

**UNIVERZITA PARDUBICE**  
Fakulta elektrotechniky a informatiky

**AUTONOMNÍ ROBOTICKÉ VOZIDLO**

Bc. Jan Knotek

Diplomová práce  
2021

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2020/2021

## **ZADÁNÍ DIPLOMOVÉ PRÁCE** (projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Jan Knotek**  
Osobní číslo: **I19295**  
Studijní program: **N0714A150005 Automatické řízení**  
Studijní obor: **Automatické řízení**  
Téma práce: **Autonomní robotické vozidlo**  
Zadávací katedra: **Katedra řízení procesů**

### Zásady pro vypracování

Cílem práce je návrh a realizace mobilního robotického vozidla. Pro orientaci vozidla v neznámém prostoru bude použita elektronická navigační jednotka vlastní konstrukce, jejíž návrh a realizace bude jedním z cílů řešení diplomové práce. Jako řídicí jednotka robota bude použita komerčně dostupná vývojová platforma s jednočipovým mikropočítačem. Pohyb robotického vozidla bude realizován čtyřmi, nezávisle řízenými stejnosměrnými motory. Řídicí software robotického vozidla bude využívat vybraný algoritmus umělé inteligence. Nedílnou součástí práce bude rešerše na dané téma, kompletní výrobní dokumentace zařízení a zdrojové kódy použitého software.

Rozsah pracovní zprávy: **60**  
Rozsah grafických prací:  
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

MATOUŠEK, D., Práce s mikrokontroléry ATMEL AVR-3.díl, edice uP a praxe, 2. vydání, BEN – technická literatura, 2006, ISBN 80-7300-209-4.  
EVERETT, H., R. Sensors for Mobile Robots, Theory and Applications. 1.vyd. A.K.Peters, Ltd. ISBN 1-568811-048-2.  
NOVÁK, P. Mobilní roboty – pohony, senzory, řízení, BEN-technická literatura, Praha 2007, ISBN 80-7300-141-1.

Vedoucí diplomové práce: **Ing. Libor Havlíček, Ph.D.**  
Katedra řízení procesů

Datum zadání diplomové práce: **6. listopadu 2020**  
Termín odevzdání diplomové práce: **21. května 2021**

L.S.

---

**Ing. Zdeněk Němec, Ph.D.**  
děkan

---

**Ing. Daniel Honc, Ph.D.**  
vedoucí katedry

V Pardubicích dne 16. listopadu 2020

## **Prohlášení**

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

Bc. Jan Knotek

## **Poděkování**

Rád bych poděkoval vedoucímu mé diplomové práce panu Ing. Liboru Havlíčkovi, Ph.D. za vedení práce, cenné rady a připomínky. Také bych rád poděkoval mé rodině za vstřícnost a podporu při realizaci diplomové práce.

V Pardubicích dne

Bc. Jan Knotek

## **ANOTACE**

*Práce se zabývá návrhem a realizací mobilního robotického vozidla. Pro orientaci vozidla v neznámém prostoru bude použita elektronická navigační jednotka vlastní konstrukce, jejíž návrh a realizace bude jedním z cílů řešení diplomové práce. Jako řídicí jednotka robota bude použita komerčně dostupná vývojová platforma s jednočipovým mikro počítačem. Pohyb robotického vozidla bude realizován čtyřmi, nezávisle řízenými krokovými motory. Řídicí software robotického vozidla bude využívat vybraný algoritmus umělé inteligence.*

## **KLÍČOVÁ SLOVA**

*Autonomní robotické vozidlo, ESP32, krokový motor, lidar, umělá inteligence*

## **TITLE**

*AUTONOMOUS ROBOTIC VEHICLE*

## **ANNOTATION**

*This thesis deals with design and implementation of a mobile robotic vehicle. Custom made electronic navigation unit will be used for vehicle orientation in unknown space. Design and realization of said navigation unit will be one of the goals of this thesis. Commercially available development platform with a single-chip microcomputer will function as a control unit. The movement of the robotic vehicle will be realized by four, independently controlled stepper motors. The control software of the robotic vehicle will be using a selected algorithm of artificial intelligence.*

## **KEYWORDS**

*Autonomous robotic vehicle, ESP32, stepper motor, lidar, artificial intelligence*

## OBSAH

|   |    |
|---|----|
| Seznam zkratk a značek .....  | 10 |
| Seznam symbolů proměnných a funkcí .....                                  | 11 |
| Seznam ilustrací .....  | 12 |
| Seznam tabulek .....  | 14 |
| ÚVOD .....  | 15 |
| 1 TEORETICKÁ ČÁST .....   | 16 |
| 1.1 ÚVOD DO PROBLEMATIKY ROBOTICKÝCH VOZIDEL.....                         | 16 |
| 1.1.1 Definice autonomního robotického vozidla.....                       | 16 |
| 1.1.2 Stupně autonomie vozidel.....                                       | 16 |
| 1.2 KROKOVÝ MOTOR .....   | 17 |
| 1.2.1 Výhody a nevýhody krokových motorů.....                             | 18 |
| 1.2.2 Druhy a možnosti zapojení vinutí krokových motorů .....             | 18 |
| 1.2.3 Řízení krokových motorů.....  | 19 |
| 1.3 LIDAR .....   | 20 |
| 1.4 VÝBĚR KOMPONENT .....   | 21 |
| 1.4.1 ESP32.....  | 22 |
| 1.4.2 Rozšiřující I/O modul pro ESP32 MCP23017.....                       | 23 |
| 1.4.3 Krokový motor NEMA17 (17HS4401) .....                               | 24 |
| 1.4.4 Řadič pro krokové motory TMC2209 – V1.2.....                        | 24 |
| 1.4.5 Senzor vzdálenosti TF-Luna .....                                    | 25 |
| 1.4.6 Hallův senzor .....   | 25 |
| 1.4.7 Senzor vzdálenosti VL53L0X.....                                     | 26 |
| 1.4.8 Gyroskopický modul BNO055 .....                                     | 26 |
| 1.4.9 Měřič proudu INA226.....  | 27 |
| 1.4.10 Nabíjecí a ochranný modul se zvyšujícím stejnosměrným měničem..... | 27 |
| 1.4.11 Nabíjecí a ochranný modul HW-293 .....                             | 28 |
| 1.4.12 Stejnosměrný snižující měnič LM2596 .....                          | 28 |
| 1.4.13 OLED displej SSH1106.....  | 28 |
| 1.4.14 Li-ionová baterie MH12210 .....                                    | 29 |
| 1.4.15 Stejnosměrný externí napájecí adaptér .....                        | 29 |
| 1.5 DISKUZE NAD VÝBĚREM KOMPONENT .....                                   | 30 |
| 1.5.1 Vývojová deska.....   | 30 |

|       |  |    |
|-------|--|----|
| 1.5.2 | Motory.....                                  | 30 |
| 1.5.3 | Řadič pro krokové motory .....               | 31 |
| 1.5.4 | Senzor vzdálenosti .....                     | 32 |
| 1.5.5 | Napájení .....                               | 32 |
| 1.5.6 | Měření proudu.....                           | 33 |
| 1.6   | PROGRAMOVÁNÍ DESEK ESP32 .....               | 33 |
| 1.6.1 | Arduino .....                                | 34 |
| 1.7   | MATLAB.....                                  | 35 |
| 1.7.1 | Porovnání nástrojů GUIDE a AppDesigner.....  | 36 |
| 1.8   | UMĚLÁ INTELIGENCE .....                      | 37 |
| 1.8.1 | Úvod do problematiky umělé inteligence ..... | 37 |
| 1.8.2 | Stavový prostor a jeho prohledávání.....     | 38 |
| 2     | PRAKTICKÁ ČÁST.....                          | 42 |
| 2.1   | NÁVRH ELEKTRONIKY VOZIDLA .....              | 42 |
| 2.1.1 | Napájecí část .....                          | 43 |
| 2.1.2 | Část řídicí elektroniky vozidla .....        | 45 |
| 2.1.3 | Návrh DPS .....                              | 46 |
| 2.1.4 | Návrh lidaru .....                           | 53 |
| 2.1.5 | HW nastavení řadičů krokových motorů .....   | 54 |
| 2.1.6 | Práce s řadiči krokových motorů TMC2209..... | 55 |
| 2.2   | NÁVRH SOFTWARE VOZIDLA .....                 | 57 |
| 2.2.1 | Popis programového kódu vozidla.....         | 57 |
| 2.2.2 | Hledání nejkratší cesty vozidla k cíli ..... | 61 |
| 2.2.3 | GUI aplikace .....                           | 65 |
| 2.3   | PRŮBĚH JÍZDY VOZIDLA .....                   | 67 |
| 2.3.1 | Manuální režim .....                         | 67 |
| 2.3.2 | Autonomní režim .....                        | 67 |
| 2.3.3 | Vizualizace dat o vozidle .....              | 69 |
| 2.4   | VZHLED A PARAMETRY VOZIDLA .....             | 72 |
| 2.4.1 | Vzhled vozidla .....                         | 72 |
| 2.4.2 | Parametry vozidla .....                      | 73 |
| 2.5   | TESTOVÁNÍ AUTONOMNÍ JÍZDY VOZIDLA.....       | 75 |
| 3     | ZHODNOCENÍ .....                             | 78 |
| 4     | ZÁVĚR .....                                  | 79 |



|                          |    |
|--------------------------|----|
| POUŽITÁ LITERATURA ..... | 80 |
| PŘÍLOHY .....            | 84 |

## SEZNAM ZKRATEK A ZNAČEK

|                  |   |
|------------------|---|
| A/D              | Analogově-digitální                             |
| BLE              | Bluetooth Low Energy                            |
| CPU              | Central processing unit                         |
| DHCP             | Dynamic Host Configuration Protocol             |
| DIY              | Do it yourself                                  |
| DPS              | Deska plošných spojů                            |
| EN               | Enable  |
| GPIO             | General-purpose input/output                    |
| GPS              | Global Positioning System                       |
| GPU              | Graphics processing unit                        |
| GUI              | Graphic User Interface                          |
| IoT              | Internet of Things                              |
| I/O              | Input/output                                    |
| I <sup>2</sup> C | Inter-Integrated Circuit                        |
| IDE              | Integrated Development Environment              |
| LTE              | Long Term Evolution                             |
| OLED             | Organic light-emitting diode                    |
| PID              | Proporcionálně integračně derivační (regulátor) |
| SPI              | Serial Peripheral Interface                     |
| SRAM             | Static Random Access Memory                     |
| ToF              | Time-of-Flight                                  |
| TCP/IP           | Transmission Control Protocol/Internet Protocol |
| UART             | Universal Asynchronous Receiver and Transmitter |
| UI               | Umělá inteligence                               |

## SEZNAM SYMBOLŮ PROMĚNNÝCH A FUNKCÍ

|             |  |
|-------------|--|
| $f(v_i)$    | hodnotící funkce   |
| $G$         | množina cílových stavů   |
| $g$         | prvek množiny cílových stavů   |
| $g(v_i)$    | cena cesty z počátečního vrcholu do vrcholu $v_i$                        |
| $GR$        | graf   |
| $h(v_i)$    | odhad ceny cesty z vrcholu $v_i$ do nejbližšího cílového vrcholu (stavu) |
| $I_{RMS}$   | střední kvadratický proud, A   |
| $R_{SENSE}$ | snímací odpor, $\Omega$ ,  |
| $S$         | množina všech možných stavů úlohy  |
| $s_0$       | počáteční stav úlohy   |
| $s_i$       | i-tý stav  |
| $s_k$       | k-tý stav  |
| $S_p$       | stavový prostor  |
| $v_i$       | i-tý vrchol  |
| $V_{VREF}$  | referenční napětí, V   |
| $\phi$      | množina operátorů pro přechod mezi jednotlivými stavy                    |
| $\varphi$   | operátor přechodu  |

## SEZNAM ILUSTRACÍ

|  |    |
|--|----|
| Obr. 1.1 – Způsoby zapojení fázových vinutí krokového motoru („Krokové motory”, 2021)                                      | 19 |
| Obr. 1.2 – Lidar (Kilián, 2018).....   | 21 |
| Obr. 1.3 – ESP32 .....   | 22 |
| Obr. 1.4 – Specifikace ESP32 (Stehlík, ©1998–2021).....  | 23 |
| Obr. 1.5 – Rozšiřující I/O modul MCP23017.....   | 23 |
| Obr. 1.6 – Krokový motor NEMA17.....   | 24 |
| Obr. 1.7 – Řadič krokových motorů TMC2209.....   | 24 |
| Obr. 1.8 – Senzor vzdálenosti TF Luna .....  | 25 |
| Obr. 1.9 – Hallův senzor .....   | 25 |
| Obr. 1.10 – Senzor vzdálenosti VL53L0X .....   | 26 |
| Obr. 1.11 – Gyroskopický modul BNO055.....   | 26 |
| Obr. 1.12 – Měřič proudu INA226 .....  | 27 |
| Obr. 1.13 – Nabíjecí a ochranný modul se stejnosměrným zvyšujícím měničem.....   | 27 |
| Obr. 1.14 – Nabíjecí a ochranný modul HW-293 .....   | 28 |
| Obr. 1.15 – Stejnosměrný snižující měnič LM2596.....   | 28 |
| Obr. 1.16 – OLED displej SSH1106.....  | 29 |
| Obr. 1.17 – Li-ionová baterie MH12210 .....  | 29 |
| Obr. 1.18 – Stejnosměrný externí napájecí adaptér .....  | 29 |
| Obr. 1.19 – Arduino IDE (Knotek, 2019).....  | 34 |
| Obr. 1.20 – Porovnání nástroje GUIDE a AppDesigner<br>(„GUIDE to App Designer Migration Tool for MATLAB”, ©1994-2021)..... | 36 |
| Obr. 2.1 – Blokové schéma elektroniky vozidla.....   | 43 |
| Obr. 2.2 – Schéma zapojení napájecí DPS .....  | 48 |
| Obr. 2.3 – Osazovací plán napájecí DPS .....   | 48 |
| Obr. 2.4 – Navržená napájecí DPS .....   | 49 |
| Obr. 2.5 – Navržená napájecí DPS s vylitým polygonem .....   | 49 |
| Obr. 2.6 – Schéma zapojení řídicí DPS .....  | 51 |
| Obr. 2.7 – Navržená řídicí DPS .....   | 52 |
| Obr. 2.8 – Osazovací plán řídicí DPS.....  | 52 |
| Obr. 2.9 – Navržená řídicí DPS s vylitým polygonem .....   | 53 |
| Obr. 2.10 – Blokové schéma lidaru .....  | 53 |
| Obr. 2.11 – Schéma pro nastavení proudového omezení řadiče TMC2209 .....   | 54 |

|   |    |
|---|----|
| Obr. 2.12 – Prohledávání otevřeného prostoru při využití Manhattanové metriky ..... | 62 |
| Obr. 2.13 – Prohledávání otevřeného prostoru při využití Euklidovské metriky .....  | 63 |
| Obr. 2.14 – Vývojový diagram A* algoritmu .....                                     | 64 |
| Obr. 2.15 – GUI aplikace pro ovládání mobilního vozidla .....                       | 66 |
| Obr. 2.16 – Diagram přechodů sekvenčního automatu autonomního pohybu vozidla.....   | 68 |
| Obr. 2.17 – Vizualizace dat na OLED displeji .....                                  | 70 |
| Obr. 2.18 – Vizualizace dat v GUI aplikaci.....                                     | 70 |
| Obr. 2.19 – Vizualizace dat po startu programu (sériový monitor).....               | 71 |
| Obr. 2.20 – Vizualizace dat po stisku tlačítka INFO (sériový monitor).....          | 72 |
| Obr. 2.21 – 3D model vozidla.....   | 72 |
| Obr. 2.22 – Zkonstruované vozidlo .....   | 73 |
| Obr. 2.23 – Závislost spotřeby elektrické energie na rychlosti vozidla.....         | 75 |
| Obr. 2.24 – Testování autonomní jízdy vozidla – počáteční stav.....                 | 76 |
| Obr. 2.25 – Testování autonomní jízdy vozidla – konečný stav.....                   | 77 |

## **SEZNAM TABULEK**

|  |    |
|--|----|
| Tab. 2.1 – Technické parametry vozidla.....                        | 74 |
| Tab. 2.2 – Záznam měření spotřeby elektrické energie vozidla ..... | 74 |

# ÚVOD

Autonomní vozidla jsou aktuálně velmi zajímavé téma pro různé odborníky. Řada společností se na vývoji těchto automobilů podílí. V tomto projektu se však jedná o autonomní robotické vozidlo, neboli vozidlo, které je vybavené (mikro)počítačovým systémem s množstvím snímačů. Obecně mezi ně patří například radar, lidar, GPS nebo třeba kamerové systémy schopné v reálném čase získávat data ze svého okolí (Cerman, 2019).

V této práci bude využito vlastní konstrukce lidarů. Pro zajištění autonomie vozidla bude využito jednoho z algoritmů umělé inteligence, konkrétně algoritmu A\*. Umělá inteligence je v dnešní době na vzestupu. Hovoří se o ní nejen ve vědeckém světě, nýbrž i inženýři v současné době hojně aplikují poznatky umělé inteligence do praxe.

Přestože se algoritmy umělé inteligence často tvoří na počítačích, v této práci se jedná o aplikaci algoritmů na mikroprocesor. Konkrétně se jedná o aplikace algoritmů prohledávání stavového prostoru na robotické vozidlo. V současné době lze nalézt mnoho simulací a programů, které umožňují snadno pochopit princip chování těchto algoritmů. Cílem práce je nalézt nejkratší cestu z bodu A do bodu B právě pomocí těchto algoritmů.

Kromě programové části práce je zkonstruováno vozidlo, které umožňuje pohyb v prostoru. K jeho „vidění“ je využit již zmíněný lidar, který mapuje v reálném čase okolí vozidla. Vozidlo dále obsahuje standardně čtyři kola, která jsou ovládána pomocí čtyř krokových motorů. Jako řídicí jednotka je využit modul ESP32. Jedná se o moderní vývojovou mikroprocesorovou desku s dostatečným množstvím paměti. Pro účely této práce jsou navrženy dvě DPS s konektory pro připojení všech potřebných komponent projektu. První DPS je tvořena převážně konektory pro řídicí elektroniku, tedy pro připojení řídicí jednotky, rozšiřující desky pro řídicí jednotku, řadičů pro krokové motory apod. Druhá deska je pak tvořena konektory pro napájení vozidla (nabíjecí a ochranné moduly, stejnosměrné měniče a měřice proudu).

# 1 TEORETICKÁ ČÁST

## 1.1 ÚVOD DO PROBLEMATIKY ROBOTICKÝCH VOZIDEL

V současné době existuje množství mobilních robotických vozidel. Existují profesionální řešení od firem, ale i množství tzv. DIY projektů. Tyto projekty často tvoří zájemci o elektrotechniku, programátoři, ale i obyčejní lidé se zájmem o tuto činnost. Zatímco firmy si své know-how většinou pečlivě střeží, poloprofesionální řešení od obyčejných lidí lze často nalézt na internetu (například formou různých návodů). Tato řešení pak slouží jako inspirace dalším lidem.

Mezi aktuálně exponované druhy autonomních robotických vozidel patří například:

- vozidla umožňující jízdu po čáře,
- vozidla vyhýbající se překážkám,
- dopravní vozítka (využití například ve výrobních halách),
- robotické vysavače,
- vozidla s manipulátorem apod.

Cenová relace pak závisí na požadované činnosti vozidla. Při tvorbě DIY projektů se konstruktéři pohybují v rámci tisícikorun (cena pouze za hardware). Záleží také, zda využívají originálních dílů nebo například různých klonů desek z Číny. Rozdíl v ceně takového vozidla pak může být znatelný. Pokud bude brán v potaz typ vozidla, který je součástí této práce, pak se cenová kalkulace bude pohybovat kolem 7 500 Kč. Jedná se však cenu za hardware nakoupený v Číně, tedy s množstvím různých klonů apod. Pokud by byly brány v potaz originální komponenty, pak by cena byla značně vyšší.

### 1.1.1 Definice autonomního robotického vozidla

Autonomní mobilní vozidlo je vozidlo, které je schopné se pohybovat samostatně. Obsahuje mikropočítačový systém a další senzory. Obecně se autonomie vozidla dle standardizační organizace Society of Automotive Engineers dělí do šesti stupňů (0 až 5) (Cerman, 2019).

### 1.1.2 Stupně autonomie vozidel

Stupeň 0 představuje vozidlo bez využití jakékoliv automatizace.

Stupeň 1 zahrnuje částečně automatizované funkce korigované řidičem. Elektronika může mírně zasahovat do činnosti řidiče, nicméně vozidlo může vykonávat takovouto činnost



pouze jednu naráz. Příkladem může být adaptivní tempomat, kdy systém na základě určitých informací ovládá podélný pohyb (rychlost) vozidla, zatímco řidič ovládá směr.

Stupeň 2 obsahuje funkce jako stupeň 1, nicméně lze tyto funkce kombinovat. Typickým příkladem může být systém automatického parkování. Člověk však musí být stále ve střehu a v případě potřeby zasáhnout. Systémy stupně 2 jsou podporované současnou legislativou, která obsahuje určité předpisy a zákonné rámce provozu na pozemních komunikacích.

Stupeň 3 umožňuje vozidlu autonomní jízdu za určitých podmínek. Existují různé jízdní scénáře, které vozidlo dokáže zvládnout. Mezi ně patří například autonomní jízda na dálnici nebo autonomní jízda v dopravní zácpě. Systém by tyto scénáře měl zvládnout, přesto se vyžaduje i činnost řidiče v případě selhání systému. Řidič tedy systém neřídí, nicméně musí být připraven zasáhnout (udává se do 10 s). Vozidlo s funkcemi stupně 3 proto stále obsahuje standardní ovládací prvky (volant, pedály apod).

Stupeň 4 představuje již plně automatizovaný systém, nicméně stále zůstává možnost vozidlo řídit ručně.

Stupeň 5 pak představuje systém koncipovaný již jako bezpilotní. Vozidlo je tedy schopno bezpečně zvládnout jakoukoliv dopravní situaci a neumožňuje jej řídit ručně (Cerman, 2019).

V rámci této práce autonomní robotické vozidlo odpovídá stupni autonomie 3. To znamená, že vozidlo je schopné realizovat určitý jízdní scénář. Stále však existuje možnost ručního ovládání v případě selhání systému. Myšleným scénářem je jízda z bodu A do bodu B nejkratší možnou cestou. Tuto jízdu zajišťuje algoritmus umělé inteligence aplikovaný v řídicí jednotce vozidla.

## **1.2 KROKOVÝ MOTOR**

Krokový motor je synchronní točivý stroj, který je napájen impulsy stejnosměrného proudu. Magnetické pole vzniká postupným napájením jednotlivých fází. Pohyb motoru je (obzvláště při nízkých rychlostech) nespojitý. Rotor motoru se pohybuje mezi stabilními polohami v určitém úhlu (krok motoru). Počet kroků je pak dán počtem pólových dvojic motoru, nicméně jej lze ovlivnit způsobem ovládání („Krokový motor“, 2021).

### 1.2.1 Výhody a nevýhody krokových motorů

Krokové motory jsou hojně používané, jejich uplatnění lze nalézt třeba v robotice, při přesném polohování. V této kapitole budou uvedeny jejich výhody a nevýhody.

#### Výhody krokových motorů:

- úhel natočení hřídele je přímo úměrný počtu vstupních impulsů,
- rychlost otáčení motoru je přímo úměrná frekvenci vstupních impulsů,
- řízení možné v otevřené smyčce bez nutnosti polohové zpětné vazby,
- přesná a rychlá reakce na zrychlení, zpomalení i na změnu směru otáčení,
- nekumulativní chyba polohování,
- velký točivý moment při nízkých rychlostech, není třeba využívat převodovky,
- velký statický moment,
- obousměrný provoz,
- aretace v zastaveném stavu motoru,
- motor lze přetížit i zastavit bez poškození,
- rotor je u velké části motorů uložen v přesných kuličkových ložiscích,
- vysoká životnost, robustnost.

#### Nevýhody krokových motorů:

- jsou nevhodné pro velké rychlosti otáčení,
- při určitých rychlostech může dojít k rezonancím,
- při ztrátě kroku je ztracena poloha poháněného stroje.

Vzhledem k potřebě přesného polohování a převažujícím výhodám, jsou pro tuto práci zvoleny právě krokové motory („Krokové motory”, 2021).

### 1.2.2 Druhy a možnosti zapojení vinutí krokových motorů

Krokových motorů existuje více druhů, stejně tak jako způsobů zapojení vinutí. Mezi druhy krokových motorů se řadí následující:

- motor s proměnnou reluktancí,
- motor s rotorem tvořeným permanentním magnetem,
- hybridní motor.

Nejstarší z uvedených druhů krokových motorů je motor s proměnnou reluktancí, který se již prakticky nevyužívá. Motor s rotorem tvořeným permanentním magnetem již své uplatnění nalezne, a to zejména ve spotřební a automobilové technice. Hybridní motory jsou

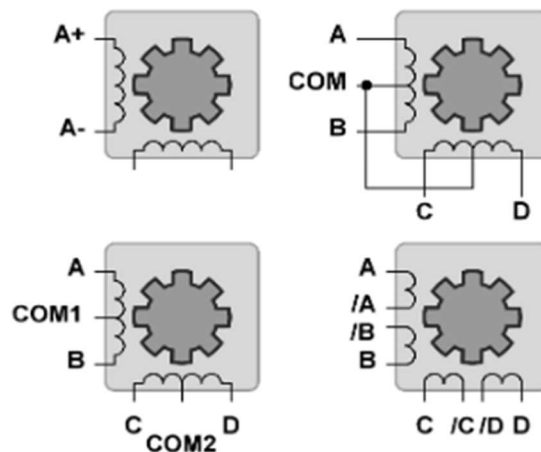
pak téměř výhradně zastoupeny v průmyslových aplikacích. Motory této konstrukce vykazují nejlepší dynamické i momentové parametry („Krokové motory”, 2021).

Motory NEMA17, využití v tomto projektu, jsou právě hybridní konstrukce.

### Zapojení vinutí fází

Krokové motory jsou vyrobeny s různě zapojeným vinutím. Mezi nejběžnější způsoby patří zapojení:

- bipolární (4 vodiče),
- unipolární (5 vodičů),
- unipolární nebo bipolární, vinutí spojena do série (6 vodičů),
- unipolární (vinutí spojena do série), bipolární (vinutí spojena do série nebo paralelně), 8 vodičů („Krokové motory”, 2021).



Obr. 1.1 – Způsoby zapojení fázových vinutí krokového motoru („Krokové motory”, 2021)

Tyto možné způsoby zapojení vinutí zachybuje obr. 1.1. Motory NEMA17 lze tedy zapojit bipolárně i unipolárně. V této práci jsou motory zapojeny pomocí 4 vodičů, tedy bipolárně.

### 1.2.3 Řízení krokových motorů

Krokový motor lze řídit postupným spínáním fází v různých režimech:

- čtyřtaktí s magnetizací jedné fáze,
- čtyřtaktí s magnetizací dvou fází,
- osmitaktí,
- mikrokrokování („Krokové motory”, 2021).

V této práci bude jako způsob řízení využito mikrokrokování.

## **Mikrokrokování**

Krokový motor se pohybuje po krocích. Ty jsou dány konstrukcí motoru. U využitého motoru NEMA17 to je 200 kroků na jednu otáčku, úhel kroku je tedy  $1,8^\circ$  (jedna otáčka je  $360^\circ$ ).

Při mikrokrokování se proudy v jednotlivých vinutích mění po malých krocích. Tím dochází k rozdělení kroku na mnoho menších tzv. mikrokroků. Při mikrokrokování se využívají všechna vinutí naráz, nicméně se nevyužívají s plným výkonem („Krokové motory”, 2021).

Na trhu jsou dostupné řadiče, které umožňují až 256 mikrokroků (příkladem může být využitý řadič TMC2209).

Výhodou použití mikrokrokování je dosažení vyšších otáček při dobrém průběhu točivého momentu. Dochází tak k uklidnění chodu motoru („Krokové motory”, 2021).

## **Řídicí obvody krokových motorů**

Úkolem řídicího obvodu krokového motoru je následující:

- měnit směr magnetického toku obvodem motoru změnou směru proudu vinutím,
- omezit velikost proudu vinutím na bezpečnou velikost, zároveň však zajistit co nejrychlejší nárůst proudu ve vinutí, aby bylo dosaženo plného využití motoru („Krokové motory”, 2021).

## **1.3 LIDAR**

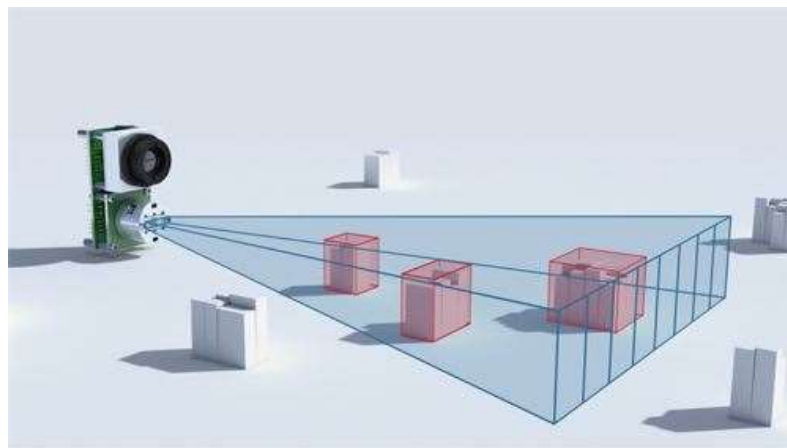
Lidar je technologie, která slouží k měření vzdáleností objektů. Lidar ke své funkci využívá laserový paprsek. Princip funkce spočívá v měření času mezi vysláním, odrazem a zachycením odrazu. Na základě této hodnoty logika vypočítá vzdálenost objektu kolem sebe nebo v určitém směru. Lidar osvětluje okolí světlem a měří odražené impulsy za pomoci snímače (ten je umístěn v blízkosti vysílače). Po vyslání světla z vysílače se světlo pohybuje, dokud nedosáhne cíle. Zde se část světelné energie odrazí zpět k emitoru. Časový rozdíl mezi vyzařovaným a přijatým pulsem určuje vzdálenost cíle (Kilián, 2018).

### **Využití lidarů:**

- meteorologie,
- zemědělství (analýza výnosu plodin),
- archeologie (vytváření digitálních trojrozměrných modelů archeologických lokalit),
- lesnictví (měření výšky porostu, mapování množství stromů, detekce ploch listů apod.),
- řízení a navigace u některých autonomních automobilů (Kilián, 2018).

V tomto projektu má lidar obdobné využití jako u autonomních vozidel, kde nejčastěji plní roli detektoru překážek a funguje jako rotující snímač, který mapuje okolní prostředí vozidla (příklad viz obr. 1.2). Využití lidarů umožňuje získat mapu, která poskytuje informace o tom, kde se v okolí vozidla nacházejí potenciální překážky a v jaké pozici je vůči nim vozidlo. Tato data jsou průběžně zpracovávána softwarem automobilu/vozidla tak, aby nedošlo k možné kolizi.

Aktuální cena lidarů se pohybuje v rozmezí cca 2 až 3 tisíc korun. V tomto projektu je však lidar sestaven z cenově dostupných komponent. Jedná se o využití krokového motoru a laserového senzoru pro měření vzdálenosti. Pro správnou funkci lidarů je senzor připojen přes sběrný kroužek tak, aby nedocházelo k poškození vodičů. Cena lidarů se v tomto projektu pohybuje kolem 1 300 Kč.



Obr. 1.2 – Lidar (Kilián, 2018)

## 1.4 VÝBĚR KOMPONENT

Jednotlivé komponenty jsou voleny na základě určitých požadavků. Mezi nejdůležitější požadavky patří následující:

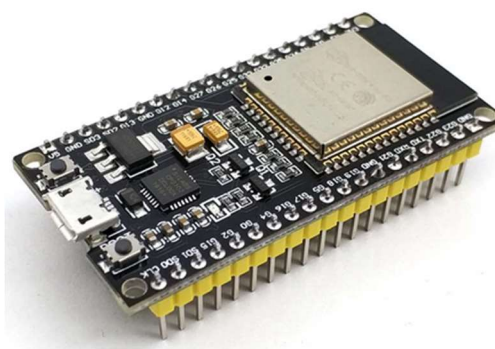
- malé rozměry,
- snadná implementace,
- nízká cena.

Prvním kritériem jsou malé rozměry. K základní mikroprocesorové desce (ESP32) je připojeno množství komponent a rozměry vozidla taktéž limitují prostor pro použitou elektroniku. Snahou je tedy využití komponent s co nejmenšími rozměry.

Dalším kritériem je snadná implementace. Komponenty jsou voleny též na základě dostupných knihoven, dále pak na základě existence příkladů, jak s tímto komponentem pracovat.

Senzor vzdálenosti je následně volen i na základě co nejmenšího počtu připojovaných vodičů. Komponenty jsou zvoleny také podle ceny, jelikož je tento projekt poměrně finančně nákladný. Snahou je volba komponent s příznivou prodejní cenou. Proto je většina komponent zakoupena ze zahraničních internetových obchodů.

### 1.4.1 ESP32



Obr. 1.3 – ESP32

Jako řídicí jednotka vozidla byla zvolena vývojová deska ESP32 (obr. 1.3). Jedná se o desku firmy Espressif, která je kompatibilní s deskami Arduino. Lze ji programovat v integrovaném vývojovém prostředí od Arduina (Arduino IDE), případně pomocí jazyka Python. Tato deska poskytuje na rozdíl od svého předchůdce ESP8266 vyšší výkon, ale i řadu nových funkcí. Deska umožňuje využití technologie WiFi, ale také Bluetooth verze 4.2 s podporou BLE (Bluetooth Low Energy). Procesor pak obsahuje dvě výpočetní jádra. SRAM paměť má velikost 512 kB. Deska disponuje také větším počtem GPIO (vstupně-výstupních) portů. Dále deska obsahuje větší počet pinů s podporou sběrnic SPI, I<sup>2</sup>C a UART (Stehlík, ©1998–2021).

Detailní specifikaci řídicí jednotky ESP32 zobrazuje obr. 1.4.

| Categories   | Items                       | Specifications  |
|--------------|-----------------------------|---|
| Wi-Fi        | Standards                   | FCC, CE, TELEC, KCC   |
|              | Protocols                   | 802.11 b/g/n/d/e/i/k/r (802.11n up to 150 Mbps)<br>A-MPDU and A-MSDU aggregation and 0.4 $\mu$ s guard interval support |
|              | Frequency range             | 2.4 ~ 2.5 GHz   |
| Bluetooth    | Protocols                   | Bluetooth v4.2 BR/EDR and BLE specification   |
|              | Radio                       | NZIF receiver with -98 dBm sensitivity  |
|              |                             | Class-1, class-2 and class-3 transmitter  |
|              |                             | AFH   |
| Audio        | CVSD and SBC                |   |
| Hardware     | Module interface            | SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, I2C, IR   |
|              |                             | GPIO, capacitive touch sensor, ADC, DAC, LNA pre-amplifier  |
|              | On-chip sensor              | Hall sensor, temperature sensor   |
|              | On-board clock              | 26 MHz crystal, 32 kHz crystal  |
|              | Operating voltage           | 2.2 ~ 3.6V  |
|              | Operating current           | Average: 80 mA  |
|              | Operating temperature range | -40°C ~ 85°C *  |
|              | Ambient temperature range   | Normal temperature  |
| Package size | 18 mm x 20 mm x 3 mm        |   |
| Software     | Wi-Fi mode                  | Station/softAP/SoftAP+station/P2P   |
|              | Security                    | WPA/WPA2, WPS, TLS  |
|              | Encryption                  | WEP/TKIP/AES/RSA/ECC/SHA  |
|              | Firmware upgrade            | UART Download / OTA (via network) / download and write firmware via host  |
|              | Software development        | Supports Cloud Server Development / SDK for custom firmware development   |
|              | Network protocols           | IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT  |
|              | User configuration          | AT instruction set, cloud server, Android/iOS App   |

Obr. 1.4 – Specifikace ESP32 (Stehlík, ©1998–2021)

### 1.4.2 Rozšiřující I/O modul pro ESP32 MCP23017

Tento rozšiřující 16bitový modul (obr. 1.5) nabízí využití 16 vstupně-výstupních pinů. Modul komunikuje po sběrnici I<sup>2</sup>C. V projektu slouží pro rozšíření počtu vstupů/výstupů řídicí desky ESP32.

Specifikace rozšiřujícího I/O modulu MCP23017 jsou následující:

- rozměry: délka 26,5 mm, šířka 21 mm,
- hmotnost: 3 g,
- provozní napětí: 1,8 ÷ 5,5 V,
- vysokorychlostní rozhraní I<sup>2</sup>C (100 kHz, 400 kHz, 1,7 MHz),
- konfigurovatelný zdroj přerušení,
- provozní teplota: -40 až +85 °C.



Obr. 1.5 – Rozšiřující I/O modul MCP23017

Údaje jsou převzaty z („MCP23017 Bidirectional 16-Bit I/O Expander”, ©2010-2020).

### 1.4.3 Krokový motor NEMA17 (17HS4401)

Jedná se o krokový motor s úhlem kroku 1,8 stupně (obr. 1.6). Vyznačuje se nízkou hlučností a kvalitním zpracováním. Tento projekt obsahuje celkem pět motorů. Čtyři tyto motory slouží k pohybu vozidla a poslední motor pro otáčení lidarů.

Specifikace krokového motoru NEMA17 jsou následující:

- rozměry: délka 42 mm, šířka 42 mm, výška 40 mm, délka hřídele 24 mm, průměr hřídele 5 mm,
- hmotnost: 294 g,
- úhel kroku: 1,8 °,
- přídržný točivý moment: 4 kg / cm,
- aretační točivý moment: 150 g / cm,
- proud cívkou: 1,7 A,
- provozní teplota: -20 až +50 °C.

Údaje jsou převzaty z („1,8° 42mm Hybrid Stepper Motor-NEMA17”, ©1990–2021).



Obr. 1.6 – Krokový motor NEMA17

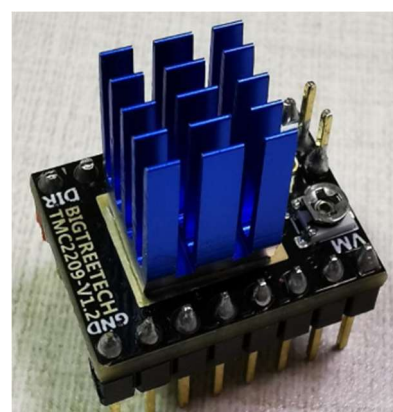
### 1.4.4 Řadič pro krokové motory TMC2209 – V1.2

Tento řadič (obr. 1.7) slouží k ovládání krokového motoru. Pomocí tohoto řadiče je možné ovládat nejen rychlost, ale i směr otáčení motorů. Těchto řadičů je v projektu využito celkem pět (čtyři pro ovládání motorů sloužících pro pohyb vozidla, jeden pro umožnění rotace lidarů).

Specifikace řadiče TMC2209 – V1.2 jsou následující:

- rozměry: délka 20,3 mm, šířka 15,3 mm,
- hmotnost: 3 g,
- provozní napětí: 4,75 ÷ 28 V,
- maximální špičkový proud: 2,8 A,
- maximální proud: 2 A,
- počet mikrokroků: 2, 4, 8, 16 (lze rozdělit až na 256 částí),
- pracovní mód: STEP/DIR nebo UART.

Údaje jsou převzaty z („BIGTREETECH TMC2209-V1.2”, 2021).



Obr. 1.7 – Řadič krokových motorů TMC2209



### 1.4.5 Senzor vzdálenosti TF-Luna

TF-Luna (obr. 1.8) je cenově výhodný jednopaprskový optický senzor. Umožňuje rychlou a stabilní detekci objektů na vzdálenosti do 8 metrů. Obsahuje vlastní diagnostiku, vzdálenou správu a aktualizace firmwaru, nastavitelnou rychlost komunikace a skenování, výstupní datový formát, vzdálený reset apod. Poskytuje vynikající výkon v aplikacích při získávání informací o vzdálenostech předmětů, osob, překážek o velikostech 3 cm<sup>2</sup> a větších, Senzor je využit v lidarů vozidla a slouží k měření vzdálenosti objektů od vozidla.

Specifikace senzoru vzdálenosti TF-Luna jsou následující:

- rozměry: délka 35 mm, šířka 13 mm, výška 21 mm,
- hmotnost: 5 g,
- snímací vzdálenost: 0,2 až 8 m,
- efektivní snímací vzdálenost: 0,3 až 4 m,
- rozlišení: 1 cm,
- frekvence snímání: až 250 Hz,
- komunikace: UART/I<sup>2</sup>C,
- průměrná spotřeba: 0,35 W,
- provozní napětí: 3,7 až 5,2 V,
- provozní teplota: -10 až +60 °C.

Údaje jsou převzaty z („Snímač vzdálenosti TF-Luna”, n. d.).



Obr. 1.8 – Senzor vzdálenosti TF Luna

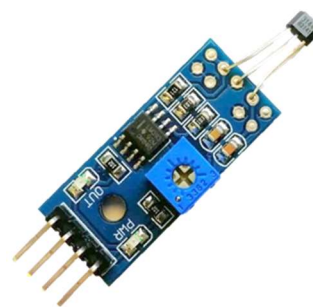
### 1.4.6 Hallův senzor

Hallův senzor (obr. 1.9) je senzor, který využívá Hallova jevu a v tomto projektu slouží ke zjištění výchozí pozice krokového motoru lidarů. Přestože senzor umožňuje práci s analogovým pinem, využit je pouze digitální pin. Důvodem je prostá detekce magnetu v okolí.

Specifikace Hallova senzoru jsou následující:

- rozměry: délka 45 mm, šířka 14 mm, výška 6 mm,
- hmotnost: 3 g,
- provozní napětí: 3,3 ÷ 5 V,
- výstup senzoru: analogový, digitální.

Údaje jsou převzaty z (Knotek, 2019).



Obr. 1.9 – Hallův senzor

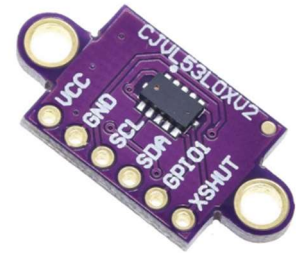
### 1.4.7 Senzor vzdálenosti VL53L0X

VL53L0X (obr. 1.10) je moderní senzor vzdálenosti, který pracuje na laserovém principu. Jedná se o ToF (Time of Flight) senzor. Využití této technologie umožňuje absolutní měření vzdálenosti bez ohledu na cílovou barvu a odrazivost detekovaného povrchu. V této práci slouží k tomu, aby vozidlo nenarazilo do překážky. Senzor je umístěn v přední části vozidla.

Specifikace senzoru vzdálenosti VL53L0X jsou následující:

- rozměry: délka 25 mm, šířka 10,4 mm,
- hmotnost: 2 g,
- maximální vzdálenost: 2 m,
- rozlišení: 1 mm,
- frekvenční rozsah: 50 Hz,
- typické zorné pole: 27 °,
- provozní napětí: 2,6 až 5,5 V.

Údaje jsou převzaty z („VL53L0X Time-of-Flight Laser Ranging Sensor“, ©2010-2020).



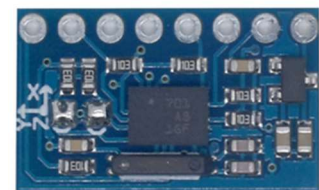
Obr. 1.10 – Senzor vzdálenosti VL53L0X

### 1.4.8 Gyroskopický modul BNO055

Jedná se o modul gyroskopu pro mikrokontrolery. Tento modul (obr. 1.11) obsahuje kromě gyroskopu také akcelerometr a magnetometr. V projektu slouží ke stabilizaci směru jízdy vozidla.

Specifikace gyroskopického modulu BNO055 jsou následující:

- rozměry: délka 20 mm, šířka 12 mm,
- hmotnost: 2 g,
- provozní napětí: 3 ÷ 5 V,
- čip: BNO055,
- rozsah gyroskopu:  $\pm 2000$  °/s,
- komunikační rozhraní: I<sup>2</sup>C.



Obr. 1.11 – Gyroskopický modul BNO055

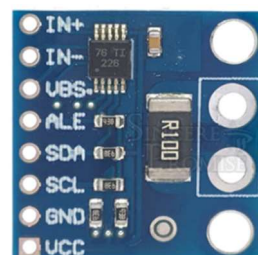
Údaje jsou převzaty z („3-osý gyroskop a akcelerometr GY-521, MPU6050“, 2021).

### 1.4.9 Měřič proudu INA226

Tento modul (obr. 1.12) umožňuje přesné měření proudu, napětí i výkonu. V projektu slouží ke zjištění spotřeby elektrické energie vozidla (motorů) a řídicí elektroniky.

Specifikace měřiče proudu INA226 jsou následující:

- rozměry: délka 20,9 mm, šířka 19,4 mm, výška 1,6 mm,
- hmotnost: 2 g,
- provozní napětí:  $2,7 \div 5,5$  V,
- vysoká přesnost měření,
- konfigurovatelná možnost průměrování,
- 16 programovatelných adres.



Obr. 1.12 – Měřič proudu INA226

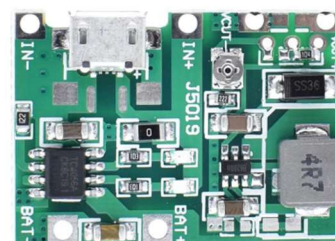
Údaje jsou převzaty z („INA226 current/power sensor”, ©2010-2020).

### 1.4.10 Nabíjecí a ochranný modul se zvyšujícím stejnosměrným měničem

Tento modul (obr. 1.13) umožňuje dobíjení Li-ionové baterií bez nutnosti jejich vyjmutí z vozidla. Umožňuje též ochranu proti přepětí, podpětí i nadproudu. Součástí modulu je i zvyšující stejnosměrný měnič. Ten umožňuje získat volitelné napětí (pomocí potenciometru umístěného na desce), které je vyšší než na vstupu modulu. Tento projekt jej využívá k úpravě napětí při napájení řídicí elektroniky vozidla.

Specifikace nabíjecího modulu se zvyšujícím měničem jsou následující:

- rozměry: délka 33 mm, šířka 23 mm,
- hmotnost: 4 g,
- vstupní napětí:  $4,5$  V  $\div$   $8$  V,
- výstupní napětí:  $4,3$  V  $\div$   $27$  V,
- dobíjecí napětí:  $4,2$  V
- dobíjecí proud: max.  $1$  A,
- vybíjecí proud: max.  $2$  A,
- napětí Li-ionové baterie:  $3,3 \div 4,2$  V,



Obr. 1.13 – Nabíjecí a ochranný modul se stejnosměrným zvyšujícím měničem

Údaje jsou převzaty z („Lithium Battery Charger”, ©2010-2020).

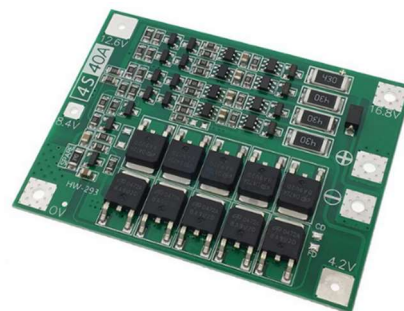
### 1.4.11 Nabíjecí a ochranný modul HW-293

Nabíjecí a ochranný modul HW-293 (obr. 1.14) slouží k nabíjení baterií (MH12210) pro napájení motorů vozidla. Taktéž slouží jako ochrana proti nadproudu, zkratu apod.

Specifikace nabíjecího a ochranného modulu HW-293 jsou následující:

- rozměry: délka 61 mm, šířka 41 mm, výška 4 mm
- hmotnost 10 g,
- provozní napětí: 12,6 V,
- nabíjecí napětí: 12,6 ÷ 13,6 V.
- nabíjecí proud: max. 20 A,
- vybíjecí proud: 40 A.

Údaje jsou převzaty z („Lithium battery protection board”, ©2010-2020).



Obr. 1.14 – Nabíjecí a ochranný modul HW-293

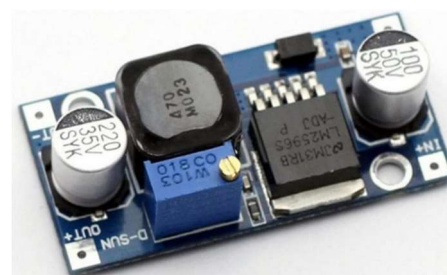
### 1.4.12 Stejnosemřný snižující měnič LM2596

Tento snižující stejnosměrný měnič (obr. 1.15) umožňuje získat volitelné napětí (pomocí potenciometru umístěného na desce), které je nižší než na vstupu modulu. Tento projekt jej využívá k úpravě napětí při nabíjení baterií externím napájecím adaptérem.

Specifikace stejnosměrného snižujícího měniče jsou následující:

- rozměry: délka 43 mm, šířka 21 mm, výška 14 mm,
- hmotnost: 9 g,
- vstupní napětí: 4,5 ÷ 40 V,
- výstupní napětí: 3 ÷ 35 V,
- maximální výstupní proud: 3 A,
- maximální účinnost: 92 %
- provozní teplota: -45 až +85 °C.

Údaje jsou převzaty z („Stepdown nastavitelný měnič s LM2596 DC-DC”, 2016) .



Obr. 1.15 – Stejnosemřný snižující měnič LM2596

### 1.4.13 OLED displej SSH1106

Jedná se o OLED displej typu SSH1106 (obr. 1.16), který podporuje komunikaci přes I<sup>2</sup>C. Tento displej v práci slouží k zobrazení dat o činnosti vozidla a je umístěn v jeho střední části.

Specifikace OLED displeje SSH1106 jsou následující:

- rozměry: délka 35,4 mm, šířka 33,5 mm, výška 3 mm,
- hmotnost: 7 g,
- úhlopříčka: 1,3 palce,
- rozlišení: 128×64 px,
- komunikační rozhraní: I<sup>2</sup>C,
- provozní teplota: -30 až +80 °C.



Obr. 1.16 – OLED displej SSH1106

Údaje jsou převzaty z („IIC I2C OLED display 1,3" 128x64 Bílý”, 2019).

#### 1.4.14 Li-ionová baterie MH12210

Li-ionové baterie (obr. 1.17) slouží k napájení motorů a řídicí elektroniky vozidla.

Specifikace Li-ionové baterie MH12210 jsou následující:

- rozměry: průměr 18 mm, výška 65 mm,
- hmotnost: 46 g,
- kapacita: 3400 mAh,
- napětí: 3,7 až 4,2 V,
- typ baterie: NCR18650B.



Obr. 1.17 – Li-ionová baterie MH12210

Údaje jsou převzaty z („Rechargeable Li-ion battery”, ©2010-2020).

#### 1.4.15 Stejnoseměrný externí napájecí adaptér

Jedná se o stejnosměrný napájecí adaptér určený k nabíjení Li-ionových baterií (obr. 1.18). V projektu slouží k nabíjení baterií vozidla.

Specifikace stejnosměrného externího napájecího adaptéru jsou následující:

- rozměry: délka 113 mm, šířka 51 mm, výška 31 mm,
- hmotnost: 168 g,
- vstupní střídavé napětí: 100 ÷ 240 V, 50/60 Hz,
- výstupní stejnosměrné napětí: 12,6 V,
- výstupní proud: 3 A.



Obr. 1.18 – Stejnoseměrný externí napájecí adaptér

Údaje jsou převzaty z („Lithium Battery Charger”, ©2010-2020).

## 1.5 DISKUZE NAD VÝBĚREM KOMPONENT

### 1.5.1 Vývojová deska

Při výběru vývojové desky byly brány v potaz dvě možné platformy. První z nich je komerčně známá a dostupná platforma Arduino. Jako druhá možnost se jeví vývojová deska ESP32.

Arduino disponuje určitými výhodami. Výhodou je množství příkladů, které je u platformy Arduino mnohem větší než u ESP32. Jeho nevýhoda však spočívá v neexistenci WiFi čipu, který je pro tento projekt vhodný.

Vývojová deska ESP32 disponuje množstvím paměti potřebné pro tvorbu algoritmů umělé inteligence. Dále se snadno programuje. Navíc umožňuje využít technologie Bluetooth a WiFi. Právě technologie WiFi je v tomto projektu využita. Cena je taktéž menší při srovnání s deskami Arduino. Rozměry jsou taktéž menší. ESP32 dále umožňuje efektivnější a širší možnost práce s PWM signálem a taktéž pracuje na vyšší frekvenci.

S ohledem na tyto skutečnosti byla zvolena deska ESP32.

### 1.5.2 Motory

Výběru motorů předcházelo dlouhé rozmyšlení ohledně všech pozitiv i negativ jednotlivých řešení. Množství motorů použitelných v tomto projektu je celá řada. Jako hlavní dva favorité byly vybrány stejnosměrné motory s enkodéry a krokové motory.

Nicméně na internetu se lze dozvědět i o stejnosměrných i střídavých elektromotorech, které by teoreticky také bylo možné využít. Nicméně od tohoto nápadu bylo upuštěno, vzhledem k principu jejich ovládání a napájení. Taktéž jejich cena je ve srovnání se zbylými typy motorů mnohonásobně vyšší.

Pro tuto práci mohou být vhodným řešením stejnosměrné nebo krokové motory.

#### **Stejnoseměrný motor s enkodérem**

Stejnoseměrné motory s enkodéry se jeví jako dobrá volba. Mezi jejich výhody patří snadné ovládání a možnost využití vysokých otáček motoru. Existence enkodéru pak umožňuje získat informaci o orientaci a natočení motoru. Taktéž existuje množství příkladů, návodů a vzorových kódů, které umožňují tyto motory využít. Jejich cena se (v zahraničí) pohybuje kolem 350 Kč.

Vyšší otáčky motoru jsou však kompenzovány nižším točivým momentem. Standardní motory dostupné v zahraničních internetových obchodech bohužel neumožňují plnit

požadovanou činnost. Točivý moment naprázdno je sice dostatečný, avšak při zatížení značně klesá. Ve výsledku by to znamenalo, že se vozidlo s těmito motory sice rozjede, ale například na místě se už neotočí. Pokud by bylo více zatíženo, pak by se nemuselo ani rozjet.

Na internetu lze nalézt motory s vyšším točivým momentem i v záběru, nicméně už jsou dražší a méně snadno dostupné.

Další nevýhodou je nepřesnost jejich provozu. Motory mohou být vyrobeny různou technologií, nemusí se vždy rozjet úplně stejnou rychlostí a jejich koordinaci by bylo třeba určitým způsobem zajistit. Mezi běžně používaná řešení patří například regulace otáček jednotlivých kol pomocí softwarové verze PID regulátoru. Bylo by tedy třeba zavést zpětnou vazbu a řídit motory zpětnovazebně.

Na základě těchto skutečností bylo od stejnosměrných motorů s enkodéry upuštěno.

## **Krokový motor**

Krokové motory v principu umožňují přesné polohování. Proto se tyto motory uplatňují například v robotice, kde je přesné polohování nutné. Krokové motory využívají pevně definovaného kroku. Pokud jsou dodrženy parametry motoru, pak nedochází ke tzv. ztrátě kroku. Taktéž lze využít tzv. mikrokrokování, což umožňuje plynulejší chod motoru. Mezi jeho obrovské výhody patří velký točivý moment. Ve srovnání s běžnými stejnosměrnými motory se jedná o mnohonásobné hodnoty. Taktéž cena je nižší. Jejich běžná cena se pohybuje kolem 250 Kč. Naopak konstrukce krokových motorů je poměrně robustní a tyto motory se vyznačují velkou mechanickou trvanlivostí („Krokové motory”, 2021).

Mezi nevýhody krokových motorů patří spotřeba elektrické energie. Taktéž při vyšších rychlostech dochází ke snížení točivého momentu motoru. Další nevýhodou může být nižší plynulost chodu motoru ve srovnání se stejnosměrnými motory.

Na základě těchto skutečností bylo rozhodnuto využít krokové motory, které v kombinaci s vhodným řadičem tvoří ideální řešení pro tento typ projektu.

### **1.5.3 Řadič pro krokové motory**

Řadičů pro krokové motory existuje na trhu celá řada. Běžně dostupné a cenově přijatelné jsou např. řadiče s označením A4988 nebo DVR8825.

Pro tento projekt se však zajímavě jeví řadič TMC2209. Tento řadič má oproti ostatním řadu výhod. Při řízení tímto řadičem je pro motor těžké ztrácet kroky. Řadič totiž obsahuje řadu příkazů, které je možné naprogramovat tak, aby k tomu nedocházelo. Řadič také zajišťuje tišší chod motoru oproti jeho konkurentům. Další výhodou je větší chladičí podložka pro umístění

chladiče. Řadič také může bránit motoru před jeho roztřesením. Podstatnou výhodou je pak množství módů, která řadič obsahuje. Jako nejdůležitější se jeví mód „CoolStep“, který pomocí regulace protékajícího proudu jednotlivými cívkami umožňuje ušetřit až 75 % energie (při optimálním nastavením jeho parametrů) („TMC2209 Datasheet“, 2021).

Mezi nevýhody tohoto řadiče určitě patří jeho cena, která převyšuje cenu samotného krokového motoru a pohybuje se kolem 260 Kč. Další nevýhoda spočívá v absenci návodů pro práci s tímto řadičem. Jelikož se řadič prodává jako příslušenství k 3D tiskárnám, na internetu lze prakticky nalézt jen návody pro použití při 3D tisku.

Přesto se pro tento projekt jeví ideálně řadič TMC2209, který slouží jako náhrada za starší modely A4988 nebo DRV8825.

#### **1.5.4 Senzor vzdálenosti**

Mezi komerčně dostupné senzory vzdálenosti patří senzory, které jsou založeny na dvou fyzikálních principech. Využívají se ultrazvukové nebo laserové senzory.

Mezi typického zástupce ultrazvukových sensorů patří model HC-SR04 příp. jeho novější model HY-SRF05. Ultrazvukové senzory se vyznačují nízkou cenou a množstvím různých příkladů a návodů. Mezi nevýhody pak patří pevně definovaný pracovní úhel a neschopnost spolehlivě detekovat různě nakloněné překážky.

Jako typického zástupce laserových sensorů vzdálenosti lze uvést laserový senzor VL53L0X nebo novější VL53L1X. Přímo pro lidary existují senzory od firmy Benewake. Typickým zástupcem může být senzor TF-Luna, který umožňuje měřit do vzdálenosti 8 metrů. Tato firma však pro lidary nabízí celou řadu sensorů. Laserové senzory se vyznačují vyšší přesností měření, nižší spotřebou a kompaktními rozměry. Mezi jejich nevýhody pak patří cena, která je i mnohonásobně vyšší než u ultrazvukových sensorů.

Pro potřeby tohoto projektu se lépe jeví laserový princip měření vzdálenosti. Jako senzor vzdálenosti pro potřeby lidaru je tedy zvolen model TF-Luna. Dále je na vozidle využit laserový senzor VL53L0X.

#### **1.5.5 Napájení**

Pro napájení jsou zvoleny osvědčené Li-ionové baterie MH12210. Jedná se o baterie s poměrně velikou kapacitou 3400 mAh. Pomocí sériového zapojení baterií je dosaženo požadovaného napětí (12 V) a naopak zapojením této sériové kombinace paralelně je dosaženo dostatečné kapacity pro chod vozidla.



Baterie jsou využity pro napájení motorů i elektroniky, nicméně toto napájení je odděleno. Pro oba typy napájení je k dispozici nabíjecí modul. Není tedy nutné vyjmout baterie pro jejich nabití. Pro nabíjení a ochranu baterií pro řídicí elektroniku byl zvolen modul se zvyšujícím stejnosměrným měničem od výrobce GREAT IT. Tento modul sdružuje nabíjecí a ochranný modul HW-107 a stejnosměrný zvyšující měnič, který poskytuje 5 V pro napájení elektroniky vozidla.

Naopak pro nabíjení a ochranu baterií pro motory vozidla byl zvolen modul HW-293. Výhodou tohoto modulu je schopnost práce s vysokými proudy (až 40 A), ochrana baterií a nízká cena. Jeho nevýhoda pak spočívá ve velikosti tohoto modulu.

K úpravě napětí je využito měniče stejnosměrného napětí. Ten je zvolen na základě malých rozměrů a výsledné ceny. Měnič není třeba nijak programovat, stačí pomocí odporového trimru na desce nastavit požadované výstupní napětí.

### **1.5.6 Měření proudu**

Možnosti měření proudu za pomoci využití mikrokontrolerů je vícero. Pro měření proudu lze využít například moduly MAX4080, dále moduly řady INA (INA169, INA219, INA226, INA3221 apod.) nebo například moduly řady ACS (ACS712, ACS758), příp. WCS1800.

Pro tuto práci byly zvoleny měřiče proudu řady INA, konkrétně INA226. Důvodem je nízká cena modulu, existence knihoven a návodů pro obsluhu modulu, ale také vhodné rozměry. Navíc lze za pomoci těchto měřičů získat informace o proudu až do výše např. 8 A (maximální proud je omezen velikostí bočníku, na kterém se vyhodnocuje úbytek napětí).

## **1.6 PROGRAMOVÁNÍ DESEK ESP32**

Řídicí jednotka ESP32 se programuje prostřednictvím známého prostředí Arduino IDE. Přestože existují i jiné možnosti jak programovat tyto desky, bylo zvoleno prostředí Arduino IDE. Umožňuje totiž snadno implementovat volně dostupné knihovny z internetu, pomocí kterých lze pracovat s jednotlivými komponentami. Samotné programování pak probíhá obdobně jako u desek Arduino.

Arduino (tedy i ESP32) lze programovat v jazyce C nebo C++. Nejjednodušším řešením však zůstává využití knihovny Wiring. Ta je v současné době velmi používána. Kvůli její komplexnosti lze o této knihovně hovořit jako o samostatném programovacím jazyce (Knotek, 2019).

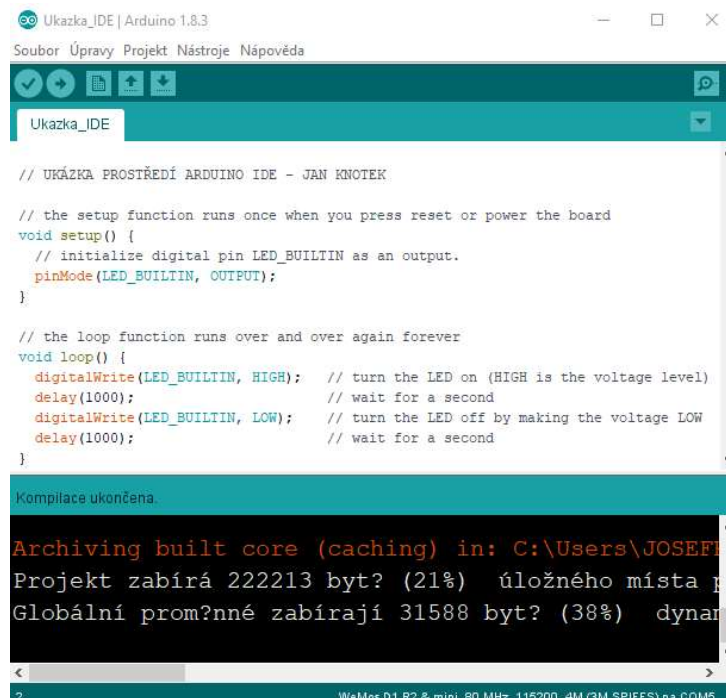
## 1.6.1 Arduino

Arduino je otevřená elektronická platforma, založená na jednoduché vývojové desce a prostředí, pomocí kterého lze tvořit programové kódy a nahrát tyto kódy do desek Arduino. Arduino lze využít pro řadu projektů. Lze jej využít pro práci s pasivními i aktivními elektronickými součástkami, s různými moduly, senzory nebo např. motory (Knotek, 2019).

Pro Arduino existuje řada kompatibilních modulů a rozšiřujících desek. Arduino komunitu tvoří řada lidí, existují internetová fóra a pro jeho vývojové prostředí lze nalézt velké množství knihoven. Právě proto bylo toto prostředí zvoleno (Knotek, 2019).

### Arduino IDE

Vzhledem ke kompatibilitě desek Arduino a ESP32 lze k jejich programování využít již zmiňované prostředí Arduino IDE. Arduino IDE (Integrated Development Environment) je software pro programování desek Arduino (tím pádem i desek kompatibilních s Arduinem). Vývojové prostředí (obr. 1.19) se skládá z textového editoru pro psaní kódu, textové konzole, prostoru pro zprávy, lišty s příkazy pro běžné funkce apod. Pomocí tohoto prostředí lze programy nahrávat do vývojových desek kompatibilních s Arduinem a komunikovat s nimi (Knotek, 2019).



```
Ukazka_IDE | Arduino 1.8.3
Soubor Úpravy Projekt Nástroje nápověda
Ukazka_IDE

// UKÁZKA PROSTŘEDÍ ARDUINO IDE - JAN KNOTEK

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

Kompilace ukončena.
Archiving built core (caching) in: C:\Users\JOSEF...
Projekt zabírá 222213 byt? (21%) úložného místa p...
Globální proměnné zabírají 31588 byt? (38%) dynar...
2 WeMos D1 R2 & mini, 80 MHz, 115200, 4M (3M SPIFFS) na COM6
```

Obr. 1.19 – Arduino IDE (Knotek, 2019)

## 1.7 MATLAB

MATLAB je inženýrský nástroj a interaktivní prostředí pro technické a vědecké účely. Slouží k technickým výpočtům, analýze dat, vývoji a vizualizaci algoritmů apod. Tento nástroj využívá řada inženýrů a vědců po celém světě. MATLAB umožňuje řešit úlohy napříč různými obory. Poskytuje řešení v oblastech jako je aplikovaná matematika, zpracování signálu a komunikace, strojové učení, finanční analýza, návrh řídicích systémů, robotika a mnoha dalších. Příkladem aktuálního využití systému MATLAB mohou být např. robotické sondy vesmírných programů, monitorovací systémy ve zdravotnictví, řešení bezpečnosti automobilů nebo například inteligentní energetické sítě nebo LTE systémy („MATLAB®”, ©1991-2021). MATLAB nabízí následující prostředky:

- výkonný programovací jazyk pro numerické výpočty a vývoj algoritmů,
- interaktivní prostředí s grafickými nástroji pro návrh a řešení problémů,
- nástroje pro matematické modelování, simulaci a analýzu dat, měření a testování,
- tisíce vestavěných funkcí z oblasti matematiky, statistiky, optimalizace a manipulace s daty,
- grafické funkce pro vizualizaci dat,
- vývojové nástroje pro zlepšení kvality kódu a maximalizaci výpočetního výkonu,
- nástroje pro tvorbu aplikací s grafickým uživatelským rozhráním (GUIDE, AppDesigner),
- funkce pro integraci algoritmů v jazyce MATLAB s externími aplikacemi a jazyky,
- paralelní a distribuované výpočty, GPU výpočty, práce s rozsáhlými daty („MATLAB®”, ©1991-2021).

Systém MATLAB dále nabízí více než 80 nadstavbových aplikačních knihoven funkcí. Tyto knihovny lze do prostředí MATLAB snadno přidávat i jeho po úspěšné instalaci („MATLAB®”, ©1991-2021).

Pro účely této práce je MATLAB využíván jako nástroj, který umožňuje tvorbu aplikací s grafickým uživatelským rozhráním. Aktuálně existují dvě základní možnosti práce s grafickým uživatelským rozhráním (GUI) v MATLABu. První z nich je starší, ale stále používaný nástroj GUIDE. Druhou možností je pak využití nového moderního nástroje pro tvorbu aplikací AppDesigner.

Součástí práce je pak aplikace, která umožňuje ovládání vozidla a vizualizaci aktuálních dat.

## 1.7.1 Porovnání nástrojů GUIDE a AppDesigner

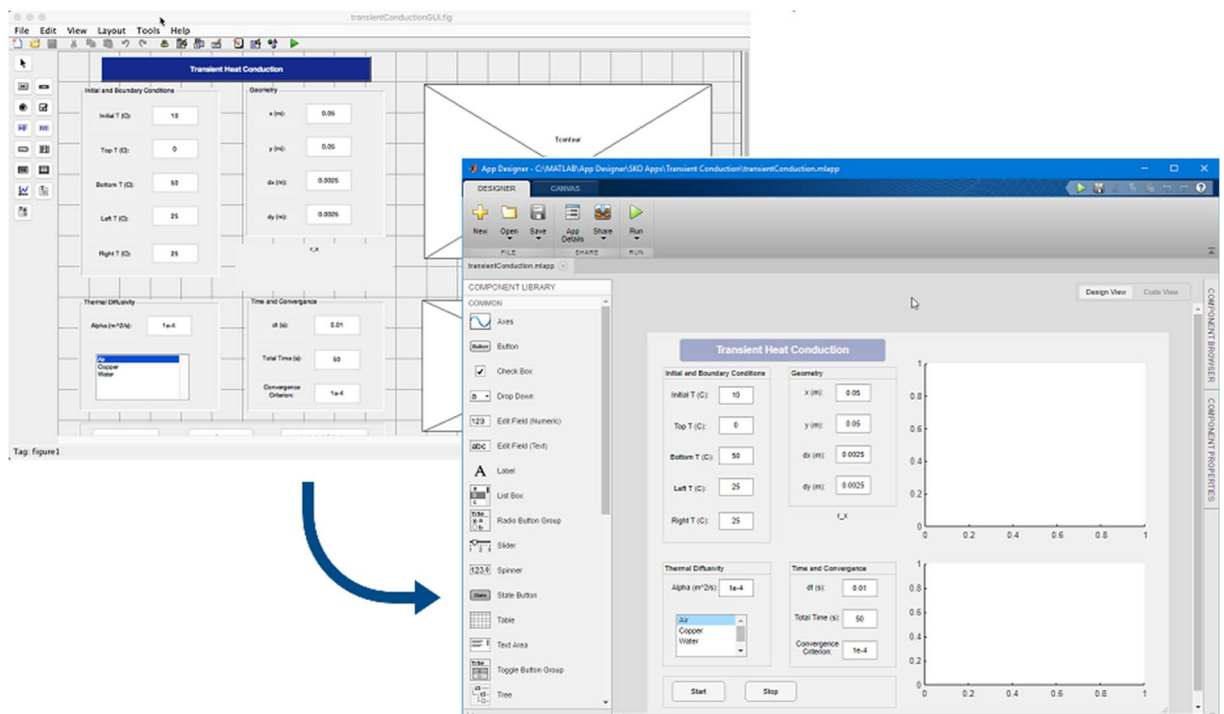
### GUIDE

GUIDE je původní prostředí pro tvorbu aplikací, které bylo představeno v roce 1996. Aktuálně je k dispozici ve verzi 5. V tomto prostředí lze vytvořit samostatné okno se standardními prvky – ovládacími (tlačítka, posuvníky atd.) a zobrazovacími (textové pole, grafy). Vlastnosti těchto prvků je možné programově měnit a události těchto objektů spojit s vlastní programovou obsluhou (Kupka, 2021).

Nicméně podpora tohoto nástroje bude v budoucích verzích MATLABu ukončena a společnost MathWorks doporučuje využívat nástroj AppDesigner (obr. 1.20).

### AppDesigner

AppDesigner je moderní vývojové prostředí pro tvorbu GUI, které bylo představeno v MATLABu verze R2016a. Aktuálně se jedná o doporučené prostředí pro tvorbu aplikací s grafickým uživatelským rozhraním. Cílem tohoto nástroje je úplná náhrada nástroje GUIDE. Tento nástroj obsahuje prakticky vše co GUIDE, avšak umožňuje využít i řadu dalších funkcí. Nejen proto je v tomto projektu využit právě AppDesigner („Comparing GUIDE and App Designer”, ©1994-2021).



Obr. 1.20 – Porovnání nástroje GUIDE a AppDesigner („GUIDE to App Designer Migration Tool for MATLAB”, ©1994-2021)

## 1.8 UMĚLÁ INTELIGENCE

### 1.8.1 Úvod do problematiky umělé inteligence

Umělá inteligence (UI) je souhrnný název pro systémy a stroje vytvořené za účelem efektivního provádění úkolů, které usnadňují lidskou práci. Principem UI je napodobení funkce lidské inteligence. Strojům je tak umožněno samostatně přemýšlet a rozhodovat. Umělá inteligence má (stejně jako člověk) potenciál pro další zlepšování a zefektivnění činností. K tomuto účelu využívá informace, se kterými pracuje (Kodůusková, 2021).

Existuje řada kritérií pro hodnocení míry umělé inteligence. Mezi ně patří například nalezení optimální varianty na základě zadaných kritérií, schopnost učit se z předchozích zkušeností, nalézat znaky podobnosti, adaptovat se při změnách okolních podmínek apod. Mezi další kritéria pak patří např. schopnost abstrakce (zjednodušení problému), generalizace (konstrukce obecného závěru) nebo třeba schopnost predikce (předvídaní) (Doležel, 2019).

Umělá inteligence se využívá v řadě aplikací.

#### **Příklady využití UI v praxi:**

- hry a hraní her,
- inteligentní osobní asistent (např. Siri od firmy Apple),
- strojový překlad (např. Google translate),
- samořiditelné automobily (např. automobil firmy Google),
- rozpoznávání obličejů,
- plánování přepravy zboží,
- filtrování spamů apod (Doležel, 2019).

#### **Další oblasti využití UI:**

- řízení procesů,
- počítačové vidění,
- porozumění textu,
- analýza, syntéza a rozpoznávání mluvené řeči,
- robotika,
- plánování a optimalizace,
- expertní systémy,
- bioinformatika,
- dolování dat,

- zpracování přirozeného jazyka (Doležel, 2019).

Umělá inteligence spojuje více vědních disciplín dohromady. Tato vědní disciplína stojí na těchto základních stavebních kamenech:

- teorie grafů,
- teorie fuzzy množin,
- teorie rozhodování a teorie her,
- teorie pravděpodobnosti a statistika,
- umělé neuronové sítě,
- umělé formální jazyky,
- metody rozhodování (Doležel, 2019).

Umělou inteligenci tvoří tři základní oblasti:

- metody prohledávání stavového prostoru,
- metody strojového učení,
- meta-heuristické optimalizační algoritmy (Doležel, 2019).

S těmito oblastmi práce využívá oblast metod prohledávání stavového prostoru. Pomocí vybrané metody je umožněn autonomní pohyb vozidla.

## 1.8.2 Stavový prostor a jeho prohledávání

### Stavový prostor

Stavový prostor definuje možné stavy a operátory pro přechod mezi stavy úlohy. Definice stavového prostoru je následující

$$S_P = (S, \phi), \tag{1.1}$$

kde  $S = \{s\}$  – množina všech možných stavů úlohy,

$\phi = \{\varphi\}$  – množina operátorů pro přechod mezi jednotlivými stavy (Šimková, 2021).

Platí, že k-tého stavu  $s_k$  je možné dosáhnout aplikací operátoru přechodu  $\varphi_{ik}$  na stav  $s_i$ . Stav  $s_i$  je tedy předchůdcem stavu  $s_k$  a obráceně stav  $s_k$  je následníkem stavu  $s_i$ . Následníkem vrcholu  $v_i$  v grafu  $GR$  se rozumí každý vrchol tohoto grafu, do kterého vede z  $v_i$  hrana. Předchůdcem vrcholu  $v_i$  v grafu  $GR$  se rozumí každý vrchol tohoto grafu, ze kterého vede z  $v_i$  hrana. Počáteční stav úlohy je označován jako  $s_0$ . Dále se definuje množina cílových stavů  $G =$

$\{g\}$  , které představují řešení problému. Řešení problému může být jedno nebo jich může být více. Platí  $G \subset S$  (Šimková, 2021).

Řešení úlohy ve stavovém prostoru je myšleno nalezení takové posloupnosti operátorů, jejichž postupná aplikace zajistí přechod z počátečního stavu  $s_0$  do některého z cílových stavů množiny  $G$  (Šimková, 2021).

Stavový prostor úlohy může být velmi rozsáhlý a nemusí být vždy možné v konečném čase prověřit všechny možné stavy a přechody mezi nimi. Existuje tedy celá řada strategií pro prohledávání stavového prostoru (Šimková, 2021).

### **Prohledávání stavového prostoru**

Prohledávání stavového prostoru je jednou ze základních oblastí umělé inteligence. Při hledání řešení se obvykle uvažují určité zjednodušující předpoklady:

- práce s konečným počtem stavů, diskrétní reprezentace,
- plně pozorovatelné prostředí – vše co ovlivňuje úlohu je známé,
- deterministické prostředí – každá akce má jeden přesně definovaný výstup pro každý stav,
- statické prostředí – změna stavu je možná pouze na základě námi vykonané akce, žádné vnější události (Doležel, 2019b).

Další zjednodušující předpoklady:

- přesné vymezení cílových stavů,
- řešením problému je posloupnost po sobě vykonaných akcí (nevyskytují se žádné paralelní děje, vše probíhá sériově),
- off-line plánování – stav systému se nemění v průběhu plánování (Doležel, 2019b).

### **Cíle a jejich definice**

Cíl závisí na úloze. Je také možné zadat více alternativních cílů. Cíl může být vyjádřen buď explicitně nebo pouze pomocí podmínek, které musí splňovat.

Příklady typů cílů:

- dosažení cílového stavu,
- dosažení cílového stavu při splnění podmínek,
- dosažení cílového stavu s maximální efektivností,
- vykonání zadaného úkolu (Doležel, 2019b).

V tomto projektu půjde o dosažení cílového stavu s maximální efektivností. Cílem je dostat robotické vozidlo z bodu A do bodu B nejkratší možnou cestou.

### **Metody prohledávání stavového prostoru**

Tyto metody se dělí na dvě základní kategorie:

- neinformované metody,
- informované metody.

Neinformované metody aplikují jednotlivé operátory slepě, prochází tedy stavový prostor bez informace o výhodnosti aplikace daného operátoru. Tyto metody tedy nijak nehodnotí výhodnost cesty. Výhodou je jejich jednoduchost. Nevyžadují výpočet žádné hodnotící funkce (Šimková, 2021).

Příklady neinformovaných metod:

- prohledávání do hloubky,
- prohledávání do šířky,
- prohledávání do šířky dle ceny,
- obousměrné prohledávání (Šimková, 2021).

Informované metody při hledání využívají různé heuristické znalosti. Jejich snahou je odhadnout, který z možných přechodů je z hlediska řešení problému výhodný. Tyto metody využívají další znalosti o řešeném úkolu. Tyto znalosti pak lze zakomponovat do tzv. heuristické funkce  $h$ . Heuristická funkce odhaduje cenu cesty z vrcholu  $v_i$  do nejbližšího cílového vrcholu (stavu). Tato funkce umožňuje redukovat množství prohledávaných stavů. Řešení je pak závislé na kvalitě heuristické funkce (Šimková, 2021).

Heuristická funkce:

- závisí na řešeném problému,
- pro jednu úlohu existuje více možností jejího vyjádření,
- je nezáporná,
- pro cílový stav musí být nulová,
- musí být přípustná (nesmí nadhodnocovat cenu cesty k cíli),
- musí být konzistentní (Doležel, 2019c).

Příklady informovaných metod:

- hladové heuristické hledání,
- paprskové prohledávání,



- algoritmus A\*.

Pro hodnocení výkonu algoritmů prohledávání stavového prostoru existují určité ukazatele:

- úplnost (existence řešení),
- optimálnost (existence nejlepšího řešení) (Doležel, 2019b).

Ukazatele výkonu algoritmů z hlediska náročnosti:

- časová složitost (čas nutný k nalezení řešení tzn. počet vrcholů vygenerovaných v průběhu řešení),
- paměťová náročnost (množství paměti nutné k nalezení řešení tzn. maximální počet vrcholů uložených v paměti) (Doležel, 2019b).

Ukazatele výkonu algoritmů z hlediska náročnosti jsou pak vyjádřeny pomocí asymptotické složitosti (Doležel, 2019b).

Pro implementaci algoritmu umělé inteligence byl zvolen A\* (A star algoritmus). Algoritmus A\*.

Jedná se o jednu z informovaných metod prohledávání stavového prostoru. Strategie algoritmu spočívá v expanzi (získání všech následníků) stavu s nejnižší celkovou cenou cesty. Využívá následující hodnotící funkci

$$f(v_i) = g(v_i) + h(v_i), \quad (1.2)$$

kde  $g(v_i)$  - cena cesty z počátečního vrcholu do vrcholu  $v_i$ ,

$h(v_i)$  - odhad ceny cesty z vrcholu  $v_i$  do nejbližšího cílového vrcholu (stavu) (Doležel, 2019c).

Algoritmus byl vybrán z několika důvodů. Algoritmus vykazuje dobré výsledky při prohledávání stavového prostoru. Pro nalezení cesty neprochází velkou část stavového prostoru, hledání tedy bude rychlejší. Taktéž tento algoritmus nalezne využití v reálných aplikacích.

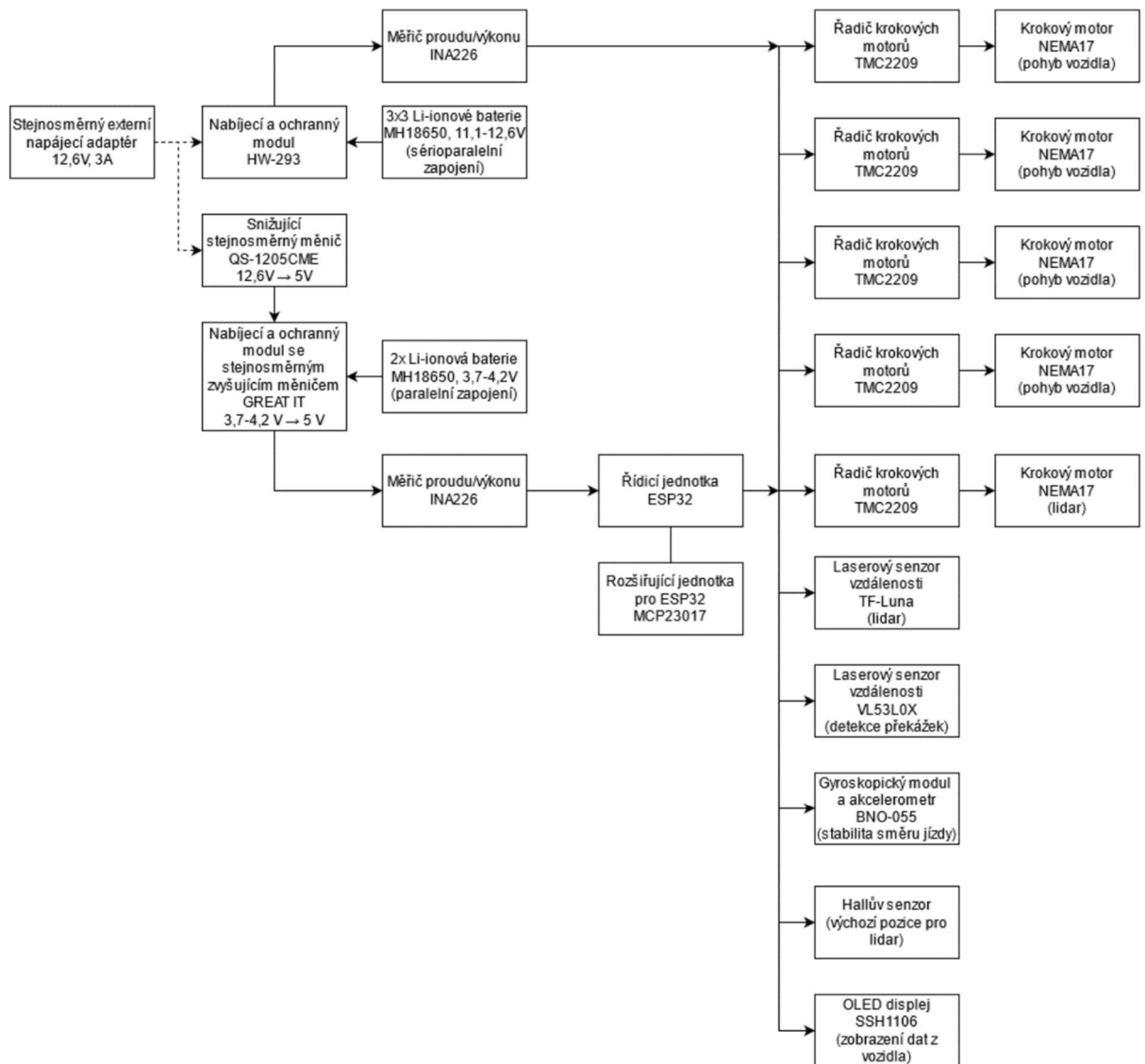
## **2 PRAKTICKÁ ČÁST**

Praktická část práce je věnována návrhu a realizaci autonomního robotického vozidla. Řídicí jednotkou je vývojová deska ESP32. Vozidlo je navrženo jako standardní vozidlo se čtyřmi koly. Tato kola jsou nezávisle řízena pomocí čtyř krokových motorů. Další krokový motor je pak využit pro konstrukci lidarů. Všechny krokové motory jsou řízeny pomocí řadiče krokových motorů. Vozidlo dále obsahuje senzor vzdálenosti (ochrana před případným nárazem do překážky) a gyroskop pro snazší zajištění stability směru jízdy. Napájecí část je řešena pomocí Li-ionových baterií a nabíjecích/ochranných modulů. Součástí návrhu elektroniky vozidla jsou i dva senzory pro měření proudu. Tím je zajištěno měření spotřeby elektrické energie vozidla. Pro úpravu napětí na patřičné hodnoty jsou k dispozici dva měniče stejnosměrného napětí. Návrh elektroniky vozidla je uveden na obr. 2.1.

Kromě hardwarové části práce se tato kapitola zabývá také tvorbou software a samotného průběhu jízdy vozidla. K dispozici je také informace o vytvořené aplikaci, pomocí které lze vozidlo ovládat.

### **2.1 NÁVRH ELEKTRONIKY VOZIDLA**

Návrh elektroniky vozidla je znázorněn pomocí blokového schématu na obr. 2.1.



Obr. 2.1 – Blokové schéma elektroniky vozidla

Elektronika vozidla obsahuje celkem tři části. První z nich je napájecí část, druhá část obsahuje řídicí elektroniku vozidla, třetí část tvoří lidar, Hallův senzor a OLED displej. Pro první dvě části vozidla je vytvořena samostatná DPS, která zásadním množstvím redukuje celkový počet vodičů vozidla.

### 2.1.1 Napájecí část

Napájecí část elektroniky vozidla obsahuje následující komponenty:

- li-ionové baterie MH18650,
- napájecí a ochranný modul HW-293,
- nabíjecí a ochranný modul se zvyšujícím stejnosměrným měničem,
- snižující stejnosměrný měnič QS-1205CME,

- zvyšující stejnosměrný měnič MT3608,
- měřič proudu/výkonu INA226 (2 ks),
- stejnosměrný externí napájecí adaptér.

### **Li-ionové baterie MH18650**

Napájení vozidla a elektroniky je řešeno pomocí Li-ionových baterií MH18650. Napájení vozidla i elektroniky je navrženo odděleně. Jedna Li-ionová baterie dle údajů výrobce poskytuje napětí 3,7 až 4,2 V.

Vzhledem k tomu, že jsou motory vozidla navrženy na 12 V, stačí zapojit tři baterie do série a přivést jejich napětí na příslušné řadiče (kromě řadičů je napětí z těchto baterií přivedeno i na měřič proudu/výkonu motorů vozidla). Nicméně vzhledem k požadavku většího proudu a větší kapacity jsou baterie zapojeny sérioparalelně (vždy tři baterie paralelně, celkem tři kombinace paralelního zapojení baterií v sérii). Tím je zajištěna dostatečná kapacita a dostatečný proud.

Pro napájení elektroniky je třeba přivést napětí 5 V. V tomto případě stačí jedna baterie (3,7 až 4,2 V) a následné zvýšení napětí na 5 V pomocí zvyšujícího měniče napětí. Vzhledem k požadavku na vyšší kapacitu jsou zapojeny dvě baterie paralelně. Tyto baterie tedy napájí řídicí jednotku ESP32 (včetně jejího rozšiřujícího I/O modulu), zvyšující stejnosměrný měnič, oba senzory vzdálenosti a měřič proudu/výkonu pro část řídicí elektroniky vozidla.

### **Napájecí a ochranný modul HW-293**

Napájecí a ochranný modul HW-293 slouží k nabíjení baterií pro napájení motorů vozidla (napájení vozidla probíhá pomocí externího napájecího adaptéru). Tento modul také zajišťuje ochranu proti nadproudu, zkratu, ochranu proti vybití a přebití. Tato varianta modulu je vhodná pro 3 články Li-ion v sérii. Paralelně lze článků zapojit více, v tomto případě jsou zapojeny vždy 3 články paralelně.

### **Nabíjecí a ochranný modul se zvyšujícím stejnosměrným měničem**

Nabíjecí a ochranný modul od firmy GREAT IT slouží k nabíjení baterií pro napájení elektroniky vozidla. Tento modul také zajišťuje ochranu proti nadproudu, zkratu, ochranu proti vybití a přebití. Tento modul je vhodný pro napájení jedné baterie, nicméně paralelně jich lze zapojit více. V tomto případě jsou zapojeny 2 články paralelně. Modul také obsahuje stejnosměrný zvyšující měnič, který slouží k úpravě napětí poskytovaného baterií pro napájení řídicí elektroniky vozidla (3,7 až 4,2 V) na napětí potřebné pro napájení jednotlivých komponent pro řízení vozidla (5 V).

## **Snižující stejnosměrný měnič QS-1205CME**

Snižující stejnosměrný měnič QS-1205CME slouží k úpravě napětí poskytovaného bateriemi pro napájení vozidla (12,6 V) na napětí potřebné pro napájení napájecího a ochranného modulu HW-107 (5 V).

## **Měřič proudu/výkonu INA226**

Měřiče proudu/výkonu INA226 jsou využity dva. První z nich slouží k měření protékajícího proudu řídicí elektronikou vozidla, druhý pak pro měření proudu tekoucího motory. Využití těchto modulů umožňuje monitorovat spotřebu elektrické energie dvou hlavních částí vozidla (motory, elektronika).

## **Stejnosemřný externí napájecí adaptér**

Stejnosemřný externí napájecí adaptér slouží k nabíjení baterií pomocí napájecího a ochranného modulu HW-293. Tento adaptér není přímo součástí vozidla. V případě vybití baterií vozidla je třeba tento adaptér k vozidlu připojit. Adaptér ze síťového střídavého napětí vytvoří stejnosměrné napětí o hodnotě 12,6 V. Adaptér také zajišťuje vstupní nadproudovou ochranu, ochranu výstupního proudu a ochranu proti zkratu.

### **2.1.2 Část řídicí elektroniky vozidla**

Část řídicí elektroniky vozidla obsahuje následující komponenty:

- řídicí vývojová deska ESP32,
- rozšiřující I/O deska pro ESP32 CJMCU-2317,
- optický senzor vzdálenosti VL53L0XV2,
- laserový senzor vzdálenosti TF-Luna,
- řadiče krokových motorů TMC2209 (5 ks),
- gyroskopický modul BNO055.

#### **Řídicí vývojová desku ESP32**

Řídicí vývojová deska ESP32 slouží k celkovému řízení vozidla. Obsahuje programový kód zajišťující všechny potřebné úkony. Pomocí naprogramovaného ESP32 lze ovládat jeho rozšiřující I/O modul, ovládat řadiče motorů a zajišťovat tak pohyb motorů a tím i pohyb kol vozidla. Dále lze získávat a vyhodnocovat údaje z ostatních modulů. Programový kód této desky též umožňuje autonomní pohyb vozidla pomocí algoritmu umělé inteligence. Součástí kódu je i přenos dat přes WiFi do aplikace vytvořené v MATLABu (i naopak).

## **Rozšiřující I/O deska pro ESP32 CJMCU-2317**

Rozšiřující I/O deska pro ESP32 CJMCU-2317 umožňuje využít další vstupně/výstupní piny desky ESP32. Její aplikace umožňuje v budoucnu možnost připojení dalšího zařízení k ESP32.

## **Laserový senzor vzdálenosti VL53L0XV2**

Laserový senzor vzdálenosti VL53L0XV2 slouží k detekci překážek před vozidlem. Vzhledem k tomu, že se lidar (který mapuje okolí) točí s určitou rychlostí, nemusí při velmi rychlém pohybu vozidla zaznamenat překážku. Z tohoto důvodu je v popředí vozidla umístěn tento senzor. V případě příliš malé vzdálenosti mezi vozidlem a překážkou dojde k jeho zastavení. Maximální vzdálenost, kterou je senzor schopen detekovat jsou dva metry.

## **Laserový senzor vzdálenosti TF-Luna**

Laserový senzor vzdálenosti TF-Luna je součástí lidaru vozidla. Slouží k mapování okolí v 2D prostoru. Získává tedy informaci o vzdálenosti k překážkám v okolí vozidla. Senzor je přes sběrný kroužek upevněn v plastové konstrukci a pomocí krokového motoru je umožněno jeho otáčení. Senzor je schopný detekovat překážku na vzdálenost až 8 metrů.

## **Řadiče krokových motorů TMC2209**

Řadiče krokových motorů TMC2209 slouží k ovládání krokových motorů vozidla. Jeden z nich je určen k natáčení lidaru, další 4 pak slouží pro samotný pohyb vozidla.

## **Gyroskopický modul BNO055**

Gyroskopický modul BNO055 slouží k stabilitě směru jízdy. Přestože modul obsahuje ještě akcelerometr a magnetometr, tyto funkce nejsou ve vozidlu využity.

### **2.1.3 Návrh DPS**

Návrh desek plošných spojů je proveden v prostředí Eagle. Obě desky (napájecí resp. řídicí elektroniky) jsou realizovány jako dvouvrstvé. Vzhledem k omezenému prostoru na vozidle (tím i omezené ploše DPS) a množství vodičů by realizace desky jako jednostranné nebylo vhodné. Navíc pro zhotovení desky bylo využito zahraniční firmy, která umožňuje zakázkovou výrobu na míru a dokáže bez problému zhotovit vícevrstvé DPS.

Desky jsou tvořeny převážně konektory typu dupont, do kterých se připojují jednotlivé komponenty. Při návrhu hardware vozidla byly brány v potaz možné změny v konstrukci, proto je možné veškeré komponenty snadno vyměnit. Toto řešení je též výhodné v případě, že dojde

k poškození některé z komponent. Dále desky obsahují konektory do DPS pro připojování různých vodičů (vodiče od motorů, napájení, I<sup>2</sup>C komunikace apod.) a napevno připájené součástky pro správnou funkci zapojení (např. rezistory pro UART komunikaci a filtrační kondenzátory).

Pro obě DPS je k dispozici schéma zapojení, osazovací plány a pohled na navrženou desku z prostředí Eagle.

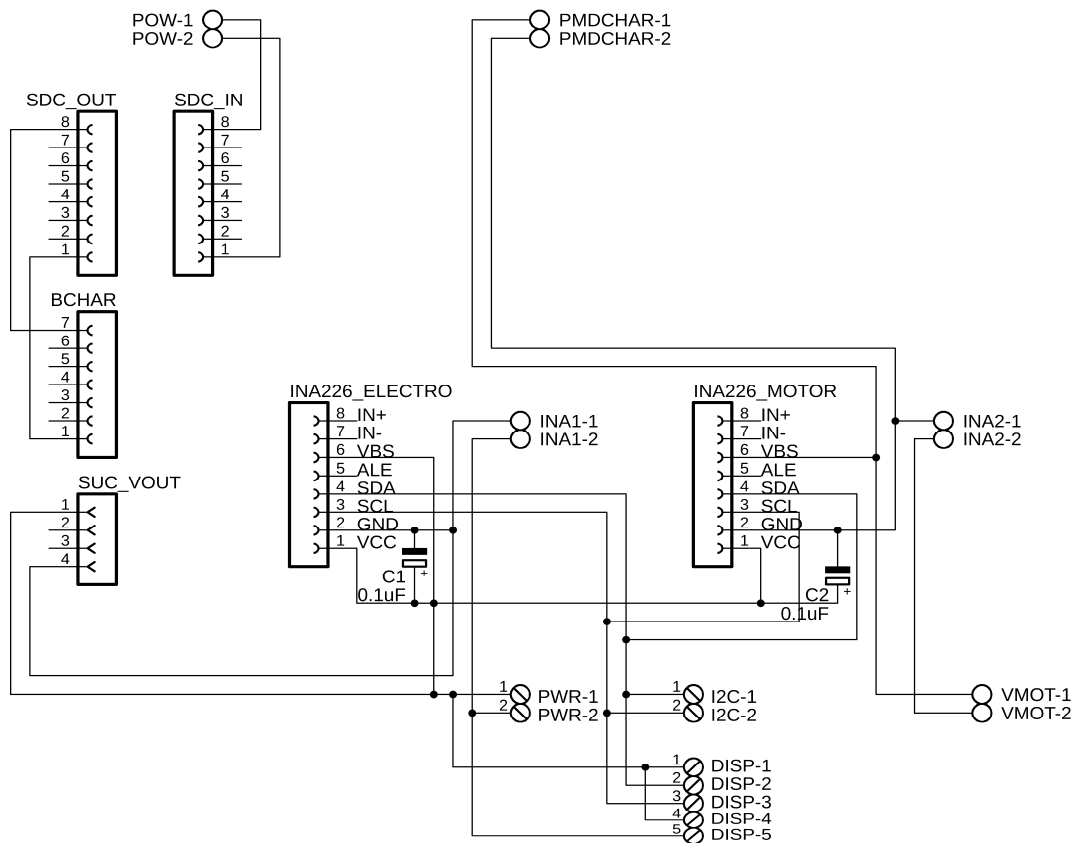
### **DPS pro napájecí elektroniku**

DPS napájecí části elektroniky vozidla obsahuje konektory pro následující komponenty:

- nabíjecí a ochranný modul HW-293 (Battery protection board),
- nabíjecí a ochranný modul se zvyšujícím stejnosměrným měničem (Battery Charger, Step up-converter),
- snižující stejnosměrný měnič HW-411 (Step down convertor LM2596, HW-411),
- měřič proudu/výkonu INA226 (INA226),
- OLED displej SSH1106 (DISP).

Dále pak obsahuje konektory pro připojení různých vodičů (napájení, motory, I<sup>2</sup>C komunikace) a kondenzátory.

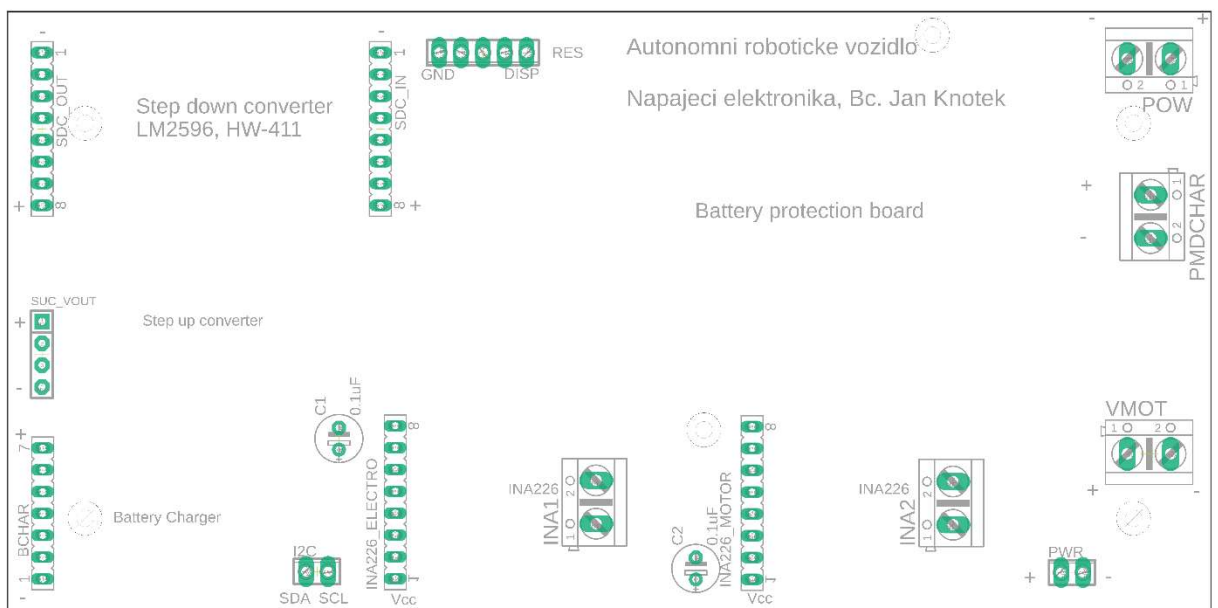
Obr. 2.2 zachycuje návrh napájecí DPS v prostředí EAGLE.



Obr. 2.2 – Schéma zapojení napájecí DPS

Dalším krokem bylo rozmístění součástek, resp. převážně konektorů napájecí DPS.

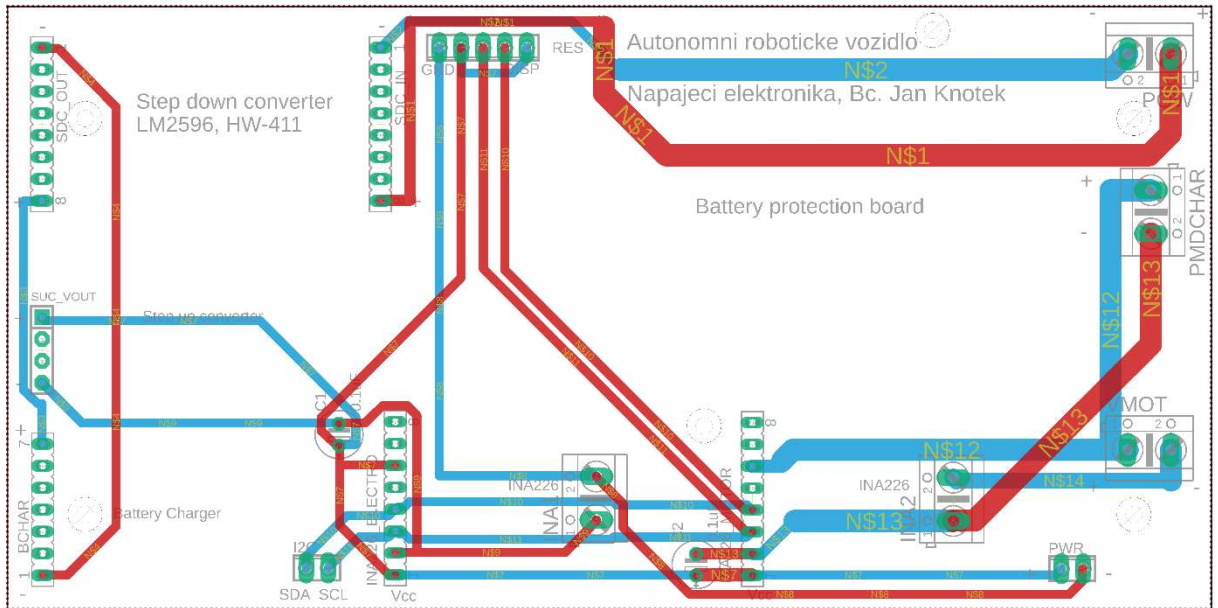
Obr. 2.3 zobrazuje osazovací plán této desky.



Obr. 2.3 – Osazovací plán napájecí DPS

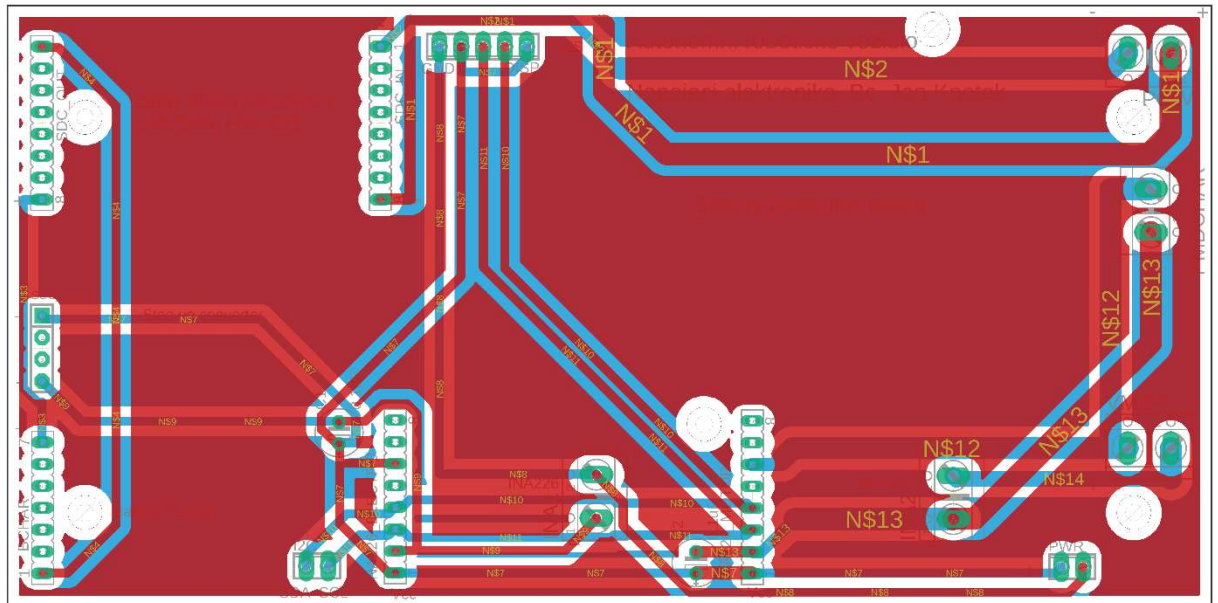


Následně bylo třeba propojit vodivé cesty a upravit jejich šířku tam, kde to bylo třeba (viz obr. 2.4).



Obr. 2.4 – Navržená napájecí DPS

Pro snížení možnosti rušení byla DPS navržena s vylitým polygonem (viz obr. 2.5).



Obr. 2.5 – Navržená napájecí DPS s vylitým polygonem

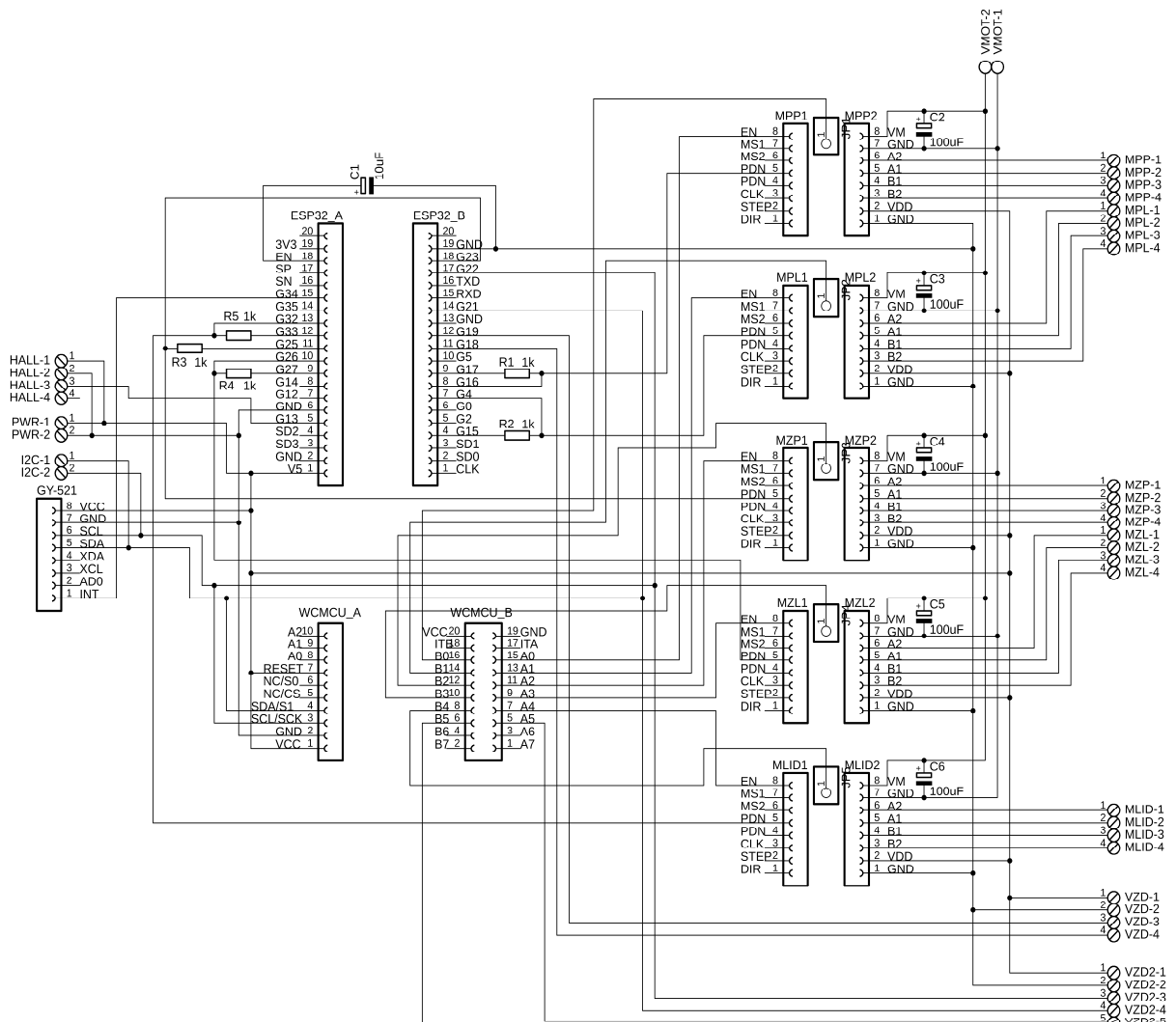
## **DPS pro řídicí elektroniku**

DPS řídicí části elektroniky vozidla obsahuje konektory pro následující komponenty:

- řídicí vývojovou desku ESP32 (ESP32\_A, ESP32\_B),
- rozšiřující I/O desku pro ESP32 CJMCU-2317 (WCMCU\_A, WCMCU\_B),
- laserový senzor vzdálenosti VL53L0XV2 (VZD2),
- laserový senzor vzdálenosti TF-Luna (VZD),
- Hallův senzor (HALL),
- řadiče krokových motorů TMC2209 (MPL1(2), MPP1(2), MZL(2), MZP(2), MLID1(2)),
- gyroskopický modul BNO055 (GY-521).

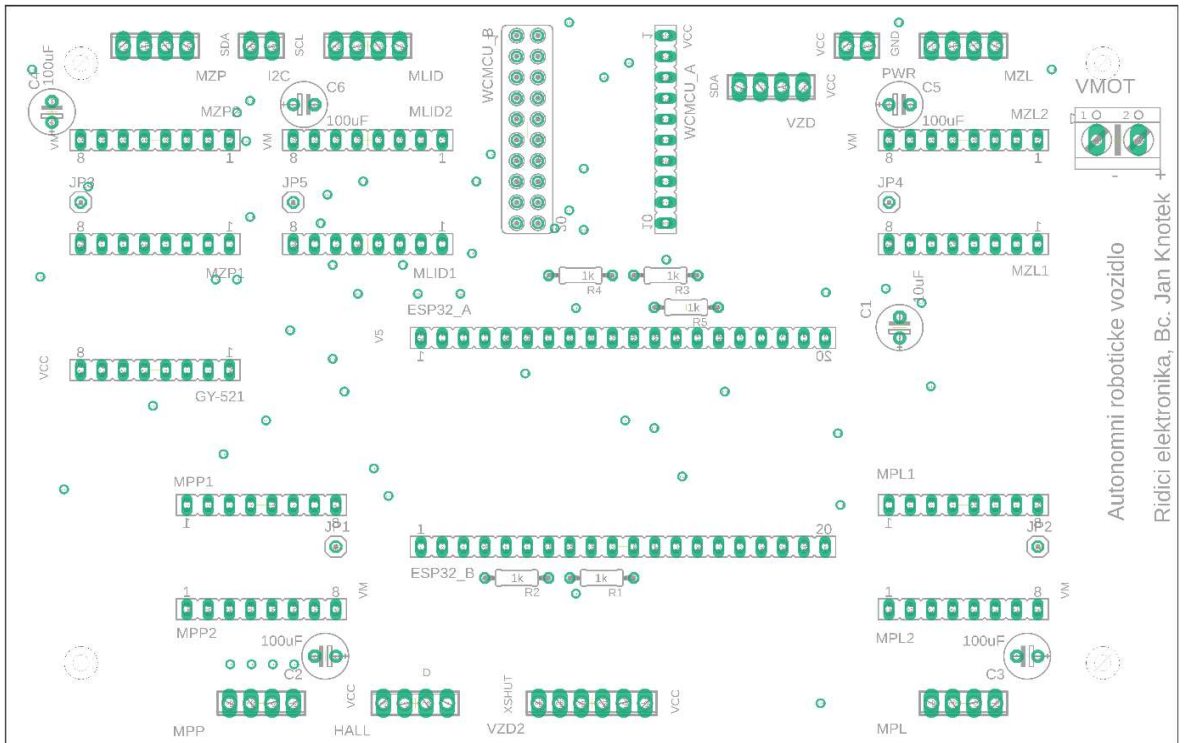
Dále pak obsahuje konektory pro připojení různých vodičů (napájení, motory, I<sup>2</sup>C komunikace), rezistory pro UART komunikaci a kondenzátory.

Postup práce byl v případě řídicí DPS obdobný, nejprve došlo k návrhu zapojení (obr. 2.6).



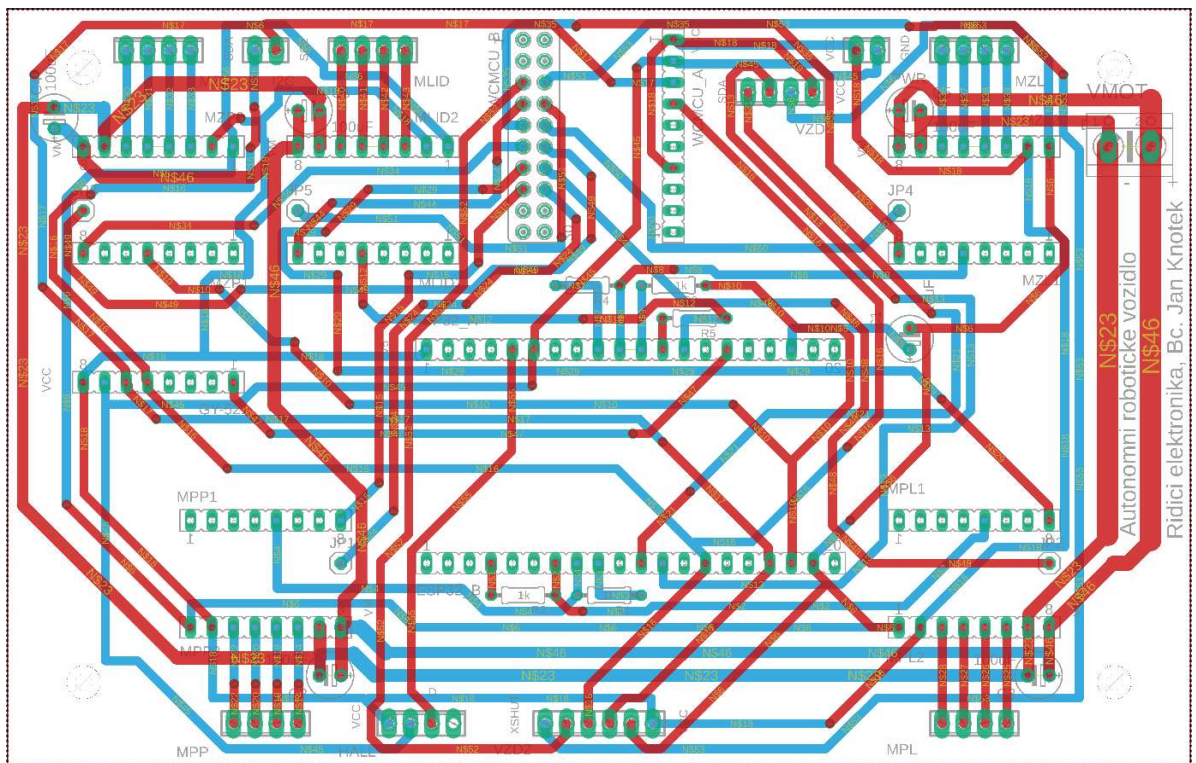
Obr. 2.6 – Schéma zapojení řídicí DPS

Dalším krokem bylo rozmístění součástek, resp. převážně konektorů řídicí DPS. Obr. 2.8 zobrazuje osazovací plán této desky.



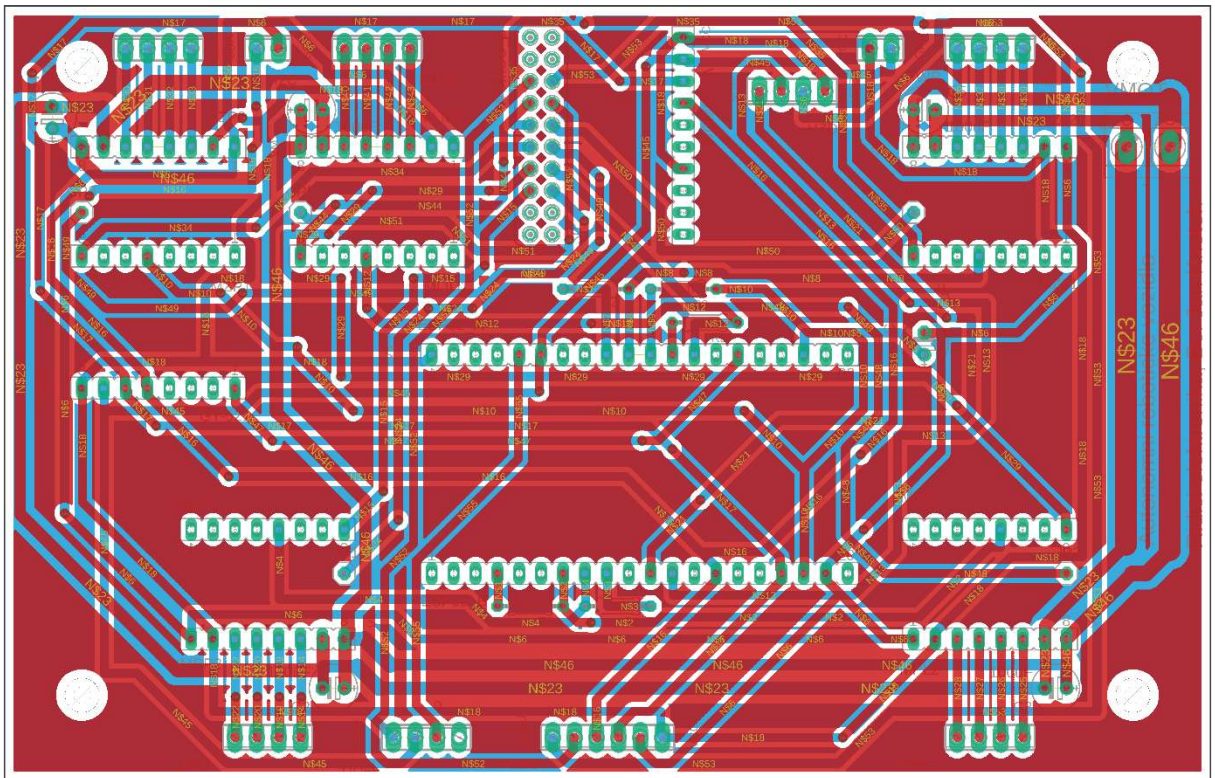
Obr. 2.8 – Osazovací plán řídicí DPS

Následně bylo třeba propojit vodivé cesty a upravit jejich šířku tam, kde to bylo třeba (obr. 2.7).



Obr. 2.7 – Navržená řídicí DPS

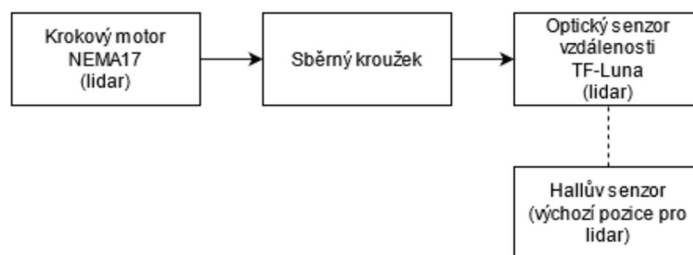
Pro snížení možnosti rušení byla DPS navržena s vylitým polygonem (obr. 2.9).



Obr. 2.9 – Navržená řídicí DPS s vylitým polygonem

#### 2.1.4 Návrh lidaru

Lidar je zkonstruován z krokového motoru NEMA 17, jehož hřídel je přes sběrný kroužek přivedena na optický senzor vzdálenosti TF-Luna. Sběrný kroužek zajišťuje přenos signálu tak, aby nedošlo k poškození vodičů při otáčivém pohybu motoru. Pro správnou funkci lidaru je nutné znát výchozí pozici natočení senzoru. Pro tento účel je využito Hallůva senzoru, který umožňuje detekci magnetického pole. Senzor vzdálenosti TF\_Luna je umístěn v plastové konstrukci, do níž je zabudován magnet. Při otáčení lidaru je v určitém okamžiku magnet dostatečně přiblížen k Hallůvu senzoru tak, že jej může detekovat. Tím je dosaženo výchozí pozice. Blokové schéma lidaru zobrazuje obr. 2.10.

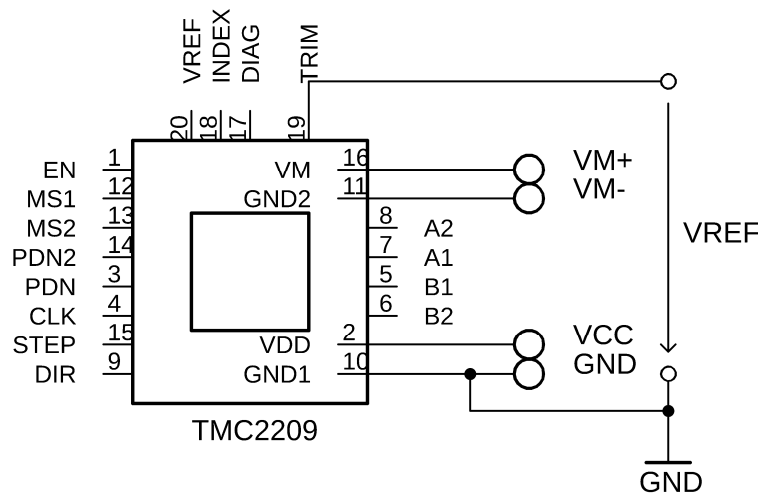


Obr. 2.10 – Blokové schéma lidaru

### 2.1.5 HW nastavení řadičů krokových motorů

Před začátkem práce s krokovými motory je třeba ručně nastavit proudové omezení řadičů. Každý krokový motor pracuje se specifickým proudem, proto je třeba řadiči umožnit práci s touto hodnotou proudu. Toho je dosaženo měřením a nastavováním referenčního napětí na trimru řadiče. Toto nastavování probíhá různě, vždy podle typu použitého řadiče. Princip je však stejný, pouze se liší vzorec pro výpočet proudu.

Nastavování maximálního proudu řadičem probíhá při připojení napájecího napětí (VCC) a napětí potřebném pro chod motorů (VM), avšak bez připojených motorů. Nastavení probíhá tak, že se odporovým trirem ladí referenční napětí (VREF) proti zemi viz obr. 2.11. Použitý řadič TMC2209 disponuje přímo vývodem VREF, není tedy nutné při nastavování proudového omezení držet voltmetr ve středu trimru, ale je možné napětí změřit přímo na pinu VREF.



Obr. 2.11 – Schéma pro nastavení proudového omezení řadiče TMC2209

Obr. 2.11 představuje schéma pro nastavení proudového omezení. Napětí motorů (VM) je dáno specifickou hodnotou pro daný motor. V případě použitého motoru NEMA 17 se jedná o 12 V. Napájecí napětí řadiče (VCC) pak představuje hodnotu napětí od 3,3 V do 5 V.

Při nastavování proudového omezení se vychází ze vzorce

$$I_{RMS} = \frac{325mV}{R_{SENSE} + 20m\Omega} \cdot \frac{1}{\sqrt{2}} \cdot \frac{V_{VREF}}{2,5V}, \quad (2.1)$$

kde  $I_{RMS}$  – střední kvadratický proud, A,

$R_{SENSE}$  – snímací odpor,  $\Omega$ ,

$V_{VREF}$  – referenční napětí, V („BIGTREETECH TMC2209-V1.2”, 2021).

Střední kvadratický proud je dán typem motoru, v případě použitých motorů NEMA 17 se jedná o 1,5 A. Snímací odpor je specifický pro každý typ řadiče, v případě použitého TMC2209 se jedná o 110 mΩ. Při programování řadiče však lze tento odpor pomocí software změnit.

Vzhledem k tomu, že pro nastavení proudu je třeba znát referenční napětí, bude dále uvedeno vyjádření  $V_{VREF}$  ze vztahu (2.1)

$$V_{VREF} = \sqrt{2} \cdot 2,5 I_{RMS} \cdot \frac{R_{SENSE} + 20m\Omega}{325mV}. \quad (2.2)$$

Dosazením do vztahu (2.2) je dáno požadované referenční napětí

$$V_{VREF} = \sqrt{2} \cdot 2,5 \cdot 1,5 \cdot \frac{110m\Omega + 20m\Omega}{325mV} = 2,12 V. \quad (2.3)$$

Při opomenutí nastavení proudového omezení mohou nastat dva případy:

- motory nebudou plnit svoji funkci dle očekávání (proudové omezení na minimum),
- dojde k poškození řadiče, resp. motorů (proudové omezení na maximum).

Od výrobce by měl řadič dorazit s nastaveným proudovým omezením na minimum, první případ je tedy častější. Při nedostatečném proudu se sice motory roztočí, nicméně budou vykazovat malý točivý moment.

Naopak druhý případ by mohl nastat, pokud je proudové omezení nastaveno na maximum. Pak (obzvláště při použití motorů s malým  $I_{RMS}$ ) teoreticky dochází k poškození řadičů příp. motorů.

### 2.1.6 Práce s řadiči krokových motorů TMC2209

Řadič krokových motorů TMC2209 obsahuje řadu výhod oproti ostatním typům řadičů (např. oproti A4988 nebo DRV8825). Nicméně jeho správné nastavení je pracnější než u ostatních řadičů.

Řadič může pracovat ve standardních režimech STEP/DIR a UART („TMC2209 Datasheet”, 2021).

Režim STEP/DIR představuje řízení motorů pomocí software. Ve vývojovém prostředí je třeba správně generovat pulzy pro ovládání krokových motorů. STEP představuje krok a DIR

představuje směr otáčení motorů. Taktéž je možno využít signály INDEX a ERROR, které vracejí informace o stavu řadiče („TMC2209 Datasheet”, 2021).

Pro účely této práce bylo zvoleno druhého režimu. V režimu UART ovládá STEP a DIR signály CPU řadiče. Ten přistupuje k diagnostickým informacím a konfiguruje řadič skrze rozhraní UART. CPU pak obstarává řízení pohybu a řadič řídí krokový motor a optimalizuje výkon motoru („TMC2209 Datasheet”, 2021).

### **Pokročilé funkce řadiče TMC2209**

Řadič TMC2209 implementuje pokročilé funkce, které jsou exkluzivní pro produkty firmy TRINAMIC. Tyto funkce přispívají k větší přesnosti, spolehlivosti, vyšší energetické účinnosti a plynulejšímu pohybu motoru:

- StealthChop2™ - bezhlučný, vysoce přesný algoritmus pro neslyšitelný pohyb a neslyšitelné zastavení motoru,
- SpreadCycle™ - vysoce přesné řízení proudu umožňující velké dynamické pohyby,
- MicroPlyer™ - umožňuje využít 256 mikrokroků a tím zajišťuje plynulost pohybu,
- StallGuard4™ - umožňuje návrat motoru do původní pozice bez využití senzoru, chrání koncové spínače v případě přetížení motoru,
- CoolStep™ - využívá měření StallGuard k přizpůsobení proudu motoru pro nejlepší účinnost a nejnižší zahřátí motoru a řadiče („TMC2209 Datasheet”, 2021).

Kromě těchto vylepšení výkonu nabízejí řadiče motorů TRINAMIC ochranná opatření pro detekci a ochranu proti zkratovaným výstupům, přerušení výstupu, přehřátí a podpětí pro zvýšení bezpečnosti a zotavení z poruch zařízení („TMC2209 Datasheet”, 2021).

Z uvedených pokročilých funkcí řadiče TMC2209 je využíváno funkce StealthChop a CoolStep.

#### **StealthChop**

StealthChop je využíván při nízkých otáčkách motorů. Tato funkce zajišťuje téměř neslyšný chod motorů (včetně chodu i zastavení při nízkých otáčkách).

#### **CoolStep**

CoolStep je využíván pro snížení spotřeby elektrické energie vozidla. Pohání motor optimálním proudem. Ke své funkci využívá StallGuard, který měří zatížení motorů. Poté nastaví proud na minimální hodnotu požadovanou při daném zatížení. Tím je zajištěno šetření energie a udržování řadiče v chladu.



## **Diagnostika problémů s řadiči**

Řadiče TMC2209 rovněž obsahují diagnostické výstupy (DIAG a INDEX). Oba výstupy poskytují důležité informace o stavu řadiče. Aktivní výstup DIAG ukazuje, že řadič nemůže fungovat normálně nebo že je detekováno zablokování motoru. Výstup INDEX pak signalizuje nulovou pozici čítače mikrokroků („TMC2209 Datasheet“, 2021).

V této práci je využito pouze výstupu DIAG, který následující informace:

- detekce zablokování motoru,
- reset při zapnutí,
- informace o podpětí,
- informace o přehřátí řadiče,
- informace o zkratu („TMC2209 Datasheet“, 2021).

Výstup DIAG je přiveden jako digitální vstup přes rozšiřující I/O desku k řídicí jednotce ESP32, která pozoruje tento vstup a v případě jeho aktivace upozorní na možný výskyt problému s řadičem.

## **2.2 NÁVRH SOFTWARE VOZIDLA**

Jádrem projektu je vývojová deska ESP32, kterou je nutné příslušným způsobem naprogramovat. Program je tvořen v prostředí Arduino IDE. Jako způsob programování bylo zvoleno objektově orientované programování. Program umožňuje vozidlu plnit jeho úkoly – zajišťuje tedy možnost ovládání motorů, získávání údajů ze senzorů, umožňuje vozidlu komunikovat s GUI aplikací apod. Mezi jeho nejdůležitější úkoly patří možnost autonomní jízdy. Tato kapitola obsahuje popis software vozidla. Převážně se jedná o popis programového kódu.

### **2.2.1 Popis programového kódu vozidla**

Programový kód se skládá ze tří hlavních částí. První částí je deklarace tříd, druhou část tvoří funkce setup a třetí částí je funkce loop. V této kapitole budou všechny tři části kódu popsány.

#### **Deklarace tříd**

Program obsahuje řadu tříd. Každá třída obsahuje konstruktor, metodu setup a další metody třídy. V konstruktoru dochází k vytvoření potřebných atributů, v metodě setup pak k potřebné konfiguraci (například definici pinů, vlastností řadiče, senzorů apod.).

Využité třídy a jejich metody (vyjma konstruktorů a metod s názvem setup):

- Engine – zajišťuje chování motoru, obsahuje ukazatel na objekt softwarového sériového portu, rozšiřujícího I/O modulu a řadiče motoru. Třída je tvořena následujícími metodami:
  - Start – spuštění řadiče krokového motoru,
  - Stop – zablokování řadiče krokového motoru,
  - SetSpeed – nastavení požadované rychlosti motoru vozidla,
  - SpeedAct – aktualizace rychlosti vozidla,
  - CheckDiag – kontrola diagnostického výstupu viz kapitola 2.1.6,
  - Test\_connection – kontrola navázání komunikace s řadiči,
  - Disp – výpis dat o řadiči krokového motoru.
- I2C\_Scan – umožňuje zjistit I<sup>2</sup>C adresu využitých komponent.
- ClFrontDistance – zajišťuje obsluhu předního senzoru vzdálenosti, obsahuje odkaz na objekt rozšiřujícího I/O modulu a na ovladače senzoru. Třída je tvořena následujícími metodami:
  - Disp – výpis údaje o vzdálenosti předního senzoru,
  - CorrectSpeed – úprava rychlosti vozidla dle vzdálenosti předního senzoru.
- ClAstar – zajišťuje A\* algoritmus (více viz kapitola 2.2.2) a export dat do GUI aplikace (více viz 2.2.3 ). Třída je tvořena následujícími metodami:
  - SetDest – nastavení cíle,
  - MapReset – resetování mapy,
  - MapToStates – uložení mapy do pole „States“,
  - MapPrint – výpis mapy,
  - ASearch – průběh A\* algoritmu,
  - CreatePathToDest – vytvoření cesty do cíle,
  - StraightPathFree – testování „přímé viditelnosti vozidla“ (viz 2.2.2),
  - MapUpdatePoint – aktualizace bodu mapy,
  - MapUpdate – aktualizace mapy okolí vozidla, cesty a cíle,
  - AngleNextPath – určení úhlu pro natočení vozidla do směru následující jízdy,
  - PathAllExport a další metody pro export dat do GUI aplikace.
- ClLidar – zajišťuje vytvoření mapy okolí vozidla pomocí laserového snímače vzdálenosti. Obsahuje ukazatel na rozšiřující I/O modul a na ovladač senzoru TF-Luna. Třída je tvořena následujícími metodami:
  - Disp – výpis stavu laserového senzoru,

- PointfromPolar – zajišťuje vložení bodu získaného lidarem ve formě polárních souřadnice,
- Filter – filtrace osamocených bodů,
- Scan – zajišťuje skenování okolí vozidla (využívá motor lidaru),
- DataExportBegin a další metody pro export dat do GUI aplikace.
- CIDirectionControl – zajišťuje práci s gyroskopem, obsahuje ukazatel na objekt gyroskopu, PID regulátoru a rozšiřujícího I/O modulu. Třída je tvořena následujícími metodami:
  - GyroDisplayDetails – zobrazuje data o senzoru (jméno, adresu apod.),
  - GyroDisplayStatus – zobrazuje vnitřní data o senzoru (stav, chyby apod.),
  - GyroDisplayCalStatus – zobrazuje data o kalibraci senzoru,
  - GyroDisplayOrientation – zobrazuje data poskytnuté senzorem (souřadnice),
  - GyroGetZOrientation – získání informace o úhlu natočení vozidla,
  - Start – odstartování regulace úhlu,
  - Stop – zakázání natáčení,
  - Run – periodicky spouštěná metoda, zajišťuje načítání a regulaci úhlu natočení vozidla. Zde je využito PID regulátoru.
- EngineControl – zajišťuje ovládání motorů pro pohyb vozidla, obsahuje ukazatele na objekty třídy Engine. Třída je tvořena následujícími metodami:
  - Start – spuštění motorů,
  - Stop – zastavení motorů,
  - CheckDiag – kontrola diagnostického výstupu viz kapitola 2.1.6 ,
  - SetSpeed – nastavení požadované rychlosti motorů vozidla,
  - SpeedAct – aktualizace rychlosti vozidla,
  - Run – start pohybu vozidla.
- CIMeasurementUIA – zajišťuje obsluhu měřičů proudu, obsahuje ukazatel na objekt tohoto měřiče proudu (INA226). Třída je tvořena následujícími metodami:
  - CheckConfig – zajišťuje konfiguraci měřiče proudu,
  - Disp – výpis proudu, napětí a výkonu.
- CIDisplay – zajišťuje ovládání OLED displeje, obsahuje ukazatel na objekt tohoto displeje. Třída je tvořena metodou UpdateDisplay, která zajišťuje výpis údajů o vozidle na displej.

## **Funkce setup**

Funkce setup je vestavěná funkce prostředí Arduino IDE a vykoná se pouze jednou. V této funkci se volají jednotlivé metody Setup příslušných tříd. Dochází tedy k počáteční konfiguraci atributů.

V této funkci se také pracuje s dynamickým alokováním paměti. Dochází k alokování určitého počtu bytů pro získání většího množství paměti potřebného pro práci A\* algoritmu.

## **Funkce loop**

Funkce loop je další vestavěná funkce prostředí Arduino IDE, na rozdíl od funkce setup se jedná o nekonečnou smyčku. Tato funkce obsahuje dvě hlavní části – příjem příkazů a provádění opakovaných činností.

### **Příjem příkazů**

Příjem příkazů je možný sériovou linkou (v případě připojeného ESP32 pomocí USB kabelu k prostředí Arduino IDE) pomocí sériového monitoru nebo pomocí Wifi při využití protokolu TCP/IP. Pro ladění kódu lze využít sériové linky, avšak pro autonomní pohyb vozidla je využívána komunikace WiFi.

Tato část kódu se zabývá zpracováním získaných dat – přijímá a obsluhuje příkazy (např. spuštění motorů, zobrazování dat na displeji apod.).

### **Provádění opakovaných činností**

Mezi opakovaně prováděné činnosti se řadí tři nejdůležitější činnosti. První z nich je PID regulace úhlu natočení vozidla. Ta se provádí s frekvencí 10 Hz a zajišťuje tak natočení vozidla do požadované pozice. Existují zde dvě sady parametrů – pro jízdu a stání vozidla. Dále se zde zajišťuje se stejnou frekvencí ujetí požadované vzdálenosti.

Třetí opakovanou činností je již samotný pohyb vozidla, který se řídí tzv. sekvenčním automatem. Jedná se o strukturu switch – case, která umožňuje definovat jednotlivé fáze jízdy vozidla.

Sekvenční automat autonomního pohybu vozidla obsahuje následující stavy:

0. čekání na příkaz,
1. skenování okolí,
2. provádění A\* algoritmu,
3. start otáčení vozidla,
4. otáčení vozidla,
5. start přímé jízdy,

6. přímá jízda,
7. kontrola cíle,
8. náhlé zastavení,
9. dosaženo cíle + čekání na příkaz,
10. nedosaženo cíle + čekání na příkaz,
11. čekání po otočení vozidla,
12. start otáčení vozidla na nejbližších 90 ° v následujícím směru jízdy,
13. otáčení na nejbližších 90 ° v následujícím směru jízdy,
14. čekání po otočení 90 °,
15. opakované skenování.

Detailní průběh činnosti vozidla dle tohoto stavového automatu je popsán v kapitole 2.3.2.

### 2.2.2 Hledání nejkratší cesty vozidla k cíli

Pro hledání nejkratší cesty vozidla k cíli je využito algoritmu A\*, který je implementován v prostředí Arduino IDE. Algoritmus využívá údajů z lidarů. Okolo bodu, poskytnutého lidarem, je vytvořen čtverec o straně 30 cm. Tento čtverec tak tvoří „překážku“ a vozidlo se jí musí vyhnout.

Jelikož algoritmus A\* představuje informovanou metodu prohledávání stavového prostoru, je dostupná určitá heuristika umožňující odhad ceny cesty do cílového stavu. K heuristickému odhadu je možné využít řadu metrik. Každá metrika poskytuje určité výhody. Mezi nejběžnější z nich patří Manhattanová metrika a Euklidovská metrika.

Manhattanová metrika je definována jako

$$h(v_i) = |x_1 - y_1| + |x_2 - y_2|, \quad (2.4)$$

Euklidovská metrika je definována jako

$$h(v_i) = (x_1 - y_1)^2 + (x_2 - y_2)^2, \quad (2.5)$$

kde  $h(v_i)$  - odhad ceny cesty z vrcholu  $v_i$  do nejbližšího cílového vrcholu (stavu),

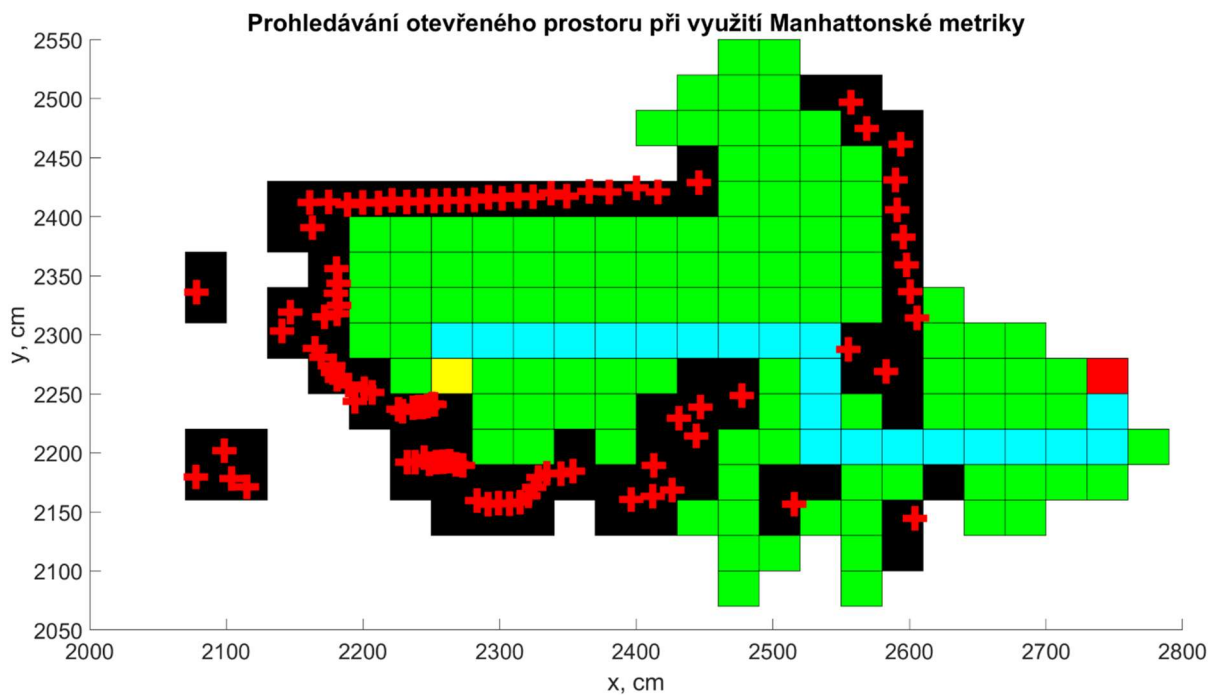
$x_1$ - souřadnice x počátečního bodu, cm,

$y_1$ - souřadnice y počátečního bodu, cm,

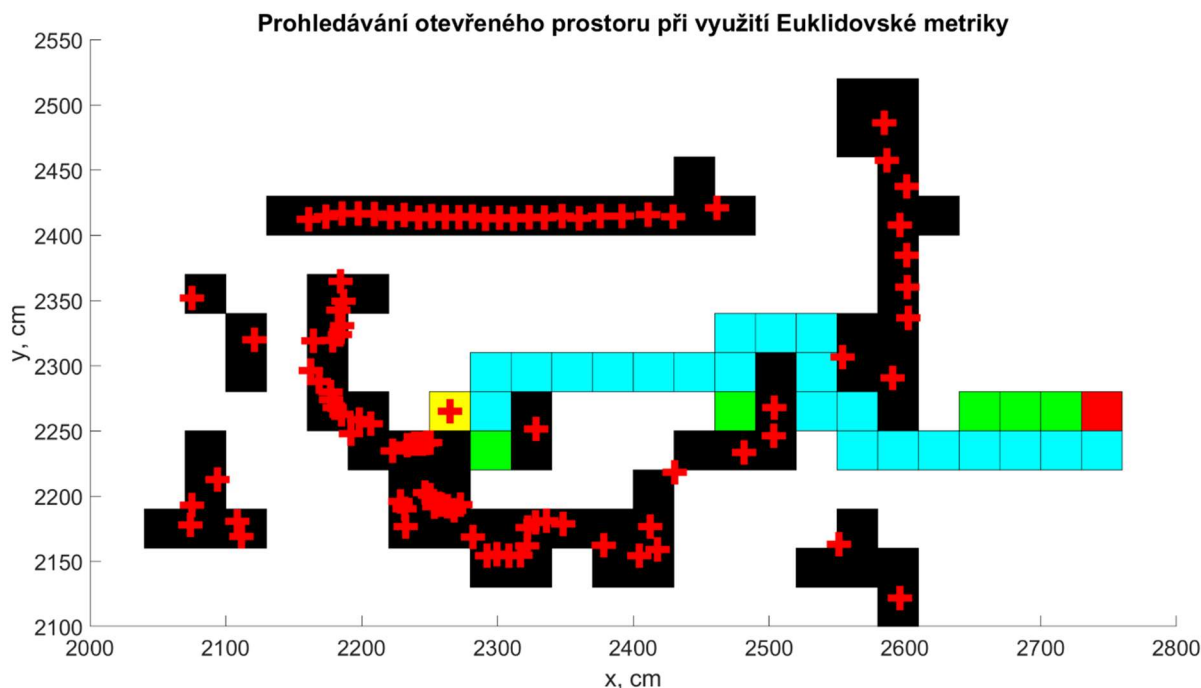
$x_2$ - souřadnice x koncového bodu, cm,

$y_2$ - souřadnice y koncového bodu, cm.

Manhattanská metrika nachází uplatnění v úzkých uzavřených prostorách mezi ně patří např. bludiště. Naopak Euklidovská metrika se lépe uplatňuje v otevřených prostorech. Pro porovnání rozdílu využití těchto metrik v prohledávání otevřeného prostoru je na obr. 2.12 a obr. 2.13 uvedena mapa okolí lidarů vnitřního pokoje budovy.



Obr. 2.12 – Prohledávání otevřeného prostoru při využití Manhattanové metriky



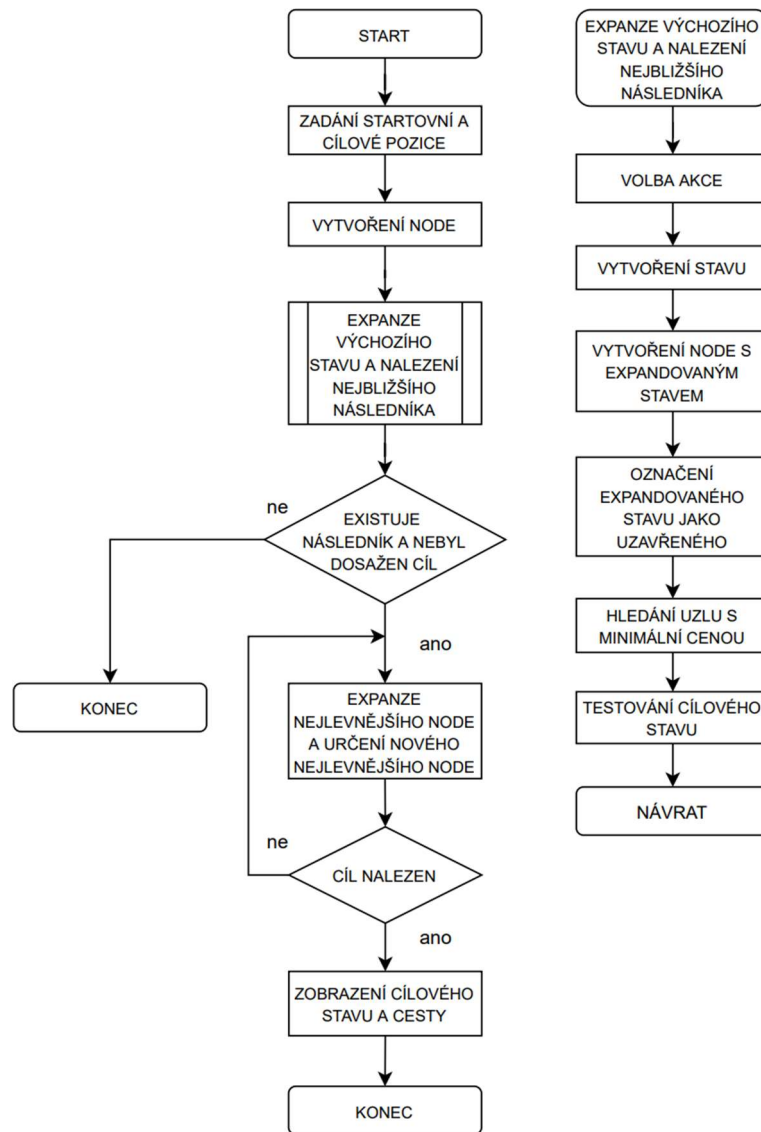
Obr. 2.13 – Prohledávání otevřeného prostoru při využití Euklidovské metriky

Obr. 2.12 a obr. 2.13 zobrazují prohledávání otevřeného prostoru pomocí již zmíněných metrik. Na obrázcích jsou vidět čtverce a „plusy“. Červené „plusy“ představují data z lidarů, konkrétně body naměřené senzorem vzdálenosti TF-Luna. Černé čtverce o délce strany 30 cm okolo nich pak představují potenciální překážku pro vozidlo. Žlutý čtverec signalizuje vozidlo. Červený čtverec pak ukazuje cílovou pozici. Zelené a modré čtverce jsou spjaty s prohledáváním otevřeného prostoru. Zelené čtverce představují prostor, který algoritmus A\* prohledal, než našel cíl. Modré čtverce pak představují nejkratší cestu k cíli. Po této trase by se pak vozidlo vydalo.

Z těchto obrázků je patrné, že prohledávání otevřeného prostoru při využití Euklidovské metriky je značně efektivnější a rychlejší, proto byla tato metrika využita pro nalezení nejkratší cesty vozidla k cíli.

## Průběh hledání nejkratší cesty vozidla k cíli

Na obr. 2.14 je uveden vývojový diagram popisující průběh algoritmus A\* pro hledání nejkratší cesty vozidla k cíli.



Obr. 2.14 – Vývojový diagram A\* algoritmu

Na počátku prohledávání stavového prostoru je třeba zjistit startovní pozici. Ta je zjištěna za pomoci lidarů. Po jedné (ne celé) otočce lidarů dochází ke zjištění počáteční pozice vozidla. Následně je třeba v aplikaci nastavit cílovou pozici. Cílová pozice je dána vzdáleností k cíli (v centimetrech) a úhlem (ve stupních), pod kterým se cíl nachází. Dalším krokem je vytvoření node.

Node je struktura obsahující:

- jednoznačný identifikátor vrcholu,
- stav,



- identifikátor bezprostředního předchůdce vrcholu,
- akce, která vedla k přechodu z rodiče na současný vrchol,
- cena cesty z kořene (počátečního stavu) k tomuto stavu,
- odhad ceny cesty z počátečního vrcholu do nejbližšího cílového stavu.

Následně je provedena metoda zajišťující expanzi výchozího stavu a nalezení nejbližšího následníka. V této metodě dochází k volbě akce (dopředu, doprava, dozadu, doleva). Po zvolení akce dochází k vytvoření stavu. Zde se určuje cena jednotlivé akce a vypočítává se hodnotící funkce, tedy celková cena. Ta je dána součtem ceny z počátečního do aktuálního vrcholu a heuristického odhadu ceny z aktuálního vrcholu do cíle. K tomuto odhadu je využita již zmíněná Euklidovská metrika. Po jednotlivém provedení všech přístupných akcí dochází k vytvoření node s expandovaným stavem. Následně je expandovaný stav označen jako uzavřený a dále se již neprohledává. Poté již dochází k hledání otevřeného uzlu s minimální cenou. Poslední částí metody je test dosažení cílového stavu.

Průběh hledání cílového stavu pokračuje, dokud existuje nějaký následník a nebyl dosažen cíl. Do té doby pravidelně dochází k expanzi nejlevnějšího node a určování nového nejlevnějšího node. Když dojde k nalezení cíle dochází k zobrazení cílového stavu a cesty.

Vozidlo se tímto algoritmem řídí, nicméně nekopíruje přímo cestu vytvořenou algoritmem, jelikož by jezdilo pravoúhle. Při výběru dočasného cíle se zaměřuje na místo, které je v algoritmem vygenerované cestě nejbližší k cíli a zároveň je přímo viditelné. Tím je zajištěna optimální trasa vozidla.

### 2.2.3 GUI aplikace

Pro ovládání vozidla a vizualizace dat byla vytvořena GUI aplikace za pomoci AppDesigneru MATLABu. Řídící jednotka ESP32 komunikuje s aplikací prostřednictvím WiFi pomocí TCP/IP protokolu. ESP32 tvoří server a aplikace se stává klientem. IP adresu řídicí jednotky přiděluje DHCP server.

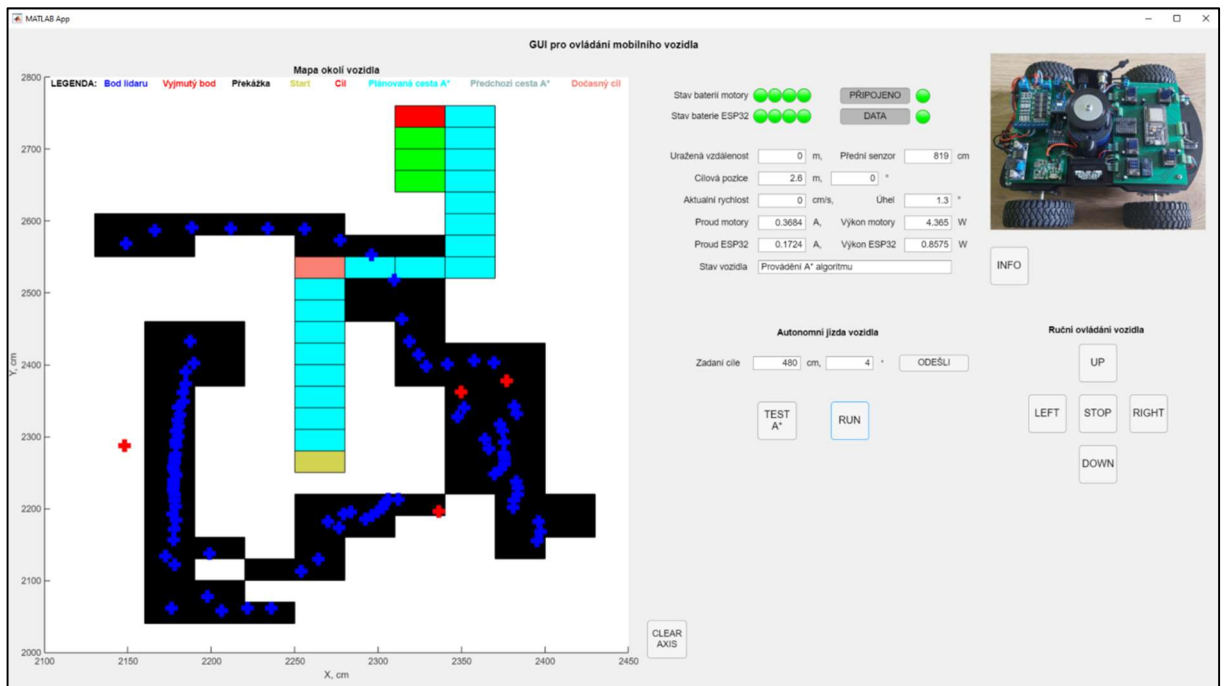
Aplikace zajišťuje základní činnosti potřebné pro ovládání vozidla – připojení a odpojení TCP/IP klienta, příjem dat, odeslání informace o cílovém stavu apod. Dále obsahuje tlačítka pro ruční ovládání vozidla, ukazatele pro zobrazení dat o vozidle a mapu okolí vozidla.

Mapa okolí vozidla obsahuje následující údaje:

- bod lidarů (modré „plus“),
- vyjmutý resp. vyfiltrovaný bod (červené „plus“),
- překážka (černý čtverec),

- start (zlatý čtverec),
- cíl (červený čtverec),
- plánovaná cesta A\* algoritmu (světle modrý čtverec),
- předchozí cesta A\* algoritmu (šedý čtverec),
- dočasný cíl (běžový čtverec).

Vzhled aplikace ukazuje obr. 2.15.



Obr. 2.15 – GUI aplikace pro ovládání mobilního vozidla

## Funkce aplikace

Aplikace umožňuje ovládání vozidla a vizualizaci jeho dat. Aplikace reaguje na události (stisk tlačítka, změnu stavu apod.). Při spuštění aplikace je nutné se připojit k TCP/IP serveru, tedy k ESP32. Jakmile dojde k úspěšnému připojení, rozsvítí se indikační LED.

Následně je třeba umožnit příjem dat (opět úspěšný příjem dat je indikován pomocí LED). Data jsou přijímána ve formě kódu. Data obsahují nejrůznější informace o stavu vozidla. Ty jsou pak v aplikaci extrahovány a zobrazeny v editovatelných oknech aplikace. Okolí vozidla je zobrazeno pomocí grafu.

Následně uživatel volí, zda chce vozidlo ovládat v manuálním nebo autonomním režimu. Při ručním ovládání je umožněno uživateli aplikace ovládat vozidlo pomocí tlačítek (příkazy rovně, doprava, dolů, doleva a zastav).

V autonomním režimu je potřeba zadat cílový bod – a to ve formě vzdálenosti k cíli (v centimetrech) a úhlu, pod jakým se cíl nachází (ve stupních). Pomocí tlačítka se údaj odešle

do řídicí jednotky a ta jej zpracuje. Lze také využít tlačítko pro testování A\* algoritmu, které zajistí otočení lidarů a vykreslení mapy. Samotný autonomní pohyb vozidla se spouští tlačítkem RUN.

Nezávisle na zvoleném režimu je možné tlačítkem INFO vyslat data do sériového monitoru v prostředí Arduino IDE (při propojení USB kabelem). Tato data obsahují informace o stavu motorů, gyroskopu, předního senzoru vzdálenosti apod. Slouží tedy ke kontrole správné funkce vozidla.

## **2.3 PRŮBĚH JÍZDY VOZIDLA**

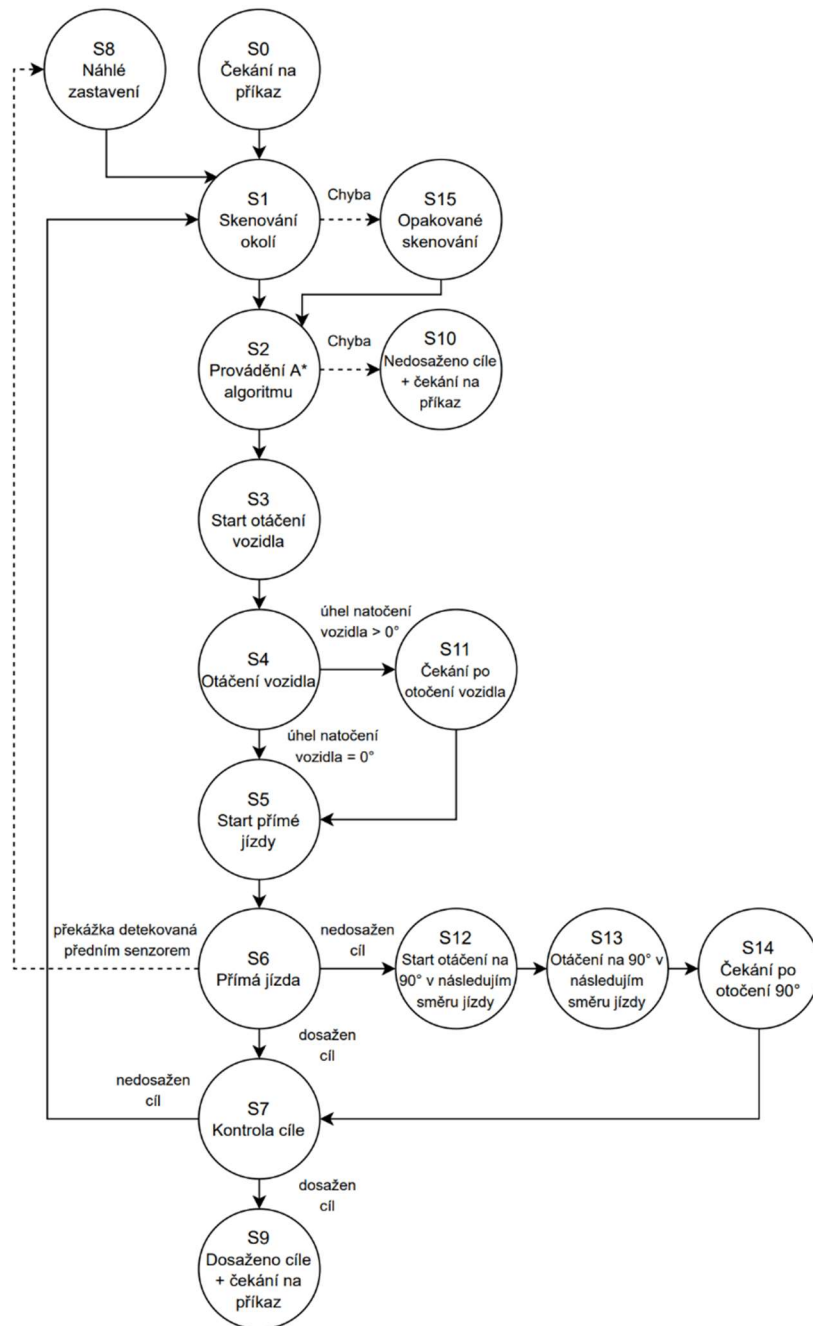
Vozidlo může pracovat v manuálním nebo autonomním režimu.

### **2.3.1 Manuální režim**

Vozidlo v manuálním režimu je možné ovládat z aplikace. Možnosti vozidla jsou standardní. Vozidlo umožňuje jízdu vpřed, zatáčení, zrychlení, zpomalení a zastavení. Mapování okolí je možné spustit, když je vozidlo zastaveno.

### **2.3.2 Autonomní režim**

Vozidlo v autonomním režimu je schopné jízdy z bodu A do bodu B nejkratší možnou cestou za pomoci již zmíněného A\* algoritmu. Činnost vozidla je řízena sekvenčním automatem uvedeným v kapitole 2.2.1. Tomuto automatu odpovídá diagram přechodů (obr. 2.16).



Obr. 2.16 – Diagram přechodů sekvenčního automatu autonomního pohybu vozidla

Vozidlo po zapnutí napájení očekává příkazy. Nachází se v stavu 0 – čekání na příkaz. Po zadání cílového bodu dochází ke skenování okolí (stav 1). V této části programu vozidlo pomocí lidarů skenuje terén a určuje výchozí pozici. V tomto okamžiku také dochází ke generování mapy okolí vozidla. Poté dochází k provádění A\* algoritmu (stav 2). Jakmile vozidlo zjistí informaci o „dočasném“ cíli (vozidlo za dočasný cíl považuje bod, který je nejbližší k cíli a zároveň na něj vozidlo „vidí“), dojde k nasměrování vozidla právě k tomuto cíli (stavy 3 a 4). Pokud se vozidlo nenatočilo, pokračuje se startem přímé jízdy (stav 5). Pokud

se natočilo, pokračuje se přes krok 11, kde je implementována pauza (1 s) pro dokončení natáčení. Čekání je prováděno proto, aby se vozidlo stačilo ustálit na požadované pozici (tato pozice je regulována PID regulátorem). Dalším krokem je již přímá jízda k dočasnému cíli (stavy 5 a 6). Pro samotnou jízdu vozidlo využívá „rampy“. Při rozjíždění vozidla dochází k postupnému zvyšování rychlosti. Jakmile vozidlo dosáhne provozní rychlosti (5000 mikrokroků/s), jede konstantní rychlostí. Před dočasným cílem opět dochází k postupnému snižování rychlosti tak, aby vozidlo zastavilo na požadované pozici. Pokud vozidlo dojede do cíle, pokračuje se do stavu 7 (kontrola cíle) a přejde se do stavu 9 (dosaženo cíle + čekání na příkaz). Pokud se však nejedná o cíl, přechází se do stavu 12 (start otáčení vozidla na nejbližších 90 ° v následujícím směru jízdy), poté do stavu 13 (otáčení vozidla na nejbližších 90 ° v následujícím směru jízdy) a 14 (čekání po otočení 90 °). Poté se přejde do stavu 7 (kontrola cíle), provede se natočení mapy a cíle a přejde se do stavu 1 – dochází ke znovuspuštění algoritmu. Dochází tedy k novému skenování okolí a provádění algoritmu. Následně se vozidlo rozjede k dalšímu cíli. Pokud se již jedná o finální cílový stav, vozidlo zastaví a přejde do režimu čekání na pokyny (stav 9).

Pokud během jízdy vozidla dojde k chybě (například k překročení určité velikosti pole nebo když je cíl nedostupný), program přejde do stavu 10 (nedosaženo cíle + čekání na příkaz).

Během jízdy vozidla se též může stát, že se před vozidlem náhle objeví překážka. V tom případě se přejde do stavu 8 – náhlé zastavení. Dojde k zakreslení překážky do mapy, natočení mapy a vozidlo pokračuje dalším skenováním okolí dále (stav 1).

Poslední případ, který může nastat, se týká stavu 1. Když čas skenování není dle očekávání (tzn. že například lidar nebyl ve výchozím stavu nebo selhal Hallův senzor), přejde se do stavu 15, kde se provádí to samé jako ve stavu 1, nicméně se jedná o jiný krok. Důvodem je existence možnosti zastavení vozidla právě například při chybě Hallova senzoru, kdy by se lidar neustále točil.

### **2.3.3 Vizualizace dat o vozidle**

Při zapnutém napájení dochází k vizualizaci dat o vozidle. Informace jsou dostupné na OLED displeji na vozidle, ale také v aplikaci vytvořené v AppDesingeru MATLABu. Pro ladění kódu je možné si data nechat také zobrazit pomocí sériového monitoru v prostředí Arduino IDE.

První dva případy zobrazují následující informace o vozidle:

- ukazatele nabití baterie – odděleně pro elektroniku a motory,

- uražená vzdálenost,
- vzdálenost a úhel k cíli,
- vzdálenost předního senzoru,
- aktuální rychlost a úhel,
- proud motory a ESP32,
- výkon motorů a ESP32,
- stav vozidla.

Na obr. 2.17 je zobrazen OLED displej, který zobrazuje data o vozidle.



Obr. 2.17 – Vizualizace dat na OLED displeji

Dále jsou data dostupné v GUI aplikaci (obr. 2.18).



Obr. 2.18 – Vizualizace dat v GUI aplikaci

Při propojení vozidla s prostředím Arduino IDE pomocí USB kabelu lze zobrazovat data pomocí sériového monitoru, což je velmi výhodné při ladění programového kódu.

Po startu programu dochází k výpisu údajů o připojených komponentách (obr. 2.19). K dispozici jsou například informace o I<sup>2</sup>C adresách jednotlivých zařízení, o navázání spojení s řadiči motorů, připojení k síti WiFi apod.

```
16:02:12.696 ->
16:02:12.696 -> START =====
16:02:12.696 -> PSRAM does not work
16:02:12.696 -> Heap for malloc: 217792
16:02:12.696 -> Heap after malloc: 211764
16:02:12.696 -> =====EngineControl =====
16:02:12.977 -> =====Front distance laser sensor initialize ...
16:02:12.977 -> Scanning I2C. SDA pin:21, SCL pin:22 ... 0x0F, 0x20, 0x29, 0x3C, 0x40, 0x41, done
16:02:12.977 -> Change I2C address to 0xF
16:02:13.196 -> Scanning I2C. SDA pin:21, SCL pin:22 ... 0x0F, 0x20, 0x29, 0x3C, 0x40, 0x41, done
16:02:13.243 -> Long range mode is not active.
16:02:13.243 -> Normal mode is active, timing budget 33 ms. No high speed and No high accuracy.
16:02:13.243 -> OK
16:02:13.243 -> =====Gyroscope BNO055 initialize ...
16:02:13.680 -> BNO055 is detected OK.
16:02:14.680 -> Sensor: BNO055
16:02:14.680 -> I2C address: 0x29
16:02:14.680 -> Driver Ver: 1
16:02:14.680 -> Unique ID: 55
16:02:14.680 -> Max Value: 0.00 xxx
16:02:14.680 -> Min Value: 0.00 xxx
16:02:14.680 -> Resolution: 0.01 xxx
16:02:14.915 -> System Status: 0x5
16:02:14.915 -> Self Test: 0xF
16:02:14.915 -> System Error: 0x0
16:02:14.915 -> Gyroscope sensor calibration status (0..data ignored,3..Full calibrated):
16:02:14.915 -> System:0 Gyroscope:3 Accelerometer:0 Magnetometer:0
16:02:15.071 -> =====Testing connection to steppers ...
16:02:15.071 -> Testing connection to steppers: FrontLeft
16:02:15.071 -> Testing connection to steppers: RearLeft
16:02:15.117 -> Testing connection to steppers: RearRight
16:02:15.117 -> Testing connection to steppers: Lidar
16:02:15.117 -> =====Lidar laser sensor TFLuna initialize ...
16:02:15.117 -> Scanning I2C. SDA pin:18, SCL pin:19 ... 0x10, done
16:02:15.117 -> Device Address: 0x10
16:02:15.117 -> System Reset: Passed
16:02:15.617 -> Get Firmware Version: 3.0.9
16:02:16.102 -> Get Serial Number: T3300330012015
16:02:16.633 -> Get Time: 1336
16:02:17.633 -> Get Frame Rate: 0
16:02:17.633 -> =====Measurement Control =====
16:02:17.633 -> Device Address: 0x41
16:02:17.633 -> Mode: Shunt and Bus, Continuous
16:02:17.633 -> Samples average: 16 samples
16:02:17.633 -> Bus conversion time: 1.100ms
16:02:17.633 -> Shunt conversion time: 1.100ms
16:02:17.680 -> Max possible current: 0.82 A
16:02:17.680 -> Max current: 0.82 A
16:02:17.680 -> Max shunt voltage: 0.08 V
16:02:17.680 -> Max power: 29.49 W
16:02:17.680 -> =====Measurement Motors =====
16:02:17.680 -> Device Address: 0x40
16:02:17.680 -> Mode: Shunt and Bus, Continuous
16:02:17.680 -> Samples average: 16 samples
16:02:17.680 -> Bus conversion time: 1.100ms
16:02:17.680 -> Shunt conversion time: 1.100ms
16:02:17.680 -> Max possible current: 7.80 A
16:02:17.680 -> Max current: 6.55 A
16:02:17.680 -> Max shunt voltage: 0.07 V
16:02:17.680 -> Max power: 235.92 W
16:02:17.680 -> =====Display=====
16:02:17.680 -> Type SH1106. Device Address: 0x3C, display is sucesfully initialized.
16:02:17.727 -> =====WiFi connecting to LonkKnotUFC
16:02:17.822 -> MAC: C:11:E1:AB:62:24
16:02:18.305 -> .
16:02:18.305 -> WiFi connected.
16:02:18.305 -> IP address: 192.168.10.122
16:02:18.353 -> Motors auto disable.
```

Obr. 2.19 – Vizualizace dat po startu programu (séριοvý monitor)

Důležitá data pro kontrolu správné funkce zařízení můžeme získat z GUI aplikace po stisku tlačítka INFO. Dochází k zobrazení údajů dle obr. 2.20.

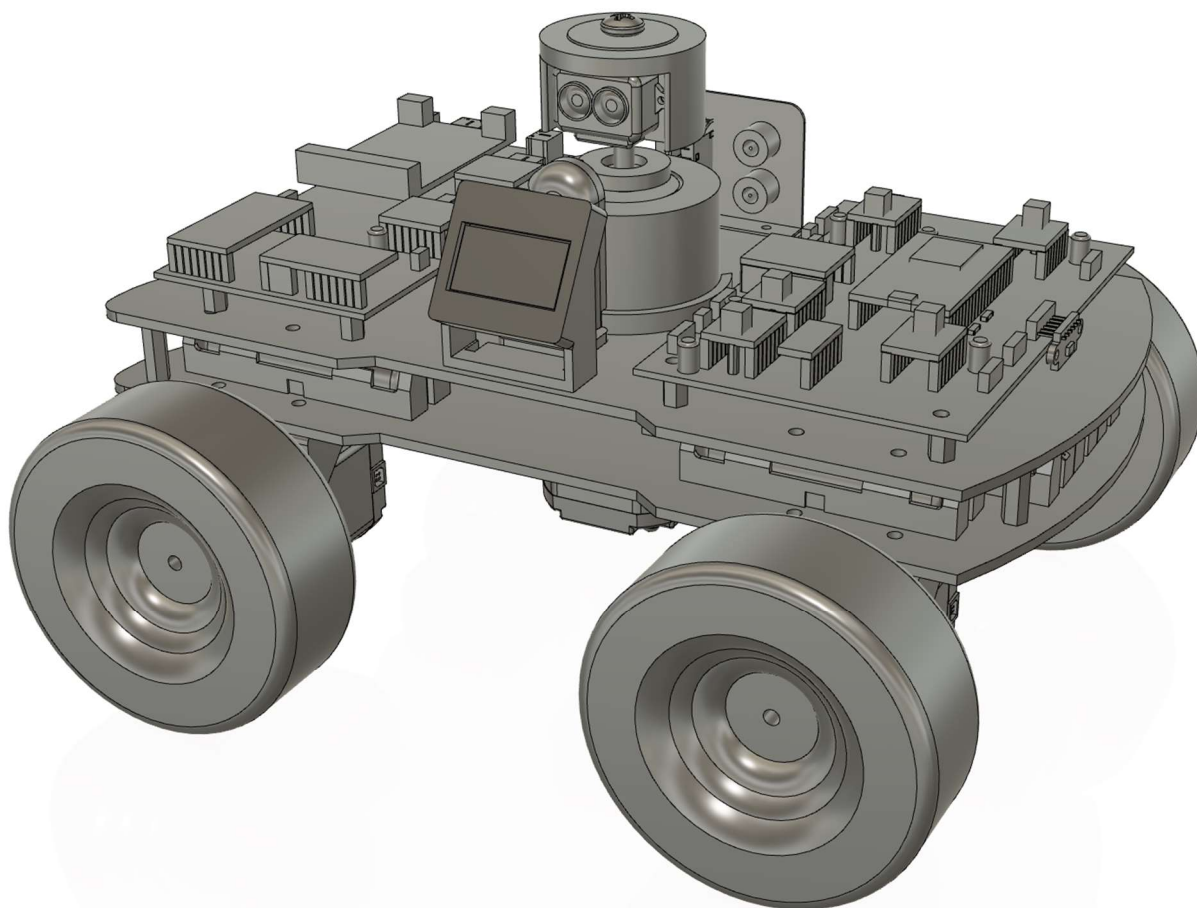
```
COM3
16:01:33.524 -> Data z serial monitoru: 9
16:01:33.524 -> Gyroscope sensor calibration status (0..data ignored,3..Full calibrated):
16:01:33.524 -> System:0 Gyroscope:3 Accelerometer:0 Magnetometer:0
16:01:33.524 -> Gyroscope orientation X: 359.8750 Y: -0.5625 Z: -0.8750
16:01:33.571 -> Measurement Control, Bus voltage: 4.98125 V, Bus power: 0.14000 W, Shunt voltage: 0.00282 V, Shunt current: 0.02820 A
16:01:33.571 -> Measurement Motors, Bus voltage: 11.98875 V, Bus power: 0.48000 W, Shunt voltage: 0.00042 V, Shunt current: 0.04020 A
16:01:33.571 -> Lidar laser sensor: Status: READY
16:01:33.571 -> Front distance [mm]: 843
16:01:33.571 -> FrontLeft --- Speed: 0, SG_RESULT: 0, STALL_VALUE: 50, SEMIN: 160, SE..MAX: 192, mA: 633, TSTEP: 1048575, TPFWMTHRS: 95, TBL: 1, msteps: 8
16:01:33.617 -> FrontRight --- Speed: 0, SG_RESULT: 0, STALL_VALUE: 50, SEMIN: 160, SE..MAX: 192, mA: 633, TSTEP: 1048575, TPFWMTHRS: 95, TBL: 1, msteps: 8
16:01:33.617 -> RearLeft --- Speed: 0, SG_RESULT: 0, STALL_VALUE: 50, SEMIN: 160, SE..MAX: 192, mA: 633, TSTEP: 1048575, TPFWMTHRS: 95, TBL: 1, msteps: 8
16:01:33.664 -> RearRight --- Speed: 0, SG_RESULT: 0, STALL_VALUE: 50, SEMIN: 160, SE..MAX: 192, mA: 633, TSTEP: 1048575, TPFWMTHRS: 95, TBL: 1, msteps: 8
16:01:33.664 -> Lidar --- Speed: 0, SG_RESULT: 0, STALL_VALUE: 50, SEMIN: 160, SE..MAX: 192, mA: 633, TSTEP: 1048575, TPFWMTHRS: 95, TBL: 1, msteps: 32
 Automatické scrollování  Zobrazit časové razítko Nová řádka (NL) 115200 baudů Vymazat výstup
```

Obr. 2.20 – Vizualizace dat po stisku tlačítka INFO (sériový monitor)

## 2.4 VZHLED A PARAMETRY VOZIDLA

### 2.4.1 Vzhled vozidla

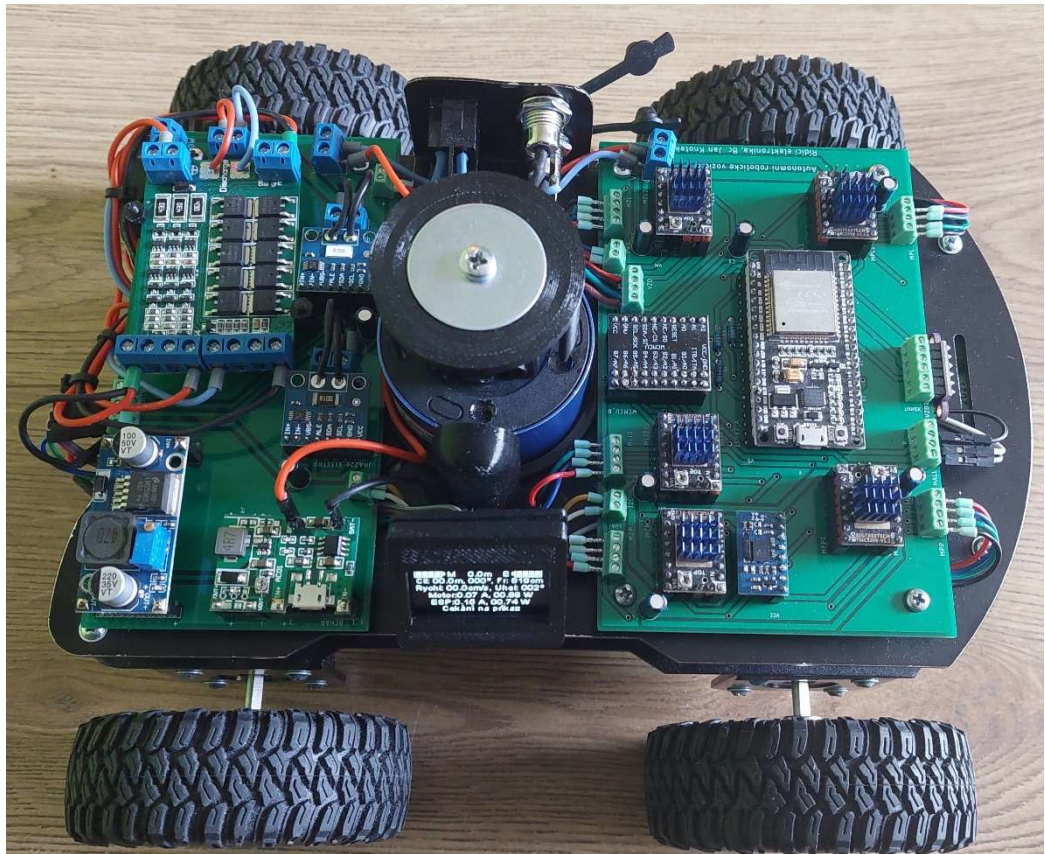
Pro představu o vzhledu vozidla slouží jeho 3D model (obr. 2.21). Model byl vytvořen v prostředí Autodesk Fusion 360.



Obr. 2.21 – 3D model vozidla



Reálnou konstrukci vozidla pak zachycuje obr. 2.22.



Obr. 2.22 – Zkonstruované vozidlo

Reálná konstrukce vychází 3D modelu vozidla a z popisu návrhu HW dle kapitoly 2.1.

#### 2.4.2 Parametry vozidla

V tab. 2.1 jsou uvedeny technické parametry zkonstruovaného vozidla. Při programování řídicí desky se využívají jako jednotky rychlosti mikrokroky za sekundu. Provozní rychlost vozidla je 5 000 mikrokroků/s. Pro názornější představu o rychlosti vozidla je tato rychlost přepočítána na metry za sekundu. Proud a výkon (spotřeba) je získávána z měřičů proudu INA226. Vzhledem k tomu, že spotřeba řídicí elektroniky vozidla je oproti spotřebě motorů prakticky zanedbatelná (jedná se řádově o jednotky procent), v tab. 2.1 jsou uvedeny hodnoty pouze pro motory.

Tab. 2.1 – Technické parametry vozidla

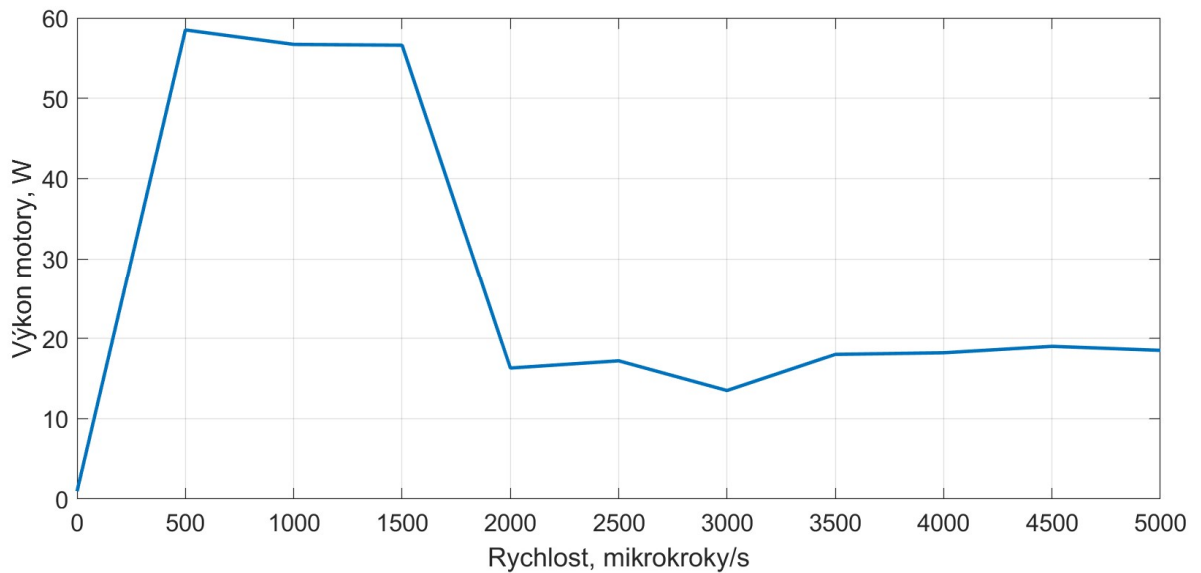
| <b>Parametr</b>                         | <b>Hodnota</b>                            |
|---|---|
| Rozměry                                 | délka 27 cm, šířka 24,5 cm, výška 18,5 cm |
| Hmotnost                                | 3,9 kg                                    |
| Provozní rychlost                       | 0,63 m/s                                  |
| Maximální rychlost                      | 1,89 m/s                                  |
| Proud při provozní rychlosti            | 1,6 A                                     |
| Spotřeba energie při provozní rychlosti | 18,5 W                                    |
| Klidový proud                           | 0,08 A                                    |
| Klidová spotřeba energie                | 0,97 W                                    |

Pro názornější představu o spotřebě elektrické energie a proudu tekoucí elektronikou a motory vozidla bylo provedeno měření. Toto měření bylo provedeno pro rozsah rychlostí vozidla 0 – 5 000 mikrokroků/s. Jeho výsledky zobrazuje tab. 2.2.

Tab. 2.2 – Záznam měření spotřeby elektrické energie vozidla

| <b>Rychlost,<br/>mikrokroky/s</b> | <b>Proud motory, A</b> | <b>Výkon motory,<br/>W</b> | <b>Proud ESP32, A</b> | <b>Výkon ESP32,<br/>W</b> |
|-----------------------------------|------------------------|----------------------------|-----------------------|---------------------------|
| 0                                 | 0,08                   | 0,97                       | 0,14                  | 0,72                      |
| 500                               | 5,70                   | 58,50                      | 0,51                  | 2,50                      |
| 1 000                             | 5,60                   | 56,70                      | 0,51                  | 2,53                      |
| 1 500                             | 5,48                   | 56,60                      | 0,51                  | 2,53                      |
| 2 000                             | 1,43                   | 16,30                      | 0,23                  | 1,20                      |
| 2 500                             | 1,51                   | 17,20                      | 0,24                  | 1,20                      |
| 3 000                             | 1,56                   | 13,50                      | 0,24                  | 1,20                      |
| 3 500                             | 1,60                   | 18,00                      | 0,24                  | 1,20                      |
| 4 000                             | 1,60                   | 18,20                      | 0,24                  | 1,20                      |
| 4 500                             | 1,70                   | 19,00                      | 0,25                  | 1,20                      |
| 5 000                             | 1,60                   | 18,50                      | 0,22                  | 1,20                      |

Obr. 2.23 ukazuje závislost spotřeby energie na rychlosti vozidla. Z grafu je zřejmé, že vozidlo v klidové pozici spotřebovává minimální množství energie. Při nízké rychlosti naopak dochází k nejvyššímu nárůstu spotřeby (až 58,5 W), která od 2 000 mikrokroků/s klesne na 16,3 W a poté mírně roste.

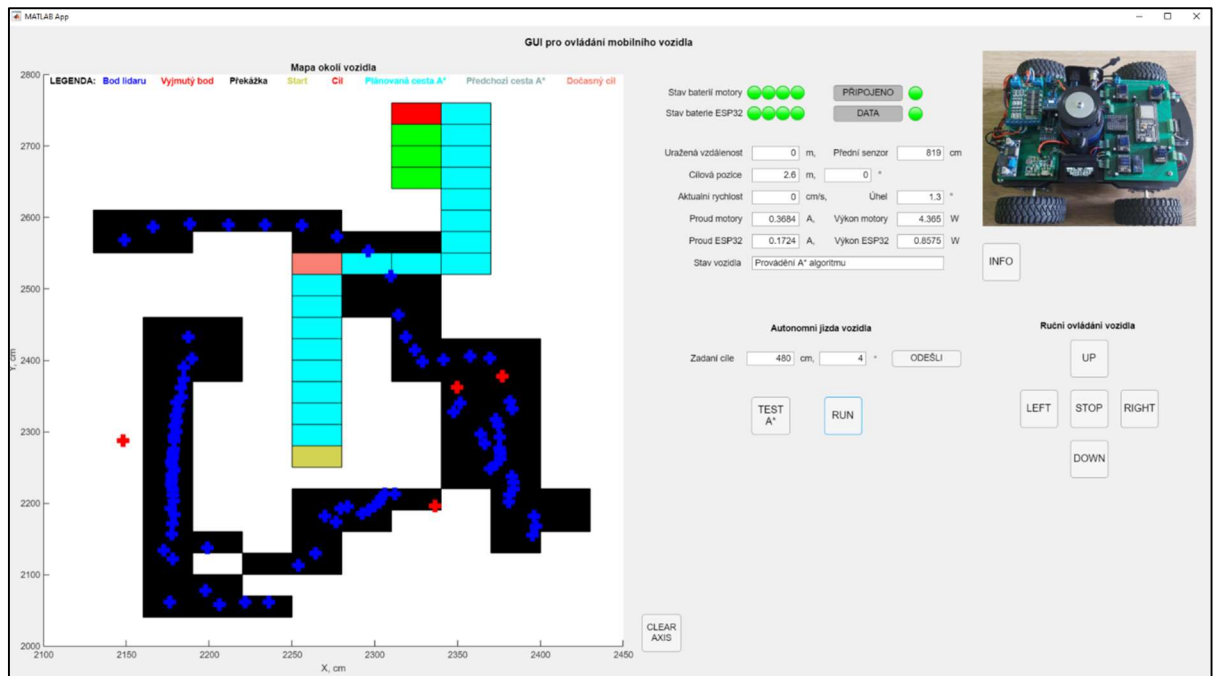


Obr. 2.23 – Závislost spotřeby elektrické energie na rychlosti vozidla

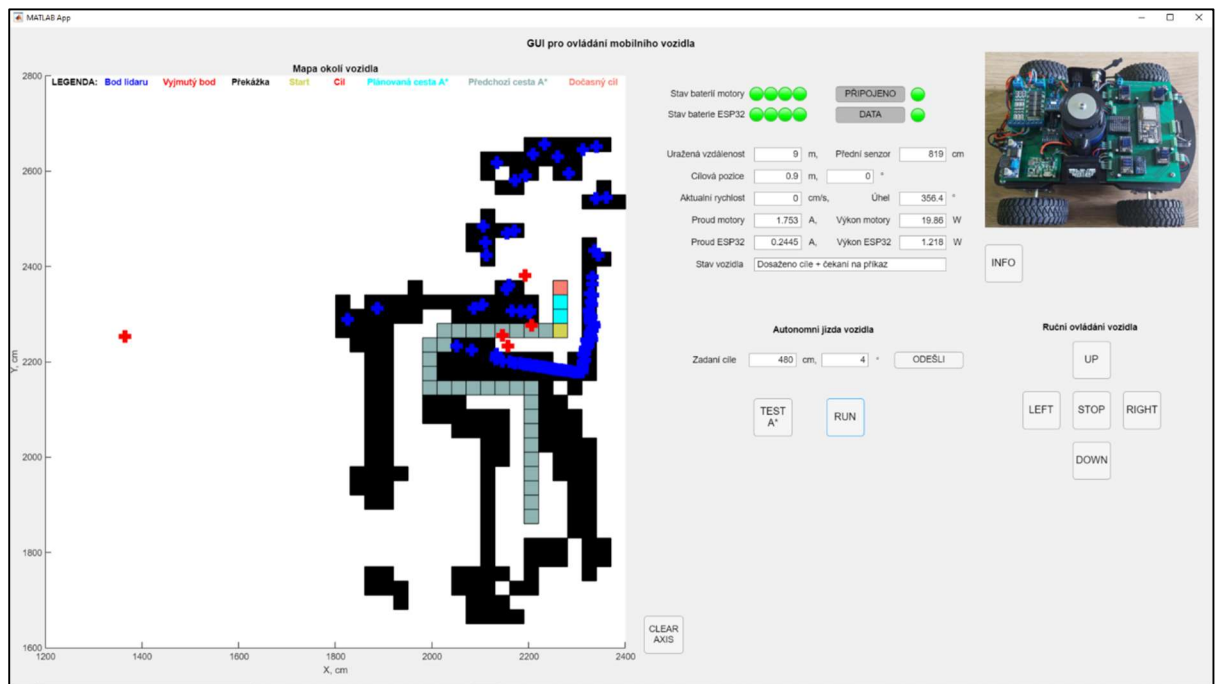
## 2.5 TESTOVÁNÍ AUTONOMNÍ JÍZDY VOZIDLA

Ladění a testování autonomní jízdy vozidla probíhalo uvnitř bytu. Celkem bylo vozidlo testováno na dvou trasách. Záznam první z nich je uveden v této kapitole. Jedná se o trasu z obývacího do dětského pokoje. Vozidlo startovalo uprostřed obývacího pokoje a jeho cíl se nacházel přibližně 5 m daleko v jiném pokoji.

Prvním krokem bylo zadání cíle do aplikace a dále došlo ke spuštění autonomní jízdy vozidla pomocí tlačítka RUN (obr. 2.24). Poté se již vozidlo pohybovalo dle stavového automatu uvedeného v kapitole 2.2.1. Výsledkem testování vozidla bylo dosažení požadovaného cíle (obr. 2.25).



Obr. 2.24 – Testování autonomní jízdy vozidla – počáteční stav



Obr. 2.25 – Testování autonomní jízdy vozidla – konečný stav

Obr. 2.25 ukazuje, že vozidlo skutečně došlo do cíle. Dále je na obrázku zobrazena celá cesta, kterou vygeneroval A\* algoritmus. Vozidlo tuto cestu z větší části projelo, nicméně tam kde to bylo možné, si cestu zkrátilo (viz „přímá viditelnost“ – kapitola 2.2.2).

### 3 ZHODNOCENÍ

Výsledkem této práce je návrh, realizace a oživení autonomního robotického vozidla. Vozidlo dokáže pracovat v manuálním i autonomním režimu a lze jej ovládat z vytvořené aplikace. Aplikace též umožňuje vizualizovat zjištěná data. Navíc je na vozidle umístěn OLED displej, který rovněž zobrazuje data o vozidle. Vozidlo obsahuje dvě DPS. Desky plošných spojů jsou koncipovány tak, aby bylo možné většinu komponent snadno vyměnit. Proto se na nich ve většině případů objevují pouze konektory, do kterých se pak nasouvají jednotlivé komponenty (řadiče, senzory apod.).

Řešení bylo realizováno dle požadavků zadání. Výsledkem je vozidlo schopné autonomního pohybu, které obsahuje elektronickou navigační jednotku vlastní konstrukce (lidar). Jako řídicí jednotka byla zvolena komerčně dostupná vývojová platforma s jednočipovým mikroprocesorem ESP32. Pohyb vozidla je pak realizován čtyřmi nezávisle řízenými krokovými motory. Řídicí software využívá vybraný algoritmus umělé inteligence (A\* algoritmus), který slouží k nalezení nejkratší cesty vozidla k cíli.

Zadání práce bylo v některých místech, kde to bylo vhodné, rozšířeno. Jedná se například o vytvoření aplikace pro ovládání a vizualizaci dat nebo třeba o umístění displeje na vozidlo. Dalším vhodným rozšířením je zajištění možnosti dobíjení baterií bez nutnosti jejich vyjmutí z vozidla. To je zajištěno pomocí ochranných a napájecích modulů na napájecí DPS vozidla.

Pro realizaci vozidla bylo využito především komponent ze zahraničních internetových obchodů (kromě motorů a řadičů krokových motorů). Tím došlo k výrazné redukci celkové ceny vozidla, která se i tak pohybuje kolem 7 500 Kč. Při nákupu komponent v České republice by se částka významným způsobem navýšila.

## 4 ZÁVĚR

Autonomní robotické vozidlo se podařilo úspěšně realizovat dle požadavků zadání. Vozidlo je schopné autonomního pohybu a dokáže nalézt nejkratší cestu k cíli. Kromě toho je možné vozidlo kvůli bezpečnosti kdykoliv zastavit. Vozidlo lze také ovládat v ručním režimu.

Projekt zabral poměrně velké množství času (řadově se jedná o stovky hodin práce), nicméně výsledkem je propracovaná konstrukce. Realizace zadání diplomové práce pro mě byla značným přínosem. Během realizace se vyskytnuly situace, které vedly k více či méně zdařilým řešením jednotlivých částí řešení vozidla. Takovýto způsob práce mě naučil být trpělivý, jelikož bez trpělivosti se návrh a realizace konstrukce takového typu neobejde. Samotná realizace vyžadovala použití znalostí nejenom z oboru automatizační techniky, ale i návrhu mechatronických konstrukcí, programování jednočipových mikropočítačů a tvorby aplikací pro osobní počítače.

Jako další osobní přínos lze uvést to, že jsem se setkal s realizací algoritmu umělé inteligence, která je v dnešní době na vzestupu a má potenciál využití do budoucna. Dále jsem se naučil pracovat s jednočipovým mikropočítačem ESP32, který kvůli integraci WiFi čipu nalezne uplatnění například v IoT. Taktéž práce s krokovými motory nabídla řadu problémů a činností k řešení. Návrh vozidla také obsahoval tvorbu dvou DPS, které bylo třeba navrhnout, nechat vyrobit a oživit.

Vozidlo lze dále upravovat a vylepšovat. Pro zobrazení údajů o vozidle by bylo možné kromě aplikace a displeje na vozidle využít ještě například webovou stránku. Vozidlo by dále bylo možné doplnit o další bezpečnostní prvky (například o zadní senzor vzdálenosti) apod.

Přestože vozidlo lze dále vylepšovat a zdokonalovat, svůj účel plní dobře.

## POUŽITÁ LITERATURA

- 1,8° 42mm Hybrid Stepper Motor-NEMA17, ©1990–2021. GM electronic [online]. Praha [cit. 2021-02-01]. Dostupné z: <https://www.gme.cz/data/attachments/dsh.775-138.1.pdf>
- Absolute Orientation BNO055 Sensor, ©2010-2020. *AliExpress* [online]. [cit. 2021-02-21]. Dostupné z: <https://www.aliexpress.com/item/32809065847.html?spm=a2g0s.9042311.0.0.27424c4dEYWtdz>
- BIGTREETECH TMC2209-V1.2, ©2021. GitHub [online]. China: Shenzhen Bigu Technology [cit. 2021-02-01]. Dostupné z: <https://github.com/bigtreetech/BIGTREETECH-TMC2209-V1.2/blob/master/manual/TMC2209-V1.2-manual.pdf>
- CERMAN, Aleš, 2019. Autonomní jízda silničních vozidel. *Elektro: odborný časopis pro elektrotechniku* [online]. Praha: FCC Public, 14. 6. 2019, 2019(6), 36-38 [cit. 2021-01-31]. Dostupné z: <http://www.odbornecasopisy.cz/elektro/clanek/autonomni-jizda-silnicnich-vozidel--3746>
- Comparing GUIDE and App Designer, ©1994-2021. MathWorks® [online]. Natick: The MathWorks [cit. 2021-02-03]. Dostupné z: <https://www.mathworks.com/products/matlab/app-designer/comparing-guide-and-app-designer.html>
- DOLEŽEL, Petr, 2019a. *Základy umělé inteligence I: Úvod do předmětu*. Pardubice.
- DOLEŽEL, Petr, 2019b. *Prohledávání stavového prostoru a plánování*. Pardubice.
- DOLEŽEL, Petr, 2019c. *Prohledávání stavového prostoru a plánování: Hledání řešení*. Pardubice.
- GUIDE to App Designer Migration Tool for MATLAB, ©1994-2021. MathWorks® [online]. Natick: The MathWorks [cit. 2021-02-03]. Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/66087-guide-to-app-designer-migration-tool-for-matlab>
- INA226 current/power sensor, ©2010-2020. *AliExpress* [online]. [cit. 2021-02-01]. Dostupné z: <https://www.aliexpress.com/item/33005541425.html?spm=a2g0s.9042311.0.0.27424c4d2Iugue>



- KILIÁN, Karel, 2018. Čím se LIDAR liší od radaru a jaká je jeho role v autonomních vozidlech. Vtm [online]. Praha: CZECH NEWS CENTER, 12. října 2018 [cit. 2021-02-01]. Dostupné z: [https://vtm.zive.cz/clanky/cim-se-lidar-lisi-od-radaru-a-jaka-je-jeho-role-v-autonomnich-vozidleh/sc-870-a-195431/default.aspx](https://vtm.zive.cz/clanky/cim-se-lidar-lisi-od-radaru-a-jaka-je-jeho-role-v-autonomnich-vozidlech/sc-870-a-195431/default.aspx)
- KNOTEK, Jan, 2019. Domáci automatizace s využitím mikropočítače [online]. Pardubice [cit. 2021-02-01]. Dostupné z: <https://hdl.handle.net/10195/73206>. Bakalářská práce. Univerzita Pardubice, Fakulta elektrotechniky a informatiky. Vedoucí práce Libor Havlíček.
- KOŘOUSKOVÁ, Barbora, ©2021. Využití umělé inteligence (nejen) v podnikání. Rascasone [online]. Praha: Rascasone [cit. 2021-02-03]. Dostupné z: [https://www.rascasone.com/cs/blog/vyuziti-umele-inteligence-ai?utm\\_term=um%C4%9B%C3%A1%20inteligence&utm\\_campaign=CZ\\_SEARCH\\_PWA&utm\\_source=adwords&utm\\_medium=ppc&hsa\\_tgt=kwd-312845998482&hsa\\_mt=e&hsa\\_cam=9686037817&hsa\\_kw=um%C4%9B%C3%A1%20inteligence&hsa\\_net=adwords&hsa\\_acc=2107111182&hsa\\_grp=99319030237&hsa\\_src=g&hsa\\_ad=426841551481&hsa\\_ver=3&gclid=CjwKCAiAu8SABhAxEiwAsodSZFrohQ0h7cThfQLwC\\_UYOkqDXMefWg-nT-UG6S9GoDsAmpKcCjY4RoCUM0QAvD\\_BwE](https://www.rascasone.com/cs/blog/vyuziti-umele-inteligence-ai?utm_term=um%C4%9B%C3%A1%20inteligence&utm_campaign=CZ_SEARCH_PWA&utm_source=adwords&utm_medium=ppc&hsa_tgt=kwd-312845998482&hsa_mt=e&hsa_cam=9686037817&hsa_kw=um%C4%9B%C3%A1%20inteligence&hsa_net=adwords&hsa_acc=2107111182&hsa_grp=99319030237&hsa_src=g&hsa_ad=426841551481&hsa_ver=3&gclid=CjwKCAiAu8SABhAxEiwAsodSZFrohQ0h7cThfQLwC_UYOkqDXMefWg-nT-UG6S9GoDsAmpKcCjY4RoCUM0QAvD_BwE)
- Krokové motory, ©2021. Profirobot [online]. Praha, 2013 [cit. 2021-01-31]. Dostupné z: <http://profirobot.cz/wp-content/uploads/2018/03/Krokov%C3%A9-motory.pdf>
- Krokový motor, ©2021. Penta servis [online]. Holice: PENTA SERVIS, 2021 [cit. 2021-01-31]. Dostupné z: <https://www.pentaservis.cz/pojem/krokovy-motor/>
- KUPKA, Libor, 2021. Softwarové prostředky pro řízení. Pardubice.
- Lithium Battery Charger, ©2010-2020. AliExpress [online]. [cit. 2021-02-03]. Dostupné z: <https://www.aliexpress.com/item/4000322660703.html?spm=a2g0s.9042311.0.0.27424c4dHkmJf3>
- Lithium battery protection board, ©2010-2020. AliExpress [online]. [cit. 2021-02-01]. Dostupné z: <https://www.aliexpress.com/item/4000042036512.html?spm=a2g0s.9042311.0.0.27424c4dCr82OB>
- MATLAB®, ©1991-2021. Humusoft: ve službách vědy a techniky [online]. Praha: HUMUSOFT [cit. 2021-02-03]. Dostupné z: <https://www.humusoft.cz/matlab/details/>

- MCP23017 Bidirectional 16-Bit I/O Expander, ©2010-2020. AliExpress [online]. [cit. 2021-02-01]. Dostupné z:  
[https://www.aliexpress.com/item/4001066322161.html?spm=a2g0o.productlist.0.0.5d79450c1XnNK0&algo\\_pvid=eecbb10d-e51b-4d66-8f32-a4fdfa2e5a14&algo\\_expid=eecbb10d-e51b-4d66-8f32-a4fdfa2e5a14-0&btsid=2100bdde16110606393758077ebe15&ws\\_ab\\_test=searchweb0\\_0,searchweb201602\\_,searchweb201603\\_](https://www.aliexpress.com/item/4001066322161.html?spm=a2g0o.productlist.0.0.5d79450c1XnNK0&algo_pvid=eecbb10d-e51b-4d66-8f32-a4fdfa2e5a14&algo_expid=eecbb10d-e51b-4d66-8f32-a4fdfa2e5a14-0&btsid=2100bdde16110606393758077ebe15&ws_ab_test=searchweb0_0,searchweb201602_,searchweb201603_)
- MT3608 DC-DC Step Up Converter, ©2010-2020. AliExpress [online]. [cit. 2021-02-01].  
Dostupné z:  
<https://www.aliexpress.com/item/1005001557483971.html?spm=a2g0s.9042311.0.0.27424c4dyTgdvj>
- IIC I2C OLED display 1,3" 128x64 Bílý, ©2019. *Drátek.cz* [online]. ECLIPSERA [cit. 2021-04-16]. Dostupné z: <https://dratek.cz/docs/produkty/0/735/1527497643.pdf>
- Rechargeable Li-ion battery, ©2010-2020. AliExpress [online]. [cit. 2021-02-01]. Dostupné z:  
<https://www.aliexpress.com/item/VariCore-New-Original-18650-NCR18650B-Rechargeable-Li-ion-battery-3-7V-3400mAh-For-Flashlight-use-Free/32889223073.html?spm=a2g0s.9042311.0.0.27424c4d1IQyYJ>
- Snímač vzdálenosti TF-Luna. *ControlTech* [online]. Ovčáry [cit. 2021-02-21]. Dostupné z:  
<https://eshop.controltech.cz/eshop/data/TF-Luna.pdf>
- STEHLÍK, Petr, ©1998–2021. ESP32 je tu. Co přinese nástupce ESP8266? *Root.cz: Informace nejen ze světa Linuxu* [online]. 1.9.2016 [cit. 2021-02-01]. ISSN 1212-8309. Dostupné z:  
<https://www.root.cz/clanky/esp32-je-tu-co-prinese-nastupce-esp8266/>
- Stepdown nastavitelný měnič s LM2596 DC-DC, ©2016. *Drátek.cz* [online]. Havlíčkův Brod: ECLIPSERA [cit. 2021-02-01]. Dostupné z: [https://dratek.cz/arduino/1303-stepdown-nastavitelny-menic-s-lm2596-dc-dc.html?gelid=CjwKCAiAo5qABhBdEiwAOtGmbvDuAgv8IGlu\\_UnesSt7-wnI30jgAwde6NpzBBKXOxiqFONRFjDz5BoCWCUQAvD\\_BwE](https://dratek.cz/arduino/1303-stepdown-nastavitelny-menic-s-lm2596-dc-dc.html?gelid=CjwKCAiAo5qABhBdEiwAOtGmbvDuAgv8IGlu_UnesSt7-wnI30jgAwde6NpzBBKXOxiqFONRFjDz5BoCWCUQAvD_BwE)
- ŠIMKOVÁ, Pavlína, ©2021. Prohledávání stavového prostoru. *Docplayer* [online]. DocPlayer [cit. 2021-02-03]. Dostupné z: <http://docplayer.cz/29739414-1-prohledavani-stavoveho-prostoru.html>
- TMC2209 Datasheet, ©2021. Trinamic: New part of Maxim Integrated [online]. Hamburg [cit. 2021-02-03]. Dostupné z:

[https://www.trinamic.com/fileadmin/assets/Products/ICs\\_Documents/TMC2209\\_Datasheet\\_V103.pdf](https://www.trinamic.com/fileadmin/assets/Products/ICs_Documents/TMC2209_Datasheet_V103.pdf)

USB nabíječka baterií, ©2016. Drátek.cz [online]. Havlíčkův Brod: ECLIPSERA [cit. 2021-02-01]. Dostupné z: <https://dratek.cz/docs/produkty/0/944/1424114838.pdf>

VL53L0X Time-of-Flight Laser Ranging Sensor, ©2010-2020. *AliExpress* [online]. [cit. 2021-02-21]. Dostupné z:  
<https://www.aliexpress.com/item/32960610502.html?spm=a2g0s.9042311.0.0.27424c4dzbGV38>

# **PŘÍLOHY**

**A – CD**

**Příloha k diplomové práci**  
Autonomní robotické vozidlo  
Bc. Jan Knotek

**CD**

## **OBSAH**

- 1 Text diplomové práce ve formátu PDF
- 2 Zdrojový kód mikropočítače ESP32
- 3 Zdrojový kód GUI aplikace
- 4 3D model vozidla
- 5 Schémata zapojení DPS
- 6 Fotodokumentace