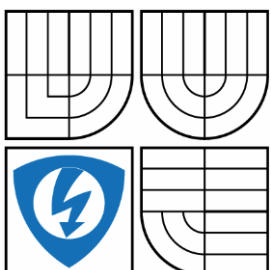


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## ELEKTRONICKÝ SUBSYSTÉM MALÉHO MOBILNÍHO ROBOTU

ELEKTRONIC SUBSYSTEM OF A SMALL MOBILE ROBOT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

LUDĚK PODOLAN

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. LUDĚK ŽALUD, Ph.D.

BRNO 2013



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Bakalářská práce

bakalářský studijní obor  
**Automatizační a měřicí technika**

**Student:** Luděk Podolan

**ID:** 125597

**Ročník:** 3

**Akademický rok:** 2012/2013

## NÁZEV TÉMATU:

**Elektronický subsystém malého mobilního robotu**

## POKYNY PRO VYPRACOVÁNÍ:

- Navrhněte elektroniku pro řízení čtyř hlavních motorů robotu.
- Vytvořte systém pro správu nabíjení akumulátorů (battery management).
- Navrhněte použití enkodérů pro předložený typ podvozku.
- Navrhněte a otestujte komunikační protokol mezi předloženým mikrokontrolérem a proximním skenerem.
- Navrhněte a otestujte rozhraní pro komunikaci s uživatelem pomocí LCD displeje.

## DOPORUČENÁ LITERATURA:

H.R. Everett, Sensors for Mobile Robots, A K Peters/CRC Press, 1995, ISBN 1568810482

**Termín zadání:** 11.2.2013

**Termín odevzdání:** 27.5.2013

**Vedoucí práce:** doc. Ing. Luděk Žalud, Ph.D.

**Konzultanti bakalářské práce:**

**doc. Ing. Václav Jirsík, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt**

Bakalářská práce se zabývá návrhem a stavbou mobilního robotu s kolovým diferenciálním podvozkem. Cílem práce bylo navrhnout elektronický subsystém robotu, umístit jej do robotu a poté otestovat. Každá kapitola se nejdříve zabývá teoretickým rozбором daného problému a až poté jeho praktickou realizací. V práci jsou uvedeny možnosti výběru prvků pro stavbu robotu s popisem vybrané varianty. Centrálním řídicím prvkem robotu je AVR mikrokontrolér vsazený do vývojového prostředí EvB4.3. Díky LCD displeji vzniklo jednoduché menu pro snadné ovládání robotu uživatelem. Nabití akumulátoru je v robotu programově hlídáno díky možnosti využití A/D převodníku mikrokontroléru. V posledních dvou kapitolách je zmíněno použití enkodérů s možností následného návrhu regulátoru otáček motorů. Je zde také popsána komunikace mezi proximitním skenerem zajišťující jízdu robotu středem koridoru. V závěru práce jsou pro názornost uvedeny fotky finální podoby robotu.

## **Klíčová slova**

Elektronický subsystém, mobilní robot, kolové platformy, akumulátor, vývojový kit EvB4.3, Atmega644p, řízení stejnosměrného motoru, H-můstek, správa nabíjení akumulátorů, stabilizace napětí, enkodér, ARDUINO, SPI sběrnice

## **Abstract**

The topic of this bachelor thesis is design and construction of a mobile robot with wheeled differential platform. The main goal was to design, implement and test an electronic sub-system for a small mobile robot. Each chapter firstly deals with theoretical aspects of relevant topics and after that describe the process of construction. The thesis shows various ways of construction of the robot with the process of selection of the optimal one. The main controlling part of the robot is the AVR microcontroller placed on the EvB4.3 development board. A simple menu was created on an LCD display for communication between the user and the robot. Battery management is controlled by an analogue to digital converter. The last two chapters focus on encoders and communication between the robot and a proximity sensor. Pictures of the robot are enclosed at the end of the thesis.

## **Keywords**

Electronic subsystem, mobile robot, wheeled platforms, battery, development board EvB4.3, Atmega644p, DC motor controlling, H-bridge, battery management, voltage regulation, encoder, ARDUINO, SPI bus.

### **Bibliografická citace:**

PODOLAN, L. *Elektronický subsystém malého mobilního robotu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2013. 71s. Vedoucí bakalářské práce byl doc. Ing. Luděk Žalud, Ph.D.

## **Prohlášení**

„Prohlašuji, že svou bakalářskou práci na téma Elektronický subsystém malého mobilního robotu jsem vypracoval samostatně pod vedením vedoucího bakalářské práce, s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: **27. května 2013**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu bakalářské práce doc. Ing. Lud'ku Žaludovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Dále děkuji kolegovi Vítu Přecechtělovi za odbornou pomoc a věcné rady při stavbě a programování robotu. Také děkuji své rodině za veškerou pomoc při studiu a zpracování této závěrečné práce.

V Brně dne: **27. května 2013**

.....  
podpis autora

# Obsah

<b>1. Úvod.....</b>	<b>9</b>
1.1 Téma práce .....	9
1.2 Co je to robot?[1] .....	9
1.3 Aplikace mobilního robotu[1] .....	10
1.4 Typy mobilních robotů .....	10
<b>2. Prvky pro stavbu mobilního robotu .....</b>	<b>11</b>
2.1 Kolové platformy mobilních robotů .....	11
2.1.1 Diferenciální podvozek (differential drive) .....	11
2.1.2 Synchronní podvozek (synchro drive) .....	12
2.1.3 Trojkolový podvozek s řízeným předním kolem (tricycle drive) .....	13
2.1.4 Ackermannův podvozek (Ackermann steering nebo kingpin steering) .....	13
2.1.5 Podvozky se všesměrovými koly .....	13
2.1.6 Pásové podvozky .....	14
2.1.7 Vybraný podvozek .....	15
2.2 Výběr a popis řídicí elektroniky robotu .....	15
2.3 Akumulátory .....	17
2.3.1 Pb - Olovený akumulátor .....	18
2.3.2 NiCd – Nikl-kadmiový akumulátor .....	18
2.3.3 NiMh - Nikl metal hydridový akumulátor .....	19
2.3.4 Li-ion a li-pol akumulátory .....	19
2.3.5 Vybraný typ akumulátoru .....	20
<b>3. Elektronika pro řízení čtyř hlavních motorů robotu.....</b>	<b>22</b>
3.1 Základní vlastnosti stejnosměrného motoru[14] .....	22
3.1.1 Princip činnosti[14] .....	22
3.2 Popis motorů ve vybraném podvozku .....	25
3.3 Budicí obvody stejnosměrných obvodů[14] .....	26
3.4 Dvojitý H-můstek pro řízení stejnosměrných motorů .....	27
3.4.1 DPS pro paralelní zapojení motorů .....	29
3.5 Ukázka zdrojového kódu z mikrokontroléru .....	30
<b>4. Komunikace s uživatelem pomocí LCD displeje .....</b>	<b>34</b>
4.1 Popis LCD displeje posazeného na kitu .....	34
4.2 Schéma zapojení LCD k AVR mikrokontroléru .....	35
4.3 Popis stáhnuté knihovny pro obsluhu LCD .....	36
4.3.1 Definování portu ovládajícího LCD .....	36
4.3.2 Základní funkce knihovny .....	36
4.4 Popis a ukázka vytvořeného menu .....	38
4.4.1 Zdrojový kód .....	38
4.4.2 Ukázka obrazovek menu .....	39
<b>5. Správa nabíjení akumulátorů (battery management) a stabilizace napětí .</b>	<b>40</b>
5.1 Nabíječ Li-Pol akumulátorů .....	40

5.2	Indikace nabití baterie pomocí LED diod .....	41
5.3	Hlídaní napětí akumulátoru pomocí mikrokontroléru.....	42
5.3.1	Dělič napětí .....	42
5.3.2	Program mikrokontroléru.....	44
5.4	Stabilizační deska pro 9V .....	46
<b>6.</b>	<b>Návrh použití enkodérů.....</b>	<b>47</b>
6.1	Princip a typy enkodérů[30] .....	47
6.2	Popis vybraného čipu .....	48
6.3	Popis navržené DPS pro umístění čipu .....	48
6.4	Možnosti zapojení uvedeného čipu .....	50
6.5	Popis mikrokontroléru na platformě ARDUINO .....	50
6.6	Programová část mikrokontroléru .....	51
6.6.1	Popis SPI sběrnice pro přenos mezi AVR a ARDUINEM .....	52
<b>7.</b>	<b>Komunikace s mikrokontrolérem obsluhující proximitní skener .....</b>	<b>55</b>
7.1	Popis vstupně výstupních pinů mikrokontroléru.....	55
7.2	Popis smluvené komunikace .....	55
7.3	Ukázka zdrojového kódu mikrokontroléru .....	56
<b>8.</b>	<b>Finální podoba robotu .....</b>	<b>57</b>
8.1	Fotky robotu .....	57
8.2	Celkové blokové schéma robotu .....	59
<b>9.</b>	<b>ZÁVĚR.....</b>	<b>60</b>
	<b>Literatura.....</b>	<b>61</b>
	<b>Seznam obrázků .....</b>	<b>65</b>
	<b>Seznam symbolů, jednotek a zkratk .....</b>	<b>67</b>
	<b>Seznam příloh .....</b>	<b>68</b>

# 1. ÚVOD

## 1.1 Téma práce

Cílem této práce je navrhnout a sestavit mobilní robot schopný autonomního či řízeného pohybu s možností pohybu ve vnitřním i venkovním prostředí. Pro zajištění bezproblémové činnosti robotu je důležité vybrat a následně správně zapojit vhodné prvky, jako například podvozek, akumulátor, řídicí a ochrannou elektroniku.

Téma práce bylo rozděleno do dvou celků. V mé práci se zabývám stavbou robotu, výběrem komponent, jejich kompletací a návrhem vhodného algoritmu pro obsluhu jednotlivých částí robotu mikrokontrolérem Atmega644p umístěným na vývojovém prostředku EvB4.3. Jedná se o vývojovou desku obsahující periferie potřebné k jednoduché obsluze robotu, jako tlačítka, LED diody pro signalizaci, bzučák, segmentový displej a také LCD displej. Díky možnosti zobrazení nápisu na LCD displeji bylo vytvořeno jednoduché menu, ve kterém si uživatel může vybrat, jak se robot bude pohybovat. Nutností při užívání modelářských akumulátorů je jejich ochrana proti vybití pod určitou mez. Nabíjení akumulátoru je hlídáno dvěma způsoby. Indikační pás LED diod zobrazuje postupným rozsvěčováním stav baterie. Druhým způsobem je přesnější ochrana pomocí analogově-digitálního převodníku mikrokontroléru, díky kterému je možno přesněji kontrolovat napětí akumulátoru, zobrazit jej na displeji a ve spolupráci se bzučákem vydávat akustický signál při vybití. V práci je také nastíněn princip a zapojení enkodérů zajišťujících možnost regulace otáček motorů. Pro ukázkou programování zde budou uvedeny pouze části zdrojových kódů. Kompletní zdrojové kódy jsou uvedeny na příloženém DVD.

Druhou částí tématu se zabývá ve své práci týmový kolega Vít Přecechtěl, jehož úkolem je zpracování dat z proximitního skeneru a návrh řídicího algoritmu pro průjezd koridorem. Data ze skeneru jsou čtena pomocí dalšího vývojového prostředku osazeného mikrokontrolérem ARM. Jelikož je tento mikrokontrolér výkonnější, veškeré složitější výpočty probíhají v něm. V práci bude uveden princip komunikace mezi oběma mikrokontroléry. Robot obohacený o tento skener bude ve spolupráci s enkodéry, již umístěnými v robotu, schopen autonomního pohybu středem koridoru a vyhýbání překážkám.

## 1.2 Co je to robot? [1]

Robot je obvykle elektromechanický stroj, který je schopen se pohybovat v prostoru, vnímat svoje okolí, případně toto okolí i měnit. Mobilní robot může mít rozdílnou míru autonomie, od teleoperovaného až po úplně autonomního.

Teleoperovaný mobilní robot pouze vykonává příkazy lidského operátora. Robot může být ovládán úplně (podobně jako autíčko na vysílačku) nebo mohou být některé

činnosti částečně autonomní. Takový robot bývá označován za semiautonomní. Plně autonomní robot je schopen činnosti zcela bez lidského zásahu, případně je činnost lidského operátora omezena na zadávání cílů robotu, který sám volí nejlepší způsob dosažení zadaného cíle.

### 1.3 Aplikace mobilního robotu [1]

Nejvíce aplikací mobilních robotů nacházíme v průmyslovém, výzkumném a armádním prostředí. Všechna tato prostředí mají společnou vlastnost, že se v něm nepohybují osoby nepoučené o možnostech daného robotu. To snižuje nároky na použitého robota, který není nucen se vyrovnat s nepředvídatelným chováním lidí.

V průmyslu se mobilní roboty nasazují ve výrobních halách či skladech k dopravě materiálu. Častým důvodem nasazení mobilních robotů je prostředí nebezpečné nebo nevhodné pro člověka, jako jsou provozy s vysokou hladinou hluku, vysokou teplotou, nebezpečnými plyny apod. Automobilový průmysl využívá značné množství robotických přístupů při vývoji různých asistenčních zařízení usnadňující řízení, GPS navigace, udržování odstupu od předchozího automobilu, sledování krajnice. Armádní prostředí také nahrává mobilním robotům. Bezpilotní letouny schopné plnit úkoly v autonomním nebo teleoperovaném režimu. Mobilní roboty se používají k odstranění námořních i pozemních min, k průzkumu nebezpečných oblastí. Automatické řízení aut v zásobovací koloně přispívá k bezpečnosti vojáků. Stačí řídit první vůz kolony, a ostatní následují autonomně.

### 1.4 Typy mobilních robotů

V následující podkapitole jsou uvedeny informace ze skript předmětu Základy robotiky, dostupné z [2].

Mobilní roboty je možné rozdělit podle toho, v jakém prostředí se pohybují na roboty operující:

- Na souši (terrestrial)
- Ve vodě (aquatic)
- Ve vzduchu (airborne)
- Ve vesmíru (space)

Jsou pochopitelně možné i kombinace – např. tzv. obojživelné stroje, tj. vozidla schopná pohybu na souši i ve vodě. Roboty pohybující se po souši je dále možno rozdělit na platformy kolové (wheeled), pásové (tracked) a kráčeující (limbed).

Jelikož je v mé práci použit kolový podvozek, který obsahuje několik vlastností pásového podvozku, v následujícím uvedu popis a vlastnosti těchto dvou typů podvozků.

## 2. PRVKY PRO STAVBU MOBILNÍHO ROBOTU

V následujících kapitolách o mobilních platformách jsou uvedeny informace ze skript předmětu Základy robotiky, dostupné z [2].

### 2.1 Kolové platformy mobilních robotů

Než si rozebereme jednotlivé typy podvozků, měli bychom si říct něco o typech kol. Kola mohou být aktivní či pasivní, tedy hnaná nebo vlečená (tlačená). Dále mohou mít určitý počet stupňů volnosti. Kolo s jedním stupněm volnosti se pohybuje pouze kolem jedné osy rovnoběžné s povrchem, po kterém se robot pohybuje (například zadní kolo automobilu s náhonem předních kol). Dále mohou mít kola dva stupně volnosti. V tomto případě se kolo pohybuje ve směru dvou os. Tento příklad si můžeme představit jako přední kolo bicyklu. Otáčí se kolem osy rovnoběžné s povrchem a zároveň osy propojené s říditky při zatáčení. V mobilní robotice mohou být manévry těchto pohybů zajištěny motory či servomechanismy.

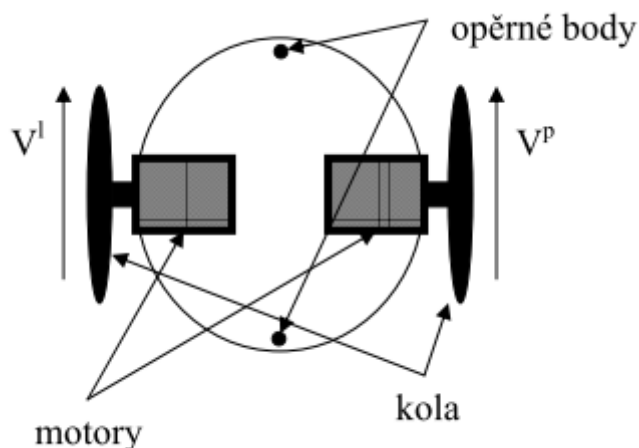
#### 2.1.1 Diferenciální podvozek (differential drive)

Jde o nejjednodušší a také nejpoužívanější druh podvozku, který se velmi často využívá pro aplikace mobilních robotů ve vnitřních prostorách budov. Nelze ale samozřejmě vyloučit použití tohoto typu podvozku ve venkovních destinacích. Jedná se také o nejoblíbenější řešení pro amatéry a nadšence v oblasti robotiky.

Kinematiku tohoto typu podvozku lze popsat na obrázku [Obrázek 2-1]. Podvozek obsahuje dvě aktivně řízená kola s jedním stupněm volnosti. K pohonu těchto kol se nejčastěji využívají stejnosměrné motory s enkodéry nebo krokové motory. K tomuto typu podvozku je také vhodné zakomponovat všesměrová kolečka, sloužící pro zajištění stability celého podvozku.

Mezi podstatné výhody tohoto typu podvozku patří možnost využití docela přesné odometrie (schopnost lokalizace robotu z vlastních systémů, např. enkodérů zabudovaných v motorech) pro navigaci robotu. Další výhodou tohoto podvozku je snadná manévrovanost. Robot s tímto typem podvozku je schopen otočit se na místě.

Uvedu také jednu z nevýhod tohoto podvozku a tou je neschopnost překonat vyšší překážky. Při pohybu například přes vyšší hrbol se může převrátit a tím pádem může dojít k poškození elektroniky nebo kapotáže.

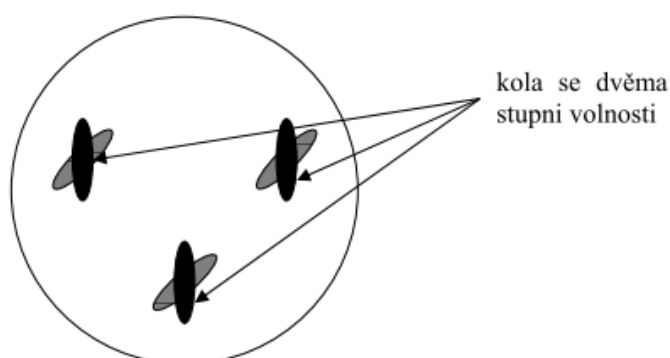


Obrázek 2-1 - Nákres diferenciálního podvozku [2]

### 2.1.2 Synchronní podvozek (synchro drive)

Typickým uspořádáním kol tohoto typu podvozku je rovnostranný trojúhelník. Každé z kol má dva stupně volnosti, čímž je ve většině případů nutno použít dvou motorů, či motoru a servomechanismu pro každé kolo. Cílem tohoto uspořádání je synchronizovaný pohyb, kdy se kola otáčejí stejnou rychlostí a na stejnou stranu. Této spolupráce lze dosáhnout vhodným návrhem řídicího systému nebo vhodným uspořádáním kol na podvozku. Oproti mechanické synchronizaci, kdy je třeba pouze dvou motorů na libovolný počet kol, při elektronické synchronizaci je třeba dvou motorů na jedno kolo.

Velkou výhodou podvozku s mechanickým provázáním kol je jeho jednoduchost v oblasti řízení. Není zde nutno provádět žádné matematické operace či transformace pohybu k řízení jednotlivých kol. Zjednodušeně řečeno. Stačí zadat úhel natočení jednoho kola a např. přes řetěz se nám otočí i ostatní kola o stejný úhel.



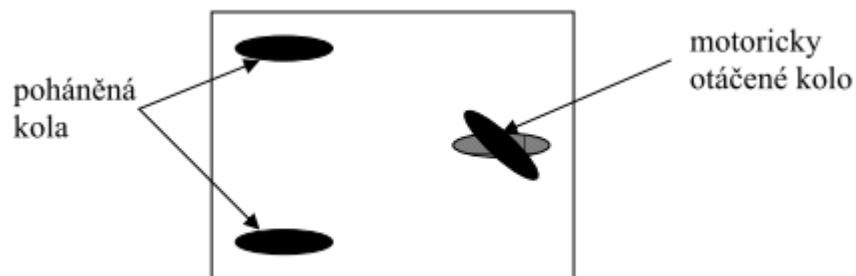
Obrázek 2-2 - Nákres synchronního podvozku [2]

### 2.1.3 Trojkolový podvozek s řízeným předním kolem (tricycle drive)

Uspořádání kol tohoto typu podvozku je většinou také ve tvaru trojúhelníku. Dvě hnaná kola umístěná vzadu plus jedno motoricky natáčené kolo vepředu. Tento podvozek si lze představit například jako dětský kočárek stylu trojkolky, jenom s koly obohacenými o motory.

Výhodou podvozku je jednoduché řízení, kdy jedním motorem ovládáme rychlost zadních kol, tedy rychlost celého podvozku, a druhým motorem natočení předního kola, tedy směr jízdy.

Nevýhodou je zde nemožnost rotace na místě, tedy i velká pravděpodobnost uvíznutí v komplikovaných prostorech.

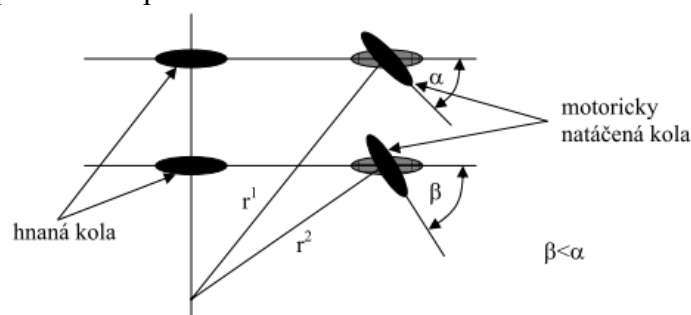


Obrázek 2-3 - Schéma trojkolového podvozku [2]

### 2.1.4 Ackermannův podvozek (Ackermann steering nebo kingpin steering)

Jde o typ podvozku známý z automobilů. Pohyb zajišťují dvě zadní poháněná kola s natáčením předních kol. Natočení předních kol nebývá ve většině případů synchronní např. z důvodu menšího natočení jednoho z kol při průjezdu zatáčkou.

Tohoto typu podvozku se nejvíce využívá u větších robotů operujících v terénu. Nevýhodou je opět otočení podvozku na místě.



Obrázek 2-4 - Nákres Ackermannova podvozku [2]

### 2.1.5 Podvozky se všesměrovými koly

Alternativou k výše uvedeným typům kol v podvozcích jsou tzv. všesměrová či složená kola. Jedná se systém složených kol, která jsou schopná nést podvozek

ve dvou osách. Tímto způsobem je možné zkonstruovat všesměrově se pohybující mobilní platformu.

Jako příklad mohou posloužit kola na obr. [Obrázek 2-5], která mají na svém obvodu množství pasivních válečků zajišťujících již zmiňovaný pohyb. Podvozek by tedy mohl vykonávat pohyb libovolným směrem, s libovolnou rotací a navíc by tyto dva pohyby mohl vykonávat současně. Kdyby byl pohyb „přidaných“ kol zablokován, chovalo by se kolo jako běžné kolo s jedním stupněm volnosti.

Hlavní nevýhodou tohoto podvozku je naprosto nemožné použít odometrii k navigaci podvozku, protože celý systém podvozku je založen na prokluzu kol, který nelze přesně spočítat. Prokluz závisí také na povrchu válečků a vlastnostech povrchu, po kterém se robot aktuálně pohybuje. Možností navigace by mohl být například GPS systém, který nevyužívá vnitřních systémů robotu.

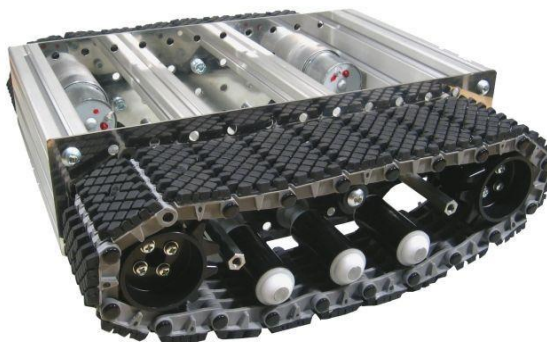


Obrázek 2-5 - Ukázka všesměrového kola Mecanum [3]

### 2.1.6 Pásové podvozky

Vlastnosti těchto podvozků se téměř rovnají vlastnostem diferencially řízeného podvozku s odlišností propojení dvou diferencially kol pásem. Pás zajišťuje podvozku lepší přilnavost s projížděným povrchem. Pásové podvozky umožňují robotům projet podstatně náročnější terény než ostatní podvozky.

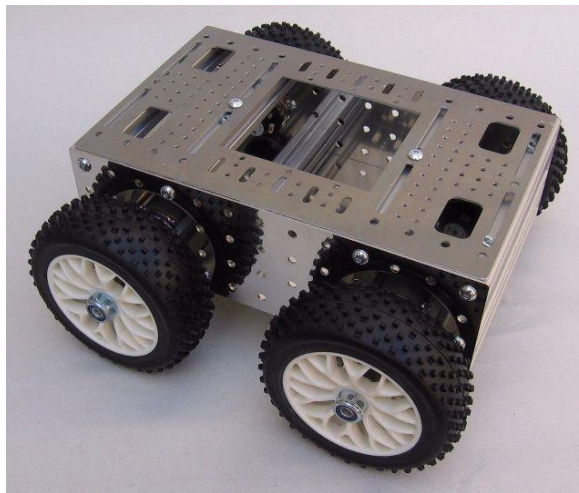
Nevýhoda těchto podvozků je v zatáčení. Natočení celého podvozku znamená puštění naplnu všech kol. Jedna strana vzad a jedna strana vpřed. Tento pohyb způsobuje u robotů vynaložení největší energie, tím pádem i brzké vybití napájecích akumulátorů při častém zatáčení. Při pohybu vpřed či vzad není spotřeba tak velká. Další důležitostí je dimenzace součástek na vyšší proudy z důvodu jejich odběru při tomto náročnějším pohybu.



Obrázek 2-6 - Příklad pásového podvozku [4]

## 2.1.7 Vybraný podvozek

Podvozek je možno buď vyrobit podomácku z nakoupených součástek anebo si jej koupit v kompletním sestavení jako jsem to udělal já. U menších modelů je do podvozku také zakomponována převodovka pro lepší převodový poměr kol a jeho lepší manévrovatelnost. Podvozky mohou být velikosti modelu pásového podvozku pro tank o rozměrech formátu stránky A5, či větší o půdorysu samotné konstrukce bez kol o velikosti A4. Svůj podvozek jsem také vybíral pro jeho robustnost a možnost uvezení až 4kg nákladu. Podvozek s koly i motory byl objednan z internetového obchodu firmy Snail Instruments.



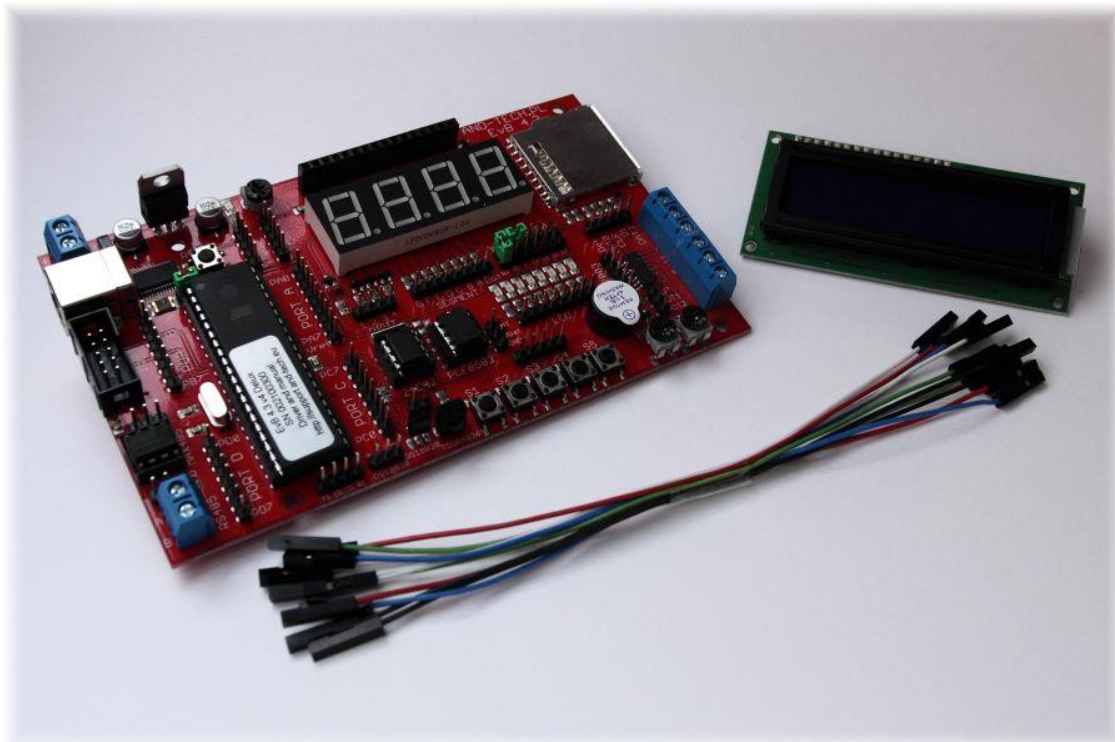
Obrázek 2-7 - Zakoupený diferenciální podvozek [36]

Jedná se o čtyřkolový podvozek s pohonem všech kol. Každé kolo je poháněno samostatným motorem. Pro mou aplikaci jsem použil zapojení, kdy je každá strana podvozku ovládána synchronně. Z popisu je vidno, že se jedná o diferenciálně řízený typ podvozku uvedený v kapitole 2.1.1 jenom s tím rozdílem, že na každé straně jsou dva motory. Podvozek takto získá větší stabilitu, není tedy potřeba použít podpůrných koleček. Podrobnější vlastnosti motorů budou uvedeny v jiné kapitole.

## 2.2 Výběr a popis řídicí elektroniky robotu

V dnešní době je možné na současném trhu vybrat ze široké škály řídicích komponent. Je možné sáhnout po mikroprocesorové desce Arduino, obsahující AVR procesor, či desce osazené procesorem typu ARM nebo zvolit ještě výkonnější miniaturní počítač Raspberry Pi velikosti kreditní karty. V této části bych nechtěl rozebírat podrobně všechny možné varianty, ale podat obecný náhled na mnou vybranou variantu.

Z tohoto nepřehledného množství možných řídicích mikrokontrolérů (mikroprocesor obohacený o periferie, vstupy a výstupy, A/D převodník a další potřebné součásti) jsem si pro svou práci vybral kit EvB4.3, který je možné zakoupit v internetovém obchodě [www.shop.onpa.cz](http://www.shop.onpa.cz). Na uvedené internetové adrese je uživateli přístupna bohatá podpora, což je velké plus. Vlastnosti dílčích částí kitu budou blíže popsány v kapitolách, ve kterých jsou jednotlivé periferie využívány.



**Obrázek 2-8 - Vývojový kit EvB4.3 s drátovými propojkami [5]**

Na desce vývojového kitu EvB4.3 je možno najít následující části [6]:

- procesor AVR ATmega16 (ATmega32 nebo ATmega644p),
- hodiny reálného času - RTC (Real Time Clock) - PCF8583,
- sériová paměť EEPROM v provedení I2C - AT24C02,
- přijímač IR (infračerveného záření) na 36kHz - TSOP4836,
- teplotní senzor – 1wire Dallas sběrnice - DS18B20,
- převodník RS232/RS485,
- patice pro MMC/SD paměťové karty,
- tlačítko – 5x,
- LED dioda - 8x,
- separátní výkonový výstup 500mA – 5x,
- analogové potenciometry – 2x,
- sedmi segmentový LED display – 4x,
- interface ISP (programování),
- interface USB,
- volné napájecí piny +5VDC – 5x,
- volné zemnicí piny – 5x,
- LCD display 2x16 znaků – 1x.

Ve standardním balení je ke kitu dodáváno 10ks drátových propojek.

Nyní budou uvedeny informace k programování procesoru vsazeného do kitu. V datasheetu kitu, který je možno stáhnout na již uvedené stránce [www.shop.onpa.cz](http://www.shop.onpa.cz) najdeme na straně 23 návod, jak nahrát do procesoru program. Drátovými propojkami propojíme podle obrázku piny ISP konektoru a programovacího konektoru procesoru.

Poté už nám jen stačí stáhnout z uvedených stránek program AND-LOAD, správně jej nastavit podle manuálu a nahrát námi požadovaný soubor.

Celý svůj projekt jsem programoval programovacím jazykem C v programu AVR Studio. Nejnovější verze je dostupná na serveru [www.atmel.com](http://www.atmel.com). Zde je nutné se zaregistrovat. Po dokončení registrace Vám bude zaslán e-mail s odkazem na stažení programu. Při překompilování vytvořeného projektu v AVR Studiu dojde ve složce s projektem k vytvoření souboru s příponou hex, který poté nahrajeme do procesoru pomocí již zmíněného programu AND-LOAD.

V této práci budou uvedeny části kódu obsahující funkce ke správnému chodu jednotlivých částí robotu bez ukázky nahrávání. Kompletní zdrojové kódy jsou uvedeny na příloženém DVD.

Ještě přidávám jednu vlastní zkušenost a tou je, že pokud nebylo programování nikdy Vaším cílem a nemáte vůbec tušení o nějakých funkcích, registrech a podobných tématech v oblasti mikrokontrolérů, je vhodné si na začátku přečíst některou z knížek věnujících se tomuto tématu. Jedním z výborných autorů je pan Váňa s knihou Mikrokontroléry ATMEL AVR – Programování v jazyce C, z nakladatelství BEN. Uvádí zde názorné a praktické příklady práce s mikrokontroléry, kdy čtenář dostane jakýsi všeobecný náhled na věc.

Na úvod neuškodí znát také několik základních vlastností mikrokontroléru. Z katalogového listu mikrokontroléru bychom vyčetli, že mikrokontrolér Atmega664p obsažený na kitu obsahuje 4 porty, pojmenované písmeny A až D, každý po 7 pinech využitelných pro další aplikace. Každý z pinů obsahuje další specifické funkce, jako například funkce A/D převodníku, čítače, časovače, funkce sériového rozhraní, SPI rozhraní, a další pro nás nedůležité funkce.

## 2.3 Akumulátory

Akumulátorem rozumíme technologický vynález, který dokáže akumulovat energii, většinou se jedná o energii elektrickou. Jedná se o sekundární článek, což znamená, že před použitím akumulátoru je nutné jej prvně nabít. Akumulátor se skládá z několika článků o určitém jmenovitém napětí. U každého typu baterie jsou uvedeny hraniční hodnoty napětí pro nabíjení, vybíjení a skladování akumulátorů. Jeden z hlavních parametrů, na který je vhodné se při koupi dívat, je kapacita uvedena v Ah (Ampér hodinách), případně mAh (mili-Ampér-hodinách). Čím větší kapacita, tím větší výdrž napájeného zařízení a pochopitelně i větší pořizovací cena.

Akumulátory se dají rozdělit podle různých kritérií a můžeme jich najít velké množství. V této kapitole bude uvedeno rozdělení podle jejich chemického složení. Rozdělení akumulátorů a jejich stručný popis byl vybrán z [7].

Nejběžněji používané akumulátory jsou:

1. Olověné akumulátory – **Pb**
2. Nikl-kadmiové – **NiCd**
3. Nikl-metal hydridové – **NiMh**
4. Lithium-iontové - **Li-ion**, lithium-polymerové - **Li-pol**

### 2.3.1 Pb - Olověný akumulátor

Mezi hlavní výhody tohoto akumulátoru patří jeho vysoká kapacita, dlouhá životnost a nízké samovybíjení.

Vzhledem ke své hmotnosti, která je docela velká pro menší modely, se hodí do větších modelů jako například vzducholodí, lodí, či větších automobilů. Nevýhodou těchto akumulátorů je také jejich velikost, schopnost nedodávat větší proudy a ekologická nešetrnost (olovo).

Tento typ akumulátorů můžeme najít například v osobních automobilech. Napětí jednoho článku je 2V.



Obrázek 2-9 - Pb akumulátor, 12V, 18Ah [8]

### 2.3.2 NiCd – Nikl-kadmiový akumulátor

Při procházení jakéhokoliv internetového obchodu s elektronikou a akumulátory lze najít následující vlastnosti nikl-kadmiových akumulátorů.

Výhodou je schopnost dodání vysokých proudů. Velkou výhodou je počet nabíjecích cyklů (cca 1000), provozní teplota (snese proudy až do -40°C). Další výhodou je nízká pořizovací cena a rychlé nabíjení.

Nevýhodou těchto akumulátorů je vyšší hmotnost. Při procházení internetu jsem našel akumulátory s velkou proudovou kapacitou, ale malou napěťovou kapacitou. Např. 4500mAh při napětí 1,2V. Na druhou stranu není třeba často akumulátor dobíjet při dlouhodobějším provozu zařízení. Díky výrobě z kadmia jsou tyto akumulátory taktéž ekologicky nešetrné jako předchozí olověné akumulátory.

Tento typ baterií můžeme nalézt například v podobě nabíjecích baterií do příručních fotoaparátů, svítilen a dalších zařízení pracujících při nízkém napětí, které je na jednom článku 1,2V.



Obrázek 2-10 - NiCd akumulátor, 1,2V, 4500mAh [9]

### 2.3.3 NiMh - Nikl metal hydridový akumulátor

Při stejných rozměrech jako NiCd dosahují vyšší kapacity a mají o něco nižší hmotnost. Zároveň se nehodí do zařízení s vyšším aktuálním proudovým zatížením. Mezi další výhody patří rychlé nabíjení a větší ekologická šetrnost než NiCd.



Obrázek 2-11 - NiMh akumulátor, 1,2V, 800mAh [10]

Nevýhodou těchto akumulátorů je rychlé samovybití, nemožnost dodávání vyšších proudů a nižší počet nabíjecích cyklů.

Možnost využití je stejná jako u NiCd, tedy fotoaparáty, svítilny, vysílačky, autíčka na dálkové ovládání, atd. Napětí na jednom článku je 1,2V.

### 2.3.4 Li-ion a li-pol akumulátory

Poslední dva typy akumulátorů budou díky svým společným vlastnostem uvedeny v jedné podkapitole.

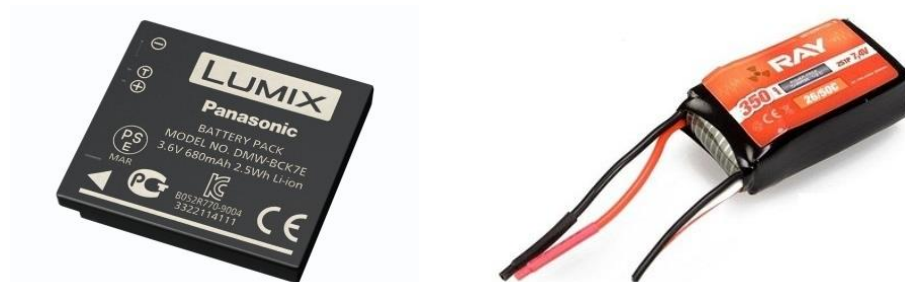
Mezi výhody obou akumulátorů patří jejich vysoká kapacita, nízká hmotnost, velmi malé samovybití a vysoký počet nabíjecích cyklů.

Nevýhodou je v porovnání s ostatními typy vyšší pořizovací cena a díky nutnosti nabíjení malými proudy dlouhá doba nabíjení. Lithium-iontové akumulátory mají ještě jednu nevýhodu a tou je neschopnost dodávání vyšších proudů.

Možností využití obou typů je velká spousta. Li-ion akumulátory jsou používány například ve fotoaparátech nebo mobilních telefonech, jejichž hlavním zdrojem energie nejsou „obyčejné tužkové baterie“, tedy NiMh baterie, ale li-ion akumulátory

Li-pol akumulátory se zase díky svým malým rozměrům a velké proudové i napěťové kapacitě využívají v modelářství. Spousta modelů, jako například roboty,

letadla na dálkové ovládání, modely aut a další, jsou napájeny tímto druhem akumulátoru.



Obrázek 2-12 - Li-ion akumulátor, 3,6V, 680mAh [11] (vlevo) a li-pol akumulátor, 7,4V, 350mAh [12] (vpravo)

### 2.3.5 Vybraný typ akumulátoru

V předchozích podkapitolách byly uvedeny nejběžněji používané akumulátory v modelářství a elektronických přístrojích.

V počátcích mé práce jsem nejprve chtěl využít pro stavbu robotu dvou olověných paralelně zapojených akumulátorů. Díky paralelnímu zapojení získáme při stejném napětí dvojnásobnou proudovou kapacitu. Při výběru podvozku a podrobnějším prostudování vlastností jednotlivých typů jsem se rozhodl jednak díky hmotnosti, rozměrům, proudové i napěťové kapacitě pro li-pol modelářský čtyř článkový akumulátor zakoupený v modelářském obchodě [www.rcmarket.cz](http://www.rcmarket.cz). Jeho pro mě důležité parametry jsou:

- jmenovitá kapacita 2700mAh,
- jmenovité napětí 16,8V,
- 4 sériově zapojené články, každý o jmenovitém napětí 4,2V,
- hmotnost 246g.



Obrázek 2-13 - Vybraný Li-Pol akumulátor, 16,8V, 2700mAh [13]

Výpočet proudového zatížení robotu:

- Špičkový odběr všech motorů je přibližně 3500mA
- Zbývající části elektroniky mají celkový odběr přibližně 500mA

Celkový odběr robotu je tedy 4000mA.

Z následujícího výpočtu můžeme uvažovat dobu pohybu robotu při vybraném typu akumulátoru:

$$Doba\ pohybu = \frac{Kapacita\ baterie}{\check{S}pičkové\ proudové\ zatížení} * 60\ minut = \frac{2700}{4000} * 60 = 40\ minut$$

V praxi nebude robot pod špičkovou zátěží, ale bude střídat různé pohyby a vydrží tedy mnohem déle. Experimentálně bylo zjištěno, že největšího odběru dosahuje robot při zatačení, jelikož robot otáčí pravou a levou stranou do protisměru, motory jsou tedy v největším zátěži.

Důležité upozornění:

**Baterie nesmí být vybity pod svou uvedenou mez. V dalších kapitolách bude uvedeno ošetření k ohlídání této meze. Při dosažení minimálního napětí je nutno robot vypnout a akumulátory dobít. Při nadměrném vybití může dojít k explozi akumulátoru!!!**

Další důležitou vlastností, na kterou jsem bral ohled při mém výběru, je jmenovité napětí. Motory jsou napájené napětím o velikosti 12V a řízené H-můstkem (popsán dále), na kterém vzniká úbytek napětí 2V. V návrhu bylo důležité počítat s napěťovou rezervou, aby nedocházelo k výpadkům napájení motorů po chvíli jízdy robotu.

## 3. ELEKTRONIKA PRO ŘÍZENÍ ČTYŘ HLAVNÍCH MOTORŮ ROBOTU

### 3.1 Základní vlastnosti stejnosměrného motoru [14]

Stejnoseměrný motor s permanentním magnetem je často používaným typem motoru v oblasti pohonu mobilních robotů. Mezi jeho výhody patří zejména velice příznivý poměr výkon/hmotnost, relativně snadné řízení otáček, u běžných typů pak také cena, dostupnost a široká nabídka. Mezi nevýhody patří složitější a tedy dražší rychlostní a zejména polohové řízení v porovnání například s krokovým motorem, dále díky komutátoru je pak tento motor zdrojem elektromagnetického rušení a také není bezúdržbový (např. v porovnání s krokovým motorem).

Stejnoseměrný motor obvykle pracuje na relativně vysokých otáčkách a nízkém momentu, což je pro potřebu mobilních robotů nevýhodné. Řešením je použít převodovku. Tato převodovka může být již součástí daného motoru. Spolu s převodovkou se může motor dodávat také již se senzorem polohy nebo rychlosti. Výhodou je celková kompaktnost takového pohonu.

Většina stejnosměrných motorků má dva elektrické kontakty. Po připojení jmenovitého napětí (s ohledem na katalogové údaje motoru) na tyto svorky se rotor nezatíženého motoru roztočí a jeho otáčky se ustálí na jmenovitých. Změnou polaritý docílíme změnu smyslu otáček, změnou napětí pak změnu rychlosti otáčení.

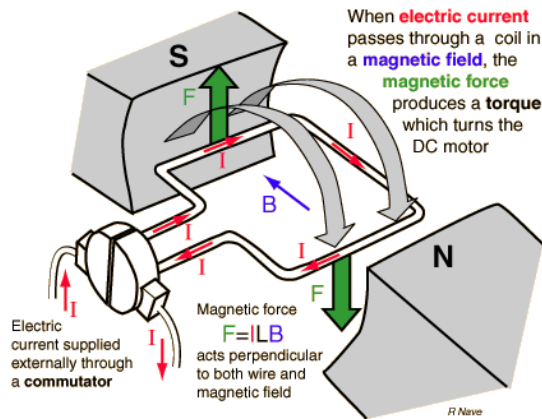
Příklad malého stejnosměrného motorku můžeme vidět na [Obrázek 3-1].



Obrázek 3-1 - Příklad stejnosměrného motorku s převodovkou [14]

#### 3.1.1 Princip činnosti[14]

Otáčení rotoru stejnosměrného motoru a permanentním magnetem – viz principiální schéma na [Obrázek 3-2], je způsobeno silou  $F = ILB$  vytvářející silový moment  $M$  na ramenech rovnajícím se poloměru závitů  $W$  smyčky (pro úhel  $\alpha = 0^\circ$ ). Tyto síly jsou dvě – navzájem opačné. Obecně pak moment  $M$  se rovná dvakrát síla  $F$  působící na rameni  $(W/2) \cdot \sin\theta$ , kde úhel  $\theta = 90 - \alpha$  – viz. [Obrázek 3-2].



Obrázek 3-2 - Řez stejnosměrným motorem, princip činnosti [15]

$$M = 2ILB \frac{W}{2} \sin\theta = ILBW \sin\theta \quad (3.1)$$

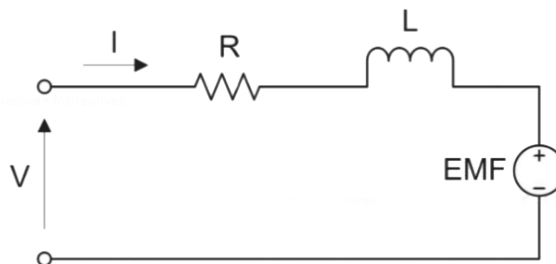
$$M = IBAsin\theta, \text{ kde } A = LW \text{ je plocha závitů smyčky}$$

Ve skutečnosti je jedna smyčka nahrazena cívkou složenou z N smyček – závitů. Na každý závit působí moment M, takže celkový moment působící na cívku je:

$$M = NIBAsin\theta \quad (3.2)$$

Kontinuální otáčení rotoru vyžaduje změnu polarity protékajícího proudu, což je zajištěno mechanicky komutátorem.

Pro vhodný výběr motoru je nutno znát jeho základní parametry a charakteristiky odvoditelné ze zjednodušeného modelu na následující straně [Obrázek 3-3].



Obrázek 3-3 - Náhradní schéma stejnosměrného motoru [16]

Poměr výstupního mechanického výkonu k elektrickému vstupnímu výkonu je roven účinnosti. V důsledku typických ztrát jako tření, odpor vinutí a dalších je tento poměr menší než 1. Typicky pro malé motorky je v rozsahu 0,5 – 0,75, pro velké 0,9.

Při otáčení rotoru, a tedy i cívky, v magnetickém poli permanentního magnetu statoru, je generován elektrický proud. Jeho změna v čase indukuje elektromotorické napětí  $u_{emf}$  opačné polarity než je napájecí napětí, závislé na parametrech motoru tj. na mg. indukci B v mezeře pro vinutí, na počtu a délce závitů rotoru a na průměru, na

němž je umístěno vinutí. Tyto parametry motoru se v katalogu motorů vyjadřují rychlostní konstantou  $k_n$ , která je definována vztahem pro rychlost  $n$  v ot./min,

$$n = k_n \cdot u_{emf} \quad (3.3)$$

Čím se tedy rotor rychleji otáčí, tím je toto napětí vyšší. Protože napětí  $u_{emf}$  je, jak již bylo zmíněno, v opačné polaritě vůči napájecímu napětí  $U$ , dochází k poklesu proudu tekoucím vinutím:

$$U = RI + u_{emf} \quad (3.3)$$

$$I = \frac{U - u_{emf}}{R} \quad (3.4)$$

Dále také platí, že při nulových otáčkách  $\omega = 0$ , nedochází k indukovanému napětí  $u_{emf}$  a proud tekoucí rotorem nazýváme proud nakrátko a je roven:

$$I_S = \frac{U}{R} \quad (3.5)$$

Když se motor otáčí, elektromotorické napětí  $u_{emf}$  je přímo úměrné rychlosti otáčení  $\omega$ :

$$u_{emf} = k_e \cdot \omega, \quad (3.6)$$

kde  $k_e$  je zpětná elektromotorická konstanta.

Po dosazení:

$$U = RI + k_e \cdot \omega \quad (3.7)$$

Podobně je i kroučící moment  $M$  úměrný proudu protékajícím vinutím a na magnetickém poli. Řízen je tedy pouze protékajícím proudem. Moment  $M$  vzrůstá lineárně s proudem s proporcionalní konstantou  $k_m$ , nazývanou momentová konstanta:

$$M = k_M \cdot I \quad (3.8)$$

Dosazením obdržíme:

$$U = \frac{MR}{k_M} + k_e \cdot \omega \quad (3.9)$$

Mechanický výkon  $P_m$  na hřídeli je elektrický výkon  $P_e$  minus ztráty na odporu vinutí:

$$P_m = P_e - I^2R \quad (3.10)$$

Po dosazení:

$$M\omega = UI - I^2R \quad (3.11)$$

$$k_M I \omega = (IR + k_e \cdot \omega)I - I^2R \quad (3.12)$$

a po roznásobení a upravení rovnice obdržíme rovnost:

$$k_M = k_e = k_M \quad (3.13)$$

Takže po dosazení za  $k_e = k_M$ :

$$U = \frac{MR}{k_M} + k_M \cdot \omega \quad (3.14)$$

Z této rovnice nám už zbývá jenom vyjádřit vztah pro úhlovou rychlost  $\omega$ , zajímá nás její závislost:

$$\omega = -\frac{R}{k_M^2} M + \frac{U}{k_M} \quad (3.15)$$

Jedná se o rovnici přímkou s negativním sklonem.

Z posledního odvozeného vztahu plyne, že otáčky motoru lze ovlivnit změnou napájecího napětí motoru. Pokud je zapotřebí snížit otáčky motoru při současném zvýšení jeho momentu – což je v robotice častý požadavek, je nutno použít mechanickou převodovku.

Převodovka tvoří s motorem pohon. Pro správný výběr pohonu je nutno uvažovat požadovaný mechanický moment a rychlost výstupu pohonu, které jsou rozhodující pro výběr velikosti převodovky.

### 3.2 Popis motorů ve vybraném podvozku

V předchozích dvou podkapitolách byly uvedeny základní vlastnosti a princip stejnosměrného motoru. V této podkapitole budou uvedeny podrobnější informace o motorcích, které jsou obsaženy v zakoupeném podvozku.

Jak již bylo řečeno, v podvozku jsou obsaženy 4 stejné motorčky. Zde budou uvedeny informace převzaté z [17].

Jedná se tedy o stejnosměrné motory GM37-110 s převodovkou a převodovým poměrem 52,5:1. Napájení motorků jsem si mohl zvolit mezi 12V a 24V. Při návrhu jsem uvažoval celkové napájení robotu 12V, proto byla tato možnost zvolena i pro motory. Odběr jednoho z motorů bez zátěže je 85mA, při 0ot/min je odběr 850mA. Rychlost otáčení bez zátěže je 110ot/min. Rozměry motoru jsou 37x52mm. Motor je zobrazen na obrázku [Obrázek 3-4]:

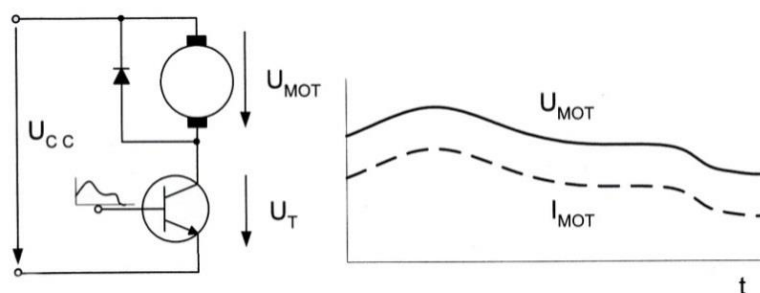


Obrázek 3-4 - Stejnosměrný motor s převodovkou GM37-110 [17]

### 3.3 Budicí obvody stejnosměrných obvodů [14]

Jak již bylo zmíněno v kapitole o principu činnosti stejnosměrného motoru, otáčky lze řídit pomocí změny napájecího napětí. Nejjednodušší – plynulá změna napětí na motoru však není vhodná z energetického hlediska. Nejjednodušším příkladem řízení je tzv. jedno-kvadrantový proudový regulátor [Obrázek 3-5]. Nazývá se tak proto, jelikož může otáčky řídit pouze v jednom směru a moment může generovat taktéž v jednom směru. Výkonový tranzistor, případně výkonový operační zesilovač je ovládán řídicím napětím. Celkové napájecí napětí se rozdělí na úbytek na výkonovém tranzistoru

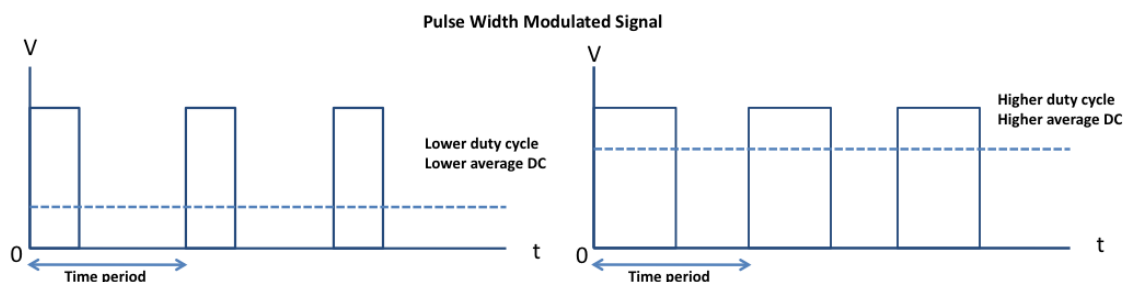
a úbytek na motoru. Ztráty u malých výkonů jsou zanedbatelné, tudíž tento způsob řízení může převážet nad jinými z důvodu slabého rušení spolu s cenou výkonové jednotky oproti celkové účinnosti.



Obrázek 3-5 - Jednokvadrantový proudový regulátor otáček [14]

V případě, že potřebujeme kromě řízení proudu měnit také jeho polaritu (z důvodu změny směru otáčení) musíme použít bipolárního způsobu napájení motoru. To lze v jednom případě bipolárním zdrojem, kde je jeden tranzistor uzavřen a druhý řízen. V současné době se dává přednost můstkovým spínačům („H-můstek“) a to v provedení lineárním (pro spojité řízení velikosti proudu) nebo spínacím – pro PWM řízení. Oba typy vyplývají z můstkového zesilovače. Odlišnost je v typech použitých tranzistorů.

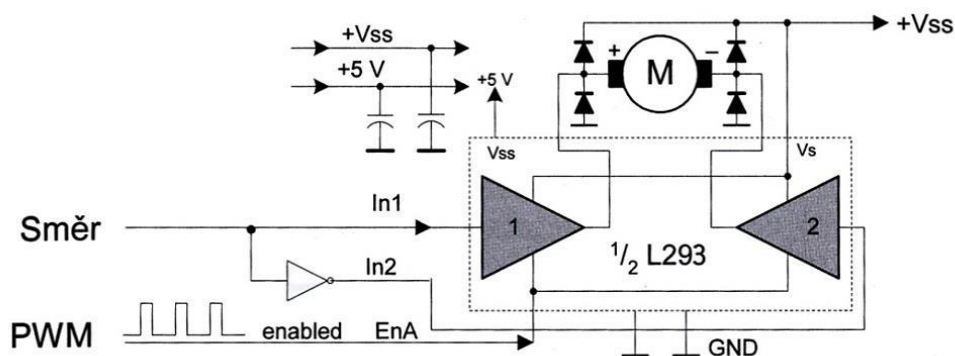
Další a pro našeho robota důležitý způsob řízení proudu protékajícím motorem je založen na rychlé změně plného napětí na motoru – napětí zapnuto/vypnuto. Konstantní napájecí napětí je časově rozděleno do impulzů s konstantní frekvencí a řízenou šířkou – tzv. PWM (pulse width modulation) [Obrázek 3-6]. Protože jsou výkonové tranzistory (zde spínací) buď zcela otevřeny nebo zcela zavřeny, nevznikají na nich téměř žádné ztráty. Otáčky motoru budou úměrné střední hodnotě protékajícího proudu.



Obrázek 3-6 - Průběh proudu při PWM [19]

Příklad řízení otáček motoru pomocí PWM je nakreslen na [Obrázek 3-7] využívající polovinu integrovaného obvodu L293. V následující kapitole o řízení a regulaci otáček motoru si ukážeme příklad zapojení tohoto obvodu pro oba motory našeho robota.

Jedná se o obvod, který obsahuje dva identické můstkové spínače (každý tvořen dvěma samostatnými polovodičovými H-mosty) včetně potřebné logiky zabráňující současnému sepnutí koncových tranzistorů jedné větve. Maximální frekvence PWM signálu je u tohoto obvodu 5kHz, maximální proud 600mA na kanál. Podrobnější informace můžeme čerpat např. v datasheetu [20] tohoto obvodu, či na dalších různých internetových stránkách.



Obrázek 3-7 - výkonová část můstkového proudového regulátoru otáček využívající signál PWM [14]

Jako výhody tohoto způsobu řízení proudu lze uvést nízké výkonové ztráty, vysokou účinnost a dostupnost i pro vysoké proudy (výkony). Mezi nevýhody patří vznik elektromagnetického rušení na rozhlasových frekvencích, vysoké ztráty při zastaveném motoru.

### 3.4 Dvojitý H-můstek pro řízení stejnosměrných motorů

V minulé kapitole byl uveden popis můstkového regulátoru, který dokáže řídit dva stejnosměrné motory. Pro řízení čtyř motorů jsem si vybral řídicí H-můstek z již zmiňovaného internetového obchodu firmy Snail Instruments. Má označení Bridge20 a jeho základem je obvod L298, jež vychází z již popsaného obvodu L293. L298 je stavěn na vyšší napětí i proudy. H-můstek je zobrazen na následujícím obrázku [Obrázek 3-8].

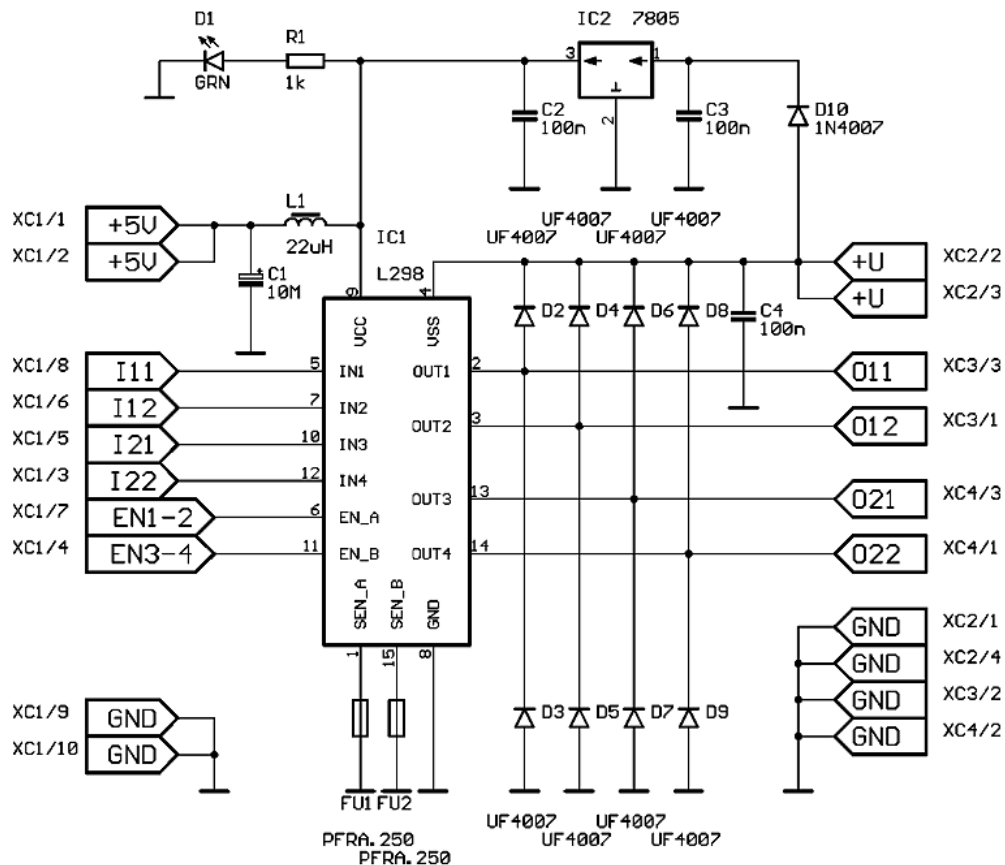


Obrázek 3-8 - Dvojitý H-můstek k řízení až 4 stejnosměrných motorků [21]

Napájecí napětí je výrobcem udáváno v rozsahu +9V až +28V. Při napájení H-můstku je nutno počítat s úbytkem napětí na můstku a to přibližně +2V. To znamená, že při napájení motorů 12V budeme potřebovat zdroj o minimálním napětí 14V. Propojení můstku s řídicím procesorem a motory je zajištěno konektory se zámkem (na obrázku H-můstku dva bílé konektory). Konektor pro řídicí signály není na obrázku vidět. Je schován i s obvodem L298 za černým chladičem.

Můstek dokáže řídit na každé větvi motor s proudovým zatížením 2A. V datasheetu[22] je také uvedena možnost paralelního zapojení dvou motorů k jedné větvi, která byla využita a bude popsána v následující kapitole.

Je zde také obsažen 5V stabilizátor napětí pro napájení vnitřní logiky L298 a dalších externích zařízení. Celý H-můstek je jištěn proti přepětí, či zkratům. Schéma zapojení celého můstku je uvedeno na následujícím obrázku [Obrázek 3-9].



Obrázek 3-9 - Schéma zapojení H-můstku [22]

Řízení motorů je zajištěno následujícím způsobem. L298 obsahuje vždy tři řídicí vstupy pro každý motor. Vstupy jsou označeny EN1-2 (EN3-4), IN1 (IN3) a IN2 (IN4). Přivedením logické 1 na vstup EN1-2 (EN3-4) zapneme otáčení motoru. Otáčky motoru je také možno regulovat přivedením PWM signálu (vysvětleno v kapitole 3.3) taktéž na vstup EN1-2 (EN3-4). Signály přivedené na vstupy IN1 (IN3) a IN2 (IN4) řídí směr otáčení. V následující tabulce Tabulka 3-1 jsou uvedené možnosti řízení otáčení.

Tabulka 3-1 – Hodnoty signálů k řízení motorů

IN1 (IN3)	IN2 (IN4)	EN1-2 (EN3-4)	Funkce
High	Low	High nebo PWM	Otáčení vpravo
Low	High	High nebo PWM	Otáčení vlevo
High	High	High nebo PWM	Motor zkratován (brzda)
Low	Low	High nebo PWM	Motor zkratován (brzda)
X	X	Low nebo 0	Motor odpojen

Vysvětlení:

High – hodnota logické 1

Low – hodnota logické 0

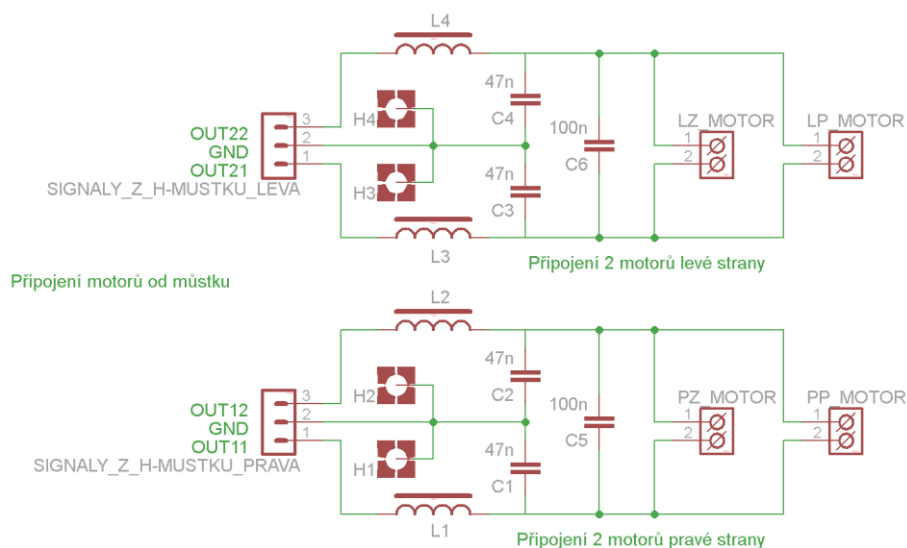
PWM – signál pulsní šířkové modulace

X – libovolná úroveň

Napětí na jednotlivých vstupech nesmí překročit +5V.

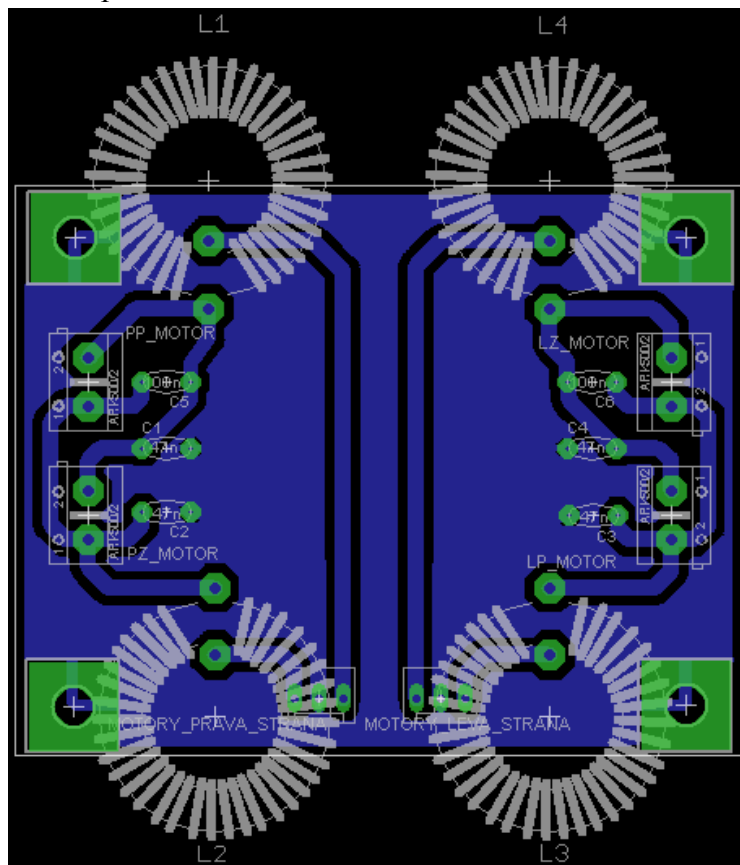
### 3.4.1 DPS pro paralelní zapojení motorů

V této kapitole je uveden způsob paralelního zapojení dvou motorů k jedné větvi H-můstku. Jak již bylo řečeno, jedna větev H-můstku je schopna řídit motor s proudovou zátěží 2A. V datasheetu můstku [22] jsem našel možnost zapojení dvou motorů k jedné větvi. Pro můj podvozek ideální možnost. Zvolil jsem tedy zapojení, které umožňuje ovládat synchronně každou stranu kol. Tedy posláním jediného signálu do H-můstku dojde k roztočení obou motorů jedné strany současně. Při poslání spouštěcího signálu do druhé větve L298 dojde pochopitelně k roztočení motorů opačné strany robotu. Díky této desce jsem tedy nemusel pořizovat dva H-můstky a ušetřil jsem i dalších 6 pinů řídicího mikrokontroléru. V zapojení [Obrázek 3-10] jsou obsaženy kondenzátory a cívky zajišťující vyrušení šumu z řídicího signálu. Nejsou nutné, ale zvyšují spolehlivost celého zapojení. Proto jsem je také do své práce zahrnul.



Obrázek 3-10 - Schéma DPS pro paralelní zapojení motorů

Pro názornost je uveden obrázek desky plošných spojů [Obrázek 3-11], kterou bylo nutné navrhnout a nechat vyrobit. Výroba byla zadána ve školní laboratoři. Osazení součástkami jsem provedl sám. Obrázek výsledné osazené desky umístěné uvnitř robotu bude uveden na konci práce.



Obrázek 3-11 - DPS paralelního zapojení motorů

### 3.5 Ukázka zdrojového kódu z mikrokontroléru

V této kapitole je pro ukázkou uvedena funkce pro pohyb robotu dopředu nahraná do mikrokontroléru. Podle této funkce jsou odvozeny další funkce zajišťující pohyb robotu dozadu, doleva, doprava a stop.

```
void go_forward()           //Function for forward movement
{
    DDRA = 0b11110000;
    PORTA = 0b01010000;    //Sending values on PORTB for H-bridge

    //Left side - forward
    //OUT11 - PIN7 - HIGH
    //OUT12 - PIN6 - LOW

    //Right side / forward
    //OUT21 - PIN5 - HIGH
    //OUT22 - PIN4 - LOW
}
```

Jelikož se jedná o první funkci programu uvedenou v mé práci, bude rozebrána podrobněji. Další funkce už budou podobného charakteru.

Nejdůležitějšími částmi mikrokontroléru pro nás v této kapitole budou vstupně výstupní piny, dále jen V/V piny. Naučíme se s nimi pracovat, jednoduše tedy spínat piny H-můstku a tím pádem rozjet motory.

Jak již bylo řečeno v kapitole o H-můstku, je nutné do něj přivést 3 signály. Dva pro určení směru pohybu a jeden pro ovládání rychlosti.

V mé aplikaci jsem si zvolil k ovládání směru otáčení motorů PORTA, piny PA7 až PA4. Záměrně budu uvádět stejné označení komponent jako v programu pro mikrokontrolér, aby se s nimi čtenář rychleji seznámil.

Při používání V/V pinů je nutné provést nastavení pinu buď jako vstupního nebo jako výstupního. V našem případě potřebujeme signál z mikrokontroléru odeslat, proto piny PA7 až PA4 nastavíme jako výstupní zapsáním logické 1 a tedy vložním následujícího příkazu:

```
DDRA = 0b11110000; //Setting PA7-PA4 as output.
```

Tento řádek se bude opakovat ve všech dalších funkcích zajišťujících pohyb.

Chceme-li odeslat signál z mikrokontroléru do zařízení, pošleme ho nastavením příslušného portu do logické 1, tak jak nám nařizuje tabulka v kapitole o H-můstku.

Zavoláním následující funkce nastavíme robot pro pohyb vpřed.

```
PORTA = 0b01010000; //Sending values on PORTB for H-bridge
```

Nyní zbývá už jen na piny EN1-2 a EN3-4 poslat signál s povelom ke spuštění. Toho můžeme dosáhnout dvěma způsoby. A to buď sepnutím těchto dvou pinů přímo nastavením do logické 1 nebo pomocí již zmiňované pulsní šířkové modulace, neboli PWM. Vzhledem k šetrnosti a zacházení s motory je lepší využít PWM, a tím i hladšího průběhu rozjezdu, než spustit motory ihned na plný výkon.

Ke generaci PWM signálu využijeme vnitřní periferie mikrokontroléru a tou je čítač/časovač s možností generace PWM signálu.

Jeho správnou funkci nastavíme v programu zavoláním vytvořené funkce `init_PWM()`, která obsahuje nastavení čítače/časovače2. Podrobné nastavení nalezneme opět v datasheetu mikrokontroléru [23].

```
void init_PWM(void)
{
    //Fast PWM Mode
    TCCR2A |= (1 << COM2A1) | (0 << COM2A0); //Compare Output mode
    TCCR2A |= (1 << COM2B1) | (0 << COM2B0); //Compare Output
    //Setting type of waveform generation - Fast PWM
    TCCR2A |= (1 << WGM21) | (1 << WGM20);
    TCCR2B |= (0 << WGM20);

    //Prescaler settings - 128
```

```

TCCR2B |= (1 << CS22) | (0 << CS21) | (1 << CS20);

//Init TimerCounter2 value to 0 for no wheel rotation
OCR2B = OCR2A = 0;
}

```

Pro ukázkou a vysvětlení je uvedena funkce pro zaslání PWM signálu na výstupní piny čítače, tedy piny PD7 a PD6, které jsou propojeny s piny H-můstku. Jednotlivé proměnné je nutné nadefinovat před volanou funkcí.

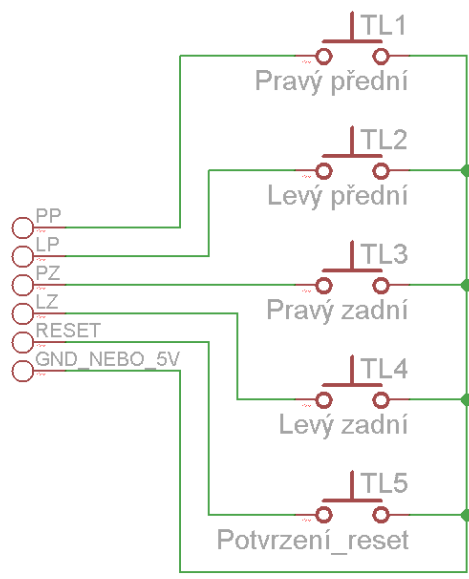
```

void acceleration(void)
{
    for(i=0; i < 20; i++)
    {
        PWM_value1 += 20;
        if(PWM_value1 <= 100) PWM_value1 = 100;
        if(PWM_value1 >= 255) PWM_value1 = 255;
        //Writing PWM value to Timer/Counter2 register to create a PWM
        signal on PD7 and PD6 pins
        PWM_value1_char = PWM_value1;    //OCR2B and OCR2A are 8-bit
                                         registers so we need PWM value
                                         in char
        OCR2B = OCR2A = PWM_value1_char;
    }
}

```

Ve funkci je obsažen `for` cyklus pro periodické narůstání hodnoty PWM po 20. Tato hodnota byla experimentálně zjištěna jako přijatelná. Dále jsou přidány podmínky pro ošetření mezních stavů. Nejnižší hodnota PWM nastavena na 100 a nejvyšší na 255. Při hodnotě PWM nižší než 100 docházelo k pouhému pískání motorů bez pohybu a tedy jejich postupnému poškozování. Předposledním řádku dochází k přepsání číselné hodnoty `PWM_value1` na znakovou hodnotu, která je nutná k zápisu do registrů `OCR2B/A`. Tato hodnota je poté odeslána na již zmiňované piny PD7 a PD6, které jsou propojeny s řídicími piny H-můstku pro levou a pravou stranu. Funkce pro zastavení bude analogická, jenom s rozdílem periodického snižování PWM hodnoty.

Funkce zajišťující pohyb mohou být vloženy například do části zdrojového kódu, jež se stará o možnost ovládní robotu uživatelem manuálně a to pomocí vytvořeného dálkového ovladače, uvedeného na obrázku [Obrázek 3-13]. Pro rychlejší pochopení zapojení je také uvedeno schéma [Obrázek 3-12] manuálního ovladače. Z krabíčky s tlačítky vede 6 drátů, kdy 5 z nich jsou logické úrovně tlačítek a šestý z nich je připojen k opačným koncům tlačítek, představující společný vodič. Tento vodič můžeme na kytu připojit buď k +5V nebo na GND. Nejdříve je tedy nutno nastavením bitu patřičného pinu v registru `DDRx` do logické 0 nastavit pin jako vstupní. Poté zapsáním logické 1 do stejného bitu registru `PORTx` připojíme pull-up rezistor, který nám ošetřuje možnost spínání signálem GND. Ke čtení stisku tlačítek je využita funkce `bit_is_set(PORT, PIN)`, jejíž deklarace (říkáme, co uvedená funkce dělá) je opět uvedena na začátku programu.



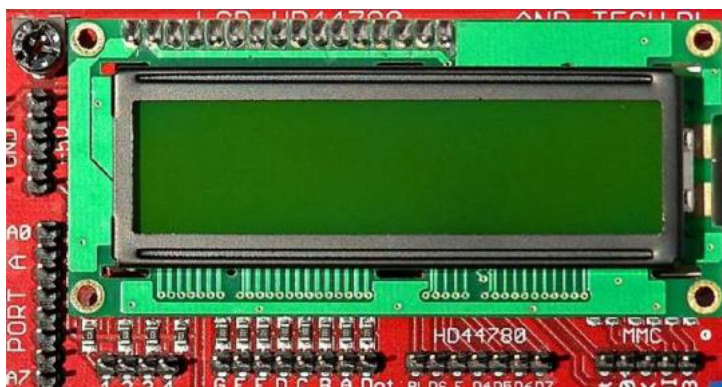
**Obrázek 3-12 - Schéma zapojení manuálního ovladače**



**Obrázek 3-13 - Manuální ovládání pro robot**

## 4. KOMUNIKACE S UŽIVATELEM POMOCÍ LCD DISPLEJE

V této kapitole bude pojednáno o vnější komunikaci uživatele s robotem. Na kitu mohou být využity tři způsoby indikace zvolené položky v menu. Jednou z možností je využití postupného rozsvěcování LED diod na kitu. Další z možností je využití segmentového displeje, k jehož ovládání je potřeba většího počtu ovládacích pinů mikrokontroléru. Poslední a pro mě nejpříjemnější možností zobrazení jakýchkoli informací je využití LCD displeje, na kterém je vytvořeno jednoduché menu. Uživatel robotu je tedy přímo vybízen vybrat jednu z možností a robot bude konat pohyby, jež si uživatel vybere.



Obrázek 4-1 - Detail LCD displeje [6]

### 4.1 Popis LCD displeje posazeného na kitu

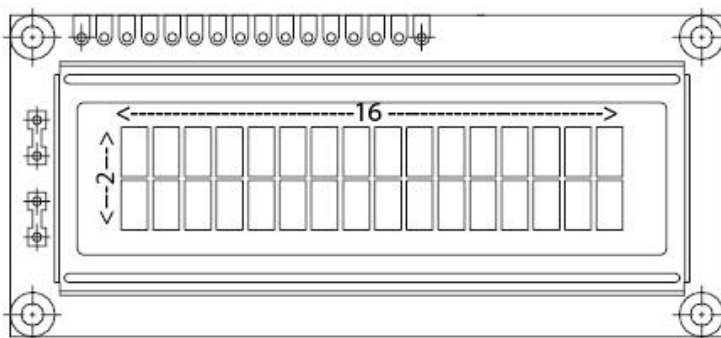
Displej je osazen 16 piny, jejichž vlastnosti budou popsány v následující tabulce Tabulka 4-1 převzaté z [24] a upravené podle vlastních potřeb.

Tabulka 4-1 - Popis pinů LCD displeje

Číslo pinu/pinů	Signál	Popis
1	GND	0V
2	Ucc	+5V
3	Contrast Voltage(kontrast)	v našem případě pomocný trimr
4	R/S Register Select (volba registru)	0=zápis do registru, 1=zápis dat
5	R/W Read/Write from LCD	0=Zápis do LCD, 1=Čtení z LCD
6	E Clock	Každých 450ns dojde k odeslání kladného pulsu k inicializaci přenosu
D0-D3	Read from LCD	Piny potřebné k čtení dat z LCD
D4-D7	Write to LCD	Piny potřebné k zápisu dat do LCD

V této práci se zabývám pouze možností odeslání dat na displej, piny D0-D3 nebudou tedy v zapojení vůbec využity.

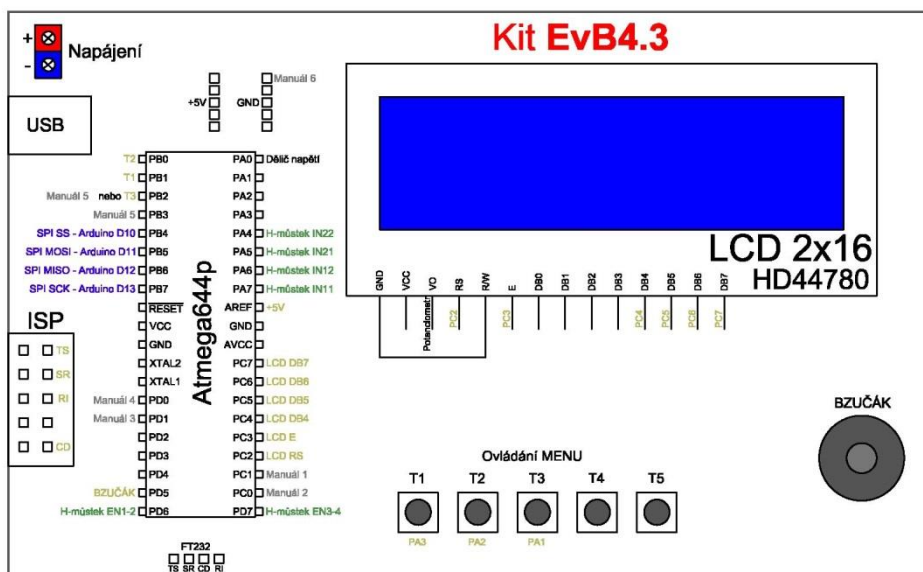
Displej je osazen standardně využívaným řadičem HD44780, což mu dává možnost rozmístění znaků na pomyslné mapě displeje 16x2. To znamená, že máme možnost využití 2 řádků o 16 znacích na délku. Je vhodné vysvětlit tuto problematiku ihned na začátku, protože rozložení znaků pro nás bude klíčové při zápisu znaku, či celých řetězců na LCD. Pro pochopení je uveden obrázek [Obrázek 4-2].



Obrázek 4-2 - Nákres LCD displeje s vysvětlením pozic znaků [25]

## 4.2 Schéma zapojení LCD k AVR mikrokontroléru

První a nejdůležitějším krokem, který je nutné před zápisem dat do LCD displeje udělat je výběr a konfigurace použitých pinů v knihovně, která bude popsána dále, a jejich analogické propojení na řídicí desce. Nejvhodnější volbou v tomto případě je obsazení jednoho portu mikrokontroléru pouze pro ovládání LCD displeje. V mém případě byl zvolen PORTC se zapojením pinů uvedeným na následujícím schématu [Obrázek 4-3].



Obrázek 4-3 - Schéma kitu EvB4.3 s popisky zapojení LCD k PORTu C

## 4.3 Popis stáhnuté knihovny pro obsluhu LCD

Displejů podobného typu je v dnešní technicky vyspělé době využito ve velkém množství elektronických zařízení a systémů. Ve volné chvíli pátrání na internetu se mi podařilo najít potřebnou knihovnu obsahující funkce k obsluze tohoto LCD. Je dostupná z následujícího odkazu [24].

### 4.3.1 Definování portu ovládajícího LCD

Knihovna stažená z uvedené stránky obsahuje 2 soubory, které je nutno přidat do projektu se zdrojovými kódy vložením následujícího textu `#include "lcd.h"`

V souboru s příponou `.h` je nutno správně nadefinovat jaký port a které piny byly pro komunikaci mikrokontroléru s LCD displejem vybrány. Pro příklad uvádím část zdrojového kódu, kde bylo nutno tato data změnit.

```
#define LCD_PORT          PORTC          /**< port for the LCD lines */
#define LCD_DATA0_PIN    4              /**< pin for 4bit data bit 0 */
#define LCD_DATA1_PIN    5              /**< pin for 4bit data bit 1 */
#define LCD_DATA2_PIN    6              /**< pin for 4bit data bit 2 */
#define LCD_DATA3_PIN    7              /**< pin for 4bit data bit 3 */
#define LCD_RS_PORT      PORTC          /**< port for RS line */
#define LCD_RS_PIN       2              /**< pin for RS line */
#define LCD_E_PORT       PORTC          /**< port for Enable line */
#define LCD_E_PIN        3              /**< pin for Enable line */
```

### 4.3.2 Základní funkce knihovny

Knihovna pro obsluhu LCD displeje obsahuje mnoho funkcí. V mém projektu jsem zatím využil pouze základní z nich a ty budou nyní rozebrány. Při programování byl zjištěn jeden z důležitých poznatků. Uvedené funkce je nutno volat v opakujících se cyklech programu. Nejvhodnějšími použitými cykly jsou `while` cyklus doplněný vhodnou podmínkou běhu programu, či `if` cyklus splňující stejný účel.

#### 4.3.2.1 Inicializace displeje

První využitou a potřebnou funkcí je funkce `void lcd_init(uint8_t dispAttr)`, která nám zajišťuje počáteční inicializaci displeje. Výstupní parametr funkce není žádný, jelikož je před názvem funkce obsaženo slovo `void`, což znamená v překladu „prázdný“. Jejím vstupním parametrem vepsaným do kulatých závorek musí být jeden z následujících příkazů:

- `LCD_DISP_OFF` - vypnutý displej
- `LCD_DISP_ON` - zapnutý displej, vypnutý kurzor
- `LCD_DISP_ON_CURSOR` - zapnutý displej, zapnutý kurzor
- `LCD_DISP_CURSOR_BLINK` - zapnutý displej, blikající kurzor

### 4.3.2.2 Určení pozice displeje a smazání textu

Další možnou funkcí zajišťující pohyb po displeji a možnost výpisu textu na libovolné pozici je funkce `void lcd_gotoxy(uint8_t x, uint8_t y)`. Funkce opět nevrací žádný parametr. Vstupními hodnotami jsou celočíselné proměnné `x` a `y` představující souřadnice na displeji. Jedinou důležitou vlastností je číslování od nuly. Pro horizontální souřadnici `x` používáme tedy hodnoty 0-15, pro vertikální `y` souřadnici 0 a 1. Princip pohybu je také naznačen na obrázku [Obrázek 4-2].

V mnoha případech potřebujeme vymazat veškerý text na displeji. K tomuto účelu slouží funkce `void lcd_clrscr(void)`. Funkce nemá žádný vstupní ani výstupní parametr. Stačí ji tedy ve zdrojovém kódu zavolat pouhým napsáním textu `lcd_clrscr()`.

### 4.3.2.3 Vložení textu

K dokonalosti nám už chybí jen funkce zajišťující, jak již bylo zmíněno v opakujícím se cyklu, zápis textu na LCD. Touto funkcí je `void lcd_puts(const char *s)`.

Nyní je důležité si říct o přípravě textu před odesláním do LCD pomocí popsané funkce. Tuto část rozdělíme do dvou podoblastí. Problematika bude vysvětlena na vypsání informace ohledně napětí akumulátoru. Na prvním řádku budeme chtít vypsát text BATERIE a na druhém řádku měnicí se text U=16.80V.

Text baterie se nebude průběžně měnit, proto jej můžeme do funkce zadat konstantně přímo. K tomuto můžeme využít i funkce `lcd_gotoxy`, kterou přejdeme na požadovanou pozici a funkcí `lcd_puts("BATERIE")`; vypíšeme text. Další možností je vypuštění funkce `lcd_gotoxy` a vložení řetězce znaků o 16 znacích, kdy doplníme volná místa mezerou. Výsledek tedy bude zapsán funkcí `lcd_puts(" BATERIE ")`. Mě osobně přijde druhá možnost vypsání textu s mezerami přijatelnější a jednodušší.

Nyní se dostáváme k měnicímu se textu využitelného např. u měřených hodnot, hodnot čítačů a dalších. Před odesláním textu na displej je nutno vytvořit proměnnou typu `string` s názvem například „`text_voltage`“ vložení kódu `string text_voltage[32]`. Číslo v hranatých závorkách značí možný počet znaků v řetězci `text_voltage`. Při uložení námi požadovaného textu bude automaticky počet změněn.

K uložení měnicí se hodnoty využijeme funkci `sprintf(char *__s, const char *__fmt, ...)`, obsaženou v interní knihovně AVR studia `stdio.h`, kterou je nutné již vysvětleným způsobem vložit do projektu. Funkce má následující vlastnosti. Prvním vstupním parametrem funkce je řetězec, do něhož budeme zapisovat, pro nás je to již deklarovaný (oznámíme programu, že určitá proměnná je vytvořena) řetězec znaků `text_voltage`. Dalším vstupním parametrem je řetězec znaků uvedený v uvozovkách s patřičným textem. Zde `"%.2d%.1dV"`.

Za procenta je nutno dosadit celou a desetinou část čísla, tedy `U_radix_part`, `U_decimal_part`, přečtenou a převedenou z registru A/D převodníku, jak bude vysvětleno v kapitole o hlídání napětí akumulátoru. Tyto proměnné vkládáme jako další parametry funkce v pořadí, v jakém jsou v řetězci znaky `%` obsaženy. Jednoduše řečeno, každému `%` odpovídá jedna proměnná.

Nakonec už jen zapíšeme proměnnou `text_voltage` jako vstupní parametr funkce `lcd_puts`. Výsledná část kódu bude vypadat následovně:

```
printf(text_voltage, "%.2d.%.1dV", U_radix_part, U_decimal_part);  
lcd_gotoxy(6,1);  
lcd_puts(text_voltage);
```

Zde jen opět připomínám nutnost definovat proměnné a možnost přidání mezer do zapsaného řetězce s napětím a vypuštění příkazu `lcd_gotoxy`.

## 4.4 Popis a ukázka vytvořeného menu

V této kapitole bude uvedena pouze část zdrojového kódu vytvořeného MENU zobrazovaného na LCD displeji. Další text obrazovek a použitých funkcí je analogicky doplněn v dalším `case` příkazu. Z toho plyne, že je možné zobrazit jakékoliv množství obrazovek. Stačí pouze doplnit další `case` příkazy.

### 4.4.1 Zdrojový kód

Před následujícím textem je ve zdrojovém kódu uvedena funkce pro čtení stisku tlačítka, kterým dojde k aktivaci pohybu robotu.

```
while(start == false)  
{  
    . //podmínky zajišťující průchod menu pomocí zvyšování hodnoty v proměnné  
    . //option(volba), ošetření mezních stavů menu  
    .  
    .  
        switch (option)  
        {  
            case 1:  
            {  
                lcd_puts(" UKAZKOVY MOD ");  
                lcd_gotoxy(0,1);  
                lcd_puts("      zmenit  >>");  
            }  
            break;  
        }  
}
```

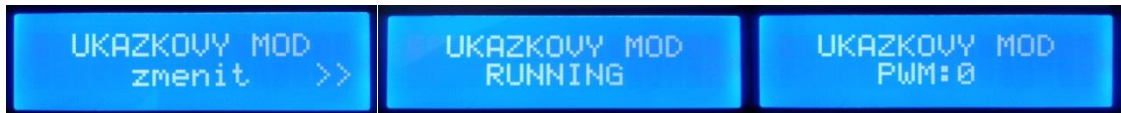
Jakmile dojde k již zmiňované aktivaci pohybu robotu, či jiné vykonávané funkce, dojde ke změně proměnné `start` na hodnotu `true` a program skočí do podmínky s hlavičkou `while(start == true)`, která obsahuje analogický příkaz `switch`. V tomto novém příkazu dojde k poslání textu `RUNNING` do LCD pro indikaci chodu programu nebo pohybu robotu a také ke spuštění potřebné funkce. Při chodu ukázkového programu a programu manuálního ovládání je text `RUNNING` přepsán aktuální PWM hodnotou.

```

while(start == true)
{
    switch(option)
    {
        case 1:
        {
            lcd_puts(" UKAZKOVY MOD ");
            lcd_gotoxy(0,1);
            lcd_puts("  RUNNING  ");
            _delay_ms(500); //Čekej 0,5 sekundy
            illustration_mode();//Funkce zajišťující pohyb
        }
        break;
    }
}

```

#### 4.4.2 Ukázka obrazovek menu



Obrázek 4-4 - LCD: Ukázkový mód - průchod menu (vlevo), volba potvrzena (uprostřed), zobrazení hodnoty PWM signálu při jízdě (vpravo)



Obrázek 4-5 - LCD: Manuální ovládání - průchod menu (vlevo), volba potvrzena (uprostřed), zobrazení hodnoty PWM signálu při jízdě (vpravo)



Obrázek 4-6 - LCD: Scanner - průchod menu (vlevo), volba potvrzena - čekání na spojení se skenerem (vpravo)



Obrázek 4-7 - LCD: Enkodéry - průchod menu (vlevo), volba potvrzena (vpravo) - při stisku dvou tlačítek dojde k inkrementaci počítadla pulzů pro pravou a levou stranu



Obrázek 4-8 - Indikace nabití baterie

## 5. SPRÁVA NABÍJENÍ AKUMULÁTORŮ (BATTERY MANAGEMENT) A STABILIZACE NAPĚTÍ

### 5.1 Nabíječ Li-Pol akumulátorů

Nabíjení veškerých akumulátorů a baterií je věda na dlouhou dobu. Baterie je možné nabíjet spoustu způsoby, je možné si vyrobit vlastní nabíječ, nebo pořídit odzkoušený a za přijatelnou cenu i rozumný nabíječ. Při nákupu akumulátorů v RCMarketu v Brně jsem si pořídit pro mé účely naprosto postačující nabíječ Equilibrium Mini V2 zobrazený níže [Obrázek 5-1].



Obrázek 5-1 - Mikroprocesorový nabíječ akumulátorů Equilibrium Mini s příslušenstvím, [26]

Z akumulátoru jsou vyvedeny dva svazky kabelů zakončeny konektory. Jeden svazek tlustších drátů je určen k hlavnímu přívodu do baterie. Další svazek 5 vodičů je určen pro balancer baterie, jež slouží k rovnoměrnému nabití jednotlivých článků baterie. Výstupní vodiče balanceru jsou opatřeny konektorem, který nemusí vždy sedět. Kvůli tomuto problému se využívá balanční destička uvedena na obrázku [Obrázek 5-1] vpravo dole. Konektor balanceru baterie není připojen přímo do balanční destičky. Spojení je přerušeno DIN konektorem, díky kterému není nutno baterii z robotu při nabíjení vyndat. Další nutností je do nabíječky připojit dvěma banánky již zmiňovaný hlavní dvou vodičový přívod baterie. Foto robotu při nabíjení bude zobrazeno na konci práce.

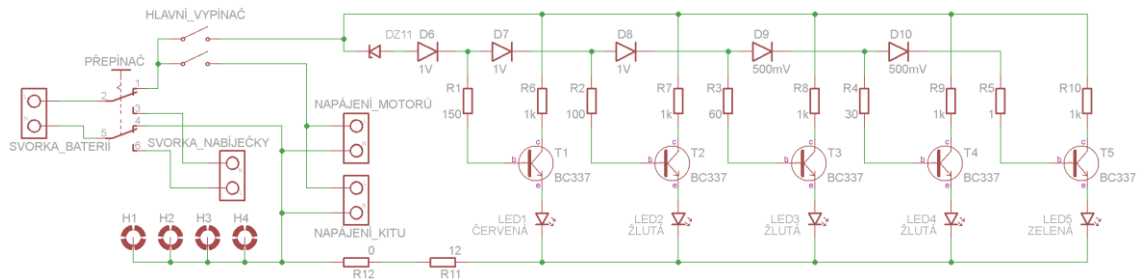
Nyní je možno nabíječ připojit k síti 220V přes přiložený adaptér. Na displeji se zobrazí jednoduché přehledné menu. Z nabídky vybereme nabíjení Li-Pol čtyř článkových akumulátorů. Další důležitou možností je výběr nabíjecího proudu. Maximální výrobcem udávaný nabíjecí proud je 2,7A. Všeobecně je známo, že je lepší nabíjet akumulátory menšími proudy po delší dobu. Tímto také přispíváme k šetrnému zacházení s bateriemi. Při nabíjení tedy vždy volím nabíjecí proud 1A.

## 5.2 Indikace nabití baterie pomocí LED diod

V této kapitole bude jednoduše vysvětlen princip indikace napětí na akumulátoru. Schéma bylo převzato z návodu nalezeného na serveru [www.pandatron.cz](http://www.pandatron.cz). Jedná se o návod určený pro autobaterie. Pro svou potřebu jsem přepočítal hodnoty, obměnil součástky, schéma [Obrázek 5-2] překreslil a upravil k obrazu svému.

Činnost tohoto zařízení je následující. Při zvyšujícím se napětí na autobaterii, či akumulátoru dochází k postupnému otevírání tranzistorů T1-T5 a tím pádem i rozsvěcování barevně odlišených diod.

Při plně nabitém akumulátoru svítí všechny LED diody. Při poklesu napětí pod 15V dojde k postupnému zhasínání jednotlivých LED po přibližně 0,5V. Hraniční hodnota, kdy svítí pouze červená LED je 13V. Nyní je nutné robot dále nepoužívat a akumulátor nabít.



Obrázek 5-2 - Schéma zapojení DPS s indikací nabití akumulátoru, část s tranzistory převzata z [27]

Pro ukázkou je na následujícím obrázku vlevo [Obrázek 5-3] zobrazena deska plošného spoje. Deska je navrhnutá podle předchozího schématu. Deska téměř koresponduje se schématem. Je zde pouze malý rozdíl. Deska byla původně navržena bez přepínače odpojujícího veškerou elektronickou část robotu od baterie při jejím nabíjení.

Schéma je tedy aktuální, deska je upravena pomocí drátových propojek. Součástky mimo desku jsou propojeny přímo v robotu. Výsledný pás LED diod je možné vidět na následujícím obrázku [Obrázek 5-3] vpravo nebo fotkách celkového pohledu na robot v poslední kapitole.



Obrázek 5-3 - Návrh DPS pro základovou desku s indikací nabití akumulátoru (vlevo), foto pásu LED diod na robotu (vpravo)

## 5.3 Hlídaní napětí akumulátoru pomocí mikrokontroléru

V minulé podkapitole byl uveden popis indikace napětí akumulátoru pomocí postupného rozsvěcování LED diod. V tomto případě stačilo uživateli pohlédnout na světelný pás a ihned viděl, zda je robot schopen jízdy, či ne. V této podkapitole si uvedeme metodu pro přesnější kontrolu napětí akumulátoru a to pomocí A/D převodníku obsaženého v mikrokontroléru. Jeho základní vlastnosti budou nyní uvedeny a jsou převzaty z datasheetu mikrokontroléru [23].

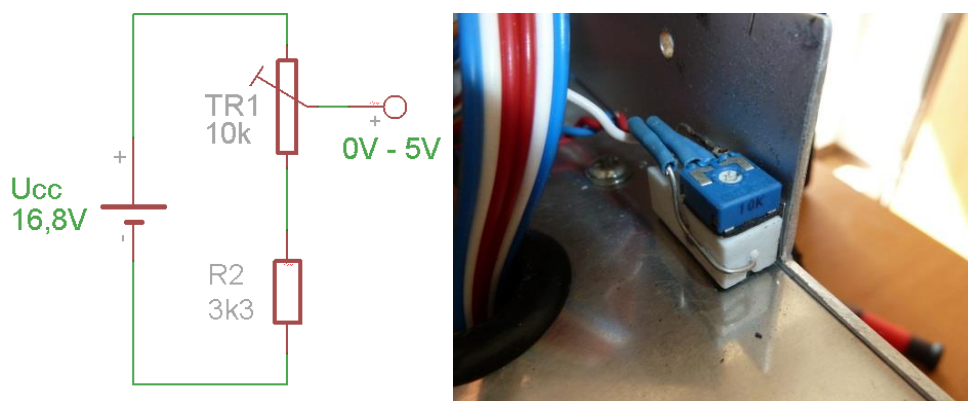
Důležitým parametrem pro mé využití je rozlišení 10bitů. To znamená, že vstupní analogová hodnota přivedená na vstupní kanál A/D převodníku bude lineárně rozprostřena na  $2^{10}$  digitálních hodnot binárního kódu. Zjednodušeně to znamená, že analogový rozsah 0V – 5V odpovídá 0-1023( $2^{10}$ ) hodnotám v mikroprocesoru.

Pro možnost měření napětí na kanálu A/D převodníku je nutné zvolit referenční napětí. Je to napětí, vůči němuž převodník porovnává aktuální měřenou hodnotu. Referenční napětí může být buď generováno externě nebo interně, přímo v mikrokontroléru. Nastavením bitů `REFS1` a `REFS0` v registru `ADMUX` můžeme interní referenční napětí buď vypnout, nastavit jako 1,1V nebo 2,56V. Jelikož na kitu EvB4.3 jsou vyvedeny hřebínky s 5V, využil jsem možnosti externího referenčního napětí. Pokud zvolíme tuto možnost, je nutné na pin `AREF` mikrokontroléru přivést toto referenční napětí. Nastavením správných hodnot do bitů `MUX4` – `MUX0` v registru `ADMUX` nastavujeme správný vstupní kanál převodníku pro měřené napětí.

Registr `ADCSRA` řídí a informuje nás o průběhu převodu. Výsledek převodu v 8 bitovém čísle si můžeme přečíst v registru `ADCH`. Příklad zdrojového kódu bude uveden v následujícím textu.

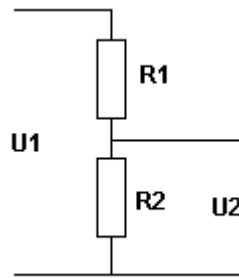
### 5.3.1 Dělič napětí

Jelikož referenčnímu napětí odpovídá maximální hodnota +5V, bylo nutné upravit také napětí z baterie přiváděné na měřicí kanál A/D převodníku. Toho dosáhneme děličem napětí s nastavitelným trimrem (proměnným odporem). Díky tomuto zařízení můžeme tedy jednoduše měřit napětí na akumulátoru pomocí mikrokontroléru. Ukázka zapojení děliče napětí je zobrazena v následujícím obrázku vlevo [Obrázek 5-4].



Obrázek 5-4 - Dělič napětí - schéma s vypočítanými hodnotami (vlevo), umístění v robotu (vpravo)

Obecné schéma děliče napětí je nakresleno na následujícím obrázku [Obrázek 5-5]:



Obrázek 5-5 - Obecné schéma děliče napětí [28]

Pro sestavení děliče je nutno vypočítat hodnoty součástek. Typické hodnoty rezistorů je možno najít v jakémkoli obchodě s elektronikou. Hodnota rezistoru  $R_2$  byla zvolena z dostupných součástek, pokud vyjde nedostupná hodnota pro trimr  $TR_1$ , je nutno  $R_2$  upravit a provést výpočet znovu.

Výchozí hodnoty pro výpočet:

- Maximální napětí akumulátoru  $U_1 = 16,8V$
- Maximální možné napětí na vstupu A/D převodníku  $U_2=5V$
- Zvolená hodnota odporu  $R_2 = 3k3 = 3300\Omega$

Nutno vypočítat hodnotu trimru  $R_1$ . Na konci výpočtu bude odpor  $R_1$  nahrazen příslušnou hodnotou trimru.

Dále budeme vycházet ze vztahu pro výpočet děliče napětí:

$$\frac{R_2}{R_1 + R_2} = \frac{U_2}{U_1} \quad (5.1)$$

Nyní je nutno vyjádřit  $R_2$ :

$$R_2 = \frac{U_2}{U_1} ((R_1 + R_2)) \quad (5.2)$$

Úpravou dostáváme následující vzorec pro  $R_1$ :

$$R_1 = R_2 \left( \frac{U_1}{U_2} - 1 \right) \quad (5.3)$$

Nyní je možné do rovnice dosadit příslušné hodnoty a vypočítat výslednou hodnotu uvedené výše:

$$R_1 = 3300 \left( \frac{16,8}{5} - 1 \right) = 7788\Omega \quad (5.4)$$

Hodnotu rezistoru  $R_1$  je nutno nahradit nastavitelným trimrem o vyšší hodnotě. Nejbližší přípustná katalogová hodnota trimru je  $10k\Omega$ . Nastavením děliče napětí pomocí voltmetru stáhneme maximální hodnotu na nutnou hodnotu  $R_1$ , tedy  $7788\Omega$ .

Tímto dosáhneme lineární závislosti mezi napětím akumulátoru a možným vstupním napětím A/D převodníku.

Jedinou nevýhodou tohoto zapojení je občasné překontrolování nastavení trimru například pomocí přesnějšího multimetru.

## 5.3.2 Program mikrokontroléru

Pomocí této inicializační funkce nastavíme správný chod A/D převodníku.

```
void init_AD(void)      //Init function for AD converter
{
    ADMUX = 0x00;      //Reference voltage set for reading from AREF pin,
                       //pins MUXn not set because I am not using gain
    //Register ADCSRA controls conversation and informs us about it
    ADCSRA |= (1 << ADEN);    //ADC Enable
    ADCSRA |= (1 << ADSC);    //ADC Start Conversation
    ADCSRA |= (1 << ADATE);   //Auto Trigger Enable - automatic start of
                               //conversation - trigger source is selected in
                               //ADCSRB register
    ADCSRA |= (1 << ADIF);   //ADC Interrupt Flag - by setting this bit ADC
                               //interrupt flag will be cleared
    ADCSRA |= (1 << ADIE);   //ADC Interrupt Enable
    //Prescaler settings
    ADCSRA |= (1 << ADPS2);   //Prescaler settings 000-2; 001-2; 010-4; 011-
                               //8; 100-16; 101-32; 110-64; 111-128
    ADCSRA |= (1 << ADPS1);   //In our case the highest prescaler (128)
    ADCSRA |= (1 << ADPS0);
}
```

Následujícími příkazy v programu přečteme hexadecimální číslo z ADC registru a poté provedeme přepočítání na celou a desetinnou část. Opět zmiňuji, že je nutné jednotlivé proměnné nadefinovat před hlavní smyčkou programu.

```
ADC_value = ADC;
//voltage convert from ADC register
U_mv = ((long)ADC_value*16800/1023);
//Getting radix part by modulo dividing
U_radix_part = ((U_mv % 100000) - (U_mv / 1000))/1000;
//Getting decimal part by modulo dividing
U_decimal_part = ((U_mv % 1000) - (U_mv / 10))/100;
```

Příkazem `printf`, který je obsažen v knihovně `string.h`, kterou je nutno vložit do programu příkazem `#include <string.h>` na začátku programu, spojíme celou a desetinnou část do jednoho řetězce a příkazem `lcd_puts` pošleme text s informací o napětí na LCD displej.

```
//connection for radix and decimal part and saving to one string text_V
printf(text_voltage, "%.2d%.1dV", U_radix_part, U_decimal_part);
lcd_puts(text_voltage);
```

Další pro mě důležitou vlastností využití A/D převodníku k měření napětí je možnost zapojení do projektu buzzer neboli klakson obsažený také na kitu EvB4.3.

Přivedeme-li na pin klaksonu kladný puls z mikrokontroléru, dojde k jeho sepnutí a tedy vydávání akustického signálu. Této možnosti jsem taktéž využil ke svému

projektu. Při poklesu napětí akumulátoru pod 14V dojde k celkovému zastavení robotu, napsání na LCD textu „!!!POZOR!!!, BATERIE VYBITA“ a přerušovanému houkání klaksonu.

Důležitá část kódu, ve které dojde v podmínce `if` k porovnání měřené hodnoty s konstantní hodnotou 14000mV a poté k rozhodnutí, zda je klakson sepnut, či ne a nápisy zobrazeny, je uvedena v následujícím textu:

```
ISR(ADC_vect)
{
    ADC_value = ADC;
    if ( (((long)ADC_value*16800/1023)) < 14000) //discharged battery
    {
        while(1){
            start = false;
            lcd_gotoxy(0,0);
            lcd_puts("  !!!POZOR!!! ");
            lcd_gotoxy(0,1);
            lcd_puts(" BATERIE VYBITA ");
            BUZZER_ON; //Zapnutí klaksonu
            _delay_ms(1000); //Čekej sekundu
            BUZZER_OFF; //Vypnutí klaksonu
            _delay_ms(1000); //Čekej sekundu
        } else BUZZER_OFF;
    }
}
```

Celá tato funkce je vložena do rutiny programu obsluhující tzv. přerušení. Pokud skončí převod A/D převodníku, dojde k nastavení příslušného bitu v řídicím registru a dojde k vyvolání přerušení pro vektor přerušení `ADC_vect`. Zjednodušeně řečeno, pokud dojde k ukončení převodu a zapsání hodnoty napětí do příslušného registru, procesor přeruší jakoukoliv činnost, skočí do funkce s názvem `ISR(ADC_vect)` a vykoná instrukce obsažené v této funkci. V našem případě se jedná o kontrolu, zda je akumulátor vybit nebo ne.

Pro pochopení principu A/D převodníku bylo řečeno vše. Pro hlídání připojení děliče ke kanálu A/D převodníku byla do uvedené funkce také přidána podmínka zastavení robotu při poklesu napětí pod 1V. V případě li-pol baterie to není možné, jelikož minimální dovolené napětí na akumulátoru je přibližně 13V. V mém případě hodnota 1V symbolizuje pouze odpojení děliče napětí, například při napájení kitu přes USB při jeho programování. Robot v tomto případě není zapnut.

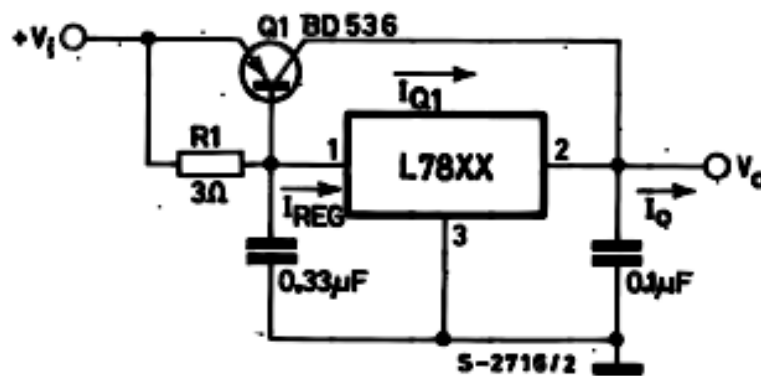


Obrázek 5-6- Stavů baterie - Nabito (vlevo nahoře), průběh vybíjení (nahore uprostřed), baterie téměř vybita (vpravo nahoře), baterie odpojena (vlevo dole), baterie vybita (vpravo dole)

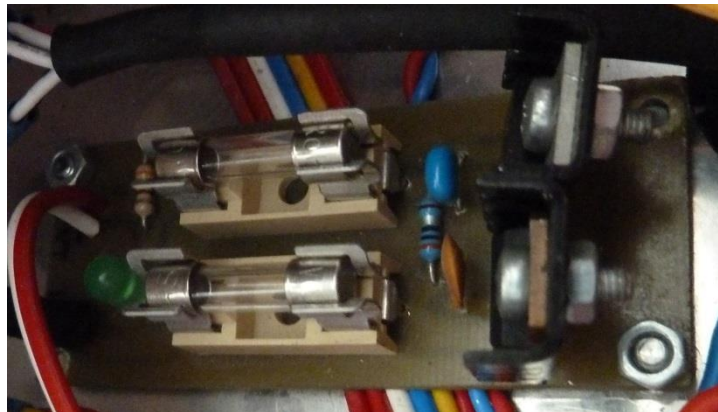
## 5.4 Stabilizační deska pro 9V

Pro napájení kitu EvB4.3 a Arduina bylo nutné zajistit nižší napájecí napětí. K těmto účelům je nejuvhodnější použít stabilizátor napětí. V mém případě jsem využil stabilizační desky vytvořené již dříve pro předchozí projekty. Součástí této destičky je běžně dostupný stabilizátor napětí 7805. V zapojení s bipolárním tranzistorem BD536 nebo jeho možnou náhradou BD912, je schopen dodávat vyšší proud do připojeného zařízení. Na této desce jsou také umístěny ochranné pojistky pro případ zkratu.

Schéma zapojení je uvedeno na obrázku [Obrázek 5-7], dostupné v datasheetu [29] pod názvem High Current Voltage Regulator. Na dalším obrázku [Obrázek 5-8] je možno vidět výslednou vyrobenou desku.



Obrázek 5-7 - Schéma zapojení stabilizátoru 7809 [29]



Obrázek 5-8 - Vyrobená a osazená DPS pro stabilizaci napětí

## 6. NÁVRH POUŽITÍ ENKODÉRŮ

### 6.1 Princip a typy enkodérů [30]

Základním principem funkce enkodérů je měření přítomnosti či nepřítomnosti nějakého dobře detekovatelného materiálu na otáčejícím se kolečku. V případě tachometru pro jízdní kolo je to magnet, v případě kuličkové myši je to díra v kolečku atd.

V robotické praxi se nejčastěji setkáváme s enkodéry optickými. V případě *reflexních* (odrazových) enkodérů jsou zdroj „světla“ i jeho přijímač společně umístěny na jedné straně kolečka, na kterém jsou reflexní a matné plošky. Jejich střídání před přijímačem při otáčení kolečka na něm způsobuje změny napětí v důsledku změny množství detekovaného světla. Výstupem je obdélníkový *signál*, kde každý obdélník odpovídá jedné reflexní či matné ploše. Při jejich rovnoměrném rozmístění po obvodu kolečka odpovídá jeden obdélník otočení vždy o shodný počet stupňů. Enkodérům generující takovýto pravidelný obdélníkový signál říkáme *inkrementální*. U *transmisivních* enkodérů jsou zdroj a přijímač umístěny na opačných stranách děravého kolečka. Reflexní plošky jsou zde suplovány dírami.

V případě znalosti, na kterou stranu se kolečko otáčí, z výše uvedených informací poměrně snadno získáme ujetou vzdálenost. Zkalibrovat enkodéry můžeme například tak, že necháme robota kousek popojet, přesně změříme ujetou vzdálenost, spočítáme počet „obdélníků“ naměřených během tohoto pohybu a změřenou vzdálenost vydělíme tímto počtem. Získáme tím „délku“ jednoho „obdélníku“ (často se tomuto údaji říká *enkodérový tik*).

V případě, že informaci o směru otáčení nemáme, z výše popsaného enkodéru ji nezískáme. Pro zjištění směru otáčení potřebujeme na kolečku ještě jeden senzor, který je v ideálním případě namontován tak, že jeho signál je s původním signálem fázově posunut a to nejlépe o  $90^\circ$ . Fázový posuv znamená, že se nám hrany obdélníků jednotlivých signálů neslévají (signály jsou posunuty v čase). Posuv o  $90^\circ$  znamená, že hrana obdélníku jednoho signálu nastává uprostřed obdélníku signálu druhého.

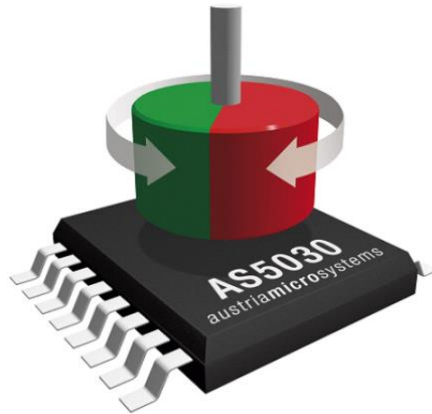
Mimo výše popsaných *inkrementálních* enkodérů existují i tzv. *absolutní*. Zatímco inkrementální enkodéry pouze detekují změnu polohy o pevně daný minimální krok, absolutní enkodéry nám říkají přímo polohu v rámci 360 stupňů.

Další, běžnější variantou, je použití potenciometru. Poloha se pak jednoduše určí pomocí A/D převodníku. Toto řešení je například použito snad ve všech levných modelářských servech.

## 6.2 Popis vybraného čipu

Po dohodě s panem Benešem z ústavu Automatizace byly objednány 3 vzorky absolutních enkodérů firmy Austria MicroSystems. Vzorky [31] je možné objednat pro své účely zcela zdarma.

Jedná se tedy o absolutní bezkontaktní magnetický enkodér [Obrázek 6-1]. Před enkodérem je tedy nutno vložit diametrálně (kolmo na osu) pólovaný magnet a při jeho otáčení dojde ke generaci, či změně PWM signálu. Možnost zvolení typu výstupu nám zajišťuje množství možných zapojení, která budou uvedena dále.



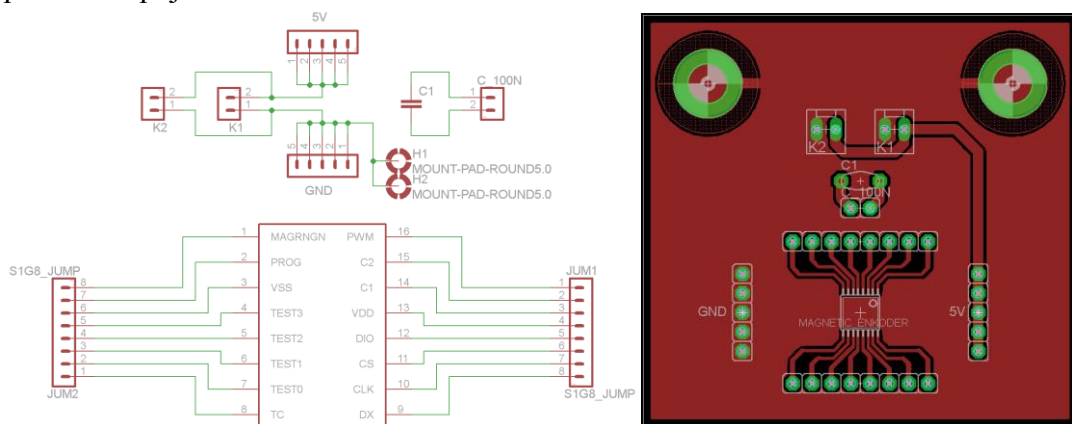
Obrázek 6-1 - Absolutní bezkontaktní enkodér AS5030 s diametrálně pólovaným magnetem [31]

Pomocí jeho výstupu můžeme přesněji měřit otočení kol o určitý úhel a tedy měřit vzdálenost.

## 6.3 Popis navržené DPS pro umístění čipu

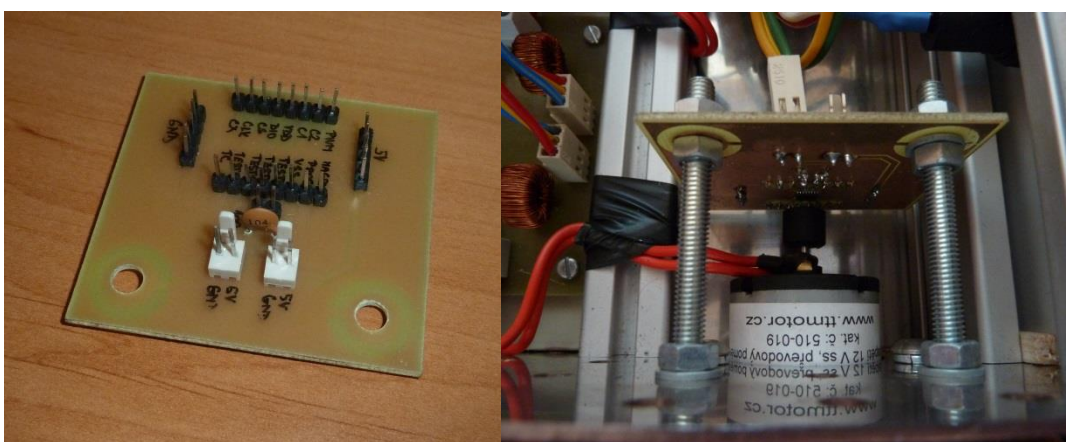
Na samotný čip by nebylo jednoduché například napájet tenké drátky, jelikož jeho nožičky jsou téměř miniaturních rozměrů. Proto bylo jednodušší navrhnout univerzální desku plošného spoje, kam byl čip umístěn a kde bude možné měnit jeho zapojení.

Na obrázku níže [Obrázek 6-2] je uvedeno schéma a k němu odpovídající deska plošného spoje.



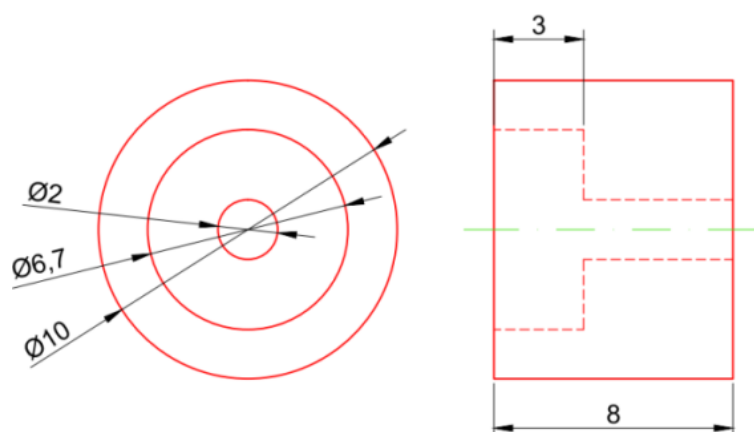
Obrázek 6-2 - Schéma DPS pro enkodérový čip (vlevo) a navržená DPS k výrobě (vpravo)

Z čipu jsou pomocí oboustranných kolíků vyvedeny jednotlivé piny. Dále je na desce umístěn kondenzátor o kapacitě 100nF. Hodnota je uvedena v datasheetu enkodéru, dostupném na stránkách výrobce [31]. Napětí potřebné k napájení +5V je přivedeno z H-můstku na konektor se zámkem K1 či K2. Na desce jsou ty konektory dva kvůli možnému napájení další enkodérové desky. Tohoto bylo využito při zapojení dvou desek, jedna pro pravou stranu robotu a druhá pro levou, jelikož jsou motory každé strany sesynchronizovány. Dále na desce najdeme dva montážní otvory spojené s nulovým potenciálem sloužící pro upevňovací šrouby spojené s podvozkem robotu. Více nám ukáže fotka její hotové podoby a výsledné umístění vyrobené desky v robotu [Obrázek 6-3].



Obrázek 6-3 - Vyrobený a osazená DPS s AS5030 na spodní straně (vlevo) a její umístění v robotu (vpravo)

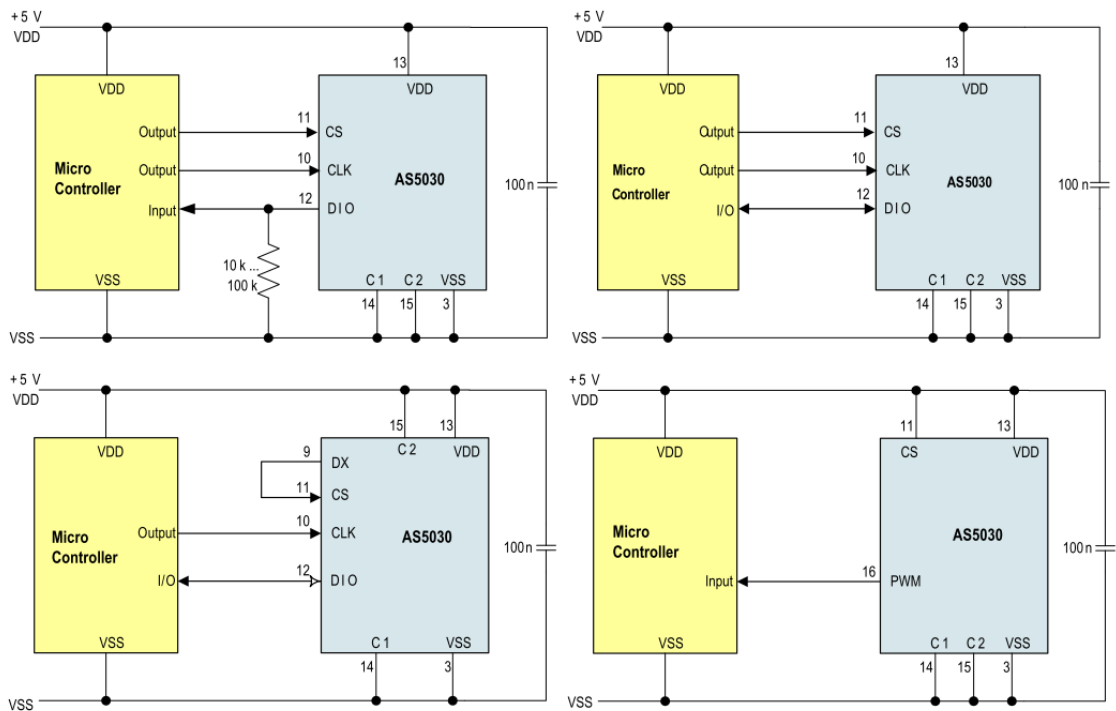
Z obrázku je také vidět, že bylo nutné vymyslet uchopení magnetu umístěného před čip. Na zadní části motoru je vyvedena hřídelka otáčející se směrem a rychlostí pohybu kola. Nejjednodušším způsobem bylo tedy navrhnout a vysoustružit pouzdro magnetu nasunuté na tuto hřídelku. Pro představu uvádím výkres pouzdra i s rozměry [Obrázek 6-4].



Obrázek 6-4 - Nákres pouzdra k umístění magnetu na otáčející se hřídelku motoru

## 6.4 Možnosti zapojení uvedeného čipu

V této podkapitole budou naznačeny možná zapojení čipu [Obrázek 6-5]. Zatím se mi nepodařilo zjistit, které ze zapojení bude pro mou aplikaci nejvýhodnější. Funkčnost enkodérů a návrh regulátoru zajišťující zvyšování otáček při zatížení robotu bude navrhnout v budoucnu. Každé ze zapojení má své specifické vlastnosti zjistitelné opět z datasheetu součástky uvedeného na webových stránkách výrobce [31].

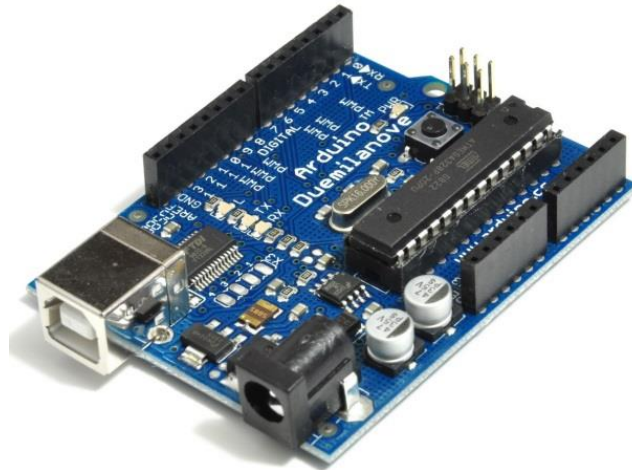


Obrázek 6-5 - Možnosti zapojení čipu AS5030 [31]

## 6.5 Popis mikrokontroléru na platformě ARDUINO

Čtení dat z čipu je možné více druhy mikrokontrolérů. Z důvodu nedostatku pinů a funkcí na vývojovém kitu EvB4.3 jsem se rozhodnul využít dalšího vývojového prostředku, se kterým jsem pracoval již dříve. Jak již napovídá název kapitoly, jedná se o vývojový prostředek Arduino. Lze sehnat více typů tohoto zařízení, s odlišnými vlastnostmi a rozměry. Já si již dříve pro základy programování pořídil Arduino Duemilanove [Obrázek 6-6]. Jedná se o podobný prostředek jako kit EvB4.3 jen obsahuje méně periférií a vyžaduje méně znalostí z oblasti programování.

Výhodou tohoto prostředku je jeho jednoduchost a dostupnost množství materiálů a návodů na internetu. Bližší informace o jeho možnostech, ukázky zdrojových kódů a zapojení je možné najít na domovské stránce výrobku [32].



Obrázek 6-6 - Arduino Duemilanove [33]

## 6.6 Programová část mikrokontroléru

V této části uvedu stručný výpis zdrojového kódu z Arduina. Jedná se o testovací program, ve kterém je pomocí stisku dvou tlačítek připojených k pinům 2 a 3 symbolizován signál z enkodérů. Při stisku tlačítka dojde v podstatě k sepnutí vstupu a tedy pomyslné skokové změně signálu, tedy PWM.

Chceme-li, aby mikrokontrolér reagoval na tato sepnutí kdykoliv za běhu programu je nutno připojit tlačítka na vstupy externího přerušení, které bylo již vysvětleno. V obsluze externího přerušení, funkce `encoder1` a `encoder2`, dojde k čítání pulsů. Pomocí proměnných `value1` a `value2` by byly navrženy regulátory. Jak již bylo zmíněno, správný zápis nastavení přerušení v mikrokontroléru je uveden na oficiálních stránkách projektu Arduino [32], v sekci Reference.

Zde je uvedena ukázka kódu, v komentářích je uvedena funkčnost každého příkazu.

```
#include <avr/io.h>           //Vložení knihovny pro vstupy/vystupy
#include <avr/interrupt.h>    //Knihovna pro preruseni

int left_encoder_pin = 2;    //Signal z levého enkoderu priveden na pin 2
int right_encoder_pin = 3;   //Signal z praveho enkoderu priveden na pin 3
volatile int value1 = 0;     //Počet pulsu z leveho enkoderu vynulovan
volatile int value2 = 0;     //Počet pulsu z praveho enkoderu vynulovan

void setup()
{
    //Interrupt0 na pinu 2, obsluhou je funkce encoder1, RISING =
    //inkrementace promenne value1 pri nabezne hrane
    attachInterrupt(0, encoder1, RISING);
    //Interrupt1 na pinu 3, obsluhou je funkce encoder2, RISING =
    //inkrementace promenne value1 pri nabezne hrane
    attachInterrupt(1, encoder2, RISING);
    Serial.begin(19200);     //Start seriove komunikace pri prijem dat do pc
}
```

```

void loop()                //Nekonecna smycka
{
  interrupts();           //Povoleni preruseni
  while(1);               //Nekonecna smycka
}
//Pri stisku prvnioho tlačítka dojde k vykonani instrukci v této funkci
void encoder1(void)
{
  value1++;               //Inkrementace poctu pulsu leveho enkoderu
  if(Serial.available())  //Pokud je seriova komunikace zahajena
  {
    Serial.println("LEFT:"); //Vypis do terminalu
    Serial.print(value1);    //Vypis hodnotu pulsu
  }
}
//Pri stisku druheho tlačítka dojde k vykonani instrukci v této funkci
void encoder2(void)
{
  value2++;               //Inkrementace poctu pulsu praveho enkoderu
  if(Serial.available())  //Pokud je seriova komunikace zahajena
  {
    Serial.println("RIGHT:"); //Vypis do terminalu
    Serial.print(value2);    //Vypis hodnotu pulsu
  }
}

```

Jedná se o nástin možného řešení. Z praxe mi vyplynulo, že výsledné řešení může být odlišné než navrhnuté. V procesoru obsaženém v kitu EvB4.3 je programování podobné. Při zapojení méně přídatných komponent by bylo možné tento princip využít.

Dalším nutným požadavkem pro čip enkodéru je vygenerování signálu systémových hodin, obdoba PWM signálu, jen s rozdílem konstantní šířky pulzu. Tento signál je nutno vygenerovat pomocí čítače/časovače a přivést na jeden z pinů čipu AS5030. Pro dva enkodéry, pravou i levou stranu je nutno tyto signály vygenerovat dva. Pro tuto potřebu je tedy nutné mít v procesoru dostupné dva čítače/časovače. Bohužel v kitu EvB4.3 jsem jeden použil na generaci PWM signálu pro řízení otáček motorů. Výstupy dalších dvou jsou obsazeny pro jiné zařízení robotu. Proto jsem se rozhodnul navrhnout řešení v podobě propojení obou čipů s Arduinem a odesláním signálu z Arduina do kitu EvB4.3 pomocí SPI sběrnice, která bude popsána v následující podkapitole.

### 6.6.1 Popis SPI sběrnice pro přenos mezi AVR a ARDUINEM

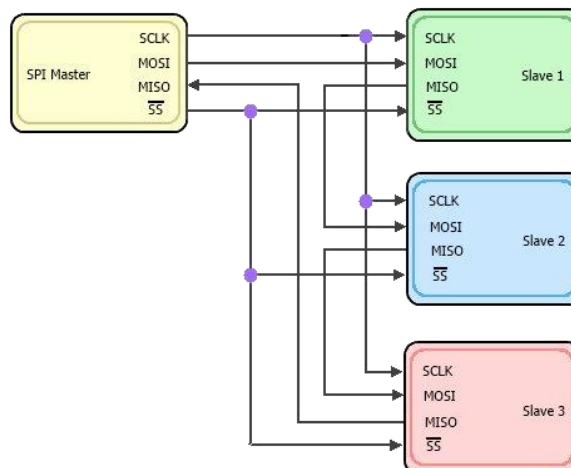
V mnoha zařízeních využívajících více mikrokontrolérů je nutná jejich vzájemná komunikace. Možností je spousta, například využití komunikačních sběrnic UART, I<sup>2</sup>C, SPI nebo jednoduše pomocí spínání V/V pinů. Pro složitější datové přenosy je výhodné využít sběrnice. Pokud je naším cílem pouze rozsvícení LED diody, postačí nám V/V piny mikroprocesoru. V této podkapitole bude zmíněna SPI sběrnice. Informace byly vybrány z článku na serveru Wikipedie [34].

**SPI (Serial Peripheral Interface)**, neboli sériové periferní rozhraní se využívá ke komunikaci mezi řídicími mikroprocesory a ostatními mikroprocesory. Komunikaci zajišťuje společná sběrnice. Adresace se provádí pomocí zvláštních vodičů, které při logické nule aktivují příjem a vysílání zvoleného zařízení. Zajišťují piny SS (Slave Select) nebo CS (Chip Select).

Před započítím komunikace je nutno určit typ zařízení připojeného k této sběrnici. Může se jednat buď o **Master** nebo **Slave**.

- **Master**
  - řídí komunikaci pomocí hodinového signálu
  - pomocí SS signálu (někdy CS) určuje, se kterým zařízením na sběrnici bude komunikovat
- **Slave**
  - pokud je aktivován pomocí SS nebo CS, vysílá data podle hodinového signálu

Schéma zapojení je uvedeno na následujícím jednoduchém schématu [Obrázek 6-7]. Při komunikaci pouze dvou zařízení nebudou připojena Slave zařízení s označeními 2 a 3.



Obrázek 6-7 - Schéma zapojení mikrokontrolérů pomocí SPI sběrnice [35]

Průběh komunikace:

- Pro zahájení komunikace nastaví Master logickou 0 na SS vstupu zařízení, se kterým chce komunikovat.
- Dojde ke generaci hodinového signálu a k odeslání dat,
  - MISO(Master In a Slave Out) – Master je vstup, Slave je výstup,
  - MOSI(Master Out a Slave In) – Master je výstup, Slave je vstup.
- Jakmile jsou data odeslána, mohou nastat dva případy,
  - komunikace pokračuje = generování hodinového signálu, hodnota SS se nemění,

- komunikace je ukončena = Master přestane vysílat hodinový signál a nastaví SS do logické 0.
- Délka vyslaných dat je buď 8bitů = 1byte nebo 16bitů =Word.  
Vztah mezi hodinovým signálem a daty se určuje dvěma konfiguračními bity, které se v původní dokumentaci nazývají **CPOL** a **CPHA**. Nastavení hodnot těchto bitů možno zjistit z katalogových listů uvedených zařízení.

Pro komunikaci mezi dvěma zařízeními je vhodné vymyslet vlastní zprávu podle svých potřeb. Pro představu bude uvedena funkce pro odeslání zprávy z Arduina do AVR procesoru.

```
void send_message_to_AVR(void)
{
    // Nastavení hodnoty na pinu SS do logické 0 pro volbu zarizeni
    digitalWrite(SS, LOW); // SS is pin 10
    // Poslani rezerze znak po znaku pres SPI i seriovou linku
    for (message; c = *message; message++)
    {
        SPI.transfer(c); //Transfer message char by char to SPI
        Serial.print(c); //Transfer message char by char to Serial Monitor
    }

    // Po skonceni komunikace dojde k nastaveni SS na logickou 1
    digitalWrite(SS, HIGH);

    delay(1000); // Cekej 1s
}
```

Zprávu uloženou do proměnné `message` bylo nutné vymyslet a definovat na začátku programu. Pro svůj účel byl vymyšlen řetězec znaků `L:0000 R:0000\n`.

L a R značí levou a pravou stranu robotu. První ze 4 nul představuje pohyb kol každé strany dopředu nebo dozadu. Tento směr bude určen proximitním skenerem popsaným v další kapitole. Při nutnosti pohybu dopředu bude na místo nuly zapsán znak `+`. Analogicky při pohybu dozadu znak `-`. Znak `\n` značí konec zprávy.

Další tři číslice jsou určeny pro odečítanou hodnotu z enkodérů. Rozlišení čipu AS5030 je 10bitů,  $2^{10}$  je tedy 255, což je maximální hodnota z čipu.

Tento řetězec znaků je tedy Arduinem odeslán do AVR mikrokontroléru umístěném na kitu EvB4.3. Po přijetí znaků je tedy možné navrhnout pomocí aktuálně nastavené hodnoty PWM signálu k pohybu motorů a přijaté hodnotě z enkodérů regulátor otáček motorů.

## 7. KOMUNIKACE S MIKROKONTROLÉREM OBSLUHUJÍCÍ PROXIMITNÍ SKENER

Dalším úkolem robotu je pohyb středem koridoru za pomoci laserového dálkoměru, o jehož obsluhu se stará další mikrokontrolér, nyní značky ARM umístěný na dalším vývojovém kitu. Tato úloha je cílem bakalářské práce mého týmového kolegy, Vítka Přecechtěla. Výhodou této práce je z praktického hlediska nezávislost těchto dvou témat. Každý z nás má svou část, na které může pracovat. Společnou částí je pouze jednoduchá komunikace, která byla předem vymyšlena a bude probrána v této kapitole.

### 7.1 Popis vstupně výstupních pinů mikrokontroléru

Snad ve většině užívaných mikrokontrolérů jsou obsaženy vstupně výstupní piny. Piny mohou být spínány buď pozitivní nebo negativní logikou. V případě pozitivní logiky je programem mikrokontroléru čteno 0V jako logická 0. Při přivedení +5V na tento pin je hodnota čtena jako logická 1. V případě negativní logiky je tomu naopak.

Tento problém je ovšem ve většině mikrokontrolérů ošetřen připojením tzv. pull-up rezistoru. Při připojení tohoto rezistoru dojde k záměně negativní logiky za pozitivní. Uvedu jednoduchý příklad. V kitu EvB4.3 jsou tlačítka připojena na signál GND. Při čtení v mikrokontroléru bychom však chtěli číst hodnotu při připojení k +5V. Například jednoduchým zápisem `DDRA |= (0 << PA1)` nastavíme pin A1 jako vstupní. Pull-up rezistor připojíme zápisem `PORTA |= (1 << PA1)`. Nyní když budeme chtít číst hodnotu z pinu, můžeme využít funkce `bit_is_set(PORTA, PA1)`, která při hodnotě +5V na vstupu vrací hodnotu 1, v opačném případě 0. Pokud by nebyl připojen pull-up rezistor využívala by se funkce `bit_is_reset` se stejnými vstupními hodnotami. Rozdíl by byl pouze v návratových opačných hodnotách. Nastavení hodnot v registrech by mělo být uvedeno v katalogových listech či návodech pro jednotlivá zařízení či mikrokontroléry.

Při použití jednoho tlačítka se problém zdá téměř zanedbatelný. Ovšem při použití více tlačítek a složitějších systémů je vhodné tuto problematiku znát. Jinak může docházet ke zbytečným chybám při programování.

### 7.2 Popis smluvené komunikace

Pro komunikaci ARM mikrokontroléru s Arduinem budou použity pouze 4 vodiče připojené na 4 piny každého z procesorů. Každý z vodičů bude symbolizovat jeden pohyb robotu. Pohyb dopředu, pohyb dozadu, otáčení doleva a otáčení doprava. Při sepnutí některého z pinů dojde také v Arduinu ke změně odesílané zprávy `L:0000 R:0000\n` do mikrokontroléru AVR na kitu EvB4.3. Pokud bude dálkoměr požadovat

například pohyb dopředu, zpráva bude změněna na `L:+000 R:+000\n`, značící pohyb obou stran robotu kladným směrem, tedy dopředu. Pro zbývající pohyby bude změna této zprávy analogická.

Jelikož dálkoměr dokáže číst různé vzdálenosti od robotu a jeho řídicí ARM procesor je výkonnější než AVR procesor obsažený v kitu či Arduinu, budou veškeré výpočty ujeté vzdálenosti a možné trajektorie počítány právě v něm.

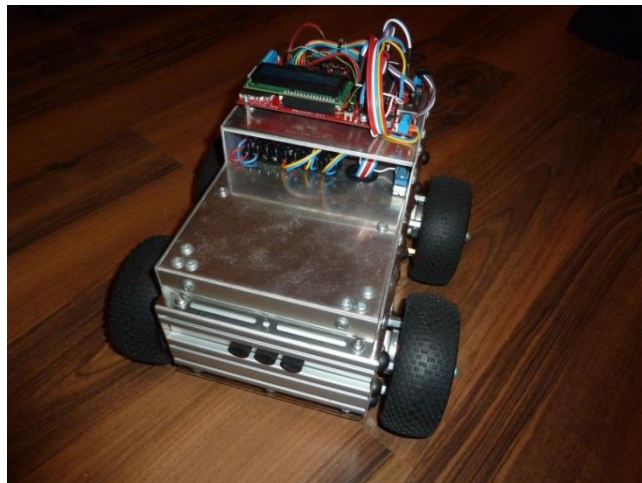
### 7.3 Ukázka zdrojového kódu mikrokontroléru

V této podkapitole je uveden zdrojový kód z Arduina zajišťující změnu zprávy vysílané do kitu EvB4.3 pomocí SPI sběrnice. Pro testování byla do podmínek zajišťujících změnu znaků ve zprávě přidána možnost simulace sepnutí pinů ARM mikrokontrolérem odesláním písmen přes terminál sériové linky programu pro obsluhu Arduina.

```
void loop (void)
{
  movement = Serial.read();          //Cteni znaku z terminalu pocitace
  state_forward = digitalRead(forward_pin);      //Pri sepnuti pinu dojde k
  state_backward = digitalRead(backward_pin);    //nastaveni stavu
  state_left = digitalRead(turn_left_pin);
  state_right = digitalRead(turn_right_pin);
  //Zmena paramaetru zpravy
  if((state_forward == LOW) | (movement == 'a')) //Pohyb dopredu
  {
    message[2] = '+';          //left side forward
    message[9] = '+';          //right side forward
  }
  else if(state_backward == LOW | (movement == 'b')) //Pohyb dozadu
  {
    message[2] = '-';          //left side backward
    message[9] = '-';          //right side backward
  }
  else if(state_left == LOW | (movement == 'c')) //Otoceni doleva
  {
    message[2] = '-';          //left side backward
    message[9] = '+';          //right side forward
  }
  else if(state_right == LOW | (movement == 'd')) //Otoceni doprava
  {
    message[2] = '+';          //left side forward
    message[9] = '-';          //right side backward
  }
  else //Stop
  {
    message[2] = '0';          //left side stop
    message[9] = '0';          //right side stop
  }
} //Konec nekonecne smycky
```

## 8. FINÁLNÍ PODOBA ROBOTU

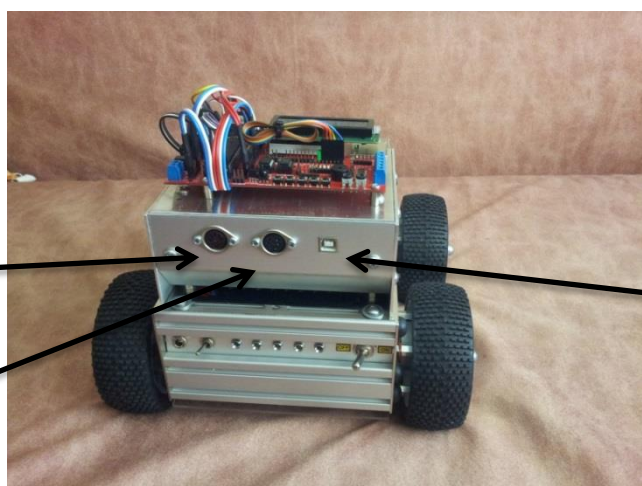
### 8.1 Fotky robotu



Obrázek 8-1 - Pohled na robot ze přední strany

Konektor  
balancéru

Konektor  
manuálního  
ovládání

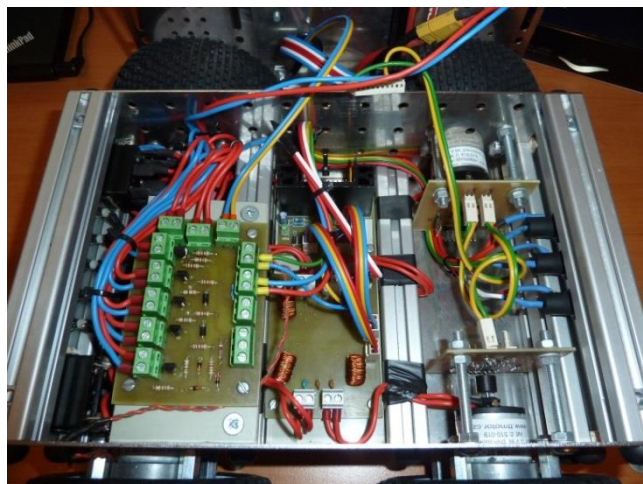


Programovací  
konektor  
Arduina

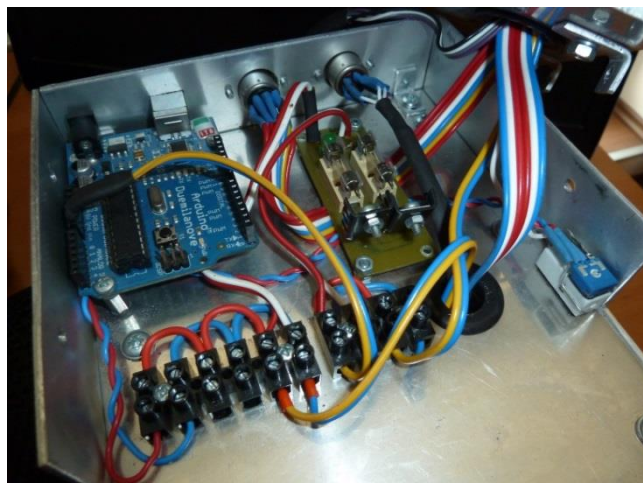
Obrázek 8-2 - Pohled na robot ze zadní strany s popisky konektorů



Obrázek 8-3 - Spodní strana robotu, obohacená o LED pásek



Obrázek 8-4 - Vnitřní elektronika robotu je pomyslně rozdělena na tři části. Základová deska s indikací vlevo. H-můstek s deskou pro paralelní zapojení uprostřed. Enkodéry s pojistkami vpravo



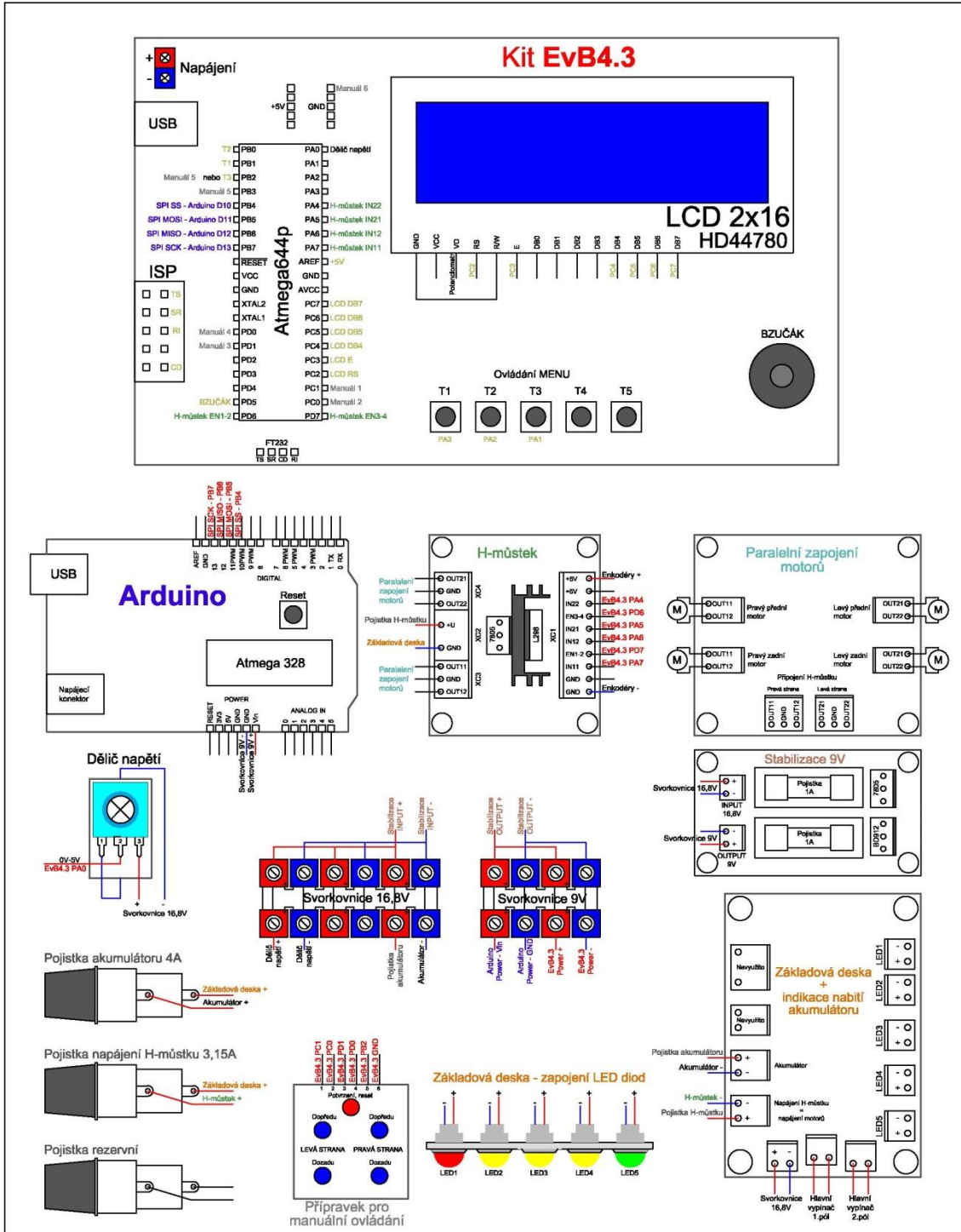
Obrázek 8-5 - Elektronika horní paluby - Arduino, stabilizační deska a svorkovnice pro rozvod napětí



Obrázek 8-6 - Nabíjení robotu - pro spojení balanční destičky s robotem byl vyroben propojovací kabel

## 8.2 Celkové blokové schéma robotu

Na následujícím obrázku [Obrázek 8-7] je uvedeno celkové schéma zapojení součástí robotu. Propojovací drátky či kabely jsou zde symbolizovány textem, kam příslušný vodič vede. Barevné odlišení nadpisů součástí nám pomůže k jednodušší orientaci ve schématu. Detailnější obrázek i lze nalézt na příloženém disku.



Obrázek 8-7 - Celkové blokové schéma zapojení robotu vytvořené v programu AutoCAD

## 9. ZÁVĚR

V této práci byl navržen a zkonstruován mobilní robot schopný autonomního i ovládaného pohybu. Bylo využito znalostí z předchozího projektu předmětu Základy robotiky, ve kterém jsem načerpal znalosti v oblasti řízení stejnosměrných motorů mikrokontrolérem. Robot se podařilo zkonstruovat a je plně funkční.

Uspadněním manuální práce a shánění dalších nutných věcí pro robot byl vývojový prostředek EvB4.3. Díky jeho periferiím bylo možné robot snadněji ovládat a nebylo nutné například vymýšlet uchopení spínačů a tlačítek pro průchod menu a spuštění hlavního programu robotu.

Velkým přínosem v mé práci byl LCD zobrazovač umístěný na kitu EvB4.3. Díky tomuto zařízení mohlo být vytvořeno jednoduché menu, pomocí něhož může uživatel snadněji komunikovat s robotem. V případě problémů v programovací části bylo také možné zobrazit různé nápisy na LCD displeji.

Další nutností bylo seznámení se s principem analogově-digitálního převodníku, který robotu zajišťuje přesné hlídání napětí akumulátoru a nemůže dojít k jeho nadměrnému vybití a tím možné explozi akumulátoru i celého robotu. Při častějším používání robotu a tedy i většímu opotřebení akumulátoru by bylo vhodné využít existujícího mikrokontroléru, který je přímo určen ke hlídání nabití akumulátorů a zajišťuje ochranu proti nadměrnému vybití některého z článků.

K robotu je nyní také možno připojit manuální ovladač, díky kterému je možné s robotem jezdit přesně podle vlastních příkazů. V budoucnu bych chtěl robot obohatit o Wi-fi modul, kterým by bylo možné řídit robot dálkově přes PC či mobilní telefon. K robotu bych také chtěl připojit kameru, či mobilní telefon s kamerou. Díky těmto zařízením by bylo možné přenášet do PC obraz z okolí robotu. Vznikl by tak dálkově řízený robot průzkumník.

# LITERATURA

- [1] KOŠNAR, Karel. Skupina inteligentní a mobilní robotiky, Katedra kybernetiky, Fakulta elektrotechnická, České Vysoké Učení Technické v Praze. In: *Mobilní robotika* [online]. 2010 [cit. 2013-05-06]. Dostupné z: [http://pandatron.sk/?1745&mobilni\\_robotika](http://pandatron.sk/?1745&mobilni_robotika)
- [2] Šolc F., Žalud L., Robotika, Vysoké učení technické v Brně, 2006, Brno (cs)
- [3] *RobotShop: Robotics at your service!* [online]. 2013 [cit. 2013-05-06]. Dostupné z: <http://www.robotshop.com/content/images//100mm-left-mecanum-wheel-a.jpg>
- [4] Podvozek: Tank 03. *Shop: Snail Instruments* [online]. 2013 [cit. 2013-05-06]. Dostupné z: [http://shop.snailinstruments.com/index.php?main\\_page=product\\_info&cPath=77&%20products\\_id=380](http://shop.snailinstruments.com/index.php?main_page=product_info&cPath=77&%20products_id=380)
- [5] Vývojový kit EvB. *Www.kamnar.cz* [online]. [cit. 2013-05-06]. Dostupné z: <http://www.kamnar.cz/profikamna/vyvojovy-kit-evb.html>
- [6] *Návod na obsluhu vývojové desky EvB 4.3* [online]. 3. vyd. 2010 [cit. 2013-05-06]. Dostupné z: [http://shop.onpa.cz/download/EvB43\\_cz\\_manual\\_rev3.pdf](http://shop.onpa.cz/download/EvB43_cz_manual_rev3.pdf)
- [7] Akumulátory. *Akumulátory* [online]. 2004 [cit. 2013-05-06]. Dostupné z: [http://www.mo-na-ko.net/ruzne\\_acupack.htm](http://www.mo-na-ko.net/ruzne_acupack.htm)
- [8] Olověný akumulátor WP18-12, 12V. *GM Electronic* [online]. [cit. 2013-05-06]. Dostupné z: <http://www.gme.cz/pb-akumulatory/oloveny-akumulator-wp18-12-12v-18ah-kung-long-p540-054/>
- [9] Akumulátor GP 4500mAh D. *GM Electronic* [online]. [cit. 2013-05-06]. Dostupné z: <http://www.gme.cz/nicd-akumulatory/akumulator-gp-4500mah-d-p540-160/>
- [10] Akumulátor NEXCELL AAA NiMH 800mAh. *GM Electronic* [online]. [cit. 2013-05-06]. Dostupné z: <http://www.gme.cz/nimh-akumulatory/akumulator-nexcell-aaa-nimh-800mah-p540-198/>
- [11] Lumix Battery Pack. *Panasonic* [online]. [cit. 2013-05-06]. Dostupné z: [http://media.panasonic.eu/images/DMW-BCK7E\\_1WebA1001001A11A06B53314C29161.jpg](http://media.panasonic.eu/images/DMW-BCK7E_1WebA1001001A11A06B53314C29161.jpg)

- [12] G3 RAY LI-POL 350MAH/7,4 26/50C AIR PACK. In: *RCMarket Brno* [online]. 2010 [cit. 2013-05-06]. Dostupné z: <http://www.rcmarket.cz/zbozi/akumulatory-baterie/lipol/g3-ray-li-pol-350mah74-2650c-air-pack.html>
- [13] G3 RAY LI-POL 2700MAH/14,8 26/50C AIR PACK. In: *RC Market Brno* [online]. 2010 [cit. 2013-05-06]. Dostupné z: <http://www.rcmarket.cz/zbozi/akumulatory-baterie/lipol/g3-ray-li-pol-2700mah148-2650c-air-pack.html>
- [14] NOVÁK, Petr. Mobilní roboty - pohony, senzory, řízení. Praha : Nakladatelství BEN - technická literatura, 2005. 248 s. ISBN 80-7300-141-1.
- [15] DC Motor Operation. *HyperPhysics* [online]. Canada, 2000 [cit. 2013-05-06]. Dostupné z: <http://hyperphysics.phy-astr.gsu.edu/hbase/magnetic/motdc.html>
- [16] Understanding Eccentric Rotating Mass (ERM) Vibration Motor: Understanding ERM Vibration Motor Characteristics. *PRECISION MICRODRIVES* [online]. 2013 [cit. 2013-05-06]. Dostupné z: <http://www.precisionmicrodrives.com/application-notes-technical-guides/application-bulletins/ab-004-understanding-erm-characteristics-for-vibration-applications>
- [17] Stejnoseměrný motor GM37-110. *Shop - SnailInstruments* [online]. 2013 [cit. 2013-05-06]. Dostupné z: [http://shop.snailinstruments.com/index.php?main\\_page=product\\_info&cPath=75\\_148&products\\_id=655](http://shop.snailinstruments.com/index.php?main_page=product_info&cPath=75_148&products_id=655)
- [18] H.R. Everett, Sensors for Mobile Robots, A K Peters/CRC Press, 1995, ISBN 1568810482
- [19] Lab 9: Pulse Width Modulation (PWM) using PIC CCP module. In: *Embedded Lab: An online teaching laboratory for Microcontrollers and Embedded Systems* [online]. 2013 [cit. 2013-05-06]. Dostupné z: <http://embedded-lab.com/blog/?p=1443>
- [20] L293 QUADRUPLE HALF H-DRIVER: Datasheet. In: *Shop - GM Electronic* [online]. 1990 [cit. 2013-05-06]. Dostupné z: <http://www.gme.cz/dokumentace/399/399-017/dsh.399-017.1.pdf>

- [21] Bridga 2x2A. *Shop: SnailInstruments* [online]. 2013 [cit. 2013-05-06]. Dostupné z: [http://shop.snailinstruments.com/index.php?main\\_page=product\\_info&cPath=68\\_69&products\\_id=783](http://shop.snailinstruments.com/index.php?main_page=product_info&cPath=68_69&products_id=783)
- [22] Bridge20: Datasheet. *Dvojitý H-můstek k řízení stejnosměrných motorů* [online]. [cit. 2013-05-06]. Dostupné z: <http://www.rotta.cz/PDF/bridge20.pdf>
- [23] Atmega644: Datasheet. *Atmel Corporation* [online]. 2012 [cit. 2013-05-06]. Dostupné z: <http://www.atmel.com/Images/doc2593.pdf>
- [24] Interfacing a HD44780 Based LCD to an AVR. *Peter Fleury Online* [online]. 2003 [cit. 2013-05-06]. Dostupné z: <http://homepage.hispeed.ch/peterfleury/avr-lcd44780.html>
- [25] LCD44780 pin layout. *Www.blogspot.com* [online]. [cit. 2013-05-06]. Dostupné z: <http://1.bp.blogspot.com/-Tvl5cufcz4M/TtVFbCr9KtI/AAAAAAAAAUE/QfhyoMzgxGQ/s1600/16x2+char+lcd+pinout+4+bit+interface.png>
- [26] EQUILIBRIUM MINI V2 12/220V. *RCMarket Brno* [online]. 2010 [cit. 2013-05-06]. Dostupné z: <http://www.rcmarket.cz/zbozi/nabijeni/nabijece/equilibrium-mini-v2-12220v.html>
- [27] Indikátor napětí autobaterie. *Pandatron: Elektrotechnický magazín* [online]. 200-2013 [cit. 2013-05-06]. Dostupné z: [http://pandatron.cz/?108&ukazatel\\_napeti\\_autobaterie](http://pandatron.cz/?108&ukazatel_napeti_autobaterie)
- [28] Odporový dělič. *Ondřej Weisz: Domácí stránky* [online]. 2000 [cit. 2013-05-06]. Dostupné z: <http://web.telecom.cz/oweisz/rdelic.htm>
- [29] L7800 Series Voltage Regulators Datasheet [online]. [cit. 2011-11-23]. Dostupné na URL: <<http://www.gme.cz/dokumentace/330/330-149/dsh.330-149.1.pdf>>
- [30] Enkodéry. DLOUHÝ, Martin a Zbyněk WINKLER. *Robotika.cz* [online]. 2003 [cit. 2013-05-06]. Dostupné z: <http://robotika.cz/guide/encoders/en>
- [31] Automated Micro Systems. *Magnetic Position Sensors: AS5030 Rotary Position Sensor* [online]. 2013 [cit. 2013-05-06]. Dostupné z: <http://www.ams.com/eng/Products/Magnetic-Position-Sensors/Magnetic-Rotary-Position-Sensors/AS5030>

- [32] Arduino Duemilanove. *Oficiální stránky projektu* [online]. [cit. 2013-05-06]. Dostupné z: <http://arduino.cc/en/Main/ArduinoBoardDuemilanove>
- [33] Arduino Duemilanove. *Roboti pro zábavu* [online]. 2009 [cit. 2013-05-06]. Dostupné z: [http://www.roboti.4fan.cz/wp-content/uploads/Duemilanove\\_31.jpg](http://www.roboti.4fan.cz/wp-content/uploads/Duemilanove_31.jpg)
- [34] Serial Peripheral Interface. In: *Wikipedia: Otevřená encyklopedie* [online]. 2003 [cit. 2013-05-06]. Dostupné z: [http://cs.wikipedia.org/wiki/Serial\\_Peripheral\\_Interface](http://cs.wikipedia.org/wiki/Serial_Peripheral_Interface)
- [35] Schéma zapojení SPI sběrnice. *JaxCoder* [online]. 2008-2011 [cit. 2013-05-06]. Dostupné z: [http://jaxcoder.com/Pictures/Article%20Pics/spi\\_2.png](http://jaxcoder.com/Pictures/Article%20Pics/spi_2.png)
- [36] Podvozek MOB-03. *Snail Instruments: Obchod* [online]. 2013 [cit. 2013-05-06]. Dostupné z: [http://shop.snailinstruments.com/index.php?main\\_page=product\\_info&cPath=77&products\\_id=382](http://shop.snailinstruments.com/index.php?main_page=product_info&cPath=77&products_id=382)

# SEZNAM OBRÁZKŮ

<b>Obrázek 2-1</b>	- Nákres diferenciálního podvozku [2].....	<b>12</b>
<b>Obrázek 2-2</b>	- Nákres synchronního podvozku [2] .....	<b>12</b>
<b>Obrázek 2-3</b>	- Schéma trojkolového podvozku [2] .....	<b>13</b>
<b>Obrázek 2-4</b>	- Nákres Ackermannova podvozku [2].....	<b>13</b>
<b>Obrázek 2-5</b>	- Ukázka všesměrového kola Mecanum [3] .....	<b>14</b>
<b>Obrázek 2-6</b>	- Příklad pásového podvozku [4].....	<b>14</b>
<b>Obrázek 2-7</b>	- Zakoupený diferenciální podvozek [36] .....	<b>15</b>
<b>Obrázek 2-8</b>	- Vývojový kit EvB4.3 s drátovými propojkami [5] .....	<b>16</b>
<b>Obrázek 2-9</b>	- Pb akumulátor, 12V, 18Ah [8].....	<b>18</b>
<b>Obrázek 2-10</b>	- NiCd akumulátor, 1,2V, 4500mAh [9] .....	<b>19</b>
<b>Obrázek 2-11</b>	- NiMh akumulátor, 1,2V, 800mAh [10] .....	<b>19</b>
<b>Obrázek 2-12</b>	- Li-ion akumulátor, 3,6V, 680mAh [11] (vlevo) a li-pol akumulátor, 7,4V, 350mAh [12] (vpravo) .....	<b>20</b>
<b>Obrázek 2-13</b>	- Vybraný Li-Pol akumulátor, 16,8V, 2700mAh [13].....	<b>20</b>
<b>Obrázek 3-1</b>	- Příklad stejnosměrného motorku s převodovkou [14] .....	<b>22</b>
<b>Obrázek 3-2</b>	- Řez stejnosměrným motorem, princip činnosti [15].....	<b>23</b>
<b>Obrázek 3-3</b>	- Náhradní schéma stejnosměrného motoru [16].....	<b>23</b>
<b>Obrázek 3-4</b>	- Stejnosměrný motor s převodovkou GM37-110 [17] .....	<b>25</b>
<b>Obrázek 3-5</b>	- Jednokvadrantový proudový regulátor otáček [14] .....	<b>26</b>
<b>Obrázek 3-6</b>	- Průběh proudu při PWM [19] .....	<b>26</b>
<b>Obrázek 3-7</b>	- výkonová část můstkového proudového regulátoru otáček využívající signál PWM [14] .....	<b>27</b>
<b>Obrázek 3-8</b>	- Dvojitý H-můstek k řízení až 4 stejnosměrných motorků [21].....	<b>27</b>
<b>Obrázek 3-9</b>	- Schéma zapojení H-můstku [22].....	<b>28</b>
<b>Obrázek 3-10</b>	- Schéma DPS pro paralelní zapojení motorů .....	<b>29</b>
<b>Obrázek 3-11</b>	- DPS paralelního zapojení motorů .....	<b>30</b>
<b>Obrázek 3-12</b>	- Schéma zapojení manuálního ovladače .....	<b>33</b>
<b>Obrázek 3-13</b>	- Manuální ovládání pro robot.....	<b>33</b>
<b>Obrázek 4-1</b>	- Detail LCD displeje [6].....	<b>34</b>
<b>Obrázek 4-2</b>	- Nákres LCD displeje s vysvětlením pozic znaků [25].....	<b>35</b>
<b>Obrázek 4-3</b>	- Schéma kitu EvB4.3 s popisky zapojení LCD k PORTu C .....	<b>35</b>
<b>Obrázek 4-4</b>	- LCD: Ukázkový mód .....	<b>39</b>
<b>Obrázek 4-5</b>	- LCD: Manuální ovládání.....	<b>39</b>
<b>Obrázek 4-6</b>	- LCD: Scanner .....	<b>39</b>
<b>Obrázek 4-7</b>	- LCD: Enkodéry .....	<b>39</b>
<b>Obrázek 4-8</b>	- Indikace nabití baterie .....	<b>39</b>
<b>Obrázek 5-1</b>	- Mikroprocesorový nabíječ akumulátorů Equilibrium Mini s příslušenstvím, [26] .....	<b>40</b>
<b>Obrázek 5-2</b>	- Schéma zapojení DPS s indikací nabití akumulátoru, část s tranzistory převzata z [27].....	<b>41</b>

<b>Obrázek 5-3</b>	- Návrh DPS pro základovou desku s indikací nabití akumulátoru (vlevo), foto pásu LED diod na robotu (vpravo) .....	<b>41</b>
<b>Obrázek 5-4</b>	- Dělič napětí .....	<b>42</b>
<b>Obrázek 5-5</b>	- Obecné schéma děliče napětí [28].....	<b>43</b>
<b>Obrázek 5-6</b>	- Stavby baterie.....	<b>45</b>
<b>Obrázek 5-7</b>	- Schéma zapojení stabilizátoru 7809 [29] .....	<b>46</b>
<b>Obrázek 5-8</b>	- Vyrobená a osazená DPS pro stabilizaci napětí.....	<b>46</b>
<b>Obrázek 6-1</b>	- Absolutní bezkontaktní enkodér AS5030 s diametrálně pólovaným magnetem [31] .....	<b>48</b>
<b>Obrázek 6-2</b>	- Schéma DPS pro enkodérový čip (vlevo) a navržená DPS k výrobě (vpravo).....	<b>48</b>
<b>Obrázek 6-3</b>	- Vyrobený a osazená DPS s AS5030 na spodní straně (vlevo) a její umístění v robotu (vpravo).....	<b>49</b>
<b>Obrázek 6-4</b>	- Nákres pouzdra k umístění magnetu na otáčející se hřídelku motoru . .....	<b>49</b>
<b>Obrázek 6-5</b>	- Možnosti zapojení čipu AS5030 [31] .....	<b>50</b>
<b>Obrázek 6-6</b>	- Arduino Duemilanove [33] .....	<b>51</b>
<b>Obrázek 6-7</b>	- Schéma zapojení mikrokontrolérů pomocí SPI sběrnice [35].....	<b>53</b>
<b>Obrázek 8-1</b>	- Pohled na robot ze přední strany .....	<b>57</b>
<b>Obrázek 8-2</b>	- Pohled na robot ze zadní strany s popisky konektorů .....	<b>57</b>
<b>Obrázek 8-3</b>	- Spodní strana robotu, obohacená o LED pásek .....	<b>57</b>
<b>Obrázek 8-4</b>	- Vnitřní elektronika robotu.....	<b>58</b>
<b>Obrázek 8-5</b>	- Elektronika horní paluby.....	<b>58</b>
<b>Obrázek 8-6</b>	- Nabíjení robotu.....	<b>58</b>
<b>Obrázek 8-7</b>	- Celkové blokové schéma zapojení robotu vytvořené v programu AutoCAD .....	<b>59</b>

# SEZNAM SYMBOLŮ, JEDNOTEK A ZKRATEK

Ah	Ampér hodina
mAh	mili Ampér hodina
mA	mili Ampér
kHz	kilo Hertz
nF	nano Farad
LED	Light Emitting Diode
LCD	Liquid Crystal Display
ARM	Advanced RISC Machine
GPS	Global Positioning System
AVR	Alf, Vegard, RISC procesor
A/D	Analog to Digital
ISP	Serial Peripheral Interface
USB	Universal Serial Bus
PWM	Pulse Width Modulation
DC	Direct Current
SS	Slave Select
CS	Chip Select
GND	Ground
V/V	Vstupně/Výstupní
Pb	Olovo
NiCd	Nikl kadmium
NiMh	Nikl metal hydrid
Li-ion	Lithium ion
Li-pol	Lithium polymer

# SEZNAM PŘÍLOH

## NÁVRH HARDWAROVÉ ČÁSTI PRO ROBOT

<b>A. Základová deska s indikací nabití akumulátoru.....</b>	<b>69</b>
1. Deska plošného spoje – bottom (strana spojů).....	69
2. Osazovací výkres DPS – top (strana součástek).....	69
<b>B. Deska pro paralelní zapojení motorů.....</b>	<b>70</b>
1. Deska plošného spoje – bottom (strana spojů).....	70
2. Osazovací výkres DPS – top (strana součástek).....	70
<b>C. Enkodéry.....</b>	<b>71</b>
1. Deska plošného spoje – bottom (strana spojů).....	71
2. Osazovací výkres DPS – top (strana součástek).....	71
3. Osazovací výkres DPS – bottom (strana spojů).....	71

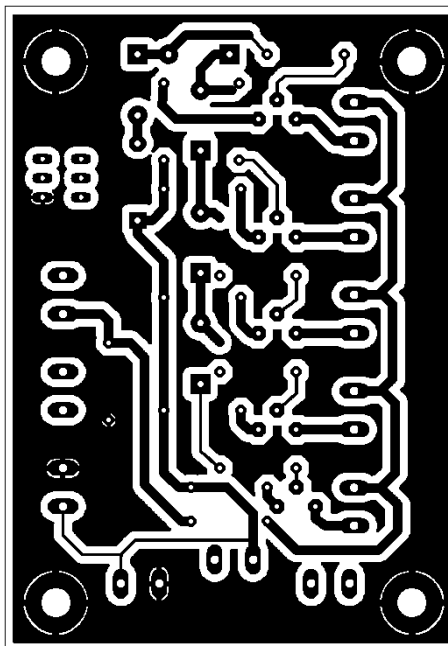
### Přílohy na DVD:

1. Elektronická verze této bakalářské práce
2. Projekt v programu AVR Studio pro mikrokontrolér Atmega644p
3. Projekt v programu Arduino pro Arduino Duemilanove
4. Projekt v programu AutoCAD – blokové schéma zapojení robotu
5. Projekty v programu Eagle
  - a. Dálkové ovládání
  - b. Enkodéry
  - c. Napěťový dělič
  - d. Základová deska s indikací nabití akumulátoru
  - e. Zapojení motorů k H-můstku
6. Fotky a obrázky schémata robotu
7. Datasheety a manuály
  - a. Absolutní magnetický enkodér AS5030
  - b. AVR Atmega644p
  - c. H-můstek
  - d. Obvod L298
  - e. Stabilizátory série 7800

# NÁVRH HARDWAROVÉ ČÁSTI PRO ROBOT

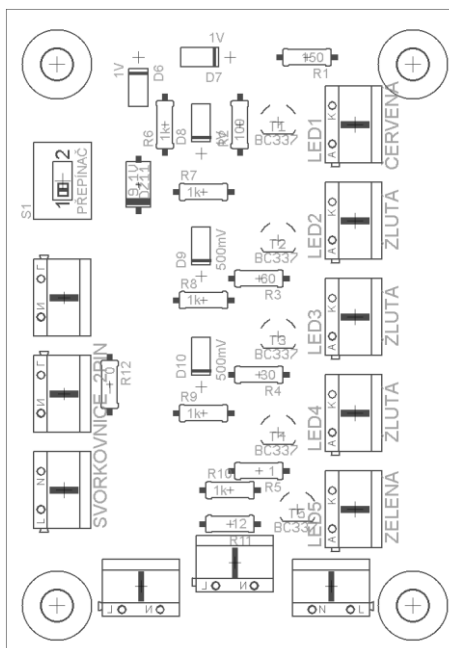
## A. Základová deska s indikací nabití akumulátoru

### 1. Deska plošného spoje – bottom (strana spojů)



Rozměry 60x80mm, měřítko M1:1

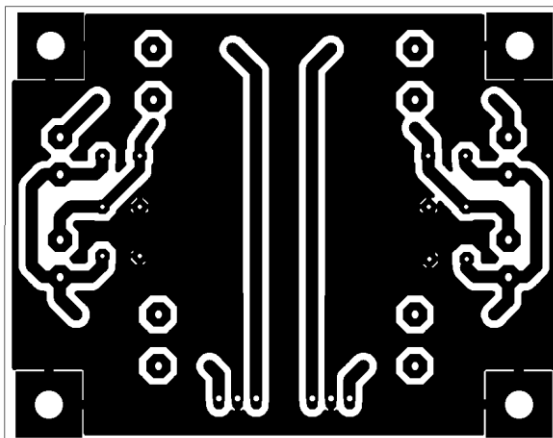
### 2. Osazovací výkres DPS – top (strana součástek)



Rozměry 60x80mm, měřítko M1:1

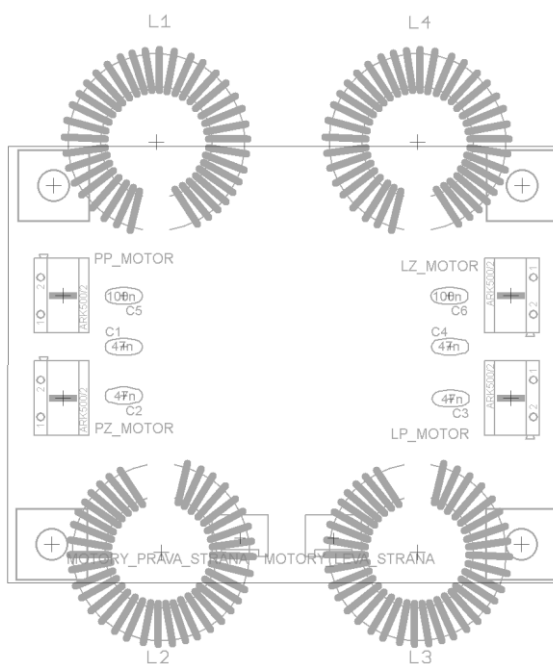
## B. Deska pro paralelní zapojení motorů

### 1. Deska plošného spoje – bottom (strana spojů)



Rozměry 75x56mm, měřítko M1:1

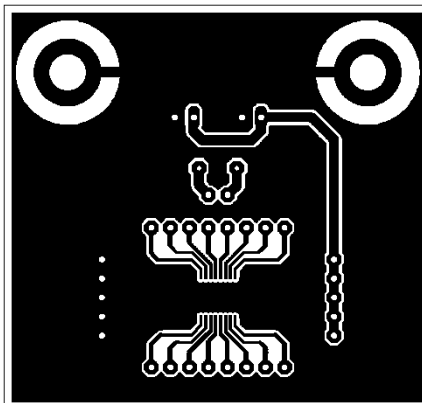
### 2. Osazovací výkres DPS – top (strana součástek)



Rozměry 75x56mm, měřítko M1:1

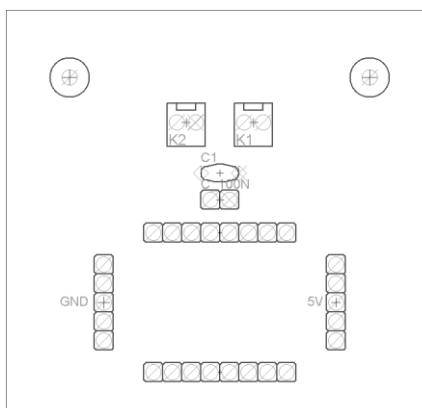
## C. Enkodéry

### 1. Deska plošného spoje – bottom (strana spojů)



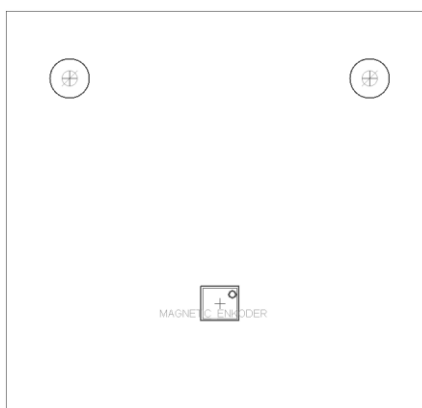
Rozměry 57x54mm, měřítko M1:1

### 2. Osazovací výkres DPS – top (strana součástek)



Rozměry 57x54mm, měřítko M1:1

### 3. Osazovací výkres DPS – bottom (strana spojů)



Rozměry 57x54mm, měřítko M1:1