

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

ZIGBEE MODUL PRO PLATFORMU LEGO MINDSTORM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

DARIUS KOČÁR

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

ZIGBEE MODUL PRO PLATFORMU LEGO MINDSTORM

ZIGBEE MODUL FOR LEGO MINDSTROM PLATFORM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Darius Kočár

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jaroslav Ráb

BRNO 2008

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav informačních systémů

Akademický rok 2007/2008

Zadání bakalářské práce

Řešitel: **Kočár Darius**

Obor: Informační technologie

Téma: **ZigBee modul pro platformu Logo MindStorm**

Kategorie: Počítačové sítě

Pokyny:

1. Seznamte se s platformou Logo MindStorm, komunikačním rozhraním ZigBee a moduly MicroChip pro ZigBee.
2. Navrhněte model lokální komunikace mezi robotem LogoMindStorm a modulem MicroChip, dále přes ZigBee mezi moduly ZigBee.
3. Implementujte podporu komunikace přes ZigBee.
4. Proveďte testování komunikace a porovnejte jej s komunikačním rozhraním Bluetooth, které je standardně použito.

Literatura:

- Logo MindStorm dokumentace

Při obhajobě semestrální části projektu je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Ráb Jaroslav, Ing., UIFS FIT VUT**

Datum zadání: 1. listopadu 2007

Datum odevzdání: 14. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
612 66 Brno, Božetěchova 2

doc. Ing. Jaroslav Zendulka, CSc.
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Darius Kočár**
Id studenta: 78936
Bytem: Potoční 720/22a, 682 01 Vyškov
Narozen: 23. 10. 1985, Vyškov
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
bakalářská práce

Název VŠKP: ZigBee modul pro platformu Logo MindStorm
Vedoucí/školitel VŠKP: Ráb Jaroslav, Ing.
Ústav: Ústav informačních systémů
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

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

Článek 2 Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3 Závěrečná ustanovení

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

Brně dne:

.....
Nabyvatel

D. Kocín
.....
Autor

Abstrakt

Cílem této práce je rozšíření platformy Lego MindStorm (dále LEGO) o ZigBee modul. Toto rozšíření umožní dálkově aktivovat pohyb robota. Dále je srovnáno výsledné řešení s již zabudovaným Bluetooth rozhraním. Tomu předchází analýza komunikačních požadavků a návržení komunikačních protokolů. Dále zhodnocení možností již předem daných, neměnných vlastností jako komunikační možnosti NXT Kostky, rozhraní ZigBee modulu, případně jeho řídicího obvodu.

Klíčová slova

ZigBee, LEGO, Lego MindStorm, I²C

Abstract

The goal of this thesis is extending Lego Mindstorm platform with ZigBee modul. This extension will enable remote control of robots motors. Further is comparsion between designed solution and already embeded Bluetooth. It is preceded by analysis of communication requirements and protokol designing. Further is evaluation of NXT Brics predefined communication abilities and ZigBee modules capabilities.

Keywords

ZigBee, LEGO, Lego MindStorm, I²C

Citace

Kočár Darius: ZigBee modul pro platformu Lego MindStorm. Brno, 2008, bakalářská práce, FIT VUT v Brně

ZigBee modul pro platformu Lego MindStorm

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Jaroslava Rába

Další informace mi poskytli přátelé.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Darius Kočár
1. srpna 2008

Poděkování

Děkuji Ing. Jaroslavu Rábovi a Ing. Ondřej Ryšavý, Ph.D za vedení a pomoc při řešení problémů spjatých s tvorbou práce. Dále chci poděkovat kolegovi Davidu Zemanovi z Fakulty elektrotechniky a komunikačních technologií za pomoc při tvorbě schémat.

© Darius Kočár, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah	1
Úvod	2
1 Platforma Lego MindStorm	4
1.1 NXT Kostka.....	5
1.1.1 USB sběrnice.	5
1.1.2 I ² C	6
1.1.3 Bluetooth.....	7
1.2 Rozšiřující čidla	7
1.2.1 I ² C senzory.....	8
2 ZigBee vývojový kit	9
2.1 ZigBee	9
2.1.1 Mesh síť.....	9
2.2 USART Modul.....	11
2.3 MSPP Modul	12
2.4 A/D Module	12
2.5 Jednotlivé piny.....	12
3 Návrh řešení	13
3.1 NXT Kostka – ZigBee modul.....	13
3.1.1 Komunikace na I ² C sběrnici	14
3.2 ZigBee modul – ZigBee modul	15
4 Implementace	16
5 Testování komunikace a srovnání s Bluetooth.....	21
5.1 Teorie testování komunikace.....	21
5.1.1 Teoretická výhodnost Mesh sítí.....	21
6 Závěr	26
Literatura	27
Zkratky.....	29
Seznam příloh	30
Tabulka 3 – Výčet programovacích jazyků [5]:	1

Úvod

Počítače jsou již běžnou součástí našich životů. A pokud jim chceme rozumět, je třeba se s nimi setkávat již od útlého mládí. Toho jsou si již vědomy i firmy vyrábějící hračky pro děti. Vznikají tak navenek jednoduché počítače, které hlavně pobaví. Některé ale poskytují více než zábavu. Je to například platforma Lego MindStorm (dále Lego) od společnosti LEGO. Tuto platformu můžeme dále vylepšovat, tvořit pro ni další rozhraní, senzory a de facto vše, co umí digitálně či analogově komunikovat. Výrobce ale vybavil tuto platformu v bezdrátových technologiích pouze rozhraním Bluetooth. Pokročilejší rozhraní, jako je WiFi, či energeticky nenáročné ZigBee chybí. Tato volba výrobce pak v určitých aplikacích, jako například kooperace více robotů na rozsáhlejšímu prostoru velmi omezuje vlastní využití robota.

Z hlediska naplnění zadaného obsahu a cílů, přehlednosti a logické posloupnosti jsem tuto práci zpracoval v šesti relativně ucelených kapitolách, jejichž stručná charakteristika následuje.

První kapitola je věnována základnímu seznámení s platformou Lego MindStorm. Je proveden její zevrubný popis fyzického složení a konfigurace, jsou uvedeny příklady využití této platformy. Zvláštní pozornost je věnována analýze poskytovaných komunikačních možností a analýze možností pro další vývoj nového hardware. Tato analýza vychází zejména z Hardware Development Kitu. Návrh vlastního technického řešení vychází z poskytnutých možností, především pak ze zabudovaných komunikačních schopností tzv. NXT Kostky a ZigBee modulu, kompatibility převodníků mezi různými komunikačními sběrnici/protokoly.

Druhá kapitola se zabývá charakteristikou a specifickými vlastnostmi ZigBee modulu, rovněž z pohledu ohodnocení nabízených možností a možnostmi samotného komunikačního modulu. Dále je v této kapitole rozebrána konkrétní vývojová souprava, na které se bude celé zařízení vyrábět.

Ve třetí kapitole jsou teoreticky rozebrány možnosti a technické postupy, kterými lze dosáhnout požadovaného cíle a funkčnosti. Tato kapitola je relativně rozsáhlá, protože se při podrobnější analýze ukázalo, že je zde příliš mnoho omezujících požadavků z třetích stran. Například zmíním pouze master rozhraní na komunikační sběrnici PC platformy Lego MindStorm nebo obsazené komunikační sběrnice na mikroprocesoru od ZigBee. Tyto a další skutečnosti podstatným způsobem omezily výsledný návrh.

Další kapitola popisuje a shrnuje praktickou realizaci výsledného návrhu. Jsou identifikovány a analyzovány problémy, které se při vlastní praktické realizaci vyskytly. Je shrnuta výsledná teoretická funkčnost celku. Je zde popsáno provedení testů odolnosti soustavy proti možným chybám, které se mohou vyskytovat jak na rozhraní mezi Legem a ZigBee modulem tak mezi bezdrátovými moduly navzájem.

V předposlední kapitole je provedeno teoretické srovnání bezdrátové technologie ZigBee s technologií Bluetooth, které již Lego standardně obsahuje. Dále jsou v kapitole uvedeny výsledky teoretických testů dosahu ovládání a odolnosti proti vybraným druhům rušení.

Poslední závěrečná kapitola shrnuje hlavní aspekty a výsledky celé práce. Je zhodnoceno naplnění obsahu a cílů práce. Jsou uvedena rovněž slabá místa výsledného návrhu. Jsou navrženy možnosti dalšího budoucího vývoje a funkcionalit, kterou mohou být přidány či vylepšeny.

Podstata řešení v souladu se zadáním a cílem této práce se v zásadě koncentruje na vyřešení dvou základních oblastí a to:

- Řešení lokální komunikace mezi robotem Lego MindStorm a modulem Microchip
- Řešení komunikace mezi jednotlivými ZigBee moduly

Tyto dvě základní oblasti, resp. části jsem vypracovával sice odděleně a zároveň společně s ohledem na jejich vzájemnou provázanost.

1 Platforma Lego MindStorm

Platforma Lego

MindStorm je výukovou platformou od společnosti LEGO. Je založená na jednoduchém skládání komponent jako kostek Lega. Její hlavní částí je tzv. NXT Kostka. Tato základní komponenta obsahuje veškerou inteligenci a napájení



všech částí s připojenými čidly. **Obrázek 1: NXT Kostka s připojenými čidly [4]**

K té se potom připojují

jednotlivé další komponenty pomocí čtyř analogových/digitálních portů (viz obrázek 1). NXT Kostka dále obsahuje 3 hybridní porty na řízení aktuátorů (aktuátor je mechanické zařízení pro pohyb či ovládání mechanismu nebo systému).

NXT kostka je mozkiem MINDSTORMS® robota. Je to inteligentní, počítačem řízená LEGO® kostka, která umožňuje robotu MINDSTORMS „oživnout“ a provádět různé operace.

Stručný popis základních charakteristik NXT kostky:

Motorové porty

NXT má tři výstupní porty sloužící pro připojení motorů - Porty A, B a C

Senzorové porty

NXT má čtyři vstupní porty pro připojení senzorů - Porty 1, 2, 3 a 4.

USB port

Připojením USB kabelu do USB portu a stáhnutím programů z počítače lze rozšiřovat schopnosti NXT kostky (či naopak nahráním dat z robota do počítače). Je možné také použít bezdrátové technologie Bluetooth jak pro nahrávání do NXT kostky tak stahování dat z NXT kostky

Reproduktor

Je možná tvorba programů s reálnými zvuky a jejich reprodukce za běhu programu.

NXT Tlačítka

Oranžové tlačítko: Zapnout / Potvrdit / Spustit

Světle šedé šipky: Požívány pro pohyby doleva a doprava v NXT menu.

Tmavě šedé tlačítko: Vymazat / Jít zpět.

NXT obrazovka

NXT má mnoho zobrazovacích vlastností – více o těchto vlastnostech lze nalézt v MINDSTORMS

NXT Users Guide, která je součástí balení NXT kitu. Jsou zde rovněž uvedeny detailní informace o jednotlivých ikonách a položkách nabídky menu.

Technická specifikace [4]

- 32-bit ARM7 mikrokontrolér
- 256 KBytes FLASH, 64 KBytes RAM
- 8-bit AVR mikrokontrolér
- 4 KBytes FLASH, 512 Byte RAM
- Bluetooth bezdrátová komunikace (Bluetooth Třída II V2.0 kompatibilní)
- USB port plné rychlosti (12 MBit/s)
- 4 vstupní porty, 6-vodičový digitální kabel (Jeden port obsahuje a IEC 61158 Type 4/EN 50 170 kompatibilní rozšiřující port pro budoucí použití)
- 3 výstupní porty, 6-vodičový digitální kabel.
- 100 x 64 pixel LCD plně grafický displej.
- Reproduktor - 8 kHz zvukovou kvalitou. Zvukový kanál s 8-bit rozlišením a 2-16 kHz vzorkováním.
- Zdroj napětí: 6 AA baterií

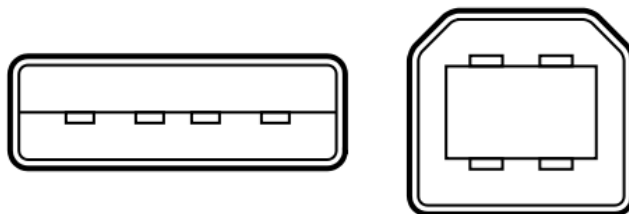
1.1 NXT Kostka

Základním výpočetním prvkem NXT kostky je 32-bitový ARM-7 mikrokontrolér. Dále je zde instalován 8-bitový AVR mikrokontrolér, který již není uživateli, ani vývojáři dostupný. Tento AVR mikrokontrolér funguje jako rozšiřující rozhraní pro hlavní ARM-7. [15]

Nyní podrobněji rozeberu všechny komunikační možnosti, které nám NXT Kostka nabízí.

1.1.1 USB sběrnice.

USB – Universal Serial Bus. Jak již název napovídá, jedná se o univerzální sériovou, centrálně řízenou sběrnici. V dnešní době se vyskytuje ve dvou standardech a to USB 1.1 a USB 2.0.



Obrázek 2: USB (typ A a B) konektory [1]

Tyto standardy se liší pouze přenosovou rychlostí. Pro USB 1.1 je maximální přenosová rychlost 12 MBit/s. Pro USB 2.0 je to 480 MBit/s. V obou případech je signál přenášen po metalických spojích. Je již připraven také standard USB 3.0, který dosahuje dalšího několikanásobného zvýšení přenosové rychlosti přidáním optického kabelu. NXT Kostka má ale podporu pouze USB 2.0. Maximální délka kabelu je tedy stanovena na 5m. [1]

Jak jsem již na začátku zmínil, jedná se o centrálně řízenou sběrnici, která obsahuje pouze jeden „Master“ prvek. Tento prvek řídí celou sběrnici. Na základě jeho dotazů odpovídají jednotlivá koncová zařízení.

Pracovní napětí USB je 5V. Připojená zařízení mohou z hostitele odebírat proud až 500mA.[1] Tato hodnota však není ultimativní. U většiny dnešních počítačů je napájení USB vyřešeno připojením přímo na zdroj a krátkodobě je takto možno odebírat i větší proud (např. roztočení ploten disku připojeného přes USB). Dlouhodobě to však není vhodné řešení, protože zbylé části sběrnice již nejsou na takový proud dimenzovány (např. kabely) a může docházet k nadměrnému zahřívání.

V budoucnu bude USB sběrnice stále používanější a již se chystají její další vylepšení jako např. WUSB (Wireless USB), či zmiňované USB 3.0 (s optickým přenosem dat). Vždy ale půjde o rozšiřování základních vlastností USB, a to velmi rychlé a jednoduché připojení zařízení na krátkou vzdálenost. V tomto ohledu má USB perspektivu, že bude i nadále v budoucnosti podporováno.

Naše NXT Kostka, kterou máme pro tuto práci k dispozici má rozhraní USB Slave – jako koncové zařízení. Tento fakt v kombinaci s tím, že ZigBee modul neobsahuje USB vůbec znamená, že pokud bychom se museli vydat touto cestou, tak by nezbylo, než implementovat USB na mikroprocesoru. Nebo ještě lépe, zakoupit rozšiřující USB Host modul. Musíme ale vzít v úvahu příliš vysoké komunikační napětí 5V [1], kterého není schopen stabilizátor mikroprocesor u ZigBee modulu dosáhnout [15]. Muselo by dojít k dalším úpravám formou přidání minimálně jednoho dalšího stabilizátoru.

1.1.2 I²C

I²C, neboli IIC je zkratka pro Inter-Integrated Circuit. Jedná se o multi-master sběrnici vyvinutou firmou Philips. Celá sběrnice sestává z jednoho Master prvku, jednoho nebo více Multi-master zařízení a dalších Slave zařízení. Všechny tyto prvky jsou propojeny pouze 2 obousměrnými vodiči.

Na celé sběrnici může být maximálně 128 zařízení.[2]

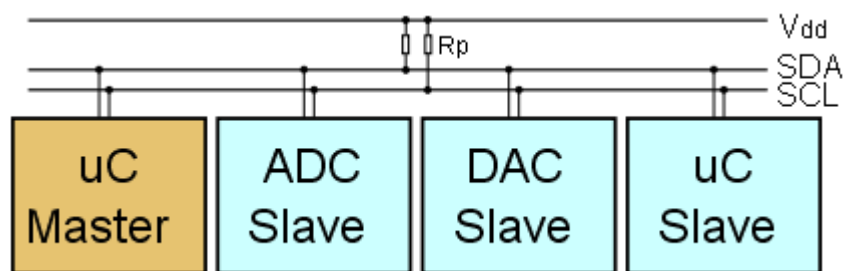
Dva vodiče po kterých probíhá komunikace jsou SDA a SCL.

- SDA – Synchronous Data.
- SCL – Synchronous Clock

Maximální délka

těchto vodičů je dána jejich nejvyšší přípustnou kapacitou 400pF.

Sběrnice je řízena Master či Multi Master prvky. Tato zařízení také generují hodinový signál, podle kterého závislá zařízení odpovídají. To klade velký důraz na jejich kvalitu, zvláště při dodržování vysokých kmitočtů (Pozn.: Licenční politika firmy Philips dovoluje implementaci I²C



Obrázek 3: Jednoduché schéma s jedním řídicím (mikrokontrolér) a třemi řízenými zařízeními (ADC, DAC a jiný mikrokontrolér) s pull-up rezistory Rp [2]

zdarma. Poplatky jsou nutné až za použité adresy. Ty jsou 7 Bitové, z nich je pak několik vyhrazeno pro speciální účely).

Komunikace probíhá tak, že stanice vyšle na sběrnici adresu zařízení, se kterým chce komunikovat. Pokud se takové zařízení na sběrnici nachází, tak mu odpoví ACK. [1] Pro řízení komunikace je použita metoda detekce kolize. Během vysílání musí neustále porovnávat vysílané Bity se skutečným stavem SDA. Je-li zjištěn rozdíl mezi očekávaným a skutečným stavem linky SDA, je to indikace kolize mezi několika stanicemi. Vzhledem k charakteru sběrnice (otevřené kolektory) může k této situaci dojít, pokud určitá stanice vysílá úroveň H, zatímco jiná stanice vysílá úroveň L. Stanice, která na lince zjistí úroveň L zatímco sama vysílá úroveň H musí vysílání okamžitě ukončit. K řízení komunikace většinou dochází během vysílání několika prvních Bitů, kdy je vysílána adresa přijímací stanice. Pokud by se např. dvě stanice současně pokusily o zápis do stejného obvodu, nastane kolize až při přenosu vlastních zapisovaných dat. V krajním případě, kdy několik stanic současně zapisuje stejná data na stejnou adresu, nemusí být kolize vůbec detekována.[2]

NXT Kostka je i v tomto případě velmi omezená. Sběrnice I²C je sice obsažena na všech 4 portech, ale pouze 1 port podporuje plnou rychlost 3,4MBit/s.[15] a všechny porty umí pracovat pouze v Master režimu. Z toho nám vychází, že u ZigBee modulu musí být Slave. Ostatní parametry jako spotřeba a pracovní napětí jsou ideální, nebo alespoň vhodné.

1.1.3 Bluetooth

Bluetooth již ze své bezdrátové podstaty není vhodný pro komunikaci s dalším bezdrátovým modulem. Nesmíme rovněž zapomenout, že pracují na stejných „volných“ frekvencích 2,4GHz. Implementace Bluetooth by u strany ZigBee modulu musela být řešena rozšiřujícím modulem. Bluetooth je také relativně velmi energeticky náročné (viz Kapitola 5 – Testování komunikace a srovnání s Bluetooth). Toto rozhraní již dále nebudeme uvažovat.



Obrázek 4: Logo Bluetooth [14]

1.2 Rozšiřující čidla

Pro komunikaci s okolním světem je zde přítomno rozhraní Bluetooth, USB a dále již zmíněná čidla. Základní čidla, která systém LEGO nabízí jsou následující:

1. Ultrazvuk na měření vzdáleností,
2. Světelný senzor s LED diodou, který může sloužit jako indikátor světla či jako upozornění na bezprostřední přiblížení k překážce,
3. Dotykový senzor,
4. Zvukový senzor – mikrofon.

Potom je zde již mnohem více či méně amatérských zapojení, která s NXT Kostkou komunikují buď pomocí USB, nebo mnohem častěji pomocí I²C sběrnice obsažené ve 4 portech. Pro komunikaci se ZigBee modulem nám ale může posloužit i jedno z čidel.

1.2.1 I²C senzory

Je zde možnost, aby například jeden ze senzorů místo dat, která má poskytovat, tak po propojení se



Obrázek 5: Digitální čidla. Zleva: čidlo zvuku, ultrazvukové čidlo vzdálenosti, optické čidlo vzdálenosti. [4]

ZigBee modulem poskytoval informace ze ZigBee sítě. Je to sice zvláštní možnost, ale komunikací přes některý z I²C senzorů bychom dosáhli velké abstrakce a de facto nám odpadla jakákoliv starost o údržbu našeho rozšíření. Výrobce stále zachovává podporu pro všechna svá zbylá čidla. Dále by podpora ZigBee byla podporována na úrovni celého Firmware NXT Kostky. Nevýhody jsou ale nasnadě, zatemňování původního návrhu čidel, nespolehlivost, atd.

2 ZigBee vývojový kit

ZigBee je jméno pro soupravu vysoko úroňových protokolů používající malá, nízko-příkonová rádia založená na [IEEE 802.15.4 standardu](#) [8]. Používá se pro WPAN (Wireless Personal Area Network) jako například bezdrátová sluchátka. Tato technologie je cílena jako konkurence Bluetooth. Má být ale levnější a pracovat s nižším odběrem.

2.1 ZigBee

ZigBee modul lze stručně charakterizovat jako levný, energeticky nenáročný bezdrátový standard pro mesh sítě. Nízká cena dovoluje této technologii, aby byla široce nasazována v aplikacích sloužících k



Obrázek 6: ZigBee modul

ovládání a kontrolování různých zařízení či služeb. Nízká energetická náročnost umožňuje delší pracovní čas při práci s bateriemi s menší kapacitou. Architektura mesh sítí zase přináší velkou spolehlivost přenosů na dlouhé vzdálenosti.[7][11]

2.1.1 Mesh síť

Mesh síťování je způsob, jak směřovat data, hlas mezi uzly. Základní výhodou Mesh sítí je redundance spojení. Dovoluje to nepřetržitá spojení a automatické přenastavení směrování v případě poruchy některých uzlů či výpadku linek. Toho je dosaženo předáváním paketů mezi jednotlivými uzly, dokud není dosaženo cílové destinace. Mesh síť, jejíž uzly jsou spojeny každý s každým se nazývá plně propojená síť. Mesh síť se liší od normálních sítí v tom, že zařízení sítě se mohou spojit každý s každým skrze vícenásobné „skoky“ a obecně nejsou mobilní. Mesh síť můžeme označit za druh Ad-Hoc sítě. Z tohoto důvodu jsou Mobilní Ad-Hoc síť (MANET – Mobile Ad-Hoc Network) a Mesh síť úzce spjaty, ale MANET síť navíc řeší problémy související s mobilitou uzlů.[12]

Mesh síť jsou tzv. sebezotavující, tj. síť může fungovat i když se pokazí uzel, nebo dojde ke ztrátě spojení. Výsledkem je velice spolehlivá síť. Tento koncept je aplikovatelný jak na bezdrátové síť, tak i na drátové.

Bezdrátové mesh síť jsou nejčastější aplikací mesh architektury. Bezdrátové mesh síť byly původně vyvinuty pro vojenské účely, ale v posledním desetiletí prošly znatelnou evolucí. Jak klesají ceny rádiových vysílačů, tak se jednotlivé bezdrátové produkty vyvinuly a nyní podporují více stanic na mesh uzel s dalšími uzly poskytující specifické funkce – jako například klientský přístup,

směrování zpětnou oklikou, nebo prohledávání pásma pro velmi rychlé předávání v mobilních aplikacích. Design uzlu mesh sítě se také stal více modulárním – jedno zařízení může podporovat více typů bezdrátových karet – každá pracující na své vlastní frekvenci.[12]

ZigBee Aliance, standard definující ZigBee také zveřejnila aplikační profily umožňující mnoha OEM výrobcům vytvářet vzájemně kompatibilní produkty.[7]

Pro nekomerční požití je ZigBee specifikace volně k dispozici. Pokud se ovšem jedná o komerční využití ZigBee protokolu, tak je třeba stát se členem ZigBee aliance na minimálně „vstupní úrovni“. (Toto členství stojí 3500\$ ročně a přináší přístup k zatím nezveřejněným specifikacím a právo vytvářet produkty pro trh dle těchto specifikací.[7][11]).

ZigBee jako standard pracuje jak na průmyslových frekvencích, tak i výzkumných a zdravotnických. Jde tedy o 868 MHz v Evropě, 915 MHz v zemích jako USA, Austrálie a 2,4 GHz ve skoro všech státech světa. Tato technologie si klade za cíl být jednodušší a levnější jak ostatní WPAN jako Bluetooth. Dodavatelé ZigBee čipů typicky poskytují integrované řešení jak s rádiem, tak i mikrokontrolérem s 60KB – 128KB flash paměti. Bezdrátové karty jsou ale také dostupné jako oddělená zařízení, která je možno použít s obecným mikrokontrolérem. Toto je případ, který se týká i Demonstračního kitu firmy Microchip, který mi byl poskytnut pro vývoj. [8][11]

V roce 2006 se cena ZigBee rádia blížila již k 1\$ a cena za celou soupravu rádia, procesoru a paměti se pohybovala kolem 3\$. Pro srovnání cena Bluetooth čipu je nyní pod 3\$.[13]

První zásobník, který byl pro ZigBee vydán se nyní jmenuje ZigBee 2004. Druhé vydání se jmenuje ZigBee 2006. Rozdíl je hlavně ve výměně [MSP/KVP](#) struktury použité v ZigBee 2004 za „cluster knihovnu“. ZigBee 2004 zásobník je dnes již víceméně zastaralý. Bohužel PICDEM Z 2,4GHz demo kit, který jsem měl k dispozici používá tento „zastaralý“ standard ZigBee 2004.

Nynější verze zásobníku ZigBee 2007 obsahuje 2 profily:

- První (nazývaný jednoduše ZigBee) pro domácí a jednodušší průmyslové použití. Tento profil má menší prostorovou náročnost, ale obsahuje pouze základní možnosti.
- Druhý profil (nazývaný ZigBee Pro). Tento profil nabízí více možností jako multicasting, směrování více-k-jednomu a vysoký stupeň zabezpečení pomocí symetrického šifrování SKKE.

ZigBee 2007 je plně zpětně kompatibilní se zařízeními používajícími zásobník ZigBee 2006. Zařízení se ZigBee 2007 se mohou připojit do ZigBee 2006 a lepších sítí. Kvůli mírným odlišnostem ZigBee Pro zařízení se musí chovat jako Koncová zařízení ZED.[6][7][11]

Ke své práci jsem dostal vývojový kit PICDEM Z 2,4GHz od firmy Microchip. Podle firemní informace se jedná o jednoduchou na používání a všemi oceňovanou vývojářskou platformu.[9] Kit obsahuje vysílač od Chipcomu a Mikrokontrolér PIC18F4620 pro jeho řízení. Pro vzájemnou

komunikaci je použita SPI sběrnice. Pracovní napětí obou součástek je 3,3V [9]. To je dodáváno přes napěťový regulátor z 9V baterie či síťového zdroje.

Samotný komunikační protokol mezi modulem Chipcomu CC2420 a Mikrochip PIC18F4620 je velmi složitý a samotná firma zrazuje od jeho vlastní implementace.[9] Místo toho je v balení dodáván program ZENA, s jehož pomocí dojde

k vygenerování potřebných zdrojových souborů a

souborů pro linker. Ty jdou dále přeložit s nemálo varováními opět na proprietárním Microchip vývojovém prostředí MPLab IDE (Integrated development Enviroment). Bohužel již samotný ZENA software mne při nastavení modulu na Chipcon CC2420 zrazoval od jakéhokoliv dalšího vývoje s tím, že daný produkt už je zastaralý. Vzhledem tomu, že jsem neměl na výběr jiné možnosti, takže jsem pokračoval dále. Pro plnou funkčnost je také nutné nainstalovat MPLAB C18 překladač pro správné přeložení zdrojových kódů. K nalezení je na stránkách výrobce studentská verze zdarma, či na přiloženém DVD.

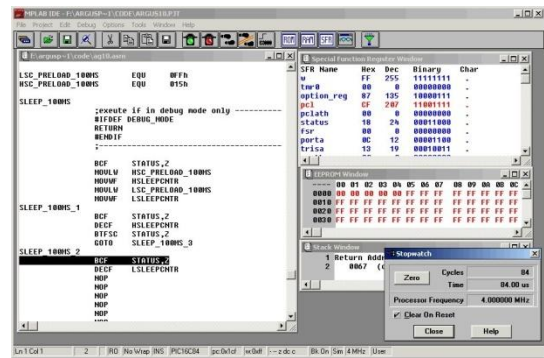
Nyní rozebereme dostupná rozhraní, pomocí kterých by bylo možné komunikovat s platformou Lego MindStorm.

2.2 USART Modul

Mikrokontrolér PIC18F4620 obsahuje USART modul, který podporuje sériovou komunikaci RS-232 (Standardní sériový port), RS-485 a LIN 1.2. Tento modul je ve výchozím nastavení použit jako debugovací konzole s výstupem na sériový port. Tato vlastnost ale ve finálním produktu nebude vůbec nutná. Problém nastává v tom, že NXT Kostka nemá sériový port. To si vyžaduje použití převodníku na USB, či I²C sběrnici.

RS-232 na USB Host adaptér de facto neexistuje. Je to z důvodů, že USB Host řídí celou sběrnici, která není zrovna triviální a pomalá.

Pro I²C sběrnici sice je mnoho adaptérů, ale všechny podporují chování jako I²C zařízení Master, popřípadě Multi-Master. I²C Master koncové zařízení je již ale samotná NXT Kostka, takže další zařízení na sběrnici již mohou být pouze Slave. Pokud by byla alespoň Multi-Master, tak na sběrnici by mohly být přítomny další Multi-Master zařízení. Problém v „RS-232 na I²C Slave“ je dle výrobců v tom, že I²C Slave zařízení musí velmi přesně zachovávat časování vysílané Masterem, což není dle údajů od výrobce jednoduché. Závěr je tedy takový, že tuto možnost lze uvažovat jako alternativní řešení, ale pouze pod vyššími výrobními náklady (100\$ a více)



Obrázek 7: Vývojové prostředí MPLab

2.3 MSPP Modul

Master Synchronous Serial Port Module. Tento modul umožňuje komunikaci po 3-vodičovém SPI a I²C sběrnici v režimech Master, nebo Slave. Z předchozího popisu základních charakteristik a jejich analýz pro potřeby této práce je konstatováno, že LEGO na svých portech vyžaduje I²C Slave zařízení. Tímto by byla komunikace mezi Kostkou a ZigBee vyřešena. Bohužel tento modul je již zabrán pro komunikaci pomocí SPI s vysílací částí. I²C piny a registry modulu jsou shodné s těmi pro SPI. Tímto je jakýkoliv paralelní běh obou sběrnic vyloučen a toto řešení je dále nepoužitelné.

2.4 A/D Module

Tento modul by bylo možné použít pouze pro velice jednoduchou komunikaci (je/není napětí). Jsou zde jistě ne zrovna vhodné vlastnosti jako zpoždění převodu analogového napětí na digitální hodnotu. Vzhledem k této a dalším nevhodným vlastnostem jsem od tohoto řešení ustoupil.

2.5 Jednotlivé piny

Další komunikaci je možno provádět pomocí 36 digitálních I/O portů. Porty pracují s napětím 3,3V. Několik z nich je již použito pro vlastní funkčnost demonstračního kitu. Zbylé jsou volně k použití. Přímou komunikaci přes tyto piny nelze. Není kam zapojit na druhé straně – NXT Kostce. Je však možné je použít pro rozšíření celého kitu o další modul např. I²C. K tomuto účelu je od firmy Philips dodávána řada zařízení PCF, která poskytují mnoho rozhraní pro sběrnici I²C. Pro zadané řešení je konkrétně vhodný model PCF8574P. Jedná se o 8 Bitový I/O expandér pro I²C sběrnici pracující v režimu slave. Řešení komunikace tímto způsobem je možné a v zásadě vhodné pro náš účel.

3 Návrh řešení

V této kapitole je teoreticky rozebrán postup, jakým se došlo k výslednému způsobu realizace. Je zde také zdůvodněno, proč je vybraný výsledný způsob řešení ze všech uvažovaných nejvhodnější. Dále jsou zde diskutovány možnosti dalšího vývoje. Způsob implementace zvoleného nejvhodnějšího návrhu je popsán v následující kapitole.

3.1 NXT Kostka – ZigBee modul

Při návrhu řešení je postupováno od strany NXT Kostky, protože její rozhraní jsou nejvíce omezující. Z možných řešení se jako nejlepší varianta nabízí použití sběrnice I²C. Na straně NXT Kostky je implementována jako Master. Z toho vychází nutnost, aby na straně ZigBee modulu byla implementována jako Slave. Toho můžeme dosáhnout dvěma způsoby.

- Pomocí již vestavěného USART Modulu.
- Nebo rozšiřujícím modulem, který implementuje I²C Slave.

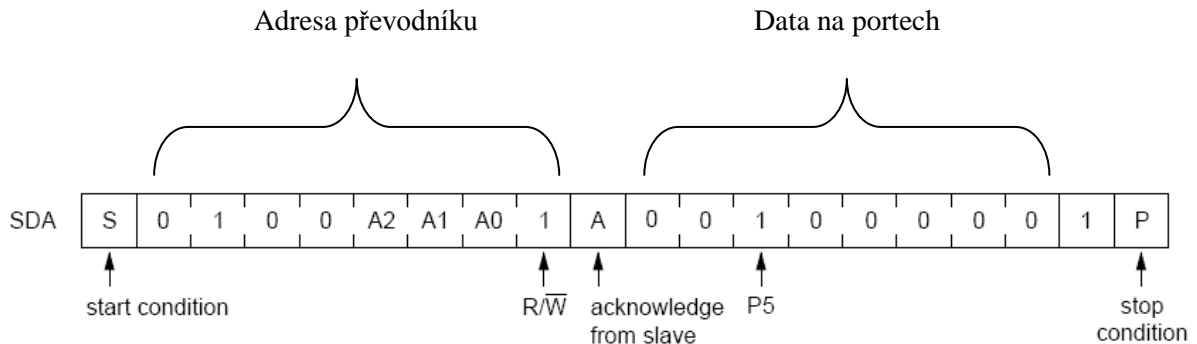
První možnost by byla velice komplikovaná. Muselo by se vyřešit sdílení pinů I²C s SPI sběrnici. To by i následně vyžadovalo ochranu proti ztraceným paketům, které by nebyly doručeny z důvodu, že je zrovna ona sběrnice vypnuta. Toto řešení jsem ponechal jako záložní, pokud by se u jiných objevil nečekaný, těžko řešitelný problém.

Druhý způsob je velice zajímavý již jenom tím, že by se veškerá režie týkající se komunikace po I²C přenesla mimo mikrokontrolér. Tím by se výrazně snížilo výpočetní zatížení mikrokontroléru. Následuje výčet charakteristických vlastností modulu PCF8574:

- Provozní napětí od 2,5 do 6 Volt
- Nízký proudový odběr v klidu – 10 μ A maximálně
- I²C na paralelní port rozšíření
- Obvod přerušení
- DIP16, nebo prostor šetřící SO16, SSOP20 balení.

PCF8574 je křemíkový CMOS obvod. Poskytuje obecné rozhraní pro rozšíření kompatibilních mikrokontrolérů o I²C sběrnici. Zařízení se skládá z 8 Bitových kvazi obousměrných portů a rozhraní I²C-sběrnice. PCF8574 má nízké energetické nároky a obsahuje výstupy schopné dodávat vysoké proudy, tj. schopnost přímo napájet LED diody. Obsahuje také port přerušení, který může být připojen k logice přerušení na mikrokontroléru. Nastavením přerušení může I/O obvod informovat mikrokontrolér o přijatých datech aniž by musely komunikovat přes I²C. To znamená, že PCF8574 může zůstat jednoduchým závislým zařízením. PCF8574 a PCF8574A verze se liší pouze v jejich adresách. V zapojení je použit PCF8574P. Jedná se o specializaci PCF8574. PCF8574P je ve

schránce DIP16. DIP16 schránka je schránka z umělé hmoty, kde jsou jednotlivé porty vyvedeny ve formě nožiček ve dvou rovnoběžných řadách. [16]



Obrázek 8: Detail komunikace na I²C sběrnici [16]

Vstupy a výstupy jsou při normálním zacházení chráněny proti elektrostatickým výbojům. Pokud je však potřeba zajistit mikročip absolutně, je třeba přistoupit k bezpečnostním opatřením pro zacházení se zařízeními MOS. Vhodné náměty lze nalézt v „Data Handbook IC12“ v kapitole „Handling MOS devices“.[16]

Provozní napětí splňuje jak napájení z Lega – 5V, či ze zdroje mikrokontroléru – 3,3V. Klidový proudový odběr 10 μ A [16] je také zanedbatelný oproti 100 000 μ A mikrokontroléru [15]. Díky rozšíření na paralelní port je možné použít pouze tak širokou sběrnici, kolik dat chceme přenášet. V našem případě je požadována pouze vzdálená aktivace motorů robota přes jiné ZigBee RFD (koncové zařízení). Obvod přerušení nebudeme používat, protože komunikace bude pouze směrem ZigBee – Robot.

3.1.1 Komunikace na I²C sběrnici

Komunikace na I²C probíhá stylem dotaz - odpověď. Při použití převodníku PCF8574 má pak konkrétní sémantika komunikace následující význam: volání převodníku – odpověď obsahující napětí na portech. V případě zapisovací transakce volání převodníku: data k nastavení na porty – potvrzení od převodníku. Grafické zobrazení čtecí transakce je znázorněno na obrázku 8. [16]

Komunikace na I²C sběrnici začíná startovacími Bity následované adresou zařízení, se kterým chce master či multimaster zařízení komunikovat. Veškerá komunikace nyní probíhá mezi volajícím a volaným zařízením až do výskytu stop Bitů. Obecně není omezen počet Bytů přenesených mezi koncovými zařízeními. Každý Byte složený z 8 Bitů je následován jedním potvrzovacím Bitem. Potvrzovací Bit je úroveň H (Hight) na sběrnici generovaná vysílačem vždy, když master vygeneruje puls pro potvrzovací Bit. [2] [16]

Slave přijímač, který je adresován musí generovat potvrzení po přijetí každého Bytu. Totéž platí i pro Master a Multimaster zařízení přijímající Byte od slave vysílače. Zařízení, které potvrzuje musí stáhnout napětí na lince SDA (na obě linky jsou použity Pull-up rezistory, takže výchozí hodnota je H) v průběhu potvrzovacího pulzu. Ve výsledku je tedy SDA linka ve stabilním L (Low) stavu po celou periodu potvrzovacího pulzu. Zpoždění při nastavování a udržování stavu musí být brány v úvahu. [2] [16]

3.2 ZigBee modul – ZigBee modul

Jak již bylo zmíněno v kapitole o ZigBee Vývojovém kitu, tak vlastní komunikační protokol je relativně složitý a k jeho implementaci slouží software ZENA. Tento software nenabízí žádnou základnu pro vývoj dalších vrstev. Spíše se snaží o co nejméně náročnou implementaci základních (zapnuto/vypnuto) a mírně vyšších funkcí (stmívač) v duchu ZigBee, tj. nízké energetické náročnosti. Pro náš účel použijeme jeden vypínač on/off, jehož výstup bude zapojen na I²C expandér, který ho přes I²C sběrnici dále předá NXT Kostce. Pro další vývoj by ale bylo zapotřebí přenášet libovolná datová slova.

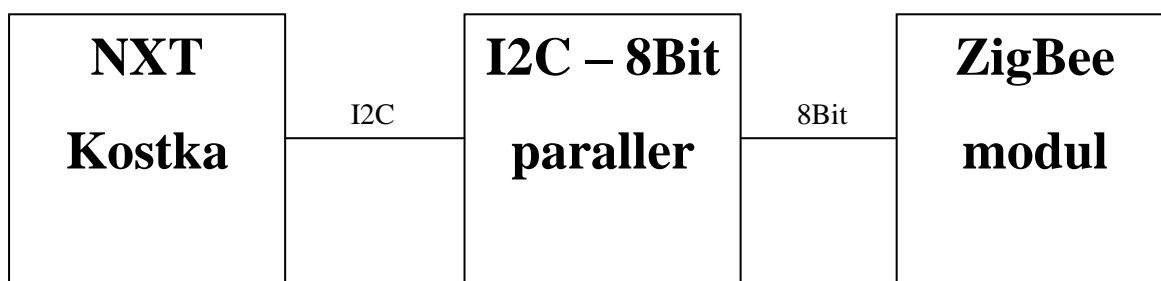
Vlastní komunikace na ZigBee síti může být volitelně šifrována. Toto již však není požadováno a proto toto téma nechávám pro další výzkum a vývoj.

Po zvážení všech pro a proti jednotlivých řešení jsem pro výslednou implementaci zvolil řešení s I²C a „vypínačem“. Tj. pro komunikaci mezi NXT Kostkou a ZigBee modulem bude použita I²C sběrnice, kde NXT kostka bude Master a ZigBee modul bude přes rozšiřující modul PCF8574 v pozici Slave. V komunikačním protokolu mezi ZigBee moduly bude použit mód „vypínač“. Jeho vypínání ale nebude ovládat světlo v kuchyni, jak autoři zamýšleli, ale pohyb robota.

4 Implementace

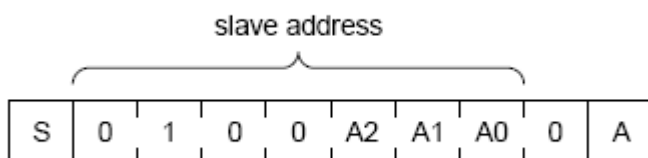
Implementace sestává z vlastního sestavení obvodu a jeho spájením. Dále bylo třeba sestavit ukázkový program demonstrující funkčnost navrženého řešení. Tato část vychází hlavně z „Hardware development kitu“. Podle specifikací zde uvedených bylo celé zařízení navrženo a sestaveno. Bylo třeba pečlivě zvážit veškeré parametry, jako například napájecí napětí a proudový odběr převodníku tak, aby navržené řešení bylo plně funkční. Zaměřím se tedy hlavně na konkrétní zvláštnosti a problémy se kterými jsem se setkal.

Struktura celé komunikace je uvedena na blokového schématu na obrázku 9. Dle specifikací I²C rozhraní obsažených v příloze [1] - LEGO MINDSTORMS NXT Hardware Developer Kit je



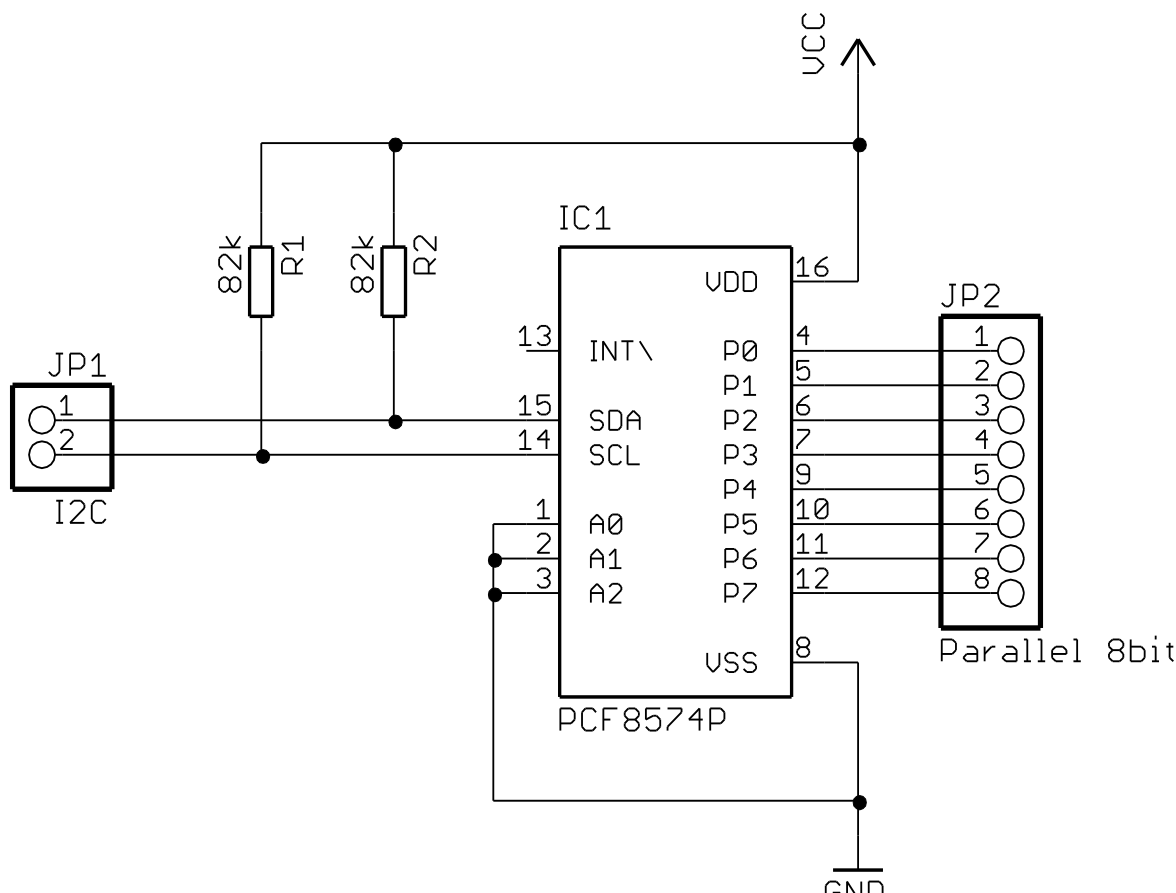
Obrázek 9: Blokové schéma zapojení

podstatné, že na oba porty SDA (datový kanál) i SCL (hodinový signál) musí být použity pull-up rezistory o hodnotě 82kΩ. Použil jsem standardní rezistory s tolerancí 1%. Jako další je třeba pořídit konektory pro tři vodiče, které budou použity jako výstup na I²C sběrnici + zem. Podadresa zařízení



Obrázek 10: Adresa slave převodníku PCF8574 [16]

se u obvodu PCF 8574 určuje pomocí tří pinů A0, A1, A2. Tyto piny jsem uzemnil, čímž vznikla adresa 111. Tuto adresu budu později používat pro adresaci ZigBee modulu. Z 8 vstupně/výstupních portů je při tomto zapojení použit pouze jeden, který má hodnotu zapnout či vypnout motory. Ostatní porty je možno použít pro další, rozšiřující funkce. Vzhledem k tomu, že zadání práce blíže nevyžaduje případné další funkce je pro demonstraci správnosti navrženého řešení postačující funkce ovládání motorů. Při implementaci dalších funkcí by se postupovalo analogickým způsobem. Ostatní porty jsou uzemněny. Mohou být použity později. Schéma výsledného zapojení je na obrázku 8. Po celkovém spájení zapojení na univerzálním plošném spoji, který je součástí ZigBee vývojového modulu jsem přistoupil k ožívování. Pokud je navržené řešení správné a nedošlo k chybám či



Obrázek 10: Schéma zapojení převodníku I2C - 8bit

poškozením při fyzickém provedení obvodu měla by výsledná sestava správně fungovat ihned po připojení ke zdroji napájení.

Oživení zapojení sestává z napsání jednoduchého programu, který se převodníku na adrese 0101110 dotáže na hladiny napětí na jednotlivých pinech. Programování NXT Kostky je možné v mnoha různých jazycích. Jejich výčet je uveden [v tabulce 1](#), která je součástí příloh této práce. Pro naši potřebu je vhodný jazyk pro standardní firmware s nízkou úrovní podporou I²C, který by v ideálním případě byl podobný C++, či Pythonu. Po zvážení všech požadavků jsem zvolil jazyk NXC. Je obsažen ve velice silném vývojovém prostředí BrixCC.

Bric Command Center, neboli BrixCC je Windows (95, 98, ME, NT, W2K, XP, Vista) program běžně znám pod názvem IDE (Integrated Development Kit) pro vývoj aplikací pro NXT(všechny verze) Scout, Cybermaster, a Spybot programovatelných kostek s použitím Davea Bauma Not Quite C (NQC) programovacího jazyku. Podporuje také Scout, RCX2, a Spybot s využitím MindScript(tm) od firmy LEGO a LASM(tm) programovacího jazyku prostřednictvím Mindstorms 2.5 SDK(Software Development Kit). Podporuje také programování RCX kostek v C, C++, Pascal, Forthu, a Javě s použitím brickOS, pbForth, and leJOS alternativních firmwarů. BrixCC nyní také podporuje programování nové LEGO Mindstorms NXT kostky pomocí Not

eXactly C (NXC), Next Byte Codes (NBC), a jednoduchého, na kostce přítomného jazyka zvaného NPG. [3]

Program jsem tedy psal přímo v IDE BrixCC a ladil na připojené NXT Kostce. Je zde použito pouze standardních schopností jazyka. Knihovny jsou požitý pouze standardní. Makra jako S1, či LCD_LINEx mají mnoho aliasů a mezi těmito aliasy je můžeme libovolně zaměňovat. Výsledný kód programu, který ovládá jízdu robota vpřed a daný stav současně vypisuje na displej NXT Kostky potom může vypadat následovně.

Program v jazyce NXC ovládající pohyb vpřed na základě dat přijatých přes I²C:

```
1  #include "NXCDefs.h"
2
3
4  byte bufLSWrite[] = {0x4F}; ///< Adresa zarizeni +1(cteme data)
5  +0(zapisujeme)
6  /** Funkce pro vymaskovani Bitu
7   * @param b Byte kde se Bit nachazi
8   * @param Pozice Bitu
9   * @return Hodnotu Bitu. */
10 inline bool bit (const byte b, const int pos ) {
11     byte tmp = b << pos;
12     tmp = tmp >> 7;
13     return tmp;
14 }
15
16 task main()
17 {
18     byte outbuf[]; ///< Buffer pro prijem dat
19     byte tmp;      ///< Pomocna promena
20     bool result;   ///< Vysledek
21
22     // Zapneme sezor
23     SetSensorLowspeed(S1);
24     // Konstanta prijmanych B
25     const int count = 1;
26     // Prijata data
27     int x;
28     while (1) {
29         // Prijem dat od ZigBee
30         x = I2CBytes(S1, bufLSWrite, count, outbuf);
31         // Kotrola chybovych stavu
32         while ((outbuf[0] == 255) || (outbuf[0] == 0))
33             x = I2CBytes(S1, bufLSWrite, count, outbuf);
34
35         bool run = bit(outbuf[0], 4);
36         if (run) {
37             OnFwd(OUT_BC, 30);
38         } else {
39             Off(OUT_BC);
40         }
41         //ClearScreen ();
42         TextOut(10, LCD_LINE1, "Fwd:");
43         NumOut(40, LCD_LINE1, run, 100);
44         Wait (100);
45     }
46
47     Wait (2000);
48 }
```

Veškerá dokumentace k výše uvedeným funkcím jazyka NXC může být nalezena na webových stránkách NXC projektu. Zdůrazním zde některé části kódu, které nemusí být ihned jasné, ale jsou nutné pro další tvorbu.

Jako první je důležité nastavení I²C. Ke správné funkčnosti, je třeba všude nastavit port, na který je zařízení připojeno. V ukázkovém příkladu to je S1 = Port 1. S1 je makro a má mnoho dalších aliasů.

Dále výpočet adresy zařízení se provádí tak, že vezmeme konstantu pro tento typ převodníku 01000000 (0x40) a přičteme k ní nastavené hodnoty na adresních Bitech A0 – A2 posunuté o jednu vlevo. Teda například 1110 pro samé jedničky. Logickým součtem dostaneme I²C adresu našeho zařízení. Poslední Bit (ten nejvíce vpravo) nastavíme na 1 pro čtení nebo 0 pro zápis.

Poslední zvláštností je, že hodnoty 0 a 255 se neberou v úvahu. Funkce *I²CBytes* a její aliasy *I²CRead*, *LowSpeedRead*... vrací „0“, pokud není žádné zařízení připojeno, „255“ vrací zase při chybě v komunikaci (Když například přijímáme více dat, než bychom měli, či dojde k přeslechu).

Tento kód vypisuje pouze přijatá data ze ZigBee modulu. Pro jejich zpracování je ještě potřeba vymaskovat jednotlivé Bity pomocí operátorů Bitového posunu `>>` a `<<`. Toto je realizováno pomocí funkce/subrutiny „bit“. Její parametry jsou uvedeny v komentářích, které lze pomocí programu oxygen převést do programové dokumentace v HTML anebo Latexu.

Po vlastním zapojení všech součástí a naprogramování robota vše fungovalo přesně dle předpokladů. Při opakovaném testování funkcí bylo občas třeba déle počkat, až se zařízení vzájemně najdou. Tedy až se koncový RFD modul připojí do sítě vytvářené Koordinátorem. Pokud se zařízení chovají neočekávaně až nedeterministicky tak je pravděpodobné, že napájecí baterie jsou již vybity. ZigBee kit totiž nemá žádnou kontrolu napětí, které mu je poskytováno.

5 Testování komunikace a srovnání s Bluetooth

V této kapitole je obsaženo testování komunikace ZigBee modulů a srovnání navrženého řešení s možnostmi Bluetooth, které je standardně použito v NXT Kostce.

5.1 Teorie testování komunikace

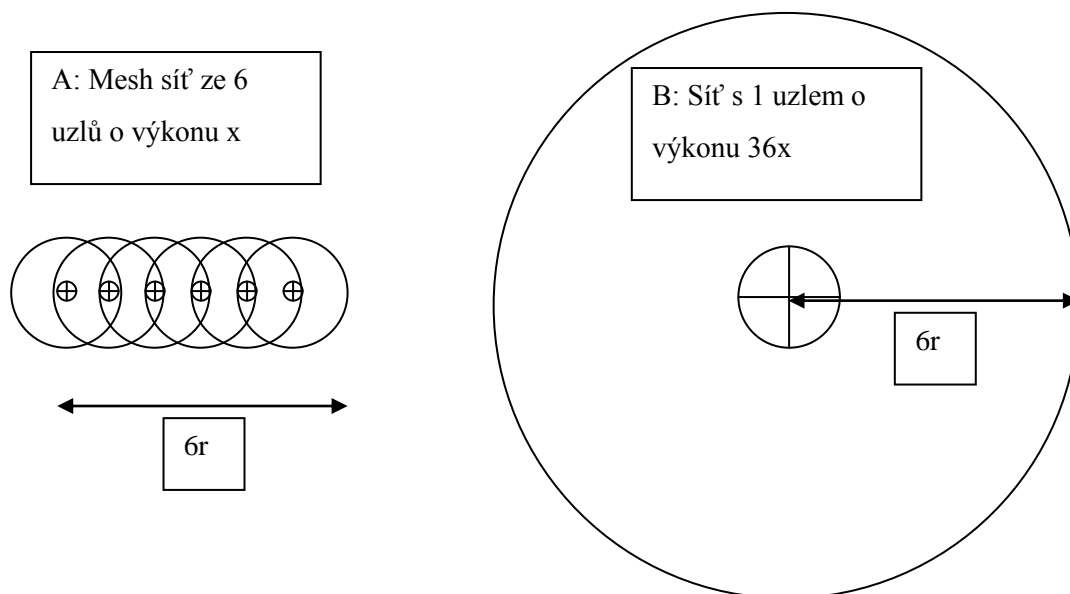
Při teoretickém srovnání těchto dvou technologií jsem čerpal z mnoha různých materiálů. Již na začátku je třeba si uvědomit, že tyto dvě technologie, ač se zdají být velmi podobné mají úplně jinou cílovou skupinu zákazníků a aplikací, pro které jsou určeny a kde jsou nasazovány.

Bluetooth jako starší z nich má sloužit pro tvorbu PAN sítí, které nepředpokládají soustavný provoz v řádu dní. Má sloužit hlavně jako jednoduchý prostředek pro synchronizaci a výměnu dat mezi zařízeními jako jsou notebook, mobilní telefon, GPS modul apod. na malé vzdálenosti.[14]

Naproti tomu ZigBee standard je míněn jako velmi spolehlivý protokol pro přenos malého množství dat v polních podmínkách, tj. v podmínkách, kde může nastávat velké rušení a kde je vyžadována nízká energetická náročnost a vysoká spolehlivost. Této spolehlivosti se dosahuje použitím technologií Mesh sítí. Velkou redundancí uzlů sítě a zároveň nízkým výkonem (a z toho vyplývající nízkou spotřebou) lze takto dosahovat přenosů na velké vzdálenosti při velké spolehlivosti a zároveň nízké energetické náročnosti.

5.1.1 Teoretická výhodnost Mesh sítí

Praktické využití jedné z vlastností mesh sítí lze vidět na následném obrázku a obecném příkladu, který ho následuje. Tato vlastnost – preposílání paketů přes více uzlů samozřejmě není jediná a pomocí „repeater“ módu ji lze nastavit i na obyčejných AP.



Obrázek 11: Schéma ke vzorovému příkladu srovnání Mesh sítě se sítí s infrastrukturou.

Příklad: Pro ideální případ, kdy je útlum prostředí rovnoměrný a po celé ploše se nikde nevyskytuje žádné rušení ani odrazy spočítáme celkový vyzařovaný výkon pro možnosti A i B.

Tabulka 1: Výpočet výše uvedeného příkladu.

A	B
Jeden uzel o výkonu x má dosah odvozen z rovnice $x = \pi r^2$ Z obrázku dále vyplývá, že celkový dosah při použití minimálního počtu uzlů je $6r$. Celkový vysílaný výkon (a tedy i celková spotřeba navrženého řešení) je $6x$	Nyní vyjdeme z potřebného dosahu $6r$. Po dosazení do rovnice $y = \pi(6r)^2$ dostaneme $y = 36\pi r^2$ Dalšími úpravami a substitucí $x = \pi r^2$ nám vyjde y – tedy celkový vysílaný výkon (a tedy i celková spotřeba navrženého řešení) rovna $36x$

Příklad názorně ukazuje praktické snížení potřebného výkonu k přenesení bezdrátového signálu na větší vzdálenost. Samozřejmě pokud bychom přenášeli signál pouze mezi dvěma body, tak nepoužijeme všesměrové antény, ale to platí pro obě možnosti. Obě řešení mají své výhody i nevýhody.

U řešení A lze dosáhnout velké redundance a při dobrém nastavení úsporných procesů a využívání pouze části sítě lze dosáhnout také energetické úspory. Nevýhoda tohoto řešení tkví ve složitější údržbě a správě takovéto sítě, kdy už nelze nastavovat každý z uzlů ručně. Dále je třeba při stavbě složitější sítě mít podrobný návrh požadavků a významných komunikačních toků, abychom mohli síť správně navrhnout. Také vlastní rozmístění uzlů do reálného světa je více než

problematické (zamezení proti krádežím, zajištění aspoň občasného napájení...). I přes všechny tyto nevýhody kladná vlastnosti Mesh sítí převažují.

Řešení B se používá při standardních sítích s infrastrukturou. Jedná se o statické řešení, která lze těžko přizpůsobovat měnícím se požadavkům na síť a výpadkům jak zařízení, tak i jednotlivých linek. Jeho výhodou je snadná udržovatelnost, jasná topologie sítě. Dále je třeba fyzicky udržovat pouze jasně vymezené uzly. Nevýhody jsou omezené přizpůsobování se změnám v síti, nároky na energii, velká kolizní doména (v ukázaném příkladu).

Pro lepší názornost a následnou diskusi je uvedena tabulka porovnání základních vlastností ZigBee a Bluetooth.

Tabulka 2: Srovnání ZigBee s Bluetooth (Informace čerpány z [6][8][9][11][14])

	ZigBee	Bluetooth
Frekvenční pásmo	2,4 GHz	2,4GHz
Dosah navrhovaný	10-100metrů	10 metrů ⁽¹⁾
Dosah v praxi	Až 400 metrů	Až 100 metrů(Třída 1)
Přenosová rychlost	20-250 KBit / s	1Mbit / s až 3Mbit / s ⁽²⁾
Energetická náročnost[typická životnost na baterii]	1 rok	Několik dní
Bezpečnost	128 Bit AES ⁽⁴⁾	64bit, 128 Bit s vlastními algoritmy ⁽³⁾⁽⁴⁾
Počet zařízení v síti	Až 65535	8
Škálovatelnost (sítě)	Velmi vysoká	Nízká
Spolehlivost	Vysoká	střední
Komplexnost (Náročnost na použití v praxi)	Nízká	Vysoká

- (1) Jedná se o Class 2 zařízení. Bluetooth vysílače se dělí do 3 tříd které se liší maximálním vyzařovaným výkonem. Class 3 - 1mW Class 2 – 2,5mW Class 3 – 100mW
- (2) Základní návrh specifikace Bluetooth v 1.0 specifikoval maximální rychlost 1MBit/s, ale další standardy 2.0 a 2.1 EDR(Enhanced Data Rate) přidávají podporu komunikace rychlostí až 3Mbit/s
- (3) Bluetooth používá vlastní šifrovací algoritmy odvozené od SAFER+ block cipher.
- (4) Větší bezpečnosti je možné dosáhnout použitím zabezpečení na vyšších síťových vrstvách (například až na aplikační vrstvě).

Jak již jsem dříve zmínil a tak i z tabulky vyplývá, že každý standard je poplatný potřebám pro které vznikl. Z tohoto důvodu si tedy musíme ujasnit vlastnosti, které od bezdrátového rozhraní pro daný účel očekáváme my. Jsou to tedy následující:

- **Spolehlivost** (Odolnost proti rušení a nejlépe i tzv. „selfhealing“, tedy automatické zotavování se z poruch sítě)
- **Nízká energetická náročnost** – minimálně týdný provozu v pohotovostním režimu.
- **Bezpečnost**, tedy šifrované spojení minimálně na takové úrovni, aby bylo v dnešní době neprolomitelné.

To jsou tedy hlavní požadavky, které od použité komunikační technologie požadujeme. Dále jsou již méně podstatné záležitosti týkající se náročnosti nasazení technologie v praxi, licenční

podmínky, dostupnost technické podpory atd. Tyto požadavky jsou sice nepodstatné, ale při vyvážení hlavních požadavků se podle nich budeme dále rozhodovat.

Při posuzování **spolehlivosti** je třeba porovnat použité frekvenční pásmo. ZigBee sítě mohou pracovat na frekvencích 868 MHz v Evropě, 915 MHz v zemích jako USA, Austrálie a 2,4 GHz ve většině států světa.[8][11] To je výhoda oproti Bluetooth, které je stavěno pouze pro frekvenční pásmo 2,4GHz. Na této centrální frekvenci [7] lze v dnešní době provozovat zařízení bez potřeby vlastnit licenci (bezlicenční pásmo). To je sice určitý klad, ale licenční problematika není předmětem této práce. Vyšší spolehlivost je dosahována a zajišťována především provozem v licencovaném pásmu, kde máme zaručeno, že nebude docházet k vzájemnému rušení s jinými zařízeními. Se ZigBee tedy máme možnost omezeného výběru frekvence. Dalším faktorem ovlivňujícím složitost je i komplexnost daného protokolu. Čím je protokol složitější, tím více je náchylný k výskytu chyb při jeho implementaci. Těmto chybám se nedá u složitějších protokolů vyvarovat.

Dalším bodem je **nízká energetická náročnost**. U tohoto požadavku je zřejmá výrazná přednost ZigBee před Bluetooth. ZigBee obsahuje mnoho rutin jak pro vlastní snížení odběru v aktivním režimu, tak i protokoly pro uspávání celého zařízení v pravidelných intervalech. Teoretické hodnoty výdrže jsou k nalezení v tabulce 2.

Poslední z důležitých vlastností je **bezpečnost**. Oba protokoly poskytují různé možnosti zabezpečení spojení již na síťové vrstvě. Bezpečnost je u Bluetooth zajištěna pomocí vlastních algoritmů[6][14]. U ZigBee je to pomocí Amerického standardu pro zabezpečení AES. Ale praxe ukazuje, že rozšířenější bezpečnostní standardy (a nejen ty) bývají také předmětem většího zájmu o prolomení ze strany zkušených uživatelů. Mimo jiné i z tohoto důvodu byla bezpečnost Bluetooth již několikrát prolomena. Při posuzování této vlastnosti má mírnou přednost ZigBee. Dalšího zvýšení bezpečnosti můžeme dosáhnout na vyšších síťových vrstvách například pomocí SSL / TLS.

Z předchozí analýzy jednotlivých požadavků jasně vyplývá, že ZigBee je pro naši aplikaci vhodnější a jeho vlastnosti se spíše slučují s požadavky na bezdrátovou komunikaci robota. Můžeme tedy konstatovat, že použitím ZigBee modulu jsme celkově rozšířili komunikační možnosti robota.

6 Závěr

Na základě analýzy a zhodnocení možností vybraných parametrů základních komponent, které byly k dispozici pro řešení byl navržen model lokální komunikace mezi robotem Lego MindStorm a modulem MicroChip. Byla fyzicky implementována podpora komunikace přes ZigBee a provedeno testování komunikace. Bylo potvrzeno, že navržené řešení a dané zapojení je funkční a že tímto způsobem je možné rozšířit NXT kostku o ZigBee modul. Testováním bylo prokázáno, že komunikační protokol je spolehlivý a po navázání spojení lze NXT Kostku ovládat pomocí koncového zařízení ZigBee.

Pokud by se však zadaná úloha a dané téma mělo řešit znovu, tak bych rozhodně nedoporučoval použít zadaný vývojový kit od MicroChipu. Hlavním negativním aspektem je skutečnost, že I²C rozhraní je v tomto vývojovém kitu vyloučeno z použití, protože sdílí piny s SPI sběrnici a ta je již použita pro komunikaci s modulem ZigBee. To vedlo k nutnosti použít hardwarový převodník, který částečně omezil výsledný návrh zařízení, zejména co do kompaktnosti fyzického provedení a komplexnosti komunikačního protokolu. Převodník byl sestaven mimo hlavní demonstrační desku (z důvodu možného snadného poškození demonstračního kitu). S demonstračním kitem je propojen napájecími vodiči a jedním, případně více vodiči datovými. Po dalších úpravách je navržené řešení možné použít v projektu ILERT.

Z porovnání protokolů ZigBee a Bluetooth vyplývá, že ZigBee je vhodnější pro tento typ použití, tj. na NXT kostce, která je napájena bateriemi a předpokládá se její použití v proměnných podmínkách.

Jako další cíle pro využití ZigBee modulu na platformě Lego MindStorm bych viděl plné zapojení robota do složitějších sítí, kde by i sám robot poskytoval data ze svých senzorů pro ostatní členy sítě. Dále bych doporučil implementaci zabezpečení komunikace na vyšších síťových vrstvách a nakonec, jako nejvyšší cíl integrování výše zmíněných funkcí do standardního firmwaru.

Literatura

- [1] WIKIPEDIA, otevřená encyklopedie, Universal Serial Bus, Červen 2008, dokument dostupný na adrese: http://cs.wikipedia.org/wiki/Universal_Serial_Bus
- [2] WIKIPEDIA, otevřená encyklopedie, I²C, Červen 2008, dokument dostupný na adrese: <http://cs.wikipedia.org/wiki/I²C>
- [3] SOURCEFORGE.NET, projektu BrixCC, dokument, program a zdrojové kódy dostupné na adrese <http://bricxcc.sourceforge.net/>
- [4] LEGO Group, NXT, Červen 2008, dokument dostupný na adrese http://mindstorms.LEGO.com/Overview/The_NXT.aspx
- [5] Steve Hassenplug, NXT Programming software, Červen 2008, dokument dostupný na adrese: <http://www.teamhassenplug.org/NXT/NXTSoftware.html>
- [6] WIFI™, BLUETOOTH™, ZIGBEE™ AND WIMAX™, H. LABIOD (INT, Evry, France), H. AFIFI (ENST, Paris, France), C. DE. SANTIS (INT, Evry, France), 2007
- [7] ZigBee Alliance: ZigBee Specification v 1.0 ZigBee Alliance Board of Directors, 2004, dokument dostupný na <http://www.zigbee.org>
- [8] IEEE: IEE 802.15 WPAN™ Task Group 4 (TG4), srpen 2006, dokument dostupný na <http://www.ieee802.org/15/pub/TG4.html>
- [9] Flowers D.: AN965 AN965 – Microchip Stack for ZigBee Protocol[datasheet], Microchip Technology Inc., 2006, USA. Dokument dostupný na adrese <http://ww1.microchip.com/downloads/en/AppNotes/00965c.pdf>, nebo na přiloženém DVD
- [10] ZigBee Specification. Technická zpráva, ZigBee Alliance, 2008.
- [11] WIKIPEDIA, Otevřená encyklopedie, ZigBee, červen 2008, dokument dostupný na <http://cs.wikipedia.org/wiki/ZigBee>
- [12] WIKIPEDIA, Otevřená encyklopedie, Mesh networking, červen 2008, dokument dostupný na http://en.wikipedia.org/wiki/Mesh_networking
- [13] Jon Adams, Bob Heile, Busy as ZigBee, IEEE, říjen 2006, dokument dostupný na <http://www.spectrum.ieee.org/oct06/4666>

- [14] WIKIPEDIA, Otevřená encyklopedie, Bluetooth, červen 2008, dokument dostupný na <http://en.wikipedia.org/wiki/Bluetooth>

- [15] LEGO MINDSTORMS NXT Hardware Developer Kit, The LEGO group, 2006, dokument dostupný na CD přiloženém k NXT kitu.

- [16] Datasheet_PCF8574P, Philips Electronics, 1997, dokument dostupný na www.datasheetscatalog.com

Zkratky

ACK	Acknowledgement
AES	Advanced Encryption Standard
AP	Access Point
DVD	Digital Versatile Disc
H	Hight
HTML	Hyper Text Markup Language
I ² C = IIC	Iter Integrated Circuit
IDE	Integrated Development Enviroment
IEEE	Institute of Elektrical and Elktronic Engineers
L	Low
LED	Light Emiting Diode
NQC	Not Quite C
OEM	Original Equipment Manufacturer
SDK	Software Development Kit
SKKE	Symetric-Key Key Exchange
SPI	Serial Periheal Interface
USB	Universal Seríal Bus
WPAN	Wireless Personal Area Network
ZED	ZigBee End Device

Seznam příloh

Příloha 1. LEGO MINDSTORMS NXT Hardware Developer Kit (na přiloženém DVD)

Příloha 2: BricxCC IDE (na přiloženém DVD)

Příloha 3: LabView (na přiloženém DVD)

Příloha 4. Ukázkové zdrojové texty v NXC (na přiloženém DVD)

Příloha 5. PIC18F2525/2620/4525/4620 Data Sheet Kit (na přiloženém DVD)

Příloha 6. PCF 8574A Data Sheet Kit (na přiloženém DVD)

Příloha 7. Výčet programovacích jazyků

Tabulka 3 – Výčet programovacích jazyků [5]:

Vlastnosti	NXT-G Retail	NXT-G Educational	RoboLab 2.9	NBC	NXC	RobotC	NI LabVIEW Toolkit	leJOS NXJ	pbLua	LEJOS OSEK
Typ jazyka	Grafický	Grafický	Grafický	Assembly	C-like	C	Grafický	Java	Lua	ANSI C
Firmware	Standardní	Standardní	Standardní(#1)	Standardní	Standardní	Standardní(#1)	Standardní	Vlastní	Vlastní	Vlastní
IDE (obsaženo?)	Ano	Ano	Ano	Ano	Ano	Ano	Ne (#6)	pluginy pro Eclipse a NetBeans.	Ne (#7)	Eclipse CDT(GCC+ATMEL SAM-BA)
Windows	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano (#7)	Ano
Mac OSX	Ano	Ano	Ano	Ano	Ano	Net Yet	Ano	Ano	Ano (#7)	Ne
Linux	Ne	Ne	Ne	Ano	Ano	Ne	Ne	Ano	Ano (#7)	Ne (maybe)
Události	Ne	Ne	Ano	Ne	Ne	Ano	Ne	Standardní Java events		Ano (OSEK RTOS)
Multithreading	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano		Ano (OSEK RTOS)
Bluetooth Brick to PC	Ano	Ano	Ne	Ano	Ano	Ano	Ano	Ano	Ano	Ano
Bluetooth Brick to Brick	Ano	Ano	Ne	Ano	Ano	Ano	Ano	Ano	Ano	Zatím ne
Bluetooth Brick to Other Device	Ne	Ne	Ne	Ne	Ne	Ano	Ne	Ano	Ano	Ne

I²C podpora	(#5)	(#5)	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano (US pouze)
File System	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Zatím ne	Net plánováno
Floating Point	Ne	Ne	Ano	Ne	Ne	Ano	Ne?	Ano	(#8)	Ano
Datalog	Ne	Ne	Ano	Ne	Ne	Ano	Ne	Ano	Ne	Net Yet
Jak získat	Included With retail version of NXT	Available with Educational Version of NXT	Available with Educational Version of NXT	BricxCC Web Site	BricxCC Web Site	RobotC.net or LEGO Education	LabVIEW toolkit Site (#6)	Free download z lejos Web Site	pbLua Site	LEJOS OSEK site
Podpora zařízení 3. stran	Ano	Ano	?	?	Ano	Ano	?	Ano	?	?
Co chcete dělat... (#2)										
Pohnout s roboty, aniž znáte jazyk	Ano	Ano	Ano							
Naučit se programovat s NXT				Ne						
Psat rychlé programy				Ano	Ano	Ano		Možná?	Ano	Ano
Psat rychle programy	Ano	Ano	Ano							
Naučit se				?	Ano	Ano	Ano	Ano	Ano	Ano

pokročilé programovat										
Vhodné pro (#3)	Lidé začínající s NXT dělající jednoduché úkoly	Školy začínající s NXT	Školy přecházející z RXT na NXT	Pokročilí programátoři.	C programátoři	Realtime aplikace	LabVIEW uživatelé, či lidé chtějící vylepšit NXT-G	Java programátoři	Vzdělání uživatelé	Vestavěná zařízení, C/C++ programátoři.
	NXT-G Retail	NXT-G Educational	RoboLab 2.9	NBC	NXC	RobotC	NI LabVIEW Toolkit	leJOS NXJ	pbLua	LEJOS OSEK
Testovací program (#4)										
Testovaná verze	1.1	2.9							13c	
Rychlost (smyček/min)	762	73k			4285	103k	750/5350 (#9)		18k	2695
Rychlost (#10)						134k/412k			31k	334k/18144k
Paměť (bytes)	10704 Bytes	559 Bytes			1428 Bytes	561 Bytes	8084/1890		1750 Bytes	18240 (včetně firmware)
Čas k napsání	10 min	~20 min			30 min	30 min	15 min/80 min			30 min
Program	Kód, Grafický	Kód, Grafický			Kód	Kód1, Kód2, Autorovi komentáře	Kód, Grafický / Kód1 + Kód2, Grafický		Kód 1 Kód 2	Kód

Ostatní komponenty (Fakty & Názory)	Tento soft je určen pro děti, ale je příliš omezující pro dospělé	Stejně jako Reatil verze, jen pro „Academy“ roboty	LEGO tvrdí, že tohle je poslední verze LoboLabu.				Může být použit ke tvorbě bloků fungujících v NXT-G nebo může tvořit programy ke stažení přímo do NXT Kosky, nebo tvoří PC programy ovládající NXT Kostaku.		Ve vývoji	LEJOS OSEK je NET část leJOS projektu, ale používá části leJOS Kódu.
--	---	--	--	--	--	--	---	--	-----------	--

Poznámky:

- (1) RoboLab a RobotC požívají standardní firmware.
- (2) **Nic** z těchto není "**Konečná odpověď**" v programování, ale možná vám pomohou ve výběru nevhodnější varianty.
- (3) Toto je spíše názor těch, kteří tento software používají.
- (4) Vytvořil jsem "[testovací program](#)" který běží na všech platformách abych porovnal rychlost, paměťovou náročnost jednotlivých jazyků.
- (5) Standardní firmware podporuje PC, ale NXT-G nemá přímé funkce k jejich ovládní.
- (6) LabVIEW Toolkit pro NXT je add-on pro LabVIEW. Z toho důvodu musíte mít vlastní kopii LabVIEW.
- (7) Programy jsou kompilovány na LEGO NXT brick
- (8) Lua podporuje něco podobné plovoucí řádové čárce. Více Inko na jejich stránkách.
- (9) Dva NI programy byly napsány pomocí The Labview Toolkit & vysoko-úrovňových volání a Toolkitu a nízko-úrovňových částí.
- (10) Většina softwaru pracuje rychleji, pokud je optimalizována. Tato řádka má optimalizované hodnoty