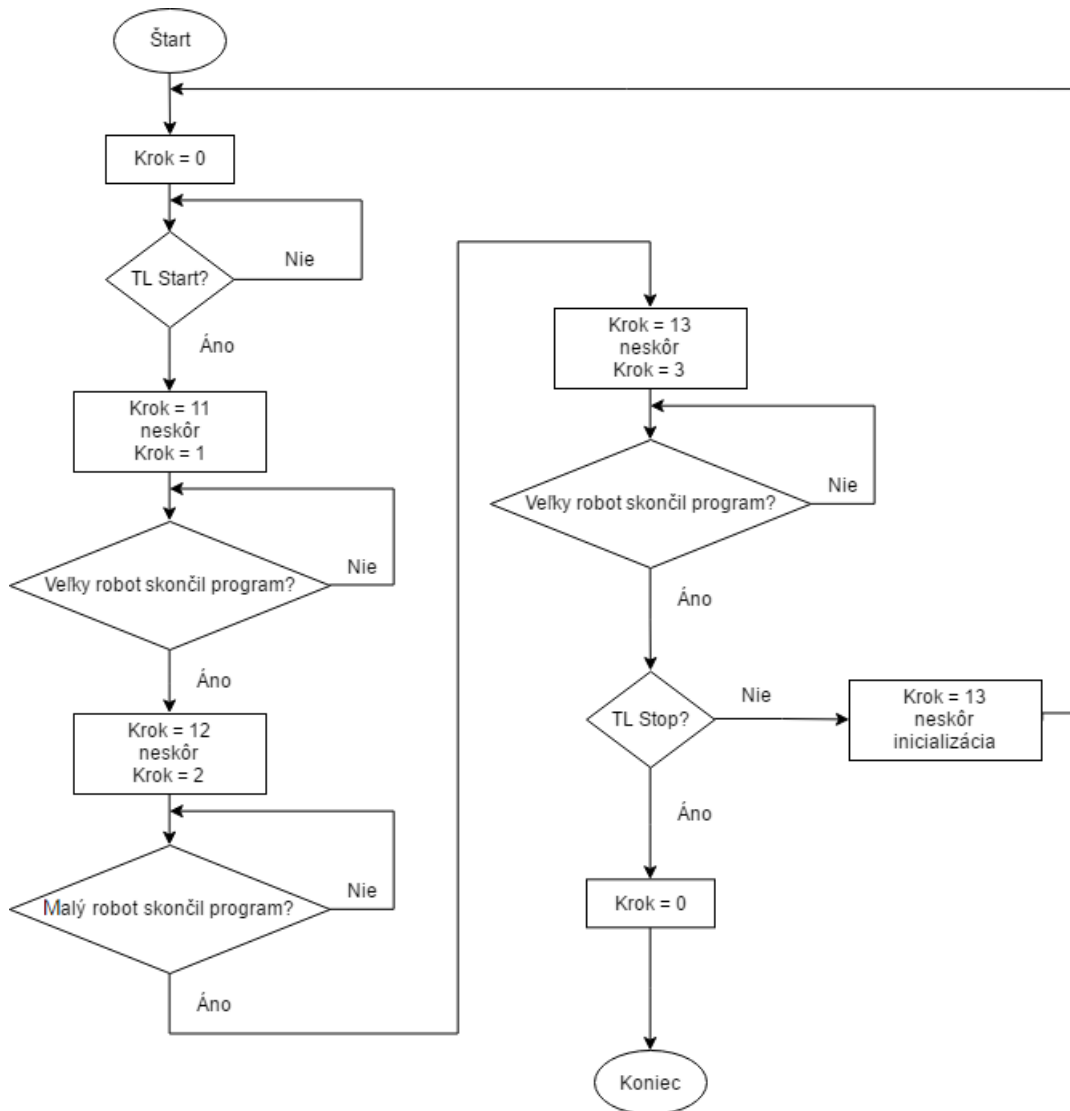
V tejto časti programu je riešené zastavenie a opätovné spustenie simulácie pomocou riadiacich signálov robotov.

FC100 - inicializácia pri spustení

Nastavenie premenných do počiatočného stavu pri zapnutí napájania riadiaceho PLC.

FC3 - prepínanie medzi krokmi programu

Signálom štart sa simulácia uvedie do činnosti. Prepínanie medzi simulovanými operáciami je riešené pomocou signálov robotov, ktoré signalizujú bezpečné ukončenie danej operácie. Po prijatí tejto informácie, program prejde do ďalšieho kroku a opäť sa čaká na signál ukončenia operácie od príslušného robota. Signálom Stop sa program nezastaví hneď, ale až po prevedení celého jedného simulačného cykľusu. Na okamžité zastavenie simulácie slúži signál Emergency_stop_cmd, poprípade signál Pause.



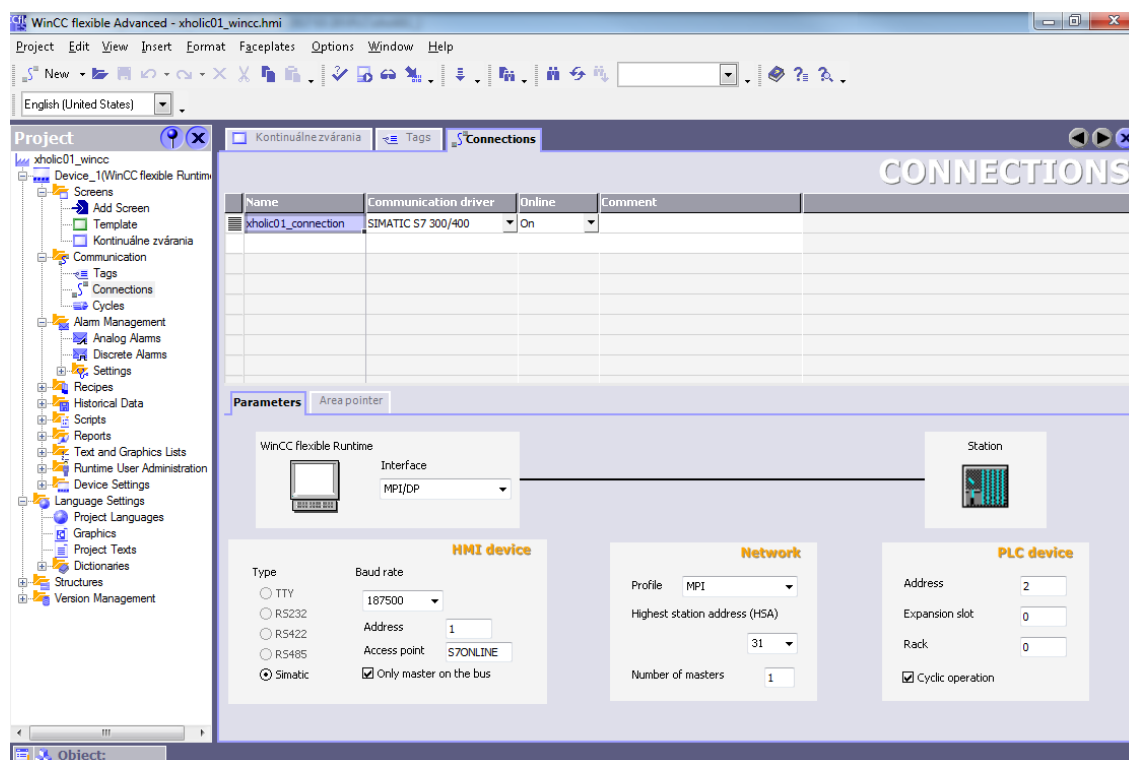
Obr. 3.7: Vývojový diagram funkcie FC3 - prepínanie medzi krokmi programu.

Celý riadiaci program sa nachádza v prílohách tejto diplomovej práce.

3.3 Vizualizácia simulačného procesu

Posledná časť práce sa venuje zhotoveniu vizualizačného programu, pomocou ktorého je možné ovládať simulovaný proces v programe Tecnomatix. Tento vizualizačný program som zhotovil v prostredí WinCC flexible Advanced.

Prvým krokom po vytvorení nového projektu v programe WinCC je správne definovanie spojenia cez, ktoré bude riadiace PLC a zariadenie, na ktorom je spustený vizualizačný program komunikovať. Táto aplikácia využíva na prenos riadiacich signálov komunikáciu typu MPI. Nastavenie komunikácie je znázornené na obr. 3.8.

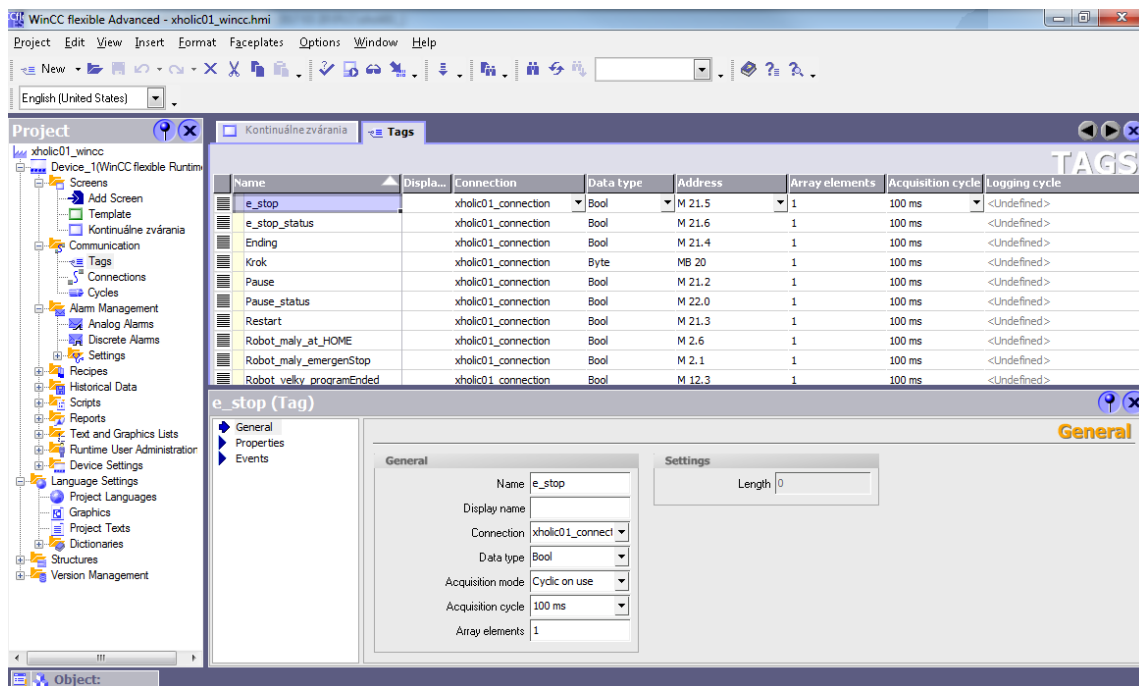


Obr. 3.8: Definovanie komunikácie v programe WinCC flexible Advanced.

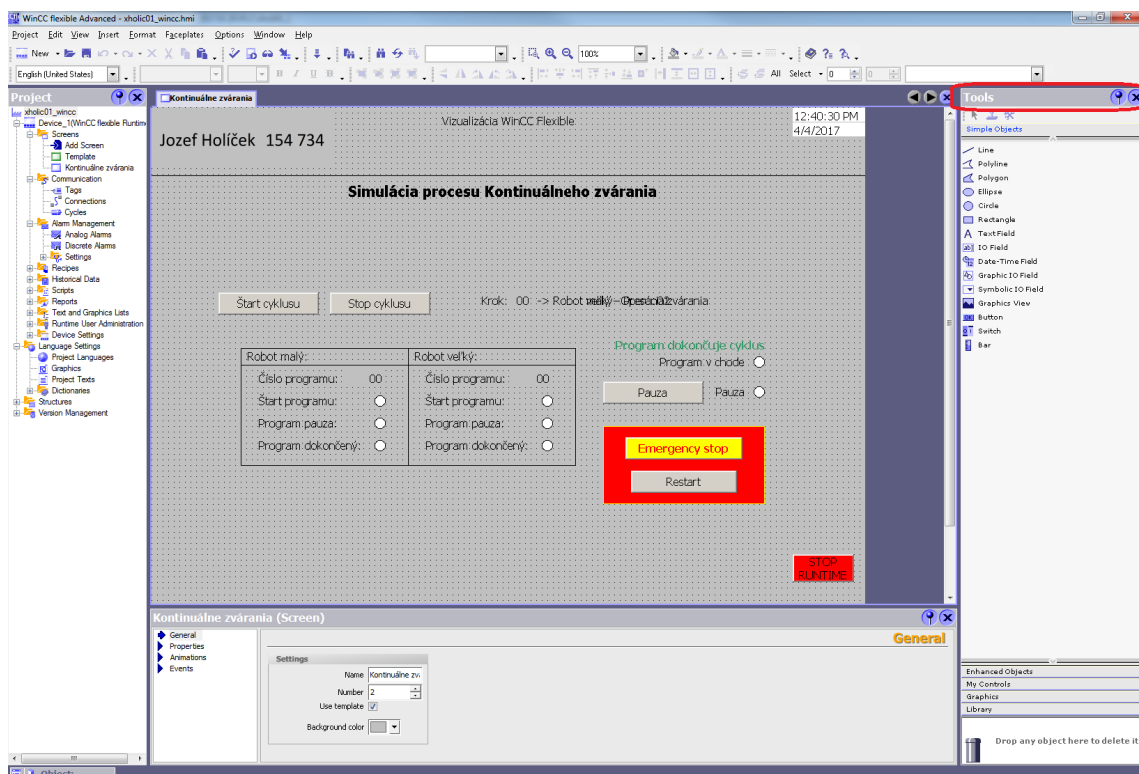
Ďalším krokom je definovanie všetkých premenných, ktoré budeme na ovládanie využívať. Tieto premenné sa definujú do tabuľky **Tags**. Pre prehľadnosť a uľahčenie programovania vizualizácie doporučujem používať názvy premenných rovnaké, ako v riadiacom programe, ktorý je zhotovený vo vývojovom prostredí Step7. Na obr. 3.9 je zobrazený príklad definovania premenných v programe WinCC.

Posledným a najdôležitejším krokom je správne definovanie grafických objektov a ich vlastností. V aplikácii využívam základné objekty ako textové polia, tlačidlá a signalizačné svetlá. Všetky základné objekty sa nachádzajú v záložke **Tools**, v pravej časti vývojového prostredia.

Vizualizačný program sa nachádza v prílohách tejto diplomovej práce.



Obr. 3.9: Definovanie premenných v programe WinCC flexible Advanced.



Obr. 3.10: Tvorba vizualizačného programu v prostredí WinCC flexible Advanced.



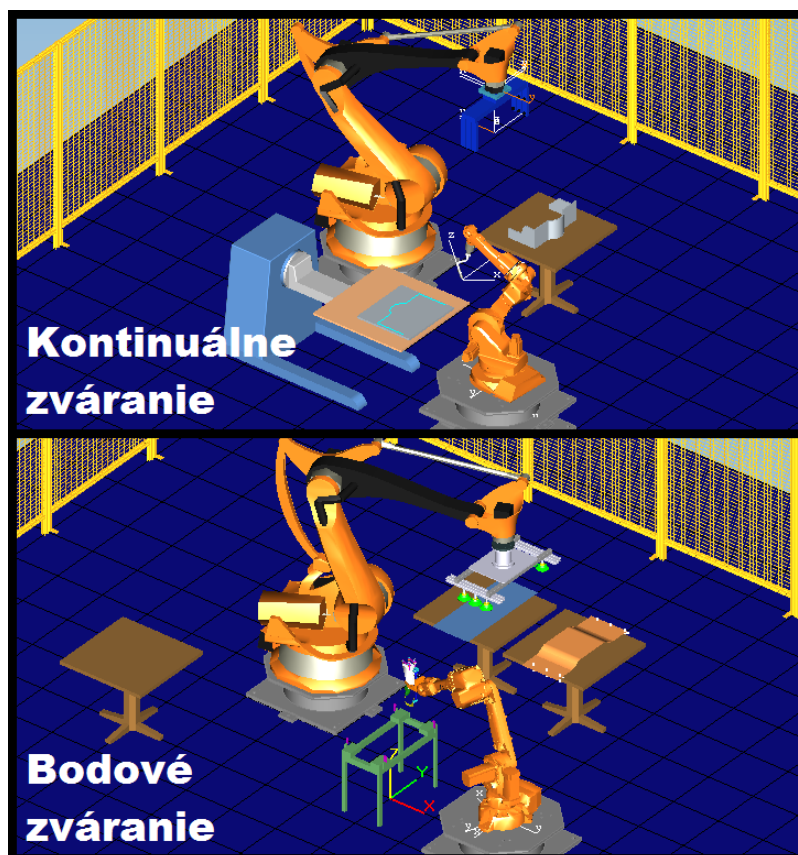
Obr. 3.11: Runtime vizualizačného programu.

4 ZÁVER

Cieľom tejto diplomovej práce, bolo oboznámenie sa s programom Tecnomatix Process Simulate. Je to program, ktorý slúži na modelovanie a simulovanie výrobných procesov, pričom jednou z hlavných úloh bolo oboznámenie sa s virtuálnym riadením stacionárnych priemyselných robotov slúžiacich na kontinuálne a bodové zváranie. Program Tecnomatix Process Simulate pritom poskytuje možnosť simulovania viacerých rôznych operácií a jeho portfólio inštrukcií je veľmi široké.

Podarilo sa mi zhotoviť dva modely výrobných liniek, simulujúce kontinuálne a bodové zváranie prípravkov. V oboch vytvorených simulačných pracoviskách sa dráhy programovaných robotov prekrývajú, pričom medzi robotmi a predmetmi v ich dosahových vzdialenostiach nedochádza ku kolíznym stavom.

Obe virtuálne pracoviská, ktoré som zostrojil v tejto diplomovej práci sú znázornené na obr. 4.1. Tieto pracoviská pozostávajú z dvoch stacionárnych robotov. Väčší z robotov je určený na manipulovanie s materiálom a menší robot je určený na zváranie. Ďalej sa na pracoviskách nachádzajú mnou vymodelované zvaracie stoly, ktoré slúžia na uchytenie a manipuláciu s materiálmi určenými na zváranie.



Obr. 4.1: Zhotovené pracoviská kontinuálneho a bodového zvárania.

Simuláciu kontinuálneho zvarovania som navyše prispôbil tak, aby bolo možné pracovisko ovládať pomocou PLC SIMATIC rady 300. Taktiež som pre toto PLC vytvoril riadiaci program, ktorým sa jednotlivé operácie robotov spúšťajú podľa požadovanej sekvencie. Na komunikáciu medzi PLC a počítačom so simulačným programom využívam rozhranie OPC servera.

Funkciu programu som následne overil pomocou vizualizačného programu zhotoveného v prostredí WinCC flexible Advanced.

Výsledkom tejto práce je návod, pomocou ktorého je možné zostrojiť simuláciu kontinuálneho alebo bodového zvaracieho procesu.

LITERATÚRA

- [1] Digitální podnik. Digital Factory [online]. Plzeň, 2011 [cit. 2016-12-23]. Dostupné z URL: <<http://digipod.zcu.cz/index.php/cs/digitalni-tovarna>>.
- [2] PÁSEK, Jan a Vlastimil BRAUN. Automatizace procesů II: Úroveň řízení výroby. FEKT VUT v Brně, 2014.
- [3] Tecnomatix 1: Co je digitální továrna? [online]. Brno: Marek Pagáč, 2012 [cit. 2016-11-07]. Dostupné z URL: <<http://www.konstrukter.cz/2012/06/29/tecnomatix-1-co-je-digitalni-tovarna/>>.
- [4] Tecnomatix [online]. Batislava: SOVA Digital a.s., 2014 [cit. 2016-12-23]. Dostupné z URL: <<http://www.ova.sk/sk/produkty/tecnomatix>>.
- [5] Manipulators: Additional Robot Axes [online]. Hohenstein-Ernstthal: FÖRSTER welding system, 2011 [cit. 2016-11-10]. Dostupné z URL: <<http://www.forster-welding-systems.com/en/manipulators/additional-robot-axes.html>>.
- [6] Industrial Robots [online]. New Taipei City: IELTS in Taiwan, 2015 [cit. 2016-11-10]. Dostupné z URL: <<https://ieltsintaiwan.wordpress.com/2015/11/17/ielts-reading-yes-no-not-given-industrial-robots/>>.
- [7] OPC DataHub: What is OPC? [online]. Ontario, 2010 [cit. 2017-03-01]. Dostupné z URL: <<http://www.opcdatahub.com/WhatIsOPC.html>>.
- [8] GRABCAD COMMUNITY: VACUUM GRIPPER ROBOT [online]. ccrdgb, 2015 [cit. 2017-04-01]. Dostupné z URL: <<https://grabcad.com/library/vacuum-gripper-robot>>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

CAD	Computer Aided Design
PLC	Programmable Logic Controller
OPC	Object Linking and Embedding for Process Control
STL	Statement List
IP	Internet Protocol
CEE	Cyclic Event Evaluation
MPI	Message Passing Interface

A OBSAH PRILOŽENÉHO CD

Všetky priložené simulačné programy boli zhotovené v programe Tecnomatix Process Simulate Version 13.0.

- CD koreňový adresár priloženého CD
 - PLC.....podklady pre program PLC
 - VIPA OPC Server - ini folder...zložka pre inicializáciu OPC komunikácie
 - ini.zip
 - xholic01_PLC.zip.....PLC program v Step7
 - Tecnomatix.....simulačné programy
 - čisté projekty.....pracoviská bez definovaných operácií a programov
 - bodové_zvaranie_bez programu.psz
 - bodové_zvaranie_bez programu-ModifiedComponents.zip
 - kontinuálne_zvaranie_bez programu.psz
 - kontinuálne_zvaranie_bez programu-ModifiedComponents.zip
 - xholic01 bodove_zvaranie_static_sim.psz.....statická simulácia
 - xholic01 bodove_zvaranie_static_sim-ModifiedComponents.zip
 - xholic01 kontinualne_zvaranie_plc_sim.psz.....virtuálne riadenie
 - xholic01 kontinualne_zvaranie_plc_sim-ModifiedComponents.zip
 - xholic01 kontinualne_zvaranie_static_sim.psz.....statická simulácia
 - xholic01 kontinualne_zvaranie_static_sim-ModifiedComponents.zip
 - Video zostrihané video z oboch simulácií
 - xholic01_video.mp4
 - Vizualizácia.....vizualizačný program vo WinCC
 - xholic01_WinCC.zip
 - Diplomová práca - Holíček.pdf hlavný súbor diplomovej práce