

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

## ŘÍZENÍ ROBOTY POMOCÍ FITKITU

BAKALÁŘSKÁ PRÁCE

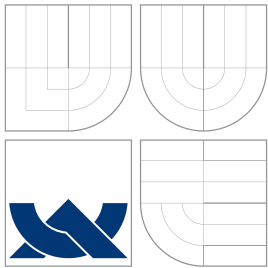
BACHELOR'S THESIS

AUTOR PRÁCE

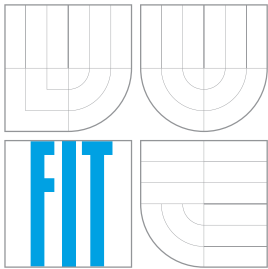
AUTHOR

TOMÁŠ NOVOTNÝ

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

## ŘÍZENÍ ROBOTY POMOCÍ FITKITU

ROBOT CONTROL USING FITKIT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

TOMÁŠ NOVOTNÝ

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JAROSLAV ROZMAN

BRNO 2008

## Zadání bakalářské práce

Řešitel: **Novotný Tomáš**  
Obor: Informační technologie  
Téma: **Řízení robota pomocí FITkitu**  
Kategorie: Umělá inteligence

**Pokyny:**

1. Seznamte se s jednotlivými funkcemi robota a s jeho senzory
2. Seznamte se s funkcí sběrnice I2C a jejím použitím na FITkitu
3. Navrhněte funkce pro ovládání robota a čtení jeho senzorů
4. Navržené funkce implementujte

**Literatura:**

- BasicATOMPro Syntax Manual, manuál k procesoru

Při obhajobě semestrální části projektu je požadováno:

- první tři body zadání

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: **Rozman Jaroslav, Ing.**, UITS FIT VUT

Datum zadání: 1. listopadu 2007

Datum odevzdání: 14. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
Fakulta informačních technologií  
Ústav inteligentních systémů  
602 00 Brno, Božetěchova 2

---

doc. Dr. Ing. Petr Hanáček  
*vedoucí ústavu*

**LICENČNÍ SMLOUVA  
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO**

uzavřená mezi smluvními stranami

**1. Pan**

Jméno a příjmení: **Tomáš Novotný**  
Id studenta: 79307  
Bytem: Kubíkova 2294/18, 628 00 Brno  
Narozen: 18. 08. 1985, Brno  
(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: Řízení robota pomocí FITkitu  
Vedoucí/školitel VŠKP: Rozman Jaroslav, Ing.  
Ústav: Ústav inteligentních 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 zpracování 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 užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnožení.
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.

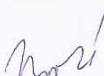
## Článek 3 Závěrečná ustanovení

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.

V Brně dne: .....

.....

Nabyvatel

  
.....

Autor

## Abstrakt

Práce se zabývá teoretickým popisem, návrhem a implementací dálkově ovládaného robota s několika čidly, který je postaven na dostupných zařízeních. K řízení na straně robota je využíván FITkit, který komunikuje s počítačem, zpracovává informace získané z čidel a řídí ovládání motorků. Většina senzorů používá rozdílné rozhraní, ať jde o sběrnici I<sup>2</sup>C, PWM nebo analogové signály. Zpracování řídí mikrokontrolér na FITkitu, zdrojové kódy jsou psány v jazyce C. FPGA je využíváno pro komunikaci pomocí COM portu. Při návrhu a implementaci byl kladen důraz na pozdější rozšíření robota. Výsledkem práce je dálkově ovládaný robot se sonary, kompasem, akcelerometrem a optickými enkodéry.

## Klíčová slova

Robot, řízení, senzor, FITkit, I<sup>2</sup>C, PWM, A/D převod, 4WD1, sonar, SRF08, kompas, CMPS03, akcelerometr, ACC7260, optický enkodér, QME-01, HandyPort HPS-120

## Abstract

This paper deals with the theoretical background, design and implementation of the remote controlled robot, which is shipped with various sensors. Work is based on the available devices. FITkit is used for controlling sensors, motors and communication with the PC at the robot side. The different interfaces are used by almost every sensor. These are the I<sup>2</sup>C bus, PWM or analog signals. The biggest part of the controlling is done by the FITkit using programming C language. The FPGA is used just for the communication over the COM port. Later expansion was in the mind during the design and implementation. The result of this work is remote controlled robot with the sonars, compass, accelerometer and optical encoders.

## Keywords

Robot, control, sensor, FITkit, I<sup>2</sup>C, PWM, A/D conversion, 4WD1, sonar, SRF08, compass, CMPS03, accelerometer, ACC7260, optical encoder, QME-01, HandyPort HPS-120

## Citace

Tomáš Novotný: Řízení robota pomocí FITkitu, bakalářská práce, Brno, FIT VUT v Brně, 2008

# Řízení robota pomocí FITkitu

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Jaroslava Rozmana. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Tomáš Novotný  
14. května 2008

## Poděkování

Rád bych poděkoval Ing. Jaroslavu Rozmanovi za odborné vedení a zapůjčení senzorů potřebných pro práci.

© Tomáš Novotný, 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

<b>1 Úvod</b>	<b>3</b>
<b>2 Teoretický rozbor</b>	<b>4</b>
2.1 4WD1 Robot Kit	4
2.2 Atom Bot Board	4
2.3 FITkit	5
2.3.1 Mikrokontrolér	5
2.3.2 FPGA	5
2.4 I <sup>2</sup> C	5
2.5 PWM	6
2.6 Sonar	6
2.7 Kompas	7
2.8 Akcelerometr	7
2.9 Optický enkodér	8
2.10 Scorpion	8
2.11 HandyPort HPS-120	9
<b>3 Návrh řešení</b>	<b>10</b>
3.1 Senzory	10
3.1.1 Sonar a kompas	10
3.1.2 Akcelerometr	11
3.1.3 Optický enkodér	11
3.2 Pohyb	11
3.3 Řízení	12
3.4 Napájení	12
3.5 Shrnutí	13
<b>4 Implementace</b>	<b>14</b>
4.1 I <sup>2</sup> C	14
4.1.1 Hardwarová část	14
4.1.2 Práce se sběrnici	16
4.1.3 Zapojení FITkitu	16
4.1.4 Sonar	17
4.1.5 Kompas	18
4.2 A/D převod	19
4.3 Impulsy	20
4.3.1 Převodník úrovní	20
4.3.2 Výpočet rychlosti	20



4.4	PWM	20
4.4.1	Převodník úrovní	21
4.4.2	Práce s časovačem	21
4.4.3	Zapojení FITkitu	22
4.5	COM port	22
4.5.1	Napájení HPS-120	23
4.5.2	Komunikace s FPGA	23
4.6	Napájení	23
4.7	Shrnutí zapojení FITkitu	24
<b>5</b>	<b>Závěr</b>	<b>25</b>
	<b>Literatura</b>	<b>26</b>

# Kapitola 1

## Úvod

Konstrukce robotů už dnes není pouze výsadou výzkumných týmů a tvorba těch jednodušších se začíná objevovat na velkém množství míst. Spolu s tímto jevem se zvyšuje poptávka po senzorech, motorech a dalších částech potřebných pro konstrukci robotů. Za přijatelné ceny jsou dnes běžně dostupná různá čidla a zařízení. Spolu s mikrokontroléry získáme mocný nástroj pro tvorbu rozličných projektů, ať už pro zábavu nebo seriózní průmyslové aplikace.

Jelikož je toto prostředí zajímavé a vyvstává zde velké množství možných problémů k řešení, jeví se jako první potřebný krok pro další rozvoj vytvoření základního stavebního kamene. Pojem „základní kámen“ nabízí fantazii velký prostor. Může jít o podvozek na kolečkách, helikoptéru nebo třeba loď. Obklopeni zdmi kanceláří a laboratoří sáhneme právě po první nabízené variantě, která nabízí rozsáhlý prostor pro využití. Odrazovým můstkem bude dálkově ovládaný robot s několika různými čidly, který bude veškeré získané informace předávat na vzdálený počítač.

Pro práci bude stěžejní jedno z vybavení dostupné na Fakultě informačních technologií. To tvoří robot 4WD1 doplněný senzory, který je ovládaný deskou Atom Bot Board. Její funkční rozšíření je však omezené. Proto přichází na řadu mocnější zařízení – FITkit. Vybaven výkonným mikrokontrolérem a programovatelným hradlovým polem dává do budoucna velký potenciál. Nejprve bude potřeba nahradit desku Atom Bot Board a dále pokračovat rozšiřováním o další čidla. V návrhu a implementaci musí být brán ohled na budoucí rozšiřování, tedy šetřit prostředky v používaných zařízeních. Práce je rozdělena do tří hlavních částí.

Kapitola Teoretický rozbor sestává z popisu zařízení a technologií, se kterými se v praktické části pracuje. Stručně je popsán i aktuální stav ovládaní robota. U každé části je uveden zdroj, který může sloužit k získání dalších informací. Velká část technických specifikací je dostupná on-line. Zde uvedený popis popisuje základní rysy a informace potřebné pro projekt.

Další kapitola Návrh řešení podrobně popisuje vybraná zapojení a moduly, které budou na mikrokontroléru využity. Pokud existuje více možností řešení, jsou různé přístupy popsány a zároveň zdůvodněn výběr. V závěru části je navržené řešení shrnuto.

V předposlední kapitole nazvané Implementace je podrobně popsáno zapojení jednotlivých částí, přičemž je doplněna schémata zapojení. Dále obsahuje vysvětlení a popis převodníků úrovní, které jsou pro správnou funkčnost mikrokontroléru a některých periférií nezbytné. Důležitou částí jsou i výpočty parametrů pro hodinové signály některých modulů mikrokontroléru.

# Kapitola 2

## Teoretický rozbor

V kapitole budou probrány principy a vlastnosti zařízení a součástí robota, které máme k dispozici. Z těchto částí bude robot sestaven. Stručně bude prezentováno i současné řešení ovládání robota deskou Atom Bot Board v projektu [19].

### 2.1 4WD1 Robot Kit

Základem pro tělo robota je 4WD1 Robot Kit od firmy Lynxmotion [8]. Jde o konstrukci pro vnitřní i venkovní použití. V našem případě je rozšířena o další plošinu. Ve vzniklém prostoru je umístěna část elektroniky (deska scorpion a sonary). FITkit je připevněn na horní plošinu. Následující obrázek ukazuje robota bez horní plošiny a s doplněnou horní plošinou (první dvě konstrukce).

Pohyb je zajištěn pomocí čtveřice motorků GHM-04 (Gear Head Motor) [9] o napájecím napětí 7,2 V. Pro snímání rychlosti a směru pohybu motorků jsou doplněny optickými enkodéry QME-01, které jsou blíže popsány v části 2.9.

V současné době se konstrukce nevyrábí. Nahrazena je novou verzí A4WD1 (Aluminum 4WD1). Ta není vybavena ani základní plošinou (lze ji doplnit), jde tedy jen o podvozek s kolečky. Jedná se o dvě poslední konstrukce na následujícím obrázku.



Obrázek 2.1: Lynxmotion 4WD1 a A4WD1 bez horní plošiny a s horní plošinou

### 2.2 Atom Bot Board

V současné době je robot ovládán pomocí desky Atom Bot Board [12]. Ta je připravena pro připojení 24 a 28 pinových mikrokontrolérů Basic Atom, Basic Atom Pro, Basic Stamp-2, OOPic-C atd. Na desce je kromě patice pro mikrokontrolér 16 digitálních I/O pinů,

4 analogové, 3 tlačítka a signalizační diody, speaker, konektor pro připojení Play Station 2 a CANON9 konektor pro programování.

Celkové ovládání je tedy limitováno schopnostmi připojeného mikrokontroléru. Nahrazením této desky FITkitem získáme mnohem více možností při práci s robotem.

## 2.3 FITkit

Platforma FITkit [5] slouží na Fakultě informačních technologií Vysokého učení technického v Brně pro výuku a praktickou realizaci softwarových a hardwarových projektů. Obsahuje mikrokontrolér, FPGA (Field-Programmable Gate Array), LCD displej, klávesnici s 16 tlačítky, pinheaders, DRAM a řadu konektorů. Jde tedy o univerzální řešení, které může být nasazeno ve spoustě projektů, kdy jedním z případů bude i ovládání robota. Dále bude podrobněji popsán mikrokontrolér a FPGA.

### 2.3.1 Mikrokontrolér

FITkit je osazen MCU (Micro-Controller Unit – mikrokontrolér) MSP430F168IPM firmy Texas Instruments [23]. Má nízké napájecí napětí (od 1,8 V do 3,6 V) a nízký příkon (v aktivním režimu 330  $\mu$ A při 1 MHz a 2,2 V). Jádro tvoří 16-bitový RISC mikroprocesor na rychlosti až 8 MHz (při 3,6 V). Vybaven je flash pamětí 48 KB + 256 B a 2 KB RAM. Pouzdro má 64 pinů, z toho je až 48 GPIO (General Purpose Input/Output).

Obsahuje několik modulů. USART0 (Serial Communication Interface), který lze přepnout do jednoho z režimů asynchronní UART, synchronní SPI nebo synchronní I<sup>2</sup>C. USART1 je podobný, jen nenabízí I<sup>2</sup>C. Mezi další moduly patří dva 16-bitové časovače (Timer\_A a Timer\_B), 12-bitový A/D a D/A převodník, komparátor a watchdog.

Pro přístup ke většině periférií na FITkitu je využíváno rozhraní SPI, které je připojeno do FPGA. Pro adresaci jednotlivých komponent v FPGA (například LCD displej, COM port) je využíván vlastní protokol [26].

### 2.3.2 FPGA

Dalším čipem FITkitu je programovatelné hradlové pole XC3S50-4PQ208C řady Spartan 3 firmy Xilinx.

Obsahuje 50 000 logických hradel, 1 728 logických buněk, 4 násobičky 18×18 bitů, až 124 uživatelských I/O, až 24 různých I/O standardů, 2 jednotky pro správu hodin a 12 kb distribuované paměti RAM.

## 2.4 I<sup>2</sup>C

Jde o sériovou sběrnici vyvinutou firmou Philips v roce 1982. Od té doby prošla několika revizemi. Název představuje zkratku Inter-Integrated Circuit. Sběrnice je tvořena dvěma vodiči a je orientována 8-bitově. V nejrychlejším režimu přenosu (High-speed) je možné dosáhnout rychlosti až 3,4 Mb/s. Je možné připojit několik zařízení typu master nebo slave, která jsou adresována pomocí unikátních 7 nebo 10-bitových<sup>1</sup> adres. Za ní následuje jeden bit, který určuje, zda se jedná o zápis (logická nula) nebo o čtení (logická jednička). Počet zařízení na sběrnici je limitován jen dostupností unikátních adres a celkovou kapacitou sběrnice, jejíž maximální hodnota může být 400 pF.

<sup>1</sup>10-bitové adresování je rozšíření, původní revize obsahovala jen 7-bitové.

V současné době je sběrnice využívána ve velkém množství zařízení jako paměti, A/D a D/A převodníky, senzory, LCD displeje a další.

Nejnovější revize 03 je z června 2007 a vytvořila ji firma NXP (založená firmou Philips). My se budeme zabývat předchozí revizí 2.1 z ledna 2000 [16], která je implementována v mikrokontroléru na FITkitu. Novinkou v nejnovější revizi je především rozšíření o další přenosovou rychlost Fast-mode Plus (až 1 Mb/s), která může pracovat na delších sběrnicích. Zároveň je zpětně kompatibilní se stávajícími rychlostmi.

Implementace sběrnice v zařízení není od 1. října 2006 zpoplatněna (ve standard a fast módu). Přidělení slave adresy zařízení však zůstává zpoplatněno částkou 2500 € [14].

Vodiče na sběrnici jsou pojmenovány SDA (Serial Data) a SCL (Serial Clock). Jak z názvů vyplývá, jeden se používá pro přenos dat a druhý pro přenos hodinového signálu. Tyto vodiče jsou při nečinnosti na sběrnici ve stavu logická jedna díky pull-up rezistorům (každý na jednom vodiči). Standard definuje tři přenosové rychlosti: 100 kb/s Standard-mode, 400 kb/s Fast-mode a 3,4 Mb/s High-speed mode.

## 2.5 PWM

Technika PWM (Pulse-width modulation – Pulsně šířková modulace) [20] slouží ke generování impulsního průběhu s konstantní periodou, ve kterém se mění šířka (střída) impulsu. Právě pomocí ní se přenáší informace. Pokud mluvíme o střídě 50 %, znamená délka impulsu 50 % z periody. U střídě 25 % je impuls čtvrtinový. PWM většinou slouží k ovládní elektromotorů, lze ho například využít také jako jednoduchý D/A převodník.

## 2.6 Sonar

Sonar (z anglického SOund NAvigation and Ranging) je jedno ze zařízení používaných pro detekci předmětů v okolí robota. Dalšími možnostmi jsou například infračervená nebo dotyková čidla.

Princip je vcelku jednoduchý. Zařízení měří čas, který trvá od vyslání do přijetí odraženého zvukového signálu. Ze znalosti rychlosti zvuku v prostředí (v našem případě vzduch) lze následně vypočítat vzdálenost objektu. Na prakticky stejném principu je založena echo-lokace netopýrů.



Obrázek 2.2: Sonar SRF08

Měření však nemusí být přesné a může být nepříznivě ovlivněno několika faktory. Zvuk může například narazit pod takovým úhlem, že dojde k jeho odražení od vysílajícího sonaru a ozvěna není zachycena. Falešné měření může také způsobit odraz od několika předmětů, který se vrátí za podstatně delší dobu – to vede k závěru, že objekty jsou od nás mnohem dále. Při použití více sonarů může docházet k příjmu zvuků vyslaných jiným sonarem. Tato

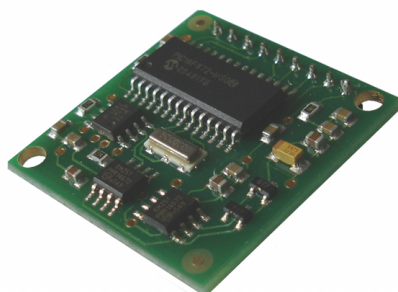
situace může nastat i při použití jednoho sonaru na robotovi, pokud je v okolí další robot se sonarem, který pracuje na stejné frekvenci.

Pro použití na robotovi jsou k dispozici sonary SRF08 firmy Devantech [4]. Ty jsou zároveň vybaveny světelným čidlem. Pracují na frekvenci 40 kHz, komunikačním rozhraním je pro ně sběrnice I<sup>2</sup>C a napájení je 5 V. Pro změření vzdálenosti je nejprve potřeba sonaru odeslat žádost, pak počkat 65 ms a poté číst výsledek. Rozsah měření je od 3 cm do 6 m. Výsledky mohou být v několika jednotkách (palce, centimetry nebo mikrosekundy) a ve dvou módech (změřená vzdálenost a vstup vhodný pro umělé neuronové sítě).

U sonaru je možné změnit jeho adresu. Jak bylo popsáno výše, zařízení na sběrnici musí mít jedinečnou adresu. Většinou se používá na sběrnici sonarů více.

## 2.7 Kompas

Dalším senzorem robota je kompas. K dispozici je CMPS03 firmy Devantech [2]. Pro komunikaci lze využít PWM signál nebo sběrnici I<sup>2</sup>C. Napájení je 5 V.



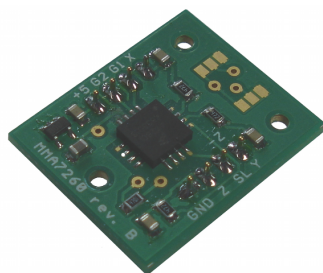
Obrázek 2.3: Kompas CMPS03

Při použití PWM reprezentuje úhel šířka pulsu. 1 ms představuje 0° a 36,99 ms znamená 359,9°. Mezi pulsy je signál na nízké úrovni 65 ms. U I<sup>2</sup>C komunikace máme možnost číst ve dvou formátech. Prvním je přečtení jednoho bajtu, který reprezentuje celý úhel. Navracená hodnota leží v rozmezí 0–255, přičemž nula odpovídá 0°. Jde o méně přesné, zato však jednodušší řešení. Druhou možností je přečíst dva bajty, výsledek je pak 0–3599, což přesně odpovídá úhlu (po vydělení deseti).

Tento digitální kompas je zkalibrován při výrobě. Při prvním vyzkoušení však dával při točení do všech směrů výsledky v rozmezí 240–300 stupňů. Po kalibraci začal ukazovat přesně (porovnáváno s buzolou). Kalibraci není potřeba provádět při každém startu, informace jsou uloženy v mikrokontroléru kompasu.

## 2.8 Akcelerometr

Mezi další senzory patří tříosý akcelerometr ACC7260 [22]. Umožňuje měřit zrychlení způsobené gravitací nebo nerovnoměrným pohybem. Napájení je 5 V a hodnoty zrychlení jsou předávány analogovým signálem. Vnitřní senzor pracuje na 3,3 V, takže i výsledné napětí nepřekročí tuto hodnotu. Na desce Atom Bot Board není připojen a nebyl ještě používán.



Obrázek 2.4: Akcelerometr ACC7260

Rozsah měření je nastavitelný na 4 hodnoty:  $\pm 1,5$  g,  $\pm 2$  g,  $\pm 4$  g a  $\pm 6$  g. Ty můžeme vybrat pomocí dvojice plošek na modulu (přepájením) nebo při provozu nastavením příslušných logických úrovní na pinech G1 a G2. Citlivost je závislá na zvoleném rozsahu. Při  $\pm 1,5$  g je to 800 mV/g, u  $\pm 6$  g je to 200 mV/g. Nulové zrychlení je reprezentováno polovinou napájecího napětí senzoru, tedy 1,65 V. Výstupní hodnoty napětí by měly být v rozmezí 0,45 V – 2,85 V.

## 2.9 Optický enkodér

Pro měření rychlosti a směru pohybu lze využít čtveřice optických enkodérů QME-01 (Quadrature Motor Encoder) [10] se senzorem firmy US Digital (označení senzoru je E4P-120-079-HT).



Obrázek 2.5: Optický enkodér QME-01 na motorku GHM-04 [10]

Napájecí napětí má hodnotu 5 V a výstupem je dvojice TTL signálů (2 kanály – A a B). Za jedno otočení hřídele přijde 120 impulsů na jednom kanálu. Pomocí fázového posuvu signálů zjistíme směr otáčení (signály A předbíhají B nebo obráceně).

## 2.10 Scorpion

Deska scorpion firmy Robot Power [18] slouží k ovládání motorů kol a případně dalšího motoru. Konstruována je jako duální H-můstek. Napájení z baterií může být v rozmezí 4,8 V až 25 V.



Obrázek 2.6: Scorpion Dual RC H-Bridge Motor Controller [11]

Ovládání motorků probíhá pomocí dvou PWM signálů s periodou 20 ms [11]. Pokud je šířka pulsu 1,5 ms, motory stojí. Při přibližování k 1 ms se motory začínají točit na jednu stranu, u 2 ms se točí na druhou stranu. Na výběr jsou dvě řízení: *mix* a *tank*. U prvního se jedním signálem řídí zatáčení a druhým rychlost. U řízení typu *tank* se každým signálem ovládá jedna strana motorků. Výběr je prováděn pomocí jumperu na desce.

K desce je možné připojit ještě jeden motor. U něj je ale ovládání umožněno jen od nuly až po plný výkon, zpětný chod nelze využít. Slouží například pro připojení zbraně.

## 2.11 HandyPort HPS-120

Dvojice zařízení HandyPort HPS-120 [6] firmy HandyWave Co., Ltd. slouží pro bezdrátovou komunikaci. Nahrazuje křížený sériový kabel (laplink).



Obrázek 2.7: HandyPort HPS-120

Pro bezdrátovou komunikaci využívá standard Bluetooth. Pracuje na frekvenci 2,4 GHz až 2,4835 GHz, dosah je až 100 metrů při přímé viditelnosti. Maximální přenosová rychlost je 115,2 kbps. Napájení je 5 V až 12 V. Pokud se spokojíme s továrním nastavením<sup>2</sup>, není potřeba žádná konfigurace a na koncových zařízeních není potřeba instalace software. V případě, že potřebujeme změnit parametry, k zařízení se připojujeme pomocí terminálového programu (například Hyper Terminal).

<sup>2</sup>Rychlost 9600 bps, 8 data bitů, bez parity, 1 stop bit, bez řízení toku.



# Kapitola 3

## Návrh řešení

Na základě teoretických znalostí bude navrženo řešení pro praktické zpracování. Při návrhu nás bude zajímat především úspora vývodů mikrokontroléru a FPGA. Jde o parametr, který je velmi důležitý při případném budoucím rozšiřování robota. Zároveň, pokud to bude možné, chceme využít knihoven FITkitu a dostupných modulů v mikrokontroléru.

### 3.1 Senzory

Pro zjištění informací o prostředí máme k dispozici čtyři typy senzorů: sonary, kompas, akcelerometr a optické enkodéry. U sonaru nemáme v možnostech komunikace na výběr – jedinou možností je sběrnice I<sup>2</sup>C. Akcelerometr a optický enkodér také nenabízí výběr. U kompasu jsou možnosti dvě: PWM a I<sup>2</sup>C.

#### 3.1.1 Sonar a kompas

U zmiňované sběrnice je nespornou výhodou prakticky neomezené množství zařízení, které můžeme připojit za pomoci dvou vodičů. Vzhledem k omezenému množství vývodů na straně mikrokontroléru a toho, že sonar nabízí jen sběrnici I<sup>2</sup>C, nemá smysl u kompasu využívat PWM. Jeho použití by zabralo další vodič, navíc sběrnice poskytuje přesnější výsledky.

Sběrnice I<sup>2</sup>C je v mikrokontroléru implementována pouze na USART0 a pokud nechceme provádět vlastní programovou implementaci, musíme použít právě toto rozhraní. Na FITkitu je rozhraní využíváno v režimu UART při komunikaci s terminálem (přes USB kabel). Toto spojení však nepotřebujeme, využíváme vlastního (bezdrátového) na dostupném COM portu. Po startu tedy můžeme terminál ukončit a přepnout USART0 na sběrnici I<sup>2</sup>C. Pro další rozšiřování robota je sběrnice velmi výhodná. K dispozici je velké množství zařízení počínaje paměťmi, A/D a D/A převodníky a dalšími.

Mikrokontrolér nabízí možnost využít několik přerušení při práci se sběrnici (například připraven k přenosu, připraven ke čtení). Vektor přerušení je však obsazen obsluhou odeslání pro režim UART (mikrokontrolér sdílí tyto vektory pro všechny režimy). Protože nechceme zasahovat do knihovny, přerušení nemůžeme využít. K dispozici tak máme jen příznakový registr, který můžeme číst ve smyčce. Čekací smyčka může být zdrojem „zatuhnutí“ systému, pokud budeme čekat na možnost odeslání, ale zařízení bude mít poruchu. Proto musíme ve smyčce sledovat příznak, který nás informuje o tom, že nedošlo očekávané potvrzení od protějšku.

### 3.1.2 Akcelerometr

Akcelerometr nám informace o zrychlení předává pomocí trojice analogových signálů (pro každou z os x, y a z jeden). Mikrokontrolér disponuje jedním 12-bitovým A/D převodníkem s osmi kanály. Kladné referenční napětí můžeme vybrat z několika zdrojů, mezi které patří interní 1,5 V a 2,5 V, signál  $AV_{CC}$  a externě připojený signál. Ten však nesmí přesáhnout hodnotu  $AV_{CC}$ , která je v případě FITkitu 3,3 V. Nám nejlépe vyhovuje dostupný signál  $AV_{CC}$ , který přesně odpovídá napájecímu napětí senzoru. Poloviční napětí 1,65 V určující zrychlení 0 g tedy bude přesně odpovídat polovině z maximální naměřitelné hodnoty.

U měření pomocí A/D převodu můžeme využít přerušení. Po zahájení můžeme pokračovat v dalším běhu programu a jakmile dojde k převodu signálu z poslední osy, aktualizujeme hodnoty zrychlení. Není potřeba žádného aktivního čekání.

### 3.1.3 Optický enkodér

K dispozici máme čtyři enkodéry. Každý předává informace o rotaci pomocí dvou signálů, to celkově znamená osm signálů.

Pro záchyt událostí v čase a generování PWM signálu (potřebných pro desku scorpion) jsou vhodné časovače. K dispozici jsou dva: `Timer_A` a `Timer_B`. Jejich funkce jsou velmi podobné, největší rozdíl je v počtu záchytných/porovnávacích registrů. U prvního čítače jsou k dispozici tři, u druhého sedm. Čítače mohou běžet ve třech režimech:

1. Čítání nahoru (až po hodnotu v prvním porovnávacím registru)
2. Nepřetržitě čítání (od 0 do 0xffff stále dokola)
3. Čítání nahoru a dolů (jako čítání nahoru s rozdílem, že po dosažení hodnoty se nezačíná od 0, ale čítá se k 0)

Pro záchyt časových intervalů nejvíce vyhovuje druhý režim. Výběr čítače je dán, v případě `Timer_A` není možné číst všechny enkodéry. Pracovat tedy budeme s `Timer_B`, který i tak může obsloužit maximálně sedm signálů. V tomto případě však využijeme celkově druhý čítač. Protože je to do budoucna nevhodné řešení, budeme snímat z každého enkodéru pouze jeden signál. To nám vystačí na výpočet rychlosti u každého kola.

Směr pohybu můžeme zjistit z dalších dvou zdrojů: z vlastního vysílaného PWM signálu pro ovládání motorků a přesněji akcelerometrem (tím můžeme také měřit rychlost). Zároveň z každého signálu můžeme zjistit, jestli nedošlo k zablokování nějakého motorku nebo vyjetí robota na překážku tak, že kola nemají kontakt s povrchem.

Modul `Timer_B` poběží v nepřetržitém režimu a signál bude zachytáván na vzestupné hraně pomocí přerušení. Díky tomu můžeme dále využívat funkci `DelayMS()` z knihovny `libfitkit`, která pracuje právě s druhým čítačem.

## 3.2 Pohyb

Deska scorpion pro ovládání motorků využívá dvou PWM signálů. Pro jednoduché řízení využijeme režim *mix*. Jedním kanálem ovládáme zatáčení, druhým rychlost. Pokud stojíme na místě a začneme zatáčet, robot se začne točit kolem své vlastní osy. Na každé straně robota jsou zapojeny dva motorky. Celkově ovládáme pomocí dvou signálů veškerý pohyb robota.

Pro generování PWM signálu se čítač využívá v režimu čítání nahoru. Do prvního registru se vloží délka periody a do dalších se vkládají jednotlivé časové úseky (určujeme střídu), ve kterých chceme vyvolat nějakou událost. Tou může být vyvolání přerušení nebo provedení akce s vybraným pinem (například nastavení, reset, změna). U PWM je nejjednodušší metodou využít událost s pinem. Vybereme jeden z možných pinů a nastavíme vlastnost, že se má pin při dosažení periody nastavit na logickou jedničku a u dosažení hodnoty v registru se má pin resetovat. Tím získáme signál o požadované střídě, jejíž hodnotu určíme jen nahráním příslušné hodnoty do registru. Obsluha je velmi jednoduchá a proces není potřeba složitě programově ošetřovat.

Poslední záležitostí je výběr čítače. Timer\_A má tři registry, což je pro nás přesný počet (jeden je pro určení periody a další dva pro určování střídy). Teoreticky by bylo možné sloučit generování PWM i odměrování časových intervalů v modulu Timer\_B – první registr použít pro periodu PWM, další dva pro generování střídy a zbylé čtyři pro záchyt impulsů. Problémem však je, že poslední vstup/výstup časovače (TB6) není vyveden na pinheaders, ale je využíván na FITkitu jako signál #FPROG.

### 3.3 Řízení

Robot má být řízen ze stanoviště. Pro bezproblémovou práci v nejrůznějších terénních podmínkách je potřeba využít bezdrátové spojení. Dostupný bluetooth modul HandyPort HPS-120 se připojuje do COM portů. Na FITkitu je k dispozici jeden výstup. Ten však není přímo propojen s mikrokontrolérem. V cestě je FPGA, které bude zatím v našem případě sloužit jako brána k perifériím na FITkitu. Komunikace probíhá v režimu SPI na USART1. Využití spojení s FPGA je do budoucna velmi vhodné, jelikož díky malému obsazení FPGA může být doplněna další funkcionalita.

### 3.4 Napájení

Nezbytnou součástí robota je napájení veškeré elektroniky a motorků. Jak bylo v předchozí kapitole popsáno, většina připojovaných zařízení pracuje na 5 V. Mikrokontrolér FITkitu však pracuje na 3,3 V. Proto je potřeba upravit propojení tak, aby nedošlo k poškození některé části.

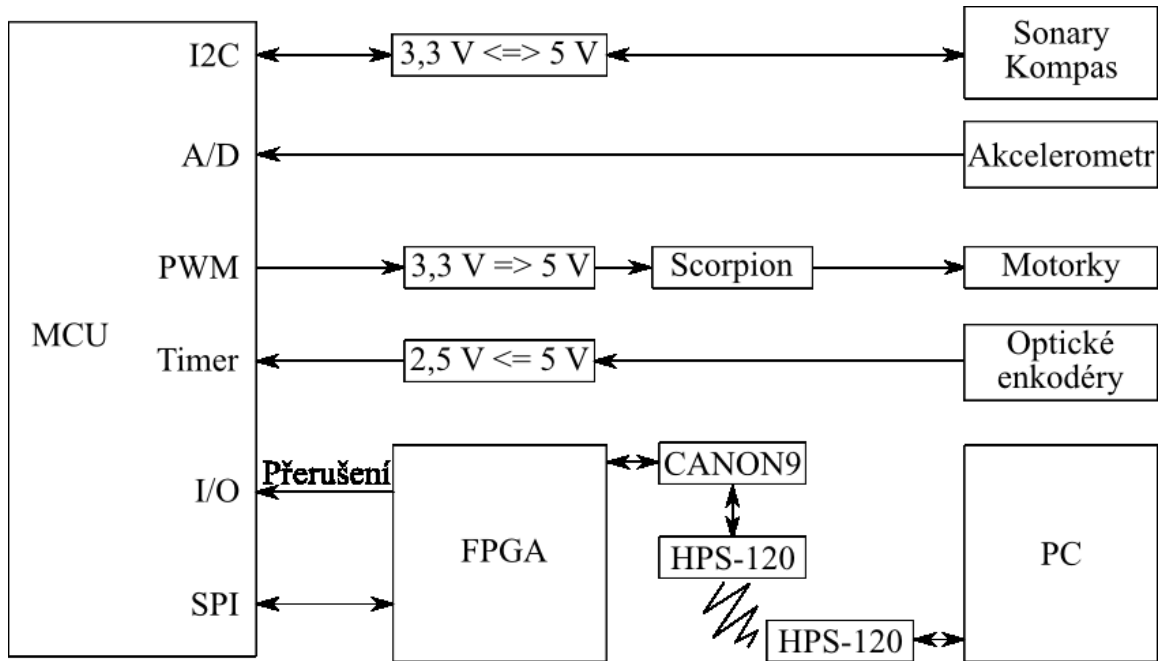
Podle [24] může mít vstupní napětí na pinech maximální hodnotu  $V_{CC} + 0,3$ . V případě FITkitu se  $V_{CC} = 3,3 V$ , tedy na vstup můžeme přiložit maximálně 3,6 V. Výstupy z FITkitu mají maximální napětí rovno  $V_{CC}$ .

Jak je vidět, vstupy nejsou tolerantní pro větší napětí a pokud nechceme mikrokontrolér poškodit, musíme vytvořit převodníky úrovní. U sběrnice I<sup>2</sup>C musí být tento převodník na obou vodičích obousměrný. Sběrnice sice podle specifikace umožňuje propojit zařízení s různými úrovněmi napětí, ale zařízení nižších úrovní musí být tolerantní vůči maximálnímu napětí na sběrnici. V našem případě tomu tak není.

Jako zdroj nám poslouží dvojice baterií o napětí 7,2 V. Jedna bude přes desku scorpion připojena přímo k motorkům. Druhá bude připojena na 5 V regulátor napětí. O převod na 3,3 V pro pull-up rezistory I<sup>2</sup>C se postará FITkit, který disponuje piny s tímto napětím.

### 3.5 Shrnutí

Pro práci se třemi sonary a kompasem využíváme USART0 v režimu I<sup>2</sup>C. Akcelerometr připojíme na 3 kanály z A/D převodníku. Ovládání motorků zajistíme pomocí PWM na Timer\_A, u Timer\_B obsadíme 4 kanály optickými enkodéry. Dálkové ovládání připojujeme na konektor CANON9 přes FPGA, se kterým komunikujeme rozhraním USART1 v režimu SPI. Propojení je úsporné a umožňuje v budoucnu využívat další nevyužité periferie mikrokontroléru a především FPGA. Následující obrázek shrnuje propojení.



Obrázek 3.1: Celkové zapojení zařízení

# Kapitola 4

## Implementace

Nedílnou součástí práce je praktická část. V té byla vytvořena aplikace pro ovládání popsaných částí robota. V následujících částech je popsáno, který soubor obsahuje funkce pro příslušnou periférii. Aplikace je uložena na přiloženém CD. V adresářové struktuře SVN FITkitu má být umístěna v `FITkit/apps/robot_control`.

Při vytváření sběrnic a dalších propojení bylo velmi dobrým pomocníkem kontaktní nepájivé pole, jelikož jednotlivá zapojení mohla být rychle a lehce měněna. Jedinou nevýhodou bylo připojování pinů, klasické byly vcelku krátké a občas propojení ztratilo kontakt. Nyní k jednotlivým částem robota.

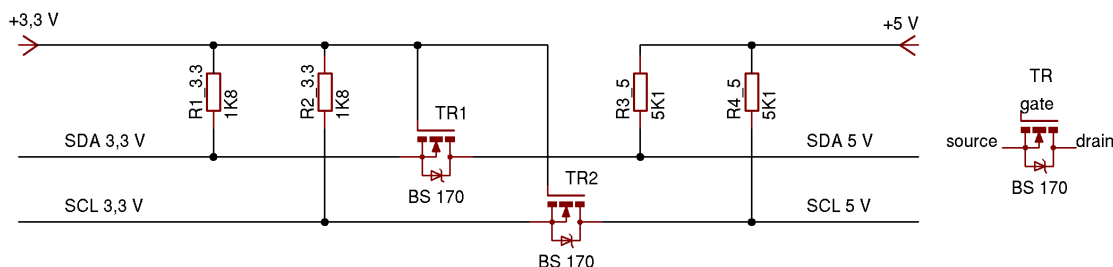
### 4.1 I<sup>2</sup>C

Pro práci se sběrnicí bylo nejprve potřeba vytvořit oboustranný převodník úrovní 3,3 V a 5 V tak, aby nedošlo k poškození mikrokontroléru. Pro následnou snadnou práci se zařízeními na sběrnici I<sup>2</sup>C byl vytvořen soubor funkcí v jazyce C (hlavičkový a zdrojový soubor `i2c.h` a `i2c.c`).

#### 4.1.1 Hardwarová část

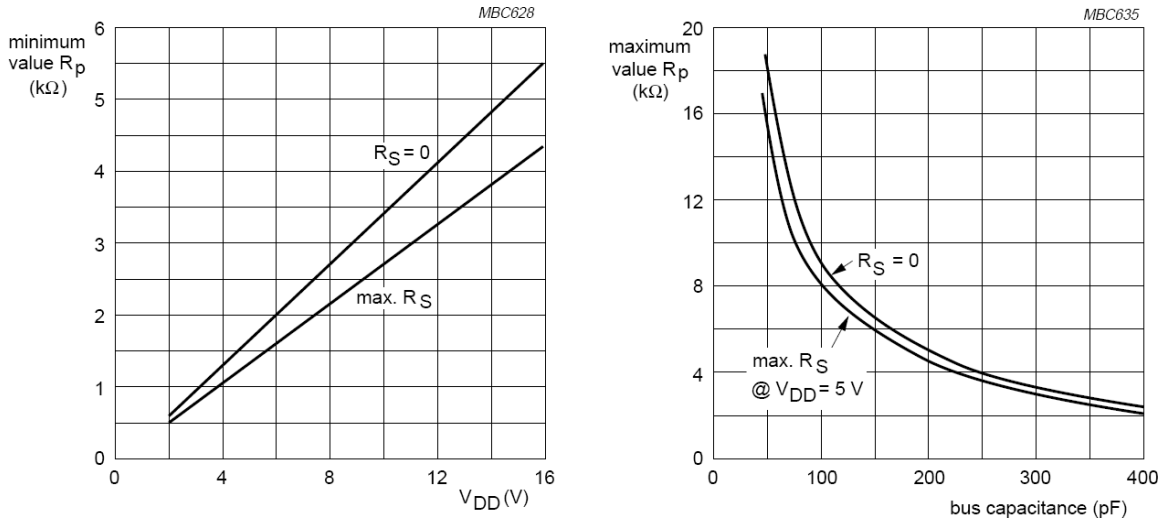
V [16] je navrženo řešení, které odděluje části s různým napětím pomocí dvojice tranzistorů MOS-FET (pro každý vodič jeden). Následující schéma podle navrženého řešení ukazuje zapojení převodníku. Tranzistory jsou BS 170 [15].

Na 3,3 V části sběrnice je připojen mikrokontrolér, na 5 V sonary a kompas. Sběrnici je možné doplňovat o další zařízení.



Obrázek 4.1: Schéma zapojení sběrnice I<sup>2</sup>C

Na sběrnici musí být dále přítomny pull-up rezistory. Jejich minimální a maximální hodnota je odvozena podle následujících grafů.



Obrázek 4.2: Grafy pro výpočet hodnot rezistorů  $R_p$ . Převzato z [16].

V našem případě je  $R_s$  rovno 0 (tyto rezistory nepoužíváme). Pro 3,3 V část byly vybrány odpory 1,8 kΩ, u 5 V pak 5,1 kΩ. Jak je vidět z grafů, hodnoty mohou být v relativně velkém rozmezí. Podle [13] odpovídá kapacitě sběrnice 400 pF přibližně 20–30 zařízení nebo 10 m kabelu. V našem případě jde o velmi malou sběrnici se třemi zařízeními (případně více při zapojení většího množství sonarů) a celkovou délkou kabelů přibližně 1 m.

Kapacita 3,3 V části sběrnice byla měřena digitálním multimetrem. Dvojice vodičů vedoucích z mikrokontroléru k MOS-FET tranzistorům má kapacitu 33 pF a délku 60 cm.

Další částí je nastavení hodinového signálu, který generuje jen master (mikrokontrolér). Budeme komunikovat maximálně na rychlosti 100 kb/s (standard-mode), frekvence nesmí překročit 100 kHz. Pro nastavení SCL se v mikrokontroléru využívá registrů I2CTCTL (dvěma bity se nastavuje zdroj hodinového signálu), I2CPSC (nastavení předděličky), I2CSCLH (délka SCL high) a I2CSCLL (délka SCL low). Hodinový signál, který vstupuje do modulu I<sup>2</sup>C se označuje I2CIN. Výsledný signál, který nastavíme pomocí registrů, má označení I2CCLK.

Jako zdroj I2CIN byl vybrán druhý externí oscilátor, který má frekvenci 7,3728 MHz (zdroj SMCLK). Druhou možností je první externí oscilátor (ACLK), ten má ale frekvenci 32,768 kHz, což je méně než 100 kHz. Výsledná hodnota předdělení se vypočítá I2CPSC + 1 (k hodnotě v registru se přičítá jednička a pak se dělí). Není vhodné do registru vkládat hodnoty větší jak 4 [25]. K nastavení by měly sloužit zbývající dva registry I2CSCLH a I2CSCLL. K jejich hodnotám se přičítá číslo 2. Pokud je hodnota v registru nižší než 4, výsledná hodnota je vždy 5.

Vztah pro výpočet výsledného signálu je

$$I2CCLK = \frac{I2CIN}{(I2CPSC + 1) \cdot ((I2CSCLH + 2) + (I2CSCLL + 2))}$$

Jelikož chceme dosáhnout maximální frekvence 100 kHz, po dosazení získáme

$$100 \geq \frac{7372,8}{(I2CPSC + 1) \cdot ((I2CSCLH + 2) + (I2CSCLL + 2))}$$

Úpravou vznikne vztah (neznámý jmenovatel je větší jak nula, můžeme násobit)

$$(I2CPSC + 1) \cdot ((I2CSCLH + 2) + (I2CSCLL + 2)) \geq 73,728$$

Hodnota v I2CPSC může být v rozmezí 0–4. V registrech I2CSCLH a I2CSCLL by měla být stejná hodnota (maximálně lišící se o jedna). Zvolíme tedy I2CPSC = 0 a vypočítáme.

$$\begin{aligned} 0 + 1 \cdot (2 \cdot (I2CSCL + 2)) &\geq 73,728 \\ 2 \cdot I2CSCL + 4 &\geq 73,728 \\ I2CSCL &\geq 34,864 \end{aligned}$$

Nejbližší větší celé číslo je 35. S tímto nastavením získáme frekvenci

$$I2CCLK = \frac{7372,8}{(0 + 1) \cdot ((35 + 2) + (35 + 2))} \doteq 99,63 \text{ kHz}$$

Podle [25] musí být frekvence I2CIN alespoň 10× větší než I2CCLK×výsledná hodnota předdělení. V našem případě je podmínka splněna:

$$\begin{aligned} I2CIN &> 10 \cdot I2CCLK \cdot (I2CPSC + 1) \\ 7372,8 &> 996,3 \end{aligned}$$

#### 4.1.2 Práce se sběrnici

Pro jednoduchou práci se sběrnici byla vytvořena knihovna funkcí. Její hlavičkový soubor se jmenuje `i2c.h`. V něm najdeme deklarace funkcí pro inicializaci, odeslání a příjem dat a čekací smyčky.

Přenos na sběrnici začíná vždy master zařízením (těch může být přítomno několik). Slave zařízení si vybere jedinečnou 7 bitovou adresou<sup>1</sup> a osmým bitem určí operaci (1 – čtení nebo 0 – zápis).

Následuje *start* stav, který zahajuje přenos. Od této chvíle je sběrnice považována za obsazenou. Stav je indikován přechodem z logické jedničky do logické nuly na SDA, zatímco SCL zůstává nezměněno (tedy logická jednička).

Podobně je reprezentován *stop* stav. Přechod z logické nuly do logické jedničky na SDA, přičemž SCL zůstává beze změny. Po tomto stavu je sběrnice považována za uvolněnou.

Speciálním stavem je *repeated start* (zkráceně *restart*), který je generován místo stavu *stop* (ten bude generována na konci). Občas je potřeba odeslat příkaz a hned začít číst výsledek, tedy změnit směr přenosu v průběhu jednoho vysílání. Po ukončení přenosu stavem *stop* by mohl jiný master zahájit přenos a přerušit tak naše plánované čtení. Proto po odeslání dat odešleme znovu stav *start* následovaný adresou zařízení a bitem pro zápis. Celý přenos ukončíme stavem *stop*. V rámci jednoho přenosu může master provést neomezené množství *repeated start*, avšak ukončen musí být právě jedním stavem *stop*.

#### 4.1.3 Zapojení FITkitu

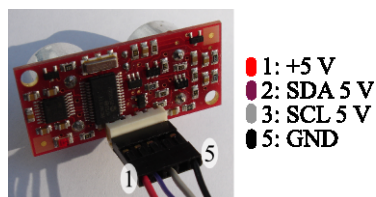
Velká část vývodů mikrokontroléru je vyvedena na pinheaders s označením JP9. Vývody z USART0 se liší pro mód UART a I<sup>2</sup>C. Pro sběrnici potřebujeme vývody SDA a SCL. Na straně mikrokontroléru jde o porty P3.1 a P3.3 (nožičky číslo 29 a 31). Signál SDA je přiveden

<sup>1</sup>Existuje speciální *general call* adresa určující všechna zařízení. Jde o 7 nul.

na JP9(16) a signál SCL najdeme na JP9(14). Správné zapojení je velmi důležité, protože při prohození sběrnice nefunguje. Barvy vodičů jsou pro sběrnici celkem čtyři, protože je rozdělena označení pro 3,3 V a 5 V část. Pro 3,3 V má SDA modrou a SCL bílou barvu. U 5 V je SDA vodič fialový, SCL šedý.

#### 4.1.4 Sonar

Sonary SRF08 jsou připojeny na 5 V část sběrnice I<sup>2</sup>C. Pro jednoduchou práci se sonarem byl vytvořen soubor funkcí. Deklarace najdeme v hlavičkovém souboru `srf08.h`. Zapojení ukazuje následující obrázek.



Obrázek 4.3: Zapojení SRF08

Jelikož se předpokládá, že sonarů bude na jedné sběrnici více, je možné vyměnit adresu z nabízených 16. Výchozí adresa je 0xe0. Jak je vidět, tato adresa má 8 bitů. Podle specifikace však existují pouze adresy o délce 7 bitů. V tomto případě uvádí výrobce dva typy adres – liché pro čtení ( $n + 1$ ) a sudé pro zápis ( $n$ ). Ve specifikaci [16] však takové určení není zmíněno. Adresu o 8 bitech navíc není možné zapsat do registru adresy v mikrokontroléru. Proto musí být posunuta o jeden bit doprava (vydělena 2). Výchozí adresa tedy bude mít tvar 0x70. Operace čtení nebo zápis se pak určuje příznakem v kontrolním registru přenosu.

Nabízené adresy sonarů jsou 0xe0, 0xe2, 0xe4, . . . , 0xfc, 0xfe. Na sonaru je červená SMD dioda, která po připojení napájení signalizuje adresu. Nejprve se jednou dlouze rozsvítí. Poté blikne krátce podle adresy. Pokud neblíkne, je nastavena výchozí adresa (0xe0). Jedno bliknutí znamená následující adresu (0xe2). Maximum je 15 zablíkaní, což označuje poslední adresu 0xfe.

Registr	Význam pro čtení	Význam pro zápis
0		Zahájení (provedení) akce
1		Omezení <i>maximum analogue gain</i>
2	Vyšší bajt výsledku	Změna dosahu měření ( <i>range</i> )
3	Nižší bajt výsledku	

Tabulka 4.1: Používané registry sonaru SRF08 [4]

Sonar se jeví jako zařízení se 36 8-bitovými registry (adresy 0–35), které mají význam podle čtení nebo zápisu. Pro čtení jsou dostupné všechny, pro zápis jen první tři. Tabulka shrnuje námi používané. Pro čtení nás zajímá registr 2 a 3, ve kterém najdeme horních a dolních 8 bitů z výsledku. V registru 1 je hodnota ze světelného senzoru. Registr 0 pro zápis je *command register*. Zápisem hodnoty určíme, co za akci chceme provést. Číslo 0x51 značí měření v centimetrech, dalšími příkazy můžeme například měnit adresu. Registr číslo 1 slouží ke změně *maximum analogue gain*, registr 2 ke změně dosahu měření (*range*). Změna adresy je zapsána do paměti sonaru, ostatní nastavení se ztrácí po odpojení napájení.



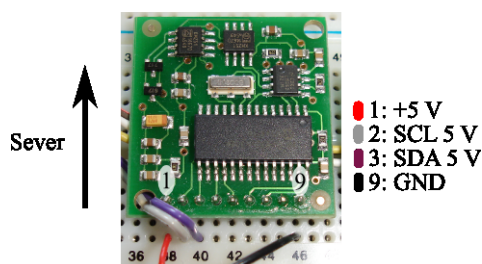
Komunikace probíhá tak, že nejprve odešleme žádost o měření. Začíná se stavem *start* na sběrnici, následuje adresa zařízení s příznakem zápisu a po ní hodnota 0 (zápis do *command register*) a 0x51 (měření v centimetrech). Přenos je ukončen stavem *stop*. Podle specifikace stačí počkat 65 ms, ale po této době sonar bez změny vnitřního nastavení (*range*) nedodal výsledky. Po uplynutí 66 ms už byly většinou k dispozici. V příkladu zapojení a demonstračního kódu [3] je také vkládána pauza 66 ms. Aby mikrokontrolér nečekal 65 ms a mohl provádět další operace, není vložena přímo čekací funkce. Místo toho se při inicializaci zahájí měření a při každém průchodu hlavní smyčkou se kontroluje, jestli už neuběhla požadovaná doba. Pokud ano, zažádá se o výsledek a spustí se další měření.

Po uplynutí stanovené doby tedy můžeme pokračovat čtením výsledku. Začíná se zase stavem *start* následovaným adresou zařízení a příznakem zápisu. Hodnota je 2 (chceme druhý registr). Provedeme stav *restart*, odešleme zase adresu zařízení s příznakem čtení a v této chvíli získáváme výsledek. Sonar SRF08 provádí autoinkrementaci. Posílá data od registru, který jsme specifikovali. Získáme tak tedy i druhý bajt výsledku, který je v registru číslo 3.

Při testování sonaru v pokoji byly vždy navraceny hodnoty maximálně 55 cm. Podle technické specifikace se má v tomto případě změnit *analogue gain*. Výchozí hodnota je na maximum (při měření se začíná na nejnižší hodnotě 94 a pokračuje se k nejvyšší 1025). Změnou na nejnižší hodnotu (zápisem do registru 1) začaly být výsledky přesné. Při namíření do volného prostoru (tedy většího než 6 metrů) vrátil sonar většinou hodnotu 0, což znamená, že nedetekoval žádný objekt. Proto bylo toto nastavení zahrnuto do inicializační funkce sonaru. Pro zkrácení doby měření na 65 ms bylo do inicializační funkce zahrnuto také mírné omezení dosahu měření. Stále je však nad hranicí 6 metrů, které dokáže sonar zpracovat. Standardně je měření nastaveno na 11 metrů.

#### 4.1.5 Kompas

Kompas CMPS03 [2] je také, stejně jako sonar, připojen na 5 V část sběrnice. Pro práci byl vytvořen také soubor funkcí, jejichž deklarace najdeme v hlavičkovém souboru *cmps03.h*. Zapojení ukazuje následující obrázek.



Obrázek 4.4: Zapojení CMPS03

Podobně jako u o sonaru je výchozí adresa uvedena jako 8-bitová 0xc0. Převedením na 7 bitů získáme 0x60. U kompasu se nepočítá, že jich bude přítomno na sběrnici více. Adresu tedy není možné měnit.

Kompas se jeví jako zařízení se 16 registry (adresy 0–15, viz. následující tabulka používaných registrů). Všechny kromě posledního jsou určeny pro čtení. Ve většině jsou systémové informace. Pro měření je důležitý registr 1, kde najdeme výsledek jako jeden bajt (úhel převedený na 0–255). V registrech 2 a 3 je stejná informace, pouze rozšířena na dva bajty (0–3599). Poslední registr 15 je určen pro zápis a slouží jako jeden ze způsobů kalibrace

kompasu. Ta probíhá zapsáním hodnoty 255 do registru, přičemž hodnotu zapisujeme celkem  $4\times$  a vždy máme kompas namířen na jednu ze světových stran. Druhou možností kalibrace je použití pinu 6. Kompas namíříme na jednu ze světových stran a pin 6 spojíme se zemí (0 V). Proces opakujeme  $4\times$  pro každou světovou stranu. Nezáleží na pořadí, v jakém na světové strany ukazujeme. Po kalibraci metodou s pinem začal kompas dávat přesné výsledky (porovnáváno s buzolou).

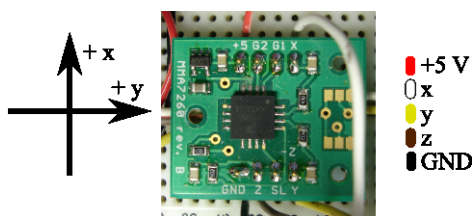
Registr	Význam
2	Vyšší bajt výsledku
3	Nižší bajt výsledku

Tabulka 4.2: Používané registry kompasu CMPS03 [2]

Čtení změřeného úhlu je relativně jednoduché. Komunikaci zahájíme stavem *start* na sběrnici. Pokračujeme zasláním žádosti o čtení z příslušného registru kompasu (1 nebo 2). Poté na sběrnici provedeme *restart* a zašleme adresu stejnou jako v předchozím kroku, ale s příznakem čtení. Kompas začne hned odpovídat. Díky autoinkrementaci stačí zaslat žádost jen o první registr (například 2), další jsou zasílány automaticky.

## 4.2 A/D převod

U akcelerometru, zatím jediného zařízení využívajícího A/D převodník, nebylo díky jeho internímu napájecímu napětí potřeba tvořit převodník úrovní. Jeho tři výstupy jsou přímo propojeny na první tři kanály A/D převodníku. Referenčním napětím je interní zdroj  $AV_{CC}$  o napětí 3,3 V. Pro práci s akcelerometrem byl vytvořen soubor funkcí, jejichž deklarace jsou v souboru `acc7260.h`.



Obrázek 4.5: Zapojení ACC7260

Rozsah měření byl zvolen na  $\pm 1,5$  g. V tomto případě nabízí modul nejvyšší citlivost 800 mV/g. Zároveň robot nedosahuje takové akcelerace, aby využil vyšší nabízený rozsah. V tomto rozsahu není potřeba upravovat signály G1 a G2 pro určení rozsahu.

Celkové zapojení je tedy velmi jednoduché. Z osmi pinů na modulu jich je zapojeno pět. Napětí 5 V, signál GND a osy x, y a z. Modul je osazen v nepájivém kontaktním poli.

Výpis a výpočet hodnot zrychlení je na první pohled napsán příliš složitě. Celé číslo se převádí na desetinné několika operacemi. To je způsobeno absencí konverzní specifikace `%f` v knihovně funkci `printf()` v mikrokontroléru. Není tedy možné přímo tisknout argumenty typu `float`.

Při provozu se objevil problém, že v klidovém stavu akcelerometr ukazoval v osách x a y zrychlení přibližně  $+0,2$  g. V ose z bylo zrychlení správné – tedy přibližně 1 g. Nejprve se nabízelo hledat chybu u A/D převodu, například příliš rychlé přepínání mezi kanály a vznik parazitních kapacit. Při měření digitálním multimetrem přímo na výstupech (zatížených

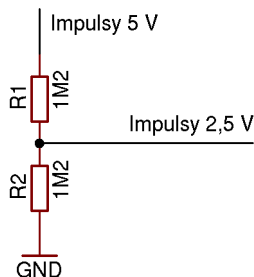
i nezatížených) však bylo zjištěno, že převod je správný. Na osách x a y skutečně nebylo napětí 1,65 V indikující 0 g, ale u osy x bylo přibližně 1,838 V a u osy y 1,809 V. Protože není známa příčina, byla v mikrokontroléru od naměřených hodnot odečtena přebytečná hodnota. Chyba je pravděpodobně konstantní, protože s kompenzací ukazovala osa x i y při namíření dolů očekávanou hodnotu 1 g.

## 4.3 Impulsy

U optických enkodérů nám rychlost vyjádří počet impulsů za jednotku času. Rychlost budeme měřit u každého motorku zvlášť. Vzhledem k signálům TTL logiky budeme muset pro mikrokontrolér vytvořit vstupní převodník úrovní z 5 V na hodnotu přijatelnou pro mikrokontrolér. Deklarace funkcí pro práci s enkodéry najdeme v souboru `qme01.h`.

### 4.3.1 Převodník úrovní

Podle [1] byl vytvořen převodník úrovní (napěťový dělič). Hodnoty odporů 1,2 M $\Omega$  jsou taktéž převzaty. Výsledné výstupní napětí je 2,5 V.



Obrázek 4.6: Schéma zapojení převodníku úrovně napětí pro výstup z optických enkodérů

### 4.3.2 Výpočet rychlosti

Enkodéry jsou připojeny na vnitřní hřídel motoru, za jednu její otáčku naměříme 120 impulsů. Vnitřní hřídel se točí v poměru 50:1 vůči vnější. Pro jednu otočku kola na vnější hřídeli tedy získáme 6000 impulsů. Tento údaj specifikuje i [21].

Pneumatiky mají průměr 3,6". Za jednu otočku robot ujede vzdálenost 28,727 cm. Impulsy budeme měřit v intervalech minimálně 65 ms, použijeme totiž již připravené odměřování pro dobu mezi zahájením měření a výsledkem u sonaru (měření může být o chvíli delší, přesný čas však víme vždy). Z počtu impulsů za jednotku času a znalosti vzdálenosti pro jednu otočku jsme schopni snadno vypočítat rychlost příslušného kola. Jeden impuls reprezentuje přibližně velmi malou vzdálenost 0,00479 cm.

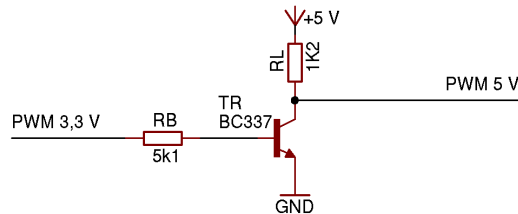
## 4.4 PWM

Pro ovládání motorků pomocí desky scorpion bylo potřeba vytvořit výstupní převodník úrovní z 3,3 V na 5 V z důvodu spolehlivosti zapojení. Jednoduchá práce se zařízením je zajištěna souborem funkcí v jazyce C (hlavičkový a zdrojový soubor `scorpion.h` a `scorpion.c`).

#### 4.4.1 Převodník úrovní

Podobně jako u zařízení na sběrnici I<sup>2</sup>C i scorpion je zařízení pracující na 5 V. Při testování pracovala deska zapojená na napětí 5 V a ovládaná pomocí 3,3 V signálu z mikrokontroléru. V [18] je popsáno, že příští verze firmware již bude obsahovat ovládání pomocí příkazů s TTL logikou. V té je logická jednička na vstupu reprezentována napětím od 2,0 V. Proto ovládání pravděpodobně funguje.

Protože ale chceme spolehlivé zapojení, musí být vytvořen výstupní převodník úrovní. Vytvořen byl podle [1]. Je sestaven z bipolárního tranzistoru NPN a dvou rezistorů. Zapojení má nevýhodu, že signál invertuje. Jednoduchou změnou nastavení generování PWM signálu v mikrokontroléru však můžeme výstup invertovat také, takže po průchodu převodníkem má signál požadované hodnoty. Zapojení musí být celkem dvě (dva signály – rychlost a zatáčení). Jedno ukazuje následující schéma. Pro zapojení byl vybrán tranzistor BC 337-16-TAP [17]. Hodnota R<sub>L</sub> byla zvolena 1,2 kΩ, R<sub>B</sub> bylo dopočítáno na hodnotu 5,1 kΩ.



Obrázek 4.7: Schéma zapojení převodníku úrovně napětí pro desku scorpion

#### 4.4.2 Práce s časovačem

Na mikrokontroléru je PWM signál generován modulem Timer\_A, který čítá nahoru v režimu *compare*. Výstupní mód je nastaven na *set/reset*. Hodnoty signálu jsou určeny trojicí registrů. V TACCRO je uložena perioda signálu. TACCR1 a TACCR2 slouží k určení střídy.

V nastaveném módu je signál nastaven do logické jedničky, pokud čítač dosáhne hodnoty v jednom z registrů TACCR1 nebo TACCR2. Při dosažení hodnoty v TACCRO (periody) je signál resetován, tedy nastaven do logické nuly. Z popisu je zřejmé, že signál má invertovaný průběh. Toto chování je žádoucí, neboť převodník úrovní na výstupu signál invertuje. Pokud by nebyl zařazen, použitý mód by byl *reset/set*.

Pohyb robota je určován pouze úpravou dvojice registrů. Ostatní nastavení je provedeno při inicializaci časovače (perioda, režim, hodinový signál atd.). Důležitým nastavením je zdroj hodinového signálu řízený registrem TACTL, který je vybrán dvojicí bitů. Jako u sběrnice I<sup>2</sup>C je vybrán SMCLK o frekvenci 7,3728 MHz. Vstupní předdělení může nabývat hodnot 1, 2, 4 a 8. Vybrána je hodnota 8, tím získáme časovou jednotku  $t_{TA}$  čítače Timer\_A

$$t_{TA} = \frac{1}{\frac{7,3728}{8}} \doteq 1,085\mu s$$

Hodnotu pro periodu 20 ms ukládáme do registru TACCRO. Snížena je o jedna, čítání probíhá od nuly. Získáme ji jednoduše:

$$TACCRO = \frac{T}{t_{TA}} - 1 = \frac{20\,000}{1,085} - 1 \doteq 18\,432$$

Podobným výpočtem získáme hodnotu pro 1 ms (minimum), 1,5 ms (střed – nic se ne-děje) a 2 ms (maximum). Hodnoty shrnuje následující tabulka.

Hodnota	Čas [ms]	Hodnota v registru
Minimum	1	921
Střed	1,5	1381
Maximum	2	1842
Perioda	20	18432

Tabulka 4.3: Hodnoty registrů čítače pro ovládání desky scorpion

### 4.4.3 Zapojení FITkitu

Propojení je provedeno na pinheaders JP9. Vývodů pro Timer\_A je několik, na mikrokontroléru jsou to porty P1.1 – P1.3 a P1.5 – P1.7. Porty P2.3 a P2.4 nenabízejí první výstup (TA0). Port P1.1 není na FITkitu vyveden na pinheaders. Porty P1.3 nebo P1.7 se používají pro přerušení od FPGA. Většinou se propojuje P1.7 (viz. část 4.5.2). Využit tedy můžeme z nabízených prakticky všechny. Stačí nám dva poslední výstupy (TA1 a TA2). První výstup je napojen na registr TACCRO, ve kterém je uložena perioda. Žádnou akci tedy neprovádí. Vybereme porty P1.2 a P1.3, které jsou přivedeny na JP9(29) a JP9(30). Barva vodičů je žlutá a zelená.

## 4.5 COM port

Připojený počítač komunikuje s robotem pomocí terminálu připojeného na COM portu. Bezdrátová komunikace využívá zařízení HPS-120. Pro obě strany se jeví spojení jako propojení dvou počítačů (křížený kabel – laplink).

Na straně FITkitu je využit dostupný COM port, který je připojen k FPGA. Komunikace mezi FPGA a mikrokontrolérem je zajištěna přes SPI s využitím přerušení. Obsluha v mikrokontroléru je podobná obsluze terminálu připojeného přes USB, jen jsou využity funkce s odlišnými názvy.

Na straně počítače se připojujeme ke COM portu pomocí terminálového programu. Parametry připojení jsou: rychlost 115 200 bps, 8 data bitů, 1 stop bit, bez parity. Zařízení HPS-120 musí být nastaveny také na tyto hodnoty. Při běhu je neustále vypisován následující řádek:

```
o Sonary: 25 31 24 [cm] Kompas: 119 [st] Akc: -0.003 +0.002 +1.002 [g] Sco:
1381 1381 Rych: 0 0 0 0 [100*cm/s]
```

Prvním znakem je animace nafukující se bubliny, která při konstantních hodnotách senzorů znázorňuje běh. Následují tři hodnoty ze sonarů od levého po pravý, stupně naměřené kompasem a zrychlení v osách x, y a z naměřené akcelerometrem. Dvě stejná čísla za zkratkou Sco jsou délky PWM signálu posílané desce scorpion. Hodnota 1381 znamená střed a tím pádem robot stojí. Poslední informací jsou rychlosti otáčení jednotlivých kol, začíná se u předních levým a pravým, pokračuje se u zadních levým a pravým.

Ovládání pohybu robota je pomocí kláves i, j, k, l. Ty tvoří tvar šipek. Pomocí klávesy o se robot zastaví (PWM signály se přepnou okamžitě do výchozího stavu).

### 4.5.1 Napájení HPS-120

Možností napájení zařízení je více. Na straně počítače je nejjednodušší metodou propojení napájecího konektoru s USB portem počítače pomocí dodávaného kabelu. U robota je zapojení obdobné, místo dodávaného je však využit kabel z projektu [19], který je místo do USB připojen do nepájivého kontaktního pole.

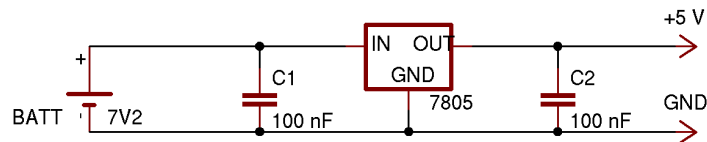
### 4.5.2 Komunikace s FPGA

Jak již bylo zmíněno, komunikace mezi FPGA a mikrokontrolérem probíhá přes SPI. Aby byl mikrokontrolér informován o přijetí a odeslání dat, musí být zapojeno přerušení od FPGA. To je zajištěno propojením pinů JP10(5) a JP9(26). Při vzniku přerušení se vytvoří 8-bitový vektor přerušení, kde jednotlivé bity určují přerušení. Využity jsou zatím dva – jeden pro příjem a druhý pro odeslání. Jakmile neprobíhá přenos přes SPI, je vyvoláno přerušení na pinu a mikrokontrolér přijme vektor. Podle jeho obsahu provede příslušnou obsluhu.

Problémem při práci zůstává nepřesnost v komunikaci mezi mikrokontrolérem a FPGA. Data z FPGA přichází posunuta o jeden bit doleva. Přijímané bajty je tedy potřeba v mikrokontroléru operací posunutí upravit.

## 4.6 Napájení

Nedílnou součástí robota je napájení, v našem případě z baterií. Robot má dva akumulátory o napětí 7,2 V. Jedním je napájena veškerá elektronika a druhý slouží pro napájení motorků. Pro převod ze 7,2 V na 5 V (pro elektroniku) byl použit stabilizátor napětí 7805 (v pouzdře TO-220). Zapojení zobrazuje následující schéma. Použité keramické kondenzátory mají hodnotu 100 nF.



Obrázek 4.8: Schéma zapojení 5 V stabilizátoru napětí 7805

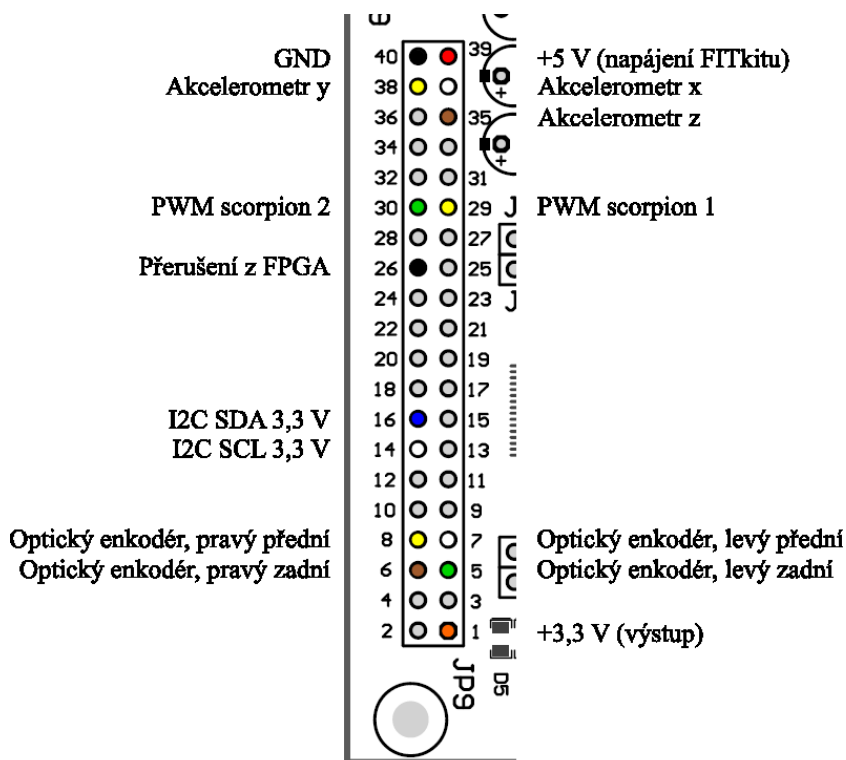
FITkit je připojen k regulátoru přes piny JP9(39) (5 V) a JP9(40) (GND).

## 4.7 Shrnutí zapojení FITkitu

Následující tabulka a obrázek shrnuje zapojení pinů na JP9 (v levém dolním rohu FITkitu). V tabulce jsou uvedeny čísla pinů a jejich význam. Obrázek znázorňuje umístění.

Pin JP9	Barva	Funkce	Port mikrokontroléru
1	Oranžová	3,3 V (pro I <sup>2</sup> C)	
5	Zelená	Vstup optického enkodéru, levý zadní	P4.2
6	Hnědá	Vstup optického enkodéru, pravý zadní	P4.3
7	Bílá	Vstup optického enkodéru, levý přední	P4.0
8	Žlutá	Vstup optického enkodéru, pravý přední	P4.1
14	Bílá	I <sup>2</sup> C SCL 3,3 V (hodinový signál)	P3.3
16	Modrá	I <sup>2</sup> C SDA 3,3 V (data)	P3.1
26	Černá	Přerušení z FPGA, připojeno na JP10(5)	P1.7
29	Žlutá	PWM scorpion 1	P1.2
30	Zelená	PWM scorpion 2	P1.3
35	Hnědá	Akcelerometr osa z	P6.2
37	Bílá	Akcelerometr osa x	P6.0
38	Žlutá	Akcelerometr osa y	P6.1
39	Červená	Napájení FITkitu 5 V (z regulátoru z baterie)	
40	Černá	GND (z regulátoru z baterie)	

Tabulka 4.4: Zapojení JP9 na FITkitu.



Obrázek 4.9: Zapojení JP9 na FITkitu. Podklad převzat z [5].

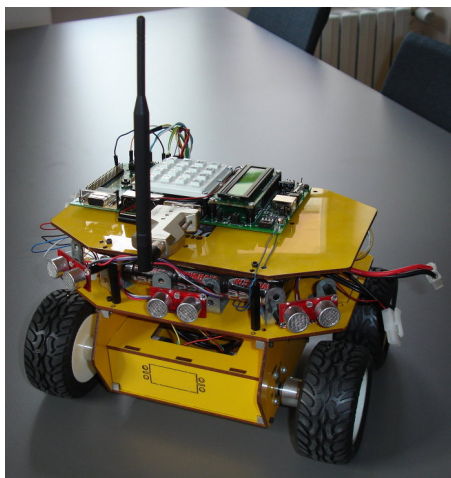
# Kapitola 5

## Závěr

Výsledkem práce je dálkově ovládaný robot s množstvím čidel, pomocí kterých získává nej-různější informace o sobě a okolním prostředí. Ze tří sonarů umístěných v přední části získáme přehled o blížících se předmětech, kompas nám napoví ve vlastním natočení a akcelerometrem zjistíme vlastní polohu. Problémy s otáčením kol můžeme identifikovat za pomoci optických enkodérů.

Jak bylo v úvodu popsáno, tento robot může sloužit jako odrazový můstek. Jedno z velmi zajímavých rozšíření by mohlo být připojení kamery a přenos obrazu na řídicí počítač. To přináší několik problémů (především přenosové pásmo), ale výsledek by mohl být zajímavý – spolu s autonomním pohybem a vlastní umělou inteligencí by bylo možné sledovat chování v prostoru a zkoušet různé řídicí algoritmy. Další možností rozšíření by mohlo být připojení robotického ramene, ostatně autor [7] ovládání jednoho takového vytvořil a v závěru práce upozorňuje, že zajímavá by byla jeho mobilizace.

Vedlejším produktem práce s FITkitem bylo zprovoznění nástrojů pro práci pod operačním systémem NetBSD, který používám. Tato oblast však není předmětem mé práce, proto zde není popsána. Důležitá jsou však zpracovaná čidla a knihovny pro práci s nimi. To umožňuje vybrat si jen potřebnou část a tu použít ve svém projektu. Zajímavá je také sběrnice I<sup>2</sup>C, protože se mi nepodařilo nalézt dostupnou knihovnu nebo ukázkovou aplikaci pro práci s ní na FITkitu. Při práci také vzniklo několik převodníků úrovní, které mohou pomoci v dalších aplikacích. Na závěr fotografie výsledného robota.



Obrázek 5.1: Výsledný robot



# Literatura

- [1] Bierl, L.: *Interfacing the 3-V MSP430 to 5-V Circuits*. Texas Instruments, Říjen 2002, [online], [cit. 2008-11-05].  
URL <http://focus.ti.com/lit/an/slaa148/slaa148.pdf>
- [2] Coe, G.: *CMPS03 documentation*. Devantech Ltd, [online], [cit. 2008-11-05].  
URL <http://www.robot-electronics.co.uk/htm/cmeps3doc.htm>
- [3] Coe, G.: *SRF08 to Basic Atom documentation*. Devantech Ltd, [online], [cit. 2008-11-05].  
URL <http://www.robot-electronics.co.uk/htm/srf08atom.shtml>
- [4] Devantech Ltd: *SRF08 Ultra sonic range finder*. [online], [cit. 2008-11-05].  
URL <http://www.robot-electronics.co.uk/htm/srf08tech.shtml>
- [5] Fučík, O.: FITkit : Úvod. [online], [cit. 2008-11-05].  
URL <http://www.fit.vutbr.cz/kit>
- [6] HandyWave: *HPS-120 HandyPort-Serial*. [online], [cit. 2008-12-05].  
URL [http://www.handywave-usa.com/downloads/HPS\\_120\\_manual\\_v2.0\\_english.pdf](http://www.handywave-usa.com/downloads/HPS_120_manual_v2.0_english.pdf)
- [7] Lengyel, K.: *Ovládání robotického manipulátoru mikrokontrolérem*. bakalářská práce, FIT VUT v Brně, Brno, 2007.
- [8] Lynxmotion, Inc.: *4WD 1*. [online], [cit. 2008-11-05].  
URL <http://www.lynxmotion.com/Category.aspx?CategoryID=57>
- [9] Lynxmotion, Inc.: *Gear Head Motor - 7.2vdc 50:1 175rpm (6mm shaft)*. [online], [cit. 2008-11-05].  
URL <http://www.lynxmotion.com/Product.aspx?productID=96&CategoryID=11>
- [10] Lynxmotion, Inc.: *Quadrature Motor Encoder w/Cable*. [online], [cit. 2008-11-05].  
URL <http://www.lynxmotion.com/Product.aspx?productID=448>
- [11] Lynxmotion, Inc.: *Scorpion Dual RC H-Bridge Motor Controller*. [online], [cit. 2008-11-05].  
URL <http://www.lynxmotion.com/Product.aspx?productID=89&CategoryID=10>
- [12] Lynxmotion, Inc.: *Atom Bot Board*. 2004, [online], [cit. 2008-11-05].  
URL <http://www.lynxmotion.com/images/data/abpdf01.pdf>
- [13] NXP Semiconductors: *Facts*. [online], [cit. 2008-11-05].  
URL [http://www.nxp.com/products/interface\\_control/i2c/facts/](http://www.nxp.com/products/interface_control/i2c/facts/)

- [14] NXP Semiconductors: *I<sup>2</sup>C Licensing information*. [online], [cit. 2008-11-05].  
URL [http://www.nxp.com/products/interface\\_control/i2c/licensing/](http://www.nxp.com/products/interface_control/i2c/licensing/)
- [15] ON Semiconductor: *BS170 Small Signal MOSFET 500 mA, 60 Volts*. Srpen 2005, [online], [cit. 2008-11-05].  
URL [http://www.onsemi.com/pub\\_link/Collateral/BS170-D.PDF](http://www.onsemi.com/pub_link/Collateral/BS170-D.PDF)
- [16] Philips Semiconductors: *The I<sup>2</sup>C-bus specification*. Leden 2000, [online], [cit. 2008-11-05].  
URL [http://www.nxp.com/acrobat\\_download/literature/9398/39340011.pdf](http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf)
- [17] Philips Semiconductors: *BC817; BC817W; BC337 45 V, 500 mA NPN general-purpose transistors*. Leden 2005, [online], [cit. 2008-11-05].  
URL [http://www.nxp.com/acrobat\\_download/datasheets/BC817\\_BC817W\\_BC337\\_5.pdf](http://www.nxp.com/acrobat_download/datasheets/BC817_BC817W_BC337_5.pdf)
- [18] Robot Power: *Scorpion HX User Manual*. Červenec 2004, [online], [cit. 2008-11-05].  
URL [http://www.robotpower.com/downloads/RobotPower\\_ScorpionHX\\_UserManual.pdf](http://www.robotpower.com/downloads/RobotPower_ScorpionHX_UserManual.pdf)
- [19] Rozman, J.: *Mobilní autonomní robot*. Projekt FRVŠ 2818.
- [20] Schwarz, J.; Růžička, R.; Strnadel, J.: *Mikroprocesorové a vestavěné systémy Studijní opora*. 2006.
- [21] Snail Instruments: *Robotika - motory a kola, pohonné jednotky*. [online], Poslední aktualizace 16. března 2008, [cit. 2008-11-05].  
URL <http://www.snailinstruments.com/cze/robotics/motors.php>
- [22] Snail Instruments: *ACC7260*. Srpen 2004, [online], [cit. 2008-11-05].  
URL <http://www.snailinstruments.com/docs/acc7260.pdf>
- [23] Texas Instruments: *MSP430F168*. [online], [cit. 2008-11-05].  
URL <http://focus.ti.com/docs/prod/folders/print/msp430f168.html>
- [24] Texas Instruments: *MSP430x15x, MSP430x16x, MSP430x161x Mixed Signal Microcontroller (Rev. E)*. Srpen 2006, [online], [cit. 2008-11-05].  
URL <http://www.ti.com/lit/gpn/msp430f168>
- [25] Texas Instruments: *MSP430x1xx Family Users's guide*. 2006, [online], [cit. 2008-11-05].  
URL <http://focus.ti.com/lit/ug/slau049f/slau049f.pdf>
- [26] Vašíček, Z.: FITkit : Firmware - Propojovací systém FITkitu. [online], [cit. 2008-11-05].  
URL <http://merlin.fit.vutbr.cz/FITkit/docs/firmware/spifitkit.html>

# Seznam příloh

1. CD, na kterém jsou umístěny zdrojové kódy a další dokumenty v adresářích nebo souborech:

- `robot_control.zip`
  - Zdrojové soubory pro umístění do adresářové struktury FITkitu (včetně programové dokumentace)
- `FITkit.rev271.zip`
  - Zabalená SVN FITkitu, se kterou pracují výše zmíněné zdrojové kódy
- `bp_xnovot88.pdf`
  - Tato bakalářská práce
- `manuals/`
  - Manuály k použitým zařízením
- `photos/`
  - Fotografie robota a zapojení
- `videos/`
  - Videá při dálkovém ovládní robota