


Ovládání laboratorního modelu Mindstorms NXT (spike) pomocí PC

PC control of laboratory model Mindstorms NXT (spike)

Bc. Jakub Reich

Diplomová práce
2008

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2007/2008

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jakub REICH**

Studijní program: **N 3902 Inženýrská informatika**

Studijní obor: **Informační technologie**

Téma práce: **Ovládání laboratorního modelu Mindstorms NXT
(spike) pomocí PC**

Zásady pro vypracování:

1. Provedte literární studii softwarového a hardwarového vybavení robota NXT.
2. Provedte studii programovacího jazyka, možnosti komunikace atd. (Java, Labview, C++, DotNet a další).
3. Provedte návrh ovládacího programu (možno i webové aplikace) s ohledem i na jiné předměty vyučované na FAI.
4. Provedte návrhy laboratorních úloh a zpracujte pro ně dokumentaci.

Rozsah práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **PERDUE, David J. THE UNOFFICIAL LEGO MINDSTORMS NXT INVENTOR S GUIDE.** Megan Dunchak; Christina Samuell. 1st edition. San Francisco : No Starch Press, Inc, c2008. 296 s. ISBN 978-1-59327-154-1.
2. **ASTOLFO, Dave, FERRARI, Mario, FERRARI, Giulio. BUILDING ROBOTS WITH LEGO MINDSTORMS NXT.** Audrey Doyle. 1st edition. Burlington : Syngress Publishing, Inc, c2007. 447 s. ISBN 978-1-59749-152-5.
3. **SCHOLZ, Matthias Paul. Advanced NXT: The Da Vinci Inventions Book.** Jennifer Whipple. Apress, c2007. 369 s. ISBN 978-1-59059-843-6.
4. **GASPERI, Michael, HURBAIN, Philippe, HURBAIN, Isabelle. Extreme NXT : Extending the LEGO MINDSTORMS NXT to the Next Level.** Susannah Davidson Pfalzer. [s.l.] : Apress, c2007. 286 s. ISBN 978-1-59059-818-4.
5. **BOOGAARTS, Martijn, et al. THE LEGO MINDSTORMS NXT IDEA BOOK : design, invent, and built.** Nancy Sixsmith, Megan Dunchak. 1st edition. San Francisco : No Starch Press, Inc, c2007. 344 s. ISBN 978-1-59327-150-3.
6. **BAGNALL, Brian. Maximum LEGO NXT : Building Robots with Java Brains.** Edited by Sylvia Philipps. 1st edition. Canada : Variant Press, c2007. 505 s. ISBN 978-0-9738649-1-5.

Vedoucí diplomové práce:

Ing. Roman Šenkeřík

Ústav aplikované informatiky

Datum zadání diplomové práce:

20. února 2008

Termín odevzdání diplomové práce:

19. května 2008

Ve Zlíně dne 20. února 2008



prof. Ing. Vladimír Vašek, CSc.

děkan



doc. Ing. Ivan Zelinka, Ph.D.

ředitel ústavu

ABSTRAKT

Tato diplomová práce se zabývá analýzou hardwarových a softwarových vlastností robotické vývojové sady LEGO MINDSTORMS NXT: Cílem je vytvoření aplikace vhodné k ovládání laboratorního modelu „Spike“ pomocí PC a zprostředkování získaných znalostí ucelenou a přehlednou formou laboratorních úloh k dalšímu studiu, v návaznosti na předměty vyučované na Univerzitě Tomáše Bati ve Zlíně. Práce vychází z konceptu, vystavěném na využití toolboxu RWHT, vyvinutého univerzitou Rheinisch-Westfaelische Technische Hochschule v CÁCHách, pro matematický software MATLAB společnosti Mathworks.

Klíčová slova: Mindstorms NXT, robot, laboratorní úlohy, MATLAB, RWHT toolbox

ABSTRACT

This master thesis analyzes hardware and software features of the LEGO MINDSTORMS NXT robotics development kit. Main aim is to create the application suitable to control the laboratory model „Spike“ by a computer and mediation of an obtained knowledge in a compact and well-arranged form supposed to further studies, in relationship to subjects taught on Thomas Bata University in Zlin. The concept of this thesis is based on usage of the Mathworks MATLAB software with the RWHT toolbox, developed on Rheinisch-Westfaelische Technische Hochschule in Aachen.

Keywords: Mindstorms NXT, robot, laboratory tasks, MATLAB, RWHT toolbox

Rád bych na tomto místě poděkoval vedoucímu diplomové práce Ing. Romanu Šenkeříkovi za podněty, rady a odborné vedení mé diplomové práce. Dále bych chtěl poděkovat rodinným členům za projevenou morální a lidskou podporu, která dopomohla vzniku tohoto projektu.

Motto

“
Fantazie je důležitější než vědění.

”

ALBERT EINSTEIN (* 1879 – † 1955)

Souhlasím s tím, že s výsledky mé bakalářské práce může být naloženo podle uvážení vedoucího bakalářské práce a vedoucího katedry. V případě publikace budu uveden jako spoluautor.

Prohlašuji, že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve Zlíně, 19. 5. 2008

.....
Jakub Reich

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 NXT HARDWARE	10
1.1 VÝVOJ MINDSTORMS NXT.....	10
1.2 PROGRAMOVATELNÁ KOSTKA NXT	12
1.2.1 Výstupní a vstupní porty	13
1.2.2 LCD displej	16
1.2.3 I ² C.....	17
1.2.4 Rozhraní Bluetooth	19
1.3 SERVOMOTORY	19
1.4 STANDARDNÍ SENZORY NXT.....	22
1.4.1 Dotykový senzor.....	23
1.4.2 Světelný senzor	24
1.4.3 Zvukový senzor	25
1.4.4 Ultrazvukový senzor	27
1.5 SENZORY A HARDWARE EXTERNÍCH FIREM.....	28
2 NXT SOFTWARE	34
2.1 LABVIEW	35
2.2 NXT-G	38
2.2.1 RobotC	40
2.3 ALTERNATIVNÍ PROGRAMOVACÍ JAZYKY.....	41
II PRAKTICKÁ ČÁST	43
3 APLIKACE PRO ŘÍZENÍ ROBOTU SPIKE	44
3.1 MATLAB	45
3.2 RWTH TOOLBOX.....	46
3.3 VOLBA ADAPTÉRU BLUETOOTH.....	48
4 NÁVRH LABORATORNÍCH ÚLOH	49
4.1 ÚLOHA Č. 1 – OVLÁDÁNÍ SERVOMOTORŮ ROBOTU NXT	49
4.2 ÚLOHA Č. 2 – SNÍMÁNÍ HODNOT SENZORŮ ROBOTU NXT	62
4.3 ÚLOHA Č. 3 – ŘÍZENÍ LABORATORNÍHO MODELU SPIKE.....	66
ZÁVĚR	70
ZÁVĚR V ANGLIČTINĚ	72
SEZNAM POUŽITÉ LITERATURY	74
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	76
SEZNAM OBRÁZKŮ	77
SEZNAM TABULEK	79

SEZNAM PŘÍLOH.....80

ÚVOD

Robotika se řadí mezi prestižní a perspektivní vědní obory současnosti. Zabývá se technologií robotů, jejich studiem, návrhem, výrobou a uplatněním. Robotika patří mezi moderní multidisciplinární obory, zahrnující znalosti elektrotechniky, mechaniky, teorie řízení, měřicí techniky, umělé inteligence a celé řady dalších disciplin. Přestože roboty mohou mít různý vzhled a schopnosti, obvykle se robotem rozumí zařízení (autonomní integrovaný systém) vykonávající činnost pohybového nebo manipulačního charakteru, v cílené interakci s reálným okolím, na základě určité formy samostatného řízení. Robot zpracovává sensoricky informace svých vnitřních stavů a údaje o okolním prostředí a na základě tvorby modelu a řízení je schopen toto prostředí aktivně ovlivňovat.

V roce 2006 uvedla společnost LEGO na trh novou generaci svých modulárních robotických systémů, stavebnici MINDSTORMS NXT, která spojuje konstrukční přednosti léty prověřených umělohmotných kostek s inteligentním programovatelným jádrem 32 bitového mikropočítače a jeho vstupně výstupních periferií, zahrnujících čtyři typy standardních senzorů a tři servomotory. Stavebnice nabízí jednoduchý a přesto účelný přístup k návrhu, konstrukci a programování laboratorních modelů robotů, který našel uplatnění ve vzdělávacím procesu po celém světě.

Tato diplomová práce se zabývá rozborem základních hardwarových a softwarových charakteristik sady MINDSTORMS NXT a přibližuje přehlednou a srozumitelnou formou laboratorních úloh využití nástrojové sady RWTH, toolboxu matematického softwaru Matlab, pro jejich řízení. Nástrojová sada byla vyvinuta na univerzitě Rheinisch-Westfaelische Technische Hochschule v Cáchách, kde slouží jako laboratorní nástroj pro výuku v inženýrském oboru elektrotechnika. Cílem je seznámit potenciálního studenta s funkcionalitou toolboxu, vlastnostmi využití bohatého programového zázemí matematického softwaru Matlab a metodikou řízení laboratorních modelů pomocí PC obecně.

Práce svým obsahem navazuje na předměty Matlab, Algoritmizace úloh a Mikropočítače, na Fakultě aplikované informatiky UTB ve Zlíně. Jednotlivé oddíly diplomové práce jsou koncipovány tak, aby mohly být případně použity jako manuál laboratorních úloh.

I. TEORETICKÁ ČÁST

1 NXT HARDWARE

1.1 Vývoj MINDSTORMS NXT

MINDSTORMS NXT vypustila společnost LEGO na trh na sklonku roku 2006. Skutečný počátek historie této sady lze ovšem datovat do roku 1998, kdy byla představena první stavebnice MINDSTORMS: Robotic Invention System (RIS) s programovatelnou kostkou Robotic Control Explorer (RCX), která představovala úspěšné sloučení evoluce pěti dekad modulárních konstrukčních technik stavebnic LEGO, s léty rozvíjenými vzdělávacími metodami v oboru počítačových věd na univerzitě Massachusetts Institute of Technology (MIT). [4]



Obr. 1 Inteligentní kostka RCX

Své stavebnice produkuje LEGO již od konce 40. let minulého století. Počínaje obyčejnými důvěrně známými do sebe zapadajícími plastovými stavebními kostkami, až po dnešní vysoce technické části, zahrnující ozubená kola, hřídele, senzory, napájecí zdroje, motory či pneumatické díly. Ačkoli se styl modelování neustále vyvíjí, nové součástky jsou navrženy zpravidla ke zpětné kompatibilitě, využitelnosti a plné slučitelnosti se starými. Tento charakteristický rys a rozsáhlý inventář existujících částí jsou jedním z důvodů, proč je LEGO atraktivní volbou pro konstrukci modelů robotů a dalších rozmanitých zařízení.

MIT se zabývá výzkumem využití informační techniky ve vzdělávání od druhé poloviny sedmdesátých let 20. století. Původní název stavebnice MINDSTORMS byl přitom převzat z titulku knihy jednoho z tamních významných vědeckých pracovníků, Seymoura Paperta. Koncept práce vycházel nejen ze zprostředkování a vyhodnocení testů studentů, ale také užití počítače studenty, jako nástroje k řešení logických problémů a vlastního výzkumu. S technologickým postupem a miniaturizací tištěných obvodů se staly počítače ideálním prostředkem k vytvoření nespočetného množství prototypů programovatelných kostek.

RIS znamenal revoluční posun ve vývoji. Široké veřejnosti se poprvé dostal do rukou jednoduchý, přehledný a přitom komplexní systém, který umožňoval snadno vymyslet i sdílet jednotlivé návrhy. Každý mohl bez obtíží přesně realizovat svou představu, dokonce bez hlubších znalostí tradičních oborů, jako jsou stavební konstrukce, kovodělné práce, elektronika nebo programování. RIS se také stal výhodnou volbou vzdělávacích institucí, které potřebovaly odolnou a mnohonásobně použitelnou platformu pro inženýrské laboratorní projekty.

Kostka RCX zapouzdřovala osmibitový mikropočítač se třemi vstupy a třemi výstupy, infračerveným komunikačním rozhraním, malým reproduktorem a digitálním LCD displejem o čtyřech zobrazovaných jednotkách. Programování pomocí osobního počítače, probíhalo na bázi intuitivního grafického vývojového prostředí. Pro své přílišné zjednodušení byl však jazyk až na nejprimitivnější modely prakticky nepoužitelný.

Komunita lidí pohybujících se v okolí produktu započala horečně s jeho zpětným zkoumáním a vynaložila značné úsilí na jeho zdokonalení. Brzy se na Internetu objevily alternativní programovací jazyky a byly publikovány návrhy designu nových domácích senzorů. Tlak ze strany zákazníků postupně donutil firmu LEGO k uveřejnění Softwarové vývojové soupravy a další dokumentace, čímž učinil z RCX uživatelsky více otevřenou platformu.

Vydáním odlehčené verze Robotic Discovery System (RDS) s nepřepisovatelnou pamětí ROM se společnost dopustila strategického omylu. Výrobek byl komunitou ignorován a odsouzen. Zatímco elektronika ve světě se každý měsíc posunula vpřed, LEGO i nadále namísto zdokonalování svých produktů vyrábělo slabší a limitovanější verze. Vývoj MINDSTORMS na osm následujících let virtuálně zamrzl.

1.2 Programovatelná kostka NXT

Jádro stavebnice MINDSTORMS NXT tvoří inteligentní programovatelná kostka NXT, mikropočítač, zapouzdřující hlavní procesor, koprocessor, paměti, obvody vstupně výstupních periférií a napájecí zdroj (šest baterií typu AA). Lze hovořit o řídicím (kognitivním) systému sestavovaného robotu. Řídicím systémem se rozumí rozhraní shromažďující vnitřní a vnější sensorická data, na jejichž základě se provádí rozhodování o posloupnosti a jednotlivých operací vykonávaného programu. Mimo to, umožňují informace zpracované řídicím systémem robotu zkonstruovat model okolního prostředí, podle kterého se v návaznosti na danou úlohu rozhoduje o změně polohy, popřípadě akci koncového efektoru. Efektorem je míněno zařízení, jehož stav lze ovládat změnou napětí iniciovanou mikrokontrolerem (nejběžněji pohon motoru).



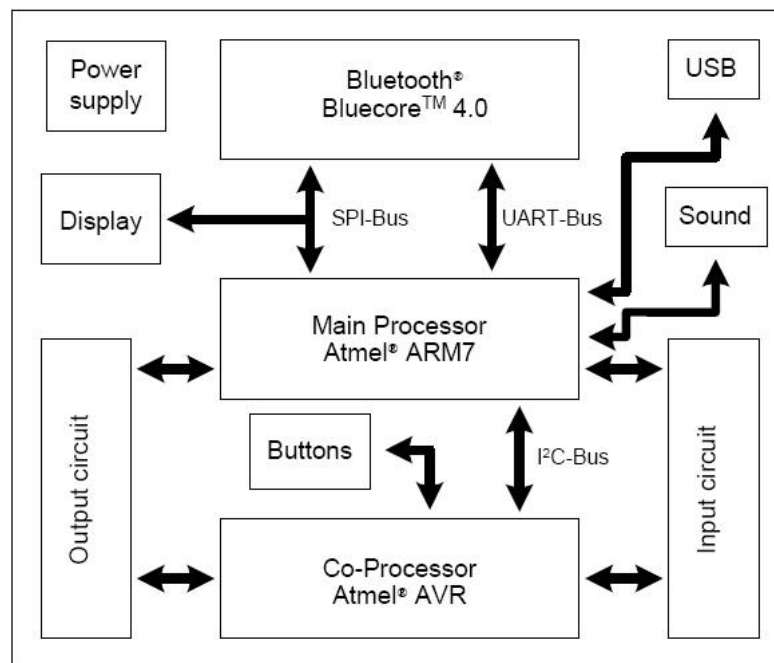
Obr. 2 Inteligentní kostka NXT

Hlavní, dvaatřicetibitový mikroprocesor Atmel AMR7, obsahuje 256 KB přepisovatelné paměti FLASH, 64 KB operační paměti RAM a pracuje na frekvenci 48 MHz. Osmibitový koprocessor AVR, taktovaný na frekvenci 8 MHz, obsahuje 4 KB paměti FLASH a 512 B operační paměti.

Na vrchní straně přístroje jsou orientovány tři výstupní porty A, B, C, sloužící k ovládání připojených motorů a port vysokorychlostní sériové sběrnice USB 2.0 (s přenosovou rychlostí 12,1 Mb/s), přes niž lze z nadřazeného počítače do paměti mikropočítače zaznamenat řídicí program nebo další systémové soubory. Na spodní straně se nalézají čtyři výstupní porty (označené čísly 1 až 4), vyhrazené pro sběr informací z připojených senzorů. Pro úspěšnou práci se systémem potřebují uživatelé nutně přístup k jeho nastavení a řízení. Uživatelské rozhraní (HMI), umístěné na čelním panelu programovatelné kostky NXT, je tvořeno čtyřmi ovládacími prvky (tlačítky) a monochromatickým černobílým LCD

displejem o rozlišení 100 x 64 pixelu. Displej umožňuje nejen navigaci položkami menu implementovaného firmwaru, ale také zobrazení libovolné informace z uživatelského programu. Ke zvukové signalizaci, případně přehrávání složitějších zvukových záznamů, jako je hudba, slova, fráze, číslice, atp. byl zahrnut do těla kostky reproduktor o vzorkovací frekvenci 2-16 KHz s osmibitovým kódováním. Zabudovaný čip standardu Bluetooth 2.0 EDR dovoluje bezdrátovou komunikaci s okolními zařízeními (např. mobilní telefon, osobní počítač, PDA). [1], [2], [4]

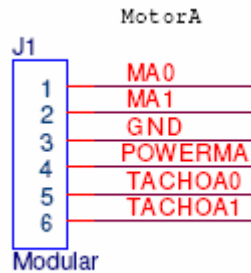
Následující schéma na obrázku Obr. 3, uvádí detailněji propojení jednotlivých komponent mikropočítače a funkcí, které jsou inteligentní kostkou řízeny.



Obr. 3 Schéma zapojení komponent kostky NXT

1.2.1 Výstupní a vstupní porty

Jak bylo zmíněno v předcházející kapitole, obsahuje kostka NXT tři výstupní porty, určené k řízení připojených efektorů. Efektory jsou přivedeny šestižilovou průmyslovou telefonní sběrnicí, zakončenou standardizovanými konektory RJ12. Šestižilové vedení je koncipováno tak, aby výstupní zařízení mohlo odesílat informace zpět do mikropočítače, bez nutnosti využití vstupního portu (vyhrazeno pro vstupní signál enkodéru motoru). Schematické zapojení portu A uvádí obrázek Obr. 4. Zbývající porty B a C jsou identické. [9]



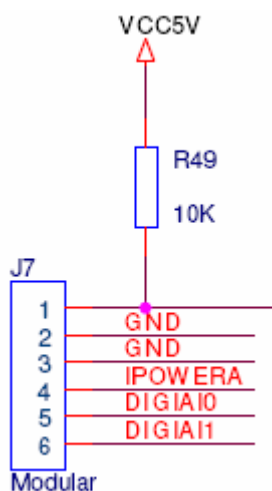
Obr. 4 Výstupní port

Popis pinů výstupního portu A:

- MA0 a MA1, výstupní signály pro řízení akčních členů. Tyto signály jsou řízeny interním řadičem motoru, který může dodat nepřetržitý proud 700 mA a špičkový proud přibližně 1 A. Výstupní modulační signál PWM ovlivňuje výkon procentuálním rozložením dodávaného proudu. Řadič motoru je vybaven vestavěnou ochranou proti přehřátí, která v případě nutnosti automaticky odpojí příslušný výstup.
- GND, zemnicí signál vyhrazený pro výstupní porty
- POWERMA, výstupní napětí 4,3 V je vnitřně přivedeno na všechny piny výstupních a vstupních portů. Pokud překročí proud, odebíraný zároveň na všech portech, maximální dovolenou hranici 180 mA, automaticky poklesne bez předchozího varování hodnota tohoto výstupního napětí. Přivedení napětí na zemnicí pin vede k restartování kostky.
- TACHOA0 a TACHOA1 jsou vstupy Schmittova klopného obvodu, umístěného mezi portem a vstupními piny mikroprocesoru ARM7. Schmittův klopný obvod je speciální komparátor s hysterezí, jehož výstup závisí nejen na hodnotě vstupu, ale i na jeho původním stavu. Signály umožňují detekci fázového rozdílu mezi nimi. Standardní firmware využívá těchto signálů k určení rychlosti a směru otáčení motoru.

Čtyři vstupní porty poskytují zázemí pro sensorická měření fyzikálních vlastností okolního reálného prostředí. Šestižilová uživatelská sběrnice umožňuje na jednom portu snímat jak analogové, tak digitální signály. Kterýkoli ze vstupních portů lze použít k připojení libovolného senzoru.

Vstupní port číslo 4 obsahuje navíc rozšíření o implementaci komunikačního čipu rozhraní RS485, který se nachází za běžným obvodem. Tento port umožňuje inteligentní kostce NXT vysokorychlostní obousměrnou komunikaci ve vícebodové síti na velké vzdálenosti. Díky práci na bázi otevřeného komunikačního protokolu provozní sítě P-NET, dovoluje připojení externích zařízení, jako jsou počítače, inteligentní senzory, akční členy nebo paměťové nadstavby, užitím běžně dostupného dvoužilového vedení. Sériově zapojené komponenty přitom komunikují nezávisle na sobě. Koncept přenosu se nejvíce podobá sběrnici USB zkombinované se strukturou počítačových sítí 10Base2 (tenký Ethernet). [9]



Obr. 5 Vstupní port

Popis vstupního portu 1:

- Pin 1 (ANA), vstupní analogový signál přivedený na vstup desetibitovému A/D převodníku koprocesoru AVR. Současně také funguje jako generátor elektrické energie pro aktivní senzory podporované systémem RIS. Všechny vstupní analogové signály portů jsou vzorkovány stejnou frekvencí 333 Hz. Aktivní analogové senzory vyžadují před zahájením měření žádané veličiny přivedení napájecího napětí po dobu 3 ms
- IPOWERA, slouží obdobně jako POWERMA k vyvedení výstupní napětí 4,3 V a má shodné vlastnosti.
- DIGIAIO a DIDIAI1, digitální vstupně/výstupní piny kompatibilní s komunikačním rozhraním I²C o rychlosti 9600 bit/s. NXT pracuje ve vztahu k vnějším zařízením pouze jako master a vyžaduje začlenění zdvihadího rezistoru na jejich komunikačních vstupech. Detaily sběrnice I²C rozvádí kapitola.

1.2.2 LCD displej

Každý pixel LCD displeje je složen z molekul tekutých krystalů, uložených mezi dvěma transparentními elektrodami a dvěma polarizačními filtry, přičemž osy polarizace jsou na sebe navzájem kolmé. Bez krystalů mezi filtry by světlo procházející jedním filtrem bylo blokováno filtrem druhým. Molekuly tekutých krystalů se nacházejí bez elektrického proudu v neuspořádaném stavu. S přivedení elektrického proudu jsou molekuly srovnány s mikroskopickými s kolmými drážkami na elektrodách. Jednotlivé molekuly se srovnají do spirálové struktury, přičemž světelné záření procházející filtrem je při průchodu tekutým krystalem rotováno, což mu umožňuje projít i druhým filtrem. Polovinu záření absorbuje první polarizační filtr. Mimo toho je ale celá sestava průhledná. [10]

V okamžiku přivedení elektrického proudu na elektrody jsou molekuly tekutých krystalů taženy rovnoběžně s elektrickým polem, což snižuje rotaci vstupujícího světla. Jestliže nejsou tekuté krystaly vůbec stočené, procházející světlo se polarizuje kolmo k druhému filtru, je zcela blokováno a pixel se jeví jako nerozsvícený. Řízením stočení krystalů v pixelu lze kontrolovat kvantitu procházejícího světla, a tedy celkovou svítivost pixelu.

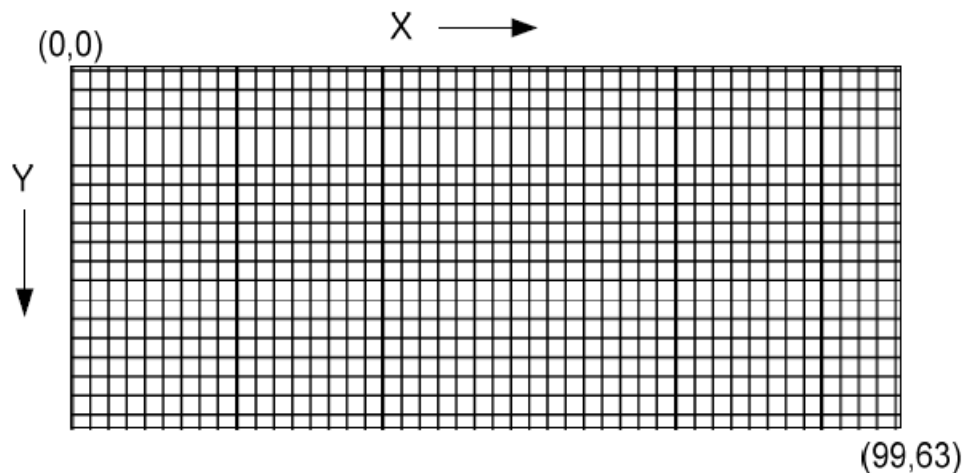
Je obvyklé srovnat polarizační filtry tak, aby bez přívodu elektrické energie byly pixely průhledné, a při průchodu elektrického proudu se stanou neprůhlednými. Pro dosažení speciálních efektů se však někdy využívá i opačného uspořádání.

Elektrické pole potřebné pro rychlé srovnání molekul tekutých krystalů však způsobuje jejich „vystrčení“, což vede k poškození displeje. Tento problém se kompenzuje použitím střídavého proudu.

Pro snížení ekonomických nákladů jsou v elektronice LCD displeje často multiplexovány. V multiplexovaném displeji se elektrody na jedné straně displeje seskupují vertikálně a každá skupina má svůj zdroj napětí. Na druhé straně se elektrody seskupují horizontálně, přičemž každá tato skupina má svůj spotřebič napětí. Matice je konstruována tak, aby každý pixel měl unikátní kombinaci zdroje a spotřebiče. Elektronika pak řídí spínání zdrojů a spotřebičů.

Mezi podstatné faktory pro hodnocení LCD displeje patří rozlišení, rozměry zobrazované plochy, doba odezvy, typ mřížky (pasivní nebo aktivní), pozorovací úhel, podpora barev, jas, kontrast, poměr stran a vstupní porty.

Černobílý bodový displej LCD s rozlišením 100x64 pixelů, pracující v pozitivním reflexním módu, byl do kostky NXT vestavěn pro zlepšení uživatelského rozhraní. Zorná plocha displeje je 26 x 40.6 mm. Zobrazování na displeji řídí interní řadiče UltraChip1601, který komunikuje po vnitřní sběrnici SPI s mikrokontrolerem AVM. Rychlost sběrnice je 2 MHz. Doba odezvy displeje je 17 ms. Data alokovaná paměti mikropočítače se řadí do dvourozměrného pole. Pozorovací úhel displeje je 180 stupňů.



Obr. 6 Souřadnicový systém LCD displeje

1.2.3 I²C

Rozhraním I²C (Inter-Integrated Circuit) se rozumí průmyslový standard multi-masterové sériové sběrnice, vyvinutý společností Philips Semiconductors počátkem 80. let minulého století. Používá se pro připojení nízkorychlostních periferních zařízení, vyžadujících jednoduchou digitální komunikaci. Dvou vodičové vedení, zahrnující hodinový kanál SCL (Synchronous Clock) a datový kanál SDA (Synchronous Data) umožňuje připojit až 128 různých komponent. Připojené stanice se dělí na řídicí, tzv. master, které kontrolují počátek a konec přenosu a generují taktovací signál a řízené (slave). [12]

Z elektrického hlediska jsou oba kanály zapojeny jako otevřený kolektor. Maximální délku vodičů omezuje nejvyšší přípustná kapacita 400 pF. Připojením vodičů ke kladnému napětí přes zdvihací odpor se zajistí jejich vysoká napěťová úroveň v klidovém stavu. Během vysílání jsou jednotlivé datové bity na SDA přenášeny tak, ke změně logické úrovně na SDA dochází pouze, je-li SCL v úrovni L. Toto pravidlo se poruší jen při vysílání START a STOP bitu, rozlišujících počátek a konec komunikace.

Maximální frekvence signálu SCL stanoví verze protokolu I²C na 100 kHz nebo 400 kHz. Pro oba mezní kmitočty je určen minimální povolený čas setrvání SCL v úrovni L a H. Při komunikaci a přenosu dat synchronizují jednotlivé stanice své generátory taktovací frekvence tak, aby časový úsek přechodu do úrovně H na SCL byl odměřován vnitřními časovači každé stanice až od okamžiku, kdy SCL opravdu dosáhne úrovně H (protože SCL je obvod s otevřeným kolektorem, může setrvávat v úrovni L i v okamžiku kdy se daná stanice pokouší přejít na úroveň H). Obdobně je od spádové hrany odměřována doba trvání úrovně L na SCL. Tento princip dovoluje některé ze stanic zpomalit přenos. Pomalá stanice může podržet po určitou dobu signál SCL v úrovni L a tím zabránit vysílající stanici ve vyslání dalšího bitu. Sběrnice I²C neumožňuje plně duplexní přenos, v každém okamžiku vysílá pouze jedno zařízení. Všechna zařízení připojená na sběrnici mají nastavenou individuální adresu o délce 7 nebo 10 bitů a zabudovaný mechanismus komunikace pomocí I²C sběrnice. Po zachycení impulsu START srovnávají všechny obvody svou adresu s adresou, která je vysílána na sběrnici. Zjistí-li některý z obvodů shodu, že vysílání je určeno právě jemu, potvrdí přijetí adresy bitem ACK. Potom přijímá nebo vysílá další data. Několik adres je na I²C vyhrazeno pro speciální účely.

Každý vysílaný byte či adresa je následována vysláním jednoho bitu ACK. Vysílající stanice jej přenáší na úrovní H. Přijímající stanice potvrzuje přijetí připojením SDA do úrovně L po dobu jeho vysílání. Jestliže vysílající stanice nedostane potvrzení příjmu, ukončí vysílání podmínkou STOP.

Pro řízení komunikace se využívá metoda s detekcí kolize. Každá stanice zahájí vysílání právě tehdy, je-li předtím sběrnice v klidovém stavu. Během vysílání musí být neustále srovnávány odeslané bity s reálným stavem SDA. Pokud je zaznamenán rozdíl mezi očekávaným a skutečným stavem linky SDA, značí to indikaci kolize mezi několika stanicemi, pokoušejícími se o komunikaci. Vzhledem k povaze sběrnice může dojít k situaci, kdy určitá stanice vysílá úroveň H, zatímco jiná stanice vysílá úroveň L. Stanice, která na lince indikuje úroveň L zatímco sama vysílá H, je nucena vysílání ihned ukončit. K řízení komunikace obvykle dochází při vyslání několika počátečních bitů, kdy se odesílá adresa přijímací stanice. Pokud by se dvě stanice pokusily ve shodný okamžik o zápis do stejného obvodu, nastane kolize až při přenosu vlastních zapisovaných dat. V mezním případě, kdy zapisuje několik stanic současně data na shodnou adresu, nemusí být kolize detekována vůbec.

1.2.4 Rozhraní Bluetooth

Bluetooth je bezdrátová komunikační technologie, definovaná standardem IEEE 802.15.1, sloužící k přenosu dat mezi dvěma či více elektronickými zařízeními, na krátké vzdálenosti při nízkém rozsahu frekvenčního pásma. Existuje v několika vývojových verzích, přičemž dříve nejčastěji implementovanou verzí 1.2 nahrazuje nyní novější specifikace Bluetooth 2.0 EDR (Enhanced Data-Rate), disponující modulační technikou $\pi/4$ -DQPSK, zvyšující datovou propustnost na trojnásobnou hodnotu oproti starší verzi 1.2 (2,1 Mbit/s). Tímto se dosahuje dlouhodobější výdrže baterii, protože samotné navázání spojení i přenos probíhají v mnohem kratší době, než u starších verzí Bluetooth. [13]

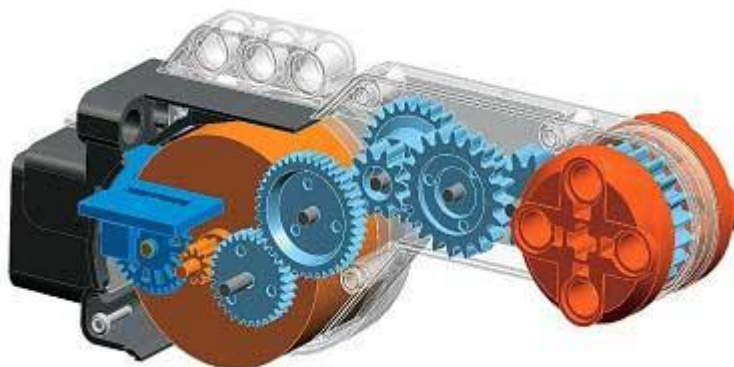
Bluetooth pracuje, obdobně jako Wi-Fi, v ISM pásmu 2,4 GHz. Využívá přenosové metody FHSS, při níž se během jedné sekundy provede 1600 skokových změn (přeladění) mezi 79 frekvencemi v rozestupu 1 MHz. Zmíněný mechanismus zvyšuje odolnost spojení proti šumovému rušení na shodném kmitočtu. Je definováno několik výkonových úrovní (2,5 mW, 10 mW, 100 mW), jimž odpovídá komunikace do vzdálenosti přibližně 10 až 100 m. Udávané hodnoty ovšem platí jen ve volném prostoru. Pokud jsou mezi komunikujícími zařízeními překážky (typicky například zdi), dosah rychle klesá. Obvykle však nedochází ke skokové ztrátě spojení, ale postupně narůstá množství chybně interpretovaných paketů.

Řídící mikropočítač NXT podporuje bezdrátovou komunikaci na bázi Bluetooth pomocí zabudovaného čipu CSR BlueCore 4, verze 2. Byla jim nahrazena starší přenosová metoda programovatelné kostky RCX, pracující na principu infračerveného signálu. Kostku lze připojit souběžně až ke třem dalším zařízením, ale v jednom okamžiku komunikuje právě s jedním. Přenosovou činnost mezi dvěma zařízeními popisuje komunikační protokol, vycházející z profilu SPP (Serial Port Profile), vytvářející spojení pomocí emulované sériové linky RFCOMM. Jedná se o souhrn instrukcí v oblasti služeb poskytovaných aplikacím, definování funkcí a postupů nutných pro interoperabilitu mezi zařízeními Bluetooth. Pro redukování spotřeby pracuje čip ve výkonové třídě Class 2, s dosahem přibližně 10 m (v otevřeném prostoru až 15 m). [1], [2], [4], [5]

1.3 Servomotory

Základním a typickým představitelem efektoru je elektromotor. Elektromotorem se míní elektrický točivý stroj, přeměňující elektrickou energii na mechanickou práci. Nejčastěji

bývá v těle robotu aplikován jako součást lokomočního (pohybového) nebo manipulačního ústrojí. Lokomočním ústrojím se nazývá technický prostředek (podvozek), jehož specifickou vlastností je pohyb. Patří k charakteristickým rysům mobilních robotů s rozmanitými typy podvozků (kolové, pásové, kráčející, atd.). Mobilita označuje stupeň pohybové volnosti, který se vyskytuje u robotů s podvozkem, či jiným mechanismem, dovolujícím pohyb.



Obr. 7 Vnitřní převodování motoru NXT

Většina elektromotorů pracuje na elektromagnetickém principu, vycházejícím z Lorentzova zákona síly, vzájemném působení dvou polí, vytvářených průchodem elektrického proudu obvodem. Existují i motory stavějící na rozdílných elektromechanických jevech, jako jsou elektrostatické síly, piezoelektrický efekt či tepelné účinky průchodu elektrického proudu.

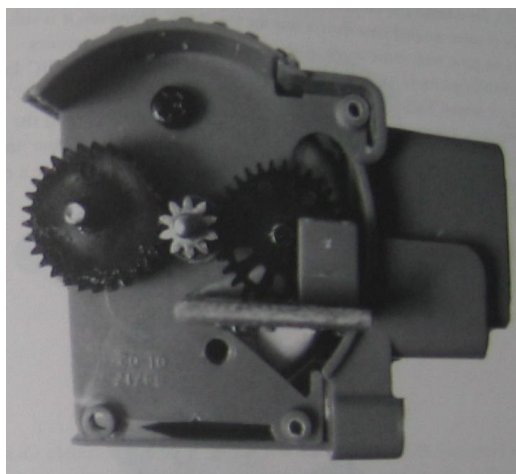
Servomotorem neboli servem, se míní motor pro pohony (především elektrické), u nichž lze na rozdíl od běžného motoru přesně nastavit polohu natočení osy. Mimo robotiku se aplikují například pro posuvy CNC strojů, nastavení čtecích hlaviček pevných disků nebo k ovládání rádiově řízených modelů. Elektrické servomotory bývají kontrolovány prakticky bez výjimky tranzistorovými měniči s pulzní šířkovou modulací (PWM).

Při pulzní šířkové modulaci, neboli PWM (Pulse Width Modulation) dochází ke změně šířky pulzu nosného signálu (signálu s konstantní frekvencí), přičemž rychlost otáčení elektromotoru je přímo úměrná šířce pulzu. Zařízení provádějící PWM modulaci, se nazývá pulzní šířkový modulátor. Principem je komparace - superpozice řídicího, modulačního signálu se signálem tvaru pily o vyšším konstantním kmitočtu splňující Shannon-Kotělnikovu podmínku. Výsledné impulzy za komparátorem jsou šířkově modulovány a jejich plocha odpovídá energii modulačního signálu

Poloha hřídele servomotoru se zjišťuje elektricky, pomocí fotoelektrického snímače (enkodéru) nebo rozkladače (selsynu). U zařízení vyžadujících nižší přesnost je možné aplikovat levnější optické snímání kódového kotoučku či proužku. Signál snímače polohy se přivádí zpětnou vazbou na regulátor, jenž komparuje skutečnou polohu motoru s žádanou hodnotou. Na základě rozdílu žádané a skutečné polohy regulátor řídí měnič a tak nastavuje motor na žádanou polohu.

Rotační senzor kostky RCX byl známý svou špatnou spolehlivostí při velmi nízkých a velmi vysokých otáčkách. Stabilitu snímání senzoru se snažili vývojáři robotických systémů zajistit optimalizací zdrojového kódu programu. Teprve integrací enkodéru přímo do těla motoru v sadě NXT vzrostla spolehlivost a funkcionalita celého komponentu, jež dovoluje vývoj sofistikovanějších mechanismů. Interaktivní servomotor umožňuje přesné měření rychlostí a vzdáleností v různých datových formátech, zahrnujících stupně, otáčky a čas. [1], [2], [5]

Princip enkodéru servomotoru NXT se podobá mechanismu starší kuličkové myši. Soustavu tvoří černé kolečko s dvanácti dírami, schopné pomocí optoelektronického snímače detekovat 24 stavových přechodů zapnuto/vypnuto na každou otáčku. Světelný paprsek generovaný senzorem dopadá skrz děrování na fotocitlivou buňku, kde jsou impulsy snímány a převedeny na informaci o rychlosti otáčení. Enkodér zajišťuje nejen detekci pohybu a jeho rychlosti, ale slouží také jako synchronizační jednotka při zdvojeném zapojení motorů.



Obr. 8 Útroby enkodéru motoru NXT

1.4 Standardní senzory NXT

Měření je proces, pro jehož udržení je zapotřebí určitý tok energie nebo látkový tok. Je to nejdůležitější, nejpracnější a nejobsáhlejší část experimentální činnosti vedoucí k určení hodnoty žádané veličiny. Měřicí proces obvykle prostřednictvím čidla tuto energii více či méně ovlivňuje – spotřebovává. Čidlo (senzor, receptor) pak danou energii transformuje na výslednou informaci o stavu nebo průběhu procesu sledovaného objektu. Nositelem této informace je opět určitá forma energie nebo akumulace látky generované měřicím čidlem. Není-li však tato forma energie vhodná pro další zpracování informace používají se převodníky pro transformaci této energie na energii vhodnou pro další zpracování. Měření je realizováno technickými měřicími prostředky, které společně tvoří buď měřicí kanál, nebo celý měřicí systém. [8]

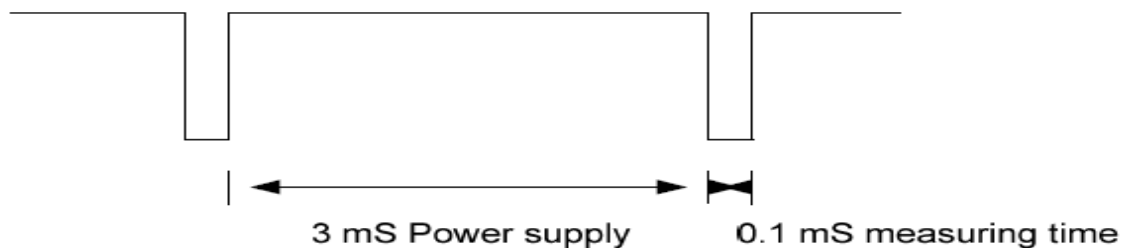
Vlastnosti technických prostředků měření jsou charakterizovány statickými a dynamickými parametry a parametry spolehlivosti. [7] Statické parametry popisují provozně technické vlastnosti přístrojů a jejich jednotlivých prvků v ustáleném stavu. Při sledování časových průběhů proměnných veličin od počátku změny do ustálení odezvy je daný průběh procesu měření charakterizován dynamickými parametry. Příčinou přechodových dějů jsou hmotné, tepelné, elektrické a magnetické kapacity, setrvačnosti, tlumení hmot měřicích prvků, impedance, rychlosti změn fyzikálních a chemických parametrů měřených medií atp. Dynamické parametry jsou definovány časovou konstantou nebo přechodovou funkcí, váhovou funkcí, frekvenční charakteristikou, frekvenční funkcí, dynamickou chybou. Spolehlivost je definována jako schopnost zařízení plnit funkci podle daných požadavků po definovaný časový úsek.

Senzorem se rozumí zařízení sloužící k měření hodnoty fyzikální veličiny a její převod na unifikovaný signál (nejčastěji elektrický), vhodný k dalšímu zpracování. Rozlišují se senzory interní zpětné vazby, monitorující vnitřní stav robotu a externí zpětné vazby, které dovolují robotu okolní prostředí a interagovat s vnějším světem. Informace snímané senzory zlepšují použitelnost a univerzálnost robotu, jeho schopnost vypořádat se změnami prostředí, v němž se pohybuje, určení polohy a orientace v prostoru, detekování překážek a vyhýbání se kolizím, detekování žádaných či chybových stavů, kontrole výsledků akce robotu, atd. Slouží k vytváření a udržování aktuálního modelu robotu a modelu okolního světa.

Z hlediska napájení se rozlišují senzory NXT:

- **Aktivní**, které ke své činnosti vyžadují kromě měřicího signálu také zdroj vnějšího napájení.
- **Pasivní**, kterými prochází pouze měřicí signál.
- **Digitální**, využívající komunikační sběrnici I²C.

Pro zachování zpětné kompatibility se staršími senzory systému RIS (vyrobenými pro inteligentní kostku RCX) jsou dodrženy u stávajícího generátoru kostky NXT shodné parametry napájení a měřicího intervalu. Generátor poskytuje proudový výstup o maximální hodnotě 18 mA a řídí energii dodávanou aktivním sensorům. Nejdříve se vyše 3 ms napájecí impuls a během následující 1 ms je snímána analogová hodnota měřené veličiny. [9]



Obr. 9 Časový průběh snímání na senzorech

1.4.1 Dotykový senzor

Dotykový senzor je typem spínače s jedním stabilním stavem. Rozlišují se hodnoty zapnuto (1), kdy stlačením spínač uzavře vodivě elektrický obvod a umožní průchod detekovatelného signálu a vypnuto (0), kdy silou návratového pružného elementu je elektrický obvod rozpojen. Případně lze detekovat také přechod mezi těmito dvěma stavy. Pro svou jednoduchost a intuitivnost nalézá dotykový senzor uplatnění v celé řadě aplikací, například při manuálním ovládání strojů, zjišťování přítomnosti výrobku, u dorazu pojízdných zařízení, při detekci kolize s překážkou, atp. [1], [2], [4], [5]



Obr. 10 Dotykový senzor

1.4.2 Světelný senzor

Světelný senzor je konstruován jako soustava fotoelektricky citlivého prvku, fototranzistoru a zdroje světelného záření, elektroluminiscenční LED diody. Fototranzistorem se míní bipolární polovodičová křemíková součástka, jejíž emitorový přechod umožňuje přístup světla. Zapojuje se společným emitorem, přičemž vnější zdroj se připojuje mezi kolektor a emitor tak, aby kolektorový přechod byl polarizován závěrně. Báze zpravidla nebývá vyvedena. Funkcionalita spočívá v otevírání emitorového přechodu dopadajícím zářením. Množství uvolněných nosičů se zvyšuje úměrně s intenzitou záření a je zesilováno jako bázový proud bipolárního tranzistoru. Vlivem tohoto zesilovacího účinku mívají fototranzistory vyšší citlivost na záření než fotodiody. Neosvětleným fototranzistorem prochází kolektorový proud, zvaný proud za temna I_0 , určený zbytkovým proudem tranzistoru ICE_0 . Voltampérové charakteristiky mají tvar výstupních charakteristik bipolárního tranzistoru, parametrem je zde namísto proudu báze osvětlení E .

Světlo-vyzařující diodou LED se rozumí elektronická polovodičová součástka vybavená P-N přechodem. Prochází-li přechodem elektrický proud v propustném směru, přechod emituje úzkospektrální nekoherentní světlo. Ve speciálních případech může vyzařovat i jiné druhy záření. Tento jev je označován jako elektroluminiscence, přeměna elektrické energie ve světelnou.

Světelný senzor se užívá k měření hodnoty světlosti snímané plochy nebo intenzity dopadajícího světelného záření a dovoluje také nepřímou určit, dle odstupňování šedi odraženého světla, barvu snímané plochy. Informace získané senzorem prezentuje kostka jako procentuální údaj od 0 do 100%, kde hodnota 0% představuje nejmenší množství

dopadajícího světelného záření a 100% množství maximální. Zdroj světelného záření lze kdykoli programově vypnout, takže senzor pak slouží pouze ke snímání intenzity okolního světla.

Během snímání odraženého světla je zapotřebí počítat s interferenčními vlivy všech vnějších zdrojů světelného záření, jež při měření působí. Množství plochou odraženého světla ovlivňuje řada faktorů – převážně její barva, struktura a druh materiálu, či vzdálenost od zdroje záření. Černé objekty odrážejí méně záření než světlé, přičemž záleží i na matnosti či lesklosti daného povrchu. Od vzdálenějších objektů se k senzoru vrací méně záření než od bližších. Tyto faktory jsou vzájemně závislé, což znamená, že určit přesně návratový údaj snímaný senzorem, lze pouze při udržení konstantních, dobře známých počátečních podmínek. [1], [2], [4], [5]



Obr. 11 Světelný senzor

1.4.3 Zvukový senzor

Zvukem se rozumí vibrace (mechanické vlnění), šířené pružnými, pevnými, kapalnými či plynnými látkami nebo plasmou, které lze zachytit sluchovým orgánem. Ve vakuu se zvuk nešíří. Materiál, který je nosičem zvukového signálu, se nazývá medium. Plynnými, kapalnými a plasmatickými látkami, se zvukový signál šíří jako podélné vlnění. Pevnými látkami se zvuk šíří jako vlnění podélné nebo příčné. Při podélném (longitudálním) vlnění je amplituda kmitů rovnoběžná se směrem šíření vlny. Při příčném (transversálním) vlnění je amplituda vlnění kolmá ke směru šíření vlny. Zdrojem zvuku může být každé chvějící se těleso. O vlnění v okolí zdroje zvuku nerozhoduje pouze jeho chvění, ale také okolnost, zda je předmět dobrým nebo špatným zářičem zvuku. Tato vlastnost závisí zejména na jeho

geometrickém tvaru. Zvukový signál je charakterizován všeobecnými vlastnostmi vlnění, mezi něž patří frekvence, délka vlny, perioda, amplituda, intenzita, rychlost a směr.

Lidské vnímání zvuku je limitováno v intervalu frekvencí 20 Hz až 20 kHz, přičemž v průběhu života dochází k jeho zhoršení vlivem opotřebení sluchových receptorů. Zvuk s frekvencí nižší než 20 Hz se nazývá infrazvuk. Zvuk s frekvencí vyšší než 20 kHz se označuje jako ultrazvuk.

Senzorem určeným pro snímání zvukového signálu je zvukový senzor, označovaný také jako mikrofon. Princip snímání vychází z přeměny mechanických kmitů akustického vlnění na elektrický signál. Lze měřit intenzitu signálu v rozsahu vnímatelném lidským uchem, kdy jednotkou je 1 dBA (adjusted decibel) nebo obecnější signály, zahrnující i frekvence mimo tento rozsah, kdy jednotkou je 1 dB (decibel). Decibel představuje nejen jednotku pro měření hladiny intenzity zvuku, ve skutečnosti se jedná o obecné měřítko podílu dvou hodnot, které se používá v mnoha dalších oborech. Jedná se o fyzikálně bezrozměrnou míru, obdobně jako například procento. Avšak na rozdíl od něj je decibel jednotka s logaritmickou mírou, jejíž definice souvisí s objevením Fechner-Weberova zákona (lidský sluch vnímá zvukové podněty logaritmicky k jejich intenzitě).



Obr. 12 Zvukový senzor

Přibližný vztah mezi intenzitou zvuku, decibely a procentuální hodnotou, zobrazovanou kostkou NXT uvádí následující tabulka.

dB	Vnímání člověkem	Hodnota senzoru
10	Práh slyšitelnosti	0%
20	Hluboké ticho, bezvětří, akustické studio	
30		4 - 5 %
40	Tlumený hovor, šum v bytě, tikot budíku	
50	Klid, tichá pracovna, obracení stránek novin	
60	Běžný hovor	10 - 30 %
70	Mírný hluk, hlučná ulice, běžný poslech televize	
80	Velmi silná reprodukováná hudba, vysavač v blízkosti	
90	Silný hluk, jedoucí vlak	
100	Sbíječka, přádelna, maximální hluk motoru	30 - 100 %
110	Velmi silný hluk, živá rocková hudba, kovárna kotlů	
120	Startující proudové letadlo	
130	Práh bolestivosti	
140	Akustické trauma, 10 m od startujícího proudového letadla	

Tab. 1 Vztah vnímání intenzity zvuku člověkem

1.4.4 Ultrazvukový senzor

Jedním z nejvýznamnějších snímacích zařízení sady NXT MINDSTORMS je ultrazvukový senzor, často označovaný jako sonar. Sonar (z anglického SOund Navigation And Ranging - zvuková navigace a zaměřování) je elektronické zařízení, vycházející z principu radaru, určené k zaměření a zjištění vzdálenosti objektů pomocí ultrazvukového vlnění. Proces snímání, kdy se zvuk vysílaný senzorem odrazí od předmětu zpět do místa vysílání, kde je zachycen a z celkového času, jenž uplynul od okamžiku vyslání zvukové vlny do okamžiku zpětného příjmu odražené vlny (ozvěny neboli echa) je vypočítána vzdálenost, se nazývá echolokace. Tento princip využívají mimo jiné některé specializované lékařské elektronické přístroje (sonografy) či bývá aplikován při měření hloubky moře.

Ultrazvukový senzor měří vzdálenost v centimetrech a palcích. Je schopen snímat vzdálenost v rozsahu 0 až 2,5 metrů s přesností +/- 3 cm. [1], [2], [4], [5]



Obr. 13 Ultrazvukový senzor

1.5 Senzory a hardware externích firem

Díky zavedení standardizovaného digitálního komunikačního rozhraní I²C, lze využít vstupní porty programovatelné kostky NXT k připojení velkého množství různých typů senzorů. Z dlouhodobé, rentabilní a perspektivní spolupráce společnosti LEGO s externími firmami vzešla celá řada kvalitních výrobků. Mezi nejvýznamnější zástupce mezi vývojáři a prodejci externích senzorů pro sadu NXT patří společnosti Vernier, HiTechnic a Mindsensors.com.

Společnost Vernier vyvinula senzorový adaptér pro sadu NXT, vhodný k připojení asi třiceti typů senzorů, které sama vyrábí. V široké škále se objevuje například akcelerometr, průtokoměry, světelné senzory, barometry, kolorimetry, tlakové senzory, termoelektrické články, senzory pro měření pH, O₂ turbidity, magnetického pole, atd. Podrobnější informace o jednotlivých senzorech a jejich kompletní výčet lze získat na www stránkách výrobce.

Společnost HiTechnic se zabývá výrobou senzorů a rozbočovačů kompatibilních s robotickými systémy LEGO MINDSTORMS. Senzory jsou dodávány bez konektorových vodičů, které jsou součástí běžné stavební sady. Senzory jsou podporovány softwarem LEGO MINDSTORMS Education NXT verze 1.1 a firmwarem verze 1.05. Pro používání je třeba doinstalovat NXT G-bloky, zdarma dostupné na www stránkách výrobce. Mezi podporované senzory se řadí například gyroskopy, kompas, IR senzor nebo

Společnost Mindsensors.com produkuje senzory, multiplexory a další hardware připojitelný k NXT, s podporou vývojových nástrojů NXT-G, RobotC, LeJOS a NXC (viz

kapitola 2. NXT SOFTWARE). Pro získání přehledu o možnostech rozšiřujících senzorů, rozebírá následující výčet stručně některé z vlastností senzorů Mindsensors.com:

- **Kamerový subsystém**, poskytuje zpracování obrazu v reálném čase, využitelné pro sledování vodící linie nebo až osmi uživatelsky definovaných rozdílně barevných objektů s obnovovací frekvencí 30 obrázků za sekundu, nastavitelným zaostřením čočky a výstupem statistických funkcí těchto sledování. Lze jej připojit nejen přes standardní vstupní port ke kostce NXT, ale také přes rozhraní USB přímo k osobnímu počítači.



Obr. 14 Webová kamera

- **RCX senzorový multiplexor pro NXT**, slouží k připojení starších senzorů, vyvinutých pro původní robotickou sadu RCX. RXMux dovoluje připojit až čtyři různé druhy senzorů, zahrnující dotykový, světelný, rotační a teplotní senzor.



Obr. 15 Senzorový multiplexor RCX

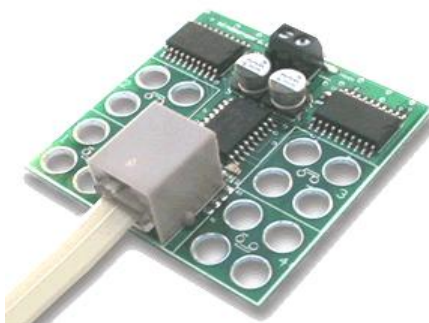
- **Vysoce senzitivní akcelerační senzor**, je třísměrový digitální akcelerometr se čtyřmi programově nastavitelnými stupni citlivosti. Lze jej použít nejen pro měření statického a dynamického zrychlení ve více směrech, ale také jako senzor naklonění nebo ke sledování rychlostního profilu. Pro komunikaci používá kompatibilní I²C

protokol. Detekuje zrychlení již od hodnoty 10 mG v závislosti na nastavení citlivosti pomocí programu. Rozlišení náklonu ve všech osách je 2,5 stupně. Podporované senzitivity jsou 2.5 G, 3.3 G, 6.7 G, 10.0 G. Senzor podporuje automaticky detekovatelnou paralelní architekturu, což znamená, že lze připojit dva senzory pracující nezávisle na sobě do jediného portu.



Obr. 16 Akcelerační senzor

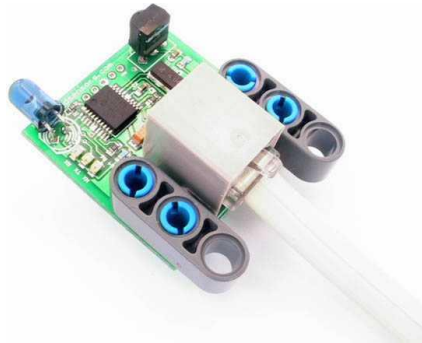
- **Multiplexor NXT motorů**, je jednoduchý a flexibilní multiplexor, speciálně navržený jako řadič pro připojení středně výkonných motorů RCX (a několika dalších typů motorů ze stavebních sad LEGO) ke kostce NXT. Multiplexor se připojuje přímo k digitálnímu portu, přičemž energie pro jeho napájení je dodávána z externího zdroje. Je navržen jako H-přemostění motorové jednotky pro napětí 35 V a proud 0,6 A. Lze ovládat až čtyři motory nezávisle na sobě v dopředném, zpětném, brzdícím a plovoucím režimu, při možnosti nastavení 256 rychlostí pro každý z nich. Podporuje čtení a zapisování všech nastavení parametrů motoru.



Obr. 17 Multiplexor NXT motorů

- **Komunikační rozhraní RCX – NXT**, je zařízení vyhrazené pro datové propojení programovatelných kostek RCX a NXT. Pomocí NXT lze udělovat povely nebo snímat data ze samostatného RCX zařízení nebo části složitějšího robotického

systemu řízeného kostkou RCX. Zařízení podporuje obousměrnou komunikaci pomocí infračerveného rozhraní.



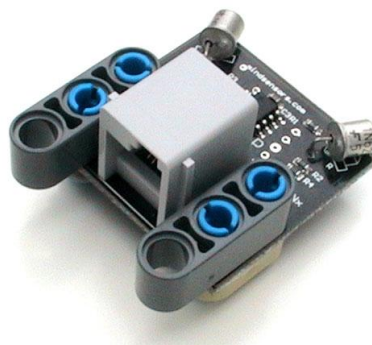
Obr. 18 Rozhraní RCX - NXT

- **Sony Playstation 2 rozhraní**, je rozhraní sloužící k připojení bezdrátového ovladače dreamGEAR MAGNAFORCE 2.4GHZ RF pro Sony Playstation 2. Ovladač obsahuje dva analogové joysticky a 16 tlačítek funkčních tlačítek, jejichž stisknutí lze zachytit programovatelnou kostkou. Pokročilý vícesměrový D-pad a analogové joysticky poskytují ideální manuální kontrolu a ovládání robotu.



Obr. 19 Rozhraní PS2

- **Duální IR detektor překážek**, je senzor určený k odhalení překážek v cestě robotu. Detektor je navržen tak, že tato pomyslná tykadla, mohou detekovat objekty v rozmezí od 6 "do 18" (15 cm až 45 cm). Toto čidlo je schopné zjistit překážky na přední-levé straně, přední-pravé straně nebo přímo před sebou.



Obr. 20 Detektor překážek

- **Magnetický kompas**, je digitální senzor dovolující detekovat odchylku od severního magnetického pólu Země. Lze jej využít například ke sledování trasy udržující určitý kurz. Obvod dvouosého ortogonálního magnetického kompasu vyvinula firma Honeywell. Senzor umožňuje vícenásobné čtení s výstupem celých nebo reálných čísel. Rozlišení odchylky je 0,1 stupně při programování v jazyce RobotC či NBC a 1,44 stupně při práci s jazykem NXT-G.



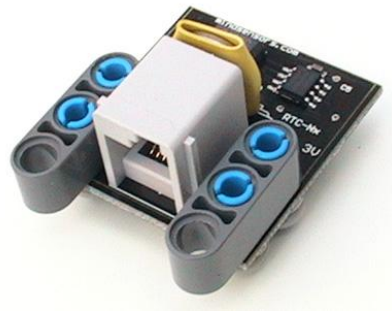
Obr. 21 Magnetický kompas

- **Pneumatický tlakový senzor**, se využívá při konstruování robotických systémů, obsahujících pneumatické díly. Lze jím měřit tlak produkovaný pneumatickým systémem v rozsahu od 0 do 35 PSI nebo od 0 do 240 kg/cm².



Obr. 22 Pneumatický senzor

- **Obvod reálného času**, využívá k běhu energie dodávané ze standardního portu a jako zálohu baterii se životností až deset let. Dovoluje počítat sekundy, minuty, hodiny, dny v týdnu a měsíce, měsíce, roky a přestupné roky s platnou kompenzací do roku 2100. Je ideální jako zázemí pro aplikace s ukládáním dat.



Obr. 23 Obvod reálného času

- **Vysoce precizní IR senzory, pro krátké, střední a dlouhé vzdálenosti, jsou digitální senzory využívající měření vzdálenosti objektů pomocí infračerveného záření. Vysoké rozlišení umožňuje odečítat hodnoty s přesností milimetru. Intervaly se dle použitého typu čidla pohybují v rozmezí 4 až 30 cm, 10 až 80 cm a 20 až 150 cm.**



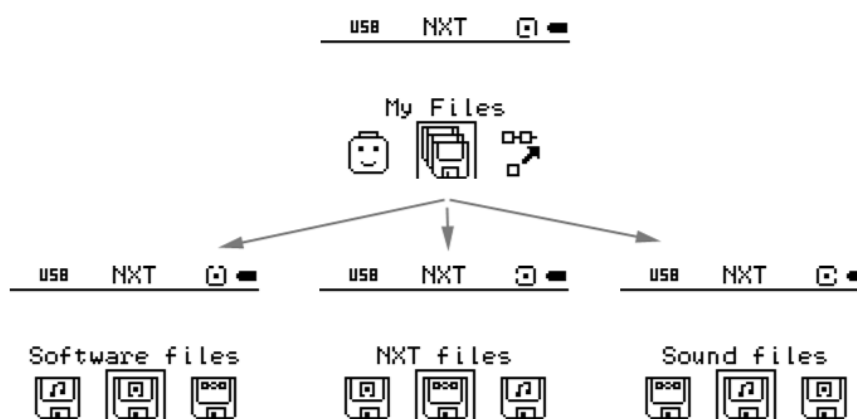
Obr. 24 Infračervený senzor

2 NXT SOFTWARE

Obdobně jako složitější osobní počítače, využívá i inteligentní kostka NXT vlastní operační systém. Operačním systémem se myslí programová sada sloužící především k tomu, aby byly aplikacím transparentním způsobem zpřístupněny prostředky počítače (resp. mikropočítače), aby vytvářel potřebné operační prostředí pro chod aplikací. Systémové prostředky zahrnují nejen obsluhu hardwaru (senzory, motory, tlačítka, LCD displej, časovače, paměti, atd.), ale také strojový čas a přístup ke komunikačním tokům, či prostředky nehmotné povahy (souborový systém, menu, grafika, ...). Operační systém dodávaný výrobcem v zabudované flash paměti je označován jako firmware. [1], [2], [4]

Souborový systém NXT dovoluje uchovávat až 64 souborů. Souborový systém označuje způsob organizace informací (dat ve formě souborů) tak, aby bylo možné je snadno vyhledat a přistupovat k nim. Firmware umožňuje uživateli vytvářet, mazat, přejmenovávat a modifikovat soubory a jejich obsah. Názvy uložených souborů umožňují maximální délku 15 znaků, s tříznakovou extenzí oddělenou od jména souboru tečkou (obdobně jako v operačním systému MS DOS). Extenze všech souborů začíná písmenem r (např., spustitelné soubory mají extenzi .rx). Tabulka ukazuje přiřazení extenzí k jednotlivým typům souborů.

Uživatelské rozhraní kostky NXT zobrazuje tyto soubory ve formátu strukturovaného menu. Ukázkou hierarchie menu a jeho podmenu uvádí následující obrázek. Zobrazovací utilitu NeXTScreen vyvinul John Hansen.



Obr. 25 Hierarchie menu firmwaru NXT

Datový prostor pro ukládání programů je značně limitován velikostí flash paměti inteligentní kostky, což omezuje jejich rozsáhlost a funkcionalitu. Částečnou kompenzaci

tohoto omezujícího faktoru lze dosáhnout odstraněním některých nepodstatných systémových souborů, například demo programů, zvukových souborů, či souborů grafického rozhraní nebo optimalizací zdrojového kódu programu s využitím opakujících se sekvencí a funkcí.

Zdrojový kód vytvořený pomocí programovacích prostředků na PC je předkompilován do kompaktního formátu, vhodného ke spuštění a vykonání jádrem mikropočítače. Tento kompaktní programový formát se nazývá bytecode. Označení vyplývá z anglického termínu byte, jednotky množství dat v informatice. Takové množství informace může reprezentovat například celé číslo od 0 do 255 nebo jeden znak. Jeden byte je obvykle nejmenší objem dat, se kterým dokáže mikropočítač (resp. procesor) přímo pracovat. Posloupnost bytů pak tvoří sérii instrukcí popisující žádané chování programu.

Bytecode ovšem není přímo spustitelným kódem. Operační systém zahrnuje aplikaci tzv. interpretu, která za běhu (v reálném čase) překládá jednotlivé části bytcodeu do strojově vykonatelných instrukcí. Interpretované jazyky jsou běžné – mezi nejznámější zástupce se řadí platforma Java. Java používá překlad do Java bytcodeu, který je přeměněn na strojově orientovaný kód pomocí interpretu pracujícího v reálném čase, kterému se říká Java Virtual Machine (JVM).

Společnost LEGO zvolila pro svůj výrobek jako primární programovací prostředí graficky orientovaný jazyk, vycházející ze softwaru LabVIEW. Nicméně časem byla vytvořena celá řada více či méně komplexních programovacích jazyků, stavějících na konvenčním, lexikálně orientovaném programování (jazyky na bázi C, C++, Java, atp.).

2.1 LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) je platforma a vývojové prostředí vizuálního programovacího jazyku firmy National Instruments. Grafický jazyk nazývaný „G“, byl původně vyvinut pro počítače Apple Macintosh v roce 1986. LabVIEW se běžně používá pro sběr dat, ovládání přístrojů a průmyslovou automatizaci. Pracuje na různých systémových platformách, zahrnujících Microsoft Windows, různé varianty systémů UNIX, Linux a Mac OS. [11]

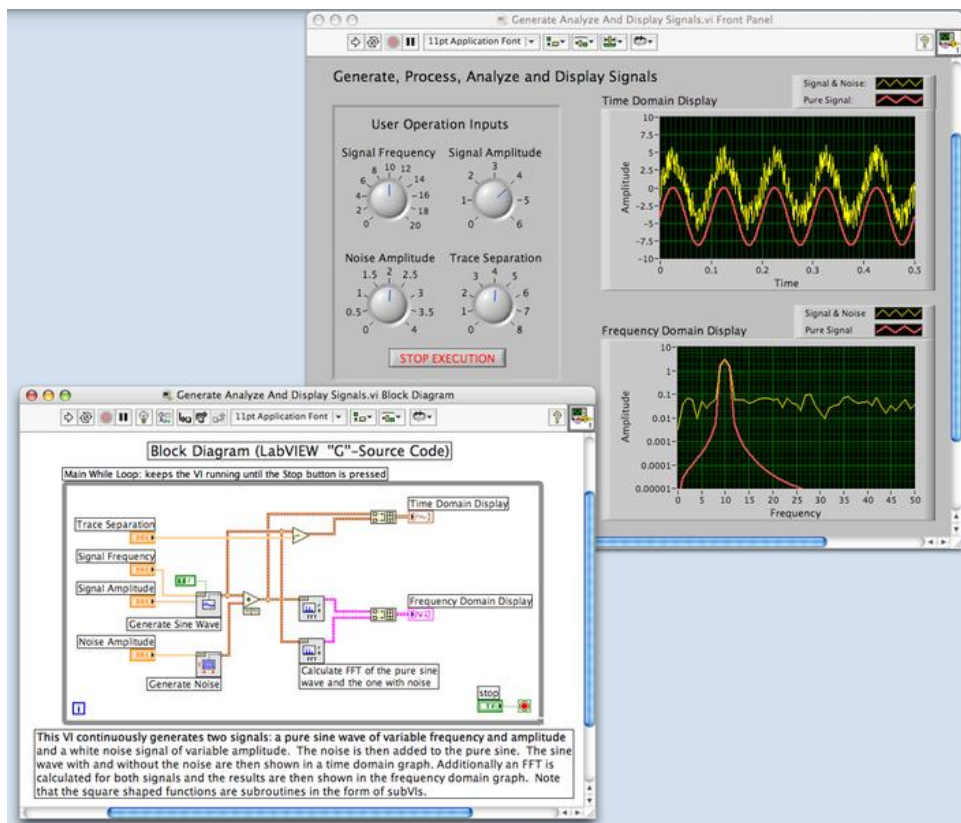
Programovací jazyk G patří do skupiny dataflow jazyků. Chování programu je podmíněno strukturou grafických blokových diagramů (zdrojového kódu LV), jejichž funkční uzly

spojuje programátor pomocí tažení čar. Tyto čáry představují proměnné a každý uzel může být vykonán v okamžiku příchodu dat na jeho vstup. Vzhledem k tomu, že se data mohou objevit na vstupech několika uzlů současně, dovoluje jazyk G paralelní zpracování. Mnohoprocesový a mnohovláknový hardware pak automaticky využívá vestavěný plánovač úloh, který přiřadí jednotlivým funkčním uzlům vlákna operačního systému, připravená k provedení.

Často dochází k omylu, že dataflow architektura je náchylná k uvážnutí. Datový tok však jednoznačně určuje pořadí vykonávaných kroků. Proto definuje grafická syntaxe chování programu stejně dobře jako kterýkoli konvenční, kódově orientovaný jazyk. Kromě toho nevyžaduje LabVIEW deklarace proměnných. Typ proměnné (spojnice mezi uzly) je stanoven daty procházejícími mezi jednotlivými uzly. Tato vlastnost se nazývá polymorfismus.

LabVIEW zahrnuje do vývojového procesu tvorbu uživatelského rozhraní (tzv. předních panelů). Programy a podprogramy se nazývají virtuální instrumenty (VI). Každý virtuální instrument má tři základní složky: blokové schéma, přední panel a konektorové okno. Konektorová okna reprezentují jednotlivé subnástroje virtuálních instrumentů. Ovládací prvky a indikátory na předním panelu umožňují uživateli kontrolovat vstup a výstup dat spuštěného virtuálního instrumentu. Nicméně přední panel lze použít také jako programové rozhraní. Pokud je vložen jako uzel do blokového schématu, definuje vstupy a výstupy uzly skrz svá konektorová okna. To znamená, že každý VI lze snadno testovat předtím, než je vestavěn jako podprogram do většího celku.

Grafický přístup také dovoluje uživatelům bez hlubších znalostí programovacích jazyků snadno vytvářet aplikace pouhým přetažením virtuální reprezentace laboratorních zařízení, s nimiž jsou obeznámeni. Tvorbu malých, jednodušších aplikací obsahuje softwarová dokumentace. Což na jedné straně představuje výhodu, ale existuje také určité nebezpečí podcenění odborné znalosti potřebné pro dobrou kvalitu "G" programování. Pro složitější algoritmy nebo rozměrnější kódy je důležité, aby programátoři měli rozsáhlé znalosti o zvláštní LabVIEW syntaxi a topologii jeho paměti. Nejmodernější LabVIEW vývojové verze nabízejí možnost budování stand-alone (samostatně spustitelných) aplikací. Dále lze vytvářet distribuované aplikace, snadněji implementovatelné vzhledem k paralelní povaze G-kódu, které komunikují prostřednictvím schématu klient/server.



Obr. 26 Ukázka pracovní plochy vývojového prostředí LabVIEW

Předností LabVIEW vůči ostatním vývojovým prostředím je rozsáhlá podpora přístupu k měřicímu přístrojovému vybavení. Ovladače a abstraktní vrstvy rozmanitých typů nástrojů a sběrnici jsou zahrnuty nebo jsou k dispozici pro začlenění. Abstraktní vrstvy nabízejí standardní softwarové rozhraní pro komunikaci s hardwarovým zařízením. Pokud jde o výkonnost, LabVIEW obsahuje překladač, který produkuje nativní kód pro platformu procesoru. Grafický program je přeložen do spustitelného strojového kódu pomocí interpretace syntaxe a její kompilace. Spustitelné a zdrojové kódy jsou sloučeny do jediného souboru. Spustitelný kód běží s pomocí LabVIEW run-time engine, který obsahuje některé předkompilované části pro plnění nejběžnějších úkolů, které jsou definovány v G jazyce. Run-time engine redukuje kompilace a také zajišťuje konzistentní rozhraní pro různé operační systémy, grafické systémy, hardwarové komponenty, atd. Run-time prostředí činí kód přenosným napříč platformami. Obecně platí, že LV kód může být pomalejší než ekvivalentně sestavený C kód, i když rozdíly často souvisí více s programem pro optimalizaci výkonu, než vlastní rychlostí.

V uživatelském balíčku poskytuje mnoho knihoven s celou řadou funkcí pro sběr dat, generování signálů, matematických operací, statistiky, zpracování signálu, analýzy a mnoha

grafických prvků. Počet bloků pro pokročilé matematické funkce, jako je integrace, filtrování a další speciálních vlastností, obvykle spojených se sběrem dat ze senzorů je obrovský. Kromě toho obsahuje LabVIEW obsahuje textově založené programové složky nazývané MathScript s dalšími funkcemi pro zpracování signálu, analýzu a matematické operace. MathScript mohou být integrovány s využitím grafického programování a použitím m-file souborů, skriptovací syntaxe, která je obecně kompatibilní s programem MATLAB.

Plně objektově orientovaný charakter kódu umožňuje opětné použití bez úprav. Tak dlouho, dokud datové typy vstupů a výstupů jsou konzistentní, budou dva dílčí virtuální instrumenty zaměnitelné.

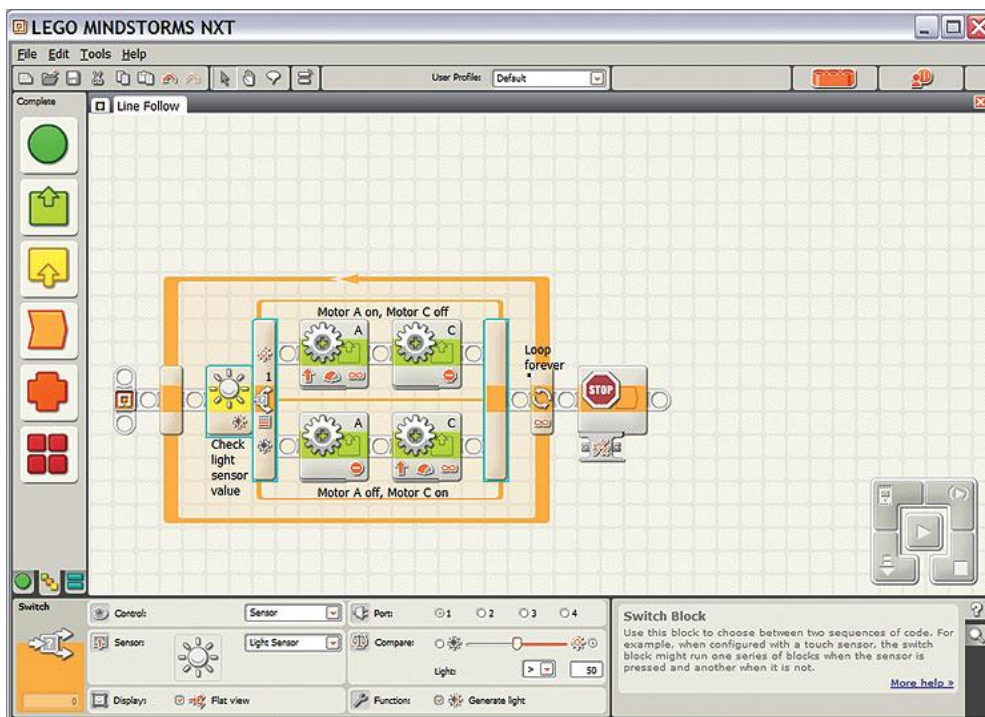
National Instruments se stále více zaměřuje na schopnost nasazení LabVIEW kódu pro rozrůstající se počet koncových zařízení, jako Phar Lap OS, PocketPC, PDA, FieldPoint moduly a speciální FPGA desky. Existuje nízkonákladová verze LabVIEW Student Edition zaměřené na vzdělávací instituce pro účely studia. K dispozici je také aktivní komunita uživatelů LabVIEW kteří komunikují přes několik e-mailových skupin a internetových fór.

2.2 NXT-G

NXT-G je integrované grafické vývojové prostředí (IDE) vyvinuté z profesionálního softwaru LabVIEW firmy National Instruments, dodávané jako standardní součást sady NXT. NI a LEGO udržují dlouhodobý, kooperativní vztah, který začal v roce 1998 s rozvojem ROBOLABu, programovacího softwaru vytvořeného pro původní školní produkty LEGO MINDSTORMS RCX. ROBOLAB, který je rovněž založen na NI LabVIEW, vyzdvihnul LEGO MINDSTORMS na přední pozici mezi softwarovými produkty pro výuku robotiky, jakožto účelný nástroj pro pedagogy na celém světě.

Důvodem vzniku bylo zaměření se na uživatele s nízkými zkušenostmi v programování pomocí konvenčních jazyků, či programování obecně. Zřetel je proto kladen především na intuitivnost, nativitu a jednoduchost vývojového prostředí i procesu. Program se vytváří obdobně jako v LabVIEW na základě metodiky drag and drop – umístováním a propojováním funkčních bloků a nastavením jejich vlastností. Každý funkční blok zastupuje jednu konkrétní činnost, např. řízení motorů, snímání hodnot senzorů, nastavení časovačů, matematické operace, přehrávání zvuků, atp. Uživateli je také umožněno

vytvářet pomocí tradičních programovacích metod vlastní kódové bloky, či jejich skupiny a začleňovat je jako podcelky hlavního programu. [1], [2], [4], [5]



Obr. 27 Ukázka pracovní plochy vývojového prostředí NXT-G

Některé nevýhody NXT-G:

- Grafické programovací rozhraní není vhodné pro kompozici rozsáhlejších programů.
- Deklarace a práce s uživatelsky definovanými proměnnými je komplikovaná.
- Nepřítomnost polí (datových struktur, které sdružují daný počet prvků o stejné velikosti) a jejich obslužné rutiny mezi základními bloky.
- Nemožnost pracovat a zobrazovat na LCD displeji jiná než celá čísla. Neexistuje možnost operovat s čísly s plovoucí desetinou čárkou.
- Jiná vývojová prostředí poskytují lepší zázemí pro debugging programů.
- Vývojový proces v grafickém jazyce je často časově náročnější, než programování v jazyce s textově orientovanou syntaxí.

2.2.1 RobotC

ROBOTC je výkonný programovací jazyk na bázi jazyku C pro prostředí operačního systému Windows, určený k psaní a ladění programů NXT a jediný programovací jazyk na této úrovni, který nabízí komplexní, realtime debugger. Je křížovou platformou, která umožňuje studentům učit se typu programování v syntaxi jazyku C, používané často v moderním vzdělání a profesionálních aplikacích. [1], [2], [4], [5]

RobotC nabízí nastavitelný prediktivní textový editor s šablonami kódových struktur, statistickými funkcemi, kvalitním překladačem jazyku C a interaktivním odlad'ovacím programem. Dokáže nepřetržitě sledovat až padesát různých příčin softwarového selhání. Podporuje uživatelsky definované zarážky (brakepoints) ve zdrojovém kódu. Pokročilým nastavením lze přizpůsobit funkcionalitu úrovni zkušeností jednotlivých uživatelů.

RobotC poskytuje inženýrský přístup k designování a analýze systémů, rychlému vytváření prototypů, řešení problémů, týmové práci a kritickému myšlení. K dispozici jsou ukázkové aplikace zahrnující příklady z reálného světa. Obsahuje jednoduše použitelné integrované řídicí algoritmy. Uzavřené smyčky PID regulátorů, pro řízení motorů, s nastavitelnými parametry proporcionální, integrální a derivační složky. Umožňuje nekomplikovaný přístup ke sběru a zpracování dat s výstupem do Microsoft Excel a rozsáhlé množství zpětnovazebních a řídicích strategií k experimentálním účelům.

Rozšířené složku systému pracují se standardy jazyku C, lze využít ke komunikaci pomocí zavedených průmyslových komunikačních protokolů nebo definovat uživatelsky vytvořené řešení. Dále je možné programově nízko-úrovňově a vysoko-úrovňově ovládat hardware, vyvíjet, přidávat a přizpůsobovat senzory, či rozhraní rychlých periférií, jako jsou klávesnice nebo displej, experimentovat se souběžnými vícevláknovými aplikacemi, plánovacími algoritmy a dalšími funkcemi a analyzovat, testovat a revidovat bezdrátová spojení mezi více robotickými systémy. Integrovány jsou také pojmy z mnoha dílčích disciplín elektrických a výpočetních věd: programování a vytváření rozhraní mikro počítačů, analogová a digitální elektronika, počítačové inženýrství, řízení a komunikace.

2.3 Alternativní programovací jazyky

Uvolněním zdrojového kódu firmwaru inteligentní kostky NXT a poskytnutím detailní softwarové vývojové sady SDK, učinilo LEGO smělý krok vůči programovým vývojářům, kteří se pokoušeli modifikovat stávající či navrhnout vlastní firmware, programovací nebo uživatelské rozhraní pro tuto komponentu. Od srpna roku 2006 byla na základě poznatků uveřejněných v SDK vytvořena komunitou celá řada aplikací. [1], [2], [4], [5]

Mezi nejvýznamnější se řadí:

- **NBC/NXC/BricxCC** – NBC (Next Byte Codes) a NXC (Not Exactly C) jsou jazyky pracující na standardním firmwaru kostky NXT. NBC je jednoduchý jazyk, syntaxí podobný assembleru neboli jazyku symbolických adres. Assembler je programovací jazyk nejnižší úrovně, závislý na strojovém kódu procesoru. Název assembler se používá i pro překladač, který tento jazyk překládá do strojového kódu. NXC se řadí do rodiny vyšších textově orientovaných programovacích jazyků, podobných svou syntaxí jazyku C. Vychází z instrukcí assembleru NBC. BricxCC (Bricx Command Center) definuje pokročilé integrované uživatelské vývojové rozhraní zahrnující jazyky NBC a NXC. Všechny tři platformy jsou relativně stabilní a rozvinuté.
- **pbLUA** – je implementací dynamického skriptovacího jazyku LUA pro platformu NXT. Lua je odlehčený, reflexivní, imperativní a procedurální programovací jazyk navržený jako skriptovací jazyk s rozšířitelnou sémantikou. Název je odvozen z portugalského slova pro měsíc. Lua byl vytvořen v roce 1993 Robertem Ierusalimským, Luizem Henriquem de Figueiredo, a Waldemar Celesem, členy Computer Graphics Technology Group v PUC-Rio na Pontifical University of Rio de Janeiro v Brazílii. Lua je dynamicky psaný jazyk určený k použití jako rozšiřující nebo skriptovací jazyk, dostatečně kompaktní, aby se vešel na nejrůznější hostitelské platformy. Podporuje pouze malý počet atomárních datových struktur jako logické hodnoty, čísla (double - s přesností plovoucí desetinné čárky ve výchozím nastavení), a řetězce. Typické datové struktury jako jsou pole, množiny, hashovací tabulky, seznamy a záznamy mohou být zastoupeny pomocí jediné nativní datové struktury, tabulky, což je v podstatě heterogenním mapa. V

důsledku toho zabírá úplný referenční interpret se základním jazykem pouze 150KB a snadno jej lze přizpůsobit pro širokou škálu aplikací.

- **LeJOS NXJ** – je software nahrazující původní firmware programovatelné kostky NXT. LeJOS obsahuje interpret Java Virtual Machine, který umožňuje využívat podmnožinu funkcí programovacího jazyku Java. LeJOS rovněž poskytuje rozsáhlé knihovny, které podporují různé zajímavé vyšší funkce, například navigační nebo na chování založené robotiky. Byl vyvinut pro sadu RCX. Přejít na novější verzi MINDSTORMS NXT proběhnul v lednu 2007. Nová verze je mnohem rychlejší (přibližně 15x), disponuje větší pamětí v nabídce systému, podporou Bluetooth a mnoha dalších funkcí kostky NXT.

II. PRAKTICKÁ ČÁST

3 APLIKACE PRO ŘÍZENÍ ROBOTU SPIKE

Model Spike je jedním ze čtyř základních typů robotů sestavitelných z dílů stavební sady MINDSTORMS NXT. Jedná se o štíru podobný mobilní robot s kráčejším podvozkem, vybavený dvěma servomotory pro ovládání pohybu na každé straně a jedním pro řízení ocasu / žahadla s dotekovým senzorem na jeho konci (viz obrázek pod textem). Dále tělo robotu obsahuje v přední části umístěný ultrazvukový senzor, dovolující snímat překážky do vzdálenosti až 2,5 m a zvukový senzor pro přijímání audio signálů. Chování robotu řídí inteligentní programovatelná kostka NXT. Je možné jeho dovybavení dalšími druhy senzorů.



Obr. 28 Laboratorní model robotu Spike

Aplikace pro řízení laboratorního modelu Spike by měla splnit tyto základní požadavky:

- Řízení motorů, dodávaného výkonu, počtu otáček, směru otáčení a tedy řízení směru a rychlosti pohybu robotu.
- Řízení pohybu ocasu, jeho vystrčení a zatažení, simulující jedový útok reálného živočicha, včetně snímání stisknutí dotekového snímače na jeho konci.
- Snímání hodnot ultrasonického senzoru, dovolující zaznamenání a obcházení překážek.
- Snímání hodnot zvukového senzoru, umožňující např. zadávat robotu povely pro započítí a ukončení činnosti.

- Kontrolu a řízení všech vnitřních programových pochodů robotu.
- Zobrazení informací na displeji robotu
- Zabezpečení možnosti komunikace s nadřazeným řídicím prvkem, například PC, schopným, přijímat, odesílat a zpracovávat data získaná robotem.
- Univerzální přístup a dostatečné vývojové zázemí k uplatnění řídicí aplikace pro jiné typy laboratorních robotů NXT, nasazení při řešení v charakterově obdobných úlohách

Jako dostatečně silný vývojový nástroj pro tvorbu aplikací, určených k ovládání robotů NXT se ukázal toolbox RWTH matematického software MATLAB společnosti Mathworks. Toolbox byl vytvořen na univerzitě Rheinisch-Westfaelische Technische Hochschule v Cáchách.

3.1 MATLAB

MATLAB je kvalitní integrované vývojové prostředí pro vědeckotechnické výpočty, modelování, návrhy algoritmů, simulace, analýzu a prezentaci dat, paralelní výpočty, měření a zpracování signálů, návrhy řídicích a komunikačních systémů. MATLAB je nástroj jak pro pohodlnou interaktivní práci, tak pro vývoj širokého spektra aplikací. [14]

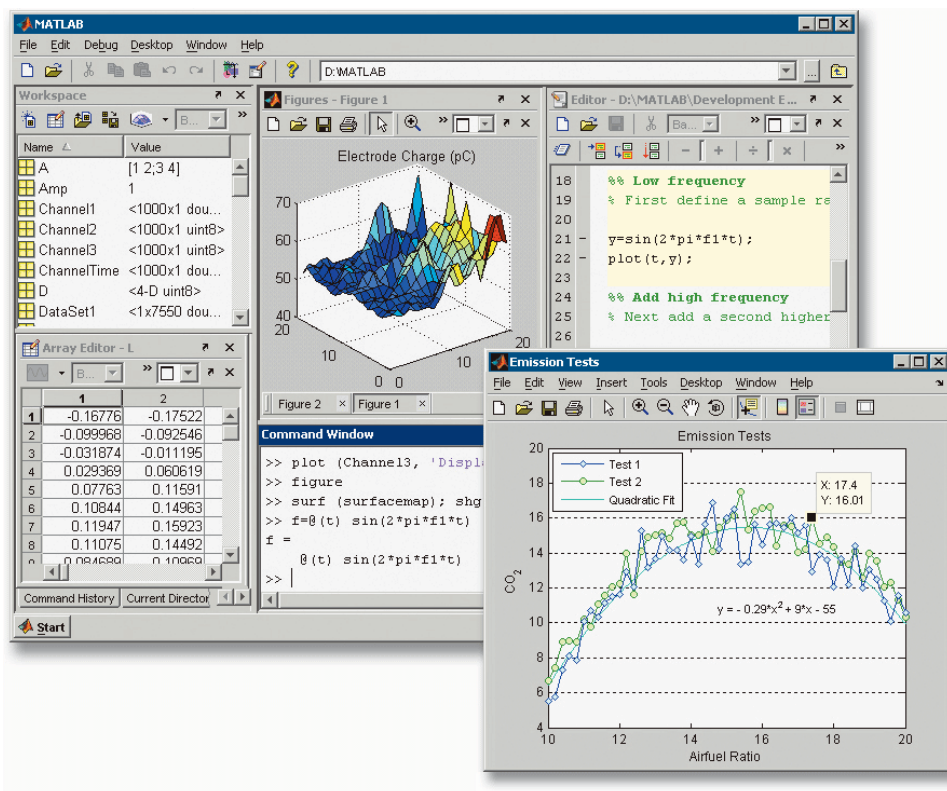
Výpočetní systém MATLAB se během uplynulých let stal celosvětovým standardem v oblasti technických výpočtů a simulací nejen ve sféře vědy, výzkumu a průmyslu, ale i v oblasti vzdělávání. Je považován za přelom nejen z hlediska rozsahu, integrace a kvality produktu, ale především z hlediska vztahu k uživateli a jeho pohodlí při práci.

MATLAB poskytuje svým uživatelům mocné grafické a výpočetní nástroje, ale i rozsáhlé specializované knihovny funkcí spolu s výkonným programovacím jazykem čtvrté generace. Knihovny jsou svým rozsahem využitelné prakticky ve všech oblastech lidské činnosti.

Díky své architektuře je MATLAB určen zejména těm, kteří potřebují řešit početně náročné úlohy a přitom nechtějí nebo nemají čas zkoumat matematickou podstatu problémů. Za nejsilnější stránku MATLABu je považováno mimořádně rychlé výpočetní jádro s optimálními algoritmy, které jsou prověřeny léty provozu na špičkových

pracovištích po celém světě. MATLAB byl implementován na všech významných platformách (Windows, Linux, Solaris, Mac).

Všechny laboratorní úlohy byly vytvořeny na patnáctidenní trial verzi MATLAB 7.6, přístupné licencovaným uživatelům (licence UTB ve Zlíně) z oficiální webové stránky společnosti MathWorks - <http://www.mathworks.com/products/MATLAB/tryit.html>.



Obr. 29 Ukázka pracovní plochy softwaru MATLAB

3.2 RWTH Toolbox

RWTH toolbox je rozšiřující nástrojovou sadou matematického softwaru MATLAB pro ovládání robotů MINDSTORMS NXT, pomocí bezdrátové komunikace Bluetooth. Spadá do kategorie open source projektů pod GNU General Public License (GPL). Zdrojové kódy softwaru pod GPL mohou být svobodně upravovány, používány a šířeny, a to obvykle bezplatně (případně za cenu distribučních nákladů). Vývoj toolboxu byl inspirován studentským projektem, uskutečněným na univerzitě RWTH v Čáchách – „MATLAB meets LEGO MINDSTORMS“, zaměřeným na studenty elektrotechniky a proto je určen především pro vzdělávací účely. [15]

Soubor funkcí vychází z bezdrátové LEGO MINDSTORMS NXT Bluetooth komunikačního protokolu pro ovládání inteligentní kostky NXT prostřednictvím bezdrátového spojení Bluetooth. Ačkoliv spojení Bluetooth není, pro svou vysokou latenci, doporučeno pro ovládání robota v reálném čase prakticky vůbec, poskytuje tento toolbox funkce pro přímo komunikaci s robotem.

Hlavní výhodou tohoto konceptu dálkového ovládání je umožnit spojení robota s aplikací složitých matematických operací a vizualizace v MATLABu. RWTH soubor nástrojů otevírá široké spektrum možností vybavit roboty, umělou inteligenci a dalšími zlepšeními za použití všech funkcí MATLABu a výpočtů pro digitální zpracování signálu. Výhodou je také možnost exportovat aplikaci vyvinutou v prostředí MATLABu, jako samostatně spustitelný program.

Toolboxem RWTH jsou podporovány funkce pro:

- Bluetooth komunikace
- Snímání NXT senzorů (dotykový, světelný, zvukový a ultrasonický)
- Řízení NXT servomotorů
- Monitorování sekundárních systémů (stav baterie, přehrávání zvuků, ...)
- Pomocné funkce implementované pro zefektivnění programování

Mezi charakteristické vlastnosti toolboxu RWTH patří (ve srovnání s ostatními programovacími jazyky, pracujícími s kostkou NXT):

- Velké zvýšení funkcionality díky využití výpočetního výkonu a paměti PC
- Téměř neomezená velikost programu (ve srovnání s NXT)
- Ovládání více robotů pomocí jediného programu (limitováno pouze počtem souběžně instalovaných Bluetooth adaptérů)
- Vzdálené vykonávání programů v prostředí MATLAB, zatímco na kostce NXT probíhá samostatný program
- Lze využít přídatný hardware jako webové kamery, joysticky apod.
- Pokročilá podpora ladění zahrnující užití zářezek, spouštění krok po kroku nebo inspekci proměnných za běhu programu

- Možnosti pokročilé síťové a Internetové komunikace
- Působivá 2D a 3D vizualizace dobře zdokumentovaná dokumentace
- Použití stávajících programů, toolboxů nebo knihoven, např. neuronových sítí, zpracování obrazu, databázových komponent
- Jednoduše použitelný profilovací program pro odhalení potíží v programu
- Využití uživatelské příručky softwaru MATLAB pro snadnější návrh aplikací a formulářů metodou drag and drop
- Stabilita toolboxu byla úspěšně testována více než 300 studenty, aplikována na více než 75 NXT robotech a 150 počítačích s různými operačními systémy

3.3 Volba adaptéru Bluetooth

Pro vytvoření bezdrátového spojení pomocí rozhraní Bluetooth je zapotřebí, aby obě zařízení tuto komunikaci podporovala. Moderní příruční počítače (Notebooky) v dnešní době již běžně pracují s vestavěnými Bluetooth adaptéry. Problém však nastává, je-li nutné komunikovat s PC. Je také nutné brát zřetel nato, že ne každé zařízení je zcela kompatibilní s kostkou NXT. Vhodným řešením je využití adapterů připojitelného přes sběrnici USB. Cena takového zařízení se pohybuje v řádu stovek korun. Výčet kompatibility některých adaptéru Bluetooth uvádí následující tabulka převzatá z oficiálních stránek <http://mindstorms.lego.com/eng/Overview/Bluetooth.aspx>.

Název zařízení	Třída	OS	Kompatibilita
Abe UB22S (LEGO)	II	WinXP	Špatná
		Linux	Velmi dobrá
AnyCom	I	WinXP	Špatná
		Linux	Špatná
AVM - BlueFRITZ! USB v2.0	I	WinXP	Vynikající
		Linux	Vynikající
Belkin - F8T012 Bluetooth	I	WinXP	Špatná
		Linux	Špatná
Conceptronic - CBT200U2 Bluetooth V2.0	I	WinXP	Dobrá
		Linux	Dobrá
Level One - MDU-0025USB Bluetooth 2.0	I	WinXP	Velmi dobrá
		Linux	Velmi dobrá

Tab. 2 Kompatibilita Bluetooth adaptéru

4 NÁVRH LABORATORNÍCH ÚLOH

Navrhované laboratorní úlohy jsou zaměřeny na praktické využití nástrojové sady RWTH pro řízení robotických modelů stavebnice MINDSTORMS NXT. Předpokládají dostatečnou orientaci studenta ve vývojovém prostředí matematického softwaru MATLAB, užívání jeho funkcí a tvorby grafického uživatelského rozhraní. Cílem těchto úloh je zejména srozumitelnou formou uvést studenta do problematiky řízení robotických systému pomocí PC. Úlohy nevyčerpávají zcela dané téma, ani možnosti použití nástrojové sady RWTH, po jejich absolvování by však student měl být schopen samostatným konstruktivním způsobem uplatnit získané poznatky pro tvorbu náročnějších aplikací.

Posloupnost úloh je vystavěna hierarchicky tak, aby každá z nich tvořila uzavřený celek teoretických a praktických znalostí a přitom logicky navazovala na úlohy předcházející. Toto pořadí je doporučeno zachovávat především tam, kde náročnější úlohy vychází z konceptů použitých v úlohách předcházejících (Úloha č. 3).

Následující kapitoly ukazují výčet nejpodstatnějších částí laboratorních úloh a jejich vzorové vypracování. Kompletní zadání laboratorních úloh je součástí elektronické přílohy ve formátu PDF.

4.1 Úloha č. 1 – Ovládání servomotorů robotu NXT

Úkol měření:

Ovládání servomotorů robotu NXT

- Seznamte se s hardwarovými vlastnostmi inteligentní programovatelné kostky NXT, funkcionalitou výstupních portů a charakteristickými vlastnostmi servomotorů
- Seznamte se s konceptem a vlastnostmi RWTH toolboxu matematického softwaru MATLAB pro řízení robotů Mindstorms NXT pomocí PC
- Seznamte se podrobně s principy vysokoúrovňových funkcí toolboxu RWTH pro konfiguraci a navázání spojení pomocí bezdrátové komunikace Bluetooth
- Seznamte se podrobně s principy vysokoúrovňových funkcí toolboxu RWTH pro řízení servomotorů a snímání jejich základních parametrů

- Navrhněte grafické uživatelské rozhraní, pokrývající ovládání a snímání všech základních parametrů servomotorů NXT. Uživateli bude umožněno nastavení portu ovládaného motoru, příkonu, mezního úhlu, volba aktivace lineární rampy, synchronizace více motorů a snímání parametrů každého motoru
- Otestujte navrženou aplikaci a zhodnoťte kvalitu řízení i komponent kostky NXT

Seznam použitých přístrojů a součástek:

- Inteligentní programovatelná kostka NXT
- Sada servomotorů NXT
- Počítač s matematickým softwarem MATLAB 7.4 a vyšší
- Adaptér pro Bluetooth komunikaci

Funkce pro konfiguraci a navázání spojení pomocí Bluetooth:

Na počátku každé práce s matematickým softwarem MATLAB je dobré „vyčistit“ si pracovní prostředí, což se provede příkazy

```
Clear all;  
Close all;
```

Pro komunikaci s inteligentní kostkou NXT je nutné použít komunikační protokol SPP (Serial Port Profile), který ve své podstatě funguje jako obousměrný virtuální sériový port. To je důvodem, proč lze odesílat a přijímat data z MATLABu prostřednictvím sériového portu. Pro zpracování více připojení nebo nastavení Bluetooth adapterů na rozdílných počítačích slouží inicializační soubor s údaji pro funkce MATLABu, který musí být přítomen v aktuálním adresáři nebo jehož cestu lze dohledat prostřednictvím nastavení.

Formát inicializačního souboru vypadá následovně:

```
[Bluetooth]  
SerialPort=COM3  
BaudRate=9600  
DataBits=8  
SendSendPause=10  
SendReceivePause=30  
TimeOut=2
```

Konfigurace určuje číslo sériového portu, přenosovou rychlost, počet datových bitů. Parametr Timeout má vliv pouze při použití OS Windows. Stanoví čas, během něhož

Bluetooth čeká na „ztracená data“. Výchozí vnitřní nastavení MATLABu na hodnotu 10 způsobuje obtíže se zamrznutím v některých řídicích programech na některých počítačích (příčiny nejsou dosud nalezeny). Nastavením proměnné na hodnotu 2 lze získat velmi stabilní chod se vzácnými pozastaveními. Nižší hodnoty mohou vést ke skutečné ztrátě paketů, která nebyla prozatím zkoumána. Velmi důležité je, že NXT mohou zdánlivě ztrácet pakety / příkazy, protože vstupní vyrovnávací paměti (nebo fronta) jsou omezené velikosti. Jelikož nelze o tomto získat detailnější informace, jsou nastaveny vysílací a přijímací funkce tak, aby mezi jednotlivými přenosovými operacemi vyčkávali. V tomto případě budeme požadovat 10 ms prodlevu mezi dvěma po sobě jdoucími odesílacími operacemi. Na druhé straně, 30 ms pauza je nutná mezi odesílací a přijímací zprávou (a naopak). To by mělo dát NXT dostatek času, k přepínání mezi přenosovými módy.

Pro vytvoření inicializačního souboru je možné použít standardní textový editor nebo pohodlnější způsob zavoláním grafického uživatelského rozhraní příkazem:

```
BT_MakeConfigFile;
```

Následující funkce pracují stejně dobře pod Windows i Linuxem s jednou výjimkou, handle navrácený Windows má sériovou strukturu, zatímco handle vracený Linuxem je jednoduchý skalár. Všechny interní funkce na to berou zřetel. Před získáním nového handlu je dobré zrušit všechna předchozí spojení. Pro aktuální inicializační soubor (např. bluetooth.ini) se tak učiní příkazem

```
BT_CloseAllHandles ('bluetooth.ini');
```

který uzavře všechny porty přiřazené jím otevřené. Více drastickou, ale často nevyhnutelnou metodou je použít příkaz pro uzavření všech sériových portů:

```
BT_CloseAllHandles;
```

Nyní lze otevřít spojení. Je praktické se ujistit, zda je kostka NXT skutečně připojena k rozhraní Bluetooth, před zavoláním této funkce:

```
h = BT_OpenHandle ('bluetooth.ini', 'check');
```

Volitelný argument „check“ způsobí, že funkce odesílající kontaktní paket nevrátí handle na nové spojení do doby, není-li obdržena odpověď. Tento rys je velmi užitečný k detekci poruchy nebo uzavřeného Bluetooth spojení před vykonáním programu. Nastavení výchozího globálního handlu a jeho opětovné získání provádí funkce

```
BT_SetDefaultHandle (h);
```

```
handle = BT_GetDefaultHandle;
```

Uzavření komunikace pro jednu nebo všechny konfigurace portů lze provést funkcemi

```
BT_CloseHandle (h);
BT_CloseAllHandles ('bluetooth.ini');
```

Funkce pro řízení servomotorů NXT a snímání jejich základních parametrů:

Tyto funkce poskytují jednoduchý vysokoúrovňový přístup k ovládání všech vlastností servomotorů NXT. Hlavní myšlenkou je zadat nejdříve číslo motoru (výstupního portu), se kterým se bude pracovat a posléze sled po sobě jdoucích příkazů, ovlivňující nastavení parametrů motoru. Poté co jsou všechny žádané parametry nastaveny, stačí vyslat příkaz k provedení.

Při určení nastavovaného motoru lze použít čtyři druhy syntaxe:

```
SetMotor (0);
SetMotor ('0');
SetMotor 0;
SetMotor (MOTOR_B);
```

Standardní způsob, vstup pomocí řetězce, fancy syntaxi nebo použití symbolické konstanty zastupující čísla portů (MOTOR_A, MOTOR_B, MOTOR_C). Záleží na uživateli, kterou metodiku si zvolí, nicméně doporučované je používání symbolických konstant.

Funkce *GetMotor* vrací číslo portu právě nastavovaného motoru. Může být užitečná u funkcí, které vyžadují tuto hodnotu jako vstup, například

```
NXT_ResetMotorPosition (GetMotor, true);
```

Je-li aktivní port nastaven, lze začít se specifikací parametrů. Následující příklad uvádí provedení jedné otočky motoru s výkonem 50%. Po specifikaci parametrů jsou odeslány příkazy k vykonání.

```
SetMotor (MOTOR_A)
    SetPower (50);
    SetAngleLimit (360);
SendMotorSettings
```

Vyžaduje-li se nekonečný běh motoru (resp. do okamžiku dalšího nastavení), používá se úhlový limit 0

```
SetMotor (MOTOR_A)
    SetPower (50);
    SetAngleLimit (0);
```

```
SendMotorSettings
```

Často se požaduje, aby motor rotoval přesnou konstantní rychlostí při různém zatížení. To je uskutečnitelné spuštěním regulace rychlosti. Stejně jako všechny ostatní příkazy, i tento ovlivňuje v daný okamžik pouze motor, jenž je právě nastavován.

```
SpeedRegulation on;
```

Jeden pozoruhodný účinek na vnitřní rychlosti regulace je, že motor zvyšuje výkon v případě, že požadované otáčky nemohou být splněny. Např. je-li nastaven pohyb s výkonem 20%, bez regulace rychlosti, ale robot je příliš těžký, ztrácí tah (i když lze slyšet motory, že stále běží). Regulace otáček umožní robotu vnitřně zvýšit svůj výkon a uvést se tak do pohybu, jakoby proti němu nepůsobila žádná síla. Hranici 100% výkonu však nelze překonat.

Účelným módem je synchronizace dvou motorů. Synchronizace koriguje dva motory tak, že působí, jako by byly spojeny nápravou.

```
SetMotor (MOTOR_B)  
    SyncToMotor (MOTOR_C);  
    SetPower (50);  
SendMotorSettings
```

Ve výše uvedeném příkladu robot začne jízdu vpřed (za předpokladu, že motory 1 a 2 ovládají pohyb kol). Od bodu, kdy je zavolána funkce *SyncToMotor*, všechna nastavení ovlivňují oba synchronizované motory. Funkce *SendMotorSettings* bude pak vnitřně posílat pakety pro každý z motorů, jako kdyby byly skutečně spojeny. Pouze synchronizací dvou motorů lze provést pohyb po křivce nebo kruhový pohyb.

```
SetMotor (MOTOR_B)  
    SyncToMotor (MOTOR_C);  
    SetPower (50);  
    SetTurnRatio (100);  
    SetAngleLimit (200);  
SendMotorSettings
```

Parametr Turn Ratio rozděluje výkon mezi motory. Hodnota 100 znamená, že jeden motor se roztočí dopředu a druhý dozadu, což má za následek maximální možný rotační účinek. Hodnota 50 nastaví chod motorů tak, že jeden běží a druhý se zastaví. Hodnotami 1 a 49 lze dostat pohyb po křivce (obě kola se roztočí v rozdílných otáčkách). 0 rozběhne oba motory směrem vpřed. Zápornými hodnotami lze dosáhnout zrcadlových výsledků. Synchronizace motorů se ukončuje funkcí

```
SyncToMotor off;
```

Je potřeba připomenout, že motory mohou v jednom okamžiku buď synchronizovány, nebo může být použita jejich regulace. Před začátkem regulaci rychlosti je nutné ručně vypnout synchronizaci a naopak. Regulaci nebo synchronizaci lze resetovat také funkcí *StopMotor*, viz dále.

```
SyncToMotor off;
SpeedRegulation on;
```

Speciální nastavení se používá pro lineární nárůst rychlosti motoru, rostoucí lineární rampu, která v podstatě plynule zrychluje ze staré ho výkonu na nový. Parametr je platný pouze, pokud byl zadán úhel omezení *SetAngleLimit*. Jedná se o rotační vzdálenost, za kterou motor upraví svůj výkon na novou hodnotu.

```
SetPower (100);
SetAngleLimit (1080);
SetRampMode up;
```

Obdobně se provede lineární zpomalení

```
SetPower (0);
SetAngleLimit (1080);
SetRampMode down;
```

popřípadě vypnutí příkazem

```
SetRampMode off;
```

Všechna předcházející nastavení synchronizace motorů lze ovšem provést voláním jediné funkce. Následující příklad uvádí stav kdy MOTOR_A je synchronizovaný s MOTOR_B s výkonem 50%, mezní úhel je 360 stupňů, regulace rychlosti je vypnuta, mód rampy je nastaven na lineární nárůst a parametr koeficient otáčení (Turn Ratio) na hodnotu 25. Tato syntaxe zkracuje zdrojový kód, ale zhoršuje přehlednost jeho zápisu.

```
SendMotorSettings(MOTOR_A, 50, 360, 'off', MOTOR_B, 25, 'up')
```

Vypnutí otáčení motoru lze učinit opět několika způsoby:

```
StopMotor(GetMotor, 'off');
StopMotor(GetMotor, 'brake');
StopMotor('all', 'off');
```

Prvním příkazem se aktivnímu motoru vypne napájení a ten přechází do režimu volnoběhu (Coast mode), kdy je možné libovolně manipulovat s jeho rotační částí. Druhé nastavení dovoluje motor aktivně zabrzdít a držet v jeho stávající pozici. Funguje jako nouzová brzda pro řízení robotů (s poměrně vysokou spotřebou energie). Poslední příkaz odešle

synchronní povel k zastavení na všechny tři porty motorů v jeden okamžik. Je výhodné použít tuto funkci na začátku a konci programu, kdy přecházejí všechny motory do klidového stavu. Zároveň výše zmíněný kód uvádí další případ užití funkce *GetMotor*.

Důležitou součástí řízení motoru je snímání hodnot jeho rotačního senzoru. Část kódu pod tímto textem uvádí všechny snímatelné proměnné servomotoru. Pro přehlednost jsou vysvětlivky zachovány jako poznámky zdrojového kódu. Výsledky snímání jsou uloženy do proměnné *out*.

```
out = GetMotorSettings(MOTOR_C);
out.IsRunning           % určuje, zda je motor v chodu
out.Power               % aktuální výkon motoru
out.AngleLimit          % aktuální omezení úhlu, 0 znamená bez omezení
out.TurnRatio           % aktuální koeficient otáčení

out.SpeedRegulation     % určuje, zda probíhá regulace rychlosti
out.SyncToMotor         % určuje, je-li motor synchronizován, -1 značí
                        nesynchronizován

out.TachoCount          % vnitřní, neresetovatelný čítač otáček (ve
                        stupních)
out.Angle               % aktuální, resetovatelná, pozice natočení motoru

out.MotorBrake          % logická proměnná, určující, zda je motor aktivně
                        brzděn
```

Vnitřně tyto funkce samozřejmě využívají funkce nižší úrovně *NXT_GetOutputState*. Je dobré si povšimnout, že hodnota *.Angle* je skutečně vnitřní proměnná *NXT BlockTachoCount*, kterou lze nastavit pomocí *ResetMotorAngle()* nebo *NXT_ResetMotorPosition (GetMotor, true)* (tj. to, co se nazývá relativní pozice motoru v oficiální dokumentaci Mindstorms). Příklad resetování čítače otáček motoru:

```
ResetMotorAngle (MOTOR_A);
```

Proměnná *.TachoCount* na druhé straně reprezentuje čítač otáček podobný počítadlu kilometrů v automobilu. Udává ve stupních celkovou hodnotu otáček motoru, od chvíle kdy byla programovatelná kostka *NXT* zapnuta. Tato hodnota není resetovatelná (může však být resetována, při novém spuštění externího programu, které lze vynutit pomocí *NXT_StartProgram*). Lze ji také nastavit zpět na nulu, jestliže motor vykoná stejný počet otáček v opačném směru. Nejeefektivnějším způsobem je restart kostky *NXT*.

Chování další užitečné funkce objasňuje následující kód

```
SetMotor (MOTOR_A)
    SetPower (20);
    SetAngleLimit (360);
SendMotorSettings
WaitForMotor (GetMotor);
NXT_PlaySound(440, 200);
```

Motor je nastaven tak, aby provedl otáčku o 360 stupňů při příkonu 20%. Funkce *WaitForMotor* zajistí, aby následující akce (v tomto přehrání tónu o frekvenci 440 Hz) byla vykonána až po dokončení této rotace. *WaitForMotor* lze použít také jako funkci se dvěma parametry. V tom případě druhý parametr značí čas v sekundách, po němž bude činnost motoru přerušena. Pokud činnost skončí dříve, funkce *WaitForMotor* se automaticky ukončí. Této vlastnosti se využívá při uvážnutí robotu v nekonečné smyčce nebo jako alternativního pokračování programu.

Výstupního portu lze využít také ke spínání (např. světel).

```
SwitchLamp (MOTOR_C, 'on');
pause(1)
SwitchLamp (MOTOR_C, 'off');
```

Je-li první parametr nastaven jako all, bude povel odeslán na všechny tři výstupní porty:

```
SwitchLamp ('all', 'off');
```

Postup návrhu:

Po prostudování materiálů zabývajících se hardwarovým vybavením stavebnice MINDSTORMS NXT a seznámení se s hlavními rysy servomotorů a nástrojové sady RWTH se přistoupí ke konfiguraci některých nastavení programovatelné kostky a její zpřístupnění komunikaci pomocí rozhraní Bluetooth.



Obr. 30 Kostka NXT

Kostka se zapíná oranžovým tlačítkem On/Enter, umístěným v centru pod LCD displejem, které slouží také k potvrzování jednotlivých kroků a vstupů do nižších úrovní menu. Pohyb v menu se provádí šedými šipkami vpravo a vlevo od startovacího oranžového tlačítka. Poslední šedé tlačítko Zpět ve spodnější centrální části slouží ke zrušení volby nebo návratu do vyšší úrovně menu. V nejvyšší úrovni vyvolá volbu vypnutí programovatelné kostky.

Nejdříve je vhodné si nastavit kostku tak, aby nepřecházela po určitém čase do režimu spánku – což přeruší napájení a spojení pomocí bezdrátové komunikace (a může způsobit potíže s uváznutím sériových portů v „mrtvém“ stavu). Pomocí šipek vyhledáme menu Settings, označené obrázkem francouzského klíče. Stisknutím tlačítka Enter se vstoupí do podmenu a zde vybereme položku Sleep Mode, označenou obrázkem kolečka s čárkou v horní části. Pomocí šipek nastavíme výchozí hodnotu na Never a potvrdíme tlačítkem Enter. Tlačítkem Zpět se vrátíme do nejvyšší úrovně menu.



Obr. 31 Menu Settings

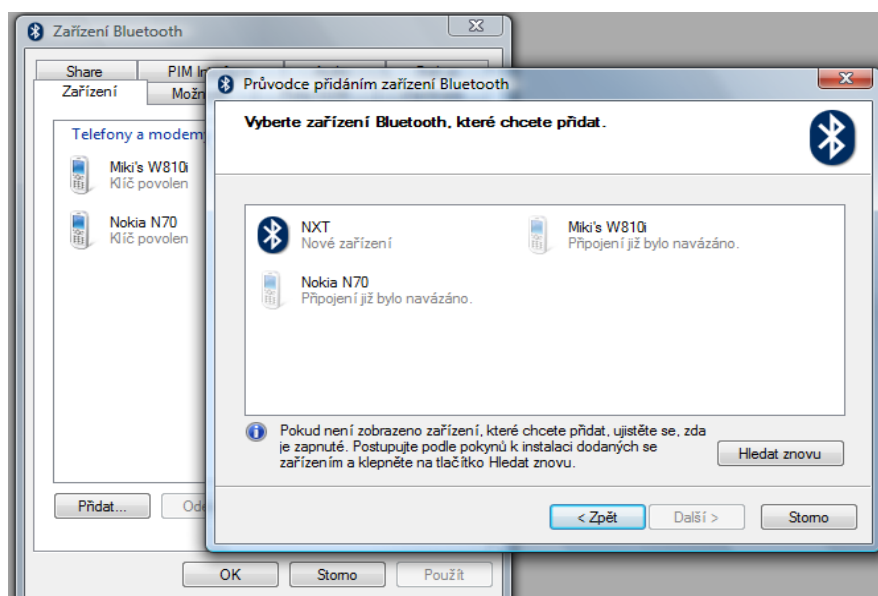
V hlavním menu se zvolí šipkami položka připojení Bluetooth, označená typickou značkou tohoto rozhraní. V podmenu nastavíme položky Visibility (zobrazená symbolem oka) do stavu Visible – viditelný, aby bylo zařízení rozpoznatelné pomocí PC a položku On/Off do stavu On, čímž se rozhraní zapíná a je připraveno k navázání komunikace. Kostka musí zůstat v zapnutém stavu po celou dobu komunikace, jinak může dojít k obtížím.



Obr. 32 Menu Bluetooth

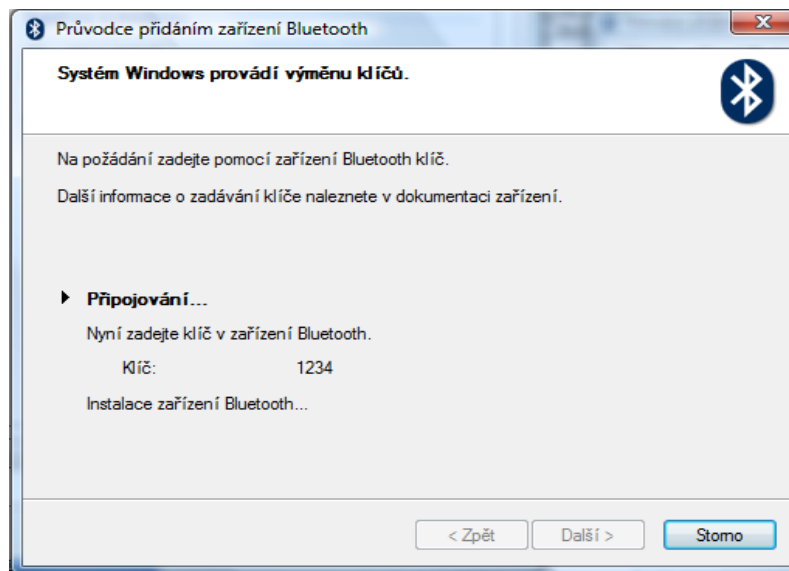
Nyní již lze vyhledat přichystaný mikropočítač pomocí PC. Následující postup popisuje zpřístupnění komunikace v operačním systému Windows Vista, v němž byly laboratorní úlohy vyvíjeny. Je potřeba poznamenat, že konkrétní kroky záleží také na nastavení vzhledu a funkcionality samotného systému Windows. Obdobně lze postupovat i při práci v ostatních systémech společnosti Microsoft. Ke zprovoznění komunikace v dalších operačních systémech (Linux, Mac) je doporučeno nastudovat si jejich dokumentační materiály.

Z nabídky Start zvolíme položku Ovládací panely a vybereme Zařízení Bluetooth. Po kliknutí na záložku Zařízení a tlačítko Přidat se otevře průvodce přidáním zařízení Bluetooth. Zaškrtneme volbu Zařízení je nastaveno a připraveno k rozpoznání a přesuneme se do následujícího okna kliknutím na tlačítko Další. Po prohledání dostupného okolí indikuje počítač programovatelnou kostku NXT.



Obr. 33 Průvodce přidáním zařízení Bluetooth

Dvojklikem na ikonu nového zařízení se otevře okno vyžadující zadání klíče. Zaškrtneme volbu Použit klíč nalezený v dokumentaci a do příslušného textového pole vepíšeme hodnotu 1234, což je standardní klíč, přednastavený v kostce NXT. Po potvrzení tlačítkem Ok počítač čeká na ověření klíče druhou stranou.



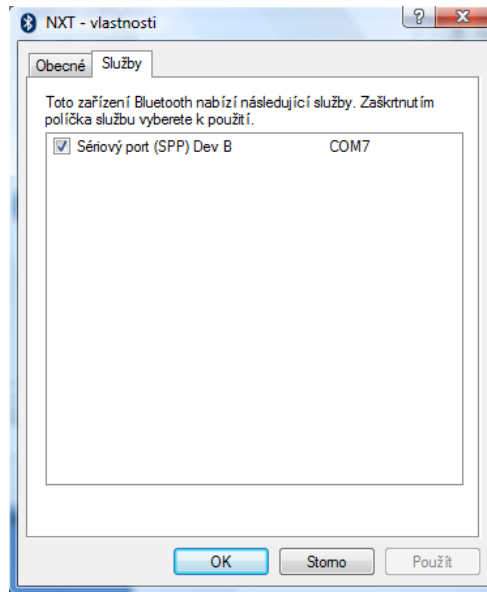
Obr. 34 Připojení k zařízení Bluetooth

Na displeji kostky NXT se automaticky zobrazí zadávací obrazovka potvrzení klíče. Jelikož jsme zvolili standardní klíč 1234, který je přednastavený stačí pouze potvrdit volbu stisknutím oranžového tlačítka Enter. Proběhne instalace ovladačů sériového portu pro obousměrnou komunikaci. V okně průvodce ukončím instalaci stisknutím tlačítka Dokončit.



Obr. 35 Potvrzení klíče

Je dobré se přesvědčit, zda ve vlastnostech v záložce Služby právě připojeného zařízení NXT je nastavena komunikace pomocí obousměrného sériového portu SPP. Kostka je připravena ke komunikaci s toolboxem RWTH.



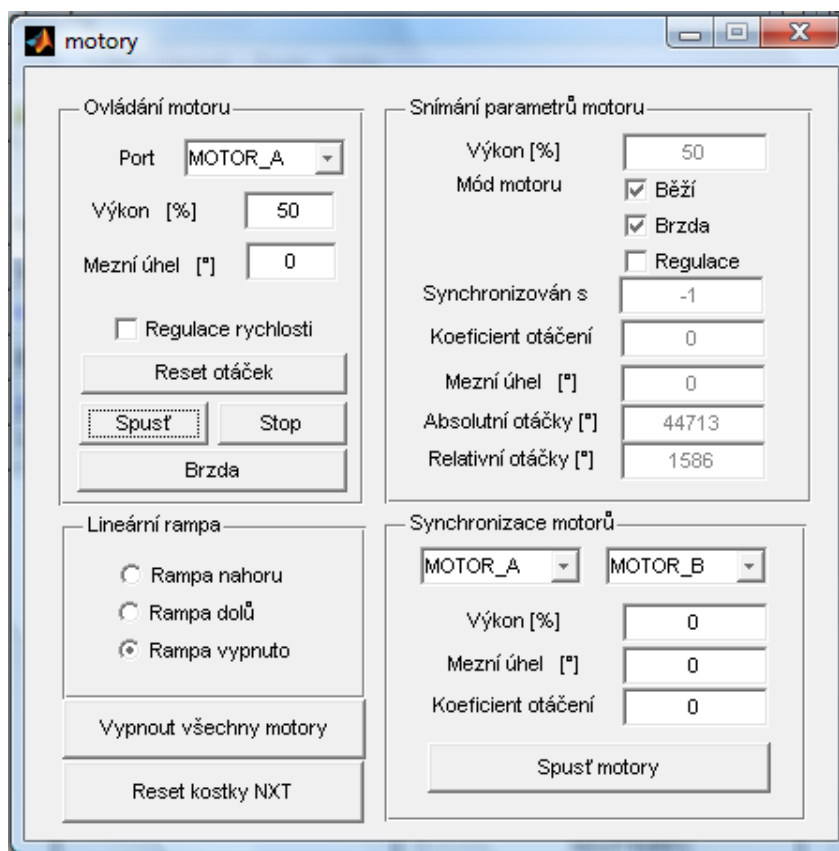
Obr. 36 Vlastnosti zařízení Bluetooth

Po spuštění matematického softwaru MATLAB vyčistíme nejdříve uživatelský prostor voláním funkcí *Close all* a *Clear all*. Poté je potřeba konfigurovat parametry připojení Bluetooth, nejlépe voláním funkce *BT_MakeConfigFile*, případně pomocí externího textového editoru a voláním příslušných funkcí získat nový komunikační handle. Dbáme zejména na to, aby číslo portu odpovídalo portu, na němž je připojeno zařízení NXT. Zbylé položky mohou zůstat nezměněny. Zavoláním funkce *guide* se otevře nástroj pro tvorbu grafického uživatelského rozhraní. Z nabídky zvolíme prázdnou GUI aplikaci – Blank GUI (Default).

Na čistou pracovní plochu postupně tažením umístíme vhodné prvky z levého panelu nástrojů a přizpůsobíme jejich vlastnosti na námi požadované. Otevření editoru vlastností se provádí dvojklikem na daný prvek. Snažíme se zachovat srozumitelná jména jednotlivých umístěných komponent, na něž se budeme odkazovat pomocí volacích funkcí. Vyšší přehlednosti aplikace docílíme rozčleněním okna pomocí panelů. V případě nutnosti přizpůsobíme velikost okna aplikace tažením černého čtverce v pravém dolním rohu. Po úspěšném dokončení návrhu vzhledu vygenerujeme m-file soubor popisující funkcionalitu okna a jeho prvků.

Oživení grafického rozhraní dosáhneme doplněním programové rutiny do volacích funkcí jednotlivých prvků. Hodnoty parametrů prvků jsou předávány pomocí handle sdružujícího tyto prvky (nejčastěji pojmenovaného „handles“). Atributy získáváme a nastavujeme voláním funkcí *set()* a *get()*.

Vzorová aplikace by měla vypadat asi takto



Obr. 37 Vzorová aplikace pro řízení motorů

Závěr:

Navržené grafické uživatelské rozhraní slouží k řízení servomotorů NXT a snímání jejich základních parametrů. Panel ovládání motoru v levé horní části nabízí uživateli možnost určit výstupní port, příkon motoru v procentech, mezní úhel otočení (rotaci ve stupních) a možnost aktivovat regulaci rychlosti, která zvyšuje příkon v případě, že motor nedosahuje požadované rychlosti. Spuštění a zastavení motoru se provede tlačítky Spust' a Stop (přičemž stop nejdříve motor aktivně zabrzdí a posléze vypne). Tlačítko Reset otáček vynuluje relativní čítač otáček motoru. Tlačítko Brzda přepne motor do stavu aktivního brzdění. Nastavení konstantní rychlosti nebo lineárního nárůstu případně poklesu rychlosti motoru umožňuje panel Lineární rampa, umístění v levé části pod panel ovládání motoru. Lineární rampa je panel checkboxů, které dovolují uživateli aktivovat vždy pouze jednu variantu.

V pravé horní části jsou zobrazovány výstupní údaje snímaných parametrů motoru. Údaje se aktualizují s každým spuštěním a ukončením činnosti servomotoru, případně změnou

výstupního portu motoru v panelu Ovládání motoru. Výtčet snímaných parametrů je patrný z obrázku.

V pravé spodní části je situováno ovládání synchronizace motorů. Uživatel může provést volbu synchronizovaných portů výběrem ze seznamu, nastavení příkonu, mezního úhlu a koeficientu otáčení, který určuje, kterým směrem se budou oba motory při synchronizaci točit. Vypnutí všech běžících motorů a reset čítače kostky NXT lze provést tlačítky v levé spodní části okna.

Při testování aplikace se projevíly nepřesnosti při snímání údajů enkodéru, vestavěného v útrokách každého servomotoru a potíže se synchronizací motorů, které vedly k občasným potížím, způsobeným pravděpodobně nedostatečnou mezerou mezi odesíláním jednotlivých požadavků pomocí bezdrátového rozhraní Bluetooth. Nastavený mezní úhel po vykonání činnosti motoru pravidelně zcela neodpovídal požadovaným otáčkám (S odchylkou přibližně 5-10 stupňů na 100 stupňů mezního úhlu).

Realizace programu je součástí elektronické přílohy diplomové práce.

4.2 Úloha č. 2 – Snímání hodnot senzorů robotu NXT

Úkol měření:

Snímání hodnot senzorů robotu NXT

- Seznamte se obecnými vlastnostmi senzorky a jejího využití
- Seznamte se s hardwarovými vlastnostmi inteligentní programovatelné kostky NXT, funkcionalitou vstupních portů a charakteristickými vlastnostmi standardních senzorů stavebnice MINDSTORMS NXT
- Seznamte se podrobně s principy vysokoúrovňových funkcí toolboxu RWTH pro konfiguraci a navázání spojení a snímání hodnot ze senzorů
- Navrhněte grafické uživatelské rozhraní, pokrývající ovládání a snímání všech základních parametrů senzorů kostky NXT a případné další statistické zpracování získávaných dat. Uživateli bude umožněna volba portu a použitého senzoru. Návrh bude obsahovat výstup snímaných hodnot do grafu. Přizpůsobte výstupní zobrazení os grafu, aktuálnímu snímanému senzoru

- Otestujte navrženou aplikaci a zhodnoťte kvalitu snímání i přesnost senzorů kostky NXT

Seznam použitých přístrojů a součástek:

- Inteligentní programovatelná kostka NXT
- Dotykový senzor NXT
- Světelný senzor NXT
- Zvukový senzor NXT
- Ultrasonický senzor NXT
- Počítač s matematickým softwarem MATLAB 7.4 a vyšší
- Adaptér pro Bluetooth komunikaci

Funkce pro odečítání hodnot senzorů NXT:

Následující funkce nabízí snadný a přehledný způsob, jak přistupovat ke snímání všech hodnot standardních senzorů, připojitelných k programovatelné kostce NXT. Ultrazvukový senzor používá digitální rozhraní a musí být zpracován na nižší úrovni pomocí složitějších mechanismů, než ostatní senzory, ale při využití funkcí vysoké úrovně programování nelze zaznamenat prakticky žádný rozdíl. Jediné, co stojí za zmínku je, že funkce *GetUltrasonic* je asi 1,5 až 2 krát pomalejší než ostatní funkce *Get** (protože vnitřně potřebuje odeslat 3 nebo někdy i 4 pakety namísto dvou). Funkce *Open** používají ke své činnosti nízkourovňovou funkci *NXT_SetInputMode*, zatímco funkce *Get** volá interně nízkourovňovou funkci *NXT_GetInputValues*. Je nezbytné používat funkci *CloseSensor* k vypnutí konkrétního čidla, např. pro vypnutí červené LED diody světelného senzoru v aktivním režimu, čímž se sníží výrazně spotřeba energie. K označení snímaného portu lze použít standardní čísla portů, které uvádí dokumentace sady MINDSTORMS NXT (0 – 3) nebo lépe čitelné symbolické konstanty *SENSOR_1* až *SENSOR_4*.

Spuštění snímání hodnot senzoru se provádí funkcemi:

```
OpenSwitch(SENSOR_1);  
OpenSound(SENSOR_2, 'DB');  
OpenLight(SENSOR_3, 'ACTIVE');  
OpenUltrasonic(SENSOR_4);
```

Zvukový senzor lze samozřejmě zpřístupnit v režimech DB a DBA. Světelný senzor lze zpřístupnit pro čtení v aktivní ACTIVE a inaktivním INACTIVE režimu. Měření aktuální hodnoty senzoru a její přečtení uvádí následující příklad:

```
if GetSwitch(SENSOR_1)
end
if GetSound(SENSOR_2) < 100
end
if GetLight(SENSOR_3) > 1000
end
if GetUltrasonic(SENSOR_4) < 30
end
```

Funkce *GetSwitch* snímá, zda byl dotykový senzor sepnut (výstupními hodnotami je 0 a 1). Funkce *GetSound* snímá aktuální intenzitu zvuku v dB nebo dBA, dle nastaveného režimu (hodnoty se pohybují od 0, pro nejnižší intenzitu do 1023 pro nejvyšší intenzitu). Funkce *GetSound* vrací aktuální intenzitu světelného záření v aktivním nebo inaktivním módu (hodnoty 0 až 1023). Funkce *GetUltrasonic* vrací vzdálenost objektu od ultrazvukového senzoru v cm (hodnoty 0 až 100).

Následující zdrojový kód cyklu snímajícího počet překročení určité hranice měřené hodnoty, který plně vytěžuje strojový čas počítače

```
OpenSound(SENSOR_2, 'DB');
while(podmínka)
    if GetSound(SENSOR_2) > 500
        ClapCount = ClapCount + 1;
    end
end
```

lze ovšem nahradit jiným, využívajícím nízkourovňové funkce *NXT_SetInputMode* (vyjma ultrazvukového senzoru, který vnitřně používá zcela jiný režim), jenž umožňuje počítání těchto hodnot automaticky, uvnitř kostky NXT a přitom lze stále vykonávat jiné jednoduché funkce.

```
NXT_SetInputMode(SENSOR_2, 'SOUND_DB', 'PERIODCOUNTERMODE', 'dontreply');
NXT_ResetInputScaledValue(SENSOR_2);
if GetSound(SENSOR_2) > 500
end
```

```
% zde je možné vykonat jakoukoli funkci například pozastavení na 10s
pause(10)
```



```
data = NXT_GetInputValues (SENSOR_2);  
ClapCount = data.ScaledVal;
```

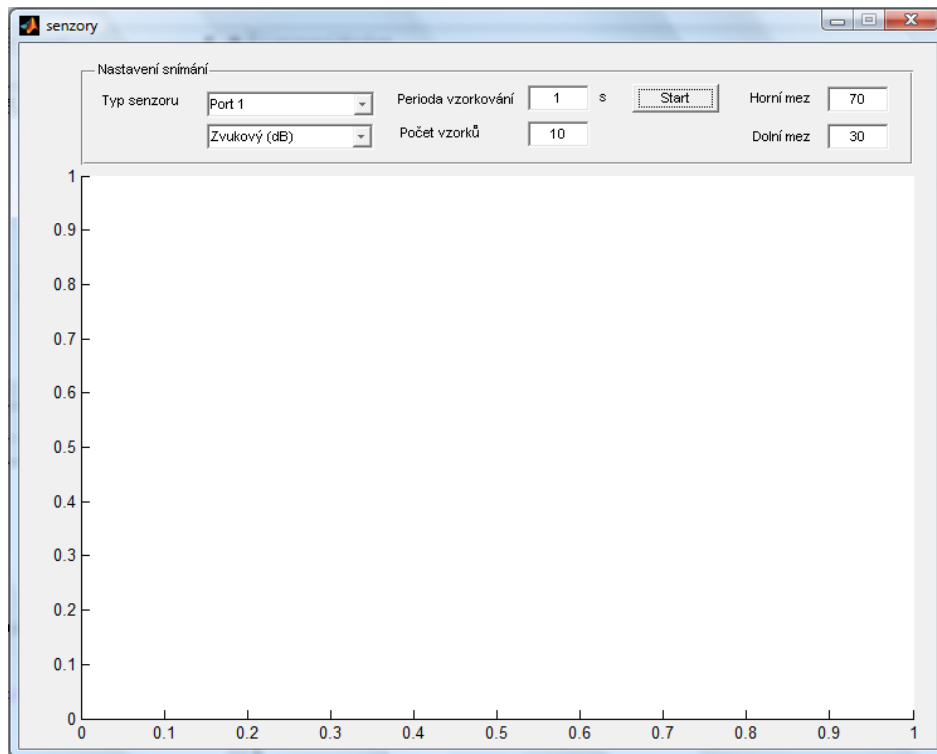
Kostka NXT spočetla překročení prahu vnitřně. Po ukončení čtení se použijí funkce pro uzavření senzoru.

```
CloseSensor (SENSOR_1);  
CloseSensor (SENSOR_2);  
CloseSensor (SENSOR_3);  
CloseSensor (SENSOR_4);
```

Postup návrhu:

Po seznámení se principy sensoriky a vlastnostmi standardních senzorů robotické stavební sady MINDSTORMS NXT a jejich funkčnímu propojení s nástrojovou sadou RWTH postupujeme obdobným způsobem jako při návrhu v laboratorní úloze č. 1. Grafické uživatelské rozhraní by mělo obsahovat prvky pro volbu portu a typu připojeného senzoru, dále, nastavení periody vzorkování a zobrazení výstupu naměřených veličin do grafu. Rozsah os grafu by se měl přizpůsobit snímané veličině.

Vzhled vzorové aplikace je na obrázku



Obr. 38 Vzorová aplikace pro snímání senzorů

Závěr:

Navržené grafické uživatelské rozhraní slouží k zobrazení hodnot snímaných na standardních senzorech sady MINDSTORMS NXT. Aplikace je rozdělena do dvou částí. První z nich, Nastavení snímání, zahrnuje ovládací prvky pro konfiguraci zvoleného portu a typu senzoru, dále pak nastavení periody vzorkování a počet odebraných vzorků a v neposlední řadě nastavení horní a dolní meze snímané veličiny. Počátek snímání se řídí spuštěním tlačítka Start. Od periody vzorkování se během zobrazení hodnot do grafů odečítá čas potřebný k vykonání těchto instrukcí, použitím interních funkcí MATLABu tic a toc, což by mělo korigovat případný rozdíl.

Druhou část rozhraní tvoří během měření periodicky aktualizovaný graf pro výstup naměřených hodnot. Měření hodnot je umožněno pro všechny typy senzorů a portů. Rozsah os grafů se automaticky přizpůsobuje typu snímané veličiny a délce vzorkování. Data získaná při měření nejsou dále nijak zpracovávána, jejich výstup je pouze do grafu. Aplikace je upotřebitelná především k ověření funkcionality jednotlivých senzorů. Doplnění další programové rutiny pro zpracování statistických údajů měření nabízí širokou škálu možností a uplatnění funkcí softwaru MATLAB.

Při měření se projeví nepříznivé vlivy rušení a ztrát paketů na přenosové lince bezdrátového spojení.

Realizace programu je součástí elektronické přílohy diplomové práce.

4.3 Úloha č. 3 – Řízení laboratorního modelu Spike

Úkol měření:

Řízení laboratorního modelu Spike

- Seznamte se s hardwarovými vlastnostmi inteligentní programovatelné kostky NXT, funkcionalitou vstupních a výstupních portů a charakteristickými vlastnostmi servomotorů a senzorů NXT. Použijte podklady laboratorních úloh 1 a 2
- Seznamte se s konceptem a vlastnostmi RWTH toolboxu matematického softwaru MATLAB pro řízení robotů Mindstorms NXT pomocí PC a jeho komunikačních možností přes rozhraní Bluetooth. Použijte podklady laboratorních úloh 1 a 2

- Navrhněte grafické uživatelské rozhraní, pokrývající ovládání a snímání všech základních parametrů modelu Spike
- Navrhněte programovou rutinu pro manuální ovládání robotu
- Navrhněte programovou rutinu pro řízení pohybu robotu z místa A do místa B. Vstupem bude relativní pozice cílového místa vůči robotu. Robot uchovává informaci o směru a vzdálenosti cílové polohy. Narazí-li robot při cestě k cíli na překážku, snaží se ji objet (například metodikou „Neustále vpravo“). Jestliže překážku není možné objet, zobrazí program informaci o nedostupnosti cílové pozice. Dynamický model robotu odhadněte experimentálním způsobem.
- Při návrhu využijte sadu systémových, případně pomocných funkcí toolboxu RWTH (například k akustické signalizaci překážky, atp.)
- Otestujte navrženou aplikaci a zhodnoťte kvalitu řízení i komponent kostky NXT

Seznam použitých přístrojů a součástek:

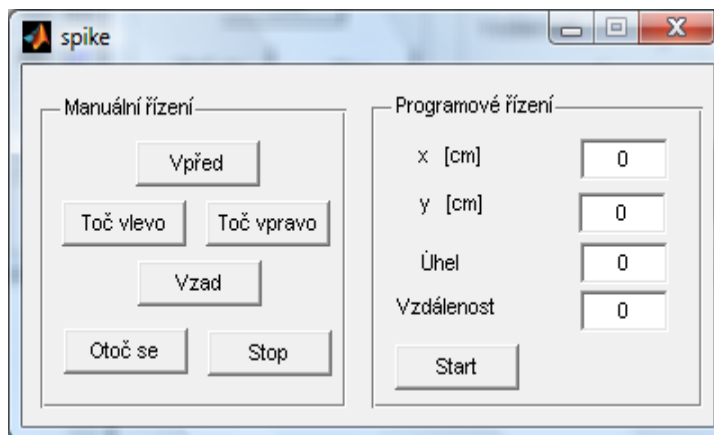
- Inteligentní programovatelná kostka NXT
- Sada servomotorů NXT
- Sada senzorů NXT
- Počítač s matematickým softwarem MATLAB 7.4 a vyšší
- Adaptér pro Bluetooth komunikaci

Postup návrhu:

Po seznámení se s hardwarovými vlastnostmi inteligentní programovatelné kostky NXT, funkcionalitou vstupních a výstupních portů a charakteristickými vlastnostmi servomotorů a senzorů NXT, a jejich funkčnímu propojení s nástrojovou sadou RWTH postupujeme obdobným způsobem jako při návrhu v laboratorních úlohách č. 1 a č. 2. Aplikace musí obsahovat prvky pro manuální řízení a programové řízení robotu. Potřebnou rutinu ovládání motorů programujeme experimentálně, na základě vlastností materiálu povrchu, po němž se robot pohybuje. Snažíme se docílit co nejvyšší přesnosti jednotlivých pohybů, protože odchylky vedou k hromadění chyby při jejich vykonávání.

Cílem je navrhnout vhodnou strategii robotu pro pohyb do cílové pozice, počítající s případnou překážkou v přímém směru. K určování překážek je vhodné použít

ultrazvukového senzoru robotu, případně kontrolovat umístění překážek vystrčením „žahadla“ modelu.



Obr. 39 Vzorová aplikace pro ovládání robotu Spike

Závěr:

Navržené grafické uživatelské rozhraní slouží k ovládání laboratorního modelu Spike, vytvořeného na bázi modulární robotické stavební sady MINDSTORSM NXT. GUI je rozděleno do dvou částí, z nichž první, označená jako panel Manuální řízení, zpřístupňuje manuální ovládání pohybu robotu a to ve směrech vpřed, vzad, otočení se o 90 stupňů vpravo, otočení se o 90 stupňů vlevo, obrat robotu o 180 stupňů a zastavení chodu všech motorů. Druhá část aplikace, panel označený jako Programové řízení, slouží uživateli k zadání souřadnic pozice, na kterou se má robot po stisknutí tlačítka Start vydat. Mimo to je zobrazen v tomto oddílu ještě úhel – směr k cíli a vzdálenost. Pro oživení robotu byla napsána programová rutina, zabezpečující jeho pohybové funkce. Úhly otáčení a rovinného pohybu byly určeny experimentálně.

Funkcionalita programu je následující: Robot porovná zadané souřadnice se svou relativní pozicí, která je stanovena jako bod $[0,0]$ souřadného systému. Provede se výpočet vzdálenosti a úhlu natočení vůči cílové pozici. Na základě těchto údajů provede robot korekci úhlu vzhledem k cílové pozici tak, aby se po jejím provedení mohl pohybovat k cíli po přímce (zúžení pohybu a prostoru do jednoho směru). Následně robot provede detekci pomocí ultrazvukového senzoru, umístěného ve své přední části. Na základě hodnot snímaných ultrazvukovým senzorem je zvolena některá z variant pohybu – pokud robotu nebrání nic v cestě, posune se o stanovený úsek směrem k cíli, pokud je před robotem zjištěna překážka, provede se „úkok stranou“, při němž robot provede pohyb

směrem vpravo o přesně stanovený úsek. Po přepočítání souřadnic a úhlu cíle se postupuje obdobným způsobem cyklicky, dokud robot nedosáhne cílové pozice. Jestliže ani určitý počet úroků nevede k možnosti dosáhnout cíle, provede robot hlášení akustickým znamením.

U manuálního i programového řízení robotu se projevily vlivy rušení bezdrátového přenosu Bluetooth, které zpomalovaly celkovou rychlost komunikace, zapříčiněné ztrátou paketů a tedy celkovou přesnost řízení. Zlepšení lze dosáhnout naprogramováním některých programových rutin přímo v jádru kostky NXT a ty posléze pouze volat funkcemi řídicího toolboxu RWTH, čímž se dosáhne odlehčení komunikačního rozhraní Bluetooth. Pro podrobnější informace je potřeba nastudovat dokumentaci grafického jazyku NXT-G nebo jiného vývojového prostředí, umožňujícího vytvářet programy vykonatelné v jádru kostky. Mimo to se projevily odchylky při určování otáček, způsobené nepřesností vnitřních enkodérů servomotorů.

Pohyb robotu závisí také na prostředí, v němž se robot pohybuje, zejména na vlastnostech materiálu podložky. Rutinu ovládající pohyb robotu je proto nezbytné se změnou prostředí upravit pro nové vlastnosti. Během řízení je potřeba dbát na to, aby baterie zůstávala ve stavu plně nabitá, kdy je zaručen 100% výkon robotu. Rutinu lze obdobným způsobem upravit pro jiné typy robotů.

Realizace programu je součástí elektronické diplomové práce.

ZÁVĚR

Ačkoli se robotická stavební sada MINDSTORMS NXT může jevit na první pohled pouze jako nákladná hračka pro mládež (což jistě byl původní ekonomický záměr společnosti LEGO), její koncept vychází především z obohacení, zefektivnění a zkvalitnění vzdělávacího procesu v oboru robotiky, informatiky a výpočetních věd. Spolupráci s univerzitou MIT v USA, která existuje již od poloviny 70. let minulého století, se podařilo vyvinout snadno pochopitelný a přitom komplexní a dobře dostupný výrobek pro návrh a modelování robotických modulárních systémů, který si vydobyl nezastupitelnou pozici v učebnách a univerzitních laboratořích po celém světě.

Jak vyplynulo ze studie charakteristických vlastností hardwarových a softwarových komponent, vychází operabilita stavebnice z užití inteligentní programovatelné kostky, mikro počítače s 32bitovým jádrem, který řídí činnost všech vnitřních okruhů i periférií, včetně vestavěného čipu bezdrátové komunikace Bluetooth. Pomocí čtyř vstupních a tří výstupních portů lze monitorovat vnitřní a vnější procesy a přizpůsobovat jim model robotu i okolního prostředí (a aktivně tak na něj působit). Výstupní porty slouží k připojení servomotorů a vstupní k připojení senzorů. Standardně obsahuje sada čtyři typy senzorů – dotykový, světelný, zvukový a ultrasonický. Díky zabudovanému standardizovanému digitálnímu komunikačnímu rozhraní sběrnice I²C lze snadno připojit řadu koncových zařízení externích, jako jsou senzory, ovladače či multiplexery rozšiřující komunikačních schopnosti například o spojení s webovou kamerou či starší programovatelnou kostkou RCX.

Oficiální vývojové programové prostředí je odvozeno z profesionálního softwaru LabVIEW společnosti National Instruments, na bázi grafického programovacího jazyku G. Je určeno především uživatelům, kteří nemají hlubší zkušenosti s programováním pomocí konvenčních lexikálních jazyků. Proto je jeho funkcionalita omezená. Během krátkého období vznikla řada fundovanějších prostředí pro různé platformy nízkoúrovňového i vysokoúrovňového programování (odvozeniny jazyků assembler, C, Java, atp.).

Z rozsáhlého výběru akceptovatelných řešení byla pro své nesporné výhody zvolena možnost ovládání robotu pomocí nástrojové sady RWTH, vyvinuté k tomuto účelu na univerzitě Rheinisch-Westfaelische Technische Hochschule v Cáchách, která pracuje jako toolbox matematického softwaru MATLAB. Toolbox zahrnuje všechny funkce podstatné

pro efektivní snímání vnitřních a vnějších vlivů působících na robota a jeho následné řízení. Komunikace probíhá prostřednictvím protokolu Bluetooth SPP, který je virtuální obdobou tradičního sériového portu. Mimo to patří mezi hlavní přednosti obrovská programová svoboda, prakticky neomezená velikost programů (oproti jiným metodám programování inteligentní kostky NXT) a v neposlední řadě rozsáhlost funkcionálního zázemí samotného MATLABu, čítající desítky až stovky nástrojů pro práci s analýzou signálů, neuronovými sítěmi, automatickým řízením, sítěmi, atd.

Navrženy jsou tři laboratorní úlohy, které nahlíží z různých aspektů na řízení modelu robotu pomocí PC. První se zaměřuje na kontrolu servomotorů, ovládání příkonu, počtu otáček, synchronizace a snímání důležitých vnitřních veličin. Druhá úloha uvádí do problematiky sensoriky, standardních senzorů pracujících s kostkou NXT a snímání jejich hodnot pomocí toolboxu RWTH. Poslední úloha sdružuje prakticky vědomosti získané v úlohách předcházejících. Slouží k manuálnímu a programovému řízení kompletního modelu robotu. V praktické části diplomové práce jsou nastíněny pouze podstatné části zadání laboratorních úloh a jejich vzorové vypracování. Samotné úlohy jsou součástí elektronické přílohy ve formátu PDF.

Laboratorní úlohy nevyčerpávají zcela potenciál využití řídicího toolboxu RWTH. Snahou je především přiblížit případnému studentovi přehlednou formou základní funkcionalitu, obecné principy praktikované v robotice, propojitelnost s hardwarovými a softwarovými prostředky stavebnice MINDSTORMS NXT a jejich aplikovatelnost na široké spektrum úloh obdobného charakteru tak, aby se po úspěšném absolvování dobře orientoval v dané problematice a svedl cíleně a účelně uvést získané poznatky do praxe.

ZÁVĚR V ANGLIČTINĚ

You can think that construction set MINDSTORMS NXT is just an expensive toy for children (what was surely first intension of LEGO company), but its concept is enrichment, more effective and qualitative education system in branch of robotic and informatics. By cooperation with MIT in USA, which exists from half of the 70th years of last century, was created easily understandable, but complex and available product for designing and modeling of robotic modular systems, which is not substitutable in university labs all around the world.

It was ensued on study of characteristic properties of hardware and software components, that operatibility of construction sets is based on usage of intelligent programmable brick, microcomputer with 32 bit core, which control all activities of inner circuits and peripheries including chip for wireless Bluetooth communication. By using of four input and three output ports you can monitor inner and outer processes and adapt the model of robot and surrounding environment (and appeal on it actively). Output ports are for connection of servomotors, input for connection of sensors. Usually you can find four types of sensors in the set – touch, light, sound and ultrasonic. Thanks to built-in standardized digital communication interface of I2C it can be easily connected a lot of external products like sensors, controllers or multiplexers extending communication skills like the connection of web camera or older programmable brick RCX.

Official development program interface is derived from professional software National Instruments product LabVIEW, on base of G the graphic programmable language. It is addressed to users, who dont have deeper experience with programming in conventional lexical languages. Thats the reason why the functionality is quite limited. There is a lot of interfaces for different types of platform of low-level and high-level programming (derivates of languages like assembler, C, Java, etc.).

From the wide range of acceptable solutions it was chosen the option to control the robot by the tool set RWTH created for this purpose on Rheinisch-Westfaelische Technische Hochschule University in Aachen, which is designed as a toolbox for mathematical software MATLAB. This toolbox includes all functions important for effective reading of inner and outer influences appealing the robot and its later control. Communication is going trough Bluetooth SPP protocol, which is virtual version of traditional serial port. By

the way its main advantage is huge program freedom, practical unlimited size of programs (in compare with other methods of programming the intelligent NXT brick) and not finally extensiveness functional base of MATLAB itself, counting tens or hundreds of tools for working with signals, neuron nets, automated control, networks etc.

There are three lab tasks designed, which think of different aspects of control of the robot by the PC. First is aimed on servo motors control, controlling the power input, number of rotations, synchronization and reading important inner values. Second one is taking place in problem of sensors, standard sensors working with NXT brick and reading its values by the RWTH toolbox. Last task includes practical knowledge gained from last two tasks. It deals with manual and program control of the complete robot model. In practical part of this master thesis you can find sketched main parts of lab tasks and its model solution. Task are included as electronic attachment in PDF format.

Lab tasks are not emptying potential of the controlling toolbox RWTH. The main endeavor is just to enclose basic functionality, common principles practiced in robotic, communication with hardware and software instruments of MINDSTORMS NXT construction set and its option of applications to wide spread of tasks with analogous character in that way to students, so they can be well oriented in this problematic and able to apply gained knowledge in the practices.

SEZNAM POUŽITÉ LITERATURY

- [1] PERDUE, David J. *THE UNOFFICIAL LEGO MINDSTORMS NXT INVENTOR'S GUIDE*. Megan Dunchak; Christina Samuell. 1st edition. San Francisco : No Starch Press, Inc, c2008. 296 s. ISBN 978-1-59327-154-1.
- [2] ASTOLFO, Dave, FERRARI, Mario, FERRARI, Giulio. *BUILDING ROBOTS WITH LEGO MINDSTORMS NXT*. Audrey Doyle. 1st edition. Burlington : Syngress Publishing, Inc, c2007. 447 s. ISBN 978-1-59749-152-5.
- [3] SCHOLZ, Matthias Paul. *Advanced NXT: The Da Vinci Inventions Book*. Jennifer Whipple. Apress, c2007. 369 s. ISBN 978-1-59059-843-6.
- [4] GASPERI, Michael, HURBAIN, Philippe, HURBAIN, Isabelle. *Extreme NXT : Extending the LEGO MINDSTORMS NXT to the Next Level*. Susannah Davidson Pfalzer. [s.l.] : Apress, c2007. 286 s. ISBN 978-1-59059-818-4.
- [5] BOOGAARTS, Martijn, et al. *THE LEGO MINDSTORMS NXT IDEA BOOK : design, invent, and built*. Nancy Sixsmith, Megan Dunchak. 1st edition. San Francisco : No Starch Press, Inc, c2007. 344 s. ISBN 978-1-59327-150-3.
- [6] BAGNALL, Brian. *Maximum LEGO NXT : Building Robots with Java Brains*. Edited by Sylvia Philipps. 1st edition. Canada : Variant Press, c2007. 505 s. ISBN 978-0-9738649-1-5.
- [7] HRUŠKA, František. *Technické prostředky automatizace II*. 2. vyd. Zlín: Univerzita Tomáše Bati, Fakulta aplikované informatiky, 2006. 112 s. ISBN 80-7318-397-8
- [8] HRUŠKA, František. *Technické prostředky automatizace III: (senzory, jejich principy a funkce)*. 2. vyd. Zlín: Univerzita Tomáše Bati, Fakulta aplikované informatiky, 2005. 118 s. ISBN 80-7318-315-3
- [9] LEGO MINDSTORMS NXT Hardware Developer Kit. Dostupný z WWW:
<[http://cache.lego.com/upload/contentTemplating/MindstormsOverview/otherfiles/2057/LEGO%20MINDSTORMS%20NXT%20Hardware%20Developer%20Kit\(37A0CF630-CCE5-4AAF-91FA-D1E7C911817C.zip\)](http://cache.lego.com/upload/contentTemplating/MindstormsOverview/otherfiles/2057/LEGO%20MINDSTORMS%20NXT%20Hardware%20Developer%20Kit(37A0CF630-CCE5-4AAF-91FA-D1E7C911817C.zip))>

[10] Displej z tekutých krystalů. Dostupný z WWW:

<<http://cs.wikipedia.org/wiki/LCD>>

[11] Labview. Dostupný z WWW:

<<http://en.wikipedia.org/wiki/Labview>>

[12] Inter-Integrated Circuit. Dostupný z WWW:

<<http://cs.wikipedia.org/wiki/I2C>>

[13] Bluetooth. Dostupný z WWW:

<<http://cs.wikipedia.org/wiki/Bluetooth>>

[14] Matlab. Dostupný z WWW:

<<http://www.humusoft.cz/matlab/matlab.htm>>

[15] RWTH toolbox. Dostupný z WWW:

<<http://www.mindstorms.rwth-aachen.de/>>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

CNC	Computer Numerical Control
GPL	General Public License
HMI	Human-machine Interface
I ² C	Inter-Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
JVM	Java Virtual Machine.
LCD	Liquid Crystal Display
LED	Light-emitting Diode
MIT	Massachusetts Institute of Technology
NI	National Instruments
PC	Personal Computer
PDA	Personal Digital Assistant
PS2	Playstation 2
PWM	Pulse-width Modulation
RCX	Robotic Control Explorer
RDS	Robotic Discovery System.
RIS	Robotic Invention System
RWTH	Rheinisch-Westfälische Technische Hochschule
SCL	Synchronous Clock
SDA	Synchronous Data
SDK	Software Developers Kit
SONAR	Sound Navigation And Ranging
USB	Universal Serial Bus
WI-FI	Wireless Fidelity

SEZNAM OBRÁZKŮ

Obr. 1 Inteligentní kostka RCX	10
Obr. 2 Inteligentní kostka NXT	12
Obr. 3 Schéma zapojení komponent kostky NXT	13
Obr. 4 Výstupní port	14
Obr. 5 Vstupní port	15
Obr. 6 Souřadnicový systém LCD displeje.....	17
Obr. 7 Vnitřní převodování motoru NXT	20
Obr. 8 Útroby enkodéru motoru NXT	21
Obr. 9 Časový průběh snímání na senzorech.....	23
Obr. 10 Dotykový senzor	24
Obr. 11 Světelný senzor	25
Obr. 12 Zvukový senzor.....	26
Obr. 13 Ultrazvukový senzor	28
Obr. 14 Webová kamera	29
Obr. 15 Sensorový multiplexor RCX	29
Obr. 16 Akcelerační senzor.....	30
Obr. 17 Multiplexor NXT motorů	30
Obr. 18 Rozhraní RCX - NXT	31
Obr. 19 Rozhraní PS2	31
Obr. 20 Detektor překážek.....	32
Obr. 21 Magnetický kompas	32
Obr. 22 Pneumatický senzor	32
Obr. 23 Obvod reálného času.....	33
Obr. 24 Infračervený senzor.....	33
Obr. 25 Hierarchie menu firmwaru NXT.....	34
Obr. 26 Ukázka pracovní plochy vývojového prostředí LabVIEW	37
Obr. 27 Ukázka pracovní plochy vývojového prostředí NXT-G.....	39
Obr. 28 Laboratorní model robotu Spike	44
Obr. 29 Ukázka pracovní plochy softwaru MATLAB.....	46
Obr. 30 Kostka NXT.....	56
Obr. 31 Menu Settings	57

Obr. 32 Menu Bluetooth	58
Obr. 33 Průvodce přidáním zařízení Bluetooth	58
Obr. 34 Připojování k zařízení Bluetooth	59
Obr. 35 Potvrzení klíče	59
Obr. 36 Vlastnosti zařízení Bluetooth.....	60
Obr. 37 Vzorová aplikace pro řízení motorů	61
Obr. 38 Vzorová aplikace pro snímání senzorů.....	65
Obr. 39 Vzorová aplikace pro ovládání robotu Spike.....	68

SEZNAM TABULEK

Tab. 1 Vztah vnímání intenzity zvuku člověkem	27
Tab. 2 Kompatibilita Bluetooth adaptérů.....	48

SEZNAM PŘÍLOH

P I Elektronické přílohy a verze bakalářské práce na CD