



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA STROJNÍHO INŽENÝRSTVÍ**  
**ÚSTAV AUTOMOBILNÍHO A DOPRAVNÍHO**  
**INŽENÝRSTVÍ**

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF AUTOMOTIVE ENGINEERING

## **MOBILNÍ ZAŘÍZENÍ PRO POHYB V NEBEZPEČNÉM PROSTŘEDÍ**

MOBILE DEVICE FOR MOVEMENT IN DANGEROUS ENVIRONMENT

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**JAKUB BARTUŠEK**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. et. Ing. MARTIN BILÍK**

BRNO 2015

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automobilního a dopravního inženýrství

Akademický rok: 2014/15

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

student(ka): Jakub Bartušek

který/která studuje v **bakalářském studijním programu**

obor: **Stavba strojů a zařízení (2302R016)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

### **Mobilní zařízení pro pohyb v nebezpečném prostředí**

v anglickém jazyce:

### **Mobile devices for movement in dangerous environment**

Stručná charakteristika problematiky úkolu:

Práce je koncipována jako rešerše v oblasti mobilních zařízení pohybujících se v nebezpečných prostředích.

Cíle bakalářské práce:

Rešerše mobilních zařízení pro pohyb v nebezpečných prostředích.

Porovnání jednotlivých variant.

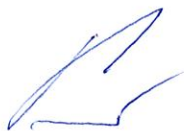
Seznam odborné literatury:

www.infozdroje.cz

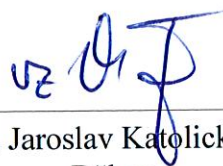
Vedoucí bakalářské práce: Ing. et Ing. Martin Bilík

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2014/15.

V Brně, dne 25.11.2014



prof. Ing. Václav Píštěk, DrSc.  
Ředitel ústavu



doc. Ing. Jaroslav Katolický, Ph.D.  
Děkan



## ABSTRAKT

Práce se zabývá návrhem mobilního robotu pro pohyb v neznámém prostředí. Úvod práce se věnuje řešením v oblasti možností konstrukce mobilních robotů, výběru vhodných typů senzorů, algoritmům pro plánování trasy a řízení robotu. Druhá část práce se zabývá samotnou konstrukcí robotu. Robot je poté sestaven a jsou na něm testovány vybrané algoritmy pro plánování trasy v praxi.

## KLÍČOVÁ SLOVA

Mobilní robot, Differenciální podvozek, Arduino, PWM, BUG algoritmy, Plánování trasy, Mikrokontroler, DC motor, Řízení robotu, Sensory

## ABSTRACT

This thesis deals with design of mobile robot for movement in dangerous environment. The first part of this thesis pays attention on area of mobile robot construction, different types of suitable sensors, motion planning algorithms and robot control. Second part is about construction and design of the robot. Motion planning algorithms are used for this robot and tested in environment.

## KEYWORDS

Mobile robot, Differential chassis, Arduino, PWM, BUG algorithm, Motion control, Microcontrollers, DC motor, Robot control, Sensors



## **BIBLIOGRAFICKÁ CITACE**

BARTUŠEK, J. Mobilní zařízení pro pohyb v nebezpečném prostředí. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2015. 47 s. Vedoucí bakalářské práce Ing. et Ing. Martin Bilík.



## ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. et Ing. Martina Bilíka s použitím literatury uvedené v seznamu.

V Brně dne 29. května 2015

.....

Jméno a přímení



## PODĚKOVÁNÍ

Tímto bych chtěl poděkovat vedoucímu práce panu Ing. et Ing. Martinu Bilíkovi za odborné vedení a ochotný přístup při vypracování bakalářské práce. Dále bych chtěl poděkovat své přítelkyni Lee Mikundové a rodině za podporu ve studiu.



## OBSAH

1	Úvod.....	9
1.1	Cíl práce.....	9
1.2	Dělení robotů.....	10
1.3	Druhy mobilních robotů .....	10
1.4	Příklady robotů .....	11
2	Konstrukční možnosti jednotlivých dílů.....	13
2.1	Druhy podvozků .....	13
2.1.1	Diferenciální podvozek .....	13
2.1.2	Trojkolový podvozek s řízeným předním kolem.....	14
2.1.3	Ackermanův podvozek .....	14
2.2	Motory .....	15
2.2.1	DC motory .....	15
2.2.2	Krokové motory.....	16
2.2.3	Modelářské servomotory .....	16
2.3	Mikrokontroler .....	17
2.4	Senzory .....	17
3	Plánování trasy, Algoritmy.....	18
3.1	Plánování trasy .....	18
3.2	Algoritmy pro plánování cesty .....	18
3.2.1	Informované algoritmy .....	19
3.2.2	Neinformované algoritmy.....	21
4	Mechanická a Elektronická konstrukce .....	25
4.1	Volba použitých součástek.....	25
4.2	Konstrukce robotu .....	26
4.3	Podvozek .....	28
4.4	Matematický model robotu .....	28
4.4.1	Kinematika.....	28
5	Řídicí systém.....	30
5.1	Řídicí deska – Arduino .....	30
5.1.1	Obecně o Arduino .....	30
5.1.2	Architektura Arduina .....	31
5.1.3	Napájení .....	32
5.1.4	Programovací jazyk .....	32
5.1.5	Arduino Příslušenství a Knihovny .....	33
5.2	USB Host Shield .....	34





6	Pohonný systém.....	35
6.1	Motory .....	35
6.2	Řízení motorů.....	35
6.2.1	L293D .....	36
6.2.2	PWM.....	37
7	Senzorický systém .....	38
7.1	Externí senzory .....	38
7.1.1	Ultrazvukový senzor HC-SR04 .....	38
7.2	Interní senzory .....	39
7.2.1	Enkodér .....	39
8	Manuální režim.....	40
9	Autonomní režim .....	41
10	Závěr .....	43
	Seznam zdrojů .....	46
	Seznam obrázků .....	47



# 1 ÚVOD

## 1.1 CÍL PRÁCE

Cílem práce je sestavit jednoduchý mobilní robot ze součástí běžně dostupných na trhu. Základní funkce robota, kterou budeme vyžadovat je manuální ovládání pomocí bezdrátového ovladače.

Dalším cílem je vytvořit a implementovat autonomní režim, ve kterém se bude robot schopný vyhnout překážkám v cestě směrem k cíli. Cílem se rozumí buď předem nastavená pozice, nebo počáteční pozice při ztrátě signálu. Tím je myšlena schopnost robota vrátit se na místo, ze kterého vyjel.

Požadavky, kladené na řešení, jsou následující:

- napájení pomocí baterie.
- malá a snadná konstrukce.
- snadná obsluha a případné úpravy/opravy.
- snadné ovládání, pokud možno bezdrátové.



## 1.2 DĚLENÍ ROBOTŮ

Roboty lze dělit podle několika různých hledisek.

Dělení dle mobility:

- stacionární – nemohou se pohybovat z místa na místo (například průmyslové manipulátory).
- mobilní – mohou se přemisťovat (například vesmírné sondy a vozítka na Marsu).

V dalším textu budou uvedeny pouze mobilní roboty.

## 1.3 DRUHY MOBILNÍCH ROBOTŮ

Roboty lze rozdělit podle několika kritérií, mezi základní dělení patří:[1]

- dálkově řízené – jsou řízeny operátorem, který má informaci (vizuální) o pracovním okolí robotu.
- autonomní – předpokládá se u nich schopnost samostatně řešit zadanou úlohu, např. sledování čáry, schopnost reagovat na překážky v cestě (zastavit, vyhnout se jim), umět se pohybovat v neznámém prostředí.

Dále můžeme dělit podle prostředí, ve kterém se robot může pohybovat:

- na souši – kolová a pásová vozidla.
- ve vodě – průzkumné ponorky, robotické ryby.
- ve vzduchu – kvadrkoptéry, drony, RC modely, robotičtí ptáci.
- ve vesmírném prostoru – servisní roboti, družice, satelity.

Dělení podle účelu nasazení:

- manipulační – pro manipulaci s předměty.
- montážní – jako součást montážních linek.
- servisní – pro opravu a servis nehod na obtížně přístupných místech.
- inspekční – pro kontrolu stavu pozorovaného jevu.
- průzkumné – pro prozkoumávání neznámého prostoru.
- vojenské – pyrotechničtí roboti, drony.
- zdravotnické – lékařské manipulátory, laserové skalpely.
- určené pro zábavu - kvadrkoptéry, drony, RC modely.

Dělení podle typu podvozků:

- kolové.
- pásové.
- kráčející.
- plazivé.
- šplhající.
- hybridní.
- inteligentní.

## 1.4 PŘÍKLADY ROBOTŮ

### ASIMO – Honda

*(Mobilní – Autonomní – Pro pohyb na souši – Chodící – Zdravotnické účely)*

Mezi veřejností se jedná snad o nejznámějšího zástupce robotů. Byl představen v roce 2000. ASIMO byl konstruován, tak, aby se co nejvíce podobal člověku a mohl tak ho tak nahradit v běžném lidském kontaktu a prostředí, jedná se tedy o humanoidní robot. ASIMO tak může sloužit jako společník pro staré nebo těžce nemocné lidi, dokáže reagovat na lidské jednání i řeč. Není vhodný pro použití v náročném prostředí a terénu, tak jako třeba roboty od společnosti Boston Dynamics.

Asimo je vysoký 130cm a váží 50kg. Maximální rychlost jakou se může pohybovat je 5,7 km/h, při běžné chůzi je to 2,7 km/h. Celkově má ASIMO 57 stupňů volnosti (degrees of freedom = DOF)

Zajímavostí je to, že Asima lze ovládat pomocí speciální pokrývky hlavy. Ta vysílá podle myšlenek, které představují snímané elektrické výboje z pokožky hlavy, člověka základní příkazy k pohybu.

Celkově je jeho provoz velmi energeticky náročný, protože pro každý DOF je potřeba aktuátor a jeho interní baterie tak vydrží na hodinu provozu. [12]



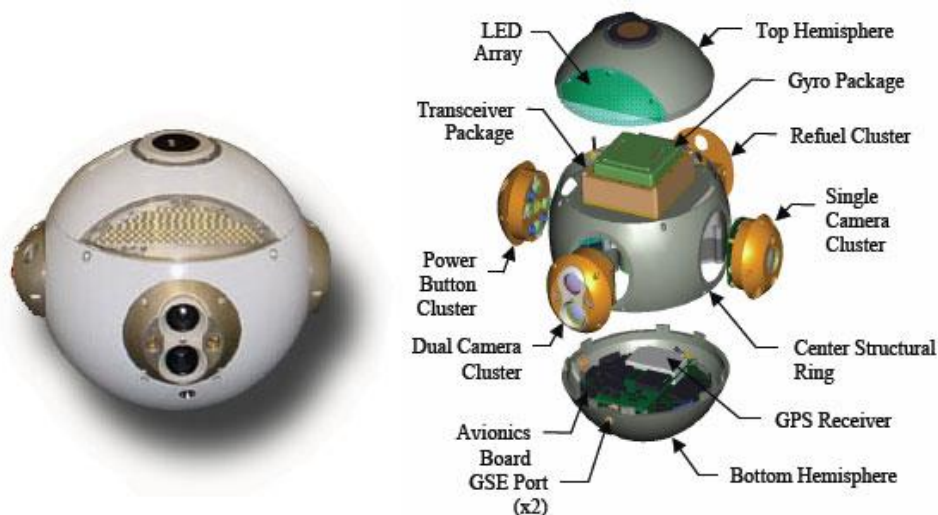
Obr. 1 ASIMO. [12]

### Mini AERCam – NASA

*(Mobilní – Autonomní – Pro pohyb ve vesmíru – Létající – Inspekční / Servisní)*

Tato robotická kamera byla vyvinuta NASA v roce 2005 na kontrolní práce ve vesmírném prostoru. Je to volně létající robot vybaven 12-ti tryskami na stlačený plyn, které umožňují pohyb a natáčení ve všech směrech. V autonomním režimu je schopna se orientovat ve složitém trojrozměrném prostoru.

Má tvar koule o průměru 190mm a váží 4,5kg. [2]



Obr. 2 Mini AERCam. [2]

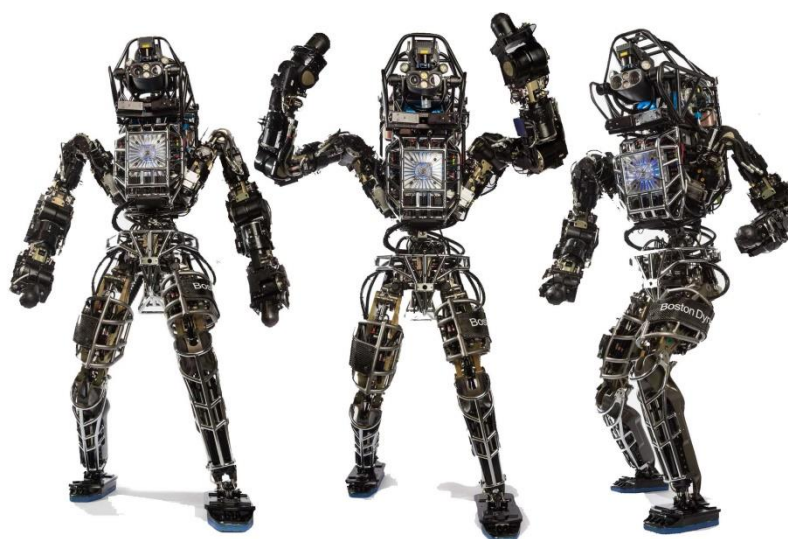
## ATLAS – Boston Dynamics

*(Mobilní – Autonomní – Pro pohyb na souši – Chodící – Multifunkční)*

Atlas je vysoce mobilní humanoidní robot konstruovaný pro náročný terén a provoz firmou Boston Dynamics za podpory projektu DARPA (Defense Advanced Research Projects Agency). Dokáže šplhat, běhat, manipulovat a interagovat s okolním prostředím. Je vysoký 180cm a poprvé byl představen v roce 2013.

Kloubové ruce dovolí ATLASu používat nástroje vytvořené pro použití člověkem. Cílem pro vývoj ATLASu bylo vyvinout robot pro použití v podmínkách, ve kterých nemůže člověk přežít. ATLAS je tak schopen používat nástroje, ovládat páky, pracovat na elektrických přístrojích a mnoha podobných akcí. Je vybaven velkým množstvím senzorů, stereo kamerou, laserovým dálkoměrem. Má 28 DOF.

Napájení je řešeno pomocí kabelu a externího zdroje. [5]



Obr. 3 ATLAS. [5]

## 2 KONSTRUKČNÍ MOŽNOSTI JEDNOTLIVÝCH DÍLŮ

### 2.1 DRUHY PODVOZKŮ

Volba podvozků je první a velmi důležitý problém k řešení. Tento proces je důležitý pro způsob řízení a jeho volba značně ovlivňuje i výsledný řídicí systém robotu.

V následujících kapitolách je provedeno porovnání různých druhů podvozků, které je možné pro projekt využít. Dále porovnáním jejich parametrů, výhod a nevýhod bude zvolen nejvhodnější podvozek pro mobilní robot.

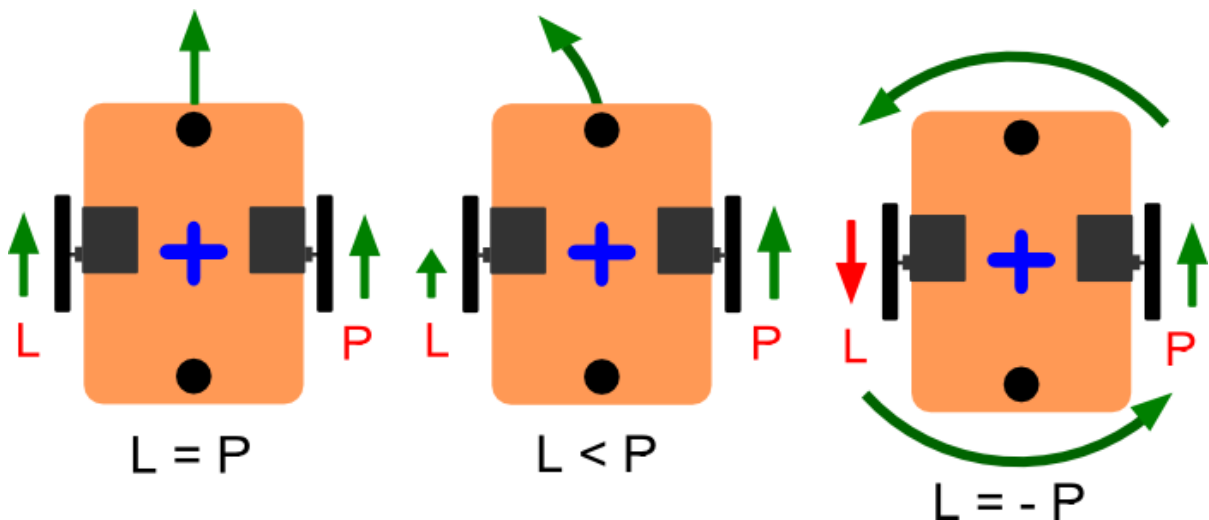
#### 2.1.1 DIFERENCIÁLNÍ PODVOZEK

Jedná se o podvozek, jehož pohyb je závislý na dvou řízených motorech, uložených na opačné straně robotu. Díky tomu lze měnit směr a rychlost pouze pomocí změny otáček na jednotlivých kolech. Z toho je tedy odvozený název „diferenciální“, protože je robot řízen „diferencí“ (rozdílem) mezi otáčkami jednotlivých kol.

Pokud se tedy obě kola točí stejnou rychlostí a stejným směrem, dochází k přímočarému pohybu tím směrem. (viz. Obr. 4 vlevo)

Pokud se obě kola otáčejí stejnou rychlostí ale opačným směrem, dochází k rotaci robotu okolo středu otáčení. (viz. Obr. 4 vpravo)

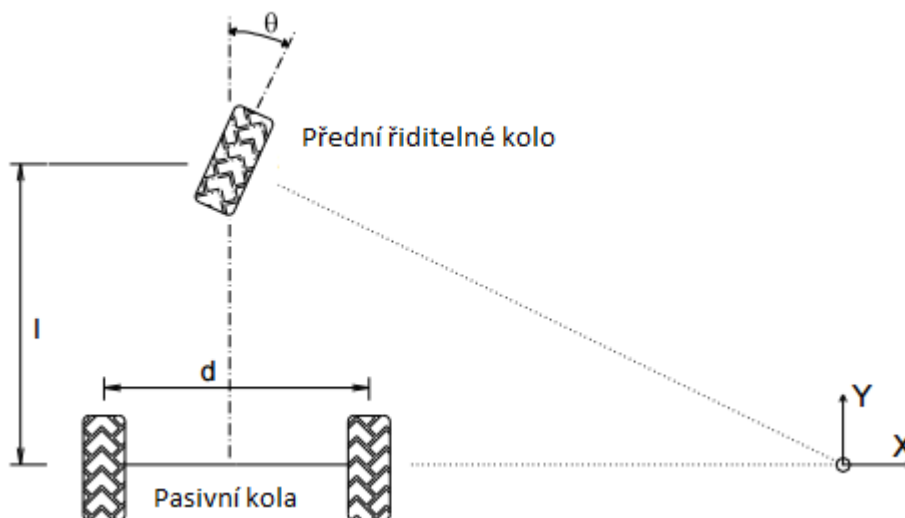
Pokud se kola otáčejí stejným směrem ale jinou rychlostí, robot zatáčí směrem od rychleji se otáčejícího kola. (viz. Obr. 4 uprostřed)



Obr. 4 Schéma diferenciálního řízení.

### 2.1.2 TROJKOLOVÝ PODVOZEK S ŘÍZENÝM PŘEDNÍM KOLEM

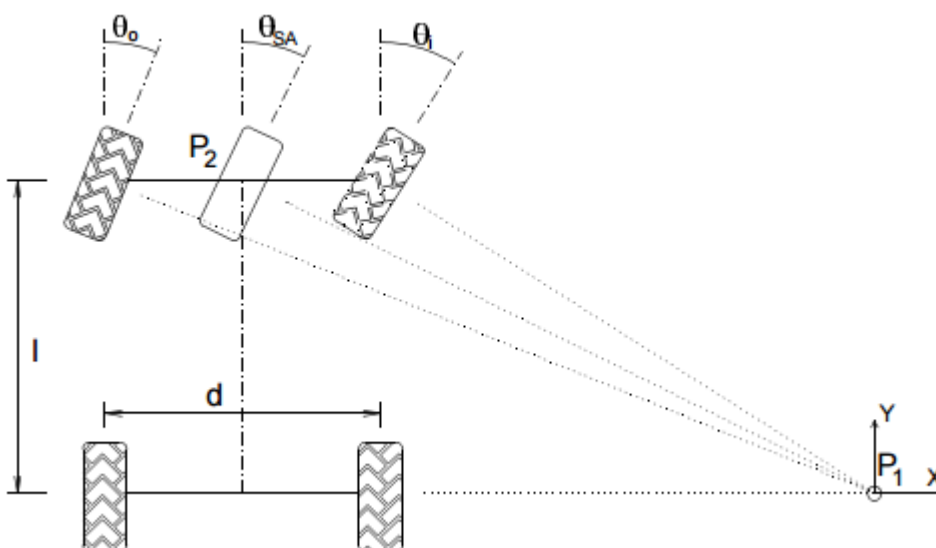
Skládá se z jednoho řízeného předního kola, které řídí směr pohybu a ze dvou pasivních zadních kol, která řídí rychlost pohybu nebo naopak, kdy je pasivní přední kolo, které ovlivňuje rychlost, a směr se řídí zadními.



Obr. 5 Schéma trojkolového podvozku. [3]

### 2.1.3 ACKERMANŮV PODVOZEK

S Ackermanovým podvozkem je seznámen každý, kdo se pohybuje v automobilovém průmyslu. Jedná se o jak konstrukčně, tak výpočtově, velmi složitý typ podvozku. Je vhodný pro vysoké rychlosti a má malou spotřebu energie. Jeho velkou nevýhodou je holonomní omezení, to se robotu nedovolí s podvozkem otočit na místě, jako v případě diferenciálně řízeného podvozku, proto není pro řešení úplně vhodné.



Obr. 6 Schéma Ackermanova podvozku. [3]

## 2.2 MOTORY

V oblasti mobilních robotů se nejčastěji používají rotační elektromotory. Z elektromotorů se z důvodu příznivého poměru výkon/hmotnost používají zejména stejnosměrné komutátorové motory. Jako alternativní pohon v oblasti školních robotů se používají krokové motory (nízká cena, snadné ovládání, není zapotřebí převodovky) a modelářské servomotory (snadné ovládání, zabudovaná převodovka, polohové řízení). [1]

V následujících kapitolách jsou probrány tyto tři druhy motorů a popsán základní princip jejich funkce.

### 2.2.1 DC MOTORY

Jedná se o často používaný typ motoru v oblasti pohonu mobilních robotů. Mezi jeho výhody patří velmi příznivý poměr výkon/hmotnost, relativně snadné řízení otáček, dostupnost a nízká cena a široká nabídka. Mezi nevýhody patří složitější polohové a rychlostní řízení v porovnání s krokovým motorem.

DC motor obvykle pracuje ve vysokých otáčkách a nízkém krouticím momentu, což není vhodné pro potřeby mobilních robotů. Řešením je použití převodovku, která může být již součástí motoru. Zároveň může být součástí motoru senzor polohy nebo rychlosti.



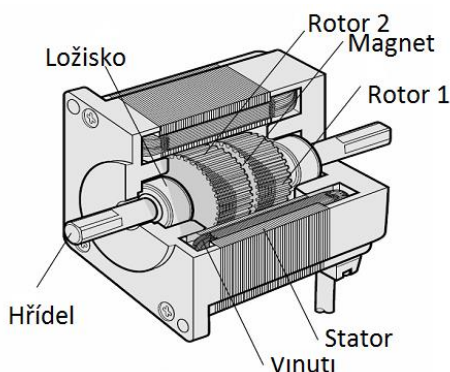
Obr. 7 Modelářský DC motorek. [7]

Tyto elektromotory mají dva elektrické kontakty. Připojením jmenovitého napětí na dané svorky se motor roztočí a ustálí se na jmenovitých otáčkách. Změnou polarity lze docílit změny směru otáčení a změnou napětí pak změny otáček. [1]



### 2.2.2 KROKOVÉ MOTORY

Krokové motory se využívají v případě potřeby pohonu, který umí přesně nastavit polohu a i přes působící síly ji udržet, pak se využívají krokové motory. V robotice se používají z důvodu snadné obsluhy. Pro precizní řízení rychlosti není nutné mít komplexní PID regulátor- Pokud se motory nepřetěžují, lze se obejít i bez zpětné vazby o natočení, počítají se pouze kroky.

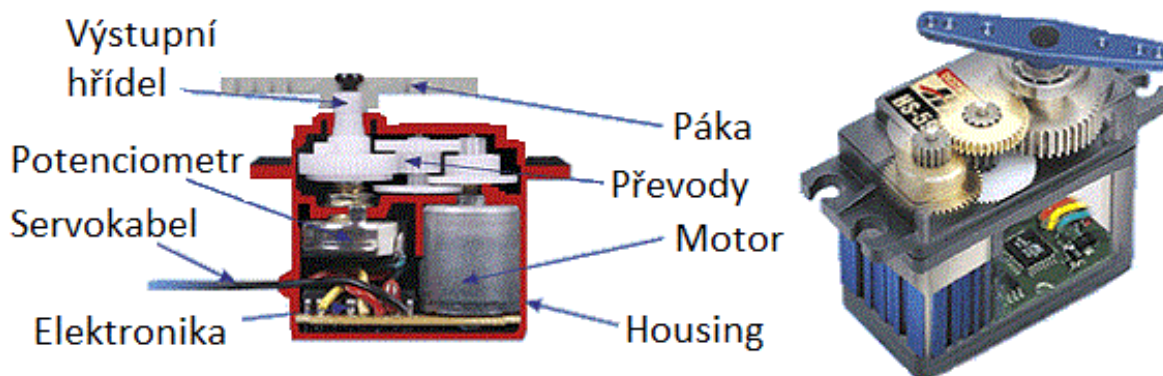


Obr. 8 Schéma krokového motoru. [7]

Princip krokového motoru je jednoduchý. Proud procházející cívkou statoru generuje magnetické pole, které přitáhne opačný pól magnetu rotoru. Vhodným zapojováním cívek dosáhneme vytvoření rotujícího magnetického pole, které otáčí rotorem. [7]

### 2.2.3 MODELÁŘSKÉ SERVOMOTORY

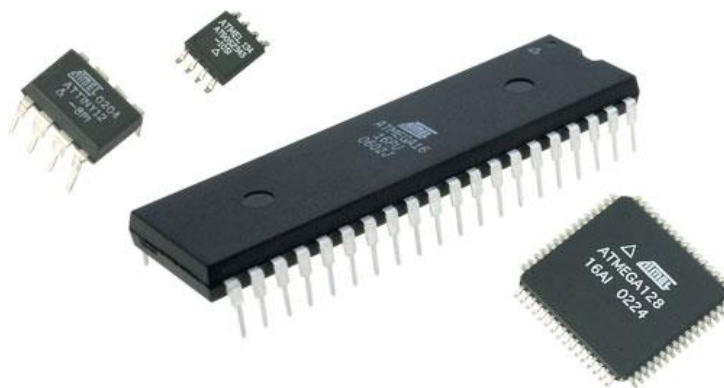
Modelářské servomotory dokáží přes své malé rozměry vyvinout velký kroutící moment. Jejich základem je několikastupňová převodovka, skládající se ze z několika (obvykle 4-5) ozubených kol. Ta je poháněna malým stejnosměrným motorkem. Napětí je uzpůsobeno používaným napětím v modelech, v rozsahu mezi 3,8-6 V. Nedílnou součástí je elektronika, která vyhodnocuje zpětnou vazbu z ramene serva a porovnává ji s řídicím signálem. Tato zpětná vazba je tvořena potenciometrem spojeným s posledním stupněm převodovky a je realizována analogovým obvodem.



Obr. 9 Schéma modelářského servomotoru. [7]

## 2.3 MIKROKONTROLER

Jedná se o monolitický integrovaný obvod obsahující kompletní mikropočítač. Tyto „jednočipy“ se vyznačují velkou spolehlivostí a kompaktností. Proto je jejich použití vhodné pro jednoúčelové aplikace, jako je řízení, regulace apod. Velmi často se s nimi dá setkat jako se součástmi embeded systémů. [11]



Obr. 10 Mikrokontroler. [7]

Jednočip je postaven tak, aby zahrnoval vše potřebné k tomu, aby mohl pracovat v požadované aplikaci bez dalších podporných obvodů. Tím je myšlena paměť pro uložení programu, operační paměť RAM a rozsáhlá sada podpurných bloků, A/D a D/A převodníky, bloky pro logické a analogové vstupy/výstupy, časovače a další aplikační logiku. [11]

## 2.4 SENZORY

Senzory jsou pro robot velmi důležité. Jedná se zdroj informací pro řídicí systém o okolí. V technickém slova smyslu je to mechanická nebo elektronická součástka, která měří určitou fyzikální veličinu a převádí ji na signál, který už řídicí systém dokáže zpracovat a následně vyhodnotit chování robotu. Senzory tak můžeme dělit na několik skupin. [10]

- Podle druhu měřené veličiny (analogová, digitální) – senzory teploty, tlaku, průtoku, mechanických veličin (posunutí, polohy, zrychlení, síly), senzory elektrických a magnetických veličin
- Podle média sloužícího pro přenos signálu (elektrické, hydraulické, pneumatické a jiné)
- Podle fyzikálního principu měření (způsobu převodu hodnoty měřené veličiny na hodnotu signálu)
- Podle styku s prostředím (kontaktní, bezkontaktní)

Senzor snímá sledovanou fyzikální, chemickou nebo biologickou veličinu a podle určitého principu ji transformuje na měřenou veličinu, většinou elektrickou. Vzniklý analogový signál je možné pomocí A/D převodníku převést na digitální a ten dále pomocí mikrokontroleru upravovat (například provést korekce naměřených hodnot).

Senzory můžeme dále dělit na dvě kategorie a to na externí – měřící parametry okolí robotu, a interní – měřící parametry robotu.



## 3 PLÁNOVÁNÍ TRASY, ALGORITMY

Řídicí systém charakterizujeme podle úkolu jaký provádí, vlastností robotu a vlastností algoritmu.

Navigace – problém hledání bezkolizní dráhy pro robot

Pokrytí – snaha o pokrytí prostoru v okolí robotu senzory

Lokalizace – využívání dat ze senzorů pro určení polohy robotu

Mapování – prozkoumávání a snímání neznámého prostoru

### 3.1 PLÁNOVÁNÍ TRASY

Plánováním cesty se rozumí schopnost robotu se dostat z bodu A do bodu B. Proto je důležité mít určité informace o prostředí, ve kterém se robot pohybuje. Ty jsou buď známé, částečně známe nebo neznáme. Dále se mohou v prostoru vyskytovat překážky, ať už pohyblivé nebo nepohyblivé. Plánování lze tedy podle těchto kritérií rozdělit do následujících variant

- Pohyb ve známém prostředí s nepohyblivými překážkami
- Pohyb ve známém prostředí s pohyblivými překážkami
- Pohyb v neznámém prostředí s nepohyblivými překážkami
- Pohyb v neznámém prostředí s pohyblivými překážkami

Prostředí můžeme reprezentovat modelem, který může být dvourozměrný a trojrozměrný. Z hlediska reprezentace prostředí ho lze dělit na dva druhy:

- spojité – prostor se považuje za spojité kontinuum.
- diskrétní – prostor je rozdělen na dílčí segmenty.

### 3.2 ALGORITMY PRO PLÁNOVÁNÍ CESTY

Pro plánování cesty robotu existuje množství metod. Lze je rozdělit podle možného způsobu použití. První možností jsou informované algoritmy. Ty využívají předem zadané kompletní data o prostředí, která jsou nahrána do paměti robotu předem. Ten si poté sám spočítá nejlepší trasu a pak se po ní vydá. Druhou možností je použití neinformovaných algoritmů, které požadují pouze data o počáteční a cílové pozici a cestu plánují za pohybu, v závislosti na datech získaných ze senzorů.

Pro tuto práci jsou nejdůležitější BUG algoritmy, proto se jimi bude práce podrobněji zabývat. U ostatních algoritmů bude zmíněn základní princip metody mapování.



### 3.2.1 INFORMOVANÉ ALGORITMY

- **Metoda potenciálových polí (metoda dekompoziční)**

Mapa (viz. Obr. 11) je rozdělena na buňky, každé buňce je přiřazena hodnota. Startu je přiřazena nejvyšší hodnota, cíli nejmenší. Robot pak postupuje mřížkou cestou od nejvyššího čísla k nejnižšímu. Okolí překážek má přiřazena čísla větší než volný prostor a tím se robot dokáže vyhnout překážce. [7]

9	8	7	6	5	4	3	2	1	0
9	8	7	6	5	4	3	2	1	1
9	8	7	6	5	4	3	2	2	2
9	8	7					3	3	3
9	8	8	8	9	10		4	4	4
9	9	9	9	9	9		5	5	5
10	10	10	10	9	8		6	6	6
11	11	11	10	9	8	7	7	7	7
	12	11	10	9	8	8	8	8	8
		11	10	9	9	9	9	9	9

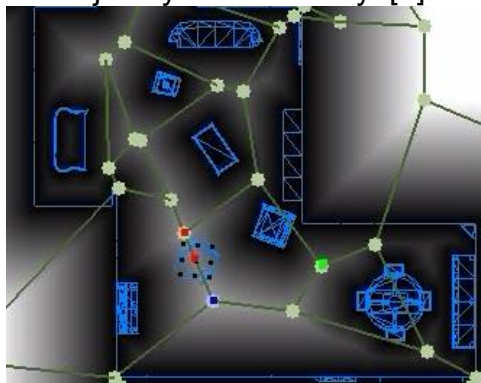
Obr. 11 Příklad potenciálového pole. [7]

- **Grafový přístup (metoda dekompoziční)**

Používá frontu pro uchování nezpracovaných vrcholů. Z fronty jsou vybrány vrcholy, do kterých vede hrana, a jsou nenavštívené. Předpokládá se, že virtuální hrany existují vždy mezi sousedními vrcholy. Algoritmus dělí buňky do tří kategorií, nenavštívené (nemají přiřazenou vzdálenost), mrtvé (již navštívené a všichni jejich sousedi byli také zpracováni) a živé (již navštívené, ale mající ještě nenavštívené sousedy) [7]

- **Pseudo-Voronoi diagramy (metoda dekompoziční)**

Nejkratší cesta není vždy tou nejideálnější volbou. Někdy je například potřeba držet se daleko od překážek. Voronoi hrana je definována tam, kde se setkají vlny ze dvou různých překážek. Takto předzpracovanou strukturu lze s výhodou využít i pro opakované hledání cest. Ze startu i z cíle je spuštěno prohledávání do šířky a čeká se, než najde cestu na některou Voronoi hranu. Poté se postupuje po hranách až do části obsahující cíl, kde druhé prohledávání do šířky definuje zbylou část cesty. [7]

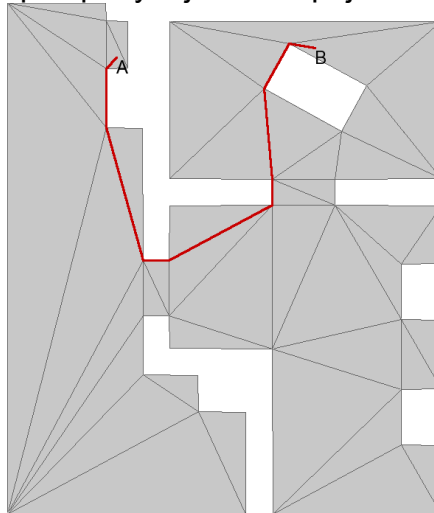


Obr. 12 Pseudo-Voronoi diagram. [7]



- **Graf viditelnosti (metoda RoadMaps)**

V této metodě, jsou za vrcholy grafů považovány rohy překážek. Pokud lze tyto vrcholy spojit čarou, pak jsou tyto vrcholy navzájem viditelné a není mezi nimi překážka. Robot se pak pohybuje mezi spojnicemi těchto vrcholů. [1]



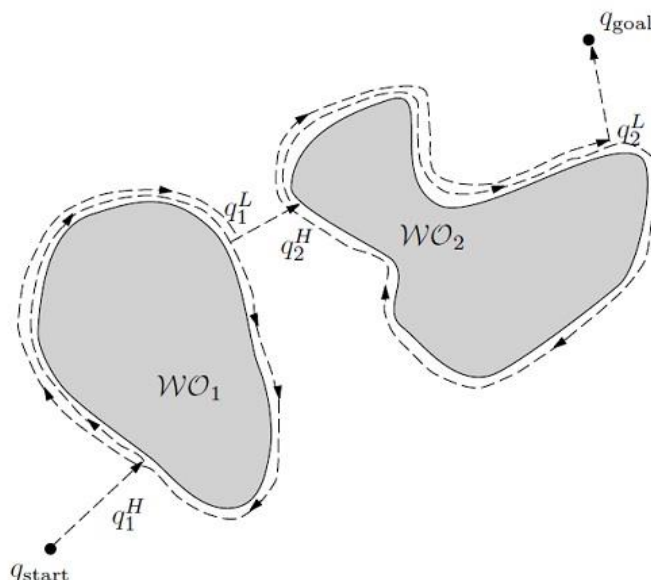
Obr. 13 Graf viditelnosti. [7]

### 3.2.2 NEINFORMOVANÉ ALGORITMY

Jako nejjednodušší možný přístup se jeví přímá cesta k cíli, dokud není detekována pomocí senzorů překážka. Té se pak robot vyhne a pokračuje dále v původním směru. Na začátku je nutno si definovat určité předpoklady o robotu a překážce. Robot je tedy pouze bod ve 2D rovině, je vybaven senzory a díky znalosti své pozice je schopen v každém okamžiku určit svoji vzdálenost k cíli. Překážka může být libovolná, ale musí jich být konečný počet s konečným obvodem, jinak by robot nebyl schopen najít cestu. [7]

#### BUG1

Robot si představíme jako bod s dotykovým senzorem, kterým je schopný detekovat překážky a zná svoji polohu a polohu cíle. V první fázi si robot propojí startovní  $q_{start}$  a cílovou pozici  $q_{goal}$  a vydá se po přímé trase. Pokud narazí na překážku, celou ji objede a najde místo, ze kterého je to nejbližší do cíle. Z něj si opět spočítá přímou trasu do cíle a po ní se vydá. To se opakuje pokaždé, když robot narazí na překážku. [2]



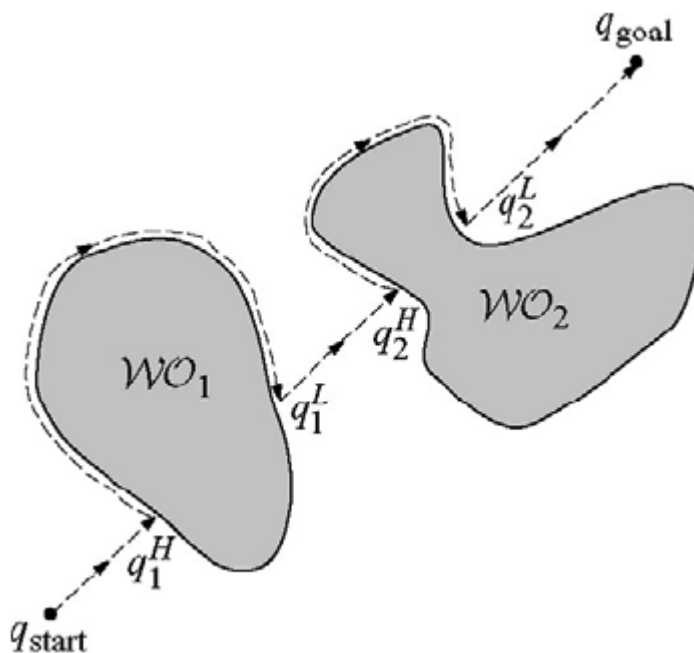
Obr. 14 Bug 1 Algoritmus s úspěšné dorazení do cíle. [2]

#### Algoritmus BUG1

1. Z  $q_{start}$  jed' do  $q_{goal}$
2. Pokud jsi v  $q_{goal}$  – KONEC
3. Pokud je v cestě překážka, označ bod  $q_i^H$ , na kterém jsi na ni narazil
4. Objed' celou překážku a najdi bod nejbližší cíli  $q_i^L$
5. Vydej se nejkratší cestou k bodu  $q_i^L$
6. Pokračuj z bodu  $q_i^L$  do  $q_{goal}$

## BUG2

Tento algoritmus má stejné základní předpoklady jako BUG1, ale jiný přístup. Zatímco BUG1 objížděl celou překážku, BUG2 je účinnější a tuto cestu si zkracuje. První část je stejná, robot si vytyčí pomyslnou přímkou  $H$  z  $q_{start}$  do  $q_{goal}$  a vyjede po ní směrem k cíli. Narazí-li na překážku, objede ji, a jakmile se dostane na průsečík s počáteční přímkou, vydá se po ní. Tento algoritmus je ale náchylnější na ztracení trasy, a v komplikovaném prostředí se může zaseknout v nekonečné smyčce. [2]



Obr. 15 BUG2 Algoritmus. [2]

### Algoritmus BUG1

1. Z  $q_{start}$  jed' do  $q_{goal}$
2. Pokud jsi v  $q_{goal}$  – KONEC
3. Pokud je v cestě překážka, označ bod  $q_i^H$ , na kterém jsi na ni narazil
4. Objed' celou překážku a najdeš-li bod na přímce  $H$  blíž k cíli než bod  $q_i^H$  jed do cíle
5. Pokud se vrátíš zpět do  $q_i^H$ , cesta neexistuje



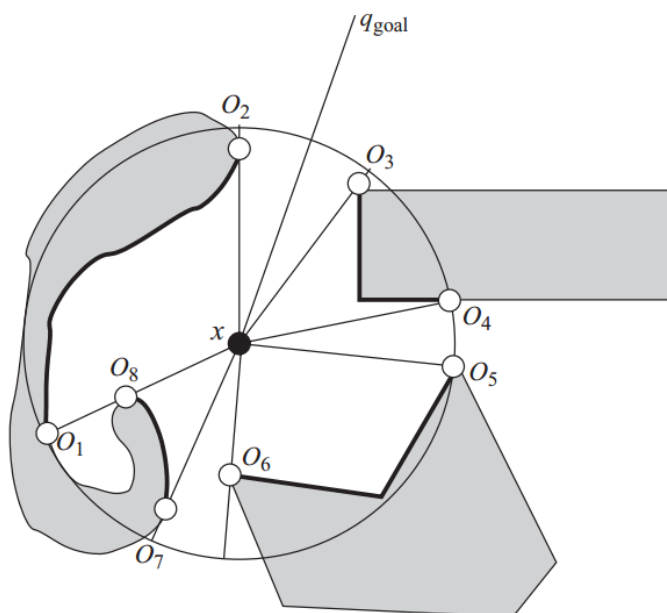
## BUG TANGENT

Vylepšením předchozích dvou variant je algoritmus BUG Tangent. V něm se uvažuje se senzorem s rozsahem  $360^\circ$  a nekonečným dosahem. Tyto senzory se modelují pomocí „raw distance function“  $\rho(x, \theta)$ , kde  $x$  je vzdálenost k překážce pod úhlem  $\theta$ .

Protože senzory mají omezený rozsah, nahrazuje se modelu pomocí „saturated raw distance function“, který má stejné parametry jako „raw distance function“ pokud je překážka v dosahu senzoru  $R$ , jinak je rovna nekonečnu. [2]

$$(1) \rho_R(x, \theta) = \begin{cases} \rho(x, \theta) \\ \infty \end{cases}$$

Tangent Bug předpokládá, že dokáže odhalit nespojitosti v  $\rho_R$  (viz. Obr.). Pro 2D prostor je interval spojitosti určen intervalem  $x + \rho(x, \theta)[\cos(\theta), \sin(\theta)]^T$ . Tam, kde tyto intervaly ztrácí spojitost, se pak vytvoří body nespojitosti  $O_i$ .

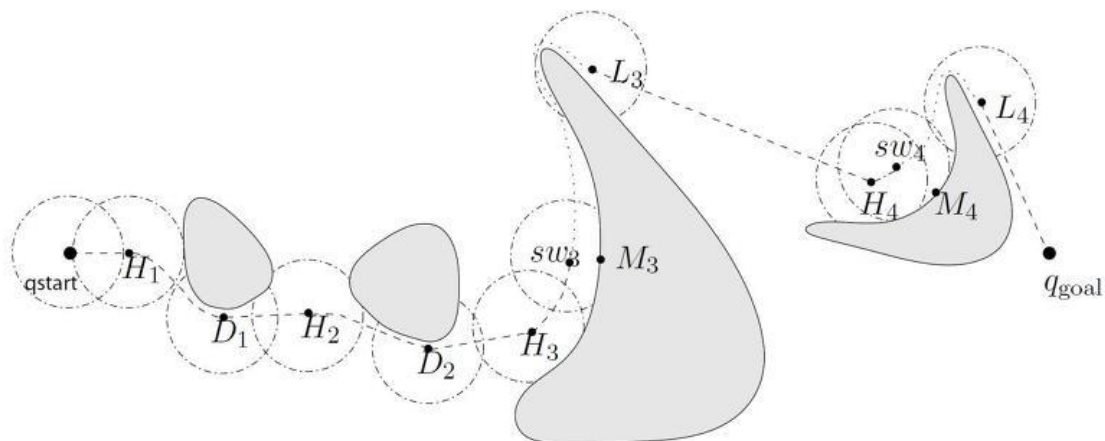


Obr. 16 Tangent BUG. [2]

### Algoritmus Tangent BUG

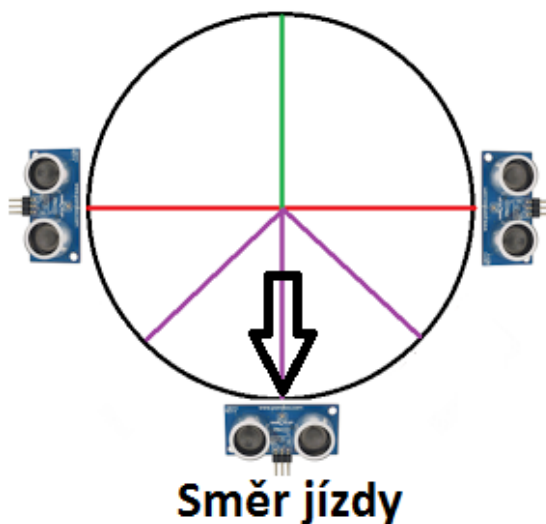
1. Z  $q_{start}$  jed' do  $q_{goal}$
2. Pokud jsi v  $q_{goal}$  – KONEC
3. Pokud robot detekuje překážku, detekuje bod nespojitosti  $O_i$
4. Jed' směrem ke zvolenému bodu nespojitosti
5. Pokud je překážka v cestě, spuštění podprogramu *objíždění překážky*
6. Pokud skončí podprogram *objíždění překážky*, jed' do  $q_{goal}$





Obr. 17 Ukázka trasy robotu pomocí BUG tangent algoritmu. [2]

Bug algoritmus předpokládá pokrytí senzory v rámci  $360^\circ$ . Toho nebude robot schopen dosáhnout. S třemi nainstalovanými ultrazvukovými senzory dosáhne uhlu pokrytí pouze  $180^\circ$ . Z toho budou plynout i jistá omezení při pohybu a mapování. Robot tak „uvidí“ pouze před sebe a do stran.



Obr. 18 Schéma umístění ultrazvukových senzorů.



## 4 MECHANICKÁ A ELEKTRONICKÁ KONSTRUKCE

### 4.1 VOLBA POUŽITÝCH SOUČÁSTEK

Nyní budou shrnuty všechny potřebné součástky, ale jejich funkci se práce bude věnovat až dále, v příslušných kapitolách.

Rozdělit je lze do pěti kategorií:

- Podvozek
- Řídící deska
- Senzory
- Motory
- Ovládání

Jako základní prvek, a zároveň základ pro všechny ostatní díly, považujeme podvozek. Ten jsme zvolili dvoukolový diferenciálně řízený. (Kap. 4.3)

Jako řídící deska bylo zvoleno Arduino, pro jeho jednoduchost, dostupnost a obrovskou uživatelskou základnu. V online podobě je dostupné velké množství OpenSource knihoven a dokumentací pro zjednodušení práce a popis jednotlivých částí a modulů. (Kap. 5)

Pro pohon robotu byly vybrány modelářské stejnosměrné elektromotory, jejich řízení budeme řešeno pomocí PWM a mikročipu L293D, což je stejnosměrný pulsní měnič pracující ve čtyřech kvadrantech. (Kap. 6)

Robot je nutné osadit senzory, aby z nich byl schopný získat data o svém okolí a aby na základě naměřených dat mohl na prostředí reagovat. Na základě rozboru byly zvoleny za vhodné následující senzory. (Kap. 7)

- Ultrazvukový senzor
- Enkodér

Pro komunikaci s ovladačem se použije Arduino „Shield“ USB Host Shield, který umožní ovládání pomocí ovladače pro PC, bezdrátové ovládání pomocí smartphonu se systémem Android nebo pomocí bluetooth ovladače. (Kap. 5.2)



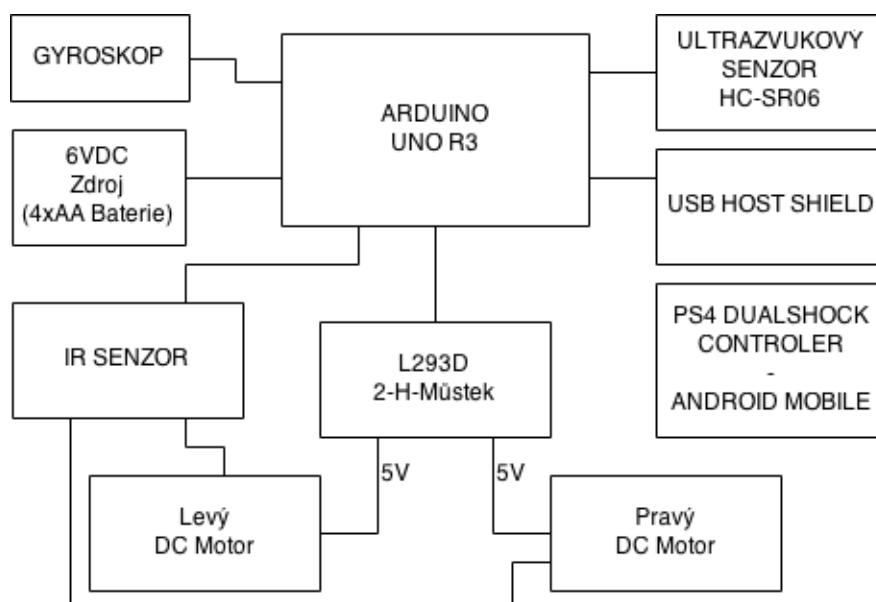
## 4.2 KONSTRUKCE ROBOTU

Na obrázku 12 je schéma robotu. Jako základní prvek slouží mikrokontroler Arduino, k němu jsou připojeny všechny potřebné periferie.

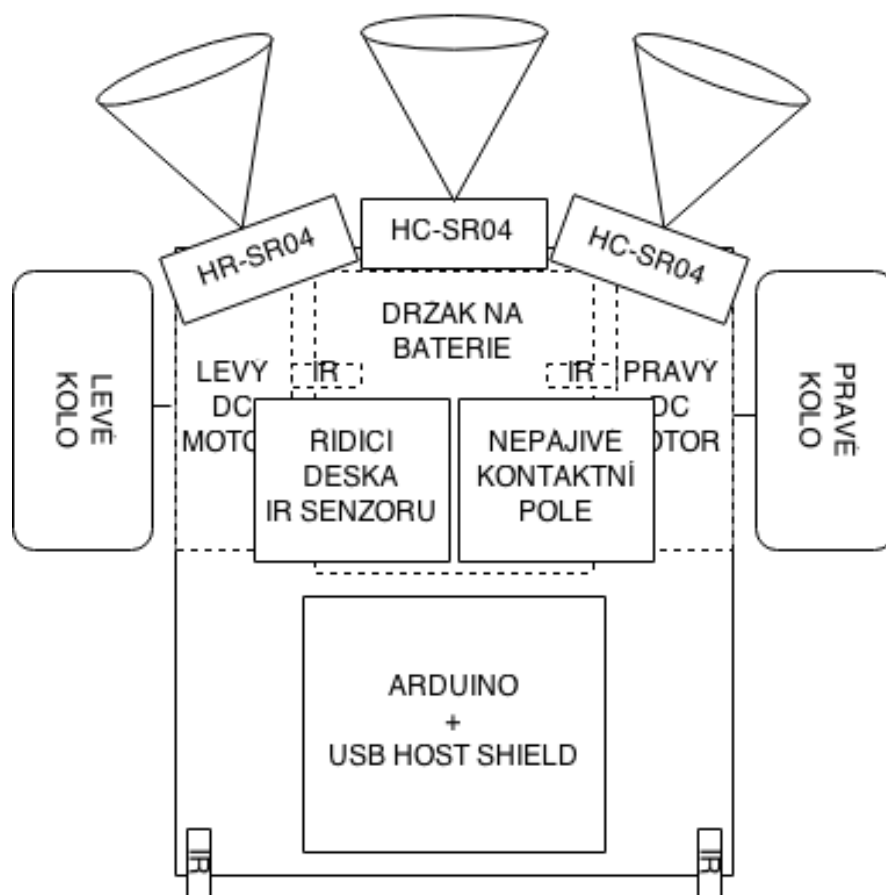
První jsou dva stejnosměrné motory, připevněné k podvozku. U motorů předpokládáme nutnost se točit oběma směry a možnost řízení otáček. To bude řešeno pomocí dvojitého H-můstku L293D. Pro jeho připojení k mikrokontroleru jsou třeba dva analogové a čtyři digitální vstupy.

Jako zdroj napětí slouží 4 AA baterie zapojené sériově, o celkovém napětí 6V a kapacitě 2700mAh, připojené na Arduino.

Na další analogové vstupy jsou připojeny ultrazvukové senzory HC-SR04, zjišťující vzdálenost robotu od překážek.



Obr. 19 Blokové schéma robotu.



Obr. 20 Layout robotu.



### 4.3 PODVOZEK

V teoretickém rozboru byly představeny tři typy podvozků, na kterých by robot mohl být postaven. Na základě rešerše byl zvolen diferenciálně řízený dvoukolový podvozek. To z důvodu jednoduchosti řízení a velmi dobré manévrovatelnosti. Robot se tak dokáže otočit na místě a snadno se vyhnout překážkám, na rozdíl od neholonomních podvozků typu Ackerman.

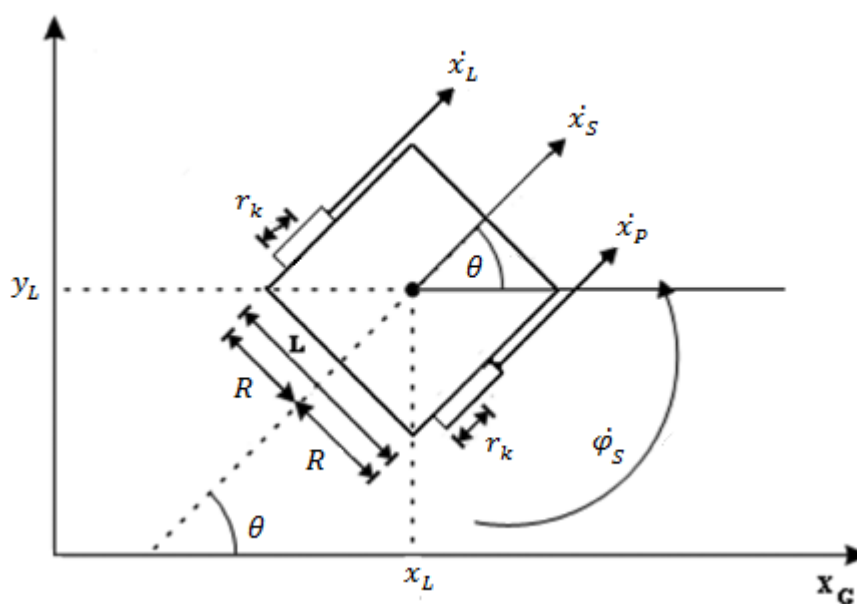


Obr. 21 Podvozek ve složeném stavu. [6]

### 4.4 MATEMATICKÝ MODEL ROBOTU

#### 4.4.1 KINEMATIKA

Při sestavení pohybových rovnic se vychází ze zjednodušeného kinematického modelu, ve kterém jsou ignorovány některé základní předpoklady, jako je např. tření a moment setrvačnosti.



Obr. 22 Kinematické schéma robotu.



Nejprve si definujeme si základní veličiny

$\dot{x}_L$  ... rychlost levého kola [m/s]

$\dot{x}_P$  ... rychlost pravého kola [m/s]

$\dot{\varphi}_L$  ... otáčky levého kola [rad/s]

$\dot{\varphi}_P$  ... otáčky pravého kola [rad/s]

$R$  ... vzdálenost kola od středu otáčení [mm]

$\dot{x}_S$  ... rychlost středu otáčení robotu [m/s]

$\dot{\varphi}_S$  ... rychlost otáčení kolem středu robotu [rad/s]

$r_k$  ... poloměr kola [m]

$x_L$  ... posunutí lokálního souřadného systému vůči globálnímu v ose  $x$  [m]

$y_L$  ... posunutí lokálního souřadného systému vůči globálnímu v ose  $y$  [m]

$\theta$  ... natočení lokálního souřadného systému vůči globálnímu [rad]

Rychlost pohybu jednotlivých kol se tedy spočítá z výrazu

$$(2) \dot{x}_i = R * \dot{\varphi}_L$$

Pomocí rychlostí obou kol jsme schopni spočítat rychlost středu robotu a zároveň úhlovou rychlost otáčení robotu kolem svého středu otáčení.

$$(3) \dot{x}_S = \frac{\dot{x}_L + \dot{x}_P}{2}$$

$$(4) \dot{\varphi}_S = \left( \frac{\frac{\dot{x}_L}{R} + \frac{\dot{x}_P}{R}}{2} \right)$$

Pohyb robotu je tak definován jako funkce závislá na následujících parametrech

$$(5) \dot{\delta}_L = f(l, r_k, \theta, \dot{\varphi}_L, \dot{\varphi}_P) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

Vztah mezi lokálním a globální souřadným systémem se provede pomocí ortogonální transformace

$$(6) R(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$(7) \dot{\delta}_G = R(\theta) \dot{\delta}_L$$

$$(8) \dot{\delta}_L = R(\theta)^{-1} \dot{\delta}_G$$

Sloučením rovnic (2),(3),(5) a (6) dostaneme výsledný vztah pro pohyb diferenciálně řízeného mobilního robotu

$$(9) \dot{\delta}_L = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \frac{r_k}{2} \begin{bmatrix} \dot{\varphi}_L + \dot{\varphi}_P \\ 0 \\ \frac{\dot{\varphi}_L - \dot{\varphi}_P}{2r_k} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

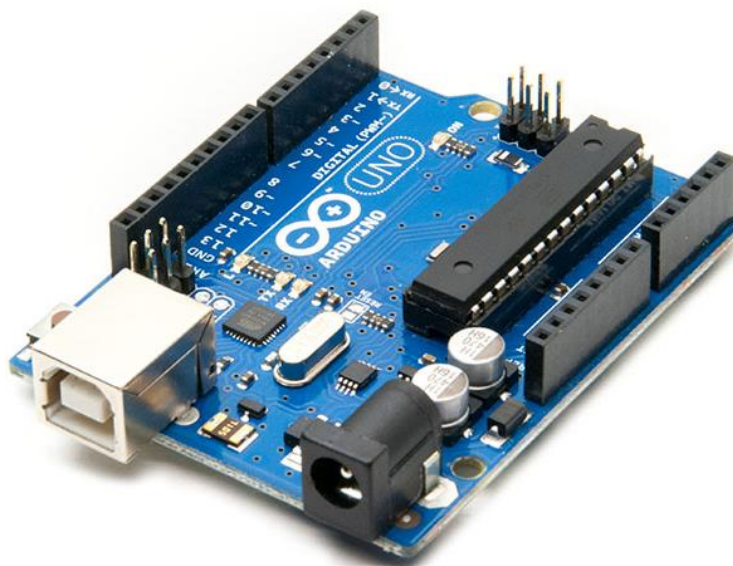


## 5 ŘÍDÍCÍ SYSTÉM

### 5.1 ŘÍDÍCÍ DESKA – ARDUINO

#### 5.1.1 OBECNĚ O ARDUINU

Arduino je open-source platforma založená na mikrokontroleru ATmega328. Má 14 digitálních I/O pinů (6 z nich lze použít jako PWM výstup), 6 analogových vstupů, 16 MHz keramický rezonátor, USB konektor, silový konektor, ICSP header a reset tlačítko [5]. Následující část se bude věnovat nejdůležitější části robota – Arduino.



Obr. 23 Arduino UNO R3. [4]

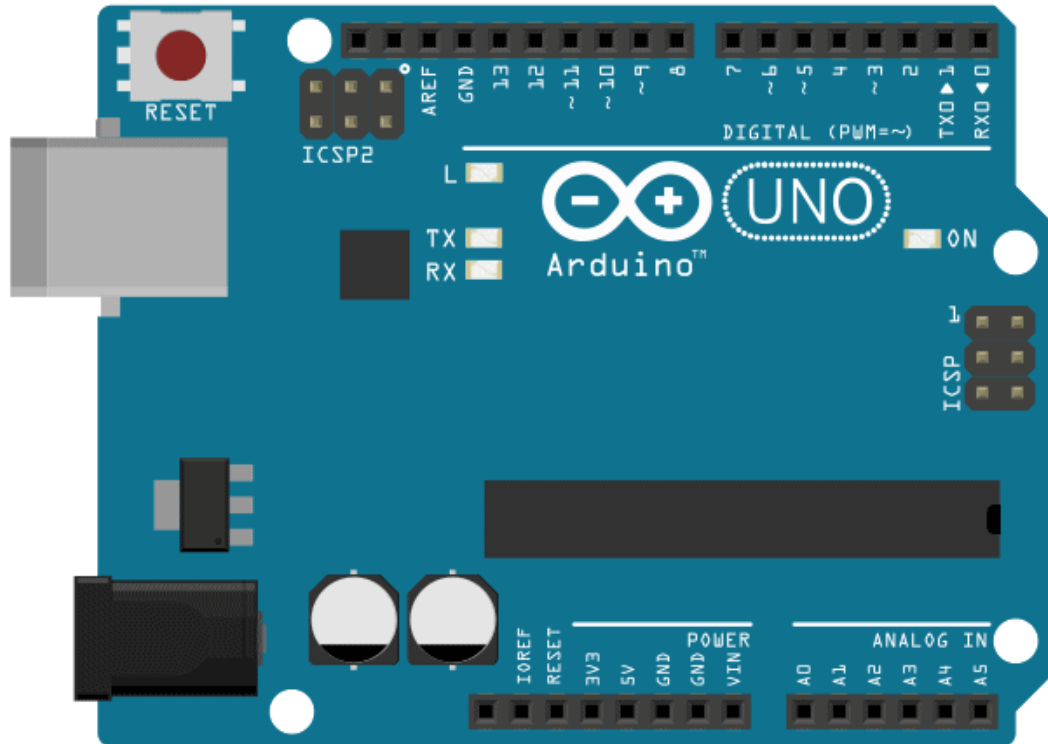
#### Shrnutí

Mikrokontroler	ATmega328
Pracovní Napětí	5V
Vstupní napětí ( )	7-12V
Vstupní Napětí (Max)	6-20V
Digitální I/O Piny	14 (6 s možností PWM výstupu)
Analogové I/O Piny	6
DC proud na I/O Pin	40 mA
DC Proud 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Taktovací frekvence	16 MHz
Délka	68.6 mm
Šířka	53.4 mm
Váha	25 g



### 5.1.2 ARCHITEKTURA ARDUINA

Arduino komunikuje s připojenými perifériemi skrz vstupní a výstupní piny. V následujícím textu to bude podrobněji vysvětleno a popsáno.



Obr. 24 Diagram desky Arduino. [4]

Na levé straně desky je USB port (šedá krabička) a napájecí „jack“ (černá krabička).

Na vrchní části, zprava doleva, je 14 pinů, číslované od 0 do 13. To jsou digitální piny. Ty mohou být individuálně nastaveny, aby složily jako vstup nebo výstup. To znamená, že digitální data mohou zapisovat nebo číst z připojených periférií. Protože se jedná o piny digitální, mají pouze dva možné stavy. HIGH (logická 1) a LOW (logická 0).

Některé digitální piny mají přednastavené funkce. Piny 0 a 1 jsou označeny jako RX a TX. Používají se k sériové komunikaci, k posílání a přijímání dat. Pin 13 má připojenou červenou LED diodu. Ta je na desce pod pinem 13 označena písmenem L. Piny 3, 5, 6, 9, 10 a 11 jsou označeny jako PWM. Tyto piny mohou simulovat analogový výstup přes digitální pin. Tento fakt bude využit pro řízení motorů.

Další pin je označen GND (země). Země se chápe část, co uzavírá obvod a umožní elektrickému proudu proudit obvodem. Do GND pinu se připojují GND piny z periferních zařízení. Pin AREF říká Arduino, jaké je nastavení napětí pro analogové piny. Na spodní části desky je šest pinů s označením 0 až 5. Jedná se o analogové výstupy. Na rozdíl od digitálních pinů, které mají jen dva stavy, analogové mají 1024, v závislosti na použitém napětí. Analogové piny lze použít pouze jako vstupy.





### 5.1.3 NAPÁJENÍ

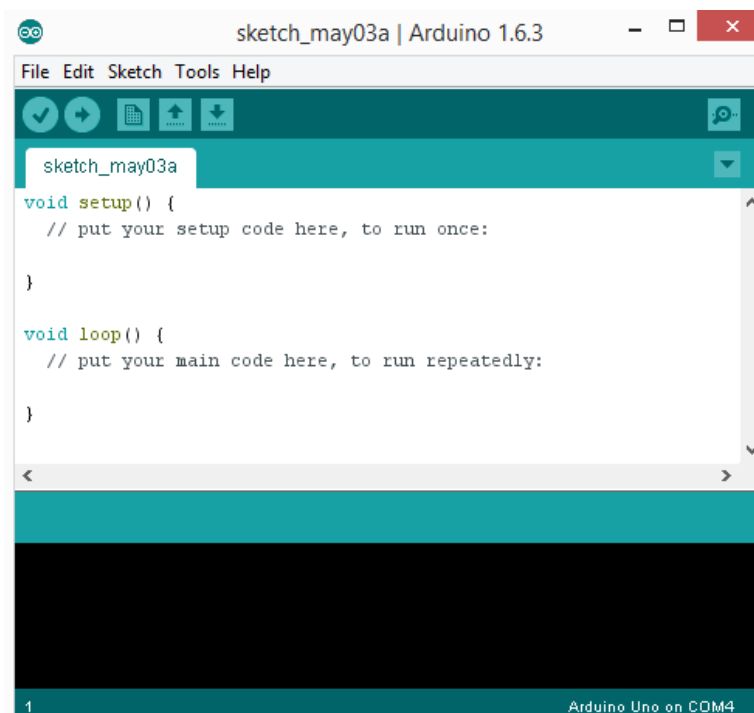
Arduino lze napájet přes USB port, nebo přes externí zdroj. Jako takový zdroj může sloužit externí baterie nebo solární články. Volba zdroje závisí na aplikaci, pro kterou bude deska použita. Bude-li se jednat o malý domácí projekt, lze využít napájení přes USB port. Při větších projektech, kde se počítá se snadnou dostupností k aplikaci, jsou vhodné externí baterie. Pro náročně dosažitelné aplikace je možno použít solárních článků pro napájení. V tom případě ale bude zapotřebí do obvodu zapojit regulátor, protože solární články neposkytují konstantní proud.

Deska dokáže pracovat s externím napětím 6-20 V. Doporučené rozpětí je 7-12V, s méně než 7V může být deska nestabilní naopak s více jak 12V se může deska přehřát a poškodit.

### 5.1.4 PROGRAMOVACÍ JAZYK

Arduino lze programovat v jazyce C nebo C++. Nejjednodušší je používat knihovnu Wiring. Ta je v současné době pro programování Arduina velmi rozšířená. Někdy se tato knihovna označuje jako samostatný programovací jazyk. [4]

Pro programování Arduina se využívá Arduino IDE (integrated development environment = integrované vývojové prostředí). Jedná se o open source knihovnu, která je napsaná v jazyce Java. Vývojové prostředí lze vidět na obrázku Obr. 25.



Obr. 25 Arduino IDE. [4]



Na obrázku [Obr. 25] je ukázkový kód, který obsahuje dva bloky programu.

- `void setup(){} -` Tělem programu je kód, který se provede jen jednou na začátku programu (při připojení napájení, při resetování).
- `void loop(){} -` Zde se píše kód, který se neustále opakuje dokola až do odpojení napájení.

Dané funkce jsou pro chod programu nezbytné a program bez nich skončí chybou.

### 5.1.5 ARDUINO PŘÍSLUŠENSTVÍ A KNIHOVNY

Mezi největší výhody práce s Arduinem jsou snadno připojitelné moduly, zvané „Shieldy“, senzory a velmi dobře zpracované knihovny. Ty pomáhají tím, že díky nim se nemusí konstruktéra soustředit na nadbytečné programování, ale přímo na problém a cíl práce.

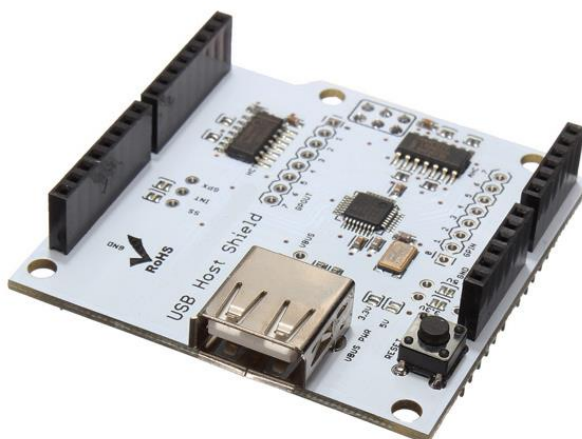
V této práci jsou použity následující knihovny:

- USB Host shield 2.0 library – pro komunikaci mezi Arduinem a dálkovým ovladačem přes USB host shield
- TLC5940 – 16 kanálový PWM kontrolér
- MatrixMath – knihovna pro práci s maticemi
- PID Library – knihovna pro PID regulátor, použitý při řízení
- PID Autotune Library – knihovna pro určení parametrů PID regulátoru



## 5.2 USB HOST SHIELD

USB Host Shield umožňuje k Arduinou připojit jiná zařízení pomocí USB portu. Shield je postaven na MAX3421E, což je periferní USB kontrolér obsahující digitální logický a analogový obvod nezbytný pro umožnění přenosu signálu z periférií v reálném čase. [4]



Obr. 26 USB Host Shield. [4]

Pomocí USB Host Shieldu lze připojit následující zařízení

- Klávesnice, myš, joystick
- Herní ovladače pro PlayStation, Xbox, Nintendo
- Mobilní telefon
- Paměťové karty / HDD
- Bluetooth adaptéry

V případě manuálního ovládání bude připojen k Shieldu bluetooth adaptér a ovládání bude řešeno pomocí ovladače pro PS4, Dualshock 4 [Obr. 27]. Vše bude fungovat bezdrátově pomocí přenosu dat přes Bluetooth.



Obr. 27 Dualshock 4.

V případě autonomního režimu je nutné se připojit přes bluetooth pomocí telefonu se systémem Android, aby bylo možné robotu zadat souřadnice cíle. Připojení přes mobil bude zároveň podporovat i manuální režim.



## 6 POHONNÝ SYSTÉM

### 6.1 MOTORY

Pro použití v mobilním robotu byl zvolen stejnosměrný motor s převodovkou 1:48 a následujícími parametry:

Parametry:

DC motor s převodovkou

Jmenovité otáčky (6V): 200RPM  $\pm 10\%$

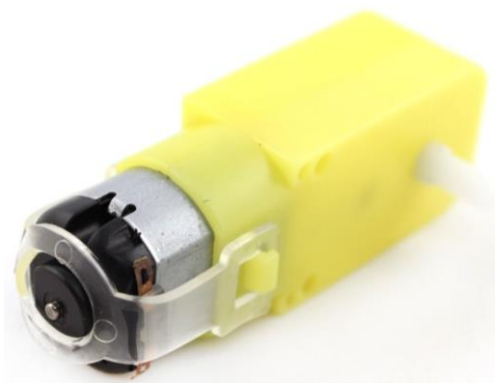
Jmenovitý proud (6V): 200mA

Jmenovité napětí (3V): 90RPM  $\pm 10\%$

Jmenovitý proud (3V): 150mA

Napětí: 3-6 V

Převodový poměr 1:48



Obr. 28 DC motor, 1:48.

### 6.2 ŘÍZENÍ MOTORŮ

Před zapojením motorů je třeba zvážit, co je po motorech požadováno pro řízení pohybu robotu.

- Regulování rychlosti otáčení
- Řízení směru otáčení

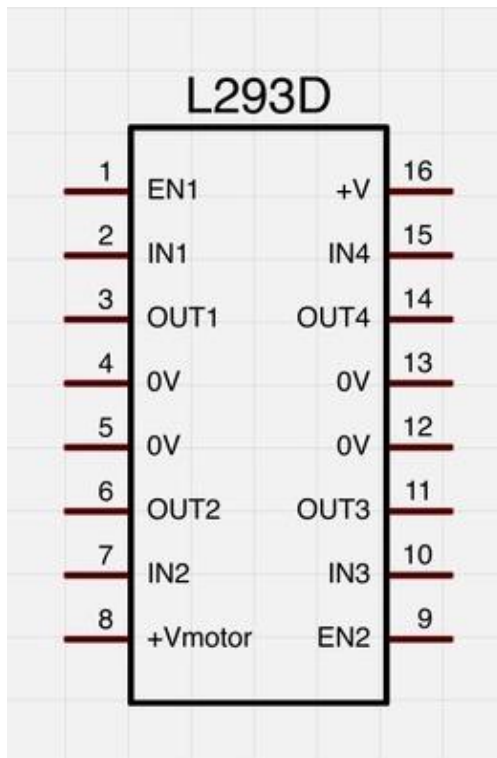
Po připojení zvoleného motoru přímo k Arduino, a puštění proudu se motor začne točit. Točí se pouze jedním směrem a konstantní rychlostí, to ale není pro robot vyhovující. V tomto případě lze řídit směr přehozením kontaktů, ale rychlost nelze řídit vůbec.



### 6.2.1 L293D

Do obvodu mezi Arduino a DC motorek bude připojen integrovaný obvod L239D (stejnsměrný pulsní měnič pracující ve čtyřech kvadrantech), který pomůže požadované vlastnosti zrealizovat. Tento čip dokáže jak řídit směr, tak otáčky dvou DC motorů pomocí PWM (pulse with modulation).

Jedná se o velmi jednoduchý H-můstek. Má dva můstky, jeden na levé, druhý na pravé straně a proto dokáže ovládat dva motory současně. Dokáže pracovat s proudem do 1A a napětím mezi 4.5V až 36V.



Obr. 29 L293D Layout.



Obr. 30 L293D Čip.

Pro začátek jsou použity tři piny, Pin1(En1), Pin2(In1) a Pin3(In2). Připojíme-li k Arduino na 5V výstup a motor je připojen na piny OUT1 a OUT2, začne se točit v jednom směru. Tento směr je směr A. Pokud je Pin1(En1) připojen k zemi (GND), motor se zastaví. Pin1(En1) všechno spouští a vypíná. A to je klíč při využití PWM a řízení rychlosti otáčení. Po zapojení Pin1(En1) zpět na 5V a Pin3(In2) na GND, pak se motor začne otáčet v opačném směru, směru B.

L293D má dva piny pro napájení, Pin8(+Vmotor) slouží pro napájení motorů, Pin16(+V) pro napájení mikročipu. Oboje lze připojit k Arduino, ale také je možno připojit externí zdroj na Pin8(+Vmotor) pokud používáme motory s vyšším jmenovitým napětím.

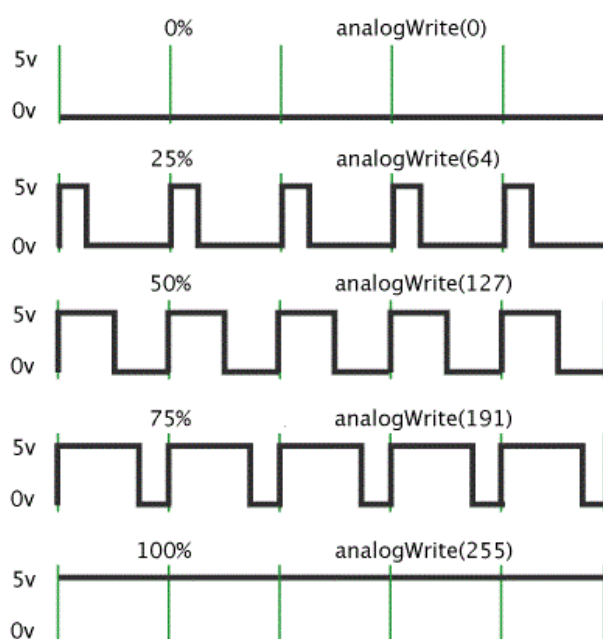
Řízení pomocí Arduina probíhá následovně. Je-li na obou vstupních pinech L293D napětí, motor se točí jedním směrem. Pokud na jednom pinu napětí není a na druhém je, motor se točí opačným směrem.



## 6.2.2 PWM

PWM, neboli pulzně šířková modulace je modulace pro přenos analogového signálu pomocí dvouhodnotového digitálního signálu. Jedná se o regulaci využívající změny šířky proudového impulsu. Napětí a proud zůstává stejný, ale mění se aktivní doba, kdy prochází proud motorem. Je to v podstatě bezztrátová regulace, protože tranzistor přechází do plně otevřeného stavu velmi rychle.

Je-li nastavený digitální výstup z Arduina, tak má stavy pouze 0 (LOW) a 1 (HIGH). Ale to k regulaci nestačí. Je potřeba nastavovat i jiné hodnoty než 0 a 1. K tomu slouží na Arduinu piny s označením PWM, které mohou posílat analogový signál v rozmezí 0-255 (0-100%). Připojení Pin1 (En1) na analogový výstup z Arduina je způsob řízení otáček motoru.



Obr. 31 Příklad PWM. [7]



## 7 SENZORICKÝ SYSTÉM

### 7.1 EXTERNÍ SENZORY

#### 7.1.1 ULTRAZVUKOVÝ SENZOR HC-SR04

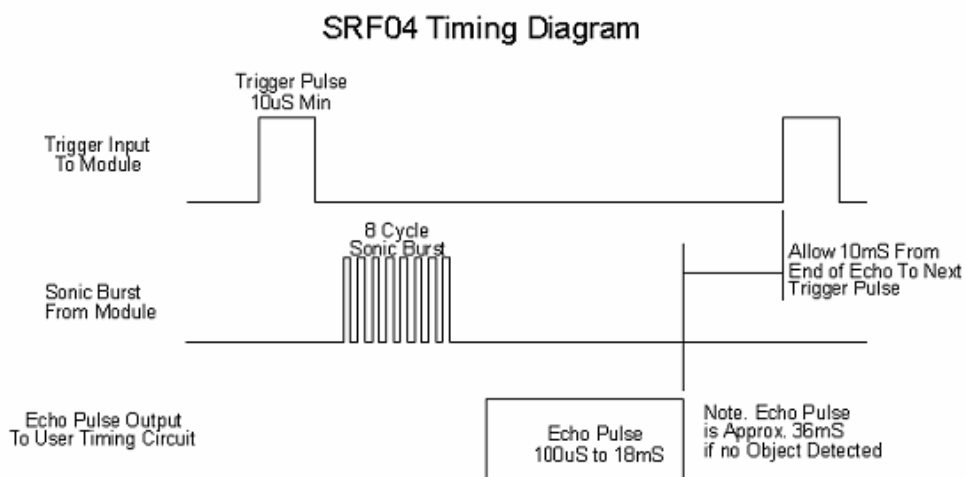
Princip měření vzdálenosti k překážce je založen na principu měření doby mezi vysláním akustického signálu a přijetím odraženého signálu – echa. [1]

Použitý senzor je schopný měřit vzdálenost v rozmezí od 3 do 300cm s přesností 3mm. Napájen je přímo z Arduina 5V/50mA. Kmitočet použitého akustického signálu je 40kHz.



Obr. 32 Senzor HC-SR04.

Princip měření vzdálenosti je následující. Na vstup Trig je vyslán puls o délce  $10\mu\text{s}$  a senzor poté automaticky vyšle osm 40kHz pulzů. Výstup Echo je nastaven na HIGH, dokud se navrátí signál odražený od překážky. Pro změření délky impulsu je využita funkci `pulseIn()`, která měří délku pulsu podle předem nadefinovaných parametrů. V konkrétním případě tedy bude sledována doba, kdy je Echo nastaveno na HIGH. Pokud je Echo puls ve stavu HIGH déle než 36ms, přepne se zpět na LOW, což vyjadřuje, že není detekována žádná překážka. Po přepnutí Echo pulsu na LOW následuje asi 10ms pauza a po ní se celý proces opakuje.



Obr. 33 HC-SRF04 Časový diagram. [1]



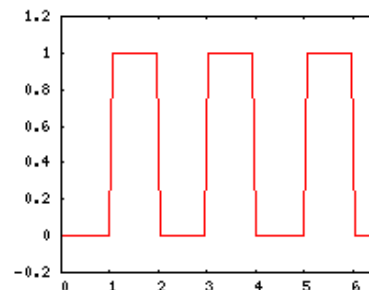
Vzdálenost k překážce je pak rovna polovině doby délky echa vynásobené rychlostí zvuku. Polovině délky proto, že je to délka, kterou signál urazí cestou k překážce a zpět. Rychlost zvuku je při teplotě vzduchu 25°C rovna 346m/s.[1]

$$(10) l_p = \frac{t_p}{2} v_{zvuku}$$

## 7.2 INTERNÍ SENZORY

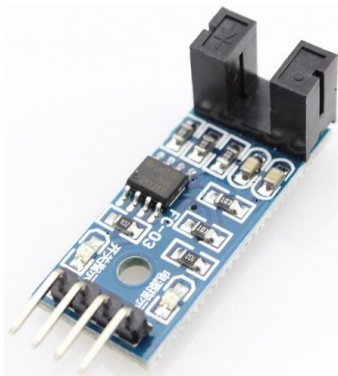
### 7.2.1 ENKODÉR

Základním principem enkodéru je měření přítomnosti nebo nepřítomnosti nějakého signálu. V robotické praxi se nejčastěji setkávám s enkodéry optickými. Ty lze ještě rozdělit na inkrementální a absolutní. Inkrementální dávají údaje o natočení, absolutní o poloze v rámci 360°. Druhá možnost je použít potenciometr a polohu určit pomocí A/D převodníku.



Obr. 34 Signál z enkodéru. [7]

Pro robot je použit foto elektrický senzor (Obr. 35) a děrovaný enkodérový disk (Obr. 36). Senzor se skládá z IR přijímače a vysílače, pokud není mezi vysílačem a přijímačem překážka, prochází signál v pořádku. Pokud je mezi nimi překážka, dojde ke snížení napětí na výstupu. Změny lze sledovat a je-li znám počet drážek v disku, pak lze stanovit i rychlost otáčení kola a ujetou vzdálenost. Součástí senzoru je i LED dioda zobrazující stav. Pokud senzor nedetekuje překážku, je výstup nastaven na LOW a informační dioda svítí. Je-li mezi vysílačem a přijímačem překážka, výstup je nastaven na HIGH a dioda nesvítí. Pracovní napětí je 3V - 5V



Obr. 35 Foto-elektrický senzor. [6]



Obr. 36 Disk Enkodéru.



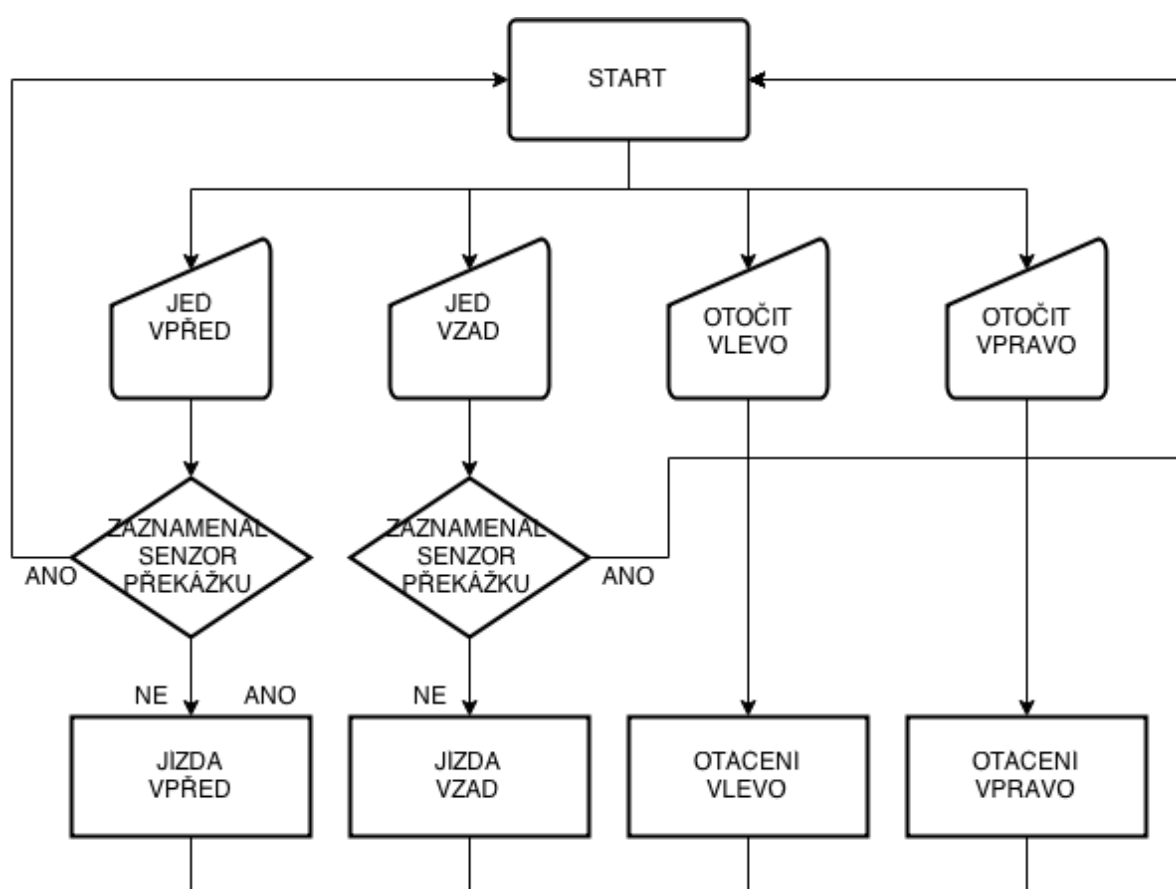


## 8 MANUÁLNÍ REŽIM

V manuálním režimu předpokládáme ovládání pomocí bluetooth ovladače, nebo aplikace na Android.

Na Obr. 37 je vidět algoritmus pro manuální režim. Po zadání příkazu JEĎ VPŘED/JEĎ VZAD robot zpracuje data ze senzorů. Pokud vyhodnotí, že je před ním nebo za ním překážka, nedovolí žádný pohyb. Pokud nebude detekována překážka, bude se moci pohybovat ve zvoleném směru.

Pro otáčení vlevo a vpravo nejsou požadovány data ze senzorů, protože se robot v podstatě otáčí na místě, a pokud by byl ve špatné pozici, ze které by se nemohl dostat, toto by mohlo pomoci.



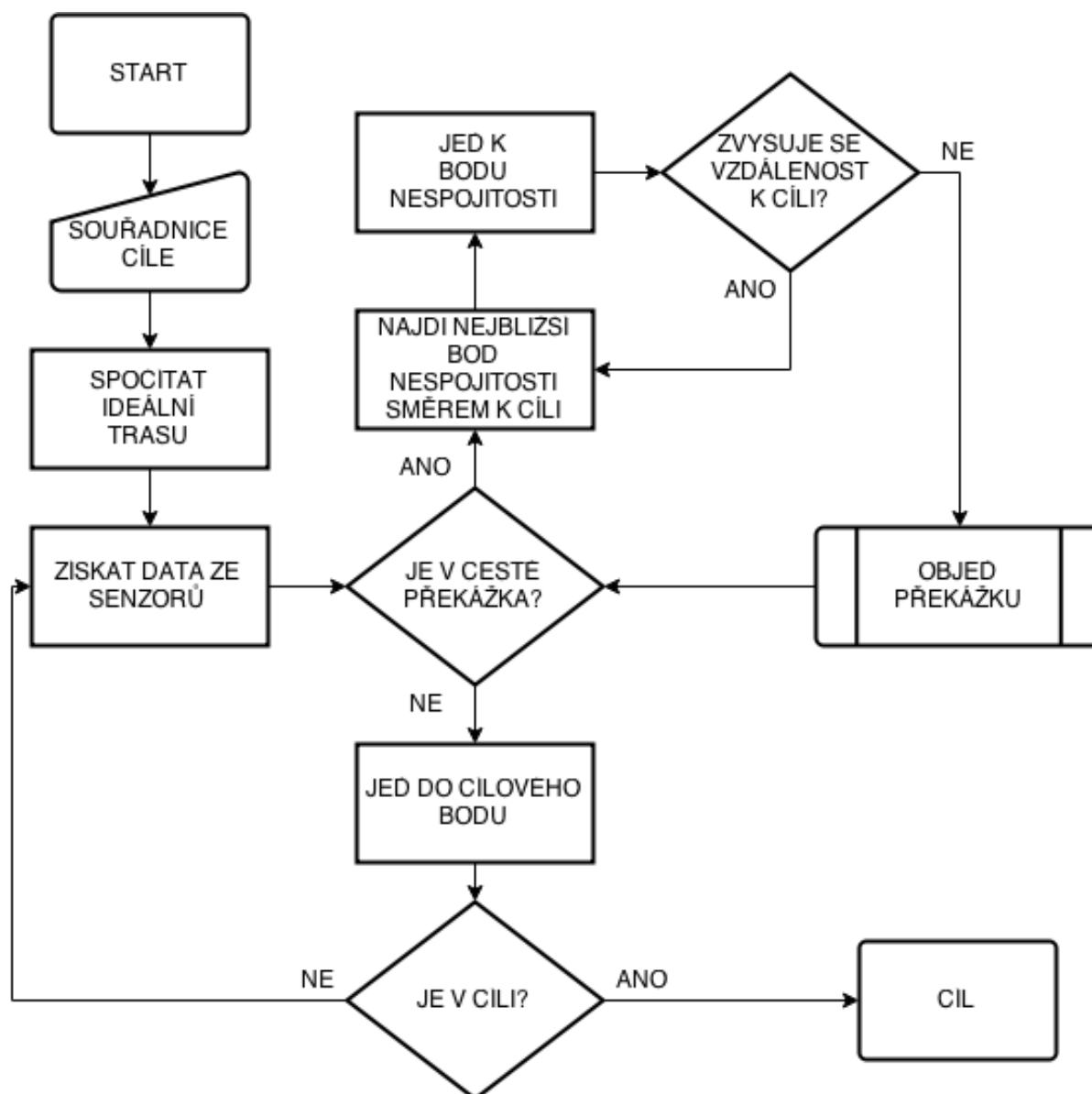
Obr. 37 Algoritmus manuálního ovládání.



## 9 AUTONOMNÍ REŽIM

V autonomním režimu se aplikuje algoritmus BUG Tangent. Robot by tak měl objíždět překážky s dostatečnou vzdáleností a nemělo by dojít ke kolizi.

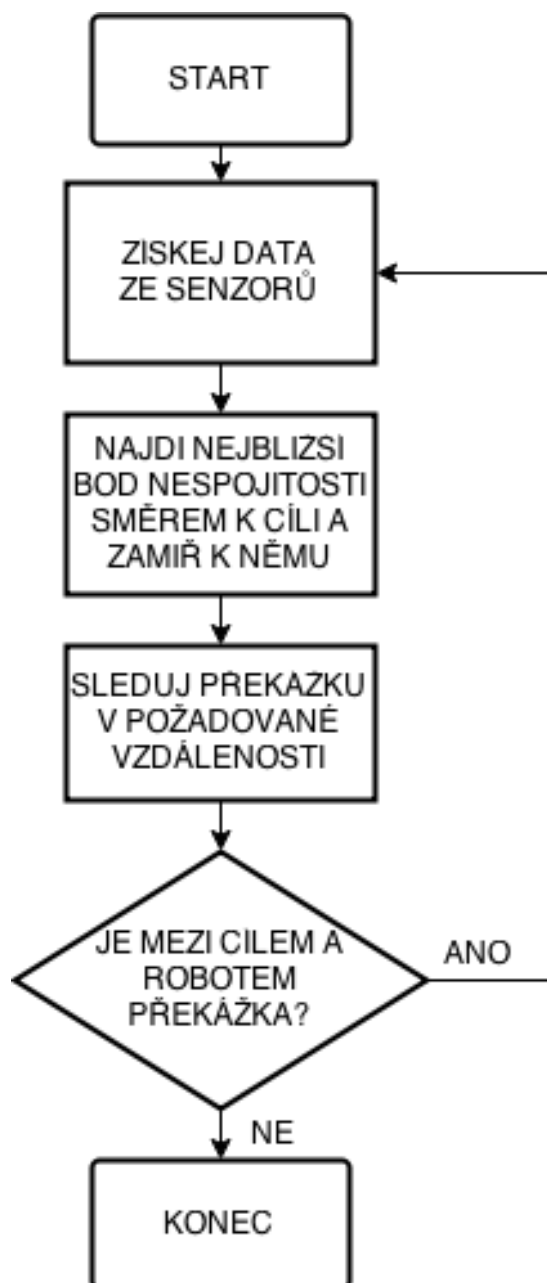
Robot si na začátku vyžádá souřadnice cílového bodu, bez nich nebude program pokračovat. Po získání těchto souřadnic provede výpočet požadovaného úhlu natočení a vyrazí směrem k cíli. Pokud senzory detekují překážku, přepne se robot z „move to goal“ algoritmu na „boundary following“, tedy na objíždění překážky. Pokud tento podprogram skončí, nevyhodnotí, že robotu brání v cestě překážka, spustí se zpět hlavní program a robot bude pokračovat k cíli.



Obr. 38 Algoritmus autonomního pohybu.



Algoritmu objíždění překážky je na Obr. 40. Robot si najde body nespojitosti, vzniklé na kružnici dané rozsahem senzorů. Poté vyhodnotí nejbližší bod nespojitosti směrem k cíli a k němu se vydá, dále sleduje stěnu v určité vzdálenosti, dokud nedojede do místa, kde už nemá žádnou překážku před sebou. Při sledování překážky robot uvažuje za primární data. Těmi jsou data ze senzoru blíže k cíli. Pokud roste vzdálenost od překážky, robot se natáčí směrem k překážce. Pokud vzdálenost klesá, robot se stáčí směrem od překážky. Tak dochází ke kopírování profilu překážky. Pokud se robot dostane do stavu, kdy mezi ním a cílem nevyskytuje překážka, ukončí podprogram a spustí opět hlavní algoritmus.



Obr. 39 Algoritmus objíždění překážky.



## 10 ZÁVĚR

Cílem práce bylo postavit mobilní robot schopný jak pohybu na základě manuálního ovládání, tak autonomního pohybu. Robot měl být schopný dostat se z bodu A do bodu B a cestou se vyhnout všem překážkám.

Robot byl postaven z volně dostupných součástek v elektrotechnických a modelářských obchodech. Byl splněn cíl jednoduché konstrukce a snadné možnosti oprav a úpravy konstrukce.

V průběhu konstrukce robotu a jeho testování největším problémem bylo napájení. Při připojení Arduina k počítači se robot choval podle požadavků, ale při připojení na externí zdroj (baterii) nedocházelo k dostatečnému napájení motorů a tak se robot nebyl schopný pohybovat. Byla vyzkoušena změna baterií, ze 4AA baterií na jednu 9V, ale to nemělo na tento problém vliv. Řešením bylo použití kvalitnějších dobíjecích baterií s kapacitou 2700mAh. Levnější baterie pravděpodobně nebyly schopny dodat požadovaný špičkový proud při rozběhu motorů.

Druhým problémem byl nedostatek prostoru na podvozku robotu. V prvních návrzích se předpokládalo, že vše půjde na podvozek osadit bez sebemenšího problému. Tento předpoklad se ale nakonec ukázal jako mylný. Při libovolném uspořádání prvků docházelo ke kolizi s montážními šrouby, jimiž byly připevněny motory a zadní kolečko. Nakonec bylo nutné vyvrtat několik nových otvorů, a přidat polystyrenovou desku na základní desku podvozku. Tak došlo k nadzvednutí prostoru pro montáž dílů a nedocházelo ke kolizi mezi spojovacím materiálem a řídicími komponentami.

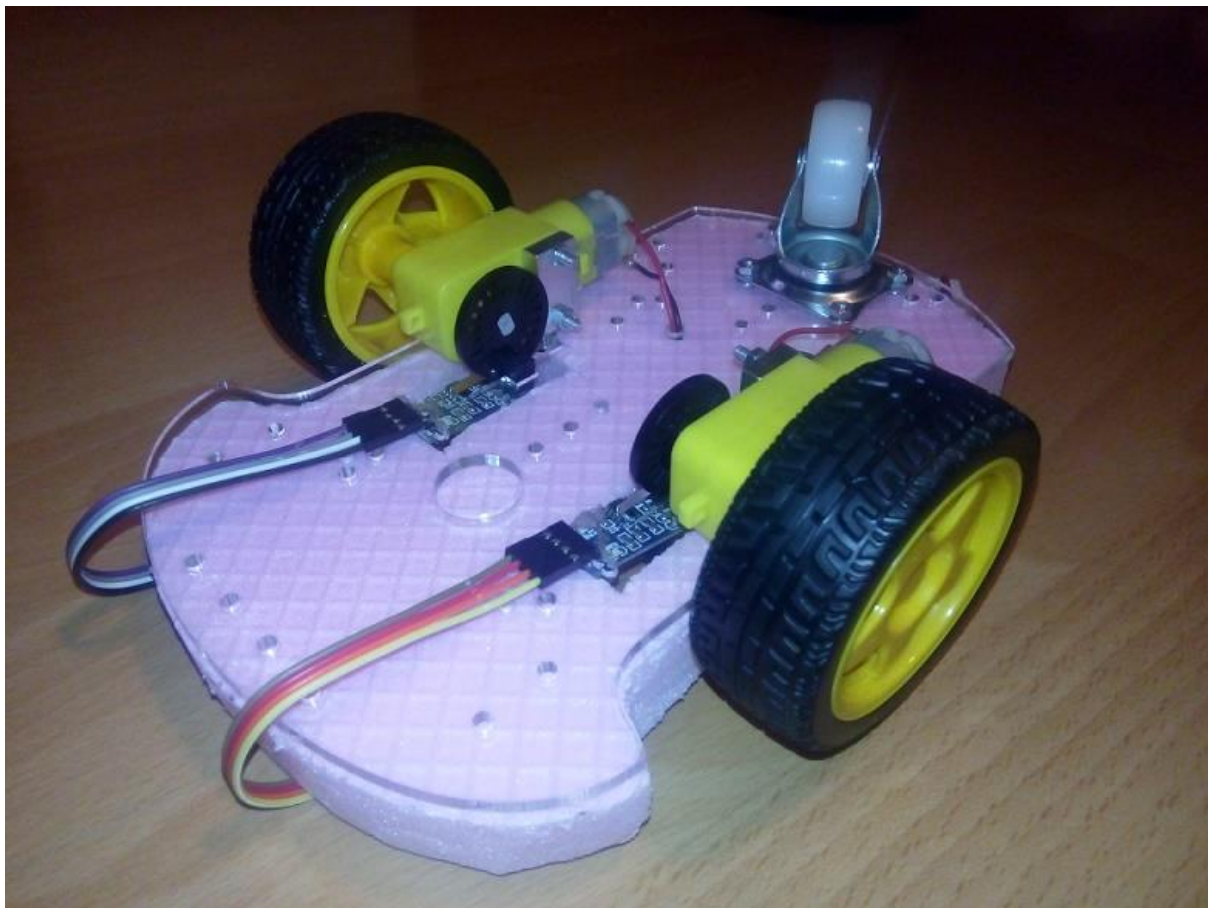
Režim manuálního ovládání byl kompletně naprogramován a otestován. Robot jezdí vpřed, vzad. Otáčí se vlevo a vpravo. Pokud má v cestě překážku, program mu nedovolí se pohybovat v jejím směru. Manuální režim pracuje pouze se čtyřmi režimy jízdy. Těmi jsou VPŘED/VZAD/OTOČENÍ VLEVO/OTOČENÍ VPRAVO.

Autonomní režim není zcela kompletní. Došlo k zpožděnému dodání jednoho senzoru, foto-elektrického. Ten je pro autonomní pohyb robota nezbytný, bez něho se robot nedokáže lokalizovat, protože neví, jakou vzdálenost ujel. Robot tedy dokáže spočítat nejkratší trasu. Dokáže detekovat překážky, ale není schopný se dostat na konkrétní místo. Tato část by po konzultaci s vedoucím bakalářské práce měla být dodělána dodatečně po odevzdání.

Po dokončení chybějícího algoritmu by měl robot být schopný po zadání cílových parametrů dojet z bodu A do bodu B. Cestou detekovat překážky a pomocí některého z implementovaných algoritmů tyto překážky objet a pokračovat k cíli. Pro autonomní režim musel být ještě doprogramován další druh pohybu, oproti manuálnímu. A tím bylo zatáčení, tedy to, že jedno kolo se točí s menšími otáčkami než druhé a v závislosti na rozdílu otáček tak dochází ke křivočarému pohybu, jehož úhel zakřivení je dán rozdílem rychlostí kol. Implementace tohoto pohybu byla nutná pro objíždění překážek. V první verzi robot vždy ujel kus trasy, natočil se, opět popojel a to se opakovalo pořád dokola. Zapojením křivočarého pohybu se stal pohyb okolo překážek plynulejší.

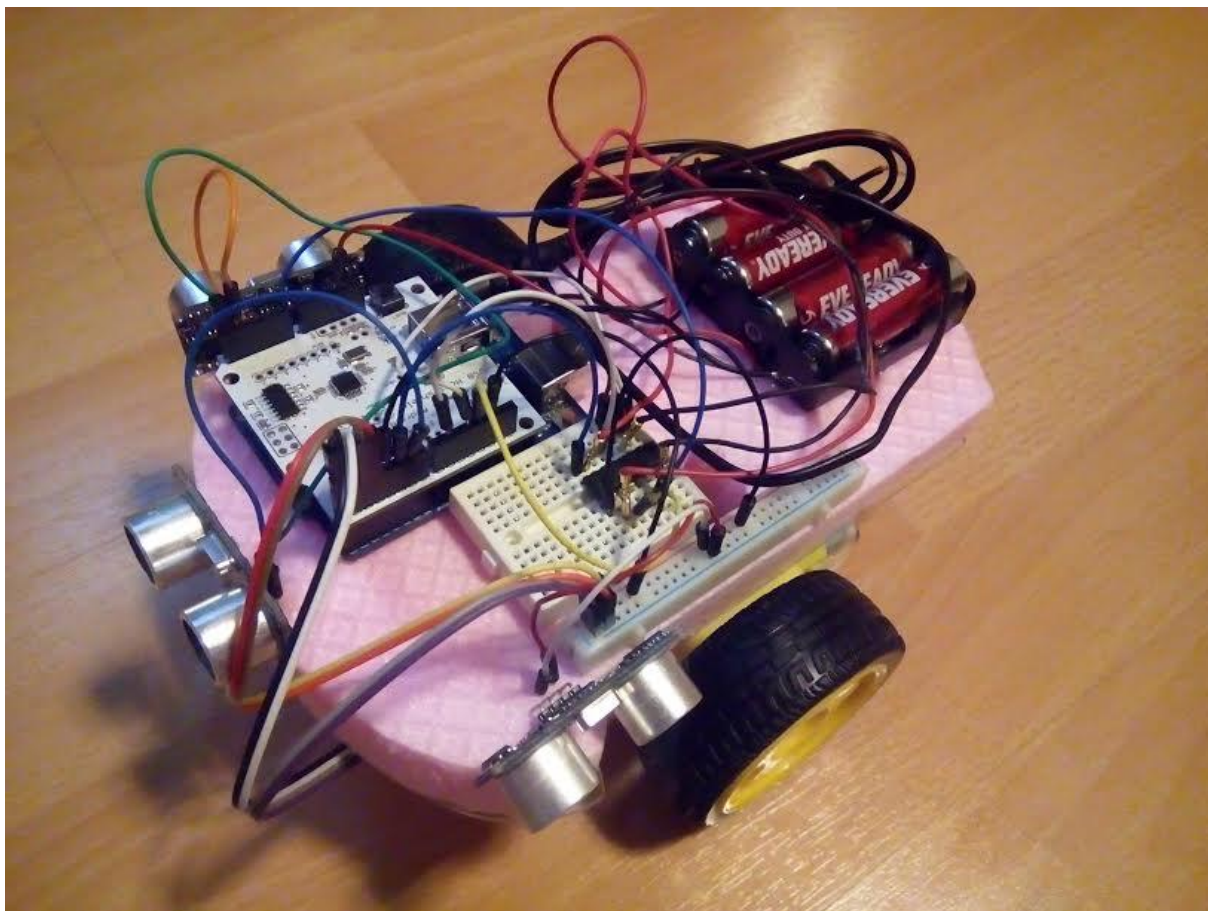


V literatuře [2] se vyskytují některé možné kritické scénáře a typy překážek, pro které implementovaný algoritmus vykazuje nepředvídatelné chování. Těmito překážkami mohou být překážky konkávního tvaru, kde robot po určité ujeté vzdálenosti podél stěny zjistí, že už je daleko, tak se otočí a zkusí hledat cestu opačným směrem a tak se dostane do smyčky. Takovéto chování bohužel nebylo možné ověřit a pokusit se podobnému chování zamezit.



*Obr. 40 Fotografie výsledného robotu – pohled zespodu.*





*Obr. 41 Fotografie výsledného robotu – pohled shora.*

Jako další cíl, mimo rámec této práce, je přesunout systém na větší podvozkovou platformu. S tím se spojuje použití lepších motorů se zabudovanými enkodéry pro lepší řízení. Dále připojení kamery a zajištění přenosu dat z kamery do ovladače. Dalším krokem bude rozšíření sensorického systému, připojením více ultrazvukových senzorů pro pokrytí kompletního okolí robotu a akcelerometru pro zdvojení vstupních dat pro odometrii.



## SEZNAM ZDROJŮ

- [1] NOVÁK, Petr. *Mobilní roboty: pohony, senzory, řízení*. 1. vyd. Praha: BEN - technická literatura, 2005, 243 s. ISBN 80-7300-141-1.
- [2] CHOSET, Howie M. *Principles of Robot Motion: Theory, Algorithms, and Implementation*. Cambridge, Mass.: MIT Press, c2005, xix, 603 p. ISBN 0262033275.
- [3] BORENSTEIN, Johann. *Navigating Mobile Robots: System and Techniques*
- [4] Arduino. [online]. [cit. 2015-05-15]. Dostupné z: [www.ARDUINO.CC](http://www.arduino.cc)
- [5] Robot Atlas. BostonDynamics. [online]. 2013 [cit. 2015-05-15]. Dostupné z: [http://www.bostondynamics.com/robot\\_Atlas.html](http://www.bostondynamics.com/robot_Atlas.html)
- [6] [www.robotstore.cz](http://www.robotstore.cz)
- [7] Robotika.cz. robotika.cz. [online]. 22.1.2005 [cit. 2015-05-15]. Dostupné z: [www.robotika.cz](http://www.robotika.cz)
- [8] Kreidl Marcel, Šmíd Radislav: *Technická diagnostika - senzory, metody, analýza signálu*, BEN - technická literatura, 2006, ISBN 80-7300-158-6
- [9] BENEŠ, Pavel. *Automatizace a automatizační technika: systémové pojetí automatizace*. 1. vyd. Brno: Computer Press, 2012-2014, 2 sv. (217; 241 s.). ISBN 978-80-251-3628-7.
- [10] BENEŠ, Pavel. *Automatizace a automatizační technika: prostředky automatizační techniky*. 5., rozšířené a aktualizované vyd. Brno: Computer Press, 2014, 304 s. ISB N 978-80-251-3747-5.
- [11] MATOUŠEK, David. *Práce s mikrokontroléry ATMEL*. 1. dotisk 2. vyd. Praha: BEN - technická literatura, 2004, 263 s.  $\mu$ C & praxe. ISBN 80-7300-094-6.
- [12] Inside Asimo Robotics. Honda. [online]. [cit. 2015-05-15]. Dostupné z: [HTTP://ASIMO.HONDA.COM/INSIDE-ASIMO/](http://asimo.honda.com/inside-asimo/)



## SEZNAM OBRÁZKŮ

Obr. 1 ASIMO [12].	11
Obr. 2 Mini AERCam [2].	12
Obr. 3 ATLAS [5].	12
Obr. 4 Schéma diferenciálního řízení.	13
Obr. 5 Schéma trojkolového podvozku [3].	14
Obr. 6 Schéma Ackermanova podvozku [3].	14
Obr. 7 Modelářský DC motorek [7].	15
Obr. 8 Schéma krokového motor [7].	16
Obr. 9 Schéma modelářského servomotoru [7].	16
Obr. 10 Mikrokontroler [7].	17
Obr. 11 Příklad potenciálového pole [7].	19
Obr. 12 Pseudo-Voronoi diagram [7].	19
Obr. 13 Graf viditelnosti [7].	20
Obr. 14 Bug 1 Algoritmus s úspěšné dorazení do cíle [2].	21
Obr. 15 BUG2 Algoritmus [2].	22
Obr. 16 Tangent BUG [2].	23
Obr. 17 Ukázka trasy robotu pomocí BUG tangent algoritmu [2].	24
Obr. 18 Schéma umístění ultrazvukových senzorů.	24
Obr. 19 Blokové schéma robotu.	26
Obr. 20 Layout robotu.	27
Obr. 21 Podvozek ve složeném stavu [6].	28
Obr. 22 Kinematické schéma robotu.	28
Obr. 23 Arduino UNO R3 [4].	30
Obr. 24 Diagram desky Arduino [4].	31
Obr. 25 Arduino IDE [4].	32
Obr. 26 USB Host Shield [4].	34
Obr. 27 Dualshock 4.	34
Obr. 28 DC motor, 1:48.	35
Obr. 29 L293D Layout.	36
Obr. 30 L293D Čip.	36
Obr. 31 Příklad PWM [7].	37
Obr. 32 Senzor HC-SR04.	38
Obr. 33 HC-SRF04 Časový diagram [1].	38
Obr. 34 Signál z enkodéru [7].	39
Obr. 35 Foto-elektrický senzor [6].	39
Obr. 36 Disk Enkodéru.	39
Obr. 37 Algoritmus manuálního ovládání.	40
Obr. 38 Algoritmus autonomního pohybu.	41
Obr. 39 Algoritmus objíždění překážky.	42
Obr. 40 Fotografie výsledného robotu – pohled zespodu.	44
Obr. 41 Fotografie výsledného robotu – pohled shora.	45