

SIMULÁCIA, NÁVRH A DSP IMPLEMENTÁCIA RIADIACICH ALGORITMOV MOBILNÉHO ROBOTA

I. Masár

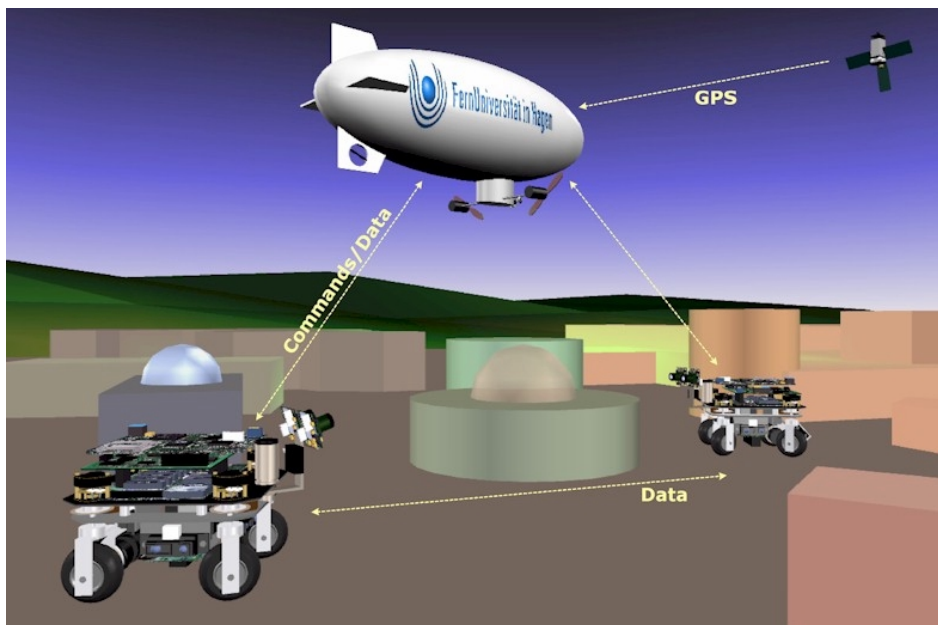
Department of Electrical Engineering
Control Systems Engineering Group, University of Hagen
Universitätsstr. 27, 580 97 Hagen, Germany
Ivan.Masar@FernUni-Hagen.de

Abstrakt

Návrh a testovanie riadiacich algoritmov autonómnych mobilných robotov prináša so sebou niekoľko problémov. Na jednej strane je to zložitosť a komplexnosť tejto úlohy, keďže na overenie navrhnutého riešenia väčšinou nestačia jednoduché simulácie samotného robota, ale je nutné modelovať aj jeho interakcie s okolím, prípadne s inými robotmi. Na druhej strane sa vynára problém s implementáciou a testovaním navrhnutých algoritmov, keďže robot musí okrem programov na riadenie pohybu vykonávať aj ďalšie úlohy, ako sú komunikácia, spracovanie dát zo senzorov vrátane spracovanie, a iné. Za účelom zjednodušenia tohto procesu sme vytvorili niekoľko podporných programov a simulačných modelov v prostredí MATLAB/Simulink, ktoré výrazne zrýchľujú a zjednodušujú jednotlivé kroky vývoja algoritmov mobilných robotov.

1 Vývoj multi-robotickej aplikácie

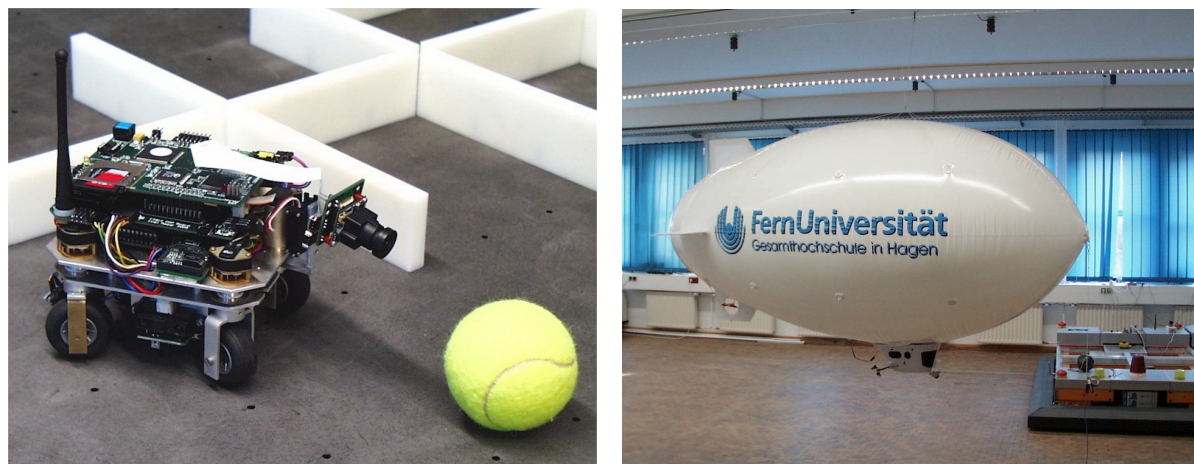
Súčasný ťažisko výskumu Katedry riadiacich systémov Univerzity v Hagene spočíva vo vývoji multi-robotickej aplikácie, pozostávajúcej z viacerých kooperujúcich autonómnych mobilných robotov. Využitím robotov rôzneho typu (pozemné, vzdušné), z ktorých každý môže získavať informácie o aktuálnej situácii pomocou rozličných senzorov je možné riešiť zadané úlohy aj v komplexnom neštruktúrovanom prostredí (Obr. 1).



Obr. 1: Multi-robotická aplikácia s rôznymi typmi robotov

2 Použité mobilné roboty

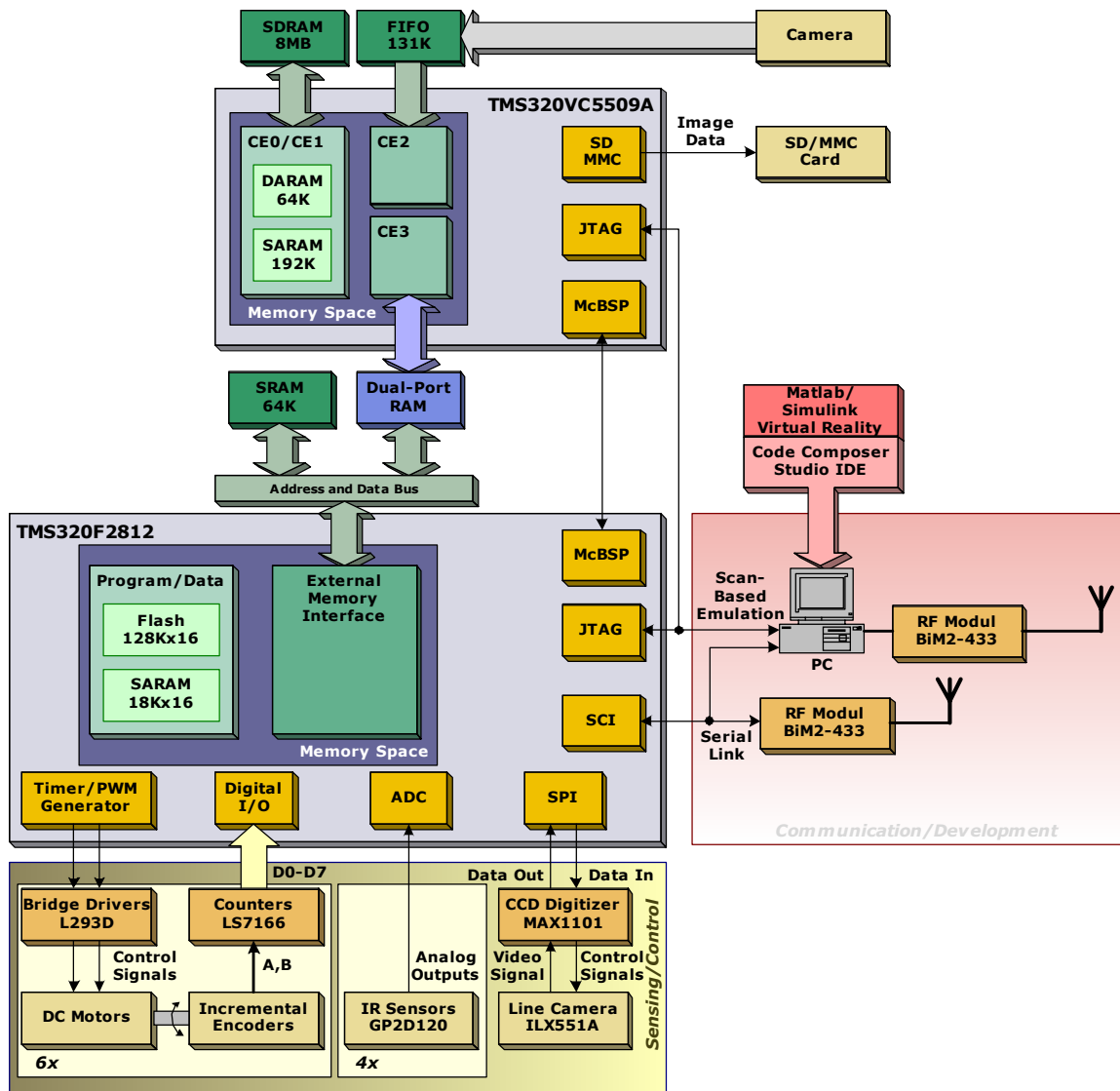
Na laboratórne experimenty bol vyvinutý mobilný robot F.A.A.K. a ako vzdušný robot, ktorého hlavnou úlohou je koordinácia pozemných robotov, bola použitá vzducholod', vybavená príslušnými senzormi a riadiacou elektronikou (Obr. 2).



Obr. 2: Mobilný robot F.A.A.K. (vľavo) a upravená autonómna vzducholod' (vpravo)

Pri rozhodovaní o riadiacej platforme pre uvedené roboty padla voľba na 32-bitové, fixed-point DSP procesory vývojovej rady C28xx firmy Texas Instruments, ktoré ponúkajú okrem výkonného výpočtového jadra aj množstvo on-chip periférií, potrebných na realizáciu digitálnych riadiacich systémov, ako sú časovače, AD-prevodník, digitálne vstupy/výstupy, generátory PWM signálov a rôzne komunikačné rozhrania. Nezanedbateľnú úlohu pri návrhu riadiacej elektroniky hrala aj jej celková hmotnosť a spotreba (predovšetkým v prípade vzducholode). Okrem toho disponujú obidva typy robotov vizuálnym systémom, pozostávajúcim z digitálnej kamery (v prípade potreby dvoch), ktorá sa môže pohybovať v dvoch osiach. Na spracovanie obrazu sú použité DSP procesory rady C55xx, taktiež od firmy Texas Instruments. Obidva typy procesorov navzájom komunikujú buď prostredníctvom zdieľanej pamäti (Dual-Port RAM), alebo pomocou sériového MsBSP portu.

Obidva typy robotov disponujú niekoľkými typmi interných/externých senzorov, ako sú inkrementálne snímače polohy jednotlivých motorov, snímače zrýchlenia (použité pri riadení pohybu pozemného robota ako aj na stabilizáciu vzducholode), infračervené snímače vzdialenosti a iné. Okrem toho sú vybavené rádiovými modulmi, ktoré umožňujú sériovú komunikáciu s robotmi a slúžia na ich diaľkové ovládanie a prenos nameraných dát a programov počas testovania aplikácií. Prepojenie jednotlivých subsystémov robota k použitým DSP procesorom v prípade mobilného robota F.A.A.K. je zrejmé z Obr. 3.



Obr. 3: Prepojenie subsystémov robota

Za účelom simulácie modelov uvedených robotov, ako aj návrhu a hardware-in-the-loop (HIL) testovania ich riadiacich algoritmov bolo v prostredí MATLAB/Simulink vyvinutých niekoľko nástrojov a simulačných knižníc, ktorých použitie bude v nasledujúcich častiach nášho príspevku demonštrované na príklade mobilného robota F.A.A.K. Jedná sa o nasledovné komponenty:

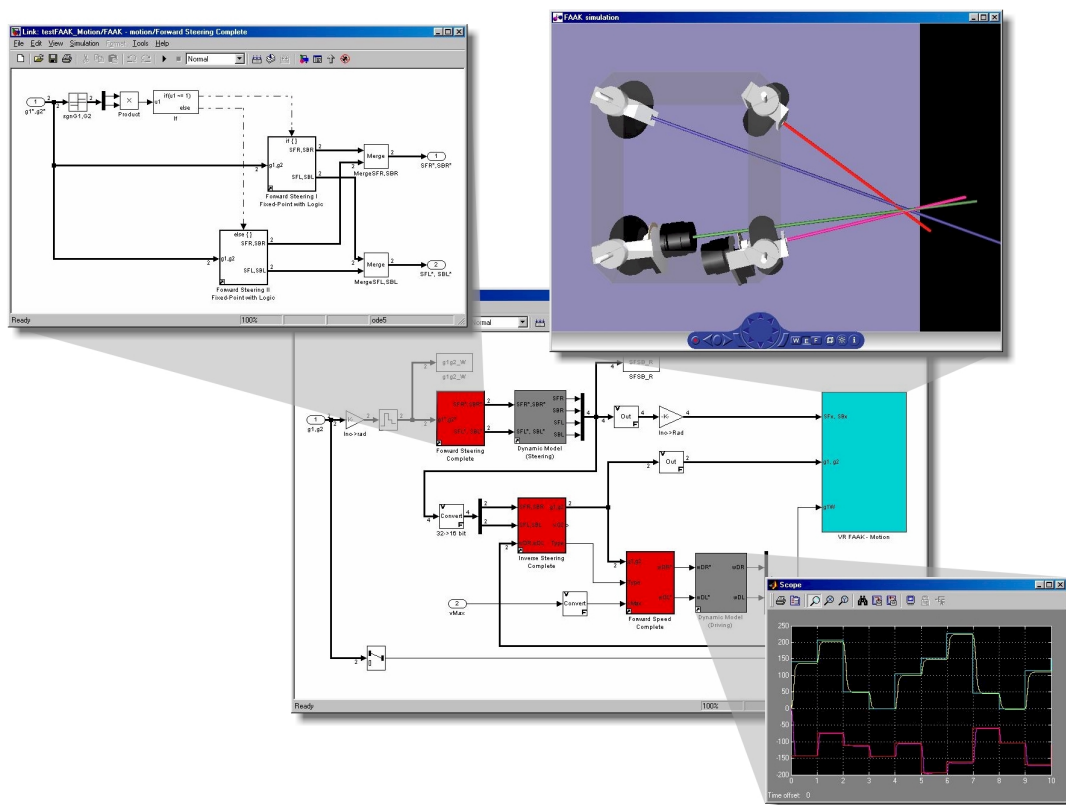
- **FaakLib** – knižnica kinematických/dynamických simulačných modelov robota a jeho elementárnych riadiacich algoritmov,
- **MapGen** – program na vytváranie modelov prostredia, ktoré je možné použiť na simuláciu interakcií robota s okolím a 3D vizualizáciu jeho pohybu pomocou Virtual Reality Toolbox-u,
- **FaakTarget** – RTW-target vyvinutý úpravou z C2000 Embedded Target-u špeciálne pre mobilný robot F.A.A.K., ktorý umožňuje automaticky generovať programový kód vo fixed-point formáte pre riadiaci DSP procesor robota,
- **FaakComm** – program na diaľkové riadenie robota, prenos dát z robota do MATLAB-u za účelom ich analýzy, ako aj prenos riadiacich algoritmov do riadiaceho procesora robota.

3 FaakLib

Keďže na testovanie navrhnutých riadiacich algoritmov robota je potrebné mať k dispozícii jeho simulačný model, bola vytvorená knižnica simulačných modelov pre Simulink **FaakLib**, ktorá obsahuje kinematické a dynamické modely robota s rôznym stupňom zložitosti. Tak je možné podľa typu testovaného algoritmu vykonávať buď komplexné simulácie robota so zohľadnením dynamiky

jeho jednotlivých osí a pohonov, alebo brať do úvahy len kinematické závislosti medzi jeho subsystémami. Okrem toho obsahuje knižnica modely senzorov umiestnených na robote a odladené algoritmy, použité na elementárne riadenie pohybu robota. Robot F.A.A.K. má totiž nekonvenčnú kinematickú štruktúru, keďže všetky jeho kolesá sú riadené, čo mu umožňuje vykonávať niektoré špeciálne pohyby, ako je napr. otáčanie sa na mieste, alebo dopredná jazda bez zmeny orientácie robota.

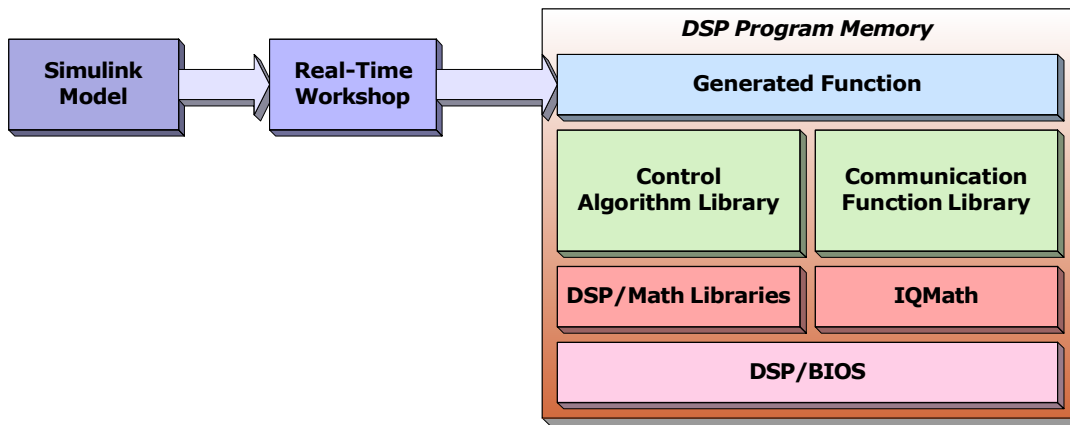
Pri návrhu riadiacich algoritmov robota sa ako veľkou výhodou ukázalo použitie blokov zo *Simulink Fixed-Point* knižnice, keďže ich bolo možné navrhnuť v rovnakom formáte s pevnou rádovou čiarkou, ako boli neskôr implementované na riadiacom DSP. Tým sa dali už počas simulácií odhaliť problémy, spôsobené zvolenou číselnou reprezentáciou veličín a optimalizovať jednotlivé kroky výpočtu tak, aby bola dosiahnutá požadovaná presnosť vykonávaných matematických operácií. Podobné testovanie na reálnom robote by naproti tomu bolo veľmi zdĺhavým procesom, pri ktorom by navyše mohlo dôjsť k poškodeniu robota v dôsledku skrytých výpočtových chýb. Takto odladené a optimalizované subsystémy na vykonávanie elementárnych riadiacich algoritmov je možné použiť aj pri automatickom generovaní kódu pre DSP. Príklad testovania algoritmu riadenia natočenia kolies robota spolu s vizualizáciou ich pohybu pomocou VRML modelu je znázornený na Obr. 5.



Obr. 4: Simulácia elementárnych riadiacich algoritmov robota s využitím blokov z knižnice *FaakLib*

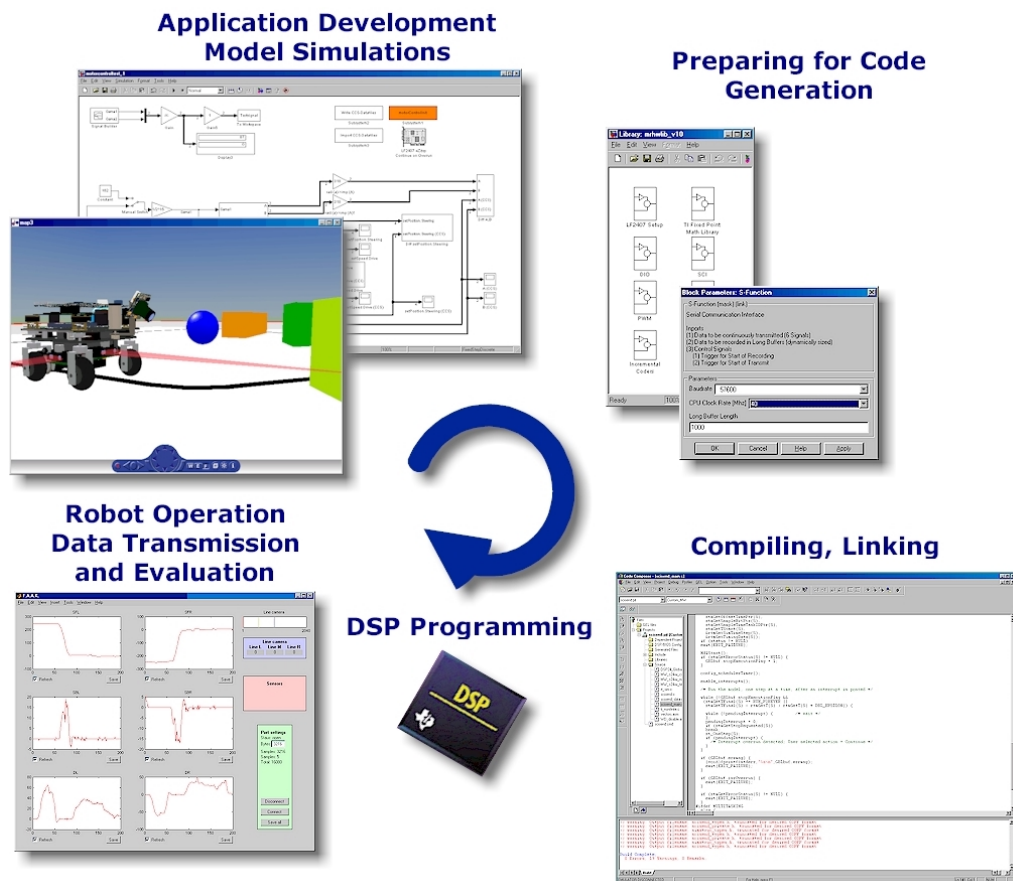
4 MapGen

Aby bolo možné simulovať pokročilejšie algoritmy riadenia mobilného robota, ako je napr. reaktívne riadenie, lokálna navigácia, plánovanie trajektórií atď., je potrebné do simulácií zakomponovať aj interakciu robota s okolím prostredníctvom jeho senzorického vybavenia. Ako bolo spomenuté, modely použitých typov senzorov boli vytvorené v knižnici *FaakLib*. Na uľahčenie návrhu prostredia robota sme vytvorili program *MapGen*. Pomocou tohto programu je možné vytvoriť okolie robota z rôznych geometrických útvarov a následne ho uložiť buď ako štruktúrovanú mapu do binárneho MAT-súboru, alebo vyexportovať ako 3D VRML model. Obidva súbory môžu byť následne použité v simuláciách a to v blokoch s modelmi senzorov (napr. v blokoch IR senzormi, merajúcimi vzdialenosť robota od prekážok) a na vizualizáciu pohybu robota v prostredí. Obidve možnosti sú znázornené na Obr. 5.



Obr. 6: Komponenty programu v DSP

Vytvorený kód je prenesený do pamäte robota pomocou sériového rozhrania, buď pomocou káblového alebo rádiového pripojenia, pričom počas prenosu je pozastavená funkcia na vykonávanie real-time riadenia. Po prenesení programového kódu je tento skopírovaný na miesto pôvodnej funkcie a robot môže začať s vykonávaním riadiaceho algoritmu. Počas behu programu ukladá robot dáta do pamäťového priestoru, rezervovaného na tieto účely, ktoré môžu byť po zastavení robota prenesené do monitorovacieho PC na vyhodnotenie. Proces vývoja algoritmov mobilného robota s využitím uvedených prostriedkov je znázornený na Obr. 7.



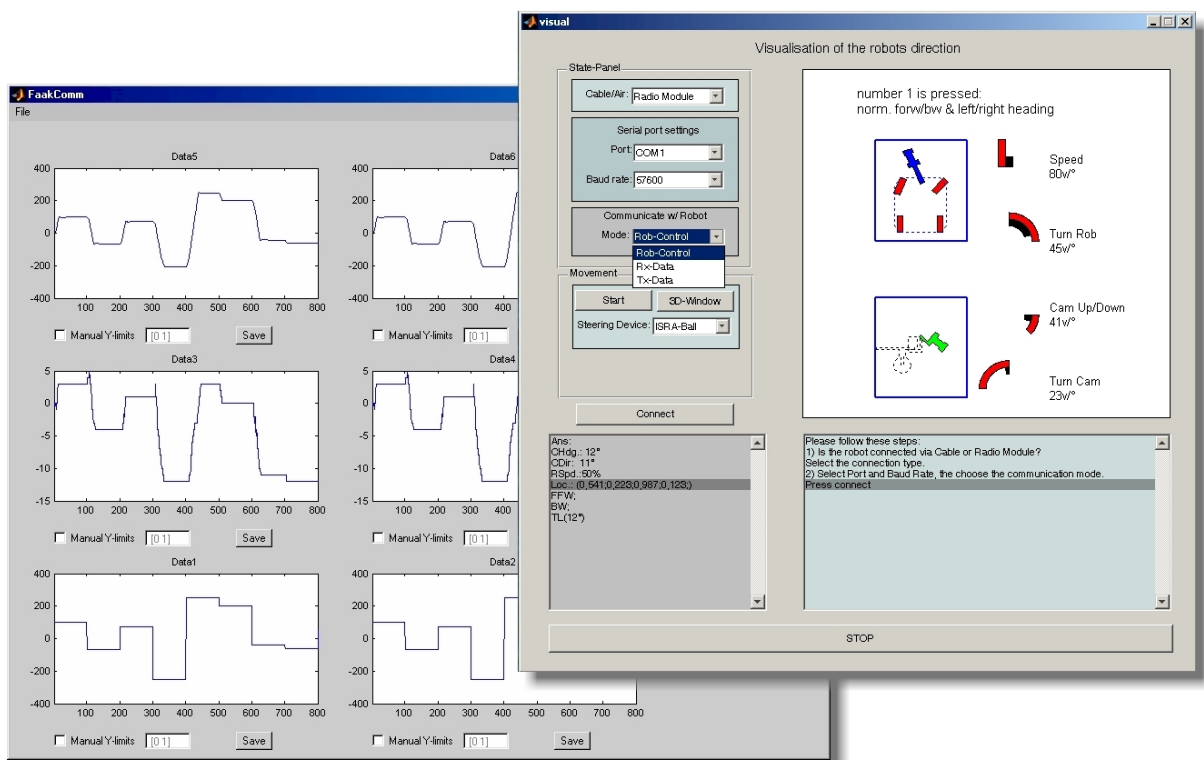
Obr. 7: Hardware-in-the-Loop návrh a testovanie algoritmov s využitím *FaakTarget*-u

6 FaakComm

Posledným predstavovaným komponentom, používaným pri HIL simuláciách robota, je program *FaakComm*. Tento slúži na sériovú komunikáciu s mobilným robotom, ktorá je využitá na tieto účely:

- diaľkové ovládanie robota, pričom riadiace príkazy sú zadávané prostredníctvom zariadenia „Teach Ball“ – jedná sa o zariadenie podobné joysticku so šiestimi stupňami voľnosti, ktoré komunikuje s PC pomocou sériového portu,
- vizualizáciu pohybu robota pomocou jeho VRML modelu, ktoré je možné použiť aj pri ovládaní robota (s využitím pohľadu na okolie z perspektívy robota),
- prenos experimentálnych dát z robota do PC za účelom ich analýzy, pričom je možné prenášať buď obmedzené množstvo dát v každej perióde vzorkovania, alebo väčšie objemy dát po zastavení robota, zaznamenané počas jeho predchádzajúcej jazdy,
- prenos skompilovaného riadiaceho algoritmu do pamäte robota.

Navrhnuté GUI (Obr. 8) k programu *FaakComm* umožňuje pohodlný výber komunikačných portov na komunikáciu s robotom a ovládacím zariadením, nastavenie spôsobu a rýchlosti komunikácie, formát prenášaných dát a iné parametre a zobrazuje aktuálne informácie o stave komunikácie a robota.



Obr. 8: Program *FaakComm* na komunikáciu s robotom

7 Záver

Naším prvým cieľom pri vývoji multi-robotickej aplikácie bol návrh integrovaného vývojového prostredia, ktoré by umožňovalo simulovať použité mobilné roboty a ich prostredie, navrhovať algoritmy na riadenie pohybu a spracovanie signálov, ako aj automaticky generovať kód pre DSP procesory, použité na robotoch a pomocou HIL simulácií ho analyzovať a ladiť. Ako najrýchlejším a zároveň najflexibilnejším riešením sa ukázala implementácia tohto prostredia na báze MATLAB/Simulink-u, vďaka ktorej prebiehajú jednotlivé kroky návrhu v jednom programovom prostredí a samotné vývojové prostredie môže byť podľa potreby ľahko rozšíriteľné o nové programové moduly alebo adaptované na nové mobilné roboty.

V ďalšom vývoji sa zameriavame na dokončenie vývoja simulačných modelov a riadenia robotickej vzducholode a simulácie kooperácie medzi robotmi, ako aj vizuálneho riadenia robotov s využitím spracovania obrazu na C55xx DSP.

Referencie

- [1] I. Masár, M. Gerke. *DSP-Based Control of Mobile Robots*. The European DSP Education and Research Symposium EDERS-2004, Birmingham, UK, 2004
- [2] *TMS320F281x Digital Signal Processors*, Texas Instruments, 2004
- [3] *TMS320 DSP/BIOS, User's Guide*, Texas Instruments 2004
- [4] *TMS320F28x Assembly Language Tools, User's Guide*, Texas Instruments 2001
- [5] *TMS320F28x Optimizing C/C++ Compiler, User's Guide*, Texas Instruments 2001
- [6] *Real-Time Workshop, User's Guide*, The Mathworks, 2004
- [7] *Embedded Target for the TI TMS320C2000TM DSP Platform, User's Guide*, The Mathworks, 2004
- [8] *Simulink Fixed Point, User's Guide*, The Mathworks, 2004