

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

BAKALÁŘSKÁ PRÁCE

Návrh systému pro nadřazené řízení
kolaborativních robotů

Plzeň, 2018

Adam Frémund

PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne

.....

Poděkování

Tímto bych chtěl poděkovat panu Ing. Martinu Goubejovi, Ph.D. za skvělé vedení práce a ochotu při poskytování cenných rad a věcných připomínek, které vedly k úspěšnému vypracování této práce.

Abstrakt

Tato práce se zabývá návrhem řídicího systému, který uživateli umožní programování v relativním souřadném systému obrobku. Před uvedením do problému jsou na začátku práce shrnuty charakteristické rysy průmyslových robotů a jejich dělení. Dále jsou popsána možná nebezpečí, která vznikají při používání průmyslových robotů a také jednotlivá řešení, jak zaručit bezpečnost robotických systémů. Další část se věnuje kolaborativní robotice a srovnání s běžně používanými průmyslovými roboty.

V praktické části je řešen hlavní cíl této bakalářské práce. Návrh nadřazeného řídicího systému byl prováděn v systému REXYGEN. Komunikace mezi řídicím systémem a kolaborativním robotem byla zajištěna pomocí komunikačního protokolu Modbus. Samotné ovládání robotu bylo programováno v grafickém rozhraní PolyScope.

Klíčová slova

průmyslové roboty, koboty, bezpečnost robotů, ISO normy, eulerovy úhly, UR3, PolyScope, Modbus, REXYGEN

Abstract

This thesis deals with the design of a control system, which allows the user to program robot in relative coordinate system of the workpiece. At the beginning of the thesis are summarized characteristic features of industrial robots and their division. Then the thesis describes the potential hazards that arise when using industrial robots, as well as individual solutions to guarantee the safety of robotic systems. Next part deals only with collaborative robotics and comparison with commonly used industrial robots.

The main aim of this thesis is solved in practical part. The design of the superior control system was performed in the REXYGEN system. Communication between the control system and the collaborative robot was ensured using the Modbus communication protocol. The control sequences were programmed in the graphical interface PolyScope.

Key words

industrial robots, cobots, safety of robot systems, ISO standards, euler angles, UR3, PolyScope, Modbus, REXYGEN

Obsah

1	Úvod	9
2	Rešerše bezpečnosti průmyslových a kolaborativních robotů	10
1	Průmyslové roboty	10
1.1	Historie průmyslové robotiky	10
1.2	Roboty a průmyslové roboty	10
1.3	Rozdělení robotů	11
1.3.1	Podle velikosti a tvaru pracovního prostoru	11
1.3.2	Dělení na základě typu řídicích systémů	13
1.3.3	Dělení na základě generování trajektorie pohybu	13
1.4	Jednotlivé části průmyslových robotů	14
2	Bezpečnost robotů	15
2.1	Úvod do problému bezpečnosti robotů	15
2.2	Nehody způsobené robotem	15
2.3	Zdroje nebezpečí způsobující nehody	16
2.4	Požadavky na bezpečnost robota	17
2.4.1	Požadavky a bezpečnostní opatření při běžném provozu	17
2.4.2	Požadavky a bezpečnostní opatření v průběhu speciálních provozních režimů	17
2.5	Zabezpečení robotů	18
2.5.1	Technická ochranná opatření	18
3	Kolaborativní robotika	21
3.1	Mezinárodní normy ISO	21
3.1.1	Technická specifikace ISO/TS 15066	21
3.2	Posuzování rizika	24
3.3	Řešení jednotlivých výrobců	25
4	Reprezentace orientace tělesa v prostoru	28
4.1	Eulerovy úhly	29
4.2	Kvaterniony	30
4.2.1	Praktické využití kvaternionů pro popis rotace v \mathbb{R}^3	31
3	Návrh komunikace a ovládání robotu	32
1	Popis řešeného problému	32
2	Seznámení s kolaborativním robotem UR3	33
2.1	Grafické rozhraní PolyScope	33
2.1.1	Inicializace robotu	33
2.1.2	Úvodní obrazovka	33
2.1.3	Spustit program	33
2.1.4	Obrazovka nový program	34
3	Návrh komunikačního rozhraní	37
3.1	Nastavení komunikace na straně robotu	37
3.2	Nastavení komunikace v řídicím systému REXYGEN	38

	3.2.1	Popis jednotlivých bloků v souboru robot_final	38
	3.2.2	Popis jednotlivých bloků v souboru exec_final	40
4		Implementace programu pro ovládání robotu	42
	4.1	Nastavení rovin v instalaci robotu	42
	4.2	Struktura programu	43
	4.2.1	Nastavení počátečních proměnných	44
	4.2.2	Akce provedené před zahájením programu	44
	4.2.3	Program robota	45
5		Testování vytvořených programů na reálném robotu	46
	5.1	Změny v řídicím systému REXYGEN	47
	5.2	Změny v grafickém prostředí PolyScope	48
	5.3	Pracoviště	48
	5.4	Testování	49
4		Závěr	51

Seznam obrázků

1	Kartézský robot [25]	11
2	Robot s válcovitým pracovním prostorem [25]	11
3	Sférický robot [25]	12
4	Scara robot [25]	12
5	Delta robot [25]	12
6	Ukázka reálného kloubového robotu určeného ke svařování [23]	13
7	Ukázka reálného robotu typu snake arm sloužícího k zacházení s jaderným odpadem [26]	13
8	Čtyři hlavní komponenty průmyslového robotu [25]	14
1	Efektivně řešený bezpečnostní systém [25]	15
2	Ochranné ploty v praxi [25]	18
3	Ochranný kryt lisu [25]	19
4	Světelné závěsy [25]	19
5	Možnost zabezpečení přístupového místa pomocí laserového bezpečnostního skeneru [25]	20
6	Nášlapná rohož - žlutá deska kolem robotu [25]	20
7	Pracovní prostor [34]	22
8	Robot používaný v módu Hand guiding [35]	23
9	Ukázka možnosti řešení módu Speed and separation monitoring [34]	23
10	Opatření pro zajištění bezpečnosti pracovišť s kolaborativními roboty [28]	25
11	Robot YuMi [27]	25
12	Robot Cobotta [29]	26
13	Roboty společnosti Universal Robots [30]	26
14	Robot Techman TM5 [37]	27
15	Vlevo - souřadnicový systém robotu, vpravo - nový posunutý souřadnicový systém [24]	28
16	Přecsní úhel [32]	29
17	Nutační úhel [32]	29
18	Rotační úhel [32]	30
19	Rotace reprezentována Eulerovými osami a úhlem [36]	31
29	Souřadné systémy [33]	32
30	Karta Pohyb	34
31	Náhled grafického rozhraní PolyScope	36
32	Příklady implementace komunikace Modbus [31]	37
33	Nastavení klienta Modbus	38
34	Ukázka řešení problému reálných čísel (souřadnice x)	39
35	Subsystem	39
36	Zjednodušený popis mapování reálného čísla	40
37	Nastavení bloku MTM	41
38	Konfigurační dialog	41
39	Vytvoření roviny v instalaci robotu	43

40	Ukázka struktury algoritmu	44
41	Nastavení hodnoty počátečních proměnných	44
42	Kód skriptu	45
43	Podprogram Creating_planes	46
44	Podprogram Robots_program	46
45	Diagram popisující celý proces potřebný pro programování robotu v rela- tivním souřadném systému obrobku	47
46	Blok pro převod souřadnic	47
47	Konfigurace ovladače Modbus master	48
48	Ukázka markeru, který je rozpoznáván kamerou v prostoru	48
49	Ukázka pracoviště	49
50	Algoritmus pro kontrolu změn v osách x,y,z	50

Úvod

Robotika je věda, která se zabývá studiem, návrhem, konstrukcí a aplikací všech robotických systémů. V dnešní době, kdy je automatizace jednou z nejvíce řešených otázek v oblasti průmyslu, již průmyslové roboty nahrazují lidskou práci. Technický pokrok v této oblasti dává stále novější a lepší technologie, díky kterým lze jednodušeji provázat vnější prostředí a řídicí systémy. Současné technologie poskytují využití různých čidel a senzorů. Díky tomu dnešní roboty dokáží vykonávat práci s mnohonásobně větší přesností a spolehlivostí než člověk.

Práce je rozdělena na dvě hlavní části. První část je především teoretická. Je zde nastíněna historie průmyslové robotiky a uvedení do problematiky bezpečnosti průmyslových robotů. V další kapitole je detailní popis robotických systémů a jejich dělení. Poté se práce věnuje bezpečnosti robotických pracovišť. Další část práce je zaměřena na kolaborativní roboty a jejich využití. Poslední kapitola teoretické části je věnována reprezentaci orientace tělesa v prostoru.

Hlavním cílem této práce bylo navrhnout systém pro nadřazené řízení kolaborativních robotů. Především šlo o návrh systému, který umožňuje programování jednotlivých pohybů robotu v libovolném relativním systému souřadnic obrobku. Nadřazený systém byl navržen v systému REXYGEN. Souřadnice obrobku poté byly získány pomocí kamery, jejíž implementace však nebyla součástí této bakalářské práce.

Rešerše bezpečnosti průmyslových a kolaborativních robotů

1 Průmyslové roboty

1.1 Historie průmyslové robotiky

Umělí pomocníci a společníci mají již dlouhou historii, avšak první plně automatizovaný stroj byl zkonstruován až v 19.století.

Jako první se v oblasti robotiky objevilo jméno George Devola, který roku 1954 podal první patent. Společnost Unimation, kterou vlastnil, vyrobila toho roku prvního průmyslového robota. Do průmyslu však byl nasazen až v roce 1961. Tyto roboty prováděly pouze jednoduché úkoly, např. přenášení věcí z místa na místo. Dlouhou dobu Unimation neměl konkurenci. To se změnilo na konci 70.let, kdy do robotiky vstoupilo několik velkých japonských společností. V Japonsku žádné americké patenty neuznávali, tudíž mohli vyrábět podobné roboty, jako ve společnosti Unimation. Roku 1978 profesor Hiroshi Makino z japonské University of Yamanashi vyvinul 3-osého robota s označením SCARA (Selective Compliance Assembly Robot Arm). Na trhu se taktéž objevila dnes již známá firma KUKA, která přišla se svým robotem složeným ze šesti elektromechanicky řízených os. Rok poté se na trh dostal první, plně elektrický, průmyslový robot řízený mikroprocesorem. Během 80. a 90. let byly vyvinuty tzv. portálové roboty.

V roce 1989 byla společnost Unimation prodána známé švýcarské firmě Stäubli. Tato firma se věnuje vývoji a prodeji robotů i nadále.

Na počátku 21. století již byly portálové roboty schopny pohybu po zakřivené dráze a taktéž se začaly objevovat roboty od firmy Reis, které disponovaly laserovým paprskem. V předchozích letech byly průmyslové roboty využívány převážně v automobilovém průmyslu. V současnosti se již hojně vyskytují ve všech odvětvích průmyslu. [12]

1.2 Roboty a průmyslové roboty

Robot je mechanický nebo virtuálně inteligentní stroj, který může provádět zadané úlohy automaticky bez pomoci nebo je možné tento stroj pomocí dálkového ovládní řídit. Robot má mít obvykle možnost senzorických vstupů - např. vidění (za pomoci údajů z kamery), dotek atd., rozpoznávání a pohybu, což znamená, že by robot měl disponovat senzory, motory a řídicími jednotkami.

Existuje několik typů robotů. Často jsou rozdělovány na základě funkce, stupňů volnosti, velikosti pracovního prostoru apod. Mezi hlavní typy robotů dnes patří mimo jiné průmyslové roboty, armádní roboty, roboty využívané ve zdravotnictví a také nesmíme zapomenout na micro a nano roboty.

Průmyslové roboty jsou multifunkční mechanická zařízení programovatelná ve třech

nebo více osách a navržena pro pohyb s předměty, materiály a nástroji nebo specializovaná zařízení, která pomocí složitějších programovatelných pohybů dokáží provádět různé úkony. Tyto roboty mají mnoho funkcí, jako například montáž různých částí výrobku, svařování, frézování, testování kvality a mnoho dalších. Celý systém průmyslových robotů nezahrnuje pouze samotné průmyslové roboty, ale také všechna související zařízení, and/or snímače potřebné pro robota k výkonu úlohy, a také komunikační rozhraní.[1]

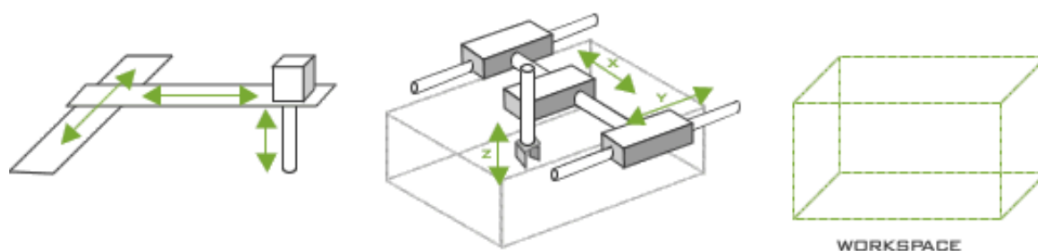
1.3 Rozdělení robotů

Roboty mohou být rozděleny podle mnoha aspektů - podle velikosti a tvaru pracovního prostoru (tento prostor je určen body, do kterých je možné dosáhnout koncovým efektem robotu), druhu kontrolního systému a podle způsobu řízení pohybu.

1.3.1 Podle velikosti a tvaru pracovního prostoru

Kartézské roboty

Hlavice robota se pohybuje v pravoúhlém souřadnicovém systému. K tomu obvykle využívá tři lineární vedení. Pracovní prostor těchto robotů má tvar krychle nebo kvádru. Takovýto robot není příliš energeticky náročný a využívá se tam, kde není potřebný větší počet stupňů volnosti. [1]



Obrázek 1: Kartézský robot [25]

Válcovité roboty

Tyto roboty mají dvě lineární osy a jednu rotační osu kolem jejich počátku. Jak již název napovídá, jejich pracovní prostor má tvar válce.



Obrázek 2: Robot s válcovitým pracovním prostorem [25]

Sférické/polární roboty

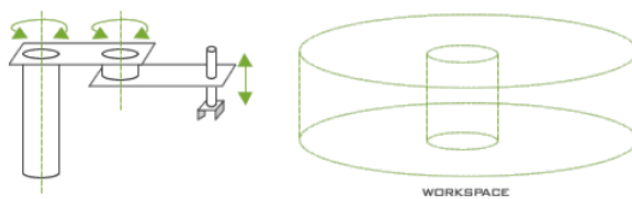
Sférické a polární roboty mají jednu lineární osu a dvě rotační osy. Pracovní prostor těchto robotů je naznačen na obrázku.



Obrázek 3: Sférický robot [25]

Scara roboty

SCARA je speciální robotická konstrukce vyvinutá na konci 70. let v Japonsku. Tyto roboty mají tři rotační klouby, které jim umožňují rychlý pohyb v horizontální rovině. Vertikální pohyb je zajištěný posledním ramenem a je v porovnání s víceosými roboty značně omezen. Oproti více-osým robotům mají však méně náročné programování. SCARA roboty se využívají především k manipulačním nebo technologicky jednodušším úkonům. [1]



Obrázek 4: Scara robot [25]

Delta roboty

Tento robot se skládá ze tří ramen připojených k univerzálním kloubům na základně. Klíčovým konstrukčním prvkem je použití lichoběžníků v rameni, které udržují orientaci koncového efektoru. Delta roboty jsou hojně využívány v továrnách, kde dochází k uchopení předmětu například na pásu a pak jeho balení. Tyto roboty dokáží být velice rychlé, za minutu mohou provést až 300 úchopů předmětu. [5] [1]



Obrázek 5: Delta robot [25]

Kloubové roboty

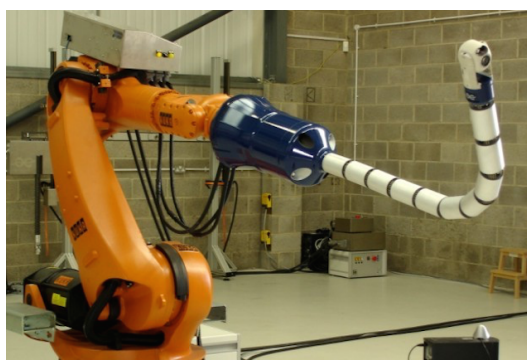
Kloubové roboty jsou nejvíce univerzálními roboty. Velmi blízce napodobují lidskou ruku, tj. rameno, biceps a předloktí, ale většinou bez zápěstí, jak je i možno vidět na obrázku níže. U šestiosých robotů, které disponují "zápěstím", je možné se s koncovým efektem robotu dostat na jakékoliv místo v prostoru a to i v jakémkoliv směru. Tyto roboty nacházejí uplatnění především při sváření, řezání, lakování nebo manipulaci s předměty. Obecně vykonávají složitější operace než již dříve zmíněné roboty. [1]



Obrázek 6: Ukázka reálného kloubového robotu určeného ke svařování [23]

”Snake arm”roboty

Snake arm roboty jsou flexibilními manipulátory s předměty. Jak již název napovídá tyto roboty jsou schopné sledovat přední část robota (efektor) celou svou pohyblivou částí kolem překážky, ale i vnitřkem překážky, pohybem jako když se plazí had. Takovéto roboty jsou konstruovány v podstatě stejným způsobem jako lidská páteř, která je propojena prostřednictvím několika obratlů. Uplatnění těchto robotů je například i při operacích (kolonoskopie, endoskopie a další). Jsou také hojně využívány odborníky na zneškodnění bomb. [1]



Obrázek 7: Ukázka reálného robotu typu snake arm sloužícího k zacházení s jaderným odpadem [26]

1.3.2 Dělení na základě typu řídicích systémů

Servo roboty

Servo roboty jsou řízeny pomocí mnoha senzorů, které nepřetržitě monitorují robota a všechny komponenty související s pozicí robota a jeho rychlostí. Informace ze senzorů jsou porovnávány s požadovanými hodnotami, které jsou dopředu naprogramovány a uloženy v paměti robota. Motory jsou poté ovládnuty tak, aby přesunuly efektor na požadovanou polohu/rychlost.

Roboty bez servomotoru

Na rozdíl od servo robotů není možné využívat zpětnou vazbu senzorů. Roboty jsou řízeny především pomocí systému mechanických přerušení a spínačů kontrolujících mezní polohy robota. [1]

1.3.3 Dělení na základě generování trajektorie pohybu

Bodové řízení (Point-to-Point path)

Takové roboty jsou naprogramovány a řízeny tak, aby se pohybovaly z jednoho diskrétního bodu do druhého, přičemž si dráhu mezi jednotlivými body volí tak, aby

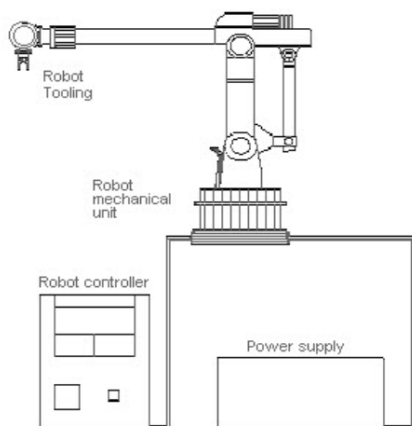
pro něj byla co nejvýhodnější. V automatickém režimu provozu se přesná cesta robota může mírně lišit v důsledku změn rychlosti a geometrie kloubů. Tuto odchylku je velice těžké předpovídat, což může vést k vytvoření potenciálního nebezpečí pro personál i samotné zařízení.

Dráhové řízení (Continuous path)

Robot, jehož cesta je řízena ukládáním velkého počtu prostorových bodů do paměti v průběhu cyklu učení, je robot, který využívá dráhové řízení. Během této doby jsou souřadnice bodů v prostoru jednotlivých os průběžně sledovány a zapisovány do paměti robota. Když je robot přepnut do automatického módu, program je přehrán z paměti a vygeneruje se duplicitní cesta. [1]

1.4 Jednotlivé části průmyslových robotů

Průmyslové roboty se skládají ze 4 hlavních částí: hlavní konstrukce robota, zdroje energie, řídicího systému a efektoru.



Obrázek 8: Čtyři hlavní komponenty průmyslového robotu [25]

Celý robot je umístěn na základně, která je pevně spojena s podlahou (může být umístěn i na pracovní desce či stole), stropem či stěnami. Na základně se nachází tělo robota, jež je tvořeno pevnými rameny spojenými pohyblivými klouby. Každý kloub představuje kinematickou dvojici. Pro řízení jednotlivých kloubů se využívají elektrické, hydraulické nebo pneumatické pohony. Na konci ramene je umístěn koncový efektor.

2 Bezpečnost robotů

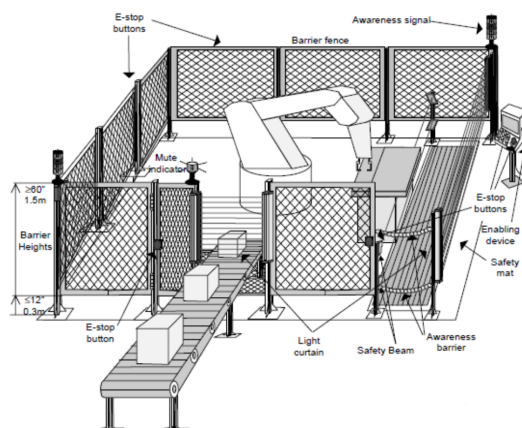
2.1 Úvod do problému bezpečnosti robotů

Roboty jsou široce využívány v průmyslu. Mohou totiž provádět úkony, které jsou například pro člověka nebezpečné, avšak jsou také hojně využívány pro monotónně se opakující práci, která může být člověku nepříjemná. Mimo to jsou průmyslové roboty na rozdíl od člověka schopny plnit složité úkony bez známky únavy. V dnešní době jsou průmyslové roboty využívány především na výrobních linkách a očekává se, že v budoucnu jejich využití i v dalších oblastech vzroste.

Bezpečnost je klíčový faktor při použití průmyslových robotů. Uvádí se, že kolem 15% nehod v průmyslu, využívající pokročilé výrobní technologie, bylo způsobeno automatizovanými výrobními stroji, včetně robotů.

Bezpečnost robotů může být interpretována mnoha způsoby. Za prvé je potřeba předejít situaci, kdy robot poškodí sám sebe. Mnohem důležitější však je vyřešit všechna bezpečnostní rizika, která by mohla vést ke zranění člověka.

Průmyslové roboty jsou programovatelné jednotky navrženy tak, aby prováděly očekávané pohyby. Bohužel pohyby lidí, kteří pracují s roboty, nelze dopředu přesně předvídat, proto je velice důležité zajistit bezpečnost robotu. Většina nehod souvisejících s robotem se stává během programování, údržby, oprav a testování. Všechny tyto úkony zahrnují interakci člověka, což vyžaduje náležité školení o bezpečnosti pro zaměstnance, kteří s robotem pracují, a také řádné používání vhodných ochranných opatření. [1]



Obrázek 1: Efektivně řešený bezpečnostní systém [25]

2.2 Nehody způsobené robotem

Při používání robotů může docházet k nehodám. Tyto nehody mohou být rozděleny do 4 kategorií:

- **Nehody způsobené nárazem nebo kolizí**

Nepředvídatelné pohyby, poruchy jednotlivých součástí robota nebo nepředvídatelné změny v programu, které přímo ovládají rameno nebo jeho periferní zařízení, mohou způsobit nehodu.

- **Nehody při kterých může dojít k zachycení např. části oblečení pracovníka**

Při neopatrném zacházení s robotem může dojít k zachycení kusu oblečení

dělníka. V horším případě dojde k zachycení končetiny či jiné části těla mezi ramenem robota a jiným periferním zařízením. V druhém případě je velké riziko stlačení, či dokonce rozdrčení končetiny.

- **Mechanické nehody**

Do této kategorie můžeme zařadit prakticky většinu všech nehod nebo poruch související se samotným robotem. Může se jednat o poškození pohonu robota, uvolnění jednotlivých součástí nebo poškození koncového efektoru.

- **Ostatní nehody**

Další nehody mohou vyplývat z práce s roboty. Zařízení, jež dodává energii robotu, představuje potenciální elektrické nebezpečí. Dojít může také k nehodě v oblasti životního prostředí. Navíc celé zařízení a kabely, které přivádí potřebnou elektrickou energii, představují možné nebezpečí výbuchu. [1]

2.3 Zdroje nebezpečí způsobující nehody

Potenciální nebezpečí strojů pro člověka lze rozdělit podle zdroje/příčiny následovně: zásah člověka, chyba řízení, neoprávněný přístup, selhání mechanické části robotu, vliv okolního prostředí, porucha v systémech zprostředkovávajících činnost robota a nesprávná instalace. Každé z těchto potenciálních nebezpečí je popsáno níže.

- **Chybný zásah člověka**

Nebezpečí spojené s lidskou interakcí může způsobit nebezpečné, nepředvídané pohyby nebo činnost robota. Velmi často dochází k chybě špatné aktivace "teach boxu" (jedná se o zařízení, pomocí kterého je možné ovládat robota krok za krokem a určit tak všechny jeho pohyby) nebo řídicího systému.

- **Chyba řízení**

Vnitřní poruchy v řídicím systému robota, chyby v softwaru a elektromagnetické rušení můžeme považovat za chyby řízení. Tyto chyby mohou nastat také kvůli poruchám v hydraulických, penumatických nebo elektrických částech řídicího systému.

- **Neoprávněný vstup**

Vstup do chráněného prostoru robota je obecně velmi nebezpečný. Toto nebezpečí se ještě zvyšuje, pokud dotyčná osoba není obeznámena s ochrannými opatřeními.

- **Mechanická porucha**

Provozní programy nemusí zahrnovat poruchu mechanického dílu robota, což může vést k chybnému nebo neočekávanému provozu robota.

- **Vliv okolního prostředí**

Elektromagnetické rušení může mít nežádoucí vliv na provádění zadaného úkonu a tím zvyšovat potenciální riziko zranění pro každou osobu pracující v blízkosti robota.

- **Porucha v systémech zprostředkovávajících činnost robota**

Pneumatické, hydraulické a elektrické systémy úzce související s řízením robota. Za největší riziko, které tyto systémy představují, se bere především vznik požáru nebo dokonce výbuch způsobený elektrickým zkratem. Velké nebezpečí také spočívá v hořlavém hydraulickém oleji.

Všimněme si, že tento seznam není seřazen v nějakém určitém pořadí (např. podle četnosti výskytu nehod způsobených právě jedním z výše zmíněných zdrojů nebezpečí). Existuje mnoho zdrojů nebezpečí a ve skutečnosti se jejich výskyt může lišit například podle odvětví průmyslu, ve kterém se robot využívá, a také podle toho, k jaké práci je využíván. V dnešní době se již díky různým technologiím daří snížit nebezpečí vzniku nehody na minimum. [1]

2.4 Požadavky na bezpečnost robota

Ze statistik vyplývá, že většina nehod a incidentů spojených s roboty se nestává při běžném provozu. Proto je při práci s průmyslovými roboty kladen důraz na bezpečnost ve speciálních provozních režimech. Mezi tyto režimy se řadí: uvedení robota do provozu, nastavování všech periferních zařízení, samotné programování, testování funkčnosti, odstraňování vzniklých problémů a samozřejmě i údržba robota. Při těchto provozních režimech je obsluha robota obvykle v nebezpečné blízkosti robota, proto jsou vyžadována vysoká bezpečnostní opatření, která by měla předejít případným nehodám. [1]

2.4.1 Požadavky a bezpečnostní opatření při běžném provozu

Použití robotů v průmyslu vyžaduje důslednou analýzu všech možných rizik, jejich zdokumentování a pokud je to třeba, tak i snížení daného rizika.

1. Opatření zabráňující volný přístup obsluhy do nebezpečné oblasti při běžném provozu zahrnují následující:
 - Zabránění přístupu do nebezpečných zón pomocí mechanických bariér.
 - Použití bezpečnostních zařízení, které reagují na přiblížení osoby k robotu (pohybové senzory, bezpečnostní náslapné rohože) a v ten samý okamžik bezpečně vypnou stroj.
 - Umožnění přístupu pouze v případě, že je celý systém v bezpečném stavu. Toho může být například dosaženo použitím zařízení pro blokování dveří, který automaticky otevře přístupové dveře až v okamžiku, když je vše bezpečně vypnuto.
2. Opatření potřebná pro prevenci úrazů způsobených například uvolněním jednotlivých částí a jejich vymrštěním:
 - Samotný návrh robota a jeho pracovního prostoru by měl brát v potaz možné uvolnění a vymrštění jednotlivých částí robota.
 - Je potřeba zavést taková opatření, při kterých, i když dojde k vymrštění poškozené části robota, bude zajištěno, že tato část neopustí ochrannou zónu kolem robota. Toho lze docílit například bezpečnostním krytem. [1]

2.4.2 Požadavky a bezpečnostní opatření v průběhu speciálních provozních režimů

Při některých speciálních režimech provozu je zapotřebí pohybů robotu (např. při nastavení a programování robotu). Žádný z těchto pohybů nesmí ohrozit zúčastněné osoby. Vzhledem k tomu by všechny pohyby měly:

- být prováděny s omezenou rychlostí, aby bylo možné zareagovat při nekontrolovaném pohybu,
- být prováděny pouze tehdy, pokud je zaručeno, že žádná část lidského těla není v nebezpečné zóně,

- být prováděny pouze tehdy, pokud je zajištěno, že při pohybu robot nenarazí na žádnou překážku nebo alespoň musí být toto riziko minimalizováno. [1]

2.5 Zabezpečení robotů

Bezpečnost je třeba brát v úvahu už od samého počátku navrhování stroje. Lze tak mnohem snadněji vyřešit rizikové faktory pomocí konstrukčních změn a často i zcela předejít použití technických ochranných opatření.

Mechanické části stroje by neměly obsahovat žádné ostré hrany nebo přesahující díly. Pokud možno by měla být eliminována místa, kde hrozí riziko úrazu.

2.5.1 Technická ochranná opatření

Pokud existuje nebezpečí, které není možné odstranit změnami konstrukčních prvků, je na místě zvolit jiné ochranné opatření. Po určení nezbytné úrovně zabezpečení se volí druh ochranného zařízení. Teprve po ověření jeho funkčnosti a vhodnosti je možné místo označit jako zabezpečené.

Ochranná opatření využívaná v dnešní době

Ochranné ploty se používají k zabránění vstupu do pracovního prostoru robota. Zároveň plot poskytuje preventivní ochranu před úlomky částí robota, jež mohou být vymrštěny z pracovního prostoru. Bariéry se rovněž využívají, pokud je potřeba zajistit, aby pracovník nevstoupil do pracovní buňky z prostoru, kde se např. nakládá a vykládá materiál (určeno pro pásové dopravníky). Ve zvláštních případech, jako je například obloukové sváření či práce s laserem, je zapotřebí, aby byl systém zcela zapouzdrěn. [1]



Obrázek 2: Ochranné ploty v praxi [25]

Pevné ochranné kryty

Slouží k mechanickému zabránění vstupu do nebezpečné oblasti, jako je například pracovní prostor robota. Mohou být pohyblivé (dveře) nebo nepohyblivé (plot). Aby kryty skutečně zamezily kontaktu s jakoukoli částí lidského těla v nebezpečném prostoru, musí být vhodně umístěny. Pokud jsou v nich otvory, je třeba vytvořit odstup ochranné bariéry od nebezpečného místa. Velikost tohoto odstupu je úměrná velikosti otvoru. Jedním z mnoha příkladů může být použití krytu v kombinaci s lisem. Díky těmto krytům není možné do pracovní části stroje vložit ruku či nohu, mechanická zábrana může být navíc doplněna elektromechanickým zámkem, který elektricky zamyká mechanickou zábranu a zároveň tím odemyká ovládání stroje - stroj tedy nelze spustit bez řádného zafixování mechanické zábrany. [1]



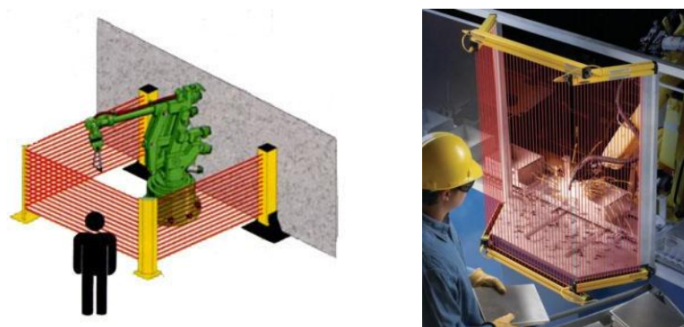
Obrázek 3: Ochranný kryt lisu [25]

Přístroje snímající přítomnost pracovníka

Tyto detektory jsou běžně používány pro zvýšení bezpečnosti při práci s roboty. Především se využívají náslapné rohože a bezkontaktní bezpečnostní přístroje, kterými jsou například světelné závěsy, mřížky, laserové skenery a chytré kamerové systémy. Senzory jako takové samy nezabraňují vstupu do daného prostoru tak jako mechanické zábrany, ale při detekci pohybu v nebezpečné zóně spustí nouzové zastavení. Pokud je to třeba, je nejdříve stroj uveden do bezpečné polohy. Toto však neproběhne okamžitě, proto se ochranné zařízení neumisťuje hned vedle nebezpečného místa, ale v bezpečné vzdálenosti od něj.

- Světelné závěsy:

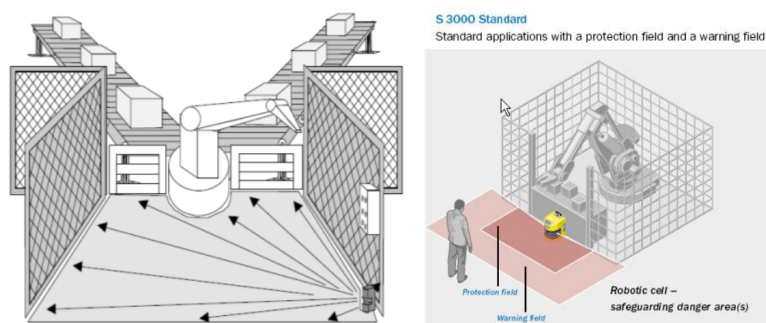
Pokud je v dráze světelného paprsku zjištěna jakákoliv překážka, chod robotu se okamžitě přerušuje. Tato metoda zabezpečení poskytuje okamžitý přístup k robotu, a může tak usnadnit práci. Nicméně světelné závěsy nedokážou chránit před vymrštěnými objekty z pracovního prostoru, proto slouží především jako doplněk k pevným ochranným krytům.



Obrázek 4: Světelné závěsy [25]

- Laserové skenovací přístroje:

Tato zařízení používají jediný laserový paprsek pro mapování oblasti a zjišťování změn, které by mohly znamenat potenciální nebezpečí.



Obrázek 5: Možnost zabezpečení přístupového místa pomocí laserového bezpečnostního skeneru [25]

- Nášlapné rohože:

Nášlapné rohože poskytují ochranné pásmo kolem robotů. Nouzové zastavení nastane po stoupnutí na podložku. Rohože však musí být v bezpečné vzdálenosti, aby nebylo možné přes rohož natáhnout končetinu až do nebezpečného prostoru.



Obrázek 6: Nášlapná rohož - žlutá deska kolem robotu [25]

- Zvukové a světelné varovné systémy:

Tento typ zabezpečení není postačující. Mohou však být použity pro zvýšení bezpečnosti.

- Kontrola parametrů stroje:

V některých případech je třeba kontrolovat různé parametry související s bezpečností. Pokud je překročena limitní hodnota některého z parametrů, ozve se varovný signál nebo se zařízení úplně zastaví. Nejčastěji se měří rychlost pohybu, teplota a tlak. Roboty jsou často vybaveny senzory pro kontrolu pozice. Jedná se o tzv. měkké meze, které mohou přizpůsobit pracovní prostor, ve kterém robot pracuje. Pokud robot opustí definovanou zónu, dochází k ochrannému zastavení robota. [1]

3 Kolaborativní robotika

Zatím jsme se v práci věnovali pouze průmyslovým robotům, které jsou používány především ve velkých výrobních halách. Nedílnou součástí v oblasti robotiky jsou i tzv. kolaborativní roboty, zkráceně "koboty". Slovo kolaborativní bylo převzato z anglického jazyka. Doslovný překlad slova *collaboration* je spolupráce. Jak již český překlad napovídá, jedná se o činnost dvou či více osob nebo subjektů, které si navzájem pomáhají. A toto je i účelem kolaborativních robotů. Narozdíl od velkých průmyslových robotů, jež musí být z bezpečnostních důvodů izolovány od lidí, jsou kolaborativní roboty využívány pro spolupráci s člověkem. Jelikož není bezpečnost zajištěna opatřeními, které byly nastaveny v předešlé části, musí být nastavena jiná bezpečnostní opatření, díky kterým má člověk možnost se pohybovat v pracovním prostoru robota, aniž by hrozilo nebezpečí úrazu.

Na trhu se pohybuje mnoho výrobců, kteří se zabývají výzkumem a vývojem kolaborativních robotů. Každý výrobce navrhuje rozdílná řešení v oblasti bezpečnosti svých robotů. Především záleží na tom pro jaké účely jsou roboty vyrobeny. Avšak nejdůležitějším faktorem je vždy zajištění bezpečnosti při obsluze robota.

3.1 Mezinárodní normy ISO

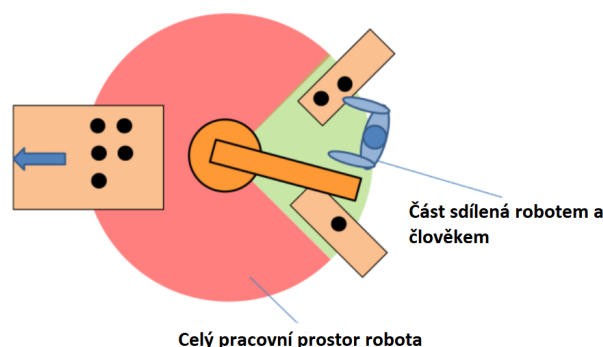
Všechny firmy, které vyvíjí kolaborativní roboty, mají vesměs podobná řešení pro zaručení bezpečnosti. Vše se totiž odvíjí od nutnosti splnit technické specifikace, které jsou dané normou *ISO 10218*. Tato norma specifikuje požadavky a pokyny týkající se bezpečného návrhu, ochranných opatření a informací týkajících se použití průmyslových robotů. Taktéž popisuje základní rizika spojená s používáním robotů a obsahuje opatření, která tato rizika odstraní nebo přinejmenším sníží pravděpodobnost výskytu těchto bezpečnostních rizik. Všechna rizika a možnosti, jak jim předejít, byly popsány v předešlé kapitole.

3.1.1 Technická specifikace ISO/TS 15066

V normách *ISO 10218-1* a *ISO 10218-2* byla kolaborativní robotice věnována pouze jedna strana, což se změnilo v únoru roku 2016, kdy byla vydána technická specifikace *ISO/TS 15066* upřesňující bezpečnostní požadavky z norem *ISO 10218-1* a *ISO 10218-2*. Autorem specifikace je technická komise. V této komisi jsou zastoupeni všichni významní výrobci průmyslových kolaborativních robotů: Universal Robots, ABB, Rethink Robotics, Kuka, Fanuc, Denso Wave a Yaskawa. Výsledný text je konsezem všech zúčastněných a všechny jejich nové kolaborativní roboty jsou s touto specifikací ve shodě. Nová specifikace *ISO/TS 15066* přináší třicet stran informací a doporučení věnovaných jen tomuto novému oboru robotiky.

Technická specifikace ovšem není standardem. Jde opravdu jen o soubor informací a doporučení pro konstruktéry robotů a robotizovaných pracovišť. Obvyklý postup je takový, že technická specifikace je po určitou dobu používána v praxi, sbírají se zkušenosti, které jsou použity k revizím specifikace, jež se následně může stát normou nebo součástí normy: v tomto případě se počítá s tím, že se stane součástí norem *ISO 10218-1* a *ISO 10218-2*.

1. Nejprve je nutné přesně definovat, co je kolaborativní robot. Podle normy je kolaborativní robot definován jako robot, který je používán v bezprostřední blízkosti člověka. Tento prostor je nazýván jako místo určené pro spolupráci a je součástí celého pracovního prostoru robota, jež může být chráněn plotem.



Obrázek 7: Pracovní prostor [34]

2. Tato norma taktéž společně s normou *ISO 10218* definuje sdílení pracovního prostoru ve čtyřech základních módech:

- Safety-rated monitored stop.
- Hand guiding.
- Speed and separation monitoring.
- Power and force limiting.

První dva zmíněné módy však nejsou až tak úplně kolaborativní. Je nutné zmínit hlavní rozdíl oproti obyčejným průmyslovým robotům. Ty s přítomností člověka dopředu nepočítají a pokud se tak stane, znamená to chybový stav a je potřeba robot připravit zpět do provozu. U kolaborativních robotů se s přítomností člověka již dopředu počítá. Pokud člověk přistoupí k robotu, ať už v jakémkoliv módu, robot přeruší svoji činnost a počká, dokud člověk opět nebezpečnou zónu neopustí. [6]

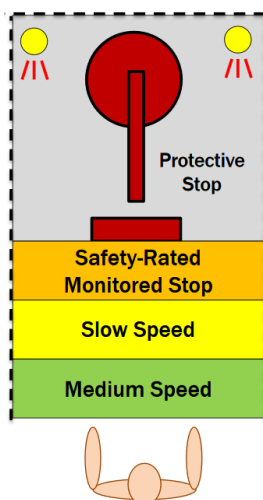
U **safety-rated monitored stop** (SRMS) je robot řízen automaticky až do okamžiku, kdy je funkce aktivována. Poté může být ovládán ručně. Po deaktivaci funkce (což nastane, když všechny osoby opustí pracovní prostor robota) je opět řízen automaticky (nejde tedy o bezpečnostní zastavení, jež by vyžadovalo restartování celého robotického systému). Tento bezpečnostní mód se používá zejména při údržbě, výměně nástroje, ale rovněž se této funkce využívá u robotů, jejichž sdíleným pracovním prostorem může být stůl za okénkem v kleci, na který člověk položí objekt určený ke zpracování robotem. [6]

Hand guiding také obvykle není brán jako provozní kolaborace. Většinou se využívá pro jednoduché nastavení trajektorie při programování úlohy. Tento mód se většinou využívá v kooperaci s SRMS módem, jež hlídá vstup člověka do sdíleného prostoru a zastaví robota. Člověk může ručně nastavit trajektorii, kterou bude robot vykonávat při plnění dané úlohy. Robot spustí svůj chod až poté, co člověk opustí sdílený prostor. [6]



Obrázek 8: Robot používaný v módu Hand guiding [35]

Speed and separation monitoring mód (SSM) již nepotřebuje klec, ve které by byl robot po dobu spolupráce uzavřen. Sdílený prostor je sledován čidly a většinou se dělí na dvě zóny. První zóna, většinou označována jako vnější zóna, způsobuje zpomalení robotu, pokud je v této oblasti detekován jakýkoliv pohyb. Tato zóna může být rozdělena i na více částí a rychlost je snižována postupně (na obrázku č.19 zelenou a žlutou barvou). Robot však nezpomaluje proto, aby rána od něj méně bolela. Tato vnější zóna musí být ještě mimo dosah ramena robotu. K přímému kontaktu mezi člověkem a robotem může dojít až ve vnitřní zóně (resp. může dojít ke kontaktu s objektem, se kterým je právě manipulováno). Proto je potřeba, aby v okamžiku, kdy člověk překročí hranici vnitřní zóny (na obrázku naznačena oranžovou barvou), byl již robot bezpečně zastaven. Žlutá zóna slouží jen k takovému zpomalení robotu, aby v případě přítomnosti člověka ve vnitřní zóně mohl včas a bezpečně zastavit, a předešel tak možnosti kontaktu s člověkem. Podmínky dané normou pro sdílení společného prostoru jsou definovány správným nastavením externího čidla. Tento mód ani nevyžaduje instalaci nějakého speciálního kolaborativního robotu, jelikož zpomalit a zastavit umí jakýkoliv robot. Mód je vhodný především tam, kde se obsluha robotu vyskytuje ve sdíleném prostoru jen po určitou dobu pracovní operace. Během této doby je pohyb robotu výrazně zpomalen či zastaven. [6]



Obrázek 9: Ukázka možnosti řešení módu Speed and separation monitoring [34]

Mód **power and force limiting** (PFL) se však už bez speciálně konstruovaného kolaborativního robotu neobejde, neboť podmínky spolupráce zde zajišťuje robot samotný. Výzkum kolaborativních robotů vyústil v technickou specifikaci ISO/TS 10566, jež doplňuje standard ISO 10218. Tato specifikace (ve velmi zjednodušeném podání) určuje, jaká síla (mechanický tlak) a rychlost nárazu ramena robotu způsobí ještě snesitelnou bolest na různých částech těla, a to za různých podmínek - například při nárazu ve volném prostoru nebo při přitlačení na pevnou překážku. Konstrukce kolaborativních robotů by měla být navržena tak, aby tyto limitní hodnoty nebyly překročeny. To je obvykle zajištěno přidáním rychlostních a momentových senzorů přímo do kloubů robotu. Tento mód je jediným módem, při kterém je umožněna trvalá a plná spolupráce řídicího systému robotu s obsluhou.

Tato funkce je motivována myšlenkou, že nezáleží na tom, dotkne-li se člověk ramene robotu, který stojí, nebo ramene robotu, který se pohybuje přiměřenou rychlostí a při kontaktu vyvíjí jen bezpečnou sílu. Nestací však, je-li bezpečný samotný robot, jelikož pokud robot nese ve svém chapadle např. ostrý, horký nebo jinak nebezpečný předmět, musí být jeho bezpečnost zajištěna bez ohledu na to, že je schopen se pohybovat jen limitovanou rychlostí a vyvinout pouze limitovanou sílu. V těchto případech je zapotřebí umístit robot do ochranné klece a veškerá přímá spolupráce s robotem není možná. S těmito situacemi je potřeba počítat již při návrhu celého řídicího systému. [6]

3.2 Posuzování rizika

Z hlediska implementace specifikace ISO/TS 10566 je možné funkci PFL bez omezení použít v oblasti, kde obsluha ucítí dotyk robotu, který jí však nezpůsobí bolest. Pokud je již kolize bolestivá, ale ještě nepůsobí žádné zranění, je možné funkci PFL doplnit o vhodné bezpečnostní kryty nebo snímače. Pokud se vyskytne riziko závažných zranění při použití funkce PFL, nelze tuto funkci použít vůbec a bezpečnost musí být zajištěna jiným způsobem.

Riziko je posuzováno stejně jako u normálních průmyslových robotů a řídí se tedy stejnými normami (ISO 10218-1 a ISO 10218-2). Podle specifikace ISO/TS 10566 je nutné navíc posoudit zamýšlené a předvídatelné kontakty robotu s člověkem, určit typ kontaktu (přechodný, kvazistatický) a vymezit část těla, na níž může dojít ke kontaktu, případně četnost těchto kontaktů.

Podle přílohy A specifikace ISO/TS 10566 je zapotřebí v první řadě zabránit kontaktu s částmi těla, které jsou výše než krk. Pokud není možné této situaci zabránit, je nutné použít ochrannou helmu a štít chránící oči a obličej. Bezpečnou konstrukcí je potřeba zajistit, že při pohybu nemůže dojít ke skřípnutí části těla mezi pohybující se částí a taktéž je nutný správný návrh pracoviště, aby se předešlo skřípnutí části těla člověka mezi robotem a ostatními částmi pracoviště nebo stěnou. Pracoviště prochází validací pomocí počítačových simulací, zkouškami při využití snímačů síly nebo předmětů, které představují části lidského těla. Tato část specifikace bude s největší pravděpodobností nadále upravována, aby validace byla ještě jednodušší a jednoznačnější.

Snížení rizika je možné docílit pomocí prostředků, jež jsou shrnuty na obrázku č.20. Lze využít všechna zmíněná doporučení, avšak preferována jsou ta v horních řádcích. Směrem dolů jde pouze o prostředky doplňující a pomocné. Riziko sice snižují, ale samy o sobě nejsou dostačující. [7]

	Opatření	Popis	Poznámka
Bezpečná konstrukce (zajišťuje výrobce robotu, chapadla, nástroje)	eliminace	eliminace nebezpečí skřípnutí nebo nebezpečného dotyku; v této části je nutný také správný návrh celého pracoviště	
	náhrada	méně nebezpečné materiály, omezení energie, rychlosti, rozsahu pohybu	
	omezení interakcí	vyloučení dotyku s obsluhou, zvláště dotyku s citlivými částmi těla (omezení rozsahu pohybu)	
Bezpečnostní prostředky a doplňující opatření	bezpečnostní prostředky	kryty, zámky, mříže, bezpečnostní spínače, funkce a systémy	parametry je třeba nastavit podle konkrétních podmínek
	doplňující opatření	nouzové zastavení, bezpečné odpojení momentu, zabezpečení přístupu, možnosti úniku z nebezpečné zóny, prevence pádu	
Opatření na straně uživatele	výstražná zařízení	signální světla, majáky, sirény, bezpečnostní tabulky a značení	
	administrativní opatření	SOP, školení, inspekce, kontroly bezpečnosti práce	
	osobní ochranné prostředky	ochranný oděv, boty, brýle, rukavice apod.	

Obrázek 10: Opatření pro zajištění bezpečnosti pracovišť s kolaborativními roboty [28]

3.3 Řešení jednotlivých výrobců

Ve všech možných dostupných informacích, které výrobci o svých robotech poskytují, se většinou vyskytují podobná řešení zajišťující bezpečnost. Hlavním prvkem pro zaručení bezpečnosti je vždy především samotná konstrukce robotu. Poté se klade důraz na rychlost bezpečného zastavení. Všechny roboty obsahují mnoho senzorů sledujících rychlost a sílu v jednotlivých částech robotu. Díky těmto senzorům jsou roboty schopné zaznamenat kontakt a přerušit vykonávání úlohy.

1. ABB: YuMi

Tento robot má dvě paže, ohebné ruce, univerzální systém podávání součástí, lokalizaci součástí založenou na kamerovém systému a jednoduché programování učení. Proto se hodí hlavně při montáži malých součástek. Bezpečnost je navíc od všech ostatních robotů zvýšena díky lehké konstrukci z horčíkové slitiny, kterou pokrývá pružný plastový plášť, díky němuž je omezena síla působící při případném kontaktu s člověkem. Robot je při detekování kontaktu schopen zastavit svou činnost v řádech milisekund. [13]



Obrázek 11: Robot YuMi [27]

2. Denso WAVE: Cobotta

Dalším příkladem může být robot Cobotta. Konstrukce tohoto robotu je uzpůsobena pro hand-guiding mód. Robot nemá žádné ostré hrany a paže robotu je tvořena zaoblenými částmi, díky čemuž je snížena možnost zachycení ruky člověka. Robot opět navíc obsahuje mnoho senzorů snímající rychlost a moment v jednotlivých kloubech. [14]

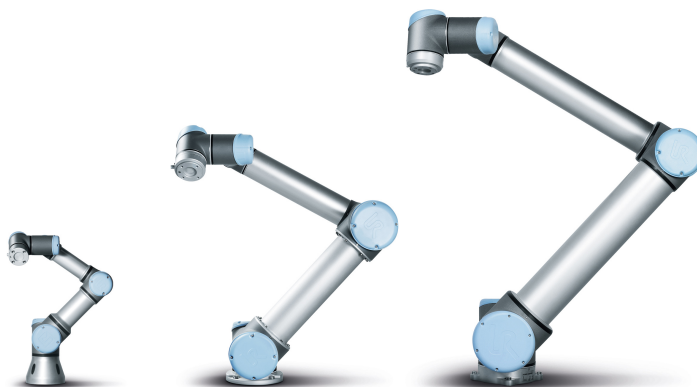


Obrázek 12: Robot Cobotta [29]

3. Universal Robots: UR3, UR5, UR10

Všechny tři roboty ze série Universal Robots mají možnost prostřednictvím ovládacího tabletu jednoduše nastavit bezpečnostní roviny. Tyto roviny slouží k určení prostoru, kde robot sníží svou rychlost. Bohužel tento druh bezpečnostního opatření lze využít pouze v případě, kdy je přesně známo, kde se bude člověk vyskytovat v prostoru robotu, protože pohyb člověka není nijak monitorován. To může vést ke zbytečnému zpomalování robotu, i když není člověk v jeho blízkosti.

Vyššího stupně bezpečnosti lze u Universal Robots robotů docílit snadno. Stačí přidat senzory, které budou s robotem komunikovat pomocí vstupního/výstupního systému. Tyto senzory snímají pohyb v daném prostoru a robot díky nim může redukovat svou rychlost.



Obrázek 13: Roboty společnosti Universal Robots [30]

4. **Techman: TM5**

TM robot má zabudovaný inteligentní systém vidění, jež umožňuje zjistit polohu předmětů, identifikaci čárových kódů, rozlišuje barvy a nabízí mnoho dalších funkcí.

Další výhodou je nepochybně možnost ručního ovládání. Jednoduše lze nastavit body trajektorie vykonávané robotem.

Ochranné ploty nebo buňky již nejsou potřeba. Robot automaticky přeruší svou činnost po detekování kolize. [15]



Obrázek 14: Robot Techman TM5 [37]

Všechny tyto roboty mají však jedno společné - jejich široké využití napříč všemi odvětvími průmyslu. Navíc jejich snadná implementace do výroby představuje mnohonásobné zrychlení.

Kolaborativní roboty zvládnou prakticky vše:

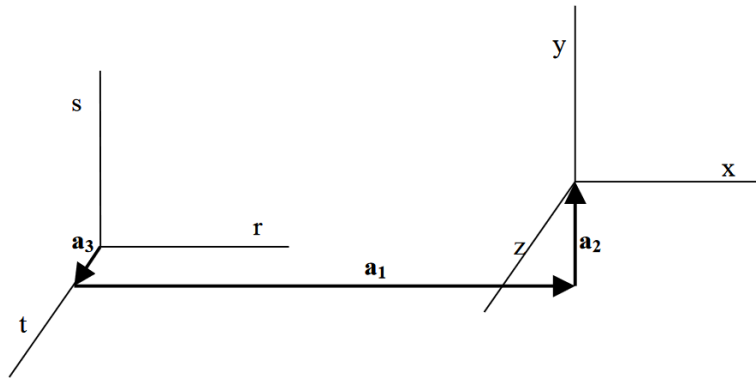
- laboratorní analýzy a testování,
- balení a paletování,
- obsluha zařízení,
- uchopení a umístění předmětů,
- lepení a sváření,
- injekční vstřikování,
- kontrola kvality atd.

4 Reprezentace orientace tělesa v prostoru

Popis orientace tělesa v prostoru je v oblasti průmyslových robotů velice důležitý pojem. Při práci s obrobky nebo jakýmkoliv tělesy totiž robot potřebuje mít přesně danou pozici a orientaci tělesa v prostoru. Orientaci tělesa v prostoru lze v podstatě chápat jako směr tohoto tělesa v prostoru. Orientace však není zcela to samé co směr. Směr je většinou spojený pouze s vektorem, který oproti tělesu nemá "tloušťku" nebo jakoukoliv jinou dimenzi kromě své délky. Pokud tedy bude vektor otočen kolem své osy, nedojde k žádné změně tohoto vektoru. Bude-li však v prostoru otočen předmět, změní se orientace tohoto předmětu. Pro popis orientace tělesa v prostoru je zapotřebí šesti souřadnic:

- 3 souřadnice pro popis referenčního bodu $t_0^0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}$

Tento referenční bod je získán posunem souřadnicového systému např. vzhledem k souřadnicovému systému určujícím orientaci základny robotu.



Obrázek 15: Vlevo - souřadnicový systém robotu, vpravo - nový posunutý souřadnicový systém [24]

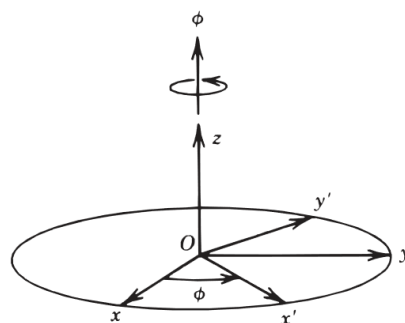
Referenční bod poté představuje počátek nového souřadnicového systému. Nový souřadnicový systém je získán postupným posunem ve směru všech os původního souřadnicového systému.

- Orientace může být popsána jedním ze způsobů:
 - souřadnicemi vektorů spojených s tělesem (n, t, b) ,
 - Eulerovými úhly (ϕ, θ, ψ) ,
 - rotační maticí R ,
 - osou - úhlem,
 - kvaternionem,
 - rotačním vektorem.

4.1 Eulerovy úhly

Při popisu rotace tuhého tělesa kolem nějaké osy je zapotřebí zavedení tří nezávislých parametrů. Těmito třemi parametry jsou právě tzv. Eulerovy úhly. Libovolné natočení tuhého tělesa lze složit ze tří otočení kolem vhodných os. A každé z těchto tří otočení je poté popsáno jedním ze tří Eulerových úhlů. Švýcarský matematik a fyzik Leonhard Paul Euler zvolil pro popis rotace tuhého tělesa tyto úhly:

1. Precesní úhel ϕ - tento úhel charakterizuje otočení kolem osy z kartézského systému souřadnic a je z intervalu $[0; 2\pi)$.

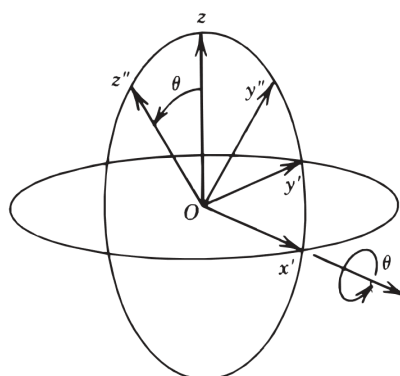


Obrázek 16: Precesní úhel [32]

Rotace kolem osy z o úhel ϕ je popsána maticí rotace:

$$\mathbf{R}_z(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2. Nutační úhel θ - tento úhel charakterizuje otočení kolem osy x' , což je nová poloha osy x po předchozích otočeních a je z intervalu $[0; \pi)$.

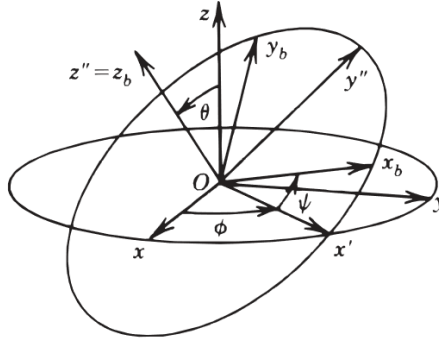


Obrázek 17: Nutační úhel [32]

Rotace kolem osy x' o úhel θ je popsána maticí rotace:

$$\mathbf{R}_{x'}(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

3. Rotační úhel ψ - tento úhel charakterizuje otočení kolem nové polohy osy z a je z intervalu $[0; 2\pi)$.



Obrázek 18: Rotační úhel [32]

Rotace kolem nové polohy osy z'' o úhel ψ je popsána maticí rotace:

$$\mathbf{R}_{z''}(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Výsledné natočení v prostoru jednoznačně popisuje matice R . Tato matice je získána vynásobením matic popisujících jednotlivá natočení:

$$\mathbf{R} = \mathbf{R}_z(\phi)\mathbf{R}_{x'}(\theta)\mathbf{R}_{y''}(\psi)$$

$$\mathbf{R} = \begin{bmatrix} \cos \phi \cos \psi - \cos \theta \sin \phi \sin \psi & -\cos \theta \cos \psi \sin \phi - \cos \phi \sin \psi & \sin \phi \sin \theta \\ \cos \psi \sin \phi + \cos \phi \cos \theta \sin \psi & \cos \phi \cos \theta \cos \psi - \sin \phi \sin \psi & -\cos \phi \sin \theta \\ \sin \theta \sin \psi & \cos \psi \sin \theta & \cos \theta \end{bmatrix}$$

Při využití metody Eulerových úhlů však může dojít k potížím. Pro některé úhly může nastat problém s nejednoznačností popisu. Tento problém se u Eulerových úhlů označuje pojmem "*gimbal lock*" a vede ke ztrátě jednoho stupně volnosti. [8] [10]

4.2 Kvaterniony

Další velice využívanou metodou pro vyjádření orientace v prostoru je metoda kvaternionů. Na rozdíl od metody Eulerových úhlů, ve které jsou využívány v reprezentaci orientace pouze tři parametry, u nichž může nastat již zmíněný problém "*gimbal lock*", kvaterniony využívají parametry čtyři. V matematice jsou kvaterniony rozšířením oboru komplexních čísel. Kvaterniony jsou definovány jako uspořádané čtveřice reálných čísel se speciálně definovanými operacemi sčítání a násobení.

Obyčejná komplexní čísla jsou tvořena pouze z reálných čísel a prvku i , který splňuje $i^2 = -1$, kvaterniony jsou tvořeny přidáním tří prvků i, j, k tak, že jsou splněny následující vztahy:

$$\begin{aligned} i^2 &= j^2 = k^2 = ijk = -1 \\ ij &= k & ji &= -k \\ jk &= i & kj &= -i \\ ki &= j & ik &= -j \end{aligned}$$

Každý kvaternion je lineární kombinací prvků $1, i, j, k$, tudíž ho lze zapisovat jako $a + bi + cj + dk$, kde a, b, c a d jsou reálná čísla. [9]

4.2.1 Praktické využití kvaternionů pro popis rotace v \mathbb{R}^3

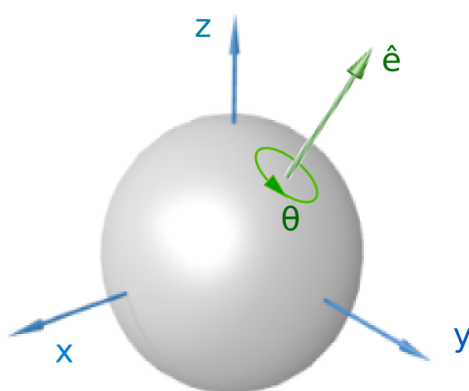
Kvaternion lze také zapsat ve tvaru $a + \mathbf{v}$, kde $a \in \mathbb{R}$ a $\mathbf{v} = v_1i + v_2j + v_3k$, kde \mathbf{v} je vektor v \mathbb{R}^3 . Pro libovolný ryze imaginární kvaternion \mathbf{v} a libovolný kvaternion $h \neq 0$ platí, že $h\mathbf{v}h^{-1}$ je opět ryze imaginární a zobrazení $\mathbf{v} \mapsto h\mathbf{v}h^{-1}$ reprezentuje rotaci v \mathbb{R}^3 . Je možné se omezit pouze na jednotkové kvaterniony $|h| = 1$, poté platí:

rotace kolem osy \mathbf{o} o úhel φ je reprezentována kvaternionem

$$h = \cos \frac{\varphi}{2} + \sin \frac{\varphi}{2} \mathbf{o},$$

kde \mathbf{o} je jednotkový vektor určující směr osy \mathbf{o} .

Podstatnou výhodou při využití kvaternionů pro popis orientace v prostoru je, že skládání rotací odpovídá pouhému násobení příslušných kvaternionů. [9]

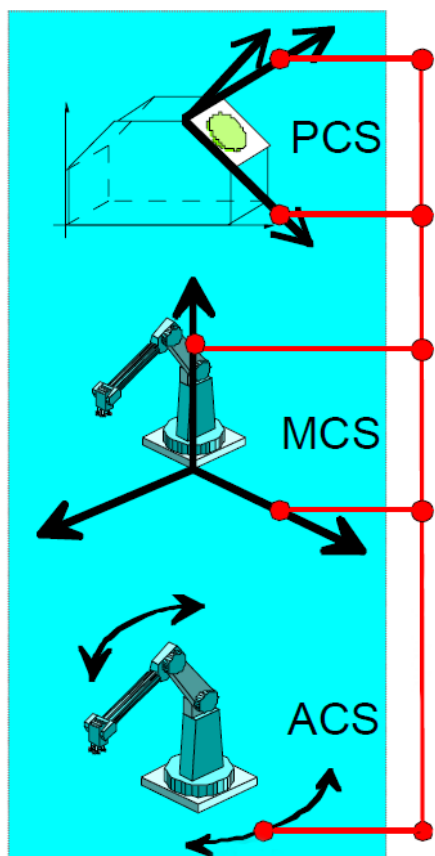


Obrázek 19: Rotace reprezentována Eulerovými osami a úhlem [36]

Návrh komunikace a ovládání robotu

1 Popis řešeného problému

Jeden z mnoha problémů při návrhu robotických systémů nastává při využití více vstupních/výstupních zařízení, např. použití kamery a robotické paže. Poté je zapotřebí zajistit správný převod mezi různými souřadnými systémy. Úkolem této práce bylo navrhnout řídicí systém, který robotu umožní provádět úkony v rámci souřadného systému výrobku, jehož orientace v prostoru je získána z kamery. Díky tomuto návrhu lze zjednodušit práci s obrobky, které již nebudou muset být připravovány vždy na stejné místo, ale jejich orientace a translace v prostoru bude libovolná. Pouze musí být splněno, že obrobek se bude vyskytovat v souřadném systému kamery, aby ho bylo možné detekovat.



- **PCS - Product/workpiece coordinate system** (souřadnice obrobku) - souřadný systém spojen s obrobkem, na kterém se provádějí nějaké operace. Pro usnadnění programování nebo při opakovaných činnostech prováděných na různých místech je výhodné zadávat pohyby relativně vůči tomuto systému souřadnic.
- **MCS - Machine coordinate system** (souřadný systém stroje) - obvykle pevný souřadný systém spojený se základnou robotu. V těchto souřadnicích je typicky reprezentován pohyb nástroje/efektoru připevněného k ramenu robotu (TCP - tool center point)
- **ACS - Axis coordinate system** (kloubové souřadnice robotu) - tyto souřadnice udávají translaci nebo rotaci jednotlivých kloubů stroje, které jsou většinou řízeny nějakým servem. V těchto souřadnicích probíhá interně regulace na úrovni polohových smyček dílčích motorů.

Obrázek 29: Souřadné systémy [33]

2 Seznámení s kolaborativním robotem UR3

Robot od firmy Universal Robots s označením UR3 je jeden z mnoha kolaborativních robotů nabízených na trhu. Pomocí UR3 je možno automatizovat v podstatě cokoliv: od montáže po natírání, od šroubování po označování, od balení po leštění či svařování.

Jedná se o rameno složené z trubic a kloubů z extrudovaného hliníku. Jeho ovládání a programování je velice snadné a to díky programovacímu rozhraní PolyScope. Ani začátečníci v oblasti programování nebudou mít žádný problém s návrhem jednoduchého programu robotu. Pro pokročilé PolyScope nabízí mnoho předefinovaných funkcí, pomocí kterých je možno naprogramovat složité operace robotu.

Universal Robots také nabízí možnost stažení Offline simulátoru. Tento simulátor umožňuje ovládání virtuálního robotu. Programovací prostředí je stejné jako u reálného robotu. Hodí se tedy pro testování složitějších operací, u kterých by mohlo při špatném naprogramování dojít k poškození reálného robotu. Návrh komunikačního rozhraní pro vzdálené ovládání a programování robotu z nadřazeného systému přes komunikaci Modbus TCP a stejně tak návrh systému umožňující programování pohybů robotu v libovolném relativním systému souřadnic obrobku byly taktéž prováděny ve virtuálním simulátoru.

2.1 Grafické rozhraní PolyScope

Pro spuštění robotu je potřeba zapojit ovládací jednotku do elektrické sítě. Spuštění celého systému je provedeno stisknutím vypínacího tlačítka na ovládacím dotykovém displeji. Systém se sám spustí. Pokud by kdykoliv došlo k nějakým potížím a bylo zapotřebí okamžitého vypnutí, lze využít červené bezpečnostní tlačítko E-STOP umístěné hned vedle vypínacího tlačítka na ovládacím terminálu.

2.1.1 Inicializace robotu

Po spuštění robotu je zapotřebí i jeho inicializace. Inicializace slouží ke správnému nastavení instalace robotu (tyto instalace mohou obsahovat různé předefinované bezpečnostní roviny a dají se načíst z uložených souborů) a taktéž nastavení váhy nákladu.

2.1.2 Úvodní obrazovka

Po inicializaci robotu se již objeví úvodní obrazovka. Na výběr je ze 4 možností:

- **Spustit program:** slouží ke spuštění již vytvořeného programu.
- **Naprogramovat robota:** slouží k vytvoření nového programu nebo k editaci již vytvořeného programu.
- **Nastavit robota:** slouží k prvotní inicializaci a nastavení robotu, dále také k nastavení jazyka, hesel atd.
- **Vypnout robota:** slouží k vypnutí robotu a ovládací jednotky.

2.1.3 Spustit program

Tato sekce slouží pro spuštění již vytvořeného programu, který bude načten z paměti.

2.1.4 Obrazovka nový program

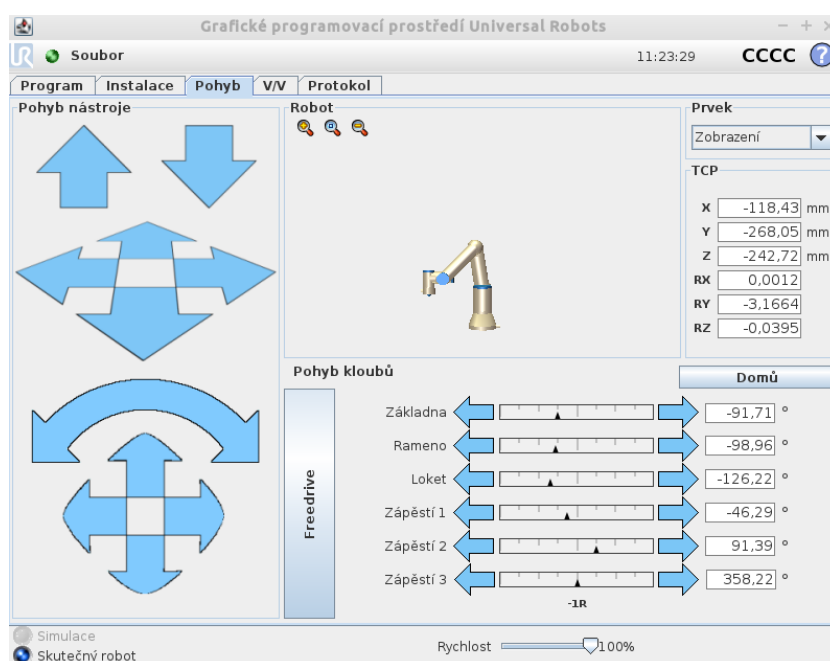
Po stisknutí tlačítka *Naprogramovat robota* se otevře nová obrazovka, ve které lze v horním levém rohu vybrat z mnoha záložek (Program, Instalace, Pohyb, atd.).

2.1.4.1 Instalace

Tato záložka slouží k nastavení parametrů robotu. Je zde možnost upravit vše, počínaje nastavením středového bodu nástroje, přes nastavení vstupů/výstupů, nastavení bezpečnostní konfigurace (omezení kloubů, bezpečnostní roviny a obecná omezení týkající se např. síly, rychlosti, atd.), konče možností nastavení připojení k síti Ethernet/IP.

2.1.4.2 Pohyb

Určeno pro ruční ovládání robotu. V okně *Pohyb nástroje* lze ovládat pozici nástroje pomocí šipek znázorňující směr pohybu.



Obrázek 30: Karta Pohyb

- **Robot** - 3D grafika znázorňuje reálnou aktuální polohu ramene robotu.
- **Poloha prvku a nástroje** - v pravém horním rohu obrazovky se nachází nástroj pro výběr prvků. Textová pole pod ním ukazují souřadnice TCP vzhledem k vybranému prvku.
- **Pohyb nástroje** - slouží k pohybu TCP v daném směru nebo k jeho natočení.
- **Pohyb kloubů** - robot umožňuje přístup k ovládání jednotlivých kloubů.
- **Freedrive** - při stisknutí a držení tlačítka *Freedrive* lze robot fyzicky uchopit rukou a přemístit do požadované polohy.

2.1.4.3 V/V

Slouží k nastavení vstupů a výstupů robotu. Tyto vstupy mohou být analogové i digitální.

2.1.4.4 Protokol

Poskytuje všechny důležité informace o aktuálním stavu robota a jeho nástrojů. Tyto informace se týkají např. teploty ovladače, síťového napětí, průměrného napájení robota. Dále jsou kontrolovány všechny klouby robotu a jejich zatížení.

2.1.4.5 Program

Nejdůležitější část grafického prostředí robotu sloužící k naprogramování robotu. Obrazovka *Program* obsahuje 4 záložky (Příkaz, Grafika, Struktura, Proměnné), navíc v levé části obrazovky je naznačena struktura daného programu (programový strom).

- 1) Příkazy je možno přidat do programového stromu přes záložku **struktura**, kde lze vybrat již mnoho předefinovaných příkazů, ale je také možnost vkládání vlastního zdrojového kódu.

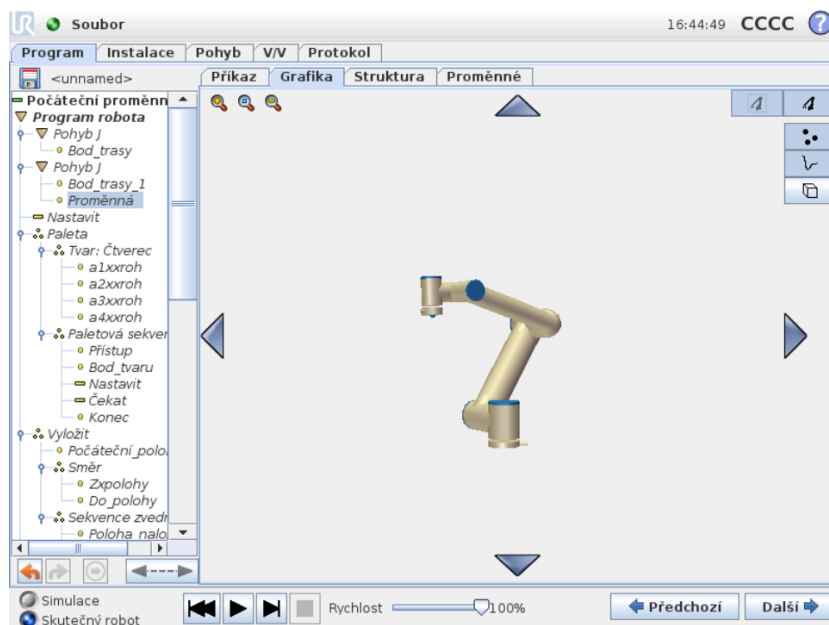
Základní příkazy:

- Příkaz **Pohyb** - pohyb robotu mezi jednotlivými body trasy. Na výběr je mezi těmito možnostmi:
 - Pevný bod trasy - pevně dané body určující trasu robotu.
 - Relativní bod trasy - relativní poloha je dána jako rozdíl mezi výchozí a cílovou polohou. Udává polohu ramene vzhledem k předchozí poloze, například "o 2cm doleva".
 - Proměnný bod trasy - bod trasy je určený proměnnou. Proměnná musí být ve tvaru pozice.
- Příkaz **Čekat** - robot čeká po dobu určenou uživatelem nebo reaguje na hodnoty ve V/V registrech.
- Příkaz **Nastavit** - slouží k nastavení digitálních nebo analogových výstupů na zvolenou hodnotu.
- Příkaz **Samostatné okno** - jedná se o zprávu, která se zobrazí na obrazovce ve chvíli, kdy program dosáhne tohoto příkazu. Chod programu se v té chvíli zastaví. Lze zadat vlastní text. Robot vyčká s vykonáváním programu až do chvíle, kdy uživatel stiskne tlačítko *OK*.
- Příkaz **Zastavit** - vykonávání programu se v tomto bodě zastaví.
- Příkaz **Komentář** - slouží programátorovi k zachování např. nějaké důležité informace o průběhu programu. Nemá vliv na vykonávání programu.
- Příkaz **Složka** - složka slouží k uspořádání a označení určitých částí programu. Samotná složka neprovádí žádné operace.

Poté je možné využít několika pokročilých příkazů:

- Příkaz **Cyklus** - základní příkazy programu se budou cyklicky opakovat. V závislosti na výběru jsou příkazy základního programu prováděny v nekonečném cyklu, v určitém počtu opakování nebo dokud je daná podmínka pravdivá. Pokud je použit cyklus s výrazem jako koncovou podmínkou, umožňuje rozhraní PolyScope nepřetržité vyhodnocování tohoto výrazu, takže lze program přerušit kdykoli během provádění, nikoli pouze po každém zopakování.
- Příkaz **Podprogram** - může obsahovat části programu, které jsou nutné na několika místech.
- Příkaz **Zadání** - slouží k zadání hodnot proměnných. Hodí se především u složitějších programů.

- Příkaz **If** - struktura příkazu If...else může změnit chování robota na základě vstupů senzorů nebo hodnot proměnných. Opět lze stejně jako u cyklu kontrolovat výraz nepřetržitě.
 - Příkaz **Skript** - tento příkaz umožňuje přístup k základnímu skriptu v reálném čase, který je prováděn ovladačem robota.
 - Příkaz **Událost** - lze použít k monitorování vstupního signálu a provedení akce či nastavení proměnných, když je vstupní signál vysoký. Například v případě, že výstupní signál přepne na vysokou hodnotu, vyčká program 100 ms a opět nastaví nízký signál.
 - Příkaz **Vlákno** - označuje paralelní proces programu robota. Vlákno lze využít k ovládní externího stroje nezávisle na ramenu robota. Vlákno může komunikovat s programem robota pomocí proměnných a výstupních signálů.
- 2) Karta **Grafika** - slouží pro nahlédnutí na aktuální pozici robota. Jednoduše tak lze kontrolovat provádění aktuálního programu.
 - 3) Karta **Struktura** - slouží k vkládání příkazů do programového stromu a k jejich editaci.
 - 4) Karta **Proměnné** - zde jsou uvedeny aktuální hodnoty všech proměnných uložených v paměti.

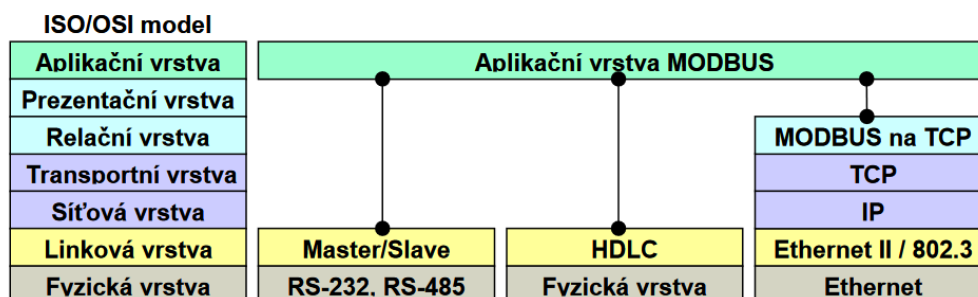


Obrázek 31: Náhled grafického rozhraní PolyScope

3 Návrh komunikačního rozhraní

Pro složitější úlohy, při kterých je zapotřebí např. komunikace mezi robotem a kamerou, jako v našem případě, je výhodné řídit robot z nadřazeného systému. Tato řešení ve většině případů umožňují více možností v oblasti programování robotu. Komunikace mezi řídicím systémem a robotem může být zprostředkována mnoha způsoby. Pro danou implementaci byla vybrána možnost komunikace pomocí protokolu Modbus.

Modbus je otevřený protokol používaný pro komunikaci mezi různými typy zařízení (PLC, dotykové displeje, I/O rozhraní atd.). V modelu ISO/OSI se jedná o komunikační protokol na úrovni aplikační vrstvy. Modbus umožňuje komunikaci typu klient-server mezi různými zařízeními a na různých typech sítí a sběrnic. Protokol podporuje mnoho komunikačních médií, např. sériové linky, optické a rádiové sítě nebo síť Ethernet s využitím protokolu TCP/IP, čehož bude využito pro následnou implementaci komunikace nadřazeného systému s robotem. Komunikace je zprostředkována metodou požadavek-odpověď. Požadovaná funkce je poté blíže specifikována pomocí kódu funkce, který je součástí daného požadavku. [18]



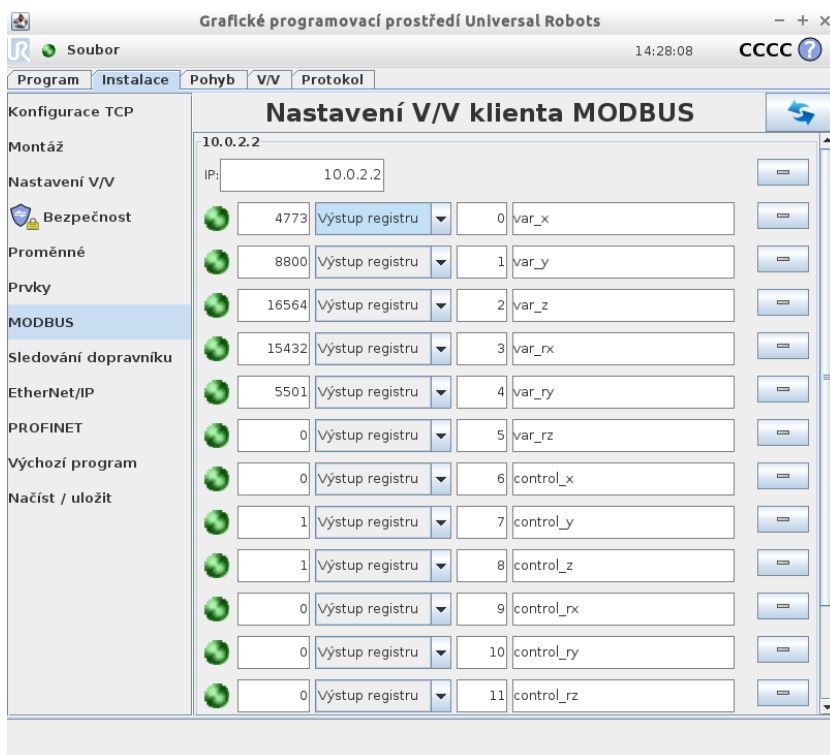
Obrázek 32: Příklady implementace komunikace Modbus [31]

3.1 Nastavení komunikace na straně robotu

Grafické rozhraní PolyScope již disponuje možností nastavit signály Modbus klient. Připojení k serveru (nebo podřízeným jednotkám Modbus) na zadaných IP adresách lze vytvořit pomocí vstupních/výstupních signálů (registry nebo digitální). Každý tento signál má jedinečný název a lze ho jednoduše použít v programech.

Nastavení Modbus klienta lze snadno docílit pomocí záložky *Instalace* v rámci již vytvořeného programu. Program, jež je popsán v této bakalářské práci, již obsahuje vlastní instalaci. Která obsahuje všechna nutná nastavení. Po každém spuštění programu je uživatel dotázán, zda-li chce načíst instalaci uloženou v programu.

V tomto případě byla zvolena možnost s využitím registrů. Na tyto registry jsou zapisovány hodnoty získané z kamery. Hodnoty jsou ještě před zápisem zpracovány řídicím systémem REXYGEN, který je až poté zapíše na příslušné 16-bitové registry.



Obrázek 33: Nastavení klienta Modbus

V konfiguraci klienta Modbus bylo zapotřebí načíst 12 hodnot. Prvních 6 hodnot určuje polohu v prostoru a 6 dalších hodnot je zapotřebí pro rozlišení kladných a záporných čísel. Pokud je konfigurace správně nastavena, budou u jednotlivých registrů svítit zelené kontrolky.

Na každém řádku je zapotřebí nastavit adresu daného registru (v tomto případě hodnoty od 0 do 11). Načtené hodnoty z registrů jsou zobrazeny hned vedle kontrolky. Všechny tyto hodnoty lze uložit jako proměnnou do paměti.

3.2 Nastavení komunikace v řídicím systému REXYGEN

REXYGEN slouží pro programování řídicích systémů bez ručního psaní kódu. Pro snadný návrh algoritmů lze využít REXYGEN Studio. Návrh je velice podobný návrhu algoritmů pomocí globálně rozšířeného produktu Matlab Simulink. REXYGEN obsahuje rozsáhlou knihovnu funkčních bloků, které lze při návrhu využít. Uživatelé jsou k dispozici jak jednoduché komparátory a časovače, tak i pokročilé bloky, které slouží ke zpracování analogových signálů a k regulaci. Navržené algoritmy lze jednoduše zkompileovat a spustit na cílovém zařízení. Při prvních pokusech představovalo cílové zařízení osobní počítač, na kterém byl spuštěn i virtuální simulátor robotu.

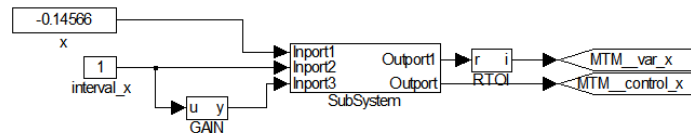
Editor REXYGEN Studio umožňuje vytvoření souborů .mdl. Tyto soubory jsou složeny pouze z bloků knihovny REXYGEN. Pro návrh komunikace byly zapotřebí dva soubory. První soubor *exec_final* nastavuje cílovou platformu a časování jednotlivých úloh, které jsou v druhém souboru *robot_final*.

3.2.1 Popis jednotlivých bloků v souboru robot_final

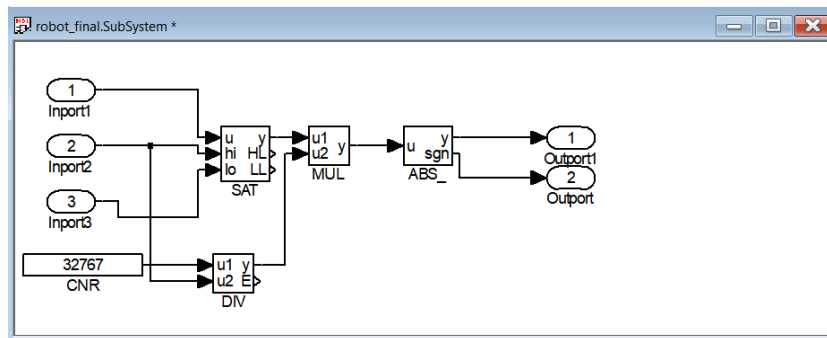
Při nastavení komunikace pomocí MODBUS TCP bylo zapotřebí vyřešit problém

v podobě 16-bitových registrů. Pro popis polohy obrobku je zapotřebí definovat souřadnice pomocí reálných čísel. Ty však není možné jednoduše zapsat na 16-bitový registr. Zápis by byl možný pomocí kódování. Bohužel systém robotu umí načítat pouze celá čísla a jen hodnoty od 0 do 32767, což je kladná část rozsahu integeru.

Řešení tohoto problému je velice jednoduché. Stačí namapovat rozsah potřebných reálných čísel na rozsah integeru. Na registr jsou poté zapisovány celočíselné hodnoty. Problém se zápisem záporných čísel, které systém robotu nedokázal číst a načítal pouze hodnotu 0, byl vyřešen přidáním pomocného registru. Na tento registr byla zapisována 0 pokud znaménko čísla bylo mínus. Pokud byly hodnoty kladné, na registr byla zapsána hodnota 1.



Obrázek 34: Ukázka řešení problému reálných čísel (souřadnice x)



Obrázek 35: Subsystem

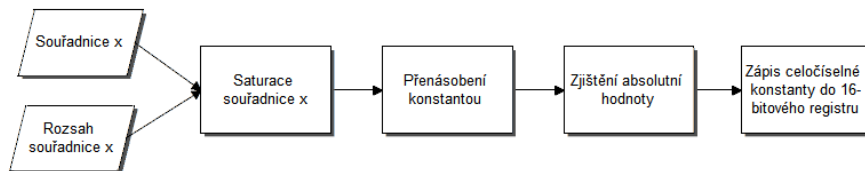
Struktura algoritmu pro namapování reálných čísel:

Popis odpovídá struktuře na dvou předešlých obrázcích.

- Proměnná x (obrázek 34) - tato proměnná reprezentuje x -ovou souřadnici obrobku. Tato proměnná je přivedena na vstup subsystemu.
- Proměnná $interval_x$ (obrázek 34) - tato proměnná reprezentuje rozsah, ve kterém jsou hodnoty x očekávány. Tato hodnota je navíc přenásobena v bloku GAIN hodnotou -1 . Obě tyto hodnoty jsou taktéž přivedeny na vstup subsystemu.
- Blok SAT (obrázek 35, saturace) - jako parametr u je na vstup přivedeno reálné číslo x . Parametry high a low udávají horní a dolní hranice. Tento blok tedy kontroluje, zda-li se parametr x vyskytuje v určeném intervalu, aby nedošlo například k poškození robotu při zadání nedosažitelných hodnot. Výstup y se tedy bude rovnat u pokud bude splněna podmínka: $lolim \leq u \leq hilim$. Pokud $u < lolim \rightarrow y = lolim$ nebo $u > hilim \rightarrow y = hilim$.
- Blok MUL (obrázek 35, násobení dvou signálů) - tento blok slouží k přenásobení reálného čísla x (vstup $u1$) konstantou (vstup $u2$), která je dána jako:

$$u2 = \frac{rozsah_integer}{rozsah_x}.$$

- Blok ABS (obrázek 35, absolutní hodnota) - napomáhá tomu, aby při zpětném dělení bylo dosaženo co nejmenší chyby. Výstup y je poté zapisován jako celočíselná konstanta do 16-bitového registru `var_x`. Výstup `sgn` je zapisován do registru `control_x`. Tato hodnota slouží ke zpětnému zjištění, zda-li se jedná o kladné či záporné reálné číslo.



Obrázek 36: Zjednodušený popis mapování reálného čísla

3.2.2 Popis jednotlivých bloků v souboru `exec_final`

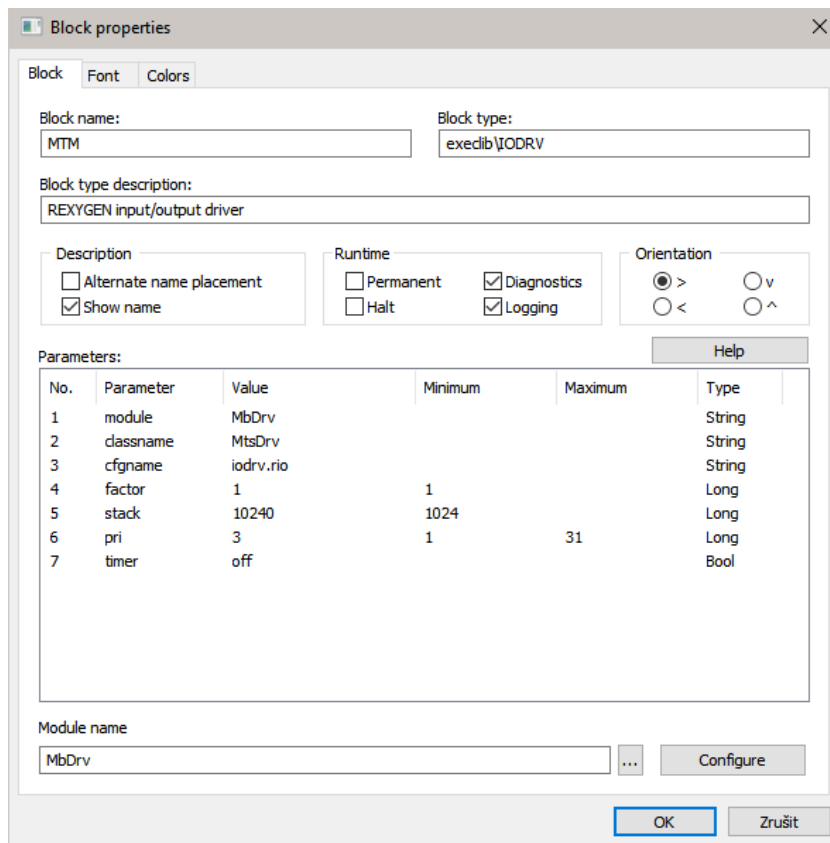
Soubor `exec_final` ve formátu `.mdl` je tzv. *hlavním souborem projektu*, ve kterém se konfigurují jednotlivé subsystemy řídicího systému REXYGEN. Tento soubor tvoří pouze jeden blok. Tím je blok `EXEC`. V systému Matlab Simulink neexistuje žádný blok, který by se svými vlastnostmi podobal bloku `EXEC`. Samotný blok `EXEC` a všechny na něj navázané bloky, nerealizují žádný výpočetní algoritmus. Slouží pouze překladači REXYGEN Compiler pro sestavení celé aplikace řídicího systému.

Konfigurace celého systému REXYGEN je složena z jednotlivých modulů (`Modules`), vstupně-výstupních ovladačů (`Drivers`), archivačního subsystemu (`Archives`) a subsystemu reálného času, obsahujícího blok `Qtask` (rychlé úlohy s vysokou prioritou) a čtyři prioritní úrovně pro plánování výpočetních procesů. Parametr `tick` určuje základní periodu, se kterou jsou spouštěny jednotlivé úlohy. Jednotlivé výpočetní úrovně (`Level0` až `Level13`) jsou poté určeny násobky základní periody. Moduly slouží k rozšiřování a doplňování řídicího systému. Vstupně-výstupní ovladače jsou implementovány jako rozšiřující moduly.

Systém REXYGEN poskytuje komunikaci pomocí protokolu Modbus. Stačí využít `MbDrv` ovladače, který podporuje jak verzi RTU, tak verzi TCP/IP. Pro verzi TCP/IP je zapotřebí připojit dva bloky. Prvním je blok `MODULE` pojmenovaný `MbDrv`, který je připojen jako modul. Druhým je blok `IODRV` pojmenovaný `MTM` a připojen jako ovladač.

Tři nejvíce důležité parametry v nastavení bloku `MTM`:

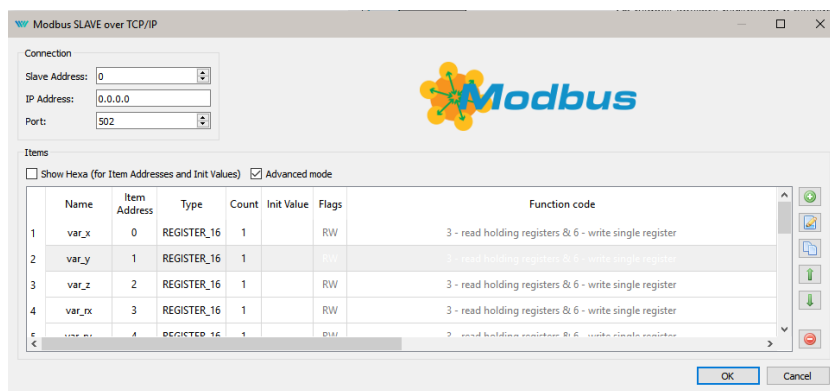
- `module` - jméno modulu, se kterým je ovladačem spojen, v tomto případě `MbDrv`,
- `classname` - třída ovladače určující roli cílového zařízení a Modbus verzi, v tomto případě `MtsDrv` (Modbus TCP/IP Slave),
- `cfgname` - jméno konfiguračního souboru ovladače, v tomto případě `iodrv.rio`.



Obrázek 37: Nastavení bloku MTM

Nastavení ovladače lze konfigurovat pomocí dialogového okna (Configure). Zde je potřeba nastavit parametry všech registrů. Ke každému registru je zapotřebí zvolit jedinečné jméno, adresu registru, zvolit jeho typ a povolit možnost čtení a zápisu pro daný registr.

V tomto případě byly zvoleny 16-bitové registry. Jména jednotlivých registrů byly pro přehlednost nastaveny stejně jako v konfiguraci na straně robotu. Nastaveno bylo 12 registrů (6 pro souřadnice a úhly, 6 pro kontrolu znaménka).



Obrázek 38: Konfigurační dialog

Posledním blokem, připojeným k bloku EXEC, je blok TASK, jehož jméno odpovídá názvu souboru robot_final. Tento blok reprezentuje algoritmus vytvořený právě v souboru robot_final. Této úloze je přiřazena výpočetní úroveň Level10.

4 Implementace programu pro ovládání robotu

Hlavním úkolem této bakalářské práce bylo navrhnout systém, který umožní programování pohybů v libovolném relativním souřadnicovém systému obrobku. Načtení hodnot získaných pomocí kamery již bylo vysvětleno v předchozí části. Pomocí řídicího systému REXYGEN jsou poté souřadnice zapsány na jednotlivé registry. V grafickém rozhraní PolyScope jsou data z registrů uložena do proměnných. Tyto proměnné jsou dostupné po celou dobu chodu programu a je možné je jakkoliv přepisovat a manipulovat s nimi.

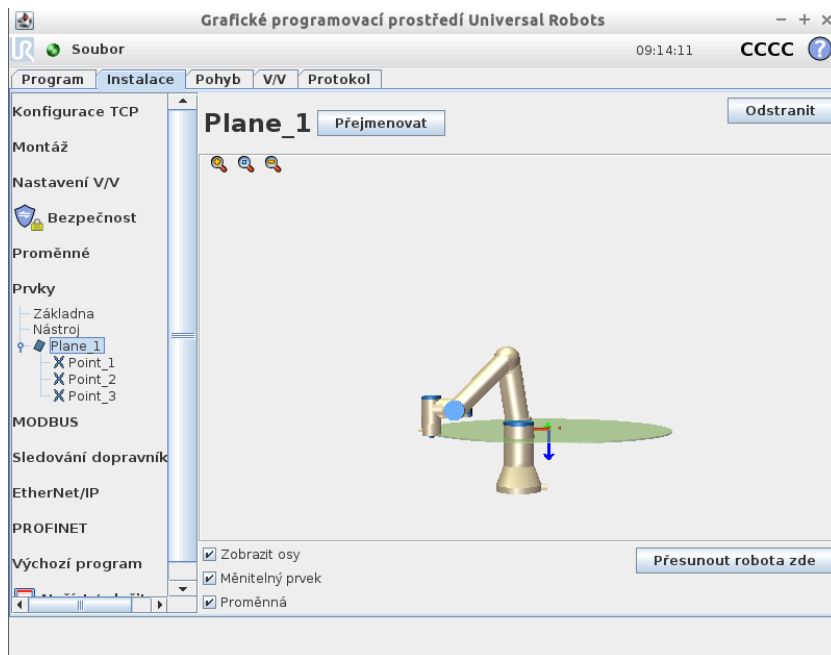
Při návrhu algoritmu byl největší problém v nedostatku informací, které se týkají širších možností v oblasti programování. Obyčejnému uživateli jsou pomocí návodu robotu zprostředkovány a popsány pouze jednoduché úkony. Know-how, které je zapotřebí při složitějším programování, např. v tomto případě pro návrh algoritmu pro práci robotu v relativních souřadnicích obrobku, není ve většině případů nikde dostupné. Toto know-how stále vlastní především distributoři robotů, kteří díky tomu navrhují robotické systémy na míru.

Pro zprovoznění ovládání bylo důležité nalézt, jaké možnosti umožňuje práce s rovinami. Tyto roviny lze jednoduše nastavit v instalaci robotu. Roviny by pak bylo možné využít jako reprezentaci umístění obrobku v prostoru. Pro algoritmus, který by umožňoval programování v libovolném relativním systému souřadnic obrobku, by však bylo zapotřebí tyto roviny v průběhu programu měnit v závislosti na datech z kamery. Roviny lze naštěstí uložit jako proměnnou a poté s ní pracovat v průběhu chodu programu.

4.1 Nastavení rovin v instalaci robotu

Pro nastavení roviny je zapotřebí v záložce *Instalace* u daného programu zvolit možnost *Prvky* a vytvořit novou rovinu. Každá nová rovina je v instalaci nastavena pomocí tří různých bodů v prostoru, které jsou postupně zadávány uživatelem. Důležitou podmínkou pro vytvoření proměnné, se kterou bude možné manipulovat, je zaškrtnutí dvou checkboxů - a to **Měnitelný prvek** a **Proměnná**. Tyto dva parametry poté uloží vytvořenou rovinu jako proměnnou do paměti a také umožní uživateli měnit parametry této roviny. Rovina poté reprezentuje polohu a orientaci obrobku v souřadnicovém systému robotu.

Tento způsob řešení má široké využití. Například při gravírování nějakého loga na předmět, jehož velikost je předem známa. Pokud by byly body zadávány na pevno v souřadnicovém systému robotu, bylo by pak nutné zajistit, že výrobek bude vždy umístěn na stejné místo. Pokud však bude využito programování v souřadnicovém systému obrobku, bude tato úloha velice zjednodušena. Předměty poté mohou klidně přijíždět na páse a robot díky informacím z kamery bude schopen reagovat na rozdílnou orientaci obrobků. Body potřebné k vygravírování loga budou tedy již zadávány vůči souřadnicovému systému obrobku a není nutné převádět souřadnice jednotlivých bodů do souřadnicového systému robotu.



Obrázek 39: Vytvoření roviny v instalaci robotu

4.2 Struktura programu

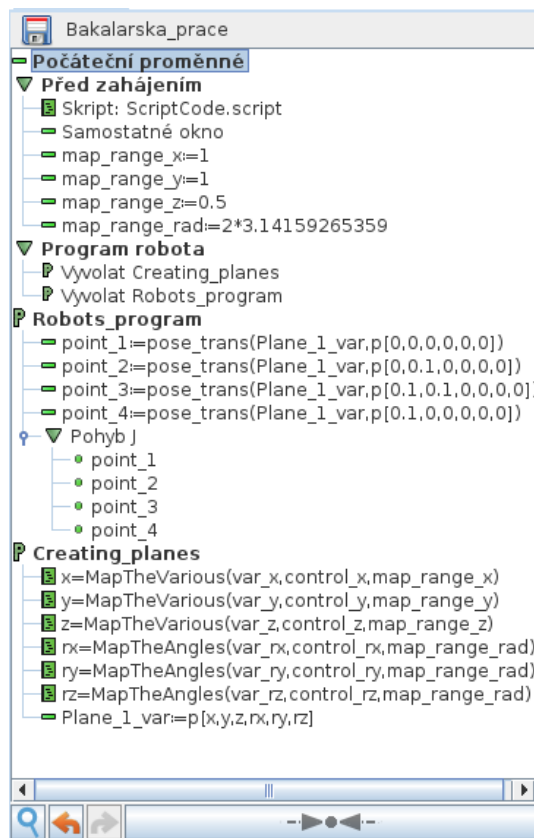
Robot od Universal Robots může být ovládán třemi různými způsoby:

- pomocí grafického uživatelského rozhraní,
- kódem skriptu,
- C-API rozhraním.

Pro ovládání robotu v libovolném relativním souřadném systému obrobku byl vytvořen v celku jednoduchý algoritmus využívající první dvě možnosti pro ovládání robotu (skript a grafické uživatelské rozhraní).

URScript je jednoduchý programovací jazyk určený pro ovládání robotu na úrovni skriptu. Podobně jako ostatní programovací jazyky má i URScript proměnné, datové typy, cykly a podmínky, funkce atd. Navíc od ostatních však URScript disponuje i přidávanými proměnnými a funkcemi, které monitorují a řídí vstupy/výstupy a také pohyby robotu.

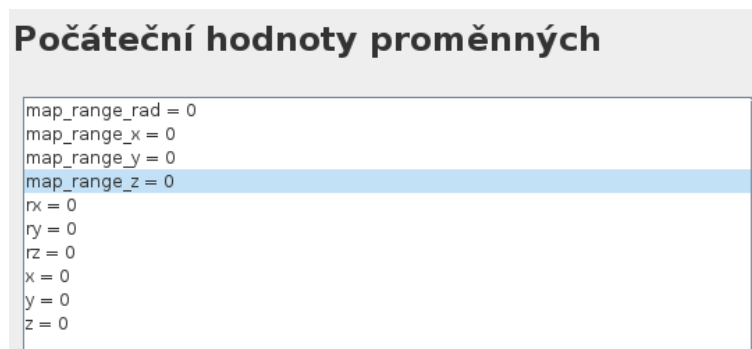
Aby vše fungovalo musí být správně nastaveny a zprovozněny všechny předešlé body řešené v této bakalářské práci. Pro vytvoření daného algoritmu již stačí po spuštění, např. prázdného souboru a načtení uložené instalace, v záložce *Program* vybrat všechny potřebné příkazy a správně nastavit proměnné. Pro zjednodušení čitelnosti kódu byly použity i již zmíněné jednoduché skripty.



Obrázek 40: Ukázka struktury algoritmu

4.2.1 Nastavení počátečních proměnných

Pro správný chod programu je zapotřebí nastavení počátečních hodnot proměnných. Pokud by tak nebylo učiněno, mohlo by se stát, že někde v paměti zůstanou uloženy hodnoty z předešlých spuštění a mohlo by dojít k chybnému stavu. V tomto případě byly všechny vytvořené proměnné nastaveny na hodnotu 0.



Obrázek 41: Nastavení hodnoty počátečních proměnných

4.2.2 Akce provedené před zahájením programu

Tato sekvence slouží k provedení úkonů před spuštěním samotného programu. Nejprve jsou nastaveny správné hodnoty pro rozsahy jednotlivých souřadnic a také rozsah radiánu. Při spuštění programu se otevře dialogové okno v grafickém rozhraní robotu a upozorní uživatele, aby zkontroloval správnost nastavených rozsahů s rozsahy, které jsou nastaveny v řídicím systému REXYGEN.

Před zahájením je taktéž zapotřebí načíst kód skriptu, který bude program využívat po celou dobu svého chodu. Tento skript byl vytvořen a uložen jako soubor.



Obrázek 42: Kód skriptu

Tento skript slouží pro mapování celých čísel zpět na reálná. Jedná se tedy o zpětný proces, který byl proveden v řídicím systému REXYGEN. Skript obsahuje funkci:

- **MapTheVarious** - tato funkce má tři vstupní parametry:
 1. *value* - celé kladné číslo v rozsahu 0 až 32767,
 2. *control* - pomocná proměnná obsahující informaci o "znaménku" celého čísla,
 3. *map_range* - rozsah do kterého je číslo zpětně mapováno.

Funkce nejprve zkontroluje, zda-li je pomocná proměnná *control* nulová nebo nenulová. Podle této informace přidá k celému číslu znaménko mínus, resp. načte celé číslo, tak jak bylo této funkci předáno. Poté už jen tuto hodnotu vydělí podílem

$$\frac{\text{rozsah_integer}}{\text{map_range}}.$$

Jako výstup této funkce je předáváno reálné číslo reprezentující souřadnice.

4.2.3 Program robota

Samotný program se skládá pouze ze dvou podprogramů. Tento způsob rozdělení byl zvolen pro jednodušší práci s robotem a především pro robustnost tohoto programu. Uživatel se tedy nemusí starat o načítání proměnných do paměti. Pouze si přizpůsobí program ovládající činnost robotu svým požadavkům.

- **Creating_planes** - tento podprogram ovládá načítání a práci s proměnnými.

První tři funkce tohoto podprogramu volají již vytvořené funkce ze skriptu. Jako parametry předávají hodnoty, které jsou získány ze 16-bitových registrů. Poslední funkcí je vytvoření roviny (souřadnicový systém obrobku) pomocí získaných reálných čísel reprezentující souřadnice.

```

P Creating_planes
- x=MapTheVarious(var_x,control_x,map_range_x)
- y=MapTheVarious(var_y,control_y,map_range_y)
- z=MapTheVarious(var_z,control_z,map_range_z)
- rx=MapTheVarious(var_rx,control_rx,map_range_rad)
- ry=MapTheVarious(var_ry,control_ry,map_range_rad)
- rz=MapTheVarious(var_rz,control_rz,map_range_rad)
- Plane_1_var:=p[x,y,z,rx,ry,rz]

```

Obrázek 43: Podprogram Creating_planes

- **Robots_program** - tento podprogram ovládá činnost robotu.

Pro ukázkou byl zvolen jednoduchý příklad. Robot v souřadnicovém systému obrobku projede čtyřmi body a opíše čtverec o rozměrech 10x10cm. Nejprve je zapotřebí vytvoření proměnných, které reprezentují jednotlivé body trasy. Tyto body jsou již vztaženy vůči souřadnicovému systému obrobku. Pohyby jsou definovány pouze jako posuny v osách x a y . Poté již stačí dát funkci *Pohyb J* jako parametry právě vytvořené body trasy.

```

P Robots_program
- point_1:=pose_trans(Plane_1_var,p[0,0,0,0,0,0])
- point_2:=pose_trans(Plane_1_var,p[0,0,0.1,0,0,0])
- point_3:=pose_trans(Plane_1_var,p[0.1,0.1,0,0,0,0])
- point_4:=pose_trans(Plane_1_var,p[0.1,0,0,0,0,0])
- Pohyb J
  - point_1
  - point_2
  - point_3
  - point_4

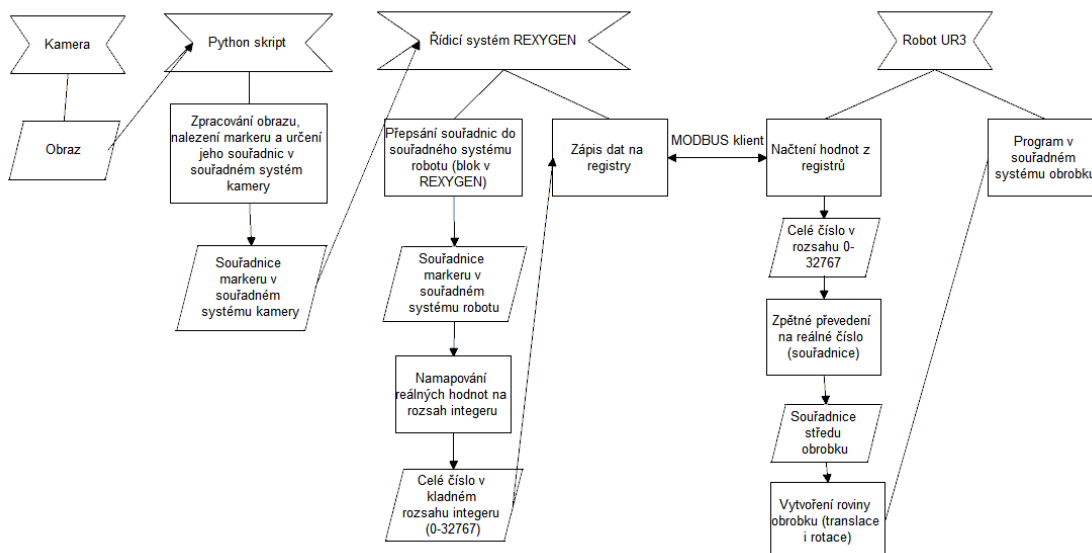
```

Obrázek 44: Podprogram Robots_program

Tento jednoduchý program byl testován pouze pro ručně zadané souřadnice v řídicím systému REXYGEN a vyzkoušen jen ve virtuálním simulátoru robotu.

5 Testování vytvořených programů na reálném robotu

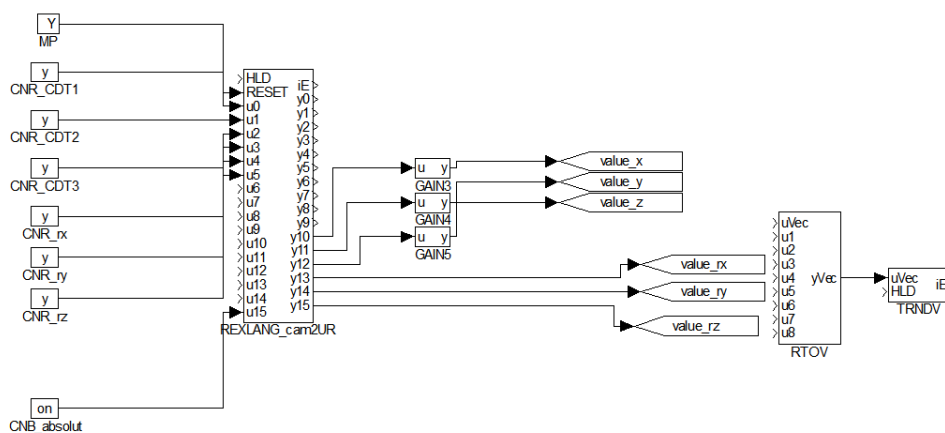
Pro testování vytvořených programů bylo zapotřebí získat translaci a rotaci obrobku v prostoru. V tomto případě byl obrobek reprezentován markerem (značkou). Tento marker je poté rozpoznán kamerou jejíž softwarovou část tvoří skript v programovacím jazyce Python. Jako výstup vrací informaci o translaci a rotaci daného markeru v souřadnicovém systému kamery, který je odlišný od toho robotického. Softwarové i hardwarové řešení kamerového systému již bylo připraveno a součástí této práce nebyl jejich návrh ani implementace.



Obrázek 45: Diagram popisující celý proces potřebný pro programování robotu v relativním souřadném systému obrobku

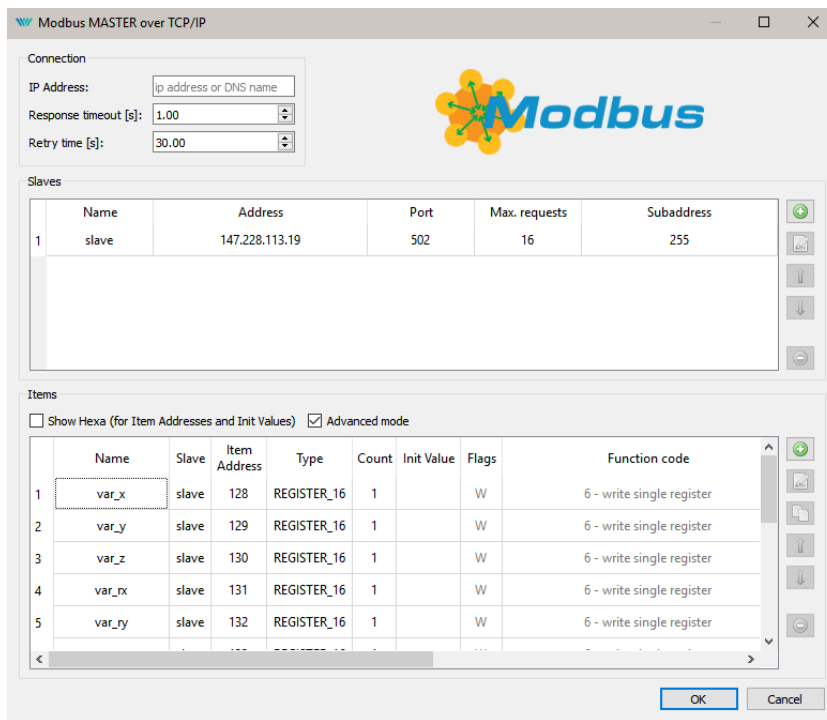
5.1 Změny v řídicím systému REXYGEN

V řídicím systému REXYGEN bylo zapotřebí přidání nového bloku, který řešil převod souřadnic mezi souřadnými systémy robotu a kamery.



Obrázek 46: Blok pro převod souřadnic

Pro připojení k reálnému robotu bylo potřeba změnit třídu ovladače z `MtsDrv` (slave) na `MtmDrv` (master). Robot poté plnil v komunikaci master-slave roli slave zařízení. V nastavení ovladače pak ještě byla přidána IP adresa cílového zařízení (adresa robotu - 147.228.113.19). Po prvním připojení k robotu bylo zjištěno, že adresy, které byly pro 16-bitové registry používány ve virtuálním simulátoru, jsou alokovány pro interní registry robotu. Pro programování byly uvolněny registry na adrese 128-255. Po změně adres již byla navázána komunikace mezi robotem a řídicím systémem REXYGEN.



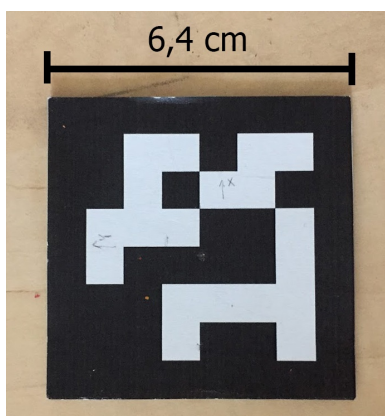
Obrázek 47: Konfigurace ovladače Modbus master

5.2 Změny v grafickém prostředí PolyScope

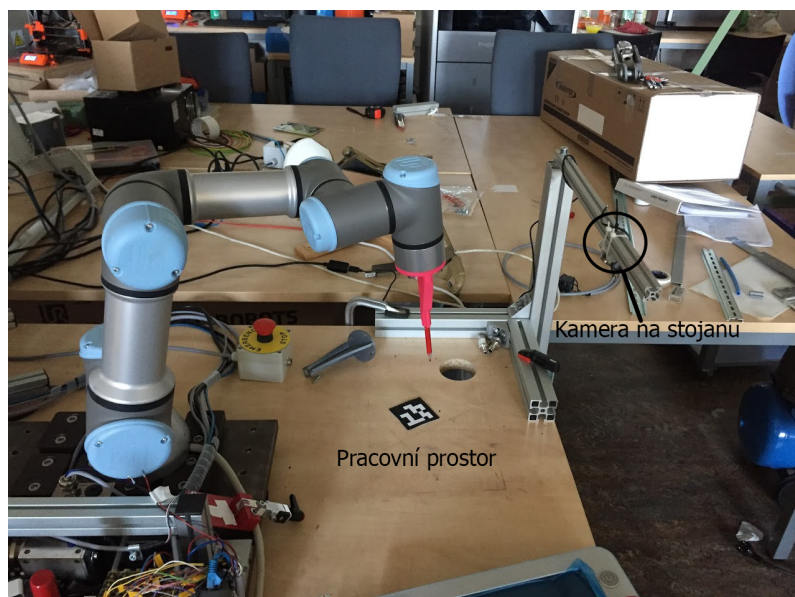
Oproti nastavení, které bylo použito při testech na virtuálním simulátoru, se změnilo pouze nastavení adres jednotlivých registrů. Tato změna byla způsobena alokováním dříve zvolených adres pro interní registry robotu.

5.3 Pracoviště

Jelikož byl robot v době testování používán katedrou kybernetiky i na jiné práce, byl pracovní prostor robotu limitován. Robot byl pevně připevněn se svou základnou k pracovnímu stolu v laboratoři. Kamera byla nainstalovaná na speciálně vytvořený stojan a byla zkalibrována přesně pro tuto aplikaci. Nástrojem robotu byla tužka. Marker byl vytištěn na papír o rozměrech 6,4 x 6,4cm.



Obrázek 48: Ukázka markeru, který je rozpoznáván kamerou v prostoru



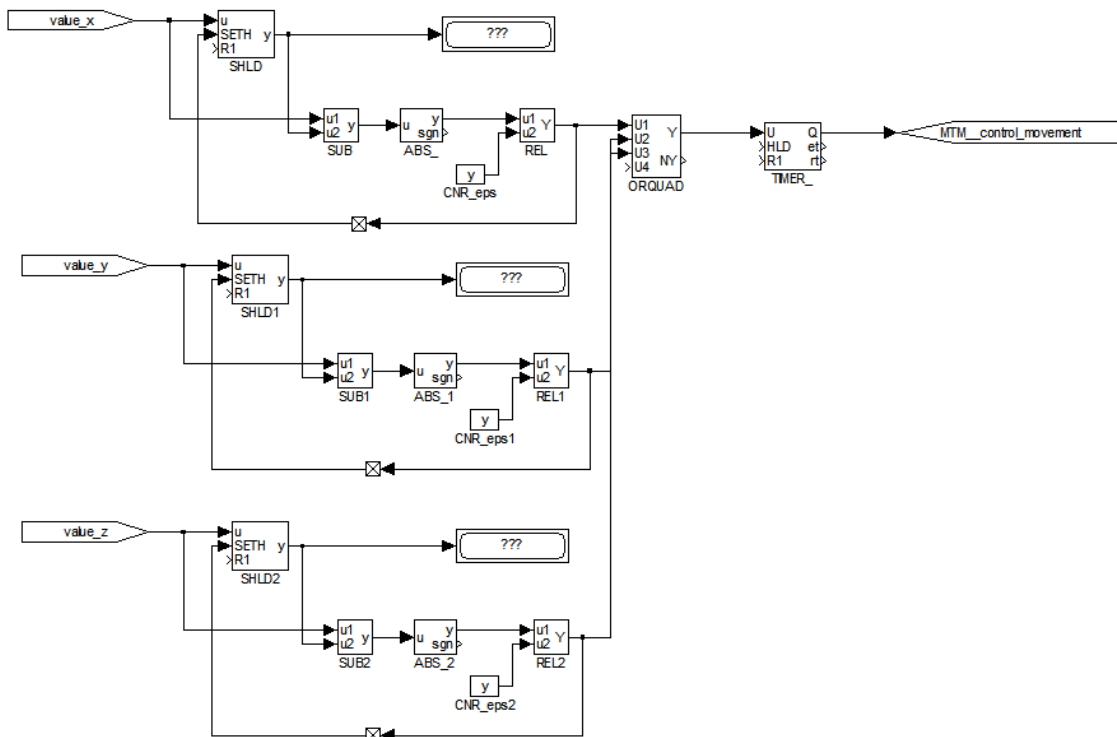
Obrázek 49: Ukázka pracoviště

5.4 Testování

Před testováním bylo zapotřebí změnit nastavení TCP (středový bod nástroje). Jelikož nástrojem robotu je tužka, muselo dojít ke změně středového bodu v ose Z (nastavení délky tužky). Všechny pohyby jsou poté robotem přepočteny vůči tomuto středovému bodu nástroje. Navíc bylo zjištěno, že je potřeba otočit souřadný systém nástroje podle osy X o 180° , aby byla shodná orientace rovin a nástroje.

Počátek souřadného systému obrobku, jehož souřadnice vrací nově přidáný blok v řídicím systému REXYGEN, je shodný se středem markeru. Pro názornou ukázkou byl vytvořen jednoduchý program, ve kterém robot projede všemi čtyřmi hranami markeru a opíše čtverec o rozměrech $6,4 \times 6,4\text{cm}$ (rozměry markeru). Při prvních pokusech byla zvolena pouze 2 sekundová pauza mezi dalším prováděním programu. Pro další pokusy byl poté vytvořen algoritmus v řídicím systému REXYGEN. Ten kontroluje zda-li se hodnota v jakékoli ose (x, y, z) změnila o více jak 2cm. Pokud ano, zapíše na kontrolní registr hodnotu 1. Robot tedy místo 2 vteřin čeká do chvíle dokud není na registru `control_movemen` hodnota 1 a provádí pohyb pouze při změně pozice markeru. Poté vyčká 2 vteřiny, aby se mohla kamera co nejlépe zkalibrovat a projede přes všechny hrany markeru.

Testováním byla ověřena funkčnost celého robotického systému a především funkčnost interakce s kamerou. Tento algoritmus je velice robustní a stačí tedy pouze změnit program, který má robot vykonat.



Obrázek 50: Algoritmus pro kontrolu změn v osách x,y,z

Závěr

Cílem této práce bylo nalézt možnost, jak zprovoznit nadřazený řídicí systém, který uživateli umožní programování robotu v relativním souřadném systému obrobku. Testy byly prováděny na robotu UR3 od společnosti Universal Robots.

V teoretické části práce byla nejprve shrnuta historie průmyslové robotiky. Dále jsou definovány základní pojmy týkající se zmíněných průmyslových robotů. Popsána jsou možná využití robotů v průmyslu, jejich dělení podle různých faktorů (tvar a velikost pracovního prostoru, typy řídicích systémů, trajektorie, atd.). V další části jsou definována všechna možná rizika, která mohou nastat při používání průmyslových robotů. Zároveň jsou shrnuty bezpečnostní prvky, které výskyt případných rizik mnohonásobně snižují. Tyto informace byly získány především z technických norem ISO 10218-1 a ISO 10218-2.

Další část je věnována kolaborativní robotice. Stručně popsána je především technická specifikace ISO/TS 15066, která poskytuje informace a doporučení věnované návrhu kolaborativních robotických systémů. Závěrem této části jsou řešení zajišťující bezpečnost u čtyř v průmyslu používaných kolaborativních robotů.

Před samotným návrhem robotického systému bylo zapotřebí zjistit možné způsoby reprezentace orientace tělesa v prostoru, v tomto případě obrobku. Této tematice se věnuje poslední kapitola v teoretické části.

V praktické části byl nejprve popsán řešený problém. Poté došlo k jednoduchému popisu grafického rozhraní PolyScope. Byly vytvořeny jednoduché příklady, které bylo možné testovat na virtuálním simulátoru. Před testováním na reálném robotu bylo nutné nalézt, jakým způsobem bude nadřazený řídicí systém komunikovat s robotem UR3. Byl zvolen komunikační protokol Modbus, který funguje na principu master-slave. Informace o souřadnicích markeru, které přicházely z kamery, byly zpracovány v řídicím systému REXYGEN a potřebné hodnoty byly zapisovány na 16-bitové registry. Pro ukázkou funkčnosti vybraného algoritmu byl vytvořen jednoduchý program, při kterém nástroj robotu (tužka) objede všechny rohy markeru.

Literatura

- [1] Guohua Cui, Dan Zhang and Marc A. Rosen *Robotic Safety: An Engineering Teaching Module*. March 2014 [online]: http://safetymanagementeducation.com/wp-content/uploads/2015/09/Robotics_Safety-Teaching_Module-Supplemental_Report_140730_FINAL_REV-with_logos.pdf
- [2] ČSN EN ISO 10218-1 Roboty a robotická zařízení -Požadavky na bezpečnost průmyslových robotů -Část 1: Roboty. Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2012.
- [3] ČSN EN ISO 10218-2 Roboty a robotická zařízení -Požadavky na bezpečnost průmyslových robotů -Část 2: Systémy robotů a integrace. Praha: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2011.
- [4] ISO/TS 15066 Robots and robotics devices –Colaborative robots. ISO, 2016.
- [5] Delta robot - Wikipedia [online]: https://en.wikipedia.org/wiki/Delta_robot
- [6] Havle Otto. Kolaborativní robotika Dočasná móda, nebo trvalý trend [online]. 2017-03-14: <http://www.controlengcesko.com/hlavni-menu/artiky/artiky/artiky/kolaborativni-robotika-docasna-moda-nebo-trvaly-trend/>
- [7] Bartošík Petr. Bezpečnost kolaborativních robotů [online]. Automa, 2017: http://www.automa.cz/Aton/FileRepository/pdf_articles/11040.pdf
- [8] Reichl Jaroslav, Všeticka Martin. Encyklopedie fyziky, popis rotace tuhého tělesa - Eulerovy kinematické rovnice [online]. 2010-09-26: <http://fyzika.jreichl.com/main.article/print/1298-popis-rotace-tuheho-telesa-eulerovy-kinematicke-rovnice>
- [9] Kvaternion - Wikipedia [online]: <https://cs.wikipedia.org/wiki/Kvaternion>
- [10] Smutný Vladimír. Robotika - Popis polohy tělesa [online]. Český institut informatiky, robotiky a kybernetiky: <http://cmp.felk.cvut.cz/cmp/courses/ROB/roblec/geometry-notecz.pdf>
- [11] Robotics Kinematics and Dynamics [online]: https://en.wikibooks.org/wiki/Robotics_Kinematics_and_Dynamics/Description_of_Position_and_Orientation
- [12] Robotika - Wikipedia [online]: <https://cs.wikipedia.org/wiki/Robotika>
- [13] Článek o budoucnosti robotiky ze stránek ABB [online]. 2015: https://library.e.abb.com/public/191cbbc9809f411aac23d1feeac1a9d0/yumi_backgrounder.pdf
- [14] Collaborative robot Cobotta - stránky výrobce [online]: <https://www.denso-wave.com/en/robot/product/collabo/cobotta.html>
- [15] Techman: TM5 - stránky výrobce [online]: <http://tm-robot.com/TM5.php>

- [16] Roberta Nelson Shea. Collaborative Robot Technical Specification ISO/TS 15066 Update [online]. Rockwell Automation:
<http://me.umn.edu/courses/me5286/robotlab/Resources/12-TR150660verview-SafetyforCollaborativeApplications-RobertaNelsonShea.pdf>
- [17] Robotics tech papers - ISO/TS 15066 Explained [online]:
https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Tech-Papers/ISO-TS-15066-Explained/content_id/6084
- [18] Ronešová Andrea, Ing. Přehled protokolu MODBUS [online]. 2005:
<http://home.zcu.cz/~ronesova/bastl/files/modbus.pdf>
- [19] RexygenStudio - Uživatelská příručka [online]. REX Controls s.r.o. 2018-03-28:
https://www.rexygen.com/doc/PDF/CZECH/RexygenStudio_CZ.pdf
- [20] Script manual - The URScript Programming Language, version 1.8 [online]. 2013-08-01:
http://www.sysaxes.com/wp-content/uploads/2013/12/scriptmanual_en.pdf
- [21] Internetová jazyková příručka [online]: <http://prirucka.ujc.cas.cz>
- [22] ČSN 01 6910: Úprava dokumentů zpracovaných textovými procesory (Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, www.unmz.cz)

Zdroje obrázků

- [23] Articulated robot - Wikipedia [online]:
https://en.wikipedia.org/wiki/Articulated_robot
- [24] Transformace souřadného systému [online] <http://www.mefi.cz/files/dokumentace/cnc8x9/navodkprogramovani/kapitola14.pdf>
- [25] Guohua Cui, Dan Zhang and Marc A.Rosen *Robotic Safety: An Engineering Teaching Module*. March 2014 [online]: http://safetymanagementeducation.com/wp-content/uploads/2015/09/Robotics_Safety-Teaching_Module-Supplemental_Report_140730_FINAL_REV-with_logos.pdf
- [26] Your questions answered: robots for extreme environments.
<https://www.theengineer.co.uk/your-questions-answered-robots-for-extreme-environments/>
- [27] Stránky výrobce ABB:
<http://new.abb.com/products/robotics/cs/prumyslove-roboty/yumi>
- [28] Bartošík Petr. Bezpečnost kolaborativních robotů [online]. Automa, 2017:
http://www.automa.cz/Aton/FileRepository/pdf_articles/11040.pdf
- [29] Cobots guide [online] :<https://cobotsguide.com/2016/06/denso-cobotta/>
- [30] Epo Machinery - stránky prodejce [online]:
<http://www.epomachinery.cz/cs/stranka/distribuce-universal-robots>
- [31] Ronešová Andrea, Ing. Přehled protokolu MODBUS [online]. 2005:
<http://home.zcu.cz/~ronesova/bast1/files/modbus.pdf>
- [32] Smutný Vladimír. Robotika - Popis polohy tělesa [online]. Český institut informatiky, robotiky a kybernetiky:
<http://cmp.felk.cvut.cz/cmp/courses/ROB/roblec/geometry-notecz.pdf>
- [33] Technical Paper, PLCopen Technical Committee 2 - Task Force, Function Blocks for motion control: Part 4-Coordinated Motion, PLCopen, 2008.
- [34] Roberta Nelson Shea. Collaborative Robot Technical Specification ISO/TS 15066 Update [online]. Rockwell Automation:
<http://me.umn.edu/courses/me5286/robotlab/Resources/12-TR150660verview-SafetyforCollaborativeApplications-RobertaNelsonShea.pdf>
- [35] Robotics tech papers - ISO/TS 15066 Explained [online]:
https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Tech-Papers/ISO-TS-15066-Explained/content_id/6084
- [36] Rigid body dynamics - Wikipedia[online]: https://en.wikipedia.org/wiki/Rigid_body_dynamics#Orientation_or_attitude_descriptions body

[37] Zlín robotics, stránky prodejce [online]: <http://www.zlinrobotics.com/>