



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

MODEL TŘÍDICÍ LINKY V SYSTÉMU TECNOMATIX PROCESS SIMULATE

SORTING LINE MODEL IMPLEMENTED BY TECNOMATIX PROCESS SIMULATE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Michal Prokop

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. Michal Prokop

ID: 154837

Ročník: 2

Akademický rok: 2016/17

NÁZEV TÉMATU:

Model třídící linky v systému Tecnomatix Process Simulate

POKyny PRO VYPRACOVÁNÍ:

- 1) Seznamte se s prostředím Tecnomatix Process Simulate a nastudujte principy práce v něm.
- 2) Vytvořte jednoduchou demonstrační úlohu, na které otestujete propojení aplikace Process Simulate a PLC.
- 3) Demonstrační úlohu řádně zadokumentujte tak, aby mohla být využita ve výuce.
- 4) Na základě získaných poznatků sestavte a oživte model výrobní linky.
- 5) Vytvořte aplikaci pro PLC, která bude model řídit.
- 6) Proveďte ověření správné funkce modelu.
- 7) Zadokumentujte tuto úlohu tak, aby mohla být využita ve výuce.

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

V této diplomové práci se zabývám vytvořením virtuálního modelu třídící linky v prostředí Tecnomatix Process Simulate. V diplomové práci zejména popisuji mimo robotické operace jako jsou: řízení dopravního pásu, generování toku materiálu, práci se sensory a simulování výrobního procesu. V druhé části diplomové práce se zabývám jejím virtuálním zprovozněním, kdy jsem vytvořil program pro PLC a pak jsem fyzické PLC spojil přes OPC Server s modelem třídící linky. Pro uživatelské ovládání jsem ještě vytvořil SCADA aplikaci. Tento model bude použit ve výuce a studenti budou vytvářet řídicí program pro PLC. V diplomové práci také popisuji jak pracovat v simulačním prostředí Tecnomatix Process Simulate.

Klíčová slova

Tecnomatix Process Simulate, digitální továrna, stacionární robot, dopravník, materiál, snímače, simulace, paletizace, virtuální zprovoznění, PLC, SCADA, OPC server

Abstract

This master's thesis is about how to create a virtual model of a sorting line by using Tecnomatix Process Simulate program. In this master's thesis is describe non-robotics operation as a controlling conveyor, material flow, working with sensors and simulating of manufactory process. In the second part of my master's thesis I realise virtual commissioning for this model of sorting line. I made program for real PLC and connected with model of sorting line via OPC server. I made SCADA application for users too. This model will be used in lessons and students will program PLC for controlling. In master's thesis I explain, how to work in simulation environment Tecnomatix Process Simulate too.

Keywords

Tecnomatix Process Simulate, digital factory, stationary robot, conveyor, material, sensors, simulation, palettizing, virtual commissioning, PLC, SCADA, OPC server

Bibliografická citace:

PROKOP, M. *Model třídící linky v systému Tecnomatix Process Simulate*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2017. 75 s. Vedoucí diplomové práce Ing. Václav Kaczmarczyk, Ph.D..

Prohlášení

„Prohlašuji, že svou diplomovou práci na téma Model třídící linky v systému Tecnomatix Process Simulate jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujícího 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.

V Brně dne: **15. května 2017**

.....

podpis autora

Poděkování

Děkuji panu vedoucímu diplomové práce Ing. Václavu Kaczmarczykovi, Ph.D. za jeho účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce a dále také za jeho vstřícnost a ochotu být kdykoliv nápomocen. Dále bych chtěl poděkovat panu Ing. Davidu Sámkovi, Ph.D., který mi poskytl technickou podporu z firmy AXIOM TECH s.r.o..

V Brně dne: **15. května 2017**

.....

podpis autora

Obsah

1	Úvod.....	9
1.1	Cíl práce.....	10
1.2	Struktura dokumentu	10
2	Digitální továrna	11
2.1	Přínos digitální továrny	11
2.2	Program Tecnomatix Process Simulate.....	13
2.3	Virtuální zprovoznění.....	14
3	Základy práce v Tecnomatix process simulate.....	16
3.1	Robot	16
3.1.1	Nastavení robota	16
3.1.2	Uchopovací zařízení	16
3.1.3	Nahrání programu do robota.....	17
3.1.4	Vytvoření signálů pro robota	18
3.2	Objekty	19
3.2.1	Vytvoření dílu	19
3.2.2	Manipulace.....	23
3.2.3	Generování toku materiálu.....	23
3.3	Dopravník	26
3.4	Snímače	30
3.4.1	Fotoelektrický snímač.....	30
3.4.2	Snímač vlastností	31
3.4.3	Snímač přiblížení	32
3.4.4	Snímač detekující dosažení polohy.....	33
3.4.5	Snímač monitorující polohu kloubu	34
3.5	Řízení simulace	34
3.5.1	Signály	34
3.5.2	Moduly.....	35
3.6	Operace.....	35
3.6.1	Řízení operací	36
3.6.2	Vytvoření operací pro zařízení	37
4	Model třídící linky	39
4.1	Rozvrhnutí linky	39
4.2	Algoritmus řešení	40
4.3	Realizace v prostředí Tecnomatix Process Simulate.....	41
4.3.1	Objekty v modelu.....	41
4.3.2	Posloupnost operací	43
4.3.3	Tok materiálu.....	45

4.3.4	Logika programu.....	46
4.3.5	Rozmístění snímačů.....	50
5	Virtuální zprovoznění	52
5.1	Připojení k PLC	52
5.1.1	Nastavení OPC serveru	53
5.1.2	Nastavení fyzického PLC v Tecnomatix Process Simulate.....	56
5.2	Řízení třídící linky z PLC.....	58
5.2.1	Úpravy v Tecnomatix Process Simulate.....	59
5.2.2	Struktura programu	59
5.2.3	Proměnné	62
5.2.4	Logika programu.....	64
5.3	Ovládací a vizualizační aplikace	66
6	Použití ve výuce.....	69
6.1	Namapování proměnných.....	69
6.2	Rozbor programu	69
6.2.1	Nový díl na dopravník	69
6.2.2	Paletizace	69
6.2.3	Světelná signalizace a nouzové zastavení.....	70
7	Závěr	71

1 ÚVOD

V této diplomové práci se zabývám vytvořením virtuálního modelu třídící linky v prostředí Tecnomatix Process Simulate. Nejprve jsem si nastudoval principy a možnosti práce v tomto simulačním prostředí. Tyto informace získané z technické dokumentace od výrobce softwaru posléze popisují v jednotlivých kapitolách. Nejprve popisují, jak nakonfigurovat stacionárního robota, jak na něj umístit pracovní zařízení a jak vytvořit a nahrát program pro řízení robota. V další kapitole jsem se zabýval vytvářením jednoduchých objektů, manipulací s nimi a následně s generováním těchto objektů jako tok materiálu. V dalších kapitolách řeším, jak ovládat dopravníkový pás, či jak nastavovat snímače a pracovat s nimi. Na závěr popisují, jak v prostředí Tecnomatix Process Simulate řídit celý proces vytvořené simulace, a to pomocí operací a signálů, jež se vyhodnocují v jednotlivých modulech. Takovému typu simulace se říká simulace řízena událostmi.

Na základě těchto získaných poznatků jsem zhotovil třídící linku s jedním stacionárním robotem, který třídí dva různé typy dílů jedoucí z dopravníkového pásu a pokládá je na palety.

Důvodem, proč jsem zvolil třídící linku k namodelování v programu Tecnomatix Process Simulate, bylo zejména to, že se na ni názorně demonstruje simulace nerobotických operací. Jelikož model bude sloužit jako laboratorní úloha, kde studenti jej budou řídit pomocí PLC, bylo potřeba vymyslet spíše jednodušší proces, aby se úloha dala stihnout naprogramovat v jedné vyučovací hodině.

V druhé půli této diplomové práce jsem pro tento model třídící linky vytvořil řídicí program pro PLC SIMATIC 300. Komunikace mezi PLC a virtuální třídící linkou je realizována přes OPC server. Takovému zprovoznění se říká virtuální zprovoznění, v odborné literatuře taktéž nazýváno z původního anglického názvu Virtual Commissioning. Takováto realizace je moderní a efektivní metodou pro návrhy výrobních linek v průmyslové automatizaci. Jelikož program, kterým se řídí virtuální model, lze bez nějakého většího doladování implementovat do reálné výrobní linky, a tak je již většina věcí řešena a odladěna před fyzickou realizací výrobní linky. Díky tomu se pak ušetří značný čas a peníze při její realizaci.

Na závěr jsem ještě pro tento model třídící linky vytvořil vizualizační a ovládací aplikaci v prostředí WinCC Flexible 2008. V aplikaci lze spustit, pozastavit nebo nouzově zastavit třídící linku. Aplikace dále pak dává uživateli informace o stavech snímačů, zda je robot nebo dopravník právě v provozu, či zobrazuje kolik dílů je již položeno na paletě.

1.1 Cíl práce

Jako hlavní cíle v této diplomové práci jsem si vytyčil zdokumentování práce ve vývojovém prostředí Tecnomatix Process Simulate, dále sestrojít v prostředí Tecnomatix Process Simulate virtuální model třídící linky. Tento model bude používán v laboratorní úloze, ve které budou studenti vytvářet PLC program, kterým se bude daný model řídit.

Druhým cílem proto je vytvoření PLC programu a otestování komunikace mezi PLC a prostředím Tecnomatix Process Simulate. PLC a program Tecnomatix Process Simulate mezi sebou bude komunikovat přes OPC server. Závěrem otestuji funkcionality celé třídící linky a to i povely pro zastavení a nouzové zastavení celé třídící linky. Proto zrealizuji ovládací aplikaci.

Tuto diplomovou práci lze použít i jako návod pro základní práci v programu Tecnomatix Process Simulate.

1.2 Struktura dokumentu

Nejprve v kapitole číslo 2 čtenáře seznamuji s problematikou digitální továrny, kde objasňuji její přínosy pro průmyslovou automatizaci, a jaké funkce nabízí. V kapitole číslo 3 popisuji základní postupy pro práci v simulačním prostředí Tecnomatix Process Simulate. V ní se čtenář dozví, jak v programu nastavit a programovat robota, jak vkládat a upravovat objekty, tvořit tok materiálu, jak simulovat dopravníkový pás, jaké snímače lze použít. Na závěr této kapitoly popisuji, jak vytvořený celkový model procesu simulovat. Ve 4. kapitole popisuji, jak jsem postupoval při návrhu třídící linky, která je stěžejním úkolem této diplomové práce. Poté jsem tento model virtuálně zprovoznil a řídit pomocí PLC, touto problematikou se zabývá 5. kapitola. Jelikož se tento model bude používat v laboratorní výuce, v předposlední kapitole číslo 6 studentům objasňuji význam použitých signálů z programu Tecnomatix Process Simulate, které budou muset při řešení této úlohy používat, a dále nastiňuji, jak tato třídící linka má fungovat. V 7. kapitole shrnuji výsledky své diplomové práce.

2 DIGITÁLNÍ TOVÁRNA

Digitální továrna je jednou z částí rozvoje průmyslové automatizace označovanou jako Průmyslová revoluce 4.0. Pod pojmem digitální továrna si lze představit rozsáhlou síť digitálních metod, trojrozměrných 3D modelů a nástrojů, které jsou včleňovány v rámci průběžného řízení dat. Základním cílem digitální továrny je komplexní a systémové plánování, projektování, ověřování a průběžné zlepšování všech důležitých struktur, procesů a zdrojů, které jsou obsaženy v reálné továrně, a které souvisí s výrobou daných výrobků. Toho je dosahováno pomocí přesných digitálních modelů a simulací a 3D vizualizací. Tímto se dále propojují všechny skupiny lidí, kteří se podílejí na budoucím výrobním procesu. Díky tomu se pak jednoduše předejde chybám, které by se jinak objevily až při spuštění výroby [1].

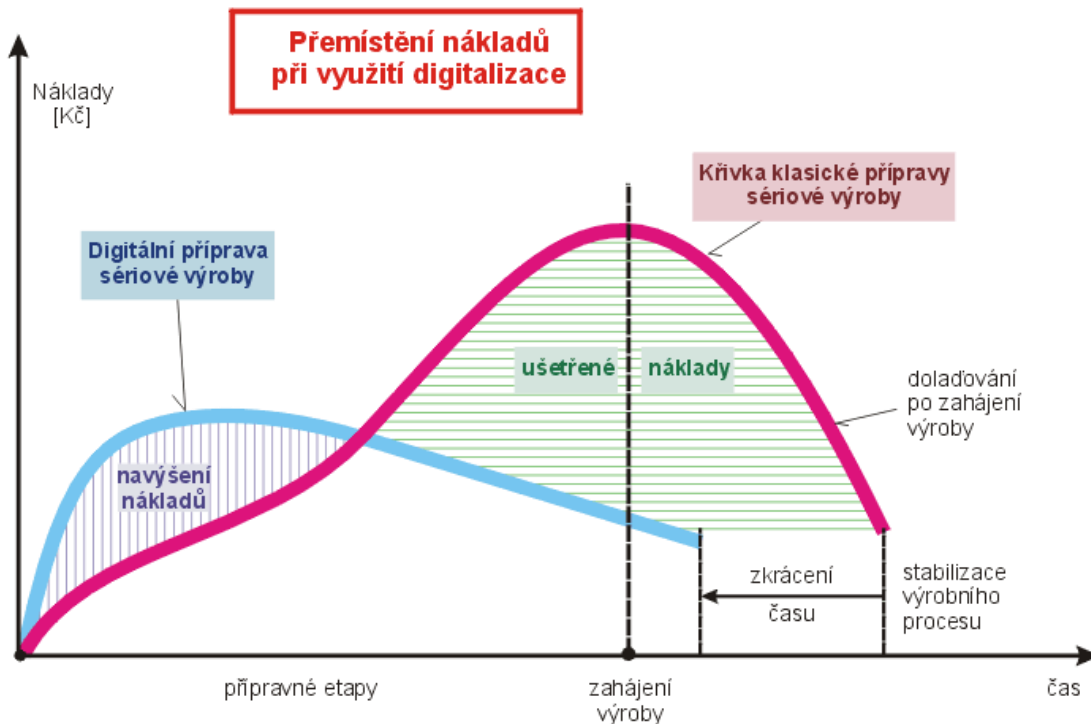
Digitální továrna tak propojuje týmy konstruktérů, mechatroniků, automatizačních a procesních inženýrů. Umožňuje tak lepší spolupráci a komunikaci a tím pádem zkrácení času plánování a tím i snížení nákladů.

2.1 Přínos digitální továrny

Digitální továrna je schopna řešit problémy vyskytující se v současných moderních výrobních linkách a strojích. Dnešními požadavky na výrobu jsou zejména snižování nákladů na výrobu a vývoj, dále pak je snaha o co nejkratší dobu dodání, čímž je možné snížit i ceny produktu. Protichůdným požadavkem naproti tomu je ovšem zvyšování složitosti výrobku, jeho flexibilita a variabilita [2].

Jelikož projekt digitální továrna propojuje všechny skupiny lidí, které se účastní na vývoji a produkci výrobku, lze tyto protichůdné požadavky řešit již při návrhu projektu. Ve virtuálním návrhu se původní návrh modelu výrobní linky, který je navržen v CAD systému, promění ve virtuální 3D model, který obsahuje kinematiku, funkční senzory a logické řídicí prvky. Díky tomu je pak možné provádět přesné simulace chování vyvíjené výrobní linky, čímž je možné zhodnotit daný návrh, zhodnotit ergonomické řešení, jelikož i pracovník je součástí digitální továrny, minimalizovat výrobní prostory, zkontrolovat možné kolize a časové ověření všech operací. Na modelu je možné i testovat různé krizové scénáře typu co se stane když [2].

Z případových studií jasně vyplývá, že jakákoliv změna plánu během průběhu procesu testování a ověřování výroby, či procesu je desetinásobně dražší, než kdyby se tato změna provedla již během návrhu a vývoje konceptu. Pokud by k této změně ovšem došlo až při sériové výrobě, byly by náklady stonásobně větší. Tento jev se označuje jako pravidlo 1:10:100, což je pro praxi nepřijatelné, a proto je nejdříve nutné ladit a odstraňovat chyby již při jeho návrhu a před spuštěním samotného testování. Toto právě umožňuje virtuální model výrobní linky [3].



Obrázek 1 Zobrazení ušetření peněz a času [4]

Navýšení vstupních nákladů ve fázi přípravy se bohatě následně vrátí v rychlejším vstupu výrobku na trh, viz Obrázek 1. A to díky tomu, že je jeho růstová fáze bezproblémovější, a tudíž i mnohem rychlejší, neboť chyby, ke kterým v této fázi často dochází, byly již eliminovány v době přípravy výroby, kde byla provedena již řada zkoušek a simulací, v níž se nakonec většina chyb opravila pomocí digitální továrny. Tabulka 1 zobrazuje procentuální přínos, který byl zaznamenán v automobilovém a leteckém průmyslu po použití digitální továrny při vývoji nových zařízení [4].

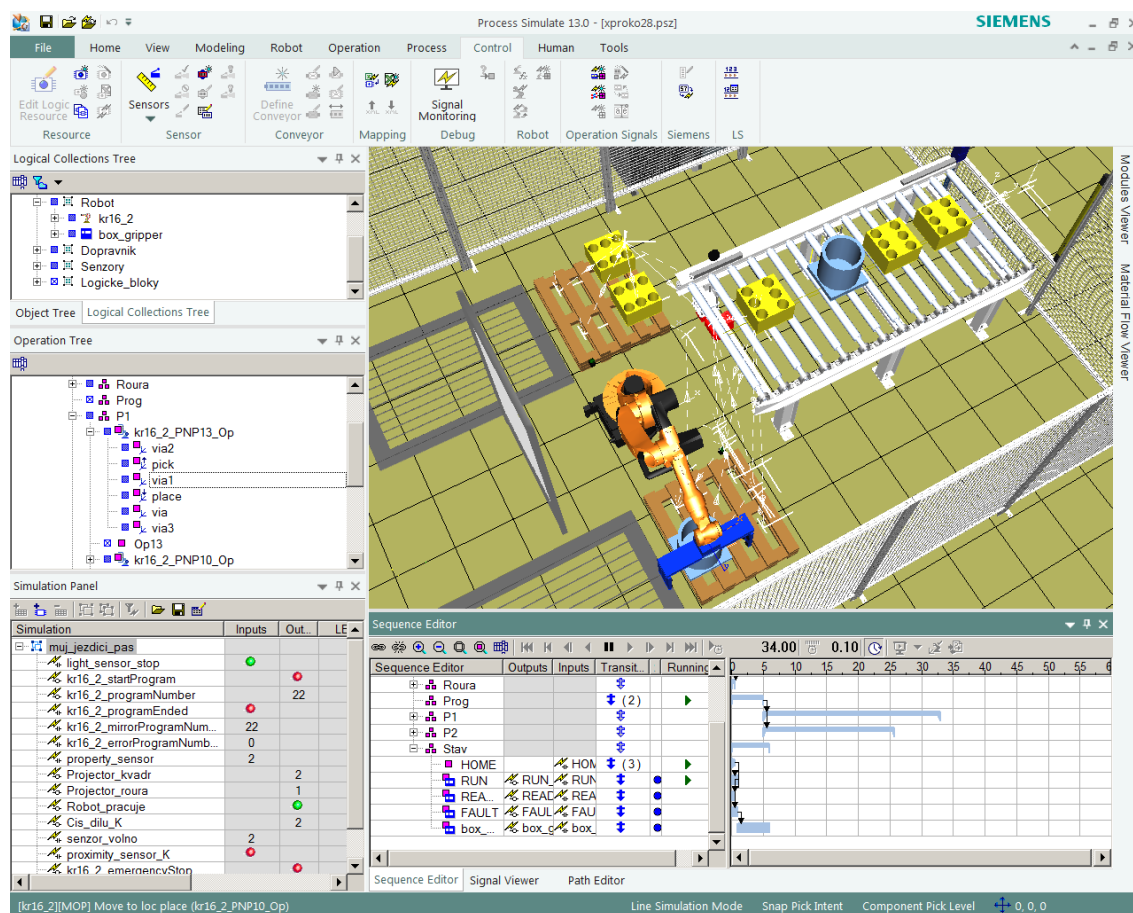
Oblast přínosu	Zlepšení
Rychlejší náběh výroby	až o 15%
Celková vyšší produktivita	až o 10%
Zvýšení produktivity stávajících výrobních zařízení	15 – 20%
Snížení potu výrobních zařízení, nástrojů, periférií a pomocného materiálu	až o 40%
Snížení investičních nákladů na nová výrobní zařízení	až o 20%
Zlepšení výrobní kvality	až o 15%
Zlepšení zralosti produktů	5 – 10%
Zkrácení projektových časů	až o 20%
Snížení počtu změnových řízení	až o 20%
Zvýšení účinnosti v oblasti komunikace a spolupráce	až o 35%

Tabulka 1 Přínos digitální továrny v automobilovém průmyslu [4]

V automobilovém průmyslu se již dnes považují metody digitální továrny za naprostou realitu. Díky tomu mohou nejvýznamnější automobilky světa odolávat konkurenčním tlakům na rychlost uvedení nového modelu i na cenu. Ze zkušenosti je jasné, že ostatní oblasti průmyslu nebudou moci zůstat stranou a bude i zde vyvíjeno úsilí na využití těchto metod [4].

2.2 Program Tecnomatix Process Simulate

Program Tecnomatix Process Simulate je součástí komplexního portfolia pro řešení problematiky digitální továrny od společnosti SIEMENS. Dalšími programy tohoto portfolia jsou Process Designer, či Plant Simulation. Tyto programy slouží pro plánování a simulaci výrobních procesů, čím se snižuje časová náročnost při uvádění do provozu a dále umožňuje detekci chyb již při vývoji, čímž se snižují finanční náklady na produkt [5].



Obrázek 2 Vývojové prostředí Tecnomatix Process Simulate

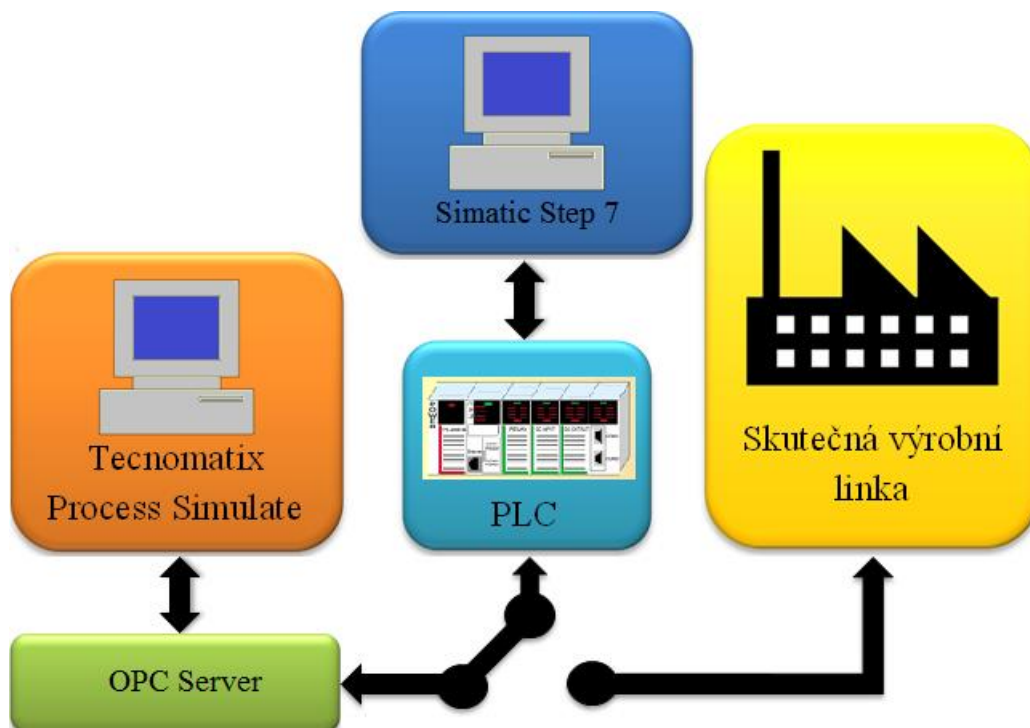
Tecnomatix Process Simulate umožňuje detailní simulaci výrobních a procesních operací. Program se převážně zaměřuje na nastavení kinematiky robotů a vytváření robotických operací jako jsou: svařování, lakování a manipulace s předměty. Dále umožňuje detekci kolízi během výrobního procesu, či nastavování kinematiky

dalších komponentů ve výrobní lince. V programu lze ověřovat montážní postupy a díky tomu se ještě před začátkem výroby mohou optimalizovat montážní procesy, činnost pracovníků a i fungování výrobních zařízení. Další oblastí simulace, kterou lze v programu simulovat, je simulace logistiky a pohybu materiálu. Program umožňuje analyzovat a vizualizovat výrobní kapacitu a výkonnost pomocí simulace diskrétních událostí. Tímto způsobem je možné rychle identifikovat kritická místa výroby, zkoumat pohyb materiálu a zobrazit využití prostředků v čase i u několika variant procesu najednou. Poslední oblastí simulace, kterou program Tecnomatix Process Simulate umožňuje, je simulace lidí a ergonomie. Díky tomu lze pak vytvářet bezpečnější a ergonomičtější pracovní postupy skutečných pracovníků [6].

Simulace výrobního procesu může být definována jako posloupnost jednotlivých operací v čase takzvaným Ganttovým digramem. Druhou možností je simulace výrobního procesu jako sled operací, které jsou vyvolány na základě nějaké události na vstupech řídicí jednotky. Tuto možnost simulace umožňuje v programu Tecnomatix Process Simulate Line Simulation mode.

2.3 Virtuální zprovoznění

Vytvořený model výrobní linky je pak možné řídit přímo v prostředí Tecnomatix Process Simulate, tato možnost se nazývá Cyclic Event Evaluation neboli cyklické vyhodnocení událostí. Další možností je přímé spojení programu Tecnomatix Process Simulate a některého z emulátorů PLC, například PLCSIM. Třetí možností je spojení tohoto digitálního modelu a reálného fyzického PLC. K tomuto propojení je potřeba OPC server, který je implementován mezi PLC a digitální model, který je vytvořen v programu Tecnomatix Process Simulate. Takovému propojení se říká Virtual Commissioning [2]. Jelikož virtuální model funguje stejně jako skutečná výrobní linka, lze PLC s již vytvořeným řídicím programem pro model bez problému připojit ke skutečné výrobní lince, jak je naznačeno na Obrázek 3.



Obrázek 3 Schéma virtuálního zprovoznění

Hlavní přínosy virtuálního zprovoznění jsou zejména:

- Verifikace a odladění programů v řídicích jednotkách jako je například PLC.
- Významná úspora času na inženýrských pracích.
- Ověření správného návrhu senzorů a jejich správného vyhodnocení v PLC.
- Kontrolování bezpečnostních funkcí a stavů a jejich blokad.
- Nasimulování různých variant konstrukčních řešení nebo řídicích programů.
- Odhalení procesních chyb.
- Optimalizace výrobního procesu.
- Významné zkrácení doby ladění a ožívování [2].


3 ZÁKLADY PRÁCE V TECNOMATIX PROCESS SIMULATE

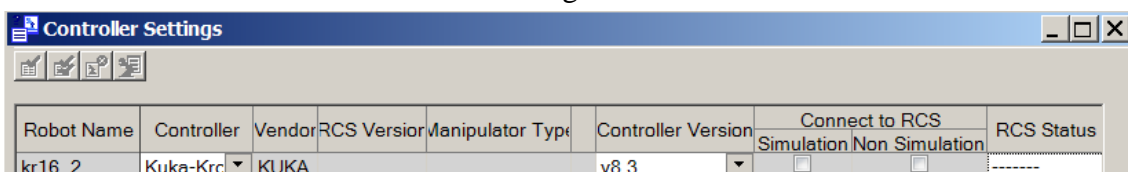
V této kapitole popisují základní postupy pro práci v daném simulačním prostředí. Jednotlivé kapitoly se zabývají postupy, jak nastavit a naprogramovat robota, jak namodelovat objekt, jak definovat tok materiálu, jak nastavit dopravníkový pás, jaké snímače lze použít a jak je nastavit. Na závěr popisují, jak celou simulaci řídit.

3.1 Robot

V simulačním prostředí lze simulovat různé robotické operace jako je svařování, lakování, nebo uchopování a umístění předmětů robotem. V programu Tecnomatix Process Simulate lze rovněž provádět offline programování robotů.

3.1.1 Nastavení robota

Po vložení robota do studie je třeba nastavit řízení robota, pravým tlačítkem myši se klikne na robota a zvolí se Controller Settings .




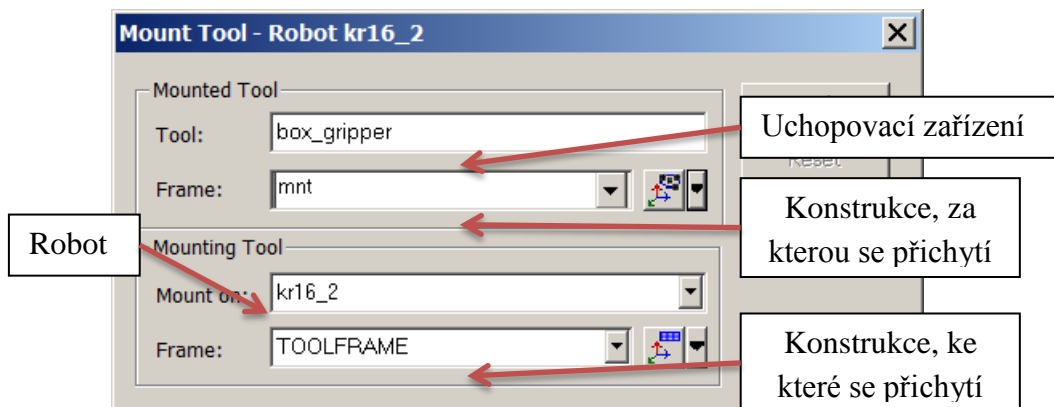
Obrázek 4 Nastavení řízení robota

V položce Controller se zvolí ovladač dle výrobce robota, pokud k vloženému robotu neodpovídá žádný výrobce z nabídky, lze zvolit implicitní nastavení. Volba verze se doporučuje co nejnovější.

3.1.2 Uchopovací zařízení

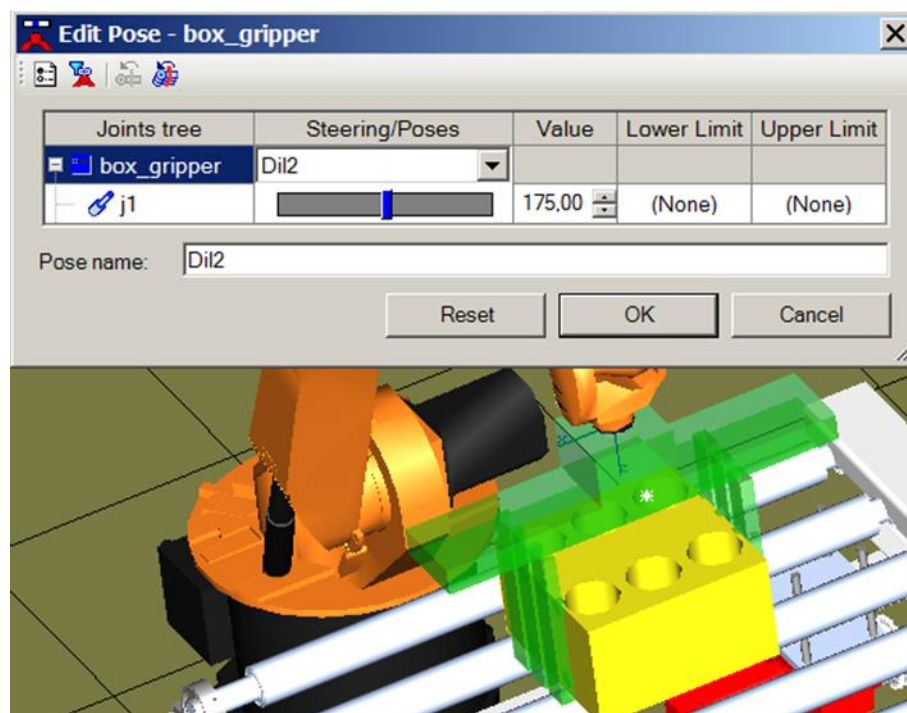
Jeden z nástrojů, kterým se dá robot osadit je uchopovací zařízení. Již vytvořené díly lze najít v adresáři Library/Resources/Robot_tools/Grippers. Jak vložit předmět je zmíněno v úvodu kapitoly 3.2 - Objekty.

Přípevnění k robotu se provádí tak, že se označí robot, k němuž má být uchopovací zařízení připevněno, a v kartě Robot v záložce Tool and Device se zvolí ikona Mount Tool .



Obrázek 5 Upevnění uchopovacího zařízení na robota

Jelikož posun čelistí při uchopení nemusí odpovídat rozměrům předmětu, který mají sevřít, lze nastavit vlastní posun čelistí. Pravým tlačítkem myši se klikne na uchopovací zařízení a vybere se Pose Editor . V něm se pak zvolí nová pozice.





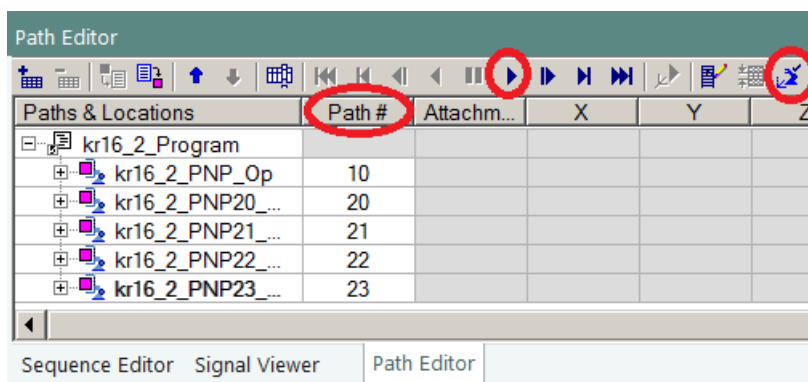
Obrázek 6 Nastavení posunu čelistí

Potom tuto nově vytvořenou pozici uchopovacího zařízení použijeme v operaci Pick and Place. Více o operacích je zmíněno v kapitole číslo 3.6 - Operace.



3.1.3 Nahrání programu do robota

Po vytvoření operací pro robota, například Pick and Place je třeba tuto operaci přidat do programu robota a nahrát jej. V záložce Robot ve skupině Program zvolit Robotic Program Inventory . V něm vytvořit nový program . V položce robot se vybere robot, do kterého má být program pak nahrán, tento program pak je nutné nastavit jako



defaultní program . Poté je třeba do takto vytvořeného prázdného programu přidat operace, které chceme, aby daný robot vykonával. K tomu slouží Program Editor . Po jeho stisknutí se přepneme do Path Editou.

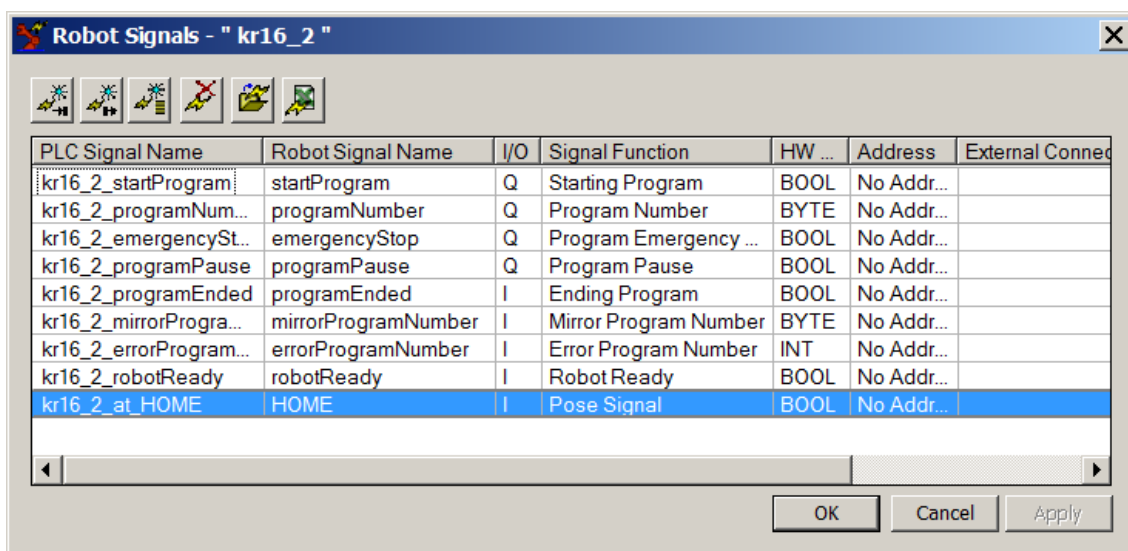


Obrázek 7 Zadání cest robotu

Zde se přidají všechny operace, které má daný robot vykonávat. V položce Path # se vepíše pro jednotlivé operace čísla cest, ty musí být vždy odlišné. Toto číslo se pak používá v signálu Program_number a slouží pro volání příslušné operace. Ještě než se program nahraje do robota, je nutné spustit simulaci zvolených operací. Nejdříve se zvolí Auto Teach  a pak spustí simulace .

3.1.4 Vytvoření signálů pro robota




Pro řízení robota z PLC jsou důležité jeho signály. Signály robota se vytváří tak, že se pravým tlačítkem myši klikne na robota a vybere se položka Robot Signals and Modules → Robot Signals , pak kliknout na ikonu Create Default Signals .




Obrázek 8 Signály robota

Tyto signály se pak využívají pro řízení simulace, viz kapitola č. 3.5 - Řízení simulace.

3.2 Objekty

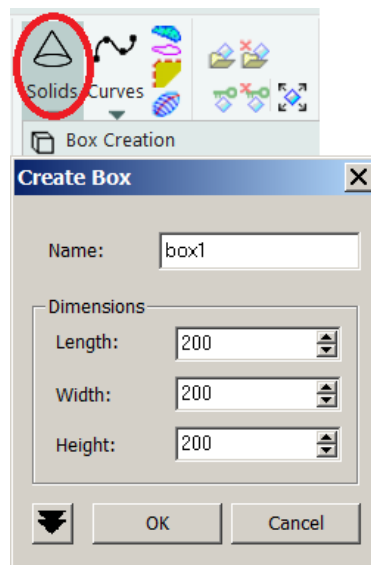
Tecnomatix Process Simulate rozlišuje dvě základní skupiny objektů: díly (parts ) a výrobní prostředky (Resources ) [8]. Již vytvořené objekty lze vkládat z knihovny. Záložka Modeling, Insert Components , pak vybrat ze správného adresáře v knihovně. Dále pak lze nové díly nebo prostředky vytvářet přímo v prostředí Tecnomatix Process Simulate.

3.2.1 Vytvoření dílu




V záložce Modeling stisknout ikonu . Zvolit ikonu Solid a pak si vybrat žádaný tvar pro díl.

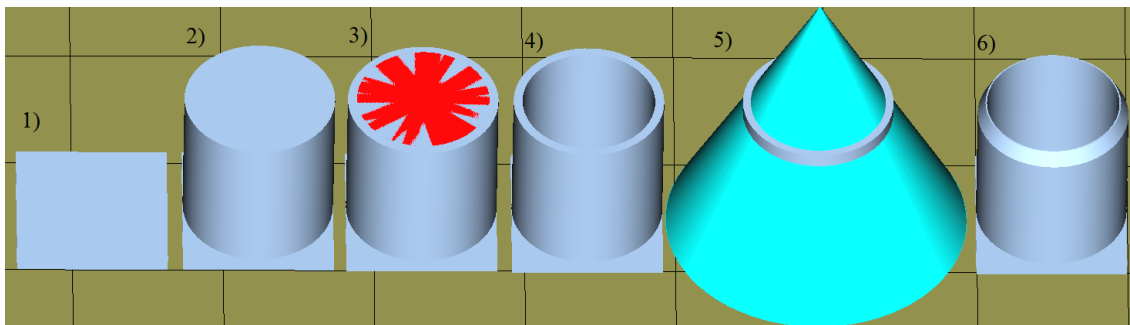
- Krychle (Box creation)
- Válec (Cylinder creation)
- Kužel (Cone creation)
- Koule (Sphere creation)
- Prstenec (Torus creation)

V tabulce pak lze zadat rozměry objektu.



Obrázek 9 Nastavení rozměrů objektu

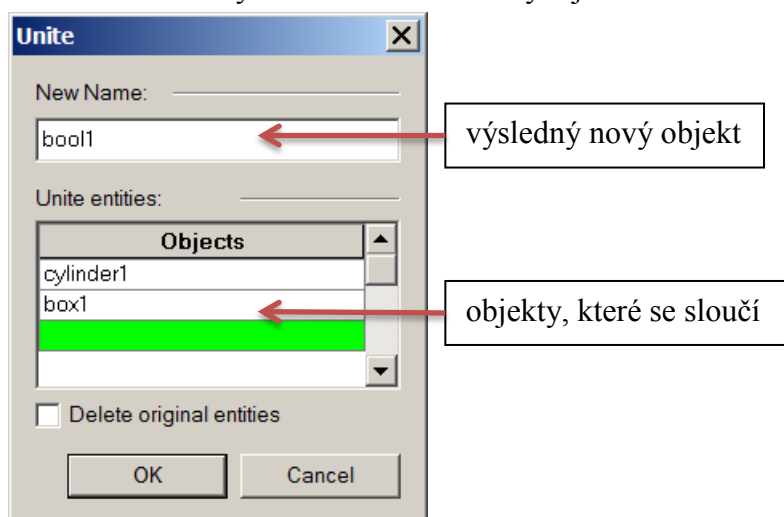
V položce Solid lze jednotlivé geometrické tělesa slučovat  nebo dělat jejich průniky , či rozdíly .



Obrázek 10 Postup modelování dílu

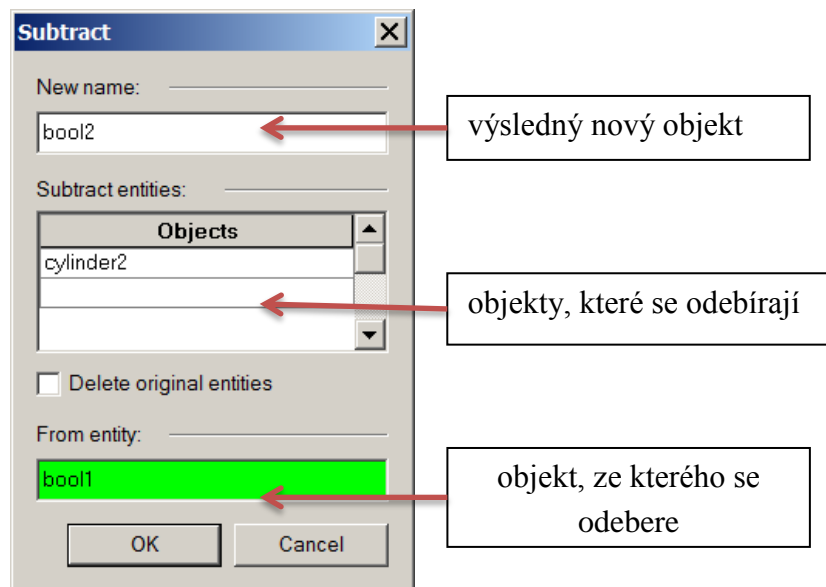
Obrázek 10 zobrazuje, jak lze takto vytvořit nový díl.

- 1) Vytvoření čtvercové základny
- 2) Přidání válcovité části. Tyto dvě části se sloučily v jednu část.



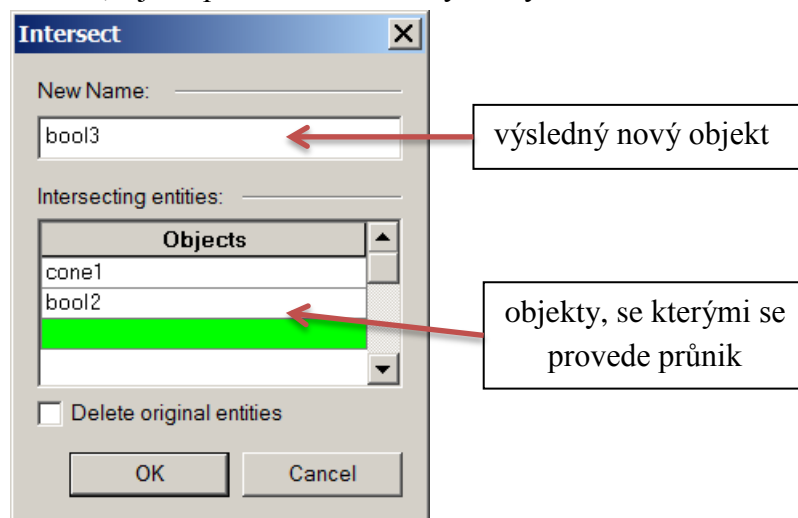
Obrázek 11 Provedení sloučení

- 3) Přidání druhého válce (červená barva), kterým se umazal vnitřek původního válce.






Obrázek 12 Provedení rozdílu

- 4) Výsledek rozdílu.
- 5) Přidání kuželu, s jeho průnikem se skosily hrany.




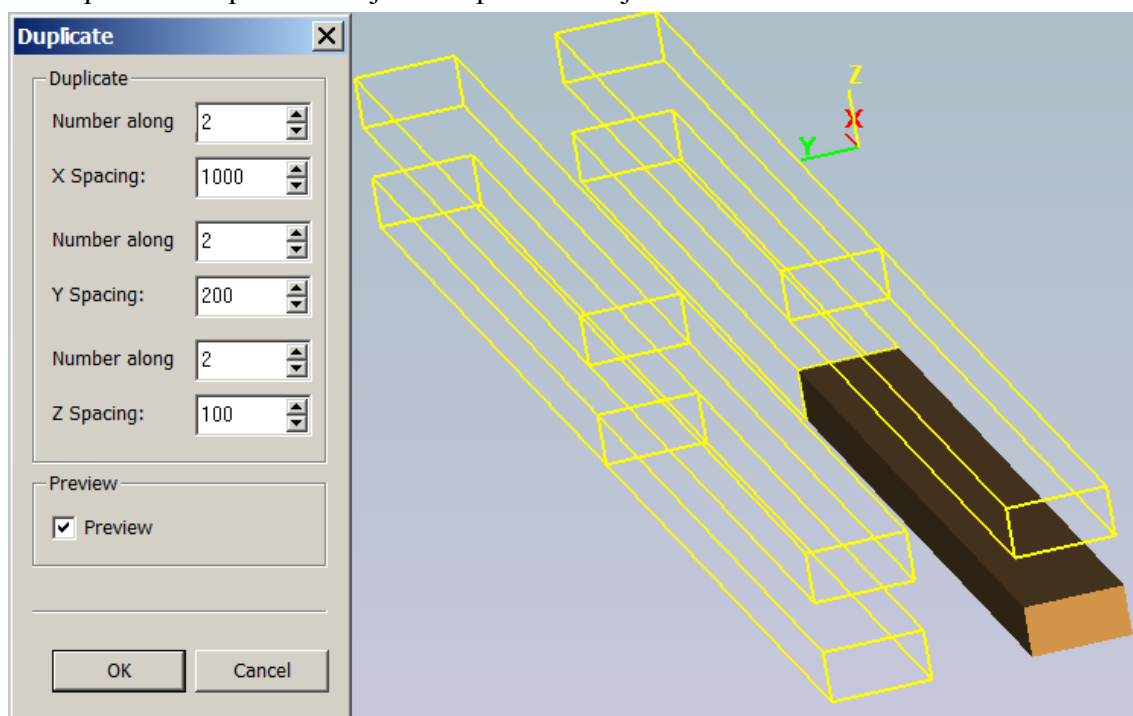
Obrázek 13 Provedení průniku

- 6) Výsledný díl.

Pro dokončení modelování objektu se stiskne ikona End modeling , ukončí se tak úpravy objektu a uloží se do knihovny. V Objecr Tree pak lze vidět, že se ikona  změnila na . Dodatečné úpravy objektu, jsou možné po stisknutí ikony Set Modeling



Další praktickou pomůckou je rozkopírování objektů .

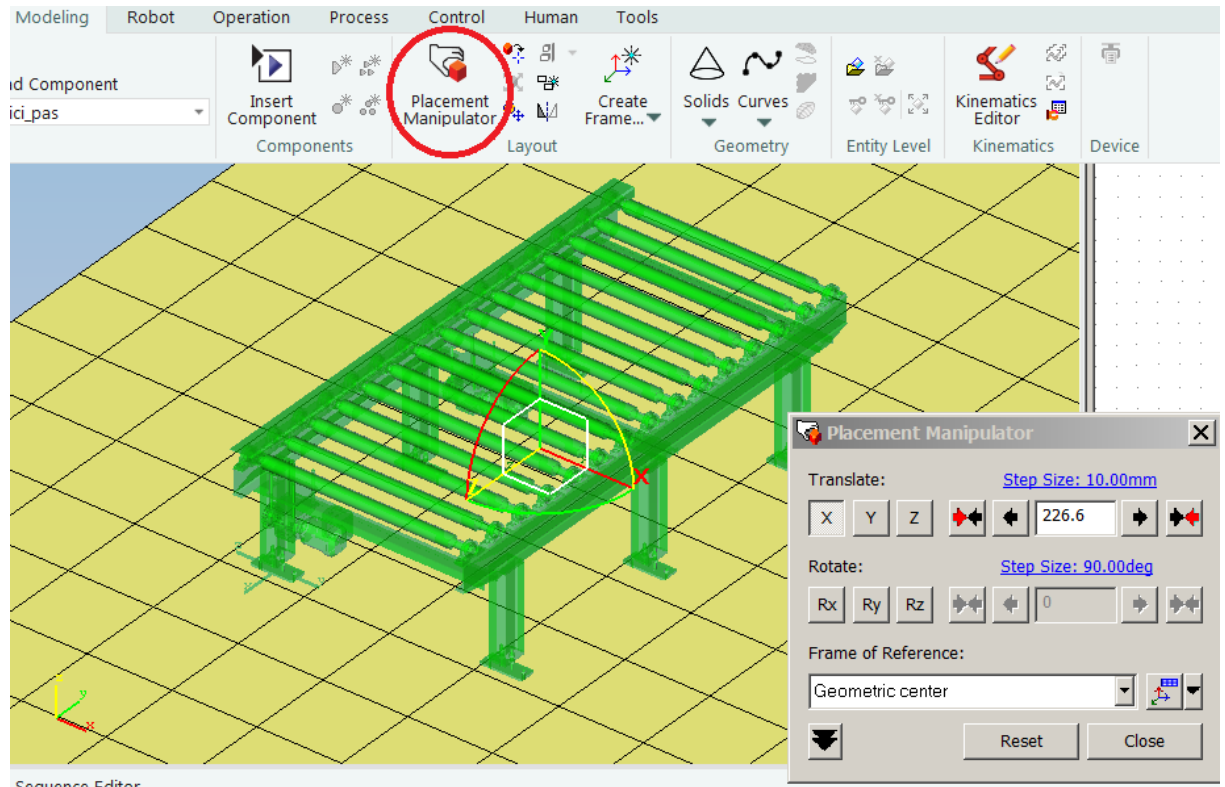


Obrázek 14 Rozkopírování objektu

Number along určuje, kolikrát bude v dané prostorové ose objekt obsažen. Spacing pak říká, jak vzdálené budou stejné hrany originálu a duplikátu od sebe.

3.2.2 Manipulace


Již vložený nebo vytvořený díl lze pak pomocí funkce Placement Manipulator vhodně umístit.



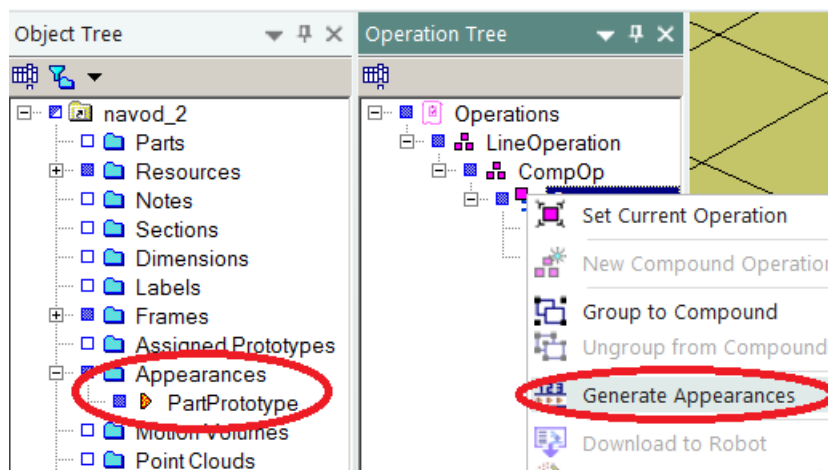
Obrázek 15 Nastavení pozice

Pozici objektu lze nastavit buď pomocí tabulky v Placement Manipulator, kde lze nastavit jednotlivé osy prostoru a rotace, viz Obrázek 15 vpravo dole. Druhou možností, jak nastavit pozici objektu, je tahem kurzoru po příslušné ose, či ploše svírající dané osy. Rotaci lze provést tahem po dané čtvrtkružnici.

3.2.3 Generování toku materiálu

Vícenásobné generování objektu je možné v simulaci řízené událostmi. Do ní se lze přepnout přes záložku Home a pak ikon Switch to Line Simulation Mode . V tomto režimu ovšem nelze pracovat s objekty definované jako Parts (díly), zde jsou nahrazeny pojmem Appearances (zjevem) [8].

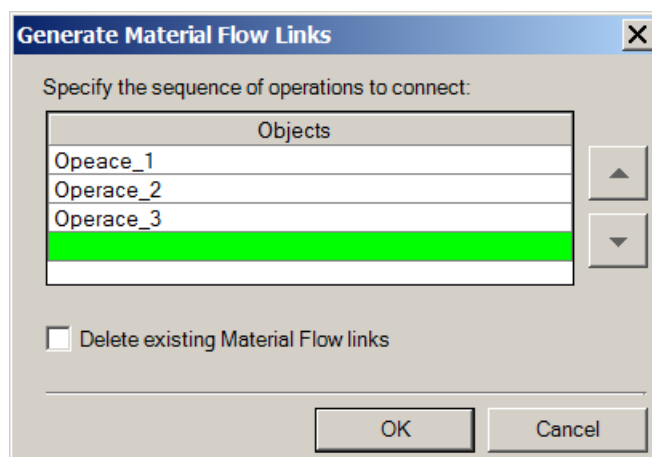
Aby bylo možné vytvořit z Parts Appearances, tak je nejprve nutné definovat pro díl nějakou operaci, například Object Flow Operation. Až pak v režimu simulace řízené událostmi lze vytvořit Appearance. Právým kliknutím myši na operaci a zvolit Generate Appearances, pak by se měl požadovaný objekt objevit v adresáři Appearances.



Obrázek 16 Vytvoření Appearance

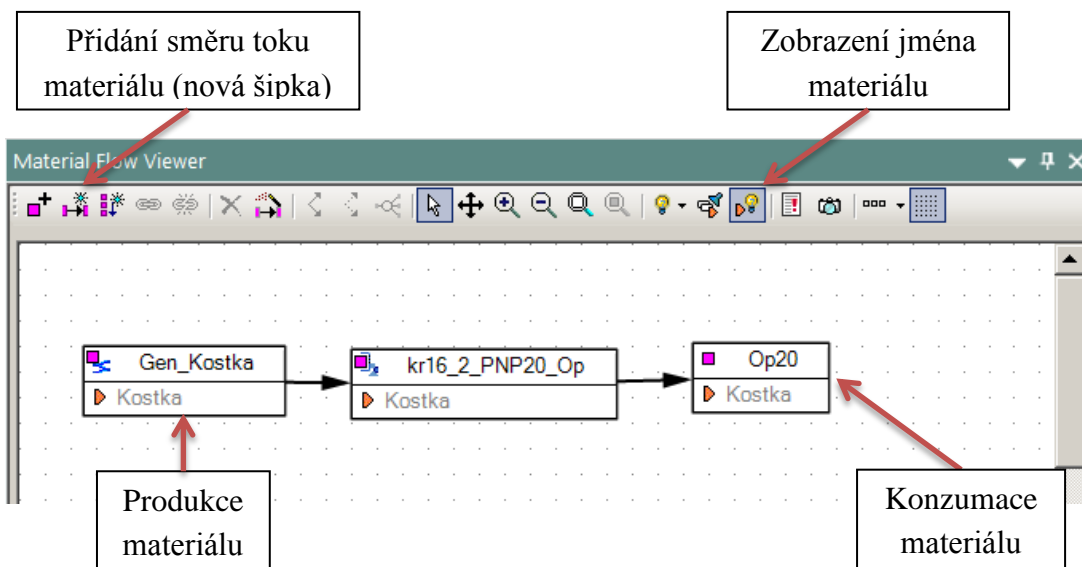
Produkcí a konzumací materiálu určuje tok materiálu (Material Flow). Ten se definuje v okně Material Flow Viewer. Toto okno se dá aktivovat přes záložku View→Viewers

Viewers → Material Flow viewer. Tok materiálu se vytváří pomocí Generate Material Flow Links [9].



Obrázek 17 Definování toku materiálu

Do tabulky Obrázek 17 se vkládají jednotlivé operace z Operation Tree, kterými materiál postupně prochází. Pro tuto názornou ukázkou bude tok materiálu vypadat následovně.




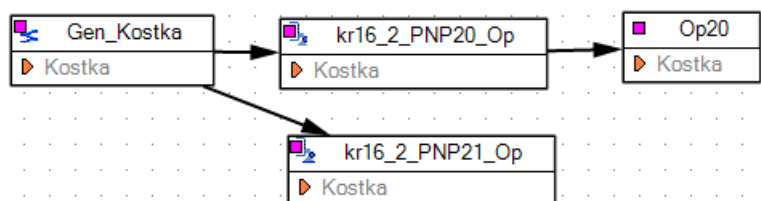
Obrázek 18 Tok materiálu

Aby bylo možné vygenerovat tok materiálu, musí být definovány alespoň dvě operace. V jedné se bude materiál produkovat a ve druhé konzumovat [8].



3.2.3.1 Větvení toku materiálu

V určitých případech je potřeba, aby se s materiálem provádělo více operací ve stejném čase, anebo materiál může procházet alternativními cestami. Například, má-li se nejprve vygenerovat materiál na dopravník a poté být položen na paletu, další díl už je ale potřeba pokládat na jinou pozici na paletě.

Nejprve se musí přidat nová operace do Material Flow Vieweru. Označí se operace ze seznamu operací (Operation Tree) a v Material Flow Vieweru se zvolí Add Operation . Následným přidáním šipky od operace generování k nové operaci pro manipulaci s předmětem se vytvořily dvě cesty pro materiál.



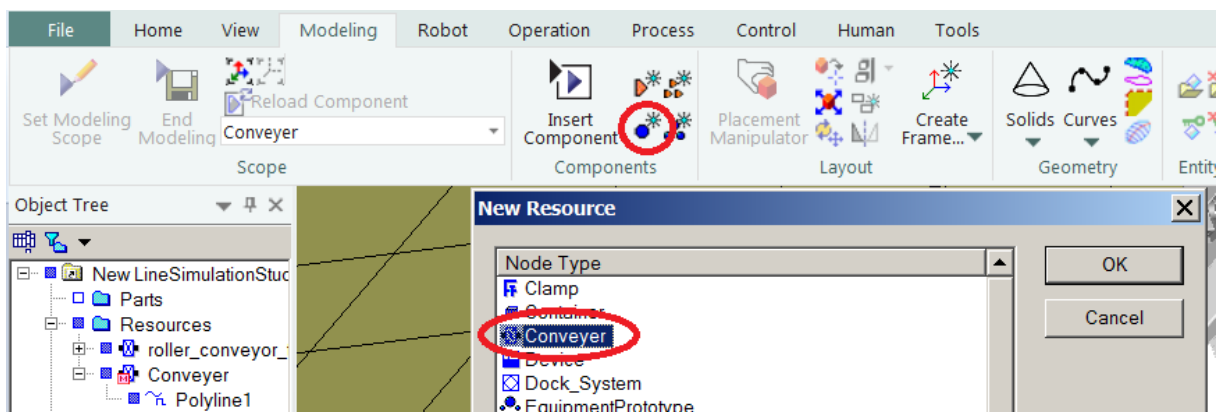
Obrázek 19 Nová cesta pro materiál

Zatím by se tyto cesty ovšem vykonávaly zároveň pro jeden a tentýž díl, což ale není žádoucí. Po označení obou šipek je pak možné tyto cesty z paralelních změnit na alternativní . Obě šipky by se pak měly zobrazovat čárkovaně. Ikonou  lze vytvořit alternativní skupinu paralelních operací.

3.3 Dopravník

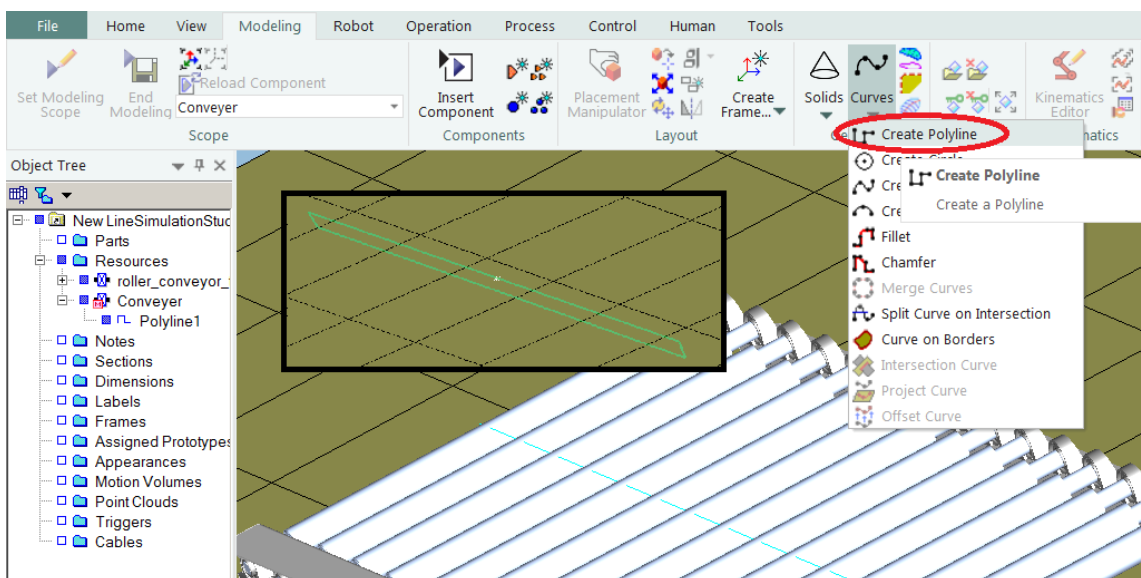
Celkový model dopravníku se skládá ze dvou částí, první část je objekt dopravníku, který slouží jen jako grafický element v simulaci, ten se vloží, jako komponent z knihovny, viz kapitola číslo 3.2 Objekty. Druhou podstatnější částí je křivka trajektorie, po níž se pak budou dopravovat předměty.

V záložce Modeling ve skupině Componentst, se zvolí Create New Resource a potom Conveyer. V okně Object Tree se pak objeví ve složce Resources položka Conveyer [9].




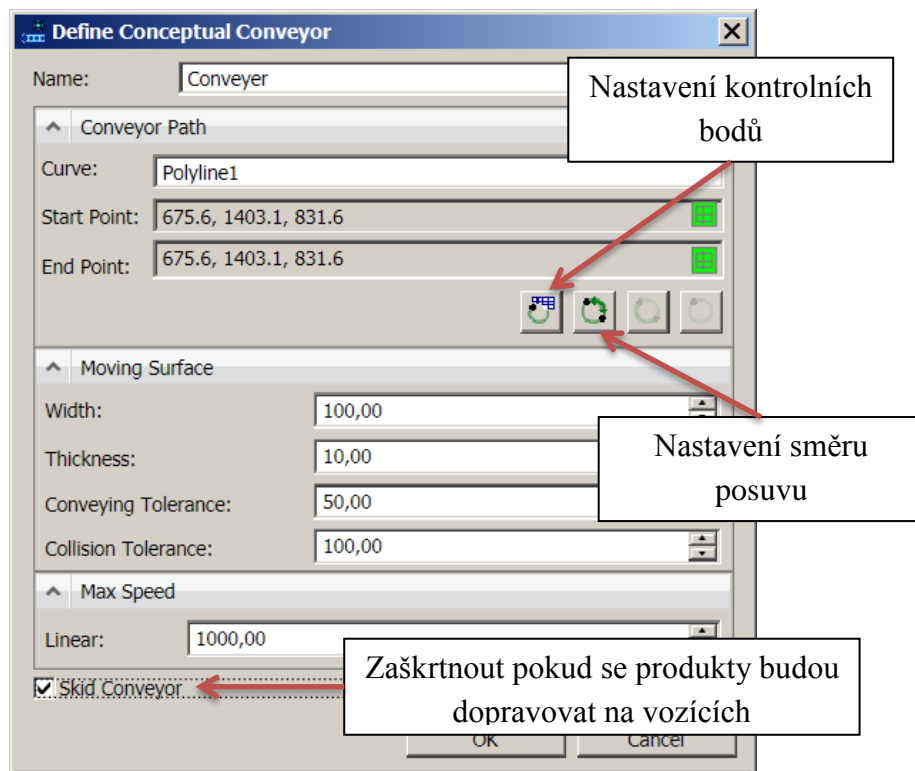
Obrázek 20 Vytvoření dopravníku

Pak je potřeba vykreslit křivku, po které se budou předměty na dopravníku pohybovat. V položce Geometry zvolit „Curves“ a vykreslit křivku kudy pak pojedou produkty. Nejvíce realisticky se jeví vykreslit smyčku kolem válečků dopravníku, pro jednoduchost vykreslení jsem zvolil hranatý tvar.




Obrázek 21 Vykreslení křivky kolem dopravníku

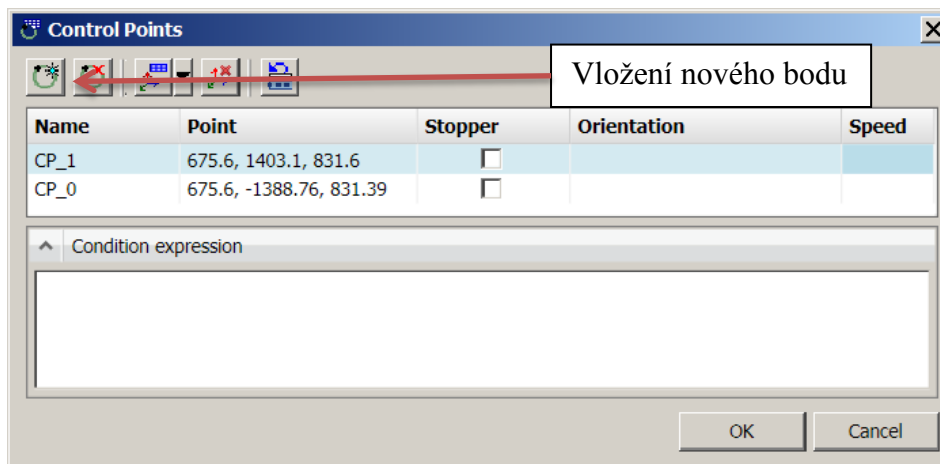
Následně je potřeba se přepnout do záložky Control a zvolit ikonu Define Conveyor . Tato ikona je aktivní při označení položky Conveyor v okně Object Tree. V tabulce se nastaví:




Obrázek 22 Nastavení parametrů dopravníku



- Curve – jméno křivky, na niž se budou pohybovat produkty (kliknout na název křivky v Object Tree)
- Moving Surface – nastavují se rozměry plochy kolem křivky, kde je ještě možné zachytávat předměty, které se budou dopravovat po dopravníku, hodnota Conveing Toleance musí být vždy menší nebo rovna hodnotě Collision Tolerance
- Max speed – maximální rychlost dopravníku


Dále v této tabulce po stisknutí tlačítka  lze nastavit kontrolní body. Jimi lze měnit rychlost nebo směr pohybu.

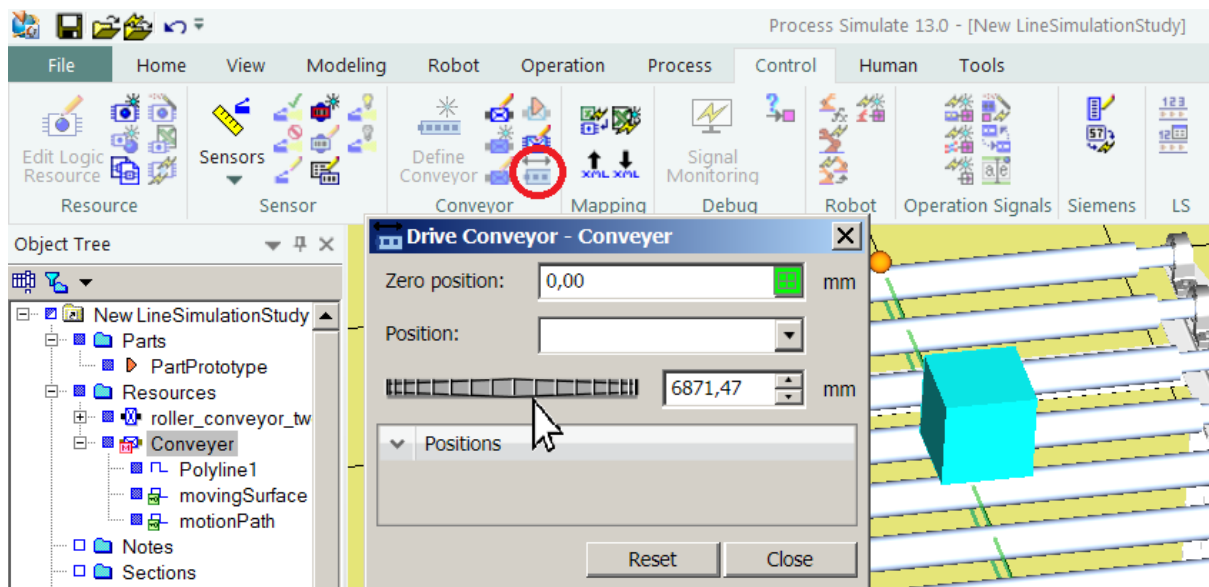


Obrázek 23 Kontrolní body


Po nastavení těchto parametrů by se v okně Object Tree měly objevit pod adresářem Conveyor položky movingSurface a motionPath. Dodatečné úpravy lze provádět po stisknutí ikony Edit Conceptual Conveyor  [9].

Nyní je třeba nastavit, který objekt bude po dopravníku jezdit. Kliknutím myši na díl se označí díl, který bude po dopravníku přepravován. Pokud simulace bude probíhat ve standardním módu, označuje se díl jako položka Part, pokud ale simulace probíhá v Line Simulation módu, musí se díl označit jako Appearance. Pokud se v grafickém náhledu žádné Appearance nenachází, je třeba pravým tlačítkem myši kliknout v Operation Tree na nějakou operaci, která je vázána pro daný díl a zvolit Generate Appearances . Nejlepší je zvolit přímo Compound Operation s názvem LineOperation, ta totiž zahrnuje všechny operace v simulaci, a tudíž se zobrazí Appearance pro všechny díly. Po označení se pak záložce Control zvolí ikona Define Conveyable Part . Zde jdou nastavit souřadnice objektu, který má být vezen, postačí ponechat 0,0,0 [9].


Kliknutím na vytvořenou křivku a pak na ikonu Drive Conveyor , lze zkontrolovat, jak se objekt pohybuje po dopravníku, samozřejmě je nutné, aby se objekt nacházel na ploše, kde je objekt možné táhnout.

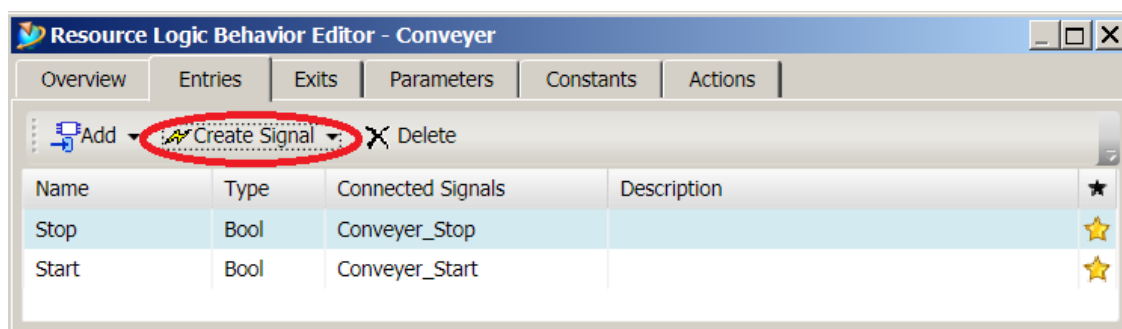


Obrázek 24 Pohyb krabice po dopravníku

V dalším bodě se nastaví logický blok a signály pro dopravník, a to kliknutím na ikonu . Dopravníku se mohou nastavit čtyři akce, a to jsou:

- Start – spuštění dopravníku
- Stop – zastavení dopravníku
- Change Speed – změna rychlosti
- Change Direction – změna směru pohybu


Následně vyskočí okno Resource Logic Behavioral Editor. Tento editor lze taktéž spustit ikonou Edit Logic Resource  [9].



Obrázek 25 Nastavení signálů




V položce Entries (vstupy) při kliknutí na Create Signal se zvolí output pro oba signály, to proto že signály jsou vztahovány k proměnným v PLC a vněm jsou to výstupní povely k zapnutí či vypnutí dopravníkového pásu.

3.4 Snímače

Vložit nový snímač do projektu lze přes kartu control v sekci sensor ikonou sensors .

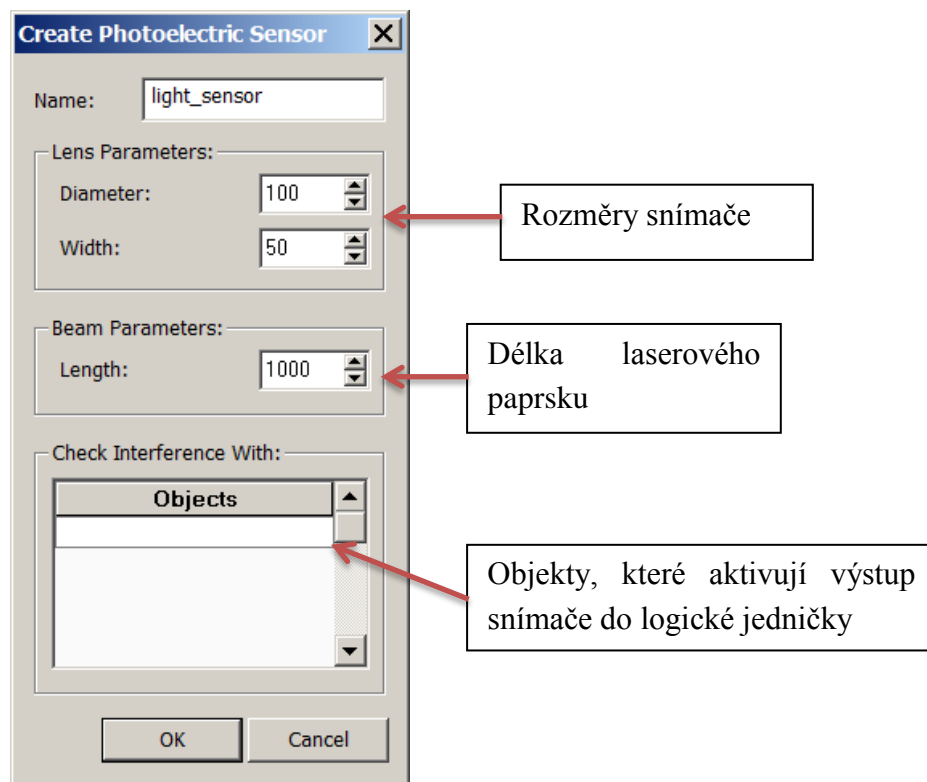
V programu Process Simulate lze vytvořit tyto typy snímačů:

- Fotoelektrický sensor (Photoelectric Sensor)
- Snímač vlastností (Property Sensor)
- Sensor přiblížení (Proximity Sensor)
- Snímač detekující dosažení polohy (Joint Value Sensor)
- Snímač monitorující polohu kloubu (Joint Distance Sensor) [9]

Aby snímač fungoval, je nutné jej aktivovat , dále pak se musí s výstupním signálem někde pracovat, například v modulech nebo v logickém bloku. K dodatečným úpravám snímače slouží ikona Edit Sensor Properties . Snímací paprsek lze zobrazit ikonou . [9]

3.4.1 Fotoelektrický snímač

Tento snímač je dvoustavový a slouží k detekci objektu nebo i více objektů.



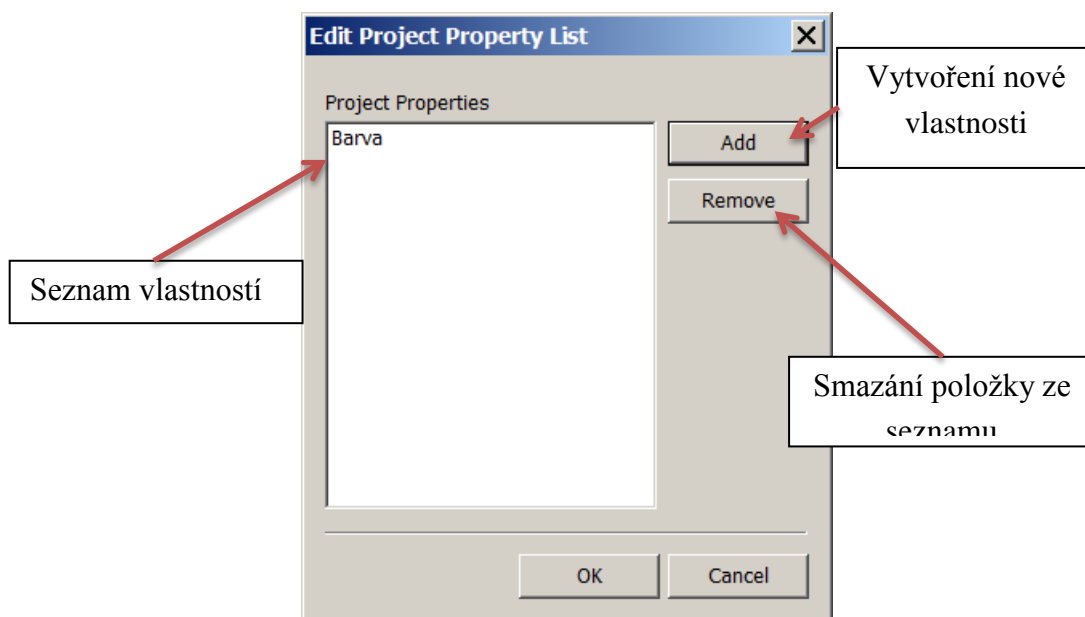
Obrázek 26 Nastavení parametrů snímače

Výstupem ze snímače je pak signál, jenž je aktivní při průchodu objektu, který má být detekován přes laserový paprsek.

3.4.2 Snímač vlastností

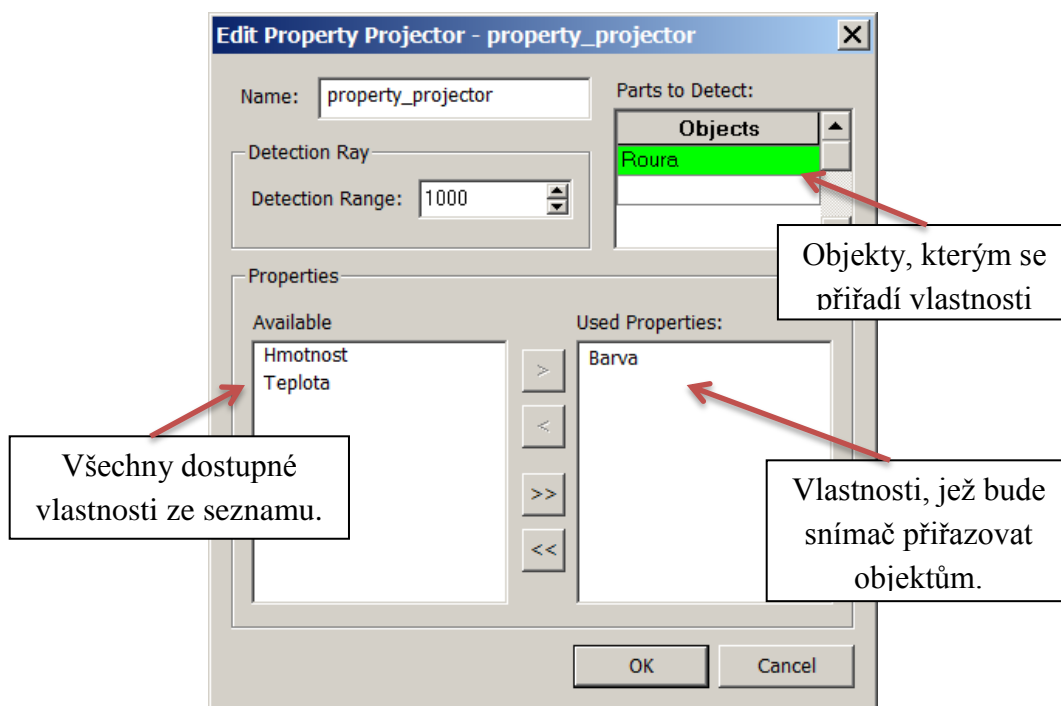
Obdobně jako fotoelektrický snímač slouží k detekci objektů. Na rozdíl od fotoelektrického je výstup snímače číslcový. Navíc detekční zóna může snímat trojrozměrný prostor.

Nejprve se musí nadefinovat vlastnosti, se kterými se bude pracovat. Vlastnosti se definují v Edit Part Simulation Property List [9].




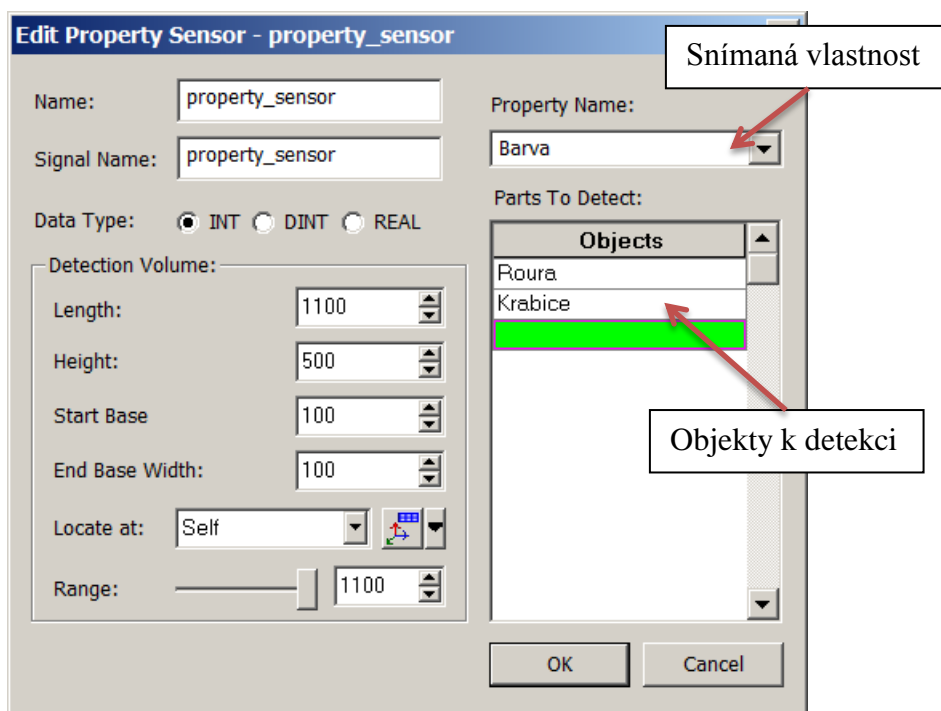
Obrázek 27 Definování vlastností

Aby bylo možné předmět detekovat, jsou k tomu potřeba dva snímače. První Property Projector [9] slouží k přiřazování vlastností určitému objektu [9].



Obrázek 28 Nastavení snímače přiřazující vlastnosti objektům


Výstupem ze senzoru jsou pak signály datového typu integer. Počet signálů pak odpovídá počtu vlastností, které má snímač přiřazovat. Číselnou hodnotu těchto signálů lze například nastavit v modulu main viz kapitola číslo 3.5.2. Objekt je pak s touto číselnou hodnotou propojen, jakmile protne paprsek projektoru. Například na Obrázek 28 projektor má výstupní signál se jménem „Property_projektor_barva“. Pokud pak objekt „Roura“ vstoupí do detekční zóny druhého snímače Property Senzor , který slouží k detekci předmětu, je pak vstupní hodnota signálu z Property Senzoru rovna hodnotě signálu „Property_projektor_barva“.



Obrázek 29 Nastavení snímače vlastností

Senzor vlastností na Obrázek 29 detekuje objekty „Roura“ a „Krabice“ a snímá vlastnost „Barva“. Abychom mohli rozlišit tyto dva objekty, je nutné mít dva signály, tudíž i dva projektory, jenž jeden nastaví číselnou hodnotu pro jeden objekt a druhý projektor nastaví číselnou hodnotu pro druhý objekt.

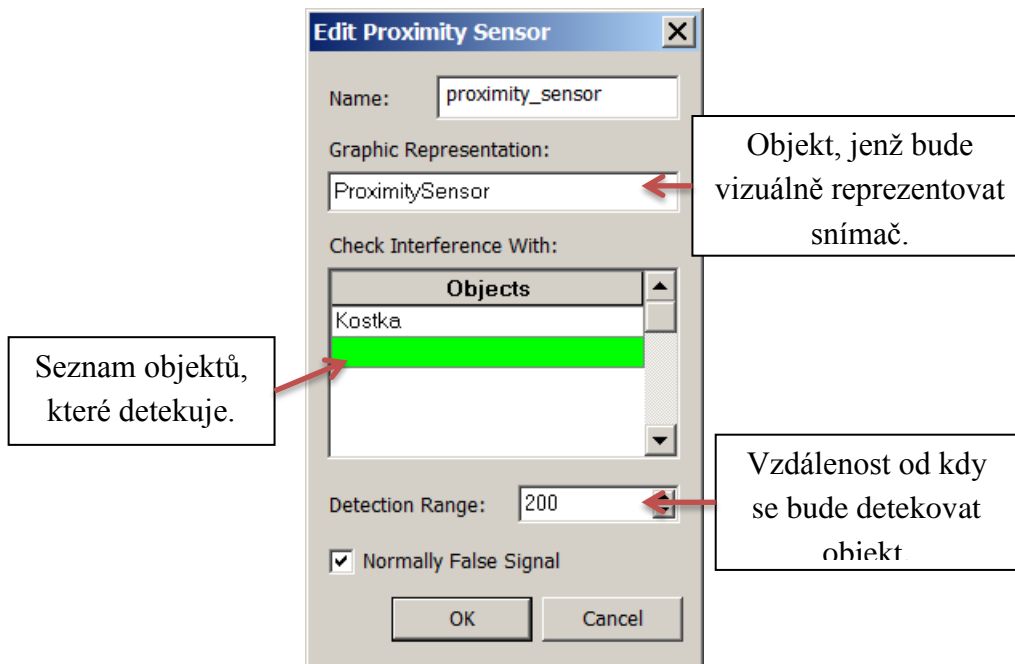
Okno zobrazené na Obrázek 29 se zobrazuje jen při aktivním Set Modeling

 Scope v záložce modeling. Jinak lze dodatečně zobrazit editační okno, ve kterém již nelze měnit šířku a výšku snímaného prostoru.

3.4.3 Snímač přiblížení

Jedná se o dvoustavový snímač detekující limitní přiblížení objektu. Snímat lze jeden nebo i více druhů objektů. Na rozdíl od fotoelektrického senzoru má tento senzor

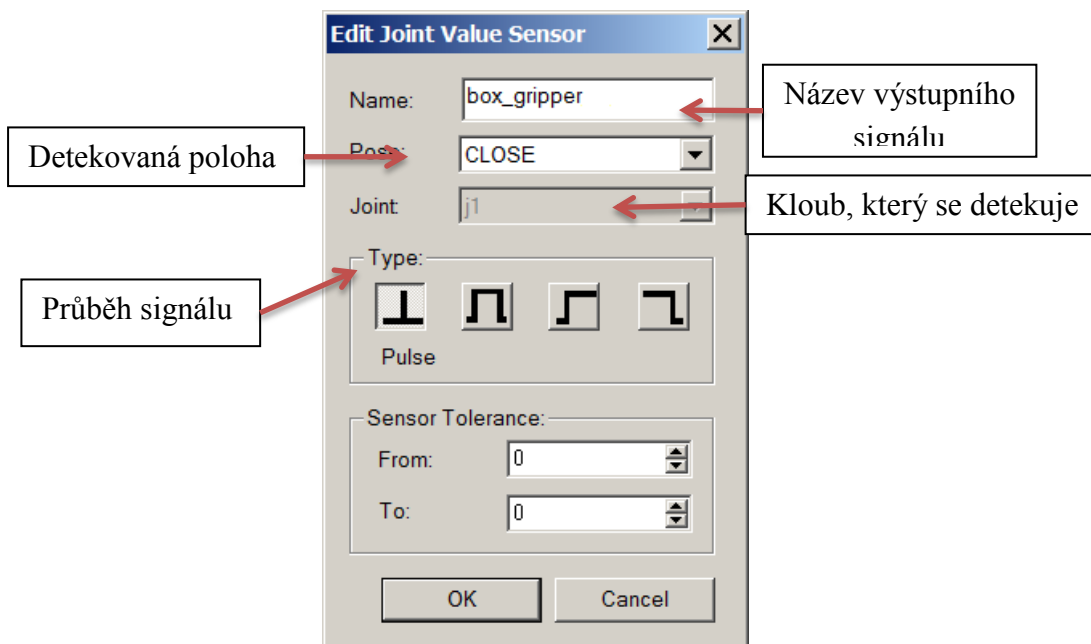
schopnost objekt detekovat v celém svém okolí, což je ale v simulaci více náročná početní operace, která tudíž zatěžuje více počítač.



Obrázek 30 Nastavení snímače přiblížení

3.4.4 Snímač detekující dosažení polohy

Tento snímač je dvoustavový, lze jím detekovat dosažení určité polohy robota nebo jeho koncové násady, například indikovat sevření uchopovacím zařízením. Snímač je možné vytvořit až po označení daného objektu, u kterého bude indikovat danou polohu.



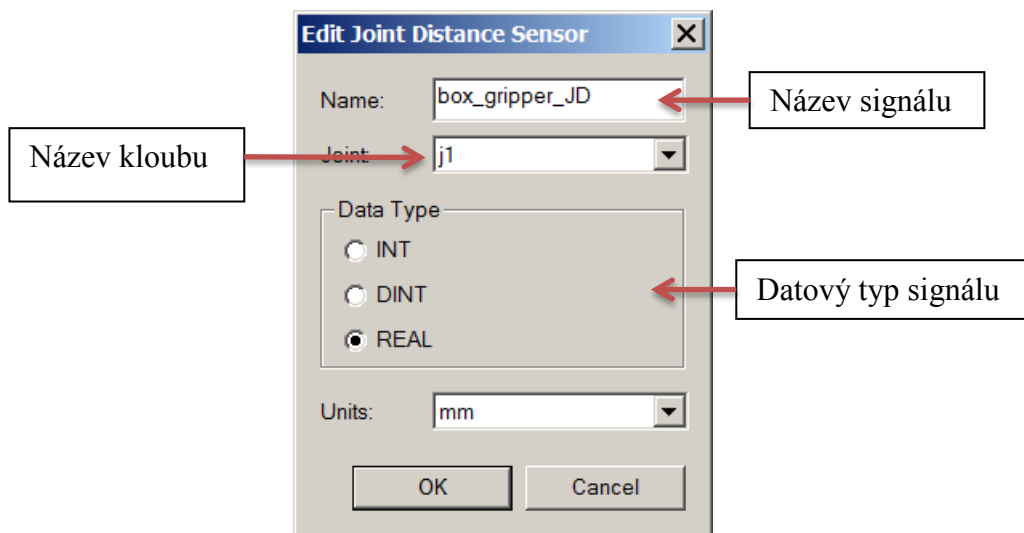
Obrázek 31 Nastavení senzoru detekující dosažení polohy

Položka „Senzor tolerance“ se používá při zadání rozsahů při obdélníkovém signálu. Signál je pak ve stavu logické jedničky právě tehdy, když je kloub v daném rozsahu. Například při poloze sevření má kloub hodnotu 150, nastaví-li se pak rozsahy -10 a 10, pak bude výstupní signál v logické jedničce, už když kloub bude nabývat hodnotu 140 až po hodnotu 160.

U nástupné hrany se nastavuje položka „From“, jakmile kloub dosáhne zadané hodnoty, bude výstupní signál ve stavu logické jedničky. Do stavu logické nuly přejde, až bude hodnota kloubu nižší než nastavená hodnota. U sestupné hrany je tomu naopak.

3.4.5 Snímač monitorující polohu kloubu

Tento snímač je číslicového formátu a udává aktuální hodnotu kloubu.



Obrázek 32 Nastavení snímače monitorujícího polohu

3.5 Řízení simulace

3.5.1 Signály





Pomocí signálů je řízena simulace procesu v Line Simulation mode. Signály lze rozdělit do více kategorií:

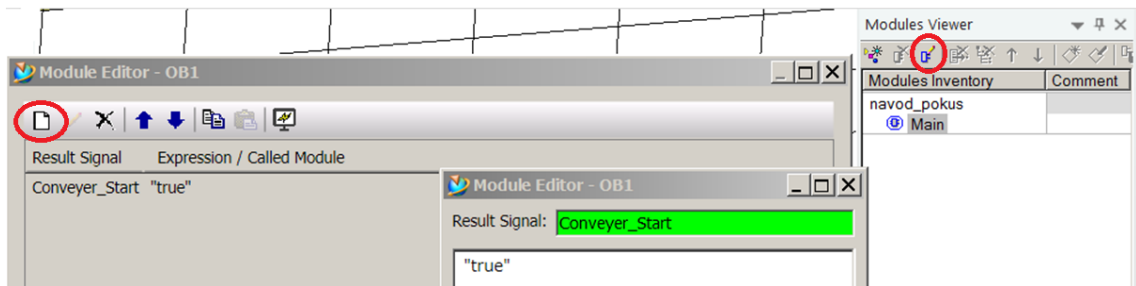
- Key Signals – Jsou definovány pouze pro program Proces Simulate. Jedná se například o automaticky vygenerované signály operací oznamující její konec.
- Display Signals – Jsou určeny pro zobrazování nějaké informace, lze je použít jako pomocné signály pro simulaci.
- Input Signals – Jsou vstupní signály do PLC
- Output Signals – Jsou výstupní signály z PLC [7]

Se signály lze pak pracovat v modulech viz kapitola Moduly a dále se používají pro přechod mezi operacemi, viz kapitola Operace.



3.5.2 Moduly

Moduly slouží pro řízení signálů v Line Simulation Modu. Je to vlastně základní prvek pro program, který vytváří logiku simulace. S moduly se pracuje v okně Modules Viewer. Ten lze zobrazit přes kartu View→Viewers→Modules Viewers.

Nový modul lze vytvořit ikonou New Module Object . Tím se vytvoří prázdný nový modul. S takto vytvořeným modulem lze pracovat pomocí Edit module . Základním prvkem je pak vstup (Entry) , do Result Signal se napíše jméno signálu, lze jej vložit i poklepnáním v signal vieweru. Tyto vstupy určují hodnotu Signálu. Delete module  pak slouží ke smazání daného modulu.



Obrázek 33 Nastavení signál v modulu

V Modules Hierarchy pak lze uspořádat moduly dle logiky programu. Jejich volání lze řídit podmínkami If-Else, ikony pro vytvoření nové podmínky:  a její zadání: . Moduly lze samozřejmě volat vícenásobně. Celý takto vytvořený program pak probíhá v jedné nekonečné super smyčce.

3.6 Operace

Pomocí operací se definují činnosti objektů v simulaci. V programu Tecnomatix Process Simulate lze vytvářet tyto operace:

- **Non-Sim Operation** – určeno zejména pro činnosti, které trvají v procesu nějakou dobu, ale nejsou v programu simulovány nebo lze použít pro čekání na další operaci
- **Object Flow Opertion** – pohyb objektů
- **Device Operation** – činnost nějakého zařízení, například otevření dveří
- **Gripper Operation** – činnost pro uchopovací zařízení, například jeho otevření
- **Weld Operation** – robotické svařování s namontovanou pistolí
- **Continuous Feature Operation** – spojitý tah po definované křivce, například laserové svařování


- **Pick and Place Operation** – uchopení a položení objektu
- **Generic Robotic Operation** – všeobecná robotická operace

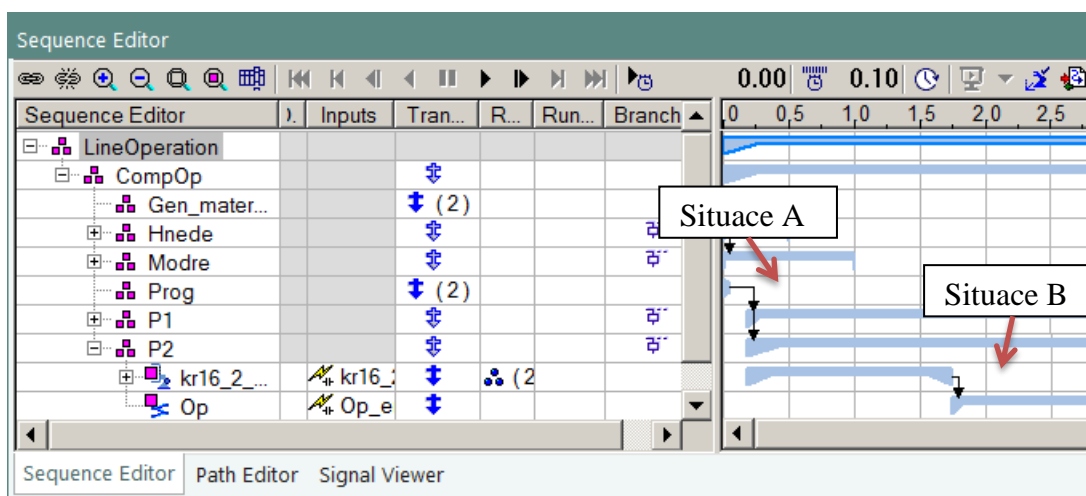
Různé operace lze taktéž spojovat do skupin operací.

3.6.1 Řízení operací

Přechod z jedné operace do druhé v Line Simulation Modu je možný jen za splnění podmínky v transition (přechod). Podmínku lze nastavit poklepnutím na modrou ikonu transition v položce Sequence Editoru. Z počátku je zde nastaveno, že přechod proběhne ukončením dané operace, například signál Op_end. Tento signál se automaticky vygeneroval při vytvoření operace s názvem Op.

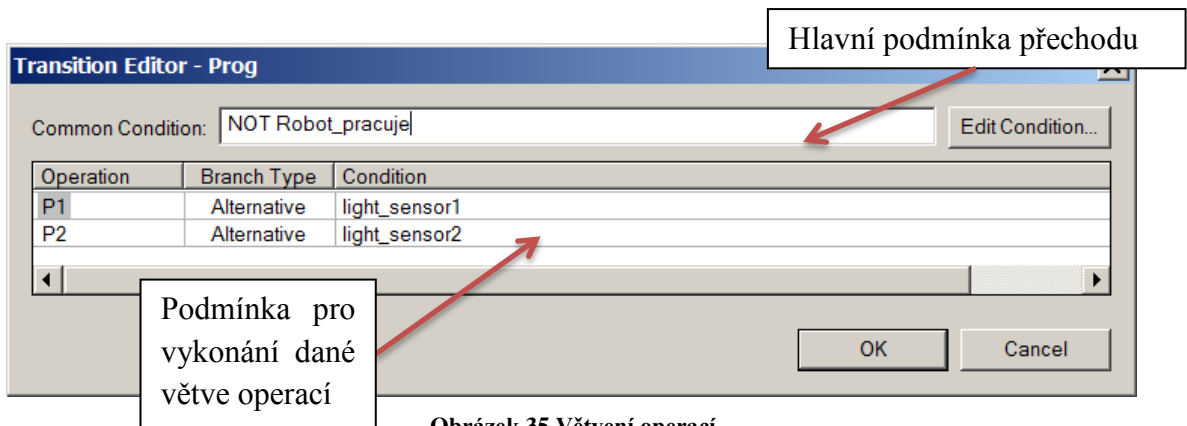
3.6.1.1 Větvení

Jak by se měly přechody mezi operacemi větvit, nám určují odkazy (Link). [9] Takto lze spojovat skupiny operací nebo jednotlivé operace v rámci jedné skupiny. Spojení se provádí tak, že se vyberou dvě položky, které chcete propojit a pak zvolit Link . Mezi operacemi by se měla zobrazit černá šipka. Na příkladu viz Obrázek 34 situace A popisuje větvení skupin operací do dvou alternativních větví.



Obrázek 34 Sekvence operací

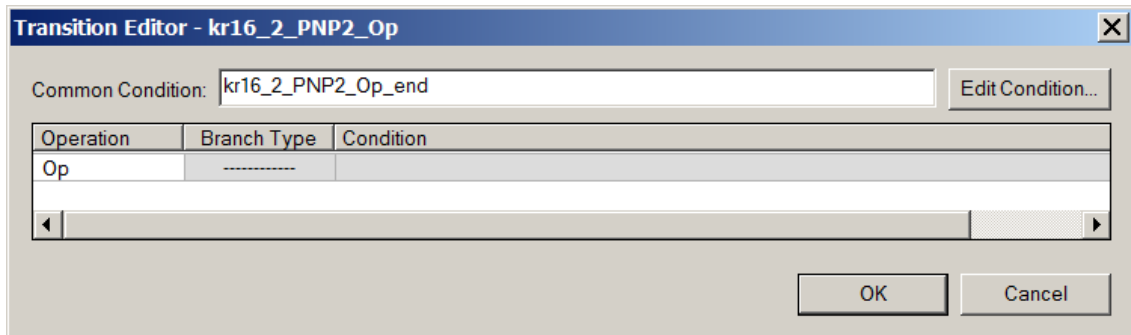
Chceme-li program větvit na více větví vykonávaných operací, lze pro dané skupiny vytvořit alternativní větve. Aby mohlo dojít k přechodu do další operace, musí být nejprve splněna hlavní podmínka. Alternativní větev se pak aktivuje, je-li splněna její podmínka [9]. Zde je to pro první větev sepnutí prvního senzoru a pro druhou větev sepnutí druhého senzoru.



Obrázek 35 Větvení operací

Lze vytvářet i větve, které se vykonají vždy při splnění hlavní podmínky pro přechod, tyto větve se nazývají simultánní (Simultaneous) [9].


V situaci B se provádí postupně dvě jdoucí operace za sebou. Nejprve robot provede operaci sebrání a položení objektu, a až poté se vykoná operace přesunutí objektu.

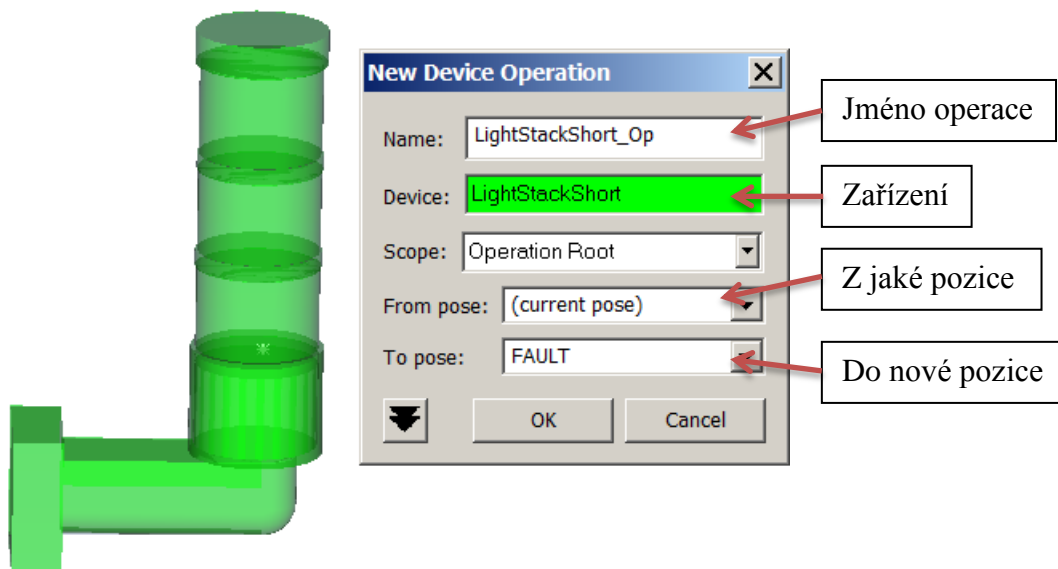


Obrázek 36 Přejít na druhou operaci

Z podmínky přechodu vyplývá, že k přechodu do operace přesunu objektu (Op) dojde, až se signál oznamující konec operace sebrání a položení.

3.6.2 Vytvoření operací pro zařízení

Na příkladu signalizačního majáku popíši, jak se vytváří operace pro zařízení. Po označení a pravým kliknutím myši na maják se zvolí New Device Operation .

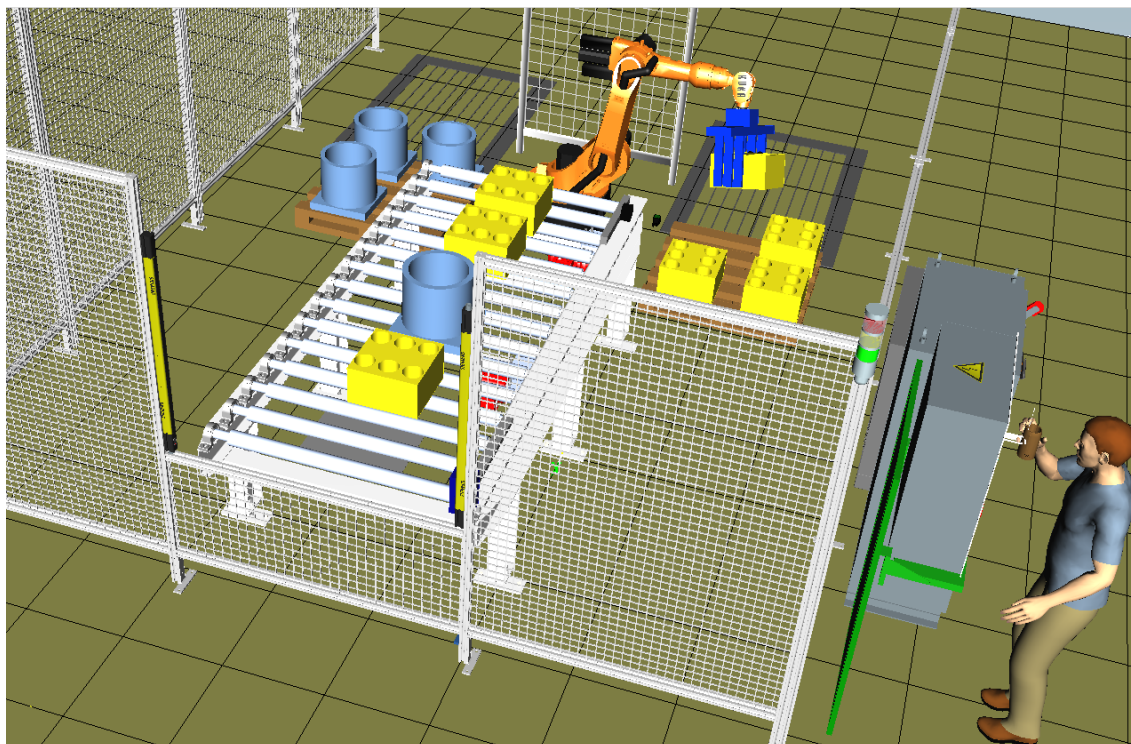


Obrázek 37 Nastavení operace zařízení

Činnost operace definuje, z jakého stavu (pozice) se má zařízení dostat do nového stavu.

4 MODEL TŘÍDÍCÍ LINKY

Jako demonstrační úlohu řešenou v prostředí Tecnomatix Process Simulate jsem zvolil třídění dvou výrobků. Výrobní linka se skládá z jednoho dopravníkového pásu, na němž se náhodně generují dva různé výrobky. Pracovně jsem tyto díly podle jejich vzhledu nazval „Kvádr“ a „Roura“. Stacionární robot na konci dopravníku tyto dva výrobky roztřídí na dvě palety.



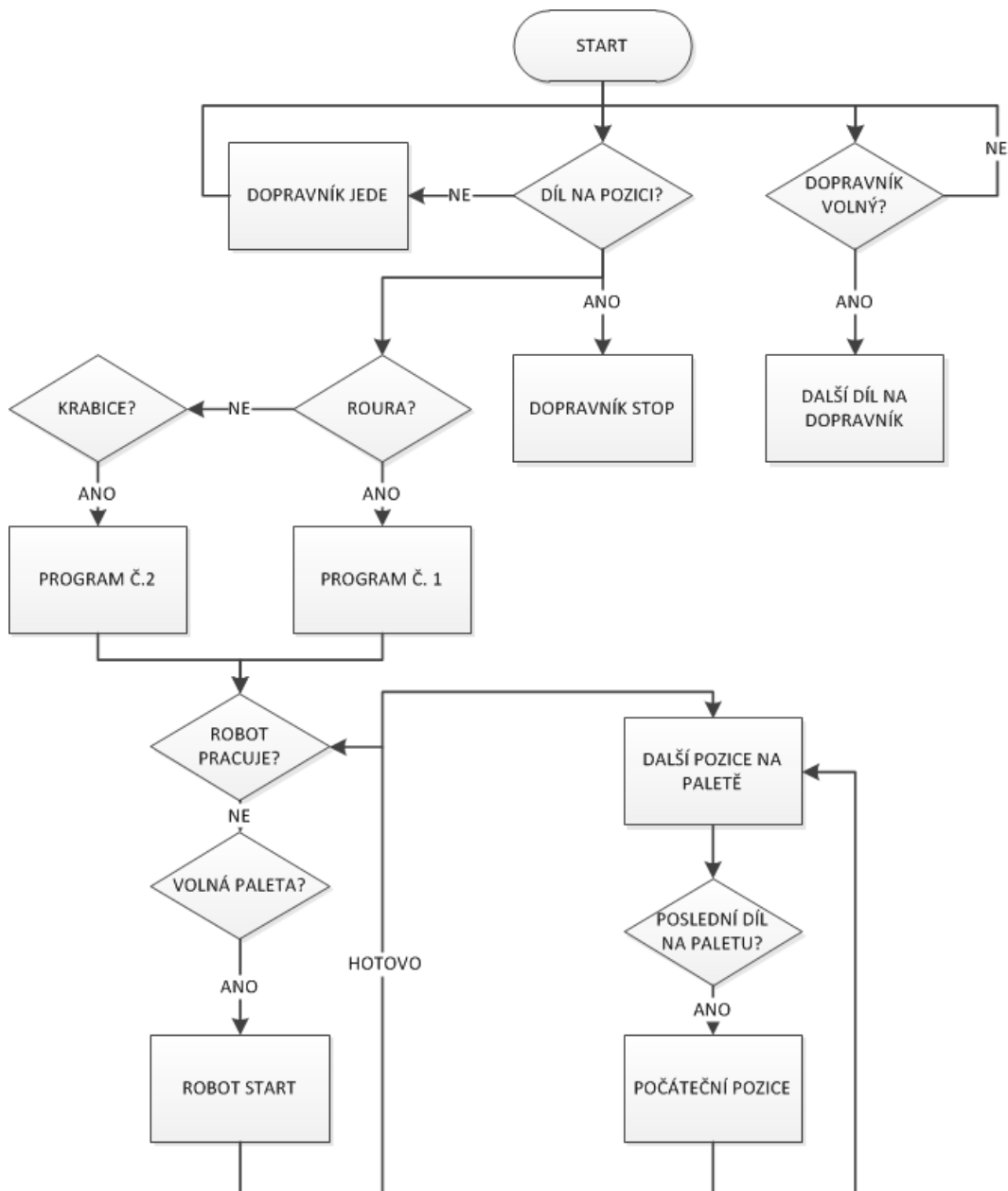
Obrázek 38 Model linky

4.1 Rozvrhnutí linky

Na dopravníkový pás se budou vkládat díly pouze tehdy, je-li volné místo na začátku dopravníku. Toto bude kontrolovat Property senzor. Jakmile díl dojíždí na konec dopravníku, indikuje ho laserový snímač, čímž se zastaví dopravníkový pás. Druhý Property senzor, který je umístěn v oblasti laserového snímače rozpoznává, zda se jedná o díl „Kvádr“ nebo „Roura“. Díl „Kvádr“ se skládá na levou paletu, na jednu paletu se pak naskládají čtyři díly. Druhý díl „Roura“ se pak skládá na pravou paletu. Robot může skládat díly jen v případě, že je na paletě ještě volné místo, to kontroluje snímač přiblížení, který je u místěn v prostoru, kde se pokládá poslední díl.

4.2 Algoritmus řešení

Vývojový diagram na Obrázek 39 popisuje, jak jsem realizoval řešení této úlohy v prostředí Tecnomatix Process Simulate. Celý tento proces pak běží cyklicky. Jelikož jsem pro řízení linky použil signály, a ty jsou aktivní jen v režimu Line Simulation Mode. Je potom pro správné fungování simulace nutné. Simulaci vždy spouštět právě v režimu Line Simulation Mode.





Obrázek 39 Vývojový diagram třídící linky

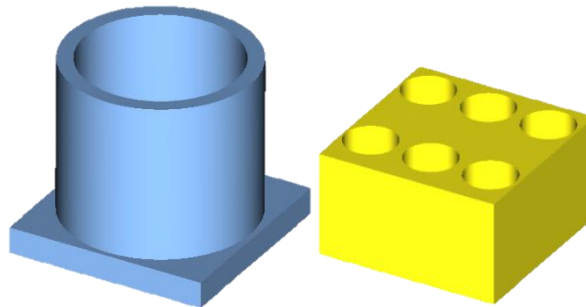
4.3 Realizace v prostředí Tecnomatix Process Simulate

Nejprve jsem realizoval řízení simulace třídící linky přímo v prostředí Tecnomatix Process Simulate. Simulace je realizována v Line Simulation módu, tudíž simulace je řízena událostmi. Následující kapitoly popisují, jak jsem postupoval při tvorbě modelu třídící linky v prostředí Tecnomatix Process Simulate.

4.3.1 Objekty v modelu

4.3.1.1 Díly

V prázdném projektu jsem si nejprve namodeloval dva díly, které se budou třídit. Tyto díly jsou položkou part, takže jsem se přepnul do standartního módu  a v kartě modeling zvolil . Podrobnější popis, jak v prostředí Tecnomatix Process Simulate upravovat objekty je popsáno v kapitole číslo 3.2 - Objekty.





Obrázek 40 Díl „Roura“ a díl „Kvadr“

Jelikož ale simulace bude řízena událostmi, což je v Line Simulation módu, je potřeba pracovat s Appearancemi (zjevy) místo parts (díly). Proto jsem ještě ve standartním módu pro oba díly vytvořil operaci, kterou přemístuji díly na začátek dopravníku. Dále jsem vytvořil jednu Compound operaci, do níž jsem vložil obě Object Flow operace. To jsem udělal z důvodu, že při přepnutí do Line Simulation módu musí být všechny operace zahrnuty ve větvi nějaké Compound operace.

4.3.1.2 Dopravník

Nejprve jsem z knihovny objektů vložil objekt dopravníku, ten má jen symbolický charakter. Pro simulaci je podstatnější vytvoření křivky trajektorie pohybu po dopravníku. Jak tuto křivku vytvořit a jak pro ni nadefinovat, které díly po ní budou jezdit, detailně popisuje kapitola číslo 3.3 - Dopravník.

Nejprve jsem si otestoval, jestli dopravník díly přesouvá pomocí ikony . Toto jsem testoval ve standartním módu, protože tato ikona je v Line Simulation módu neaktivní. Při přepnutí do Line Simulation módu jsem musel znovu zmáčknout ikonou  a pak definoval, jaké objekty se budou po dopravníku pohybovat, nyní to ale byly


appearance. Pro appearance je potřeba definovat tok materiálu. Jak se tvoří tok materiálu je popsáno v kapitole číslo 3.2.3 - Generování toku materiálu. Pro otestování funkčnosti dopravníku jsem zvolil jednoduchý tok, kdy nejprve se Object Flow operací posune díl na dopravník a pak Non-Sim operaci, jež má nastaveno několik sekund trvání, setrvává díl na dopravníkovém pásu. Zde je vhodné ještě upozornit, že je dobré nastavit dráhu Object Flow operace dostatečně dlouhou, aby nedocházelo k možnému překlápění dílů na dopravníkovém pásu.



Pro dopravník jsem ještě v jeho logickém bloku vytvořil dva výstupní signály, a to pro spuštění a zastavení signálu.


Posléze jsem na dopravník přidal optický snímač pro zastavení pásu, když díl dojíždí na konec pásu, dále dva property senzory, prvním na začátku kontroluji volný prostor na začátku pásu, aby se díly vygenerovaly jen v případě volného místa na začátku. Druhým property senzorem, jež je umístěn ve stejném místě jako optický snímač, rozpoznávám, zda se nabírá díl „Roura“ nebo „Kvádr“. Pro funkčnost property senzorů je potřeba přidat ještě property projektory. Jejich počet závisí na počtu objektů, které property senzory mají rozlišovat. V tomto případě jsem potřeboval dva property senzory. První property projektor detekoval díl „Roura“, jeho signál jsem v modulu nastavil na hodnotu 1, druhým property projektorem jsem detekoval díl „Kvádr“, signál nabývá hodnotu 2. Oba dva projektory přiřazují tím vlastnost pojmenovanou „Color“. Rozmístění snímačů je zobrazeno v kapitole číslo 4.3.5 - Rozmístění snímačů. Jak nastavovat snímače je pak popsáno v kapitole číslo 3.4 - Snímače.

4.3.1.3 Robot

Pro paletizace jsem zvolil robota KUKA kr16/2, protože svými rozměry postačuje k uchopení dílu z dopravníkového pásu a položením na palety, které jsou umístěny na levé a na pravé straně robota.

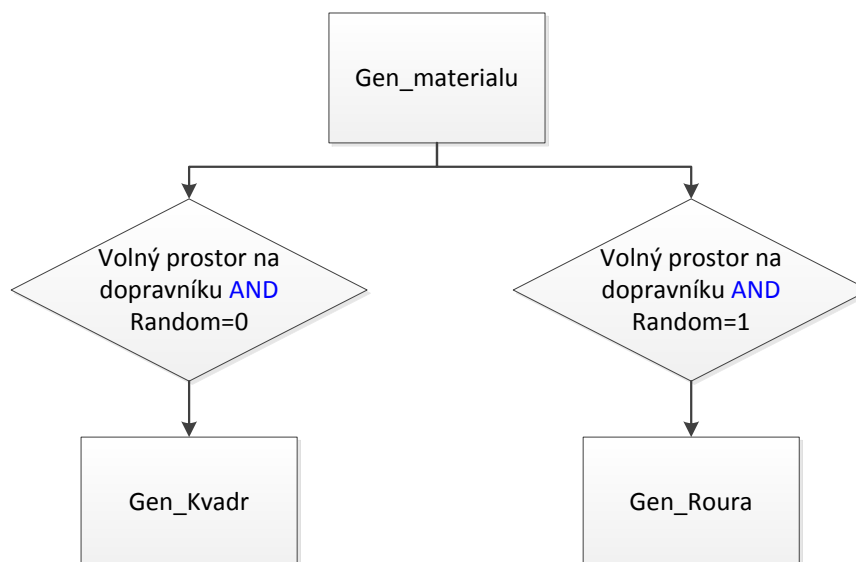
K robotu bylo potřeba připevnit ještě uchopovací zařízení, jak se takové zařízení připevňuje k robotu popisuje kapitola číslo 3.1.2 - Uchopovací zařízení. Zvolené uchopovací zařízení není zcela vhodné, protože je příliš velké, avšak vhodnější jsem v původní knihovně objektů nenašel. Abych našel přesné místo, kde se díl, který se má uchopit zastaví, jsem spustil simulaci a po zastavení pásu jsem ji přerušil. Pak jsem robotem pomocí Robot Jog  najel k dílu a nastavil pro uchopovací zařízení novou pozici pro uchycení.

Poté jsem vytvořil operaci Pick and Place. Jako uchopovací pozici jsem určil rám (Frame) daného dílu a pozici pro položení jsem určil místo na paletě. Dále bylo nutné v Path editoru přidat body, kterými má robot projet, aby nedocházelo ke kolizím. Nejprve jsem nastavil počáteční pozici robota, tu jsem nastavil tlačítkem Add Current Location . Potom jsem další průchozí body přidával tlačítkem Add Location After . Tím se aktivovalo okno Robot Jog, manipulací v něm jsem pak nastavil

požadovanou polohu robota, pokud by nestačilo hýbat jen pomocí os X Y Z, lze ještě v sekci All Joints nastavovat polohy jednotlivých kloubů robota. Pro dodatečnou manipulaci slouží ikona Manipulate Location . Takto jsem vytvořil celkem osm Pick and Place operací. Tyto Pick and Place operace jsem pak vložil do robotického programu. Jak se vytváří robotický program popisuje kapitola číslo 3.1.3 - Nahrání programu do robota. Jednotlivým Pick and Place operacím jsem přiřadil odlišné číslo (Path number). Pro pokládání „Roury“ to jsou čísla 10-13 a pro „Kvádr“ to jsou čísla 20-23. Hodnotou signálu „kr16_2_programNumber“ se pak vyvolávají příslušné Pick and Place operace.

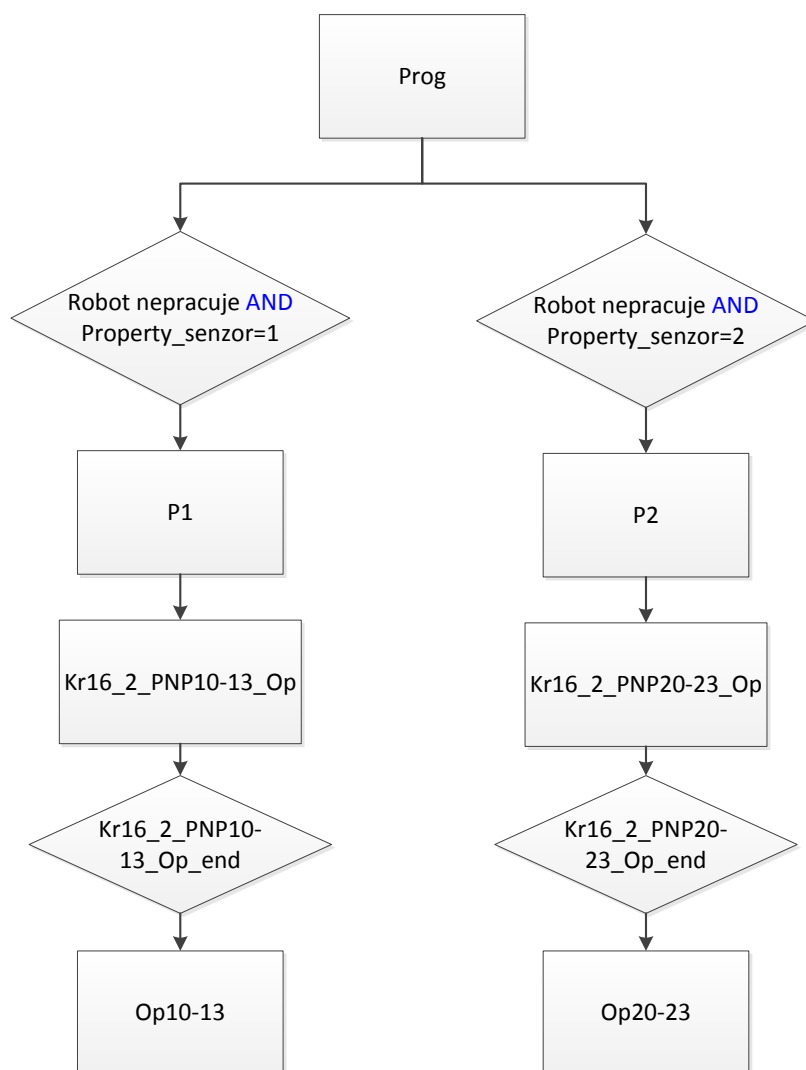
4.3.2 Posloupnost operací

Jednotlivé operace jsem seskupil do logicky tvořených skupin. První skupinou operací je generování materiálu, její název je „Gen_materialu“. V této skupině jsou obsaženy dvě Object Flow operace, a to pro příchod „Kvádrů“ nebo „Roury“ na dopravník. Jelikož nelze propojit odkazem položku skupina operací a jednotlivou operací, jsou tyto operace vloženy do skupiny „Kvádr“ respektive „Roura“. Nastavenou podmínkou přechodu se pak aktivuje buď operace pro generování „Kvádry“ nebo „Roury“, viz Obrázek 41. Společnou podmínkou pro přechod do jedné z Object Flow operace je volný prostor na začátku dopravníku, to znamená senzor „Volno=0“, pokud není podmínka splněna, vykonává se operace „Gen_materialu“, která nekoná žádnou činnost a jen se zde vyčkává na splnění podmínky přechodu. Proměnná „Random“ může nabývat hodnot nula nebo jedna, její hodnota se mění vždy po dokončení jedné z Object Flow operace, a to pomocí funkce „RANDOM(“0““2“)“.



Obrázek 41 Struktura operací pro generování materiálu

Skupina „Prog“ sdružuje skupiny operací „P1“ a „P2“, ty vykonávají proces pokládání dílů na paletu. Pick and Place operace určují, odkud kam robot nabere a položí díl. Non-Sim operace symbolizují čekání na odvoz palety.

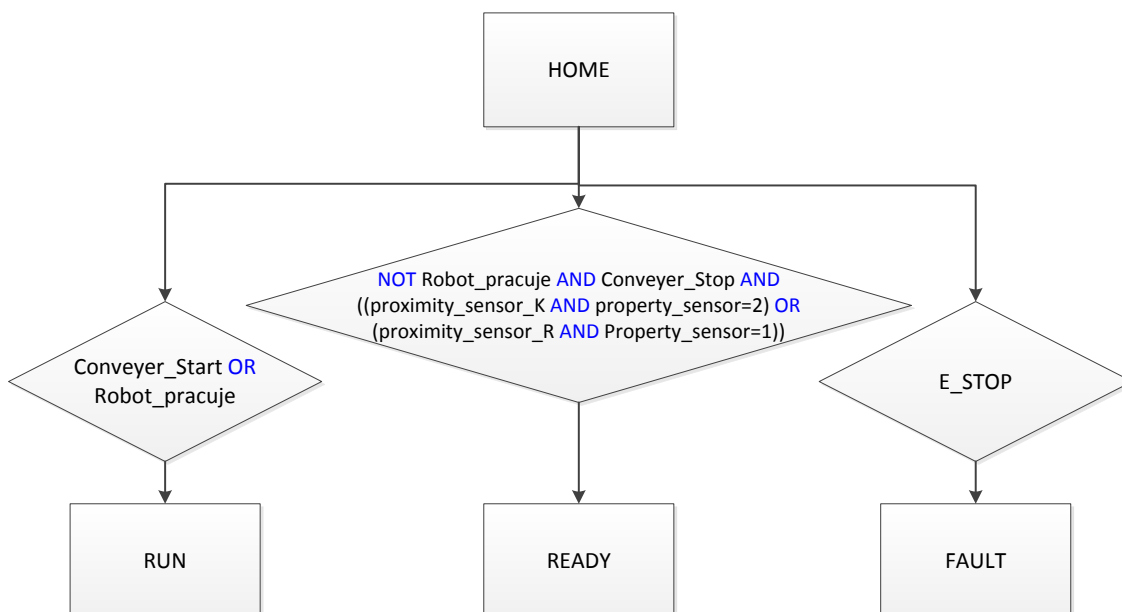


Obrázek 42 Struktura operací pro pokládání dílů

Pokud robot právě nevykonává žádnou činnost a zároveň „Property_senzor“, který rozlišuje díl na konci dopravníku ve stavu 1 nebo 2 aktivuje se program „P1“ respektive program „P2“. Až se položí poslední díl na paletu, což jsou operace „kr16_2_PNP_13“ nebo „kr16_2_PNP_23“, tak jsou aktivovány operace „Op10“ až „Op13“ nebo „Op20“ až „Op23“. Tyto operace simulují čekání na odvoz palety.

Poslední skupina se nazývá Stav. Obsahuje operace pro přepínání světla na kontrolním majáku. Na majáku může svítit:

- Zelená – linka běží
- Žlutá – linka čeká na další úkol, například paleta je plná, tudíž další stejný díl momentálně nelze položit.
- Červená – nouzové zastavení



Obrázek 43 Struktura operací pro maják

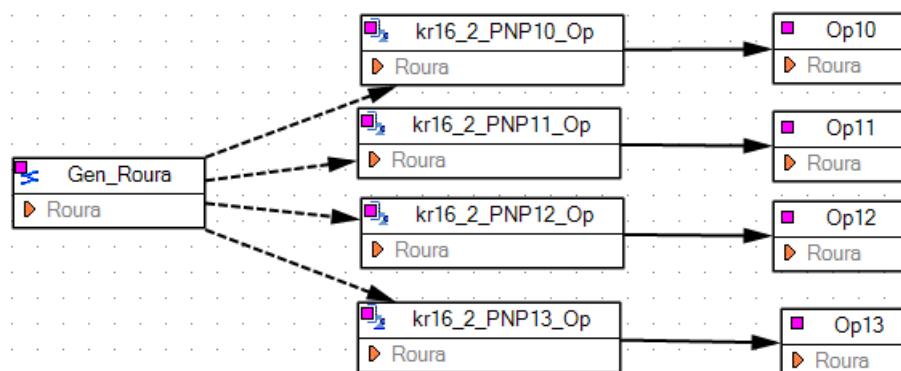
Jména operací vychází z originálního pojmenování poloh majáku. Pro přehlednost jsem tuto strukturu operací vložil do skupiny operací stav.

Tlačítkem E-Stop ve SCADA aplikace se aktivuje povel pro nouzové zastavení robota „kr16_2_emergencyStop“.

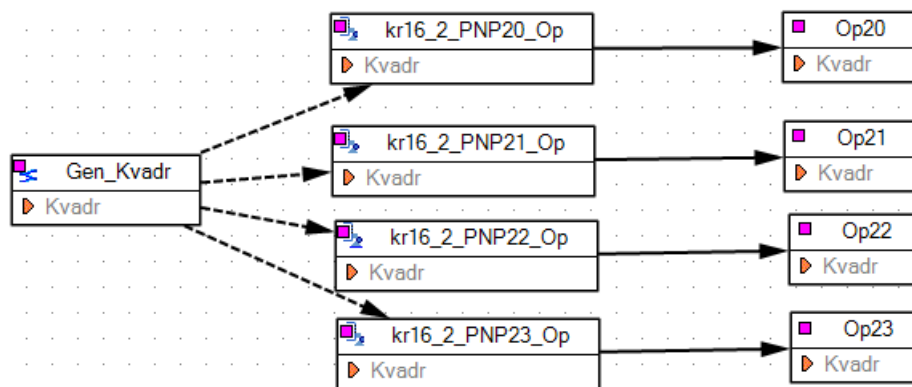
Podrobný popis jak se tvoří větvení operací a další práce s operacemi je popsáno v kapitole číslo 3.6 - Operace.

4.3.3 Tok materiálu

Oba díly mají totožný tok materiálu. Nejprve se Object Flow operací vygenerují na začátku dopravníku, pak proběhne jedna z Pick and Place operací, čím se díl přesune z dopravníku na paletu. Poslední operace symbolizuje dobu čekání, než se paleta vyprázdní. Tuto dobu jsem v operation properties nastavil na 10 sekund.



Obrázek 44 Tok materiálu dílu „Roura“



Obrázek 45 Tok materiálu dílu „Kvadr“

Čárkovaná šipka symbolizuje alternativní cestu toku materiálu. Operace „Op10“ až „Op13“ a „Op20“ až „Op23“ simulují čekání dílu, jejich vícenásobné zastoupení jsem zvolil proto, aby se zajistil korektní tok materiálu. Při volbě jen jedné Op operace, do které se sbíhaly všechny Pick and Place operace, díly na paletě zůstávaly i po uplynutí času.

4.3.4 Logika programu

Pro vytvoření řídicího programu jsem použil moduly, ve kterých jsem pracoval se signály, jak je již popsáno v kapitole číslo 7 - Řízení simulace.

Název	Datový typ	Popis
kr16_2_programEnded	BOOL	Informace o konci programu
light_sensor_stop	BOOL	Snímač k zastavení pásu
proximity_sensor_K	BOOL	Detekce posledního „Kvádrů“ na paletě
proximity_sensor_R	BOOL	Detekce poslední „Roury“ na paletě
proximity_sensor_volno	BOOL	Kontrola volného prostoru na začátku pásu
property_sensor	INT	Rozlišení typu dílu
senzor_volno	INT	Kontrola volného prostoru na začátku pásu

Tabulka 2 Vstupní signály

Název	Datový typ	Popis
Conveyer_Start	BOOL	Spuštění dopravníkového pásu
Conveyer_Stop	BOOL	Zastavení dopravníkového pásu
kr16_2_startProgram	BOOL	Spuštění robota
kr16_2_emergencyStop	BOOL	Nouzové zastavení robota
kr16_2_programPause	BOOL	Pozastavení robota
kr16_2_programNumber	INT	Číslo spouštěného programu

Tabulka 3 Výstupní signály

Název	Datový typ	Popis
Robot_pracuje	BOOL	Monitorování stavu robota
Projector_roura	INT	Číslo pro identifikaci dílu „Roura“ senzory vlastností
Projector_kvadr	INT	Číslo pro identifikaci dílu „Kvadr“ senzory vlastností
Cis_dilu_R	INT	Počítání pozice položení dílu „Roura“
Cis_dilu_K	INT	Počítání pozice položení dílu „Kvadr“
Random	INT	Náhodně vygenerované pro vygenerování nového dílu

Tabulka 4 Displej signály

Název	Datový typ	Popis
kr16_2_PNP10_Op_end	BOOL	Konec 1. položení „Roury“
kr16_2_PNP11_Op_end	BOOL	Konec 2. položení „Roury“
kr16_2_PNP12_Op_end	BOOL	Konec 3. položení „Roury“
kr16_2_PNP13_Op_end	BOOL	Konec 4. položení „Roury“
kr16_2_PNP20_Op_end	BOOL	Konec 1. položení „Kvádru“
kr16_2_PNP21_Op_end	BOOL	Konec 2. položení „Kvádru“
kr16_2_PNP22_Op_end	BOOL	Konec 3. položení „Kvádru“
kr16_2_PNP23_Op_end	BOOL	Konec 4. položení „Kvádru“

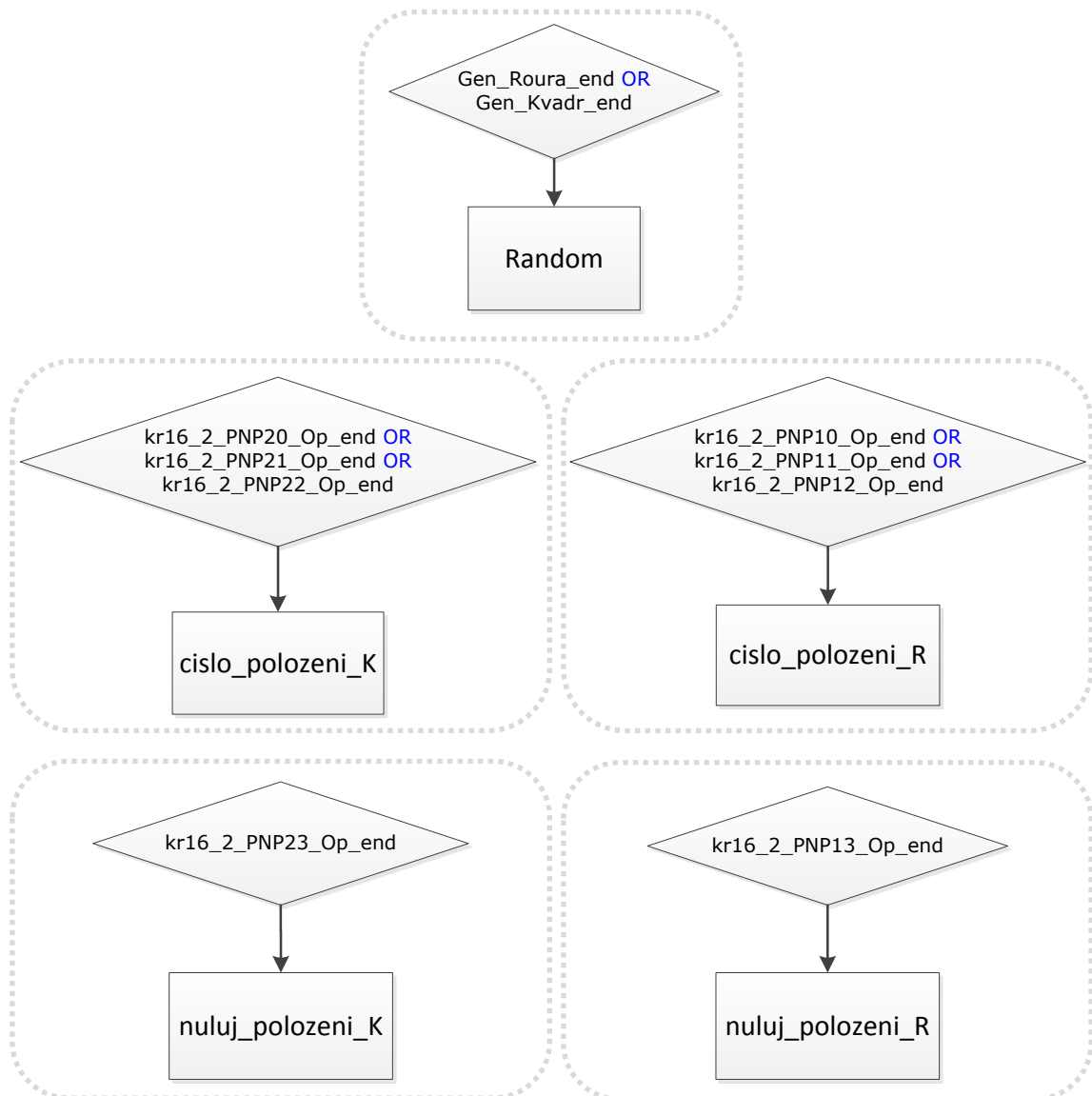
Tabulka 5 Key signály

Definování zda se jedná o vstupní nebo výstupní signál závisí na tom, zda signál je pro PLC vstupem nebo jeho výstupem. Key signály lze použít jen v programu Tecnomatix Process Simulate, vytváří se automaticky s novou operací a informují o jejím vykonání. Tabulka 5 zobrazuje jen key, které jsem využil v logice programu.

Název modulu	Popis
Zastaveni	Zastavení dopravníkového pásu
Jede	Rozjetí dopravníkového pásu
Robot_start	Povel na spuštění robota
Robot_reset	Přípravení robota k další práci
P1	Nastavení čísla programu pro díl „Rouru“
P2	Nastavení čísla programu pro díl „Kvadr“
Random	Generování čísla pro určení, který díl se objeví na pásu
Nuluj_polozeni_K	Vynulování čísla pozice pro položení u „Kvadr“
Nuluj_polozeni_R	Vynulování čísla pozice pro položení u „Roury“
Cislo_polozeni_K	Inkrementace čísla položení pro „Kvadr“
Cislo_polozeni_R	Inkrementace čísla položení pro „Rouru“

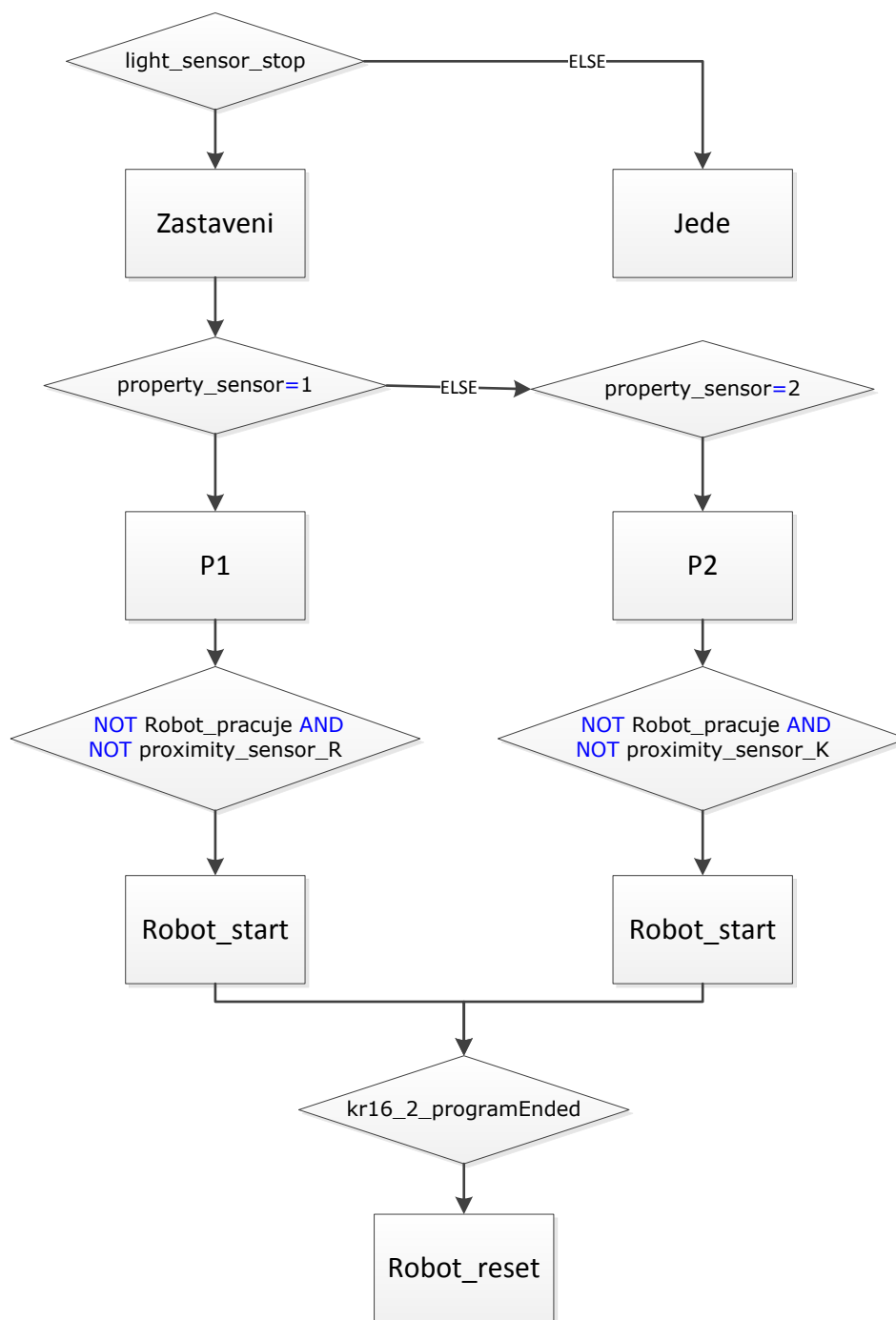
Tabulka 6 Vytvořené moduly

Tyto moduly jsem pak poskládal do logické posloupnosti větvení pomocí podmínek if-else v Modules Hierarchy. Celá hierarchie modulů se volá v nekonečné smyčce.



Obrázek 46 Hierarchie modulů pro práci s díly

V modulu „Random“ je volána funkce „RANDOM(“0“ “2“)“. Funkce náhodně vygeneruje 0 nebo 1. Hodnota z funkce se přiřadí do signálu „Random“. V modulech „cislo_polozeni_K“ a „cislo_polozeni_R“ se po splnění podmínky inkrementuje signál „Cis_dilu_K“ respektive „Cis_dilu_R“. K vynulování těchto signálů dochází v modulech „Nuluj_polozeni_K“ respektive „Nuluj_polozeni_R“.



Obrázek 47 Hierarchie modulů pro řízení třídící linky

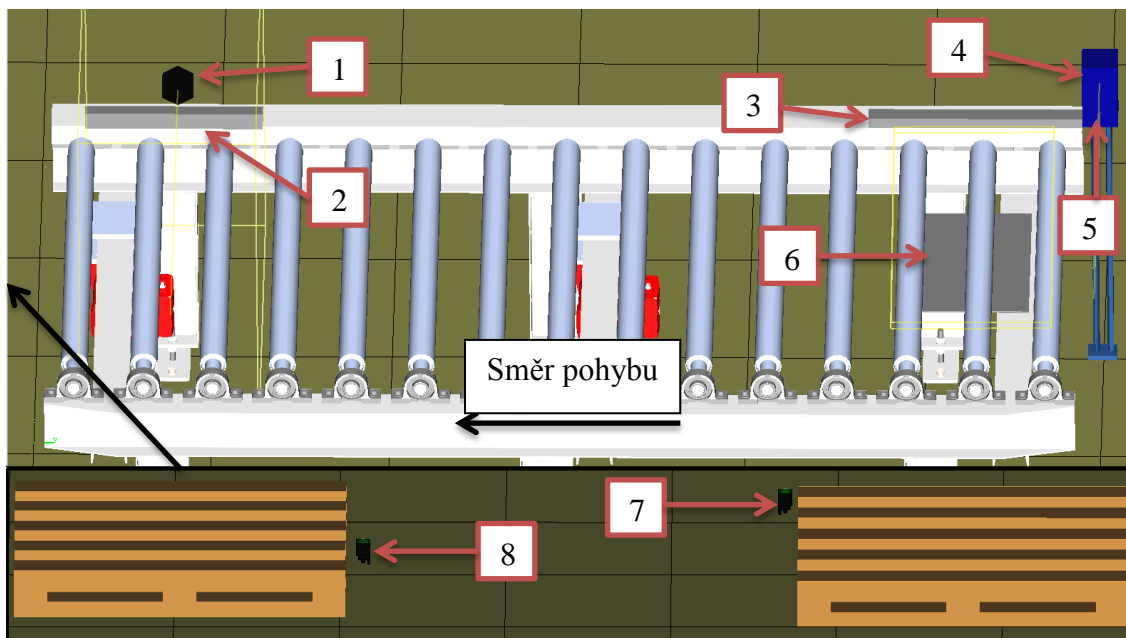
Modul „*Jede*“ je aktivní v případě neseprnutého snímače „*light_sensor_stop*“. Tento modul zapíná dopravníkový pás. Při sepnutí snímače „*light_sensor_stop*“ se aktivuje modul *Zastaveni*, ten vypíná dopravníkový pás. Modul „*P1*“ nebo „*P2*“ se aktivuje, je-li „*property_sensor*“ roven 1 respektive 2. V těchto modulech se nastaví číslo programu pro další robotickou operaci. Modulem „*Robot_start*“ se pak spouští robot. Po dokončení robotického programu se aktivuje modul „*Robot_reset*“. V něm se do logické nuly nastaví signály „*Robot_pracuje*“ a „*kr16_2_startProgram*“.

4.3.5 Rozmístění snímačů

V kapitole číslo 3.4 - Snímače detailně popisují jednotlivé typy snímačů, které lze v prostředí Tecnomatix Process Simulate použít, dále vysvětlují jejich nastavení. Pro realizaci třídící linky jsem použil následující snímače:

č.	Typ	Název	Funkce
1	Fotoelektrický snímač	light_sensor_stop	stopnutí pásu po detekci dílu
2	Snímač vlastností	property_sensor	rozpoznání dílu
3	Snímač vlastností	Volno	detekce volného prostoru na začátku dopravníku
4	Projektor	property_projector_R	přiřazení vlastnosti dílu „Roura“
5	Projektor	property_projector_K	přiřazení vlastnosti dílu „Kvadr“
6	Snímač přiblížení	Proximity_Sensor_volno	detekce volného prostoru na začátku dopravníku
7	Snímač přiblížení	proximity_sensor_R	detekce poslední „Roury“ na paletě
8	Snímač přiblížení	proximity_sensor_K	detekce posledního „Kvadr“ na paletě

Tabulka 7 Snímače



Obrázek 48 Rozmístění snímačů

Obrázek 48 zobrazuje rozmístění snímačů na třídící lince. Čísla označující snímače odpovídají číslům v Tabulka 7.

U všech snímačů jsem zvolil pozitivní logiku, takže pokud snímače detekují objekt, tak přejdou ze stavu logické nuly do stavu logické jedničky.

Snímače pro detekování volného prostoru na začátku dopravníkového pásu jsem zvolil redundantně proto, abych demonstroval možnost použití různých snímačů. První volbou byl snímač vlastností, protože na rozdíl od fotoelektrického snímače jím lze detekovat 3D prostor.

Jelikož nezáleží, jaký díl se nově objevil na začátku dopravníku, lze použít dvoustavový senzor přiblížení. Nevýhodou pro uživatele je při umístování to, že v programu nelze zobrazit detekční zónu, jak je tomu u senzoru vlastností, což nepatrně znepříjemňuje hledání vhodné polohy. Na druhou stranu by se v reálném modelu jednalo o levnější řešení než použití kamerového systému.

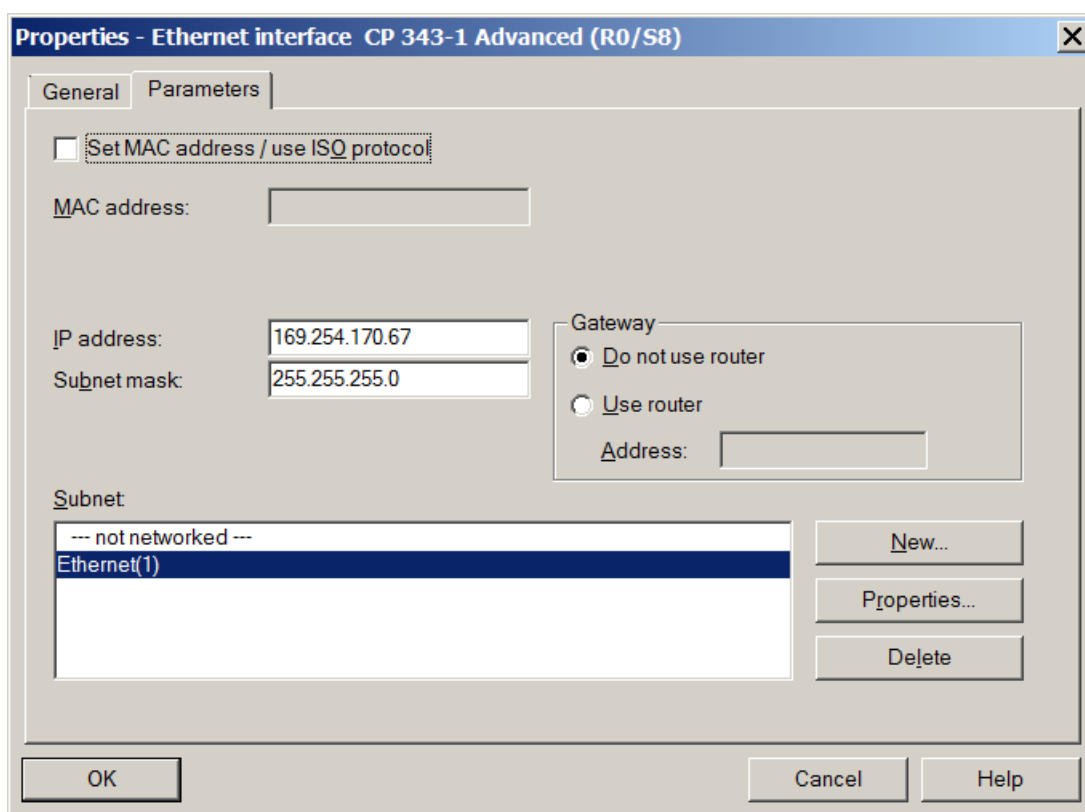
5 VIRTUÁLNÍ ZPROVOZNĚNÍ

Již vytvořený a odsimulovaný model třídící linky v programu Tecnomatix Process Simulate, lze pomocí ethernetové sítě spojit s reálným PLC. Toto PLC pak bude celý model třídící linky řídit a to tak, jak by se řídila skutečná třídící linka.

5.1 Připojení k PLC

Po úspěšném zprovoznění třídící linky v prostředí Tecnomatix Process Simulate, jsem realizoval propojení modelu této třídící linky s fyzickým PLC, neboli virtuální zprovoznění. Ke spojení programu Tecnomatix Process Simulate a PLC slouží OPC server, v této diplomové práci jsem použil program od firmy Siemens Vipa OPC server. Jako komunikační síť jsem zvolil Ethernet, zvaný jako PROFINET.

Nejprve jsem v hardwarové konfiguraci PLC mimo jiné nastavil pro síťový modul CP 343-1 Advanced odpovídající IP adresu PLC, se kterým jsem v laboratoři pracoval.

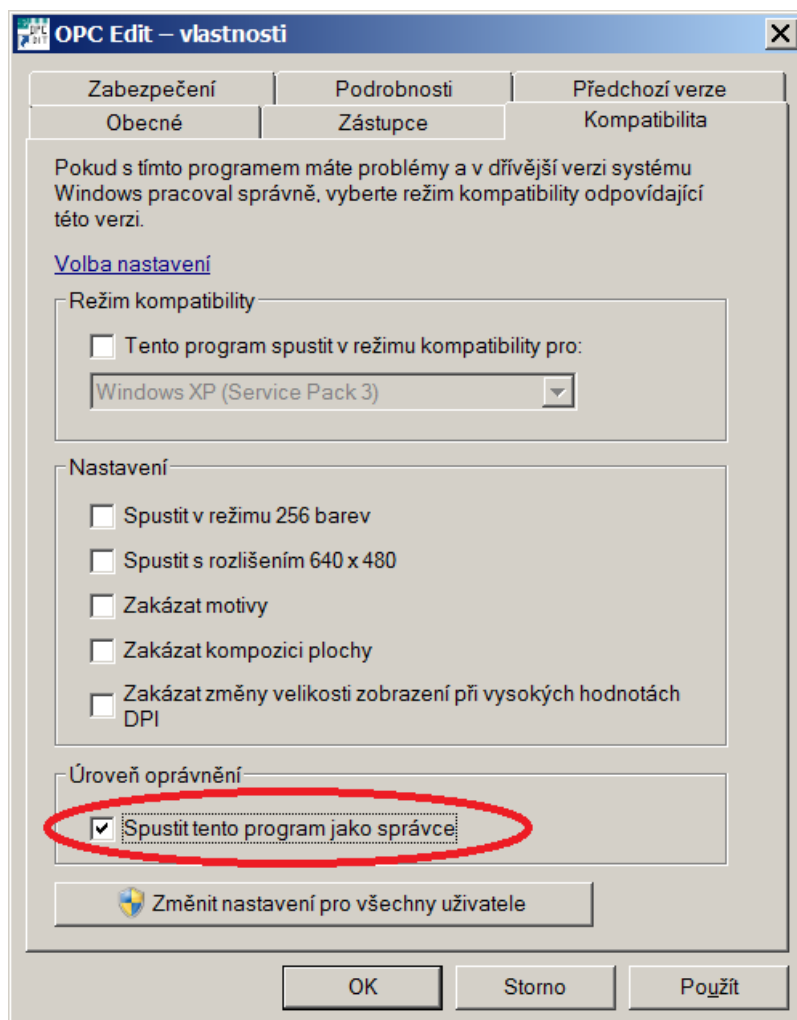


Obrázek 49 Nastavení ethernetové komunikace v PLC

V počítači, na němž se spouští simulace výrobní linky, jsem IP adresu nastavil na 169.254.170.2

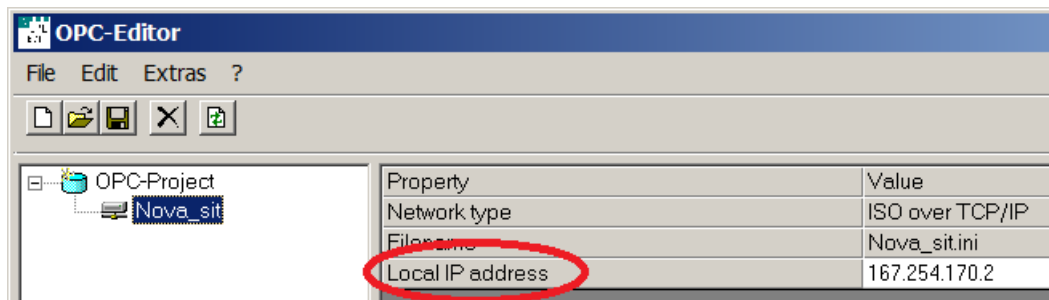
5.1.1 Nastavení OPC serveru

Pro správné fungování OPC serveru je nutné tento program spouštět jako správce. Pravým kliknutím na ikonu vybrat vlastnosti zvolit záložku kompatibilita a zaškrtnout Spustit tento program jako správce.



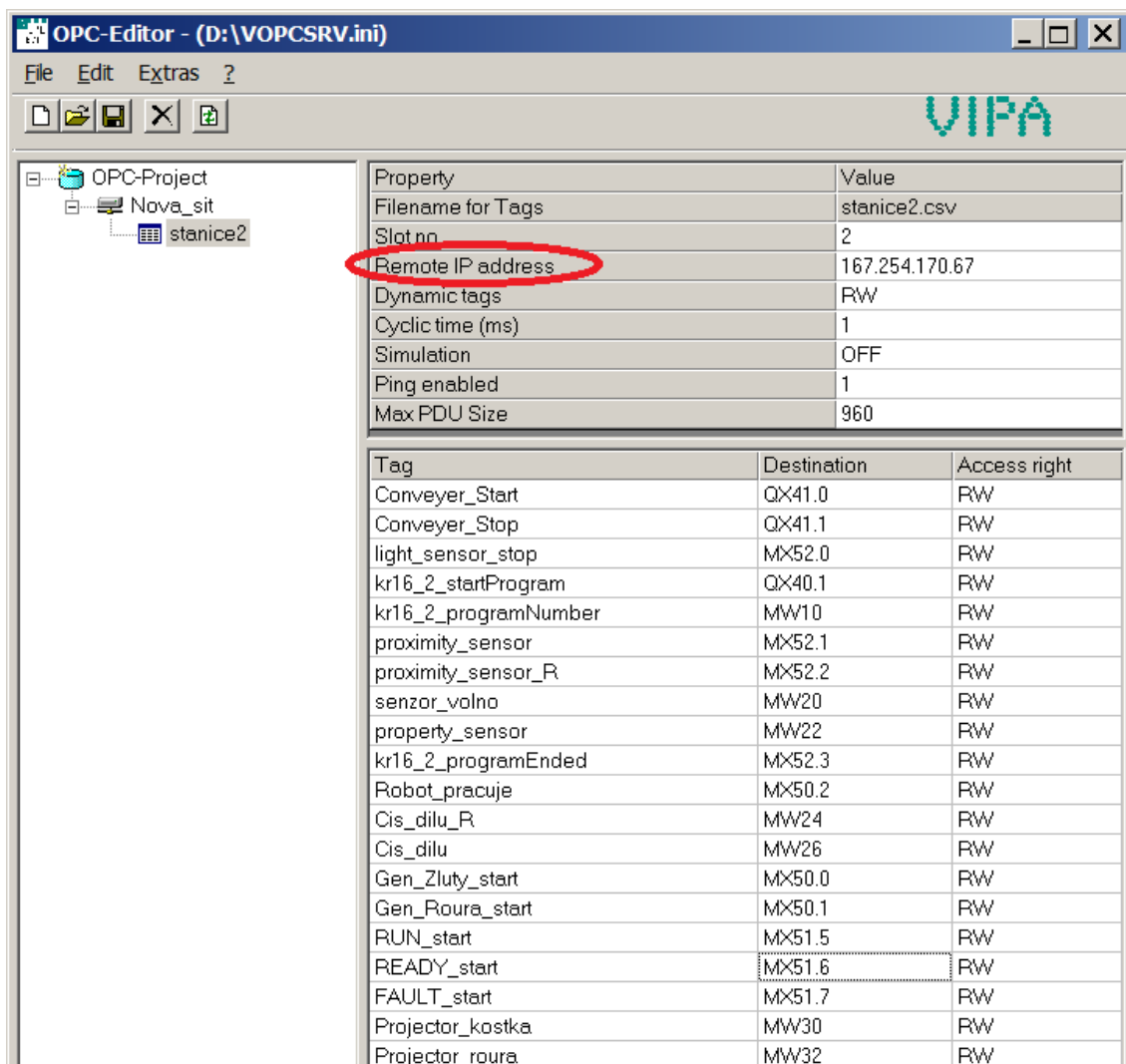
Obrázek 50 Nastavení spouštění OPC serveru jako správce

Po vytvoření nového projektu (File → New project) je potřeba pravým tlačítkem myši kliknout na ikonu OPC-Project a zvolit Add new network. Poté síť pojmenovat a zvolit ISO/Over TCP/IP. Do položky Local IP address je dobré vyplnit IP adresu počítače, na kterém bude spuštěna simulace z programu Tecnomatix Process Simulate.



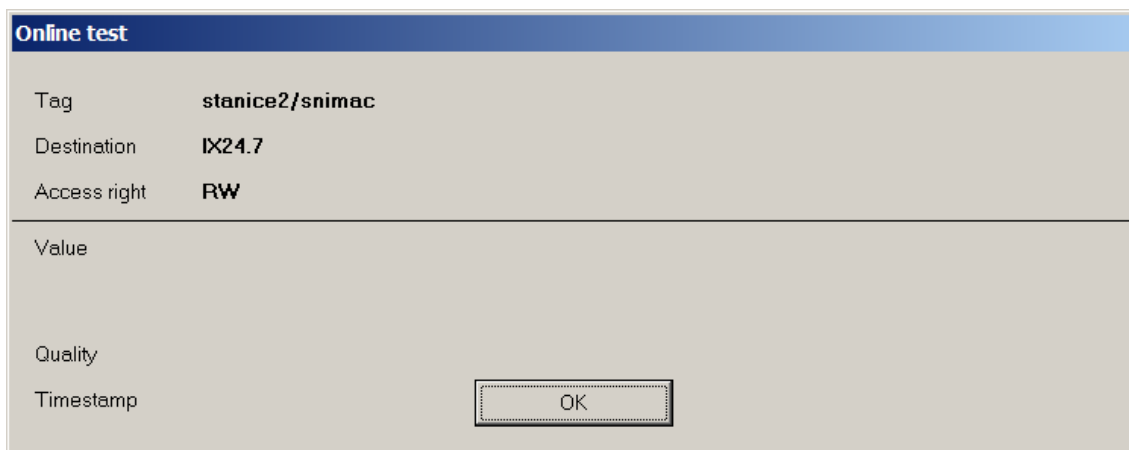
Obrázek 51 Nastavení sítě

Pak se k síti přidá nové PLC a do položky Remote IP address se zadá IP adresa příslušného PLC. Nyní je vhodné projekt uložit, první uložení se musí provést přes Save project as. Uložení projektu se vygeneruje CSV soubor, do kterého lze jednoduše připisovat tagy z programu Tecnomatix Process Simulate.




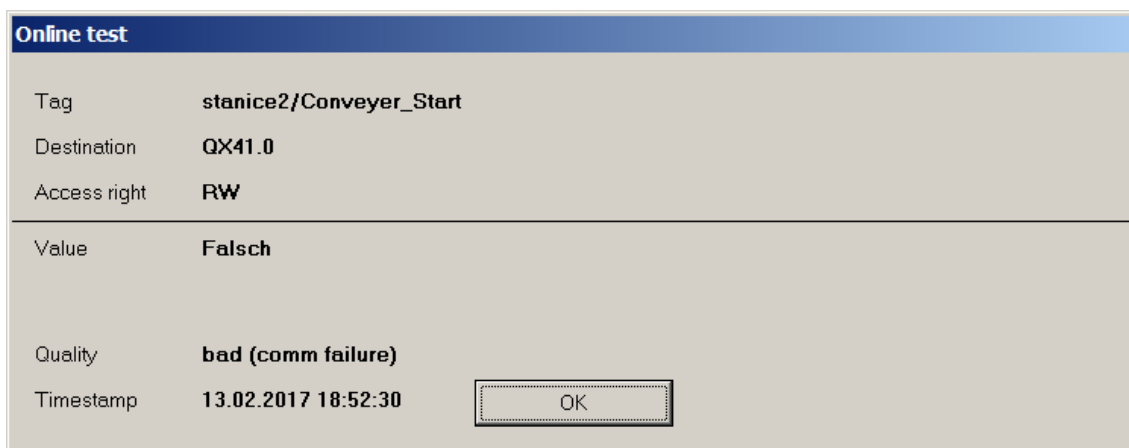
Obrázek 52 Nadefinování tagů

Pokud se nyní provede kontrola hodnoty proměnné (Check online value), lze vidět, že se neobjevují žádné informace, viz Obrázek 53.



Obrázek 53 Chybně nadefinovaná proměnná

Aby se informace o proměnné správně zobrazovaly, je třeba po každé nově přiřazené proměnné spustit rekonfiguraci OPC serveru . Pokud by rekonfigurace nešla spustit, lze tento problém ještě vyřešit tak, že lze přes task manager ukončit proces VOPCSRV.exe a znovu spustit OPC server.

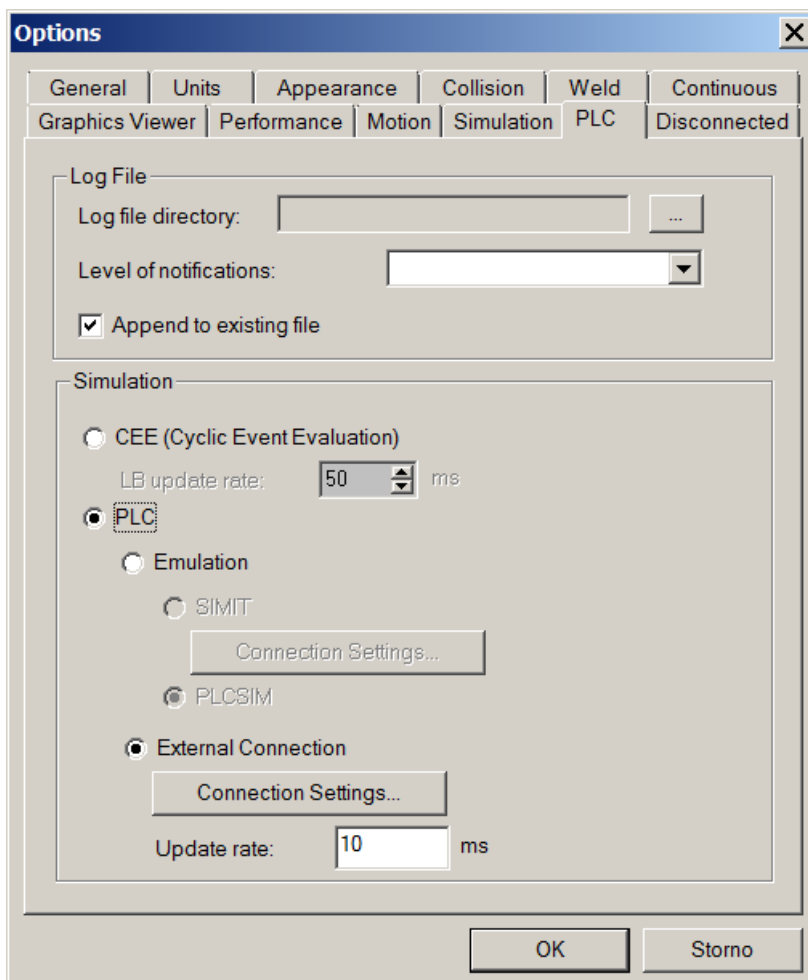


Obrázek 54 Správně nadefinovaná proměnná

Obrázek 54 již zobrazuje správně propojenou proměnou mezi PLC, OPC serverem a programem Tecnomatix Process Simulate. (Položka Quality je bad, protože počítač nebyl připojen do sítě s PLC)

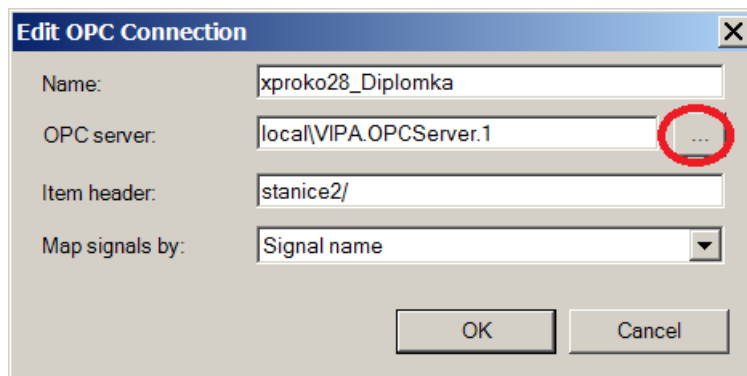
5.1.2 Nastavení fyzického PLC v Tecnomatix Process Simulate

V programu Tecnomatix Process Simulate je potřeba při simulaci, která je řízena událostmi, změnit v nastavení (File Options) v záložce PLC simulaci z cyklického vyhodnocování událostí (CEE) na PLC a dále zvolit Externí připojení.



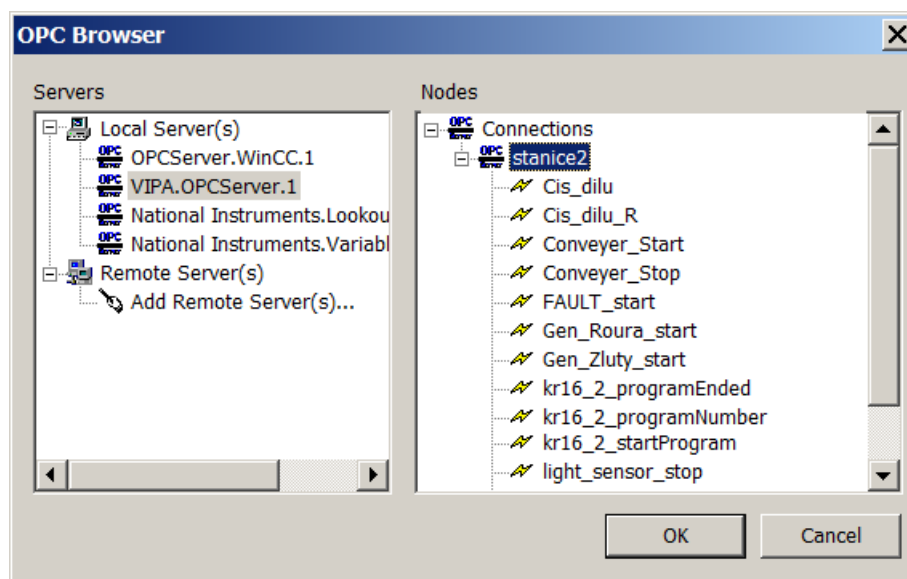
Obrázek 55 Přepnutí na řízení fyzickým PLC

Dále pak je potřeba nastavit připojení (Connection Settings) a vybrat položku add.



Obrázek 56 Přidání OPC připojení

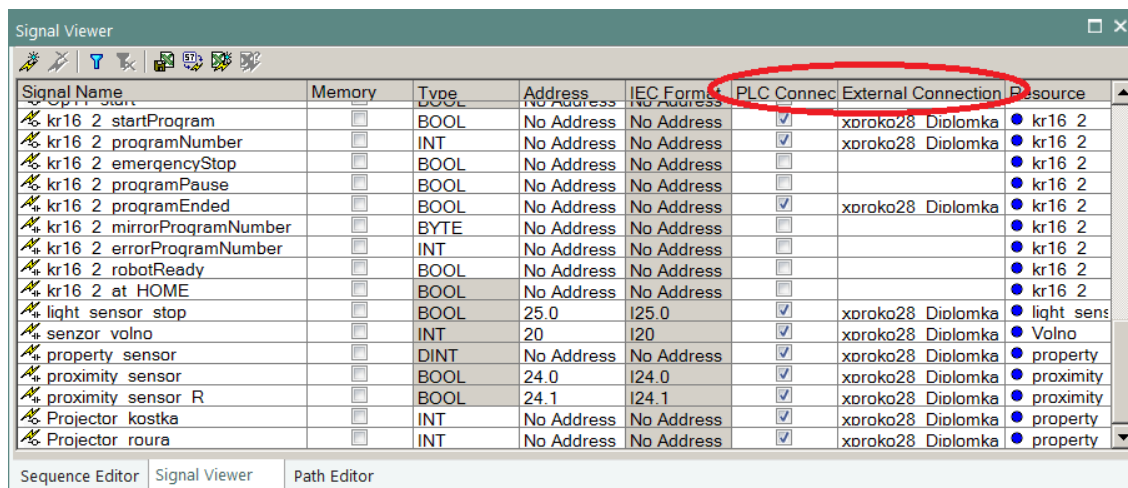
V okně se nejprve vybere nový OPC server.



Obrázek 57 Zvolení OPC serveru

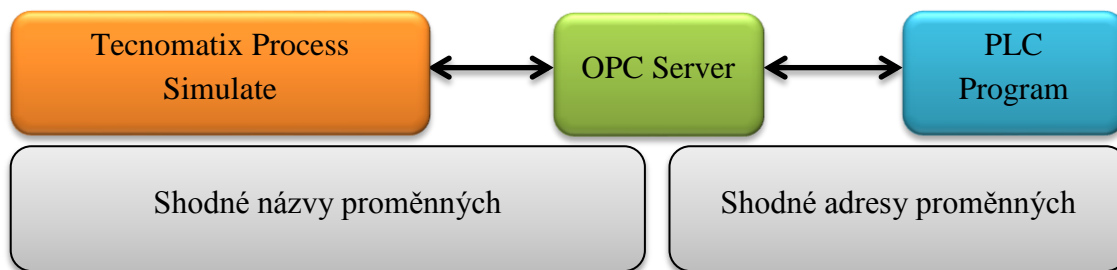
Pro tuto diplomovou práci je to tedy VIPA.OPCServer.1, dále pak se vybere Connections a stanice2. Pokud jsou proměnné v OPC serveru správně nastavené, zobrazují se i zde. Důležité je pak v okně viz Obrázek 56 v položce Item header dopsat ještě „/“, jinak by se při spuštění simulace objevila chybová hláška o špatně definované položce.

Posledním krokem před spuštěním simulace je potřeba ještě nastavit, které signály se mají k PLC připojit. To se nastaví v Signal Vieweru.



Obrázek 58 Zvolení signálu, které se propojí s PLC

Signály, které jsou potřeba k připojení s PLC se zaškrtnou a v položce External Connection se vybere příslušné jméno OPC spojení. Adresy se nemusejí zadávat, musí ovšem odpovídat jméno signálu a jméno tagu v OPC serveru. Pro spojení OPC serveru a PLC programu je nutné mít shodně nastavené adresy proměnných.



Obrázek 59 Schéma propojení proměnných

Dále pak jsem zjistil, že vstupní signály například ze snímačů, aby správně fungovaly, je potřeba nastavit v OPC serveru jako proměnné typu M (markery), nikoliv jako input. Když jsem je nastavil jako klasický fyzický vstup, tak se jejich stav měnil pouze v Tecnomatix Process Simulate, ale v OPC serveru a v PLC programu nikoliv.

5.2 Řízení třídící linky z PLC

Po úspěšném propojení virtuální třídící linky vytvořené v prostředí Tecnomatix Process Simulate a PLC jsem realizoval program pro PLC, který řídí třídící linku totožně jako program, který jsem již vytvořil v prostředí Tecnomatix Process Simulate.

Třídící linka je řízena PLC SIMATIC řady 300. Program jsem vytvářel ve vývojovém prostředí SIMATIC Manager Step7. Dané PLC se skládalo z modulů, jež jsou obsaženy v Tabulka 8.




Název modulu	Popis	Katalogové číslo	Adresy vstupů/výstupů
PS 307 5A	Napájecí zdroj	6ES7 307-1EA00-0AA0	
CPU 315F-2 DP	Procesor	6ES7 315-6FF01-0AB0	
FAI6x13Bit	6 analogových vstupů	6ES7 336-1HE00-0AB0	IW 8 – IW 18
AO4x12Bit	4 analogové výstupy	6ES7 332-5HD01-0AB0	QW 272 – QW 278
FDI24xDC24V	24 digitálních vstupů	6ES7 326-1BK00-0AB0	I 24.0 – I 26.7
FDO10xDC24V/2A	10 digitálních výstupů	6ES7 326-2BF01-0AB0	Q 40.0 – Q 41.1
CP 343-1 Advanced	Komunikační procesor	6GK7 343-1GX21-0XE0	

Tabulka 8 Hardwarová konfigurace PLC

5.2.1 Úpravy v Tecnomatix Process Simulate

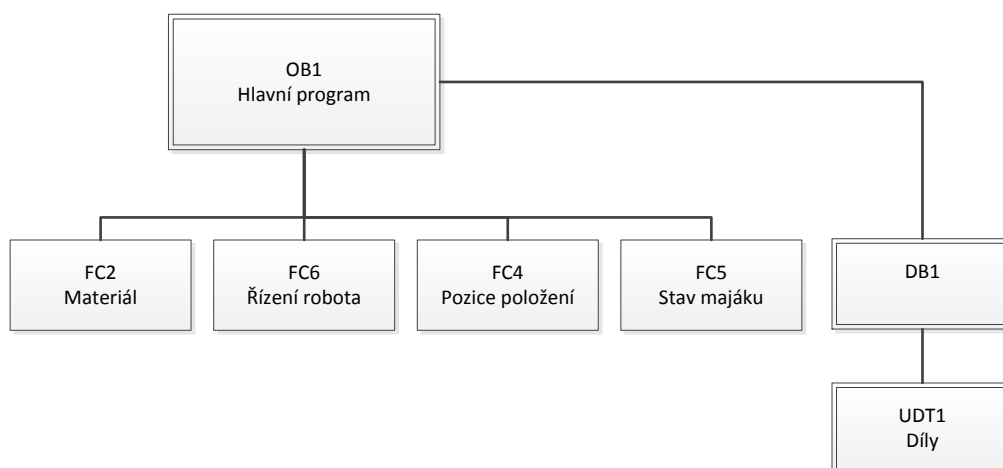
Po přepnutí z interní simulace v Tecnomatix Process Simulate nejsou aktivní moduly, ve kterých se pracovalo se signály, ani nejsou aktivní přechody operací.

Proto jsem musel nejprve signály, které jsou potřebné pro simulaci, ale nehodí se je řídit PLC programem, vyřešit jinak. Signál „Random“, který se měnil funkcí „RANDOM ()“ na hodnotu 1 nebo 2. Jeho hodnotu jsem původně měnil v modulu, ale nyní ji měním v logickém bloku s názvem „LB_signal_random“. Podmínkou pro změnu signálu je, že došlo ke spuštění robota, aby se náhodné číslo vygenerovalo pouze jednou, tak se i detekuje nástupná hrana spouštěcího signálu. Signály „projector_roura“ a „projector_kvadr“, které přiřazují číslo vlastnosti dílům, které se pak detekuje property senzory. Jejich hodnotu jsem taktéž nastavil přes logický blok.

Aby bylo možné aktivovat všechny flow non-sim a device operace, musel jsem vytvořit k nim startovací signály. Startovací signály lze vytvořit v záložce control v kartě Operation signal. Ikonou  se vygenerují startovací signály pro všechny Object Flow operace. Pro Non-Sim a Device operace je potřeba nejdříve danou operaci označit v Operation tree a pak zvolit ikonu pro Non-Sim operaci  a pro Device operaci . Ty jsou v programu Tecnomatix Process Simulate brány jako vstupní signály. Do PLC programu jsem je namapoval jako proměnné typu M (markery). Aby operace proběhla, musí být daný startovací signál po celou dobu operace ve stavu logické jedničky. I když při simulaci z PLC nejsou aktivní přechody a odkazy mezi operacemi, je potřeba stále dodržovat definovaný tok materiálu.

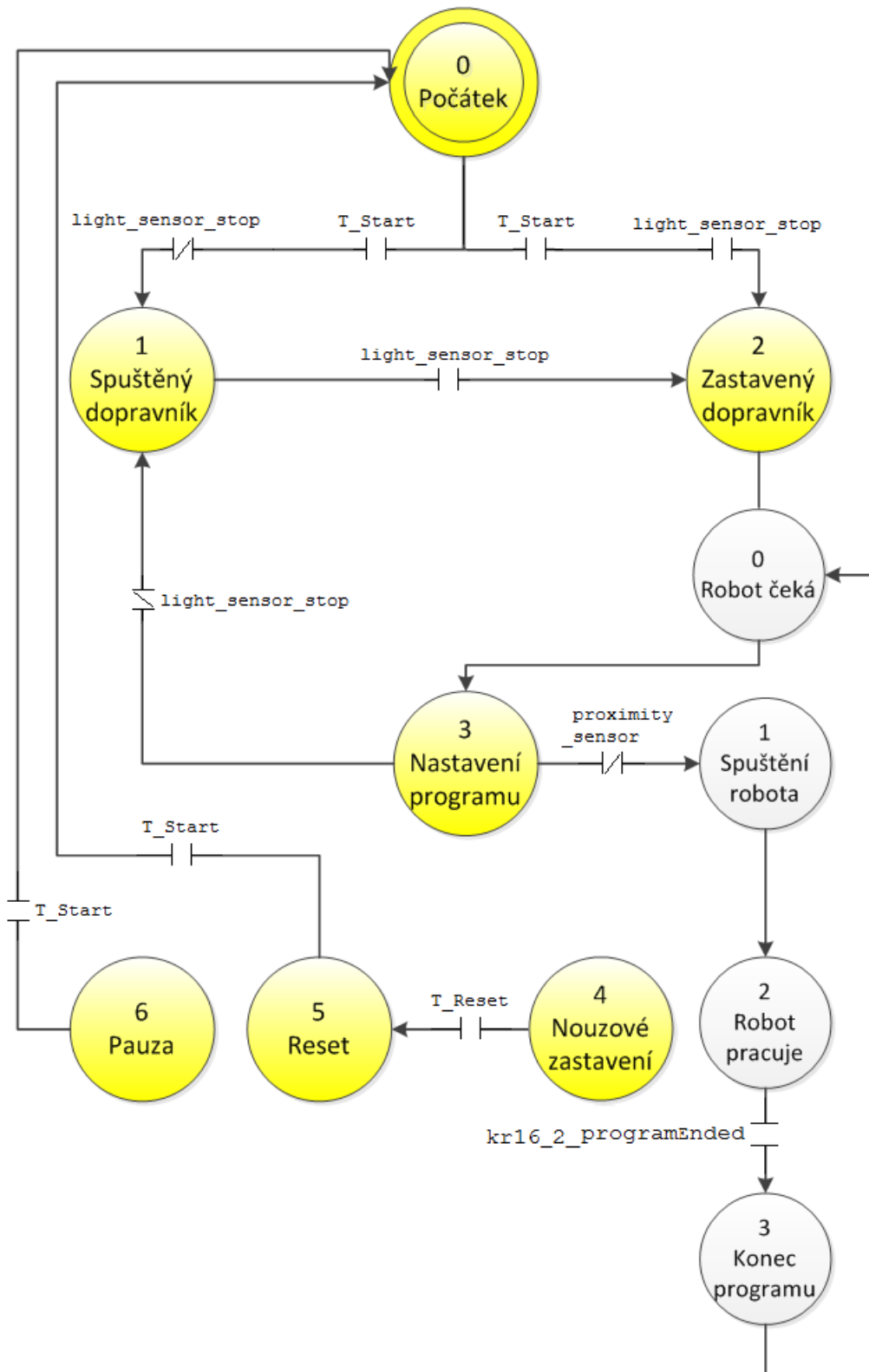
5.2.2 Struktura programu

Hlavní program pro třídící linku je vytvořen v organizačním bloku OB1. V něm jsou pak volány funkce. Pro reset programu se používá organizační blok OB 100.



Obrázek 60 Hierarchie programu

- OB1 obsahuje stavový automat pro proces třídící linky.
- Funkce FC2 dává povel pro příchod nového materiálu na dopravník a taky povel pro vyprázdnění palety po jejím naplnění.
- Funkce FC6 obsahuje stavový automat popisující činnost robota během paletizace.
- Funkce FC4 nuluje čítače, kterými se počítá číslo dalšího položení. Výstup z čítače se ukládá do proměnné typu INT. Touto proměnnou se pak určuje číslo programu pro robota.
- Funkce FC5 spíná světelnou signalizaci na kontrolním majáku.
- Data blok DB1 obsahuje pomocné proměnné pro monitorování stavu robota a proměnné tlačítek ze SCADA aplikace, dále se v něm volá UDT1 pro monitorování stavů dílu „Kvádr“ a dílu „Roura“.



Obrázek 61 Schéma stavů třídicí linky

Stavy 4, 5, a 6 mohou být aktivovány z jakéhokoliv stavu. Tyto stavy se aktivují zmáčknutím příslušného tlačítka v ovládací aplikaci. Stavy vyznačené žlutou barvou

popisují stavy stavového automatu celé třídící linky. Stavů zobrazených šedě jsou stavy podružného stavového automatu popisující chování stacionárního robota. Ze stavu 2 – „Zastavený dopravník“ do stavu 3 – „Nastavení programu“ se přejde, je-li robot ve stavu 0 – „Čeká“.

5.2.3 Proměnné

Proměnné, které jsou používány v PLC programu lze rozdělit do tří kategorií. První kategorií jsou proměnné používané jak v programu PLC, tak i jako signály v Tecnomatix Process Simulate. Jedná se zejména o signály ze snímačů, příkazy pro ovládání robota a dopravního pásu. Jejich vzájemné provázání zajišťuje OPC server. Snímače jsem chtěl původně namapovat jako fyzický vstup do PLC, aby program fungoval jako v reálném případě. Jelikož ale při takovém nastavení se měnil stav proměnné pouze v programu Tecnomatix Process Simulate a nikoliv už v OPC serveru a tudíž ani v PLC programu, musel jsem je změnit na proměnné typu M (markery).

Název	Adresa	Datový typ	Popis
kr16_2_emergencyStop	Q 40.0	BOOL	Nouzové zastavení robota
kr16_2_startProgram	Q 40.1	BOOL	povel pro spuštění robota
kr16_2_programPause	Q 40.2	BOOL	Pozastavení programu robota
box_gripper_open	Q 40.3	BOOL	Otevření uchopovacího zařízení po nouzovém zastavení
Conveyer_Start	Q 41.0	BOOL	povel pro rozjetí dopravníku
Conveyer_Stop	Q 41.1	BOOL	povel pro zastavení dopravníku
light_sensor_stop	M 52.0	BOOL	Snímač detekující díl na konci pásu
proximity_sensor_K	M 52.1	BOOL	Snímač detekující poslední Kvádr na paletě
proximity_sensor_R	M 52.2	BOOL	Snímač detekující poslední Rouru na paletě
proximity_sensor_volno	M 53.2	BOOL	Snímač detekující volný prostor na začátku pásu
senzor_volno	MW 20	INT	Snímač detekující volný prostor na začátku pásu
property_sensor	MW 22	INT	Snímač rozlišující typ dílu
kr16_2_programNumber	MW 10	INT	Číslo spouštěného programu robota
Random	MW 60	INT	Náhodně vygenerované číslo pro určení jaký díl se objeví na pásu
Gen_Kvádr_start	M 50.0	BOOL	Povel pro nový Kvádr na pás
Gen_Roura_start	M 50.1	BOOL	Povel pro novou Rouru na pás
RUN_start	M 51.5	BOOL	Aktivace zelené barvy na majáku
READY_start	M 51.6	BOOL	Aktivace žluté barvy na majáku

FAULT_start	M 51.7	BOOL	Aktivace červené barvy na majáku
kr16_2_programEnded	M 52.3	BOOL	Konec programu robota
vyprazdnit_K	M 53.0	BOOL	Povel pro vyprázdnění palety s „Kvádry“
vyprazdnit_R	M 53.1	BOOL	Povel pro vyprázdnění palety s „Rourami“

Tabulka 9 Společné proměnné

Pro lepší přehlednost jsem v PLC programu použil stejné názvy proměnných, jak je tomu v programu Tecnomatix Process Simulate. Proměnné pro vyprázdnění palet „vyprazdnit_K“ a „vyprazdnit_R“ jsou ve stručnosti spojeny se signály „Op20-23_start“ respektive „Op10-13_start“, které aktivují operce simulující čekání na odvoz palety. Tím, že jsem signály z programu Tecnomatix Process Simulate namapoval na stejnou adresu, jsem zajistil jejich společné spínání. Důvodem, proč se simuluje čekání pro jednotlivé díly na paletě zvlášť je ten, aby byl zachován správný materiálový tok.

Proměnné používané jen v programu PLC jsou deklarovány v datovém bloku DB1. Používají se pro vnitřní logiku programu a monitorují různé stavy třídící linky.

Název	Adresa	Datový typ	Popis
Kvadr.Stav	DB1.DBX 2.0	BOOL	Pokládání „Kvádrů“
Roura.Stav	DB1.DBX 6.0	BOOL	Pokládání „Roury“
Kvadr.Cislo	DB1.DBW 4	INT	Pozice položení pro „Kvádr“
Roura.Cislo	DB1.DBW 8	INT	Pozice položení pro „Rouru“
Stav_linky	DB1.DBW 10	INT	Číslo stavu linky
Stav_robot	DB1.DBW 12	INT	Číslo stavu robota

Tabulka 10 Proměnné používané jen v PLC

Jelikož pro oba díly jsou potřeba stejné pomocné proměnné pro monitorování stavu a počítání pozice pro položení dílu na paletu, rozhodl jsem se vytvořit uživatelský datový blok UDT1, ve kterém tyto proměnné deklarují. Toto řešení by mělo přínos zejména v případě potřeby rozšířit tuto třídící linku o další díly, které by se třídily.

Poslední skupinou proměnných jsou proměnné tlačítek z vizualizace, ty jsou rovněž deklarovány v datovém bloku DB1.

Název	Adresa	Datový typ	Popis
T_Start	DB1.DBX 0.0	BOOL	Start linky
T_Pause	DB1.DBX 0.1	BOOL	Pozastavení linky
T_Reset	DB1.DBX 0.2	BOOL	Reset
E_Stop	DB1.DBX 0.3	BOOL	Nouzové zastavení

Tabulka 11 Markery pro tlačítka z vizualizace

5.2.4 Logika programu

Logika programu vychází z rozvahy řešení, které je popsáno v kapitole číslo 4.2. Výsledná činnost programu v PLC je totožná s programem vytvořeném v programu Tecnomatix Process Simulate. V hlavním organizačním bloku OB1 je řešen algoritmus třídící linky dvěma stavovými automaty. První automat řeší stavy, ve kterých se vyskytuje třídící linka. Druhý automat popisuje stavy vyskytující se při činnosti robota. Zbývající funkce jsou nezávislé na stavech ve stavových automatech. Celkový program napsaný v ladder diagramu lze vidět v příloze číslo 2.

5.2.4.1 Řízení dopravníku

Řízení dopravníkového pásu je řešeno stavy č. 1 a 2. Stav č. 1 zapíná pojezd dopravníkového pásu. Dopravníkový pás je spuštěn, je-li snímač „*ligh_senzor_stop*“ ve stavu logické nuly. Tím se setuje proměnná „*Conveyer_Start*“ a restuje proměnná „*Conveyer_Stop*“.

Druhým stavem se dopravníkový pás zastavuje. K tomu dojde, pokud sepne snímač „*ligh_senzor_stop*“ do logické jedničky. Zde se oproti předešlé větvi setuje proměnná „*Conveyer_Stop*“ a restuje proměnná „*Conveyer_Start*“. Dopravníkový pás se ještě zastavuje při stavu Pauza (6) a při stavu Nouzového zastavení (4). Ty aktivuje operátor, zmáčkne-li tlačítko Pauza nebo E-Stop v ovládací aplikaci.

5.2.4.2 Řízení robota

Stav linky č. 3 aktivuje činnost robota. V tomto stavu se určuje, jestli se bude provádět program pro pokládání dílu typu „Roura“ nebo „Kvádr“. Což záleží na výstupu snímače „*property_sensor*“. Je-li jeho výstup roven jedné, „*property_sensor*“ detekoval, že se jedná o díl typu „Roura“.

Číslo programu („*kr16_2_programNumber*“) se vypočítává ze součtu čísla 10 a čísla pozice („*DB1.roura.cislo*“) pro položení dílu na příslušnou paletu. Číslo programu pro pokládání dílu „Roura“ jsou: 10, 11, 12 a 13. Analogicky pak funguje i druhá větev v dané příčce, pro pokládání dílu typu Kvádr. Díl se rozpozná, je-li výstup ze snímače „*property_sensor*“ roven dvěma. Číslo programů pak jsou 20, 21, 22 a 23.

K přechodu do stavu robota č. 1, v němž se spouští robot nastane, je-li volné místo na paletě. To je v případě, že „*proximity_sensor_R*“ respektive „*proximity_sensor_K*“ je ve stavu logické nuly. Ve stavu robota č. 1 se setuje proměnná „*kr16_2_startProgram*“, čímž nastane pohyb robota podle nastaveného programu. Po té již přechází robot do stavu č. 2, tento stav oznamuje, že robot je již v pohybu. Během tohoto stavu je resetována proměnná „*kr16_2_startProgram*“. Přechod do dalšího stavu nastane až v případě, že do PLC přijde proměnnou „*kr16_2_programEnded*“ informace o dokončení programu. Ve třetím stavu se inkrementuje příslušný čítač, kterým se monitoruje na jakou další pozici, se má další díl

položít. Na závěr se proměnná „stav_robot“, kterou se monitoruje stav robota, nastaví do nuly, což oznamuje, že je robot připraven vykonat další operaci.

5.2.4.3 Práce čítačů

Čítače „Citac_R“ a „Citac_K“ sloužící k počítání pozice položení dílu „Roura“ respektive „Kvádr“ na příslušnou paletu jsou inkrementovány po dokončení programu robota, což je stav robota č. 3. Jestli proběhlo pokládání „Roury“ nebo „Kvádru“ určuje, jestli je proměnná „stav_roura“ respektive „stav_kvadr“ ve stavu logické jedničky. Tyto proměnné se natavují při volbě programu ve stavu linky č. 3. Po inkrementaci daného čítače je daná proměnná resetována.

Ve funkci FC4 se výstupy z čítačů ukládají do proměnných „DBI.roura.cislo“ a „DBI.kvadr.cislo“. Čítač se nuluje, jestli sepnul snímač detekující poslední díl na paletě („proximity_sensor_R“ respektive „proximity_sensor_K“). Pro jistotu jsem přidal i podmínku, pokud by došlo k přetečení čísla určující pozici na paletě.

5.2.4.4 Spínání signalizačních světel

Spínání světel na kontrolním majáku je řešeno ve funkci FC5. Zelená signalizace označující, že linka je v provozu, svítí, pokud je v provozu dopravníkový pás nebo robot.

Žlutá barva signalizuje, že linka je připravena k další práci, ale nepohybuje se dopravníkový pás ani robot. K takovému stavu dojde například, když na konci dopravníku je díl, který patří na již zaplněnou paletu. Aktivace tohoto stavu se řeší ve druhé větvi. Stav „READY“ je aktivní, pokud robot nevykonává žádnou činnost a ani dopravník není v pohybu a na konci dopravníku je díl, který patří na zaplněnou paletu. Poslední podmínka je tam zejména proto, aby nedocházelo ke spínání toho to stavu na velmi malý krátký čas, kdy dopravník je zastaven a robot poté, co dokončil jenu operaci, nezačal ještě novou.

Červená kontrolka značící stav stop, je aktivní v případě nouzově zastavené třídící linky.

5.2.4.5 Produkce a konzumace materiálu

V první větvi funkce FC2 je řešeno generování materiálu na dopravníkový pás. Ke generování dalšího dílu může dojít, jen pokud je začátek dopravníkového pásu volný. To kontrolují snímače „proximity_sensor_volno“ a „sensor_volno“, jenž je typu snímač vlastností. O tom, zda se vygeneruje díl „Roura“ nebo „Kvádr“ rozhoduje proměnná „Random“, její hodnota se určuje v logickém bloku „LB_signal_random“, jenž je vytvořen v programu Tecnomatix Process Simulate. Po startu programu robota se v logickém bloku náhodně vygeneruje buď 0, nebo 1. Pokud je výstupem 0 bude se generovat na dopravníkový pás díl „Kvádr“, je-li výstupem 1, generuje se díl „Roura“.

Druhou činností funkce FC2 je konzumace materiálu, tím se rozumí vyprázdnění palety po jejím zaplnění. Sensory „*proximity_sensor_K*“ a „*proximity_sensor_R*“, detekují přítomnost čtvrtého a tudíž i posledního dílu na paletě. Při sepnutí senzoru se aktivuje povel pro vyprázdnění příslušné palety, tato operace trvá 10 sekund. Dobu lze změnit v programu Tecnomatix Process Simulate, změnou doby trvání Non-Sim operací „*Op10*“ až „*Op13*“ pro díl typu „Roura“ a pro díl typu „Kvadr“ to jsou operace „*Op20*“ až „*Op23*“.

5.2.4.6 Zásah operátora

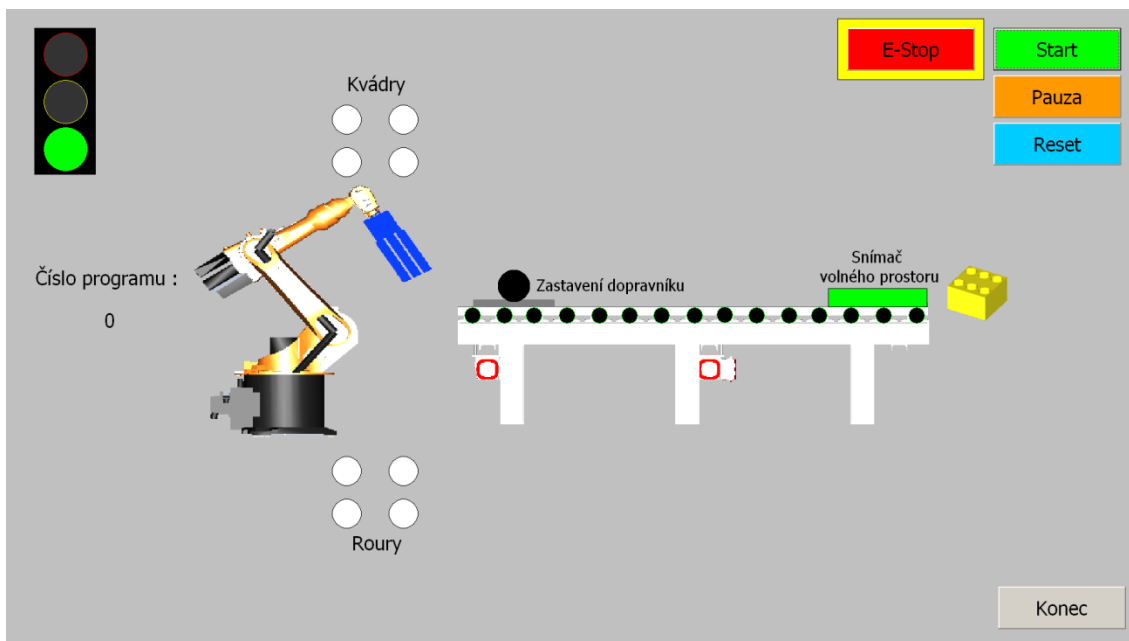
Operátor může tlačítky E-Stop, Reset a Pauza aktivovat stavy Nouzové zastavení (4), Reset (5) a Pauza (6). Při stavu nouzového zastavení se aktivuje proměnná „*kr16_2_emergencyStop*“, čímž se nouzově zastaví robot, dále se zastaví dopravníkový pás. K opětovnému spuštění linky může dojít, až se aktivuje tlačítkem Reset stav pro reset. Při tomto stavu dochází k vyprázdnění palet a robota, čelisti robota se rozevřou, pokud držely díl. Dále jsou vynulovány oba čítače a stav linky i stav robota se nastaví do nuly. Stavem Pauza se pozastaví činnost robota a zastaví se i dopravníkový pás. Jejich spuštění je možné zmáčknutím tlačítka start, při tomto stavu není potřeba aktivovat stav reset.

5.3 Ovládací a vizualizační aplikace

Ovládací aplikaci pro třídící linku jsem vytvořil v programu WinCC flexible 2008. Aplikace zobrazuje stav dopravníku, robota, signalizačního majáku, stavy snímačů, položené díly na paletu, jaký díl přichází na dopravníkový pás a jaký díl detekoval property senzor na konci dopravníkového pásu.

Uživatel v aplikaci může tlačítkem Start spustit třídící linku. Tlačítko Pauza slouží k pozastavení robota. Ke znovu rozjetí robota stačí zmáčknout opět tlačítko start. Tlačítkem E-Stop se aktivuje nouzové zastavení robota, při něm se zastaví i dopravníkový pás. Opětovné spuštění třídící linky je možné až po provedení resetu. To je zajištěno tím, že po zmáčknutí tlačítka E-Stop zmizí z nabídky tlačítko Start.

K opětovnému spuštění třídící linky může dojít až po zmáčknutí tlačítka Reset. Při resetu se vyprázdní obě palety, popřípadě i robot a vynulují se čítače.



Obrázek 62 Ovládací aplikace

K propojení SCADA aplikace a PLC programu se používají tagy, ty jsou jak v programu PLC tak i v programu WinCC namapovány na stejnou adresu. Tyto tagy v aplikaci slouží k ovládní světelné signalizace a jiné pak v programu PLC slouží pro spínání tlačítek. Přehled použitých tagů a jejich význam zobrazuje Tabulka 12.

Název	Adresa	Datový typ	Popis
T_Start	DB 1 DBX 0.1	BOOL	Spuštění linky
T_Pause	DB 1 DBX 0.2	BOOL	Přerušení programu robota
T_Reset	DB 1 DBX 0.3	BOOL	Restartování celé třídící linky
E_Stop	DB 1 DBX 0.4	BOOL	Nouzové zastavení
Cis_dilu_K	DB 1 DBW 4	INT	Zobrazení položených „Kostek“
Cis_dilu_R	DB 1 DBW 8	INT	Zobrazení položených „Rour“
proximity_senzor_K	M 52.1	BOOL	Zobrazení zaplněné palety pro „Kostky“
proximity_senzor_R	M 52.2	BOOL	Zobrazení zaplněné palety pro „Roury“
Gen_kvádr	M 50.0	BOOL	Nová „Kvádr“ na dopravníku
Gen_roura	M 50.1	BOOL	Nová „roura“ na dopravníku
light_sensor_stop	M 52.0	BOOL	Díl na konci pásu
senzor_volno	MW 20	INT	Monitorování stavu na začátku pásu
property_senzor	MW 22	INT	Zobrazení dílu na konci pásu
Robot_pracuje	DB 1 DBX 0.0	BOOL	Zobrazení stavu robota
kr16_2_programNumber	MW 10	INT	Číslo programu robota
Conveyer_Start	Q 41.0	BOOL	Zobrazení stavu dopravníku
RUN_start	M 51.5	BOOL	Zelené světlo na majáku
READY_start	M 51.6	BOOL	Žluté světlo na majáku
FAULT_start	M 51.7	BOOL	Červené světlo na majáku

Tabulka 12 Požívané tagy ve vizualizaci

6 POUŽITÍ VE VÝUCE

Úkolem studentů v laboratorní úloze bude vytvořit řídicí program v pro PLC, jímž se bude ovládat model třídící linky. Model třídící linky vytvořený v programu Tecnomatix Process Simulate budou mít již studenti k dispozici a nebudou jej nikterak upravovat.

Třídící linka bude mít za úkol roztřídit dva typy dílů a uložit vždy čtyři stejné díly na paletu. Tyto díly se na dopravníkový pás generují v nahodilém pořadí. Pokud je paleta plná, robot čeká, až se paleta vyprázdní a začne pokládat novou paletu.

6.1 Namapování proměnných

Proto, aby bylo možné řídit model třídící linky z PLC je nutné, aby signály určené pro externí připojení z programu Tecnomatix Process Simulate měly v OPC serveru shodné názvy a stejné adresy a datový typ jak k nim odpovídající proměnné v PLC programu. Takové nastavení společných proměnných lze vidět Tabulka 9 v kapitole číslo 5.2.3 - Proměnné.

6.2 Rozbor programu

Celkový řídicí program pro PLC lze rozdělit do tří hlavních úseků. Nejprve je třeba vyřešit příchod nových dílů na dopravníkový pás. Poté spuštění robota a volba správného programu pro paletizace dílu. Na závěr se řeší odebrání zaplněné palety.

Kromě samotného procesu třídění se ještě řeší světelná signalizace a nouzové zatavení.

6.2.1 Nový díl na dopravník

Nový díl na dopravníkový pás může být vložen v případě, že se na začátku dopravníkového pásu nenachází žádný díl. To lze kontrolovat snímačem „*senzor_volno*“, jehož výstup je roven nule v případě volného prostoru. Volný prostor lze taktéž kontrolovat snímačem „*proximity_senzor_volno*“, jehož výstup je v logické nule v případě volného prostoru.

Náhodnost, jaký díl se objeví na dopravníku, určuje proměnná „*Random*“, jeho hodnota je 0 nebo 1. Například když proměnná „*Random*“ je 0, objeví se na dopravníku díl „Kvádr“, a když je proměnná „*Random*“ rovna jedné objeví se díl „Roura“.

6.2.2 Paletizace

Pro zapínání a vypínání dopravníkového pásu jsou určeny dvě výstupní proměnné. Pro spuštění je to „*Conveyer_Start*“ a pro vypnutí „*Conveyer_Stop*“. Na konci dopravníkového pásu je fotoelektrický senzor „*light_sensor_stop*“, ten je určen

pro zastavení dopravníkového pásu, když jeho parsek protne díl na dopravníkovém pásu, čím přejde ze stavu logické nuly do stavu logické jedničky.

Dalším snímačem na konci dopravníkového pásu je „*property_senzor*“, ten je určen k rozlišení typu dílu. Pokud se jedná o díl „Roura“, tak je jeho výstup roven jedné a když je to díl „Kvadr“, tak je výstup ze snímače roven dvěma.

Pro spuštění robota je výstupní proměnná „*kr16_2_startProgram*“ v logické nule. Při ukončení práce robota se aktivuje do logické jedničky proměnná „*kr16_2_programEnded*“. Jakou operaci robot vykoná, rozhoduje proměnná „*kr16_2_programNumber*“. Pro pokládání dílu „Roura“ nabývá proměnná „*kr16_2_programNumber*“ hodnot 10 až 13. Pro položení na první pozici je to číslo 10, pro položení na poslední pozici je to číslo 13. Pro díl „Kvadr“ to jsou čísla 20 až 23.

Robot nemůže zahájit novou operaci, pokud je příslušná paleta již zaplněna, to kontrolují senzory „*proximity_sensor_R*“ pro paletu s „Rourami“ a pro paletu s „Kvadrmi“ „*proximity_sensor_K*“. Je-li paleta zaplněna, je potřeba aktivovat povel na její vyprázdnění. Povel pro díl „Roura“ se nazývá „*vyprazdnit_R*“ a pro díl „Kvadr“ to je povel „*vyprazdnit_K*“.

Pro určení čísla programu je dobré vytvořit si čítače, které počítají na jakou další pozici, se má další díl položit. Dále je vhodné si monitorovat, zda již robot nevykonává nějakou operaci, aby nedošlo povelom „*kr16_2_startProgram*“ spuštění další operace, přitom ještě robot nedokončil předešlou operaci.

6.2.3 Světelná signalizace a nouzové zastavení

U třídící linky je světelný maják, který signalizuje, v jakém stavu se nachází linka. Zelená barva značí, že linka je v provozu. Aktivuje se povelom „*RUN_start*“. Žlutá barva se aktivuje povelom „*READY_start*“ a značí, že je linka pozastavena, ale může se rozběhnout. To je například, když je paleta plná a další díl na konci dopravníku tudíž nelze nikam položit, nebo když se robot zastavil příkazem „*kr16_2_programPause*“. Červená barva se aktivuje povelom „*FAULT_start*“ a značí nouzový stav. Ten nastává po aktivaci „*kr16_2_emergencyStop*“.

Po aktivaci „*kr16_2_emergencyStop*“ se robot nouzově zastaví, dále pak je potřeba ještě aktivovat proměnnou „*Conveyer_Stop*“ k zastavení dopravníku. K opětovnému spuštění třídící linky může dojít až po úplném restartování třídící linky.

7 ZÁVĚR

V počátku své diplomové práce jsem si nejprve z programové dokumentace nastudoval, jak pracovat ve vývojovém prostředí Tecnomatix Process Simulate. Tyto poznatky jsem pak v jednotlivých podkapitolách zaznamenal a sepsal manuál pro toto vývojové prostředí. V něm popisují, jak nastavit robota a vytvořit program pro robota, základní práci s materiálem, jako je jeho vymodelování, manipulace a generování toku materiálu. Dále jsem popsal, jak nadefinovat dopravníkový pás, jak fungují snímače a jak celý proces výrobní linky odsimulovat v programu Tecnomatix Process Simulate. Tyto informace jsem nabyl zejména z programové dokumentace, popřípadě jsem komunikoval se supportem od firmy AXIOM TECH, a to konkrétně s panem doktorem Davidem Sámkem, kterému patří mé poděkování.

Na základě těchto poznatků jsem pak zhotovil virtuální model třídící linky, kde stacionární robot třídí dva odlišné díly, které jedou na dopravníkovém pásu. Tyto díly se generují náhodně, a proto je snímačem rozlišují, aby je pak robot umístil na správnou paletu.

Tento model třídící linky jsem následně připojil přes VIPA OPC server k PLC SIMATIC 300. Pro toto PLC jsem naprogramoval program, kterým jsem tuto třídící linku řídil. Tento program je pro přehlednost řešen jako stavový automat, jež popisuje jednotlivé stavy, ve kterých se třídící linka může nacházet. Takovéto realizaci se říká virtuální zprovoznění.

Dále jsem v programu WinCC flexible 2008 vytvořil jednoduchou SCADA aplikaci, ve které se operátorovi zobrazuje, zda se pás pohybuje, jestli robot pokládá předmět nebo stojí, anebo je nouzově zastaven. Aplikace dále zobrazuje počet dílů na paletě a také stavy jednotlivých snímačů. Operátor pak může tlačítky spustit celou třídící linku, nebo ji pozastavit, anebo nouzově zastavit a celý proces třídění vyresetovat.

Cílem této diplomové práce bylo vytvoření modelu třídící linky, který se bude používat v laboratorních úlohách, ve kterých budou studenti vytvářet řídicí PLC program pro tento model třídící linky. PLC program a SCADA aplikaci jsem v této diplomové práci řádně zdokumentoval a otestoval, popsal jsem detailně smysl všech použitých proměnných. Tento popis pak studentům srozumitelně objasňuje, který signál z programu Tecnomatix Process Simulate k čemu slouží. V kapitole číslo 6 je popsána problematika řešení dané úlohy.

Použitá Literatura

- [1] Digitální továrna. *AXIOM TECH* [online]. Zlín: AXIOM TECH, 2016 [cit. 2017-02-26]. Dostupné z: <http://www.axiomtech.cz/24751-digitalni-tovarna-tecnomatix>
- [2] AXIOM TECH S. R. O. Virtual Commissioning pomáhá navrhovat robotické linky. *Automa* [online]. Děčín: Automa-časopis pro automatizační techniku, 2016, **21**(5), 2 [cit. 2017-02-26]. ISSN 1210-9592. Dostupné z: http://automa.cz/Aton/FileRepository/pdf_articles/54805.pdf
- [3] Virtuální zprovoznování strojů s pomocí Process Simulate. *Konstruktér* [online]. Brno: Nová média, 2015 [cit. 2017-02-27]. Dostupné z: <http://www.konstrukter.cz/2015/01/30/virtualni-zprovoznovani-stroju-s-pomoci-process-simulate/>
- [4] Digitální podnik: přínosy. *Digital factory* [online]. Plzeň: Západočeská univerzita, 2017 [cit. 2017-04-28]. Dostupné z: <https://www.digipod.zcu.cz/index.php/digitalni-tovarna/prinosy>
- [5] Siemens PLM Software. *Objevte Tecnomatix (Digitální továrnu)* [online]. SRN: SIEMENS, 2016 [cit. 2016-12-22]. Dostupné z: http://www.plm.automation.siemens.com/cz_cz/products/tecnomatix/index.shtml
- [6] Tecnomatix Process Simulate. *Digital Factory* [online]. Plzeň: Západočeská univerzita v Plzni, 2013 [cit. 2016-12-22]. Dostupné z: <http://digipod.zcu.cz/index.php/cs/oblasti-nasazeni/robotika/tecnomatix-process-simulate>
- [7] KONOPA, Miroslav. *Simulation of Production Processes*. Praha, 2013. Diplomová práce. ČVUT. Vedoucí práce Ing. Pavel Burget, Ph.D.
- [8] ŠIMEK, Pavel. *Simulace výrobních linek*. Praha, 2015. Diplomová práce. ČVUT. Vedoucí práce Ing. Pavel Burget, Ph.D.
- [9] Tecnomatix eMServer Platform 13.0 Help. *PLM Automation* [online]. SRN: SIEMENS, 2016 [cit. 2016-11-29]. Dostupné z: https://docs.plm.automation.siemens.com/tdoc/tecnomatix/13/tecnomatix_eMS/#uid:index

Seznam obrázků

Obrázek 1 Zobrazení ušetření peněz a času [4]	12
Obrázek 2 Vývojové prostředí Tecnomatix Process Simulate	13
Obrázek 3 Schéma virtuálního zprovoznění	15
Obrázek 4 Nastavení řízení robota.....	16
Obrázek 5 Upevnění uchopovacího zařízení na robota	17
Obrázek 6 Nastavení posunu čelistí.....	17
Obrázek 7 Zadání cest robotu	18
Obrázek 8 Signály robota	18
Obrázek 9 Nastavení rozměrů objektu.....	19
Obrázek 10 Postup modelování dílu	20
Obrázek 11 Provedení sloučení	20
Obrázek 12 Provedení rozdílu	21
Obrázek 13 Provedení průniku	21
Obrázek 14 Rozkopírování objektu	22
Obrázek 15 Nastavení pozice	23
Obrázek 16 Vytvoření Appearance.....	24
Obrázek 17 Definování toku materiálu.....	24
Obrázek 18 Tok materiálu	25
Obrázek 19 Nová cesta pro materiál.....	25
Obrázek 20 Vytvoření dopravníku	26
Obrázek 21 Vykreslení křivky kolem dopravníku.....	26
Obrázek 22 Nastavení parametrů dopravníku	27
Obrázek 23 Kontrolní body	28
Obrázek 24 Pohyb krabice po dopravníku.....	29
Obrázek 25 Nastavení signálů	29
Obrázek 26 Nastavení parametrů snímače.....	30
Obrázek 27 Definování vlastností.....	31
Obrázek 28 Nastavení snímače přiřazující vlastnosti objektům.....	32
Obrázek 29 Nastavení snímače vlastností	32
Obrázek 30 Nastavení snímače přiblížení	33
Obrázek 31 Nastavení senzoru detekující dosažení polohy.....	33
Obrázek 32 Nastavení snímače monitorujícího polohu.....	34
Obrázek 33 Nastavení signál v modulu	35
Obrázek 34 Sekvence operací.....	36
Obrázek 35 Větvení operací	37
Obrázek 36 Přejít na druhou operaci	37
Obrázek 37 Nastavení operace zařízení.....	38
Obrázek 38 Model linky	39

Obrázek 39 Vývojový diagram třídící linky	40
Obrázek 40 Díl „Roura“ a díl „Kvádr“	41
Obrázek 41 Struktura operací pro generování materiálu	43
Obrázek 42 Struktura operací pro pokládání dílů	44
Obrázek 43 Struktura operací pro maják	45
Obrázek 44 Tok materiálu dílu „Roura“	45
Obrázek 45 Tok materiálu dílu „Kvádr“	46
Obrázek 46 Hierarchie modulů pro práci s díly	48
Obrázek 47 Hierarchie modulů pro řízení třídící linky.....	49
Obrázek 48 Rozmístění snímačů	50
Obrázek 49 Nastavení ethernetové komunikace v PLC	52
Obrázek 50 Nastavení spuštění OPC serveru jako správce	53
Obrázek 51 Nastavení sítě	54
Obrázek 52 Nadefinování tagů	54
Obrázek 53 Chybně nadefinovaná proměnná	55
Obrázek 54 Správně nadefinovaná proměnná	55
Obrázek 55 Přepnutí na řízení fyzickým PLC	56
Obrázek 56 Přidání OPC připojení	56
Obrázek 57 Zvolení OPC serveru	57
Obrázek 58 Zvolení signálu, které se propojí s PLC	57
Obrázek 59 Schéma propojení proměnných	58
Obrázek 60 Hierarchie programu	59
Obrázek 61 Schéma stavů třídící linky	61
Obrázek 62 Ovládací aplikace	67

Seznam tabulek

Tabulka 1 Přínos digitální továrny v automobilovém průmyslu [4].....	12
Tabulka 2 Vstupní signály	46
Tabulka 3 Výstupní signály	46
Tabulka 4 Displej signály	47
Tabulka 5 Key signály	47
Tabulka 6 Vytvořené moduly	47
Tabulka 7 Snímače.....	50
Tabulka 8 Hardwarová konfigurace PLC	58
Tabulka 9 Společné proměnné.....	63
Tabulka 10 Proměnné požívané jen v PLC	63
Tabulka 11 Merkery pro tlačítka z vizualizace.....	63
Tabulka 12 Požívané tagy ve vizualizaci.....	68

Seznam příloh

- 1 DVD
 - 1.1 Elektronická verze diplomové práce
 - 1.2 Model třídící linky vytvořený v programu Tecnomatix Process Simulate
 - 1.3 Video manuál pro práci v Tecnomatix Process Simulate
 - 1.4 Program pro PLC
 - 1.5 SCADA aplikace vytvořená v programu WinCC flexible 2008
 - 1.6 Výpis programu pro PLC