

CONTROL OF EXPERIMENTAL WALKING ROBOT USING SIMULATING MODEL

R. Grepl, S. Věchet, M. Bezdíček, M. Švehlák, J. Chmelíček¹

Summary: *This paper describes the scheme of remote operator control of mobile walking robot. Control structure is appropriate for low level control of robot (full operator supervision) and for initial phase of robot design. Kinematic and dynamic models are built in Matlab–Simulink–SimMechanics environment. Computation speed of model allowed real–time control of robot. Adopted approach is tested on small experimental physical model - quadruped walking robot with four legs and 12 d.o.f. overall.*

Keywords

walking robot, mobil robot control, Matlab–SimMechanics

1 Úvod

Konstrukce a řízení kráčejících robotů představuje globálně velmi zajímavý zdroj vědeckých problémů. Hlavní motivací je potenciální zvýšení některých jejich užitných vlastností, srovnáme-li je s roboty s klasickým (kolovým, pásovým) podvozkem. Příkladem může být např. možnost pohybu v složitém terénu, překonávání překážek, pohyb po vertikálních plochách atd. Uvažovanými oblastmi nasazení mobilních robotů je inspekce a zásahy v nebezpečném prostředí, pyrotechnické práce, kosmonautika, vojenství.

S tím jak roste podobnost kráčejících robotů s jejich živými vzory (hmyz, zvířata,... lidé), zvyšuje se i obtížnost návrhu a řízení. K tradičním problémům mobilní robotiky (lokální a globální navigace, rozpoznávání překážek, tvorba map prostředí, počítačové vidění apod.) přibývají náročné otázky konstrukce pedového mechanismu, volby a optimalizace pohonů, uspořádání noh robotu (počet noh a počet stupňů volnosti), synchronizace pohonů a v neposlední řadě otázka stability robotu.

Při návrhu koncepce řízení mobilního robotu je nutné uvážit míru podílu operátora (člověka) na vlastním řízení robotu. Z tohoto hlediska je možno řízení rozdělit takto:

1. Plně operátorové řízení – zde máme u kráčejícího robotu na mysli přímé řízení pohybu jednotlivých noh operátorem. Vhodné při počáteční fázi návrhu, testování algoritmů a při nouzových a extrémních situacích (např. překonávání náročných překážek).



¹ Ing. Robert Grepl, Ing. Stanislav Věchet, Ing. Milan Bezdíček, Michal Švehlák, Ing. Jaroslav Chmelíček:
Laboratoř mechatniky a robotiky, ÚT AV ČR a ÚTMB FSI VUT v Brně,
Technická 2, 616 69, Brno, e-mail: grepl@umt.fme.vutbr.cz

2. Částečně autonomní řízení – operátor zadává lokální cíle činnosti (přemístění nebo otožení celého robotu, průzkum okolí robotu apod.), jejich realizaci obstará řídicí systém robotu sám. Operátor může v případě potřeby zasáhnout. V případě krácejícího robotu to může znamenat, že operátor zadává směr pohybu robotu, samotnou chůzi však robot vykonává sám.
3. Plně autonomní činnost – robot sám zná globální cíl chování a sám provádí kroky k jeho splnění.

Zcela autonomní činnost robotu ve složitém prostředí a s náročnými požadavky chování je obtížně realizovatelná, pro řadu úloh však postačuje malá míra samostatnosti robotu – např. pro pyrotechnické aplikace je teleoperátorové řízení vhodné.

Dalším souvisejícím problémem je implementace řídicích algoritmů vyvíjených a testovaných na PC v prostředí mikrokontroleru robotu. Klíčovou otázkou je zde spotřeba energie, ale i časová (a tím i personální a finanční) náročnost převodu kódu. Často se proto přesun „intelligence“ do vlastní řídicí jednotky robotu provádí postupně.

2 Formulace problému

Doposud se naše práce prováděné v rámci výzkumných projektů soustředily na návrh kinematických inverzních modelů, statické a dynamické analýzy, dále pak na tvorbu a identifikaci tepelných modelů. Některé výsledky byly zveřejněny v článcích [8, 9, 10, 7]. Získali jsme takto určité zkušenosti, které jsou využívány při stavbě čtyřnohého robotu projektu AV ČR [5, 6]. Pro rychlé praktické ověření některých algoritmů a výpočetních metod jsme dále přistoupili k stavbě malého experimentálního robotu. Vývoj řídicího software není stále ještě uzavřen.

Současně se započítím stavby fyzikálního modelu se objevily úvahy o způsobu řízení. Tento příspěvek se věnuje popisu prvotního testovacího dálkového řízení robotu operátorem, které bylo realizováno v prostředí Matlab–Simulink.

Požadavky na řídicí systém byly definovány takto:

- malá časová náročnost tvorby testovacího řídicího systému
- implementace inverzního kinematického modelu celého robotu
- implementace bloků pro komunikaci s robotem po sériové lince
- využití dynamického modelu pro určení stability robotu
- simulační model musí pracovat v reálném čase na dostupném počítači PC

Hlavním cílem návrhu tohoto řízení bylo testování a úprava řídicího kódu v mikrokontroleru a také vlastní mechanické konstrukce (upevnění servopohonů, měření praktické zatížitelnosti robotu apod.).

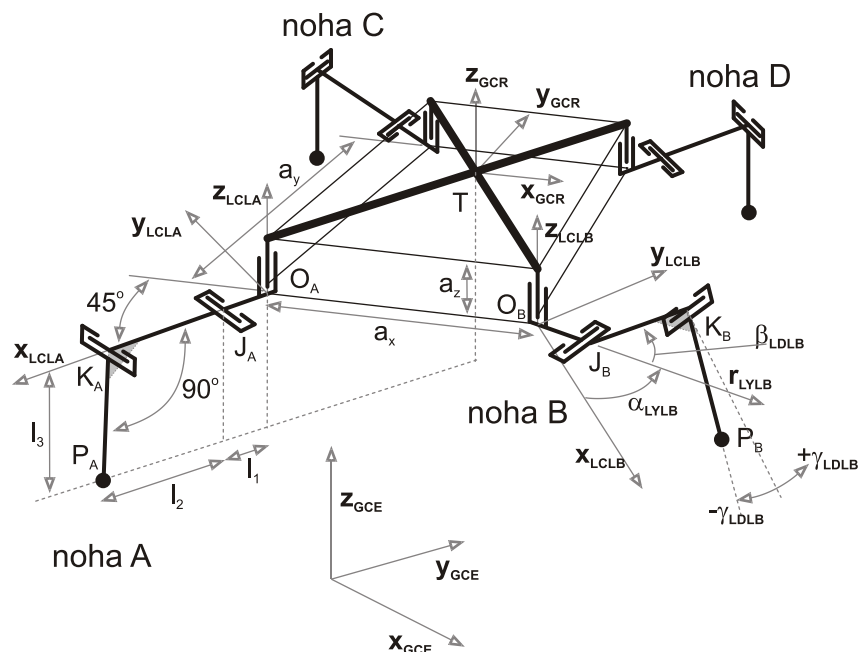
3 Postup návrhu řídicího modelu

3.1 Inverzní kinematický model založený na principu polohové regulace těla robotu

S respektováním prvních tří bodů požadavků formulovaných v předchozím odstavci jsme přistoupili nejprve k návrhu inverzního kinematického modelu (IKM) realizovaného čistě v Si-

mMechanics. Nebylo nutné formulovat přímý ani inverzní analytický model robotu nebo jeho nohy, postup je popsán např. v [9].

Pomocí definice geometrie jednotlivých částí robotu (obr. 1) a vazeb mezi nimi byl vytvořen model mechanismu. Vazby mezi konci noh a terénem jsou sférické. Neuvažujeme tedy prokluz nohy ani tuhosti podložky a koncového efektoru. V rotačních vazbách odpovídajících umístění servopohonů není zavedeno silové ani kinematické působení.



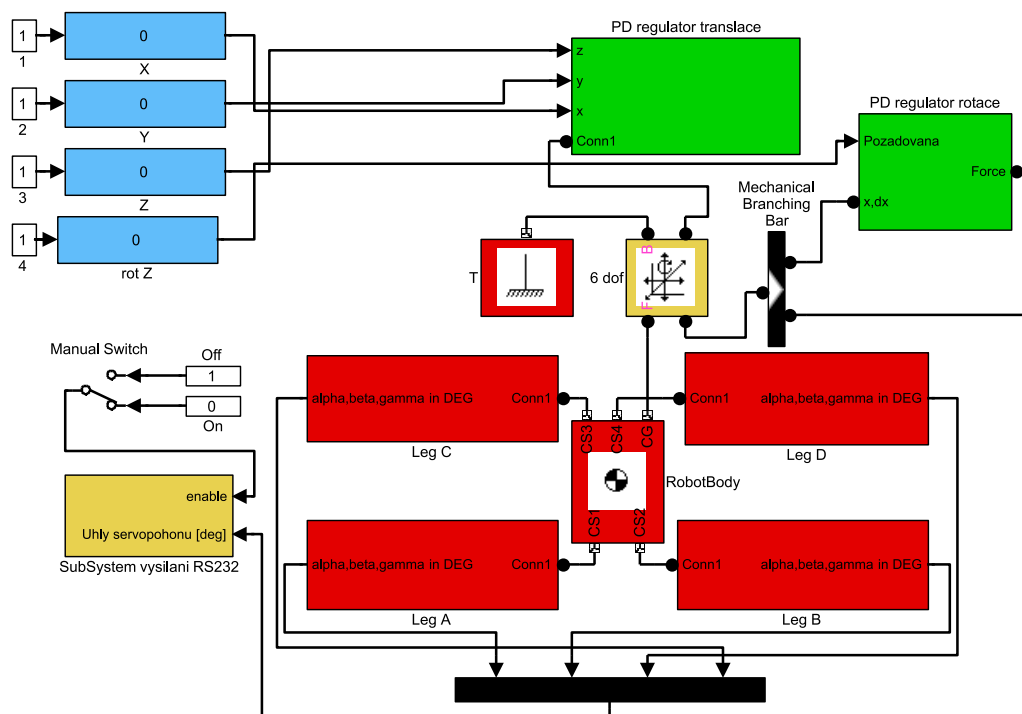
Obr. 1: Geometrie robotu a definice souřadných systémů

Tělo robotu je volně umístěno v prostoru a má tak šest stupňů volnosti. Pro každý z těchto stupňů volnosti je připojen PD polohový regulátor. Schéma simulačního modelu je uvedeno na obr. 2. Vlastní uživatelské rozhraní je provedeno Simulinkovskými prvky `slider gain`. Uživatel tak může jednoduše měnit polohu těla robotu a jeho natočení vůči glob. souř. systému. Model dále umožňuje provést změnu počáteční polohy robotu, ladění nulové pozice jednotlivých servopohonů (korekce konstrukčních a montážních nepřesností), provádění některých měření zatížitelnosti robotu apod. Není možné řídit chůzi robotu nebo samostatný pohyb jednotlivých jeho noh.

Pomocí bloků `sensor` jsou v příslušných rotačních vazbách odečítány hodnoty potřebného natočení servopohonů. Následně jsou upraveny a pomocí komunikačních bloků (popsány v odstavci 3.3) odeslány po sériové lince do řídicího mikrokontroleru robotu.

3.2 Struktura výsledného řídicího modelu v prostředí Simulink

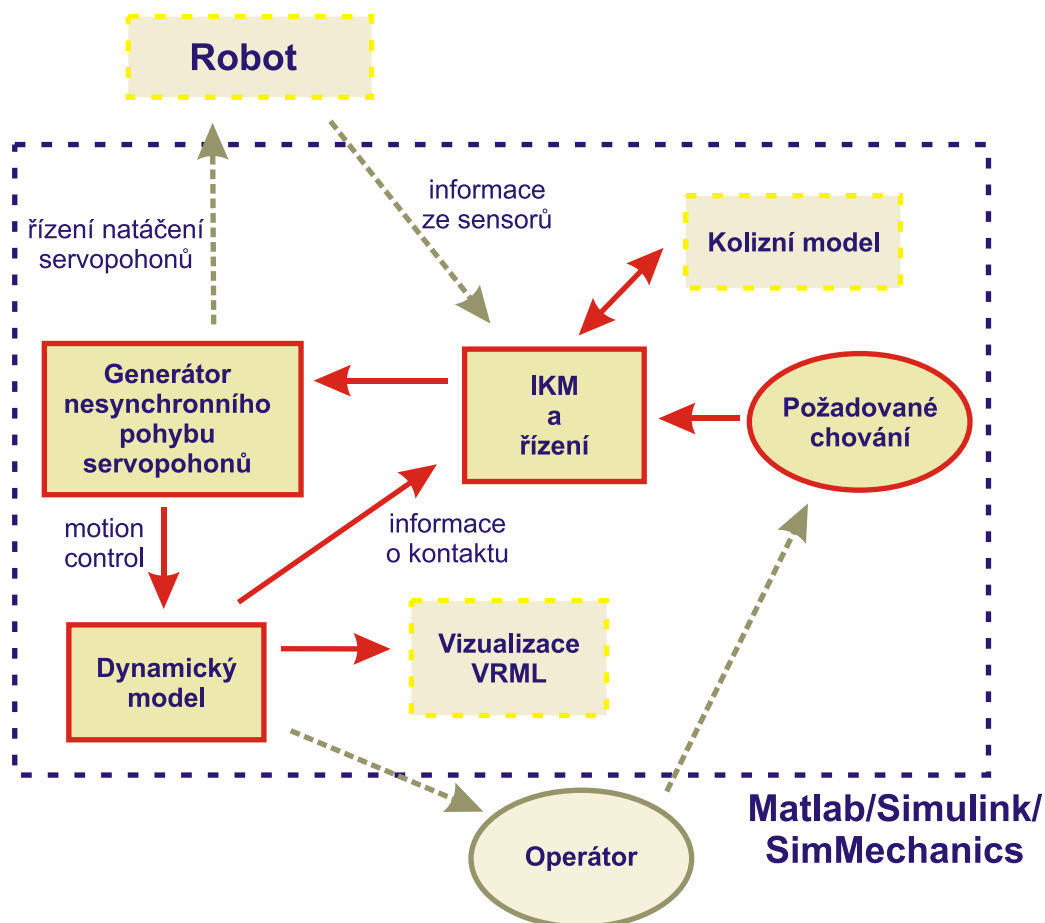
Po provedení experimentů potřebných k úpravám konstrukce a programového kódu mikrokontroleru jsme se zabývali zdokonalováním řídicího modelu. Dalším požadavkem byla implementace dynamického modelu pro určení stability (DyMS) robotu. Filosofie výpočtu je stejná jako v [7], problémem je návrh takové struktury modelu, která by umožnila výpočet i vizualizaci v reálném čase. Provedli jsme několik pokusů o rozšíření předchozího IKM o DyMS, ale rychlost simulace nebyla uspokojivá.



Obr. 2: Simulační schéma řídicího modelu založeného na polohové regulaci těla robotu

Výsledkem různých úvah a experimentů s kombinacemi bloků je struktura schématicky znázorněná na obr. 3. Odpovídající model v Simulinku je na obr. 4. Strukturu modelu lze charakterizovat takto:

- IKM je realizován pomocí bloku `matlab fcn`. Zde se provede výpočet IKM realizovaného vícevrstvou dopřednou neuronovou sítí a vykonají se potřebné transformace kinematických veličin. Z důvodů omezení náročnosti výpočtů je tato funkce volána pouze v případě, kdy algoritmus potřebuje získat souřadnice uzlových bodů trajektorie. Mezi těmito body se servopohony pohybují nesynchronizovaně pomocí lineární interpolace. Pokud je vzdálenost uzlových bodů dostatečně malá (se zřetelem k tuhosti celé konstrukce robotu), není tento nesynchronní pohyb na závadu celkovému chování robotu. Simulace dynamického modelu v prostředí SimMechanics a popsané řídicí bloky mají zcela odlišné požadavky na velikost simulačního kroku. Proto jsou bloky IKM a generování nesynchronního řízení uzavřeny do bloků `enabled and triggered subsystem`, které toto i na základě vnějších signálů (dosažení uzlového bodu, ztráta stability) ošetří.
- Generátor nesynchronního pohybu působí kinematicky `motion actuating` pomocí bloků `actuator` na jednotlivé rotační vazby (servopohony) DyMS v SimMechanics. Tento je zcela shodný s modelem popsaným v předchozím odstavci 3.1. Rozdíl je pouze v jeho vazbou s terénem. Místo sférické vazby je robot uložen na lineárních pružinách a tlumičích. Tím je vyřešen problém kinematické přeurčenosti soustavy. Blíže je problematika popsána v [7]. Vazba pružinou a tlumičem je realizována jako kontaktní úloha (síla se přenáší pouze pokud je síla kladná).
- Velikostí sil v kontaktu nohy s terénem je dána míra stability robotu. Hodnoty lze použít zpětně v řídicím algoritmu.



Obr. 3: Schéma struktury řídicího modelu realizovaného v Simulink

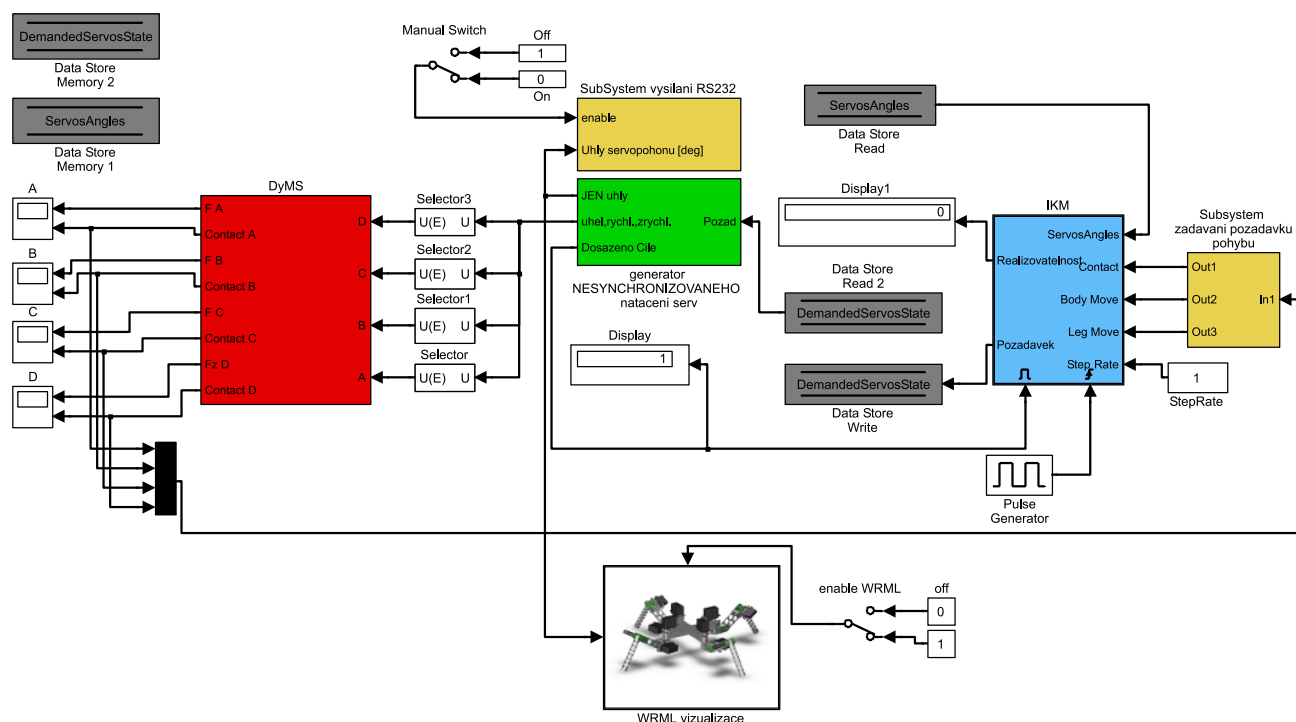
- Pokud jde o samotné možnosti zadávání požadovaného chování robotu, vycházíme z předpokladu, že veškeré možné akce robotu lze při předpokladu stabilního pohybu sestavit z pohybu těla robotu (kdy jsou tři nebo čtyři nohy jsou v kontaktu s terénem) a přemístění volné nohy. Z časových důvodů (vývoj algoritmu) jsme omezili pohyby těla robotu na translaci ve směru tří os a rotaci kolem svislé osy z , další rozšíření o zbývající dvě rotace ale není problémem.

3.3 Ostatní bloky simulačního modelu

Velmi stručně se ještě zmíníme o částech simulačního modelu, jejichž popis přesahuje rámec tohoto příspěvku, ale ve schématu se vyskytují.

RS232 Blockset je volně dostupná množina bloků navržených pro Matlab–Simulink. Vytvořil ji Leonardo Daga a lze ji najít na adrese:
<http://digilander.libero.it/LeoDaga/Simulink/RS232Blockset.htm>.

Vizualizace ve VRML je podrobněji představena v práci J. Chmelíčka na této konferenci [3]. Pomocí Virtual Reality Toolboxu pro Matlab je realizováno propojení simulačního modelu



Obr. 4: Výsledný simulační model pro řízení

s virtuálním světem. Nastavením přiměřené vzorkovací periody se dosáhne malého nároku na zdroje potřebné pro vizualizaci.

Řešení kolizí je opět popsáno v [3]. Robot je konstruován tak, že při složitějších pohybech může dojít ke kontaktu dvou sousedních noh, což je pochopitelně nežádoucí. Řídící jednotka nebo operátor o tom musí dostat informaci nebo musí být automaticky znemožněna realizace kolizní pozice robotu.

4 Závěr

V tomto příspěvku byl stručně nastíněn přístup k návrhu simulačního modelu v prostředí Matlab–Simulink určeného pro testovací úlohy řízení experimentálního čtyřnohého kráčejičícího robotu. Na počátku práce stál požadavek na maximální rychlost tvorby modelu, který se podařilo splnit, zvláště v jednodušším případě popsaném v 3.1. Na základě provedených experimentů byly odladěny komunikační a řídicí algoritmy v mikrokontroleru Atmel ATMEGA128 a provedeny některé úpravy konstrukce mechanické části robotu.

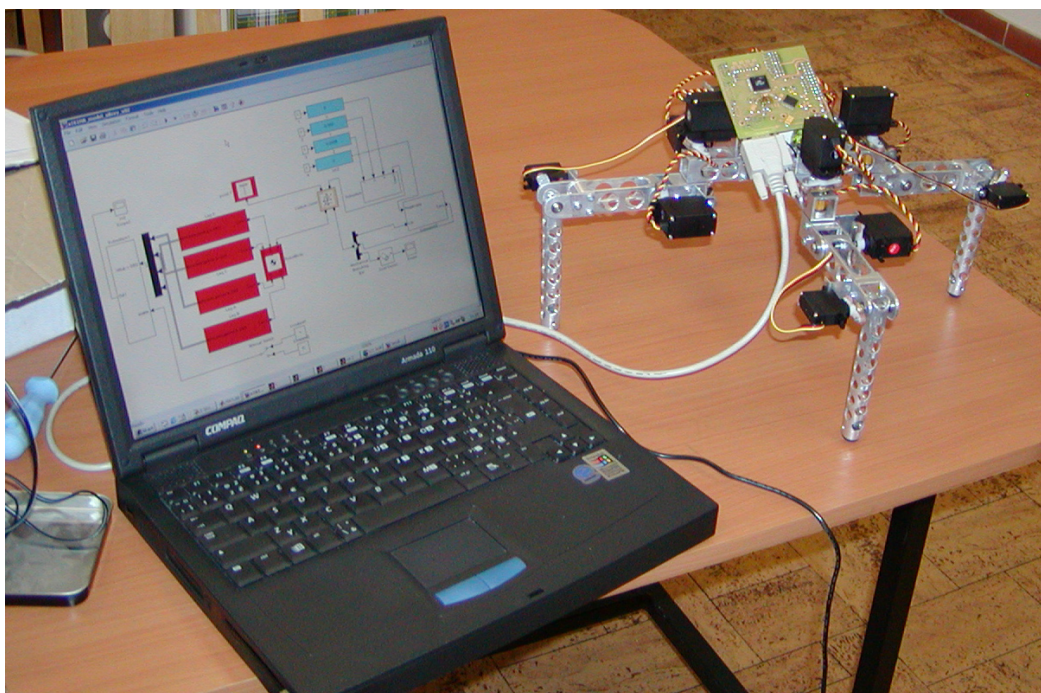
Další detailní informace jsou publikovány na této konferenci v příspěvcích [1, 2, 3]. Projekt je v současné době stále aktivní, některé výsledky může zvědavý čtenář najít také na internetových stránkách: <http://www.umt.fme.vutbr.cz/~rgrepl>.

Poděkování

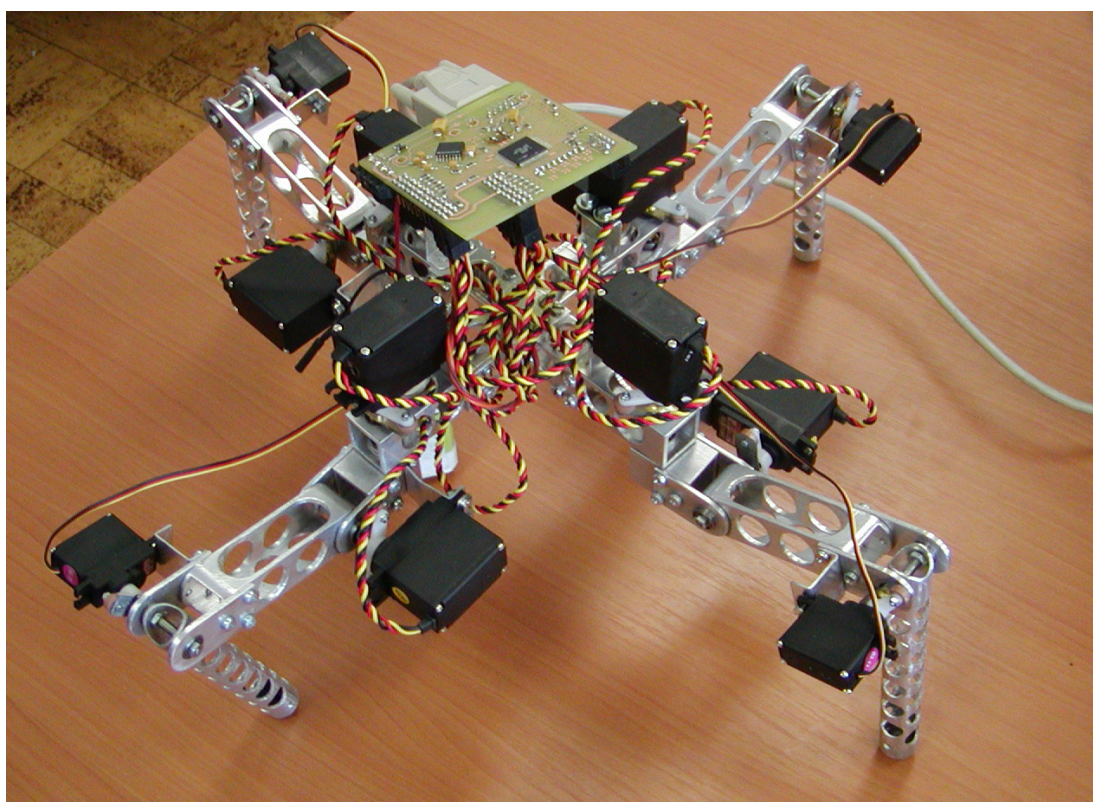
Práce vznikla za podpory pilotního projektu ÚT AV ČR č. 52020 „Realizace základních řídicích členů kráčivého robota“ a projektu MŠMT MSM 262100024 „Výzkum a vývoj mechatronických soustav“.

Reference

- [1] BEZDÍČEK, M., GREPL, R., ŠVEHLÁK, M., CHMELÍČEK, J.: *Artificial Neural Network Application To Walk Of A Four Legged Robot*, Inženýrská mechanika 2004, Svratka, Česká republika, 2004
- [2] ŠVEHLÁK, M., GREPL, R., VĚCHET, S., BEZDÍČEK, M., CHMELÍČEK, J. : *Design Of Small Laboratory Quadruped Robot*, Inženýrská mechanika 2004, Svratka, Česká republika, 2004
- [3] CHMELÍČEK, J., BEZDÍČEK, M., GREPL, R., ŠVEHLÁK: *Collision States Detection For Quadruped Robot Motion*, Inženýrská mechanika 2004, Svratka, Česká republika, 2004
- [4] WOOD, G.D.: *Simulating mechanical systems in Simulink with SimMechanics*, The MathWorks Inc., www.mathworks.com, 2002
- [5] HOUŠKA, P., BŘEZINA, T., SINGULE, V.: *Koncepce mechatronické soustavy nohy kráčivého robota*, Inženýrská mechanika 2002, Svratka, Česká republika, 2002
- [6] HOUŠKA, P., BŘEZINA, T., SINGULE, V.: *Control and synchronization drive unit of leg of walking robot*, Mechatronics, Robotics and Biomechanics 2003, international conference, Hrotovice, Czech Republic, 2003
- [7] GREPL, R.: *Dynamic model for stability control of walking robot*, Colloquium Dynamics of Machines 2004, Praha, 10.-11.2., 2004
- [8] GREPL, R.: *Kinematic and Dynamic Modelling of Quadruped Walking Robot*, Mechatronika 2003, 6th International Symposium On Mechatronics, 18.-20. června, Trenčianské Teplice, Slovensko, 2003
- [9] GREPL, R., ONDRUŠEK, Č.: *Inverzní kinematické modelování v SIMMECHANICS a jeho aplikace na robotické mechanismy*, Inženýrská mechanika 2003, 12.-15. května, Svratka, Česká republika, 2003
- [10] GREPL, R., KRATOCHVÍL, C.: *Robotic Leg Inverse Dynamic Modelling and Design Optimization in SimMechanics*, Mechatronics, Robotics and Biomechanics 2003, mezinárodní konference, Hrotovice, Česká republika, 2003
- [11] GREPL, R., KRATOCHVÍL, C.: *Inverzní kinematický model robotické nohy metodou Lazy Learning*, Výpočtová mechanika 2002, 29.-31. října, Nečtiny, Česká republika, 2002



Obr. 5: Ukázka propojení počítače s fyzikálním modelem během počátečních testů



Obr. 6: Detailní foto robotu během testování