

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ

Katedra řídicí techniky



Využití robota LEGO MINDSTORMS při výuce

Bakalářská práce

Praha 2009

Student:

Pavel Trojánek

Vedoucí práce:

Ing. Martin Hlinovský, Ph.D.

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne 5.května 2009

.....

podpis

Poděkování

Tímto bych chtěl poděkovat Ing. Martinu Hlinovskému Ph.D., vedoucímu bakalářské práce za výbornou spolupráci a podporu. Panu Aleši Kapicovi děkuji za pomoc při práci na webových stránkách! Dále děkuji všem, se kterými jsem mohl práci konzultovat a kteří tak přispěli k její vyšší kvalitě.

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra řídicí techniky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Pavel Trojánek**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Kybernetika a měření

Název tématu: **Využití robota Lego Mindstorms ve výuce**

Pokyny pro vypracování:


1. Seznamte se s možnostmi robota Lego Mindstorms (současný stav, HW vybavení).
2. Nastudujte možnosti programování robota Lego Mindstorms a připravte návod pro instalaci základních typů programovacích prostředí.
3. Proveďte návrhy tří soutěžních úloh s řízením ve třech různých programovacích prostředích včetně dokumentace s konstrukcí robota a způsobu jeho naprogramování.
4. Vytvořte webové stránky k výuce předmětu "ROBOTI".

Seznam odborné literatury:

Dodá vedoucí práce

Vedoucí: Ing. Martin Hlinovský, Ph.D.

Platnost zadání: do konce zimního semestru 2009/10


prof. Ing. Michael Šebek, DrSc.
vedoucí katedry




doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 27. 2. 2009

Abstrakt

Tato práce pojednává o využití Robotu LEGO MINDSTORMS při výuce. Věnuje se současnému stavu využívání stavebnice, rozebírá její hardwarové specifikace a konečně se věnuje jejím výukovým možnostem. Součástí práce je také popis a návod na instalaci tří programovacích prostředí a ukázka tří úloh, které může robot plnit. Poslední část práce se zabývá webovými stránkami pro předmět Roboti.

Abstract

The bachelor thesis analyses the options of working with Robot LEGO MINDSTORMS as with an educational tool. At the first, it describes the present state of the robot usage in general. Then it concerns of hardware specifications and finally it describes the possible practical usage of the robot, meaning the instructions for installing three kinds of development environments. There is also shown an illustration of three challenges that can be programmed by students. The last part of the thesis covers the documentation of web pages for the class “Robots”.

Obsah

1. Úvod do bakalářské práce	13
1.1. Cíl práce	13
1.2. Obsah práce	13
2. Seznámení s Robotem LEGO MINDSTORMS	14
2.1. Úvod do druhé kapitoly	14
2.2. Robot LEGO MINDSTORMS	15
2.3. Historie	15
2.4. Současná situace ve světě	16
2.5. Osobní zkušenost z účasti na soutěži „Roborace“	17
2.6. Využití robota při pořádání „Dne otevřených dveří“	18
2.7. Využití robota pro ROBO SOUTĚŽ 2009	19
Příprava a průběh soutěže	19
Technická stránka	20
Zhodnocení soutěže	20
2.8. Využití robota pro výuku předmětu Roboti	22
2.9. Hardwarové vybavení robota	24
Přehled HW vybavení robota	24
NXT technické rozhraní	24
Hlavní procesor Atmel ARM7	25
Komunikace uživatele s NXT kostkou	25
Napájení	26
Výstupní porty	26
Vstupní porty	28
Propojovací kabely	30
Senzory	30
Aktivní Senzory	31
Pasivní senzory	32
Nadstandardní nabídka senzorů	34
I ² C komunikace	36
Display	37
Bluetooth	38
Zvuk	40
Motory	40
Závěr ke kapitole 2	41
3. Možnosti programování robota	42
3.1. Úvod do třetí kapitoly	42
3.2. Přehled možností programování robota	42
3.3. Návod pro instalaci základních typů prostředí	43
NXT-G	43
Úvod	43
Instalace prostředí	44
Zdroje a návody	44
NXC	44
Úvod	44
Návod na instalaci programovacího prostředí	45
LeJOS NXJ	45
Úvod	45
Návod na instalaci programovacího prostředí	47
4. Návrh soutěžních úloh	49
4.1. Úvod do čtvrté kapitoly	49
4.2. Obsah čtvrté kapitoly	49
4.3. Úloha číslo 1: „Bludiště“	49
Pravidla pro úlohu „Bludiště“	49

Konstrukční řešení robota	52
Konstrukční řešení robotů z Robo soutěže 2009	53
Programátorské řešení úlohy	54
Vývojový diagram programu	55
Funkce a konstanty poskytnuté NXT API	56
Samotné tělo programu	56
Závěr k návrhu úlohy „Bludiště“	58
4.4. Úloha číslo 2.: „Překážky“	59
Pravidla pro úlohu „Překážky“	59
Konstrukční řešení robota	61
Programovací řešení robota	62
Závěr k úloze „Překážky“	66
4.5. Úloha číslo 3: „Čára“	67
Pravidla	67
Soutěžní plán	68
Konstrukční řešení	70
Programátorské řešení	70
Vývojový diagram pro úlohu „Čára“	70
Samotné programátorské řešení	71
Závěr k úloze „Čára“	74
4.6. Závěr k oddílu zabývajícím se návrhem soutěžních úloh	74
5. Webové stránky k výuce předmětu ROBOTI	75
5.1. Obsah páté kapitoly	75
5.2. Cíl webových stránek	75
5.3. Základní údaje	76
Softwarové nároky na uživatele	76
Použité technologie	76
Základní části zobrazení	76
A. Hlavička	77
B. Menu	77
C. Hlavní okno	77
D. Patička	77
5.4. Obsah stránek	78
Aktuálně	78
Organizace	79
Programovací prostředky	79
Odkazy	79
ROBO SOUTĚŽ 2009	79
5.5. Struktura webu	79
5.6. Obsah jednotlivých adresářů a funkčních souborů	80
5.7. Styly	84
Formátování rozložení stránky	84
Formátování textu, tabulek, seznamů	85
5.8. Závěr k páté kapitole	85
6. Závěr	86
SEZNAM POUŽITÝCH ZKRATEK	87
SEZNAM POUŽITÝCH ZDROJŮ	87
SEZNAM OBRÁZKŮ	93
SEZNAM TABULEK	94
OBSAH ELEKTRONICKÉ PŘÍLOHY	94

1. ÚVOD DO BAKALÁŘSKÉ PRÁCE

1.1. Cíl práce

Cílem bakalářské práce je prozkoumat problematiku Robota LEGO MINDSTORMS jako výukového prostředku, seznámit se se současným stavem využití robota, nastínit jeho hardwarové specifikace a zjistit jeho softwarové možnosti co se týče jeho programování. Dále je cílem navrhnout tři soutěžní úlohy pro předmět Roboti na Katedře řízení, Fakultě Elektrotechnické, Českého vysokého učení technického v Praze. Posledním posláním práce je vytvoření webových stránek pro již zmíněný předmět.

Toto téma bakalářské práce jsem si vybral, protože se domnívám, že stavebnice je velmi zajímavý výukový prostředek. Věřím, že díky této práci se zlepší mé programovací schopnosti, že lépe porozumím robotice a že pomohu k propagaci zajímavých a perspektivních programů na ČVUT.

1.2. Obsah práce

Nepočítáme-li úvod a závěr, je práce rozčleněna do čtyř samostatných kapitol. Kapitola 2 seznamuje s danou problematikou a rozebírá hardwarové možnosti robota. V kapitole 3 pak nalezneme popis softwaru robota a možnosti jeho programování. Kapitola 4 se věnuje návrhu soutěžních úloh pro předmět Roboti. V kapitole 5 čtenář nalezne dokumentaci k webovým stránkám vytvořeným pro předmět Roboti. Přidanou hodnotou práce je bohatý seznam zdrojů, ze kterých bylo pro práci čerpáno a ze kterých mohou čerpat i zájemci o danou problematiku.

2. SEZNÁMENÍ S ROBOTEM LEGO MINDSTORMS

2.1. Úvod do druhé kapitoly

MINDSTORMS je název produktové řady prodávané společností Lego. Společně s dalšími řadami produktů jako je například LEGO SYSTEM a LEGO TECHNIK patří do řady produktů odstupňovaných dle věku vhodného pro práci s daným produktem [19].

Robot LEGO MINDSTORMS je na jednu stranu "hračka" vhodná pro děti od dvanácti let. Na druhou stranu je to ale také pečlivě vyvinutý programovací nástroj celosvětově používaný školními a univerzitními pracovišti (viz kapitola 2.4), soutěžními týmy, ale i individuálními programátory se zájmem o robotiku a řízení.

Robot představuje variabilní systém umožňující použití pro nesčetné množství úloh v různých programovacích jazycích (viz kapitola 3). Programátor má díky rozmanitému výběru stavebních dílů možnost využít zcela libovolné konstrukční řešení a navrhnout tak robota využívajícího odpovídající senzory a servo motory, které set LEGO MINDSTORMS nabízí. Dalo by se říci, že snad jedině, čím je člověk limitován, jsou jeho vlastní fantazie.

Díky stavebnici LEGO MINDSTORMS se LEGO společně s předními vědeckými institucemi podílí na účinném vzdělávání mladé generace a studenti na Českém vysokém učení technickém díky ní mohou proniknout do základů robotiky a moderního řízení.

O stavebnici bylo řečeno:

„LEGO MINDSTORMS NXT je důkazem zdařilé snahy LEGO zpřístupňovat dětem nejmodernější poznatky vědy a techniky. Společně s předními vědeckými institucemi se tak podílí na účinném vzdělávání mladé generace.“ [36]

„Je málo dětských hraček, ze kterých jsou tatínci nadšenější než děti. Je málo výukových pomůcek, ze kterých jsou učitelé nadšenější než žáci. Je málo kreativnějších a zábavnějších produktů, než tato skvělá stavebnice. Seznamte se s LEGO MINDSTORMS NXT.“ [37]

„Nemáte nějakého předchystaného robota, kterého jen oživujete. Sami si musíte vymyslet, jak daný cíl co nejlépe dosáhnout kombinací vhodně pospojovaných dílů a vhodného programování. Neladíte tak jen program, ale upravujete i kostky a rozmístění čidel. Celé je to

přítom velmi názorné a jednoduché. Takže se můžete soustředit skutečně na samotný problém, ne na zbytečné překážky.“[38]

2.2. Robot LEGO MINDSTORMS

Pro čtenáře, který se s tímto produktem setkává poprvé, je třeba robota před jeho bližším popisem stručně představit. Samotná stavebnice nabízí možnost nesčetného množství konstrukčních řešení. Pro představu velikosti robota uvádím na Obr. 2.1 jeho rozměry. Nákup dílů stavebnice je v České Republice možný u firmy Eduxe [21], která nabízí kromě základní sady senzorů a motorů i další možnosti rozšíření.



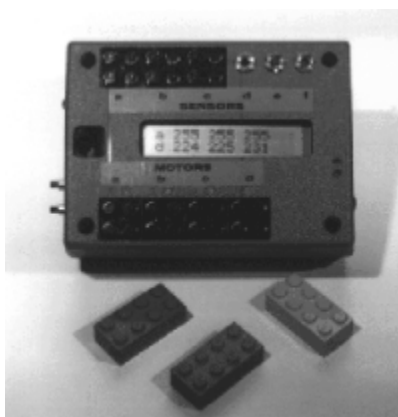
Obr. 2.1: Rozměry robota

2.3. Historie

Když byla v roce 1930 v Dánsku založena společnost LEGO, asi málokdo tušil, kam až se v roce 2009 dostane. Že bude ovšem dnes stejně jako ve třicátých letech neméně úspěšně prodávat hračky dětem po celém světě a rozvíjet jejich možnosti fantazie, můžeme rozhodně soudit z motta, které do firmy vložil její zakladatel Ole Kirk Christiansen: *"Pouze to nejlepší je dostatečně dobré"*.

Podívejme se blíže na některé časové mezníky:

- Rok 1980: Ve společnosti LEGO založena sekce "Vzdělávacích produktů".
- Rok 1986: V prodeji první LEGO produkt, který je vládán počítačem.
- Rok 1988: MIT a LEGO navazují spolupráci za účelem vyvinutí "inteligentní kostky", která LEGO ožíví skrze naprogramování uživatelem
- Podzim 1998: LEGO vypouští do prodeje řadu MINDSTORMS (kostka RCX , jazyk NQC)
- Rok 2006: Nová řada stavebnice Lego Mindstorms zvaná Mindstorms NXT je uvedena do prodeje
- Rok 2008: 10 let po vypuštění první stavebnice z řady MINDSTORMS do prodeje se LEGO chystá vydat novou verzi Mindstorms NXT 2.0., která nabídne větší kapacitu paměti, nové senzory, nový grafický programovací jazyk, rychlejší Bluetooth komunikaci nebo zabudovaný mp3 přehrávač [24].



Obr. 2.2 programovatelná NXT kostka vyvinutá na MIT

2.4. Současná situace ve světě

Stavebnice LEGO MINDSTORMS je v současné době používána vzdělávacími institucemi téměř po celém světě k vyučování základů robotiky a programování.

Už od mladého věku tak mohou být studenti pomocí stavebnice podporováni v týmové práci. Zadávané projekty s otevřeným koncem podporují u studentů fantazii a kreativitu. Týmová práce na projektech je většinou vedena zkušenějšími studenty, kteří mladším poskytují cenné rady a pomáhají jim rozvinout jejich schopnosti ve vedení týmu.

Práce s robotem též umožňuje studenty uvést do problematiky robotiky v praxi a nabídnout jim exkurze do výzkumných pracovišť.

Mezi takováto pracoviště například patří:

- Union College, New York, USA [67]
- University of Arizona [68]
- Massachusetts Institute of Technology [69]
- RWTH Aachen [70]
- ETH Zurich [71]
- Univerzita Stuttgart [72]
- a samozřejmě také České Vysoké Učení Technické v Praze. [73]

Kromě školních učeben se s robotem můžeme setkat i na soutěžích, kde jednotliví studenti či studentské týmy s roboty plní úkoly a zdokonalují se tak ve studiu robotiky, kybernetiky a programování.

Příkladem mohou být:

- Soutěž Kyber Robot na Technické univerzitě v Liberci [74]
- Robofest v Michiganu [75]
- Soutěž First lego league v Holandsku [76]
- Roborace: soutěž středoškolských a vysokoškolských týmů na univerzitě ve Stuttgartu [72]

2.5. Osobní zkušenost z účasti na soutěži „Roborace“

Protože osobní zkušenost s novým výukovým prostředkem je nenahraditelná, proběhla na podzim roku 2008 ve Stuttgartu osobní návštěva 4 členů z řad studentů a pedagogů Katedry řídicí techniky FEL ČVUT. Návštěva na soutěži Roborace přinesla bohaté zkušenosti, zejména v oblasti nákupu, péče o stavebnice, ale i organizace celé soutěže.

Na Institutu teorie systémů a automatického řízení běží soutěž už osm let a za tuto dobu výrazně stoupl zájem studentů o obor řízení. Do soutěže jsou rovněž přizvány střední školy a tři nejlepší týmy z kategorie středních škol postupují do finále i v kategorii vysokoškolských studentů.

Příprava soutěže začíná už v dubnu, zatímco samotná soutěž se koná až v listopadu. Během této doby probíhá návrh soutěžní úlohy a doladování pravidel. Soutěžní týmy po čtyřech až šesti členech mají cca jeden měsíc na to, aby svého robota libovolné konstrukce připravili pro soutěž.

Samotná soutěž poté probíhá tři hodiny ve dvou kolech, kdy mají týmy možnost po prvním neúspěšném pokusu (pakliže je neúspěšný) robota přestavět či přeprogramovat. Celá soutěž se nese ve velmi soupeřivém duchu a přítomni jsou jak sponzoři, tak i fanoušci z řad studentů a přátel soutěžních týmů.

Jak již bylo řečeno, celá soutěž má pozitivní dopad na rozvoj týmové práce, základů řízení, robotiky a programování [72].



Obr. 2.3: Soutěž Roborace na Universitě ve Stuttgartu

2.6. Využití robota při pořádání „Dne otevřených dveří“

Katedra řízení, na Fakultě Elektrotechnické, Českého Vysokého Učení Technického při pořádání dvou dnů otevřených dveří použila robota pro propagaci nejen nového studijního oboru Kybernetika a robotika, ale také pro propagaci práce celé katedry. Předvedený robot měl velký úspěch, což dokazovali přichozí studenti, kteří se na místo po skončení oficiálního programu vraceli s individuálními dotazy.



Obr. 2.4: Lego Robot při dni otevřených dveří

2.7. Využití robota pro ROBO SOUTĚŽ 2009

Jak již bylo řečeno, robot LEGO MINDSTORMS je výborným výukovým prostředkem a taktéž dobře slouží jako prostředek pro robotické soutěže. Dne 7. 5. 2009 proběhla na Katedře řízení, FEL ČVUT v Praze ROBO SOUTĚŽ 2009. Soutěže se zúčastnilo tříčlenných 17 týmů. Úkolem družstev bylo splnit úlohu „Bludiště“ (viz kapitola 4.3).

Příprava a průběh soutěže

Jednotlivá družstva měla od vyhlášení soutěže čas 14 dní na přihlášení. Tuto možnost dostali studenti z programu Elektrotechnika a Informatika, oboru Kybernetika a měření a Softwarové technologie a management na Fakultě Elektrotechnické, ČVUT v Praze.

Po přihlášení do soutěže měla družstva tři týdny čas se na splnění úlohy připravit. Tento čas byl potřeba zejména na seznámení se s robotem, návrh a realizaci konstrukčního řešení a naprogramování robota.

Soutěžní den byl pak velmi podobný konceptu Roborace (kapitola 2.5). Soutěžící měli 90 minut na to, aby svého robota na soutěžním bludišti vyzkoušeli a doladily chyby ať už v konstrukci, či v softwarovém řešení. Poté následovala dvě soutěžní kola. Každý z týmů měl jeden pokus na to, aby se svým robotem projel bludištěm. Úspěšnost družstev byla přibližně

60%. Jako výsledek se počítal lepší z časů z obou soutěžních kol. Mezi oběma koly byla 15 minutová přestávka, při které měli soutěžící možnost robota doladit. Nesměli však využít soutěžní bludiště.

Z 10 týmů, které s robotem dokázali bludiště projet, získaly tři nejrychlejší dle pořadí hodnotné ceny.

Technická stránka

Organizační a technická příprava pro soutěž byla poměrně náročná a trvala přibližně dva měsíce. Během této doby proběhlo:

- Zajištění finančních prostředků pro soutěž (stavebnice, ceny, výroba soutěžního pole – bludiště, propagace soutěže)
- Výroba a úpravy soutěžního plánu – bludiště
- Testování úlohy a její splnitelnost
- Propagace soutěže na ČVUT
- Komunikace s medií
- Příprava audiovizuální techniky pro soutěžní den (snímání hřiště kamerami, ozvučení)
- Softwarová podpora (vytvoření webových stránek, nahrávání výsledků na web)
- ... a další pomocné práce.

Zhodnocení soutěže

Dle průzkumu mezi soutěžícími byla soutěž přijata s nadšením. Důvodem přihlášení jednotlivých týmů do soutěže nebyly jen hodnotné ceny, ale hlavně to, že „škola přišla s něčím novým a zajímavým“. Studenty vesměs lákalo vyzkoušet si práci s robotem, tedy s něčím s čím se ještě nesetkali. Velká část týmů by se soutěže ráda zúčastnila příští rok znovu.

Soutěž splnila očekávání organizátorů. Podařilo se připravit generálku pro předmět Roboti – vyzkoušet, jak se bude studentům s roboty pracovat, organizaci vypůjčování stavebnic a celkově vyzkoušet, zda je stavebnice vhodným výukovým prostředkem.

Soutěž ukázala, že roboti jsou prostředkem využitelným pro prezentaci školy a katedry jako takové. Během celé soutěže vládla výborná atmosféra podporovaná diváky a

zúčastněnými. Soutěžní týmy si vyráběly svá vlastní trička s motivy soutěže a k soutěži přistupovali velmi zodpovědně.

Studenti při přípravě na soutěž aplikovali znalostí z jiných předmětů a využívali programovacího jazyka, na který byli zvyklí.

Vzhledem k tomu, že se soutěž konala prvním rokem, je jistě mnoho věcí, které lze do příštího ročníku zlepšit. Rozhodně stojí za to pozvat více medií a přizvat více diváků, protože je škoda, že se na tak skvělou událost nepřišlo podívat více zájemců. Čas přípravy na soutěž byl přiměřený, nicméně soutěžící by uvítali její posun blíže k začátku semestru, aby se soutěž nekonala před zápočtovými termíny. Počet týmů byl rovněž zvolen přiměřeně organizačním možností. Obě soutěžní kola se vešla i s přestávkou do dvou hodin.

V letošním roce se zúčastnil pouze jeden student ze střední školy. Na příští ročník se nabízí možnost buď pozvat více týmů ze středních škol, nebo udělat speciální kolo pro střední školy a nejlepší středoškolské týmy pozvat do vysokoškolské soutěže.

Myslím, že se ČVUT podařilo zorganizovat něco, co tu ještě nebylo. Myslím, že soutěž otevřela brány pro nové studijní programy otevřené od září 2009 a přispěla tak k prezentaci nových výukových metod.



Obr. 2.5: Plakát pro ROBO SOUTĚŽ 2009

2.8. Využití robota pro výuku předmětu Roboti

Robot LEGO MINDSTORMS je od počátku vyráběn a vyvíjen jako výukový nástroj a motivační prostředek pro studenty, kteří aplikují své znalosti v praxi a okamžitě vidí výsledky své práce. Studenti, kteří se s robotem seznámí a budou s ním plnit zadané úlohy, mají možnost proniknout hlouběji do oborů elektrotechniky a informatiky.

Programování

K práci s robotem studenti potřebují mimo jiné základy programování. Robot tak může být dobrým prostředkem pro výuku různých programovacích prostředků. Počínaje základy v grafickém prostředí a konče objektově orientovaným přístupem a více vláknovými aplikacemi v Javě. Programováním si studenti osvojí analyzování problémů, návrh algoritmů využívajících nejrůznější struktury, procedury, šablony a dostanou se tak do kontaktu se zpracováním dat a abstraktním myšlením, které je pro programování nezbytné.

Senzory

Senzory robota nabízí k otevření taková témata, jako je kalibrace senzorů, zpracování signálů a chyby měření, stejně tak jako analogové a digitální zpracování dat. Užitečné informace z elektrických obvodů či elektroniky mohou přijít vhod tomu, kdo by se blíže zajímal o funkci senzorů či hardware inteligentní kostky.

Systémy a modely

Robot LEGO MINDSTORMS není vybaven prostředky pro modelaci složitých systémů, přesto může plnit úlohy přenesené z reálného světa, které plní složitější zařízení, a dát tak uživateli šanci do této problematiky proniknout.

Řízení

Návrh systémů či zařízení splňující očekávané požadavky na jejich chování. I tak lze přistoupit k práci s robotem. Ať už robot přemísťuje objekt, prohledává prostor, či zajišťuje stabilitu inverzního kyvadla, vždy je třeba navrhnout řídicí algoritmus, který úlohu robotovi pomůže vyřešit. K tomu je vždy bezpodmínečně nutné využít zpětné vazby. Příkladem mohou být senzory umožňující robotovi reagovat na podněty z okolí, či PID regulace při řízení motorů, kdy se za pomoci rotačních senzorů zabudovaných v servomotorech udržuje žádaná konstantní rychlost, což je žádané zejména při různém zatížení robota, či při pohybu po různě

nakloněné podložce. PID regulace se taktéž využívá při bezchybném zatáčení, kdy se zjišťuje přesný úhel otočení motoru.

Aplikovaná robotika

Díky Robotu LEGO MINDSTORMS mají studenti šanci proniknout do základů rozhodování a plánování, zpracovávat chování robota a využívat prvky umělé inteligence. Mohou ovládat robota v reálném čase, zapojit paralelní zpracování náročných úloh, či nechat komunikovat a spolupracovat více robotů najednou.

Příkladem složitých úloh dále rozložitelných do subsystémů mohou být:

- Navigační systémy: Například senzor sdělí robotovi, kde se nachází. Naprogramovaná řídicí jednotka řekne robotovi, jak má zpracovat tuto informaci a motory se rozpohybují tak, aby robot dosáhl požadovaného výsledku.
- Plánovací systémy: Robot je schopný plánovat pohyb dopředu a naopak své plánování měnit na základě čtení údajů ze senzorů
- Snímací systémy: Ať už elektrické či mechanické snímání dat vede ke zpracování údajů, které mění rozhodování robota, a ten na základě těchto dat musí dále rozhodovat.

Fyzika, Matematika

Kinematika, přeměny energie, působení sil na těleso. I s těmito základními poznatky z oblasti fyziky se robot potýká při plnění úloh. Člověk, který se hlouběji o robota zajímá, určitě ocení znalosti z oblasti elektrických motorů, akustiky a optiky. Matematické znalosti přijdou vhod při přepočtech jednotek, výpočtu prahových hodnot senzorů a využívání geometrie při výpočtech úhlů a vztahů potřebných pro výpočet přesného pohybu.

Týmová práce, inženýrský přístup

V neposlední řadě tento výukový nástroj rozvíjí tvůrčí přístup k problému a hledání originálního řešení, mnohdy za pomoci experimentů. Vzájemná spolupráce studentů v týmech vede při práci na společných projektech s otevřeným koncem k rozvoji komunikace a rozvíjení nových pracovních postupů.

Výborným příkladem k využití robota jako výukového prostředku je například [34].

2.9. Hardwarové vybavení robota

Přehled HW vybavení robota

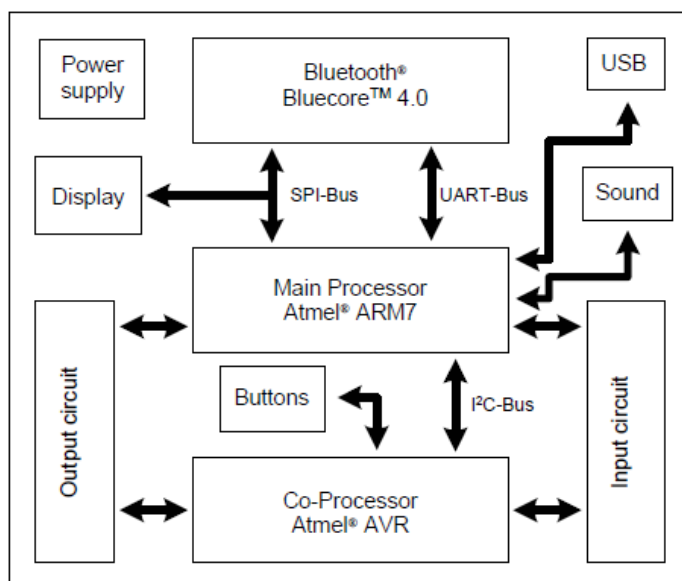
- Inteligentní NXT kostka s 32-bitovým mikroprocesorem a 256kB FLASH paměti
- 3 interaktivní servo motory s rotačními senzory k přesnějšímu řízení
- 4 senzory (zvukový, dotykový, světelný a ultrazvukový)
- 4 vstupní porty, 3 výstupní a 7 šesti žilových kabelů
- Maticový LCD display
- Zabudovaný reproduktor
- Podpora USB 2.0 a Bluetooth komunikace.



Obr. 2.6 NXT inteligentní kostka

NXT technické rozhraní

Řídící jednotkou robota je kostka NXT. Zobrazení spojení a řízení jednotlivých součástí kostky NXT je uvedeno na Obr. 2.7: (zobrazeny jsou pouze jednotky vyšších úrovní)



Obr. 2.7: Blokový diagram hardwaru kostky NXT

Hlavní procesor Atmel ARM7

Procesor je 32-bitový, běžící na 48MHz s 256ti kilobyty flashové paměti a 64mi kilobyty RAM. Do této paměti je možné nahrát až 64 souborů, které i přes odpojení NXT kostky od zdroje zůstanou v paměti uložené až do momentu příštího využití.



Obr. 2.8: procesor Atmel ARM7

Komunikace uživatele s NXT kostkou

NXT podporuje jak bezdrátovou komunikaci pomocí technologie Bluetooth rychlostí 460.8 Kbit/s , tak i rychlostí 12 Mbit/s pomocí USB 2.0 komunikace.

Napájení

Napájení kostky je možné buď šesti AA bateriemi, které ovšem neposkytují dlouhodobý pracovní komfort, nebo nabíjecím lithium-ion akumulátorem, který se vloží na místo AA baterií.



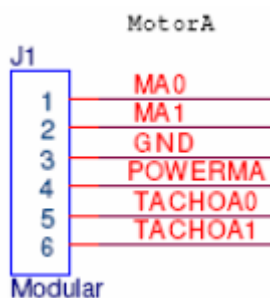
Obr. 2.9: Dobíjecí akumulátor

Výstupní porty

Kostka NXT má na vrchu tři výstupní porty, označené A, B, C, používané k řízení motorů připojených k řídicí jednotce.

K tomuto propojení slouží šesti žilový kabel, který přenáší informace k výstupnímu zařízení a z něj v případě potřeby zpět do řídicí jednotky

Na obrázku je schéma pro výstup A. Ostatní výstupy jsou obdobné.



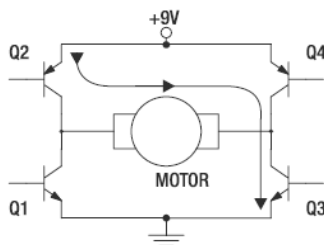
Obr. 2.10: Ukázka zapojení jednotlivých výstupních pinů

PIN1 (MA0):	PWM výstupní signál pro aktuátory
PIN2 (MA1):	PWM výstupní signál pro aktuátory
PIN3:	uzemnění
PIN4:	4.3 V
PIN5:	vstup implementující funkci Schmittova klopného obvodu.
PIN6:	vstup implementující funkci Schmittova klopného obvodu.

Tab. 2.1: Funkce jednotlivých výstupních pinů

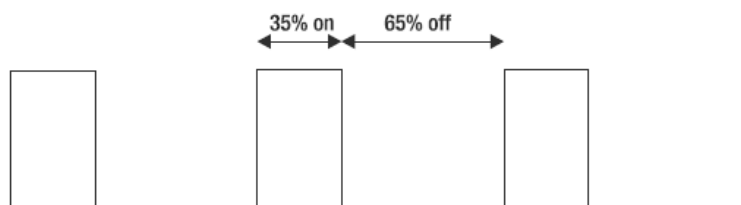
Popis jednotlivých pinů výstupních portů

- MA0 a MA1 jsou výstupní signály sloužící k řízení akčních členů. Tyto signály jsou řízeny vnitřním ovladačem motoru, který může dodávat až 700mA stejnosměrně každému výstupnímu portu a na špičce až 1A. Výstupní signál je signál PWM. Ovladač motoru má zabudovanou ochranu proti přehřátí. Pokud je tedy příliš mnoho energie nepřetržitě čerpáno z kostky, ovladač motoru automaticky přizpůsobí výstupní proud. Motory jsou řízeny obvodem zvaným H-můstek, složeným z tranzistorů. (viz Obr. 2.11.)



Obr. 2.11: Dopředný chod servo motoru

Rychlost motoru je ovládána pulzní šířkovou modulací (PWM). Tento signál řídí průměrné napětí dodávané do motorů, jak je ukázáno na obrázku pro 35% úroveň dodávané energie.



Obr. 2.12: Řízení napětí dodávaného do motorů pomocí PWM

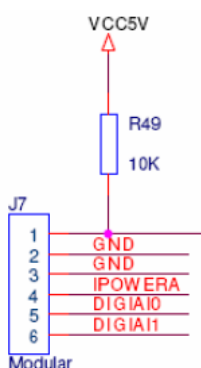
Při standardním firmwaru délka jednoho cyklu je 128 μ s, což odpovídá frekvenci 7800Hz

- Výstupní napájení (POWERMA) je propojeno se všemi vstupními a výstupními porty. Maximální výstupní proud čerpaný ze zdroje může být 180mA, to znamená, že každý port dostává přibližně 20mA. Jestliže je čerpáno více proudu, celkový výstupní proud se sníží. Pokud bude výstup zkratován, kostka bude resetována.
- Signály TACHOA0 a TACHOA1 jsou vstupní porty se zabudovaným Schmittovým klopným obvodem mezi porty a vstupními piny procesoru ARM7. Tyto signály umožňují využití fázového čidla, díky kterému je v rámci standardního firmwaru možno zpracovat tachy pulsy a detekovat směr otáčení a rychlost motoru.

Vstupní porty

Kostka NXT má čtyři vstupní porty dovolující měřit různé parametry z reálného světa. Tyto parametry závisí na typu připojeného senzoru. Rozhraní kostky, které je se senzory propojeno 6ti žilovým kabelem, umožňuje zpracování jak analogového, tak číslicového signálu ze senzoru.

Obrázek níže ukazuje schematický náčrt jednoho ze vstupů kostky.



Obr. 2.13: Ukázka zapojení jednotlivých vstupních pinů

PIN1:	Analogový vstup a možný výstup proudu.
PIN2:	uzemnění
PIN3:	uzemnění
PIN4:	4.3 V

PIN5:	číslicový I/O pin propojený s procesorem ARM7 umožňující vysokorychlostní komunikaci
PIN6:	číslicový I/O pin propojený s procesorem ARM7 umožňující vysokorychlostní komunikaci

Tab. 2.2: Funkce jednotlivých vstupních pinů

Popis pinů vstupních portů:

- Analogový vstupní pin je připojen k 10-bitovému A/D převodníku a taktěž ke zdroji proudu, který dodává energii sensorům. A/D vstupní signály jsou vzorkovány se stejnou vzorkovací frekvencí pro všechny typy sensorů a to 333Hz.
- Druhý a třetí pin jsou důležité z hlediska měření veličin senzory. Všechny signály jsou měřeny ve vztahení buď k jednomu z pinů, nebo k oběma.
- Výstupní napájení (POWER) je propojeno se všemi vstupními a výstupními porty a je hlavním zdrojem energie pro všechny senzory. Maximální výstupní proud čerpaný ze zdroje může být 180mA, to znamená, že každý port dostává přibližně 20mA. Jestliže je čerpáno více proudu, celkový výstupní proud se sníží. Pokud bude výstup zkratován, kostka bude resetována.

Tabulka níže ukazuje odběr proudu jednotlivými senzory:

Device	Measured
Touch	0
Sound	1.7 mA
Light Sensor (light off)	2.6 to 3 mA (depends on light)
Light Sensor (light on)	16.3 mA
Ultrasonic Sensor	4 mA
Motor position encoders	9 to 12 mA (depends on encoder position)
All RCX sensors and motors (via cable adapter)	0

Tab. 2.3: Odběr proudu jednotlivými senzory

- Digitální I/O piny (DIGIAI0 a DIGIAI0) jsou využívány pro číslicovou komunikaci implementovanou jako I2C komunikační kanál s rychlostí 9600 bit/s. Číslicové signály jsou přímo propojeny s mikroprocesorem NXT

Propojovací kabely

NXT přípojky jsou podobné telefonním RJ-12 přípojkám se šesti kontakty. Nicméně kolíček je na straně konektoru, místo uprostřed. Pokud bychom si chtěli vyrobit k NXT vlastní kabely, je nutné mít na paměti, že kabel RJ-12 má 6 žil na šesti pozicích, zatímco RJ-11 (standardní telefonní kabel) má 4 žíly na 6ti pozicích, RJ-9 má 4 žíly na 4 pozicích a RJ-45 (standardní síťový kabel) má 8 žil na 8mi pozicích. Více o modifikování kabelů nabízí zdroj [2].



Obr. 2.14: Propojovací kabely RJ-12 pro NXT

Senzory

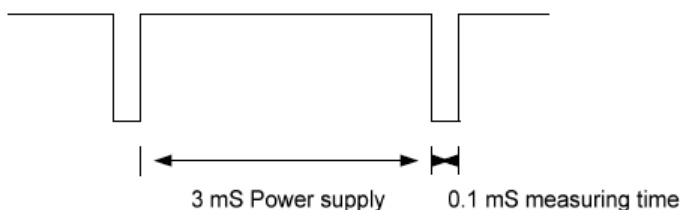
Senzory dávají robotovi informace o jeho prostředí. Ke kostce NXT jsou připojeny čtyřmi vstupními a třemi výstupními porty. Předpřipravené softwarové vybavení NXT kostky umožňuje pomocí jednoduchých programů tyto senzory vyzkoušet.



Obr. 2.15: Ukázka připojení senzorů ke kostce NXT

Aktivní Senzory

Zdroj proudu zásobuje senzory energií po dobu 3 ms a poté se 1 ms měří analogová hodnota snímaná senzorem.



Obr. 2.16: Způsob práce s aktivními senzory

Světelný senzor

Světelný senzor umožňuje robotovi reagovat na různé úrovně a barvy světelného záření. Obsahuje LED diodu jako zdroj světla, který může být zapnutý nebo vypnutý pomocí softwaru. To nám umožňuje měřit buď odražené světlo z LED, které se odráží k senzoru, nebo můžeme měřit pouze světelnou intenzitu okolí.

Na obrázku můžeme vidět LED. Fototranzistor a konektor připojené k desce s tištěnými spoji.



Obr. 2.17: Světelný senzor a jeho detaily

Fototranzistor ve světelném senzoru je mnohem citlivější na infračervené záření než relativně úzké spektrum barev, které vidíme my.

Světelný senzor umožňuje robotovi rozlišovat mezi světlem a tmou, vnímat úroveň světelného osvětlení místnosti a měřit světelnou intenzitu na barevných objektech. Na obrázku můžeme vidět vnímání barev člověkem a robotem.



Obr. 2.18: Vnímání barev člověkem a světelným senzorem

Rotační senzor

Rotační senzor je součástí servomotoru a slouží k měření otáček motoru. Tento optický senzor pracuje na principu znázorněném na Obr. 2.19. Emitory infračerveného záření generují paprsek světla, který prochází kotoučkem skrze pravidelně rozmístěné štěrby, které pouštějí světlo, které poté dopadá na fototranzistory, které toto světlo detekují a s ním i pohyb kotouče.

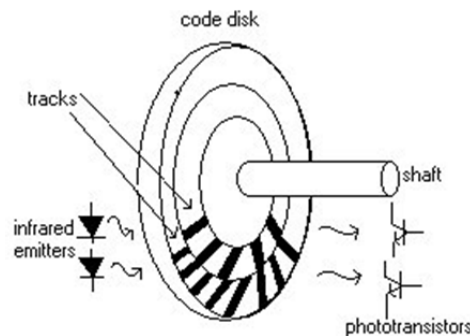


Fig 1. A rotary optical encoder

Obr. 2.19: Princip rotačního senzoru

Pasivní senzory

Senzory, které nevyžadují speciální časování pro dodávání energie a měření veličiny jako aktivní senzory, jsou nazývány pasivními senzory. Tyto senzory jsou také vzorkovány každé 3 ms, protože používají stejný A/D konvertor jako aktivní senzory

Zvukový senzor

Zvukový senzor měří různé úrovně hlasitosti zvuku. Tento senzor neoceníme v hlučné místnosti, ale můžeme ho využít v klidném prostředí, kde robot nebude rušen neadekvátními podněty.

Úroveň zvukové hlasitosti jsou měřeny v decibelech (dB). Zvukový senzor může měřit úroveň zvuku až do 90 dB. Snímané hodnoty jsou zobrazovány v procentech rozsahu, který je senzor schopen vnímat. Pro srovnání, 10-30% je normální úroveň konverzace v blízkosti senzoru a 30-100% představuje rozsah od hlasité řeči až po velmi hlasitou hudbu. Tyto rozsahy předpokládají vzdálenost od robota 1m a nižší.



Obr. 2.20: Zvukový senzor a jeho detail

Dotykový senzor

Tento senzor umožňuje robotovi reagovat na překážky v jeho okolí. Je pouze jakýmsi přepínačem, který nám nahlásí, pokud se dostane do přímého kontaktu s nějakým podnětem. Reaguje jak na stlačení, tak na uvolnění stlačitelné části a to tím způsobem, že vnímá změnu napětí, které se mění rozpojováním nebo spojováním jednoduchého elektrického obvodu a tím přerušuje či zajišťuje tok elektrického proudu.

Uvnitř dotykového senzoru můžeme najít desku s tištěnými spoji, která spojuje dotykovou část s komunikačním kabelem. Součástí desky je rezistor sériově zapojený se spínačem, zabraňující zkratu v případě zapojení do výstupního místo vstupního portu.



Obr. 2.21: Dotykový senzor a jeho detail

Číslicové pasivní senzory

Všechny senzory používající I2C komunikační kanál se nazývají číslicové, protože obsahují externí mikrokontroler ošetřující vzorkování reálného prostředí.

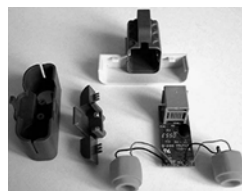
Ultrazvukový senzor

Představitel číslicových pasivních senzorů slouží k měření vzdálenosti a tím pomáhá k základnímu úkolu robota - vyhýbání se překážkám a rozpoznávání pohybu.

Senzor má dosah až jednoho metru a může tak sloužit k dobré prostorové orientaci robota.

Typickými konstrukčními prvky ultrazvukového senzoru jsou dva cylindrické objekty spojené s deskou - ultrazvukový vysílač a přijímač. Tento senzor představuje tak komplexní systém, že potřebuje svůj vlastní mikroprocesor. Senzor pracuje na bázi sonaru (podobně, jako měří vzdálenost netopýři), kdy měří čas, který trvá zvukové vlně k dosažení překážky a návratu zpět. Sonar pracuje na frekvenci 40 kHz a je schopen měřit vzdálenosti od 0 cm do 2,5 m s přesností +/- 3 cm.

Pokud je překážka představována plochou stěnou, je měření velmi spolehlivé. O měření vzdálenosti překážek tvořených malými objekty se tak říci nedá.



Obr. 2.22: Ultrazvukový senzor a jeho detail

Nadstandardní nabídka senzorů

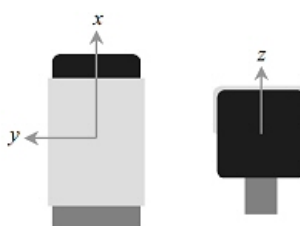
Firma Mindsensors nabízí spolu s mnoha doplňky včetně kabelů, zástrček a socketů navíc i mnoho senzorů ke kostce NXT. Více v [41].

Dalším prodejcem nadstandardní nabídky senzorů (kompas, gyroskop, senzor barev, akcelerace) je pro Českou republiku firma Eduxe s.r.o. [21]. Více o této nadstandardní řadě také na [26].

Nadstandardní řada senzorů je stejně jako jejich standardní protějšek připojována ke kostce NXT standardním NXT kabelem využívajícím I²C komunikace. Hodnoty snímané senzory jsou aktualizovány 100x za sekundu.

Akcelerometr

Senzor akcelerace obsahuje akcelerometr, který vnímá změnu rychlosti pohybu ve třech různých osách (viz Obr. 2.23: Akcelerometr). Akcelerometr měří v rozmezí od -2 g do 2 g s měřítkem přibližně 200 jednotek na g. Tento senzor taktéž měří náklon ve třech osách.



Obr. 2.23: Akcelerometr

Color Sensor

Senzor rozlišující barvy využívá tři různé barevné diody, které emitováním světla osvětlují cílový povrch a měří intenzitu barev odraženou zpět. Senzor poté zpracuje intenzitu odrazu dané barvy a spočítá číslo barvy, které vrátí programu zpět. Více informací o tomto senzoru poskytnete zdroj [27].



Obr. 2.24: Color senzor

Compass senzor

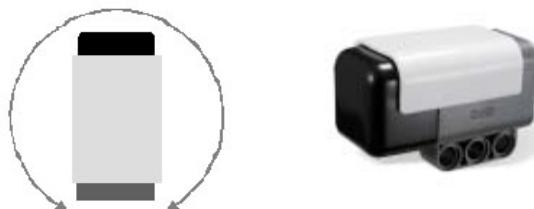
V případě tohoto senzoru se jedná o kompas, který měří magnetické pole Země a vrací hodnotu reprezentující současný směr pohybu robota v rozmezí od 0 do 359 stupňů.



Obr. 2.25: Compass senzor

Gyroskop

Gyroskopický senzor vrací počet stupňů za sekundu, o které se gyroskop otočil, stejně tak jako směr natočení. Díky tomuto senzoru můžeme robota naučit držet rovnováhu, houpat se, nebo plnit jiné funkce založené na měření otáčení robota. Na rozdíl od předchozích senzorů může gyroskop číst snímanou hodnotu až 300x za sekundu.



Obr. 2.26: Gyroskop

I²C komunikace

Implementace digitálního prostředí v kostce NXT je realizována I²C protokolem. Tento komunikační standard byl vyvinut v osmdesátých letech společností Philips Semiconductors za účelem jednoduché digitální komunikace.

I²C komunikace slouží jako digitální rozhraní pro externí zařízení komunikující s kostkou NXT. Toto umožňuje externím zařízením pracovat samostatně a s kostkou NXT si tak vyměňovat data.

Kostka NXT má 4 I²C komunikační kanály, každý pro jeden vstupní port.

I²C komunikace je však implementována pouze ve formě "maser", což znamená, že kostka NXT řídí veškeré datové toky v každém z komunikačních kanálů.

Důležitým aspektem I²C komunikace mezi dvěma zařízeními je hardwarové nastavení v každém z těchto zařízení.

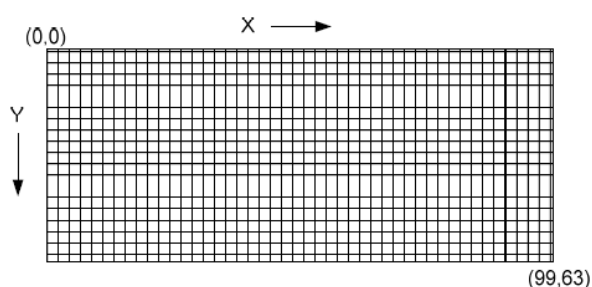
Uspořádání paměti externích zařízení

Využívání I²C komunikace poskytuje několik způsobů, jak definovat a implementovat funkci čtení a zápisu dat z a do externího zařízení. My charakterizujeme LEGO externí zařízení jako externí paměť, ze které můžeme data číst, nebo do které můžeme data zapisovat. Tím, že budeme znát specifikace stojící za každým umístěním paměti v externím zařízení, pak bude možné tato zařízení řídit a číst z nich požadovaná data.

Display

Maticový 100 x 64 pixelový LCD černobílý grafický display představuje na NXT kostce grafické uživatelské rozhraní. Zobrazovací oblast má velikost 26 x 40.6 mm.

Rozhraní SPI mezi ARM7 mikrokontrolerem a LCD-kontrolerem UltraChip 1601 SPI, běží na frekvenci 2MHz v rámci LEGO firmwaru a dvěma paměťovými mapami sloužícími k aktualizaci displeje každých 17 ms. Napájení displeje je zajištěno třemi volty. Kontrast displeje je upravitelný.



Obr. 2.27: Bitmapová mapa displeje ukazující rozvržení pixelů na displeji

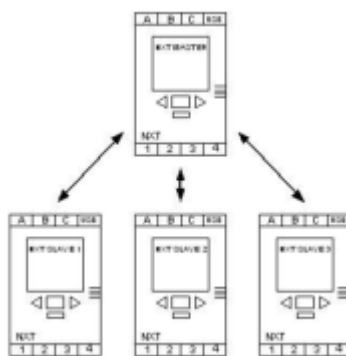
Bluetooth

Technologie Bluetooth umožňuje uživateli komunikovat s robotem jak přes počítač, tak přes mobilní telefon či jiná zařízení podporující tuto technologii. Uživateli tak umožňuje nahrávat do kostky NXT programy bez použití USB kabelu, posílání programů a příkazů z mobilního telefonu, nebo zaslání souborů mezi kostkami NXT.

Kostka NXT podporuje bezdrátovou komunikaci využitím technologie Bluetooth využitím čipu CSR BlueCore™4 v. 2. Kostka NXT tak může být současně bezdrátově propojena až se třemi dalšími zařízeními, nicméně může komunikovat současně pouze s jedním. Tato funkce je implementována využitím sériového profilu SSP (Seriál Port Profile), který je uvažován jako bezdrátový port. Díky němu je tak možné posílat soubory v rámci dvou kostek za účelem předávání informací mezi kostkami během programu. Bluetooth komunikace je možná až do vzdálenosti deseti metrů.

Bluetooth komunikace v rámci komunikace mezi kostkami NXT je nastavena jako master/slave komunikační kanál. To znamená, že jedna kostka NXT v rámci sítě kostek musí být nastavena jako master a ostatní NXT kostky komunikují přes tohoto mastera.

Obr. 2.28 ukazuje, které NXT zařízení může v rámci sítě přímo komunikovat s kterým.



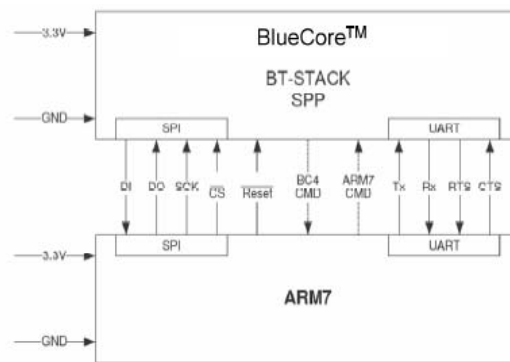
Obr. 2.28: komunikace mezi kostkami NXT v rámci Bluetooth.

Jak je ukázáno na Obr. 2.28, master NXT může být propojen se třemi dalšími kostkami najednou. Master ale může komunikovat v daném momentu pouze s jednou kostkou. Což znamená, že pokud master NXT komunikuje s NXT Slave 1 a NXT slave 3 mu začne posílat data, master tato data nevyhodnotí až do té doby, než komunikaci přepne k NXT Slave 3. Kostka NXT není schopna zastat obě funkce master, slave, protože by mohlo dojít během komunikace ke ztrátě dat.

Kostka NXT má 4 Bluetooth kanály, které slouží ke komunikaci mezi kostkami. Kanál 0 je vždy používán "NXT slave" zařízením pro komunikaci s masterem, zatímco kanály 1, 2 a 3 jsou používány pro komunikaci od mastera k "slave" zařízením.

Bluecore Chip

Funkčnost Bluetooth v rámci NXT kostky je implementována použitím samostatného čipu - CSR BlueCoreTM4 s externí 8Mbit FLASH pamětí. Firmware v rámci BlueCore integruje uživatelsky programovatelný VM-task umožňující spouštění nenáročného kódu. Řádkový interpret integrovaný do VM-tasku umožňuje dekodování a odezvy na příkazy přijaté z procesoru ARM7. (Viz Obr. 2.29)



Obr. 2.29: rozhraní mezi procesorem ARM7 a BlueCore čipem.

Zvuk

Kostka NXT obsahuje zvukový ovladač (SPY 0030A), což je diferenční zvukový zesilovací čip, který může zesilovat až dvacetinásobně výstupní hladinu zvuku. Výstup zvuku je PWM výstupní signál řízený mikrokontrolerem ARM7.

Zvuková podpora kostky NXT umožňuje spolu s odpovídajícím softwarovým vybavením uživatele vytváření vlastních melodií a převod MDI formátu do formátu spustitelného kostkou.

Motory

Tři výstupní porty kostky NXT slouží pro devíti voltové Servomotory sloužící jak uvádění robota do pohybu, tak k měření otáček motoru pomocí zabudovaných senzorů otáčení. To znamená, že můžeme ovládat pohyb robota tak, že ujede přesně zadanou vzdálenost přesně zadanou rychlostí. Motory je možné synchronizovat, což umožňuje přesný pohyb vpřed.

Jak je vidět na Obr. 2.30, motorek je převodován a jeho hnací síla pohání otočnou jednotku rychlostí až 1500 otáček za minutu. Stejněsměrně napájené motory se vždy točí příliš rychle na to, aby byly přímo připojeny na kola a jinou zátěž. Ke zpomalení se zde využívá systému ozubených kol. Senzor otáčení nenasadí přímo na osu motoru, ale je rovněž spojen s hnací silou motoru pomocí ozubených kol.

Motory mají zabudovaný rotační senzor. Tento senzor umožňuje velmi přesný pohyb. Pohyb motorů je měřen buď ve stupních, nebo v otáčkách



Obr. 2.30: 9V servo motor + detail

Zátěžové charakteristiky motoru, stejně tak jako zajímavé fotografie vnitřku servo motoru nabízí internetové zdroje od Philippa E. Hurbaina [2]. Rovnice potřebné pro identifikaci motoru uvádí taktéž zdroj [89].

Závěr ke kapitole 2

Ve druhé kapitole jsme se seznámili jak se současným využitím robota, tak i s hardwarovými součástmi, ze kterých se stavebnice skládá. Výčet údajů je omezen a proto doporučuji čtenáři se zájmem o bližší informace k problematice uvedené zdroje.

3. MOŽNOSTI PROGRAMOVÁNÍ ROBOTY

3.1. Úvod do třetí kapitoly

Stejně tak jako mnoho dalších elektronických zařízení, i kostka NXT má svůj vlastní operační systém. S tímto operačním systémem nahaným na flash paměť NXT kostky se uživatel setká prostřednictvím LCD displeje, kdy s kostkou komunikuje a pomocí operačního systému spouští aplikace. Operační systém uživateli zpřístupňuje prostředky mikropočítače pro obsluhu hardwaru (tlačítka, senzory, motory), ale i pro práci se souborovým systémem a možnostem práce s grafikou na LCD displeji.

Firmware umožňuje ukládat a spouštět programy naprogramované uživatelem, ale i předpřipravené výrobcem softwaru. Těchto programů může být díky 256ti kb paměti až 64.

Některé programovací jazyky vyžadují výměnu firmware v kostce NXT a komunikace s kostkou může být prováděna rozdílně.

3.2. Přehled možností programování robota

Následující tabulka ukazuje přehled programovacích možností robota. Ve další části se podrobněji zaměříme pouze na tři z nich.

Programovací prostředí	NXT-G	NXC	RobotC	leJOS NXJ	LEJOS OSEK	Matlab
Jazyk	Graphic	(not exactly) C	C	Java	ANSI C	RWTH Matlab Toolbox
Platformy	Windows, MAC OS	Windows, MAC OS, Linux	Windows	Windows, MAC OS, Linux	Windows, Linux	Windows
IDE (included?)	ANO	ANO	ANO	plugins for Eclipse and NetBeans.	Eclipse CDT(GCC+ATMEL SAM-BA)	ANO
Obsluha událostí	NE	NE	ANO	Standard Java events	ANO (OSEK RTOS)	?
Multivláknové aplikace	ANO	ANO	ANO	ANO	ANO (OSEK RTOS)	?
Podpora Bluetooth	ANO	ANO	ANO	ANO	ANO	ANO

Tab. 3.1: Přehled programovacích možností

Mezi méně používané pak patří například C#, RUBY, Pblua, Python[55].

3.3. Návod pro instalaci základních typů prostředí

NXT-G

Úvod

NXT-G programovací jazyk dostal jméno z programovacího jazyka využívaného programem LabVIEW, vyvinutého firmou National Instruments, který se jmenuje pouze G. Zkratka „G“ pochází z faktu, že programovací jazyk je grafický. Programy napsané v NXT-G jsou tedy poskládané z grafických bločků, pospojovaných dohromady, u kterých se nastavují jejich vlastnosti a posloupnosti. NXT-G je výsledkem práce firem LEGO a National Instruments a je základním programovacím nástrojem pro LEGO MINDSPRINT NXT a je dodáván spolu se stavebnicí. Překladač „kódu“ NXT-G je součástí softwaru dodávaného s produktem NXT. Důraz je u NXT-G kladen především na intuitivnost a jednoduchost vývojového prostředí včetně procesu programování tak, aby s robotem mohli pracovat už i žáci základních škol, kteří nemají s programováním žádné, nebo minimální zkušenosti.

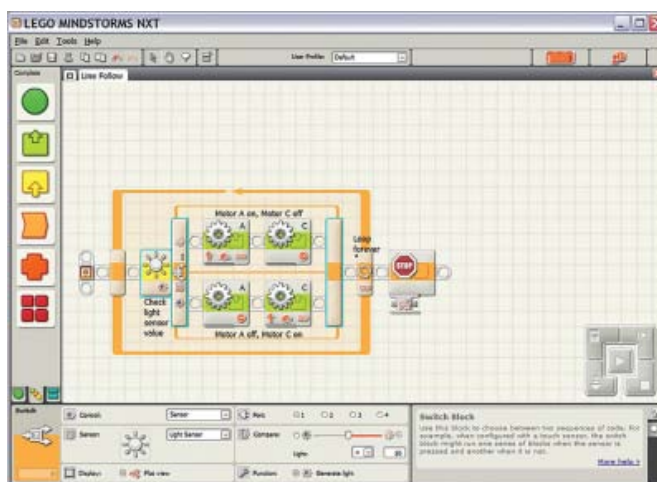
Nevýhodami NXT-G jsou naopak nevhodnost pro psaní rozsáhlejších programů, komplikovanost s uživatelsky definovanými proměnnými, nepřítomnost polí jakožto datových struktur, nemožnost pracovat s jinými než s celými čísly v rámci zobrazování čísel na LCD. Jako osobní zkušenost uvádím fakt, že vývojový proces v grafickém jazyce je časově náročnější, než vývoj v jazyce s textově orientovanou syntaxí a doporučil bych ho jen jako užitečný nástroj pro počáteční experimenty se stavebnicí ([11], [12]).

Bližší popis programovacího prostředí nabízí například literatura [15].

Instalace prostředí

K instalaci vývojového prostředí pro jazyk NXT-G je třeba nainstalovat Mindstorms Education NXT Software, který je dodáván spolu se stavebnicí. Instalace je velmi snadná a nevyžaduje speciální návod.

Tipy a triky na programování v jazyku NXT-G mohou zájemci najít v [57]. Více o používání grafických bloků ve zdroji [60].



Obr. 3.1: Grafické vývojové prostředí jazyka NXT-G

Zdroje a návody

Jako velmi slušný rozcestník mohu doporučit webové stránky Abu Dhabi Women's College [22].

NXC

Úvod

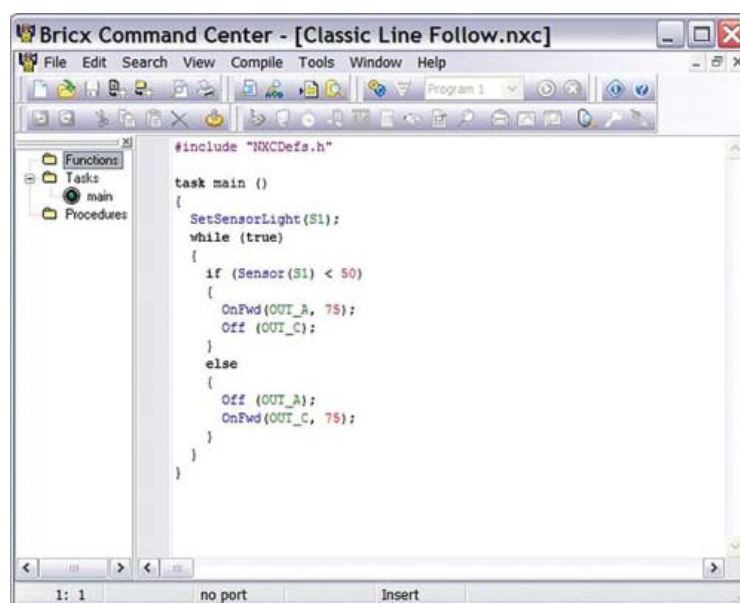
Textový jazyk odvozený od jazyka C pracuje v prostředí BricxCC na standardním firmwaru LEGO Mindstorms. Tato skutečnost je velmi příjemná pro ty, kteří chtějí programovat jak v NXT-G, tak v NXT, protože s každou změnou programovacího prostředí nemusí do kostky nahrávat nový firmware. Práce s jazykem zkracujícím spojení „Not eXactly C“ je velmi příjemná a programátor alespoň trochu znalý jazyka C si díky téměř stejné sémantice v tomto prostředí zvykne programovat velmi snadno. Další výhodou je, že se jedná

o freewarovou aplikaci. Jako nevýhodu bych uvedl někdy nepříliš snadné debugování programů. Na rozdíl od NXT-G se jedná o čistě textové programování bez grafických prvků.

Návod na instalaci programovacího prostředí.

K vývoji našich programů v NXC budeme potřebovat nainstalovat vývojové prostředí Briccc, jehož instalační program můžeme stáhnout ze stránky (<http://sourceforge.net/projects/bricxcc/>). Po nainstalování programu je nutné při jeho spuštění vybrat typ kostky a typ připojení. Proto vybereme: Port: usb, BrickType: NXT, Firmware: Standard. Programy vytvořené v tomto prostředí ukládáme s příponou (*.nxc). Po zkompilování programu můžeme program stáhnout do kostky a spustit [10].

Více o instalaci či návodech k programování robota v NXC nalezneme v [15], [54], [56]. Jako základní příručku k úvodu do programování doporučuji zdroje [9] a [10]. Náhled na uživatelské prostředí je k znázorněné na Obr. 3.2.



Obr. 3.2: Programovací prostředí BricxCC pro programovací jazyk NXC

LeJOS NXJ

Úvod

Posledním zde uváděným, ale nikoliv posledním možným způsobem, jak lze robota programovat, je programování pomocí speciálního LEGO JAVA Operačního Systému

(leJOS), který umožňuje spouštět v kostce NXT Java kód. Tento OS, nahrazující původní firmware programovatelné kostky, obsahuje interpret Java Virtual Machine, který poskytuje přímý přístup k vstupům a výstupům samotného robota, stejně tak jako k podmnožině funkcí Java API [7].

Tento produkt je šířený společností Sourceforge zdarma a je k dispozici pro operační systémy Windows, Linux a MAC OS [46].

Tento nástroj považují mezi uvedenými jako dosud nejpokročilejší. Díky rozšíření a znalosti jazyka Java mezi programátory si tak velká část uživatelů systému LEGO MINDSPORMS vybírá právě leJOS NXJ s jeho rozsáhlými knihovnami, které podporují zajímavé funkce robota.

LeJOS NXJ obsahuje:

- Nový firmware pro NXT, který obsahuje Java Virtual Machine, a který nahrazuje standardní LEGO firmware. Lego firmware může být nahrát do kostky zpět pomocí LEGO software.
- Knihovnu tříd java, kterou implementuje leJOS NXJ API.
- PC nástroje pro nahrání nového firmwaru do NXT kostky, nahrávání programů, debugování a další funkce.
- PC API pro zapisování PC programů přes USB, Bluetooth nebo LEGO komunikační protokol.
- Mnoho testovacích program.

LeJOS NXJ nabízí:

- Programování v standardním jazyku Java v IDE Netbeans nebo Eclipse.
- Objektově orientované programování.
- Prác i s preemptivní vlákny, vícedimenzionálními poli.
- Možnost rekurzivního programování.
- Synchronizace, Výjimky, javovské typy včetně float, long a string.
- Většinu z tříd java.lang, java.util and java.io.
- Výborně dokumentované "Robotics API".
- ... a mnoho dalšího [46].

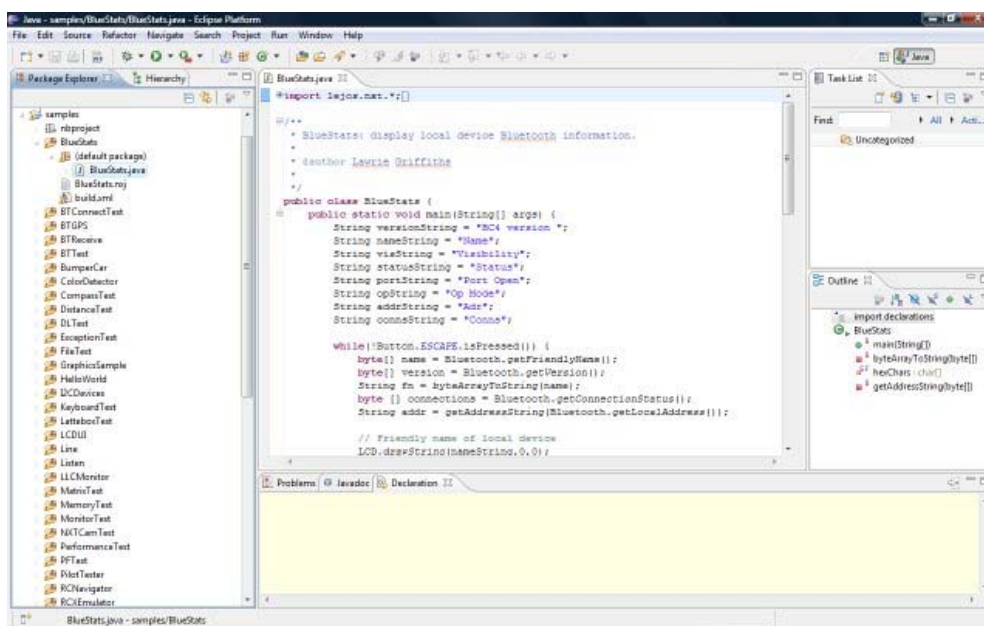
Návod na instalaci programovacího prostředí.

Javovský kód je možné psát jak ve vývojovém programovacím prostředí Eclipse, tak i v Netbeans. Osobní zkušenost s IDE Eclipse a instalace pod MS Windows Vista nabízí popis instalace pro toto prostředí. Stejně dobře popisují instalaci i cizojazyčné zdroje [8], [47], [48], ze kterých jsem čerpal.

1. Nainstalujeme originální software od od Lego Mindstorms, díky čemuž kromě programovacího jazyka NXT-G dostaneme také USB ovladač pro komunikaci počítače s kostkou.
2. Nainstalujeme nejnovější verzi JDK (java Development kit) <http://java.sun.com>
3. Podrobný návod k instalaci JDK například v [53].
4. Nainstalujeme nejnovější vydání leJOS NXJ (http://lejos.sourceforge.net/p_technologies/nxt/nxj/downloads.php)
5. Po instalaci budeme vyzváni k nainstalování nového firmwaru. Po nainstalování firmwaru je již vše připraveno pro spouštění Java programů v kostce.
6. Pro snadnou kompilaci a spouštění programů je nejsnazší nainstalovat jedno z integrovaných vývojových prostředí, které nabízí syntakticky řízený editor ukazující chyby v programu a nabízející jeho snadné debugování. V této práci bylo vybráno vývojové prostředí Eclipse.
7. Stáhneme tedy Eclipse (<http://www.eclipse.org/downloads/>). K naší práci nám bude stačit pouze balíček "Eclipse IDE for Java Developers". Balíček rozbalíme v cílovém adresáři a spustíme pomocí eclipse.exe.
8. Spolu s instalací leJOS NXJ jsme dostali také testovací programy pro naši kostku. Ty je potřeba nyní do Eclipse workspace importovat (File→Import→General→Existing Projects into Workspace).
9. Pro snadný překlad a spouštění programů v rámci prostředí Eclipse ještě do Eclipse nainstalujeme speciální plugin. Tento plugin automaticky konvertuje naše projekty do formátu přijatelného leJOS NXJ.
10. Pro instalaci našeho pluginu zvolíme „Help>Software Updates>Add Site> <http://lejos.sourceforge.net/tools/eclipse/plugin/nxj/>. Naší stránku poté vybereme a zvolíme nainstalování pluginu a po úspěšném nainstalování Eclipse restartujeme.
11. Po restartování Eclipse upravíme některá nastavení pro náš plugin volbou Window → Preferences → LeJOS NXJ. Jako NXJ_HOME zvolíme místo, kam jsme leJOS nainstalovali.

12. Při vytváření nového projektu už jen zvolíme „Konvertování do leJOS NXJ project“ a programy nyní budou po přeložení plně funkční a stažitelné do NXT kostky.

Podrobný popis začátků programování v Javě s leJOS NXJ je velmi dobře popsán ve zdroji [6], ale stejně tak i online [46]. Náhled na programovací prostředí v Eclipse nám nabízí Obr. 3.3.



Obr. 3.3: Programovací prostředí Eclipse

4. NÁVRH SOUTĚŽNÍCH ÚLOH

4.1. Úvod do čtvrté kapitoly

Jak již bylo popsáno v kapitole 2.8, Robot LEGO MINDSTORMS je mnohostranným výukovým prostředkem. Každá ze zde uvedených úloh plní podobné pedagogické cíle, kterých lze za pomoci robota dosáhnout.

V rámci úlohy je vždy u řešitele očekáváno analyzování problému, jeho následné vyřešení a zhodnocení. Ať už se tedy jedná o práci se senzory, či plánování chování robota, vždy je třeba rozumět souvislostem a jednotlivým parametrům robota, které ovlivňují jeho chování a stejně tak i v jisté míře omezují možnosti řešení jednotlivých úloh.

4.2. Obsah čtvrté kapitoly

Cílem této části bakalářské práce je navrhnout tři soutěžní úlohy pro předmět ROBOTI. Pro každou úlohu budou uvedena pravidla pro její splnění a návrh jejího řešení. Pokaždé bude využito jiné konstrukční řešení specifické pro danou úlohu. Stejně tak bude využito třech různých programovacích jazyků a v nich nastíněn možný způsob řešení dané úlohy.

4.3. Úloha číslo 1: „Bludiště“

Návrh a pravidla této úlohy se inspirovala podobným zadáním na Univerzitě ve Stuttgartu [72] a úloha byla odzkoušena před zahájením ROBO SOUTĚŽE 2009 (viz kap.2.7).

Pravidla pro úlohu „Bludiště“

Cíl úlohy

Postavit a naprogramovat robota tak, aby samostatně, bez jakékoli další pomoci projel co nejrychleji bludištěm od jeho počátku až jeho nakonec.

Vybavení

Každý tým obdrží na začátku soutěže základní soupravu LEGO® MINDSTORMS® Education (9797), soupravu technických dílů (9648 - doplňkové pasivní díly) a síťový adaptér (9833), které po skončení soutěže vrátí v kompletním stavu organizátorovi soutěže na Katedře řídicí techniky.

NXT kostku lze programovat v libovolném programovacím jazyce. Při programování NXT kostky a testování robota lze využít k napájení akumulátor, vlastní baterie nebo síťový adaptér. Při vlastní soutěži je možné použít pouze akumulátor nebo baterie.

Podrobnosti k průjezdu bludištěm

Vlastního robota lze sestavit naprosto libovolně, ovšem pouze s využitím dílů ze zapůjčených souprav. Tento robot by měl zcela samostatně a co možná nejrychleji projet bludištěm ze stanoveného startovního prostoru do stanoveného cílového prostoru na základě předem stanovené strategie řízení (skupině se nebude počítat čas, pokud bude z trajektorie průjezdu bludištěm robota zřejmé, že není použita žádná strategie řízení).

Robota lze položit do startovního prostoru (cca 40 x 40 cm) ručně. Dále již musí pokračovat samostatně bez jakékoliv dodatečné korekce (např. akustické, atp.)

Čas se měří od okamžiku, kdy robot poprvé protne startovací linii (paprsek světelné závory). Do chvíle, kdy robot poprvé projede cílovou linií (paprskem světelné závory v cílové prostoru (cca 40 x 40 cm)).

Rozměry bludiště

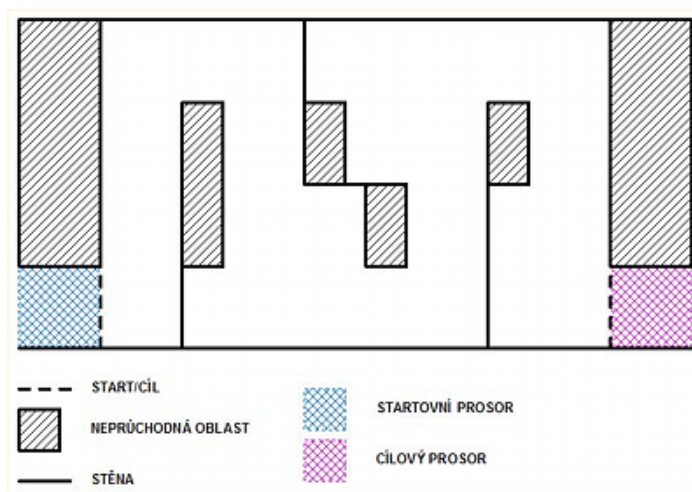
Každý robot musí bludištěm projet od začátku do konce zcela samostatně (bez jakékoliv pomoci a jeho ovládání, např. externě přes bluetooth nebo hlasovými povely). V případě porušení pravidel je tým okamžitě vyloučen ze soutěže.

Minimální vzdálenost mezi dvěma libovolnými stěnami je cca 40 cm. Všechny stěny bludiště stojí kolmo, tzn., že se v dráze nevyskytují žádné nakloněné stěny. Jsou rovné (přímkové) a nevyskytují se u nich žádná nepředpokládaná zahnutá. Výška stěny je cca 28 cm.

Obě dvě světelné závory jsou umístěny 5 cm od podkladu bludiště a vyčnívají cca 1 cm ze stěn dovnitř vlastní dráhy, takže šířka je na začátku i na konci bludiště 38 cm.

Celková plocha bludiště je 330 x 160 cm a podkladem jsou dřevěné parkety (viz Zengerova posluchárna na Fakultě Elektrotechnické, ČVUT v Praze).

Uspořádání bludiště



Obr. 4.1: Schéma soutěžního modelu

Níže uvedený vzhled bludiště je pouze jeden z možných příkladů jeho uspořádání pro vlastní soutěž. Může dojít jak ke změně polohy startovací a cílové oblasti, tak i ke změně uspořádání vnitřních stěn.



Obr. 4.2: Možný model bludiště

Průjezd bludištěm

Robot může bludištěm projet libovolným způsobem. Počítá se pouze celkový čas průjezdu. Bludiště je postaveno tak, aby nejkratší cesta mezi startovacím a cílovým prostorem nikdy nevedla zpět ke startovací linii.

Průjezd vede stále směrem k cílovému prostoru. Volba senzorů a strategie průjezdu závisí pouze na jednotlivých týmech. Při orientaci v bludišti se roboti smějí dotýkat stěn.

Vlastní soutěž

Vlastní soutěž bude probíhat dvoukolově. Výsledkem bude lepší čas z obou kol. Týmům bude umožněno 1,5 hodiny před začátkem soutěže testovat robota na variantě stejného bludiště, jaké bude použito pro vlastní soutěž. V tomto čase však není dovoleno provádět jakékoliv měření rozměrů bludiště.

Mezi oběma koly je možné vyměnit (nebo dobít) akumulátor či baterie a je také přípustné provést případné změny, opravy či úpravy softwaru.

Konstrukční řešení robota

Robot LEGO MINDTORMS nabízí k řešení úlohy celou řadu senzorů a stejně tak jako programovacích řešení i konstrukčních řešení může být mnoho, dle fantazie konstruktéra a programátora. V našem případě bylo použito konstrukční řešení uvedené na Obr. 4.3.

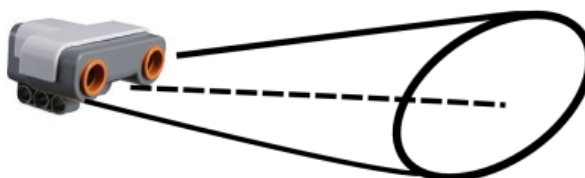


Obr. 4.3: Konstrukční řešení robota: pohled zepředu a z boku

Toto konstrukční řešení bylo vybráno z několika důvodů. Tím prvním je, že vychází z klasického návrhu Tribota popsaného běžně dostupným návodem. Samotné konstrukční řešení je poměrně časově náročné a tak byla konstrukce co nejvíce zjednodušena.

Další důvody, proč bylo vybráno toto řešení, jsou již zcela orientovány k programátorskému řešení úlohy. K průjezdu bludištěm jsem se rozhodl využít dva motory (a spolu s nimi dva rotační senzory), které budou zajišťovat pohyb robota skrz bludiště. Dále pak ultrazvukový senzor měřící vzdálenost a dva dotykové senzory vnímající náraz, či kontakt s objektem.

Jak během řešení vyplynulo, využití pouze ultrazvukového senzoru pro průjezd bludištěm je velmi obtížné, díky vnímání vzdálenosti, kterou senzor snímá v postupně se rozšiřujícím zorném úhlu (Obr. 4.4), proto nelze spoléhat pouze na něj. Tento senzor však hraje v řešení úlohy důležitou roli v detekci volného prostoru a pro vyřešení úlohy ho využijeme spolu s dalšími senzory.



Obr. 4.4: snímání vzdálenosti ultrazvukovým senzorem

K dořešení úlohy byly využity dva dotykové senzory, které zajišťují detekci kolmému kontaktu se stěnou a usnadňují robotovi orientaci v bludišti. Více o využití senzorů v následující kapitole.

Konstrukční řešení robotů z Robo soutěže 2009

Pro ukázkou uvádím několik fotografií robotů, kteří byli zkonstruováni pro Robo soutěž. Jak je z fotografií vidět, konstrukčních řešení je skutečně mnoho a je čistě na uživateli, jaké řešení zvolí.



Obr. 4.5: Konstrukční řešení robotů z Robo soutěže

Programátorské řešení úlohy

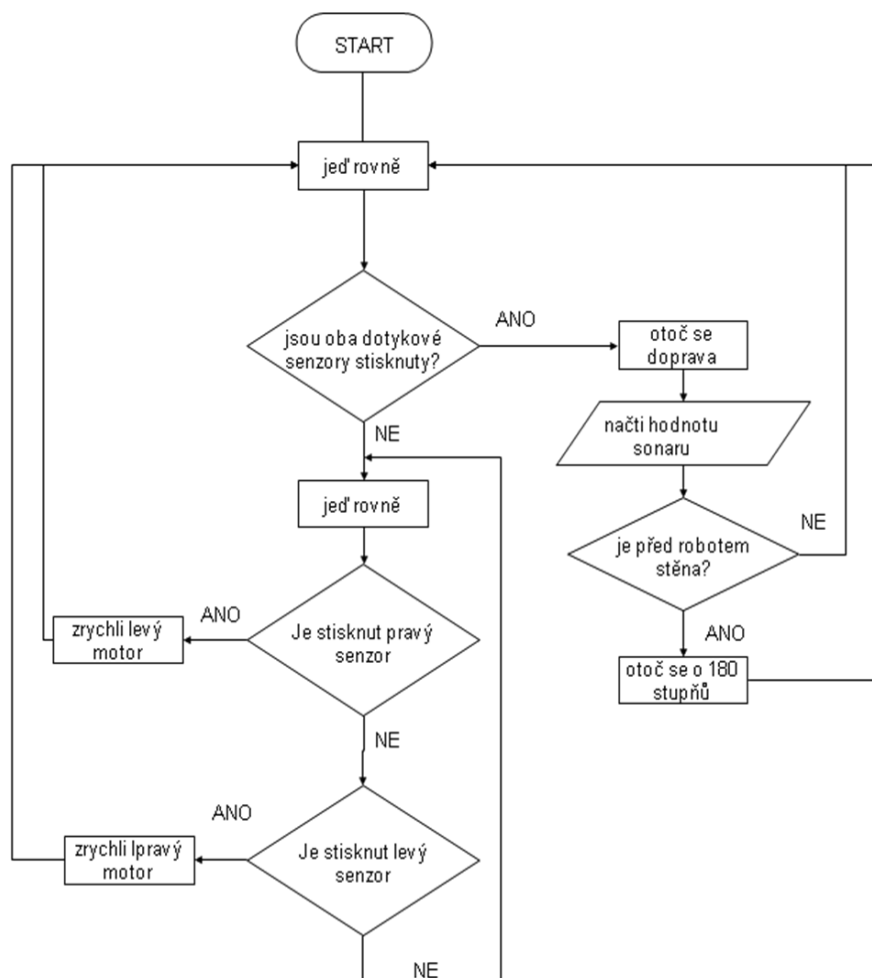
K naprogramování robota pro řešení úlohy byl využit jazyk NXC. Vnímám ho jako uživatelsky velmi příjemný a vhodný pro začínající programátory.

Programátorské rozhraní NXC definuje řadu maker, funkcí a konstant, které poskytují přístup k nejrůznějším možnostem kostky NXT jako je využití senzorů, výstupů z kostky a komunikace s kostkou, což programování značně zjednodušuje.

Popsané programátorské řešení se může lišit od skutečného zdrojového kódu. Neuvádím zde většinu výpisů na obrazovku během programu, protože jsou pro běh programu nepodstatné a věnuji se hlavně náročnějším funkcím a úkonům důležitým pro splnění úlohy.

Vývojový diagram programu

Vývojový diagram na Obr. 4.6 ukazuje průběh programu. Konkrétní forma naprogramování je rozepsána dále.



Obr. 4.6: Vývojový diagram pro úlohu "bludiště"

Funkce a konstanty poskytnuté NXT API

Pro naprogramování robota využijeme připravených funkcí (uvádím jenn ty nejdůležitější):

- Časová funkce *Wait(time)*: Zajišťuje čekání robota, nebo setrvávání robota v jednom stavu. Základní hodnota jsou milisekundy. Argumentem funkce může být výraz nebo konstanta.
- Funkce pro nastavení senzorů: *SetSensorSound(port)* slouží pro konfigurování senzorů a přiřazení vstupních portů. Hodnota z tohoto senzoru je poté načítána z funkce *Sensor(port)*.
- *OnFwd(outputs, pwr)*: funkce pro nastavení pohybu vpřed daných motorů.
- *OnRev(outputs, pwr)*: funkce nastavující zpětný pohyb u daného motoru.
- *Coast(outputs)*: funkce pro zastavení daných výstupů.
- *NumOut(x, y, value, clear = false)* Funkce sloužící k výpisu číselné hodnoty na obrazovku
- *TextOut(x, y, msg, clear = false)*: funkce sloužící k výpisu textu na obrazovku

Z připravených konstant využijeme ty pro ovládání výstupů. Bud' vždy jednoho samostatně, nebo více naráz:

OUT_A, OUT_B, OUT_C, OUT_AB, OUT_AC, , OUT_BC, OUT_ABC

Více k programovacímu jazyku NXC v literatuře [3].

Samotné tělo programu

Projděme postupně jednotlivé části kódu a zjistěme, jak robot úlohu vyřeší. Uvádím jen důležité části. Kompletní kód programu včetně komentářů je uveden v elektronické příloze.

Nejprve uvedeme funkce, které robot během průjezdu bludištěm využije.

Krátké zacouvání zajišťující vzdálení se od stěny pro snadné otočení do směru kolmého na stěnu:

Kapitola 4: Návrh soutěžních úloh

```
void backup()  
{  
    OnRev(OUT_BC, 50);  
    Wait(200);  
}
```

Otočení vpravo (argumenty jsou rychlost otočení a čas) a obdobně i otočení vlevo a otočení o 180 stupňů.

```
void pivotRight(int speed, int time){  
    OnFwd(OUT_C, speed);  
    Wait(time);  
    Coast(OUT_B);  
    Coast(OUT_C);  
}
```

Pokračujme hlavní funkcí, která zajišťuje chod programu. Provedeme nastavení senzorů.

```
task main()  
{  
  
    SetSensorTouch (S1);  
    SetSensorTouch (S4);  
    SetSensorLowspeed(S2);  
}
```

Hlavním úkolem robota bude opakovat stále stejnou proceduru:

```
while(true)  
{  
    ...  
}
```

Jet dopředu:

```
OnFwd(OUT_B, 63);  
OnFwd(OUT_C, 60);
```

Pokud robot narazí na překážku, pak zajistíme, aby se k ní postavil kolmo z důvodu dalšího přesného pohybu bludištěm. Proměnné t_r a t_l představují hodnoty dvou dotykových senzorů.

```
if (t_r == 1 && t_l == 0) {  
    OnFwd(OUT_B, 20);  
}  
  
if (t_l == 1 && t_r == 0) {  
    TextOut(0, LCD_LINE1, "sync", true);  
}
```

Poté co se robot srovná kolmo na stěnu, může pokračovat dál bludištěm. Otočí se doprava a prozkoumá průjezdnost terénu:

```
if (t_l == 1 && t_r == 1)
{
...
    backup();
    pivotRight (50, 500, sonar);
    Wait(500);
    sonar = SensorUS(S2);
    Wait(500);
}
```

V případě, že je prostor neprůjezdný, robot se otočí na druhou stranu, která dle zadání úlohy logicky průjezdná je.

```
if (sonar < 55)
{
    TextOut(0, LCD_LINE1, "wrongway");
    NumOut(1, LCD_LINE2, sonar);
    backup1();
    Coast(OUT_B);
    Coast(OUT_C);
    Wait(1000);
    //otoceni o 180 stupnu
    turnRight(100, 800, sonar);

    Coast(OUT_B);
    Coast(OUT_C);
    Wait(1000);
}
```

Za touto podmínkou končí tělo nekonečného cyklu, který se opakuje opět od začátku.

Závěr k návrhu úlohy „Bludiště“

Úlohu bludiště se podařilo úspěšně navrhnout i vyřešit. Videozáznam o jejím vyřešení je na webové stránce <http://www.youtube.com/watch?v=MMZruQwgRTU>.

Během řešení úlohy byla odhalena zejména úskalí využití ultrazvukového senzoru pro měření vzdálenosti. Dále bylo zjištěno, že se řízení motoru chová rozdílně s rozdílným stavem nabití akumulátoru. Doporučuji mít tedy akumulátor pro správné chování robota plně nabit.

Velkou část řešení úlohy představovalo ladění senzorů otáček pro správné otáčení robota o daný úhel. Velmi záleží na povrchu, po kterém se robot pohybuje, stejně tak jako na uspořádání a konstrukčním řešení pohyblivých částí. Programování bylo tedy z velké části experimentální.

4.4. Úloha číslo 2: „Překážky“

Návrh a pravidla této úlohy opět vychází ze soutěže ROBORACE pořádané na Technické univerzitě ve Stuttgartu. Návrh a řešení této úlohy bylo vytvořeno pro budoucí potřebu předmětu ROBOTI.

Pravidla pro úlohu „Překážky“

Cíl úlohy:

Z poskytnutých LEGO dílů sestavte robota, který samostatně a co nejrychleji přemístí překážky na trati do cílových prostor.

Vybavení

Každý tým obdrží na začátku soutěže základní soupravu LEGO® MINDSTORMS® Education (9797), soupravu technických dílů (9648 - doplňkové pasivní díly) a síťový adaptér (9833), které po skončení soutěže vrátí v kompletním stavu organizátorovi soutěže na Katedře řídicí techniky. Je dovoleno pouze za pomoci tohoto vybavení sestavit robota pro soutěžní úlohu.

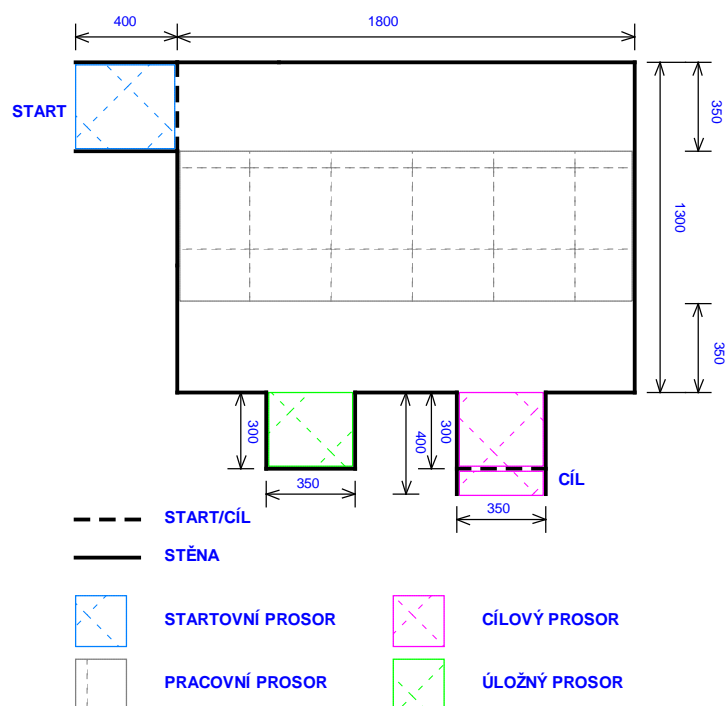
NXT kostku lze programovat v libovolném programovacím jazyce. Při programování NXT kostky a testování robota lze využít k napájení akumulátor, vlastní baterie nebo síťový adaptér. Při vlastní soutěži je možné použít pouze akumulátor nebo baterie!

Rozměry soutěžního plánu

Soutěž probíhá na dráze zobrazené na obrázku níže. Robot může být umístěn kamkoli do startovního pole před startovní čáru. Na spodní straně soutěžního plánu jsou umístěna depa, kam je třeba zavést obě překážky, postupně do depa 1 a depa 2. Depo 2 je zároveň i cílovým prostorem, kde je umístěna cílová linie.

Do těchto dep vede vždy černá čára, která nemá pravidelný tvar, ale její konec (přibližně 10 cm) je kolmý na stěnu soutěžního plánu. Černá čára je tvořená černou elektrikářskou lepicí páskou, širokou nejméně 1 cm. Povrch soutěžního plánu je tvořen bílou laminátovou deskou. Výška stěn je cca 35 cm.

Předmětem č.1 je tenisový míč, č.2 pak plechovka od nápoje.



Obr. 4.7: nákres soutěžního plánu pro úlohu "překážky"

Průjezd soutěžním plánem

Každý robot musí splnit úkol zcela samostatně. Během soutěže není dovoleno robotu jakkoliv navigovat, ať už hlasově či pomocí bluetooth.

Měří se čas okamžiku, kdy robot poprvé protne startovací linii (paprsek světelné závory). Do chvíle, kdy robot poprvé projede cílovou linií (paprskem světelné závory v cílovém prostoru (cca 40 x 40 cm)). Světelné závory vyčnívají cca 1 cm ze stěn ve výšce cca 5cm.

Samotná soutěž

Soutěž se skládá ze 2 nezávislých kol, počítá se lepší výsledek. Vítězem se stává robot, který se dostane nejrychleji do cíle a po cestě zaveze překážky do depa. Maximální čas na trati jsou dvě minuty. Mezi dvěma koly je dovoleno upravovat software a měnit baterie, či přestavět robota. Není však dovoleno jakkoliv měřit soutěžní plochu. Trať je možno testovat (nikoliv měřit) jednu hodinu před startem

V případě porušení pravidel je tým vyloučen ze soutěže.

Konstrukční řešení robota

Konstrukční řešení robota pro tuto úlohu je na první pohled složitější. K vyřešení úlohy bylo využito těchto dílů:

- Kostka NXT
- Stavební díly stavebnice LEGO MINDSTORMS
- 3x motor
- 2x dotykový senzor
- 1x ultrazvukový senzor
- 1x světelný senzor.

Konstrukce robota je zobrazena na Obr. 4.8. Je to opět jedna z mnoha možných. Klade si za cíl pomocí umístění a volby senzorů úspěšně splnit zadanou úlohu. Ultrazvukový senzor k měření vzdálenosti buse sloužit jak k detekci překážek v prostoru, tak i k lokalizování cílového prostoru pro umístění překážek.

Světelný senzor umístěný v těle robota slouží k vedení robota po černé čáře umístěné na soutěžním plánu. Dotykové senzory na konci „klepet“ slouží k detekci stěny.

V konstrukci robota jsou využity všechny tři možné výstupní porty. Dva z nich slouží pro ovládání pohybových jednotek robota a třetí pak k ovládání uchopovací části - „klepet“.

Z obrázku je patrná překážka při konstrukci robota a tou jsou propojovací kabely. Mohlo by se zdát, že jejich délka konstrukci nijak neomezuje, ale opak je pravdou. Délka kabelů ve většině případů robota omezuje v jeho pohybu a práci na soutěžním plánu. Pro konkrétní řešení tady navrhuji uživateli vyrobit kabely vlastní délky.



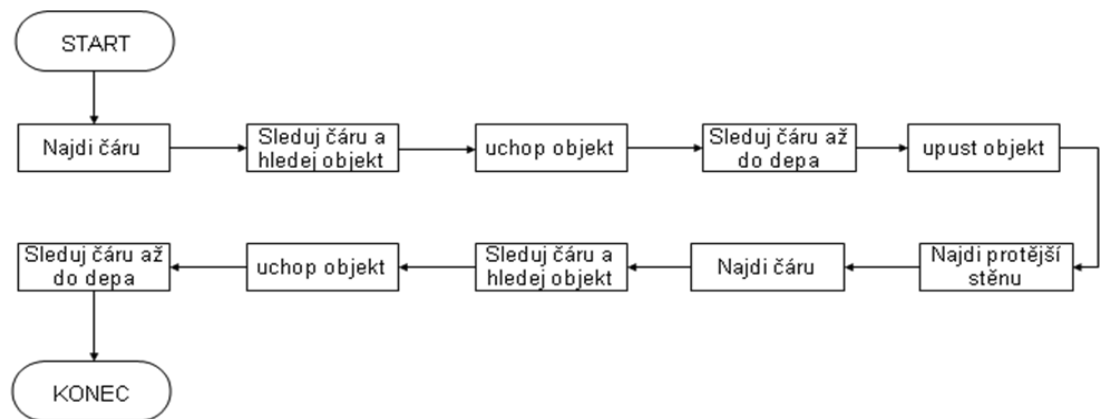
Obr. 4.8: konstrukce robota pro úlohu "překážky". Pohled z boku a zepředu

Programovací řešení robota

K programování robota byl použit firmware LEJOS NXJ pro programování v jazyce Java.

Vývojový diagram na Obr. 4.9 ukazuje způsob řešení úlohy. Úloha byla rozčleněna na několik podúloh, které jsou řešeny jako postupně jdoucí funkce v hlavním vlákně. Některé funkce se v průběhu programu opakují, a proto blíže rozeberu jen jednu z nich. Kde je potřeba, uvádím vývojový diagram.¹

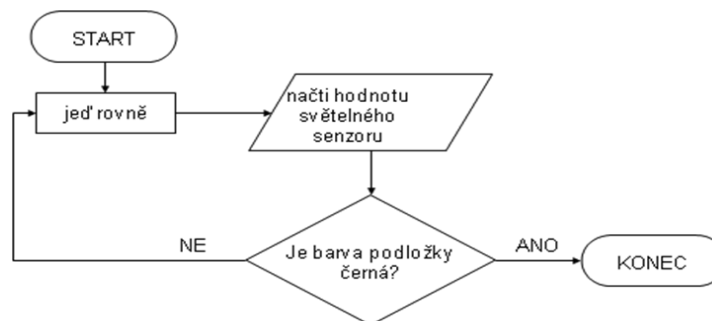
¹ Popsané programátorské řešení se může jemně lišit od skutečného zdrojového kódu. Neuvádím zde většinu výpisů na obrazovku během programu, protože jsou pro běh programu nepodstatné a věnuji se hlavně náročnějším funkcím a úkonům důležitým pro splnění úlohy.



Obr. 4.9: Vývojový diagram programu pro splnění úlohy "překážky"

Hledání černé čáry

Nejdříve musí robot nalézt černou čáru. Způsob řešení nastiňuje diagram na Obr. 4.10.



Obr. 4.10: Diagram funkce "najdi čáru"

K pohybu robota využijeme třídu „Pilot“ a její metodu „travel“, které jsou již pro nás připravené a zajišťují pohyb robota vpřed:

```
pilot.travel(20,true);
```

Pro zjištění pohybu robota vpřed využijeme opět metody třídy „Pilot“ vracející booleovskou hodnotu.

```
while(pilot.isMoving()){ ...}
```

Pro načtení hodnoty ze senzoru využijeme další z připravených metod:

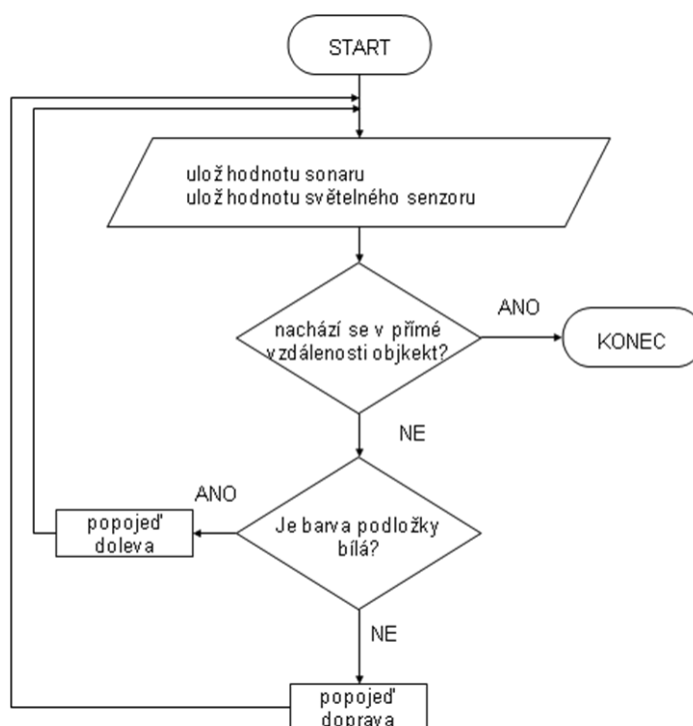
```
int sensor = light.readNormalizedValue();
```

A pokud tato hodnota bude odpovídat černé barvě, robot se zastaví (zajišťuje opět metoda třídy „Pilot“)

```
if (sensor<350) pilot.stop();
```

Sledování černé čáry

Poté, co robot našel černou čáru, tak ji sleduje do té doby, než nalezne překážku. V případě doručení překážky do depa potom robot sleduje tak dlouho černou čáru, dokud se nenachází v určité vzdálenosti od stěny depa, což je obdobné řešení jako v případě hledání překážky. Postup ukazuje vývojový diagram na Obr. 4.11.



Obr. 4.11: Funkce "sleduj čáru a hledej objekt"

Uložení hodnoty senzorů stejně tak jako přístup k hodnotě probíhá obdobně jako v předchozím případě. Individuální přístup k ovládání motorů pak vypadá následovně:

```
MotorPort.A.controlMotor(power, direction);
```

Výpis na display je vhodný pro zjišťování momentálního stavu robota. Výpis textu vypadá obdobně.

```
LCD.drawInt(value, radek_displaye, sloupec_displaye);
```


Stejně tak je možné vymazat display, či obnovit hodnotu na něm zobrazenou:

```
LCD.clear();  
  
LCD.refresh();
```

Uchopení a uvolnění objektu

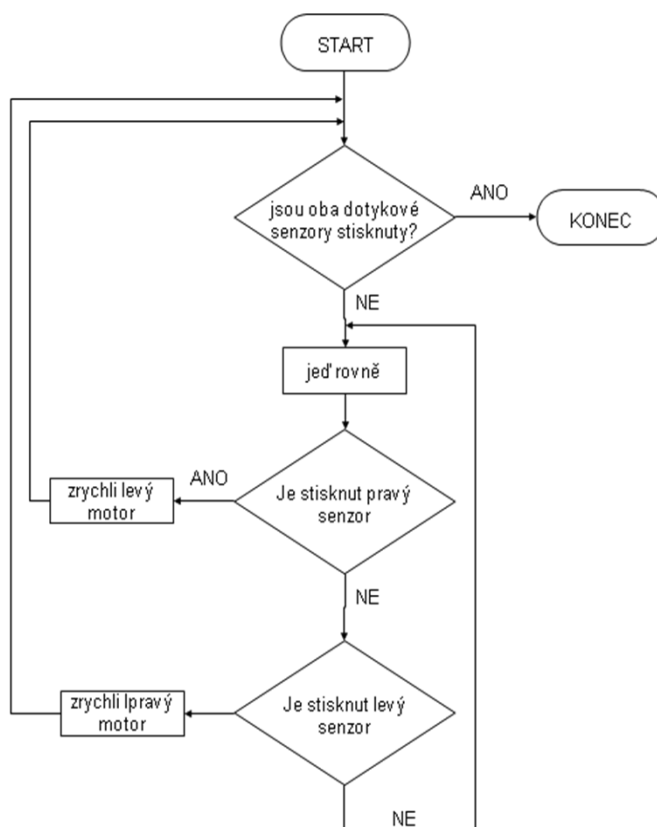
Pro tyto úkony nám bude stačit pouze využití motoru C, který pohání uchopovací část robota:

```
Motor.C.rotate(95);  
Motor.C.stop();  
Motor.C.rotate(-95);  
Motor.C.stop();
```

Nalezení stěny naproti depu

Podúlohu nalezení stěny naproti depu jsem zařadil do řešení proto, aby robot nehledal druhý objekt v prostoru náhodně, ale aby mohl opět využít toho, že se objekt nachází na černé čáře. Protože se může nacházet na určitém prostoru, začneme čáru prohledávat od samotného počátku, který nalezneme právě díky kontaktu se stěnou, kde čára začíná.

Robota nejdříve vyvedeme z depa dle již známých funkcí pro přístup k ovládní motorů. Dále se již snažíme o kolmý kontakt se stěnou, kde čára začíná. (Obr. 4.12).



Obr. 4.12: Diagram funkce pro kolmý kontakt se stěnou

Tato funkce obsahuje přístupy k senzorům, tentokrátě dotykovým, a standardní ovládání motorů využité již výše.

```

static TouchSensor touchR = new TouchSensor(SensorPort.S3);
boolean touchRight = touchR.isPressed();
    
```

Další funkce jsou již jen obdobou předcházejících, a proto je zde neuvádím. Celý kód je uveden v elektronické příloze.

Závěr k úloze „Překážky“

Návrh úlohy byl úspěšně proveden, včetně jejího programového a konstrukčního řešení. Na rozdíl od první úlohy, nebyla tato úloha vyzkoušena na reálném modelu. Díky experimentální práci se senzory by bylo potřeba programátorské řešení pro konkrétní soutěžní model ještě doladit. Odkoušení programu v domácích podmínkách ukázalo, že návrhované řešení je funkční.

Tuto úlohu hodnotím jako poměrně obtížnou a její řešitel musí být zkušenější programátor a znát možnosti ovládání robota.

Jazyk JAVA umožňuje objektově orientovaný přístup. Díky možnosti programování v prostředí Eclipse se stalo programování o mnoho příjemnější než ve zbylých dvou jazycích. Zejména snadné debugování programů a přehlednost kódu dělají práci mnohem efektivnější. Stejně tak se dá s úspěchem využít většiny Javovských tříd. Pokud bych se věnoval programování robota i nadále, rozhodně bych pokračoval v práci v tomto jazyce, o kterém píše více v kapitole 3.

4.5. Úloha číslo 3: „Čára“

Návrh této úlohy byl připraven jako zcela nová úloha pro předmět ROBOTI. Jedná se o úlohu, kde je opět využito senzorů a pohybových součástí robota. Pro tuto úlohu byl zvolen jazyk NXT-G jako ukázka další programovací možnosti.

Pravidla

Cíl úlohy

Postavit a naprogramovat robota tak, aby samostatně, bez jakékoliv další pomoci projel stanovenou dráhu podél černé čáry vyznačené na podložce a zároveň z podložky nesjel.

Vybavení

Každý tým obdrží na začátku soutěže základní soupravu LEGO® MINDSTORMS® Education (9797), soupravu technických dílů (9648 - doplňkové pasivní díly) a síťový adaptér (9833), které po skončení soutěže vrátí v kompletním stavu organizátorovi soutěže.

NXT kostku lze programovat v libovolném programovacím jazyce. Při programování NXT kostky a testování robota lze využít k napájení akumulátor, vlastní baterie nebo síťový adaptér. Při vlastní soutěži je možné použít pouze akumulátor nebo baterie.

Soutěžní plán

Podrobnosti k průjezdu bludištěm

Vlastního robota lze sestavit naprosto libovolně, ovšem pouze s využitím dílů ze zapůjčených souprav. Tento robot by měl samostatně a co možná nejrychleji projet bludiště ze stanoveného startovního prostoru (STARTU) do stanoveného cílového prostoru (CÍLE).

Robota lze položit do startovního prostoru (cca 40 x 40 cm) ručně. V případě porušení pravidel je tým okamžitě vyloučen ze soutěže.

Čas se měří od okamžiku, kdy robot poprvé protne startovací linii (paprsek světelné závory). Do chvíle, kdy robot poprvé projede cílovou linií (paprskem světelné závory v cílové prostoru (cca 40 x 40 cm)).

Po průjezdu cílovou linií je třeba dodržet následující podmínku: robot musí po opuštění čáry do 10ti vteřin zastavit a zároveň nespadnout z podložky, která končí v těsné blízkosti cílového prostoru.

Rozměry soutěžního plánu

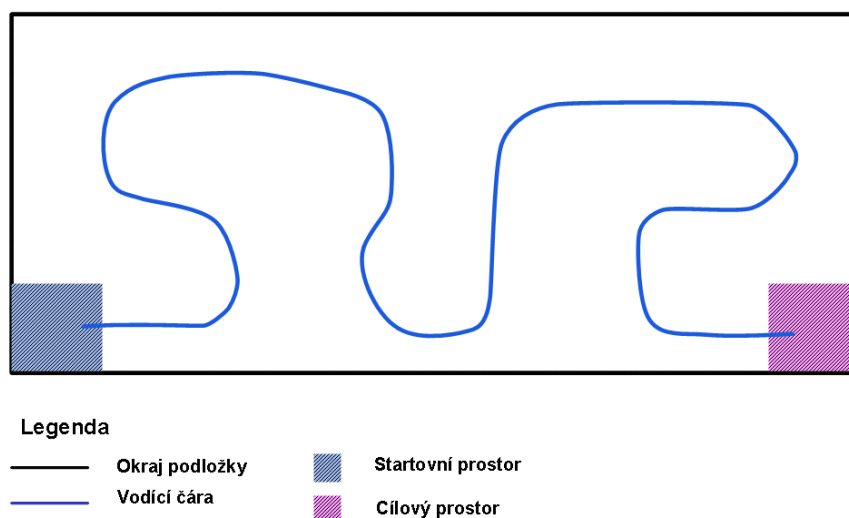
Celková délka čáry se pohybuje od jednoho do tří metrů. Čára může být jakkoliv zakroucená, neobsahuje však křížení trati. Minimální poloměr zatáčky je 20 cm. Soutěžní plán nemá žádné stěny bránící pádu robota z podložky

Světelné závory jsou umístěny v prostoru na podložce a nebrání robotovi v průjezdu podél čáry.

Celkový rozměr podložky je 200 x 90 cm a podkladem je laminátová deska.

Uspořádání soutěžního plánu

Níže uvedený soutěžní plán je pouze jeden z možných příkladů jeho uspořádání pro vlastní soutěž. Může dojít jak ke změně polohy startovací a cílové oblasti, tak i ke změně uspořádání čáry.



Obr. 4.13: Schema soutěžního plánu pro úlohu "čára"

Průjezd podél čáry

Robot může bludištěm projet podél čáry libovolným způsobem (počítá se pouze celkový čas průjezdu). Robot se nesmí odchýlit od sledované čáry více jak 20cm. V případě většího odchýlení je robot diskvalifikován ze soutěže.

Maximální čas průjezdu je stanoven na 2 minuty. Po překročení tohoto času je robot rovněž vyřazen ze soutěže.

Vlastní soutěž

Vlastní soutěž bude probíhat dvoukolově (výsledkem bude lepší čas z obou kol). Jednu hodinu před začátkem soutěže bude umožněno týmům testovat robota na variantě stejného bludiště, jaké bude použito pro vlastní soutěž. V tomto čase však není dovoleno provádět jakékoliv měření jeho rozměrů.

Mezi oběma koly je možné vyměnit (nebo dobít) akumulátor či baterie a je také přípustné provést případné změny, opravy či úpravy software.²

² Popsané programátorské řešení se může lišit od skutečného zdrojového kódu. Neuvádím zde většinu výpisů na obrazovku během programu, protože jsou pro běh programu nepodstatné a věnuji se hlavně náročnějším funkcím a úkonům důležitým pro splnění úlohy.

Konstrukční řešení

Konstrukční řešení je opět voleno jako co nejjednodušší pro úspěšné splnění úlohy. Na Obr. 4.14 je možné pozorovat umístění dvou hlavních senzorů. Jeden světelný, pro sledování černé čáry, a druhý dotykový, pro sledování povrchu desky, respektive vnímající její opuštění.

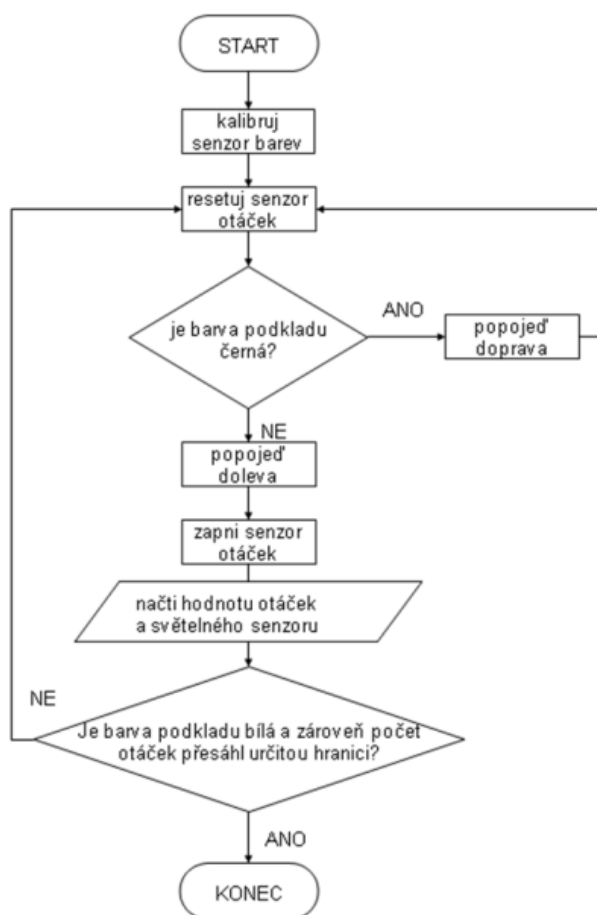


Obr. 4.14: konstrukční řešení robota pro úlohu "čára"

Programátorské řešení

Vývojový diagram pro úlohu „Čára“

Programátorské řešení opět začínáme vývojovým diagramem (Obr. 4.15), který názorně ukazuje průběh programu.

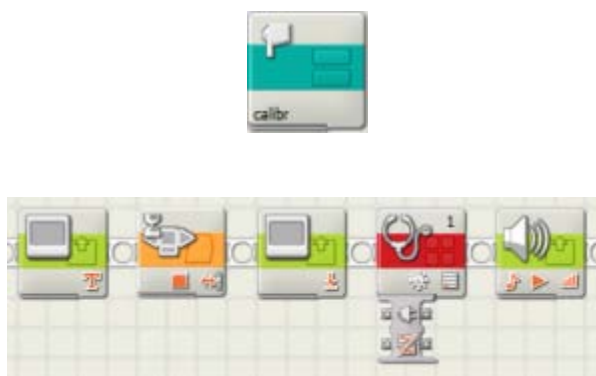


Obr. 4.15: Vývojový diagram pro úlohu "čára"

Samotné programátorské řešení

Zpřehledňování kódu

Jazyk NXT-G nabízí uživateli vytváření jakýchsi funkcí, neboli balíčků kódu, které se mohou v průběhu programu opakovat a proto je příjemnější je pro přehlednost zabalit a v případě potřeby si otevřít jen příslušný bloček s danou funkcí. Takováto ukázka je na Obr. 4.16.



Obr. 4.16: Vlastní funkce

Kalibrace senzorů

Přestože jsme prahovou hodnotu pro světelný senzor v předchozích úlohách určovali experimentálně či za pomoci technické dokumentace k senzoru, jazyk NXT-G přímo nabízí grafický bloček pro jeho kalibraci. Postupně tedy sejmeme světelným senzorem barvu čáry a barvu okolí, po kterém se bude robot pohybovat. Část této kalibrace je znázorněna na Obr. 4.16.

Bločky, u kterých je možné velmi intuitivně zadávat bližší parametry, jeden po druhém postupně slouží k výpisu textu na obrazovku, čekání na stisk tlačítka na klávesnici kostky NXT, zápisu dat a reprodukci zvukového znamení.

Smyčka běžící do opuštění podložky robotem

Po kalibraci senzorů přichází samotné tělo programu. Dle zadání úlohy robot během svého pohybu po podložce nesmí z podložky spadnout. Toto zajistíme smyčkou hlídající hodnotu dotykového senzoru:



Obr. 4.17: základní smyčka programu "čára"

V této smyčce, hlídající kontakt robota s podložkou, bude poté zbytek programu.

Provedeme resetování rotačního senzoru, který hlídá kontakt s černou čárou.

Závěr k úloze „Čára“

Úspěšně byl vytvořen návrh zadání a řešení soutěžní úlohy „čára“. Tato úloha pro řešitele zkušeného v práci s robotem není neúměrně obtížná. V našem řešení byl využit jazyk NXT-G. Jazyk NXT-G je první pohled jako velmi příjemný a názorný jazyk, který pracuje s grafickými bločky.

Z mé vlastní zkušenosti práci v tomto programovacím prostředí nedoporučuji, vzhledem k tomu, že je více časově náročná a nenabízí takové možnosti, jako ostatní programovací jazyky. Je zde velmi obtížná práce s proměnnými, funkcemi a metodami. Při větvení kódu se kód stává méně přehledným a není snadné jej spravovat. Proto bych jeho využití doporučil jen pro jednoduché programy.

Pokud by tato úloha byla programována v jiném jazyce, pak ji hodnotím jako nejjednodušší variantu ze všech tří uvedených.

4.6. Závěr k oddílu zabývající se návrhem soutěžních úloh

Úspěšně byly provedeny tři návrhy soutěžních úloh. Jejich zpracováním byly ukázány některé výhody a nevýhody konstrukčních řešení a programátorských přístupů. Robot LEGO MINDSTORMS nabízí více možností programování. Zde byly ukázány jen tři nejběžnější způsoby.

Pro začínající programátory doporučuji jazyk NXC, běžící na standardním firmwaru, který je zdarma a je velmi jednoduchý na pochopení.

Širší možnosti a lepší údržbu programů nabízí programování v jazyku JAVA běžícím na Firmwaru LEJOS NXJ. Tento jazyk je složitější na instalaci a pochopení, ale nese tu výhodu, že se s ním již většina programátorů setkala a proto pro ně bude programování robota v tomto prostředí poměrně snadné.

Grafický programovací jazyk NXT-G je pro začínajícího programátora velmi přehledný, ale jeho možnosti jsou poměrně omezené a k pokročilejšímu programování robota bych ho nepoužil.

5. WEBOVÉ STRÁNKY K VÝUCE PŘEDMĚTU ROBOTI

5.1. Obsah páté kapitoly

Cílem tohoto oddílu je informovat uživatele webových stránek o jejich poslání (kap. 5.2), struktuře a obsahu. Budoucí administrátoři stránek by zde měli nalézt možnosti jejich editace a rozšiřování.

5.2. Cíl webových stránek

„A3B99RO ROBOTI“ je nový předmět připravovaný Katedrou řídicí techniky, Katedrou kybernetiky a Katedrou měření na Fakultě Elektrotechnické ČVUT. Cílem webových stránek je poskytnout studentům informace o tomto předmětu, usnadnit vyhledávání informací pro výuku a informovat o aktivitách spojených s Robotem LEGO MINDSTORMS na zmíněných katedrách. Dalším z cílů webových stránek je nastínění možností využití robota a v neposlední řadě mohou webové stránky reprezentovat Fakultu Elektrotechnickou ČVUT tím způsobem, že budou zajímavě působit na potenciální uchazeče o studium na této fakultě. Možnost volby anglické verze webu rozšiřuje možnosti působení na větší počet uživatelů.

Webové stránky neumožňují administrační přístup a jediný možný způsob jejich editování je ruční vkládání a změna textu.

Webové stránky jsou přístupné na adrese **<http://support.dce.felk.cvut.cz/roboti>**.



Obr. 5.1: náhled webové stránky

5.3. Základní údaje

Softwarové nároky na uživatele

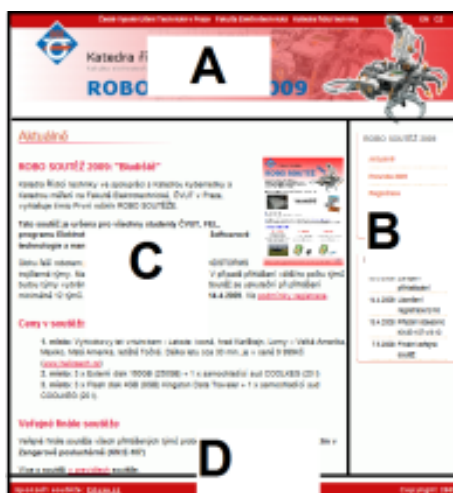
Webové stránky jsou optimalizovány pro prohlížeče Mozilla Firefox, Opera Browser a Internet Explorer verze 6 nebo vyšší. Prohlížeč musí podporovat přehrávání videí, tedy mít nainstalovaný přehrávač Flash Player.

Použité technologie

Webové stránky jsou vytvořeny jazykem HTML verze 4.01. Pro dynamickou část webových stránek je použit skriptovací jazyk PHP [13]. Pro popis vzhledu a formátování stránek je zvolen jazyk CSS [14]

Základní části zobrazení

Webová stránka se dělí na 4 části definované v kaskádových stylech. Jednotlivé části jsou znázorněny na Obr. 5.2. Tyto části jsou: A: Hlavička, B: Menu, C: Hlavní okno, D: Patička. Všechny části jsou dynamické a mění se dle parametrů volených uživatelem (volba jazyka a aktuálního obsahu stránky).



Obr. 5.2: Rozvržení webové stránky

A. Hlavička

Tento oddíl představuje titulek stránky a nese hlavní informaci o tom, co uživatel na stránkách nalezne. Ve vrchní části hlavičky se nachází určení stránek, respektive jejich příslušnost k Fakultě Elektrotechnické na ČVUT v Praze. Je zde také možnost volby jazyka.

B. Menu

Prává část webové stránky zobrazuje menu v závislosti na aktuálním obsahu stránky a jazyce, ve kterém je zobrazena. Dále jsou zde pak uvedena důležitá data či informace.

C. Hlavní okno

V hlavním okně se zobrazuje obsah volený v Menu po pravé straně.

D. Patička

Patička slouží jako informační lišta, kde jsou uvedeny informace, které nejsou aktivně vyhledávány uživateli, ale přesto nesou informace, které stránky doplňují.

5.4. Obsah stránek

V této kapitole popisují položky hlavního menu včetně nástinu informací v nich obsažených. Informace jednotlivých položek se průběžně mění. Současná verze je k datu 20. 4. 2009. Vzhled menu je znázorněn na Obr. 5.3.



Obr. 5.3: Hlavní menu webové stránky

Aktuálně

Stránka Aktuálně obsahuje aktuální informace týkající se problematiky spojené s Robotem LEGO MINDSTORMS. Stránka si také klade za cíl uživatele zaujmout a proto je tu i vloženo i motivační video s ukázkou možností robota.



Obr. 5.4: Ukázka vložené úvodní stránky

Organizace

Tato část webu informuje studenty o průběhu výuky předmětu a jeho organizaci. Text, přestože je pro předmět rozhodující, zde pro jeho obsáhlost neuvádím.

Programovací prostředky

Nezbytnou součástí této práce a stejně tak i webu je informování uživatelů o rozsáhlých možnostech programování robota. Stejně tak jako v kapitole 3, i na webu uvádím návod, jak začít se třemi základními programovacími prostředky. K tomu, jak se dostat k dalším možnostem, je samozřejmě zájemcům poskytnut návod.

Odkazy

Tato část menu čerpá hlavně ze seznamu zdrojů uvedených na konci této práce a opět zde obsah pro duplicitnost neuvádím.

ROBO SOUTĚŽ 2009

Tato položka menu, zdůrazněná kapitálkami odkazuje mimo tuto část webu na aktuální problematiku, v tomto případě na druhou část webu věnující se soutěži, která právě na Katedře řídicí techniky probíhá.

5.5. Struktura webu

[..] – Kořenový adresář

- index.php
- inc_head.php
- inc_page.php
- inc_menu.php
- inc_foot.php

- content_cs.php
- content_organization_cs.php
- content_programming_cs.php
- content_links_cs.php
- content_news_cs.php
- content_order_cs.php
- content_photogallery_cs.php
- content_race_cs.php
- content_registration_cs.php
- content_rules_cs.php

[Files]

[Img]

5.6. Obsah jednotlivých adresářů a funkčních souborů

[..] – Kořenový adresář

Vzhledem k tomu, že struktura webu není nijak složitá, nachází se většina souborů právě v kořenovém adresáři. Obsahem tohoto adresáře soubory tvořící webovou prezentaci předmětu Roboti a Robo soutěže 2009. Vzhledem k tomu, že je web vytvořen dvojjazyčně, má každý soubor obsahující prezentované informace svůj ekvivalent v cizím jazyce. Tento ekvivalent je charakterizován koncovkou `_cs` nebo `_en` v závislosti na zvoleném jazyce. V tomto případě českém a anglickém.

Soubor `index.php`

Soubor **index.php** je první stránka, která se uživateli načte při zadání webové adresy těchto stránek. Jedná se o soubor, přes který vkládáme obsah ostatních stránek (hlavního okna) a kde nastavujeme proměnné využívané během pohybu na webu.

Popis kódu obsaženého v souboru `index.php`

```
session_start();
```


Zde se jedná o zahájení práce se „session“. Session neboli relace umožňují přesnou identifikaci uživatele na serveru, udržují informace o stavu stránek a práce uživatele s nimi. Nám pomohou při práci s jazykovým nastavením stránek.

```
if (!isset($_SESSION['lang'])) {  
    $_SESSION['lang']='cs';  
}
```

Zde nastavujeme jazykové nastavení. Při vstupu na web žádné nastavení uživatelem nastaveno není, a proto musíme na začátku implicitně nějaké nastavit. Implicitně se nastavuje česká jazyková verze úvodní stránky.

```
if (isset($_GET['id'])) {  
    if ($_GET['id']=="race") {  
        $_SESSION['race']=1;  
    }  
    $TARGET="_" . $_GET['id'];  
} else {  
    unset($_SESSION['race']);  
}
```

Touto částí php kódu ošetřujeme implicitní načítání úvodní stránky při vstupu na webové stránky Roboti.

```
include("inc_lang.php");
```

Dále je důležité načíst soubor, ve kterém jsou uloženy proměnné nastavující jazykovou verzi malých částí vkládaných textů mimo hlavní okno. Tyto vkládané texty se nacházejí například v hlavičce, patičce, nebo v menu. Zde vidíme příklad:

```
$KATEDRA['cs']="Katedra řídicí techniky";  
$KATEDRA['en']="Department of control engineering";
```

Po nastavení proměnných načteme jednotlivé části webové stránky.

Soubor inc_head.php

Soubor head.php obsahuje především HTML hlavičku a začátek těla dokumentu. Jedná se o standardní HTML hlavičku, proto zde nebudu popisovat zdrojový kód.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs" lang="cs">
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<meta http-equiv="content-language" content="cs-CZ" />
<meta name="author" content="Keny Otter; keny@nativepeople.cz"
title="Kdo na to šáhne, tomu uvalím pracky!" href="./index.php" />
<title>ROBOTI | ČVUT - Katedra řídicí techniky
</title>
<link rel="SHORTCUT ICON" type="image/x-icon" href="/favicon.ico" />
<link rel="stylesheet" type="text/css" href="roboti.css"
media="screen" />
<!--
<link rel="alternate" type="application/rss+xml"
ref="http://cvut.mcdev.cz/rsshandlerext.aspx?exportId=1&dontparse=false" title="RSS kanál ČVUT" />
-->
</head>
```

Soubor inc_page.php

Soubor inc_page.php se stará o načtení vrchní části webové stránky. Dále je v něm možnost jazykové volby, která ovlivní jak načtení kaskádového stylu, tak i zobrazení prvků po celé stránce.

Část kódu starající se o nastavení jazykových parametrů:

```
<?php
    if(!isset($_GET['id'])) {
        echo '<a href="index.php?lan=en">EN </a>';
    }else{
        $id = $_GET['id']; echo '<a
href="index.php?id='.$id.'&lan=en">EN</a>';
    }
?>

|

<?php
    if(!isset($_GET['id'])) {
        echo '<a href="index.php?lan=cz">CZ </a>';
    }else{
        $id = $_GET['id']; echo '<a
href="index.php?id='.$id.'&lan=cz">CZ</a>';
    }
?>
```

Soubor inc_menu.php

Soubor inc_menu.php se stará o zobrazení novinek, které jsou vkládány přímo do HTML kódu. Hlavní funkci zde představuje menu, díky kterému si uživatel volí z nabídky obsahu hlavní strany.

Část kódu obsluhujícího uživatelskou volbu obsahu:

```
<ul>
<li><a href="index.php?id=aktualnec
"title="Aktuálně">Aktuálně</a></li>
<li><a href="index.php?id=organizace"
title="Organizace">Organizace</a></li>
<li><a href="index.php?id=programovani" title="Programovací
prostředky">Programovací prostředky</a></li>
<li><a href="index.php?id=odkazy" title="Odkazy">Odkazy</a></li>
<li><a href="./soutez/index.php" title="ROBO SOUTĚŽ 09"><b>ROBO
SOUTĚŽ 2009</b></a></li>
</ul>
```

Zřetelné je přiřazení speciální hodnoty proměnné „id“ při jednotlivých volbách v menu. Tato hodnota je předána prohlížečem v adrese URL. Tuto hodnotu využívá soubor index.php, který na jejím základě načítá hlavní obsah stránky.

Soubor inc_foot.php

Soubor inc_foot.php obsahuje pouze HTML kód pro zobrazení spodní informační lišty.

Dalšími adresáři v kořenovém adresáři jsou adresáře obsahující hlavní informace vkládané do hlavního okna webové stránky. Vložení hlavního obsahu se provádí v závislosti na parametrech:

```
include("content".$TARGET."_".$_SESSION['lang'].".php");
```

Kdy jako \$_TARGET se nám načítá aktuální id stránky a \$_SESSION se nám načítá jako aktuální jazykové nastavení.

- **content_cs.php** – úvodní stránka, která je načítána automaticky jako první, na kterou uživatel vstoupí. Obsahuje aktuální informace.
- **content_organization_cs.php** – stránka obsahující informace o organizaci předmětu
- **content_programming_cs.php** – stránka obsahující informace o možnostech programování robota.

- **content_links_cs.php** – stránka obsahující odkazy obsahující další materiály a informace k tematice.
- **content_news_cs.php** – stránka obsahující novinky k předmětu Roboti
- **content_order_cs.php** – pořadí soutěží v Robo soutěži
- **content_photogallery_cs.php** – stránka obsahující fotodokumentaci práce s roboty
- **content_race_cs.php** – aktuální informace o Robo soutěži
- **content_registration_cs.php** – informace o možnostech přihlášení do Robo soutěže
- **content_rules_cs.php** – stránka obsahující pravidla soutěže

- **roboti.css** – kaskádový styl tvořící vzhled stránek.

Každý ze souborů kromě souboru index.php má navíc svůj ekvivalent i v anglické verzi, charakterizovaný koncovkou „_en“.

[Files]

Adresář obsahující dokumenty a obrázky vkládané na stránky

[img]

Adresář obsahující obrázky tvořící vzhled stránek

5.7. Styly

Jazyk CSS zjednodušuje popis formátování webových stránek napsaných v HTML kódu. Umožňuje oddělení obsahu stránek od popisu jejich formátování a tím celou tvorbu webu zpřehledňuje. Tato technologie byla použita i při výrobě těchto webových stránek.

Formátování rozložení stránky

Pomocí kaskádových stylů je možné naformátovat rozložení jednotlivých prvků na stránce a i tyto webové stránky toho využívají.

O formátování jednotlivých prvků na stránce se starají položky:

```
body {}, #wrapper {}, .page {}, div#column-left {}, div#column-right {}, #foot {}
```

Tyto položky jednotlivé prvky na stránce zarovnávají a udávají pro ně základní vlastnosti.

Formátování textu, tabulek, seznamů

Díky využití vlastností CSS je za pomoci jednotlivých identifikátorů (značených #) a třídami (uvozenými „.“), je možné specifikovat různé formátování různých prvků v různých částech stránky.

Příkladem formátování odkazu v horní části stránky může být:

```
.page a{
    text-align: center;
    font-weight: bold;
    color: #ffffff;
    text-decoration: none;
}
```

5.8. Závěr k páté kapitole

Webové stránky k výuce předmětu ROBOTI jsou první verzí propagace tohoto předmětu. Stejně tak jako vývoj úloh a vývoj výukových metod pro práci s robotem, i tento web je připraven se vyvíjet. Tyto stránky se dne 20. 3. 2009 úspěšně podařilo spustit a od té chvíle informuje své návštěvníky a přizpůsobuje se jejich potřebám.

Jelikož je Robot LEGO MINDSTORMS poměrně nový výukový prostředek, stránky se o něm snaží uživatelům poskytnout potřebné informace a motivovat je k jeho bližšímu prozkoumání.

Webové stránky jsou přístupné na adrese <http://support.dce.felk.cvut.cz/roboti/>.

Poděkování za pomoc a práci na webových stránkách patří panu Aleši Kapicovi.

6. ZÁVĚR

Úspěšně se podařilo splnit všechny body zadání bakalářské práce, tedy popsat možnosti využití Robota LEGO MINDSTORMS při výuce. Vytvořená práce obsahuje informace, které čtenáře obohatí ve směru poznání jeho současného využití, hardwarového zpracování, programovacích možností a příkladech úloh, které jsou v rámci práce s robotem možné navrhnout. Tyto poznatky jsou zároveň shrnuty na webových stránkách (<http://support.dce.felk.cvut.cz/roboti>), které byly vytvořeny pro předmět Roboti. Tato práce také poskytuje přehled využití robota na Katedře řídicí techniky, Fakultě Elektrotechnické, Českého Vysokého Učení Technického v Praze.

Celá práce je tvořena čtyřmi samostatnými kapitolami, které zkoumanou problematiku rozebírají a podrobně popisují. V druhé kapitole je uveden popis vývoje robota, včetně informací, jakým způsobem lze s robotem pracovat. Popis Hardwarového vybavení pak poskytuje informace o inteligentní kostce NXT, která je jádrem celého systému, a stejně tak o senzorech, jejichž využití je při práci s robotem téměř nezbytné. Třetí kapitola nabízí přehled programovacích možností a uvádí autorovu osobní zkušenost se třemi konkrétními programovacími jazyky včetně shrnutí jejich výhod a nevýhod. Čtvrtá kapitola pak nabízí tři konkrétní úlohy navržené pro řešení komplexních problémů včetně nástinu jednoho z mnoha možných řešení. Pátá kapitola poté uvádí dokumentaci k webovým stránkám, které byly vytvořeny pro předmět Roboti - A3B99RO (<http://www.fel.cvut.cz/education/bk/predmety/12/57/p12579604.html>). Práci obohacuje seznam zdrojů, ze kterých bylo při jejím vytváření čerpáno a které jsou k dispozici zájemcům o bližší studium problematiky.

SEZNAM POUŽITÝCH ZKRATEK

PID regulátor: regulátor obsahující proporcionální, integrační a derivační složku

PWM: pulzní šířková modulace

LED: elektroluminiscenční dioda

SEZNAM POUŽITÝCH ZDROJŮ

Monografie

- [1] LEGO group, LEGO MINDSTORMS NXT Hardware Developer Kit, 2006
- [2] GASPERI Michael and Philippe “Philo” Hurbain with Isabelle Hurbain, Extreme NXT: Extending the LEGO MINDSTORMS NXT to the Next Level, Apress 2007
- [3] LEGO® MINDSTORMS® Education , NXT UserGuide, 2006
- [4] KNUDSEN Jonathan B, The Unofficial Guide to LEGO MINDSTORMS Robots, O’Reilly & Associates, Inc., 1999
- [5] MORAL J.A.Brena, Develop leJOS Programs Step by Step, 2008
- [6] Brian Bagnall, Maximum Lego NXT: Building Robots with Java Brains, Variant Press, 2007
- [7] Ferrari Giulio, Programming LEGO Mindstorms with Java, SYNGRESS, 2002
- [8] Butka, Jason, Installing Lejos on the NXT Mindstorm Using Eclipse IDE and Introduction to Lejos/Java programming, Lawrence Technological University, Fall 2008
- [9] Daniele Benedetteli, Programming LEGO NXT Robots using NXC., June 7 2007
- [10] Hansen, John C., NXT POWER PROGRAMMING, Variant Press, 2008
- [11] Kelly, James Floyd, LEGO Mindstorms NXT-G Programming Guide, Technology in action press, 2007
- [12] Owen Bishop, Programming LEGO MINDSTORMS NXT, Syngress, 2006
- [13] LERDORF Rasmus, Programming PHP, Second Edition, O’Reilly, 2006
- [14] MEYER, Eric, A., CSS:The definiteve Guide, Third Edition, O’Reilly, 2007.

Bakalářské a diplomové práce:

- [15] MATUŠŮ, Roman. Ovládání laboratorního modelu robota Mindstorms. Zlín 2008, Diplomová práce.(Ing.) Univerzita Tomáše Bati ve Zlíně, Fakulta Aplikované Informatiky
- [16] REICH, Jakub. Ovládání laboratorního modelu Mindstorms NXT (spike) pomocí PC. Zlín, 2008. diplomová práce (Ing.). Univerzita Tomáše Bati ve Zlíně. Fakulta aplikované informatiky
- [17] PROCHÁZKA, Michal. Řízení robota MINDSTORMS NXT pomocí PC. Zlín, 2008. diplomová práce (Ing.). Univerzita Tomáše Bati ve Zlíně. Fakulta aplikované informatiky
- [18] PIKNER, Michal. Využití stavebnice Lego při výuce. Zlín, 2008. bakalářská práce (Bc.). Univerzita Tomáše Bati ve Zlíně. Fakulta aplikované informatiky

Internetové zdroje

- [19] Books about LEGO MINDSTORMS NXT [online]: LEGO MINDSTORMS (2009-03-09), <<http://mindstorms.lego.com/>>
- [20] LEGO Průduky [online]: LEGO.COM (2009-03-09), <<http://www.lego.com/>>
- [21] HiTechnic senzory k LEGO® MINDSTORMS® NXT [online] EDUXE, výhradní distributor edukačních materiálů LEGO education, (2009-03-09), <http://www.eduxe.cz/legomenu/hitechnic_menu.htm>
- [22] Archive for the 'software' Category [online] NXT.ORG, a LEGO MINDSTORMS NXT community (2009-03-09), <<http://www.nxtasy.org/>>
- [23] LEGO MINDSTORMS History [online] ROBOT, the latest in hobby, science and consumer robotics (2009-03-09), <http://botmag.com/articles/lego_mindstorms_robot_news_2.shtml>
- [24] NXT 2.0 [online] The NXT STEP - LEGO® MINDSTORMS® NXT Blog (2009-03-02), <<http://www.thenxtstep.blogspot.com/>>
- [25] FIRST Robotics Competition [online] OREGON robotics (2009-03-09), <<http://www.ortop.org/>>
- [26] NXT Sensor Guide [online] TEAM Hassenplug (2009-03-09), <<http://www.teamhassenplug.org/NXT/>>
- [27] HiTechnic Products and Accessories [online] HiTechnic Products: (2009-03-09), <<http://www.hitechnic.com/>>
- [28] C# [online] Mindsqualls (2009-03-09), <<http://www.mindsqualls.net/>>
- [29] NXT# - Mindstorms for .NET [online] (2009-04-04,)<<http://lego.fokke.net/>>

- [30] Mindstorms Cellular Bluetooth NXT Control [online], The Mindstorms NXT Review (2009-03-09). <<http://mindstormsnxt.blogspot.com/>>
- [31] Mindstorms NXT [online] Nxtbot.com, Covering the world of consumer and personal robotics (2009-03-09) <<http://nxtbot.com/>>
- [32] First Lego League [online], Robot Challenge, Periodic Posts about Lego (TM) Mindstorms and NXT(2009-03-09) <<http://robotchallenge.blogspot.com/>>
- [33] Problem Solving with Robots [online], Lego MINDSTORMS NXT (2009-03-09), <<http://www.noucamp.org/cp1/psol/NXT.html>>
- [34] Embedded Systems - Embodied Agents, Digital Control in an Physical World [online], (2009-04-28), <<http://www.legolab.daimi.au.dk/DigitalControl.dir/index.html>>
- [35] Intro to Robotics [online], (2009-04-27), <<http://pirate.shu.edu/~wachsmut/Teaching/CSAS-Robotics/index.html>>
- [36] Robot Lego Mindstorms [online], Legacek.cz (2009-05-20), <<http://hracky-lego.legacek.cz/lego-8527-mindstorms-next-mindstorms+dp38067/>>
- [37] O čem sní roboti: Recenze Lego Mindstorms NXT [online], Zive.cz (2009-05-20), <<http://www.zive.cz/Clanky/O-cem-sni-roboti-Recenze-Lego-Mindstorms-NXT/sc-3-a-133790/default.aspx>>
- [38] O čem sní roboti: Recenze Lego Mindstorms NXT [online], Computer 23/06 (2009-05-20), <<http://www.zive.cz/Obraz-a-zvuk/Vychazi-Computer-2306---nejlepsi-sestavy-a-GPS-navigace/sc-35-sr-1-a-133179/default.aspx>>

Hardware

- [39] The NXT motor [online], Philo's Home Page (2009-03-09), <<http://www.philohome.com/nxtmotor/nxtmotor.htm>>
- [40] NXT Motor [online], Ryo's Holiday, (2009-03-09), <http://web.mac.com/ryo_watanabe/iWeb/Ryo%27s%20Holiday/NXT%20Motor.html>
- [41] Sensors and Controllers for MINDSTORMS NXT [online], mindsensors.com, (2009-04-27), <http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=57&MMN_position=24:24>
- [42] LEGO MINDSTORMS NXT Blog [online], Die NXTe Ebene, (2009-04-27), <<http://dienxteebene.blogspot.com/>>
- [43] Robotics: NXT [online], Lugnet.com, (2009-04-27), <<http://news.lugnet.com/robotics/nxt/>>
- [44] Motocube [online], (2009-04-27), <<http://www.motocube.com/>>

[45] ExtremeNXT.com [online], (2009-04-27),
<<http://www.extremenxt.com/HBANGLE.HTM>>

Programovací jazyk LeJOS NXJ

[46] LeJOSm Java for LEGO MINDSTORMS, [online] oficiální stránky projektu (2009-04-06)
<<http://lejos.sourceforge.net/> >

[47] The leJOS NXJ Tutorial [online], LEJOS, Java for LEGO Mindstorms (2009-03-18)
<<http://lejos.sourceforge.net/nxt/nxj/tutorial/index.htm>>

[48] Tutorial on how to install and run Java on Lego Mindstorms NXT using Eclipse [online],
Christoph Bartneck, Ph.D. (2009-04-04) <<http://www.bartneck.de/2008/03/04/java-lego-nxt-eclipse-tutorial/> >

[49] LeJOS, Java for Lego Mindstorms [online], Jab, Juan Antonio Breña Moral (2009-04-04),
<<http://www.juanantonio.info/jab/cms.php?id=69>>

[50] NXJ API [online], Přehled javovských tříd a programovacího rozhraní pro NXT (2009-04-06), <http://lejos.sourceforge.net/p_technologies/nxt/nxj/api/index.html>

[51] Embedded Systems - Embodied Agents, Digital Control in an Physical World [online],
(2009-04-27), <http://legolab.daimi.au.dk/DigitalControl.dir/>

[52] Java Robotics Tutorials [online], (2009-04-27), <<http://www.ridgesoft.com/tutorials.htm>>

[53] LeJOS, Java for Lego Mindstorm [online], JAB (2009-05-20),
<<http://www.juanantonio.info/jab/cms.php?id=69>>

Programovací jazyk NXC

[54] NXC [online], Lukas's website (2009-04-27), <<http://lukas.internet-freaks.net/nxt.php>>

[55] Programming Solutions for theLEGO Mindstorms NXT [online], Botmag, (2009-04-27),
<http://www.botmag.com/articles/10-31-07_NXT.shtml>

[56] NXC [online],(2009-04-27), < <http://www.debacher.de/wiki/NXC>>

Programovací jazyk NXT-G

- [57] NXT-G Tips & Tricks [online], Steve Hassenplug, (2009-04-27),
<<http://www.teamhassenplug.org/NXT/NXTGTips/>>
- [58] NXT-G Blocks [online], nxtasy.org (2009-04-28),
- [59] <<http://nxtasy.org/repository/nxt-g-blocks/> >
- [60] NXT Tutorial [online], (2009-04-27),
<http://www.ortop.org/NXT_Tutorial/html/browser_settings.html>
- [61] NXT programs [online], (2009-04-27),
<<http://www.nxtprograms.com/help/software.html>>
- [62] NXT-G Program Guides [online], (2009-04-27),
- [63] <<http://nxt-gprogramguides-1.blogspot.com/>>

Další programovací jazyky

- [64] NXT#, Mindstorms for. NET [online], (2009-04-29), <<http://nxtsharp.fokke.net/>>
- [65] ROBOTC [online], (2009-04-27), <<http://www.robotc.net/>>
- [66] LEGO Mindstorms NXT in Teaching [online], MATLAB CENTRAL (2009-04-06),
<<http://blogs.mathworks.com/loren/2008/06/30/lego-mindstorms-nxt-in-teaching/> >

Univerzitní pracoviště

- [67] Robots Rule [online], Union College (2009-03-09), <<http://cs.union.edu/csc104>>
- [68] Roboclub [online], University of Arizona (2009-03-09) <<http://roboclub.arizona.edu/> >
- [69] Introduction to EECS1, [online], Massachusetts Institute of Technology (2009-03-09)
<<http://mit.edu/6.01/mercurial/spring09/www/index.html>>
- [70] MATLAB meets MINDSTORMS RWTH, [online] RWTH Aachen (2009-03-09)
<<http://www.lfb.rwth-aachen.de/en/education/ws07/mindstorms.html> >
- [71] Autonomous mobile robotics [online], ETH Zurich (2009-03-09)
<https://www.asl.ethz.ch/education/master/mobile_robotics >
- [72] Roborace [online] Institut pro teorii systémů a řídicí techniky, Stuttgart (2009-03-01)
<<http://www.ist.uni-stuttgart.de/robofab/> >
- [73] Roboti [online] České Vysoké Učení Technické v Praze (xx-xx-xx)

- [74] Kyber Robot [online], Technická Univerzita v Liberci (2009-03-09),
<http://www.tyden.cz/rubriky/veda-a-technika/technologie/stredoskolaci-v-liberci-predvedli-roboty-vlastni-vyroby_94227.html>
- [75] Robofest [online], Lawrence Technological University (2009-03-09),
<<http://www.robofest.net>>
- [76] First Lego League [online] (2009-02-04)
<<http://www.usfirst.org/firstlegoleague/community/HomePage.html>>
- [77] Robiáda [online], ČVUT, FS (2009-02-25) <<http://mech.fsik.cvut.cz/robiada/>>
- [78] <http://homepage.mac.com/mrlaurie/robo/robochallenge.html>
- [79] CSAS 4082: Intro to Robotics, Department of Mathematics and Computer Science Seton Hall University [online], Bert G. Wachsmuth (2009-04-04)
<<http://pirate.shu.edu/~wachsmut/Teaching/CSAS-Robotics/Lectures/index.html> >
- [80] G52ARB - Advanced Robotics [online], The Univeristy of Nottingham (2009-04-04)
<http://www.cs.nott.ac.uk/~jmg/G52ARB/>
- [81] The UAE Robot Challenge [online], Abu Dhabi Women's College. (2009-04-06)
<http://www.uaerobotchallenge.com/index.htm>
- [82] FIRST LEGO League na Slovensku [online], (2009-04-04),
<http://virtuallab.kar.elf.stuba.sk/robowiki/index.php?title=FIRST_LEGO_League_na_Slovensku >
- [83] Psych269 Cognitive Robotics [online], University of California – Irvine, School of social science (2009-04-06), <<http://lejos.sourceforge.net/tutorial/essential/toc.html#hardware>>
- [84] 2007 NXT Teacher Training Materials [online], The University of Texas at Austin, (2009-04-06) <<http://www.engr.utexas.edu/dteach/teacherpdi/2007materialsNXT.cfm>>
- [85] Real Time Systems, [online], Fakulta Informatiky Masarykovy Univerzity v Brně.
<<http://www.fi.muni.cz/~xpelane/IA158/> >

SEZNAM OBRÁZKŮ

Obr. 2.1: Rozměry robota	15
Obr. 2.2 programovatelná NXT kostka vyvinutá na MIT.....	16
Obr. 2.3: Soutěž Roborace na Universitě ve Stuttgartu.....	18
Obr. 2.4: Lego Robota při dni otevřených dveří	19
Obr. 2.5: Plakát pro ROBO SOUTĚŽ 2009.....	21
Obr. 2.6 NXT inteligentní kostka.....	24
Obr. 2.7: Blokový diagram hardwaru kostky NXT.....	25
Obr. 2.8: procesor Atmel ARM7.....	25
Obr. 2.9: Dobíjecí akumulátor	26
Obr. 2.10: Ukázka zapojení jednotlivých výstupních pinů.....	26
Obr. 2.11: Dopředný chod servo motoru	27
Obr. 2.12: Řízení napětí dodávaného do motorů pomocí PWM.....	27
Obr. 2.13: Ukázka zapojení jednotlivých vstupních pinů.....	28
Obr. 2.14: Propojovací kabely RJ-12 pro NXT	30
Obr. 2.15: Ukázka připojení senzorů ke kostce NXT.....	30
Obr. 2.16: Způsob práce s aktivními senzory	31
Obr. 2.17: Světelný senzor a jeho detaily	31
Obr. 2.18: Vnímání barev člověkem a světelným senzorem.....	32
Obr. 2.19: Princip rotačního senzoru.....	32
Obr. 2.20: Zvukový senzor a jeho detail.....	33
Obr. 2.21: Dotykový senzor a jeho detail	33
Obr. 2.22: Ultrazvukový senzor a jeho detail	34
Obr. 2.23: Akcelerometr	35
Obr. 2.24: Color senzor	35
Obr. 2.25: Compass senzor	36
Obr. 2.26: Gyroskop	36
Obr. 2.27: Bitmapová mapa displeje ukazující rozvržení pixelů na displeji.....	38
Obr. 2.28: komunikace mezi kostkami NXT v rámci Bluetooth.....	39
Obr. 2.29: rozhraní mezi procesorem ARM7 a BlueCore čipem	40
Obr. 2.30: 9V servo motor + detail	41
Obr. 3.1: Grafické vývojové prostředí jazyka NXT-G.....	44
Obr. 3.2: Programovací prostředí BricxCC pro programovací jazyk NXC.....	45
Obr. 3.3: Programovací prostředí Eclipse.....	48
Obr. 4.1: schéma soutěžního modelu.....	51
Obr. 4.2: možný model bludiště.....	51
Obr. 4.3: Konstrukční řešení robota: pohled zepředu a z boku.....	52
Obr. 4.4: snímání vzdálenosti ultrazvukovým senzorem.....	53
Obr. 4.5: Konstrukční řešení robotů ů z Robo soutěže.....	54
Obr. 4.6: Vývojový diagram pro úlohu "bludiště"	55
Obr. 4.7: nákres soutěžního plánu pro úlohu "překážky"	60
Obr. 4.8: konstrukce robota pro úlohu "překážky". Pohled z boku a zepředu	62
Obr. 4.9: Vývojový diagram programu pro splnění úlohy "překážky".....	63
Obr. 4.10: Diagram funkce "najdi čáru"	63
Obr. 4.11: Funkce "sleduj čáru a hledej objekt".....	64
Obr. 4.12: Diagram funkce pro kolmý kontakt se stěnou.....	66
Obr. 4.13: Schema soutěžního plánu pro úlohu "čára"	69
Obr. 4.14: konstrukční řešení robota pro úlohu "čára"	70
Obr. 4.15: Vývojový diagram pro úlohu "čára"	71
Obr. 4.16: Vlastní funkce.....	72

Obr. 4.17: základní smyčka programu "čára"	72
Obr. 4.18: grafický blok pro ovládání rotačního senzoru	73
Obr. 4.19: rozhodovací blok pro světelný senzor.....	73
Obr. 4.20: smyčka ošetřující chování robota v bílém prostoru	73
Obr. 5.1: náhled webové stránky.....	76
Obr. 5.2: Rozvržení webové stránky	77
Obr. 5.3: Hlavní menu webové stránky	78
Obr. 5.4: Ukázka vložené úvodní stránky.....	78

SEZNAM TABULEK

Tab. 2.1: Funkce jednotlivých výstupních pinů	27
Tab. 2.2: Funkce jednotlivých vstupních pinů	29
Tab. 2.3: Odběr proudu jednotlivými senzory	29
Tab. 3.1: Přehled programovacích možností.....	42

OBSAH ELEKTRONICKÉ PŘÍLOHY

- Elektronická verze dokumentu bakalářské práce
- Zdrojové kódy k vyřešeným úlohám
- Videá se zajímavými ukázkami úloh s robotem