

Průmyslový robot Stäubli UNIMATION RS40B – uživatelský manuál

Industrial robot Stäubli UNIMATION RS40B – user's manual

Martin Tomášek



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2010/2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Martin TOMÁŠEK**
Osobní číslo: **A07111**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Průmyslový robot Stäubli UNIMATION RS40B –
uživatelský manuál**

Zásady pro vypracování:

1. Provedte literární rešerši na zadané téma.
2. V teoretické části práce popište průmyslového robota Stäubli UNIMATION RS40B a základy jeho ovládání.
3. Vytvořte elektronický manuál. Manuál bude sloužit pro výukové účely a bude obsahovat popis ovládání průmyslového robota pomocí ručního programovacího panelu (MPC), základy práce s vývojovým prostředím VAL3 studio, popis emulátoru CS8C a popis jazyka VAL3.
4. Na vhodně zvolených příkladech demonstруйте možnosti robota.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: tištěná/elektronická

Seznam odborné literatury:

1. JAZAR, Reza N. Theory of applied robotics : kinematics, dynamics, and control. New York : Springer, 2007. 693 s. ISBN 978-0-387-68964-7.
2. GHOSAL, Ashitava. Robotics : fundamental concepts and analysis. 1.pub. New Dehli : Oxford University Press, 2006. 423 s. ISBN 0195673913.
3. HAVEL, Ivan M. Robotika : úvod do teorie kognitivních robotů. 1. vyd. Praha : SNTL, 1980. 279 s.
4. KOSEK, Jiří. HTML : tvorba dokonalých www stránek. 1. vyd. Praha : Grada, 1998. 291 s. ISBN 8071696080.
5. CHOSET, Howie M. Principles of robot motion : theory, algorithms, and implementation. Cambridge, Mass. : MIT Press, 2005. 603 s. ISBN 0-262-03327-5.
6. KARGER, Adolf; KARGEROVÁ, Marie. Základy robotiky a prostorové kinematiky. Vyd. 1. Praha : ČVUT, 2000. 265 s. ISBN 80-01-02183-1.

Vedoucí bakalářské práce:

Ing. Petr Navrátil, Ph.D.

Ústav řízení procesů

Datum zadání bakalářské práce:

25. února 2011

Termín odevzdání bakalářské práce:

7. června 2011

Ve Zlíně dne 25. února 2011

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Bakalářská práce se zabývá průmyslovým robotem Stäubli UNIMATION RS40B. Teoretická část je zaměřena na historii průmyslových robotů, jejich rozdělení podle struktury, pohybové jednotky a podle druhu pohonu. Obsahuje také popis robotického ramene RS40B, jeho ovládání pomocí panelu MCP a popis programového vybavení. Praktickou část tvoří HTML stránky s popisem zařízení a ovládání robotického ramene. Dále obsahuje manuál VAL3 instrukcí, vytváření aplikací a názorné příklady.

Klíčová slova: průmyslový robot, Stäubli, manuál, VAL3, MCP

ABSTRACT

Bachelor thesis deals with an industrial robot Stäubli Unimation RS40B. The theoretical part of thesis focuses on the history of industrial robots, their distribution according to a structure, movement units and a type of drive. It also contains a description of the robotic arm RS40B, its control by using the MCP and a description of the software. The practical part consists of an HTML pages, describing the facilities and control a robotic arm. It also contains manual of VAL3 instructions, creating applications and examples.

Keywords: industrial robot, Stäubli , manual, VAL3, MCP

Na tomto místě bych chtěl poděkovat Ing. Petru Navrátilovi Ph.D., mému vedoucímu práce, za jeho ochotu a obětavost. Dále pak všem přátelům, kteří mě psychicky podporovali v průběhu celého bakalářského studia.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně 3.6.2011

.....
Martin Tomášek

OBSAH

ÚVOD.....	9
1 TEORETICKÁ ČÁST	11
1 HISTORIE	12
2 POHYBOVÉ STRUKTURY PRŮMYSLOVÝCH ROBOTŮ	15
2.1 KARTÉZSKÁ POHYBOVÁ STRUKTURA.....	15
2.2 CYLINDRICKÁ POHYBOVÁ STRUKTURA.....	16
2.3 SFÉRICKÁ POHYBOVÁ STRUKTURA	17
2.4 KLOUBOVÁ POHYBOVÁ STRUKTURA.....	18
2.5 SCARA.....	19
3 POHYBOVÉ JEDNOTKY PRŮMYSLOVÝCH ROBOTŮ	20
3.1 TRANSLAČNÍ JEDNOTKA	20
3.1.1 Koncepce s pohyblivými saněmi.....	20
3.1.2 Koncepce s teleskopickým ramenem	21
3.2 ROTAČNÍ JEDNOTKA	22
4 POHYBOVÉ JEDNOTKY PRŮMYSLOVÝCH ROBOTŮ	23
4.1 ELEKTRICKÝ POHON	23
4.2 PNEUMATICKÝ A HYDRAULICKÝ POHON	24
5 POPIS ROBOTU STÄUBLI RS40B.....	26
5.1 ROBOTICKÉ RAMENO RS40B.....	26
5.1.1 Pracovní rozsah, rychlost a nosnost	27
5.2 KONTROLÉR CS8C.....	28
5.3 RUČNÍ PROGRAMOVACÍ PANEL MCP.....	29
5.3.1 Popis MCP panelu.....	30
5.3.1.1 Volba pracovního módu (1).....	30
5.3.1.2 Tlačítko napájení motorů (2)	31
5.3.1.3 Nouzový vypínač (3).....	31
5.3.1.4 Tlačítka pro navigaci a pohyb (4)	31
5.3.1.5 Tlačítka pro volbu režimu pohybu (5)	31
5.3.1.6 Nastavení rychlosti pohybu (6)	32
5.3.1.7 Funkční klávesy (7).....	33
5.3.1.8 Alfnumerická klávesnice (8).....	33
5.3.1.9 Navigační klávesy a interface (9).....	33
5.3.1.10 Řízení aplikace (10)	34
5.3.1.11 Aktivační spínač „Dead men“ (11)	34
5.3.1.12 Rychlé ovládání I/O (12).....	35
5.3.1.13 Tlačítka Jog (13)	35
5.4 PANEL PRO VÝBĚR PRACOVNÍHO REŽIMU	35
6 PROGRAMOVÉ VYBAVENÍ.....	37

6.1	CELLS MANAGER	37
6.2	VAL3 STUDIO	38
6.3	CS8 EMULÁTOR	38
6.4	3D STUDIO	38
6.5	TRANSFER MANAGER.....	38
6.6	CS8 REMOTE MAINTENANCE	38
6.7	PLC STUDIO	38
7	PROGRAMOVACÍ JAZYK VAL3.....	39
7.1	APLIKACE.....	39
7.1.1	Programy	40
7.1.2	Datové typy	40
7.1.2.1	Jednoduché datové typy	40
7.1.2.2	Strukturované datové typy	41
II	PRAKTICKÁ ČÁST	42
8	POPIS HTML MANUÁLU	43
8.1	ÚVOD	44
8.2	MANUÁL	44
8.3	PŘÍKLADY.....	44
9	TVORBA HTML MANUÁLU.....	45
9.1	VÝVOJOVÝ PROGRAM PSPAD EDITOR	45
10	ZADÁNÍ PŘÍKLADŮ	46
10.1	PŘÍKLAD 1	46
10.2	PŘÍKLAD 2	47
10.3	PŘÍKLAD 3	48
10.4	PŘÍKLAD 4	49
	ZÁVĚR	51
	ZÁVĚR V ANGLIČTINĚ	52
	SEZNAM POUŽITÉ LITERATURY	53
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	54
	SEZNAM OBRÁZKŮ	55
	SEZNAM TABULEK.....	57
	SEZNAM PŘÍLOH.....	58

ÚVOD

Průmyslový robot je manipulátor určený k přemísťování materiálů, nástrojů a dílů, a k provádění řady naprogramovaných úkolů v oblasti výroby a produkce. Často jsou používány k činnostem, které jsou nebezpečné či nevhodné pro zaměstnance. Jsou ideální pro výroby, kde je vyžadován vysoký pracovní výkon a žádné chyby.[2]



Obrázek 1: Rameno určené ke svařování

Mezi typické činnosti průmyslových robotů patří svařování, lakování, montáž, přemísťování předmětů, balení a paletizace a kontroly výrobků. To vše s vysokou rychlostí a přesností. Ačkoliv se dají přeprogramovat na nejrůznější činnosti, často jsou naprogramovány při výrobě a pak opakují léta pořád stejnou činnost (zejména v automobilovém průmyslu). Průmyslový robot se tak stává neodmyslitelnou součástí moderních továren. [2]



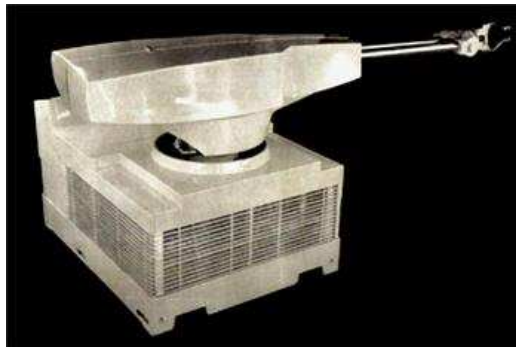
Obrázek 2: Umístění průmyslových robotů v automobilovém průmyslu

Cílem této práce je seznámit studenty s průmyslovým robotem Stäubli RS40B, jeho ovládáním pomocí ručního programovacího panelu (MPC), základy práce s vývojovým prostředím VAL3 studio, popis emulátoru CS8C, popis jazyka VAL3 a v neposlední řadě ukázky na výukových příkladech pro demonstraci jeho možností.

I. TEORETICKÁ ČÁST

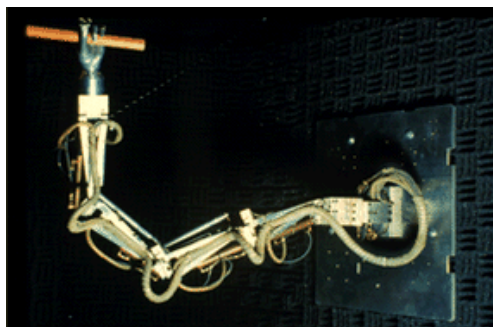
1 HISTORIE

George Charles Devol je považován za otce robotiky. V roce 1954 vynalezl prvního průmyslového robota jménem Unimate. O několik let později, Devol a Joseph F. Engelberfer, založili první společnost zabývající se roboty jménem Unimation. Jejich hlavní použití byl zejména přenos nástrojů z jednoho místa na druhé. [3]



*Obrázek 3: První průmyslový robot
Unimate*

V 70. a 80. letech 20. Století byly nadále vyvíjeny moderní průmyslové roboty. V roce 1963 byl vyroben šesti osý robot pro pomoc s péčí o postižené. V roce 1968, Marvin Minsky sestavil „chapadlové“ rameno, jež mělo 12 kloubů a dokázalo uzvednout člověka. [3]



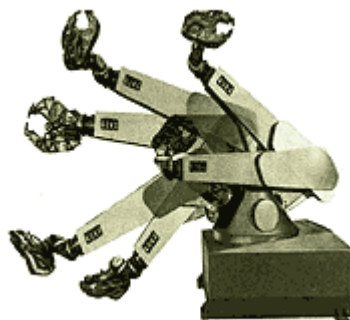
*Obrázek 4: Chapadlové rameno
Marvina Minsky*

V roce 1969 sestrojil Victor Scheinman první plně elektrické, šesti osé, kloubového robota, který byl konstruován jako lidská paže. To zajistilo lepší pohyblivost v prostoru a rozšířily se možnosti použití robota k náročnějším úkonům jako montáž a svařování. Scheinman poté dále vyvíjel roboty pro firmu Unimation, které později prodávali pod jménem PUMA (programovatelný universální stroj pro montáž). [3]



Obrázek 5: Unimation PUMA

Průmyslové roboty se poměrně rychle rozšířily po Evropě. V roce 1973 představila společnost ABB Robotics (dříve ASEA) průmyslového robota IRB 6, který byl první celosvětově rozšířený, plně elektrický a mikroprocesorem řízený robot. První dva roboty IRB 6 prodala firma ve Švédsku, kde sloužily k leštění broušení a ohýbání trubek. Také firma KUKA Robotics představila v roce 1973 robota FAMULUS, který je také jeden z prvních robotů se šesti elektromechanicky poháněnými klouby. [3]



Obrázek 6: Robot FAMULUS

V roce 1984 byla firma Unimation odkoupena společností Westinghouse Electric Corporation za 107 miliónů amerických dolarů. Westinghouse však v roce 1989 prodal firmu Inumation francouzské firmě Stäubli, která dodnes vyrábí kloubové roboty pro průmyslové použití a dokonce v roce 2004 odkoupila robotickou divizi firmy Bosh. [3]



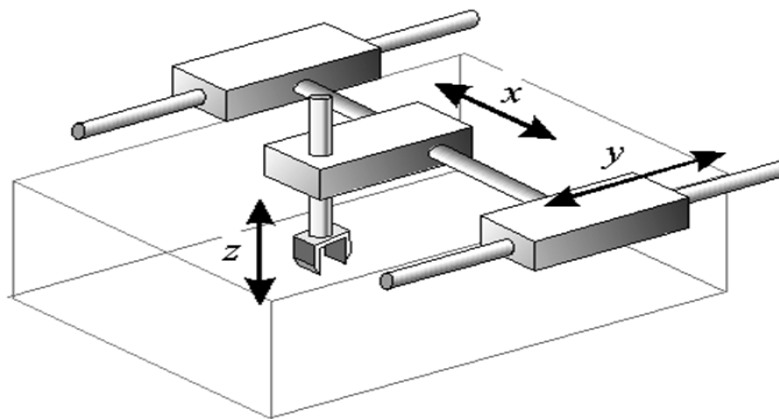
*Obrázek 7: Robotické rameno RX170
od firmy Staubli*

2 POHYBOVÉ STRUKTURY PRŮMYSLOVÝCH ROBOTŮ

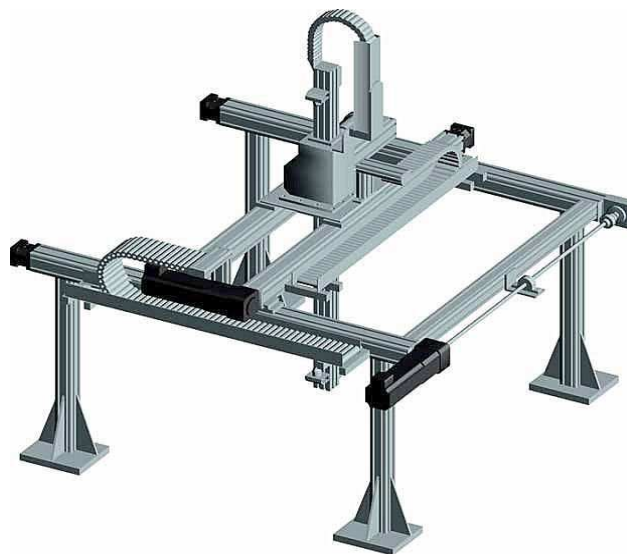
Průmyslové roboty lze rozdělit do pěti základních struktur: kartézské, cylindrické, sférické, angulární a SCARA. Klasifikace podle geometrie pracovního prostoru. [12]

2.1 Kartézská pohybová struktura

- Nedochozí ke změně orientace objektu
- Pohybuje se přímo v osách X, Y nebo Z
- Nejpřesnější pohybová struktura
- Větší prostorové požadavky
- Použití například jako sázecí stroje
- Možnost výběru požadovaných os (například jen X a Y)



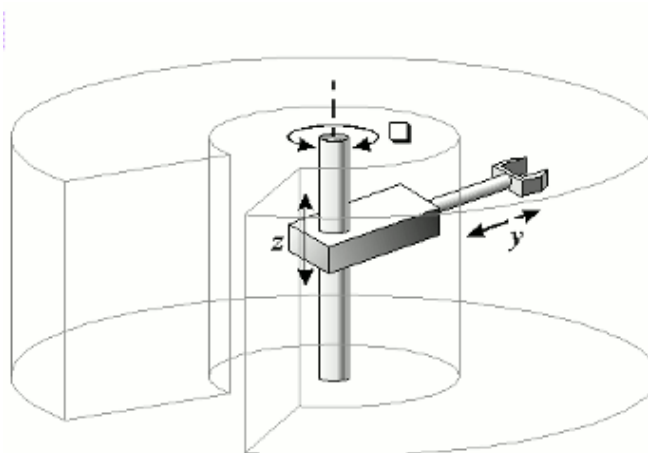
Obrázek 8: Princip pohybu v kartézské pohybové struktuře



Obrázek 9: Kartézský robot

2.2 Cylindrická pohybová struktura

- Dochází ke změně orientace objektu
- Pracovní prostor je válcový prstenec
- Jeden rotační pohyb a dva posuvné
- Obdoba robotu SCARA



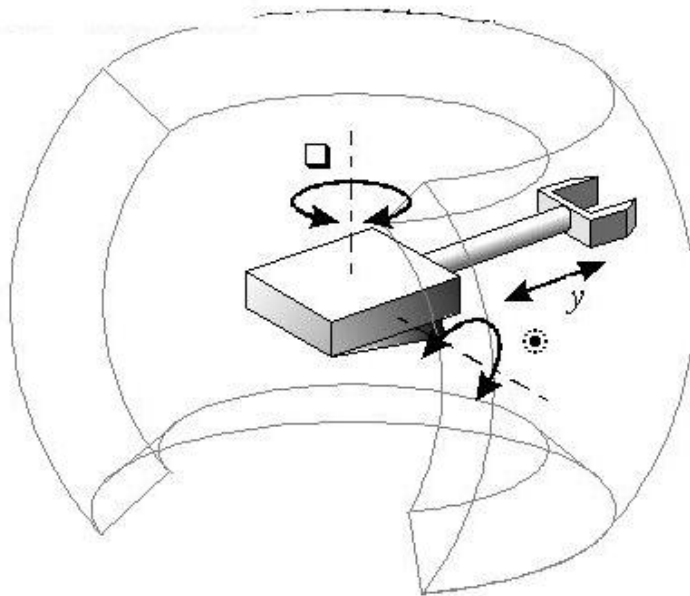
Obrázek 10: Cylindrická pohybová struktura



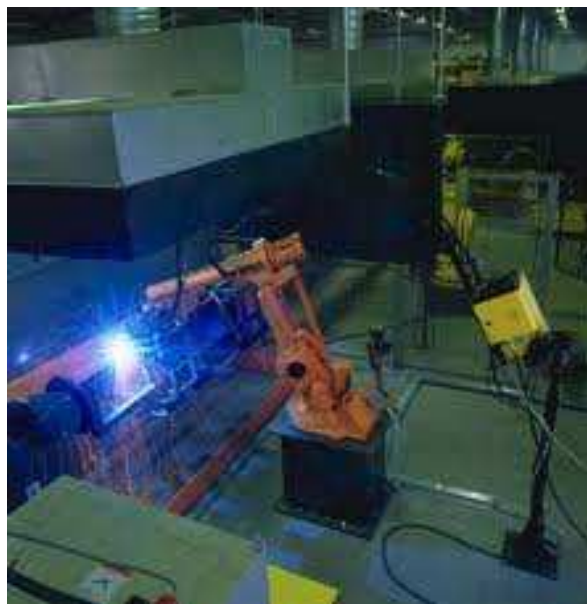
Obrázek 11: Cylindrický robot Denso

2.3 Sfěrická pohybová struktura

- Dochází ke změně orientace objektu
- Pohyb ve sférických souřadnicích
- Pracovní prostor je kulový vrchlík
- 2 rotační a jeden posuvný pohyb
- Použití pro svařování



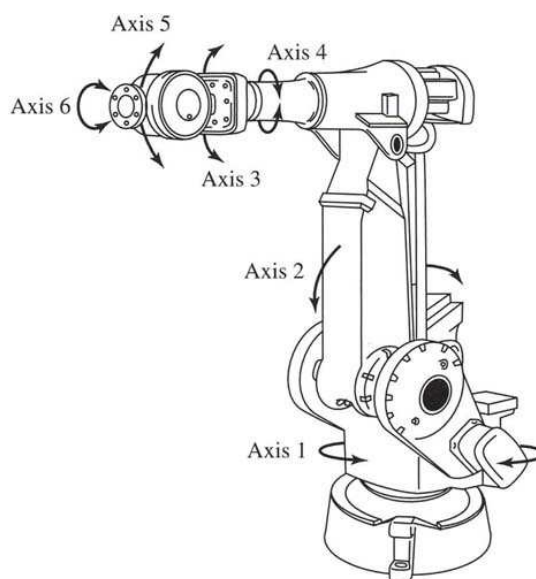
Obrázek 12: Sfěrická pohybová struktura



Obrázek 13: Sfěrický svařovací robot

2.4 Kloubová pohybová struktura

- Dochází ke změně orientace objektu
- V současné době nejrozšířenější průmyslový robot
- Podobnost lidským ramenem
- Universální využití
- Možnost rotace objektu kolem všech 3 os
- Velká volnost pohybu



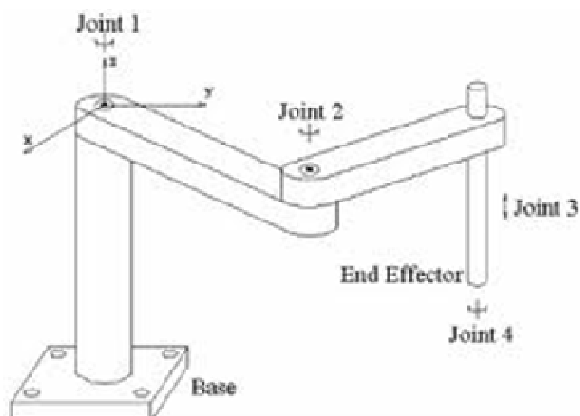
Obrázek 14: Kloubová pohybová struktura



Obrázek 15: Kloubový robot Stäubli RX260

2.5 SCARA

- Dochází ke změně orientace objektu
- 3 rotační a jeden posuvný pohyb
- Vysoká provozní rychlost



Obrázek 16: Struktura SCARA



Obrázek 17: Průmyslový robot Stäubli RS60

3 POHYBOVÉ JEDNOTKY PRŮMYSLOVÝCH ROBOTŮ

Podle konstrukčního hlediska lze rozdělit pohybové jednotky na dva základní druhy:

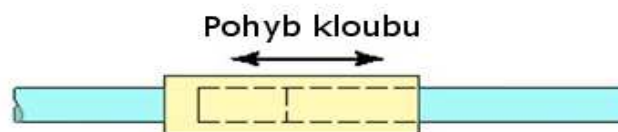
- a) Translační – vykoná přímočarý pohyb
- b) Rotační – vykonává rotační pohyb

3.1 Translační jednotka

Translační pohyb nebo také lineární pohyb je veden podél jedné z os bez jakéhokoliv rotačního pohybu. Ukázkovým příkladem je pohyb kartézského robota. Nejčastěji se vyskytuje v teleskopickém, smykadlovém nebo suportovém provedení.[12]

3.1.1 Koncepce s pohyblivými saněmi

Slouží k přesunu celého průmyslového robota v řádech metrů.



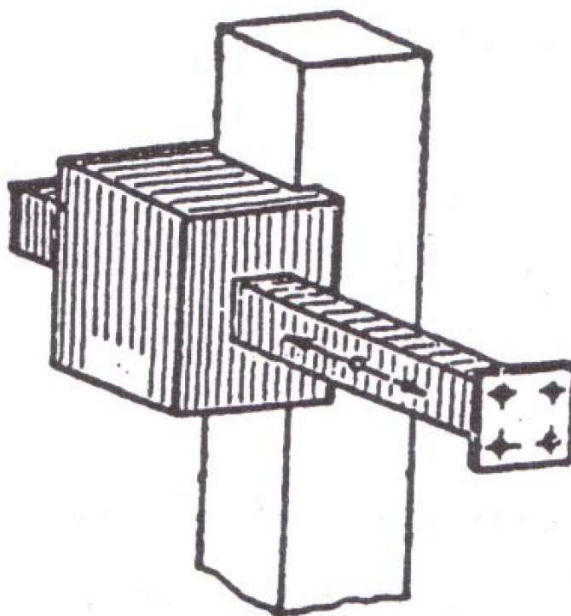
Obrázek 18: Princip pohybu pohyblivými saněmi



Obrázek 19: Průmyslový robot KUKA Jet

3.1.2 Koncepce s teleskopickým ramenem

Slouží k přesunu ramene do určité polohy. Používá se především pro jednoduché manipulátory.



Obrázek 20: Princip pohybu teleskopickým ramenem



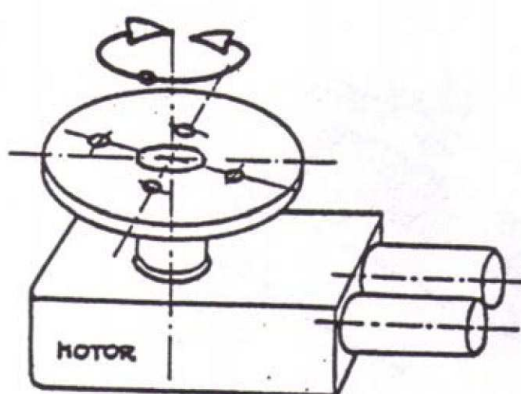
Obrázek 21: Těžká technika s teleskopickým ramenem

3.2 Rotační jednotka

Jednotka vykonává rotační pohyb. Umožňuje otáčení o více než 360° .

a) Jednotka s přímým pohonem

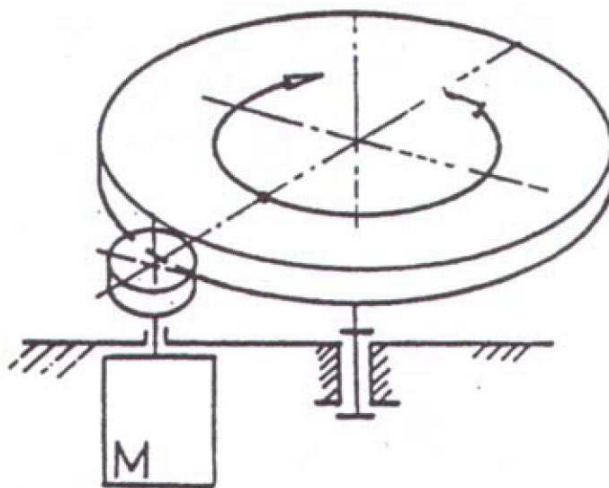
- Motor vykonává přímo rotační pohyb



Obrázek 22: Jednotka s přímým pohonem

b) Jednotka s nepřímým motorem

Motor je samostatný celek, který je spojen přímo s otáčivou jednotkou za pomoci některého typu převodu. Například ozubenými nebo řetězovými koly či řemenem.



Obrázek 23: Jednotka s nepřímým pohonem

4 POHYBOVÉ JEDNOTKY PRŮMYSLOVÝCH ROBOTŮ

K pohonu slouží motor, který nastavuje jednotlivé části robotu do požadovaných pozic. Tyto části jsou navzájem propojeny klouby. Průmyslové roboty obecně používají 3 základní druhy pohonu: elektrické, hydraulické nebo pneumatické.[5]

4.1 Elektrický pohon

V současnosti se jedná o nejpoužívanější druh pohonu průmyslových robotů, zejména u robotů s nosností ve střední třídě.

Výhody elektrického pohonu jsou snadno dostupné zdroje energie, jednoduchost vedení zdroje k motoru, jednoduché propojení s řídicími prvky, čistší provoz, nižší hlučnost oproti hydraulickým pohonům a také menší nároky na chlazení a na celkový instalovaný prostor. Velkou výhodou jsou také pořizovací a provozní náklady, které nejsou oproti jiným pohonům nižší. Také odpadá nutnost transformace energie, což má žádaný vliv na vysokou účinnost stroje. [4]

Mezi nevýhody patří nutnost použití vhodných transformačních bloků z důvodů vysokých otáček rotačních elektromotorů, složitější realizaci přímočarého pohybu, závislost na dodávce elektrické energie a značné požadavky na kvalitu provedení všech částí mnohdy složitých systémů. Také je nutné dbát zvýšených bezpečnostních předpisů z důvodu nebezpečí úrazu elektrickým proudem. [13]



Obrázek 24: Krokový motor

4.2 Pneumatický a hydraulický pohon

Pneumatický a hydraulický (tekutinový) pohon se v konstrukcích manipulátorů a robotů uplatňuje ve dvou hlavních oblastech. Hydraulický pohon v zařízeních především větších výkonů a to jak se spjitým řízením pohonu, tak i při realizaci jednoduchých pohybových funkcí. Pneumatický pohon je zajímavý pro konstrukce jednodušších manipulátorů s menší nosností a periferních prvků a zařízení automatizovaných pracovišť. [13]

Hydraulické i pneumatické pohony pracují se stejným druhem média s tekutinou. Z rozdílných vlastností kapalin a plynů se na rozdílných vlastnostech mechanismů podílí především různá poddajnost a viskozita. Jako pracovní kapaliny se v hydraulických mechanismech používají minerální oleje, pracovním médiem pneumatického pohonu je stlačený vzduch. [5]

U hydraulických pohonů se projevují v porovnání s pneumatickými pohony tyto přednosti:

- velká tuhost
- plynulý chod, možnost dosažení i malých rychlostí pohybů bez převodů a s velmi dobrou rovnoměrností
- velká účinnost

Nedostatkem hydraulického pohonu je:

- potřeba samostatného, odděleného energetického bloku
- poměrně obtížné dosažení vyšších pohybových rychlostí
- závislost viskozity kapaliny na teplotě, což se projevuje ve změně tlakových poměrů a případně i rychlosti pohybu motoru
- hořlavost některých druhů pracovních kapalin

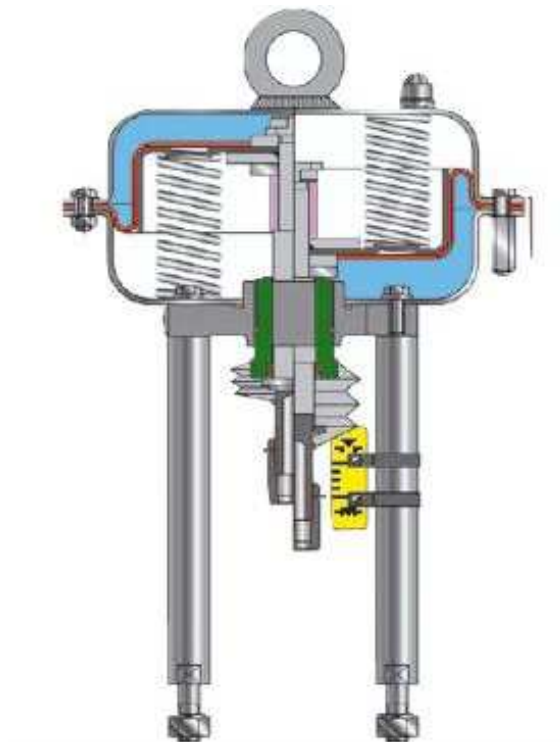
Přednosti pneumatického pohonu jsou:

- možnost připojení na centrální rozvod stlačeného vzduchu v rámci pohonu jednoduchý rozvod bez zpětného odvádění vzduchu z motoru
- možnost dosažení rychlých přímočarých pohybů
- možnost činnosti ve velkém tepelném rozsahu, ve výbušném prostředí a v provozech s nebezpečím vznícení od otevřeného ohně

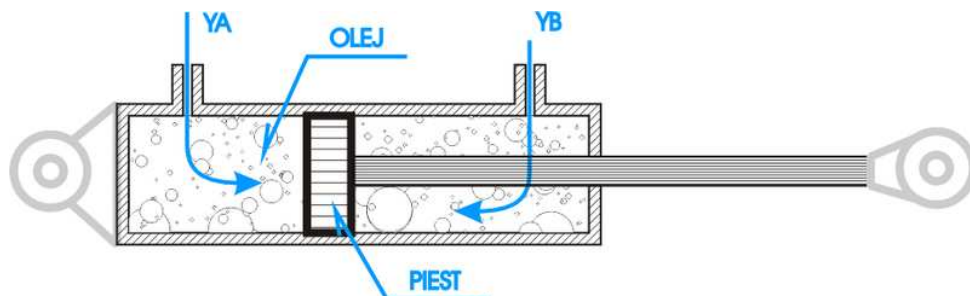
Nedostatky pneumatického pohonu:

- obtížné udržování rovnoměrného pohybu, zejména při malých rychlostech,
- poměrně komplikované mazání pohyblivých částí prvků mechanismu,
- poměrně drahý provoz, výroba stlačeného vzduchu je $(6 - 8) \cdot$ dražší než výroba elektrického proudu a asi $4 \cdot$ dražší než výroba tlakové kapaliny.

S ohledem na uvedené skutečnosti lze říci, že použití pneumatického pohonu je účelné u manipulátorů menších výkonů, jednodušších pracovních cyklů, kde se vystačí s nastavováním polohy na pevné dorazy a kde nevádí obtížné řízení rychlosti pohybu a jeho nerovnoměrnost.[5]



Obrázek 25: Pneumatický motor P1/R1



Obrázek 26: Hydraulický válec

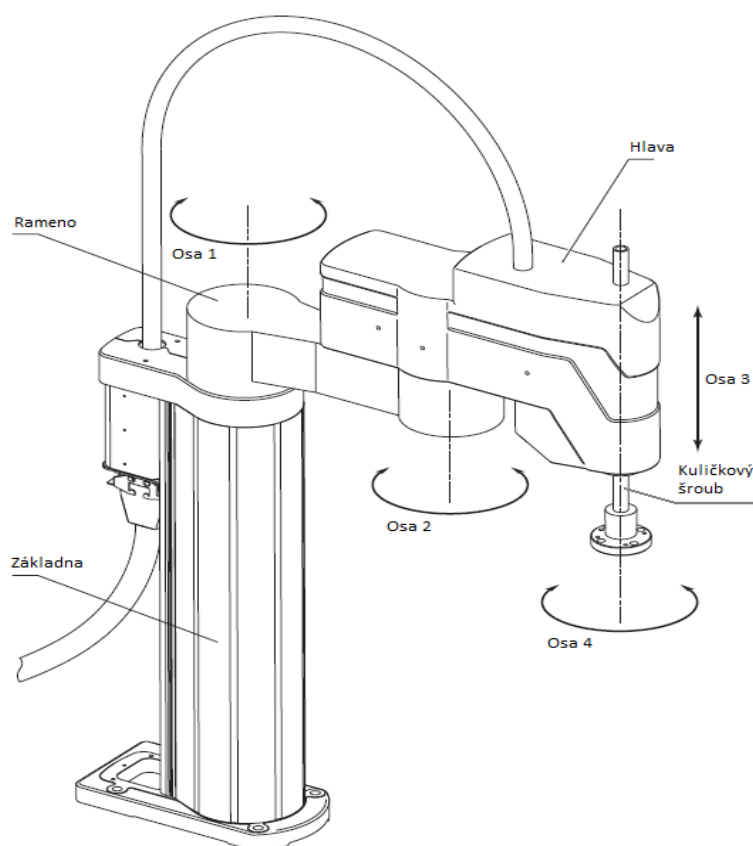
5 POPIS ROBOTU STÄUBLI RS40B

Rameno Stäubli RS40B se skládá ze čtyř základních částí: robotického ramene, kontroléru CS8C, ručního programovacího panelu MCP a z panelu pro výběr pracovního režimu.

5.1 Robotické rameno RS40B

Robot Stäubli RS40B, je 4-osý průmyslový robot typu SCARA, což znamená, že může přesunovat předměty na jakékoliv místo v X-Y-Z souřadnicích, ve svém pracovním rozsahu. První 3 osy slouží pro pohyb v horizontálním rozsahu a čtvrtá osa zajišťuje pohyb svislý, nezávisle na poloze ostatních kloubů. [10]

Výhodou průmyslových robotů typu SCARA je jejich podobnost s lidskou rukou. Podstavec nevyžaduje mnoho prostoru a umožňuje snadnou montáž, proto je lze umístit i do uzavřených prostor, kde nezaberou tolik místa jako kartézský robot a lze je umístit i na stropy či zeď. SCARA je obecně rychlejší než srovnatelné kartézské systémy. Nevýhodou je ovšem vyšší pořizovací cena. [10]



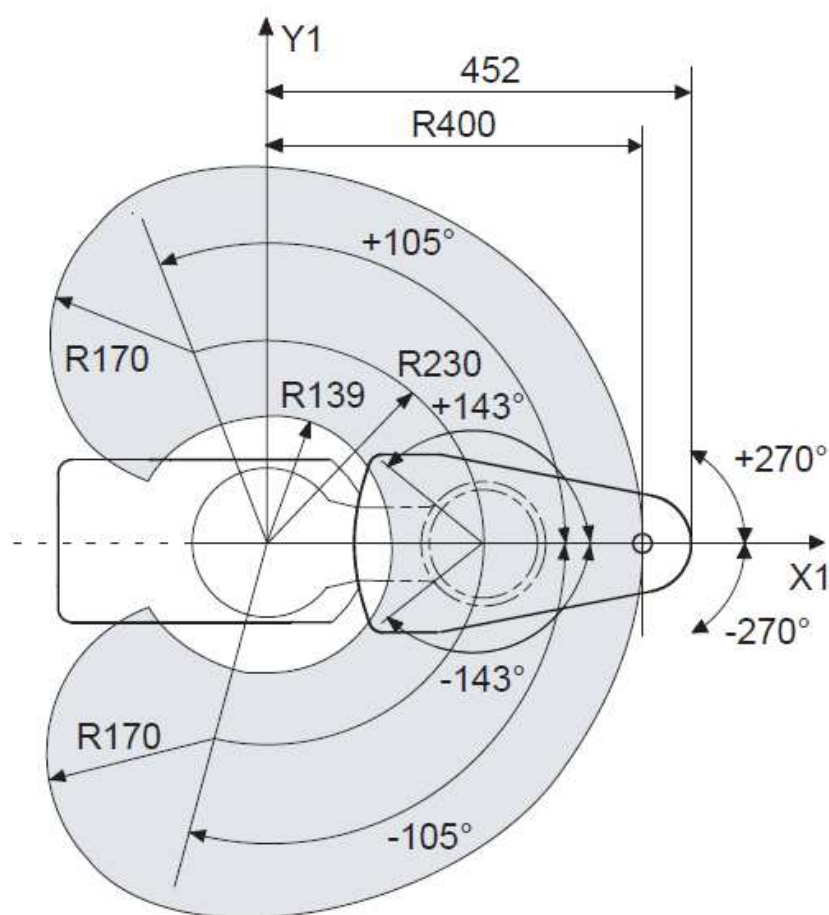
Obrázek 27: Robotické rameno RS40B

Paže se skládá ze 4 částí, které jsou vzájemně propojeny klouby. Pohyby ramene jsou generovány servomotory, které jsou na osách 3 a 4 vybaveny brzdou.

Jednotlivé části paže jsou:

- Základna
- Rameno
- Hlava
- Kuličkový šroub

5.1.1 Pracovní rozsah, rychlost a nosnost



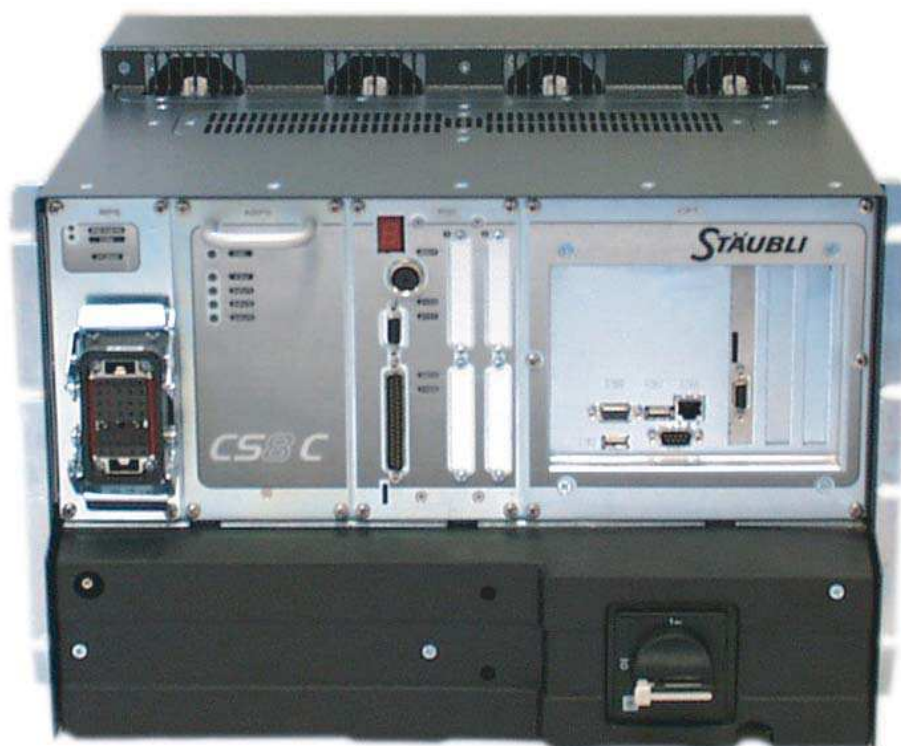
Obrázek 28: Pracovní rozsah robotického ramene

Tabulka 1 : Pracovní rozsahy

Rozsah (mm)	400
Vnitřní radius (mm)	139
Úhlový rozsah (osa 1) (°)	± 105
Úhlový rozsah (osa 2) (°)	± 143
Tah (mm)	200
Úhlový rozsah (osa 4) (°)	± 500

5.2 Kontrolér CS8C

Kontrolér CS8C je srdcem a mozkiem celého robotu. Přímá příkazy z uživatelského rozhraní a ze senzorů robotického ramene. Pomocí těchto údajů za pomoci programovacích instrukcí řídí kontrolér veškerou činnost robotu. Pro vstupní a výstupní komunikaci používá kontrolér CS8C digitální vstupy a výstupy, sériový port a usb porty. Také je možné připojit další vstupy a výstupy, které dále zvyšují možnosti robotu. [10]

*Obrázek 29: Kontrolér CS8C*

5.3 Ruční programovací panel MCP

Manuální programovací panel MCP slouží k manuálnímu ovládání ramene, spouštění a zastavování aplikací, a ke komunikaci s aplikacemi prostřednictvím uživatelského rozhraní. Obsahuje také nástroje pro tvorbu vlastních aplikací, které lze psát přímo prostřednictvím MCP panelu, který disponuje LCD displayem a alfanumerickou klávesnicí pro psaní instrukcí.



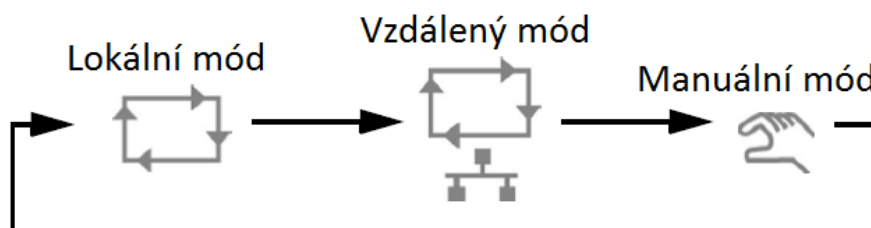
Obrázek 30: Manuální programovací panel MCP

- | | |
|--------------------------------------|-----------------------------------|
| 1 – Volba pracovního módu | 8 – Alfanumerická klávesnice |
| 2 – Tlačítko napájení motorů | 9 – Navigační klávesy a interface |
| 3 – Nouzový vypínač | 10 – Řízení aplikace |
| 4 – Tlačítka pro navigaci a pohyb | 11 – Aktivační spínač „Dead men“ |
| 5 – Tlačítka pro volbu režimu pohybu | 12 – Rychlé ovládání I/O |
| 6 – Nastavení rychlosti pohybu | 13 – Tlačítka Jog |
| 7 – Funkční klávesy | |

5.3.1 Popis MCP panelu

5.3.1.1 Volba pracovního módu (1)

Volba pracovního módu umožňuje výběr jednoho ze tří pracovních režimů (manuální, lokální, vzdálený). Zvolený mód je světelně indikován na MCP panelu. [10]



Obrázek 31: Možnosti volby pracovního módu

Manuální mód

Pohyb ramene je realizován pomocí panelu MCP a je určen zejména k manuálnímu ovládání, testování a doladování aplikace. Pohyb je omezen na rychlost 250 mm/s. Pohyby jsou prováděny, pouze pokud je stisknuto tlačítko Move/Hold, po uvolnění je pohyb přerušen. Pokud není MCP umístěn v držáku, musí být stisknuto tlačítko "Dead men". V manuálním módu nemůže být robot řízen externím zařízením. [10]

Lokální mód

Lokální mód umožňuje provádět pohyby bez jakéhokoliv lidského zásahu při maximálních rychlostech definovaných pro danou aplikaci. Pohyby jsou výsledkem příkazů napsaných v řídicím programu. Pohyby paže v lokálním módu jsou řízeny výhradně aplikací. [10]

Vzdálený mód

V tomto režimu je rameno ovládáno vzdáleným zařízením. To znamená, že některé funkce a tlačítka na MCP nejsou funkční. Záleží na zvoleném uživatelském profilu. Pokud je zvolen tento mód, může se robot začít pohybovat okamžitě po spuštění, proto by neměla být žádná osoba v jeho okolí. Rychlost robota ovládaného tímto způsobem může být maximální. [10]

5.3.1.2 Tlačítko napájení motorů (2)

Tlačítko napájení motorů umožňuje připojit nebo odpojit rameno od napájení. Zelená kontrolka indikuje, že je rameno zapnuté. Pokud v manuálním módu není MCP umístěn v držáku, musí být stlačeno aktivační tlačítko "Dead men". Pokud je zvolen vzdálený pracovní mód, toto tlačítko může být generováno externím VAL3 příkazem. [10]

5.3.1.3 Nouzový vypínač (3)

Tlačítko nouzové zastavení může být použito pouze v případě nezbytného zastavení provádění aplikace. Po stisknutí tlačítka nouzového zastavení je přerušeno napájení a rameno je zabrzděno. Ostatní zařízení je také odpojeno od zdroje napájení. [10]

5.3.1.4 Tlačítka pro navigaci a pohyb (4)

Tyto tlačítka jsou aktivní v manuálním módu a umožňují pohyb jednotlivých kloubů na ose nebo pomocí kartézských souřadnic v závislosti na volbě režimu pohybu. Jelikož je robot 4-osý, jsou tlačítka pro rotaci kolem osy X (Rx) a Y (Ry) nevyužité. Jednodušší variantou manuálního ovládání ramene je pohyb pomocí tlačítek „Jog“. [10]

5.3.1.5 Tlačítka pro volbu režimu pohybu (5)

Pokud je rameno spuštěno v manuálním módu, tyto tlačítka umožňují zvolit požadovaný režim pohybu (Joint, Frame, Tool nebo Point). Zvolený režim pohybu je indikován diodou na MCP panelu. [10]

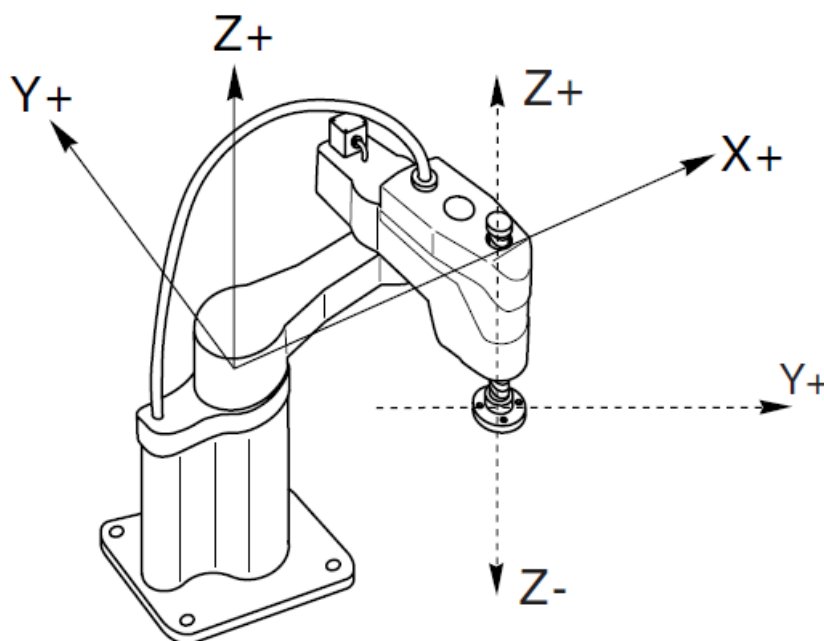
Režim pohybu joint

Tento mód slouží k pohybu jednotlivými klouby. Jelikož je robot Stäubli RS40B 4-osý, využita jsou pouze první 4 navigační tlačítka. Pohyby mohou být prováděny pomocí tlačítek pro navigaci a pohyb, a to v kladném směru (část se znaménkem „+“) nebo v záporném směru (část se znaménkem „-“). Pohyb je také možný provádět pomocí tlačítek rychlé volby. [10]

Pohyb v kartézských souřadnicích (Frame, Tool)

Tyto módy slouží pro pohyb ramenem podél tří os souřadného systému. Tyto pohyby jsou také prováděny v kladném či záporném směru. U 4-osého robota je možné využít pouze

pohybu po ose X, Y a Z a rotace kolem osy Z (RZ). V módu Frame, jsou souřadnice vztaženy k základně ramene a v módu Tool je pohyb vztažen k souřadnému systému nástroje. Tedy natočením nástroje kolem osy Z se mění i orientace jeho souřadný systém. [10]



Obrázek 32: Souřadná systém pro mód frame a tool

Point mód

Pohyb v Point módu není možné provádět pomocí navigačních tlačítek, ale pouze mezi jednotlivými body, které jsou zobrazeny na displeji MCP panelu. Tyto body lze v manuálním módu učit a upravovat. [10]

5.3.1.6 Nastavení rychlosti pohybu (6)

Tlačítko umožňuje nastavení rychlosti provádění aplikace a rychlosti pohybu ramene v manuálním módu. V manuálním módu je tato rychlost omezena na 250 mm/s. Tlačítko může být deaktivováno v závislosti na zvoleném uživatelském profilu. Pokud je tlačítko aktivní, je rychlost určována v procentech z rychlosti definované v aplikaci. [10]

5.3.1.7 Funkční klávesy (7)

Jsou použity pro výběr nabídky zobrazené nad nimi na obrazovce MCP. V závislosti na dané události jsou k těmto funkčním klávesám přiřazeny různé akce. Tyto akce jsou zobrazeny v menu na MCP displeji. [10]

5.3.1.8 Alfnumerická klávesnice (8)

Tato tlačítka jsou použita zejména pro zadávání VAL3 příkazů do aplikace. Pomocí tlačítka Shift, lze zadávat číslce a jiné znaky, které jsou na alfanumerické klávesnici zobrazeny jako druhá volba. Rozložení znaků na klávesách je stejné jako na běžné počítačové klávesnici QWERTY.

5.3.1.9 Navigační klávesy a interface (9)

Informační tlačítka a tlačítka nápovědy



Stisknutím tlačítka Help můžete kdykoliv vyvolat online nápovědu. V online nápovědě jsou funkční klávesy deaktivovány a jsou jim přiřazeny nové funkce.



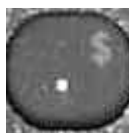
Pro opuštění online nápovědy a obnovení funkčních kláves stiskněte znovu tlačítko Help nebo tlačítko Esc.



Toto tlačítko umožňuje návrat do hlavního menu. V závislosti na uživatelském profilu může být deaktivováno.



Pomocí tohoto tlačítka se lze dostat do uživatelské části aplikace a zadávat v ní například data vyžadované pro chod aplikace.



Toto tlačítko umožňuje přechod na následující prvek, podle zvolených kritérií, které je možno zadat na alfanumerické klávesnici.

Navigační tlačítka



Tyto kurzové klávesy umožňují mimo procházení nabídek i některé další funkce popsané níže.



Rozšiřuje prvek, který byl omezen. (jako znaménko +)



Omezuje prvek, který byl rozšířen. (jako znaménko -)



Poskytuje přístup k další funkci tlačítek.



Ruší zadané hodnoty a obnovuje počáteční nastavení nebo opouští současnou stránku.



Spouští akci související s vybraným prvkem. Umožňuje změnit vybrané pole. Ověřuje správnost změněného pole.



Toto tlačítko slouží ke smazání znaku nalevo od kurzoru.

5.3.1.10 Řízení aplikace (10)

Tyto tlačítka jsou určena ke spuštění či zastavení aplikace nebo pro pozastavení pohybu ramene.



Zastaví právě prováděnou aplikaci. V závislosti na uživatelském profilu může být deaktivováno.



Umožňuje spustit vybranou aplikaci.

V manuálním módu jsou pohyby povoleny, pokud je stlačeno tlačítko

Move/Hold. Pohyby ramene jsou zastaveny, jakmile je toto tlačítko uvolněno.



V lokálním a vzdáleném módu je pohyb ramene po stlačení tlačítka přerušen a aplikace bude pozastavena. Opětovným stiskem bude aplikace obnovena opět od místa pozastavení.

5.3.1.11 Aktivační spínač „Dead men“ (11)

Tlačítko je umístěno na zadní straně MCP a má tři možné polohy:

- Uvolněné - pokud není stlačeno

- Střední pozice - tlačítko je stišťeno
- Plně stišťeno - tlačítko zůstane v této poloze, dokud nebude uvolněno

Pokud není MCP panel v držáku, je nutné v manuálním módu držet toto tlačítko po celou dobu stišťeno ve střední pozici. Pokud je aktivační tlačítko uvolněno, dojde ihned k odpojení napájení ramene. Toto tlačítko není v lokálním módu využito. [10]

5.3.1.12 Rychlé ovládání I/O (12)

V manuálním módu, tyto tlačítka mění stav digitálních výstupů, které jsou s nimi spojeny. Tlačítka mohou být přiřazeny k digitálnímu výstupu pomocí ovládacího panelu na displeji.

K přiřazení digitálních výstupů vyberte požadovaný výstup v seznamu vstupů/výstupů a stiskněte současně "Shift" a jedno z tlačítek "1", "2" nebo "3". Tato funkce nemusí být aktivní v závislosti na uživatelském profilu. [10]

5.3.1.13 Tlačítka Jog (13)

Tyto tlačítka umožňují pohyb ramene v manuálním módu pouze za pomoci jedné ruky.



Tímto tlačítkem je možné vybrat osu, po které se rameno bude pohybovat, případně kloub, který se má otočit. Vybraná osa je indikována žlutým světlem na MCP panelu v oblasti tlačítek pro navigaci a pohyb.



Pohybuje ramenem po zvolené ose nebo otáčí zvoleným kloubem. Pohyb je možný v kladném či záporném směru.

5.4 Panel pro výběr pracovního režimu

Změnu pracovního režimu lze realizovat pomocí WMS panelu. Tří polohový přepínač umožňuje vybrat jeden ze tří pracovních módů (manuální, lokální nebo vzdálený) za pomoci klíče k WMS. Zvolený mód je indikován diodou jak na WMS panelu tak i na MCP panelu. [10]

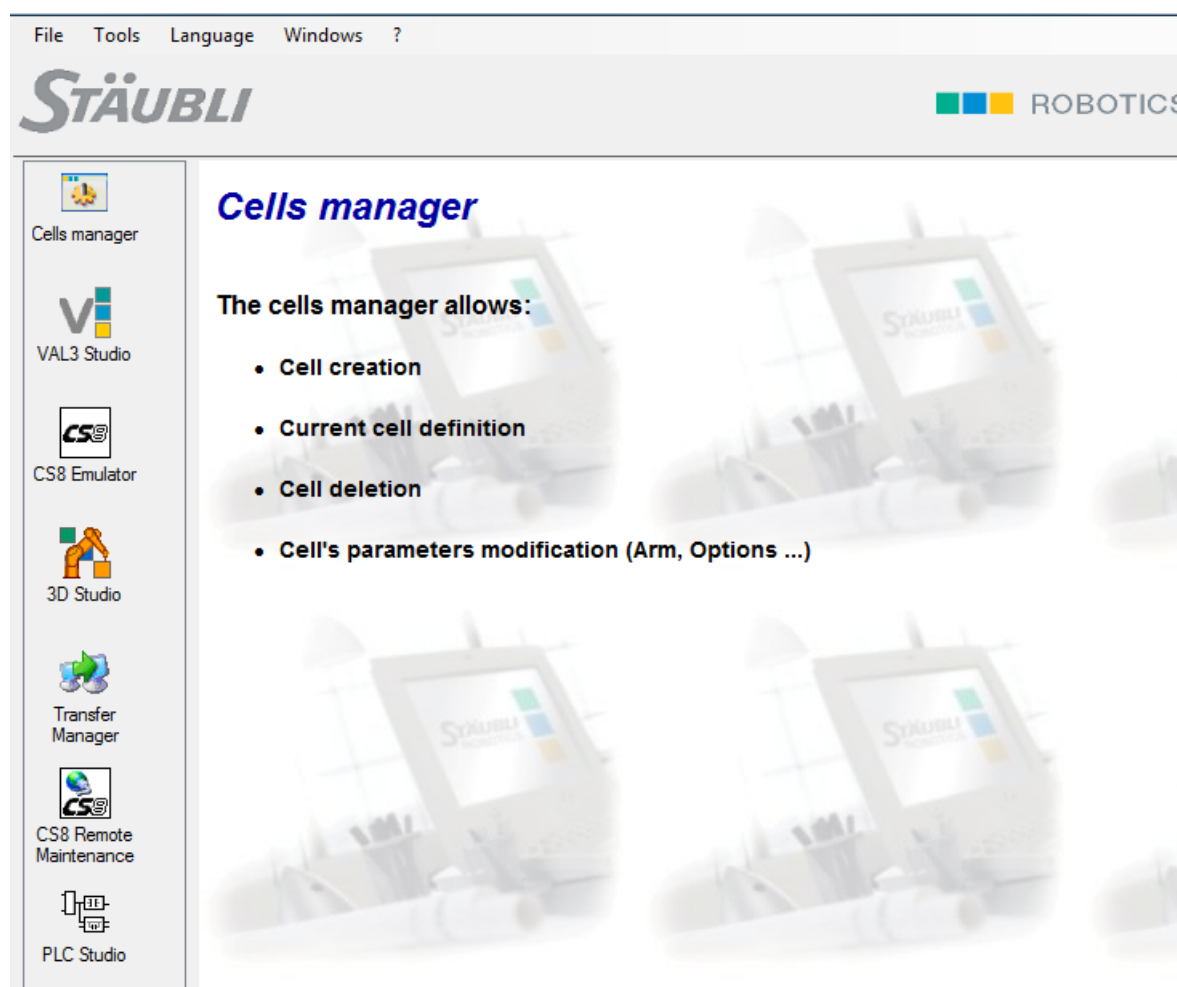
Je možné, že některé módy jsou v závislosti na uživatelském profilu zakázány. V takovém případě musí být pomocí klíče zvolen jiný pracovní mód na WMS panelu. [10]



Obrázek 33: WMS Panel pro změnu režimu

6 PROGRAMOVÉ VYBAVENÍ

Programy dodávané od výrobce jsou sloučeny v programu Stäubli Robotics Studio(SRS), který je možné nainstalovat na PC. Robotické rameno je možné ovládat přes tyto programy, nebo vytvářet vlastní aplikace a programy pro pohyb ramene. SRS obsahuje následující programy: [10]



Obrázek 34: Stäubli Robotics Studio

6.1 Cells manager

Cells manager umožňuje nastavení pracovního prostoru (buňky), ve které bude robotické rameno pracovat. Je možné nastavit adresáře pro práci počítače se zařízením. Lze definovat typ používaného zařízení, jeho upevnění a používané vstupy a výstupy. Možnosti využití a nastavení používaných doplňků jsou součástí aplikace. [10]

6.2 VAL3 Studio

VAL3 Studio slouží pro upravování a psaní aplikací. Výhodou oproti programování aplikací na MCP je přehlednost, jednoduchá ovladatelnost, příjemné uživatelské rozhraní a barevně zvýrazněný zdrojový kód. VAL3 Studio také obsahuje prostředky pro kontrolu správné syntaxe zdrojového kódu a správu aplikací v MCP panelu. [10]

6.3 CS8 Emulátor

Program SRS obsahuje také emulátor CS8C, který slouží ke kompletní simulaci funkce CS8C a MCP panelu. Lze na něm testovat aplikace, učit se ovládat robota, a to bez jakéhokoliv rizika zranění či poškození. S pomocí 3D Studia lze i vizualizovat pohyby ramene. [10]

6.4 3D Studio

Slouží k virtuálnímu zobrazení robotického ramene a jeho pohybů. Zobrazuje rameno v prostoru, ve kterém lze libovolně otáčet pohled a měnit jeho vzdálenost. Spolu s CS8 Emulátorem dokážou kompletně vizualizovat veškerou činnost ramene. Přes 3D Studio se lze i vzdáleně připojit k CS8C a sledovat tak reálné pohyby ramene na obrazovce. [10]

6.5 Transfer Manager

Přenáší data mezi kontrolérem a počítačem přes ethernet. Pomocí tohoto programu lze jednoduše přehrávat aplikace z počítače přímo do MCP a naopak. Pomocí Transfer Manageru lze také mazat. [10]

6.6 CS8 Remote Maintenance

Tímto programem lze vzdáleně ovládat robotické rameno, protože komunikuje přímo s kontrolérem. Tento program vizuálně vypadá stejně jako CS8 Emulator, rozdíl je však v tom, že komunikuje přímo s fyzickými výstupy, kterými ovládá robotické rameno. [10]

6.7 PLC Studio

Zde lze psát programy v jednom z pěti programovacích jazyků PLC, ty jsou pak převedeny na jazyk VAL3 a uloženy do kontroléru, čímž je zaručena kompatibilita s PLC programy.

7 PROGRAMOVACÍ JAZYK VAL3

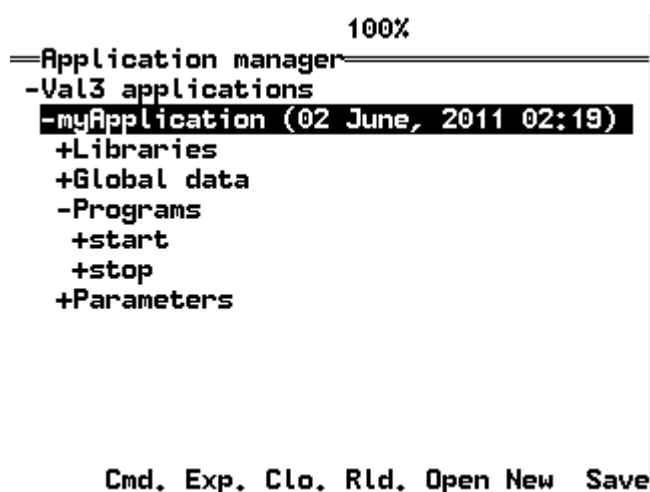
Programovací jazyk VAL3 je určen pro ovládání všech robotů firmy Stäubli. Kombinuje některé základní prvky se standardními počítačovými programovacími jazyky. Důraz je kladen zejména na nástroje pro ovládání robota, geometrické modelovací nástroje a nástroje pro správu vstupů a výstupů. [11]

Základní typy syntaxí můžeme rozdělit do několika úrovní – Základní prvky jazyka, základní proměnné typy, uživatelské rozhraní, úkoly, knihovny, řízení robota, pozice ramene a řízení pohybu.

7.1 Aplikace

VAL3 aplikace je samostatný programový balík určený pro ovládání robota a jeho vstupů a výstupů. VAL3 aplikace se skládá ze tří základních částí: programy, globální data a knihovny. Aplikace implicitně obsahuje programy „Start()“ a „Stop()“, rámec „World“ a nástroj „flange“.

Po spuštění aplikace se začnou vykonávat instrukce programu „Start()“. Po provedení všech instrukcí se spustí program „Stop()“, který aplikaci ukončí. V jednom okamžiku může být spuštěna pouze jedna aplikace.



Obrázek 35: Application manager

7.1.1 Programy

Programy jsou používány ke zjednodušení struktury aplikací, usnadňují programování, údržbu a čitelnost aplikace. Počet instrukcí programu je omezen pouze velikostí volné paměti systému. Každý program může obsahovat vlastní lokální proměnné a parametry. Jejich počet je omezen pouze velikostí zásobníku aplikace. [11]

Programy obsahují VAL3 instrukce, které mají být vykonány. Každá aplikace obsahuje hned po vytvoření programu „Start()“ a „Stop()“. Program „Start()“, je volán při spouštění aplikace a program „Stop()“ je volán pro zastavení aplikace. Tyto dva základní programy nemohou být smazány ani přejmenovány a nemají žádné parametry.

Vedle dvou základních programů, je možné vytvořit i vlastní programy, které mohou být volány kdekoliv z programu „Start()“, ale i z jakéhokoliv jiného programu. Protože jsou programy i „Start()“ a „Stop()“, lze je také libovolně volat.

7.1.2 Datové typy

Jazyk VAL3 obsahuje dva druhy datových typů a to jednoduché typy a strukturované typy. Jednoduché typy jsou proměnné, které nesou například číselnou hodnotu nebo řetězec znaků. Strukturované typy jsou složitější typová data, skládající se z několika proměnných, ale lze k nim přistupovat jako k celku. Využívají se zejména pro definici bodů v prostoru a pohybu ramene. [11]

7.1.2.1 Jednoduché datové typy

Jazyk VAL3 nabízí 6 jednoduchých datových typů, z nichž 3 slouží pro vstupy/výstupy.

- bool: logická proměnná (může nabývat pouze 2 hodnot)
- num: číselná hodnota
- string: řetězec znaků
- DIO: digitální vstup/výstup
- AIO: analogový vstup/výstup
- SIO: vstup/výstup z/do sériového portu nebo Ethernetu

7.1.2.2 *Strukturované datové typy*

Strukturované typy jsou složitější typová data, skládající se z několika proměnných, ke kterým lze přistupovat přes jejich jméno.

- trsf: nastavení geometrické transformace
- frame: nastavení kartézské geometrie
- tool: nastavení nástroje
- point: pozice v kartézských souřadnicích
- joint: pozice ramene
- config: konfigurace ramene
- mdesc: nastavení pohybu ramene

II. PRAKTICKÁ ČÁST

8 POPIS HTML MANUÁLU

Manuál byl vytvořen ve formě HTML stránek, které dohromady tvoří uživatelské prostředí pro pohyb mezi jednotlivými stránkami. Jazyk HTML je univerzálně přenositelný jazyk, tudíž by manuál měl být přístupný pod všemi operačními systémy. Pro procházení manuálu tedy postačí obyčejný internetový prohlížeč s podporou JavaScriptu a kaskádových stylů (všechny modernější prohlížeče).

Pro formátování obsahu stránek bylo využito kaskádových stylů, které zajistí jejich globální správu. Nemusí se tedy upravovat zvlášť formátování každé stránky.

Uživatelské rozhraní se skládá ze tří základních částí. V horní části okna je zobrazen titul stránky, který zůstává neměnný bez ohledu na právě otevřené stránce. V levé části je stromové menu, které slouží pro navigaci po manuálu. U každé položky je zobrazena buď ikona složky, což znamená, že položka obsahuje další seznam. Pokud je u položky zobrazena šipka, jedná se přímo o odkaz na danou stránku. Odkaz je po najetí myší zvýrazněn podtržením. Některé složky mohou být zároveň i odkazem, to indikuje podtržení po najetí myší. V horní části stromového menu se nachází dva odkazy. Jeden slouží pro rozbalení všech nabídek stromu a druhý pro sbalení všech nabídek stromu.

Pro zobrazení samotné stránky slouží hlavní menu. Název každé stránky je stejný jako název položky v levém menu.

Stránka určena pro zobrazení manuálu se jmenuje „index.html“, není však podmínkou spouštět manuál právě touto stránkou, avšak je to doporučeno, protože se jedná o výchozí stránku.



Obrázek 36: HTML uživatelský manuál

8.1 Úvod

Obsahuje účel manuálu, varovné zprávy, výstrahy a informace, a bezpečnostní pokyny pro bezpečnou práci s robotickým ramenem.

8.2 Manuál

Obsahuje samotný manuál, který popisuje základní údaje robotického ramene, obsluhu pomocí ručního programovacího panelu MCP, správu VAL3 aplikací, základy programování VAL3 aplikací a programové vybavení pro tvorbu vlastních aplikací.

8.3 Příklady

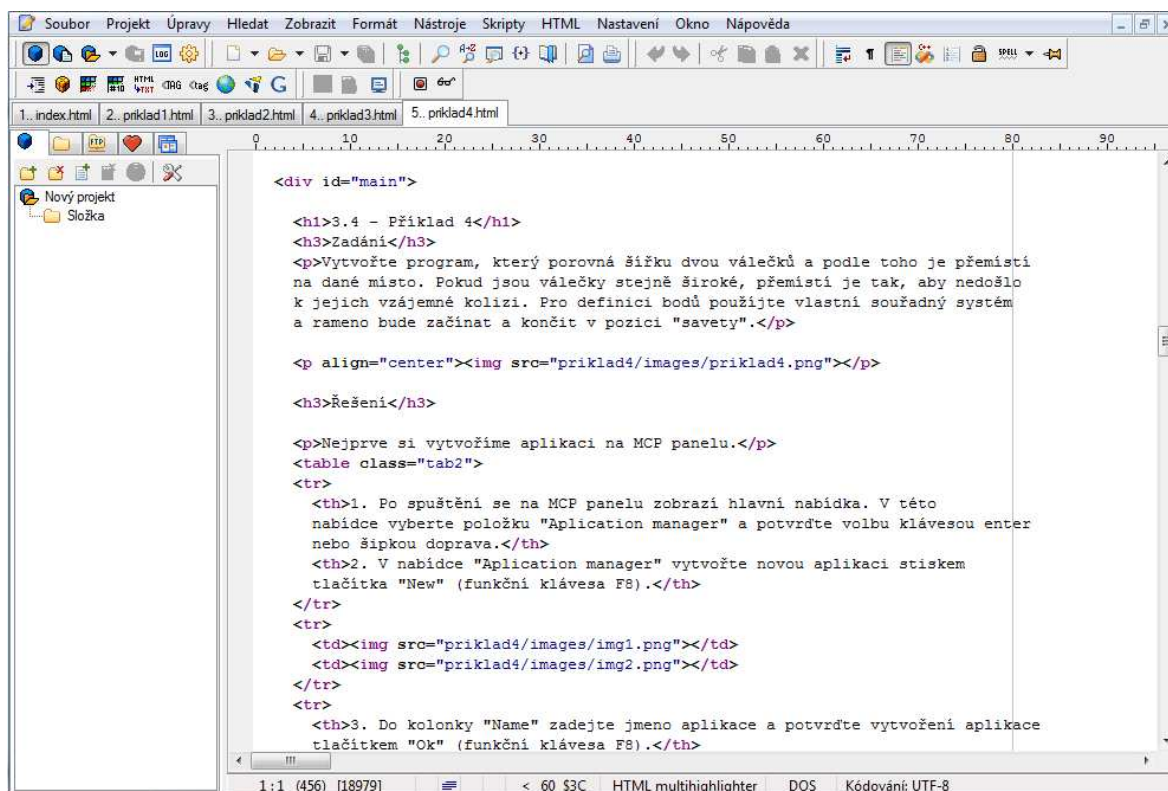
Zde jsou 4 vzorové příklady, které demonstrují možnosti průmyslového robota. V těchto příkladech je rozebráno krok po kroku vytváření zadaných aplikací. Funkce těchto příkladů je zaznamenána na videu, které se vždy nachází na konci příslušného příkladu, kde je také možné stáhnout si vzorový příklad přímo pro program VAL3 Studio.

9 TVORBA HTML MANUÁLU

Pro tvorbu manuálu byl použit program PSPad editor. Pro vytvoření stromového menu byl použit skript „Simple tree menu“, který pro jeho funkci využívá JavaScript. Stránky byly testovány v 5 nejrozšířenějších internetových prohlížečích: Microsoft Internet Explorer, Firefox Mozilla, Opera, Google Chrome a Safari. Bohužel rozevírací menu není plně kompatibilní s prohlížečem Google Chrome a vždy po otevření některé stránky se menu sbalí.

9.1 Vývojový program PSPad editor

PSPad editor je volně šiřitelný (freeware) univerzální editor pro MS Windows. Program je určený nejen k programování HTML stránek, s použitím CSS, ale i pro řadu dalších programovacích jazyků jako jsou c++, Java, Perl a další. [6]



Obrázek 37: PSPad editor

10 ZADÁNÍ PŘÍKLADŮ

V prvním příkladě se uživatel seznámí se základními instrukcemi pro pohyb, s vytvářením bodů pro určení pozice v pracovním prostoru a s manuálním ovládáním robotického ramene pro určení jeho pozice.

Druhý příklad má za úkol seznámit uživatele s vytvořením vlastního souřadného systému, deklarací bodů ve vlastním souřadném systému a pohyb v souřadném systému.

Ve třetím případě se uživatel seznámí s použitím čelistí nástroje a také se seznámí s podprogramy a s vytvářením parametrů podprogramů.

Čtvrtý příklad kompletně otestuje nabyté znalosti, kde uživatel bude muset použít podmínky pro kontrolu digitálních vstupů. Pomocí těchto vstupů je možné určit velikost uchopené součásti a podle toho ji přemístit.

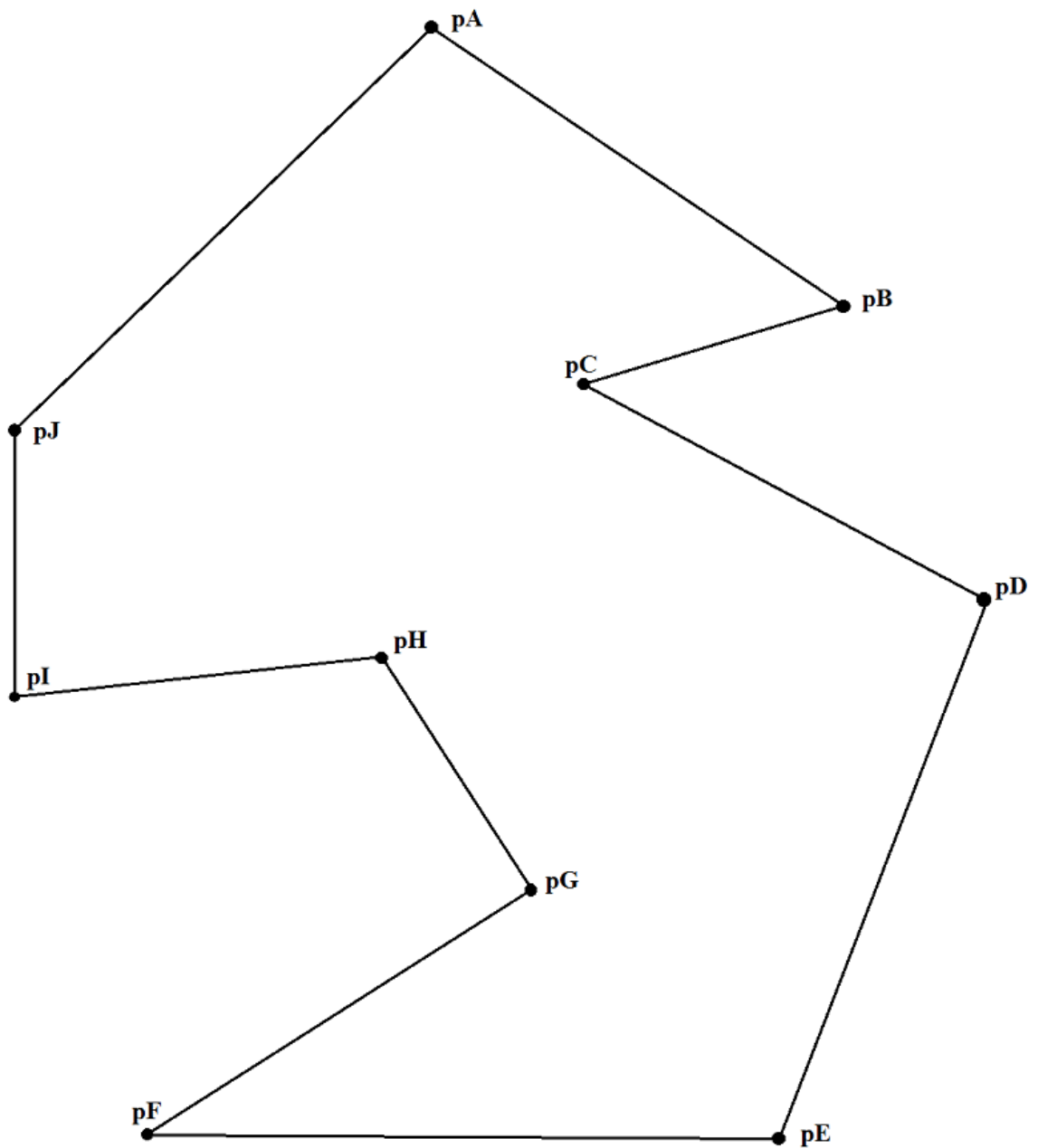
Součástí přílohy jsou i zdrojové kódy pro všechny 4 programy, které byly vygenerovány přímo z VAL3 Studia.

10.1 Příklad 1

Vytvořte program, ve kterém robot opíše manuálně definované body v abecedním pořadí s nominální rychlostí. Bude začínat a končit v bodě "safety", který bude vzdálen od pracovního prostoru.

Tabulka 2: Seznam bodů

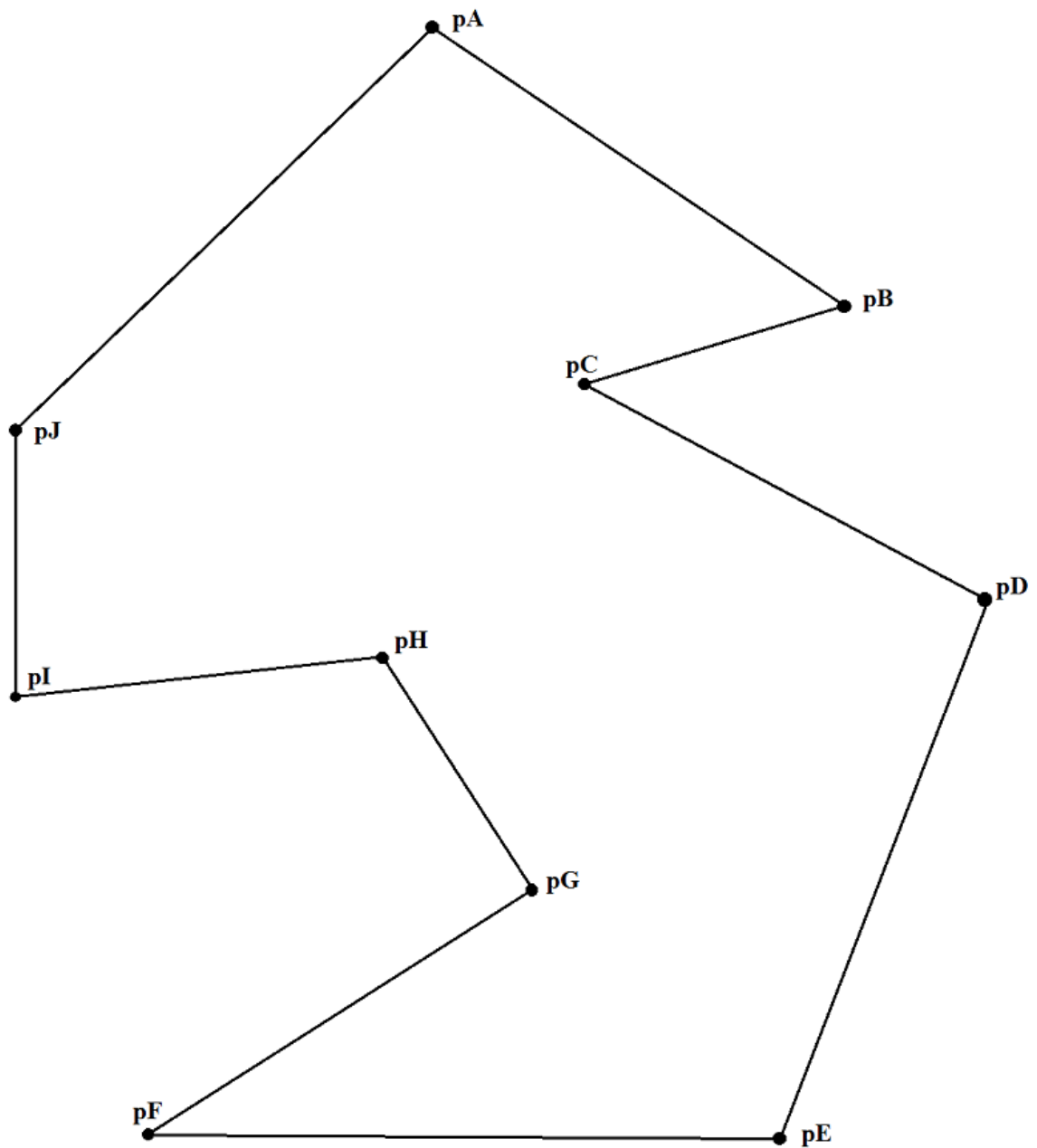
Bod	Souřadnice	Bod	Souřadnice
pA	{ 192, 5, 170, -180, 0, 0 }	pG	{ 346, 13, 170, -180, 0, 0 }
pB	{ 251, 60, 170, -180, 0, 0 }	pH	{ 307, -11, 170, -180, 0, 0 }
pC	{ 267, 21, 170, -180, 0, 0 }	pI	{ 315, -65, 170, -180, 0, 0 }
pD	{ 307, 87, 170, -180, 0, 0 }	pJ	{ 281, -68, 170, -180, 0, 0 }
pE	{ 389, 46, 170, -180, 0, 0 }	safety	{ 132, -194, 170, -180, 0, 0 }
pF	{ 389, -47, 170, -180, 0, 0 }		



Obrázek 38: Trajektorie pohybu pro příklad 1

10.2 Příklad 2

Vytvořte program, který opíše danou trajektorii se zaoblením. Vytvořte vlastní souřadný systém a vněm definujte bod shodný s počátkem souřadného systému. Pro pohyb využijte příkaz „appro“. Robotické rameno bude začínat a končit v bodě "safety", který bude shodný s počátkem souřadného systému.

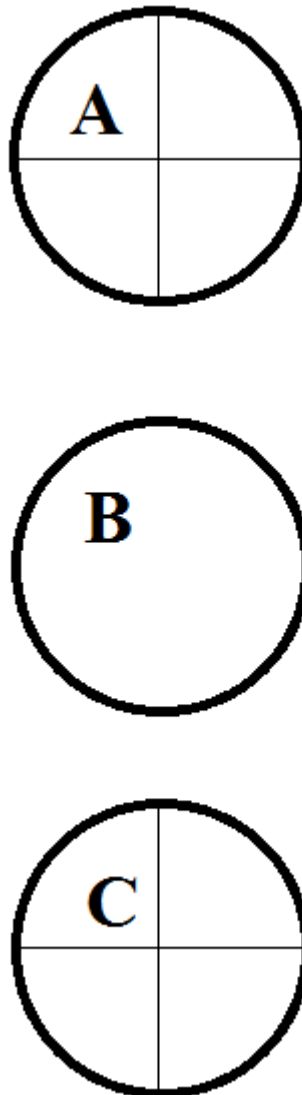


Obrázek 39: Trajektorie pohybu pro příklad 2

10.3 Příklad 3

Vytvořte program, pro výměnu dvou součástí, první součást v pozici A a druhá součást je v pozici C. Čelisti robotického ramene uchopí součást v bodě A a přesunou ji na pomocnou pozici B, z pozice C vezme druhou součást a uloží ji do pozice A, nakonec součást z pozice B přesune na pozici C. Použijte dvou podprogramů, jeden pro zvednutí součásti a

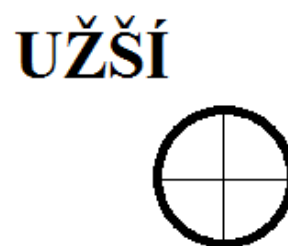
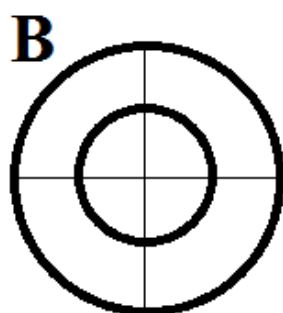
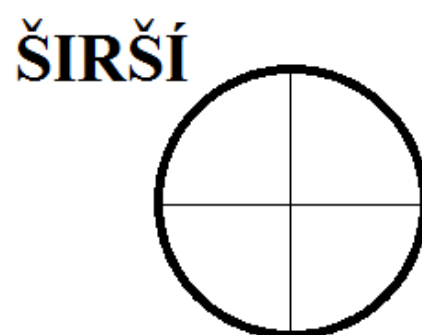
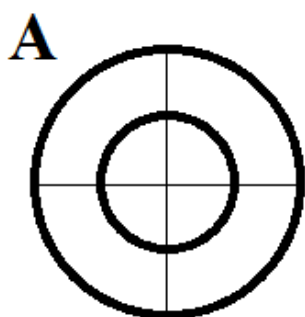
druhý pro položení součásti. Každý podprogram bude mít jeden parametr. Robotické rameno bude začínat a končit v bodě "safety", který bude vzdálen od pracovního prostoru.



*Obrázek 40: Umístění součástek
pro příklad 3*

10.4 Příklad 4

Vytvořte program, který porovná šířku dvou válečků a podle toho je přemístí na dané místo. Pokud jsou válečky stejně široké, přemístí je tak, aby nedošlo k jejich vzájemné kolizi. Pro definici bodů použijte vlastní souřadný systém a rameno bude začínat a končit v pozici "safety".



Obrázek 41: Umístění součástí pro příklad 4

ZÁVĚR

Hlavním cílem této práce bylo sestavit elektronický manuál v podobě HTML stránek, který bude sloužit pro výukové účely a bude obsahovat popis ovládání průmyslového robota pomocí ručního programovacího panelu MCP, základy práce s vývojovým prostředím VAL3 Studio, popis ovládání emulátoru CS8C a popis jazyka VAL3.

Teoretická část obsahuje historii vývoje průmyslových robotů, rozdělení podle pohybové struktury, podle pohybové jednotky a podle typu pohonu. Dále je zde popsáno robotické rameno Stäubli UNIMATION RS40B, jeho ovládání pomocí panelu MCP a struktura aplikace. Na konci teoretické části je popsáno programové vybavení, které slouží pro veškerou komunikaci počítače s kontrolérem.

V praktické části je popsán samotný HTML manuál, rozložení jednotlivých částí na stránce a popis použitých vývojových prostředí. Pro stromové menu jsem využil volně šiřitelný JavaScript, který značně usnadňuje orientaci v manuálu.

Manuál je rozdělen na tři části: úvod, manuál a příklady. Manuál se skládá z popisu ručního programovacího panelu MCP, jeho ovládáním, dále z popisu správce VAL3 aplikací a jejich programováním. Manuál je doplněn o 4 ukázkové příklady, které demonstrují možnosti jeho použití. Tyto příklady jsou v poslední části manuálu a mají za úkol seznámit uživatele s tvorbou VAL3 aplikací.

Práce předává studentovi jednak všeobecné informace o průmyslových robotech, ale hlavně jej seznámí s robotickým ramenem Stäubli UNIMATION RS40B, jeho ovládáním a tvorbou VAL3 aplikací, které jsou podporovány všemi roboty značky Stäubli.

Všechny body zadání se podařilo splnit. Problém nastal pouze při zobrazování videa v HTML stránkách. To se do jisté míry podařilo zprovoznit pomocí HTML v5, které obsahuje nástroje pro vložení videa přímo do stránek. To však snižuje kompatibilitu se staršími internetovými prohlížeči.

ZÁVĚR V ANGLIČTINĚ

The main objective of this thesis was to establish an electronic manual in HTML pages, which will be used for educational purposes and will include a description of an industrial robot control using manual programming panel MCP, basics VAL3 Studio development environment, a description of a control of the CS8C emulator and description of VAL3 language.

A theoretical part contains a history of a development of industrial robots, distribution according to movement structure, by movement control units and the type of movement drive. Longer is described robotic arm Stäubli Unimation RS40B, its control by panel MCP and application architecture. At the end of the theoretical section is described a software used for all communications between computer and controller.

A practical part describes the actual HTML manual, a configuration of its components on a page and describes a used development environment. For a tree menu, I used a freeware JavaScript, which greatly makes the orientation in the manual easier.

The manual is divided into three parts: introduction, manual and examples. The manual consists of a description of a manual programming panel MCP and its control, followed by a description of the VAL3 applications controller and its programming. The manual is completed by four sample examples that demonstrate the possibilities of its use. These examples are in the last part of the manual and are designed to familiarize users with the VAL3 applications creation.

This thesis introduces students with general information about industrial robots, but most of it familiar with the Stäubli Unimation RS40B robotic arm, its control and creating VAL3 applications that are supported by all Stäubli robots.

All entry points were fulfilled. A problem arose only when is viewing a video in HTML pages. This is to some extent succeeded Establish using HTML v5, which provides tools to embed a video directly into pages. However, this reduces a compatibility with older browsers.

SEZNAM POUŽITÉ LITERATURY

- [1] JAZAR, Reza N. Theory of applied robotics : kinematics, dynamics, and kontrol. New York : Springer, 2007. 693 s. ISBN 978-0-387-68964-7.
- [2] Learn About Robots, Industrial Robots [online]. 2011 [cit. 2011-05-04]. Dostupný z www: <http://www.learnaboutrobots.com/industrial.htm>.
- [3] Robot Worx, The History and Benefits of Industrial Robots [online]. 2011 [cit. 2011-03-04]. Dostupný z www: <http://www.used-robots.com/robot-education.php?page=benefits+of+industrial+robots>.
- [4] HAVEL, Ivan M. Robotika : úvod do teorie kognitivních robotů. 1. vyd. Praha : SNTL, 1980. 279 s.
- [5] SKAŘUPA, Jiří. Průmyslové roboty a manipulátory. Vysoká škola báňská. 2007. 229 s.
- [6] PSPad editor. [online]. 2011 [cit. 2011-05-12]. Dostupný z www: <http://www.pspad.com/>.
- [7] KARGER, Adolf; KARGEROVÁ, Marie. Základy robotiky a prostorové kinematiky. Vyd. 1. Praha : ČVUT, 2000. 265 s. ISBN 80-01-02183-1.
- [8] KOSEK, Jiří. HTML : tvorba dokonalých www stránek. 1. vyd. Praha : Grada, 1998. 291 s. ISBN 8071696080.
- [9] JAZAR, Reza N. Theory of applied robotics: kinematics, dynamics, and kontrol. New York: Springer, 2007. 693 s. ISBN 978-0-387-68964-7.
- [10] CS8C Controller : Instruction manual. Faverage: Stäubli, 2008. 190 s.
- [11] VAL3 reference manual. Faverage: Stäubli, 2008. 186 s.
- [12] JANKOVSKÝ, Dušan. Jak psat web [online]. 2011 [cit. 2011-04-15]. Dostupný z www: <http://www.jakpsatweb.cz/>.
- [13] RUMÍŠEK, Pavel : Automatizace (roboty a manipulátory), 2003. 31 s.
- [14] VAL3 reference manual. Faverage: Stäubli, 2008. 186 s.
- [15] Stäubli - Textile Machinery, Connectors and Robotics [online]. 2011 [cit. 2011-04-17]. Dostupný z www: <http://www.staubli.com/>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

3D	3 Dimensional
AIO	Analog Input Output
DIO	Digital Input Output
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
MCP	Manual Control Pendant
PLC	Programmable Logic Controller
PUMA	Programmable Universal Manipulation Arm
SCARA	Selective Compliant Assembly Robot Arm / Selective Compliant Articulated Robot Arm
SIO	Serial Input Output
USB	Universal Serial Bus
SCARA	Selective Compliant Assembly Robot Arm / Selective Compliant Articulated Robot Arm
VAL	Variable Assembly Language

SEZNAM OBRÁZKŮ

<i>Obrázek 1: Rameno určené ke svařování</i>	<i>9</i>
<i>Obrázek 2: Umístění průmyslových robotů v automobilovém průmyslu</i>	<i>9</i>
<i>Obrázek 3: První průmyslový robot Unimate</i>	<i>12</i>
<i>Obrázek 4: Chapadlové rameno Marvinina Minsky.....</i>	<i>12</i>
<i>Obrázek 5: Unimation PUMA</i>	<i>13</i>
<i>Obrázek 6: Robot FAMULUS.....</i>	<i>13</i>
<i>Obrázek 7: Robotické rameno RX170 od firmy Stäubli.....</i>	<i>14</i>
<i>Obrázek 8: Princip pohybu v kartézské pohybové struktuře.....</i>	<i>15</i>
<i>Obrázek 9: Kartézský robot</i>	<i>15</i>
<i>Obrázek 10: Cylindrická pohybová struktura.....</i>	<i>16</i>
<i>Obrázek 11: Cylindrický robot Denso</i>	<i>16</i>
<i>Obrázek 12: Sférická pohybová struktura</i>	<i>17</i>
<i>Obrázek 13: Sférický svařovací robot.....</i>	<i>17</i>
<i>Obrázek 14: Kloubová pohybová struktura.....</i>	<i>18</i>
<i>Obrázek 15: Kloubový robot Stäubli RX260</i>	<i>18</i>
<i>Obrázek 16: Struktura SCARA</i>	<i>19</i>
<i>Obrázek 17: Průmyslový robot Stäubli RS60</i>	<i>19</i>
<i>Obrázek 18: Princip pohybu pohyblivými saněmi</i>	<i>20</i>
<i>Obrázek 19: Průmyslový robot KUKA Jet.....</i>	<i>20</i>
<i>Obrázek 20: Princip pohybu teleskopickým ramenem.....</i>	<i>21</i>
<i>Obrázek 21: Těžká technika s teleskopickým ramenem</i>	<i>21</i>
<i>Obrázek 22: Jednotka s přímým pohonem.....</i>	<i>22</i>
<i>Obrázek 23: Jednotka s nepřímým pohonem</i>	<i>22</i>
<i>Obrázek 24: Krokový motor.....</i>	<i>23</i>
<i>Obrázek 25: Pneumatický motor P1/R1</i>	<i>25</i>
<i>Obrázek 26: Hydraulický válec.....</i>	<i>25</i>
<i>Obrázek 27: Robotické rameno RS40B.....</i>	<i>26</i>
<i>Obrázek 28: Pracovní rozsah robotického ramene</i>	<i>27</i>
<i>Obrázek 29: Kontrolér CS8C</i>	<i>28</i>
<i>Obrázek 30: Manuální programování panel MCP</i>	<i>29</i>
<i>Obrázek 31: Možnosti volby pracovního módu</i>	<i>30</i>

<i>Obrázek 32: Souřadná systém pro mód frame a tool</i>	<i>32</i>
<i>Obrázek 33: WMS Panel pro změnu režimu</i>	<i>36</i>
<i>Obrázek 34: Stäubli Robotics Studio</i>	<i>37</i>
<i>Obrázek 35: Application manager.....</i>	<i>39</i>
<i>Obrázek 36: HTML uživatelský manuál</i>	<i>44</i>
<i>Obrázek 37: PSPad editor</i>	<i>45</i>
<i>Obrázek 38: Trajektorie pohybu pro příklad 1</i>	<i>47</i>
<i>Obrázek 39: Trajektorie pohybu pro příklad 2.....</i>	<i>48</i>
<i>Obrázek 40: Umístění součástí pro příklad 3.....</i>	<i>49</i>
<i>Obrázek 41: Umístění součástí pro příklad 4.....</i>	<i>50</i>

SEZNAM TABULEK

<i>Tabulka 1 : Pracovní rozsahy</i>	28
<i>Tabulka 2: Seznam bodů</i>	46

SEZNAM PŘÍLOH

PI	Výstup z programu VAL3 Studio – Příklad 1
PII	Výstup z programu VAL3 Studio – Příklad 2
PIII	Výstup z programu VAL3 Studio – Příklad 3
PIV	Výstup z programu VAL3 Studio – Příklad 4
PV	CD-ROM

PŘÍLOHA P I: VÝSTUP Z PROGRAMU VAL3 STUDIO – PŘÍKLAD 1

Project properties	
Parameters	version:s8.6.4, unit:millimeter, stackSize:5000
Program Files	start.pgx stop.pgx
Data Files	priklad1.dtn
Librairies	io:io

Tool		
Public	Name	Father
	flange	

Frame		
Public	Name	Father
	world	

Point (Scara)		
Public	Name	Father
	pA	world[0]
	pB	world[0]
	pC	world[0]
	pD	world[0]
	pE	world[0]
	pF	world[0]
	pG	world[0]
	pH	world[0]
	pI	world[0]
	pJ	world[0]
	safety	world[0]

MDesc				
Public	Name		Public	Name
■	mlomSpeed			

start ()

```
begin
//bod safety
move1(safety, flange, mNomSpeed)
waitEndMove()

//bod pA
move1(pA, flange, mNomSpeed)
waitEndMove()

//bod pB
move1(pB, flange, mNomSpeed)
waitEndMove()

//bod pC
move1(pC, flange, mNomSpeed)
waitEndMove()

//bod pD
move1(pD, flange, mNomSpeed)
waitEndMove()

//bod pE
move1(pE, flange, mNomSpeed)
waitEndMove()

//bod pF
move1(pF, flange, mNomSpeed)
waitEndMove()

//bod pG
move1(pG, flange, mNomSpeed)
waitEndMove()

//bod pH
move1(pH, flange, mNomSpeed)
waitEndMove()

//bod pI
move1(pI, flange, mNomSpeed)
waitEndMove()

//bod pJ
move1(pJ, flange, mNomSpeed)
waitEndMove()

//bod pA
move1(pA, flange, mNomSpeed)
waitEndMove()

//bod safety
move1(safety, flange, mNomSpeed)
waitEndMove()
end
```

stop ()

```
begin
  popUpMsg ("Pending movement commands have been canceled")
  resetMotion()
end
```

PŘÍLOHA P II: VÝSTUP Z PROGRAMU VAL3 STUDIO – PŘÍKLAD 2

Project properties	
Parameters	version:s6.6.4, unit:millimeter, stackSize:5000
Program Files	start.pgx stop.pgx
Data Files	Priklad2.dtx
Librairies	io:io

Tool		
Public	Name	Father
	flange	

Frame		
Public	Name	Father
	fFrame	world[0]
	world	

Point (Scara)		
Public	Name	Father
	inception	fFrame[0]
	safety	world[0]

MDesc				
Public	Name		Public	Name
■	mNomSpeed			

start ()

begin

```
//bod safety  
move(safety, flange, mNomSpeed)  
waitEndMove()
```

```
//nastaveni zaobleni  
mNomSpeed.blend=joint  
mNomSpeed.leave=20  
mNomSpeed.reach=20
```

```
//bod pA  
move(appro(inception, {60, -199, 0, 0, 0, 0}), flange, mNomSpeed)  
waitEndMove()
```

```
//bod pB  
move(appro(inception, {119, -254, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pC  
move(appro(inception, {135, -215, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pD  
move(appro(inception, {175, -281, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pE  
move(appro(inception, {257, -240, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pF  
move(appro(inception, {257, -147, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pG  
move(appro(inception, {214, -207, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pH  
move(appro(inception, {175, -183, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pI  
move(appro(inception, {183, -129, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pJ  
move(appro(inception, {149, -126, 0, 0, 0, 0}), flange, mNomSpeed)  
//bod pA  
move(appro(inception, {60, -199, 0, 0, 0, 0}), flange, mNomSpeed)  
waitEndMove()
```

```
//bod safety  
move(safety, flange, mNomSpeed)  
waitEndMove()
```

end

<i>stop ()</i>
<code>begin</code> <code>end</code>

PŘÍLOHA P III: VÝSTUP Z PROGRAMU VAL3 STUDIO – PŘÍKLAD

3

Project properties	
Parameters	version:s6.4, unit:millimeter, stackSize:5000
Program Files	start.pgx stop.pgx chyt.pgx pust.pgx
Data Files	Priklad3.dtx
Librairies	io:io

Tool		
Public	Name	Father
	flange	

Frame		
Public	Name	Father
	myFrame	world[0]
	world	

Point (Scara)		
Public	Name	Father
	pA	myFrame[0]
	pB	myFrame[0]
	pC	myFrame[0]
	savety	myFrame[0]

MDesc				
Public	Name		Public	Name
■	mNomSpeed			

*chyt (point **bod**)*

```
begin
  //přemístí se nad bod
  movej(bod, flange, mNomSpeed)
  waitEndMove()

  //otevření čelistí
  io:bOut0=false

  //přemístí se blíže k součásti
  move1(appro(bod,{0,0,50,0,0,0}), flange, mNomSpeed)
  waitEndMove()
  //uchopí součást
  io:bOut0=true
  delay(1)

  //zvedne součást nahoru
  move1(appro(bod,{0,0,-50,0,0,0}), flange, mNomSpeed)
  waitEndMove()
end
```

*pust (point **bod**)*

```
begin
  //přesune součást na bod
  movej(bod, flange, mNomSpeed)
  waitEndMove()
  //přesunutí součásti níž
  movej(appro(bod,{0,0,50,0,0,0}), flange, mNomSpeed)
  waitEndMove()

  //pustí součást
  io:bOut0=false
  delay(1)

  //přemístí se výše
  move1(appro(bod,{0,0,-50,0,0,0}), flange, mNomSpeed)
  waitEndMove()
end
```

start ()

```
begin
  //bod safety
  movej(safety, flange, mNomSpeed)
  waitEndMove()

  //uchopí součást v bodě pA a položí ji v bodě pB
  call chyt(pA)
  call pust(pB)

  //uchopí součást v bodě pC a položí ji v bodě pA
  call chyt(pC)
  call pust(pA)

  //uchopí součást v bodě pB a položí ji v bodě pC
  call chyt(pB)
  call pust(pC)

  //pod safety
  movej(safety, flange, mNomSpeed)
  waitEndMove()
end
```

<i>stop ()</i>
<pre>begin popUpMsg("Pending movement commands have been canceled") resetMotion() end</pre>

PŘÍLOHA P IV: VÝSTUP Z PROGRAMU VAL3 STUDIO – PŘÍKLAD

4

Project properties	
Parameters	version:s6.6.4, unit:millimeter, stackSize:5000
Program Files	chyt.pgx pust.pgx start.pgx stop.pgx
Data Files	priklad4.dtx
Librairies	io:io

Tool		
Public	Name	Father
	flange	

Frame		
Public	Name	Father
	myFrame	world[0]
	world	

Point (Scara)		
Public	Name	Father
	pA	myFrame[0]
	pB	myFrame[0]
	pM	myFrame[0]
	pV	myFrame[0]
	savety	myFrame[0]

MDesc				
Public	Name		Public	Name
	mMdesc		■	mNomSpeed

*chyt (point **bod**)*

```
begin
  //přesune se nad bod
  movej(bod, flange, mNomSpeed)
  waitEndMove()

  //pokud jsou čelisti sevřené otevře je
  if io:bIn1==false
    io:bOut0=false
  endif

  //posune se k válečku
  movel(appro(bod,{0,0,20,0,0,0}), flange, mNomSpeed)
  waitEndMove()

  //chyti váleček
  io:bOut0=true
  delay(1)

  //posune se zpět nahoru
  movel(appro(bod,{0,0,-20,0,0,0}), flange, mNomSpeed)
  waitEndMove()
end
```

*pust (point **bod**)*

```
begin
  //přesune se nad bod
  movej(bod, flange, mNomSpeed)
  waitEndMove()

  //pokud čelisti drží váleček
  if io:bIn1==false

    //přesune se k bodu
    movel(appro(bod,{0,0,20,0,0,0}), flange, mNomSpeed)
    waitEndMove()

    //pustí váleček
    io:bOut0=false
    delay(1)

    //posune se zpět nahoru
    movel(appro(bod,{0,0,-20,0,0,0}), flange, mNomSpeed)
    waitEndMove()
  endif
end
```



```
start ()
```

```
bool mensi  
bool vetsi
```

```
begin  
    //bod savety  
    movej(savety, flange, mNomSpeed)  
    waitEndMove()  
  
    ////////////První váleček/////////  
    //volej podprogram chyt  
    call chyt(pA)  
    //pokud je váleček větší položí jej na bod pV  
    if io:bIn2==true  
        call pust(pV)  
        vetsi=true  
        //jinak jej položí na bod pM  
    else  
        call pust(pM)  
        mensi=true  
    endif  
    ////////////Druhý váleček/////////  
    //volej podprogram chyt  
    call chyt(pB)  
    //testuje zda je váleček větší  
    if io:bIn2==true  
  
        //pokud byl předchozí váleček menší položí jej na pV  
        if mensi==true  
            call pust(pV)  
        //pokud byl předchozí váleček větší položí jej na pM (dva velké)  
        else  
            call pust(pM)  
        endif  
  
        //pokud je váleček menší  
    else  
  
        //pokud byl předchozí váleček větší položí jej na pM  
        if vetsi==true  
            call pust(pM)  
        //pokud byl předchozí váleček menší položí jej na pV (dva malé)  
        else  
            call pust(pV)  
        endif  
    endif  
  
    //bod savety  
    movej(savety, flange, mNomSpeed)  
    waitEndMove()  
  
end
```

<i>stop ()</i>
begin end

PŘÍLOHA P V: CD-ROM

Obsahuje tištěnou formu práce, elektronický manuál, který byl součástí práce, soubory

*.zip, ve kterých jsou uloženy zdrojové kódy příkladů. N nosiči jsou přílohy P I, P II, P III, P IV.