

Mobilní robot jako pomůcka pro výuku programování mikropočítačů

Jiří Zmeškal

Bakalářská práce
2016



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2015/2016

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jiří Zmeškal**
Osobní číslo: **A13079**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Bezpečnostní technologie, systémy a management**
Forma studia: **prezenční**

Téma práce: **Mobilní robot jako pomůcka pro výuku programování mikropočítačů**

Téma anglicky: **A Mobile Robot as a Tool for Teaching Microcontroller Programming**

Zásady pro vypracování:

1. Prostudujte a popište principy jednoduchých mobilních robotů vhodných pro využití ve výuce.
2. Navrhněte koncepci robota a technické prostředky včetně řídicího mikropočítače.
3. Sestavte prototyp robota podle svého návrhu.
4. Vytvořte programové vybavení pro použitý mikropočítač s ohledem na jeho využití při výuce.

Rozsah bakalářské práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy – linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky. Praha: BEN, 2003. ISBN 80-730-0077-6.
2. MARGOLIS, Michael. Arduino cookbook. 1st ed. Sebastopol: O'Reilly, 2011, xxi, 631 s. ISBN 978-0-596-80247-9.
3. MASSIMO BANZI. Getting started with Arduino. 2nd ed. Farnham: O'Reilly, 2011. ISBN 9781449309879.
4. NOVÁK, Petr, Mobilní roboty – pohony, senzory, řízení, Praha: BEN – technická literatura, 2005. ISBN/EAN 80-7300-141-1 / 9788073001414.
5. PINKER, Jiří. Mikroprocesory a mikropočítače. Praha: BEN – technická literatura, 2004. ISBN 80-730-0110-1.

Vedoucí bakalářské práce:

Ing. Jan Dolinay, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

23. února 2016

Termín odevzdání bakalářské práce:

30. května 2016

Ve Zlíně dne 16. února 2016



doc. Mgr. Milan Adámek, Ph.D.
děkan



Ing. Jan Valouch, Ph.D.
ředitel ústavu

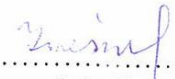
Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové/bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové/bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 25. 5. 2016


.....
podpis diplomanta

ABSTRAKT

Práce se zabývá vytvořením modelu mobilního robota, který bude sloužit jako výuková pomůcka pro programování mikropočítačů. Obsahem teoretické části je objasnění základních pojmů v oblasti mikrokontrolerů. Dále jsou popsány jednotlivé součásti, které se pro stavbu robota nejčastěji používají. V praktické části je navrhnut model robota, který se dokáže v prostoru pohybovat a vyhýbat se překážkám. Ovládání je řešeno pomocí vývojového kitu Arduino, na kterém se studenti učí programovat mikrokontrolery. Práce také uvádí programové vybavení pro tento vývojový kit. Detekce překážek je řešena formou ultrazvukových detektorů. Pohyb je realizován pomocí upravených servomotorů s vyrobenou řídicí jednotkou a servomotorem pro zatačení. Tento model bude sloužit pro popularizaci programování mikrokontrolerů mezi studenty.

Klíčová slova: mikrokontroler, Arduino, mobilní robot

ABSTRACT

This thesis deals with a mobile robot model creation. The robot shall serve as an educational aid for microcomputers' programming. The content of the theoretical part of the thesis is the explanation of the basic terms in the field of microcontrollers. Further there are described in the thesis individual components which are most often used for the construction of the robot. The model of the robot which is able to move in space and to avoid barriers is proposed in the practical part of the thesis. The control over the robot is exercised through the development kit Arduino which is used for the microcontroller programming education of students. There is programming equipment for this development kit mentioned in the thesis as well. Barriers detection is secured by ultrasound detectors. Moves are performed by adjusted servomotors with manufactured controlling unit and a turning servomotor. This model shall serve for the microcomputer programming popularisation for students.

Keywords: microcontroller, Arduino, mobile robot

Na tomto místě bych chtěl poděkovat vedoucímu práce, Ing. Janu Dolinayovi, Ph.D. za cenné rady, vstřícné jednání a věnovaný čas, které mi dopomohli ke zpracování mé bakalářské práce. Dále bych chtěl poděkovat mým rodičům za podporu při studiu.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 MIKROKONTROLER.....	11
1.1 MIKROPROCESOR	11
1.1.1 Procesory typu RISC a CISC	11
1.1.2 Architektura mikroprocesoru	12
1.2 KOMUNIKAČNÍ ROZHRANÍ	13
1.2.1 USB	13
1.2.2 RS-232.....	14
1.2.3 I2C.....	14
1.3 PAMĚŤ.....	15
1.3.1 ROM.....	15
1.3.2 PROM	15
1.3.3 EPROM.....	15
1.3.4 EEPROM.....	16
1.3.5 Paměť FLASH.....	16
1.3.6 RAM.....	16
1.4 PERIFERNÍ OBVODY	16
2 ROBOTIKA	17
2.1 HISTORIE.....	17
2.2 ROZDĚLENÍ ROBOTŮ	17
2.2.1 Mobilní roboty	17
2.2.2 Průmyslové roboty	18
2.2.3 Hobby roboty	18
3 PODVOZKY ROBOTŮ.....	19
3.1 KOLOVÝ PODVOZEK	19
3.1.1 Diferenciální.....	19
3.1.2 Ackermanův	19
3.1.3 Synchronní	20
3.1.4 Všesměrový.....	21
3.2 PÁSOVÝ PODVOZEK.....	22
3.3 KRÁČEJÍCÍ ÚSTROJÍ.....	23
4 SENZORY.....	25
4.1 INTERNÍ SENZORY	25
4.1.1 Senzory natočení	25
4.1.2 Otáčkoměry	27
4.2 EXTERNÍ SENZORY	27
4.2.1 Taktilní senzor.....	27
4.2.2 IR detektor.....	27
4.2.3 Ultrazvukový detektor.....	28
4.2.4 Kamery	28

5	POHON	29
5.1	STEJNOSMĚRNÉ MOTORY	29
5.2	SERVOMOTORY	29
5.3	KROKOVÉ MOTORY	30
6	PROSTŘEDKY PRO STAVBU ROBOTA	31
6.1	ARDUINO.....	31
6.2	MERKUR.....	31
6.3	HSET	31
II	PRAKTICKÁ ČÁST	32
7	TECHNICKÉ PROSTŘEDKY POUŽITÉ PŘI VÝROBĚ.....	33
7.1	VÝROBA PLOŠNÝCH SPOJŮ	33
7.2	ARDUINO.....	35
7.3	SERVOMOTORY	36
7.4	JEDNOTKA PRO OVLÁDÁNÍ MOTORŮ	36
7.5	ULTRAZVUKOVÝ SENZOR	39
7.6	BATERIE	40
8	SESTAVENÍ PROTOTYPU ROBOTA.....	41
8.1	SESTAVA ROBOTA	41
8.2	ELEKTRICKÉ ZAPOJENÍ ROBOTA	42
9	PROGRAMOVÉ VYBAVENÍ ROBOTA.....	44
9.1	NA ÚVOD.....	44
9.1.1	Důležité informace	44
9.1.2	Propojení s PC.....	44
9.2	PROGRAMOVÁNÍ JEDNOTLIVÝCH ČÁSTÍ ROBOTA	45
9.2.1	LED dioda	45
9.2.2	Ultrazvukový senzor	45
9.2.3	Ovládání jednotek motorů.....	47
9.2.4	Ovládání servomotoru	47
9.3	KOMPLETACE A STRUKTURA UKÁZKOVÉHO PROGRAMU ROBOTA.....	48
	ZÁVĚR	50
	SEZNAM POUŽITÉ LITERATURY.....	51
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	53
	SEZNAM OBRÁZKŮ	54
	SEZNAM TABULEK.....	55

ÚVOD

V dnešním světě nás elektronika obklopuje doslova na každém kroku. Obtížně by se hledal člověk, který v dnešní době nepoužívá mobilní telefon. Většina automobilů, které denně vidáme na ulici, má mnoho zabudovaných zařízení, jako je GPS, parkovací senzory, centrální jednotky a další vymoženosti. Domovy, obchody, výrobní linky, kanceláře, všude na těchto místech se nacházejí elektrická zařízení. A všechna tato zařízení musejí být určitým způsobem ovládána.

Pro ovládání se nejčastěji využívají mikrokontrolery, které mají vše potřebné pro svou činnost integrované v jednom pouzdře. Jsou jimi ovládány hračky pro děti, žehličky, pračky a mimo jiné také mobilní telefony, které si neustále nosíme s sebou. Díky velkému rozmachu elektroniky a velmi nízkým cenám mikrokontrolerů jsou zabudovány ve všem, kde alespoň trochu vylepší některou funkci. Pro mnoho lidí je elektronika něco nepochopitelného. Nedokáží si vůbec představit, jak funguje. Stačí jim pouze vědomí, že když zmáčknou tlačítko na ovladači televize, přepne se kanál z fotbalu na oblíbený seriál. Nevidí za tím to, že ovladač musí nejprve vyhodnotit stisknutí onoho tlačítka, informaci přenést do televize a ta následně podle přijatých informací přepne kanál. A nebo jednodušší zařízení ve formě dětské hračky, která po zmáčknutí tlačítka přehraje jednu z mála nahraných melodií.

Vyhodnocovací elektronika není černá skříňka, která záhadně funguje, ale pracuje přesně tak, jak ji člověk naprogramoval. Pokud tuto skutečnost dokážeme přijmout, otevřít pomyslnou černou skříňku a nahlédnout dovnitř, můžeme si poté vyrobit cokoli, co nás napadne. Nejjednodušší způsob, jak se s programováním seznámit, je využít předem vytvořených zařízení, na kterých dokážeme vidět výsledek našeho programování. Právě tomuto faktu je věnována celá bakalářská práce. Vysvětlí funkci základních částí mikropočítačů a uvede množství způsobů, jakými se může sestavit mobilní robot. Právě sestrojený mobilní robot, s uzpůsobenou konstrukcí a funkcemi, je využit pro popularizaci programování mikropočítačů, jelikož student uvidí veškeré své snažení přímo.

I. TEORETICKÁ ČÁST

1 MIKROKONTROLER

Jedná se o programovatelné obvody, které obsahují na jednom čipu řídicí mikroprocesor, vstupně výstupní zařízení, paměť, časovače, a/d převodníky a další periferie. Mikrokontroler sám o sobě nevykazuje žádnou činnost. Aby plnil operace, k nimž je určen, musí být nejprve k danému účelu naprogramován. Využívá se pro řídicí a kontrolní účely, kde vše potřebné je integrované v jednom pouzdře. Mikrokontroler umí zpracovávat pouze strojový kód. Pro překlad složitějších jazyků se využívají překladače. Program se zapisuje jako posloupnost instrukcí, které jsou prováděny za sebou. Toto je důležité brát v potaz při vytváření vlastního programu.

1.1 Mikroprocesor

Jedná se o hlavní část celého systému. Mikroprocesor zpracovává instrukce programu a ovládá funkci celého systému. Instrukce může vykonávat buď sám, nebo spolupracuje s ostatními částmi systému. Mikroprocesor specifikují dvě slova. Mikro – celá součástka je miniaturizovaná do jednoho pouzdra a procesor, jehož význam spočívá ve schopnosti pracovat podle připraveného programu. Mikroprocesor je schopen zpracovávat pouze jednoduché mikroinstrukce. Psát program čistě v těchto mikroinstrukcích by bylo velmi složité a nepřehledné. Proto se využívají různé jazyky pro psaní programu. Program zapsaný v takovém programovacím jazyku se pak pomocí překladače převádí do strojového kódu, který mikroprocesor dokáže zpracovat. [8]

1.1.1 Procesory typu RISC a CISC

Procesory můžeme rozdělit podle mnoha kritérií, jedním z nich je rozdělení podle instrukční sady procesoru. Jedná se o množství instrukcí, které může procesor využívat. Složitější instrukce se poté dělají posloupností instrukcí jednodušších. Čím složitější instrukce procesor vykonává, tím delší dobu mu vykonání trvá. S přibývajícím složitostí instrukce roste i složitost řadiče, a tím je procesor více zpomalován. Proto se hledá optimální řešení ve formě omezené instrukční sady, která dovoluje rychlejší chod procesoru. [8]

Dělí se na koncepce RISC a CISC.

RISC – Reduced Instruction Set Computer, programová výbava je zredukována na minimum často používaných instrukcí. Jednotlivé instrukce pracující s vnitřní pamětí jsou

vykonávány stejně dlouhou dobu a jsou zřetězovány pro rychlejší vykonávání posloupnosti instrukcí. Výsledný program je delší než u CISC, ale díky jednodušším instrukcím je výsledný čas vypracování kratší. [8]

CISC – Complete Instruction Set Computer, mikroprocesor má co nejúplnější instrukční sadu. Jednotlivé instrukce mohou být natolik složité, že zastupují celé bloky programu. Také instrukční sady bývají vytvořeny přímo pro potřeby kompilátorů, aby byl překlad ze složitějších jazyků co nejsnadnější. Vykonání těchto instrukcí poté zaměstná procesor na několik taktů. Řadič používaný v těchto typech procesoru je velmi složitý a zpomaluje chod procesoru. [8]

1.1.2 Architektura mikroprocesoru

V této kapitole bude vyjasněna stavba a základní mechanismy využívané mikroprocesorem.

Aritmeticko – logická jednotka (ALU)

Probíhají v nich aritmetické a logické operace. Může jich být v procesoru víc.

Matematický koprocessor

Pomáhají s výpočty desetinných míst. Může být jeden i více.

Řadič

Řídicí jednotka, která řídí činnost jednotlivých komponent mikroprocesoru podle prováděných instrukcí.

Registry

Vnitřní paměť mikroprocesoru. Rychlý přístup k datům uložených v nich.

Vnitřní šířka dat

Udává počet bitů zpracovaných během jednoho taktu mikroprocesoru. Díky využití 64 bitového procesoru dochází k rozšiřování adresovatelného paměťového prostoru a tím k navýšení maximální kapacity dat.

Sběrnice

Soustava vodičů zajišťující přenos signálů mezi jednotlivými komponentami mikropočítače. Sběrnice dovolují připojování dalších jednotek bez zásahu do vnitřního zapojení. V jednu chvíli smí být na sběrnici pouze jedna data, při komunikaci více zařízení

zároveň by došlo ke kolizi. Sběrnice jsou tři druhy, datová, adresová a řídicí. Datová má šířku 8 bitů nebo jejich násobek a zajišťuje přenos dat. Adresová sběrnice má šířku 16 nebo 20 bitů, podle procesoru. Zajišťuje výběr připojených zařízení. Řídicí sběrnice se poté odvíjí od architektury mikropočítače a potřeb jednotlivých zařízení. Zajišťuje správné taktování komponent počítače. [8]

Vnitřní frekvence

Též se označuje jako takt. Pro správnou činnost mikroprocesoru se musí všechny jeho vnitřní součástky řídit impulzy. Ty jsou vytvářeny taktovacími generátory.

Vnější frekvence

Taktovací frekvence pro vnější obvody. Kvůli absenci samostatného generátoru bývá vnitřní frekvence násobkem té vnější.

Napájecí napětí

Čím větší napětí procesoru je, tím větší je jeho příkon a tepelné vyzařování. Právě kvůli co nejmenší spotřebě se výrobci snaží dosáhnout co nejmenšího napětí. Dříve se používalo napětí 5V. Dnes u velmi výkonných mikroprocesorů je toto napětí sníženo na 2V. [14]

1.2 Komunikační rozhraní

Pro komunikaci dvou a více zařízení mezi sebou je nutné mezi nimi zařídit přenos dat. K tomuto účelu slouží právě sběrnice a příslušné konektory a kabely. Důraz je kladen převážně na datovou propustnost, univerzálnost, rozměry a kompatibilitu. Starší zařízení velmi často používala pro přenos dat standard RS-232. Oproti univerzálnější sběrnici USB je možné přenést méně informací za čas, ale na mnohonásobně delší vzdálenost, čehož se hlavně v průmyslu velmi často využívá. Proto často mají nové zařízení také zabudovaný standard RS-232 kvůli zpětné kompatibilitě se staršími zařízeními.

1.2.1 USB

Nejuniverzálnější druh sériového rozhraní. Existuje několik variant konektorů, které však nemají na přenosový protokol vliv. V dnešní době je USB rozhraní implementováno na většině vyráběných zařízení. Poslední verze USB 3.0 je schopna přenést až 3,2 Gb/s. Přitom sběrnice je zpětně kompatibilní. Rychlost se poté řídí nejpomalejším zařízením

zapojeným na sběrnici. Celkový počet vodičů je u Standardu 3.0 osm. Dva z nich jsou pro přenos napětí $\pm 5V$ a dva pro datový tok se zpětnou kompatibilitou s USB 2.0. Poslední dva páry tvoří kroucené páry pro Super Speed USB 3.0 ve směru vysílání a přijímání dat. Díky využití USB sběrnice je možné zařízení pomocí této sběrnice i napájet, což mnohdy velmi usnadní problém s externím napájením zařízení. Maximální délka kabelu je okolo 3 metrů. [9]

1.2.2 RS-232

Standart pro sériovou komunikaci propojující dvě zařízení navzájem. Přenos je plně duplexní, to znamená, že přenášená data putují oběma směry. Přenosové napětí je zvýšené kvůli potlačení rušení. Používá se napětí ± 5 až $\pm 15V$. Využívá se invertované logiky pro datové signály, to znamená, že log. 0 je přenášená kladným napětím a log. 1 záporným napětím. U synchronizačních signálů nedochází k inverzi. Mezi nejhlavnější vodiče patří vodiče RxD sloužící pro příjem dat, TxD pro vysílání dat a vodič GND.

Přenos rozdělujeme na asynchronní a synchronní. Při synchronním přenosu musíme mít přidaný jeden vodič pro synchronizační impulzy. Uplatnění nacházejí převážně u odesílání velkých objemů dat pomocí více vodičů. Asynchronní přenos dovoluje použití menšího počtu vodičů. Synchronizace je řešena startovací sekvencí odeslanou před odesláním vlastní zprávy, podle níž se nastaví taktovací kmitočet na přijímacím zařízení. Toto zařízení musí mít vlastní synchronizační obvod. Díky start bitům, stop bitům a paritním bitům přidaným k odesílaným datům se zpomaluje rychlost přenosu užitečných dat. Použitelná délka vodičů při zachování optimální rychlosti 19200 Bd se pohybuje okolo 15 metrů, při 9600 pak až 150 metrů. [8][16]

1.2.3 I2C

Jedná se o sériovou sběrnici, která využívá dvou vodičů a to datového a synchronizačního. Odesílání zprávy probíhá tak, že se pošle pomocí datového vodiče první část zprávy, která obsahuje značku start a adresu zařízení, s kterým se snaží navázat komunikaci. Poté se začnou posílat data po 8 bitových blocích. Po každém úspěšném přijetí bloku dat přijímač pošle hodnotu logickou 0 a tím oznámí, že dané data přijal. Zakončení odesílání je řešené značkou stop. Obě tyto značky využívají kombinaci hodnot na adresovém a datovém kabelu, které se normálně nevyskytuje. Taktovací kmitočet bývá 100kHz, 400kHz nebo 3,4

MHz. Jednotlivé uzly mají na výstupu otevřené kolektory tranzistorů jak pro data, tak pro synchronizační impulzy. Díky nim je možné připojit další zařízení. [8]

1.3 Paměť

Je to pracovní prostor procesoru, z kterého procesor čerpá a případně do nich ukládá data. Dvě hlavní paměti jsou datové, kde jsou uloženy data a programové, s kterými přímo procesor pracuje a ukládá do nich informace. Další mohou být paměti pro periferní a jiné paměťové prostory. Dynamické vlastnosti programové paměti musí plně vyhovovat taktování procesoru, jelikož vkládání čekacích taktů by zpomalovalo celý systém. Zde si uvedeme nejčastěji používané programové paměti. Kromě RAM všechny paměti uchovávají informace i po odpojení napětí.[8][15]

1.3.1 ROM

Celým názvem Read Only Memory je paměť, která je schopna po dlouhou dobu uchovat uloženou informaci i po vypnutí přívodu elektřiny. To je důležité hlavně z důvodu uchování spouštěcích informací potřebných pro správné načtení systému. Jak již z názvu vyplývá, data se dají z paměti číst, ale je velmi obtížné, mnohdy nemožné, do nich informaci uložit. Data se na ně ukládají při výrobě.[15]

1.3.2 PROM

Programmable ROM jsou čipy, které umožňují jednorázové zapsání kódu. Při výrobě jsou na všechny adresy zapsány hodnoty binární 1. Při programování jsou tyto místa vypálena, tudíž se logická 1 změní na logickou 0. Proces je jednorázový, to znamená, že data lze na čip zapsat jen jednou. [15]

1.3.3 EPROM

Vylepšená verze PROM, umožňující vymazání obsahu čipu. Děje se tak intenzivním UV zářením, které chemickou reakci znovu propojí přerušenou cestu. Binární 0 se změní na binární 1. Kvůli přístupu UV světla na čip, je nutné konstrukčně dovolit světlu, aby se dostalo k jádru čipu. Dělá se to přes křemíkové okénko. Změna chemické reakce v čipu dovoluje pouze omezený počet přeprogramování, řádově desítky. [15]

1.3.4 EEPROM

Paměti dovolující elektrické vymazání obsahu čipu. Není tedy potřeba žádné manipulace se samotnými čipy, nebo speciálního zařízení. Programování probíhá po jednotlivých adresách. Mazání i zápis dat probíhá stejným fyzikálním principem. Nemusí se tudíž již mazat celý obsah čipu. Programování je ve srovnání s EPROM nebo FLASH pomalé. [15]

1.3.5 Paměť FLASH

Obdobou paměti EEPROM. Zápis dat probíhá po jednotlivých adresách a je velmi rychlý. Mazat se musí celá paměť a je pomalejší v porovnání se zápisem. Pro větší kapacity je paměť rozdělena do sektorů, aby se nemusela přemazávat a zase zapisovat celá kapacita paměti. Buňky jsou uspořádány do řad a sloupců, tvoří vlastně takovou síť. Také rozměry těchto pamětí jsou velmi malé, což dovoluje na velmi malý čip zapsat ohromné množství informací. Díky miniaturním rozměrům je možné je aplikovat do řady specifických aplikací. [8]

1.3.6 RAM

Pocházející ze slov Random Access Memory. Z názvu vyplývá, že se jedná o paměti náhodného přístupu. Znamená to, že k jakýmkoliv datům lze přistupovat stejně rychle bez ohledu na jejich umístění v paměti, avšak nedovolují trvalé uložení dat. Po připojení napájení není v paměti obsah definovatelný. Proto musíme do této paměti nejprve nahrát data. Výhodou je, že RAM dokáže sám sebe modifikovat. V porovnání s předchozími uvedenými pamětmi jsou mnohonásobně rychlejší, což zvyšuje rychlost činnosti počítače.[8]

1.4 Periferní obvody

Jedná se o externí obvody, které pracují se signály. Patří sem převodníky, časovače, čítače nebo třeba oddělovací členy. Komunikují po sběrnici s procesorem a data ukládají do datových registrů, ze kterých poté procesor načítá data. Pokud však dojde například k přetečení čítače, skončení A/D převodu, přijetí znaku na sériové lince, jsou tyto informace uloženy ve stavových registrech. Na základě těchto stavových registrů je možné vyvolat přerušení, kde se podle závažnosti informace provedou nejdůležitější operace a poté je přerušení uvolněno. Přerušení se uplatňuje hlavně u hazardních stavů, nebo chyb různého charakteru nepříznivě ovlivňujících stav systému. [14]

2 ROBOTIKA

Robotika je věda uplatňující vědecké, technické a softwarové vědomosti. Jejím cílem je ulehčování práce člověku, nebo převzetí jeho práce úplně. Snaha o zvyšování produktivity práce a omezení lidského faktoru. Využívá se v mnoha vědních i technických disciplínách. Jejím hlavním tahounem poslední doby je automobilový průmysl, který celou svou výrobu staví právě na automatizaci. Lidé pouze kontrolují výstupní kvalitu, případně dohotovují pouze ty věci, které by byly velmi složité automatizovat. Robotika vychází ze slova robot, což je stroj vykonávající mu zadanou práci. Práci může vykonávat buď tak, že je naprogramován každý jeho pohyb, nebo se mu zadá pouze cíl a on už sám pomocí vyhodnocení situace navrhne a provede nejjednodušší řešení.

2.1 Historie

První zmínky o robotech můžeme nalézt v řecké mytologii. Bůh kovářství Hefaistos podle legend vyráběl mechanické sluhy, různé samohybné roboty a také vytvořil Talose, což byl obří robot podobající se člověku. V patnáctém století Leonardo Da Vinci navrhl mnoho robotů a mechanických výtvorů. S příchodem parních strojů a průmyslové revoluce se robotika postupně dostává do průmyslu. V počátcích stroje spíše pomáhaly lidem s prací, ale nedokázaly převzít jejich práci úplně. Tak se stalo až v 2. polovině dvacátého století. Kdy byly v Americe vyrobeny první roboty pro práci s těžkými a horkými odlitky ve slévárně, které plně přebrali lidskou práci. Následně se začala automatizovat téměř všechna odvětví průmyslu. Největším tahounem zůstává automobilový průmysl, hutnický průmysl a průmysl pracující s plasty. [10]

2.2 Rozdělení robotů

Existuje mnoho způsobů rozdělení robotů. Rozdělení podle způsobu pohybu, volnosti pohybu, podle funkce, autonomie rozhodování a podle mnoho dalších kritérií. V této kapitole budou roboty rozděleny podle prostředí, ve kterém pracují. Rozdělení budou na mobilní roboty, průmyslové roboty a hobby roboty.

2.2.1 Mobilní roboty

Jedná se většinou o elektromotorický stroj, který je schopný se sám přepravovat z místa na místo a vnímat, případně měnit, své okolí. Pro to, aby mohl reagovat na podněty, musí být také příslušným způsobem řízen. Pro autonomní pohon je důležité zajistit přísun energie po

celou dobu jízdy robota. Dělá se to převážně přídavnými bateriemi. Mobilní robot může být libovolně sestaven, pokud bude zachována jeho primární funkce, a to aby se sám dokázal pohybovat z místa na místo.

2.2.2 Průmyslové roboty

Jedná se o roboty využívané v průmyslu. Nejčastěji se vyrábí v provedení, kde mají pouze jedno rameno, ale značné množství kloubů umožňujících pohyb a rotaci v mnoha osách. Konec robota je osazen paletou nástrojů pro konkrétní činnost. Může se jednat o vrtáky, frézky, balící jednotky, svářečku a mnoho dalších nástrojů. Vyráběné jsou v mnoha velikostech podle způsobu použití. Plošné nasazování robotů do výroby zrychluje, zpřesňuje, zefektivňuje a v konečném hledisku i zlevňuje výrobu.



Obr. 1. Průmyslový robot Motoman ROBOTECH [17]

2.2.3 Hobby roboty

Za hobby roboty můžeme označit všechny roboty, které jsou vytvořeny pro zábavné, nebo výukové účely. Můžeme sem také zařadit i mobilní roboty, které svou činností nepřinášejí primárně zisk. Může se jednat o kvadruptry, RC modely, stavebnicové modely nebo roboty vytvořené pro zábavu..

3 PODVOZKY ROBOTŮ

Pro to, aby se mohl robot sám pohybovat, potřebuje ústrojí, které mu tento pohyb dovolí. Existuje mnoho způsobů, jak rozpohybovat robota. Nejběžnějším způsobem je robota umístit na kolový podvozek. Ten se ovšem nehodí do nepřístupných míst, nebo kluzkých povrchů. Proto využíváme i pásové a kráčející podvozky. Na konstrukci jsou složitější, ale výhody, které přinesou, často zastíní nevýhody spojené se složitější konstrukcí. Dalším možným způsobem pohybu robota je pomocí vrtule, nebo jejich soustavou. Ve vodě se poté používají lodní šrouby, nebo také upravená konstrukce tak, aby napodobovala pohyb ryb.

3.1 Kolový podvozek

Jedná se o nejčastěji používaný podvozek kvůli své jednoduchosti na konstrukci a množství variability jeho využití. Existuje řada způsobů, jak je možné vytvořit kolový podvozek. Jak podle počtu kol, tak způsobu jejich uspořádání a ovládání. V následující části si rozebereme nejznámější konstrukce podvozku a to diferenciální, synchronní, Ackermanův a všesměrový.

3.1.1 Diferenciální

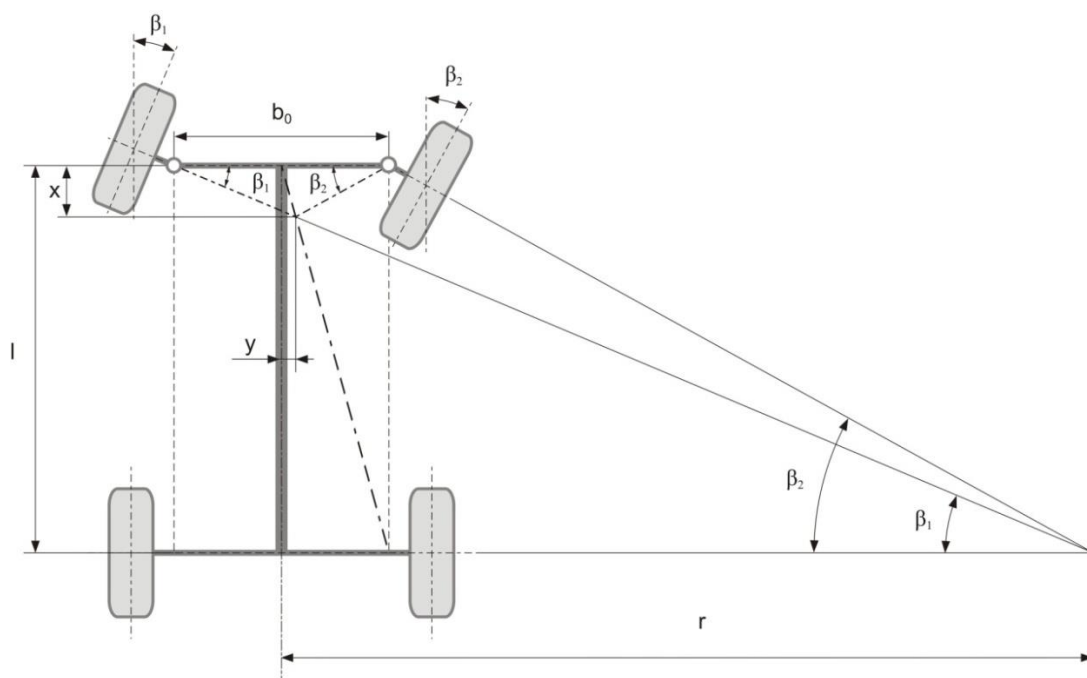
Jedná se o velmi jednoduchý způsob konstrukce podvozku. V principu máme tři kola, přičemž dvě zadní slouží pro pohyb a zatáčení a jedno kolo vpředu slouží pro udržování stability robota. Přední kolo je otočné a nemá žádný svůj pohon. Důležité u této konstrukce je to, že u zadních kol můžeme korigovat rychlost otáčení kol a také ovládat směr jejich točení. Pokud se obě kola točí stejným směrem a stejnou rychlostí, robot jede vpřed. Pokud se jedno kolo bude točit o trochu pomaleji než druhé, robot bude zatáčet na stranu kola s menší rychlostí. Čím větší bude rozdíl v rychlosti otáčení kol, tím více robot zatočí. Pokud se jedno kolo bude točit dopředu a druhé dozadu, bude se robot schopný otočit na místě. Alternativou je využití čtyř kol, kde jsou všechny kola uchycena pevně a zatáčení probíhá formou smyku. [1]

3.1.2 Ackermanův

Nejznámější typ podvozku. Využívá se například u automobilů a dětských tříkolek. Existují dvě varianty způsobu použití při čtyřech kolech. První variantou je rozdělení

hnacích a natáčecích kol. Pohon zajišťují zadní kola, která jsou pevně uchycena a nedovolují s nimi zatačet. Zatačení zase zajišťují přední dvě kola, která nejsou hnaná, ale pouze svým otáčením zajišťují otočení celého automobilu nebo robota. Tento systém byl uplatňován převážně na starších autech. Moderní auta jsou buď poháněna přední nápravou, nebo pohon zajišťuje přední i zadní náprava. Druhou variantou je spojení hnacích i otočných kol. Zadní dvě kola zůstávají pevně spojené, ovšem nepohání auto jako takové a pouze zajišťují stabilitu. Přední dvě kola zajišťují jak otáčení, tak pohon.

Problémem u této konstrukce je hlavně nerovnoměrný poloměr zatačení každého kola. Proto se tento poloměr musí upravit tak, aby se neopotřebovávalo jedno kolo mnohem více než druhé, díky většímu tření, a také, aby auto kladlo co nejmenší odpor v jízdě. Reálně se to dělá úpravou natáčení kol. Kolo blíže do středu opisuje menší poloměr kružnice než venkovní kolo (úhel β_2), tudíž se musí vnitřní kolo více natočit než venkovní (úhel β_1), aby nedocházelo k zvýšenému tření a s ním spojené nerovnoměrné opotřebení. [2] Vyobrazení můžeme vidět na obrázku 2.



Obr. 2 Ackermanův podvozek [18]

3.1.3 Synchronní

Nejčastěji bývá realizován pomocí 3 nebo 4 kol. Přitom každé kolo má svůj vlastní pohon a je schopné se otáčet kolem dvou os. Jedna osa je pro otáčení kola realizující pohon jako

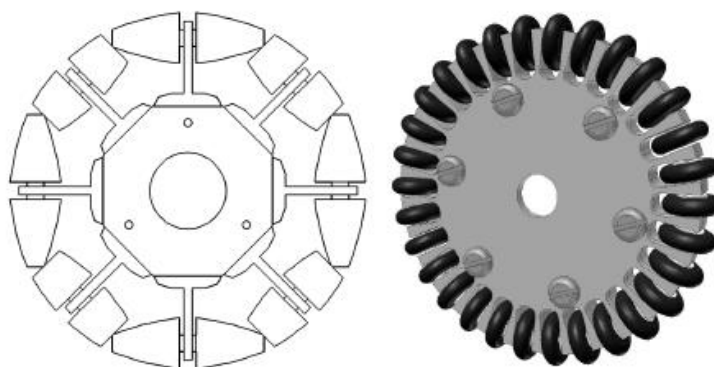
takový. Druhá osa je pak orientována kolmo na povrch, po kterém model jede a zajišťuje otáčení modelu. Kola mohou být volně otočná, to znamená, že se všechny natáčí stejným směrem. Pokud budeme uvažovat model se 3 koly, stačí poté ovládat otáčení pouze jednoho z těchto kol do potřebného směru. Ostatní kola se přizpůsobí. Jedná se o jednoduchý systém zatačení, ovšem problémem je překonávat větší nerovnosti terénu. Tento druh podvozku je díky své jednoduchosti na ovládání vhodný pro výukové aplikace. [2]

3.1.4 Všesměrový

Jedná se o konstrukci takového druhu kol, které dovolují zároveň pohyb systému vpřed a zároveň do strany nebo rotace. Konstrukce kol umožňující takový pohyb je realizováno pomocí klasického kola, které má ovšem na svém obvodu soustavu válečků. Právě tyto válečky zajišťují i boční posun kola při jízdě vpřed. Díky všesměrovému systému kol je robot schopen otáčení kolem svého těžiště. Pro snazší pochopení systému uvedeme dvě nejčastější konstrukce kol, Stanfordské a Mecanumské. [2]

Stanfordské kolo

Jak můžeme vidět na obrázku 3, způsob provedení tohoto kola je přidáním pohyblivých kroužků nebo válečků na obvod, které jsou volně otočné kolem své osy. Orientace těchto kroužků či válečků je na směr otáčení kola. Aby nedocházelo ke chvění při otáčení kola nedokonalostmi přechodů mezi jednotlivými válečky, bývá přidávána druhá řada válečků posunutých tak, aby středy válečků v druhé řadě zakrývaly díry mezi válečky první řady.



Obr. 3. Stanfordské kolo [19]

Mecanumské kolo

Oproti Stanfordské konstrukci kola má tento systém natočené válečky o 45°. Výhodou tohoto systému je lepší návaznost válečků na sebe. Nepotřebuje pro kompenzování

nerovností přidání druhé řady válečků. Hlavní nevýhodou je mrtvý úhel právě ve směru orientace válečků. Pro kompenzaci se musí kolem natáčet tak, aby se kolo do této polohy nikdy nedostalo. Další nevýhodou je rychlejší rotace válečků při pohybu vpřed. [2]



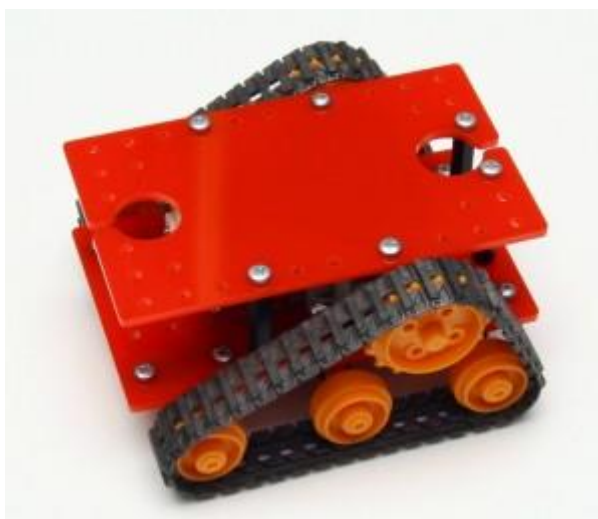
Obr. 4. Mecanumské kolo [20]

3.2 Pásový podvozek

Tento druh podvozku má oproti kolovému mnohem větší prostupnost terénem a uplatnění i v extrémních podmínkách. Používá se pro speciální aplikace a především na větších robotech. Výhody jsou převážně větší nosnost soustavy, možnost snáze překonávat převýšení a schody a lepší stabilita i na kluzkém povrchu.

Pásové podvozky jsou řízeny diferenciálně a využívají zatáčení smykem. Zatačení vytváří mnohem větší tření než jiný systém podvozku. Díky tomuto musí být poháněny výkonnějšími motory než při použití kolového podvozku. Nepřesnosti v roztečích jednotlivých opěrných bodů pásu, jejich propnutí a rozdílné tření obou pásů způsobují mnohem obtížnější jízdu po přímce na delší vzdálenosti.

Speciální konstrukce pásových podvozků lze využít v mnoha specializovaných aplikacích. Jedním z nich je užití tří pásů navzájem posunutých o 120° . Tento systém dovoluje prozkoumávání šachet a potrubí i ve svislých polohách díky možnosti zapření robota na vnitřních stěnách. Dalším je uzpůsobení podvozku pro jízdu v suti. [3]

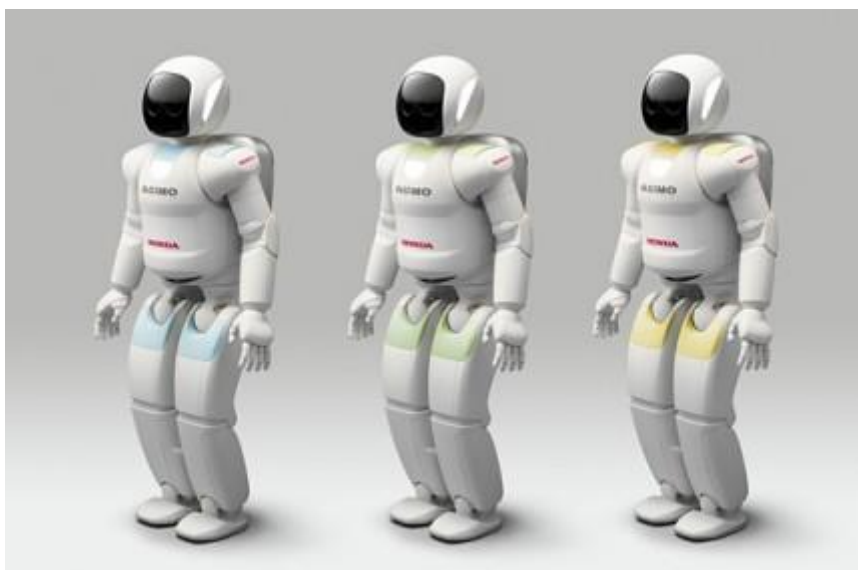


Obr. 5. Pásový odvozek [21]

3.3 Kráčející ústrojí

Kráčející podvozky dělíme podle počtu nohou. Skákající roboty začínají s konstrukcí využívající pouze jedné nohy. Avšak systém jenom jedné nohy je značně nestabilní. Většinou se jednonohé roboty používají pouze v laboratořích, a to pouze pro zkoumání dynamického chování, které se uplatňuje na soustavách s více nohou. Reálně můžeme vytvořit robota s neomezeným počtem nohou, ovšem čím více nohou, tím složitější je jejich řízení a materiální náklady. Hlavní výhodou je nejlepší prostupnost terénem z pozemních způsobů pohybu. Dokážou překonat relativně vysoké překážky a příkopy vůči svému tělu. Nevýhodou je vysoká složitost hardwarových a softwarových požadavků a energetická náročnost. [3]

Dvounohé roboty většinou slouží k experimentálním účelům a napodobují chůzi člověka. Jedním z nejdokonalejších robotů současnosti je robot jménem Asimo od společnosti Honda. S jeho výzkumem začali už v roce 1980, kdy se robot skládal pouze z nohou, které postupně odladili natolik, že k němu nakonec připojili i trup a ruce. Celý systém se zdokonaloval natolik, až v roce 2000 byla představena první generace Asimo. Tento první robot značně připomínal člověka. Avšak jeho rozhodování mělo mnoho chyb, které se díky pokročilým technologiím a softwaru dodnes zdokonaluje. [4]



Obr. 6. Nejnovější generace robotů Asimo [4]

Vícenohé systémy jsou často inspirovány přírodou. Napodobují například chůzi mravenců, čtyřnohých zvířat nebo různých housenek. Obecně platí, že čím více nohou daný robot má, tím složitější je jeho ovládání a konstrukce. Toto pravidlo neplatí u dvounohých robotů, jejichž stabilita je díky dvou opěrným bodům nestabilní a tudíž se klade mnohem větší nárok na celkovou stabilitu robota a jeho ovládání.

4 SENZORY

Pro návrh senzorického subsystému robota je nutné zvážit mnoho ovlivňujících faktorů. Je to komplexní proces, který zohledňuje vlastnosti senzorů, pracovní prostředí robota, možnost rušení senzorů navzájem, nároky na energetiku a softwarové a hardwarové vybavení robota. Návrh můžeme rozdělit podle způsobu snímání okolí a to buď na interní senzory, které se přímo zaměřují na snímání vlastních stavů robota, nebo na externí senzory, které měří parametry okolí kolem samotného robota. Samotný návrh jednotlivých systémů je vhodný rozdělit na několik částí. První částí je to návrh samotného senzoru a obslužných obvodů, druhou částí je poté propojení tohoto senzoru s vlastním řídicím ústředím robota.

Aby mohl robot správně fungovat, potřebuje dostávat informace včas a s dostatečnou přesností. Proto je důležité dbát na správný výběr detektorů pro konkrétní aplikace. [5]

4.1 Interní senzory

Interní senzory podávají ucelený přehled o stavu a natočení jednotlivých komponent, které se využívají pro měřicí účely. Může se jednat o indikaci stavu baterie, teplotu kritických částí robota a monitorování komunikace. Pro určování přesné polohy a rychlosti je nutné zajistit stálý styk s podlahou bez prokluzování, přesné měření otáček kol nebo hřídele a v neposlední řadě ze zjištěných parametrů vypočítat trajektorii v prostoru. [5]

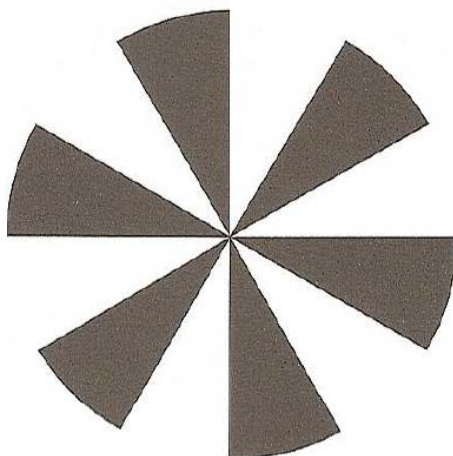
4.1.1 Senzory natočení

Provedení těchto senzorů bývá analogové a digitální. Analogové využívají změn fyzikálních veličin. Podle toho se dělí na kapacitní, odporové a indukční. Nevýhodou analogových senzorů je jejich omezené natočení. Pro konstrukci robotů jsou více využívány digitální. Mezi hlavní výhody patří neomezený rozsah měření natočení a bezdotykový způsob měření. Podle metod měření je dělíme dále na inkrementální (přírůstkové) a absolutní. [5]

Inkrementální

Vynikají vysokou rozlišovací schopností, malými rozměry a nízkou hmotností. Principem je otáčení mezikruží s pravidelným střídáním průhledných a neprůhledných oblastí. Jdoucí od středu k okraji kotoučku. Při otáčení přerušují emitované světlo LED diod, které jsou zaznamenávané na druhé straně kotoučku. Pokud světlo danou oblastí projde, pak je

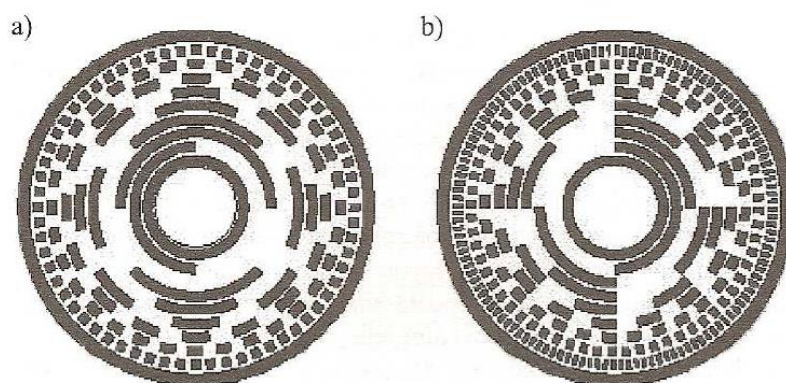
zaznamenáno, pokud je v dané oblasti stínítko, kotoučem světlo neprojde. Snímání bývá provedeno fotorezistory. Tento typ senzoru nám udává rychlost otáčení, ale nedovoluje nám určit přesnou polohu v každém okamžiku natočení. Musíme znát počáteční hodnotu, od které se vše odpočítává. Také nedokážeme určit směr otáčení pouze ze stavu kotoučku. [5]



Obr. 7. Kotouček pro inkrementální senzor [5]

Absolutní

Tento senzor je složitější než inkrementální, ale dokážeme díky němu určit přesnou polohu v každém okamžiku natočení. Také vhodným zvoleným kotoučkem dokážeme přesně určit směr otáčení. Výstupním kódem dokážeme určit přesnost natočení na jednotky stupňů. Můžeme využívat dva druhy kotoučků. Binární kódovací kotouček (obrázek vlevo) nebo využít Grayova kódu (obrázek vpravo), který je více odolný vůči chybám. [5]



Obr. 8. Kódové kotouče absolutního senzoru natočení [5]

4.1.2 Otáčkoměry

Jsou to senzory určené k měření rychlosti otáčení. Pro využití pro mobilní roboty se nejčastěji využívají indukční a impulzní otáčkoměry. Indukční využívají pro měření výstupní napětí. Impulzní měří výstupní frekvenci, a podle toho přepočítávají výslednou rychlost.

4.2 Externí senzory

Díky nim získáváme informace o okolí robota. Externí senzory můžeme dle fyzikálního hlediska dělit na dva druhy. Prvním jsou pasivní senzory, které vyhodnocují pouze přijaté záření z okolí. Druhým typem jsou aktivní, které sami generují záření, jehož odraz poté detekují. [5]

4.2.1 Taktilní senzor

Nejjednodušší provedení senzoru. Pracuje na principu sepnutí/rozepnutí kontaktního spínače při dotyku překážky. Při sepnutí nebo rozepnutí dojde ke změně logické úrovně, která je dále vyhodnocována. Zapojení většího počtu taktilních senzorů můžeme provést buď zapojením senzorů k multiplexeru, nebo zapojením jednotlivých senzorů k odporové dekádě a vyhodnocovat výstupní napětí.

4.2.2 IR detektor

Jedná se o detektor pracující v oblasti infračerveného záření. Nejčastěji se využívá vlnová délka 880 nm. Principem je vyzařování infračerveného záření a jeho následné vyhodnocení. Pokud se detekuje odrazení IR záření, je detekována překážka. Problémem je rozptyl světla a jeho následné vyhodnocení. Jelikož rozptyl roste s druhou mocninou vzdálenosti, a musíme za vzdálenost považovat cestu od IR LED diody k překážce a od překážky do detektoru. Díky rozptylu jsme však schopni detekovat odražené světlo i od šikmých překážek. Dalším problémem je absorpce světla. Každý materiál pohlcuje světlo jinak, tudíž bílý list papíru jsme schopni detekovat na vzdálenost v řádech metrů, černou matnou desku jen na několik desítek centimetrů.

Pro měření vzdálenosti a netečnost k jiným zdrojům IR záření můžeme vysílaný signál modulovat. Pouze pro detekci signálu se využívá specifického kmitočtu generovaného

záření. Detektor reaguje pouze na tento konkrétní kmitočet. Měření vzdáleností se poté provádí snižováním vyzařování diody a vyhodnocení hranice kdy ještě detekuje překážku a kdy už nedetekuje. Ovšem odraz závisí hodně na povrchu materiálu, tudíž je toto měření pouze orientační.

4.2.3 Ultrazvukový detektor

Ultrazvukové detektory pracují na frekvenci řádově téměř milionkrát pomalejší než rychlost světla. Tím pádem je vyhodnocování vzdálenosti překážek mnohem jednodušší a to jak z hlediska doby putování signálu, tak z nepotřeby složitých vyhodnocovacích obvodů. V principu se vyšle signál, který se po odražení od překážky vrací zpátky, a poté detekuje, jak dlouho danému signálu trval návrat. Z tohoto časového údaje je velmi jednoduché přes vzorec: $l = v * t$ (kde l je vzdálenost, v je rychlost zvuku a t je čas) dopočítat vzdálenost. Musíme ovšem počítat s teplotou prostředí, jelikož čím teplejší prostředí je, tím více jsou molekuly rozkmitané a tím méně brání v šíření vzruchu. Proto musíme počítat se vzorcem: $v = (331,82 + 0,61 \cdot \{t\}) \text{ m} \cdot \text{s}^{-1}$. Pro teplotu prostředí okolo 20°C uvažujeme rychlost zvuku $340 \text{ m} \cdot \text{s}^{-1}$ [6]. Při použití více ultrazvukových detektorů se musí zajistit, aby se navzájem nerušili. Provedením buď střídáním vysílání jednotlivých detektorů, nebo rozdílným kmitočtem jednotlivých detektorů.

4.2.4 Kamery

Tento typ detekce prostředí je využitelný pouze u robotů, které mají velký výpočetní výkon. Využívají se převážně dvě kamery, které pracují na principu lidského vnímání vzdálenosti. Pokud se do snímané scény dostane určitý objekt, triangulací se vypočítá jeho vzdálenost od robota. Tímto procesem se propočítá celá scéna a z ní se vytvoří síť, ve které je schopen robot pohybu. Obdobným principem je zjišťována vzdálenost hvězd ve vesmíru, kdy se porovnávají úhly vzdálených hvězd vůči nehybným galaxiím vzdálených o několik řádů dál. Přitom základnu tvoří poloměr oběžné dráhy Země. [7]

5 POHON

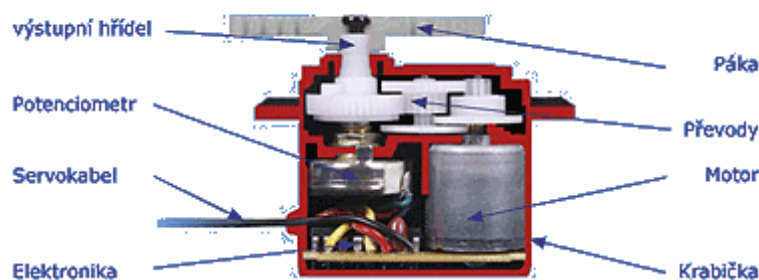
Pohonem robota rozumíme takovou soustavu součástek, které dokážou svou činností rozpohybovat robota. Pro pohyb robota se využívají nejčastěji elektrické motory. Ovládají jak kola, tak různé součásti robota (robotická noha, pásy, nohy robota). Mnohdy samotný motor pro pohyb nestačí, ale musí se přidávat různé elektronické obvody a převody. Zde si uvedeme nejčastější druhy motorů a jejich ovládání.

5.1 Stejnosměrné motory

Tyto motory pracují na přiloženém stejnosměrném napětí. Polarita napětí a její velikost přímo ovlivňuje směr a rychlost otáčení motoru. Pracuje na principu změny magnetického pole mezi dvěma permanentními magnety při přiložení proudu. Stator (neotáčivá část) je tvořena dvojicí permanentních magnetů. Uvnitř je rotor (otáčivá část), který se snaží při přiloženém napětí dostat do neutrální polohy, která leží mezi magnety. Tím že se přiblíží do této polohy, dojde k mechanickému přepólování rotoru, a tím pádem se rotor snaží dostat do další neutrální polohy. Síla způsobující tah motoru působí vždy kolmo na směr protékajícího proudu. [3]

5.2 Servomotory

Jedná se o stejnosměrné motory, které mají vlastní řídicí jednotku. Ovládají se pomocí impulsů posílaných do řídicí jednotky servomotoru. Jejich hlavní výhodou je to, že mají už vytvořené převody a podle délky impulsu se natočí o přesný úhel. Toho je využíváno především pro potřeby jednoduché a přesné polohování. Délka impulsu 1 ms odpovídá maximálnímu levému natočení hřídele a 2 ms odpovídají maximálnímu pravému natočení. Impulzem 1,5 ms poté nastavíme přesný střed otáčení. Rozsah otočení hřídele je mechanicky aretován na převodovce v rozmezí 180°. Převodový poměr nám udává poměr mezi rychlostí a tahem. Převody jsou vyráběny především z plastu, avšak pro silnější serva z kovu. Na obrázku můžeme vidět průřez samotným servomotorem. [5]



Obr. 9. Průřez servomotorem [22]

5.3 Krokové motory

Využívají se tam, kde je důležité přesné mechanické polohování. Uplatnění nacházejí například v CNC strojích, polohování hlaviček pevných disků, čteček nebo u tiskáren. Ovládání je řešeno stejnosměrnými pulzy, pomocí kterých se motor vždy pootočí. Také dokáže udržet svou polohu tam, kde je momentálně natočen. Také lze celkem snadno u jednodušších aplikací k zjištění aktuální polohy pouze počítat kroky. Konstrukčně se skládá z rotoru a statoru. Rotor tvoří permanentní magnety poskládané na hřídeli. Stator je řešen sérií cívek, které po nabuzení přitahují opačný pól rotoru. Přepínáním napájení docílíme rotace magnetického pole, které právě rotor sleduje.

Konstrukčně je můžeme rozdělit na krokový motor s pasivním rotorem, hybridní a lineární. Pasivní a hybridní má funkci sledování otáčení magnetického pole po kruhu, lineární je posun v jedné ose, nikoliv kolem osy. Dále je lze rozdělit podle způsobu buzení na bipolární a unipolární. Unipolární buzení má jeden společný výstup všech cívek. Tím je docíleno menší spotřeby, ale na úkor krouticího momentu. Má čtyři vývody. Bipolární má vlastní vývody z každé cívky, které jsou protilehlé a mají rozdílnou polaritu. Poznáme je podle šesti vývodů. [5]

6 PROSTŘEDKY PRO STAVBU ROBOTA

Na trhu lze najít nepřeberné množství výrobků, které se dají využít pro stavbu robota. Pro vlastní výrobu robota je třeba ovládací jednotky, konstrukce, pohonného ústrojí a dalších periférií. V této kapitole si uvedeme možné prostředky využitelné pro stavbu robota. Převážně se zaměříme na ovládací jednotky a konstrukční prvky robota.

6.1 Arduino

Jedná se o Open-Source platformu umožňující snadný návrh a programování elektrických zařízení. Základní modul je možné připojit přes USB k počítači a přímo pomocí pinů připojovat libovolné zařízení. Díky velké oblíbenosti mezi uživateli je právě Arduino vhodné pro výukové účely, jelikož je nepřeberné množství aplikací, kde se dá Arduino uplatnit. Programování je pro uživatele přívětivé a existuje mnoho návodů, jak daný problém řešit. Vstupně výstupní periferie přitom nemusí být od stejného výrobce, což umožňuje ještě větší možné uplatnění mezi amatéry i profesionální programátory. [12]

6.2 Merkur

Tato stavebnice má jednu obrovskou výhodu, a to je její kompatibilita s množstvím dalších stavebnic nebo součástek, díky stejné rozteči děr pro uchycení. Základním prvkem jsou samotné kovové díly, které mají různý tvar a velikost. Mají předvrtané otvory pro spojení s ostatními díly. Pomocí otvorů, šroubků, maticek a os lze vytvořit libovolnou věc dle své fantazie. Díky velké variabilitě spojování dílů může tato stavebnice sloužit jako kostra pro samotného robota. Merkur obsahuje řadu motorů, převodů a součástek, které se dají použít pro mobilní roboty.

6.3 HSET

Jedná se o robotický systém, který je tvořen množstvím desek, které se dají mezi sebou libovolně propojovat. Základním prvkem je řídicí jednotka s procesorem a USB konektorem. K němu se dají připojovat pomocí propojovacích kabelů další desky se specifickými funkcemi. Mezi ně patří desky opatřené IR čidly, bipolárními spínači, dvojíty H-můstky, desky s LED, motory a mnoho dalšího. Systém má stejnou rozteč děr jaké má Merkur, tudíž se dá spolu s Merkurem využít na konstrukci robotického systému. [11]

II. PRAKTICKÁ ČÁST

7 TECHNICKÉ PROSTŘEDKY POUŽITÉ PŘI VÝROBĚ

Pro výrobu robota bylo použito množství různých prostředků zkompletovaných dohromady. Celý návrh se odvíjel od jeho funkce, a to detekce a vyhýbání překážkám. Bylo tedy nutné navrhnout robota tak, aby se dokázal autonomně pohybovat, detekovat překážku a té se vyhnout. Také bylo potřeba jednotlivé součásti ovládat, napájet a pevně uchytit na robotu. Všechny rozvody a ovládací prvky byly vyrobeny přesně na míru tomuto robotu. Úvodem bude vysvětlena výroba plošných spojů, které byly vyrobeny pro ovládání motorů a propojení napájení a rozvodů. Dále budou popsány jednotlivé části, ze kterých se robot skládá.

7.1 Výroba plošných spojů

Pro jednoduché navrhování a vytváření plošných desek není potřeba drahé vybavení a nástrojů, jak by si mnozí mysleli. Stačí k tomu vlastní hlava, cuprextit, lihový fix, roztok chloridu železitého a případně nažehlovací fólie pro složitější zapojení. Výroba nezabere mnoho času a mnohdy velmi usnadní návrh zapojení, které by se drátově dělalo velmi složitě. Pro úpravu tvaru cuprextitu se může použít pilka na železo a rozřezat jej na potřebnou velikost.

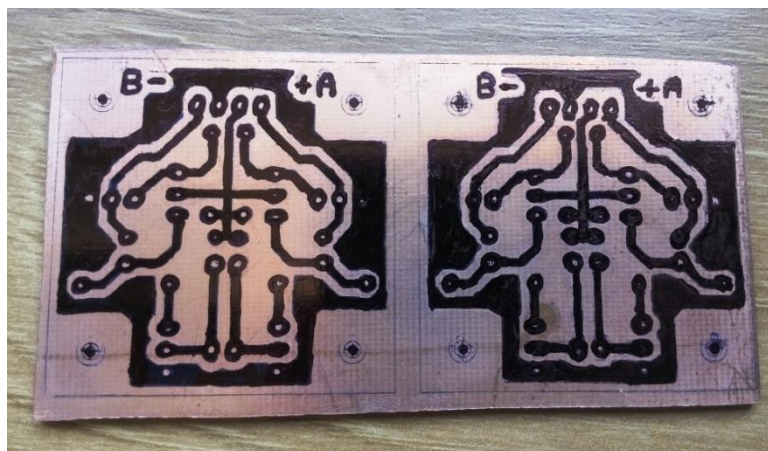
Návrh zapojení

Prvním krokem je návrh daného zapojení. Můžeme jej vymyslet z hlavy, pokud se jedná jen o jednoduché zapojení, nebo pro návrh využít specializovaných programů. Jedním z nich je program Eagle, který nám dovoluje ve free verzi navrhnout dvouvrstvou plošnou desku rozměrů deset na osm centimetrů. Pro výrobu nekomerčních plošných spojů dostačující. V Eaglu vybereme požadované součástky a vytvoříme schéma zapojení. Schéma převedeme do návrhu plošné desky a snažíme se rozmístit součástky tak, aby se žádné cesty nepřekrývaly a uspořádání součástek bylo úsporné a logické. Poté vybereme pouze vrstvy pro vlastní cesty, pájivé plošky a otvory pro součástky. Konkrétní schéma zapojení a návrh součástek na desce můžeme vidět na obrázku X a Y.

Přenesení schématu na cuprextit

Druhým krokem je nanesení schématu na předem očištěný cuprextit. Obrázek si vytiskneme na speciální fólii a přes ni to pomocí žehličky přeneseme na cuprextit. Mnohdy se stane, že se všechn inkoustový pigment nepřeneseme na desku. Vezmeme lihový fix a

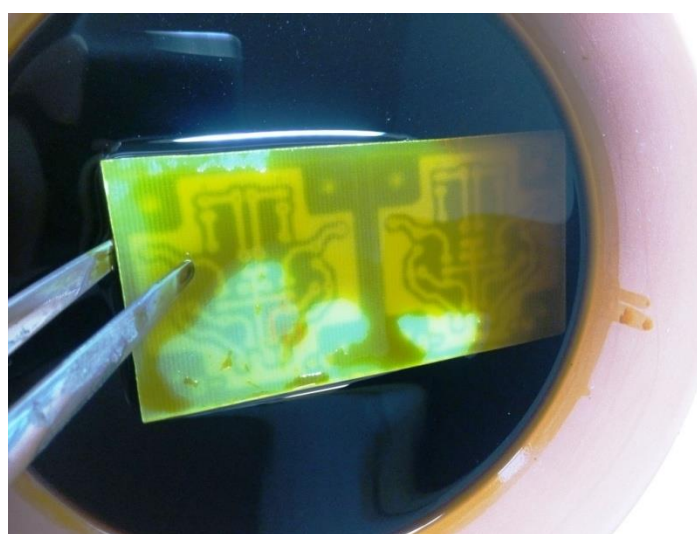
špatně přenesené cesty obtáhneme. Pokud navrhujeme celý obvod sami, pak všechny cesty vytváříme pomocí lihového fixu.



Obr. 10. Přenesené schéma na cuprextit

Leptání a vrtání

Připravíme si roztok chloridu do plastové nádoby. Poté položíme desku na hladinu chloridu nakresleným schématem dolů. Při novém roztoku a jeho dostatečném množství se vrstva mědi odleptá přibližně za 25 minut. Při nasycení roztoku leptání trvá o mnoho déle. Kontrolu provedeme sledováním průsvitnosti desky. Odleptané místa budou v roztoku tmavé, stejně jako na obrázku 11. Nebo desku vytáhneme a proti světlu sledujeme, zda je všechna měď mimo cest odleptaná. Zásadou je nenechávat desku v roztoku moc dlouho, aby se cesty nepodleptaly. Zaznačené otvory pro součástky tenkým vrtákem provrtáme.



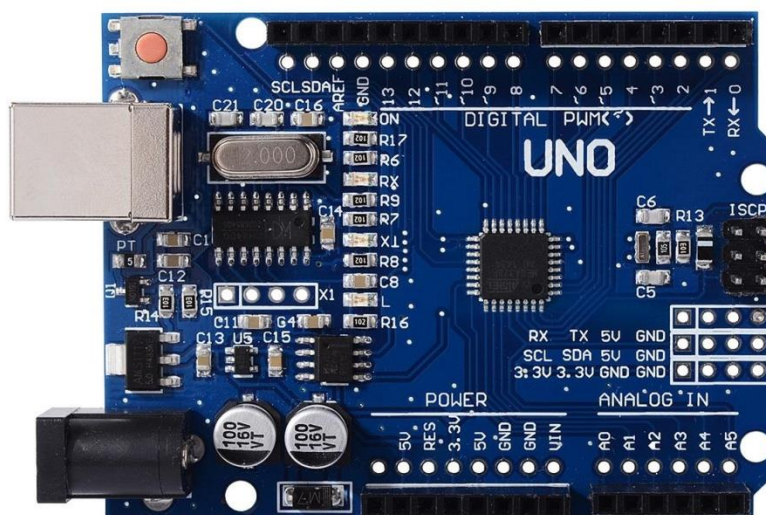
Obr. 11. Vyleptaný plošný spoj

Pájení

Součástky podle schématu osadíme a postupně zapájíme. Často se stane, že součástka nepřilne k podkladu desky. Proto využíváme při pájení kalafunu, která funguje jako odmašťovač. Pájkou s cínem ponoříme do kalafuny a zkusíme součástku znovu připájet. Pozor si musíme dát na přehřátí součástek. Pokud se pájení nedaří, počkat na zchladnutí součástky, přidat více kalafuny a zkusit to znovu.

7.2 Arduino

Pro výrobu robota je využit vývojový kit Arduino, který dovoluje propojení s PC a připojení a ovládání periferií. Tvoří základní obvod celého návrhu robota. Pro robota je použit modul klonu Arduino Uno. Vyznačuje se omezeným počtem pinů a velikostí, což je pro návrh robota dostačující. Má 14 digitálních pinů, 6 analogových pinů, piny pro zem, napájení 5V a 3,3V, připojení USB a vstup pro napájení. Pro program jsou využity piny 2 až 9 a 12. Piny 0 a 1 jsou vyhrazené a piny 10 a 11 jsou nefunkční. Program je nahráván pomocí USB konektoru přímo z PC. Psán je v softwaru vyvinutém Arduinem v jazyce Wiring. Při psaní je důležité brát v potaz vykonávání programu jako posloupnost příkazů. V programu lze nastavit, jaké piny budou programem využity pro vstup, nebo výstup.

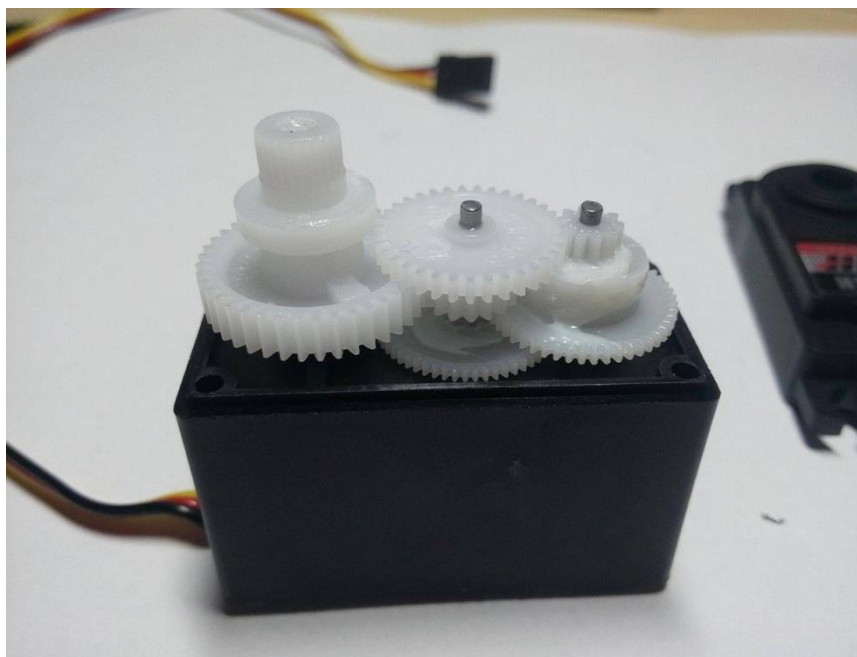


Obr. 12. Základní deska Arduino UNO [23]

7.3 Servomotory

Pro návrh robota jsou využity tři modelářské servomotory HS-311. Napájecí napětí je od 4,8 do 6V. Při menším napětí klesá výkon motoru a jeho rychlost. Při větším napětí je poté přetěžována elektronika servomotoru a při delším vystavování vyšších napětí by mohlo dojít ke zničení elektroniky. Neupravené servo má 3 vstupy. Dva slouží pro napájení a jeden slouží pro řízení natočení serva. Pomocí impulsů přesně natočíme servo do požadované polohy, čehož bylo využito při ovládání přední nápravy. Danou polohu si po celou dobu připojeného napětí udržuje.

Pro pohon jsou serva mechanicky i elektronicky upravené. V základním stavu servo nedokáže kvůli mechanickým dorazům a potenciometru otočení o více než 180°. Elektronika byla spolu s potenciometrem vyjmuta a nahrazena vlastní řídicí jednotkou. Mechanický doraz byl poté odstraněn a na výstup motorků připájeny vlastní vodiče. Na obrázku 13 je vidět u levého spodního kolečka. Vznikly tak dva stejnosměrné zpřevodované motory, které reagují na přiložené napětí.



Obr. 13. Pohled na převodovku servomotoru H-311

7.4 Jednotka pro ovládání motorů

Jelikož výstup Arduina nedokázal uspokojivě napájet zatěžované motory, byla pro napájení vyrobena dvojice upravených H můstků, které spínají motory napětím z baterie. Pro funkční zkoušky je možné přepojit konektory pro napájení na 5V z Arduina. Motory

ztratí na výkonu, ale při špatně naprogramovaných vstupech neodejde řídicí elektronika. Celý obvod byl upraven na velikost odpovídající rozteči děr odpovídající dílku v Merkuru 4x4. V principu funguje obvod jako klasický H-můstek. V obvodu je řešený tranzistory BC 639, diodami a rezistory. Klasický H-můstek nedovoloval přivést napětí na motorky větší, než přibližně 3,4 V. Napětí se vždy dorovnálo na 5V připojených na bázi zmenšené o úbytky na tranzistorech a diodě. Toto bylo vyřešeno přidáním dalších tranzistorů BC548. Zde bude popsán princip při přiloženém napětí na vstup jedné strany můstku. Schéma zapojení je na obrázku 14.

Při přiloženém napětí na vstup A je otevřen tranzistor T5. Všechny proud ze zdroje projde přes rezistor R4 a tranzistor T5, kde se uzemní. Úbytek napětí se rovná součtu napětí procházející přes tyto součástky. Ze vzorce $I = \frac{U}{R}$ se může při napětí baterie 7,4V a odporu 240Ω dopočítat ztrátový proud. Odpor tranzistoru se vzhledem k poměru odporu s rezistorem zanedbá. Ztrátový proud poté odpovídá 31 mA.

Pokud se na vstup A nepřivede napětí, tranzistor se zavře a přes odpor R4 projde proud na bázi tranzistoru T1. Z něj prochází proud na vstup 2 motoru a přes odpor R6 na bázi tranzistoru T4. Ten se otevře a motor se přes tranzistor uzemní.

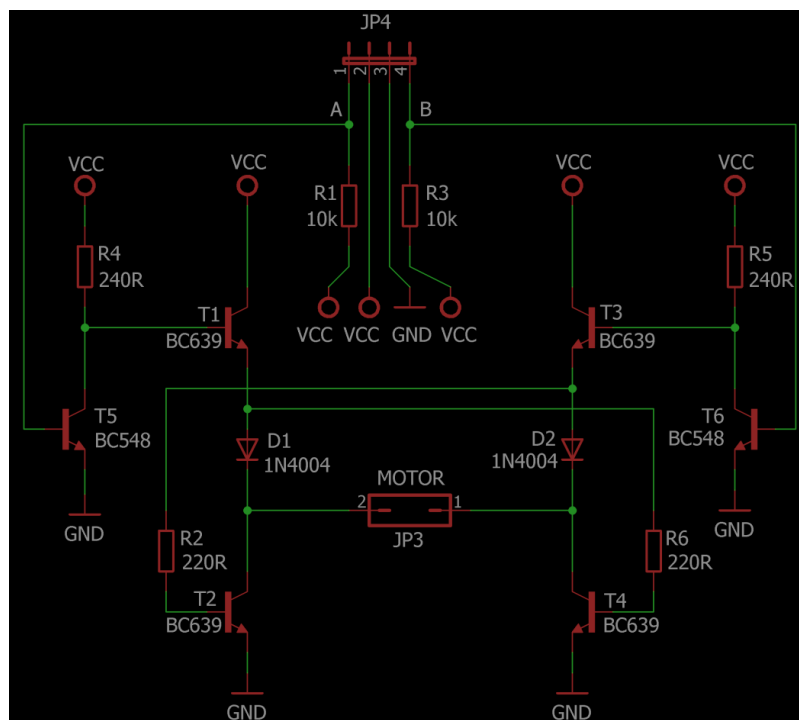
Obdobným způsobem probíhá otevírání tranzistorů na větvi B.

Tab. 1. Stav motoru pro vstupy A a B

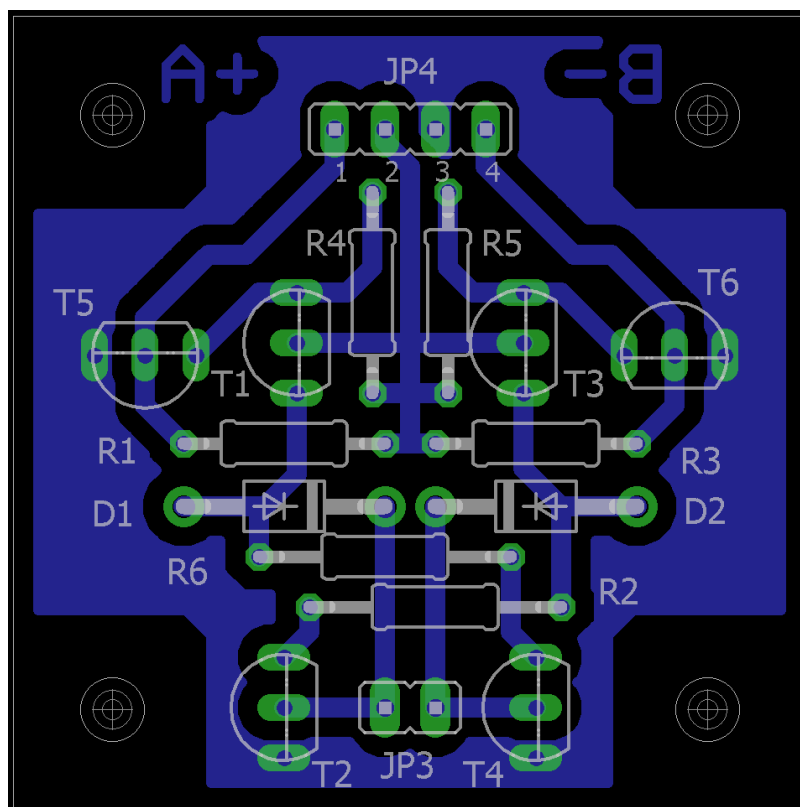
Vstup A	Vstup B	Stav motoru
0	0	zkrat
1	0	jízda vzad
0	1	jízda vpřed
1	1	zastaveno

Jak již z tabulky 1 můžeme vidět, kritická chvíle nastává, pokud jsou obě větve otevřeny. Poté dochází ke zkratu obvodu a hlavně při napájení napojeném na baterku reálně hrozí odpálení tranzistorů na desce. Pro eliminaci zkratu byly do obvodu navrženy rezistory R1 a R3, které by přidržely otevřené tranzistory po dobu startovací sekvence mikrokontroleru. Avšak ovlivňovali značně klasickou funkčnost spínání můstku přidržováním napětí i v běžném stavu, proto byly z obvodu odebrány. Pro vyřešení tohoto stavu by bylo potřeba přidat další tranzistory, které by přidržovali napětí na vstupech. Vyžadovalo by to další proměnnou v programu a zásah do plošného spoje.

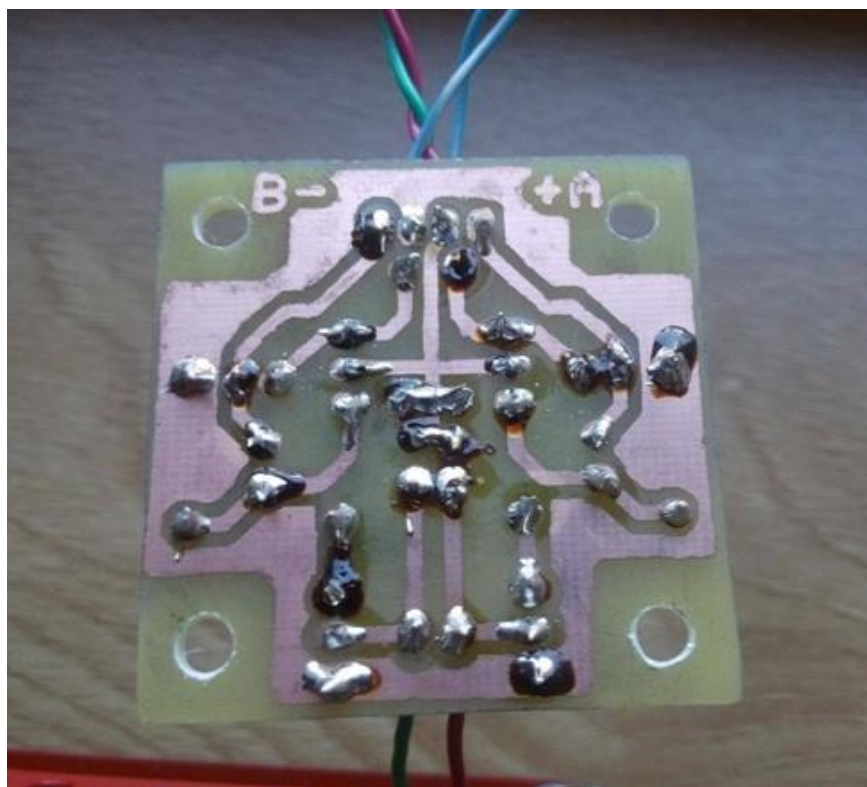
Na obrázku 15 můžeme vidět rozložení součástek na desce a na obrázku 16 vyleptanou a osazenou desku.



Obr. 14. Schéma zapojení ovládací jednotky pro motory



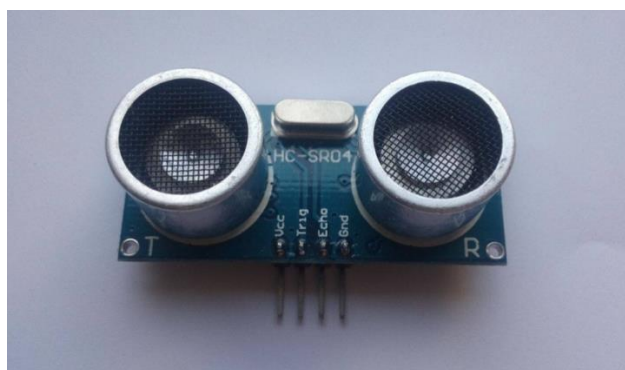
Obr. 15. Návrh součástek na plošném spoji včetně vodivých cest



Obr. 16. Pohled na spodní stranu osazeného plošného spoje

7.5 Ultrazvukový senzor

Pro měření vzdálenosti od překážek je využita dvojice ultrazvukových senzorů HC-SR04. Pracovní napájení je 5V, což odpovídá výstupu napájení z Arduina. Měřit se dá reálně do 3,75 metru. Také má problém s hladkým povrchem odrážející signál od detektoru. Udávaný rozptyl signálu je 15°. Těmto vlastnostem musí být uzpůsobeno natočení detektorů, aby správně vyhodnocovaly překážky. Má dva vstupy napájecí, vstup trigeru a echa. Pin trig slouží pro vysílání signálu a echo slouží pro přijímání.



Obr. 17. Ultrazvukový senzor HS-311

7.6 Baterie

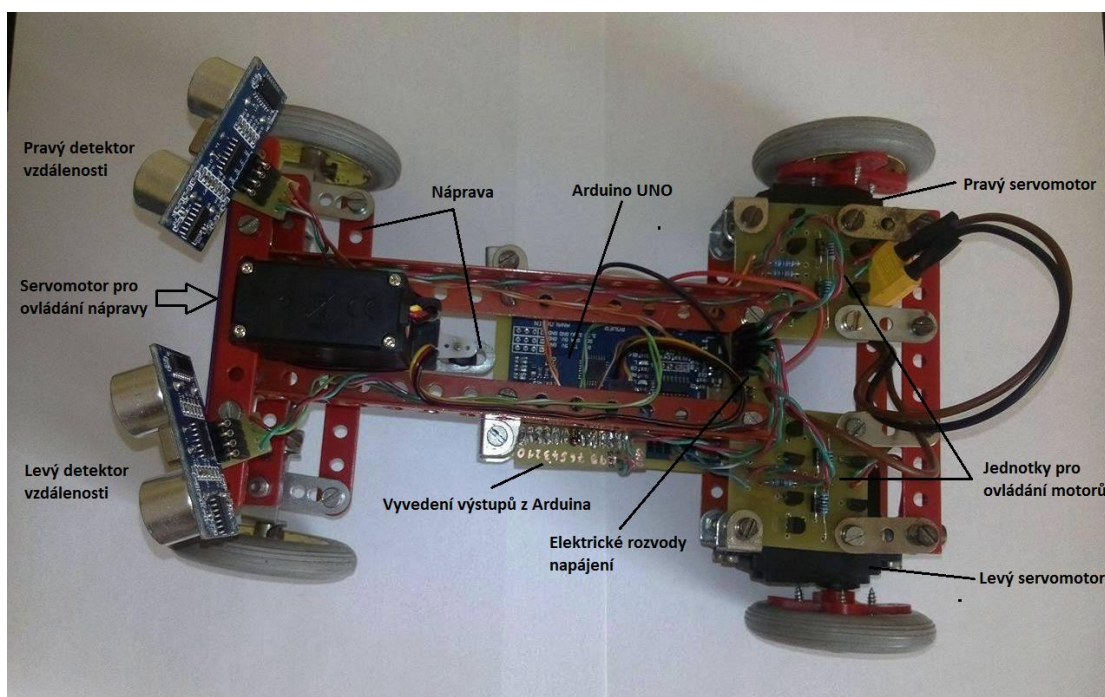
Pro napájení mobilního robota se využívají baterie, akumulátory a jakékoliv další přenosné nebo převozná zdroje elektřiny. Důraz je kladen na hmotnost, rozměry a kapacitu baterie. Pro napájení robota je využita dvoučládková Li-Pol baterie Turnigy 2S o kapacitě 1300mAh. Baterie dodává napětí od 8,2 V při plném nabití do spodní hranice 5,4 V. Pod hranicí tohoto napětí dochází k nenávratnému zničení baterie. S celým rozsahem napětí dodávaným baterií si umí stabilizátor napětí poradit. Rozměry baterie jsou 75 x 35 x 15 milimetru a váha 75 gramů. Konektor nedovoluje přímé připojení k Arduinu, tudíž bylo nutno vytvořit redukci. Vydrží napájet robota po dobu několika hodin.

8 SESTAVENÍ PROTOTYPU ROBOTA

V této části uvedeme celkový pohled na robota z již dříve popsanych součástí robota. Jako hlavní konstrukční prvek byla použita stavebnice Merkur. Ta vytváří oporu celého robota a ostatní součásti byly vytvořeny pro uchycení na tuto konstrukci. Plošné desky byly navrženy s ohledem na uchycení v předem daných roztečích děr. V následujících kapitolách si uvedeme konstrukci robota jak z technického provedení, tak elektrických napojení a rozvodů v robotu.

8.1 Sestava robota

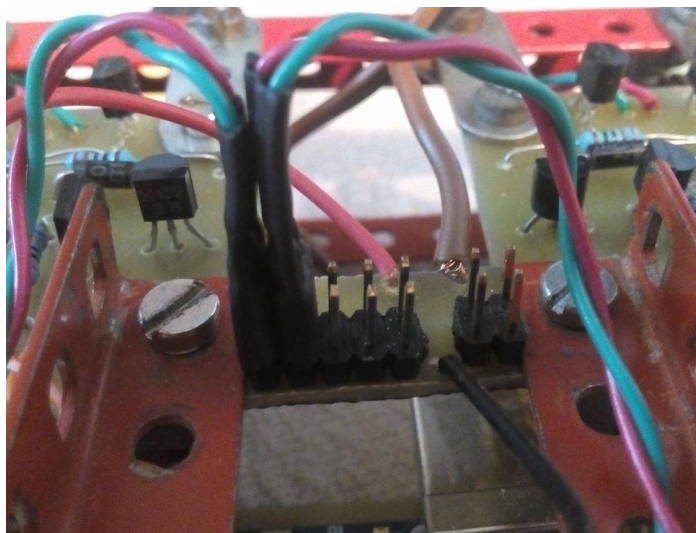
Mimo již zmiňované stavebnice Merkur uvedeme umístění a rozpis ostatních částí robota. Na obrázku 18 můžeme vidět rozmístění jednotlivých komponent. O pohon celého robota se starají upravené servomotory s řídicí jednotkou. Mezi těmito motory je vyvedení napájení Arduina a USB kabel pro propojení s PC. Také se ve vrchní části mezi motory nachází rozvod elektřiny pro všechny součásti robota. Uprostřed je poté uchycena vlastní deska Arduina s vsunutou deskou s konektory, pomocí nichž je vyveden výstup pinů k ostatním součástkám. Tuto desku je možné vyjmout. V přední části se nacházejí dva ultrazvukové detektory vzdálenosti a přední náprava robota, která je tvořena pomocí táhla a servomotoru. Podle stupně otočení serva se vytočí i kola.



Obr. 18. Popis stavby robota

8.2 Elektrické zapojení robota

Pro to, aby robot mohl fungovat, je potřeba propojit soustavou vodičů všechny jeho komponenty. Napájecí vodiče jsou vyvedeny všechny ve formě konektoru, které se dají připojit, či odpojit. Slouží k tomu napájecí svorkovnice 18. Levých 5 dvojic pinů je napájeno 5V z výstupu Arduina. Právě 2 konektory slouží pro napájení motorů, a napájení je vyvedeno přímo z baterie. Na tyto konektory lze motory přepojit pouze tehdy, kdy je program odzkoušen, jinak hrozí spálení součástek. Zem je vyvedena na straně ke středu robota, na stejné úrovni, kde je vyveden černý vodič pro zem. Červený tenký drát je poté napájení 5V z výstupu Arduina. Tlustý červený vodič je vyveden na napájení baterie. Černý a tenký červený vodič pro napětí jsou opatřeny pin konektory, které se zasunují do příslušné zdířky na desce Arduina. Lze tedy snadno napájení vypojit, aniž by se vytahovaly konektory pro jednotlivé součásti robota.



Obr. 19. Napájecí svorkovnice

V tabulce 2 jsou uvedeny barvy jednotlivých vodičů použitých při stavbě robota. Ty jsou ze součástek vyvedeny na desku osazenou piny, které se zasunou do digitálních vstupů a výstupů na Arduinu. Jejich uspořádání nedovolí špatné zasunutí do jiné části než k tomu určené. S datovými vodiči se kromě zasunutí celé osazené desky nijak nemanipuluje. Nemůže tedy dojít k prohození datových vodičů. K rozpoznání zapojení datových vodičů součástek do pinů na Arduinu slouží tabulka 3 popisující uspořádání těchto vodičů na desce Arduina.

Tab. 2. Rozpis vodičů dle barev

	VCC	GND	vstup A	vstup B
Levá jednotka ovládání motoru	rudá	zelená	modrá	modro-bílá
Pravá jednotka ovládání motoru	rudá	zelená	oranžová	oranžovo-bílá
	VCC	GND	Trig	Echo
Levý senzor	rudá	zelená	zelený	zeleno-bílý
Pravý senzor	rudá	zelená	červený	červeno-bílý
	VCC	GND	Ovládání	
Servomotor	Červená/rudá	Černá/zelená	žlutá	

Tab. 3. Rozpis zapojení pinů

	Vstup	Pin	Vstup	Pin
Levá jednotka ovládání motoru	A	2	B	3
Pravá jednotka ovládání motoru	A	4	B	5
Levý senzor	Trig	6	Echo	7
Pravý senzor	Trig	8	Echo	9
Servomotor	Ovládání	12		

9 PROGRAMOVÉ VYBAVENÍ ROBOTA

Tato kapitola bude pojednávat o programování mobilního robota a jeho jednotlivých částí. Pro pochopení celkového programového vybavení je nejdříve důležité pochopit princip jeho dílčích kroků. Teprve pak je možné se pouštět do kombinování více součástí a vytvořit tím celý kompaktní program. Všechny ukázky kódu v této kapitole berou v potaz hardwarové zapojení jednotlivých komponent.

9.1 Na úvod

První věc než se student pustí do programování robota, je důležité udělat několik dílčích kroků. Prvotním je si uvědomit a brát v potaz konstrukční a hardwarové omezení robota a za druhé správně připojit samotného robota k PC. Těmto problematikám budou věnovány následující kapitoly.

9.1.1 Důležité informace

Zapojení napájení je provedeno přes soustavu pinů, kde je velmi důležité brát v úvahu polaritu napájení. Při prohození napájecího konektoru je velmi reálné, že dojde ke zničení součástek. Proto je důležité kontrolovat, zda je konektor zapojen správně.

Připojení motorů je vždy důležité při programování nechat na nižším napětí. Přepojení na napětí baterie je možné až tehdy, pokud je program otestován po všech směrech a nedochází k přehřívání některého z tranzistorů. Při zkratu dochází k jejich zahřívání. Při napájení z Arduina trvá velmi dlouho, než se tranzistory přehřejí a zničí. Napětím z baterie je to otázka několika desítek sekund. Pokud ovládání motorů není naprogramované, je vhodné ponechat konektory vypojené, případně je ručně vypojit.

Pokud je Arduino připojeno k PC a není vyžadována mobilita, je důležité vypojit baterii, aby nedocházelo k jejímu zbytečnému vybíjení.

9.1.2 Propojení s PC

Propojení je realizováno pomocí USB konektoru na jakýkoliv USB port v PC. Před začátkem samotného programování je potřeba nainstalovat ovladač, který zařizuje komunikaci mezi daným klonem Arduina a softwarem přes který se Arduino programuje. Tento ovladač najdeme pod názvem *ch341ser.exe* a je přiložen jako příloha k bakalářské práci. Software pro programování lze volně stáhnout ze stránek Arduina [12]. V programu je důležité nastavit, na kterém portu se Arduino nachází. Vše potřebné pro nastavení je pod

položkou nástroje. Také se tam nachází výběr vývojové desky a procesoru. Zde je nutné nastavit *Arduino Mini* a procesor *ATmega328*.

9.2 Programování jednotlivých částí robota

Tato část se bude věnovat programovému vybavení robota. Přiblíží jednotlivé součásti robota z programového hlediska a uvede základní kód, který bude okomentován pro snadné pochopení studentem. Všechny nadefinované piny se řídí podle tabulky 3 rozpisu zapojení pinů. V průběhu bude kód na sebe navazovat, tím je myšleno to, že nastavení vstupů, deklarace proměnných a další příkazy budou vysvětleny jen jednou. Celkové ukázky programů včetně okomentování všech důležitých řádků jsou uvedeny v příloze.

9.2.1 LED dioda

Pro prvotní programování si zkusíme naprogramovat LED diodu. V návrhu robota není žádná zapojená LED dioda vyvedena, využijeme tedy vestavěnou diodu přímo na desce Arduina, která je připojena na výstup 13 pinu. Deklarace proměnné *dioda* datového typu integer připojené na pin 13 provedeme následovně:

```
int dioda=13;
```

V nastavení vstupů a výstupů realizováno tělem `setup()` nastavíme pin 13 jako výstup. Realizovat to můžeme následovně:

```
void setup() {  
  pinMode(dioda,OUTPUT);  
}
```

Pokud chceme nastavit daný pin jako vstupní, pak místo OUTPUT napíšeme INPUT. Rozsvícení diody poté realizujeme v hlavním těle programu `loop()`, kde příkazem

```
digitalWrite(dioda,HIGH);
```

přivedeme na výstup napětí. Zhasnutí diody se provede zapsáním LOW jako druhého parametru příkazu.

9.2.2 Ultrazvukový senzor

Ultrazvukový senzor se dá využít pro měření vzdálenosti a sledování okolí robota. Jeho přesnost je od 5 do 350 centimetrů postačující. Rozdíl je okolo půl centimetru u blízkých

předmětů. Dobře reaguje na pevné překážky (například zeď, skříň), velmi špatně na měkké překážky, které snadno utlumí zvuk (peřina, deka).

Nejprve si pomoci příkazu

```
#define ECHO 7
```

nadefinujeme piny echo a trig na příslušné piny uvedené v tabulce 3. Dále nadefinujeme proměnné *vzdalenost* a *konstanta* datového typu float.

V nastavovací části pomocí příkazu `Serial.begin(9600);` nastavíme rychlost komunikace po sériové sběrnici. Dále nastavíme pin echo jako vstupní a trig jako výstupní.

V těle programu vyšleme pomocí následujících příkazů impuls, který vygeneruje ultrazvukový signál.

```
digitalWrite(TRIG,LOW);
```

```
delayMicroseconds(2);
```

```
digitalWrite(TRIG,HIGH);
```

```
delayMicroseconds(10);
```

```
digitalWrite(TRIG,LOW);
```

Výpočet konstanty pro převod na centimetry a funkce `pulseIn`

Funkce `pulseIn` vrací počet mikrosekund od vyslání impulsu do jeho příjmu. Uvažujme rychlost zvuku $346,3 \text{ m} \cdot \text{s}^{-1}$ v suchém vzduchu, při teplotě 25°C . Rychlost zvuku převedeme na metry za mikrosekundu a vypočítáme uraženou vzdálenost. Výsledkem je poté $346,3 \cdot 10^{-6}$ metru, což odpovídá $3,463 \cdot 10^{-4}$ m. Převedeno na centimetry $3,463 \cdot 10^{-2}$ cm. Tato konstanta se podělí ještě 2, jelikož signál putuje k překážce a zpátky. Výsledná konstanta 0,017315 udává vzdálenost, kterou urazí zvuk za jednu mikrosekundu v centimetrech. Nyní stačí konstantu vynásobit s počtem mikrosekund a dostaneme výsledek. Realizováno je to následujícími příkazy.

```
vzdalenost = pulseIn(ECHOL,HIGH);
```

```
vzdalenost = vzdalenost*0.017315;
```

Nyní si pomocí příkazu `Serial.print()` vypíšeme proměnnou na sériový monitor a zde můžeme sledovat vzdálenost překážky od senzoru.

9.2.3 Ovládání jednotek motorů

Nastavení pinů se řídí podle tabulky 3, také musíme dbát na nastavení pinů popsané v tabulce 1 tak, aby nedošlo ke zkratu. Ujistíme se, že máme napájecí konektor zasunutý správnou polaritou na napětí 5V z Arduina.

Prvním krokem si nadefinujeme příslušné proměnné pro jednotlivé jednotky a nastavíme všechny proměnné jako výstupní piny. Hlavní tělo programu může vypadat následovně.

```
vpred();  
delay(5000);  
vzad();  
delay(3000);
```

Pro zjednodušení a zpřehlednění hlavního kódu si vytvoříme podprogram *vpred()* a *vzad()*. Zapišeme je následovně:

```
void vpred()  
{ }
```

V závorce je nastavení jednotlivých vstupů pro jízdu vpřed podle tabulky 1. Po vykonání podprogramu se procesor vrátí na místo volání a pokračuje v následujících instrukcích.

9.2.4 Ovládání servomotoru

Servomotor použitý na mobilním robotu slouží k otáčení táhla nápravy, a tím pádem vytáčení kol. Nejprve si v deklarační části načteme knihovnu pro ovládání serva a nadefinujeme proměnnou pro servo s názvem *zatacServo*.

```
#include <Servo.h>  
Servo zatacServo;
```

V části pro nastavení pinů přiřadíme pin pro ovládání serva.

```
zatacServo.attach(12);
```

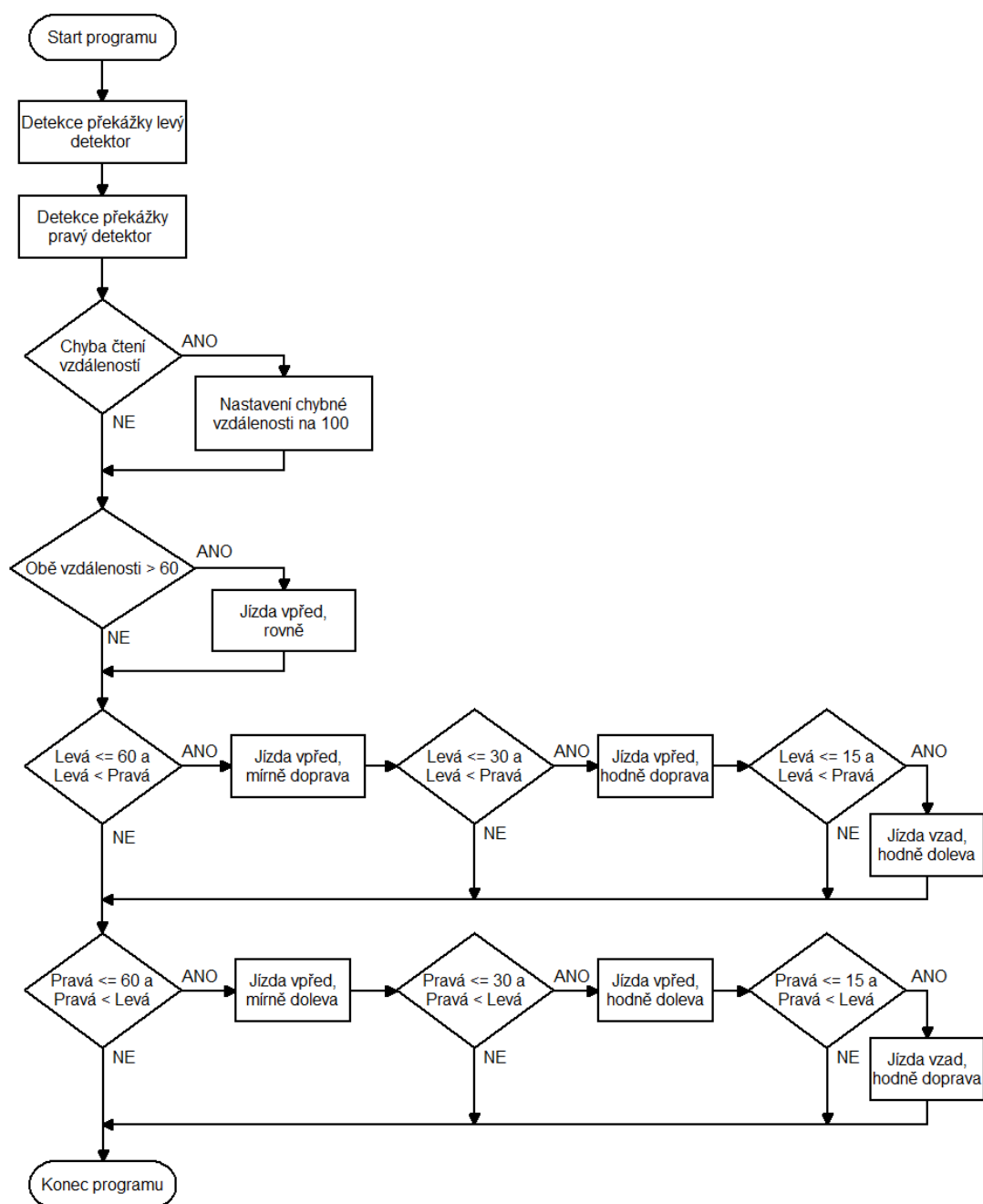
V hlavní části programu můžeme měnit úhel otočení serva zapsáním hodnoty do příkazu uvedeném na dalším řádku. Pro úhel 90° by se příkaz zapsal následovně:

```
zatacServo.write(90);
```

9.3 Kompletace a struktura ukázkového programu robota

Kompletace programu je proces, kde je jeho hlavním cílem propojit všechny dílčí části programu do celku, který vykonává požadovanou činnost. V našem případě je to ovládání pohybu robota v místnosti, detekce a vyhnutí se překážek. Zde je důležité propojit všechny úseky programu z minulé kapitoly v jeden celek, bez toho aniž by se části nepříznivě ovlivňovaly.

Pomocí následujícího vývojového diagramu na obrázku 20 bude objasněna funkce a logika ovládacího programu robota. Procesor čte instrukce programu jako posloupnost jednotlivých operací. V případě, že se dostane na konec programu, začne se vykonávat program zase od jeho začátku. Na úplném začátku programu jsou vyhodnoceny vzdálenosti obou detektorů, jejich hodnoty přepočteny na centimetry a uloženy. Všechny následující podmínky využívají hodnot v centimetrech. V případě splnění podmínky je uložena informace pro otočení nápravy a vyvolán podprogram, který řídí jízdu vpřed, nebo vzad. V případě splnění více podmínek za sebou, podmínka na nižší úrovni (na obrázku více vpravo) přepíše hodnoty zapsané předchozí splněnou podmínkou. Jelikož jízda vpřed nemá žádnou prodlevu, při jízdě vpřed je vyhodnocení rychlé. Při jízdě vzad je poté na chvíli program přerušen, aby se stihlo vozítko otočit od stěny, případně vycouvat z rohu. Celý ukázkový program je nahraný včetně komentářů na přiloženém CD.



Obr. 20. Vývojový diagram ukázkového programového vybavení robota

ZÁVĚR

Cílem této práce je seznámit studenty s programováním mikropočítačů a programování zpopularizovat, aby se v jejich řadách stalo oblíbenějším. K tomuto účelu byl vytvořen mobilní robot, kterého si může kdokoli naprogramovat. V teoretické části je uvedeno množství prostředků, s jejichž pomocí je možno mobilní robot sestavit. Pro výukové účely jsou vhodné takové roboty, které jsou levné a snadno dostupné. Z uvedených druhů robotů nejsou například vhodné průmyslové roboty, kvůli jejich složitosti a ceně. Jsou představeny nejčastěji používané podvozky, motory pohánějící robota a způsoby detekce okolí robota. Po pročtení funkce jednotlivých součástí je možné navrhnout mobilního robota s ohlédnutím na jeho požadovanou funkci.

Praktická část je zaměřena na návrh a stavbu vlastního robota, který slouží k popularizaci programování. Pro tento účel byl navržen a vytvořen mobilní robot, který dokáže detekovat překážku, a té se vyhnout. Jako mozek robota slouží mikropočítačová platforma Arduino. Díky její kompatibilitě s množstvím připojitelných zařízení byla vybrána pro řízení robota. Detekci překážek zajišťuje dvojice ultrazvukových senzorů, které jsou umístěny v přední části robota, a každá je vytočena mírně od středu osy robota. Tím je zajištěna detekce překážek blížících se jak z boku robota, tak překážek blížících se zepředu. Pohyb robota zajišťuje dvojice upravených servomotorů. Jejich ovládání je zajištěno přes vytvořené řídicí jednotky, které umožňují přivést na motory daleko větší napětí a proud, než dovoluje výstup z Arduina. Jejich schéma zapojení, návrh plošného spoje a jeho výroba je zdokumentována v bakalářské práci. Mimo jiné popisuje funkci těchto jednotek včetně průchodu proudu obvodem při možných stavech na vstupech jednotky. Programování probíhá v přívětivém softwarovém prostředí a je velmi snadné na pochopení. V příloze je poté ukázkové programové vybavení robota včetně vysvětlujících komentářů.

SEZNAM POUŽITÉ LITERATURY

- [1] CHMELÍK, Vladimír. *Mobilní kolový robot* [online]. Praha, 2015 [cit. 2016-03-26]. Dostupné z: http://www1.fs.cvut.cz/stretech/2015/sbornik_2015/1640.pdf
- [2] KONFRŠT, Pavel. *Všesměrový podvozek robota se servopohony a základní odometrií* [online]. Pardubice, 2012 [cit. 2016-03-27]. Dostupné z: https://dk.upce.cz/bitstream/handle/10195/48625/KonfrstP_VsesmerovyPodvozek_PR_2012.pdf;jsessionid=E9CB3238426F89ECEB22E058683DF156?sequence=2H. Bakalářská práce.
- [3] NEŘÁD, Karel. *Robot sledující čáru*. Zlín, 2011. Bakalářská práce. Univerzita Tomáše Bati. Vedoucí práce Doc. Mgr. Milan Adámek, Ph.D.
- [4] JAVŮREK, Karel. Honda Asimo, legendární robot s novými dovednostmi. In: *VTM.cz* [online]. [cit. 2016-04-05]. Dostupné z: <http://vtm.e15.cz/honda-asimo-legendarni-robot-s-novymi-dovednostmi>
- [5] NOVÁK, Petr. *Mobilní roboty: pohony, senzory, řízení*. Praha: BEN - technická literatura, 2005. ISBN 80-730-0141-1.
- [6] Mechanické kmitání a vlnění: Rychlost zvuku. *FYZIKA 007*[online]. [cit. 2016-04-05]. Dostupné z: <http://www.fyzika007.cz/mechanicke-kmitani-a-vlneni/rychlost-zvuku>
- [7] ZÍTKA, Michal. *Detekce pohybu v obraze* [online]. Brno, 2008 [cit. 2016-04-18]. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=8263. Bakalářská práce. Vysoké učení technické v Brně.
- [8] PINKER, Jiří. *Mikroprocesory a mikropočítače*. Praha: BEN - technická literatura, 2004. ISBN 80-730-0110-1.
- [9] USB 3.0. *SECOMP PC Plus CZ* [online]. [cit. 2016-04-22]. Dostupné z: <https://www.pcplus.cz/zkratky.asp?ZKR=72>
- [10] MIEKISCH, Jiří. *Robotika* [online]. Opava [cit. 2016-04-22]. Dostupné z: http://www.strojka.opava.cz/UserFiles/File/_sablon/VY_32_INOVACE_E-15-20.pdf

- [11] HSES: *Vývojové moduly a moduly pro robotické stavebnice* [online]. [cit. 2016-04-22]. Dostupné z: <http://www.hses.cz/informace-o-systemech/vyvojove-moduly-a-moduly-pro-roboticke-stavebnice.html>
- [12] *Arduino.cz: Webový magazín o Arduinu a elektronice* [online]. [cit. 2016-04-22]. Dostupné z: <http://arduino.cz/>
- [13] DUDÁČEK, K. *Mikrokontroléry* [online]. Plzeň, 2001 [cit. 2016-04-25]. Dostupné z: <http://home.zcu.cz/~dudacek/Pot/mikrokontrolery.pdf>.
- [14] HANULÍK, Stanislav. *Mikroprocesory*. Zlín, 2013. Střední průmyslová škola.
- [15] HANULÍK, Stanislav. *Vnitřní paměti*. Zlín, 2012. Střední průmyslová škola.
- [16] *Papouch.com: Sériový port RS232* [online]. [cit. 2016-05-12]. Dostupné z: <http://www.papouch.com/cz/website/mainmenu/clanky/jak-na-to/rs232/>
- [17] Průmyslový robot Motoman ROBOTECH [obrázek]. *Hadyna* [online]. [cit. 2016-04-22]. Dostupné z: http://www.hadyna.cz/smartwelding/motoman/fist_robot.html
- [18] Ackermanova podmínka [obrázek]. *Autolexicon.net* [online]. [cit. 2016-03-27]. Dostupné z: <http://www.autolexicon.net/cs/articles/ackermannova-podminka/>
- [19] Prostředky relativní lokalizace [obrázek]. *Marek.sk.sweb.cz* [online]. [cit. 2016-04-22]. Dostupné z: <http://marek.sk.sweb.cz/lokalizace/kapitola3.html>
- [20] DOLEJŠÍ, Pavel. *Studie a návrh všesměrového kola invalidního vozíku se zvýšenou mobilitou* [obrázek] [online]. [cit. 2016-04-24]. Dostupné z: <http://partnerstvi-energetiky.msek.cz/wp-content/uploads/zaverecne-zpravy/vsb-tu-ostava/dolejsi.pdf>
- [21] Roboty [obrázek]. *HOBBYROBOT* [online]. [cit. 2016-04-24]. Dostupné z: <http://www.hobbyrobot.cz/roboty/>
- [22] Serva [obrázek]. *Rc-auta.eu* [online]. [cit. 2016-05-09]. Dostupné z: <http://www.rc-auta.eu/uzitecne-odkazy/93-serva>
- [23] Klon Arduino UNO [obrázek]. *Maker.robotistan.com* [online]. [cit. 2016-05-22]. Dostupné z: <http://maker.robotistan.com/arduino-uno-suruculeri-nasil-yuklenir-ch340-cipli-klon/>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ALU	Aritmeticko – logická jednotka
A/D	Analogově digitální převodník
CNC	Computer Numeric Control
CISC	Complete Instruction Set Computer
EPROM	Erasable Programmable Read-Only Memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
GND	Zem
IR	Infra red
LED	Light-Emitting Diode
PC	Osobní počítač
PROM	Programmable Read-Only Memory
RAM	Random Acces Memory
RISC	Reduced Instruction Set Computer
ROM	Read-Only Memory
USB	Univerzální sériová sběrnice
VCC	Napájení

SEZNAM OBRÁZKŮ

Obr. 1. Průmyslový robot Motoman ROBOTECH [17].....	18
Obr. 2 Ackermanův podvozek [18]	20
Obr. 3. Stanfordské kolo [19]	21
Obr. 4. Mecanumské kolo [20]	22
Obr. 5. Pásový odvozek [21]	23
Obr. 6. Nejnovější generace robotů Asimo [4].....	24
Obr. 7. Kotouček pro inkrementální senzor [5]	26
Obr. 8. Kódové kotouče absolutního senzoru natočení [5].....	26
Obr. 9. Průřez servomotorem [22]	30
Obr. 10. Přenesené schéma na cuprexit	34
Obr. 11. Vyleptaný plošný spoj	34
Obr. 12. Základní deska Arduino UNO [23]	35
Obr. 13. Pohled na převodovku servomotoru H-311	36
Obr. 14. Schéma zapojení ovládací jednotky pro motory.....	38
Obr. 15. Návrh součástek na plošném spoji včetně vodivých cest	38
Obr. 16. Pohled na spodní stranu osazeného plošného spoje	39
Obr. 17. Ultrazvukový senzor HS-311	39
Obr. 18. Popis stavby robota.....	41
Obr. 19. Napájecí svorkovnice	42
Obr. 20. Vývojový diagram ukázkového programového vybavení robota.....	49

SEZNAM TABULEK

Tab. 1. Stav motoru pro vstupy A a B	37
Tab. 2. Rozpis vodičů dle barev	43
Tab. 3. Rozpis zapojení pinů	43