

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-5382-87080

DLAŽKOVO OVLÁDANÝ ROBOT TYPU HEXAPOD
BAKALÁRSKA PRÁCA

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-5382-87080

DIAĽKOVO OVLÁDANÝ ROBOT TYPU HEXAPOD
BAKALÁRSKA PRÁCA

Študijný program:	Aplikovaná informatika
Číslo študijného odboru:	2511
Názov študijného odboru:	9.2.9 Aplikovaná informatika
Školiace pracovisko:	Ústav robotiky a kybernetiky
Vedúci záverečnej práce:	Ing. Juraj Slačka, PhD.



ZADANIE BAKALÁRSKEJ PRÁCE

Študentka: **Natália Klementová**
ID študenta: 87080
Študijný program: aplikovaná informatika
Študijný odbor: 9.2.9. aplikovaná informatika
Vedúci práce: Ing. Juraj Slačka, PhD.
Miesto vypracovania: Ústav robotiky a kybernetiky

Názov práce: **Diaľkovo ovládaný robot typu hexapod**

Jazyk, v ktorom sa práca vypracuje: slovenský jazyk

Špecifikácia zadania:

Cieľom tejto práce je navrhnuť konštrukciu mobilného robota – pavúka so šiestimi nohami. Tento robot bude schopný pohybu, pričom tento pohyb bude riadený bezdrôtovo pomocou mobilnej aplikácie.

Úlohy:

1. Navrhnete vhodnú konštrukciu.
2. Vyberte vhodný hardvér a navrhnete komunikačný protokol.
3. Naprogramujete mobilnú aplikáciu pre diaľkové ovládanie.
4. Zhodnotíte výsledok.

Zoznam odbornej literatúry:

1. Adrian Tulbure – Emilian Ceuca – Adrian Ilea, Electronically controlled hexapod prototype for emergency situations, ECAI 2017, ISBN: 978-1-5090-6458-8, June 2017
2. Junke Li – Yujun Wang – Ting Wan, Design of a hexapod robot, CECNET 2012, ISBN: 978-1-4577-1415-3, April 2012

Riešenie zadania práce od: 17. 09. 2018

Dátum odovzdania práce: 13. 05. 2019

Natália Klementová
študentka

prof. Ing. Ján Murgaš, PhD.
vedúci pracoviska

prof. Dr. Ing. Miloš Oravec
garant študijného programu

SÚHRN

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Študijný program :	Aplikovaná informatika
Bakalárska práca:	Diaľkovo ovládaný robot typu hexapod
Autor:	Natália Klementová
Vedúci záverečnej práce:	Ing. Juraj Slačka, PhD.
Miesto a rok predloženia práce:	Bratislava 2019

Cieľom tejto práce je zostrojiť šesťnohého robota, schopného pohybu na diaľkové ovládanie. V tejto práci je popísaná problematika hexapodov, ich výhody, nevýhody a zhodnotenie súčasného stavu ukázaním existujúcich príkladov. Je potrebné navrhnuť a namodelovať vhodnú konštrukciu robota, jeho tela a nôh a vybrať adekvátne počet stupňov voľnosti. Následne je model vytlačený v 3D tlačiarňi. Na základe viacerých faktorov je vybraný hardvér pozostávajúci z analógových servo motorov, servo dosiek, mikropočítača, regulátora napätia a napájania. Hardvér je vsadený do modelu a naprogramovaný v prostredí Arduino. Komunikácia medzi robotom a mobilnou aplikáciou prebieha pomocou UDP datagramov. Pri programovaní softvéru robota je použitá inverzná kinematika, pričom robot je schopný chôdze rovno, do strán a otočenia na mieste. Aplikácia na diaľkové ovládanie je naprogramovaná pre mobilné telefóny s OS Android.

Kľúčové slová: hexapod, android, arduino, diaľkové ovládanie

ABSTRACT

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION
TECHNOLOGY

Study Programme:	Applied Informatics
Bachelor Thesis:	Remote controlled hexapod robot
Autor:	Natália Klementová
Supervisor:	Ing. Juraj Slačka, PhD.
Place and year of submission:	Bratislava 2019

The aim of the thesis is to create a six-legged robot which would be able to move and be controlled remotely. The problematics of hexapods, their advantages and disadvantages and the analysis of the current situation in this field is described. We had to design a suitable construction for the body and legs and choose enough degrees of freedom. The model was printed with the use of 3D printer. There are several factors we bear in mind while choosing the hardware. We choose hardware composed of analog servo motors, servo shields, microcontroller, voltage regulator and external power source. It is assembled with the printed parts and programmed in Arduino IDE. The communication between the robot and the phone application is handled with the help of UDP datagrams. The movement of the robot depends on inverse kinematics. It can move forward, to the sides and turn in place. The application for the remote controlling is programmed for phones with Android OS.

Key words: hexapod, android, arduino, remote control

Vyhlásenie autora

Podpísaná Natália Klementová čestne vyhlasujem, že som Bakalársku prácu Diaľkovo ovládaný robot typu hexapod vypracovala na základe poznatkov získaných počas štúdia a informácií z dostupnej literatúry uvedenej v práci.

Uvedenú prácu som vypracovala pod vedením Ing. Juraj Slačka, PhD..

V Bratislave dňa 21.05.2019

.....

podpis autora

Pod'akovanie

Ďakujem vedúcemu bakalárskej práce Ing. Jurajovi Slačkovi, PhD. za jeho čas, odborné vedenie, poskytnuté konzultácie a spätnú väzbu. Moje pod'akovanie zároveň patrí všetkým, ktorí mi pomáhali s dostupnosťou či už knižných, alebo internetových zdrojov, ale aj tým, ktorí ma v mojom štúdiu a vytrvalej práci podporovali.

Obsah

Úvod	1
Mobilná robotika	3
1.1 Podvozky	4
1.1.1 Kráčajúce podvozky	5
1.1.2 Počet končatín	5
1.1.3 Pohony	6
1.1.4 Počet kĺbov	7
1.1.5 Stabilita	8
1.2 Hexapod	9
1.3 Zhodnotenie súčasného stavu v oblasti kráčavých robotov	10
Mechanická konštrukcia	14
1.1 Telo	14
1.2 Končatiny	15
1.3 Hotová konštrukcia	17
Hardvér	19
1.1 Tower Pro SG90	19
1.2 PCA9685	20
1.3 WeMos D1 mini	22
1.4 Napájanie	23
1.5 YEP 20A HV (2~12S) SBEC	25
1.6 Zapojenie	26
Komunikačný protokol	27
1.1 TCP	27
1.2 UDP	28
1.3 Implementácia	28

1.3.1	Arduino.....	29
1.3.2	Android.....	29
1.3.3	Datagramy	29
Softvér hexapoda		31
1.1	Inverzná kinematika	31
1.2	Implementácia	33
1.2.1	Kalibrácia servo motorov	34
1.2.2	Trieda Joint	34
1.2.3	Trieda Leg	35
1.2.4	Globálne metódy	35
1.3	Chôdza.....	36
1.3.1	Jednonohá chôdza 5+1	37
1.3.2	Dvojnohá chôdza 4+2.....	37
1.3.3	Trojnohá chôdza 3+3	37
1.3.4	Postavenie sa a sadnutie	37
1.3.1	Rotácia na mieste.....	38
1.3.2	Chôdza rovno.....	38
1.3.3	Chôdza do strán	39
Softvér na diaľkové ovládanie		41
1.1	Používateľské rozhranie	41
1.2	Spracovanie údajov.....	41
Záver		43
Zoznam použitej literatúry		44
Prílohy		i
Príloha A: Obsah CD		ii

Zoznam obrázkov a tabuliek

Obrázok 1 Robot s jednou nohou z roku 1938 (3).....	6
Obrázok 2 Prípady stability hexapoda: staticky stabilný, staticky nestabilný a kriticky stabilný (8).....	8
Obrázok 3 a) Atlas (10) b) Spot Mini (11) c) RHex (12)	11
Obrázok 4 a) Phoenix (13) b) HEXA (14).....	12
Obrázok 5 3D modely zobrazujúce a) vonkajšiu časť tela b) vnútro vrchnej časti tela ...	15
Obrázok 6 3D model zobrazujúci a) holennú kosť b) stehennú kosť	16
Obrázok 7 Pohľad na 3D model kĺbu z dvoch strán	17
Obrázok 8 Model zloženého hexapoda v sede	17
Obrázok 9 Model zloženého robota v základnej stojacej polohe	18
Obrázok 10 Servo motor Tower Pro SG90 9g (17)	20
Obrázok 11 Perióda PWM signálu pre servo motor Tower Pro (18)	20
Obrázok 12 Servo driver PCA9685 (20)	21
Obrázok 13 WeMos D1 Mini (22).....	23
Obrázok 14 Nabitá LiPo batéria s alarmom	25
Obrázok 15 YEP 20A SBEC (25)	25
Obrázok 16 Schéma zapojenia.....	26
Obrázok 17 Rozmiestnenie servo motorov v modely.....	26
Obrázok 18 Wireshark – komunikácia PC pripojeného k prístupovému bodu hexapoda	30
Obrázok 19 Pohľad na nohu zboku (31).....	32
Obrázok 20 Pohľad na nohu zhora (31).....	33
Obrázok 21 Analógové servá nastavené na rovnakú hodnotu PWM signálu (33)	34
Obrázok 22 a) rozloženie elementov b) hotová aplikácia	42
Tabuľka 1 Význam posielačných datagramov	30

Zoznam skratiek a značiek

UDP – Používateľský datagramový protokol (User Datagram Protocol)

TCP – Protokol riadenie prenosu (Transmission Control Protocol)

2S – 2 články v sérii (2 cells in Series)

LiPo – lítium-polymérový

PWM – pulzná šírková modulácia (Pulse Width Modulation)

LED – svetelná dióda (Light-Emitting Diode)

GND – uzemnenie (ground)

IMUU – jednotka merania zotrvačnosti

CAD – počítačom podporovaný dizajn (Computer-aided design)

3D – trojdimenzionálny

IP – Internet Protocol

IDE – integrované vývojové prostredie (Integrated development environment)

WiFi – bezdrôtová vernosť (wireless fidelity)

OS – operačný systém (operating system)

API – rozhranie pre programovanie aplikácie (application programming interface)

Úvod

V dnešnom, modernom svete technológie ovládajú naše životy. Stačí sa prejsť po ulici a všimnúť si ľudí zahľadených do mobilných telefónov, v ktorých hľadajú odpovede na otázky, ktoré im napadli pred pár sekundami, či na autá, ktoré sa učia jazdiť samostatne bez potreby ľudského zásahu. Vždy ma fascinovali výdobytky modernej doby a ich fungovanie. Spôsob akým napríklad priemyselné roboty s ľahkosťou a presnosťou vykonávali svoje úlohy ma zaujal na toľko, že som si kvôli tomu vybrala bakalársku prácu, ktorá sa zaoberá robotikou. Táto bakalárska práca je pre mňa ideálne prepojenie elektroniky a informatiky s možnosťou priučenia sa niečomu novému.

Robot je stroj, ktorý je navrhnutý na automatické vykonávanie jednej alebo viacerých úloh. Zvyčajne je to elektromechanický systém, ktorý je ovládaný počítačovým programom. Roboty môžeme stretnúť v reálnom aj vo virtuálnom svete, avšak robotov z druhej možnosti skôr zaradíme medzi botov. Ich využitie je rôznorodé a najčastejšie ich môžeme vidieť vykonávať práce, ktoré sú pre človeka nebezpečné, nepríjemné alebo periodicky sa opakujúce. Za posledné desaťročie sa rozmohli najmä autonómne roboty.

Autonómny robot je robot, ktorý dokáže samostatne vykonávať požadované úlohy bez ľudskej intervencie v neštruktúrovanom prostredí. Tieto roboty majú senzory na to, aby dokázali porozumieť svojmu okoliu. Využívajú sa pri záchranných, údržbových a prieskumných úlohách, kedy nasadenie človeka môže byť riskantné.

Ďalšou kategóriou sú roboty s implementovanou umelou inteligenciou, na základe ktorej dokážu získavať skúsenosti, učiť sa a prispôbovať svojmu prostrediu. Umelú inteligenciu rozdelíme na špecializovanú a všeobecnú, ktorú sa zatiaľ vyvinúť nepodarilo. Jej veľký prínos sa očakáva najmä vo vedeckých výskumoch. S týmto sa spája aj vývoj humanoidných robotov, ktorých pohybový aparát by sa dal prirovnáť tomu ľudskému.

Pri prieskumných aktivitách, ako sú napríklad misie na Marse, sa stáva, že terén je nerovnomerný k čomu je potrebné prispôbiť podvozok. Najčastejšie k svojmu pohybu po povrchu roboty využívajú kolesá, pás alebo končatiny inšpirované živočíchmi, ktoré príroda zdokonalila počas miliónov rokov evolúcie. Pri pohybe na hrboľatom povrchu majú najväčšiu výhodu kráčajúce roboty kvôli údom, ktoré dokážu zdvihnúť cez prekážky. Medzi takéto zariadenia patria aj hexapody, ktorých lokomočný mechanizmus tvorí šesť nôh a v každom momente pohybu sú stabilné.

Táto bakalárska práca sa zaoberá kráčajúcimi robotmi so zameraním na roboty typu hexapody. Hlavným cieľom je návrh a zostrojenie konštrukcie diaľkovo ovládaného robota, ktorý bude mať šesť nôh. Na každej nohe budú tri modelárske servá ovládané pomocou Wi-Fi mikročipu ESP8266. Telo bude vymodelované a vytlačené na 3D tlačiarňi. Robot sa cez internetové pripojenie bude dať ovládať pomocou mobilnej Android aplikácie. V prvej kapitole je stručný úvod k mobilnej robotike, hexapodom a komentár k súčasnemu stavu kráčajúcich robotov. Druhá kapitola rozoberá mechanickú konštrukciu. V tretej kapitole sa venujeme použitému hardvéru. Vo štvrtej kapitole popisujeme komunikačný protokol a v posledných dvoch kapitolách sa pozrieme na softvér robota a diaľkové ovládanie.

Výsledný hexapod bude slúžiť na otestovanie pohybových algoritmov pre šesťnohé roboty. Robot sa bude dať ovládať diaľkovo pomocou mobilnej aplikácie.

Mobilná robotika

Ako sme už spomínali, robotika je interdisciplinárne odvetie vedy a inžinierstva, ktoré v sebe spája strojárstvo, elektroniku, informačné technológie a iné odvetia za účelom vytvorenia mechanizmu, ktorý by dokázal ľuďstvu uľahčiť jeho prácu a život. Robotika sa vetví podľa viacerých oblastí záujmu, avšak vo všeobecnosti roboty delíme na stacionárne – priemyselné, ktoré sú pripevnené k zemi a mobilné – roboty schopné premiestňovania sa na iné miesta, čiže nie sú fixované len na jednu lokalitu.

Práve kategória mobilných robotov je tá, ktorá ľuďom napadne ako prvá, keď sa hovorí o tejto vednej disciplíne, či už kvôli filmom s tematikou science fiction alebo mnohým vedeckým a novinárskym publikáciám a prelomom, ktoré sa dejú čoraz častejšie v tejto progresívne rozširujúcej sa oblasti. Podľa spôsobu ich fungovania ich rozdelujeme na autonómne a diaľkovo ovládané roboty.

Názov nám napovedá, že autonómne roboty využívajú autonómiu čiže samostatnosť, ktorá sa u človeka prejavuje schopnosťou vytvárať svoje vlastné názory a u stroja tým, že dokáže rozlišovať svoje prostredie, orientovať sa v ňom bez pomoci a vykonávať úlohy na, ktoré bol naprogramovaný. Avšak ako odlíšiť autonómneho robota od toho bežného? S týmto by nám mohol pomôcť nasledovný príklad. Predstavme si pomocného robota na stavbe rodinného domu, ktorého cieľom je vyzdvihnúť tehly z prízemí a vyniesť ich na poschodie. Natvrdo mu bola naprogramovaná určitá cesta k danému cieľu. Ak by sme takémuto neautonómnemu robotu so senzormi postavili do cesty prekážku, s najväčšou pravdepodobnosťou by pred ňou zastavil a nevedel ako pokračovať ďalej. Na druhej strane autonómny robot by na základe svojich výstupov zo senzorov dokázal „vymyslieť“ ako prekážku obísť a dostať sa na požadované miesto. Programovanie takéhoto algoritmu je nielen zábavnejšie ale aj užitočnejšie, lebo jeho implementácia sa dá využiť v akomkoľvek prostredí. Medzi voľne predajné autonómne roboty patria napríklad kosačky alebo vysávače, ktoré dokážu detegovať hlinu na zemi a podľa toho vedia, že túto oblasť musia čistiť intenzívnejšie.

Ľudia bez technického vzdelania si môžu myslieť, že robot na diaľkové ovládanie vlastne ani nie je robot, keď nie je autonómny. Opak je však pravdou a roboty asistované operátorom sa využívajú napríklad vo vesmírnych výskumoch a vojenstve. Čoraz väčšiu popularitu získavajú u laickej verejnosti, ktorá si doma svojpomocne dokáže postaviť svoje

vlastné autíčka a drony. K svojmu fungovaniu využívajú mikrokontrolér (malý počítač), ktorý získava informácie zo senzorov a ovládača. Medzi najpopulárnejšie metódy bezdrôtového diaľkového ovládania zaraďujeme rádio frekvenciu a infračervené žiarenie. S infračerveným žiarením sa stretávame každý deň, napríklad v ovládači od televízora alebo bio lampách. Nevýhodou je potreba krátkej vzdialenosti a žiadnych prekážok medzi objektmi. Ovládač s rádiovou frekvenciou vysiela vlny, ktoré korešpondujú binárnemu kódu pre tlačidlo, ktoré stláčame. Rádio prijímač vlny zaregistruje a dekoduje. Výhodou je väčší rozsah a prechod signálu aj cez steny. Vo svojej práci budem na ovládanie takisto využívať rádio frekvenciu, o ktorej si bližšie povieme v kapitole o komunikačnom protokole.

1.1 Podvozky

V každej sfére našich životov je užitočné mať pomocníka, ktorý by nám dokázal poskytnúť pomoc pri vykonávaní určitých aktivít. Či sa bavíme o stroji, ktorý dokáže nájsť a vyhrabať človeka zasypaného lavínou alebo drona, ktorý dokáže skontrolovať rozsah požiaru bez potreby ľudskej prítomnosti, podľa typu prostredia, v ktorom operujú, podvozky mobilných robotov rozdeľujeme do troch hlavných kategórií:

- Kolesové
- Pásové
- Kráčajúce

Jednotlivé kategórie majú svoje výhody, nevýhody a využitie v dnešnom svete, avšak v našom prípade sa budeme zaoberať otázkami týkajúcimi sa kráčajúcich robotov. Každý typ podvozku čelí trom hlavným problémom: typ prostredia, vlastnosti dotyku so zemou a stabilita.

Pri prostredí treba brať do úvahy prostredie ako také (napríklad vzduch, voda, vákuum, zem), v ktorom sa robot bude pohybovať a jeho štruktúru (napríklad či je zem mäkká a poddajná alebo vysušená a tvrdá).

Atribúty, ktoré ovplyvňujú vlastnosti dotyku so zemou sú styčné body so zemou, čiže odtlačky robota, uhol dotyku robota so zemou a jeho trenie s povrchom.

Stabilita, ktorá je popísaná v ďalšej časti, závisí od množstva a geometrie styčných bodov, či je robot staticky alebo dynamicky stabilný, jeho ťažiska a stúpania terénu. (1)

1.1.1 Kráčajúce podvozky

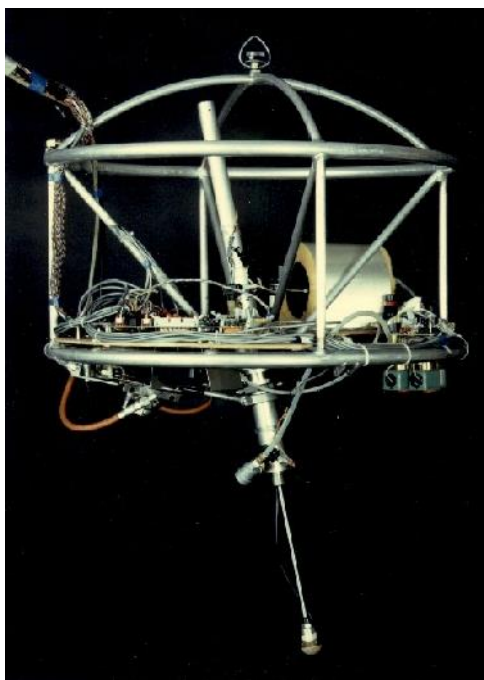
Dizajn mobilných robotov často krát býva inšpirovaný biologickými systémami, ktoré sú vo svojom jadre omnoho komplexnejšie, ak máme na mysli to z čoho sú zložené a ako fungujú. Ale prečo sa vôbec snažíme napodobniť prírodu pri navrhovaní mobilných robotov? Nemá ľudská civilizácia dostatok vedomostí na navrhnutie svojich vlastných, jedinečných dizajnov? Odpoveď na túto otázku je jednoduchá, áno má, ale prečo sa neinšpirovať prírodou, ktorá návrh končatín už vyriešila za nás? Pohybový aparát organizmov žijúcich na súši bol prispôbený k premiestňovaniu sa po nerovnom teréne, čo bol jeden z dôvodov ich prežitia.

Medzi dva hlavné typy pohybov, ktoré môžeme u suchozemských živočíchov pozorovať patria plazenie a chôdza. Plazenie sa, čiže pohybovanie sa vlnovitými pohybmi tela, je typické pre hady a dážďovky. Využívajú trenie medzi ich telom a povrchom, pomocou čoho sa hady dokážu pohybovať po stromoch či vode. Veľkou výhodou je nízka spotreba energie. Avšak častejšie sa u zvierat stretávame s chôdzou, pri ktorej využívajú nohy a ktorá je aj nám ľuďom dôverne známa. Práve pri nerovnomerných povrchoch ako sú napríklad listnaté lesy alebo kamenisté pohoria sa naplno prejavuje potenciál nôh ako vhodný spôsob premiestňovania sa. Pomocou nôh sa zviera dokáže plaziť, plávať, behať, preskakovať prekážky a iné, čo je na proti tomu spojené so zvýšenou spotrebou energie a komplikovanejšou implementáciou do strojov. Je potrebné si uvedomiť, že napríklad pri behu je len jedno chodidlo na zemi v danom okamžiku, pričom centrum nervovej sústavy zvieraťa zabezpečuje stabilitu tela, spracúva a vyhodnocuje informácie zo zmyslových orgánov a vykonáva desiatky ďalších aktivít. (2)

1.1.2 Počet končatín

Je všeobecne známe, že živočíchy sa pohybujú prostredníctvom dvoch až viacerých končatín. Žiaden z nich nemá prirodzene len jednu končatinu a to z jednoduchého dôvodu – stabilita. Takéto stvorenie by muselo väčšinu svojej energie venovať rovnováhe, nehovoriac o problémoch s premiestňovaním sa na iné miesta. Avšak pri konštruovaní strojov sa na danú problematiku pozeráme z iného uhlu. V roku 1983 bol zostrojený prvý jednonohý robot Marcom Raibertom na MIT (Massachusetts Institute of Technology). Pohyboval sa skákaním na jednej nohe, ktorá bola konfigurovateľnej dĺžky a poháňaná pneumatickým pohonom. Noha bola pripevnená k telu, ktoré bolo zložené zo senzorov, elektroniky a z boku

poháňané hydraulikou. Jeho zostrojenie bolo odôvodnené viacerými výhodami. Je jednoduchšie študovať dynamickú rovnováhu na jednej končatine, kvôli tomu, že nie je potrebné ju spájať a programovať s ďalšími končatinami. Konštrukčný čas bol nižší a robot pracoval spoľahlivejšie, lebo nebolo nutné pracovať s toľkým vybavením, pričom vzor jeho fungovania by sa dal využiť pri podvozkoch s viacerými končatinami. Tieto experimenty ukázali, že rovnováha sa dá docieľiť aj pomocou jednoduchého kontrolného systému. (3)



Obrázok 1 Robot s jednou nohou z roku 1938 (3)

1.1.3 Pohony

Na to aby končatiny boli schopné pohybu je nutné ich poháňať pomocou pohonov, inak nazývaných aj ako aktuátory, akčné členy, či motory. Prostredníctvom vyvolanej udalosti vo forme napríklad stlačeného tlačidla je možné naklápať, nastavovať či zdvíhať záťaž v podobe končatiny na rôzne pozície. Pohony rozdeľujeme na lineárne, schopné ťahania a tlačenia, a rotačné. Aktuátory podľa typu kontrolného signálu môžu byť hydraulické, pneumatické alebo elektrické.

Pneumatický pohon využíva stlačený vzduch na generovanie pohybovej energie. Množstvo sily, ktorý dokáže aplikovať na jednotku plochy je menej ako hydraulický, ale viac ako elektrický, majú rýchlu odozvu, vyžadujú najmenej údržby a nedochádza k ich poškodeniu pri preťažení. Nevýhodou je, že sa nemôžu používať pri vysokom

atmosférickom tlaku. Plyny sa rozpínajú a sú stlačiteľné s čím je spojená nežiaduca nelinearita pohybu. Vývojom v tejto oblasti sa redukujú výdavky na výrobu priemyselných robotov a robotov nevyžadujúcich také množstvo sily. (4)

Hydraulické akčné členy sú založené na teórii nestlačiteľnosti kvapalín. Konvertujú hydraulickú energiu na využiteľnú mechanickú energiu. Majú veľkú šírku frekvenčného pásma, rýchlu odozvu a skvelú precíznosť. Takisto sú schopné vygenerovať najviac sily na jednotku plochy. Kvôli týmto vlastnostiam sa využívajú na aplikácie vyžadujúce nadmernú silu najmä v strojárskom a ťažkom priemysle. Svoju cestu si hľadajú aj k robotickým aplikáciám a kvôli tomu, že tieto aktuátory sú kompaktné, dlho vydržia a svojou silou a rýchlosťou sa dajú prirovnať k ľudským svalom. Preto sa využívajú na vývoj humanoidov. Medzi nevýhody zaraďujeme vysokú cenu, 38% neúčinnosť, pri ktorej sa stráca energia, nutné chladenie motorov pri dlhodobom používaní či potencionálne znečistenie prostredia hydraulickou kvapalinou. (5)

Najvyužívanejšími pohonmi sú pohony **elektrické**. Ich ovládanie je jednoduché, sú schopné vysokej presnosti a spoľahlivosti pri sledovaní špecifickej trajektórie, majú rýchlu odozvu a k tomu všetkému aj nízku spotrebu. Na druhej strane vyžadujú pravidelnú údržbu, ľahko sa pokazia pri preťažení a takisto dochádza k ich prehrievaniu pri nadmernom používaní. Množstvo sily, ktoré sú schopné vyprodukovať na určitú plochu je veľmi nízke a preto je potrebné použitie mechanických prevodoviek (4). Najvhodnejšie je ich aplikovať do robotov, ktoré nevyžadujú vysokú rýchlosť a silu, ale skôr precíznosť kvôli mnohým opakovaniam, ako je to u priemyselných robotov.

Rozdeľujeme ich do nasledovných kategórií:

1. Jednosmerné (krokové) motory
2. Striedavé motory
3. Servomotory

1.1.4 Počet kĺbov

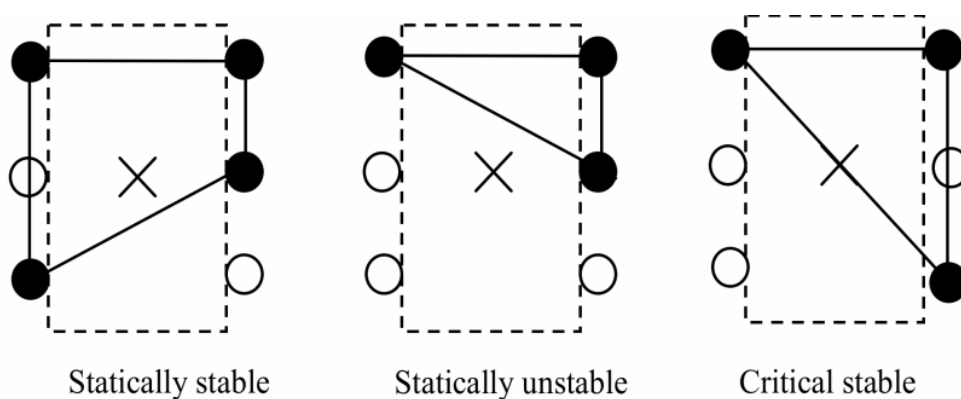
Pri návrhu končatiny sa môžeme rozhodnúť či si zvolíme článkovanú nohu alebo teleskopickú nohu. Pri prvej možnosti potrebujeme aspoň stupne voľnosti na to aby sa noha mohla pohybovať horizontálne a vertikálne. Ak chceme aby bol robot schopný pohybovať sa v akomkoľvek smere (dola, doprava) musíme mu pridať ďalší stupeň voľnosti, čo sú

v súčte tri kĺby. V závislosti od aplikácie je ich možné použiť viac napríklad na manipuláciu s objektmi. Dôležité je ich umiestnenie v konštrukcii. (6)

1.1.5 Stabilita

Dôležitou vlastnosťou pohybujúcich sa strojov je schopnosť udržať rovnováhu a neprevrátiť sa. Preto stabilitu kráčajúcich podvozkov môžeme rozdeliť na statickú a dynamickú.

Statická stabilita znamená, že robot dokáže vydržať stáť na mieste bez pádu, v každom časovom okamihu i keď nie je v pohybe alebo jeho pohyb bol zastavený v ľubovoľnej pozícii. Ďalšia definícia hovorí o tom, že robot je považovaný za staticky stabilný práve vtedy, keď sa vo všetkých časových úsekoch projekcia ťažiska nachádza vo vnútri konvexného polygónu určeného končatinami robota, ktoré sa v danom okamihu dotýkajú podložky (v anglickej literatúre sa využíva pojem „support polygon“) (7). Aby sme si bližšie vysvetlili, čo sa skrýva pod pojmom konvexný polygón predstavme si stoličku s tromi nohami. Nohy stoličky dotýkajúce sa zeme vytvárajú trojuholník, pričom stolička je staticky stabilná práve vtedy ak sa jej hmotný stred nachádza v tomto trojuholníku a jeho plocha je nenulová. O kritickej stabilite hovoríme práve vtedy, keď sa ťažisko daného telesa nachádza na úsečke konvexného polygónu spájajúcej dva vrcholy.



Obrázok 2 Prípady stability hexapoda: staticky stabilný, staticky nestabilný a kriticky stabilný (8)

Dynamická stabilita opisuje schopnosť robota udržať sa v vzpriamenom postoji počas pohybu. Na splnenie tejto definície je od robota vyžadované jeho neustále vyvažovanie sa, ktoré sa uplatňuje pri väčšine dvojnohých konštrukciách. Keď sú obe nohy na zemi je ich

konvexný polygón len priamka, kvôli relatívne malým chodidlám, ktorá sa pri chôdzi mení na jediný bod. Pri pohybe sa takisto ťažisko presúva z jednej nohy na druhú. Spomenuté dôvody sú dostatočne závažné na to, aby vývoj humanoidov kopírujúcich ľudskú chôdzu bol pre inžinierov problematický.

Kedy môžeme hovoriť o staticky stabilnej chôdzi? Chôdza kráčajúceho robota je považovaná za staticky stabilnú práve vtedy ak má minimálne štyri končatiny (pri chôdzi musí byť aspoň jedna noha vo vzduchu a tri na podložke). Jediným problémom štvornohej konštrukcie je jej rýchlosť. Preto sa v praxi skôr stretneme so šesťnohými konštrukciami, ktoré naraz môžu mať vo vzduchu až tri nohy. (1)

1.2 Hexapod

Konštrukcia robota je zložitá výzva, menovite pri kráčajúcich robotoch, kde treba brať do úvahy statickú a dynamickú stabilitu. Ak tieto atribúty nie sú splnené, robot sa nedá využiť na žiadnu aplikáciu. Slovo hexapod pochádza z gréckych slov hex (šesť) a pous (chodidlo) a opisuje šesťnohý hmyz. V posledných desaťročiach sa tento pojem začal spájať s robotikou, kde charakterizuje šesťnohé podvozky, ktoré vo svojom jadre spĺňajú pravidlá stability.

Rovnakým menom sa častuje aj Stewartova platforma, čo je plošina podopieraná šiestimi ramenami, umiestenými po dvojiciach do trojuholníkov. Hydraulikou poháňané ramená nastavujú pozíciu plošiny. Najčastejšie sa s ňou môžeme stretnúť v leteckých simulátoroch a preto je dôležité rozlišovať medzi danými mechanizmami.

V šesťdesiatych rokoch minulého storočia Space General Corporation zostrojila externe napájaného hexapoda, čo je datované ako prvé zostrojenie tohto typu vôbec. Jeho pohyb bol koordinovaný pomocou kamier. Kvôli nedostačujúcemu množstvu kĺbov a schopnosti pohybu v nerovnomerných terénoch nemohol byť využitý na dôvod, kvôli ktorému bol postavený – prieskum Mesiaca.

Prvý šesťnohý robot ovládaný digitálne cez počítačový program bol OSU Hexapod. Bol skonštruovaný v roku 1977 tímom vedcov z Ohio State University na štúdium ovládacích algoritmov. V každej nohe mal tri stupne voľnosti poháňané elektrickým motorom a celkovo vážil 136 kg. (9)

Ďalej by sme opísali zjavné výhody a nevýhody, ktoré motivovali vedcov k vývoju tejto oblasti.

Výhody:

1. Staticky a dynamicky stabilný v každom časovom okamihu
2. Schopnosť pohybovania sa v extrémne členitom teréne (dokáže sa pohybovať po schodoch, svahu, prekročiť škáru či vyliezť na ľubovoľnú prekážku primerane k svojim schopnostiam)
3. Pri poškodení končatín sa dokáže stále pohybovať
4. Končatiny môže využiť na manipuláciu s objektmi
5. Malá styčná plocha so zemou vďaka čomu prostredie zostáva nedotknuté

Nevýhody:

1. Vysoká cena kvôli potrebe použitia viacerých pohonov
2. Zložitosť hardvérovej a softvérovej implementácie
3. Komplikované „oživovanie“
4. Nízka rýchlosť na rovnom povrchu oproti kolesovým a pásovým mechanizmom
5. Každý krok je sprevádzaný nárazovou silou

Aby sa nepriaznivý vplyv nevýhod minimalizoval, inžinieri začali experimentovať s tzv. hybridmi. Hybridný robot v sebe spája výhody dvoch alebo viacerých podvozkov ako sú napríklad končatiny s chodidlami v podobe kolies spolu s vrtuľami drona.

1.3 Zhodnotenie súčasného stavu v oblasti kráčavých robotov

Väčšina výskumu a vývoja nových technológií sa uskutočňuje v Severnej Amerike a Východnej Ázii. Medzi popredných vývojárov kráčajúcich robotických systémov zaraďujeme americké firmy Agility Robotics, Dynamic Legged Locomotion a fenomenálnu organizáciu Boston Dynamics, ktorá sa tejto oblasti venuje 26 rokov. Z jej dielne pochádzajú nasledujúce výtvary.

Atlas, verejne oznámený v roku 2011, je doposiaľ celosvetovo najagilnejší humanoid, ktorý „uzrel“ svetlo sveta. Pohyby ramien, torza a nôh sú koordinované pomocou kontrolného systému čo mu zaručuje možnosť narábania s jeho celým telom. Rovnováhu si dokáže udržať pri chôdzi po nerovnomernom povrchu, behu, robení salta dozadu, či pod vplyvom externých nárazov, ktoré ho chcú zhodiť na zem. Ak už aj spadne, tak sa dokáže s ľahkosťou opäť postaviť. Hydraulikou je poháňaných jeho 28 kĺbov. Meria 150 cm a váži

75 kg, pričom je vybavený senzormi a kamerami. Jeho výrobcovia si ho predstavujú ako hlavnú postavu pri záchranných akciách, ktorých náтура je príliš nebezpečná pre ľudí.(10)

Najnovším prírastkom je štvornohý robot **SpotMini**, ktorého výzor bol inšpirovaný psovitou šelmou. Jeho existencia bola oznámená v roku 2016. Váži 25 kg (30 kg vrátane päť kĺbového ramena) a vyznačuje sa svojou tichou prevádzkou. SpotMini je vybavený stereo a hĺbkovými kamerami, IMU (jednotkou merania zotrvačnosti) a senzormi, ktoré mu pomáhajú s navigáciou. Keďže je plne elektrický je vhodný do interiérov, kde Vám rád pomôže s otvorením dverí. Robot je ovládaný pomocou diaľkového ovládača, ale zároveň je aj plne autonómny. V aktuálnom roku 2019 by sa malo dostať na trh 100 kusov, čím sa stane prvým produktom tejto spoločnosti, ktorý sa bude predávať. (11)

Ďalší autonómny robot spoločnosti Boston Dynamics vyvinutý na začiatku 21. storočia má názov **RHex**. Dosahuje rýchlosť 9 km/h, prekoná 45 stupňové stúpanie a dokonca demonštroval aj plávanie. So svojimi šiestimi, nezávisle ovládanými nohami dokáže svoju chôdzu prispôbiť drsnému terénu s minimálnou potrebou zásahu od operátora. Každá noha je ovládaná rotačným elektrickým pohonom pripevneným k telu. Je to prvý zdokumentovaný robot, ktorý preukazoval všeobecnú mobilitu v teréne. Nerobí mu problém žiadne prostredie od schodov a skál až po piesok a blato. Jeho výška je 14 cm a váži 12 kg. (12)



Obrázok 3 a) Atlas (10) b) Spot Mini (11) c) RHex (12)

Spoločnosť Lynxmotion sa vyše 15 rokov zaoberá výrobou robotických súprav pre entuziastov, ktorí si ich doma môžu poskladať a na výučbu do škôl. Zaraďujeme sem robotické ramená, kolesové, pásové a kráčajúce konštrukcie vrátane hexapodov, ktoré majú tri elektricky poháňané kĺby servomotormi od firmy Hitec na každej nohe, pevnú hliníkovú

konštrukciu, pričom pomocou aplikácie PowerPod je vytváranie programov jednoduché aj pre laikov. Na bezdrôtové diaľkové ovládanie sa využíva PS2 ovládač. K ich najpopulárnejším hexapodom patrí model **Phoenix**, ktorého dizajn tela je prirodzenejší a najviac sa podobá hmyzu v porovnaní s ostatnými modelmi. (13)

HEXA je šesťnohý, kompaktný, vysoko agilný robot spoločnosti Vincross, ktorého cieľom od roku 2016 je zoznámenie sa širokej spoločnosti s robotikou. Vďaka operačnému systému MIND založenom na Linuxe, ktorý je zároveň aj SDK (nástroj na vývoj softvéru) nie je jeho programovanie nič zložitým a zvládnu ho aj začiatočníci. K jeho senzorum, ktoré môžu byť rozšírené o ďalšie, patrí 720p kamera s nočným videním, infračervený vysielateľ, senzor na meranie vzdialenosti a trojosový akcelerometer. K svojmu fungovaniu využíva pripojenie na internet a vo všeobecnosti sa HEXA teší veľkej popularite. (14)



Obrázok 4 a) Phoenix (13) b) HEXA (14)

Predpokladá sa, že v nasledujúcich dvadsiatich rokoch dôjde k obrovskému nárastu vo výrobe a predaji interiérových robotov na celosvetovom trhu. Na mysli nemáme továrenské roboty, ale mechanizmy, ktorých úloha bude uľahčiť život bežných ľudí. Na to aby sa táto predpoveď splnila je potrebné naďalej študovať ich mobilitu, aby sa dokázali efektívne pohybovať a vykonávať svoje úlohy. Navyše netreba zabúdať ani na hardvér. Hardvér je jedným z brzdiacich činiteľov pri vývoji umelej inteligencie, ktorá potrebuje obrovskú výpočtovú silu napríklad pri tréňovaní neurónových sietí. Dnešné najvýkonnejšie počítače dosahujú inteligenciu porovnateľnú so schopnosťami mačiek, i keď súčasné mikroprocesory pracujú o šesť rádov (10^6) rýchlejšie ako neuróny. Ako môžeme tušiť toto je pre všeobecnú umelú inteligenciu nedostačujúce. Ale tak ako je už niekoľko ročným zvykom i elektronika sa bude hnať dopredu, čoho výsledkom budú rýchlejšie procesory, presnejšie a výkonnejšie

servo motory, či batérie, ktoré v sebe budú skladovať obrovské množstvá elektrickej energie. Ďalším faktorom je síce namáhavé, ale dôležité zníženie cien, tak aby boli dostupné širokej verejnosti a aby sa systémy dostali zo sterilného prostredia vývojárov do spoločnosti. Takisto robotické systémy implementované do protéz spolu s exoskeletmi by výrazne zlepšili kvalitu života hendikepovaných ľudí. V súčasnosti prebieha štvrtá priemyselná revolúcia, ktorej jedným z viacerých spúšťačov je práve robotika a je len na nás ľuďoch ako celý tento proces ovplyvní našu existenciu.

Mechanická konštrukcia

Jednou z úloh našej práce je navrhnuť vhodnú mechanickú konštrukciu, ktorá bude slúžiť ako pevný základ pre nasledovné implementovanie hardvéru a ďalších funkcionalít. Dizajn závisí aj od nasledujúcich činiteľov a preto je potrebné ich zobrať do úvahy: veľkosť robota, aktuátory, napájanie, záťaž, autonómia a okrem iného aj rozpočet. Pri návrhu konštrukcie sme využívali voľne dostupný CAD softvér s názvom DesignSpark Mechanical, ktorý 3D modely exportuje priamo do formátu STL, potrebného pri 3D tlači. Následne pomocou programu pre vytváranie rezov Ultimaker Cura sme vygenerovali G-kód pre danú tlačiareň. Na tlač prototypu bola využitá školská 3D tlačiareň Original Prusa i3 MK2. Pri tlači bol využitý zdravotne nezávadný bio plast PLA, ktorý je šetrnejší k životnému prostrediu.

1.1 Telo

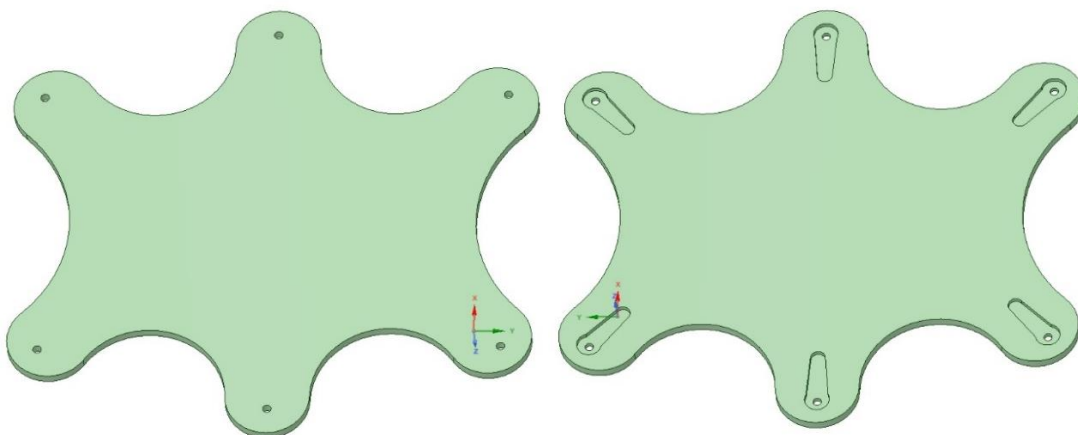
Rozlišujeme dva základné tvary tela:

1. Okrúhly/hexagonálny, pri ktorom sú končatiny umiestnené rovnomerne do kruhu
2. Podlhovastý/obdĺžnikový, pri ktorom sú končatiny symetricky umiestnené po dlhších stranách útvaru

Výzorom sme sa chceli priblížiť k malým článkonohým živočíchom a preto sme zvolili podlhovastý podvozok. Kvôli lepšej stabilite, nohy nie sú umiestnené v jednej rovine, ale stredné končatiny sú vysadené von a okrajové končatiny sú vychýlené do strán, čím získavame tvar nepravidelného šesťuholníka (15). Obrázok 5 b) vyobrazuje zárezy, ktoré slúžia na prichytenie nadstavca servo motora k vrchnej časti tela. Obrázok 5 a) slúži aj ako spodná časť.

Špecifikácia pre telo:

- rozmery: 14.5 x 15.5 x 0.03 cm
- výška vrstvy pri tlači: 0.3 mm
- výplň vnútra pri tlači: 40%
- váha hornej časti: 35 g
- váha dolnej časti: 38 g



Obrázok 5 3D modely zobrazujúce a) vonkajšiu časť tela b) vnútro vrchnej časti tela

1.2 Končatiny

Rozhodujúcejším prvkom sú končatiny, ktoré svojimi vlastnosťami špecifikujú dĺžku kroku, rýchlosť a mobilitu robota. Je dôležité navrhnuť končatinu, tak aby nespôsobovala zbytočné obmedzenia v pohybe. Hexapody majú zväčša 2 - 4 stupne voľnosti, podľa ktorých sa odvíja plynulosť pohybu. My sme sa rozhodli pre 3 stupne voľnosti, ktoré nám umožňujú pohybovať sa všetkými smermi (do strán a šikmo) a sú dostačujúce pre našu aplikáciu.

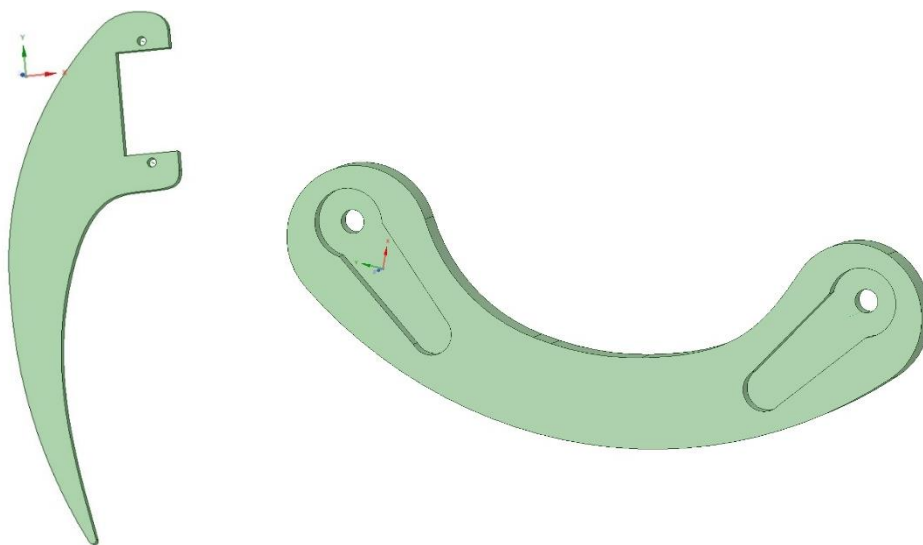
Samotný vzhľad končatín bol takisto inšpirovaný prírodou konkrétne článkonožcami. Končatiny článkonožcov sa primárne skladajú z panvičky (coxa), stehna (femur) a holene (tibia). Panvička je pripevnená k telu pomocou bedrového kĺbu a umožňuje pohyb nohy dopredu a dozadu. Pohyb končatiny hore a dole zabezpečuje stehenný kĺb priamo spojený s bedrovým. Holenný kĺb spája stehennú a holennú kosť a jeho funkciou je vyrovňovanie chôdze. Tak ako pri hmyze aj u nás je všetkých šesť končatín po oboch stranách zrkadlovým obrazom, ktoré sú inak identické.

Špecifikácia pre holennú kosť:

- rozmery: 16 x 3.5 x 0.03 cm
- výška vrstvy pri tlači: 0.2 mm
- výplň vnútra pri tlači: 20%
- váha: 4 g
- množstvo: 6 ks

Špecifikácia pre stehennú kosť:

- rozmery: 7 x 2.5 x 0.03 cm
- výška vrstvy pri tlači: 0.2 mm
- výplň vnútra pri tlači: 20%
- váha: 2 g
- množstvo: 6 ks

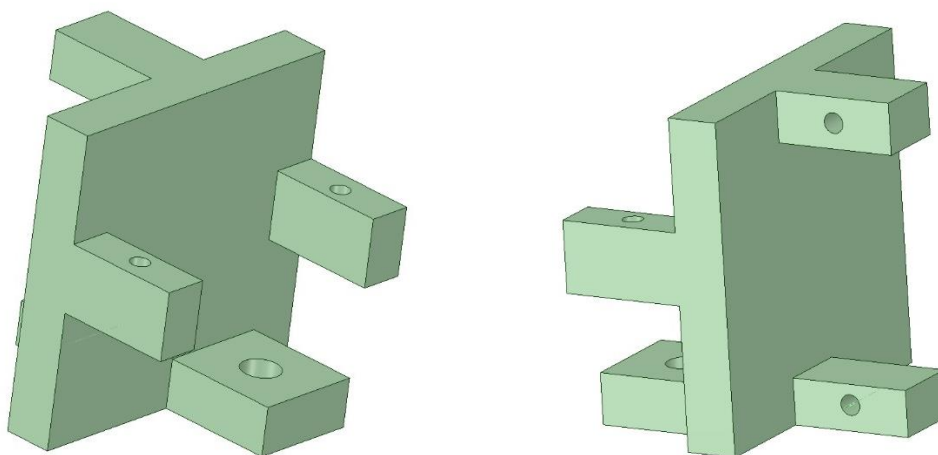


Obrázok 6 3D model zobrazujúci a) holennú kosť b) stehennú kosť

Obrázok 7 vyobrazuje model panvičky/kĺbu, ktorý spája vrchnú časť tela so spodnou. Slúži na prichytenie dvoch servo motorov vo vertikálnom a horizontálnom smere.

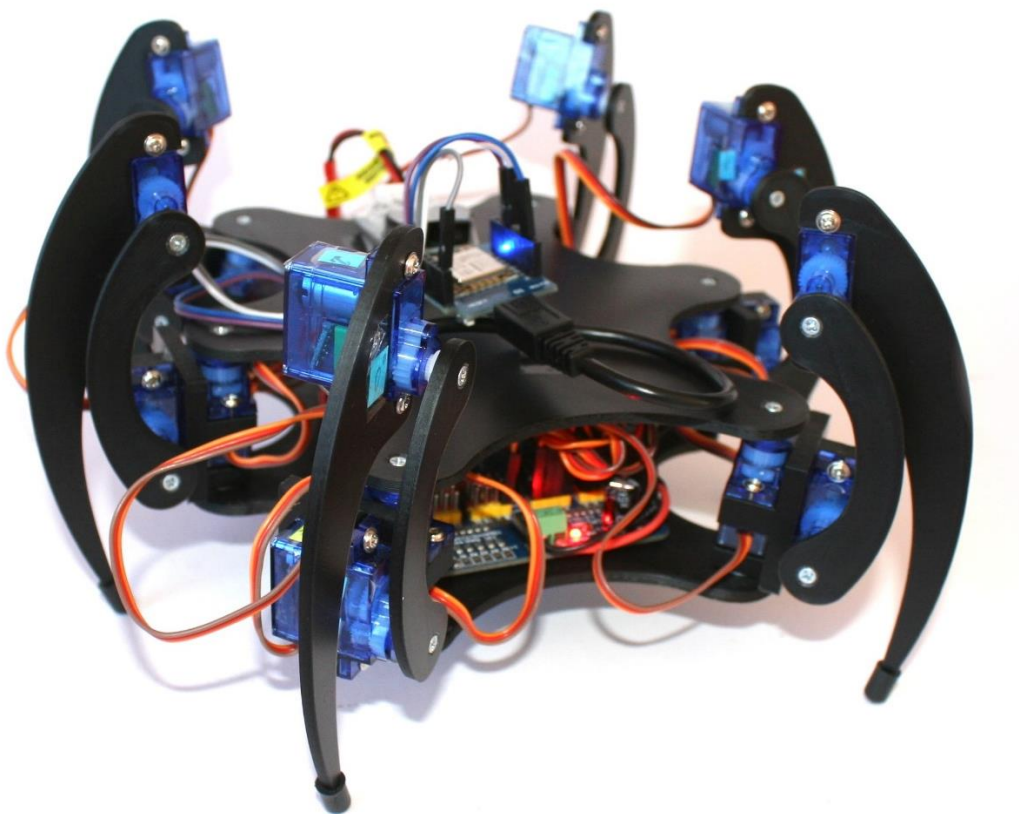
Špecifikácia pre panvičku:

- rozmery: 3.2 x 2.9 x 2.8 cm
- výška vrstvy pri tlači: 0.2 mm
- výplň vnútra pri tlači: 30%
- váha: 6 g
- množstvo: 6 ks (3 ks sú zrkadlovým obrazom)

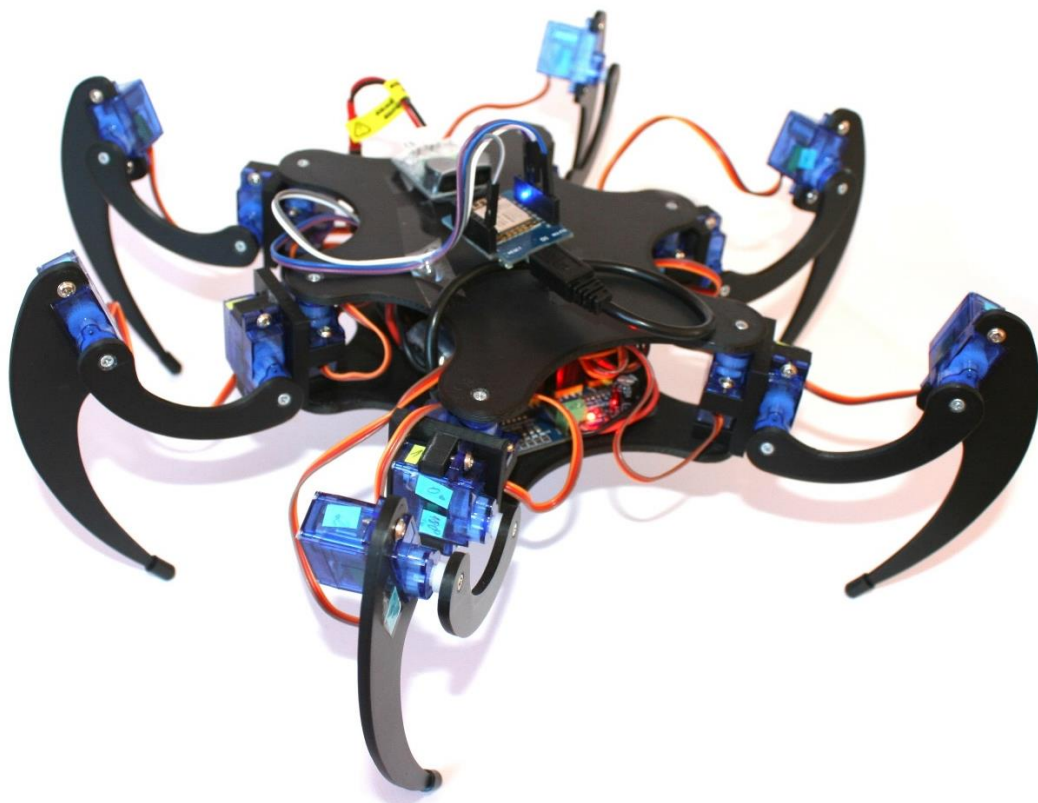


Obrázok 7 Pohľad na 3D model kĺbu z dvoch strán

1.3 Hotová konštrukcia



Obrázok 8 Model zloženého hexapoda v sede



Obrázok 9 Model zloženého robota v základnej stojacej polohe

Hardvér

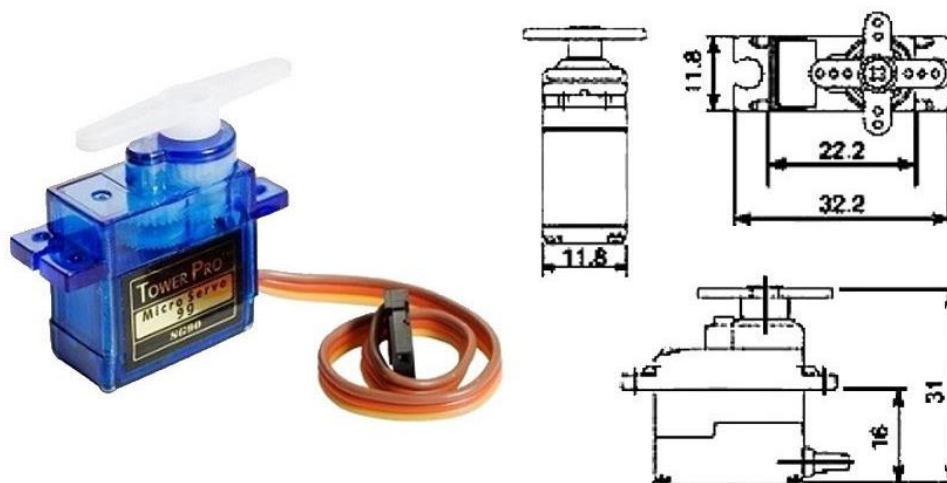
Na základe mechanického návrhu potrebujeme vybrať vhodný hardvér, ktorý by robota uviedol do pohybu. Je vhodné spomenúť, že dôležitým faktorom pri výbere elektroniky bola najmä cena.

1.1 Tower Pro SG90

Kvôli vyššie uvedeným vlastnostiam elektrických aktuátorov sme pre účely našej aplikácie vybrali modelárske servo motory značky Tower Pro SG90 9g. Je to miniatúrne, analógové servo, ktoré vzhľadom na nízku cenu svojimi parametrami milo prekvapuje. Vyrobené je z plastu a preto je náchylné na poškodenia pri veľkej záťaži, dlhodobom alebo nesprávnom používaní, ako je napríklad násilná zmena poloha serva. Ďalšou nevýhodou je to, že je analógové a tým pádom obsahuje špeciálny obvod, ktorý podľa šírky signálu a polohy potenciometra roztáča motor. Analógové servá sú pomalšie, menej presné, slabšie, ako digitálne servo motory, ktoré sú na druhú stranu drahšie, majú vyššiu spotrebu a je možné ich programovať. I napriek negatívam sme sa rozhodli zakúpiť toto servo, keďže potrebujeme až 18 kusov a klady prevyšujú zápory. Tower Pro SG90 sa dokáže otáčať približne o 180° a je riadené prostredníctvom PWM signálu. Tešia sa veľkej popularite pri zostrojovaní projektov podobného charakteru. Využíva JR konektor, ktorý pri náhodnom opačnom zapojení nespôsobí skrat. Hnedý vodič je zem, ktorá sa pripája na GND na vývojovej doske, červený vodič je napájanie, ktoré sa pripája na 5 V výstup a oranžový vodič sa používa na ovládanie, čiže je ho potrebné pripojiť na PWM výstup.

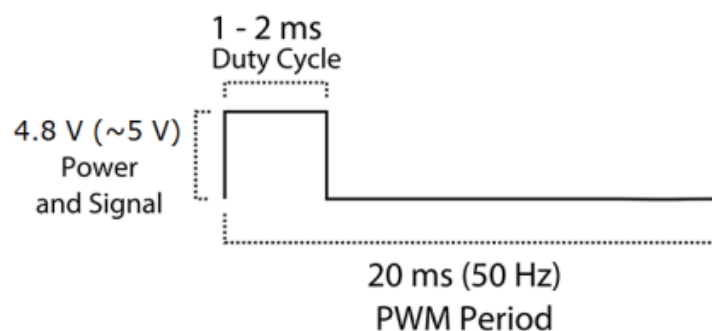
Špecifikácia:

- váha: 9 g
- rozmery: 23 x 12.2 x 29 mm
- krútiaci moment: 1.8 kg/cm (4.8 V)
- typ pohonu: POM
- rýchlosť: 0.1 sekundy/60° (4.8 V)
- pracovné napätie: 4.8 - 6 V
- typ konektora: JR (16)



Obrázok 10 Servo motor Tower Pro SG90 9g (17)

PWM, čo je skratka z anglického pulse-width modulation a má slovenský ekvivalent pulzná šírková modulácia. Je to typ digitálneho signálu, ktorý nám umožňuje meniť šírku impulzu, čiže vieme určiť ako dlho má byť signál v aktívnom a neaktívnom stave. Týmto spôsobom je možné meniť výkon motora alebo jas LED plynulo od 0% až po 100%. Na Obrázok 11 máme možnosť vidieť jednu periódu PWM signálu pre servo motor Tower Pro SG90, ktorá má dĺžku 20 ms a ktorej prevrátená hodnota je 50 Hz. Strieda daného signálu je 1 ms až 2 ms. Ak je strieda 1 ms, tak je motor v pozícií 0°, ak 1.5 ms tak v pozícií 90° a ak má strieda hodnotu 2 ms, tak motor bude v pozícií 180°. (18)



Obrázok 11 Perióda PWM signálu pre servo motor Tower Pro (18)

1.2 PCA9685

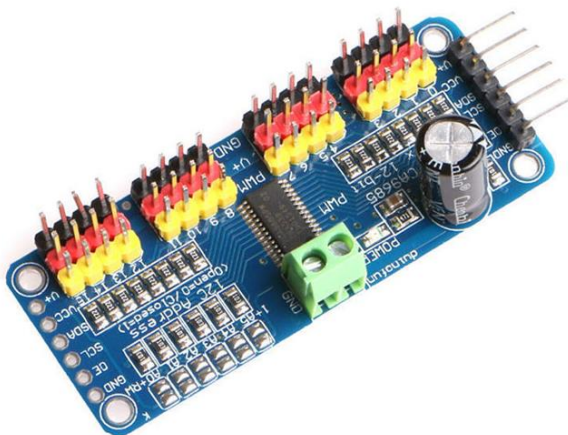
Na ovládanie všetkých 18 servomotorov sme potrebovali mikrokontrolér s dostatočným množstvom výstupov. Keďže sme mali určité požiadavky na mikrokontroléry, ktoré

väčšinou vývojovými doskami neboli spĺňané rozhodli sme sa, že na ovládanie servomotorov použijeme dva 16-kanálové PWM servo drivery PCA9685 s rozhraním I2C. Táto zbernica používa vodiče dvoch typov nazývaných ako Serial Data (SDA) a Serial Clock (SCL). Vďaka nim na ovládanie všetkých motorov potrebujeme iba 2 linky SCL a SDA prepojené s vývojovou doskou, čím sme významne ušetrili na výstupoch. Na doske sú piny VCC a GND na napájanie logiky dosky a V+ je voliteľné napájanie servo motorov, ktoré nie je využívané čipom dosky. Na napájanie servo motorov je možné použiť aj zelený terminál na vrchu dosky. Pomocou pinu OE je možné vypnúť a zapnúť všetky výstupy naraz, pričom zo začiatku sú povolené všetky, čiže OE pin je voliteľný. Na doske sa nachádza 12-bitový PWM kontrolér, ktorý dokáže ovládať 16 kanálov naraz rýchlosťou 60 Hz a tým odľahčuje mikroprocesor. Každý PWM výstup pracuje nezávisle, ale musia mať rovnakú pracovnú frekvenciu.

Špecifikácia:

- pracovné napätie: 3.3V – 5V
- výstup: 6V max. 12V
- PWM: 12-bit, 24 Hz – 1526 Hz
- rozsah teplôt: -40°C až +85°C

Aby sme PCA9685 vedeli zreťaziť s ďalšou doskou, je potrebné im priradiť unikátne adresy. Predvolenou adresou je 0x40, ktorú zmeníme premostením korešpondujúcich bitov v pravom hornom rohu. Na prepojenie dosiek použijeme prepojovacie drôtičky, ktoré navzájom pospájujeme s výstupmi GND, SCL, SDA a VCC. (19)



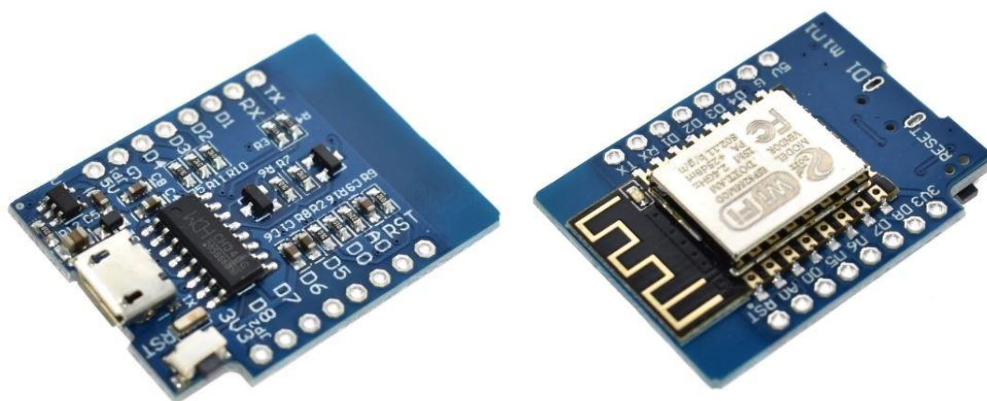
Obrázok 12 Servo driver PCA9685 (20)

1.3 WeMos D1 mini

Na to aby sme boli schopní dať servo motory do pohybu a riadiť ich potrebujeme mikrokontrolér, čo je jedno-čipový počítač slúžiaci na vykonávanie programov. Na trhu je množstvo možností, ktoré generujú PWM signál ako sú napríklad PIC, ARM, Arduino, ale aj mikroprocesor Raspberry Pie. Pre našu jednoduchú aplikáciu je postačujúca open-source platforma Arduino. Pri výbere konkrétneho typu sa musíme zamyslieť nad tým ako robota budeme ovládať na diaľku. Do úvahy spadá buď bluetooth alebo WiFi. Kvôli dôvodom uvedeným v kapitole o komunikačnom protokole sme sa rozhodli pre WiFi. Aj v tejto kategórii máme na výber z viacerých možností ako je NodeMCU ESP8266, Adafruit Feather Huzzah alebo Wemos D1 Mini. Kvôli výhodám ako sú dobre spracovaná dokumentácia, kompatibilita s Arduino a malá veľkosť, ktorá nemá žiaden vplyv na výkonnosť sme vybrali dosku Wemos D1 Mini. Ďalšou nespornou výhodou je regulátor prostredníctvom, ktorého dokážeme dosku napájať aj z 5 V zdroja. (21)

Špecifikácia:

- mikrokontrolér: ESP8266 WiFi Module
- WiFi: 802.11 b/g/n
- rýchlosť procesora: 80/160 MHz
- flash pamäť: 4 MB
- USB – Serial prevodník: CH340G
- 11 pinov s prerušeniami I2C zbernica, PWM, 1-Wire okrem pinu D0
- pracovné napätie: 3.3 V
- váha: 3 g
- rozmery: 34.2 x 25.6 mm
- kompatibilné s Arduino, MicroPython, NodeMCU
- 1 analógový vstup, USB konektor



Obrázok 13 WeMos D1 Mini (22)

1.4 Napájanie

Na napájanie 1 vývojovej dosky, 2 servo dosiek a 18 servo motorov potrebujeme externý zdroj napätia, ktorý by bol dostatočne ľahký a malý na to, aby sa dal umiestniť do modelu a zároveň dostatočne výkonný aby žiadna súčiastka nepociťovala nedostatok energie. Pri vyberaní vhodného zdroja môžeme zvoliť dve stratégie.

1. Buď si zvolíme inú batériu pre každú súčiastku, čo by mohlo byť výhodnejšie riešenie, ak by súčiastky mali rozličné pracovné napätia - obvod by sa jednoduchšie navrhoval a výsledok by mohol byť efektívnejší. Nevýhodou je, že je po vybití je potrebné nabiť viac bateriek a takisto sa môžu vybiť v rôznych časoch.
2. Druhý spôsob je výber jednej batérie pre všetky súčiastky, čím si zaručíme nižšiu hmotnosť a jednoduchšie používanie. V tomto prípade je potrebné viac premyslieť obvod, ktorý sa môže ľahko skomplikovať a na jeho sfunkčnenie bývajú potrebné regulátory napätia.

Všetky naše súčiastky majú pracovné napätie 5 V, okrem mikrokontroléra, ktorý má zabudovaný regulátor napätia z 5 V na 3.3 V. Tým pádom je pre nás výhodnejšie použiť jeden zdroj napájania. Medzi najpopulárnejšie typy externých zdrojov zaradíme NiMh batérie, tužkové batérie a LiPo batérie. Lítiovo-polymérové batérie sú zatiaľ najpokročilejšie zo všetkých uvedených druhov. Najväčšou výhodou LiPo batérií je vysoká hustota uchováanej energie a teda najnižšia hmotnosť a objem. Druhou výhodou je vysoký počet nabíjaciach cyklov, po ktorých kapacita batérie začne klesať na 60 %. Na druhej strane pri ich používaní a nabíjaní musíme byť obozretní, keďže lítium-ión vo vnútri je reaktívny so

vzduchom. Do úvahy neprichádza styk s vysokými teplotami, prebitie alebo podbitie pod 3.3 V, či použitie v projektoch, kde by hrozilo mechanické prepichnutie. Časom sa znižuje jej kapacita i keď nie je používaná. LiPo batérie sú vyrábané v rôznych veľkostiach a rozlišujeme ich na základe počtu buniek, kapacity a vysoký vybíjací prúd.

My sme zvolili akumulátor typu LiPo. Názov batérie prezrádza jej kapacitu v mAh, čo vypovedá o tom koľko prúdu musíme odoberať na to, aby sa akumulátor vybil za jednu hodinu. Označenie batérie napríklad 2S hovorí o tom, že má dva články v sérii spolu s hodnotou nominálneho napätia 7.4 V. Čím je väčšia kapacita, tým je väčšia a ťažšia samotná batéria a takisto by sme si museli dávať väčší pozor na to, aby sme motory prehriatím nezničili. Intenzita vybíjania je daná hodnotou v C. Táto hodnota vyjadruje ako rýchlo môže byť z batérie odoberaný prúd bez rizika jej zničenia. Je vhodné si uvedomiť, že stabilizátor nepustí viac ako 20 A. (23)

Výrobca motorov Tower Pro SG90 9g neuvádza aký je odber prúdu pri a bez záťaže, ale prieskumom sme zistili, že pri maximálnom vypätí odber prúdu môže dosahovať hodnoty až 750 mA – 1 A. Pri bežnom používaní sa hodnoty pohybujú okolo 500 mA. Vývojová doska s čipom ESP8266 má odber prúdu v rozsahu od 100 mA až po 500 mA v závislosti od stavu, v ktorom sa nachádza (inicializácia, prijímanie paketov atď.).

Ako ochranu pred poškodením batérie sme zakúpili tzv. Lipoly Low Voltage Alarm (2S – 4S), ktorého úlohou je upozornenie zvukovým signálom a červenými LED-kami, keď napätie batérie začne klesať pod 3.3 V a hrozí jej podbitie inak svieti na zeleno. Alarm sa pripája na servisný konektor.

Špecifikácia:

- Lipoli rozsah: 2S – 4S
- rozmery: 38 x 25 x 11 mm
- váha: 7 g
- presnosť na: 0.03 V
- hranica prahu alarmu: 3.3 V (pre každý článok) (24)



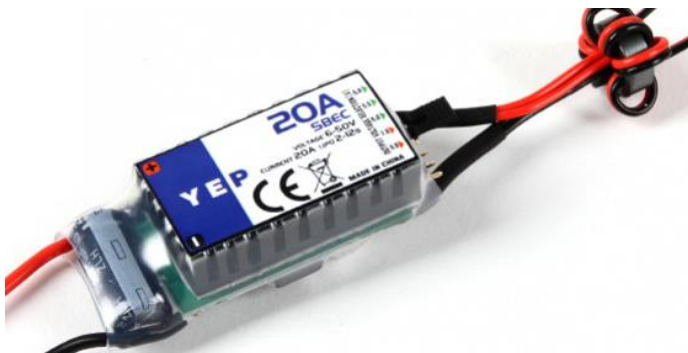
Obrázok 14 Nabitá LiPo batéria s alarmom

1.5 YEP 20A HV (2~12S) SBEC

Na konverziu napätia batérie zo 7.4 V na 5 V použijeme YEP 20A HV (2~12S) SBEC s voliteľným výstupným napätím. Táto súčiastka poskytuje spoľahlivú energiu s širokým rozsahom vstupného napätia od 6 V do 50 V. Pomocou skratovacej prepojky vieme výstupné napätia nastaviť na 5 V, 5.5 V, 6 V, 7 V a 9 V bez potreby programovania. Má v sebe zakomponovaný chladič. Vodičom, ktoré sa pripájajú na batériu sme museli vymeniť konektor za JST konektor.

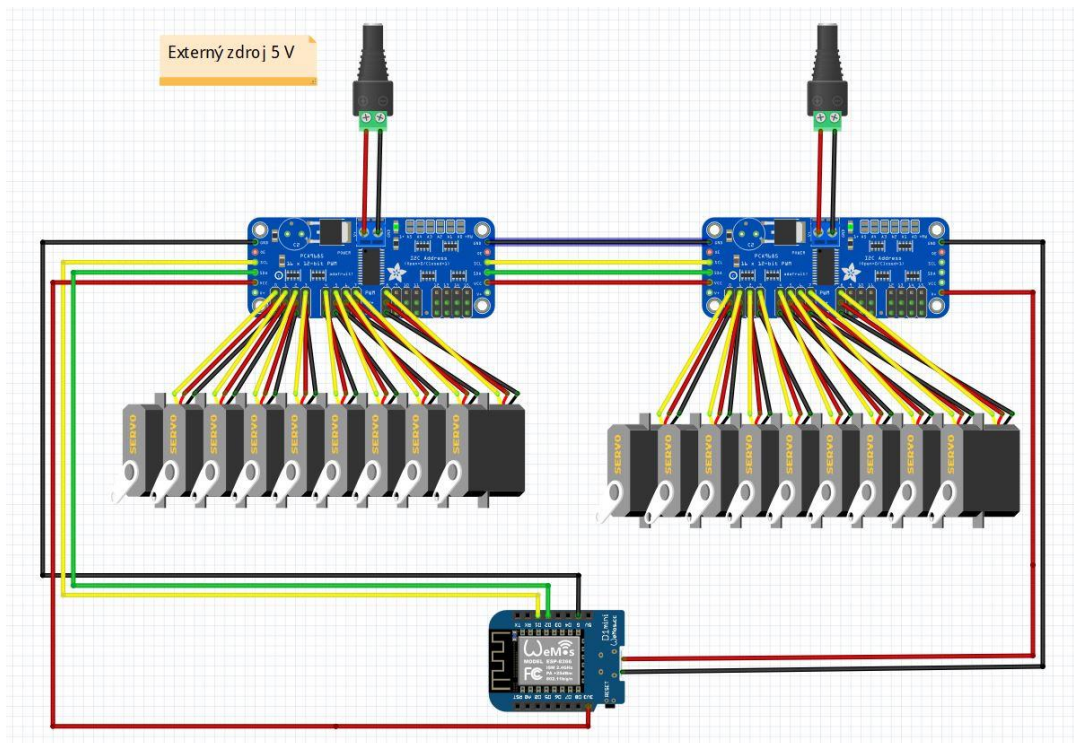
Špecifikácia:

- max. prúd: 20 A
- vstupný prúd: 2-12 článkové batérie LiPo, 6-35 článkové NI-MH/Ni-CD
- rozmery: 57 x 26 mm
- váha: 42 g (25)

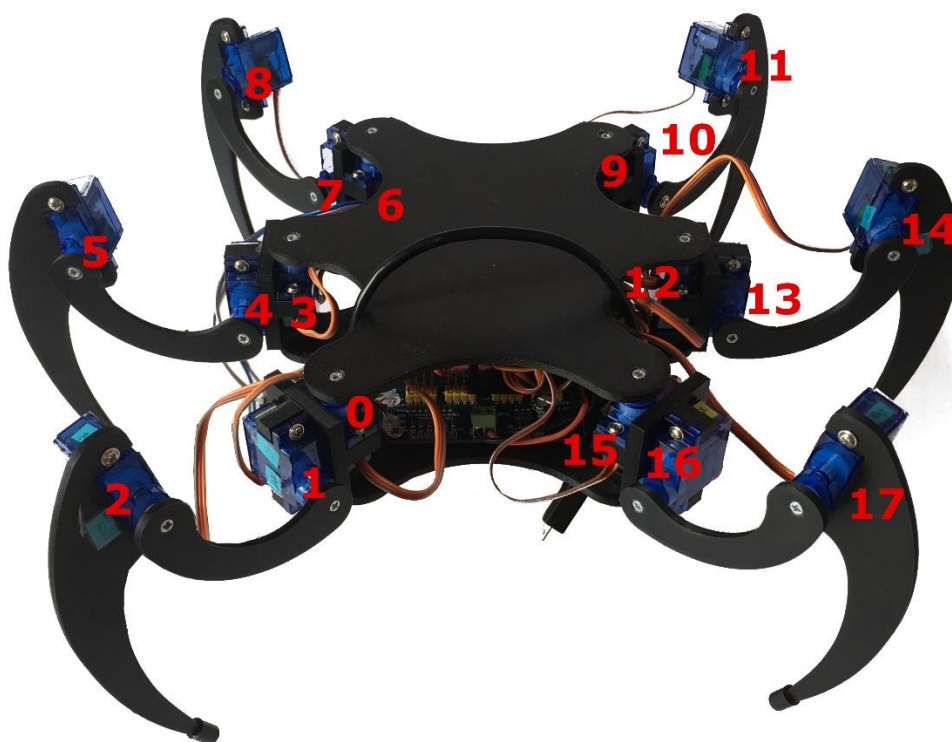


Obrázok 15 YEP 20A SBEC (25)

1.6 Zapojenie



Obrázok 16 Schéma zapojenia



Obrázok 17 Rozmiestnenie servo motorov v modeli

Komunikačný protokol

Ďalšou neoddeliteľnou súčasťou plne funkčného hexapoda je spôsob akým bude prijímať a spracovávať príkazy. Najčastejšie používanými bezdrôtovými technológiami sú Bluetooth a WiFi.

Bluetooth je spôsob akým zariadenia medzi sebou bezdrôtovo komunikujú na krátku vzdialenosť. Spočiatku mala táto technológia nahradiť len káble, ale nízka cena a nízka spotreba energie sa ukázali ako atraktívne riešenie pre mobilné zariadenia. Pracuje na báze paketov, frekvencii okolo 2.4 MHz, a kladom je, že pri prenose paketov nedochádza k ich excesívnej strate a chybám tak ako pri WiFi. Nevýhodou je relatívne krátka vzdialenosť približne 10 metrov, ktorá musí byť medzi dvoma zariadeniami.

WiFi pracuje podobne ako Bluetooth a dokáže prepojiť viac zariadení v sieti. Zariadenia je možné pripojiť na internet a komunikovať s nimi kdekoľvek vo svete. Poskytuje omnoho väčší rozsah a takisto konzumuje viac energie. My sme sa rozhodli využiť technológiu WiFi od čoho sa odvíjal aj výber hardvéru. Štandardnými protokolmi, ktoré sú postavené na IP sú TCP a UDP, ktoré sa nachádzajú v transportnej vrstve. IP protokol definuje ako adresovať a presmerovať pakety, tak aby dorazili do správnej destinácie. Sieťové priechody (po ang. gateway) sa starajú o to kam ich presmerovať.

1.1 TCP

TCP/IP alebo inak nazývaný ako Transmission Control Protocol je súbor komunikačných protokolov, ktoré slúžia na vzájomné prepojenie sieťových zariadení na internete. Používa sa aj na komunikáciu v súkromných sieťach. TCP/IP poskytuje spoľahlivú komunikáciu so schopnosťou sa obnoviť a preposlať stratené pakety. Je to obojsmerná komunikácia, čo znamená, že keď paket dorazí do destinácie, tak sú na pôvodnú adresu odoslané pakety, že boli prijaté. Skladá sa z dvoch protokolov - pre vytvorenie spojenia na úrovni TCP protokolu je potrebné mať k dispozícii možnosť nadviazania spojenia na IP úrovni. TCP špecifikuje ako zariadenia vytvárajú komunikačné kanály v sieti, spravuje ako sa správy rozdelia do menších paketov pred odoslaním a v akom poradí budú spracované. TCP/IP využíva model typu klient/server. Medzi kľúčové vlastnosti zabezpečujúce spoľahlivosť a robustnosť komunikácie patria:

- očíslovanie segmentov pre určenie poradia a detekciu duplicitných dát
- kontrolné súčty pre detekciu chýb v segmentoch
- potvrdzovanie a časovače pre detekciu a prispôsobenie sa strate alebo oneskoreniu dát (26)

1.2 UDP

UDP/IP alebo inak nazývaný ako User Datagram Protocol je jednoduchý protokol bez trvalého pripojenia používaný na prenos datagramov na danú vzdialenú IP adresu a port. Komunikácia je na báze datagramov, ktoré sú vysielané bez spätného overenia, či dorazili do cieľa. TCP poskytuje bohatšiu funkcionality, ale na druhej strane je potrebné aby vzdialené zariadenie bolo pripojené k sieťovému socketu, čo UDP nevyžaduje. Tento protokol obohacuje IP o ďalšie dve funkcionality – čísla portov a kontrolný súčet. UDP má mnoho vlastností medzi, ktoré zaradíme:

- nespoľahlivosť – po odoslaní správy nevieme zistiť, či naozaj dorazila do svojej destinácie
- nepripojenie – ak aplikácia otvorí port, tak na danú adresu môžu byť poslané pakety bez predchádzajúcej dohody či upozornenia rôznymi zariadeniami
- nenáročnosť – neobsahuje pozorovanie pripojenia či kontrolné správy
- neusporiadanosť – po odoslaní viacerých správ príjemcovi nevieme predpovedať v akom poradí prídu
- rýchlosť – nie je nutné čakať na potvrdzovaciu správu

UDP sa využíva na živý prenos videí, hovorov, online hier či iné real-time aplikácie, pri ktorých rýchlosť je nutnosťou, strata paketov nemá výrazne nežiaduce následky a implementácia vlastností spoľahlivej komunikácie by mohla mať negatívny vplyv. (27)

1.3 Implementácia

Robot bude ovládaný prostredníctvom mobilnej aplikácie pre Android. Na komunikáciu telefónu s mikrokontrolérom použijeme UDP.

1.3.1 Arduino

V prostredí Arduino IDE bolo potrebné doinštalovať knižnicu esp8266 od ESP8266 Community pre prácu s vývojovou doskou Wemos D1 Mini. Robot pri svojom fungovaní nebude potrebovať internetové pripojenie, čo znamená, že ho budeme môcť ovládať aj vo vonkajšom prostredí a preto vývojovú dosku nakonfigurujeme ako prístupový bod, na ktorý sa bude môcť telefón pripojiť pomocou metódy `createAccessPoint()`. Zvolíme sieťový priechod, masku siete a statickú IP adresu. (28)

Ak je pripojenie zriadené, ďalším krokom je špecifikovanie portu na prijímanie a odosielanie datagramov v metóde `connectUDP()`. Pre našu aplikáciu je potrebné zvoliť port z kategórie registrovaných portov v rozmedzí od 1024 – 49151. My sme zvolili port 8889. K prijímaniu datagramov dochádza vo funkcii `loop()`. Na znak prijatia príkazu z aplikácie odošleme prečítaný reťazec naspäť. (29)

1.3.2 Android

Do aplikácie je potrebné zadať IP adresu a port robota. Táto časť sa zaoberá spôsobom akým aplikácia komunikuje s ESP8266. Aplikácia je bližšie analyzovaná v kapitole 6. Na druhej strane komunikácie je mobilný telefón pripojený k prístupovému bodu. Na prijímanie a odosielanie datagramov sa používa trieda `UdpThread`, konkrétne funkcia `run()`, ktorá odošle reťazec prekonvertovaný na byty do zadanej destinácie. Následne čaká na prijatie pôvodnej správy a zistenie, či poslanie príkazu prebehlo úspešne. Premennou `timeout` bol nastavený čas v sekundách potrebný na prijatie odpovede. Ak odpoveď nepríde po tomto čase, tak je poslaná chybová hláška späť do triedy `MainActivity`. Inak je poslaná hláška hlásajúca úspech. (30)

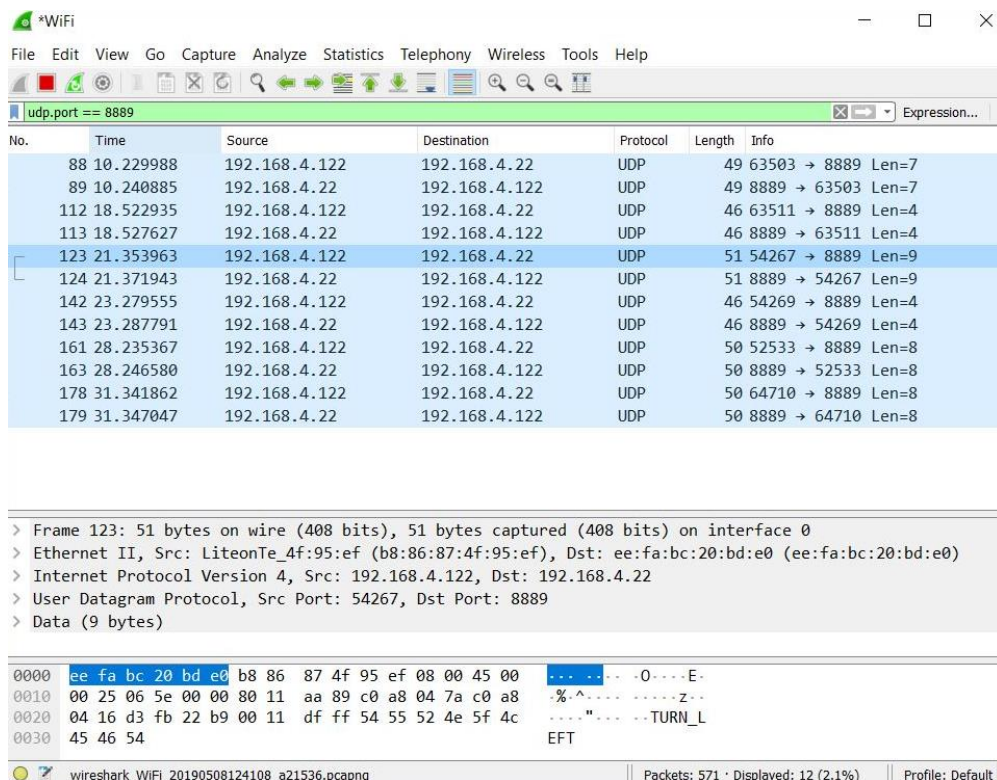
1.3.3 Datagramy

Datagramy, ktoré sa preposielajú medzi zariadeniami sú reťazce - príkazy na základe, ktorých je robot ovládaný. V tabuľke č. 1 sú uvedené jednotlivé reťazce spolu s ich funkciami. Každý reťazec okrem reťazca `STAND_UP` najprv skontroluje, či je robot v základnej stojacej polohe a ak nie je, tak je robot do tejto polohy nastavený ešte predtým ako sa nastaví do novej polohy.

Tabuľka 1 Význam posielaných datagramov

Ret'azec	Funkcia
STAND_UP	robot sa postaví do základnej stojacej polohy
SIT_DOWN	robot sa posadí zo základnej polohy
FORWARD	robot začne kráčať dopredu
BACKWARD	robot začne kráčať dozadu
LEFT	robot začne kráčať do strany doľava
RIGHT	robot začne kráčať do strany doprava
TURN_LEFT	robot sa začne otáčať na mieste doľava
TURN_RIGHT	robot sa začne otáčať na mieste doprava
STOP	robot ukončí pohyb a vráti sa do základnej polohy

Na zachytávanie datagramov medzi koncovými bodmi sme používali softvér Wireshark, ktorý slúži na analýzu sieťových protokolov. Obrázok 18 vyobrazuje komunikáciu medzi mikrokontrolérom s IP adresou 192.168.4.22 a počítačom s IP adresou 192.168.4.122 pripojeným k jeho prístupovému bodu na porte 8889. Počítač pošle UDP datagram, ktorý je následne robotom preposlaný naspäť.



Obrázok 18 Wireshark – komunikácia PC pripojeného k prístupovému bodu hexapoda

Softvér hexapoda

1.1 Inverzná kinematika

Priama kinematika je časť mechaniky opisujúca pohyb telesa či bodu. Daný pohyb je analyzovaný pomocou matematických výrazov a modelov. Ak poznáme potrebné dĺžky a uhly jednotlivých servomotorov dokážeme z nich deterministicky vypočítať polohu končatiny v karteziánskej sústave súradníc (x , y , z).

Inverzná kinematika, ako názov napovedá, je opakom priamej kinematiky. Zo známych súradníc, do ktorých chceme končatinu dostať musíme vypočítať uhly kĺbov. Jej význam vo sfére robotiky je dôležitejší, lebo zvyčajne poznáme práve koncové body a parametre pre servomotory, ktoré nám umožnia tento stav dosiahnuť sú neznáme. My budeme používať geometrický prístup. Nasledujúce rovnice zachytávajú pomocné výpočty a premenné:

$$xFemTib = x - coxaL \quad (1)$$

$$zBodyGround = z - zFemBody \quad (2)$$

$$HF = \sqrt{xFemTib^2 + z^2} \quad (3)$$

$$A_1 = \tan^{-1} \frac{xFemTib}{z} \quad (4)$$

$$A_2 = \cos^{-1} \frac{femurL^2 + HF^2 - tibiaL^2}{2HFfemurL} \quad (5)$$

$$B_1 = \cos^{-1} \frac{femurL^2 + tibiaL^2 - HF^2}{2tibiaLfemurL} \quad (6)$$

kde $xFemTib$ je vzdialenosť medzi koncom nohy na zemi a stehenným kĺbom, x je vzdialenosť stredu otáčania kĺbu panvičky od konca nohy položeného na zem a $coxaL$ je vzdialenosť kĺbu panvičky od stehenného kĺbu. $zBodyGround$ je vzdialenosť fyzického spodku tela od zeme, pričom $xFemBody$ je vzdialenosť stredu otáčania stehenného kĺbu k spodku tela a z je vzdialenosť stredu otáčania stehenného kĺbu až po zem. HF je vzdialenosť stredu otáčania stehenného kĺbu ku koncu nohy. Ďalšími premennými, ktoré sa vyskytujú v rovniciach (1), (5) a (6) sú $coxaL$, čo je vzdialenosť medzi stredom otáčania panvičky a stehenného kĺbu, $femurL$, čo je vzdialenosť medzi stredom otáčania stehenného a holenného kĺbu a $tibiaL$, čo je vzdialenosť medzi stredom otáčania holenného kĺbu

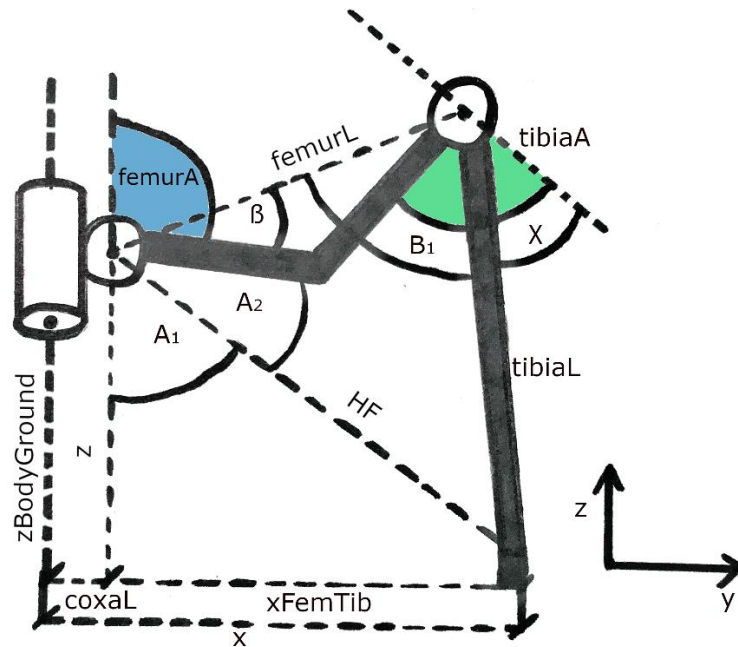
a koncom nohy. Tieto hodnoty sú definované 3D modelom. Obrázok 17 vyobrazuje uhly A_1 , A_2 a B_1 .

Obrázok 19 zachytáva modrou farbou zvýraznený uhol $femurA$ a zelenou farbou zvýraznený uhol $tibiaA$, ktoré sú definované nasledovne:

$$femurA = 180^\circ - A_1 - A_2 + \beta - \chi \quad (7)$$

$$tibiaA = B_1 - \beta + \chi \quad (8)$$

kde uhol β vyjadruje odchýlku umiestnenia násadky servo motora od $femurL$ medzi stehenným a holenným kĺbom. Uhol χ je zvieraný $tibiaL$ a priamkou vyjadrujúcej 0° pre holenný kĺb.



Obrázok 19 Pohľad na nohu zboku (31)

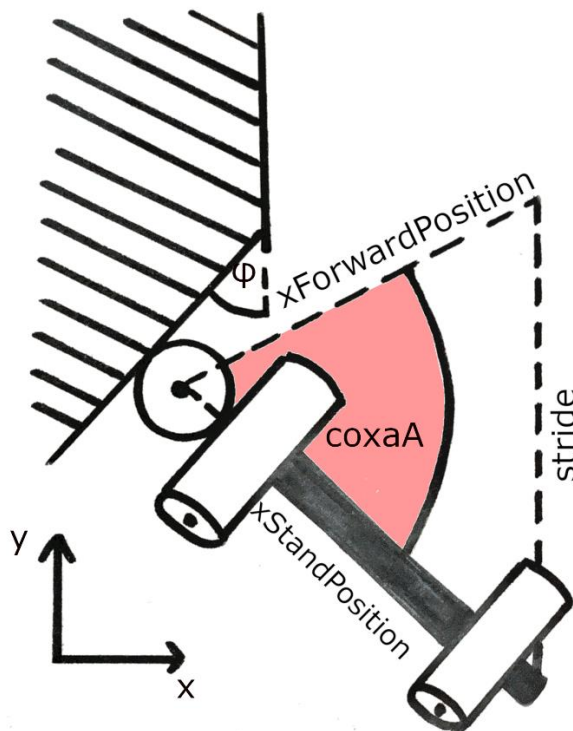
Ružovou farbou zvýraznený $coxaA$ je uhol, ktorý zvierajú priamka, kedy je noha kolmo na rovinu tela s priamkou, v ktorej je noha po vykonaní kroku a je definovaný ako:

$$coxaA = \sin^{-1} \left(\frac{y \sin \left(\tan^{-1} \left(\frac{xStandPosition \sin(90^\circ - \varphi)}{xStandPosition \sin(\varphi) + y} \right) \right)}{xStandPosition} \right) \quad (9)$$

kde $xStandPosition$ vyjadruje vzdialenosť konca nohy od servo motora znázorňujúci panvičku, keď je noha v základnej pozícii a s telom zvierá pravý uhol, φ je uhol, ktorý

zvierajú roviny, v ktorých sú umiestnené nohy a má hodnotu 42.6° , y je dĺžka kroku, ktorá býva definovaná aj ako *stride*.

Obrázok 20 znázorňuje pohľad na nohu zhora vykonávajúcu krok dozadu. Vyšrafovaná časť predstavuje telo. Hodnota *xForwardPosition* v tomto prípade vyjadruje dĺžku nohy pre krok dozadu.



Obrázok 20 Pohľad na nohu zhora (31)

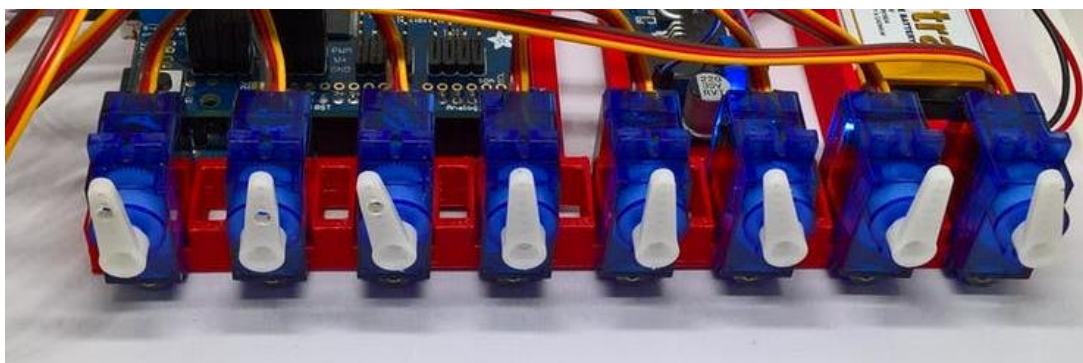
1.2 Implementácia

Pri programovaní mikropočítača ESP8266 sme používali vývojové prostredie pre Arduino. Programovacím jazykom je C/C++, pričom poskytuje knižnice pre rôzne funkcionality. Programy sa zapisujú do tzv. skečov (po anglicky sketch) a každý program musí obsahovať dve funkcie:

- `setup()` je funkcia, ktorá sa spúšťa prvotne len raz po tom, ako je doska napojená na zdroj alebo po reštarte. V tejto funkcii sa inicializujú premenné či mapujú piny.
- `loop()` je funkcia, ktorá sa spúšťa po `setup()`, pričom je vykonávaná opakovane, až do reštartu alebo odpojenia napájania. V tejto funkcii sa implementuje program na ovládanie dosky.

1.2.1 Kalibrácia servo motorov

Kvôli práci s PCA9685 sme do vývojového prostredia museli dodatočne nainštalovať knižnicu Adafruit_PWMServoDriver.h. Pri počiatočnom programovaní servo motorov sme si všimli, že každé z nich reaguje iným spôsobom na ten istý PWM signál a keďže ide o analógový obvod, tak sa medzi servami i rovnakého modelu môžu vyskytovať rozdiely. Za účelom presnejšieho pohybu sme servá nakalibrovali jednoduchým programom. Program spočíval v nájdení minimálnej a maximálnej možnej polohy, do ktorej sa servo motor dokázal otočiť bez bzučania, ktoré by indikovalo počiatok jeho poškodenia. Následne sme k týmto nameraným hodnotám na základe polohy ramena adekvátne priradili 0° , 90° a 180° a zohľadnili ich v programe. (32)



Obrázok 21 Analógové servá nastavené na rovnakú hodnotu PWM signálu (33)

1.2.2 Trieda Joint

Trieda Joint reprezentuje kĺb nohy hexapoda, teda jeden servo motor. Po zavolaní inicializačnej funkcie `setJoint()` získame inštanciu triedy obsahujúcu jedinečné id kĺbu, pin, a PWM dosku, ku ktorým je motor napojený a hodnoty minimálneho, stredného a maximálneho PWM signálu prislúchajúce 0° , 90° a najväčšej hodnote, do ktorej sa motor dostane bez poškodenia. Trieda Joint má nasledovné metódy:

- `setJoint()` – podľa zadaných hodnôt nastaví id, pin, pwm, minPwm, middlePwm, maxPwm
- `setJointPwm()` – podľa zadaného uhla a typu otočenia ($0^\circ/180^\circ$) prekonvertuje uhol v stupňoch na PWM hodnotu a zapíše ju do motora

- degreesToPulse() – prekonvertuje zadaný uhol v stupňoch na PWM hodnotu
- pulseToDdegrees() – prekonvertuje zadanú PWM hodnotu na uhol v stupňoch
- printjoint() – do serial monitoru je vypísaný reťazec obsahujúci id, pin, minPwm, middlePwm, maxPwm

1.2.3 Trieda Leg

Trieda Leg reprezentuje nohu hexapoda. Po zavolaní inicializačnej funkcie setLeg() získame inštanciu triedy obsahujúcu jedinečné id nohy, pole objektov triedy Joint, premennú reverseMovement hovoriacu o tom, či noha patrí do pravej alebo ľavej strany tela, hodnoty uhlov kĺbov, aktuálnu polohu konca nohy v karteziánskej sústave, uhol, ktorý zvierajú roviny strednej a krajnej nohy a inkrementy potrebné na zmenu jednej pozície do druhej. Trieda Leg poskytuje nasledovné metódy:

- setLeg() – podľa zadaných hodnôt nastaví atribúty id, reverseMovement, fi, a atribútom xCurrent, yCurrent, zCurrent, xInc, yInc, zInc nastaví hodnoty tak robot uloží nohy do pozície sedenia
- isLegInBasicPosition() – skontroluje či je noha v základnej stojacej polohe definovanej premennými xStandPosition, yStandPosition, zStandPosition
- getAngles() – prostredníctvom rovníc inverznej kinematiky vypočíta uhly femurAngle, tibiaAngle, coxaAngle podľa rovníc (6), (7), (8)
- setAngles() – jednotlivým kĺbom nastaví adekvátne hodnoty uhlov
- setNewPosition() – slúži na nastavenie novej pozície s bráním ohľadu na aktuálnu pozíciu nohy
- gettre a settre pre xInc, yInc, zInc
- gettre a settre pre coxaAngle, femurAngle, tibiaAngle

1.2.4 Globálne metódy

Medzi ďalšie definované metódy, ktoré nepatria do žiadnej triedy zaradíme:

- setNewPositions() – vytvorí cyklus, ktorý bude nastavovať inkrementy nohám podľa zadaných parametrov
- getIncrement() – podľa zadanej počiatočnej a konečnej hodnoty vypočíta inkrement

- `setXIncrements()` (adekvátne y alebo z) – vypočíta a nastaví xInc všetkým nohám na základe zadaných parametrov
- `resetIncrements()` – vynuluje inkrementy všetkým nohám
- `areLegsInBasicPositions()` – skontroluje, či je každá noha v základnej polohe
- `initializeLegs()` – naplní inštalácie tried `Joint` a `Leg`, čím robota nastaví do sedu

Po pripojení mikrokontroléra k zdroju napätia sa spustí funkcia `setup()`, ktorá vytvorí prístupový bod, špecifikuje port, nastaví komunikáciu so servo drivermi na frekvenciu 60 Hz a inicializuje nohy, ktoré sú definované globálnym 6-prvkovým poľom objektov typu `Leg`. Vo funkcii `loop()` dochádza k prijímaniu a odosielaniu datagramov prostredníctvom `handleCommunication()` a prijatá správa je spracovaná v metóde `handleCommand()`. Zo začiatku je robot v sede. Polohu nohy v sede definujú premenné `xSitPosition`, `ySitPosition` a `zSitPosition`. Pomocou globálnych boolean atribútov `standing`, `walkingForward`, `walkingBackward`, `walkingLeft` a `walkingRight` určujeme, v ktorom stave sa robot práve nachádza. Pred spustením akéhokoľvek pohybu (okrem postavenia sa) je najskôr skontrolované, či robot stojí v základnej polohe definovanej premennými `xStandPosition`, `yStandPosition`, `zStandPosition`. Ak v nej nie je alebo bol pohyb ukončený, tak je do nej nastavený. Zmena pozície do inej prebieha na základe inkrementov – malých krokov v mm, ktoré sú vypočítané na základe počiatočnej, koncovkej polohy a globálnych atribútov `interval` a `updateTime` vypovedajúcich o rýchlosti pohybu. Kladné alebo záporné inkrementy sa pričítavajú k pôvodným pozíciám, pričom počet cyklov závisí od `interval` a `updateTime`. Hodnoty polohy nohy sa zapisujú v cm.

1.3 Chôdza

Zvieratá a ľudia žijúci na súši sa pomocou svojich končatín premiestňujú na iné miesta podľa vzoru, ktorý nazývame chôdza. Pre kráčajúceho robota s k končatinami je množstvo možných udalostí koordinovaných medzi nohami definované ako $N = (2k-1)!$. Pre dvojnohého robota je $N = (2*2-1)! = 6$. Toto číslo opisuje nasledujúce spôsoby manipulácie s nohami: zdvihnutie a polozenie pravej nohy, zdvihnutie a polozenie ľavej nohy a súčasne zdvihnutie a polozenie oboch nôh naraz. Pre hexapoda so šiestimi nohami existuje $N = (2*6 - 1)! = 39\,916\,800$ rôznych akcií, avšak len fragment z nich je možné využiť ako vhodné spôsoby pohybu. Týmto sme chceli poukázať na komplexnosť riadenia pohybu hexapoda

(2). Rôzne spôsoby chôdze sme si v našom modeli vyskúšali pomocou vopred zadaných sekvencií. Tieto sekvencie na základe počtu prvkov poľa a prvkov v poli definujú, ktoré nohy budú vo vzduchu.

1.3.1 Jednonohá chôdza 5+1

Pri pohybe je telo robota podopierané piatimi nohami, pričom jedna noha je presunutá dopredu. Kvôli svojej náture je tento spôsob chôdze najpomalší, ale zároveň aj najrobustnejší, lebo poskytuje najstabilnejší pohyb, pričom ak dôjde k poškodeniu niektorej z nôh, mobilitu robota to neovplyvní. Na rovnakom princípe je založené aj tzv. vlnenie (ripple gait). V programe je definovaná ako `sequence[] = {0, 3, 2, 5, 1, 4}`, čo znamená, že nohy sa budú zdvíhať v poradí 0, 3, 2, 5, 1, 4.

1.3.2 Dvojnohá chôdza 4+2

Dvojnohá chôdza je definovaná štyrmi končatinami položenými na podložke zatiaľ čo sa ostatné dve presúvajú na nové pozície. Je založená na pohybe hmyzu. Nohy sú rozdelené do troch skupín – do párov, ktoré sú si zrkadlovým obrazom po stranách a presúvajú sa dopredu. Keďže telo je podopierané štyrmi nohami konštrukcia zostane stabilná aj keď sa jedna vyradí. Podmienkou je, že to nemôže byť noha v strede. V programe je definovaná ako `sequence[] = {0, 2, 1}`, čo znamená, že nohy sa budú zdvíhať po dvojiciach 0, 3; 2, 5 a 1, 4.

1.3.3 Trojnohá chôdza 3+3

Posledným typom chôdze je trojnohá chôdza, pri ktorej tri nohy zostávajú na zemi a ostatné tri sa zdvíhajú a presúvajú sa dopredu na nové pozície. Tento variant je najrýchlejší a najefektívnejší kvôli tomu, že jeden cyklus pohybu sa skladá z dvoch častí. Na druhej strane je jeho stabilita diskutabilná. Kebyže dôjde k poškodeniu končatiny, tak robot prestáva byť stabilný a v prípade pomalého kroku hrozí pád. V programe je definovaná ako `sequence[] = {0, 1}`, čo znamená, že nohy sa budú zdvíhať po trojiciach 0, 2, 4 a 1, 3, 5.

1.3.4 Postavenie sa a sadnutie

Prvotná poloha robota je sed, do ktorej ju nastaví inicializačná funkcia. Metóda `standUp()` robota postaví do tzv. základnej stojacej polohy, ktorá je východiskovou polohou pre ostatné pohyby. Funkcia `sitDown()` robota posadí zo základnej stojacej polohy do sedu.

Noha v sede je definovaná globálnymi premennými xSitPosition, ySitPosition, zSitPosition. Noha v stoji je definovaná globálnymi premennými xStandPosition, yStandPosition, zStandPosition.

```
1. function standUp(){
2.     zvysenie vysky tela zo sedu na statie
3.
4.     for postupne v zavislosti od sekvencie pre vsetky nohy
5.         zdvihnutie noh podľa sekvencie z pociatocnej pozicie
6.         polozenie noh podľa sekvencie do konecnej pozicie
7. }
```

Kód 1 Zjednodušený pseudokód pre postavenie hexapoda

1.3.1 Rotácia na mieste

Rotácia na mieste nevyužíva rovnice inverznej kinematiky a namiesto toho sme si vyskúšali nastavovanie motorov priamo. Pri tomto pohybe používame len kĺby panvičky a stehennej kosti. Ak je do metódy rotate() zadaná +1 bude sa otáčať doprava, ak -1 doľava, pričom pravá a ľavá strana musia mať nastavené hodnoty opačných polarít.

```
1. function rotate(typ smeru){
2.     for postupne pre vsetky nohy v zavislosti od sekvencie
3.         zdvihnutie noh do polovice trajektorie
4.         polozenie noh na koniec trajektorie
5.
6.     for vsetky nohy
7.         otocenie tela
8. }
```

Kód 2 Zjednodušený pseudokód rotácie hexapoda na mieste

1.3.2 Chôdza rovno

Pred tým ako robot začne kráčať dopredu, či dozadu je potrebné ho nastaviť zo základnej polohy do tzv. prípravnej polohy pomocou funkcie walkPrepare(), kedy telo robota je vystrčené dozadu a nohy zostávajú vpredú na základe poslaného argumentu (+1 vyjadruje vystrčenie tela von, -1 základnú polohu). Následne robot môže začať chodiť dopredu, pričom každý krok je vyvolaný funkciou walk(). Chôdza dozadu funguje tým istým spôsobom, jediný rozdiel je v tom, že hodnoty na y osi musia byť záporné t.j. atribúty stride

a yFowardPosition. Pri ukončení chôdze je potrebné telo presunúť späť do základnej polohy. Chôdza je definovaná globálnymi premennými yFowardPosition, zFowardPosition.

```
1. function walkPrepare(typ pohybu){
2.   for všetky nohy
3.     prenesenie tela zo zakladu dozadu/zozadu do zakladu
4. }
5.
6.
7. function walk(){
8.   for všetky nohy
9.     prenesenie tela zozadu do základnej polohy
10.
11.   for všetky nohy
12.     prenesenie tela zo zakladu dopredu
13.
14.   for postupne v závislosti od sekvencie pre všetky nohy
15.     zdvihnutie noh podľa sekvencie zozadu do polovice kroku
16.     polozenie noh podľa sekvencie dopredu
17. }
```

Kód 3 Zjednodušený pseudokód pre prípravnú funkciu a chôdzu

1.3.3 Chôdza do strán

Chôdza do strán funguje na rovnakom princípe ako chôdza dopredu a dozadu. Robot sa zo základnej polohy musí dostať do prípravnej polohy pomocou walkSidewayPrepare, kedy krajné nohy sú uložené rovnobežne so strednou nohou 1 a 5. Do tejto funkcie je potrebné zadať hodnoty xLeft, xRight a prepare. Hodnoty xLeft a xRight zadané v cm definujú dĺžku kroku. Menšia hodnota znamená, že robot sa bude hýbať daným smerom. Atribút prepare môže mať dve hodnoty: +1 vyjadruje, že nohy treba uložiť kolmo, -1 vyjadruje, že nohy treba uložiť do základnej polohy. Kráčanie je vyvolané volaním metódy walkSideway().

```

1. function walkSidewayPrepare(vzdialenost x1 od tela, vzdialenost x2 od
   tela, typ pohybu){
2.     vypocitanie vzdialenosti y1 nohy od tela na osi y pre nohy 0 a 3
3.     vypocitanie vzdialenosti y2 nohy od tela na osi y pre nohy 2 a 5
4.     vytvorenie pola, z ktoreho sa budu cerpat y hodnoty
5.
6.     for postupne pre všetky nohy v závislosti od sekvencie
7.         zdvihnutie noh zo základu
8.         uloženie noh rovnobežne so strednou nohou buď na vzdialenosť x1
           alebo x2 podľa typu pohybu
9. }
10.
11. function walkSideway(typ smeru){
12.     for postupne pre všetky nohy v závislosti od sekvencie
13.         zdvihnutie noh do polovice trajektórie
14.         polozenie noh na koniec trajektórie
15.
16.     for všetky nohy
17.         presunutie vahy na druhú stranu
18. }

```

Kód 4 Zjednodušený pseudokód pre prípravnú funkciu a chôdzu do strán

Softvér na diaľkové ovládanie

Aplikáciu pre mobilný telefón sme vyvíjali v prostredí Android Studio. Pri nastavovaní projektu sme si zvolili prázdnu aktivitu (ang. Empty Activity), programovací jazyk java a API 16: Android 4.1 (Jelly Bean) ako minimálnu API úroveň, vďaka ktorej bude aplikácia fungovať na približne 99.6% zariadení.

1.1 Používateľské rozhranie

Pred začatím písania kódu si musíme vytvoriť rozhranie pri tvorbe ktorého sme použili Constraint Layout, ktorý vytvára rozloženie v závislosti na prvkoch a rodičoch. V grafickom editore sme rozmiestnili 2 editovacie polia na zápis IPv4 adresy a portu, 9 tlačidiel slúžiacich na odosielanie datagramov a textové pole, ktoré informuje používateľa o stave komunikácie. Pridali sme aj 2 horizontálne a 1 vertikálnu líniu (ang. guideline) vďaka ktorým sa nám elementy ľahšie prepájali. Okrem nastavovania vzdialeností medzi prvkami – obmedzení na prepájanie elementov sme použili aj reťaze (ang. chain). Aplikácia je limitovaná na vertikálny smer telefónu.

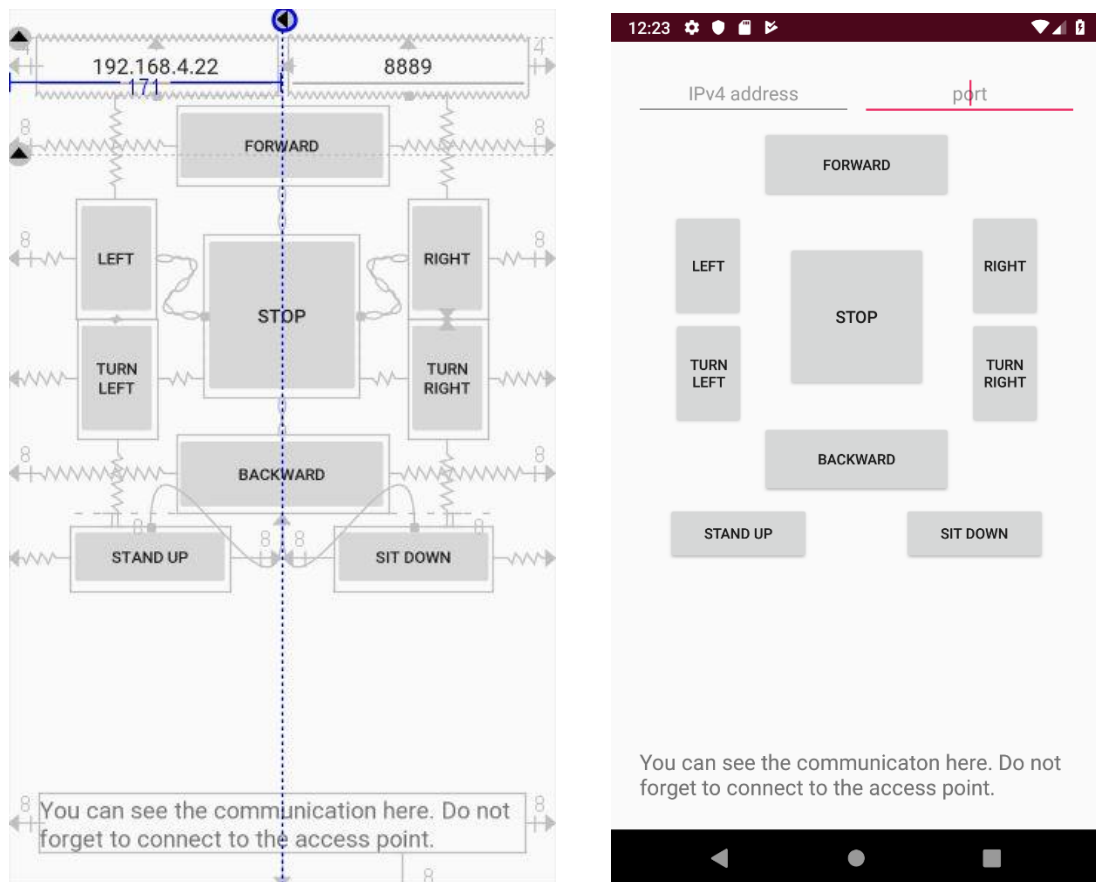
1.2 Spracovanie údajov

Na sfunkčnenie používateľského prostredia si vytvoríme dve triedy. Trieda MainActivity slúži na komunikáciu používateľa s aplikáciou. Vo vygenerovanej funkcii onCreate() si zadefinujeme elementy a tlačidlám nastavíme listener MyLovelyOnClickListener reagujúci na kliknutia. Každému tlačidlu je v konštrukторе listenera zadefinovaný reťazec, ktorý spustí príslušný pohyb hexapoda. Ak používateľ stlačí tlačidlo, tak je zavolaná funkcia onClick(), ktorá skontroluje zadaný port. Následne je zavolaná funkcia sendCommand(), ktorá skonštruuje nové vlákno typu UdpThread s IP adresou, portom, udpHandlerom a príkazom na preposlanie. UdpHandler je vnorená trieda, ktorá dedí od triedy Handler. Jej úlohou je spracovať stavy a správy poslané z vlákna. Ostatné funkcie, ktoré sa nachádzajú v triede MainActivity:

- isInteger() – skontroluje, či je zadaný reťazec celé číslo
- updateState() - zadaným reťazcom aktualizuje text v textovom poli
- threadEnd() - vynuluje premennú udpThread

- sendCommand() - vytvorí a spustí vlákno na komunikáciu

Trieda UdpThread dedí od triedy Thread a slúži na preposlanie reťazca. Vo funkcii run() dochádza k vytvoreniu socketu, skontrolovaniu IP adresy, prekonvertovaniu reťazca na pole bytov a odoslaniu a následnému prijatiu datagramu do zásobníka od ESP8266. Ak je správa prijatá, porovná sa s poslaným reťazcom a úspešnosť komunikácie sa prostredníctvom sendState() vyhodnotí v triede MainActivity, kde ho spracuje UdpHandler. Mobilný telefón musí byť pripojený na prístupový bod hexapoda.



Obrázok 22 a) rozloženie elementov b) hotová aplikácia

Záver

Cieľom tejto práce bolo zostrojiť šesťnohého robota typu hexapod. Tento robot mal byť schopný pohybu prostredníctvom diaľkového ovládania. V práci je poukázané na problematiku hexapodov, ich stabilitu, výhody, nevýhody a zhodnotenie súčasného stavu v oblasti hexapodov. Jedným z cieľov práce bolo navrhnutie vhodnej konštrukcie hexapoda. Pri návrhu sme sa inšpirovali prírodou, kvôli čomu sme zvolili podlhovastý tvar tela. Pri modelovaní končatín sme zvolili 3 stupne voľnosti vďaka, ktorým sa robot dokáže pohybovať všetkými smermi. Následne bola konštrukcia vytlačená na 3D tlačiarňi. Ďalším dôležitým krokom bol výber hardvéru, kde sme boli výrazne obmedzený rozpočtom. Keďže každá noha má 3 kĺby adekvátne sme zvolili analógové servo motory, dosky ovládajúce motory tzv. servo shiedly, regulátor napätia a napájanie. Ako mikropočítač sme zvolili Wemos D1 Mini s mikročipom ESP8266, ktorý má zabudovaný WiFi anténu. Túto charakteristiku sme využili pri komunikačnom protokole, ktorý je založený na UDP datagramoch posielajúcich reťazce, ktoré spúšťajú funkcie. Softvér robota bol naprogramovaný v prostredí Arduino a je založený na inverznej kinematike. Robot je schopný chôdze dopredu, dozadu, doľava, doprava a otočenia na mieste buď vpravo alebo vľavo. Ďalej sme si vyskúšali rôzne typy chôdze, ktoré sú typické pre hexapody. Na diaľkové ovládanie hexapoda sme si zvolili mobilný telefón s operačným systémom Android. Aplikácia bola naprogramovaná v jazyku java v prostredí Android Studio.

Vytvorený hexapod slúži ako platforma na pridanie senzorov, ktoré by mohli viesť k zabudovaniu umelej inteligencie a tým pádom by ho bolo možné využiť na prieskumné výpravy.

Zoznam použitej literatúry

1. **BÖTTCHER, Sven.** Principles of robot locomotion. [Online] Department of Computer Science SIU. 2006. [Dátum: 11. január 2019.] Dostupné na: <http://www2.cs.siu.edu/~hexmoor/classes/CS404-S09/RobotLocomotion.pdf>.
2. **HEXMOOR, Henry.** Essential Principles for Autonomous Robotics. [Online] Carbondale, Illinois: MORGAN & CLAYPOOL PUBLISHERS, 2013. [Dátum: 11. január 2019.] ISBN: 9781627050593 Dostupné na: https://www.morganclaypool.com/doi/suppl/10.2200/S00506ED1V01Y201305AIM021/suppl_file/Hexmoor_Ch1.pdf.
3. **MURTHY S. S., RAIBERT M. H.** 3D One-Leg Hopper. [Online] Cambridge, Massachusetts: Massachusetts Institute of Technology, 1984. [Dátum: 11. január 2019.] Dostupné na: http://www.ai.mit.edu/projects/leglab/robots/3D_hopper/3D_hopper.html.
4. **GASPAR BADILLO, Jose Eduardo a kol.** *Four DOF pneumatic robot design and hardware interface*. s.l. : 2017 XIII International Engineering Congress (CONIIN), IEEE, 2017. ISBN: 978-1-5386-1694-9.
5. **HYON, Sang Ho a kol.** *Design and Experimental Evaluation of a Fast Torque-Controlled Hydraulic Humanoid Robot*. s.l. : IEEE/ASME Transactions on Mechatronics, 2016, vol. 22, no. 2, 623-634 s. ISSN: 1941-014X.
6. **REZAZADEH, Siavash a kol.** Robot Leg Design: A Constructive Framework. *IEEEExplore*. [Online] 2018. [Dátum: 12. január 2019.] Dostupné na: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8466587>.
7. **KOSTROŠ, Juraj.** *Návrh a konštrukcia kráčajúceho mobilného robota*. Bratislava : FEI STU, 2010, Bakalárska práca.
8. **OLARU SORIN Mănoiu, Nițulescu MIRCEA, Viorel STOIAN.** *Hexapod robot. Mathematical support for modeling and control*. Sinaia : 15th International Conference on System Theory, Control and Computing, IEEE, 2011. ISBN: 978-973-621-321-2.
9. **GONZÁLEZ DE SANTOS Pablo, Elena GARCIA, Joaquin ESTREMER.** *Quadrupedal Locomotion: An Introduction to the Control of Four-legged Robots*. London : Springer, 2006. ISBN 978-1-84628-307-9, s. 3-5.

10. Atlas. *Boston Dynamics*. [Online] [2011]. [Dátum: 15. január 2019.] Dostupné na: <https://www.bostondynamics.com/atlas>.
11. SpotMini. *Boston Dynamics*. [Online] [2016]. [Dátum: 15. január 2019.] Dostupné na: <https://www.bostondynamics.com/spot>.
12. RHex. *Boston Dynamics*. [Online] [2002]. [Dátum: 15. január 2019.] Dostupné na: <https://www.bostondynamics.com/rhex>.
13. Phoenix Hexapod Walking Robot. *Lynx Motion*. [Online] [2014]. [Dátum: 16. január 2019.] Dostupné na: <http://www.lynxmotion.com/c-117-phoenix.aspx>.
14. HEXA: Programmable, Highly Maneuverable Robot. *Kickstarter*. [Online] San Francisco, 2017. [Dátum: 15. január 2019.] Dostupné na: <https://www.kickstarter.com/projects/1090670314/hexa-programmable-highly-maneuverable-robot/description>.
15. **ESTEBAN DELGADO, Dario a kol.** *Statically stable hexapod robot body construction*. Vilnius : 2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), 2016. ISBN: 978-1-5090-4473-3.
16. SERVO MOTOR SG90 DATA SHEET. *Imperial College Website*. [Online] Tower Pro. [Dátum: 1. február 2019.] Dostupné na: http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf.
17. TowerPro SG90 9g Mini Servo. *hobbyandyou*. [Online] [Dátum: 15. február 2019.] Obrázok dostupný na: <http://www.hobbyandyou.com/towerpro-sg90-9g-mini-servo-9-gram-servo-for-rc-robotics>.
18. Servo Motor SG-90. *Components 101*. [Online] 18. september 2017. [Dátum: 16. február 2019.] Dostupné na: <https://components101.com/servo-motor-basics-pinout-datasheet>.
19. **EARL, Bill**. Adafruit PCA9685 16-Channel Servo Driver. *Adafruit*. [Online] 2012. [Dátum: 20. február 2019.] Dostupné na: <https://learn.adafruit.com/16-channel-pwm-servo-driver>.
20. 16-Channel 12-bit PWM Servo Driver – PCA9685. *Makerlab Electronics*. [Online] [Dátum: 20. február 2019.] Obrázok dostupný na: <https://www.makerlab-electronics.com/product/16-channel-12bit-pwm-servo-driver-pca9685/>.
21. D1 mini. *WEMOS Electronics*. [Online] 28. december 2018. [Dátum: 21. február 2019.] Dostupné na: https://wiki.wemos.cc/products:d1:d1_mini.

22. Makerfocus D1 Mini NodeMcu 4M Bytes Lua WiFi Development Board Base on ESP8266 ESP-12F N Compatible NodeMcu Arduino. *Aamazon*. [Online] [Dátum: 10. november 2018.] Obrázok dostupný na: https://www.amazon.com/Makerfocus-NodeMcu-Development-ESP8266-Compatible/dp/B01N3P763C/ref=sr_1_13?keywords=esp8266+wemos+d1&qid=1558350122&s=gateway&sr=8-13.

23. **SCHNEIDER, Brian**. A Guide to Understanding LiPo Batteries. *Roger's Hobby Center*. [Online] 26. máj 2012. [Dátum: 10. marec 2019.] Dostupné na: <https://rogershobbycenter.com/lipoguide>.

24. HobbyKing™ Lipoly Low Voltage Alarm (2s~4s). *HobbyKing*. [Online] [Dátum: 9. marec 2019.] Dostupné na: https://hobbyking.com/en_us/hobbykingtm-lipoly-low-voltage-alarm-2s-4s.html?__store=en_us.

25. YEP 20A HV (2~12S) SBEC w/Selectable Voltage Output. *HobbyKing*. [Online] [Dátum: 9. marec 2019.] Dostupné na: https://hobbyking.com/en_us/yep-20a-hv-2-12s-sbec-w-selectable-voltage-output.html?__store=en_us.

26. **WIRZ, Raul a kol.** Transport Protocols for Remote Programming of Network Robots within the context of Telelaboratories for Education: A Comparative Analysis. Honolulu : 2007 16th International Conference on Computer Communications and Networks, IEEE, 2017. ISSN: 1095-2055.

27. **DORDAL, Peter L.** *An Introduction to Computer Networks 1.9.18*. Chicago : Loyola University Chicago, 2014. OCLC Number: 959614163, s. 325-326.

28. **GROKHOTKOV, Ivan**. ESP8266WiFi library. *ESP8266 Arduino core*. [Online] 2017. [Dátum: 2. apríl 2019.] Dostupné na: <https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/readme.html#quick-start>.

29. —. UDP. *ESP8266 Arduino Core*. [Online] 2017. [Dátum: 6. apríl 2019.] Dostupné na: <https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/udp-examples.html#udp-setup>.

30. Android Udp Client Example. *msdalp*. [Online] 9. marec 2014. [Dátum: 6. apríl 2019.] Dostupné na: <https://msdalp.github.io/2014/03/09/Udp-on-Android/>.

31. **LIANG, Oscar**. INVERSE KINEMATICS FOR HEXAPOD AND QUADRUPED ROBOTS. *Oscair Liang*. [Online] [2013]. [Dátum: 10. apríl 2019.] Obrázky inšpirované z: <https://oscarliang.com/inverse-kinematics-implementation-hexapod-robots/>.

32. Adafruit PWM Servo Driver Library. *Git Hub*. [Online] [2012]. [Dátum: 17. január 2019.] Dostupné na: <https://github.com/adafruit/Adafruit-PWM-Servo-Driver-Library>.

33. **LINDSAY, Jeremy**. Calibrating My Servos. *hackster*. [Online] MIT, 1. január 2018. [Dátum: 18. január 2019.] Obrázok dostupný na: <https://www.hackster.io/jeremy-lindsay/calibrating-my-servos-fa27ce>.

Prílohy

Príloha A: Obsah CD	ii
---------------------------	----

Príloha A: Obsah CD

3D modely

Priečinok obsahuje zdrojové súbory 3D modelov častí robota pre Design Spark Mechanical.

Dokumentacia

Priečinok obsahuje používateľskú príručku pre prácu s hexapodom.

Media

Priečinok obsahuje obrázky a videá z testovania robota.

Praca

Priečinok obsahuje prácu v PDF formáte.

Softver na diaľkove ovládanie

Priečinok obsahuje softvér v podobe projektu na diaľkové ovládanie pre mobilné telefóny s operačným systémom Android. Na nahranie zdrojového kódu do telefónu odporúčame použiť vývojové prostredie Android Studio.

Softver robota

Priečinok obsahuje zdrojový kód robota, ktorý je potrebný pre jeho prevádzku, pohyb a spracovanie datagramov. Kód je určený pre mikrokontrolér Wemos D1 mini a na jeho nahranie odporúčame použiť Arduino IDE.