

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY
FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

NÁVRH A REALIZACE NEJNIŽŠÍ VRSTVY ŘÍZENÍ ČTYŘNOHÉHO KRÁČIVÉHO ROBOTU

The Design and Realization of the Lowest Level Layer of the Control Unit for a Walking Robot

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **PETR OLŠA**
AUTHOR

VEDOUCÍ PRÁCE **ING. VÍT ONDROUŠEK**
SUPERVISOR

Brno, 2008

ZADÁNÍ ZÁVĚREČNÉ PRÁCE

Práce je zaměřena na vytvoření návrhu a realizaci nejnižší vrstvy řízení kráčivého robotu. Jedná se o čtyřnohý kráčivý robot se třemi stupni volnosti na nohu. Návrh jednotky řízení je podmíněn sestavením kinematického modelu nohy a následně vytvořením kinematického modelu celého robotu. Po simulačním ověření na PC je tento návrh implementován do mikrokontroleru.

Cíle:
1) Proveďte rešerši současného stavu řízení kráčivých robotů
2) Sestavte kinematický model nohy robotu se třemi stupni volnosti a ověřte jej pomocí software simulace na PC.
3) Upravený kinematický model následně implementujte do mikrokontroleru a ověřte jeho funkčnost na reálném robotu.
4) Na základě předchozích bodů navrhněte základní styl chůze robotu v rovinném prostředí bez překážek.

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: Petr Olša

Bytem: Brno – Vinohrady, Valtická 4 628 00

Narozen/a (datum a místo): 12. 3. 1985 Brno

(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta strojního inženýrství

se sídlem Technická 2896/2, 616 69, Brno

jejímž jménem jedná na základě písemného pověření děkanem fakulty:

Doc. RNDr. Ing. Miloš Šeda, Ph.D.

(dále jen „nabyvatel“)

Čl. 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
   - □ disertační práce
   - □ diplomová práce
   - ■ bakalářská práce
   - □ jiná práce, jejíž druh je specifikován jako .....

   (dále jen VŠKP nebo dílo)

Název VŠKP:	NÁVRH A REALIZACE NEJNIŽŠÍ VRSTVY ŘÍZENÍ ČTYŘNOHÉHO KRÁČIVÉHO ROBOTU
Vedoucí/ školitel VŠKP:	Ing. Vít Ondroušek
Ústav:	Ústav automatizace a informatiky
Datum obhajoby VŠKP:	

VŠKP odevzdal autor nabyvateli v*:

- ■ tištěné formě – počet exemplářů ……2……...
- ■ elektronické formě – počet exemplářů ……3……..

* hodící se zaškrtněte

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizovaní výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
   - ■ ihned po uzavření této smlouvy
   - □ 1 rok po uzavření této smlouvy
   - □ 3 roky po uzavření této smlouvy
   - □ 5 let po uzavření této smlouvy
   - □ 10 let po uzavření této smlouvy

   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: 23.5.2008

…….. ……

Nabyvatel Autor

ABSTRAKT

Cílem této bakalářské práce je návrh a realizace nejnižší vrstvy řízení kráčivého robotu.
Realizace nejnižší vrstvy řízení obsahuje:

- zavedení souřadných systémů
- řešení inverzní úlohy kinematiky robotu
- vytvoření simulačního a komunikačního programu
- implementaci inverzní úlohy kinematiky do mikrokontroleru,
- začlenění sériové komunikace mezi počítačem a robotem do řídicího systému
- realizaci kroku robotu.

Na závěr se testuje schopnost řídicího systému vykonávat povely v reálném čase.

ABSTRACT

This BA thesis is focused on the project and realization of the lowest layer of manipulation of the walking robot.
The realization of the lowest layer contains:

- installation of the coordinated systems
- the solution of the inverse task of kinematics of the robot
- creation of the simulative and communicative program
- implementation of the inverse task of kinematics into the microcontroller
- incorporation of serial communication between the PC and robot into the controlling system
- realization of robot's walking

In the end, the ability of the controlling system to accomplish the orders in real time is tested.

KLÍČOVÁ SLOVA

souřadné systémy robotu, inverzní úloha kinematiky, simulace, krok robotu

KEYWORDS

The coordinated systems of the robot, inverse task of kinematics, simulation robot's walking

PODĚKOVÁNÍ

Mé poděkování patří fakultě za vytvoření podmínek pro realizaci praktické části a poskytnutí nejmodernějšího vybavení, dále pak vedoucímu práce panu Ing. Vítu Ondrouškovi a Ing. Pavlu Houškovi Ph.D. za podnětné připomínky.

Obsah:

ZADÁNÍ ZÁVĚREČNÉ PRÁCE 3
ABSTRAKT 7
KLÍČOVÁ SLOVA 7
PODĚKOVÁNÍ 9
SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ 12
1 ÚVOD 13
2 AKTUÁLNÍ STAV ŘEŠENÉ PROBLEMATIKY 15
2.1 Rozdělení robotů 15
2.1.1 Dle technických prostředků, umožňujících pohyb robotů 15
2.1.2 Dle počtu končetin 15
2.1.3 Dle počtu stupňů volnosti 16
2.1.4 Dle použitých pohonů 16
2.1.5 Dle typu chůze 17
2.2 Geometrie nohy 17
2.2.1 Úlohy spojené s řešením geometrie 17
2.3 Fáze kroku 18
2.4 Použitý robot 18
2.4.1 Mechanické provedení 18
2.4.2 Použité pohony 19
2.4.3 Navržený řídicí systém 20
3 PRAKTICKÁ ČÁST 23
3.1 Zavedení souřadných systémů 23
3.2 Úlohy kinematiky 25
3.3 Simulační program 29
3.3.1 Popis simulačního programu 30
3.3.2 Simulace kroku 31
3.3.3 Popis tříd simulačního programu 31
3.4 Implementace kinematického modelu do mikrokontroleru 32
3.4.1 Princip řízení 33
3.5 Krok robotu 35
3.5.1 Výpočet jednotlivých bodů elipsy 36
3.5.2 Výpočty jednotlivých bodů obdélníku 37
3.5.3 Výpočty jednotlivých bodů oporné fáze kroku 38
3.6 Návrh chůze robotu 39
3.7 Přiložené zdrojové kódy 40
4 ZÁVĚR 41
5 POUŽITÁ LITERATURA 43
PŘÍLOHY 45

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

a	úhlu servomotoru jedna
b	úhlu servomotoru dva
g	úhel u paty
d	úhel odklonu úsečky AB od osy x v LS
e	úhel pro časovač
r_{VS}	polární úhel ve VS
r'	úhlu servomotoru nula
A'_x	souřadnice x bodu obdélníku
A_x	souřadnice x počátečního bodu kroku
A_{xj}	souřadnice x j-tého bodu na elipse
A'_y	souřadnice y bodu obdélníku
A_{yj}	souřadnice y j-tého bodu na elipse
B_y	souřadnice y koncového bodu kroku
HS	Hlavní souřadný systém
KS	Pomocný kartézský souřadný systém
LS	Lokální kartézský souřadný systém
N_x^{KS}	souřadnice x bodu N v KS
N_y^{KS}	souřadnice y bodu N v KS
N_P	počet pořadí chůze
P_x^{ks}	souřadnice x bodu P v KS
P_y^{ks}	souřadnice y bodu P v KS
VS	Lokální válcový souřadný systém
a	rozměr prvního segmentu nohy
b	rozměr druhého segmentu nohy
c	rozměr třetího segmentu nohy
d	pomocná proměnná
p	hodnota pro registr časovače
f	počet bodů elipsy
i	číslo nohy robotu
j	počet bodů na přímce
k_1	kružnice k_1
k_2	kružnice k_2
n	počet noh robotu
r_{VS}	polární poloměr ve VS
$\vec{u}$	směrový vektor
x_{LS}	souřadnice x v LS
xy_{LS}	rovina xy v LS
y_{LS}	souřadnice y v LS
z_{LS}	souřadnice z v LS
z_{VS}	souřadnice z ve VS

1 ÚVOD

Robot - slovo, které lákalo spisovatele už v dávných dobách a současně atribut, bez kterého si současnou ani budoucí společnost nejsme schopni představit. Již malé děti zaujme hračka v podobě robotu, která zvládne to, co dospělému člověku činí potíže. Dnes již tento významný pomocník zastupuje lidský faktor na montážních linkách, je nepostradatelným pomocníkem při těžké, pro člověka nepopulární namáhavé práci. Typickým příkladem použití robotu jsou život ohrožující situace, například při likvidaci výbušnin či při zemětřesení nebo závalech v důlních komplexech. V těchto případech byly díky robotům zachráněny mnohé lidské životy.

O těchto pozoruhodných výtvorech lidského umu jsem přečetl nejednu knihu a seznámil se s danou problematikou v mnoha článcích odborných časopisů ještě na střední škole.

Proto jsem uvítal, že se roboty objevily mezi tématy bakalářské práce. Měl jsem možnost volit mezi několika tématy , ale nejvíce mne zaujala problematika pohybu robotů, neboť se domnívám, že jde o oblast velmi zajímavou, umožňující aplikace různých prvků. Výše zmíněná problematika přímo nabádá autora práce, aby v ní uplatnil vlastní představivost. Domnívám se, že spojením tvořivých schopností se znalostmi a zákonitostmi vědy zde lze docílit solidního výsledku.

Moje práce je členěna do dvou částí. První kapitola je věnována obecným pojmům a úvodu do problematiky kráčivých robotů. Druhá kapitola se již zaměřuje na praktickou část bakalářské práce. V jejím závěru jsou shrnuty výsledky praktické části.

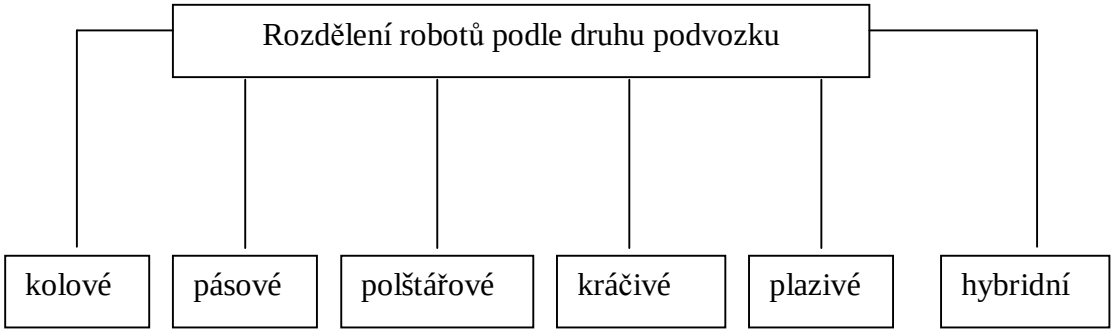
2 AKTUÁLNÍ STAV ŘEŠENÉ PROBLEMATIKY

2.1 Rozdělení robotů

Robot je obvykle elektromechanický stroj, který je schopen vnímat svoje okolí a toto okolí ovlivňovat. Roboty lze rozčlenit na 2 základní skupiny – mobilní a imobilní. Ve své práci se zabývám pouze mobilními roboty, tedy těmi, které se dokáží přesunovat v prostoru. Mobilní robot může mít rozdílnou míru autonomie od teleoperovaného až po plně autonomní.[10] Teleoperovaný robot pouze vykonává příkazy operátora. Autonomní robot je zařízení, které inteligentně propojuje senzory s řízením.[11]
Vzhledem k tomu, že lze roboty dělit podle různých kritérií, považuji za nezbytné některá z nich v úvodní kapitole představit.

2.1.1 Dle technických prostředků, umožňujících pohyb robotů

Tyto prostředky také označujeme jako podvozky mobilních robotů. Rozdělení robotů podle druhu podvozku je na Obr. 1.



Obr. 1 Rozdělení mobilních robotů podle druhu podvozku.[1]

Roboty s kolovým podvozkem se v praxi vyskytují nejčastěji. Kolo má oproti jiným způsobům pohybu jednoznačně nejlepší účinnost. Pro náročnější prostředí, se dá využít pásových robotů. [9]
Já jsem ve své práci zaměřil pozornost na pohyb kráčivý, který je pozoruhodný z hlediska možnosti překonávání nerovností terénu a případných překážek.

2.1.2 Dle počtu končetin

Počet končetin kráčivého stroje má pochopitelně velký vliv na jeho "výkonnost", tedy zejména na maximální rychlost pohybu a způsobu překonávání terénu. V historii byly sestrojeny funkční roboty s jednou končetinou stejně jako se dvanácti. Ačkoli počet noh není nijak limitován a není ani přijat "obecně uznávaný" optimální počet končetin, je evidentní a logická inklinace k určitým typům a počtům končetin. [4]
Při požadavku statické stability při pohybu je nejnižší počet noh čtyři (tři nohy pochopitelně mohou být samy o sobě také staticky stabilní, ale nám potom chybí rezerva pro nohu, která aktuálně provádí přesun …). Čtyřnohé staticky stabilní řešení však má dvě nevýhody: je nutné "uměle" měnit těžiště (obvykle se provádí pohyb baterie v jedné ose) a pohyb vždy provádí

pouze jedna končetina. Pohyb je tedy značně pomalý. Často tedy bývá aplikován šestinohý kráčivý robot, kde může být zvednutá v daném časovém okamžiku více než jedna končetina a navíc není nutno přesouvat hmotu robotu pro změnu těžiště, chceme-li dodržet podmínku statické stability.[4]

2.1.3 Dle počtu stupňů volnosti

Dalším problémem při návrhu kráčivého robotického systému je, kolik stupňů volnosti by měly mít jednotlivé končetiny. Na jedné straně je vhodný co nejvyšší počet stupňů volnosti, aby byla dostatečná volnost při polohování, na straně druhé je nutno si uvědomit, že každý aktivní stupeň volnosti znamená další pohon, což zvyšuje nejen cenu a složitost systému, ale také jeho hmotnost. Obecně je možno říci, že u robotů s vyšším počtem končetin (tj. 4-6) jsou používány nohy s nižším stupněm volnosti – zpravidla 1-3, zatímco u robotů dvounohých se běžně používá 5 stupňů volnosti.[4]
Jako stupně volnosti označujeme směry posunu a otáčení, v nichž může dojít k pohybu nebo otáčení tělesa, popřípadě bodu. V prostoru u tělesa zaznamenáváme 6 stupňů volnosti (posun podél os x, y, z a otáčení kolem výše jmenovaných os. Stupeň volnosti u nohy robotu je dán součtem stupňů volnosti jejích jednotlivých vazeb.

2.1.4 Dle použitých pohonů

Při výběru pohonu robotu bychom měli vzít v úvahu: hmotnost robotu, rychlost pohybu, požadovanou přesnost a další důležitá hlediska. Podle zvolených požadavků si pak můžeme vybrat z následujících typů motorů:

- **Stejnosměrné motory**

Jsou patrně nejjednodušší na použití, neboť jsou napájeny pouhým stejnosměrným napájením. Uvnitř stejnosměrného motoru jsou pevně umístěny permanentní magnety (stator) a volně se otáčející elektromagnety (rotor). Problém napájení pohyblivých elektromotorů je řešen mocí tzv. komutátoru.[4]

Stejnosměrné motory jsou nejčastěji používány v malých a středně velkých robotických systémech. Známými producenty stejnosměrných motorů jsou firmy Maxon Motors, Aulhaber a Mabuchi.[4]

- **Střídavé motory**

Nepoužívají komutátor pro fázování motoru. Cyklus je místo toho tvořen řídicí elektronikou. Jejich výhodou oproti stejnosměrným motorům jsou menší rozměry při stejném výkonu a možnost dosažení stejných parametrů při otáčení v obou směrech (pozn. u stejnosměrných motorů je cyklus nastaven pevně, u modelářských motorů navíc často tak, že je výrazně upřednostňován pravotočivý směr otáčení). Nevýhodou je komplikovanější a rozměrnější řídicí a výkonová elektronika a více vývodů se složitějším zapojením. [4]

- **Krokové motory**

Tyto motory umožňují přesné natáčení osy s definovaným rozlišením. Není tedy obvykle potřeba inkrementální enkodér. Výhodou je také to, že umožňují zabrždění osy. Nevýhod je hned několik: motory jsou obecně podstatně pomalejší. Nepříliš výhodný je i poměr kroutícího momentu vůči hmotnosti. Tyto motory bývají používány u malých a lehkých robotů s diferenciálním podvozkem. [4]

- **Servomotory**

Takto bývají označovány motory s integrovanou převodovkou a inkrementálním enkodérem. Někdy se servomotorem či servem myslí tento komplet včetně řídicí elektroniky. Velkou popularitu mají v oblasti malých a levných robotů tzv. modelářská serva, která obsahují minimotor, převodovku, potenciometr a řídicí elektroniku v kompaktním pouzdru se třemi vývody (napájení, zem, řídicí signál, kde délka pulsu udává natočení). Servomotory se primárně nemohou kontinuálně otáčet, jsou určeny pouze pro polohování v rozsahu 180° (Graupner) či 120 °. Je však možno klasický servomotor upravit pro kontinuální provoz se zachováním možnosti řídit rychlost otáčení v obou směrech šířkou pulsu. Tato úprava je však u většiny typů nezvratná a je nutno počítat s rychlým opotřebením motorků i převodů, protože tyto nejsou stavěny na kontinuální provoz. [4]

2.1.5 Dle typu chůze

Základní a obecným požadavkem, bez jehož splnění je chůze jako výsledek řízení kráčivého robotu nemyslitelná, je stabilita. [16]
U kráčivých robotů můžeme definovat dva typy pohybu: statickou chůzi a dynamickou chůzi. Při statické chůzi je robot v každém okamžiku ve statické rovnováze. To je obvyklý případ u čtyř a vícenohých robotů. Řešení statické stability vychází z konstrukce tzv. oporného mnohoúhelníku Obr 22.
Při dynamické chůzi se robot v některých fázích pohybu dostává do nestabilní polohy, např. se nedotýká podložky vůbec. Příkladem takového pohybu je běh nebo skákání. Výhodou dynamické chůze je možnost dosažení vyšší rychlosti, než u chůze statické. Nevýhodou je řádově vyšší náročnost na řízení takového stroje a nemožnost zastavit pohyb v každém okamžiku pohybu.[13]

2.2 Geometrie nohy

Geometrie nohy robotu je základní stavební kámen pro realizaci pohybu nohy. Každý stupeň volnosti na noze nám poskytuje lepší a hladší provedení pohybu. Na druhou stranu musíme dbát na matematickou náročnost při výpočtu, která se při vyšším stupni volnosti značně zvyšuje.

Nejprve bych rád vysvětlil základní pojem kinematika a její význam v geometrii nohy: Kinematika se zabývá geometrií a pohybem kinematického řetězce (nohy) bez ohledu na síly, které pohyb způsobují. Vzájemné polohy částí jednotlivých kinematických dvojic (kloubů, resp. polohy jednotlivých servomechanizmů) představují tzv. kloubové souřadnice robotu.[2]

2.2.1 Úlohy spojené s řešením geometrie

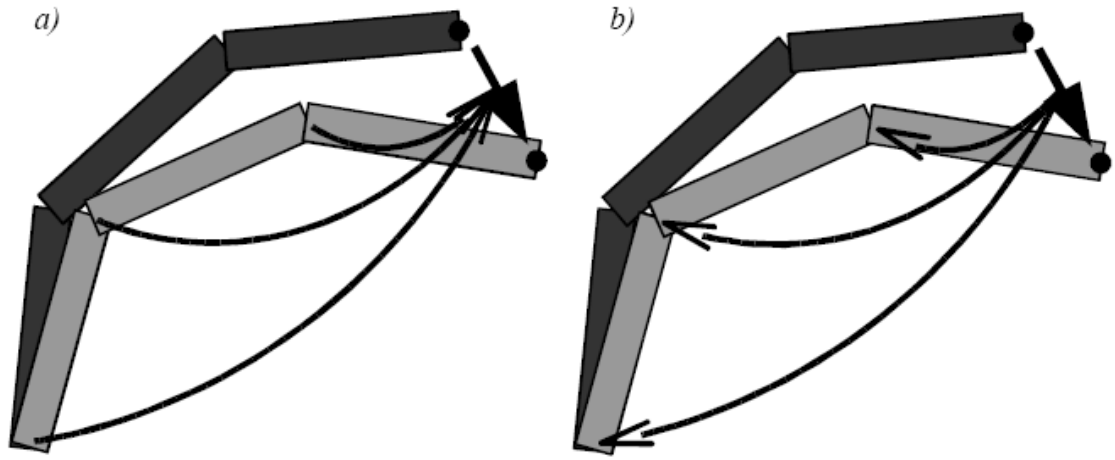
Zjištění závislosti vektoru polohy a orientace koncového členu nohy na kloubových souřadnicích se nazývá **přímá úloha kinematiky**. Naopak zjištění závislosti vektoru kloubových souřadnic na vektoru polohy a orientaci představuje **inverzní úlohu kinematiky** Obr. 2. [2], [3]

K řešení inverzní kinematiky se používá mnoho metod a jejich kombinací. Každá z nich samostatně má výhody, ale i nevýhody. Proto se často kombinují a doplňují o další přístupy.

- Inverze Jakobiánu
- Metody založené na optimalizaci
- Inverzní dynamika

Metoda Inverze Jakobiánu využívá předpokladu, že pokud existuje vztah mezi úhly jednotlivých segmentů a polohou koncového bodu, měla by také existovat závislost inverzní, mezi polohou koncového bodu a úhly jednotlivých segmentů (ne vždy však musí být jednoznačná). Pro výpočet se používá diferenciálních rovnic. [6]

V praktické části v kapitole 3.4.1 je popsán postup řešení v případě nejednoznačnosti inverzní úlohy kinematiky.

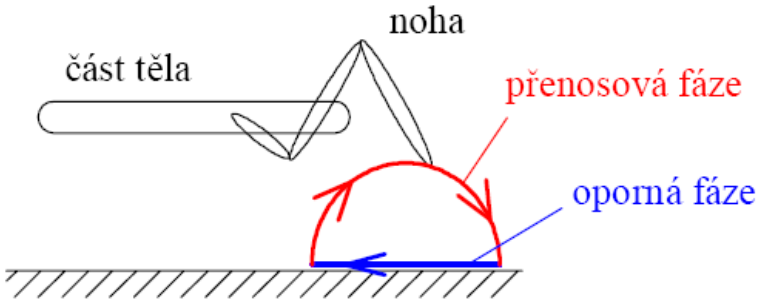


Obr. 2 Způsob ovládání kloubové struktury pomocí a) přímé b) inverzní kinematiky. [6]

2.3 Fáze kroku

Při popisu kroku se inspirujeme jedním cyklem kroku hmyzu[14], který je ukázán na Obr. 3. Každý cyklus kroku můžeme rozdělit do dvou fází:

1) oporná fáze (stance phase) – Noha se opírá o terén a v chodidle nohy existuje silová reakce terénu. [15],[16]
2) přenosová fáze (swing phase) – Noha je přenášena do nového oporného bodu. [15],[16]



Obr. 3 Dvě hlavní fáze kroku

Uskutečnění jedné přenosové a jedné oporné fáze představuje krok nohy.[16]

2.4 Použitý robot

Pro praktickou část mé práce byl zvolen robot firmy Lynxmotion typ EQ3-KT

2.4.1 Mechanické provedení

Jedná se o čtyřnohý robot, polohovaný dvanácti servomotory viz Obr 4.

Stavebnice obsahuje mechanické díly řezané laserem, spojovací materiál, servomotory, neobsahuje řídicí elektroniku.



Obr. 4 Robot od firmy Lynxmotion typ EQ3-KT. [7]

2.4.2 Použité pohony

Jako pohon jsou ve stavebnici použity servomotory Hitec HS-422 Obr. 5
Pro každý stupeň volnosti je použit jeden servomotor. Každá noha má 3 stupně volnosti, tedy celkem má robot 12 servomotorů.

- Servomotor je napájen 5V DC
- rychlost servomotoru je při 4.8V 0,23sec/60° a nebo při 6.0V 0,18sec/60°
- točivý moment je $5,5kg \cdot cm$ to se rovná přibližně $0,5N \cdot m$
- hmotnost: 1,41oz = 40g

Díky dobrému poměru hmotnosti, točivého momentu a rychlosti je to ideální volba pohonu robotu.



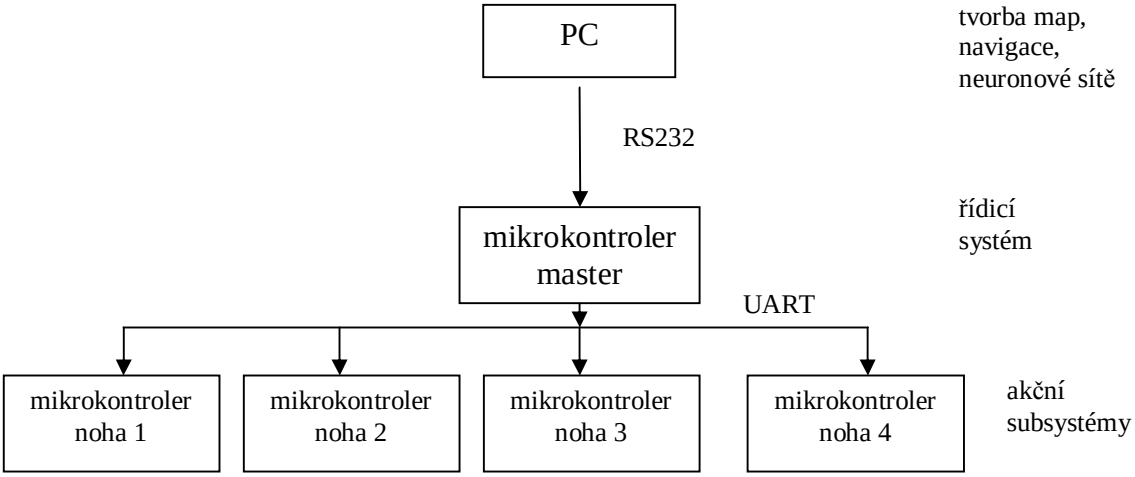
Obr. 5 HS-475HB servomotor.[7]

2.4.3 Navržený řídicí systém

Není třeba, aby koncepce řídicího systému byla založena pouze na jednom počítači nebo mikrokontroleru, který musí být schopen výkonnostně pokrýt veškeré požadavky dané aplikace. Jednotlivé požadované úlohy lze řešit také odděleně – tzv. distribuovaně. Tyto samostatné systémy pak řeší pouze omezené specializované úlohy, jako např. závislé řízení pohonů, ovládání senzoru nebo skupiny senzorů včetně vhodného předzpracování jejich dat. Právě v oblasti školních mobilních robotů je koncepce distribuovaného řízení založená na jednoduchých mikrokontrolerech poměrně rozšířená. Výhodou této koncepce také je, že v etapě vývoje daného zařízení lze na jednotlivých subsystémech pracovat odděleně, případně aplikaci postupně doplňovat o nové subsystémy, nebo stávající subsystémy nahrazovat jinými.[5]

V oblasti mikrokontrolerů postavených na 8bitových mikroprocesorech je standardním sériovým rozhraním UART [5]

Model řízení mého robotu je jen předběžný a ve finální verzi se bude lišit. Návrh byl proveden panem Ing. Houškou Ph.D. Už z návrhu vyplývá, že robot bude mít distribuovaný model řízení Obr. 6.



Obr. 6 Distribuovaný model řízení.

3 PRAKTICKÁ ČÁST

V praktické části se budu zabývat návrhem a realizací nejnižší vrstvy (souřadným systémem) robotu. Navrhnuté řešení pak realizuji v simulačním programu, který jsem pro tento účel naprogramoval. Po úspěšném odsimulování řešených problémů naprogramuji řídicí systém pro mikrokontroler jedné nohy robotu.

3.1 Zavedení souřadných systémů

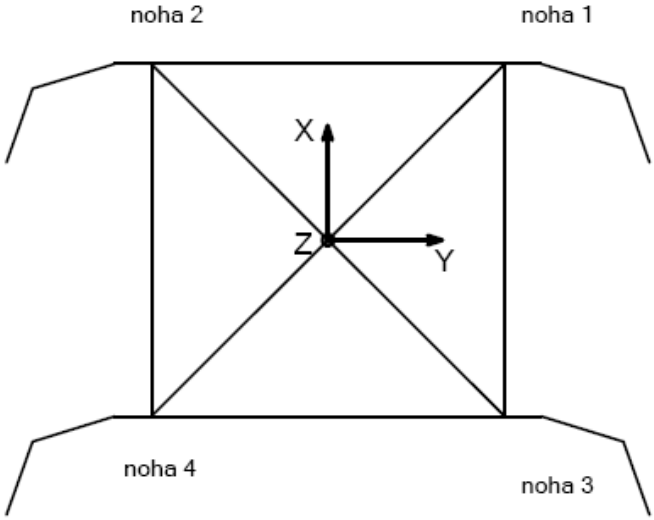
Správná volba souřadných systémů vede k dobré orientaci v prostoru a ke snazším matematickým výpočtům.
Byly zavedeny následující souřadné systémy

- Globální souřadný systém

Je spjat s prostředím. Při navigaci robotu je nutné určit jeho polohu vůči globálnímu souřadnému systému. Po určení této polohy přebírá úlohu řídicí systém, který ze získaných hodnot určí optimální trajektorii pohybu. Tento souřadný systém pro návrh řídicího systému robotu není důležitý, proto se jím dále nebudu zabývat.

- Hlavní souřadný systém (kartézský 3D) HS Obr. 7

Tento souřadný systém bude využívat řídicí systém k určení trajektorie pohybu. Jeho počátek bude ve středu obdélníku, který je tvořen 4 počátečními body Lokálních souřadných systémů. V Hlavním souřadném systému se počítají souřadnice koncových poloh pro jednotlivé nohy.



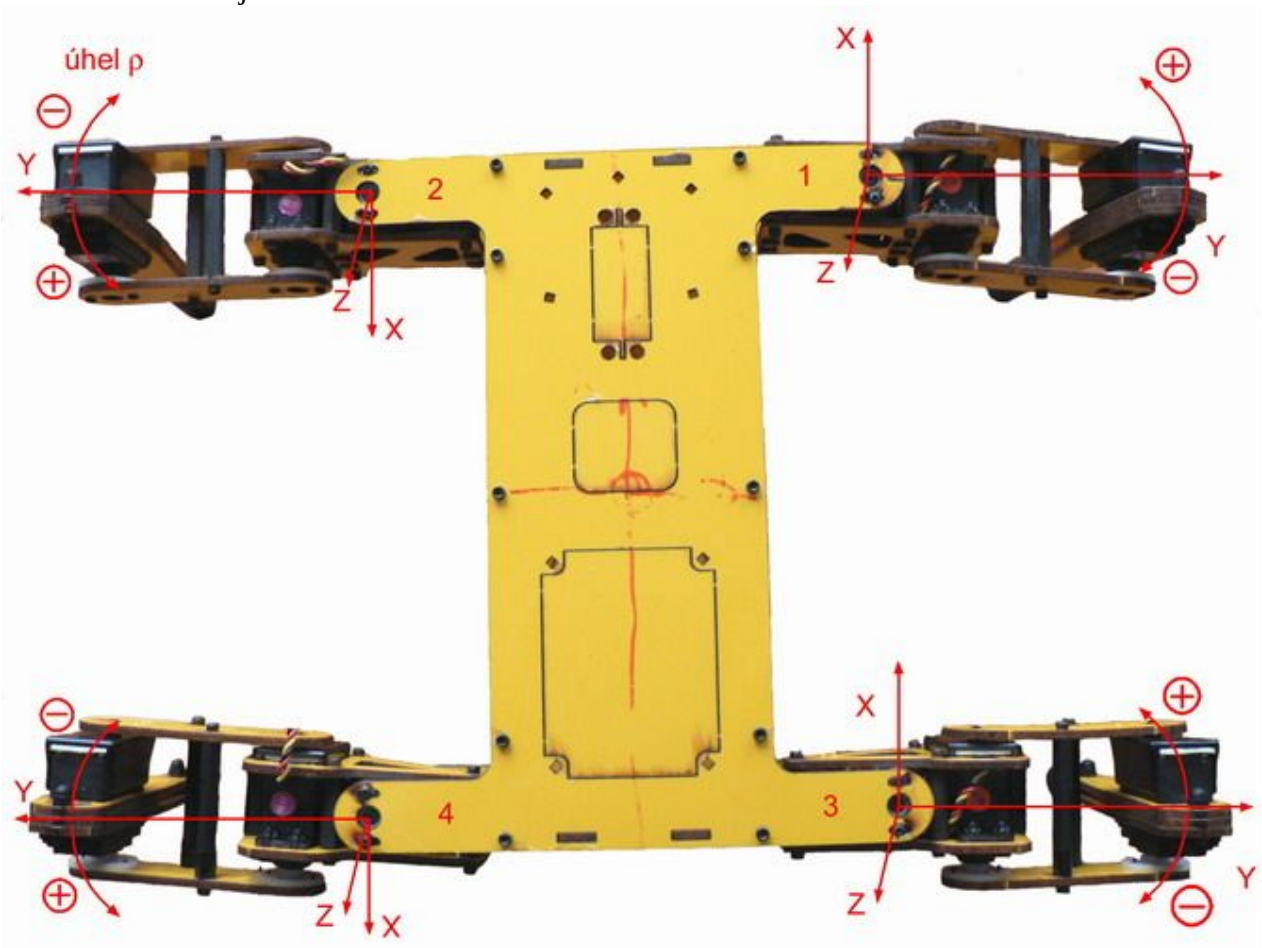
Obr. 7 Hlavní souřadný systém (pohled na robota seshora).

- Lokální kartézské souřadné systémy noh robotu LSi Obr. 8 a 9

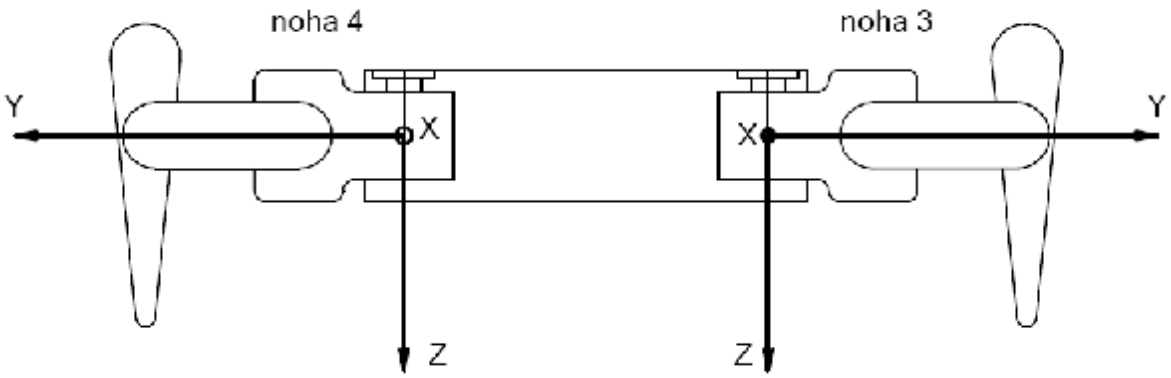
Každá noha má vlastní souřadný systém, a to kartézský $[x, y, z]_{LSi}$ kde $i \in (1,2,3,4)$ je index pro každou nohu. Osa *z* vždy směřuje dolů z důvodu kladných výsledků (stejně jako osa *z* HS). Pro nohy na pravé straně (1, 3) platí, že všechny 3 osy mají stejný směr jako osy HS. Nohy 2 a 4 mají pootočený souřadný systém o 180° po směru hodinových ručiček okolo kladné osy *z* vůči pravým nohám (1, 3). Tato volba zajistí, že osa *y* bude vždy kolmá na tělo robotu a bude vždy směrem ven z těla robotu. Díky této volbě se pak snáze realizují výpočty. Na obrázku můžeme vidět i číslování noh, které je voleno tak, že na pravé straně jsou lichá čísla pro snadnou rozšiřitelnost na šestinohý kráčivý robot.

- Lokální válcové souřadné systémy noh robotu VSi

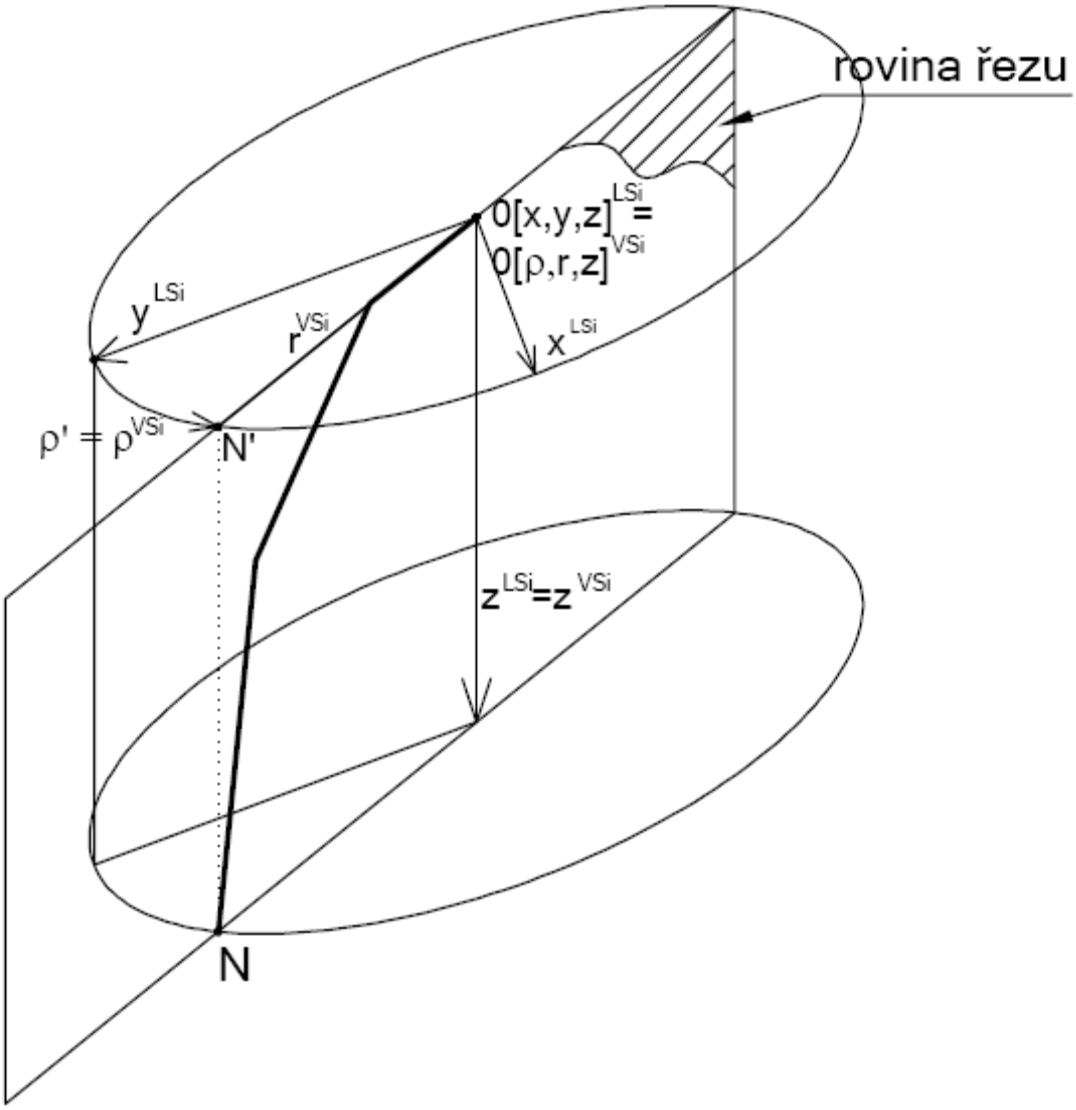
Každá noha má vlastní souřadný systém, a to válcový [ρ, r, z] $_{Vsi}$ kde $i \in (1,2,3,4)$ je index pro každou nohu. Má počátek ve stejném bodě jako LSi. Je zaveden za účelem zjednodušení výpočtu. Polární úhel ρ je měřen od osy *y* LSi a jeho smysl je na Obr. 8. Polární poloměr r je kolmá vzdálenost od koncového bodu nohy k vertikální ose *z* LSi a hodnota *z* VSi je totožná s hodnotou *z* LSi. Obr. 10



Obr. 8 Lokální kartézské souřadné systémy noh robotu (LS), pohled seshora.



Obr. 9 Lokální kartézské souřadné systémy noh robotu (LS), pohled zepředu.



Obr. 10 Lokální válcový souřadný systém (VS) a Lokální kartézský souřadný systém noh robotu (LS).

3.2 Úlohy kinematiky

Přímá úloha kinematiky, která počítá z úhlů servomotorů souřadnice koncového bodu (N), pro nás není podstatná, jelikož servomotor nemá zpětnou vazbu, která by udala úhel. Pro vytvoření řídicího programu je třeba řešit inverzní úlohu kinematiky nohy robotu, tj. sestavit rovnice, které vypočítají natočení jednotlivých servomotorů při zadání koncového bodu nohy v Lokálním kartézském souřadném systému.

Popis výpočtu:

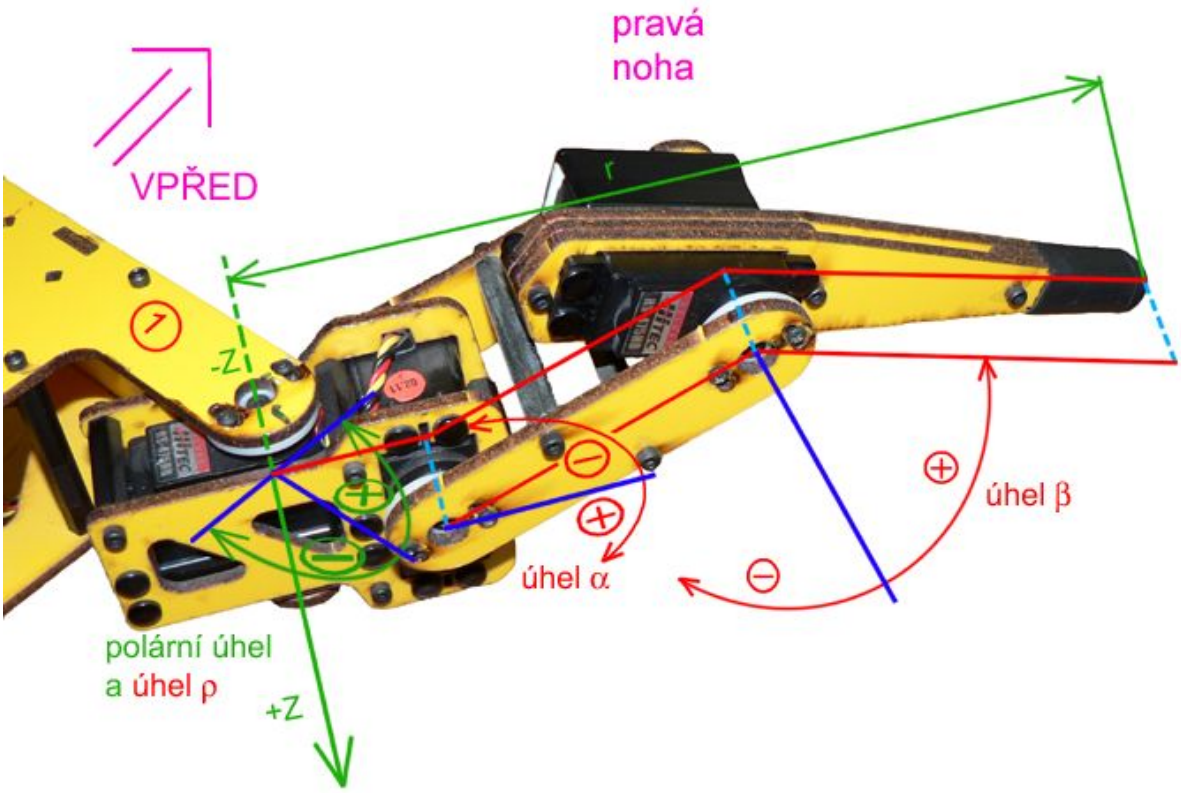
Známé jsou souřadnice $[x, y, z]^{LS}$, ty se přepočítají do $[\rho, r, z]^{VS}$.

Pro převod platí:

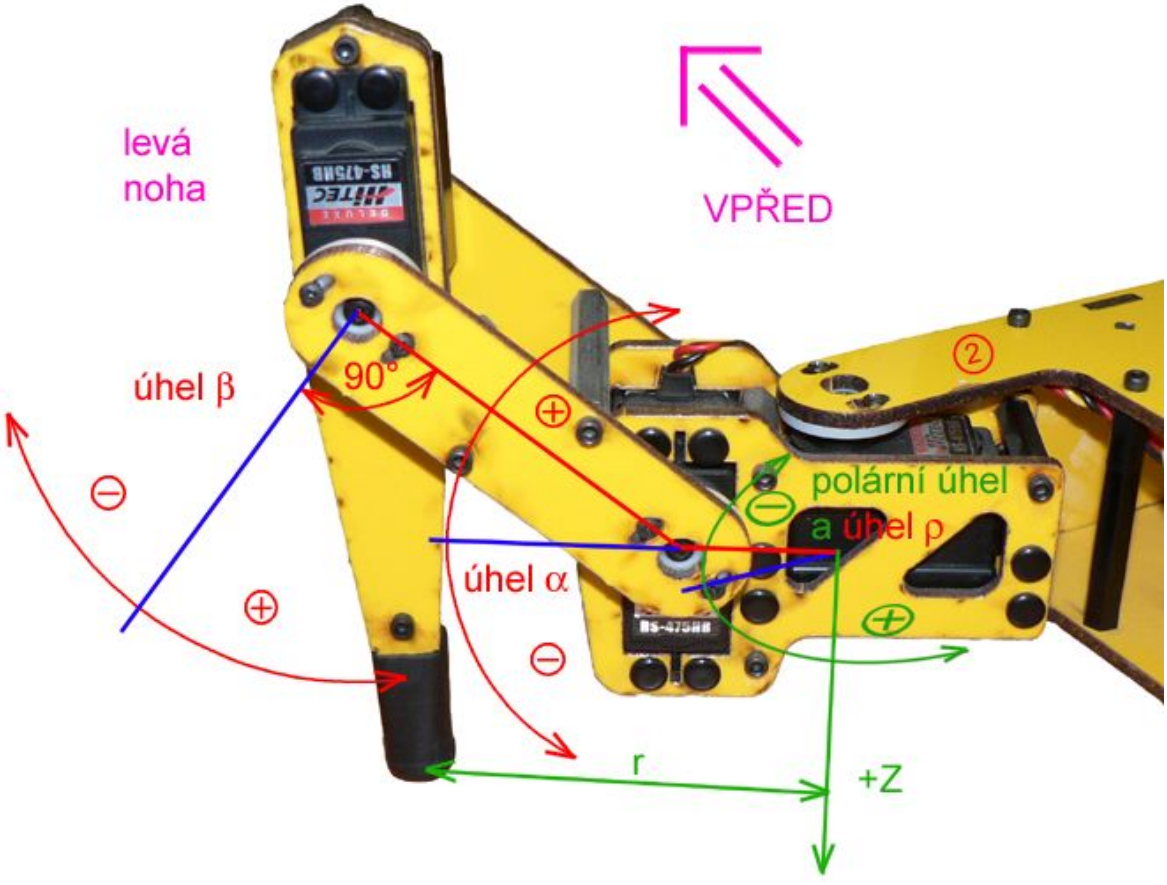
$$r_{VSi} = \sqrt{x_{LSi}^2 + y_{LSi}^2} \tag{3.1}$$

$$r_{VSi} = \arctan(x_{LSi} / y_{LSi}) \tag{3.2}$$

$$z_{VSi} = z_{LSi} \tag{3.3}$$

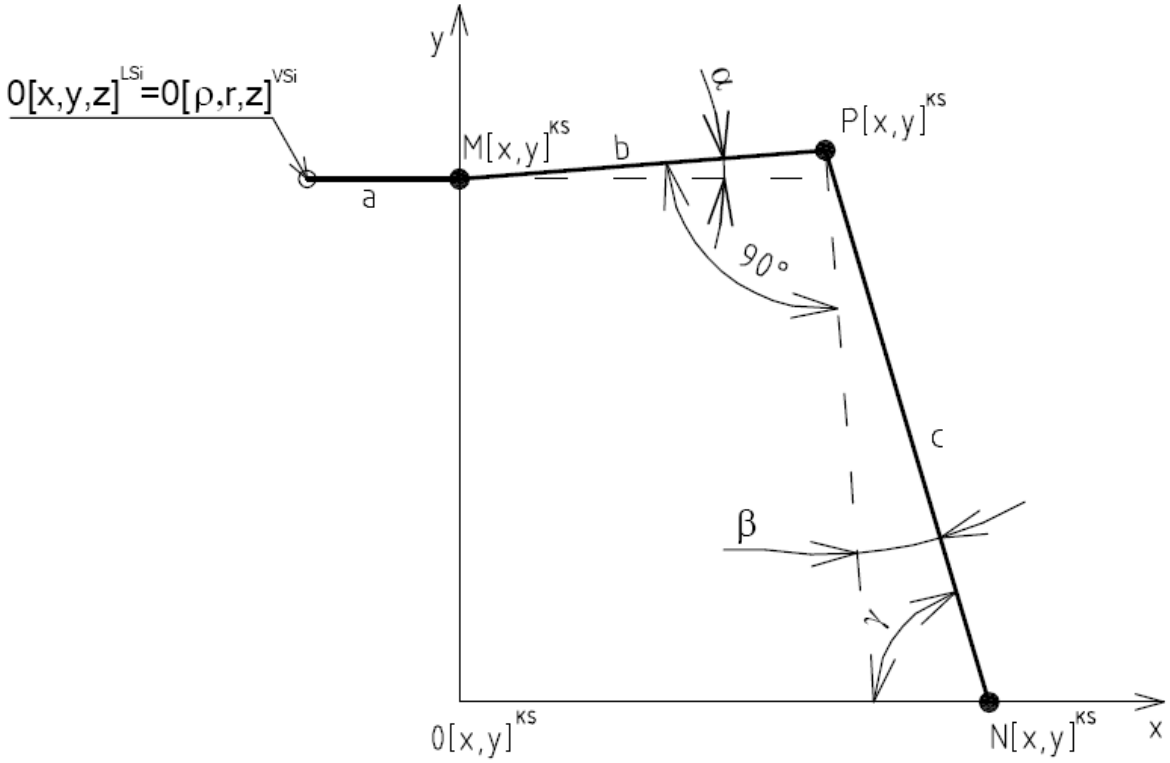


Obr. 11 Lokální válcový souřadný systém (zeleně) a úhly pravé nohy (červeně).



Obr. 12 Lokální válcový souřadný systém (zeleně) a úhly levé nohy (červeně).

Ve VS je úhel ρ roven úhlu servomotoru nula (ρ'). Zbylé dva úhly(α a β) servomotoru 1 a 2 se počítají v rovině řezu Obr. 10. V rovině řezu byl zaveden další Pomocný kartézský s. s. (2D) značený KS. Ten je posunut o hodnotu „*a*" od bodu $0[x, y, z]^{VSi}$ a jeho počátek je patrný z obrázku 13.



Obr. 13 Pomocný kartézský souřadný systém v rovině řezu.

Pro výpočet úhlu α potřebujeme znát souřadnice průsečíku P dvou kružnic (k_1, k_2), vedených z bodů M a N. Průnik kružnic se řeší soustavou dvou rovnic o dvou neznámých, které mají 2 řešení, z nichž jedno je na robotu neproveditelné.

$$\left(P_x^{KS}\right)^2 + (P_y^{KS} - z_{vs})^2 = b^2 \tag{3.4}$$

$$\underline{(P_x^{KS} - N_x^{KS})^2 + \left(P_y^{KS}\right)^2 = c^2} \Rightarrow \tag{3.5}$$

$$P_x^{KS} = \frac{N_x^{KS}}{2} + \frac{-N_x^{KS}\left(b^2 - c^2\right)}{2 \cdot d} + \frac{z_{vs}}{2 \cdot d} \cdot \sqrt{\left(\left(b+c\right)^2 - d\right) \cdot \left(d - (c-b)^2\right.} \tag{3.6}$$

$$d = \left(N_x^{KS}\right)^2 + z_{vs}^2 \tag{3.7}$$

Po získání x-ové souřadnice průsečíku P jsme již schopni y-ovou souřadnici vypočítat jednoduchou Pythagorovou větou.

$$P_y^{KS} = \sqrt{c^2 - (N_x^{KS} - P_x^{KS})^2} \tag{3.8}$$

Vztah pro úhel α je následující:

$$a = \arctan\left(\left(P_y^{KS} - N_y^{KS}\right) / P_x^{KS}\right) \tag{3.9}$$

Při výpočtu úhlu β se využívá poznatku, že součet úhlů čtyřúhelníku (0^{ks}, M, P, N) je roven 360°. Ve výpočtech byla záměrně použita goniometrická funkci arctan, protože jednak rozezná znaménko úhlu ale také z důvodu, že při realizaci výpočtu tuto jedinou funkci mám v úmyslu zjednodušit pomocí tabulkové metody výpočtu.

$$b = \frac{3.14159}{2} + a - g \tag{3.10}$$

$$g = \arctan(P_y^{KS} / (N_x^{KS} - P_x^{KS})) \tag{3.11}$$

Znaménka úhlů α a β se volí podle toho, na jaké noze je výpočet realizován viz Obr. 11 a 12.

Známé hodnoty (symbol ve vztazích 3.1-3.11)	Neznámé pomocné hodnoty (symbol ve vztazích 3.7-3.11)	Hodnoty, které chceme vypočítat (symbol ve vztazích 3.9-3.10)
Polární úhel (ρ_{VS})	Úhel u paty (γ)	Úhel servomotoru 0 (ρ')
Polární poloměr (r_{VS})	Souřadnice *x* průsečíku P (P_x^{ks})	Úhel servomotoru 1 (α)
Souřadnice (z_{VS})	Souřadnice *y* průsečíku P (P_y^{ks})	Úhel servomotoru 2 (β)
Rozměry nohy (a, b, c)	d – pomocná proměnná	
Souřadnice *x* bodu N (N_x^{ks})		

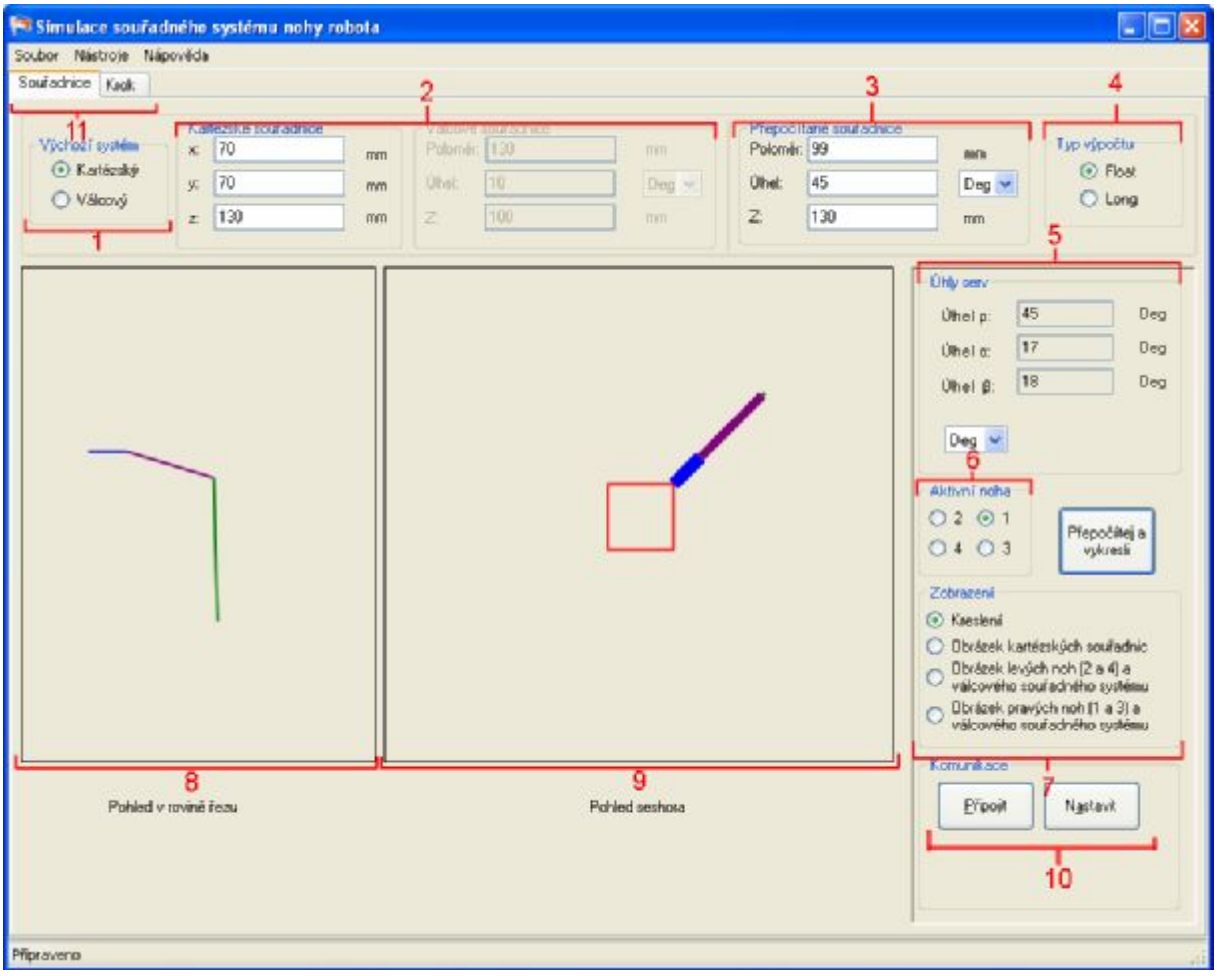
Tab. 1 Tabulka použitých proměnných

3.3 Simulační program

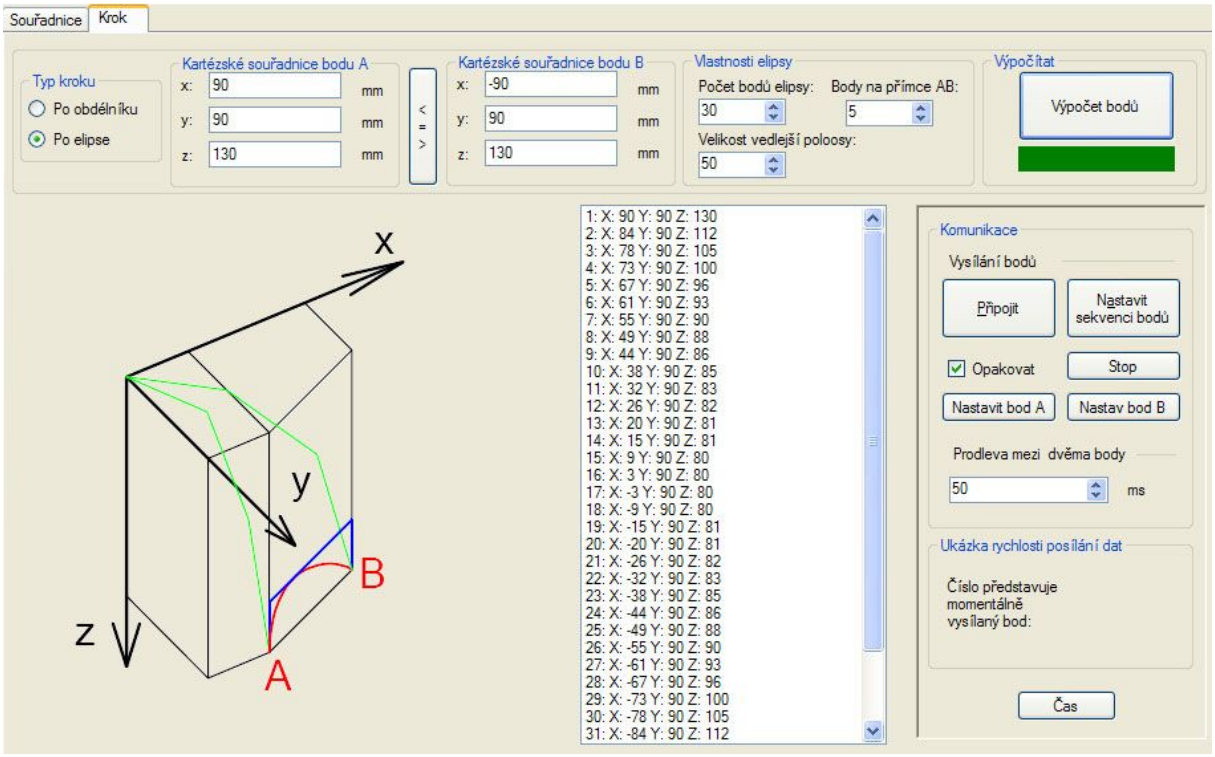
Pro návrh řídicího systému a pro ověření správnosti výpočtu popsaného v předešlé kapitole se jevilo jako rozumné řešení naprogramovat simulační program.
Cíle simulačního programu:

- Ukázat schematický obraz robotu ze dvou pohledů
- Ověřit správnost matematických výpočtů
- Informovat uživatele o přepočtech mezi jednotlivými souřadnými systémy
- Odzkoušet schopnosti pohybu jedné nohy robotu v prostoru do mezních poloh
- Komunikovat s obvody v reálném čase pomocí RS 232
- Generovat krok

Simulační program byl naprogramován ve vývojovém prostředí Visual Studio 2005 v jazyce C# [Si: ša:p].



Obr. 14 Ukázka ze simulačního programu.



Obr. 15 Generování chůze v simulačním programu.

3.3.1 Popis simulačního programu

Popis programu vychází z obrázku 14. Následující čísla vyjadřují odkazy v tomto obrázku.

1. Výchozí systém - Při práci s programem si uživatel může zvolit, který souřadný systém bude brán jako výchozí. Na výběr má ze dvou možností, buď Lokální kartézský souřadný systém (LS), označený v programu jako kartézský nebo Lokální válcový souřadný systém (VS), označený jako válcový.
2. Kartézské (válcové) souřadnice - Po výběru výchozího souřadného systému se zpřístupní jeden ze dvou groupboxů, do kterého uživatel zadá požadované hodnoty v konkrétním souřadném systému. V případě válcových souřadnic může uživatel zadat hodnotu úhlu, jak ve stupních, tak v radiánech.
3. Přepočtené souřadnice - Uživatel v programu vždy vidí přepočet z výchozího souřadného systému do druhého systému.
4. Typ výpočtu - Při realizaci programu byly naprogramovány dvě funkce. Jedna řeší výpočty v plovoucí desetinné čárce (float) a druhá používá celočíselné proměnné (long).
5. Úhly servomotorů - Po provedení výpočtu se zobrazí vypočítané úhly. Opět je možná konverze na jiné úhlové jednotky.
6. Aktivní noha - Pro výpočet úhlů je také důležité, aby program věděl, pro kterou nohu je počítá.
7. Zobrazení – Zde je možno zobrazit obrázky jednotlivých souřadných systémů a zakótování jednotlivých úhlů včetně zobrazení jejich polarity.
8. Pohled v rovině řezu – Pohled na nohu robotu v rovině řezu pro výše zadané souřadnice.
9. Pohled seshora – Pohled seshora na robota zobrazuje vždy jen aktivní nohu.
10. Komunikace – Zajišťuje spojení s robotem a posílá souřadnice bodů.
11. Krok – Záložky slouží k přepínání mezi krokem a výpočtem souřadnic bodů.

V programu dále naleznete menu, které obsahuje například formulář s maximálními hodnotami úhlů, kterých je robot schopen dosáhnout. Tyto úhly vymezují pracovní oblast noh robotu.

3.3.2 Simulace kroku

Byly navrženy a realizovány obě varianty kroku. Jak krok po obdélníku, tak po elipse. Do simulačního programu byla přidána záložka krok, kde je možno si vybrat ze široké palety nastavení stylu kroku viz Obr. 15.
Uživatel si zvolí typ kroku, buď obdélník nebo elipsu, pak zadá počáteční a koncový bod. Poté se mu odkryje příslušné nastavení.
U elipsy si volí:

- počet bodů elipsy
- velikost vedlejší poloosy
- body na úsečce AB

U obdélníku si volí:

- výšku obdélníku v ose *z*
- body na úsečce AB

Zadávat počet bodů na úsečce AB je nezbytné z důvodu, že skutečná dráha koncového bodu nohy při oporné fázi není přímka, ale oblouk – vyplývá to z konstrukce nohy. Abychom zajistili, že se noha bude vracet po přímce, nahradíme ji několika menšími úseky.
Dalším důležitým parametrem je čas mezi vysláním jednotlivých bodů. Tato časová prodleva dává komunikaci čas na to, aby vyslala souřadnice po sériové lince do MCU. V mikrokontroleru se poté tyto souřadnice přepočítají na úhly 3 servomotorů a o výsledku přepočtu je nazpět informován simulační program. Nejkratší doba, za jakou je schopna komunikace obsloužit sekvenci bodů, je 16ms. Tento čas je kritický a může se stát, že ne všechny body budou správně nastaveny, proto bylo za minimální prodlevu označeno 25ms.
Každý bod před vysláním je zkontrolován, jestli patří do pracovní oblasti robotu. Jestliže nějaký bod do pracovní oblasti nepatří, ve výčtu všech bodů je to za něj napsáno a zelený panel se změní v červený.

3.3.3 Popis tříd simulačního programu

Při tvorbě simulačního programu jsem se maximálně snažil využívat třídy obsažené v MS .NET framework. Neopomenul jsem ani robustnost programu.

3.3.3.1 Moje třídy

Nosnou třídou pro můj program je třída **LegFeatures**, která uchovává souřadnice bodu došlapu (válcové i kartézské), číslo právě použité nohy a hodnoty vypočítaných úhlů servomotorů. Dále jsou zde zahrnuty neřízené dynamicky linkované knihovny z jazyku C. Ty obsahují metody, které jsou naprogramovány pro řídicí systém robotu (přesněji pro jeho část, umístěnou v řídicí elektronice robotu) a v simulaci se používají pro ověření správnosti jejich návrhu. Přepisování těchto metod do jazyka C# by bylo zdlouhavé a zbytečné, jejich import je jednodušší. Odzkoušení metod v simulaci je hlavním důvodem jejich importu.

Aby simulační program mohl komunikovat s robotem a zadávat mu příkazy v reálném čase, bylo nutné naprogramovat třídu **Communication**, která slouží jako rozhraní mezi mojí .NET aplikací a komunikací po sériové lince. Samotná komunikace je naprogramována v C++ a její zdrojový kód jsem obdržel již hotový, pouze do ní byly začleněny 2 metody:
První: *PointCartesian_Set* přebírá kartézské souřadnice a vysílá je po sériové lince. Její návratová hodnota informuje aplikaci o tom, zda nedošlo při přenosu k chybě.

Druhá: *SetActiveLegOnMCU* přebírá jako parametr číslo nohy, ke které se vztahují vysílané souřadnice. Návratovou hodnotou je opět případný druh chyby.

Při zadávání souřadnic bodu došlapu je důležité, aby čísla byla celá, náležela do určitého intervalu, a aby velikost úhlu nepřesáhla určitý rozsah. Odůvodnění, proč jsem zvolil, že čísla budou celá vychází z toho, že robot nemá tak velkou rozlišovací schopnost, aby rozlišil údaj přesnější než 1mm. Pro účel kontroly vstupních dat byla naprogramována třída **Verification**, která kontroluje určitá editační pole a v případě špatně zadaných hodnot upozorní uživatele na chybu a poradí mu, jak ji odstranit.

Poslední třídou v mé aplikaci je třída **Gait**. Jak název napovídá, tato třída se stará o generování kroku. Umí realizovat pohyb po určité dráze (přímka, elipsa a obdélník). Třída uchovává sekvenci bodů, které tvoří dráhu kroku. Sekvence bodů je uchovávána v jednosměrně vázaném seznamu, proto není těžké jednotlivé části kroku spojovat a vytvářet tak ucelenu dráhu po různých křivkách.

Ke všem metodám naprogramovaným ve třídách Verification, LegFeatures, Communication a Gait byla napsána XML dokumentace, ve které je popsán účel metody, popis jejích parametrů a popis návratové hodnoty. Tento fakt umožňuje snazší orientaci v kódu a je pochopitelnější pro toho, kdo jej nenaprogramoval.

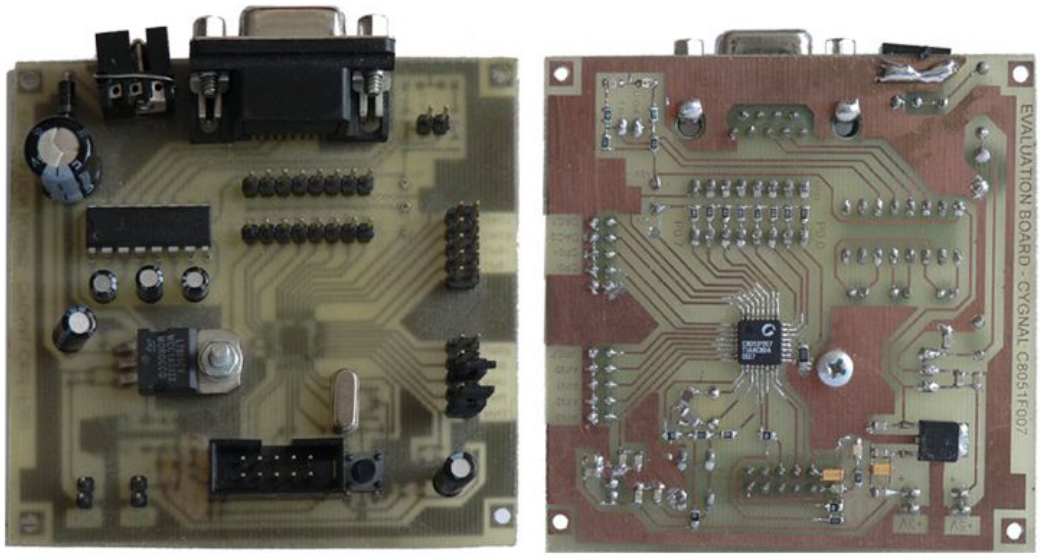
Statistika programu:

- přes 100 vizuálních komponent
- 14 kontrolních metod ve třídě Verification
- 18 výpočetních metod ve třídě LegFeatures
- více než 1000 řádků nevkládaného kódu

3.4 Implementace kinematického modelu do mikrokontroleru

Po zdárné simulaci bylo přistoupeno k tvorbě programu pro mikrokontroler. Tento mikrokontroler řídil jednu nohu robotu a byl přímo spojený po sériové lince se simulačním programem v PC. V budoucnu není problém rozšířit řídicí systém o další 4 mikrokontrolery a dosáhnout tak řídicího systému podle Obr. 6.

Řídicí systém robotu obsahuje mikrokontroler C8051F012. Součástí mikrokontroleru je 256B RAM paměti. Velikost paměti výrazně ovlivňuje způsob programování. Většina metod přebírá argumenty odkazem, proměnné jsou lokální a jsou uspořádávány do struktur, jejich význam je při výpočtu většinou několikanásobný. Zdrojový kód byl naprogramován v jazyce C ve vývojovém prostředí Visual Studa 2005. Pro překlad zdrojového kódu do strojového byl použit překladač Gnumake. Strojový kód byl poté nahrán do mikrokontroleru, umístěného na zkušební desce Obr. 16.



Obr. 16 Deska s čipem C8051 (pohled zepředu a zezadu).

3.4.1 Princip řízení

Systém řízení robotu byl navržen tak, aby akční subsystémy dostaly informace od nadřazeného systému (v mém případě počítače) a provedly operaci, pro kterou byly navrženy. Schéma řídicího systému je na Obr. 6.

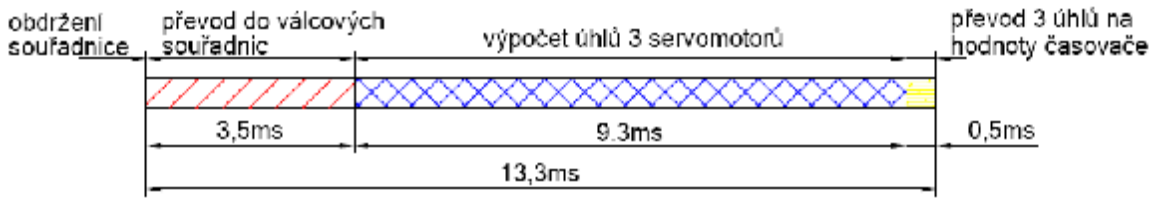
Princip řízení:

1) Na počítači se provede výpočet tvaru kroku a jiné náročnější operace
2) Cílové souřadnice došlapu se pošlou přes sériovou linku do příslušného subsystému

Následující dva body jsou řešením inverzní úlohy kinematiky.

3) V subsystému se převede bod došlapu na válcové souřadnice
4) Ve válcových souřadnicích se vypočítají 3 úhly pro 3 servomotory. Nelze vyloučit situaci, že se noha může dostat do zadaného bodu více způsoby, tedy nastane nekonečně mnoho možností nastavení 3 úhlů servomotorů pro jeden bod. V tomto případě mluvíme o nejednoznačné inverzní úloze kinematiky. Tento problém je nezbytné ošetřit. Řešení spočívá v tom, že nastavení úhlů vychází ze znalosti polohy předchozího bodu, na jejímž základě jsou nastaveny úhly pro bod, v němž je úloha inverzní kinematiky nejednoznačná.
5) V případě, že se vypočtené úhly nachází v pracovní oblasti robotu, nastaví se pro každý úhel odpovídající hodnota registru časovače. Délka časování odpovídá délce pulsu vyslaného do servomotoru. Podle délky pulsu se servomotor natočí o příslušný úhel. Jestliže nějaký úhel servomotoru zasahuje do těla robotu, ponechá se předchozí nastavení a o nedosažitelnosti je informována nadřazená vrstva.

Doba trvání od bodu tři po bod pět je znázorněna na časové ose viz Obr. 17.

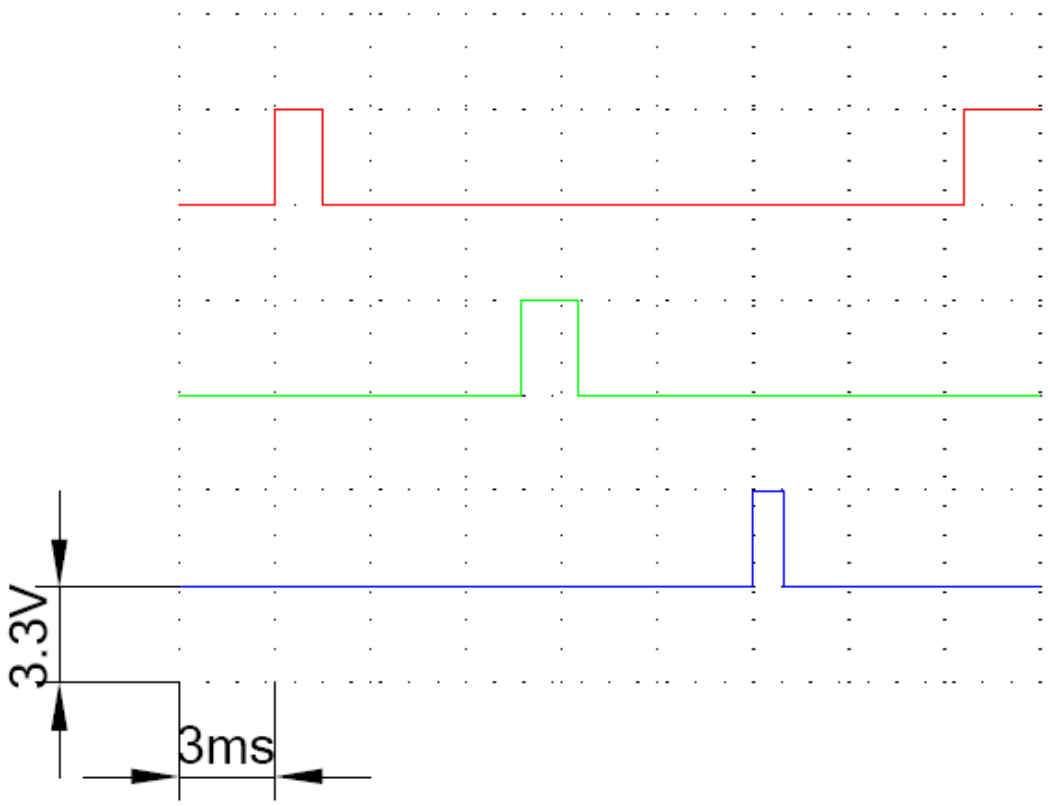


Obr. 17 Časová osa výpočtů v mikrokontroleru.

Nejdéle trvaly výpočty úhlů ve válcových souřadnicích. Je to způsobeno tím, že v matematických výpočtech bylo potřeba použít třikrát goniometrickou funkci arctan. Výpočet funkce arctan tvoří značnou část výpočtového času. Zkoumal jsem i možnost naprogramovat si vlastní funkci arctan pomocí metody CORDIC [8], ale po jejím vytvoření jsem se nedočkal zkrácení času výpočtu, naopak se zvětšily paměťové nároky. Proto jsem ponechal původní funkci nezměněnou.

Pokoušel jsem se i o převedení výpočtů z plovoucí desetinné čárky na celočíselné výpočty. Výpočty v FX (fixed point) byly stejně rychlé jako ty ve FP (floating point) a navíc stejně odmocnina a arctan musely být realizovány v plovoucí desetinné čárce (FP). V simulačním programu je možno vybrat si obě možnosti a vyzkoušet si tak přesnost obou metod výpočtu. Pro mikrokontroler byla ovšem ponechána pouze verze v plovoucí desetinné čárce. Jelikož proměnné mají během výpočtu velký rozsah, je matematika s pevnou desetinnou čárkou (FX) naprosto nevhodná pro mikrokontroler.

Postup nastavení šířky pulsu servomotoru:
Servomotor přijímá pulsy od 1ms (-50°) přes 1,5ms (0°) do 2ms (+50°) a podle šířky pulsu natočí hřídel do určitého úhlu. Prodleva mezi pulsy jednoho servomotoru je cca 20ms. Na obrázku 18 jsou vidět 3 řídicí signály 3 servomotorů.



Obr. 18 Šířkově modulované pulsy pro 3 servomotory jedné nohy.

Šířka pulsu je tvořena pomocí časovače. Rozborem bylo zjištěno, že je třeba použít dva 16bitové časovače. První z nich měl dobu časování 6ms a nastavoval vždy jeden ze 3 pinů do stavu log. 1. Druhý časovač určoval šířku pulsu podle doby jeho časování. Hodnota, od niž

bude druhý časovač časovat (čítat vzestupně) až do přetečení a vyvolání přerušení, je závislá na taktu procesoru a na módu časovače. V mém případě byl použit mód, kdy časovač využíval jednu dvanáctinu systémového času.
Z výše uvedeného plyne, že při taktu 22,1184MHz bylo využito časovačem 22,1184/12 = 1,8432MHz.
To znamená, že čítání o jeden bit trvá 1/1843,2kHz = 0,000543ms.
K vytvoření pulsu o délce 1,5ms potřebujeme tedy čítat 1,5/0,000543 = 2762 bitů.
Vzhledem k tomu, že časovač čítá vzestupně, je nutno počet čítajících bitů odečíst od $FFFF_{16}$ + 1, což je okamžik, kdy dojde k vyvolání přerušení. Tedy hodnota ***p,*** od které bude registr časovače časovat při zadání úhlu servomotoru ε v radiánech, je rovna:

$$p = FFFF_H - \frac{1,5 + \frac{e \cdot 57.29578}{100}}{0,000543} \tag{3.12}$$

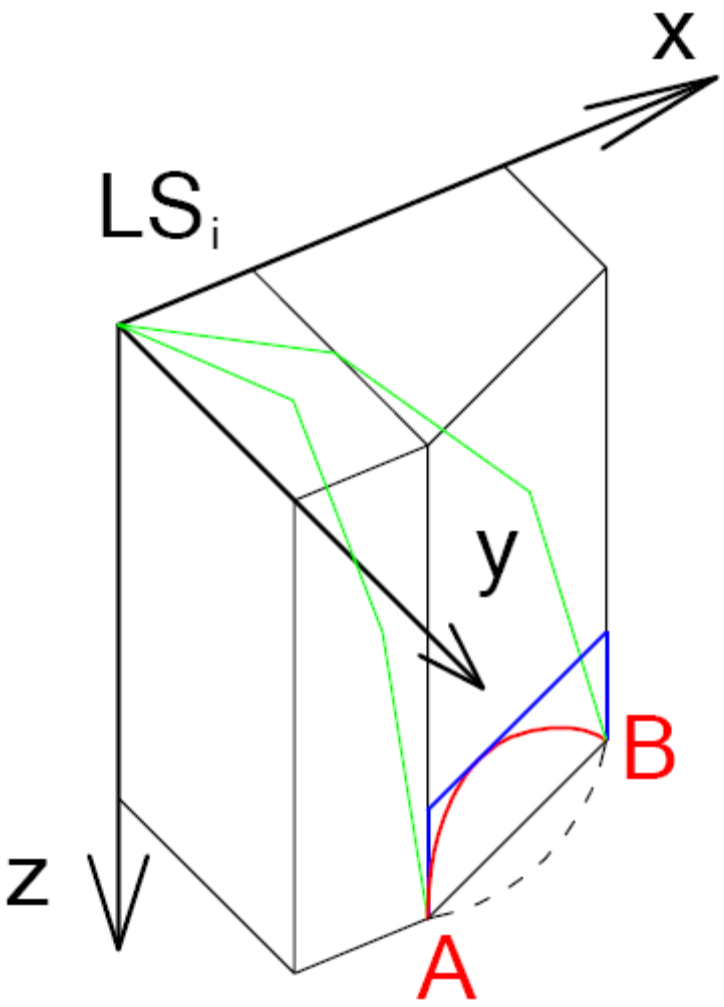
Matematická operace v mikrokontroleru s číslem větším než $FFFF_{16}$ by byla časově náročnější, proto se přičtení jedničky ve vztahu neuplatňovalo, přesnost výpočtu tím nebyla ovlivněna.

3.5 Krok robotu

Dráha koncového bodu nohy při provádění kroku byla zvolena ve tvaru obdélníku (modrá) a ve tvaru elipsy(červená) Obr. 19. Každá varianta má své výhody i nevýhody. Krok po obdélníku je méně náročný na výpočet, skládá se jen ze čtyřech bodů. Jeho nevýhodou je delší trajektorie pohybu koncového bodu nohy. Krok po elipse je kratší a tvoří jej zpravidla více bodů.

Možností, jak definovat krok po elipse v prostoru, je několik. Například UT Austin Villa robot soccer team definuje u robotu Aibo 4 parametry [13]:
1) délka elipsy
2) výška elipsy
3) pozici elipsy na ose x
4) pozici elipsy na ose y

V mém případě je elipsa jednoznačně dána:
1) počátečním bodem A
2) koncovým bodem B
3) velikostí vedlejší poloosy g



Obr. 19 Varianty kroku nohy robotu v prostoru.

3.5.1 Výpočet jednotlivých bodů elipsy

Vypočítat body elipsy v prostoru je poměrně složité. V mém případě to bylo jednodušší, protože elipsa ležela vždy v rovině kolmé na rovinu xy_{LSi}. Dalším zjednodušením bylo, že body A a B měly stejnou souřadnici *z*. Tyto zjednodušující podmínky sice neumožňují některé složitější realizace kroku, jako je například chůze do schodů, ale základní chůzi nijak neomezují.

Elipsa je dána body A, B a velikostí vedlejší poloosy *g*. Elipsa se nachází vždy v rovině dané body A, B a směrovým vektorem $\mathbf{u} = (0,0,1)$.

Postup výpočtu:
1. Výpočet velikosti úsečky dané body A, B.

$$|AB| = \sqrt{A_x^2 + B_x^2} \tag{3.13}$$

2. Velikost úsečky AB se podělí počtem bodů *f*, které má obsahovat elipsa.

$$s = \frac{|AB|}{f} \tag{3.14}$$

3. Zjistí se úhel odklonu od osy x

$$d = \arccos\left(\frac{B_x - A_x}{|AB|}\right) \tag{3.15}$$

4. Pro jednotlivé body elipsy platí:

$$A_{xj} = A_x + \cos(d)\cdot s\cdot f \tag{3.16}$$

$$A_{yj} = A_y - \sin(d)\cdot s\cdot f \tag{3.17}$$

$$A_{zj} = A_z - \sqrt{g^2 - \frac{\left(\frac{|AB|}{2}\cdot s\cdot f\right)^2\cdot g^2}{\frac{|AB|}{2}}} \tag{3.18}$$

3.5.2 Výpočty jednotlivých bodů obdélníku

Obdélník je dán body A, B a výškou *e*. Obdélník se nachází vždy v rovině dané body A, B a směrovým vektorem $\mathbf{u} = (0,0,1)$.

Realizace kroku po obdélníku umožňuje i chůzi do schodů, protože u obdélníku je možno zadat různou hodnotu souřadnic z_{LSi} počátečního a koncového bodu.
Výpočet bodů A' a B' je následující Obr. 20:

$$|AB| = \sqrt{(B_x - A_x)^2 + (B_y - A_y)^2 + (B_z - A_z)^2} \tag{3.19}$$

$$h = \arcsin\left(\frac{B_z - A_z}{|AB|}\right) \tag{3.20}$$

$$m = \arctan\left(\frac{B_y - A_y}{B_x - A_x}\right) \tag{3.21}$$

$$A'_x = A_x + \cos(m)\cdot\sin(h)\cdot e \tag{3.22}$$

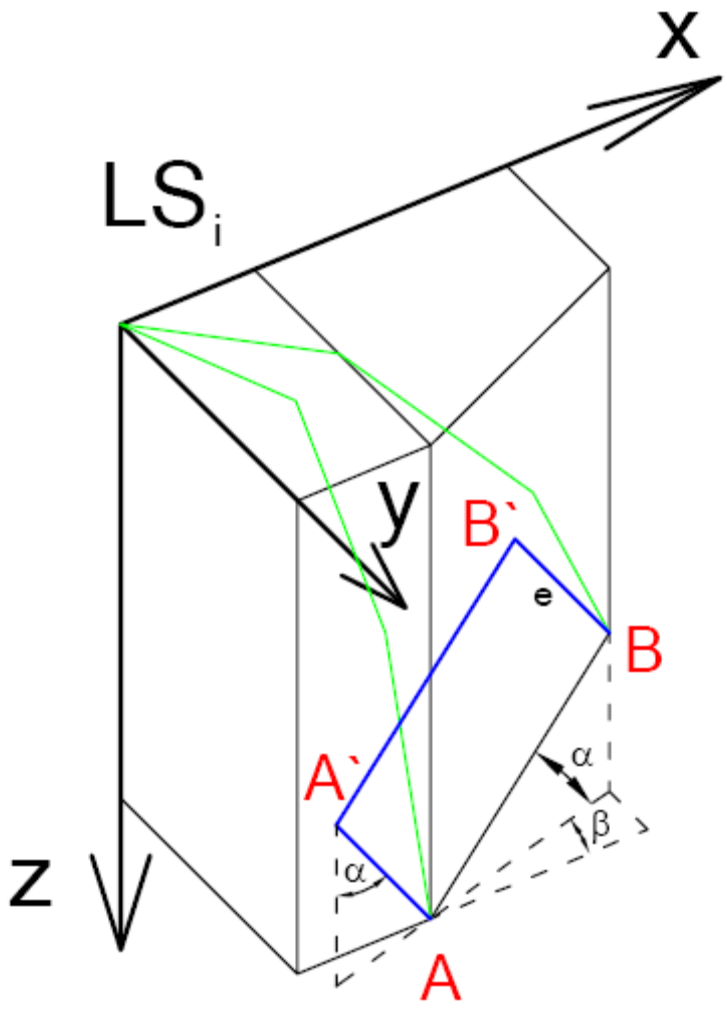
$$A'_y = A_y + \sin(m)\cdot\sin(h)\cdot e \tag{3.23}$$

$$A'_z = A_z - \cos(h)\cdot e \tag{3.24}$$

$$B'_x = B_x + \cos(m)\cdot\sin(h)\cdot e \tag{3.25}$$

$$B'_y = B_y + \sin(m)\cdot\sin(h)\cdot e \tag{3.26}$$

$$B'_z = B_z - \cos(h)\cdot e \tag{3.27}$$



Obr. 20 Krok po obdélníku v prostoru

3.5.3 Výpočty jednotlivých bodů oporné fáze kroku

Při návratu nohy z bodu B zpět do bodu A po provedení kroku (oporná fáze), je ve většině případů důležité, aby dráha byla přímka. Ve skutečnosti to ale přímka není, jedná se o křivku, která je na Obr. 19 znázorněna čárkovaně. Proto úsečku AB nahradíme několika body, čímž docílíme přímočařejší oporné fáze kroku.

Výpočet bodů na úsečce AB ($X_1 \dots X_i$) byl řešen pomocí parametrické rovnice přímky (3.34-3.36). Parametrická rovnice přímky byla zvolena, neboť díky ní lze snadno určit souřadnice bodů X_i na úsečce AB. Index *i* značí pořadí příslušného bodu na přímce z celkového počtu požadovaných bodů *j*. Je-li parametr *t* roven nule, pak je bod X roven bodu A.

Je-li parametr *t* roven jedné, pak je bod X roven bodu B.

Je-li parametr *t* v intervalu (0; 1), pak bod X leží mezi body A, B.

$$X = A + t\vec{u} \tag{3.28}$$

$$t \in (0;1) \tag{3.29}$$

Pro $\vec{u}$ platí:

$$\vec{u} = B - A \tag{3.30}$$

$$\vec{u} = (u_1, u_2, u_2)$$

$$x = a_1 + t\vec{u}_1 \tag{3.31}$$

$$y = a_2 + t\vec{u}_2 \tag{3.32}$$

$$z = a_3 + t\vec{u}_3 \tag{3.33}$$

Pro parametrickou rovnici platí:

$$x = a_1 + t\vec{u}_1 \tag{3.34}$$

$$y = a_2 + t\vec{u}_2 \tag{3.35}$$

$$z = a_3 + t\vec{u}_3 \tag{3.36}$$

Pro *i-tý* bod úsečky AB platí:

$$X_x^i = A_x + \frac{u_1 \cdot i}{j} \tag{3.37}$$

$$X_y^i = A_y + \frac{u_2 \cdot i}{j} \tag{3.38}$$

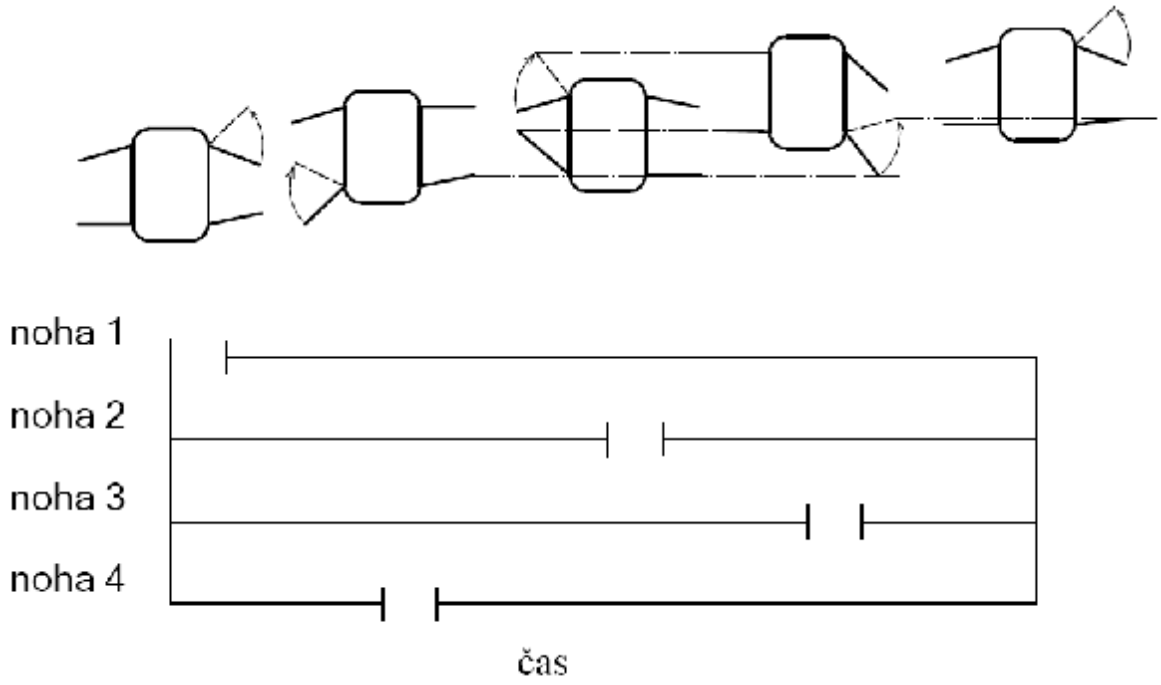
$$X_z^i = A_z + \frac{u_3 \cdot i}{j} \tag{3.39}$$

3.6 Návrh chůze robotu

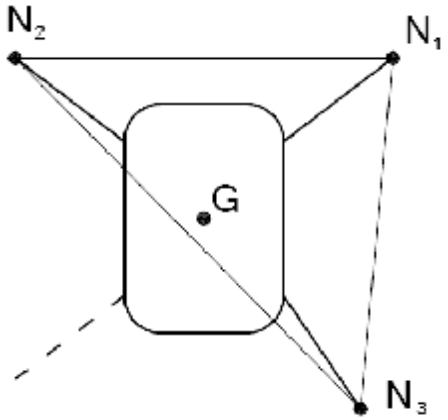
Posloupnost kroků noh, zajišťující požadovaný pohyb trupu vzhledem k terénu, je chůze.[16]

Návrhem chůze rozumím časové a prostorové zadání kroků jednotlivých noh. Volím chůzi, u které se doba součtu cyklů kroků jednotlivých noh bude rovnat době cyklu chůze a v čase se nebude měnit. Tomuto stylu chůze se říká Pevná cyklická chůze. Návrh posloupnosti kroků chůze a diagram chůze jsou zobrazeny na Obr. 21.

Návrh chůze robotu vychází z předpokladu statické stability. Motivaci k použití statické chůze podporuje fakt menší výpočtové náročnosti a jednoduchost návrhu. Při řešení statické stability vycházím z konstrukce tzv. oporného trojúhelníku Obr. 22.



Obr. 21 Návrh posloupnosti kroků chůze a Diagram chůze.



Obr. 22 Příklad oporného trojúhelníku

3.7 Přiložené zdrojové kódy

Podstatnou součástí mé bakalářské práce jsou zdrojové kódy. Jedná se především o zdrojový kód simulačního programu, jehož součástí jsou 3 projekty. První z nich, jménem *Simulace*, obsahuje grafickou část celého programu, zajišťuje vykreslování pohledů a další grafické prvky. Druhý projekt je dynamicky linkovaná knihovna *SimulationOfLeg* (symath.dll), která zahrnuje třídy popsané v kapitole 3.3.3. Poslední projekt je také DLL knihovna jménem *Servo* (servo.dll), obsahující metody, naprogramované v jazyku C.

Na přiloženém CD se nacházejí výše zmíněné projekty, jak ve formě zdrojového kódu, tak již přeložené do MSIL. Součástí jsou další dvě knihovny, deviceInterface.dll a mechLabControls.dll, které jsou nezbytné k zajištění komunikace, avšak jejich zdrojové kódy nejsou na CD, protože na jejich vývoji jsem se nepodílel. Samotný program lze spustit na libovolném počítači, na němž byl nainstalován MS .NET framework 2.0 nebo vyšší verze.

4 ZÁVĚR

V této práci byla rozebrána problematika spojená s chůzí robotu, včetně rešeršní studie. Byly zde popsány jednotlivé vývojové části řídicího systému včetně jeho hlavních funkcí. Podstatná část byla věnována návrhu souřadných systémů. Jejich správná volba pak ovlivnila řešení inverzní úlohy kinematiky a v neposlední řade i přehlednost a srozumitelnost prostorových souřadnic. Nejvíce pozornosti bylo věnováno simulačnímu programu, který umožnil ověření matematického modelu. Zároveň však sloužil jako přímé spojení s robotem a jako generátor jednoduchého kroku. V části věnované nejdůležitějšímu členu řídicího systému, tedy mikrokontroleru, který obsluhuje jednu nohu, byl popsán princip řízení servomotorů a problém s omezující velikostí paměti MCU. Nakonec byl ovšem kód natolik optimalizován, že dokonce některá paměť zbyla. Bylo zde i vysvětleno programové ošetření případů, ve kterých inverzní úloha kinematiky vychází nejednoznačně.

Poté, co došlo k úspěšnému dokončení simulace a zprovoznění řídicího subsystému, následovalo komunikační propojení obou, dosud oddělených, částí řídicího systému. Následně byl robot schopen přijímat jednotlivé souřadnice za běhu. Další fází bylo sestavení sekvence bodů, tvořících krok a zvolení vhodné prodlevy mezi jejich vysíláním. Po splnění i tohoto kroku už byla jedna noha robotu schopna bez potíží vykonávat krok. V budoucnu by bylo možné rozšířit řídicí systém o další mikrokontrolery a řídit všechny nohy současně.

PODĚKOVÁNÍ

Publikovaných výsledků bylo dosaženo za podpory Ministerstva školství, mládeže a tělovýchovy České republiky, výzkumný záměr MSM 0021630518 "Simulační modelování mechatronických soustav".

5 POUŽITÁ LITERATURA

[1] ROZSÍVAL V. *Návrh mobilního robotu s pásovým podvozkem pro inspekční činnost.* Vysoká škola báňská Technická univerzita Ostrava. Fakulta strojní, 2002. 52 s., Vedoucí diplomové práce Doc.Dr. Ing. Vladimír Mostýn.

[2] JURÁSEK, **V**. *Simulace robotických systémů: Teoretické základy* [online]. 2001, [cit. 13.února 2008] . <http://www.vegateam.cz/projekty/utar/>

[3] MOSTÝN, V. - SKAŘUPA, J. *Teorie průmyslových robotů*. 1. vydání, VŠB TU Ostrava. 2000. 147 stran.

[4] ŠOLC, F. – ŽALUD, L. *Robotika.* Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. 2006. 144s.

[5] NOVÁK, P. *Mobilní roboty: pohony, senzory, řízení.* 1. vydání. Praha: BEN – technická literatura, 2005. 248 stran. ISBN 80-7300-141-1

[6] BAŘINKA, L. *Inverzní kinematika kloubových struktur.* [online]. 2001. [cit. 10. ledna 2007] <http://62.245.66.21/projects/>

[7] Lynxmotion, Inc. 2006, Robot EQ3-KT [cit. 15. února 2007] <http://www.lynxmotion.com/>

[8] TIŠNOVSKÝ, P. Metoda CORDIC a výpočet funkcí tan, atan a length. *Fixed point arithmetic*. [online]. *2006.* [cit. 10. března 2007]. <http://www.root.cz/clanky/metoda-cordic-a-vypocet-funkci-tan-atan-a-length/>

[9] BIOLEK, Z. *Perspektivy elektroniky: sborník přednášek.* [online]. 29. března 2007. [cit. 13.února 2008]. <www.roznovskastredni.cz/dwnl/pel2007/PEL07.pdf>

[10] KOŠNAR, K. Mobilní robotika. [online]. 2007. [cit.20. února 2008] <www.roznovskastredni.cz/dwnl/pel2007/06/Kosnar.pdf>

[11] JONES, L. J. – ROTH D. *Robot Programming: A practical guide to behavior-Based Robotics.* 1st printing TAB Electronic, 2003. ISBN 978-0071427784

[12] *Řízení robotu: Přesnost kinematického modelu.* [online]. [cit. 10. ledna 2007]. <http://cmp.felk.cvut.cz/cmp/courses/roblec/rizeni.pdf>

[13] KOHL, N. – STONE, P. *Policy Gradient Reinforcement Learning for Fast Quadrupedal Locomotion.* [databázové centrum]. The University of Texas at Austin. 2004. [cit. 25. února 2008]. <http://dialog.cvut.cz/>

[14] ZHANG, W. – GUO, S. – ASAKA, K. *Characteristics Analysis of a Biomimetic Underwater Walking Microrobot.* [databázové centrum]. International Conference on Robotics and Biometics. China. 2006. [cit. 25. února 2008]. <http://dialog.cvut.cz/>

[15] CLIFFORD, R. *Gait.* [online]. [cit. 25. února 2008]. <http://www.wheelessonline.com/ortho/gait>

[16] KUTÍLEK, P. *Řízení chůze mobilního robota v konfliktním prostředí*. Brno: Univerzita obrany, 2005. Školitel Zdeněk Krupka

PŘÍLOHY

Příloha č. 1 – Přiložené CD-R médium s obsahem

- elektronická podoba této bakalářské práce (\text\BP_Olsa.pdf)
- zdrojový kód simulačního programu (\source\noha\noha.sln)
- spustitelný software:
    - simulační program (\simulaceNoha\noha.exe)
- MS .NET framework 2.0 (\dot_NET_Framework2_0\dotnetfx2.exe)
- textový soubor s podmínkami pro spuštění simulačního programu (instrukce.txt)