

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Dálkově ovládaný robotický podvozek

Ondřej Šulc

Bakalářská práce

2024

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2023/2024

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Ondřej Šulc**  
Osobní číslo: **I21097**  
Studijní program: **B0714P060001 Aplikovaná elektrotechnika**  
Téma práce: **Dálkově ovládaný robotický podvozek**  
Zadávající katedra: **Katedra elektrotechniky**

## Zásady pro vypracování

### Dálkově ovládaný robotický podvozek

Cílem práce je návrh dálkově říditelného podvozku nesoucí nezávisle polohovatelnou platformou, ve stylu tank s polohovatelnou věží, pro případnou možnost montáže aktuátorů, či senzorů.

Teoretická část práce bude obsahovat rozbor možných typů podvozků používaných pro roboty a výběr vhodného typu. Rešerši použitelných pohonů s ohledem na jejich použitelnost a případnou přesnost změny polohy. Rešerši pohonů pro řízení natočení platformy. Rešerši vhodných bezdrátových technologií pro přenos dat a řízení podvozku, jak z hlediska aplikační složitosti, tak s ohledem na řízení v reálném čase. Rešerši možností stabilizace platformy nezávisle na pohybu podvozku (uzamčení absolutního směru natočení)

Praktická část provede návrh, konstrukci a oživení konceptu podvozku robota s ovladatelnou platformou, s vybraným pohonem a konstrukcí podvozku a dálkovým obousměrným přenosem řídicích a stavových informací.

Rozsah pracovní zprávy: **30-50**  
Rozsah grafických prací:  
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

- [1] VÁŇA, V. Mikrokontroléry ATMEL AVR: popis procesoru a instrukční soubor. Praha: BEN technická literatura, 2003. 336 s. ISBN 978-80-7300-083-0.
- [2] VÁŇA, V. Mikrokontroléry ATMEL AVR: programování v jazyce C. Praha: BEN technická literatura, 2003. 216 s. ISBN 978-80-7300-102-0.
- [3] VLACH, J. Řízení a vizualizace technologických procesů. Praha: BEN technická literatura, 2002. 160 s. ISBN 978-80-86056-66-X.
- [4] BRTNÍK, B. Základní elektronické obvody. Praha: BEN technická literatura, 2011. 156s. ISBN 978-80-7300-408-8
- [5] RIPKA, P.; TIPEK, A. Master Book of Sensors. Praha : BEN, 2003. ISBN 0-12-752184

Vedoucí bakalářské práce: **Ing. Pavel Rozsival**  
Katedra elektrotechniky

Datum zadání bakalářské práce: **15. prosince 2023**  
Termín odevzdání bakalářské práce: **10. května 2024**

**Ing. Zdeněk Němec, Ph.D.** v.r.  
děkan

L.S.

**doc. Ing. Jan Pidanič, Ph.D.** v.r.  
vedoucí katedry

V Pardubicích dne 24. ledna 2024

## **Prohlášení autora**

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 9. 5. 2024

Ondřej Šulc

## **Poděkování**

Tímto bych chtěl poděkovat vedoucímu práce panu Ing. Pavlu Rozsivalovi za pomoc a vstřícný přístup při tvorbě bakalářské práce. Také bych chtěl poděkovat své rodině za podporu během celého mého studia.

## **Anotace**

Bakalářská práce se věnuje návrhu a provedení dálkově řízeného podvozku s nástavbou v podobě věže s dělem do podoby tanku. Toto je následně řízeno pomocí mikroprocesoru ESP32, který je ovládán přes webový server v kódu HTML, do něhož je implementován websocket pro rychlejší komunikaci mezi uživatelem a mikroprocesorem. V podvozku jsou dva motory a servomotor pro otáčení věže. Ve věži jsou servomotory pro elevaci, depresi a stabilizaci děla a servomotor pro „střílení“. V teoretické části jsou popsány různé typy podvozků pro roboty, zjišťování polohy podvozku, vhodné pohony pro řízení podvozku, věže, děla a vhodné bezdrátové technologie pro řízení podvozku a možnosti stabilizace platformy. V praktické části se rozebere, jaký programovací jazyk jsem zvolil, výběr vhodných součástí a popis jednotlivých částí programu.

## **Klíčová slova**

ESP32, HTML server, websocket, DC motor, servomotor, bezdrátový přenos, stabilizace

## **Title**

Remotely controlled robotic platform

## **Annotation**

This bachelor's thesis is devoted to the design and implementation of a remote-controlled chassis with a superstructure in the form of a turret and a cannon in the form of a tank. This is then controlled by an ESP32 microprocessor, which is controlled via a web server in HTML code, into which a websocket is implemented for faster communication between the user and the microprocessor. There are two motors in the chassis and a servo motor to rotate the turret. In the turret are servomotors for gun elevation, depression, and stabilization, and a servomotor for "firing." The theoretical part describes different types of chassis for robots, chassis positioning, suitable drives for chassis control, turrets, cannons, and suitable wireless technologies for chassis control and platform stabilization options. The practical part will discuss which programming language I chose, the selection of suitable components, and a description of individual parts of the program.

## **Keywords**

ESP32, HTML server, websocket, DC motor, servo-motor, wireless transmission, stabilization

# Obsah

<b>Seznam zkratk</b> .....	<b>8</b>
<b>Seznam obrázků</b> .....	<b>9</b>
<b>Seznam tabulek</b> .....	<b>9</b>
<b>Úvod</b> .....	<b>10</b>
<b>1 Teoretická část</b> .....	<b>11</b>
1.1 Typy podvozků pro roboty .....	11
1.1.1 Kolový podvozek: .....	11
1.1.2 Podvozek s pásovým pohonem: .....	11
1.1.3 Nožičkový podvozek: .....	12
1.1.4 Kombinovaný podvozek:.....	12
1.1.5 Kolejový podvozek.....	13
1.2 Typy řízení a zjišťování polohy podvozku.....	13
1.2.1 Kola (např. auto):.....	14
1.2.2 Pásy (např. tank):.....	14
1.2.3 Nohy (např. všesměrové řízení).....	14
1.2.4 Koleje: .....	14
1.3 Typy motorů pro roboty .....	14
1.3.1 Stejnoseměrný motor (DC motor):.....	15
1.3.2 Krokový motor (Stepper motor):.....	15
1.3.3 Servomotor: .....	16
1.3.4 Kombinace motorů a řadicích desek: .....	16
1.3.5 Lineární aktuátory: .....	17
1.3.6 Hybridní krokový motor:.....	17
1.4 Typy motorů pro řízení platformy .....	17
1.4.1 Stejnoseměrný motor (DC motor):.....	17
1.4.2 Krokový motor (Stepper motor):.....	17
1.4.3 Servomotor: .....	17
1.4.4 Lineární aktuátor: .....	17
1.4.5 Hydraulický pohon: .....	18
1.4.6 Elektromagnetický pohon:.....	18
1.4.7 Pneumatický pohon: .....	18

1.5	Typy bezdrátových přenosů pro roboty .....	18
1.5.1	Wi-Fi (802.11): .....	19
1.5.2	Bluetooth (802.15): .....	19
1.5.3	Zigbee (802.15.4): .....	20
1.5.4	LoRaWAN: .....	21
1.5.5	5G: .....	21
1.6	Možnosti stabilizace platformy nezávisle na pohybu podvozku .....	21
1.6.1	Gyroskop a Akcelerometr: .....	21
1.6.2	Inerciální měření (IMU): .....	22
1.6.3	Vizionární systémy: .....	22
1.6.4	LIDAR: .....	22
<b>2</b>	<b>Praktická část – součástky .....</b>	<b>23</b>
2.1	Vývojové prostředí Arduino IDE .....	23
2.2	Zapojení obvodu a schéma .....	23
2.3	Popis interních částí systému .....	25
2.3.1	Napájecí část .....	25
2.3.2	Driver pro motory H-můstek (L298N) .....	25
2.3.3	Stejnoseměrný motor DC 130 .....	26
2.3.4	Servopohon SG90 .....	26
2.3.5	Měnič napětí (step-down) – XL4015 .....	26
2.3.6	Akcelerometr a gyroskop MPU6050 .....	27
2.3.7	Podvozek robota – pásový podvozek .....	27
2.4	Funkce systému – popsání řízení .....	28
<b>3</b>	<b>Popis programu .....</b>	<b>32</b>
3.1	Řídící jednotka .....	32
3.2	Řízená jednotka .....	32
3.2.1	Knihovny a proměnné .....	32
3.2.2	Funkce setup() .....	33
3.2.3	Funkce pro dekódování websocket packetu .....	34
3.2.4	Funkce pro řízení podvozku, věže a speciální předdefinované manévry ....	35
3.2.5	Popis serveru .....	36
3.2.6	Stabilizace děla .....	37
<b>4</b>	<b>Finální zpracování .....</b>	<b>38</b>



4.1 Problémy a poruchy.....	41
4.2 Doporučení a možná vylepšení.....	41
<b>Závěr .....</b>	<b>42</b>
<b>Literatura .....</b>	<b>43</b>
<b>Příloha A – Program pro řízení robota .....</b>	<b>47</b>
<b>Příloha B – Plošný spoj pro zapojení ESP32.....</b>	<b>48</b>

## **Seznam zkratk**

HTML – HyperText Markup Language – HyperTextový značkovací jazyk

WebSocket – Komunikační protokol

DC – Direct Current – Stejnosměrný proud

ESP32 – Vývojová deska ESP32 2,4 GHz Dual-Mode Wi-Fi + Bluetooth modul antény

RC – Remote Control

Wi-Fi – Wireless Fidelity – bezdrátové připojení k internetu či k sítím

BT Bluetooth – protokol 802.15.1, bezdrátové připojení do malých sítí

GPS – Global Positioning System – globální systém určování polohy

I2C Inter – Integrated Circuit – sériový komunikační protokol

IoT – internet of Things – internet věcí

PWM – Pulse Width Modulation – pulzně šířková modulace

C – kapacita kondenzátoru

## Seznam obrázků

Obrázek 1 - Schéma pásu [9].....	12
Obrázek 2 - nejčastější zástupce malých stejnosměrných motorů [13].....	15
Obrázek 3 - průřez stejnosměrného motoru [12].....	15
Obrázek 4 - průřez krokového motoru [11].....	16
Obrázek 5 - schéma Wi-Fi sítě [10].....	19
Obrázek 6 – schéma BT sítě [19].....	20
Obrázek 7 - ESP32 .....	23
Obrázek 8 - schéma obvodu .....	24
Obrázek 9 - driver pro ovládání motorů .....	26
Obrázek 10 - Měnič napětí .....	27
Obrázek 11 - pásový podvozek [14].....	28
Obrázek 12 - první diagram.....	29
Obrázek 13 - druhý diagram.....	30
Obrázek 14 - třetí diagram.....	31
Obrázek 15 - finální tank .....	39
Obrázek 16 - vnitřek tanku .....	39
Obrázek 17 - První prototyp .....	40
Obrázek 18 - základní podvozek .....	40
Obrázek 19 - ovládací panel .....	41

## Seznam tabulek

Tabulka 1 - Vývody pinů.....	24
Tabulka 2 - Vlastnosti baterii 18650 .....	25
Tabulka 3 - Vlastnosti motoru DC130 .....	26

## Úvod

V dnešní době se RC modely stále více rozšiřují mezi obyčejné lidi, ale jejich cena je stále vysoká a při zdražování baterií se zřejmě ještě zvýší. Mezitím domácí výroba je značně levnější, zábavnější a zejména se uživatel vždy něco naučí. Navíc s rozmachem umělé inteligence se již brzo objeví humanoidní roboti, proto seznámení se s tímto tématem může být do budoucna prospěšné.

Cílem této bakalářské práce je návrh a konstrukce jednoduchého RC modelu v podobě „tanku“, který bude řízen pomocí Wi-Fi serveru. To znamená, že uživatel může zařízení ovládat ze všech přístrojů, které se připojí k Wi-Fi a disponují webovým prohlížečem. V teoretické části budou rozebrány různé druhy podvozků, se kterými se můžeme setkat u robotů, a jejich motory pro řízení a ovládání. Také možné pohony pro otáčení věže a přidaného děla, popř. jiných nástaveb. Dále možné bezdrátové přenosy pro řízení a přenos dat mezi uživatelem a modelem a stabilizace platformy nezávisle na otáčení podvozku.

Praktická část se zaměří na návrh a konstrukci daného podvozku a jeho následné zpracování, se zapojením platformy ve formě věže s polohovacím dělem a jeho následnou stabilizací. RC tank bude vybaven mikroprocesorem do kterého bude nahrán program pro přenos dat mezi uživatelem a mikroprocesorem.

Úkolem je vytvořit funkční model RC tanku, který bude ekonomicky přijatelný pro uživatele a energeticky obnovitelný. Ve finální části se představí konečný výrobek a možné zlepšení daného modelu.

Výsledky této práce by měli sloužit jako návod na tvorbu vlastního modelu, výběr daných součástí, možností technologií a přínos pro začínající uživatele, kteří nemají velkého povědomí o tvorbě, návrhu a programování daných modelů.

# 1 Teoretická část

## 1.1 Typy podvozků pro roboty

Pro malé roboty existuje spousta typů podvozků a každý z nich má své výhody a nevýhody a svůj specifický systém řízení. V této části se je pokusím popsat, a nakonec zdůvodnit jaký jsem vybral a z jakého důvodu.

### 1.1.1 Kolový podvozek:

**Diferenciální pohon:** U robotů se dá použít dva a více motorů, z nichž každý se může nezávisle pohybovat a neovlivňovat tím ostatní motory.

**Omni-kola:** Kola jsou vybavena válcovými nebo koulovými válečky, což umožní pohyb ve všech směrech nezávisle na ostatních motorech.

Často se používají kulatá hladká kola, která jsou lehce pogumovaná, ale lze použít i jiné typy.

**Výhody:** Jedná se o jednoduchý design, který lze snadno implementovat, nízká pořizovací cena a dobrá obratnost, zejména snadné otáčení na místě.

**Nevýhody:** Mezi hlavní nevýhody patří problémy s terénem a omezená stabilita robota.

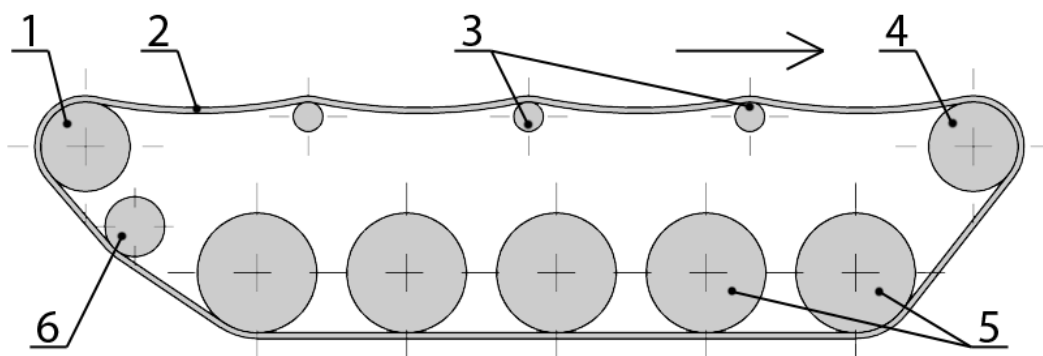
[29]

### 1.1.2 Podvozek s pásovým pohonem:

**Tankový podvozek:** Robot je vybaven dvěma pásovými pohony, což umožňuje pohyb vpřed, vzad a otáčení na místě.

Vyznačení jednotlivých částí je na obrázku (Obrázek 1 - Schéma pásu). Každý pásový podvozek musí být vybaven minimálně zadním (hnacím) kolem (1), které je ozubené a předním (napínacím) kolem (4).

Podvozek lze doplnit o pojezdová kola (5), tak aby se důležité prvky nedotýkaly povrchu, hmotnost se rozložila po celé ploše podvozků a aby pásy nesklouzly do strany. Možné je podvozek vybavit kladkami (3,6) pro pomoc při zpětném chodu tanku.



Obrázek 1 - Schéma pásu [9]

**Výhody:** Překonáváním a průchodností terénu a jeho stabilita zejména na nerovných površích, oproti ostatním podvozkům je velice nízká hodnota specifického tlaku na podložku.

**Nevýhody:** Z důvodu pásu a jejich tření, složitější konstrukce apod., dosahují tyto typy nižších rychlostí a je vyžadován silnější pohon.

[9],[29]

### 1.1.3 Nožičkový podvozek:

Mezi tento typ podvozků patří nohy a jiné končetinové mechanismy. Smyslem je se co nejvíce přiblížit lidskému modelu. Umožňují chůzi a další složitější pohyby. Pomocí motorů a servomotorů lze napodobit klouby a svaly.

**Výhody:** Nožičkový podvozek umožňuje pohyby, které připomínají chůzi a má možnost překonávat větší překážky, např. kameny.

**Nevýhody:** U těchto podvozků je složitější kinematika, zejména programování a kontrola pohybu. Tyto typy dosahují nižších rychlostí.

[29]

### 1.1.4 Kombinovaný podvozek:

Složitější podvozky, které poskytují různé kombinace podvozků. Nejčastěji jsou složeny z kol a pásových podvozků. Pro zvýšení ovladatelnosti a možnosti použití pouze jednoho motoru u pásového podvozku bez nutnosti diferenciálu.

**Výhody:** Kombinují výhody použitých podvozků. U kombinace kol a pásu se zvyšuje jejich průchodnost terénem, ovladatelnost a snižuje se pořizovací cena.

**Nevýhody:** U kombinace kol a pásu se snižuje rychlost oproti kolovému podvozku.

### 1.1.5 Kolejový podvozek

Využívá se na rovných plochách, za použití kolejí nebo přesně definované dráze. Tento způsob usnadňuje sledování čáry nebo různých vzorů. Hlavní využití je v automatizovaných linkách.

Základem jsou kolejnice, které jsou z kovu či plastu a na nich se používají speciální kola, která mohou být pogumovaná nebo s plastovými obaly pro snížení tření. Pohon je následně zajišťován pomocí libovolného motoru.

**Výhody:** Přesné sledování daných vzorů a jeho stabilita.

**Nevýhody:** Není vhodný pro použití na nerovném či nepravidelném povrchu. Nutnost použití kolejí, což znemožňuje univerzální použití.

[33]

Pro mé použití jsem zvolil **Podvozek s pásovým pohonem**, a to zejména kvůli jeho dobré průchodnosti terénem a stabilitě, která zlepšuje podmínky pro následnou stabilizaci děla, což pro mě představovalo primární důvod výběru.

## 1.2 Typy řízení a zjišťování polohy podvozku

Pro podvozky robotů existuje spousta metod pro zjišťování polohy. Mezi nejčastější patří odometrie, GPS či použití akcelerometrů. Pro zjišťování polohy budou popsány odometrie a akcelerometry.

**Odometrie:** Jedná se o metodu, která je založena na odhadu pozice z dat z enkodéru či snímače otáčení kol. Začíná se znalostí počáteční pozice robota, následně se na základě ujetých vzdáleností a otáček koleček odhaduje nová pozice. Při tomto odhadu se mohou vyskytnout chyby způsobené například klouzáním kol, nerovnostmi terénu nebo opotřebením pneumatik.

**Akcelerometr:** Tyto senzory umožňují měřit akceleraci. Začíná se znalostí počáteční polohy a následně se při pohybu seismická hmota uvnitř senzoru přesune na tlakový či kapacitní snímač a z hodnoty zrychlení se následnou dvojí integrací získá délka trajektorie. Při tomto měření může dojít k tzv. driftu, kdy se hmota při zastavení nevrátí do počáteční (nulové) pozice.

[21]

Řízení podvozku se provádí změnou chodu motorů, snížením jejich otáček nebo natočením jedné či více náprav. Popis následuje níže.

### 1.2.1 Kola (např. auto):

Pro řízení se nejčastěji vytvoří model tříkolky, kde se u čtyřkolek vytvoří virtuální přední kolo, podle něhož se následně vypočítává ujetá vzdálenost a popř. poloha.

Tříkolka se pohybuje buď po přímce, nebo po kružnici, a změna pozice a orientace závisí na úhlu natočení řídicího kola. Motory mohou být na přední i zadní nápravě. Pro jednoduché roboty se doporučuje pohon na zadní nápravu. U tohoto typu nelze zajistit otáčení na místě.

[25]

### 1.2.2 Pásy (např. tank):

Model tanku (často nazýván diferenciální podvozek) má dvě nezávisle poháněná kola.

Robot jede rovně, otáčí se na místě nebo se pohybuje po kružnici, což závisí na rozdílu rychlostí levého a pravého kola. Pro pohyb vpřed a vzad se motory točí ve stejném směru, pro otáčení robota jsou motory v protichodu. Výsledná poloha se vypočítá ze znalostí pohybu hnacích, popř. napínacích kol či akcelerometru.

[25]

### 1.2.3 Nohy (např. všesměrové řízení)

Model všesměrového robota umožňuje pohyb ve všech směrech nezávisle na orientaci. Speciální kola, či nohy umožňují robotu pohybovat se do všech stran, což zajišťuje holonomní pohyb. U noh se toto řízení používá pro klouby či chodidla. Zjišťování polohy je složitější, jelikož se musí zohlednit všechny pohyby částí noh.

[25],[29]

### 1.2.4 Koleje:

Pro kolejový podvozek se nejčastěji využívají kola, která jsou na kolejnicích. Řízení je nutné pro přední i zadní nápravu, a to stejně jako u kol. Využívá se otočného mechanismu, který umožňuje nápravám se pohybovat kolem středové osy, což umožní manévrování v obloucích. Pro větší změnu pohybu se používají výhybky. Pro tento typ lze odhad polohy určit i ze senzorů na kolejích.

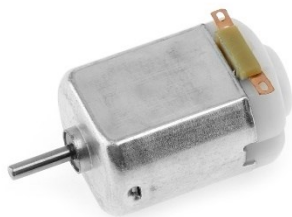
## 1.3 Typy motorů pro roboty

Pro malé roboty existuje spousta typů motorů a každý z nich má své výhody a nevýhody. Rozdílně napájení, moment motoru a rychlost otáček. V této části se je pokusím popsát, a nakonec zdůvodnit jaký jsem vybral a z jakého důvodu.

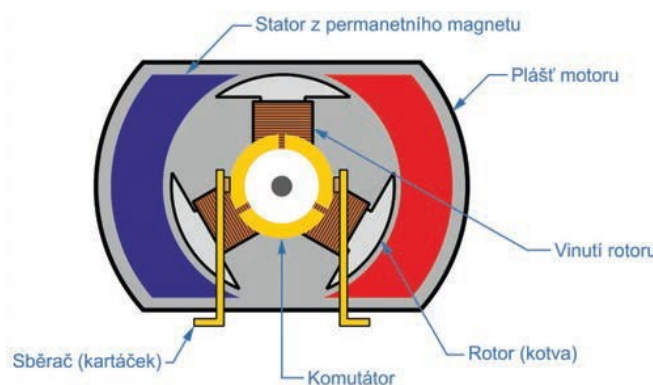


### 1.3.1 Stejnospměrný motor (DC motor):

Stejnospměrný motor s permanentním magnetem je nejčastěji používanou verzí v oblasti pohonu robotů. Tyto motory obvykle pracují na vysokých otáčkách a na nízkém momentu síly, což se pro roboty nehodí, proto se přidává převodovka, která sníží otáčky a zvýší moment síly motoru. Většina těchto motorů má dva elektrické kontakty. Po připojení jmenovitého napětí zapříčiníme roztočení motoru, při snížení napětí snížíme otáčky, při změně polaroty změníme směr otáček.



Obrázek 2 - nejčastější zástupce malých stejnosměrných motorů [13]



Obrázek 3 - průřez stejnosměrného motoru [12]

**Výhody:** Jeho poměr výkon/hmotnost a relativně snadné řízení otáček. Cena a široká nabídka těchto motorů je také značnou výhodou.

**Nevýhody:** Díky komutátoru se na tomto motoru objevuje elektromagnetické rušení. Zvýšená údržba oproti ostatním motorům a složitější polohové řízení.

[22],[2],[28]

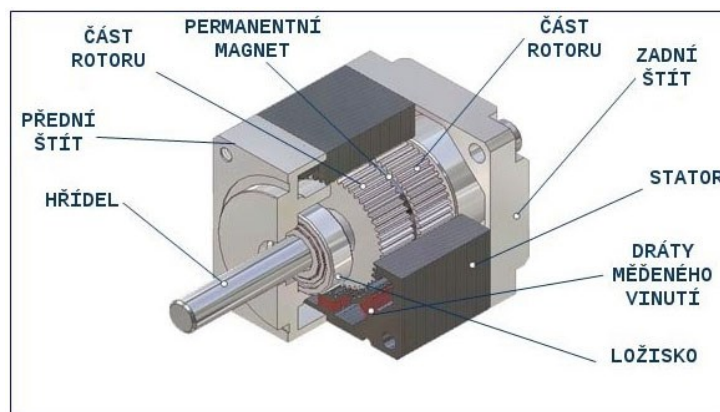
### 1.3.2 Krokový motor (Stepper motor):

V dnešní době existují dva nejvíce používané typy krokových motorů – reakční a hybridní. Tento typ motoru má speciální konstrukci se zubatým rotorem. Každý krok odpovídá pevnému úhlu otočení, který je řízen pulzy do elektromotoru. Každý impuls posune rotor o jednu fixní pozici.

**Výhody:** U těchto motorů je možná přesná kontrola stability v klidovém režimu, polohy a otáček.

**Nevýhody:** Tyto motory jsou cenově dražší a také dosahují nižších rychlostí.

Na následujícím obrázku (obrázek 4) je průřez krokového motoru, který je důležitý k popisu.



Obrázek 4 - průřez krokového motoru [11]

[28],[11],[22],[2]

### 1.3.3 Servomotor:

Servomotor obsahuje stejnosměrný motor, převodovku a výstupní hřídel. K ní je přiřazen potenciometr, který slouží jako senzor úhlu natočení. Tyto motory navíc obsahují integrovaný systém zpětné vazby, který přijímá signály o žádané poloze a přes vestavěnou elektroniku nastaví motor do požadované polohy.

**Výhody:** Přesná kontrola polohy a otáček a následná stabilita v klidovém stavu.

**Nevýhody:** Tento typ motorů má často omezený rozsah otáčení (často 180°), nebo není možné určit přesně polohu. Tyto motory dosahují vyšších pořizovacích nákladů.

[22],[28],[2]

### 1.3.4 Kombinace motorů a řadicích desek:

Tato metoda umožňuje použití motorů a k nim přidaným řadičům pro snadnější pohyb, řízení a ovládání. Často se používají H-můstky, zejména kvůli jejich jednoduchosti.

**Výhody:** Snazší ovládání a řízení oproti klasickému zapojení motorů.

**Nevýhody:** Nutnost dodat řadič na požadované napětí a proud a počítat s odběrem pro tento řadič.

### 1.3.5 Lineární aktuátory:

Převodou rotační pohyb motoru na lineární pohyb a tím vytvoří tlačnou nebo tažnou sílu. Může být realizován pomocí šroubového pohonu, válcového pohonu nebo jiných mechanismů.

**Výhody:** Tento typ je přizpůsobený pro lineární pohyb místo otáčení, často se používá pro posuvné nebo tahové aplikace. Některé druhy nabízí jednoduchou implementaci.

**Nevýhody:** Tyto motory dosahují vyšších cen a jsou omezeny rychlostí.

[27]

### 1.3.6 Hybridní krokový motor:

Tyto motory kombinují vlastnosti krokových motorů, stejnosměrných motorů a servomotorů. Jeho rotor obsahuje více zubů což umožňuje použití v náročnějších podmínkách.

**Výhody:** Tento typ umožňuje přesnou kontrolu polohy, a to díky použití krokového motoru. Také dosahuje větší účinnosti a vyšší rychlosti. Toto je spjato s větším točivým momentem.

**Nevýhody:** Tyto motory dosahují vyšších cen a mají složitější konstrukci zapříčiněnou kombinací různých motorů.

[22],[28]

Pro mé použití jsem zvolil **Kombinaci motorů a řadicích desek**. Jako motor stejnosměrný, a to kvůli jeho snadné implementaci, výkonu, rozměrům motoru a cenové přijatelnosti.

## 1.4 Typy motorů pro řízení platformy

Pro malé roboty existuje spousta typů motorů a každý z nich má své výhody a nevýhody. Rozdílné napájení, moment síly a rychlost otáček. V této části se je pokusím popsat, a nakonec zdůvodnit jaký jsem vybral a z jakého důvodu.

### 1.4.1 Stejnosměrný motor (DC motor):

[Viz 1.3.1](#)

### 1.4.2 Krokový motor (Stepper motor):

[Viz 1.3.2](#)

### 1.4.3 Servomotor:

[Viz 1.3.3](#)

### 1.4.4 Lineární aktuátor:

[Viz 1.3.4](#)

#### 1.4.5 Hydraulický pohon:

Tyto pohony využívají tlaku kapaliny (nejčastěji olej) pro následný pohyb. Jsou často používány pro velké stroje nebo tam, kde je potřeba velký točivý moment. Kapalina často odolává vyšším teplotám, má významné mazací účinky a je méně náchylná k únikům kapaliny. Použití tohoto typu pohonu je vhodné pro pomalé, ale přesné pohyby.

**Výhody:** Tento typ disponuje vysokým točivým momentem, a to i při vysokých zátěžích.

**Nevýhody:** Tyto pohony jsou vysoce komplexní. Dosahují vysokých hmotností a s tím spojenou nákladnou údržbou.

[26]

#### 1.4.6 Elektromagnetický pohon:

Jedná se o synchronní stejnosměrné motory, které pomocí střídače vysílají do statoru pulzy a ty vytvářejí magnetické pole. Řídící jednotka pak následně podle pulzů zapíná dané cívky tak aby se dosáhlo točivého magnetického pole. Tento typ motoru může poskytovat spojitý otáčivý pohyb.

**Výhody:** Tyto motory disponují vysokým výkonem, vysokou účinností a nízkou hlučností.

**Nevýhody:** Tyto motory dosahují vyšších cen a často jsou vybaveny i složitější elektronikou.

[32]

#### 1.4.7 Pneumatický pohon:

Tyto motory využívají stlačený vzduch či jiný plyn pro následný pohyb. Je občas používán pro robustní aplikace nebo tam, kde je nutné minimalizovat riziko elektrického nebo hydraulického selhání. Stlačený plyn je čistý, ale nemá vlastnosti jako mazivo a může docházet ke kondenzaci a poté je potřeba jeho následná filtrace. Vhodné pro rychlé a dynamické aplikace, ale nemají vysokou přesnost.

**Výhody:** Vynikají svým jednoduchým designem pro robustní prostředí.

**Nevýhody:** Dosahují nižší účinnosti ve srovnání s jinými motory.

[26]

Pro mé použití jsem zvolil **Servomotory**, a to kvůli jeho výkonu, rozměrům motoru a možnosti přesného otáčení.

### 1.5 Typy bezdrátových přenosů pro roboty

Pro malé roboty existuje spousta typů bezdrátových přenosů a každý z nich má své výhody a nevýhody. Vyžadují rozdílné napájení, mají rozdílná pásma pro přenos a rozsah přenosu. V této části se je pokusím popsat, a nakonec zdůvodnit jaký jsem vybral a z jakého důvodu.

### 1.5.1 Wi-Fi (802.11):

Wi-Fi se běžně používá ve vnitřních prostorech a pracuje na 2,4 GHz pásmu. Tento typ je velmi podobný ethernetovým sítím. Síť potřebuje 4 druhy fyzických komponent: distribuční systém, přístupový bod, přenosovou cestu, stanice – uživatel. Je možné si vybrat ze spousty standardů, které disponují specifickými vlastnostmi, různými přenosy ve spektru pásma, rozdílnou spotřebou energie a rychlostí přenosu. Následná tvorba aplikací a dekodování zpráv je složitější.

Na následujícím obrázku (Obrázek 5) je schéma malé Wi-Fi sítě a jak probíhá komunikace mezi nimi.



Obrázek 5 - schéma Wi-Fi sítě [10]

**Výhody:** Možnost připojení ze spousty zařízení bez nutnosti vytváření aplikace. U spousty typů, lze dosáhnout vysokých rychlostí přenosu na krátké vzdálenosti.

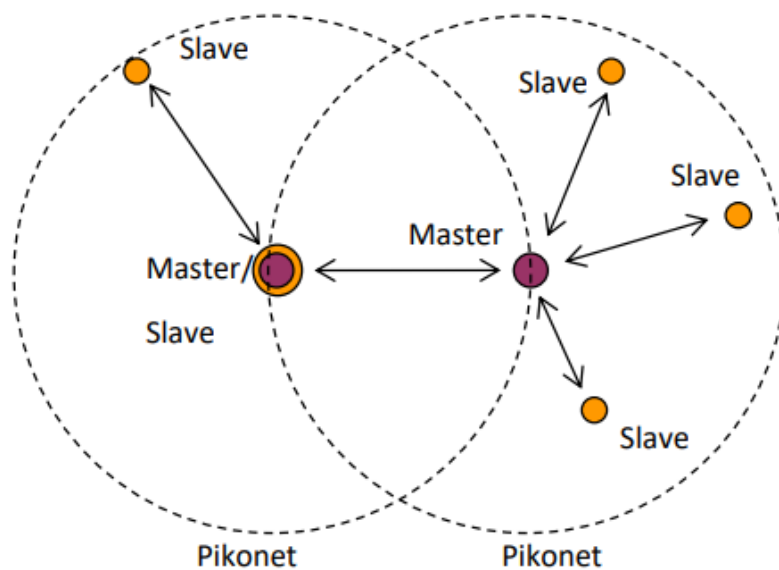
**Nevýhody:** Řízení v reálném čase má nižší latenci, ale nedosahuje vysokých rychlostí v přenosu. Nespolehlivosti na dlouhé vzdálenosti.

[18]

### 1.5.2 Bluetooth (802.15):

Bluetooth se hodí zejména pro přenos na krátké vzdálenosti a možnosti využití v jednoduchých aplikacích pro komunikaci, zejména pro spojení mezi dvěma body. Lze vytvořit piko-sítě, které obsahují 1 Master a 7 Slave zařízení. Komunikace je možná pouze Master-Slave. V dnešní době existuje spousta verzí BT a záleží na uživateli, kterou verzi se rozhodne implementovat na základě vlastností. Pracuje na pásmu 2,4 GHz, dekodování zpráv se řadí mezi jednodušší typy.

Na následujícím obrázku (Obrázek 6) je schéma malých sítí BT, které spolu vzájemně komunikují.



Obrázek 6 – schéma BT sítí [19]

**Výhody:** Snadná integrace a zapojení do spousty programů. Disponuje nižší spotřebou energie.

**Nevýhody:** Nutnost vytvořit vlastní aplikaci. Tento přenos má také nižší latenci.

[19]

### 1.5.3 Zigbee (802.15.4):

Tento přenos se používá pro sítě malých zařízení s nízkou spotřebou energie. Disponuje omezeným dosahem signálu, ale dosahem větším než Bluetooth. Primárně slouží ke sběru dat ze senzorů, které jsou v komplikovaných prostředích.

**Výhody:** Nízká spotřeba energie. Vhodné pro malé sítě s nízkou komplexitou.

**Nevýhody:** Tento typ přenosu disponuje nižší šířkou pásma a nižších přenosových rychlostí.

[24]

#### 1.5.4 LoRaWAN:

Tato metoda umožňuje nízkou spotřebu energie, ale dokáže dosáhnout vysokého dosahu signálu. Primárně slouží jako síť pro mnoho energeticky nenáročných senzorů, které jsou rozmístěny na velkou vzdálenost. Často se používá pro IoT aplikace, např. správa zařízení, osvětlení ve městech, sledování zemědělských dat.

**Výhody:** Přenos dosahuje dlouhého dosahu signálu.

**Nevýhody:** Nižší rychlosti přenosu dat a nižší šířka pásma.

[31]

#### 1.5.5 5G:

Jedná se o novou generaci mobilních sítí, která má ovšem těžší implementaci do programu a následných aplikací. Tento typ se zaměřuje na použití v mobilní sféře, ale využívá se i v průmyslových aplikacích nebo ve virtuální realitě. Pro tento typ je nutné se dohodnout s operátorem, mohlo by nastat možné rušení sítě.

**Výhody:** Vysoká přenosová rychlost a nízká latence.

**Nevýhody:** Vyšší náklady, vyšší spotřeba energie a náročnější tvorba aplikací.

[20]

Pro mé použití jsem zvolil přenos **Wi-Fi**, a to kvůli jeho snadné implementaci a možnosti připojení spousty zařízení bez nutnosti instalace aplikace.

## 1.6 Možnosti stabilizace platformy nezávisle na pohybu podvozku

Pro možnosti stabilizace platformy existuje spousta typů zařízení a součástek a slouží primárně pro udržení orientace a stability robota. V této části se je pokusím popsat.

### 1.6.1 Gyroskop a Akcelerometr:

Gyroskopy měří úhlovou rychlost zařízení v jednotlivých osách a akcelerometr měří zrychlení, což umožňují udržovat stabilitu bez ohledu na pohyb podvozku. Zařízení obsahuje setrvačnick (rotor), který zachovává polohu osy i při své rotaci.

**Výhody:** Tento typ disponuje precizním měření a rychlou odezvou.

**Nevýhody:** Mohou mít omezenou stabilitu v delším časovém horizontu kvůli kumulativním chybám.

[23]

### 1.6.2 Inerciální měření (IMU):

Kombinuje data z akcelerometrů a gyroskopů pro měření lineárních i úhlových pohybů. Některé IMU mohou také obsahovat magnetometry pro detekci magnetického pole Země.

**Výhody:** Poskytuje úplné informace o pohybu a lze ho integrovat do malých a lehkých zařízení.

**Nevýhody:** Tento typ má nižší citlivost na akcelerace a otřesy.

[23]

### 1.6.3 Vizionární systémy:

Využívá kamery a vizuální navigaci k určení absolutního směru a pozice zařízení. Následně tato vizualizace lze upravit a využít podle potřeb uživatele.

**Výhody:** Schopnost pracovat ve vnitřních prostorech nebo na místech s nízkou dostupností signálů z GPS.

**Nevýhody:** Citlivost na světelné podmínky a omezená pracovní vzdálenost

### 1.6.4 LIDAR:

Používá laserové paprsky pro měření vzdáleností k okolním objektům a vytváří trojrozměrné mapy okolí, což může být využito k udržování orientace.

**Výhody:** Přesné měření vzdáleností, efektivní i v nízké viditelnosti.

**Nevýhody:** Vysoké náklady, některé omezení ve složitých prostředích.

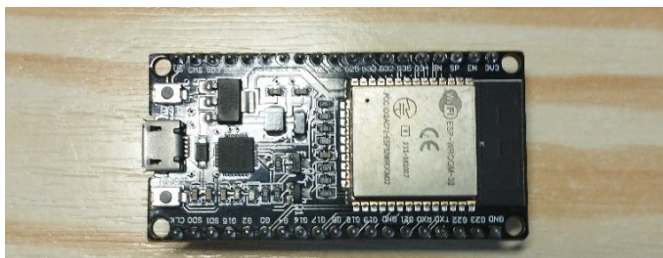
[30]

Pro mé použití jsem zvolil stabilizaci pomocí **Inerciálního měření**, a to kvůli jeho snadné implementaci, ceně i jeho přesnosti a rozměrům.



## 2 Praktická část – součástky

Pro jádro samotného robota jsem zvolil jednočipový, dvoujádrový mikroprocesor ESP-WROOM-32 (Obrázek 7), který je osazen na stejnojmenném vývojovém kitu s přidaným Wi-Fi modulem a BT modulem.



Obrázek 7 - ESP32

### 2.1 Vývojové prostředí Arduino IDE

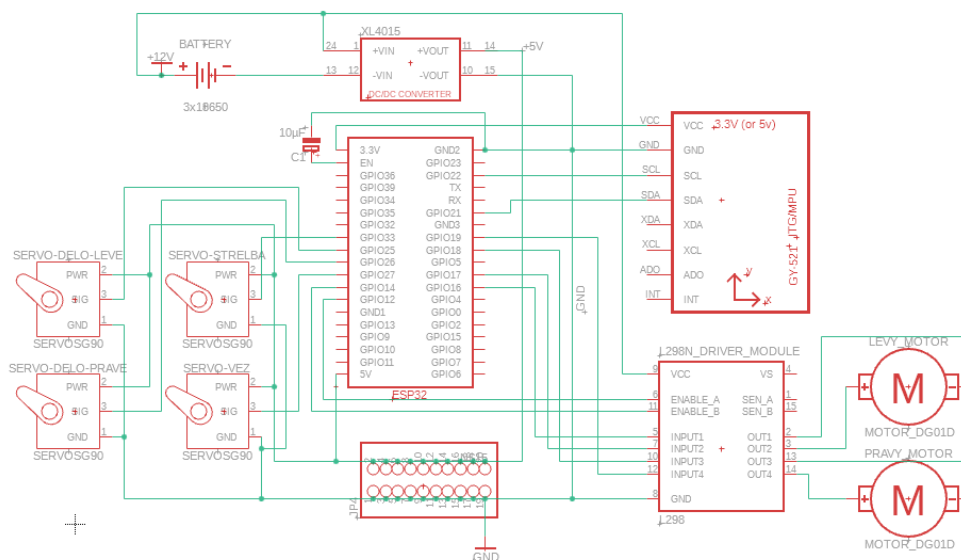
Toto prostředí vzniklo společně se vznikem platformy Arduino a postupem času se rozšířilo i na ostatní platformy. Použití je výhodné pro rozmanitost knihoven, které umožňují vytvářet různorodější programy.

Programovacím jazykem v tomto prostředí je upravený jazyk C/C++, což usnadňuje práci a stává se tak univerzálním programovacím prostředím. Většina kódu obsahuje dvě hlavní funkce: `setup()` a `loop()`. `Setup` se provede při startu mikrokontroleru a slouží pro inicializaci a nastavení různých parametrů. `Loop` se opakovaně provádí po skončení funkce `setup`.

[1]

### 2.2 Zapojení obvodu a schéma

Celý robot je napájený ze tří 3,7 V 18650 baterií, které dávají dohromady 11,1 V (při plném nabití i přes 12 V). Z tohoto napájení je vyvedena větev do H-můstku a další větev na stabilizátor 5 V pro napájení ESP32 a přidaných servomotorů pro pohyb věže a děla. K procesoru jsou, podle tabulky (Tabulka 1 – Vývody pinů), připojeny vývody k ovládání můstku, PWM signály pro ovládání servopohonů, napájení a vývody I2C sběrnice. Vývojový kit je osazen do plošného spoje, do kterého je vyvedeno napětí ze stabilizátoru.



Obrázek 8 - schéma obvodu

Tabulka 1 - Vývody pinů

Piny z ESP32	Připojení do periférii
5 V (V5)	Napájení ze stabilizátoru
EN	Uvolnění pro nahrávání ( $C = 10 \mu\text{F}$ )
GND	Zapojení na zem (nulový vodič)
G 16	Připojení do driveru (IN1)
G 17	Připojení do driveru (IN2)
G 18	Připojení do driveru (IN3)
G 19	Připojení do driveru (IN4)
G 12	PWM signál do driveru (ENA)
G 14	PWM signál do driveru (ENB)
G 25	Připojení signálu do servomotoru děla – pravé
G 26	Připojení signálu do servomotoru děla – levé
G 27	Připojení signálu do servomotoru – věž
G 33	Připojení signálu do servomotoru – střelba
G 21	Připojení I2C sběrnice – SDA
G 22	Připojení I2C sběrnice – SCL
3V3 (3,3V)	Napájení MPU6050

## 2.3 Popis interních částí systému

Dále jsem popsal jednotlivé součástky, které jsou použity v zařízení. Vybírány byly kvůli jejich účinnosti, jednoduché obsluze, vyměnitelnosti poškozených dílů a ceně.

### 2.3.1 Napájecí část

Pro napájení jsem zvolil tři 3,7V baterie v připraveném boxu. Vývody jsou následně přivedeny přes kolébkový vypínač a ochranu před vybitím do H-můstku. Vypínač je zde umístěn pro pohodlnější zapínání a vypínání robota.

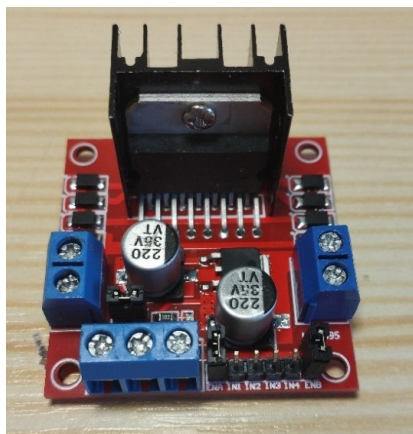
Tabulka 2 - Vlastnosti baterii 18650

Výrobce	GeB	GeB
Model	18650-3000	18650-2000
Jmenovité napětí	3,7 V	3,7 V
Jmenovitá kapacita	3000 mAh	2000 mAh
Maximální trvalý vybíjecí proud	3 A	20 A
Konstantní napětí	4,2 V	4,2 V

### 2.3.2 Driver pro motory H-můstek (L298N)

Tento modul používá čip L298N, který disponuje dobrými vlastnostmi, zejména: dobrá schopnost řízení, nízká teplota provozní teplota a anti-interface vlastnosti. Tento driver umožňuje napájení až 46 V a proud až 3 A, dlouhodoběji 2 A, pro motory je pak následné napájení v rozmezí 5–35 V v závislosti na vstupním napětí. Tento modul je vybaven i 5 V stabilizátorem pro napájení ostatních součástek, pokud ale vstupní napětí nepřesáhne 12 V je dobré zvolit samostatný stabilizátor. Tento driver lze ovládat pomocí 4 vývodů (IN1-4) díky nimž lze ovládat dva DC motory a 2 vývody pro PWM signál (ENA a ENB) pro řízení otáček. Driver je následně vybaven dvěma dvojicemi vývodu pro samotné motory (OUT1-2 pro první motor a OUT3-4 pro druhý motor).

Na následujícím obrázku (Obrázek 9) je vyfocen daný model, který jsem použil.



Obrázek 9 - driver pro ovládání motorů

### 2.3.3 Stejnosměrný motor DC 130

Motory pro řízení podvozku jsou dva DC 130 s přidanou specifickou převodovkou pro správný chod pásu. Tyto motory se liší zejména provozním napětím a momentem síly. V tabulce jsou uvedeny jednotlivé hodnoty motoru. Pro model byla vždy vybrána dvojice těchto motorů.

Tabulka 3 - Vlastnosti motoru DC130

Typ	DC 130
Jmenovité napětí	3-6 V
Spotřeba proudu při chodu	1,1 A
Volnoběžné otáčky	16500 rpm

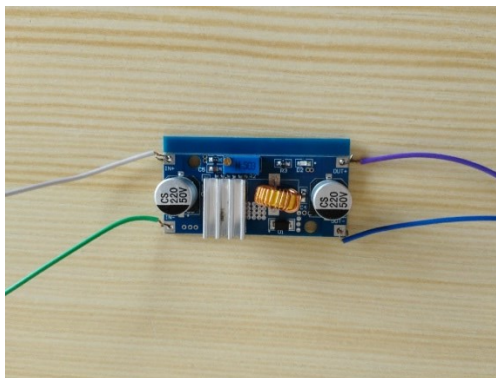
### 2.3.4 Servopohon SG90

Jedná se o plastové microservo SG90 o hmotnosti 9 g a možnosti úhlu natočení až 180°. Rychlost tohoto servomotoru je při 5 V 0.12s/60°. Provozní napětí je 3.0-7.2 V. Tento typ se vyznačuje nylonovými převody pro pevnější a silnější točivý moment než pouhý plast (v tomto případě 1.2 kg/cm). Pro mé zapojení jsem využil čtyři servomotory. Jeden pro řízení platformy (věže), další dva pro řízení elevace, deprese děla a jeho následnou stabilizaci. Poslední servomotor je pro střelbu. Tento servomotor nemá potenciometr pro určování polohy, jedná se tedy o kontinuální servomotor.

### 2.3.5 Měnič napětí (step-down) – XL4015

Jedná se o modul, který umožňuje efektivní regulaci napětí na výstupu 1,25 – 30 V při 5 A. Vstupní napětí by mělo být v rozsahu 4–38 V. Tato součástka disponuje výkonem 75 W a účinností až 96 %. Do odběru 4 A nedochází ani k významnému teplotnímu zatížení, každopádně provozní teplota je v rozsahu -40 až 85°C.

Následující obrázek (Obrázek 10) je měnič, který jsem použil.



**Obrázek 10 - Měnič napětí**

### **2.3.6 Akcelerometr a gyroskop MPU6050**

Jedná se o tříosý gyroskop a akcelerometr, který je schopný měřit natočení a zrychlení. Komunikace je možná pomocí I2C sběrnice, kterou ESP32 disponuje. Napájecí napětí je 3 až 5 V. Tento modul také disponuje AD převodníkem pro lepší přenos hodnot.

### **2.3.7 Podvozek robota – pásový podvozek**

Pásová plošina, která má možnost být napájena bateriemi AA a je poháněna dvěma stejnosměrnými motory. Navíc nabízí velkou plochu s otvory, která umožňuje instalaci různých senzorů. Motory tohoto podvozku jsou popsány v [2.3.3](#). Tento robot nedisponuje předdefinovaným napájením, nýbrž vlastním.

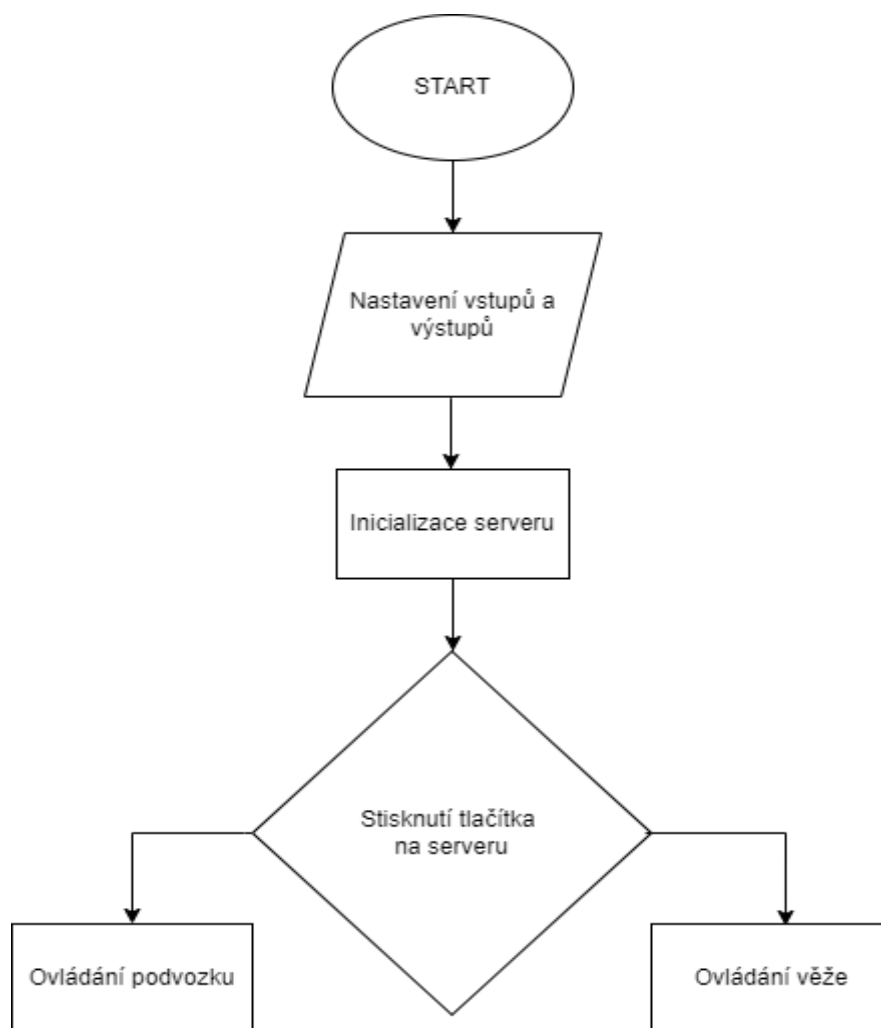
Následující obrázek (Obrázek 11) je podvozek, který jsem použil.



Obrázek 11 - pásový podvozek [14]

## 2.4 Funkce systému – popsání řízení

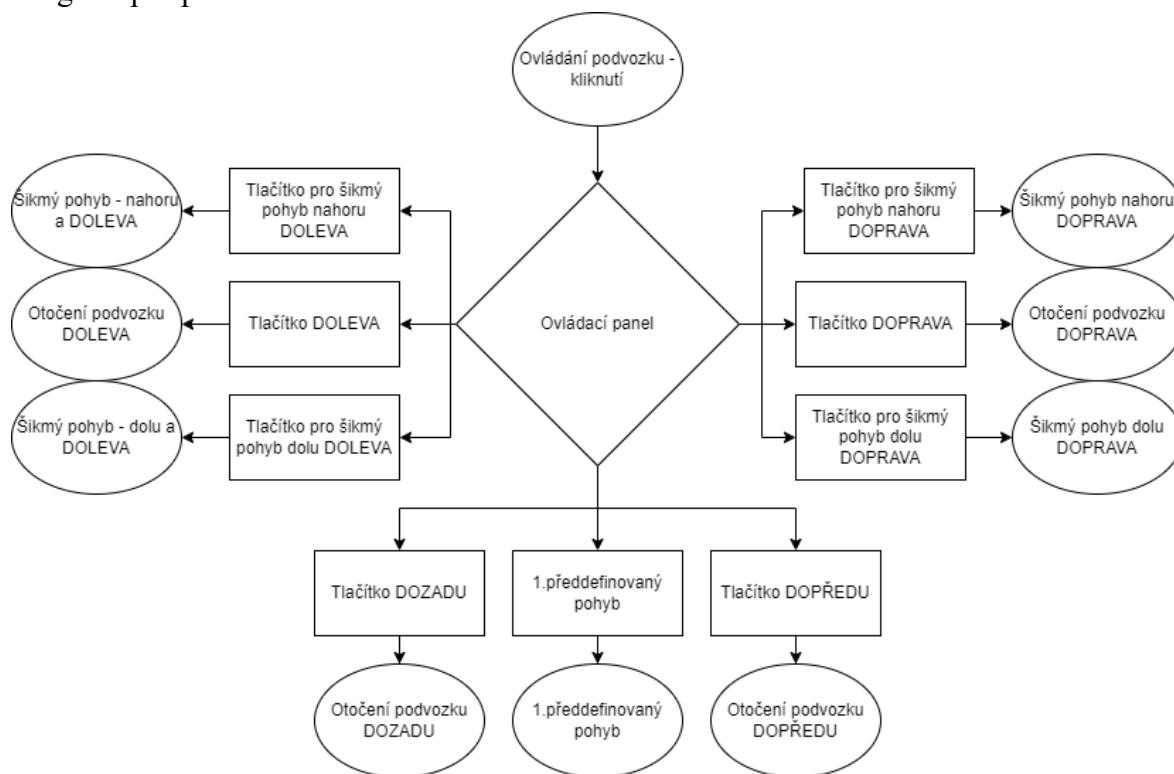
Funkce systému je popsána v přidružených vývojových diagramech. Zde je popsán první diagram ohledně hlavního programu, který proběhne po spuštění mikroprocesoru. V inicializaci se nastaví všechny potřebné vstupy a výstupy a následně se spustí Wi-Fi server, kde je potřebné ovládání. Na serveru je poté možné kliknout na potřebná tlačítka.



**Obrázek 12 - první diagram**

Ten se následně dělí na dvě části. První (horní) část je pro řízení podvozku, uprostřed je tlačítko pro střelbu a na druhou (dolní) část pro řízení věže a opět střelbu.

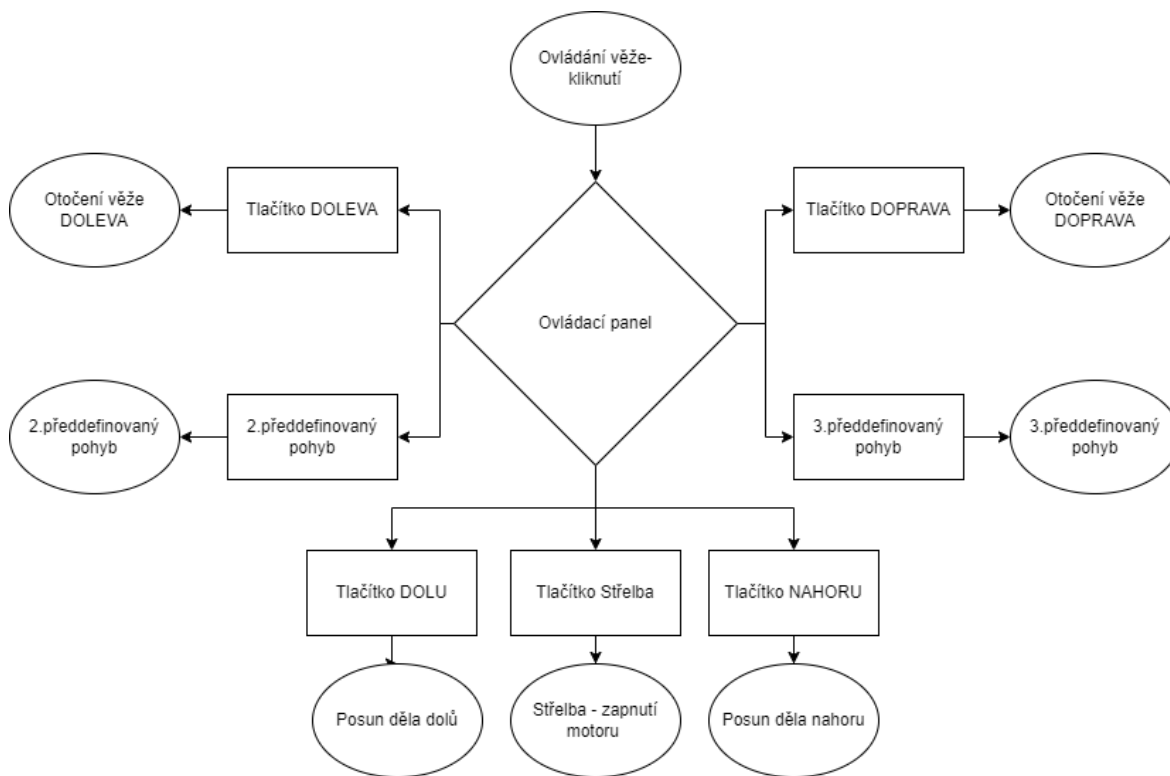
Diagram pro první část:



Obrázek 13 - druhý diagram



Diagram pro druhou část:



Obrázek 14 - třetí diagram

Pro oba diagramy je postup stejný. V případě zmáčknutí tlačítka na obrazovce se vykoná předem stanovený pohyb, který je zapsaný v konečných stavech či předdefinované speciální povely, které jsou popsány níže. U stisku tlačítka dopředu a dozadu na první části displeje se podvozek přesune dopředu či dozadu, ale u věže se dostane do horní či dolní pozice dělo.

### Předdefinované povely:

1. Předdefinovaný povel – tento povel umožní tanku jet 2 s dopředu a následně zastaví a vystřelí jeden „projektil“. V případě kontinuálního stisku tlačítka se tento povel vykonává opakovaně.
2. Předdefinovaný povel – Jedná se o otočení o 180° do levého směru.
3. Předdefinovaný povel – Jedná se o otočení o 180° do pravého směru.

## 3 Popis programu

### 3.1 Řídící jednotka

Řídící jednotka je rozhraní uživatele, které je schopno se připojit k Wi-Fi síti a disponuje webovým prohlížečem. Poté uživatel do vyhledávání napíše adresu webservru 192.168.4.1, a ta ho připojí na server pro ovládání robota. Toto je poté rozděleno do dvou částí pro jednodušší ovládání. Horní část je pro řízení podvozku a dolní část pro řízení věže ([Viz 2.4](#)). Po stisknutí tlačítka odesílá uživatelské rozhraní požadavek ve formě websocket packetu do ESP32.

### 3.2 Řízená jednotka

Celý program v ESP32 se rozděluje na několik částí, které se spouštějí v níže popsaném pořadí.

#### 3.2.1 Knihovny a proměnné

Pro správnou funkci součástí je nutné nainstalovat z prostředí Arduino IDE potřebné knihovny. Jedná se o knihovny pro funkci webservru na bázi HTTP protokolu, do kterého je vnořen websocket pro rychlejší komunikaci, vytvoření Wi-Fi spojení, funkci servomotorů a pro I2C komunikaci mezi MPU6050 a ESP32.

```
#include <Arduino.h>
#include <WiFi.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>;
#include <Wire.h>
#include <ESP32Servo.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
```

Tato část níže popisuje nastavení proměnných, které jsou používány po zbytek kódu. První odkazuje na použití maker, pro přehlednější práci při pohybu robota. Ke každému je přiřazeno číslo, které se následně používá ve switchi.

```
#define UP 1
#define DOWN 2
#define LEFT 3
#define RIGHT 4
#define STRELBA 6
#define NAHORU 7
#define DOLU 8
#define DOLEVA 9
#define DOPRAVA 5
#define STOP 0
#define PrvniPohyb 10
#define DruhýPohyb 11
#define TretiPohyb 12
#define SikmonahoruVpravo 13
#define SikmonahoruVlevo 14
```

```
#define SikmodoluVpravo 15
#define SikmodoluVlevo 16
```

Mezi nejdůležitější proměnné v tomto kódu patří val (0-2) pro zápis úhlů servomotorů, které jsou pouze hodnoty integer. Následuje tvorba PWM signálů pro řízení motorů. Pro následnou stabilizaci je vytvořen PID regulátor. Proměnná apSSID slouží pro uživatelské jméno sítě Wi-Fi a apPassword představuje heslo do dané sítě. Ostatní proměnné jsou pouze pomocné, a tedy v programu slouží pouze pro lepší orientaci či pomoc a rychlejší odezvu.

```
const int freq = 1000;
const int PWMchannel1 = 4;
const int PWMchannel2 = 4;
const int resolution = 8;

float PID, error, previous_error;
float pid_p=0;
float pid_i=0;
float pid_d=0;

double kp=0.75;
double ki=0.0001;
double kd=0;
```

### 3.2.2 Funkce setup()

Ve funkci setup se jako první vykoná funkce setUpPinModes, která nastaví piny jako výstupy (16,17,18,19) a piny 12 a 14 jako signály PWM, podle daných parametrů.

Pro stabilizaci se spustí inicializace I2C spojení s akcelerometrem.

```
void initMPU(){
  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    stabilizace = 1;
  }
  else {Serial.println("MPU6050 Found!");}
  stabilizace = 0;
}
```

[5]

Následuje spuštění ESP32 jako Wi-Fi sítě, podle parametrů. Následně se zapíše do webového prohlížeče IP adresa pro připojení. Poté se spustí server na bázi http protokolu a nastaví správné cesty pro posílání packetu – když se správně nastaví cesta odesílá se číslo 200 a na uživatelově zařízení se otevře http server podle definice, která je popsána níže, V případě špatné nastavené trasy se zobrazí chyba 404 (File Not Found, neboli stránka nenalezena).

```
WiFi.softAP(apSSID, apPassword);
IPAddress IP = WiFi.softAPIP();

server.on("/", http_GET, handleRoot);
server.onNotFound(handleNotFound);
```

```
ws.onEvent (onWebSocketEvent) ;

server.addHandler (&ws) ;
server.begin () ;
```

[6]

Následuje spuštění websocketu společně s funkcí pro dekódování websocket packetu, která se následně implementuje do již spuštěného http serveru. Server se spustí funkcí server.begin().

Poté se nastaví servomotory na piny, které jsou napsány v závorkách.

Jako poslední se spustí vlákno StabilizaceDELO, která obsahuje stabilizaci děla, a velikost zásobníku je nastavena na 40000 bajtů. Toto vlákno je pro lepší optimalizaci výkonu spuštěna na jádru 1.

### 3.2.3 Funkce pro dekódování websocket packetu

```
void onWebSocketEvent (AsyncWebSocket *server, AsyncWebSocketClient
*client, AwsEventType type, void *arg, uint8_t *data, size_t len)
{switch (type){
case WS_EVT_CONNECT:
break;

case WS_EVT_DISCONNECT:
processCarMovement ("0");
break;

case WS_EVT_DATA:
AwsFrameInfo *info;
info = (AwsFrameInfo*)arg;
if (info->final && info->index == 0 && info->len == len && info->opcode ==
WS_TEXT){
std::string myData = "";
myData.assign ((char *)data, len);
processCarMovement (myData.c_str());
}
break;
default:
break;
}}
}
```

[6]

V této funkci probíhá dekódování zpráv, které poslal uživatel. WebSocket dokáže na základě serveru poslat tři typy zpráv: připojení a odpojení k ESP32 a poslání dat. Při připojení se v mém kódu nic nestane, v případě odpojení se robot uvede do stavu STOP. Při poslání dat se nejdříve vytvoří proměnná AwsFrameInfo, která představuje packet a do ní se následně packet, který byl poslán zkopíruje. Následuje kontrola packetu na správnou délku, finální bit, indexový bit a že obsahuje text. Dále se v daném packetu odstraní již dříve zmíněné parametry a zobrazí se pouze textový string s obsahem zprávy ze serveru. Ten se následně použije ve funkci processCarMovement.

### 3.2.4 Funkce pro řízení podvozku, věže a speciální předdefinované manévry

```
void processCarMovement(String inputValue) {  
  switch(inputValue.toInt()) {  
  case UP:  
    prosba = 6;  
    break;
```

V této části kódu se rozhoduje, jaký pohyb se má vykonat. V první řadě se obsah zprávy ze serveru převede na integer pro snadnější manipulaci a následně do pomocné konstanty `prosba`. Tato konstanta poté slouží pro nastavení pohybu ve smyčce `loop`. Tato metoda byla zvolena pro rychlejší a robustnější komunikaci. V případě, že celý pohyb byl dán do funkce `processCarMovement` docházelo k výpadkům a přetížení ESP32. V případě pohybu podvozku se nastaví dané piny a PWM signál a podle toho se vykoná daný pohyb.

```
void loop() {  
  switch(prosba) {  
  case 6:  
    digitalWrite(16, HIGH);  
    digitalWrite(17, LOW);  
    digitalWrite(18, HIGH);  
    digitalWrite(19, LOW);  
    ledcWrite(PWMchannel1, 140);  
    ledcWrite(PWMchannel2, 140);  
    break;
```

Následuje případ střelby, kdy se nastaví kontinuální servomotor do správného směru střelby, tak aby se vystřelil „projektil“.

Pro ovládání věže se nastaví proměnná `prosba` do hodnoty 1 či 2 a následně v `loopu` se přičítá a odečítá 1 stupeň do maximálních 180 nebo 0 stupňů. Následně se dané hodnoty zapíše do servopohonu.

Pro ovládání děla se nastaví proměnná `prosba` do hodnoty 3 či 4 a následně v `loopu` se přičítá a odečítá 1 stupeň do maximálních 110 nebo 40 stupňů. Následně se dané hodnoty zapíše do funkce stabilizátoru.

Poté jsou v ovládání nastavené předdefinované manévry. Jedná se o otočku o 180 stupňů či jízdu se střelbou.

Jako poslední je možnost STOP či default, která zastaví celý pohyb tanku i děla.

### 3.2.5 Popis serveru

Celý server běží na HTTP protokolu a díky přidaným knihovnám do programu stačí danou část převést ve stejné formě jako pro tvorbu webových stránek. Výroba vlastního serveru podle daných motivů je tak daleko jednodušší.

```
<tr>
<td
ontouchstart='onTouchStartAndEnd("14") '
ontouchend='onTouchStartAndEnd("0") '><span class="sipky" style="transform:
rotate(135deg); display: inline-block;">&#8681;</span></td>

<td                                ontouchstart='onTouchStartAndEnd("1") '
ontouchend='onTouchStartAndEnd("0") '><span
class="sipky">&#8679;</span></td>

<td                                ontouchstart='onTouchStartAndEnd("13") '
ontouchend='onTouchStartAndEnd("0") '><span class="sipky" style="transform:
rotate(45deg); display: inline-block;">&#8679;</span></td>
</tr>
```

V této části se odehrává hlavní úkol serveru. Vytvoření specifických tlačítek, které při stisku provedou funkci `onTouchStartAndEnd`, která přidá do obsahuje packetu číslo, které se následně dekóduje v mikroprocesoru. Po uvolnění tlačítka se posílá 0, což znamená STOP. Následuje vizuální zobrazení daných tlačítek, které se definují jako `sipky` a do nich se vnoří daný symbol podle http logiky.

```
var websocketUrl = "ws:///" + window.location.hostname + "/ws";
var websocket;
function initWebSocket()

{
websocket = new WebSocket(websocketUrl);
websocket.onopen = function(event){};
websocket.onclose = function(event){setTimeout(initWebSocket, 2000)};
websocket.onmessage = function(event){};
}

function onTouchStartAndEnd(value)
{
websocket.send(value);
}
```

[6]

Tato část obsahuje funkci pro posílání daných websocket packetu pro ESP32. Jedná se o jednoduchou funkci, která pouze danou hodnotu zabalí do packetu a pošle do mikroprocesoru.

### 3.2.6 Stabilizace děla

```
void StabilizaceDELO( void * parameter) {
while(1) {
desired_angle = val-90;
jizda = prosba;

ukol = abs(jizda - zapamat);
if((ukol>=6) || (konstanta >=1)) {
kp=0.1;
ki=0.00001;
kd=0;

konstanta++;
if(konstanta >= 6){
konstanta = 0;}}
else if (jizda>=6){
kp=0.25;
ki=0.00005;
kd=0;
}
else {
kp=0.75;
ki=0.0001;
kd=0;
}

getAccReadings();
getGyroReadings();

Acceleration_angle = atan(1*(AccX)/sqrt(pow((AccY),2) +
pow((AccZ),2)))*rad_to_deg; //Y

error = Acceleration_angle - desired_angle;

pid_p = kp*error;

if((-2 >error) || (error > 2))
{
pid_i = pid_i+(ki*error);
}

pid_d = kd*(error - previous_error);

PID = pid_p + pid_i + pid_d;

previous_error = error;

if(jizda == 5){
if((-4.0<PID)&&(PID<4.0)){ // - šum při střelbě
PID = 0; }}

if((-1.5<PID)&&(PID<1.5)){ // - Šum
PID = 0;
}

rozdil = rozdil - PID;
if(rozdil < 70){
rozdil = 70;
}
}
```

```

if(rozdil > 150){
rozdil = 150;
}

rozdil2 = map(rozdil, 0, 180, 180, 0);
if(stabilizace == 1){
  val2 = map(val, 0, 180, 180, 0);
  myservo.write(val);
  myservol.write(val2);
}
else {
myservo.write(rozdil);
myservol.write(rozdil2);
}
delay (250);
zapamat = jizda;}}

```

Tento kód obsahuje nejdříve převedení úhlu pro změnu rozlišení z 0 až 180 na -90 až 90, ale zejména zjištění požadovaného úhlu stabilizace. Poté se podle daného pohybu nastaví PID konstanty pro následný výpočet. Tyto hodnoty jsou zejména sníženy pro rozjezd a zastavení tanku, ty totiž představují masivní zakmitnutí tanku. Následuje funkce pro získání hodnot z akcelerometru a gyroskopu, která pouze naslouchá, co přijde po I2C sběrnici. Poté se vypočítá úhel v ose Y a ten nastaví finální úhel pro PID regulátor. Po regulaci se výstup z regulátor odečte od hodnoty rozdíl (PWM servomotoru) a to zejména protože je MPU6050 otočeno o 180 stupňů. Pro lepší stabilizaci je zařazen i filtr pro od regulaci malých hodnot, vzniklých šumem a zvlněného signálu způsobeného vibrací při střelbě. Následuje ochrana vůči zaseknutí o podvozek nebo horní část věže. Celý tento proces proběhne po každých 250ms, a to zejména kvůli optimalizaci procesu a v případě posunu děla nedochází k nežádoucím překmitům či zásekům.

Celá verze kódu je k dispozici v příloze bakalářské práce.

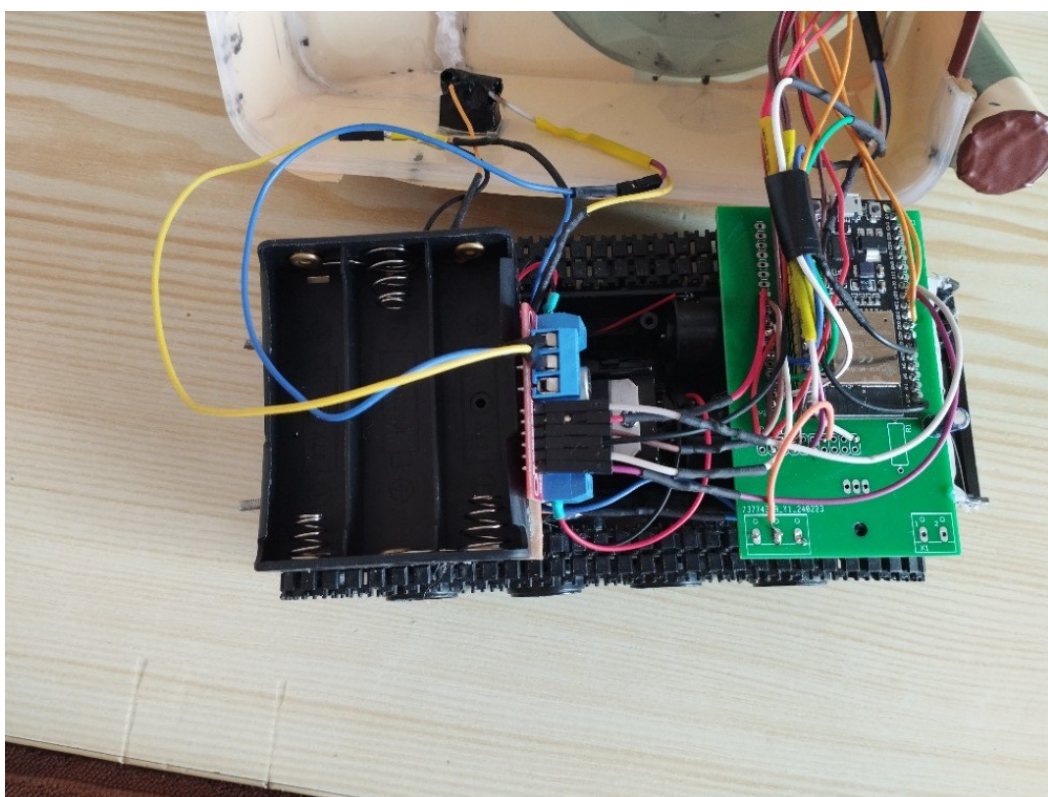
## 4 Finální zpracování

Na obrázcích (Obrázek 15,16) lze vidět finální produkt. Celá karosérie je vyrobena z plastu a osazena na podvozek. Tato karosérie je odnímatelná, je tedy možné otevřít tank a např. vyměnit baterie či upravit součástky. Na model jsem dodal barevnou kamufláž a jiné předměty specifické pro tanky. V horní části se nachází věž s dělem, opět vyrobená z plastu a z břízových špachtlí. Umístění součástek a plošného spoje bylo primárně zohledněním váhy a následného zatížení tanku.



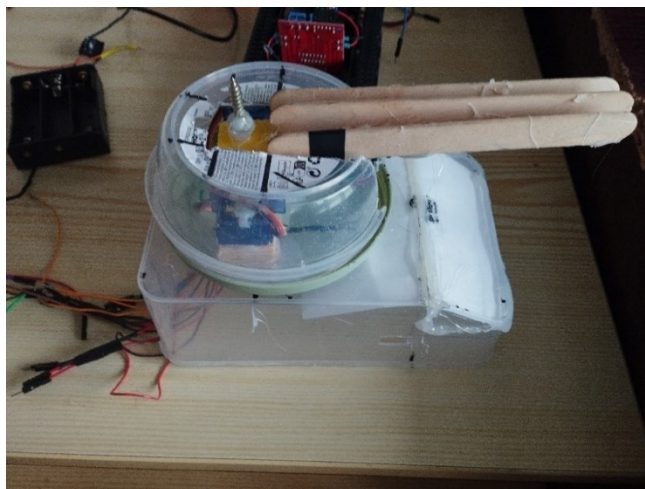


Obrázek 15 - finální tank

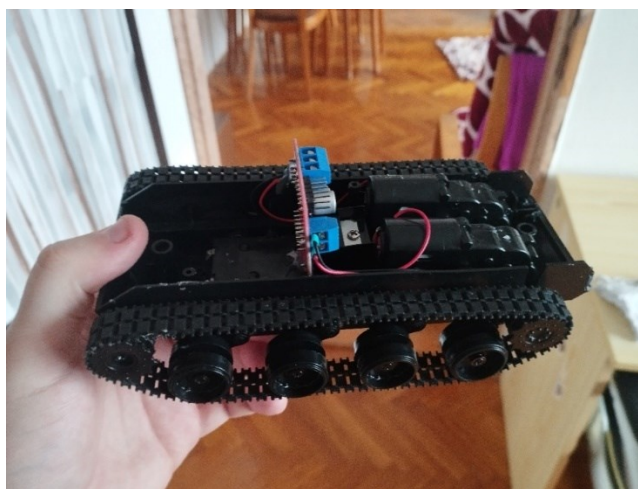


Obrázek 16 - vnitřek tanku

Na obrázcích (Obrázek 17,18) příkládám první prototyp korby s věží a základní rozložení podvozku před montáží zdroje.

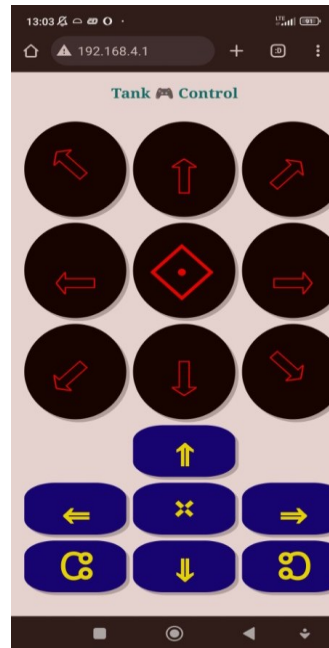


**Obrázek 17 - První prototyp**



**Obrázek 18 - základní podvozek**

Pro dělo jsem zvolil funkci praku. Servomotor slouží pro uvolnění gumičky, která slouží jako projektil, a na něm je umístěn šroub se speciálně vytvarovanými závity pro správné uvolnění. Na konci děla jsou vytvořené úchyty pro podržení gumičky. Dělo je drženo na dvou servomotorech. Uprostřed se nachází gyroskop, který je v této poloze nejvíce stabilní a dochází k nejmenšímu zkreslení. Věž se ovládá pomocí dalšího servomotoru a je přichycena pomocí dodaných šroubů a speciálních hřídelí.



Obrázek 19 - ovládací panel

Výsledné ovládání je na následujícím obrázku (Obrázek 19), kde jsou i dané symboly, které označují dané pohyby. Tyto symboly byly vybírány pro rychlou orientaci v ovládání.

Celková doba provozu tanku je zhruba kolem 10 minut. Tato hodnota se může lišit vlivem rozdílného zatížení. K tomu může dojít například třením mezi pásy RC tanku a povrchem, přes který se pohybuje.

#### 4.1 Problémy a poruchy

Během tvorby tohoto robota se vyskytla řada problémů. Mezi ty nejzávažnější patří napájení, které bylo voleno malé a slabé, to následně způsobovalo výpadky ESP32 a snižování účinnosti motorů. Jako problém se jeví i stabilizace, která trpí velkým bílým šumem, a proto dochází k občasným, nechtěným posunům děla nejčastěji během jízdy. Součástky a komponenty byli objednávány ze zahraničí a za menší finanční obnos a jejich následná kvalita byla tedy nižší. To se zejména projevilo na podvozku, který měl mnoho chyb.

#### 4.2 Doporučení a možná vylepšení

Vylepšení by bylo vhodné v oblasti napájení. Snižování spotřeby napájení, volba výkonnějších baterií včetně konstrukce lepší ochrany baterií. V úvahu připadá i výměna akumulátorového zdroje za nový.

Model lze také vylepšit pomocí věže, která se otáčí o 360 stupňů. Pro toto je potřeba speciální osazení věže, tzv. turret ring, který dokáže odstínit kabely z věže a z podvozků, tak aby nedošlo k jejich zaseknutí a poničení.

Možné vylepšení je i v rozložení váhy, tak aby se tank nekmítal při rozjezdech.

## **Závěr**

Cílem bakalářské práce bylo seznámení čtenáře s možností tvorby vlastního RC modelu pro jednoduché použití a řízení.

V teoretické části byly popsány možnosti a různé druhy prvků pro tvorbu a komunikaci robotů s uživatelem. Zde byly popsány zejména motory, typy podvozků a jejich řízení či sledování polohy. Následovalo popsání komunikace mezi uživatelem a mikrokontrolerem.

V praktické části se nejdříve pojednávalo o použitých součástkách a z jakého důvodu byly vybrány. Navíc byl popsán i programovací jazyk Arduino IDE, ve kterém lze v dnešní době naprogramovat spoustu úloh pro arduina. Následovalo rozebrání kódu a popis jednotlivých částí programu.

Ve finální části byl představen finální výrobek a výsledná verze webového serveru pro ovládání. Následně byly rozebrány i problémy a možné vylepšení daného modelu.

## Literatura

- [1] VÁŇA, Vladimír. *Mikrokontroléry Atmel AVR: popis procesoru a instrukční soubor*. Praha: BEN - technická literatura, 2003. ISBN 80-7300-083-0.
- [2] NOVÁK, Petr. *Mobilní roboty: pohony, senzory, řízení*. Robotika, 1. Praha: BEN - technická literatura, 2005. ISBN 80-7300-141-1.
- [3] BRTNÍK, Bohumil. *Základní elektronické obvody*. Praha: BEN - technická literatura, 2011. ISBN 978-80-7300-408-8.
- [4] VÁŇA, Vladimír. *Mikrokontroléry Atmel AVR: assembler*. Praha: BEN - technická literatura, 2003. ISBN 80-7300-093-8.
- [5] SANTOS, Rui. *ESP32 Web Server with MPU-6050 Accelerometer and Gyroscope*. Online. Random Nerd Tutorial. 2021. Dostupné z: <https://randomnerdtutorials.com/esp32-mpu-6050-web-server/>. [cit. 2024-03-12].
- [6] *Smartphone controlled WiFi car | esp32 | Mecanum wheels*. Online. 2021. Dostupné z: [https://www.youtube.com/watch?v=l7Ty6SZdKPI&list=PLZ3wI3cWyvm5USFtbS8Egius9jv9gMPGU&index=12&t=148s&ab\\_channel=hashincludeelectronics](https://www.youtube.com/watch?v=l7Ty6SZdKPI&list=PLZ3wI3cWyvm5USFtbS8Egius9jv9gMPGU&index=12&t=148s&ab_channel=hashincludeelectronics). [cit. 2024-03-12].
- [7] *PID brushless motor control tutorial*. Online. 2017. Dostupné z: [https://www.youtube.com/watch?v=AN3yxIBAxTA&t=210s&ab\\_channel=Electronoobs](https://www.youtube.com/watch?v=AN3yxIBAxTA&t=210s&ab_channel=Electronoobs). [cit. 2024-03-12].
- [8] *WiFi Robot Tank | ESP32 + Smartphone Controlled*. Online. 2022. Dostupné z: [https://www.youtube.com/watch?v=XCp0qFipG2o&t=617s&ab\\_channel=hashincludeelectronics](https://www.youtube.com/watch?v=XCp0qFipG2o&t=617s&ab_channel=hashincludeelectronics). [cit. 2024-03-12].
- [9] *Pásové vozidlo*. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2001-. Dostupné z: [https://cs.wikipedia.org/wiki/P%C3%A1sov%C3%A9\\_vozidlo](https://cs.wikipedia.org/wiki/P%C3%A1sov%C3%A9_vozidlo). [cit. 2024-03-12].

- [10] KULDA, Jiří. *WiFi Access Point vs WiFi Router. Jaký je v tom rozdíl?* Online. In: Dostupné z: <https://wifihome.cz/wifi-access-point-vs-wifi-router-jaky-je-v-tom-rozdil/>. [cit. 2024-03-18].
- [11] EUFACTORY, A.S. *KROKOVÝ MOTOR - CO TO JE A JAK FUNGUJE*. Online. EUFACTORY eshop. Dostupné z: [https://shop.eufactory.com/blog/12\\_krokovy-motor-uvod](https://shop.eufactory.com/blog/12_krokovy-motor-uvod). [cit. 2024-03-18].
- [12] RNDR. DAVID OBDRŽÁLEK, PH.D. | MFF UK, JIŘÍ ROTTA | JEDE ROBOT S. R. O. *MECHATRONIKA, ČÁST 2 STEJNOSMĚRNÉ A KROKOVÉ MOTORY. MODELÁŘSKÉ SERVOMECHANISMY*. Online. In: . Dostupné z: <https://www.dps-az.cz/clanky/id:84895/z-aktualniho-vydani-mechatronika-cast-2-stejnosmerne-a-krokovy-motory-modelarske-servomechanismy>. [cit. 2024-03-18].
- [13] *DC motorek 130 3V 16500 rpm*. Online. In: LaskaKit. Dostupné z: <https://www.laskakit.cz/dc-motorek-130-3v-16500-rpm/>. [cit. 2024-03-18].
- [14] *Robotický podvozek pásový plastový*. Online. In: LaskaKit. Dostupné z: <https://www.laskakit.cz/roboticky-podvozek-pasovy-plastovy/>. [cit. 2024-03-18].
- [15] DOBROVOLNÝ, Martin. *03 Aplikace mikroprocesorů 2: BAM12*. PDF. 2021.
- [16] DOBROVOLNÝ, Martin. *06 Aplikace mikroprocesorů 2 - Vlákna: BAM12*. PDF. 2021.
- [17] DOBROVOLNÝ, Martin. *07 Aplikace mikroprocesorů 2 - WiFi: BAM12*. PDF. 2021.
- [18] NĚMEC, Zdeněk. *Blok 4 – Standard 802.11: BKOSY – Komunikační systémy*. PDF.

- [19] NĚMEC, Zdeněk. *Blok 5 – Standard 802.15.1: BKOSY*. PDF.
- [20] NĚMEC, Zdeněk. *Blok 7 – 3G a vyšší generace mobilních systémů: BKOSY*. PDF.
- [21] KUPKA, Libor. *Mechatronika a robotika: Téma 5: Mapy prostředí a plánování dráhy*. PDF. Univerzita Pardubice.
- [22] KUPKA, Libor. *Mechatronika a robotika: Téma 8: Akční členy I*. PDF. Univerzita Pardubice.
- [23] KUPKA, Libor. *Mechatronika a robotika: Téma 7: Senzory II*. PDF. Univerzita Pardubice.
- [24] TESLA, Redakce. *Technologie ZigBee. Co to je, jaké má výhody a jak s ní začít?* Online. Chytrá domácnost TESLA. 2023. Dostupné z: <https://www.teslasmart.com/cz/technologie-zigbee-co-to-je-jake-ma-vyhody-a-jak-s-ni-zacit>. [cit. 2024-04-18].
- [25] WINKLER, Zbyněk. *Odometrie*. Online. 2005. Dostupné z: <https://robotika.cz/guide/odometry/en>. [cit. 2024-03-19].
- [26] KUPKA, Libor. *Mechatronika a robotika: Téma 3: Akční členy III*. PDF. Univerzita Pardubice.
- [27] BOTLAND.CZ. *Lineární aktuátory*. Online. Dostupné z: <https://botland.cz/585-linearni-aktuatory>. [cit. 2024-04-18].
- [28] *ELEKTRICKÉ POHONY V ROBOTICE*. Bakalářská práce. Ústav automobilního a dopravního inženýrství, Brno: VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ, 2018.

- [29] *Návrh, modelování a řízení všesměrového mobilního robota*. Bakalářská práce. Katedra kybernetiky, Plzeň: Západočeská univerzita v Plzni, 2016.
- [30] MENSURO S.R.O. *Co je LiDAR a jak funguje?* Online. Technologie mobilního laserového skenování GeoSLAM. Laserové skenery a software pro zpracování mračen bodů. 2020. Dostupné z: <https://www.geosken.cz/co-je-lidar-a-jak-funguje/>. [cit. 2024-04-18].
- [31] *LoRaWAN*. Online. The Things Network. Dostupné z: <https://www.thethingsnetwork.org/docs/lorawan/>. [cit. 2024-04-18].
- [32] ČERMÁK, Jiří. *Synchronní stroje – konstrukce, princip a použití*. Online. OEnergetice.cz. 2015. Dostupné z: <https://oenergetice.cz/elektrina/synchronni-stroje-konstrukce-princip-a-pouziti>. [cit. 2024-04-18].
- [33] *Podvozek (železnice)*. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2001-, 1. 7. 2023. Dostupné z: [https://cs.wikipedia.org/wiki/Podvozek\\_\(%C5%BEeleznice\)](https://cs.wikipedia.org/wiki/Podvozek_(%C5%BEeleznice)). [cit. 2024-04-18].



## **Příloha A – Program pro řízení robota**

V příloze je poskytnuta celá verze programu.

## **Příloha B – Plošný spoj pro zapojení ESP32**

V příloze je poskytnuto schéma plošného spoje pro zapojení esp32 a periférií.

Změna nastala zejména v zapojení tranzistoru a následného motoru. Ten se ve finální části vyměnil za servomotor.