



TECHNICKÁ UNIVERZITA V LIBERCI  
Fakulta mechatroniky, informatiky  
a mezioborových studií



# Řízení akčních členů elektromobilu pomocí sběrnice CAN

## Diplomová práce

*Studijní program:*

N2612 Elektrotechnika a informatika

*Studijní obor:*

Mechatronika

*Autor práce:*

**Bc. Tomáš Kubíček**

*Vedoucí práce:*

Ing. Daniel Kajzr

Ústav mechatroniky a technické informatiky





## Zadání diplomové práce

# Řízení akčních členů elektromobilu pomocí sběrnice CAN

*Jméno a příjmení:* **Bc. Tomáš Kubíček**  
*Osobní číslo:* M20000184  
*Studijní program:* N2612 Elektrotechnika a informatika  
*Studijní obor:* Mechatronika  
*Zadávací katedra:* Ústav mechatroniky a technické informatiky  
*Akademický rok:* **2021/2022**

### Zásady pro vypracování:

1. Seznamte se s dokumentací k jednotce EPAS, EPB a Linak.
2. Zapojte jednotky EPAS, EPB a Linak k jejich akčním členům a propojte řídicí systém pomocí sběrnice CAN.
3. Sestavte komunikační zprávy pro řízení úhlu natočení kol a ovládání parkovací brzdy. Navržené zprávy otestujte na vytvořeném pracovišti.
4. Vytvořte vizualizaci pro jednoduché řízení vybraných akčních členů. Vizualizaci propojte s ovladačem IFM.
5. Porovnejte řešení EPAS a Linak pro směrové řízení.

*Rozsah grafických prací:*  
*Rozsah pracovní zprávy:*  
*Forma zpracování práce:*  
*Jazyk práce:*

dle potřeby dokumentace  
40–50 stran  
tištěná/elektronická  
Čeština



### **Seznam odborné literatury:**

- [1] ŠŤASTNÝ, Jiří a Branko REMEK. Autoelektrika a autoelektronika. 6. vyd. Praha: T. Malina, 2003, 315 s. ISBN 80-86293-02-5.
- [2] JOHN, Kharl-Heinz; TIEGELKAMP, Michael. IEC 61131-3 Programming Industrial Automation Systems : Concepts and Programming Languages, Requirements for Programming Systems, Decision –Making Aids. 2nd
- [3] LINAK, Linear Actuators and Electronics User Manual 2021, [online]. [cit. 2021-10-12]. Dostupné z: <https://cdn.linak.com/-/media/files/user-manual-source/en/linak-linear-actuators-and-electronics-user-manual-eng.pdf>

*Vedoucí práce:*

Ing. Daniel Kajzr  
Ústav mechatroniky a technické informatiky

*Datum zadání práce:*

12. října 2021

*Předpokládaný termín odevzdání:*

16. května 2022

prof. Ing. Zdeněk Plíva, Ph.D.  
děkan

L.S.

doc. Ing. Josef Černožorský, Ph.D.  
vedoucí ústavu

## Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědom toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci 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.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

15. května 2022

Bc. Tomáš Kubíček

## Poděkování

Tímto bych rád poděkoval svému vedoucímu práce Ing. Danu Kajzrovi za ukázkové vedení práce a rady, které mi poskytl. Poděkovat bych chtěl i Ing. Leoši Beranovi, Ph.D. a Ing. Tomáši Myslivcovi za pomoc, kterou mi poskytli nejen při řešení této práce, ale i během studia. Dále bych rád poděkoval mým rodičům Ing. Martinu Kubíčkoví a Haně Kubíčkové za pomoc a podporu při studiu na vysoké škole.

# Řízení akčních členů elektromobilu pomocí sběrnice CAN

## Abstrakt

Cílem této práce je vytvoření funkčního servisního ovládacího systému EPAS, EPB a pohonů LA36. Tyto systémy jsou použity jako aktuátory autonomního elektromobilu vyvíjeného v rámci projektu Technické univerzity v Liberci. Tento projekt se zaměřuje na vývoj modulární platformy pro autonomní podvozky specializovaných elektrovozidel pro dopravu nákladu a zařízení. Zkráceně je označován jako projekt AnteTUL. Ovládací systém má být umožněno pomocí ovladače IFM a webové vizualizace. Jednotlivé aktuátory budou ovládány pomocí PLC prostřednictvím průmyslové sběrnice CAN. V rámci této práce jsou popsány jednotlivé úrovně systémů pro automatizaci vozidel a aktuální úroveň těchto systémů. Tato práce obsahuje podrobný popis struktury datové zprávy na sběrnici CAN a výpočet CRC. Práce se dále věnuje popisu ovladače IFM a pohonu LA36. Další část práce představuje vytvořené servisní ovládací a zapojení jednotlivých aktuátorů k PLC. Poslední část práce je zaměřena na porovnání systému EPAS a pohonu LA36.

**Klíčová slova:** Sběrnice CAN, SAE J1939, autonomní elektromobil, EPAS, EPB, LA36, webová vizualizace, ovladač IFM, AnteTUL

# Control of electric car actuators via CAN bus

## Abstract

The aim of this work is to create a functional service control of EPAS, EPB and LA36 drives. These systems are used as actuators of an autonomous electric car developed within the project of the Technical University of Liberec. This project focuses on the development of a modular platform for autonomous chassis of specialized electric vehicles for the transport of goods and equipment. Actuator control should be enabled using the IFM controller and web visualization. The individual actuators will be controlled by a PLC via the CAN bus. Within this work, the individual levels of vehicle automation systems and the current level of these systems are described. This work contains a detailed description of the data message structure on the CAN bus and CRC calculation. The work also deals with the description of the IFM controller and the LA36 drive. The next part of the work represents created service control and connection of individual actuators to the PLC. The last part of the work is focused on the comparison of the EPAS system and the LA36 drive.

**Keywords:** CAN bus, SAE J1939, autonomous electric car, EPAS, EPB, LA36, web visualization, IFM controller, AnteTUL

# Obsah

Úvod	15
<b>2 Projekt AnteTUL</b>	<b>18</b>
<b>3 Autonomní vozidla</b>	<b>22</b>
3.1 Úrovně automatizace	22
3.1.1 Úroveň 0 (Žádná automatizace)	23
3.1.2 Úroveň 1 (Podpora řidiče)	23
3.1.3 Úroveň 2 (Částečná automatizace)	23
3.1.4 Úroveň 3 (Podmíněná automatizace)	23
3.1.5 Úroveň 4 (Vysoká automatizace)	24
3.1.6 Úroveň 5 (Plná automatizace)	24
3.2 Současná úroveň autonomního řízení	24
3.3 Tesla a Waymo	26
3.3.1 Tesla	26
3.3.2 Waymo	27
3.3.3 Porovnání	27
<b>4 CAN</b>	<b>29</b>
4.1 Struktura zprávy specifikace CAN 2.0A	29
4.1.1 Začátek zprávy (Start of Frame)	30
4.1.2 Arbitrážní pole (Arbitration Field)	30
4.1.3 Řídicí pole (Control Field)	31
4.1.4 Datové pole (Data Field)	32
4.1.5 CRC pole (CRC Field)	32
4.1.6 ACK pole (ACK Field)	32
4.1.7 Konec zprávy (End of Frame)	33
4.2 Struktura zprávy specifikace CAN 2.0B	33
4.2.1 Arbitrážní pole (Arbitration Field)	33
4.2.2 Řídicí pole (Control Field)	34
4.3 Protokoly založené na sběrnici CAN	35
4.3.1 SAE J1939	35



<b>5</b>	<b>CRC</b>	<b>38</b>
5.1	Výpočet CRC . . . . .	38
5.2	Polynomiální dělení . . . . .	39
5.3	Algoritmus pro výpočet CRC . . . . .	41
<b>6</b>	<b>Lineární pohony</b>	<b>42</b>
6.1	Pohon LA36 . . . . .	42
6.1.1	Zapojení a komunikace . . . . .	44
<b>7</b>	<b>Ovladač IFM</b>	<b>52</b>
7.1	Zapojení a komunikace . . . . .	54
7.1.1	Popis zpráv pro řídicí jednotku ovladače IFM . . . . .	55
7.1.2	Popis zprávy od řídicí jednotky ovladače IFM . . . . .	57
<b>8</b>	<b>Ovládání akčních členů</b>	<b>59</b>
8.1	Zapojení . . . . .	60
8.2	Vizualizace . . . . .	62
8.2.1	Stránka EPAS (vizualizace) . . . . .	64
8.2.2	Stránka EPB (vizualizace) . . . . .	66
8.2.3	Stránka LINAK (vizualizace) . . . . .	67
8.3	Ovládání prostřednictvím ovladače IFM . . . . .	71
8.3.1	Stránka EPAS . . . . .	72
8.3.2	Stránka EPB . . . . .	73
8.3.3	Stránka LINAK . . . . .	74
<b>9</b>	<b>Porovnání a testování systému EPAS a pohonu LA36</b>	<b>76</b>
9.1	Porovnání . . . . .	76
9.2	Testování . . . . .	79
<b>10</b>	<b>Závěr</b>	<b>81</b>
	<b>Použitá literatura</b>	<b>82</b>
	<b>A Doplnkové ilustrace</b>	<b>85</b>

## Seznam obrázků

2.1	Generace 0 (zdroj autor)	19
2.2	Generace 1 (zdroj autor)	20
2.3	Ukázka pracoviště (zdroj autor)	21
2.4	Rozvaděč s řídicí elektronikou (zdroj autor)	21
4.1	Struktura datové zprávy (zdroj [16])	30
4.2	Arbitrážní pole (zdroj [16])	31
4.3	Řídicí pole (zdroj [16])	31
4.4	CRC pole (zdroj [16])	32
4.5	ACK pole (zdroj [16])	33
4.6	Standardní a rozšířený formát zprávy specifikace CAN 2.0B (zdroj [16])	35
4.7	Typické použití protokolu SAE J1939 ve vozidle (zdroj [20])	36
4.8	Struktura 29bitového identifikátoru (zdroj [21])	36
6.1	Zobrazení pohonu LA36 s popisky (zdroj [24])	44
6.2	Kabely pohonu LA36 (zdroj autor)	45
6.3	Komunikace mezi pohonem LA36 a PLC (zdroj autor)	46
6.4	Rychlost pohonu LA36 během pohybu pístnice (zdroj autor)	46
7.1	Ovladač IFM (CR1301) (zdroj [28])	52
7.2	Označení prvků ovladače IFM (zdroj autor)	53
7.3	Polohy Knoflíku (zdroj autor)	53
7.4	Označení indikátorů ovladače IFM (zdroj autor)	54
7.5	Konektor ovladače IFM (zdroj [29])	55
7.6	Příklad nastavení indikátorů ovladače IFM (zdroj autor)	57
8.1	Struktura programu (zdroj autor)	60
8.2	Schéma zapojení aktuátorů se systémy EPAS (zdroj autor)	61
8.3	Modul X20IF2772	61
8.4	Modul X20CS2770	61
8.5	Schéma zapojení aktuátorů s pohony LA36 (zdroj autor)	62
8.6	Úvodní stránka vizualizace (zdroj autor)	63
8.7	Přihlašovací okno	63
8.8	Přihlášený uživatel	63
8.9	Spuštění ovládání z vizualizace (zdroj autor)	64
8.10	Navigační lišta (zdroj autor)	64

8.11	Stránka EPAS ve vizualizaci (zdroj autor) . . . . .	65
8.12	Nastavení parametrů jednotky EPAS (zdroj autor) . . . . .	66
8.13	Stránka EPB ve vizualizaci (zdroj autor) . . . . .	66
8.14	Zobrazení zabrzděného stavu ve vizualizaci (zdroj autor) . . . . .	67
8.15	Stránka LINAK ve vizualizaci (zdroj autor) . . . . .	68
8.16	Okno pro nastavení parametrů pohonů LA36 (zdroj autor) . . . . .	69
8.17	Zobrazení aktivního chybového stavu pohonu LA36 ve vizualizaci (zdroj autor) . . . . .	70
8.18	Zobrazení chybového okna pohonu LA36 ve vizualizaci (zdroj autor) .	71
8.19	Přepínání stránek na ovladači IFM (zdroj autor) . . . . .	72
8.20	Stránka EPAS (zdroj autor) . . . . .	73
8.21	Stránka EPB (zdroj autor) . . . . .	74
8.22	Stránka LINAK (zdroj autor) . . . . .	75
9.1	Porovnání systému EPAS a pohonu LA36 (zdroj autor) . . . . .	76
9.2	Ukázka zapojení systému EPAS na pracovišti (zdroj autor) . . . . .	79
9.3	Testovací rám pohonu LA36 (zdroj autor) . . . . .	80
A.1	Ukázka pracoviště (před instalací aktuátorů) (zdroj autor) . . . . .	85
A.2	Ukázka pracoviště (po instalaci aktuátorů) (zdroj autor) . . . . .	86
A.3	Zobrazení vypnutého ovládání z vizualizace (zdroj autor) . . . . .	86
A.4	Zobrazení odbrzděného stavu ve vizualizaci (zdroj autor) . . . . .	87
A.5	Zobrazení zasunutého pohonu LA36 ve vizualizaci (zdroj autor) . . .	87
A.6	Zobrazení vysunutého pohonu LA36 ve vizualizaci (zdroj autor) . . .	88
A.7	Ovládací panel servisního technika (zdroj autor) . . . . .	88

## Seznam tabulek

4.1	Kombinace DLC bitů (podle [16]) . . . . .	31
5.1	Binární podoba polynomu 5.1 . . . . .	39
5.2	Polynomiální dělení . . . . .	39
5.3	Hodnota zbytku po dělení . . . . .	40
5.4	Binární dělení . . . . .	40
6.1	Zapojení pohonu LA36 (podle [27]) . . . . .	45
6.2	Struktura zprávy pro řídicí jednotku (podle [27]) . . . . .	47
6.3	Zpráva pro dosažení požadované polohy vysunutí pístnice . . . . .	48
6.4	Zpráva pro zasunutí pístnice . . . . .	49
6.5	Struktura zprávy od řídicí jednotky (podle [27]) . . . . .	49
6.6	Seznam chyb pohonu LA36 (podle [27]) . . . . .	50
6.7	Seznam binárních proměnných pohonu LA36 (podle [27]) . . . . .	50
7.1	Zapojení ovladače IFM (podle [29]) . . . . .	55
7.2	Struktura zpráv pro ovladač IFM (podle [29]) . . . . .	56
7.3	Vybrané kombinace v barevném prostoru RGB . . . . .	56
7.4	Příklad zpráv pro ovladač IFM . . . . .	57
7.5	Struktura zprávy od řídicí jednotky (podle [29]) . . . . .	58
7.6	Příklad zprávy od řídicí jednotky . . . . .	58

## Seznam zkratek

<b>A</b>	Ampér
<b>ACK</b>	Acknowledge, potvrzení
<b>AI</b>	Artificial Intelligence, umělá inteligence
<b>AnteTUL</b>	Projekt TUL Modulární platforma pro autonomní podvozky specializovaných elektrovozidel pro dopravu nákladu a zařízení
<b>AR</b>	Automation Runtime, operační systém
<b>AS</b>	Automation Studio, vývojové prostředí
<b>CAN</b>	Controller Area Network, průmyslová sběrnice
<b>CBFF</b>	Classical Base Frame Format, standardní formát datové zprávy
<b>CEFF</b>	Classical Extended Frame Format, rozšířený formát datové zprávy
<b>CRC</b>	Cyclic Redundancy Check, cyklický redundantní součet
<b>CSS</b>	Cascading Style Sheets, programovací jazyk
<b>CxI</b>	Ústav pro nanomateriály, pokročilé technologie a inovace
<b>DC</b>	Direct Current, stejnosměrný proud
<b>DLC</b>	Data Length Code, kód délky dat
<b>DP</b>	Data Page, stránka dat
<b>DVD</b>	Digital Video Disc, datové médium
<b>EMF</b>	Electromotive Force, elektromotorická síla
<b>EOF</b>	End Of Frame, konec zprávy
<b>EPAS</b>	Electric Power Assisted Steering, elektrický posilovač řízení a parkovací asistent
<b>EPB</b>	Electric Park Brake, elektrická parkovací brzda
<b>FD</b>	Flexible Data-rate, flexibilní rychlost přenosu
<b>FSD</b>	Full Self Driving, systém společnosti Tesla
<b>GND</b>	Ground, napěťový potenciál země
<b>GNSS</b>	Global Navigation Satellite System, globální družicový polohový systém
<b>HTML</b>	Hypertext Markup Language, hypertextový značkovací jazyk
<b>IDE</b>	Identifier Extension, označení bitu na sběrnici CAN
<b>IoT</b>	Internet of Things, internet věcí
<b>IP</b>	Internet Protocol, internet protokol
<b>IPxx</b>	Ingress Protection, označení určující stupeň krytí
<b>ISO</b>	International Organization for Standardization, mezinárodní organizace pro normalizaci
<b>kbit/s</b>	Kilobit za sekundu
<b>km/h</b>	Kilometry za hodinu

<b>LED</b>	Light Emitting Diode, dioda emitující světlo
<b>LIDAR</b>	Light Detection And Ranging, zařízení sloužící k dálkovému měření vzdálenosti pomocí laserového paprsku
<b>LSB</b>	Least Significant Bit, nejméně významný bit
<b>mA</b>	Miliampér
<b>Mbit/s</b>	Megabit za sekundu
<b>MLIT</b>	Japanese Ministry of Land, Infrastructure, Transport and Tourism, japonské ministerstvo
<b>mm</b>	Milimetr
<b>mm/s</b>	Milimetr za sekundu
<b>mph</b>	Miles per hour, míle za hodinu
<b>ms</b>	Milisekunda
<b>MSB</b>	Most Significant Bit, nejvíce významný bit
<b>N</b>	Newton
<b>OSI</b>	Open System Interconnection, standardizace komunikace v počítačových sítích
<b>PDU</b>	Protocol Data Unit, termín použitý v protokolu SAE J1939
<b>PGN</b>	Parameter Group Number, termín použitý v protokolu SAE J1939
<b>PLC</b>	Programmable Logic Controller, programovatelný logický automat
<b>Radar</b>	Radio Detection And Ranging, zařízení, které prostřednictvím rádiových vln dokáže vyhledávat cíle, identifikovat je a určovat jejich polohu
<b>RGB</b>	Red Green Blue, červená zelená modrá
<b>RTR</b>	Remote Transmission Request, označení bitu na sběrnici CAN
<b>s</b>	Sekunda
<b>SAE</b>	Society of Automotive Engineers, společnost automobilových inženýrů
<b>Sci-fi</b>	Science-fiction, vědecko-fantastický
<b>SOF</b>	Start Of Frame, začátek zprávy
<b>SRR</b>	Substitute Remote Request, označení bitu na sběrnici CAN
<b>SVG</b>	Scalable Vector Graphics, značkovací jazyk a formát souboru
<b>SW</b>	Software
<b>TUL</b>	Technická univerzita v Liberci
<b>V</b>	Volt
<b>XOR</b>	Exclusive OR, logický operátor

## Úvod

Nejspíš každý člověk na Zemi se v dnešní době setkal s termínem autonomní vozidlo. Tento termín mohl zaslechnout např. ve sci-fi filmu, dočíst se o něm v knize či odborné publikaci. Ať už se s tímto termínem člověk setkal jakkoliv, dříve byl tento termín pouze vizí budoucnosti. V současné době moderních technologií a vyspělých neuronových sítí se k této vizi lidstvo pomalu přibližuje. Je poměrně velká šance, že v blízké budoucnosti budou autonomní vozidla existovat. Nebo už existují?

Pod termínem autonomní vozidlo si nejspíš každý představí osobní automobil. Ten povětšinou slouží pro přepravu lidí nebo materiálu na pozemních komunikacích. Hlavním důvodem, který způsobuje intenzivní práci spousty společností na vývoji autonomních vozidel, je především zvýšení bezpečnosti. I sebelepší řidič nikdy nebude schopen konkurovat vyspělému autonomnímu vozidlu. Možnosti lidského řidiče jsou totiž omezené. Je to z důvodu reakční doby řidiče, která je v porovnání s řídicími systémy značně vyšší. Lidský řidič má i omezený výhled na okolí vozidla. Navíc výkon řidiče není konstantní a je závislý na mnoha okolnostech. Např. unavený řidič není tak všímavý, jako plně odpočatý řidič. Oproti tomu autonomní vozidlo může obsahovat senzory, které dokáží snímat okolí vozidla v rozsahu 360°. V závislosti na použitém řídicím systému jsou tyto údaje vyhodnocovány a aktuátorům jsou posílány příkazy v řádech desítek ms i méně. Navíc řídicí systém pracuje pořád stejně bez ohledu na dobu provozu. Na druhou stranu funkčnost autonomních vozidel je závislá na datech ze senzorů a úrovni řídicího systému. V případě poruchy senzoru nemusí řídicí systém správně vyhodnotit danou situaci a vozidlo se může začít chovat nevyzpytatelně. Dokonce ani řídicí systém nemusí být dokonalý a mohou existovat situace (stavy), které nejsou v systému definovány.

Autonomní vozidla se ale nemusí provozovat pouze na pozemních komunikacích. Moderní technologie umožňují značně zrychlovat provoz ve výrobních halách, skladovacích systémech, na stavbách atd. V těchto prostorách jsou neustále kladeny čím dál vyšší nároky na spolehlivost, efektivitu a bezpečnost. Z toho důvodu je řada vykonávaných činností člověkem automatizována a lidská síla je nahrazena strojem. V souvislosti s tím je potřeba vyvíjet i dopravní a manipulační vozidla v těchto prostorech. Tato vozidla musí danou činnost provádět efektivně a především bezpečně. V současnosti se tak nepracuje jen na vývoji autonomních osobních vozidel, ale i na vývoji autonomních užitkových vozidel.

Vývojem autonomních užitkových vozidel se zabývá mnoho společností. V tomto ohledu nechtěla TUL zůstat pozadu a v roce 2018 získala projekt s názvem Modulární platforma pro autonomní podvozky specializovaných elektrovozidel pro dopravu nákladu a zařízení. Zkráceně je tento projekt nazýván AnteTUL. Cílem projektu

AnteTUL je vytvoření autonomního elektromobilu, který bude schopen na různých pracovištích efektivně a bezpečně převážet náklad a zařízení.

Cílem této práce je vytvoření SW pro ovládání vybraných aktuátorů vyvíjeného elektromobilu v rámci projektu AnteTUL. A také vytvoření rozhraní pro servisního technika. S pomocí tohoto rozhraní bude servisní technik schopen libovolně ovládat jednotlivé aktuátory. Tato práce navazuje na magisterský projekt Řízení akčních členů elektromobilu pomocí sběrnice CAN [1]. V rámci projektu bylo probráno téma autonomních užitkových vozidel, byly popsány systémy EPAS a EPB a byla zde představena průmyslová sběrnice CAN. Tato diplomová práce rozšiřuje téma autonomních vozidel na osobní automobily, detailněji se zabývá sběrnici CAN, popisuje pohon LA36 a ovladač od společnosti ifm (dále označován jako ovladač IFM) a jejich zapojení a komunikaci po sběrnici CAN, dále popisuje vytvořené servisní rozhraní a porovnává systém EPAS s pohonem LA36.

Stejně jako jednotka řízení EPAS má i pohon LA36 zajistit u projektu AnteTUL směrové řízení kol. Pro projekt se plánuje pořízení celkem 4 pohonů LA36. Každý z těchto pohonů má umožnit zatáčení jednoho kola vyvíjeného elektromobilu. Díky tomu bude možné jednotlivá kola natáčet zcela nezávisle na natočení zbývajících kol. To umožní zatáčení kol přední i zadní nápravy a značné zvýšení manévrovatelnosti elektromobilu. Pochopitelně je nutné, aby se během pohybu elektromobilu dodržovala Ackermannova podmínka.

Z důvodu snížení poloměru otáčení (zvýšení manévrovatelnosti) se řízení všech kol většinou využívá u vozidel velkých rozměrů, která se nepohybují vysokou rychlostí. S takovými vozidly je možné se povětšinou setkat na stavbách. Existují ale i osobní automobily, které také disponují řízením všech kol. U těchto automobilů je zvláštností, že se způsob řízení kol mění v závislosti na rychlosti vozidla. Při nižších rychlostech se kola zadní nápravy otáčejí vůči kolům přední nápravy opačně. Díky tomu se sníží potřebný poloměr pro otáčení a zvýší se tak manévrovatelnost vozidla. Naopak při vyšších rychlostech se kola zadní nápravy otáčejí stejně jako kola nápravy přední. Při prudších změnách směru vozidla (např. při předjíždění) se tímto docílí značného zlepšení směrové stability. [2]

Ovladač IFM by měl v rámci projektu AnteTUL sloužit především servisnímu technikovi pro ovládání jednotlivých aktuátorů. Ke stejnému účelu by měla sloužit i webová vizualizace. Servisní technik by tedy měl mít dvě možnosti, jak vyvíjený elektromobil ovládat.

Druhá kapitola této práce se snaží blíže představit projekt AnteTUL. Jsou zde podrobněji popsány cíle projektu a aktuální stav projektu.

Třetí kapitola se zabývá autonomními vozidly. Jednotlivé podkapitoly popisují úrovně automatizovaných systémů ve vozidlech, které zavedla společnost SAE International, současnou úroveň autonomních systémů ve vozidlech a porovnává systémy společností Tesla a Waymo LLC (dále jen Waymo).

Čtvrtá kapitole je věnována sběrnici CAN. Popis sběrnice CAN je zaměřený na strukturu zprávy ve specifikaci CAN 2.0A a CAN 2.0B a na protokol SAE J1939, který ze sběrnice CAN vychází.

Pátá kapitola doplňuje čtvrtou kapitolu a zaměřuje se přímo na výpočet CRC. V této kapitole jsou ukázány možné příklady výpočtu CRC a taky způsob výpočtu



CRC v praxi.

Kapitoly šest a sedm se zabývají popisem pohonu LA36 a ovladače IFM. V těchto kapitolách je probráno zapojení systémů, komunikace prostřednictvím sběrnice CAN a možné příklady použití. V osmé kapitole je popsán výsledek práce a poslední kapitola porovnává systém EPAS a pohon LA36.

## 2 Projekt AnteTUL

Projekt AnteTUL je projekt TUL, kterému se věnuje mezioborový tým Ústavu pro nanomateriály, pokročilé technologie a inovace TUL (zkráceně CxI TUL). Tento tým se skládá z pracovníků a studentů z několika fakult TUL. [3] Hlavní řešitel projektu doc. Michal Petřů vyjádřil cíle projektu AnteTUL těmito slovy:

*„Naším cílem není autonomní vozidlo pro silniční provoz. Zaměřujeme se na elektromobilitu, která bude pomáhat lidem a zvyšovat jejich bezpečnost na různých pracovištích. Dokončili jsme vývoj funkčního prototypu s autonomními prvky pod označením Generace 0. Ten je určen pro velké haly či skladiště a nyní vyvíjíme Generaci 1, která bude jezdit v náročném venkovním terénu, například na stavbách, v dolech nebo na nezpevněných plochách a dá se využít i pro inspekci v kontaminovaných oblastech. Naše systémy by se mohly uplatnit i v domácnostech, ale také jako pomocníci handicapovaných lidí.“ [3]*

Výstupem projektu AnteTUL by měly být tedy dva autonomní elektromobily. Jeden s označením Generace 0 a druhý s označením Generace 1. Celé ovládání těchto elektromobilů z hlediska uživatele je založeno na příkazu: Dojet z bodu A do bodu B. O veškerý pohyb elektromobilu se pak starají už jen výkonné IoT (z ang. Internet of Things) jednotky. Tyto jednotky vypočítávají požadovanou trajektorii v reálném čase a na základě dat ze senzorů vytvářejí řídicí příkazy pro jednotlivé aktuátory elektromobilu. Tyto příkazy jsou předávány PLC, které komunikuje se všemi aktuátory a dohlíží na správné provedení příkazů. [3]

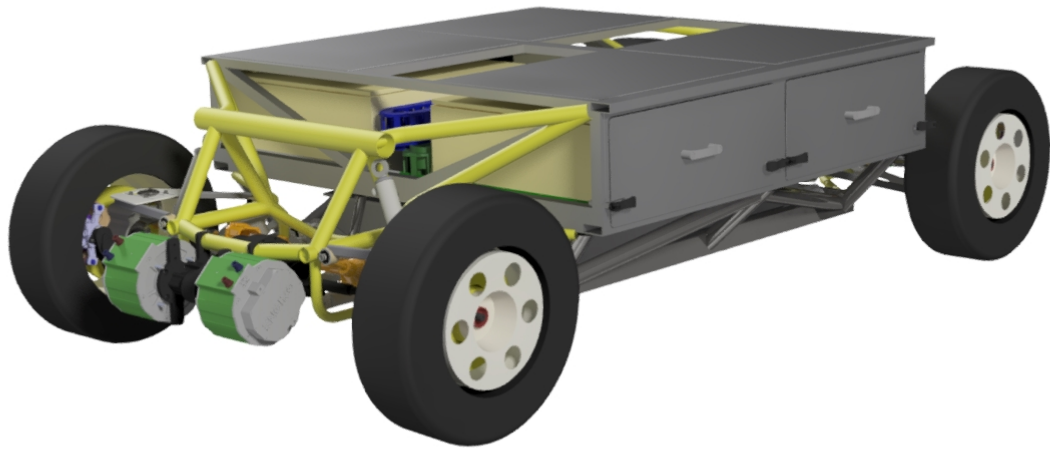
Projekt AnteTUL nespočívá jen na vytvoření funkčního autonomního elektromobilu. Projekt se zabývá např. i materiálovým výzkumem na odlehčené materiály, výběrem vhodných technologií pro skladování elektrické energie aj. Podrobnější popis projektu AnteTUL je možné najít v Online zpravodaji TUL <https://tuni.tul.cz/>. Také je možné se o něm dočíst v článku Autonomní vozidlo z Liberce pomůže zvyšovat bezpečnost přepravy v průmyslových halách i v terénu [3].

Na obr. 2.1 je vyobrazen elektromobil Generace 0 ve své holé podobě. V nejbližší době by mělo dojít k dokončení krytu, který zakryje elektronickou i mechanickou část elektromobilu a dodá tak vozidlu lukrativnější vzhled.



Obrázek 2.1: Generace 0 (zdroj autor)

V současnosti elektromobil Generace 1 není sestaven. Jeho dokončení se plánuje na konec roku 2022. Obr. 2.2 zobrazuje, jak by mohl elektromobil Generace 1 vypadat ve finální podobě.



Obrázek 2.2: Generace 1 (zdroj autor)

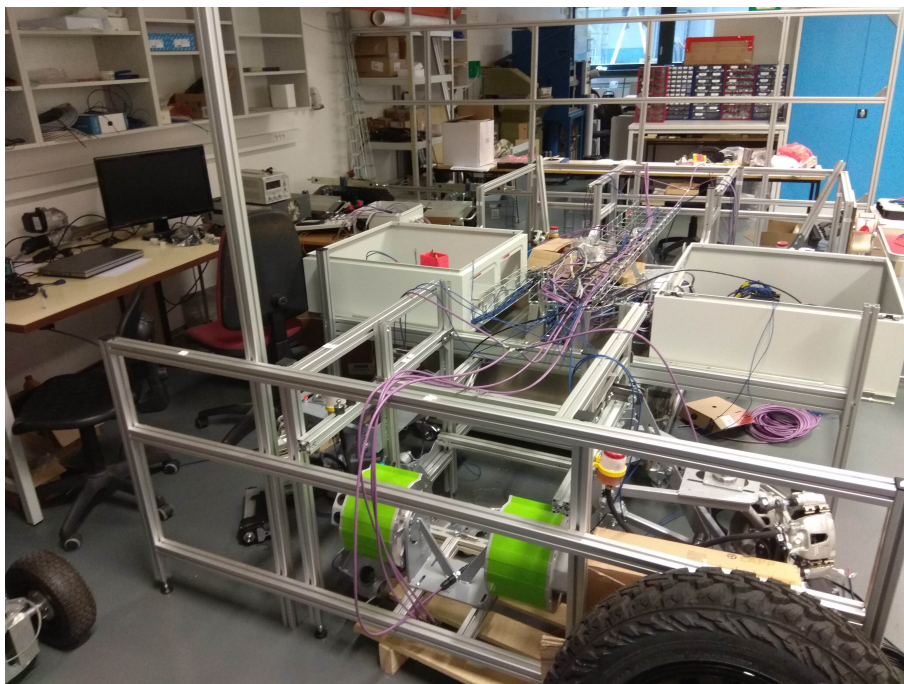
Zajímavostí na Generaci 1 je, že bude mít obě nápravy zcela identické. Nebude tedy možné říci, která náprava je přední a která zadní, jako to bývá u standardního automobilu. Z toho důvodu se rozhodlo o speciálním označení obou náprav. Jedna nese označení White (bílá) a druhá Black (černá). Obě nápravy budou barevně označeny danou barvou. Díky velkému kontrastu vybraných barev by mělo být jednoduché identifikovat, o jakou nápravu se jedná. Strany Generace 1 jsou označeny jako Levá a Pravá ve směru nápravy White.

Každé kolo elektromobilu bude poháněno vlastní pohonnou jednotkou. Ta se skládá z elektromotoru a převodovky. Kola obou náprav bude možné natáčet nezávisle na sobě. Navíc každé kolo bude s rámem elektromobilu spojeno přes hydraulický válec, který umožní nastavování výšky podvozku.

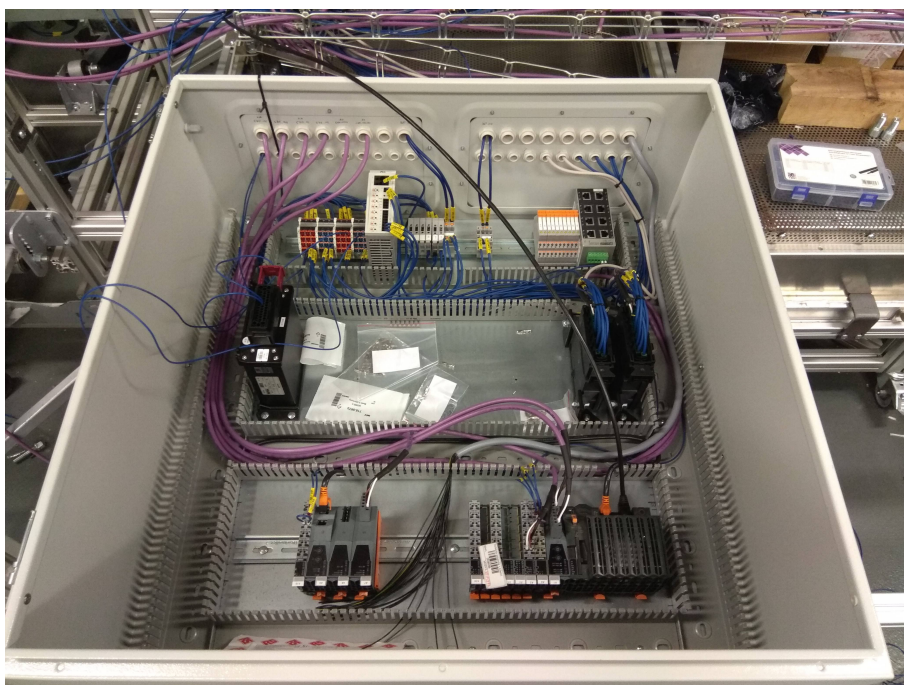
Na obr. 2.2 chybí modulární platforma, která bude umístěna na horní části elektromobilu a bude umožňovat přepravu materiálu a zařízení. Řídicí logika, hydraulické agregáty, technologie pro skladování elektrické energie a další části elektromobilu budou umístěny v prostoru mezi nápravami pod modulární platformou. Během provozu elektromobilu by byl přístup k těmto komponentám značně omezený. Z toho důvodu byl vymyšlen mechanismus, který umožňuje vysunutí skříní mimo oblast modulární platformy. V těchto skříních bude umístěna řídicí logika, hydraulické agregáty atd. Hlavní výhodou je, že při vysunutí skříně např. s řídicí logikou, budou všechny komponenty neustále propojeny. Není tak nutné při kontrole nebo výměně některé z komponent elektromobilu odpojovat celou řídicí logiku.

Aby bylo možné testovat jednotlivé aktuátory Generace 1, zapojit řídicí logiku a získat představu o umístění jednotlivých komponent na elektromobilu, bylo vytvořeno pracoviště s pevným rámem viz obr. 2.3. Na tento rám se dají namontovat

jednotlivé aktuátory, zapojit rozvaděč s řídicí elektronikou viz obr. 2.4, hydraulický agregát aj. Díky tomu bylo možné již před sestavením Generace 1 odzkoušet nejen komunikaci s řídicími jednotkami aktuátorů, ale i funkčnost rozhraní pro servisního technika.



Obrázek 2.3: Ukázka pracoviště (zdroj autor)



Obrázek 2.4: Rozvaděč s řídicí elektronikou (zdroj autor)

## 3 Autonomní vozidla

V magisterském projektu bylo probráno aktuální poznání autonomních užitkových vozidel. Tato kapitola rozšiřuje téma autonomních vozidel o další vozidla, u nichž plná automatizace ovlivní životy prakticky všech lidí. Jedná se o osobní automobily. V této kapitole jsou uvedeny jednotlivé úrovně systémů autonomního řízení<sup>1</sup> vozidel a aktuální poznání autonomních vozidel vztažené k datu 21.1.2022. Dále tato kapitola obsahuje popis systémů autonomního řízení společností Tesla a Waymo a porovnává je mezi sebou.

### 3.1 Úrovně automatizace

Mnoho výrobců automobilů a společností se zabývá automatizací vozidel. Aby bylo možné klasifikovat sofistikovanost jednotlivých technologií autonomního řízení, byl společností SAE International vytvořen standard SAE J3016.

SAE International dříve jen SAE (z ang. Society of Automobile Engineers, společnost automobilových inženýrů) ve svém standardu J3016 definovala celkem šest úrovní autonomního řízení vozidel. Tyto úrovně popisují celý rozsah automatizace vozidel, od vozidel bez jakéhokoliv systému autonomního řízení po vozidla, která jsou zcela autonomní. Je nutné podotknout, že tyto úrovně nerozdělují samotná vozidla, ale pouze jejich systémy. To znamená, že jedno vozidlo může disponovat i několika systémy autonomního řízení s různou úrovní autonomnosti. [4]

Úrovně automatizace je možné rozdělit do dvou skupin podle zapojení řidiče do řízení vozidla. V první a druhé úrovni má vozidlo pod kontrolou řidič, proto jsou autonomní systémy úrovně 1 a 2 označovány jako driver support (podpora řidiče nebo jízdní asistenti). Od třetí úrovně přebírají autonomní systémy plnou kontrolu nad vozidlem, pokud jsou zapnuty. Tyto systémy jsou označovány jako automated driving system (systémy autonomního řízení). [4]

Následující podkapitoly popisují všech šest úrovní systémů autonomního řízení. Použité překlady názvů jednotlivých úrovní odpovídají výrazům, které používá společnost Škoda Auto a.s. (dále jen Škoda) ve svém článku Pět stupňů k autonomnímu řízení [5].

---

<sup>1</sup>V rámci této práce výraz autonomní řízení (automated driving) představuje systém, který v automobilu nahrazuje osobu v roli řidiče. Tento systém dokáže ovládat jednotlivé části automobilu nebo celý automobil bez nutnosti zásahu lidského řidiče.

### 3.1.1 Úroveň 0 (Žádná automatizace)

V této úrovni vozidlo nedisponuje žádným systémem, který by dokázal (třeba jen částečně) řídit vozidlo. Do této úrovně se řadí i systémy aktivní bezpečnosti (např. parkovací senzory, hlídání mrtvého úhlu atd.). [4]

### 3.1.2 Úroveň 1 (Podpora řidiče)

Systémy patřící do této úrovně již jsou schopny vozidlo částečně řídit. Dokáží převzít kontrolu nad zatáčením (asistent pro udržování vozidla v jízdním pruhu) nebo nad akcelerací a decelerací vozidla (jakýkoliv tempomat), ale nedokáží provádět tyto akce současně. V případě, kdy systém částečně řídí vozidlo, musí mít řidič pod kontrolou zbývající část řízení. Pro tyto systémy také platí, že mohou fungovat jen za daných podmínek. Proto musí řidič na takové systémy dohlížet a být kdykoliv připraven převzít plnou kontrolu nad řízením vozidla. [4]

### 3.1.3 Úroveň 2 (Částečná automatizace)

Stejně jako v předchozí úrovni jsou i systémy druhé úrovně schopny zatáčet vozidlem a provádět akceleraci i deceleraci vozidla. Rozdíl je v tom, že v této úrovni dokáží systémy tyto akce provádět současně. Řidič tak nemusí během pohybu vozidla držet volant ani šlapat na pedály. Stále zde ale platí, že řidič musí neustále dohlížet na tyto systémy a být připraven daný úkol dokončit. Jedná se např. o systém automatického parkování. [4]

### 3.1.4 Úroveň 3 (Podmíněná automatizace)

Systémy třetí úrovně dokáží řídit celé vozidlo i za běžného provozu na pozemní komunikaci. Na tyto systémy řidič již nemusí neustále dohlížet. Oproti předchozí úrovni tyto systémy bez přestání monitorují vlastní činnost a okolní podmínky. Sami tak v dané situaci dohlíží nad tím, aby je bylo možné zapnout nebo udržet v chodu jen v případech, kdy jsou splněny všechny podmínky jejich provozu. Pokud nastane situace, kdy je systém třetí úrovně zapnut a za daných okolností již není schopen pracovat, v dostatečném předstihu požádá řidiče o převzetí kontroly nad vozidlem. V takových případech by řidič měl pokračovat v daném úkonu i s vypnutým systémem nebo by měl uvést vozidlo do stavu, ve kterém se minimalizuje riziko dopravní nehody (např. odbočení na krajnici vozovky). [4]

Systémy třetí úrovně mohou disponovat i záložním systémem, který při zadaných podmínkách může v problémových situacích zastoupit řidiče a sám tak uvést vozidlo do stavu s minimálním rizikem. Tato funkce se ale od těchto systémů neočekává. [4]

Příkladem může být systém, který dokáže řídit vozidlo na dálnici při nízké rychlosti. Jedná se především o dopravní zácpy, kdy je většinou potřeba vozidlo neustále rozjíždět a zastavovat. [4]

### 3.1.5 Úroveň 4 (Vysoká automatizace)

Ve čtvrté úrovni autonomního řízení již systémy zvládnou řídit vozidlo bez jakékoliv nutnosti zásahu řidiče. Navíc tyto systémy musí obsahovat záložní systém, který uvede vozidlo do stavu s minimální rizikem dopravní nehody. Nutná existence záložního systému je hlavním rozdílem mezi úrovněmi 3 a 4. Řidič tak nemusí na funkci systému vůbec dohlížet a ani nemusí být připraven v případě požádání převzít kontrolu nad vozidlem. Tyto systémy mohou být navrženy jak pro celou cestu, tak i pro jednotlivé dílčí části cesty (provoz na dálnici, parkování na parkovišti). Pokud řidič aktivuje např. systém autonomního řízení na dálnici, s velkou pravděpodobností nastane situace, kdy bude nutné sjet vozidlem z dálnice. Systém už dopředu vyhodnotí, že se vozidlo ocitne mimo oblast jeho použití, a vyzve řidiče, aby převzal kontrolu nad vozidlem. Řidič ale může v dané situaci např. spát nebo může mít jiný důvod, proč nepřevzme kontrolu nad vozidlem. V takovém případě se aktivuje záložní systém, který uvede vozidlo do stavu s minimálním rizikem. [4]

Díky těmto vlastnostem je možné tyto systémy používat i ve vozidlech, ve kterých řidič není vůbec přítomen a ovládat je tedy dálkově. Tato vozidla musí mít ale předem definovanou trajektorii pohybu a v rámci svého provozu musí mít vymezenou oblast pohybu (vojenská základna, kampus univerzity). [4]

### 3.1.6 Úroveň 5 (Plná automatizace)

Poslední úroveň je nejvyšším stupněm autonomního řízení. Systém této úrovně dokáže řídit vozidlo na pozemní komunikaci kdekoli na světě za stejných podmínek, ve kterých by dokázal vozidlo řídit zkušený lidský řidič. Neexistují tedy žádné podmínky (počasí, lokalita, denní doba), které by říkaly, kdy a kde je možné systém páté úrovně provozovat. Mohou nastat ale situace, ve kterých by ani lidský řidič nemohl dál vozidlo řídit (sněhová bouře, zaplavená silnice). V takových situacích si nedokáže poradit ani vozidlo vybaveno systémem Plné automatizace. Za těchto podmínek se aktivuje záložní systém, vozidlo se zastaví a čeká dokud se nepříznivé podmínky nezlepší. Přítomnost lidského řidiče v těchto vozidlech je zcela nepotřebná. I v případě poruchy systému dokáže vozidlo reagovat a téměř okamžitě dojde opět k automatickému spuštění záložního systému a vozidlo se zastaví. [4]

## 3.2 Současná úroveň autonomního řízení

V současné době se na silnici ve většině případů vyskytují automobily se systémy nulté a první úrovně. Tedy automobily, které disponují pouze výstražnými systémy nebo např. již zmíněným tempomatem. Ojedinele je možné setkat se i s vozy, jenž jsou vybaveny i systémy druhé úrovně. Na druhé úrovni autonomní řízení končí a s dalšími úrovněmi není možné se setkat.

Ačkoliv se různé výzkumné a automobilové společnosti zabývají i systémy vyšších úrovní mj. i Škoda (ta pracuje na systému 4. úrovně) není tyto systémy legálně možné využívat v běžném provozu. Hlavní překážkou, která brání použití těchto systémů, je legislativa. Ta na takové systémy není připravena. [5] Toto omezení platilo



do roku 2020.

11. listopadu 2020 získala totiž společnost Honda Motor Company, Ltd. od MLIT (z ang. Japanese Ministry of Land, Infrastructure, Transport and Tourism) certifikát na používání jejich systému autonomního řízení 3. úrovně. Aby bylo možné legálně využívat tento systém, muselo MLIT v Japonsku upravit bezpečnostní standardy a zákon o silničních vozidlech. [6] Díky tomu se stalo Japonsko první zemí na světě, ve které je možné využívat systém 3. úrovně tedy systém Podmíněné automatizace.

Tento průlomový systém nese název Traffic Jam Pilot (Pilot dopravní zácpy). Ten je součástí nového bezpečnostního systému Honda SENSING Elite, kterým disponuje vozidlo s označením Honda Legend Hybrid EX.<sup>2</sup> Traffic Jam Pilot umožňuje za určitých podmínek převzít zcela kontrolu nad řízením vozidla např. v případě, kdy se vozidlo vyskytne v dopravní zácpě na rychlostní komunikaci. Hlavní řídicí jednotka určuje polohu vozidla z trojrozměrných map s vysokým rozlišením a GNSS (z ang. Global Navigation Satellite System). Stav vozovky se určuje pomocí externích senzorů, které detekují okolí vozidla v rozsahu 360 stupňů. Na základě těchto dat řídicí jednotka rozhoduje jaké akční zásahy má vykonat (akcelerace, decelerace, zatáčení), aby se dosáhlo vysoce kvalitní a plynulé jízdy. Při vývoji tohoto systému byla kladena nejvyšší priorita na bezpečnost a spolehlivost. Proto bylo během vývoje použito 10 milionů simulací se situacemi, které mohou nastat během reálného provozu. Také se provedlo několik jízd s testovacími vozidly na rychlostních komunikacích. Celkový rozsah testovacích jízd se pohyboval kolem 1,3 milionů kilometrů. Pokud je Traffic Jam Pilot aktivován, může řidič pomocí displejů ve vozidle sledovat televizi, DVD, ovládat navigační systém a nastavit cílovou adresu. Díky tomu by se měl snížit stres řidiče a zmírnit pocit únavy, když se vyskytne v dopravní zácpě. [7]

Netrvalo to moc dlouho a došlo k povolení využívání systému Podmíněné automatizace v další zemi. Tentokrát se jedná o Německo. V této zemi se místnímu výrobcí automobilů Mercedes Benz taktéž povedlo vyvinout systém, jenž splňuje 3. úroveň autonomního řízení. Navíc získal i povolení k jeho užívání (zatím pouze v Německu). [8] Tímto systémem s označením Drive Pilot (Pilot řízení) jsou vybavena vozidla Mercedes Benz třídy S. Systém je možné aktivovat pouze na dálnicích při rychlostech menší než 37 mph (60 km/h) a pouze za určitých povětrnostních podmínek. Opět se tedy jedná o systém, který je určený především pro dopravní zácpy. Pokud je systém aktivní, snaží se dle standardu J3016 předvídat dění na vozovce a případně požádat řidiče o převzetí řízení, pokud má nastat situace, kdy již není možné tento systém používat. [9]

Standardní vozidla Mercedes Benz třídy S mají ve své výbavě již zahrnuté parkovací senzory v předním a zadním nárazníku a kamery ve zpětných zrcátkách s rozsahem 360 stupňů. Vozidla, jenž disponují systémem Drive Pilot, mají ve své výbavě navíc radary umístěné v každém rohu vozidla, dálkový radar a LIDAR směřující před vozidlo, stereo kameru umístěnou v horní části čelního skla, běžnou kameru snímající dění za vozidlem a senzor vlhkosti v předním kole. Zadní kamera slouží k detekci blikajících světel. Spolu s mikrofonom umístěným ve vozidle dokáže systém vyhodnotit přítomnost zásahového vozidla (vozidlo s přednostním právem jízdy)

---

<sup>2</sup>Leasingový prodej těchto vozidel byl zahájen 5. března 2021.

a předá kontrolu nad vozidlem řidiči. Senzor vlhkosti umožňuje zjistit stav vozovky. Konkrétně pomocí snímání hlučnosti kola zjišťuje, jak moc je vozovka mokrá. Pokud bude vozovka příliš mokrá, je pravděpodobné, že prší a déšť může rušit kamery, radary a LIDAR. V takovém případě nejsou splněny podmínky pro používání systému dle stanovených limitů a řízení vozidla je opět předáno řidiči. Stanovené limity, při kterých je možné používat Drive Pilot, nejsou dány omezenými možnostmi použitých technologií, ale jsou především uměle zavedeny výrobcem z důvodu zvýšení bezpečnosti a spolehlivosti. [9]

V budoucnosti se plánuje Drive Pilot vylepšit tak, aby jej bylo možné používat i při vyšších rychlostech. [9]

## 3.3 Tesla a Waymo

Jak společnost Testa, tak i Waymo mají velmi podobný cíl. Chtějí vytvořit systém autonomního řízení, který dokáže fungovat zcela sám bez nutné přítomnosti nebo zásahu řidiče. K dosažení tohoto cíle mají obě tyto společnosti jiný přístup.

### 3.3.1 Tesla

Společnost Tesla je největším výrobcem elektromobilů na světě. Cíl této společnosti je na automobilku poněkud neobvyklý. Tesla totiž nechce prodávat co nejvíce elektromobilů s co největším ziskem jako většina automobilek, ale chce urychlit přechod k udržitelné energii. V souvislosti s tímto přechodem se Tesla snaží i usnadnit jiným společnostem výrobu elektromobilů tím, že jim dává své patenty na technologie volně k dispozici. Výjimkou jsou patenty na baterie, které k dispozici nejsou. [10]

Od roku 2014 je majitelům vozů Tesla k dispozici jízdní asistent s názvem Autopilot. Ten může během jízdy dopomáhat řidiči zatáčením při jasně viditelných pružích nebo měnit rychlost vozidla v závislosti na aktuálním provozu vozovky. Systém Autopilot je možné rozšířit o schopnost FSD (z ang. Full Self Driving). Tato schopnost přináší další funkce, které může řidič využít. Jedná se např. o automatické zaparkování vozidla a přivolání vozidla v situaci, kdy je vozidlo zaparkované v úzkém prostoru. V takovém případě řidič nemusí být ani ve vozidle, ale musí vozidlo celou dobu sledovat. [11]

Z názvů systému Autopilot a FSD by bylo možné usuzovat, že se při jejich použití stane vozidlo zcela autonomní. To ale v současné době není pravda. Oba tyto systémy jsou jízdními asistenty, což vyžaduje přítomnost řidiče. Navíc během jejich provozu musí mít řidič ruce na volantu a být kdykoliv připraven zasáhnout a převzít řízení vozidla. [11] Z výše zmíněných vlastností systému Autopilot a FSD vyplývá, že se jedná o 2. úroveň automatizace.

Společnost Tesla oba tyto systémy neustále vyvíjí a umožňuje majitelům vozidel Tesla systém Autopilot a FSD bezdrátově upgradovat. [11]

### 3.3.2 Waymo

Waymo je americká společnost zabývající se vývojem technologií autonomního řízení. Waymo je dceřinou společností Alphabet Inc. (mateřské společnosti Google). [12] Cílem společnosti Waymo je umožnit lidem, aby se spolu se svými věcmi dostali bezpečně a snadno tam, kam chtějí. [13]

Sama společnost Waymo se neoznačuje jako výrobce automobilů s autonomní řízením, ale jako technologická firma, která se zaměřuje na autonomní řízení. K dispozici má desítky tisíc vozidel, z nichž jsou všechny vybaveny rozsáhlým množstvím senzorů, které umožňují autonomní řízení. [12]

V roce 2015 provedla společnost Waymo první autonomní jízdu na veřejných komunikacích s vozem Firefly. Tento vůz disponoval vlastními senzory, počítači a systémy pro brzdění a zatáčení, aniž by byl ve voze namontovaný volant nebo pedály. Toho roku podnikl nevidomý člověk Steve Mahan první plně autonomní jízdu na veřejných komunikacích na světě. O tři roky později spustila společnost Waymo i první komerční službu na světě pro přivolání autonomního vozidla. [13]

### 3.3.3 Porovnání

#### Senzory

Pro detekci objektů a virtualizaci prostoru kolem vozidla využívá Waymo rozsáhlé množství senzorů jako jsou kamery, radary nebo LIDAR. Tyto senzory spolu s detailními mapami vozovky vytváří model světa v rozsahu 360°. [12]

Tesla také používá kamery a jiné senzory, ale v jejich vozech není možné najít LIDAR. Místo LIDARu se snaží Tesla využívat kamerových systémů. Je to z toho důvodu, že LIDAR je velký, drahý a v současné době obsahuje pohyblivé části. Jeho použití sice usnadňuje detekci objektů kolem vozidla, ale je také velmi nákladné. [12]

#### Sběr dat pro AI

Obě společnosti využívají pro svůj systém umělou inteligenci (AI z ang. Artificial Intelligence). Aby AI fungovala spolehlivě, je potřeba ji učit. K učení AI používají obě společnosti jiná data. Společnost Waymo využívá sofistikovaných počítačových simulací a dat z vozidel, které byly testovány na skutečných silnicích. [12]

Oproti tomu Tesla využívá pouze data ze skutečného provozu. Tato data shromažďuje ze stovek tisíc vozidel, které řídí lidé spolu s jízdními asistenty. [12]

#### Reálné porovnání

3.3.2021 se objevilo na internetovém serveru YouTube video z kanálu Whole Mars Catalog s názvem Waymo vs Tesla FSD Beta. Jedná se nejspíše o první porovnání autonomních technologií společností Tesla a Waymo. [14]

Toto video bylo probráno v článku Tesla vs Waymo – Podívejte se na zřejmě první reálný souboj autonomních systémů. Autor článku Luboš Srb tento souboj

zhodnotil slovy:

*„...testování v reálném provozu proběhlo ve Spojených státech v oblasti, kterou už mají řekněme oba systémy zvládnout. Tesla i Waymo si vedou velmi dobře a zvládnou s přehledem dovést cestujícího až do cíle, přičemž během jízdy je možné sledovat na obrazovce v autech vše, co vidí i autopilot.*

*Problémy ani jednomu z porovnávaných autonomních systémů nedělají na běžné silnici ostatní auta, kolemjdoucí, ani třeba semaforey. Nicméně do cíle zvládne rychleji dojet Tesla, které stačí zhruba o 3 minuty méně než systému Waymo, ten trasu projel za zhruba 8 minut.“ [14]*

K dalšímu porovnání došlo 13.11.2021. V tento den nahrál kanál CallasEV opět na internetový server Youtube video. Na tomto videu se taktéž porovnávají jízdní schopnosti systému FSD ve vozidle Tesla se schopnostmi systému od společnosti Waymo ve vozidle Chrysler. Obě vozidla měla stejnou cílovou destinaci a startovala ze stejného místa. Na začátku videa bylo vozidlo Tesla již připraveno a čekalo se na přijetí vozidla Chrysler. To bylo přivoláno pomocí aplikace od společnosti Waymo a přijelo zcela samo bez přítomnosti řidiče. Na videu je možné vidět, že obě vozidla zvolila pro dosažení cíle jinou trasu a musela překonat mnohé překážky. V obou případech se vozidlům povedlo dojet do svého cíle s dvouminutovým rozdílem ve prospěch Tesly. Na videu není ale patrné, že za volantem vozu Tesla sedí řidič, který musí mít ruce na volantu. Oproti tomu ve voze Chrysler žádný řidič není a pasažér sedí na zadních sedačkách. [15]

## 4 CAN

Průmyslová sběrnice CAN (z ang. Controller Area Network) byla již částečně popsána v rámci magisterského projektu. Z toho důvodu se tato kapitola pouze zaměřuje na strukturu zprávy<sup>3</sup> na sběrnici CAN se standardním a rozšířeným identifikátorem a na protokol SAE J1939, který je na této sběrnici postaven.

### 4.1 Struktura zprávy specifikace CAN 2.0A

Klasický protokol CAN umožňuje posílat více druhů zpráv, jenž mají různé formáty. Jedná se o formát určený pro přenos dat (Data Frame), formát umožňující žádost o data (Remote Frame), formát určený pro oznámení chyby (Error Frame) a formát, který signalizuje příliš velké zatížení sběrnice a požaduje delší prodlevy mezi vysíláním zpráv (Overload Frame). [16] Pokud je komunikace na sběrnici bezchybová, dochází k vysílání pouze datových zpráv nebo zpráv s žádostí o data. V rámci této kapitoly jsou popsány pouze zprávy určené pro přenos dat (Data Frame). Důvodem je, že formáty Error Frame a Overload Frame se na sběrnici vyskytují pouze v případě chybové události a formát Remote frame není v rámci komunikace mezi nadřazeným systémem a řídicími jednotkami vůbec využit.

Pro přenos dat (Data Frame i Remote Frame) je možné použít dva formáty zprávy. Tyto formáty jsou označovány jako CBFF (z ang. Classical Base Frame Format, standardní formát zprávy) a CEFF (z ang. Classical Extended Frame Format, rozšířený formát zprávy). Struktura obou formátů je velmi podobná. Hlavní rozdíl je, že standardní formát využívá identifikátor zprávy o délce 11 bitů a rozšířený formát má identifikátor o délce 29 bitů (proto rozšířený). [17]

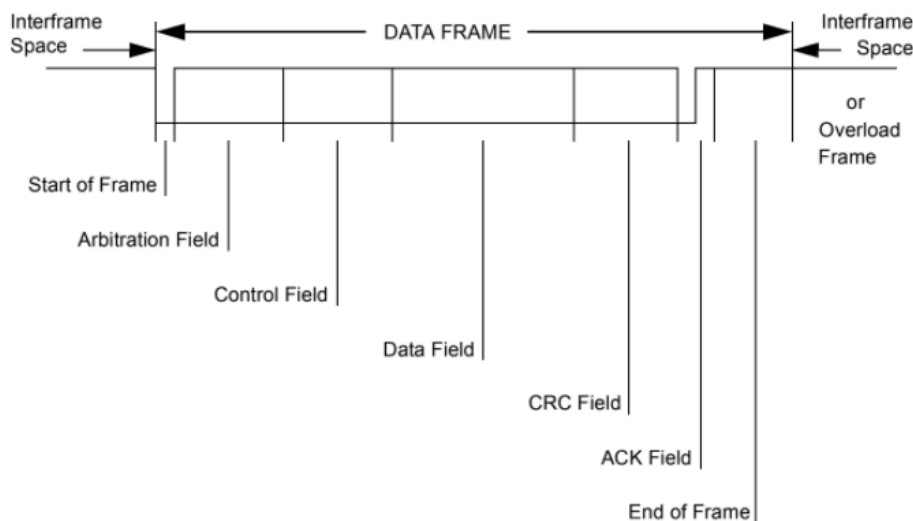
Použití jednotlivých formátů identifikátorů záleží na specifikaci sběrnice CAN. Pro verzi CAN 2.0 existují dvě specifikace. Tyto specifikace jsou označovány jako A a B. Specifikace CAN 2.0A umožňuje používat pouze standardní formát zprávy (tzn. pouze 11bitový identifikátor) a specifikace CAN 2.0B dovoluje použití obou formátů (tzn. 11bitový i 29bitový identifikátor). [16]

Na obr. 4.1 je zobrazena struktura datové zprávy. Jednotlivá pole, jenž datová zpráva obsahuje, jsou popsány v následujících podkapitolách. Vzhledem k rozdílům

---

<sup>3</sup>V anglickém jazyce je pro komunikaci na sběrnici CAN použit termín frame. Ten se v některých českých textech překládá jako rámec. Použití slova rámec při popisu komunikace na sběrnici CAN není úplně zřetelné, proto je v této práci místo slova rámec použito slovo zpráva.

mezi specifikacemi CAN 2.0A a CAN 2.0B je popis následujících částí vztažený pouze ke specifikaci CAN 2.0A, která umožňuje použití pouze 11bitového identifikátoru. Standardní formát a rozšířený formát specifikace CAN 2.0B je probrán níže.



Obrázek 4.1: Struktura datové zprávy (zdroj [16])

#### 4.1.1 Začátek zprávy (Start of Frame)

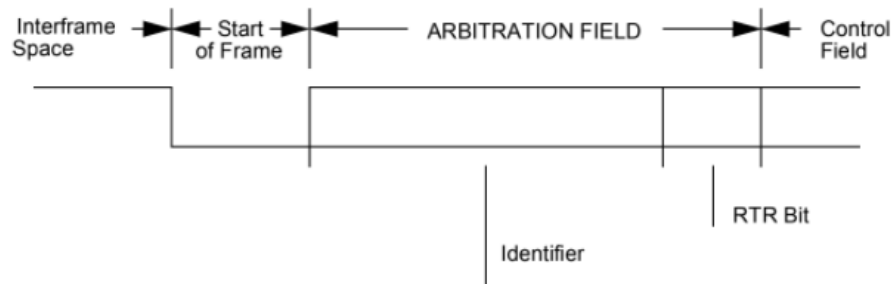
Každá zpráva poslaná po sběrnici CAN začíná bitem SOF (z ang. Start Of Frame, začátek zprávy). Tento bit je vždy dominantní (logická hodnota 0) a umožňuje všem zařízením na sběrnici, aby se mohla synchronizovat po dobu přenosu zprávy. [16] V případě, že dojde k vyslání celé zprávy nebo žádné zařízení nemá potřebu vysílat, je na sběrnici neustále recesivní hodnota (logická 1). Díky tomu, že SOF bit má dominantní hodnotu, je zaručeno, že ostatní zařízení na sběrnici zaznamenají začátek vysílání, co nejdříve a mohou se tak synchronizovat.

#### 4.1.2 Arbitrážní pole (Arbitration Field)

Po bitu SOF následuje arbitrážní pole. Toto pole obsahuje 11bitový identifikátor a RTR (z ang. Remote Transmission Request) bit. [16]

Jednotlivé bity identifikátoru jsou řazeny v sestupném pořadí. Hodnota bitu s indexem 10 se objeví na sběrnici jako první a hodnota bitu s indexem 0 jako poslední. [16]

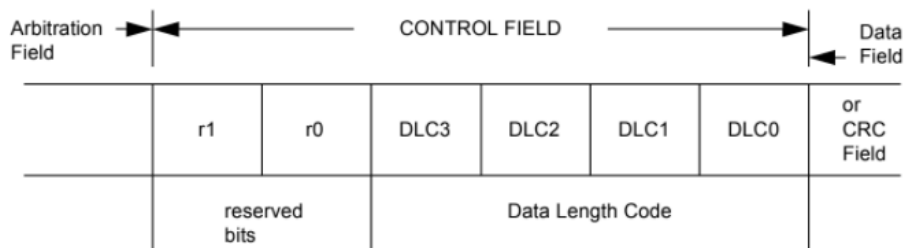
RTR bit slouží pro definování, zda se jedná o datovou zprávu nebo o zprávu s žádostí o data. Pokud chce zařízení poslat datovou zprávu, musí být hodnota RTR bitu dominantní. V případě poslání zprávy se žádostí o data musí být hodnota tohoto bitu recesivní. [16]



Obrázek 4.2: Arbitrační pole (zdroj [16])

### 4.1.3 Řídicí pole (Control Field)

Řídicí pole se skládá z celkového počtu 6 bitů. První dva bity jsou rezervovány pro možné budoucí rozšíření a během vysílání musí mít dominantní hodnotu. Hodnota zbylých 4 bitů je rovna počtu bajtů dat, které zpráva přenáší. Tyto bity jsou označeny jako DLC (z ang. Data Length Code, kód délky dat).



Obrázek 4.3: Řídicí pole (zdroj [16])

Pořadí vyslání bitů je opět sestupné. Datová zpráva umožňuje poslat data v rozsahu 0–8 bajtů. Kombinováním 4 bitů je možné získat až 16 hodnot. To znamená, že hodnoty 9 až 15 nejsou využity. Tab. 4.1 ukazuje všechny kombinace DLC bitů a jim odpovídající počet přenášených bajtů dat, které je možné v rámci jedné zprávy poslat.

Tabulka 4.1: Kombinace DLC bitů (podle [16])

Počet bajtů dat	DLC3	DLC2	DLC1	DLC0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0

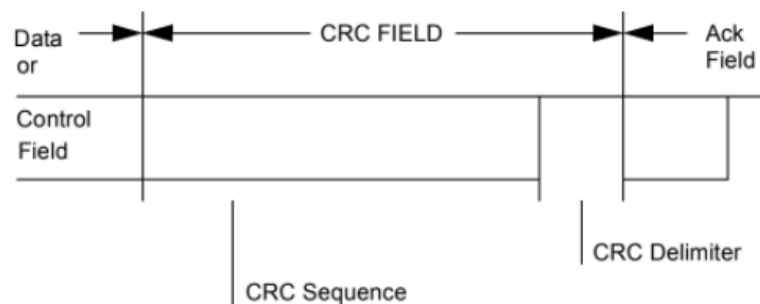
#### 4.1.4 Datové pole (Data Field)

Obsahem datového pole jsou bajty dat, které chce zařízení poslat po sběrnici. Jak bylo řečeno, je možné poslat až 8 bajtů dat. Každý bajt obsahuje 8 bitů. [16] Jedna datová zpráva může tedy obsahovat nanejvýše 64 bitů dat.

#### 4.1.5 CRC pole (CRC Field)

CRC (z ang. Cyclic Redundancy Check, cyklický redundantní součet) pole obsahuje CRC sequence a CRC delimiter. CRC sequence se skládá z 15 bitů, jejichž hodnota odpovídá zbytku po polynomiálním dělení vstupního polynomu a klíče CRC viz kapitolu 5. [16]

CRC delimiter je bit, jenž následuje bezprostředně za CRC sequence a musí mít recesivní hodnotu. [16]



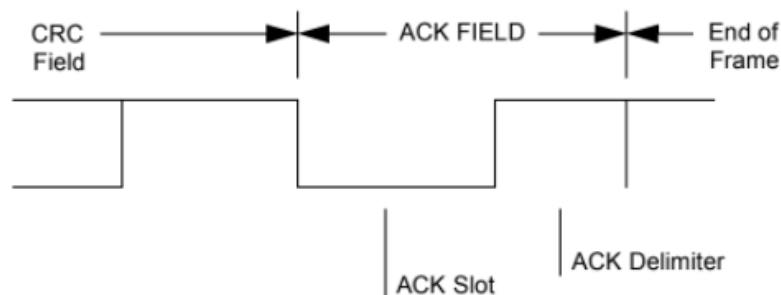
Obrázek 4.4: CRC pole (zdroj [16])

#### 4.1.6 ACK pole (ACK Field)

ACK pole (z ang. Acknowledge field, pole potvrzení) má délku 2 bitů. Jeden bit je označován jako ACK Slot a druhý jako ACK Delimiter (oddělovač). Tyto bity slouží pro potvrzení, zda vysílanou zprávu nějaké zařízení přijalo a není např. někde přerušena sběrnice. Zařízení, které zprávu vysílá, nastaví oba tyto bity na recesivní hodnotu. Pokud některé zařízení přijme zprávu s odpovídajícím CRC polem (zpráva byla přijata v pořádku), nastaví ACK Slot bit na dominantní hodnotu. Tím dojde ke změně hodnoty ACK Slot bitu z recesivní na dominantní a vysílající zařízení zjistí, že zpráva byla v pořádku odeslána a přijata. [16]

Hodnota bitu ACK Delimiter musí být vždy recesivní. Díky tomuto nastavení je ACK Slot bit vždy mezi dvěma recesivními bity (CRC Delimiter a ACK Delimiter). [16]





Obrázek 4.5: ACK pole (zdroj [16])

### 4.1.7 Konec zprávy (End of Frame)

Konec zprávy je na sběrnici CAN indikován vysláním posloupnosti sedmi bitů s re-cesivní hodnotou. Těchto sedm bitů je označováno jako EOF (z ang. End Of Frame, konec zprávy). Tyto bity musí následovat bezprostředně za ACK polem jak při přenosu datové zprávy, tak při přenosu zprávy s žádostí o data. [16] Teprve po odeslání těchto bitů může dojít k vyslání další zprávy.

## 4.2 Struktura zprávy specifikace CAN 2.0B

Jak bylo řečeno výše, specifikace CAN 2.0B umožňuje použití obou formátů identifikátoru tedy standardní i rozšířený formát. Standardní formát specifikace CAN 2.0B není zcela stejný jako standardní formát specifikace CAN 2.0A. [16] Aby bylo možné rozlišit, zda se jedná o standardní nebo rozšířený formát zprávy, byl mj. do specifikace CAN 2.0B přidán bit IDE (z ang. Identifier Extension, rozšířený identifikátor). Přidáním tohoto bitu se upravila část standardního formátu zprávy CAN 2.0B a vznikl tak rozdíl oproti specifikaci CAN 2.0A.

Hlavní struktura všech formátů datových zpráv na sběrnici CAN ve verzi 2.0 je stejná a odpovídá obr. 4.1. Datová zpráva se opět skládá z bitu SOF, z arbitrážního pole, z řídicího pole, z datového pole, z CRC pole, z ACK polem a z EOF. Rozdíl mezi jednotlivými formáty je ve struktuře arbitrážního a řídicího pole. [16] Zbylé části zprávy jsou zcela identické pro všechny formáty, proto se zde již znovu nepopisují.

### 4.2.1 Arbitrážní pole (Arbitration Field)

Arbitrážní pole rozšířeného identifikátoru se skládá z 29bitového identifikátoru, z SRR (z ang. Substitute Remote Request) bitu, z IDE bitu a z RTR bitu. Jednotlivé bity identifikátoru jsou opět řazeny sestupně. [16]

Aby bylo možné využívat oba formáty identifikátorů na jedné sběrnici CAN (specifikace CAN 2.0B), je první část arbitrážního pole stejná. Pro oba formáty obsahuje arbitrážní pole nejprve prvních (nejvyšších) 11 bitů identifikátoru. U rozšířeného formátu následuje za těmito 11 bity SRR bit. SRR bit je ve zprávě umístěn

na stejné pozici jako RTR bit u standardního formátu. Z toho důvodu se dá říct, že nahrazuje (substituuje) RTR bit. SRR bit má recesivní hodnotu. Po SRR bitu následuje bit IDE. Pro standardní formát je IDE bit roven dominantní hodnotě a je součástí řídicího pole.<sup>4</sup> U Rozšířeného formátu má IDE bit recesivní hodnotu a je součástí arbitrážního pole. Recesivní hodnota IDE bitu značí, že se jedná o zprávu s rozšířeným identifikátorem a IDE bit bude následován zbylými 18 bity identifikátoru. Arbitrážní pole rozšířeného identifikátoru je zakončené RTR bitem, který má stejný význam jako u standardního formátu zprávy ve specifikaci CAN 2.0A i 2.0B. [16]

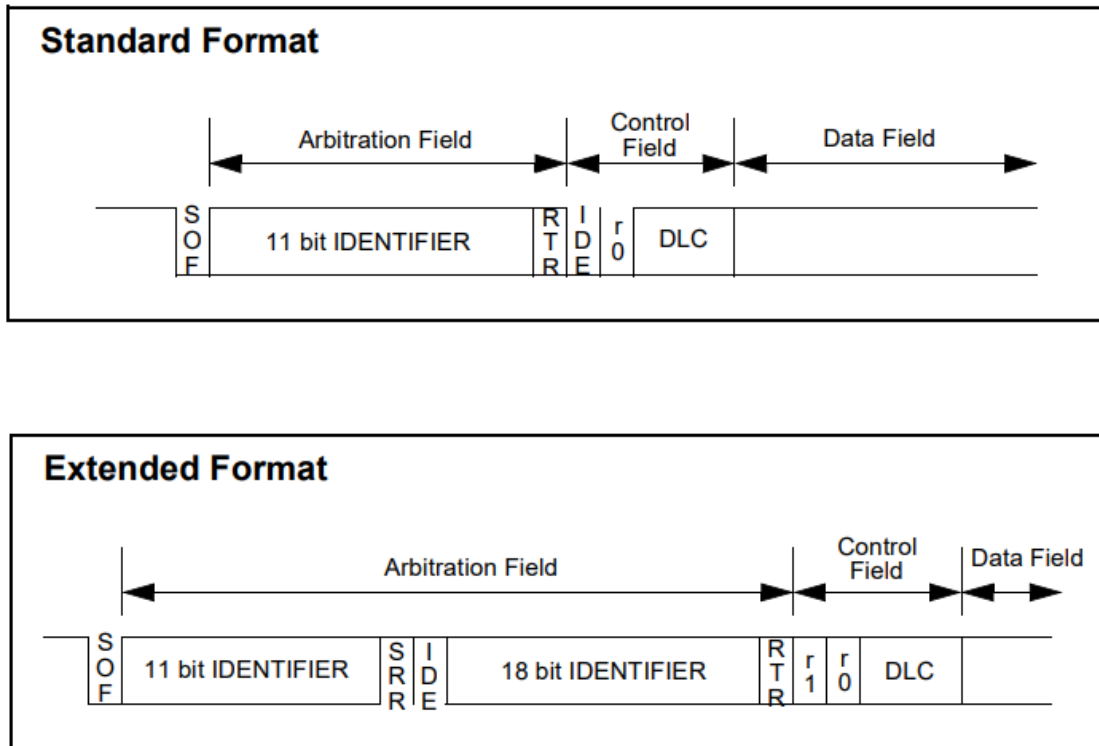
## 4.2.2 Řídicí pole (Control Field)

Řídicí pole rozšířeného formátu zprávy ve specifikaci CAN 2.0B je stejné jako řídicí pole standardního formátu zprávy ve specifikaci CAN 2.0A. Pole obsahuje celkem 6 bitů. První dva bity jsou rezervovány pro možné budoucí použití a musí mít dominantní hodnotu. Hodnota zbylých 4 bitů opět definuje počet přenášených bajtů dat. [16]

Standardní formát identifikátoru ve specifikaci CAN 2.0B obsahuje v řídicím poli navíc již zmíněný IDE bit. Pro standardní formát má IDE bit dominantní hodnotu. [16] Přítomnost IDE bitu ve standardním formátu je jediný rozdíl mezi standardním formátem zprávy ve specifikaci CAN 2.0A a CAN 2.0B. Tento rozdíl je možné vidět při porovnání obrázku 4.3, který zobrazuje řídicí pole standardního formátu zprávy specifikace CAN 2.0A, s obrázkem 4.6, který zobrazuje řídicí pole standardního i rozšířeného formátu zprávy specifikace CAN 2.0B.

---

<sup>4</sup>U standardního formátu zprávy ve specifikaci CAN 2.0A je na této pozici umístěn jeden ze dvou bitů rezervovaných pro budoucí využití.



Obrázek 4.6: Standardní a rozšířený formát zprávy specifikace CAN 2.0B (zdroj [16])

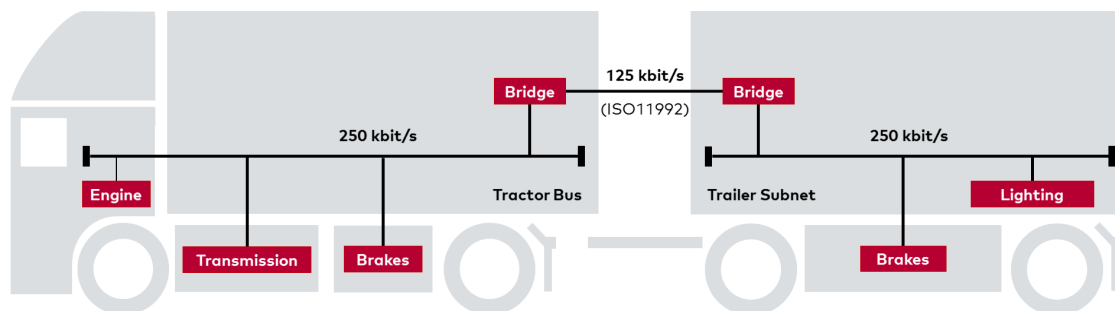
## 4.3 Protokoly založené na sběrnici CAN

Dle referenčního modelu ISO/OSI popisuje verze CAN 2.0 pouze linkovou a datovou vrstvu. [16] Existuje ale mnoho protokolů založených na sběrnici CAN, které popisují vyšší vrstvy modelu ISO/OSI. Většina těchto protokolů odpovídá aplikační a transportní vrstvě, která obsahuje některé funkce prezentační, relační a síťové vrstvy. Mezi tyto protokoly patří např. CANopen, CANopen FD, protokol SAE J1939, Isobus, Devicenet aj. Tyto protokoly jsou založené buď na klasickém protokolu CAN, nebo na protokolu CAN FD. [18]

Protokol CAN FD (z ang. Flexible Data-rate) je vylepšenou verzí klasického protokolu CAN. Klasický CAN umožňuje posílat v jedné zprávě pouze 8 bajtů dat s nejvyšší možnou rychlostí 1 Mbit/s. Protokol CAN FD dokáže přenášet až 64 bajtů dat v jedné zprávě s rychlostí vyšší než 1 Mbit/s. [19]

### 4.3.1 SAE J1939

Protokol SAE J1939 je otevřený standard, který se běžně využívá pro sériovou komunikaci mezi řídicími jednotkami v motorových vozidlech. Mezi tyto řídicí jednotky typicky patří řídicí jednotka motoru, převodovky, brzd, přístrojové desky atd. [20]



Obrázek 4.7: Typické použití protokolu SAE J1939 ve vozidle (zdroj [20])

Protokol SAE J1939 je postavený na sběrnici CAN. To znamená, že linková a datová vrstva protokolu SAE J1939 vychází z linkové a datové vrstvy klasického protokolu CAN. Oproti klasickému protokolu CAN obsahuje protokol SAE J1939 i vyšší vrstvy modelu ISO/OSI. Tyto vyšší vrstvy dále zpracovávají jednotlivé části zpráv a dávají přenášeným datům další významy.

Většina zpráv definovaná protokolem SAE J1939 je určena pro všechna zařízení, která jsou připojena na sběrnici. To znamená, že přenášená data na sběrnici nemají cílovou adresu a mohou je přijímat všechna zařízení. Tento typ zprávy se označuje jako PDU 2 (z ang. Protocol Data Unit). Pokud musí být zpráva doručena pouze jednomu konkrétnímu zařízení, může identifikátor zprávy obsahovat cílovou adresu. Tento typ zprávy je označován jako PDU 1. Identifikátor, který používá protokol SAE J1939 obsahuje 29 bitů. Jedná se tedy o specifikaci CAN 2.0B. 29bitový identifikátor zprávy je rozdělen na několik částí viz obr. 4.8. [21]

PRIORITY	RESERVED	DATA PAGE	PDU FORMAT	PDU SPECIFIC	SOURCE ADDRESS
3 bits	1 bit	1 bit	8 bits	8 bits	8 bits

Obrázek 4.8: Struktura 29bitového identifikátoru (zdroj [21])

První (nejvyšší) tři bity identifikátoru slouží pro určení priority zprávy. Pokud je hodnota těchto tří bitů nulová, má zpráva nejvyšší prioritu. Typicky mají vyšší prioritu vysokorychlostní řídicí zprávy. Zprávám obsahující data, která nemusí být posílána tak často, je přidělena nižší priorita. [21]

Po prioritních bitech následuje bit, který je rezervován pro budoucí použití a měl by mít dominantní hodnotu. Po tomto bitu obsahuje identifikátor bit, který nese označení DP (z ang. Data Page, stránka dat). Tento bit rozšiřuje počet možných skupin parametrů (Parameter Groups), které mohou být reprezentovány identifikátorem. [21]

Hodnota dalších 8 bitů určuje, zda se jedná o formát zprávy PDU 1 nebo PDU 2. Těchto 8 bitů se označuje jako PDU format. Pokud je hodnota těchto bitů v rozmezí

0x00<sup>5</sup> (0) až 0xEF (239), jedná se o formát PDU 1. Pokud hodnota PDU format odpovídá zbylým hodnotám 0xF0 (240) až 0xFF (255), je formát zprávy PDU 2. Za sekvencí bitů PDU format následuje dalších 8 bitů s označením PDU specific. Význam těchto bitů se liší podle formátu zprávy. V případě, že se jedná o formát PDU 1, obsahují bity PDU specific cílovou adresu zařízení, pro které je zpráva určena. V případě, že formát zprávy je PDU 2, umožňuje hodnota těchto bitů rozšířit počet možných skupin parametrů. Toto rozšíření je označováno jako Group Extension. [21]

Posledních 8 bitů identifikátoru zprávy obsahuje zdrojovou adresu. Tím je myšlena adresa zařízení, které zprávu vyslalo. [21]

U protokolu SAE J1939 se používá výraz Parameter Group Number (PGN). PGN obsahuje celkem 18 bitů, jejichž hodnota odpovídá sloučené hodnotě rezervovaného bitu pro budoucí použití, DP, PDU format a PDU specific. [21] Příkladem by mohla být zpráva s identifikátorem 0xCF004EE (217056494). U takto zadaného identifikátoru je nutné si nejprve uvědomit, že obsahuje pouze 28 bitů. Je to z toho důvodu, že první hodnota bitu identifikátoru je nulová a proto se nepíše. Identifikátor je možné si rozšířit o čtveřici nulových bitů, které se přidají před první hodnotu identifikátoru. Identifikátor má pak tvar 0x0CF004EE (217056494). Takto zadaný identifikátor se dá rozdělit na bajty 0x0C (12), 0xF0 (240), 0x04 (4) a 0xEE (238).

Hodnota bajtu 0x0C odpovídá v binární soustavě hodnotě  $(00001100)_2$ <sup>6</sup>. První tři nulové bity byly přidány a nemají žádný význam. Následující hodnota  $(011)_2$  přímo odpovídá prioritě zprávy. Zpráva má tedy prioritu 3. Zbylé dva nulové bity jsou hodnoty rezervovaného bitu a bitu DP. Bajt 0xF0 (240) definuje, že se jedná o zprávu PDU 2. To znamená, že bajt PDU specific obsahuje Group Extension. Bajt PDU specific je roven hodnotě 0x04 (4). Z toho vyplývá, že Group Extension má hodnotu 4. Poslední bajt 0xEE (238) je přímo adresa zařízení, které zprávu vyslalo.

PGN identifikátoru 0xCF004EE je rovno hodnotě 0x0F004 (61444). První nula byla opět přidána. Je to z důvodu, že PGN obsahuje i rezervovaný bit a DP bit a oba tyto bity mají nulovou hodnotu. [21]

Protokol SAE J1939 obsahuje i další specifikace, které vymezují jeho použití. V rámci této práce nejsou tyto specifikace dále rozebírány.

---

<sup>5</sup>Označení 0x znamená, že následující hodnota je uvedena v hexadecimální (šestnáctkové) soustavě. Toto označení se využívá pro snadnější popis jednoho bajtu. Ten obsahuje celkem 8 bitů, což odpovídá dvou hexadecimálním hodnotám. V textu za touto hodnotou následuje její ekvivalentní hodnota v desítkové soustavě uvedena v závorkách.

<sup>6</sup>Index za závorkou značí, že se jedná o číslo v binární (dvojkové) soustavě.

## 5 CRC

CRC je zabezpečení dat na sběrnici CAN založené na CRC polynomu. CRC umožňuje zjistit, zda během přenosu zprávy nedošlo k zaměně některé z binárních hodnot ve zprávě a tudíž ke ztrátě informace. Vysílající zařízení před vysláním zprávy spočte CRC a vypočtenou hodnotu vloží do CRC pole zprávy. Přijímající zařízení z přijatých dat taktéž spočítá CRC se stejným polynomem a porovná výslednou hodnotu s hodnotou poslanou ve zprávě. Ve chvíli, kdy se spočtená i poslaná hodnota CRC rovná, je zpráva označena jako správně přijatá a přijímací zařízení nastaví ACK slot bit na dominantní hodnotu viz 4.1.6. V opačném případě vyšle přijímací zařízení po bitu ACK delimiter zprávu určenou pro oznámení chyby (Error Frame). [22]

### 5.1 Výpočet CRC

Hodnota CRC je hodnota zbytku po polynomiálním dělení dvou polynomů. Jeden polynom slouží jako dělitel a druhý jako dělenec. Na sběrnici CAN je dělitel složen ze všech bitů ve zprávě před CRC polem (bitem SOF počínaje a posledním bitem z datového pole konče). Během výpočtu CRC je na celkovou hodnotu těchto bitů (bez vložených bitů) nahlíženo jako na polynom, přičemž index nejvyššího bitu odpovídá stupni polynomu. Tento polynom je označován jako vstupní polynom. Vstupní polynom je dělen dělencem, který nese označení klíč CRC. Pro sběrnici CAN je tento klíč definován jako polynom 15 stupně viz polynom 5.1.

$$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1 \quad (5.1)$$

Polynom 5.1 je možné rozepsat do následujícího tvaru.

$$1 \cdot x^{15} + 1 \cdot x^{14} + 0 \cdot x^{13} + 0 \cdot x^{12} + 0 \cdot x^{11} + 1 \cdot x^{10} + 0 \cdot x^9 + 1 \cdot x^8 + \dots + 1 \cdot x^0 \quad (5.2)$$

Takto rozepsaný polynom lze jednoduše převést do binární soustavy. Koeficienty jednotlivých mocnin  $x$  odpovídají log. hodnotám. Index každé logické hodnoty je roven hodnotě exponentu příslušného výrazu  $x$ . Např. výraz  $1 \cdot x^{14}$  odpovídá log. hodnotě 1 s indexem 14. Binární ekvivalent výše zmíněného polynomu je zobrazen v tab. 5.1.

Tabulka 5.1: Binární podoba polynomu 5.1

Index	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Log. hod.	1	1	0	0	0	1	0	1	1	0	0	1	1	0	0	1

Pro jednodušší popis se hodnota klíče CRC většinou píše v hexadecimální podobě. Binární hodnota v tab. 5.1 pak odpovídá hodnotě 0xC599 (50585).

## 5.2 Polynomiální dělení

Během výpočtu podílu dvou polynomů je vždy nejvyšší mocnina dělece dělena nejvyšší mocninou dělitele. Výsledný výraz tohoto podílu je součást řešení a je následně roznásoben dělitelem. Tímto vynásobením se získá polynom, který se odečte od dělece. Polynom, který vznikne po odečtení, se stane novým dělencem a proces se opakuje až do doby, kdy nejvyšší mocnina dělece bude menší než nejvyšší mocnina dělitele. Polynom, který již nebude možné dále dělit dělitelem, je označován jako zbytek po polynomiálním dělení.

Postup polynomiálního dělení je ukázán v tab. 5.2. Hexadecimální ekvivalent hodnoty dělece v tab. 5.2 odpovídá hodnotě 0x350 (848) a hodnota dělitele je rovna 0xB (11). Polynomy, které vznikly roznásobením dělitele s částí řešení a budou odečteny od dělece, jsou označeny zeleně. Zbytek po polynomiálním dělení je vyznačen červeně.

Tabulka 5.2: Polynomiální dělení

	Dělece	Dělitel	Mezivýpočet
0	$x^9 + x^8 + x^6 + x^4$ $-(x^9 + x^7 + x^6)$	: $x^3 + x^1 + x^0 =$	$+x^6$
1	$0 + x^8 - x^7 + x^4$ $-(x^8 + x^6 + x^5)$		$+x^5$
2	$0 - x^7 - x^6 - x^5 + x^4$ $-(-x^7 - x^5 - x^4)$		$-x^4$
3	$0 - x^6 + 2 \cdot x^4$ $-(-x^6 - x^4 - x^3)$		$-x^3$
4	$0 + 3 \cdot x^4 + x^3$ $-(3 \cdot x^4 + 3 \cdot x^2 + 3 \cdot x^1)$		$+3 \cdot x^1$
5	$0 + x^3 - 3 \cdot x^2 - 3 \cdot x^1$ $-(x^3 + x^1 + x^0)$		$+x^0$
6	$0 - 3 \cdot x^2 - 4 \cdot x^1 - x^0$		

Celý výsledek se zbytkem po polynomiálním dělení je zobrazen v rovnici 5.3.

$$\frac{x^9 + x^8 + x^6 + x^4}{x^3 + x^1 + x^0} = x^6 + x^5 - x^4 - x^3 + 3 \cdot x^1 + x^0 + \frac{-3 \cdot x^2 - 4 \cdot x^1 - x^0}{x^3 + x^1 + x^0} \quad (5.3)$$

Jak bylo řečeno výše, hodnota CRC je rovna zbytku po dělení. CRC je ale posloupnost bitů. To znamená, že zbytek po dělení musí být binární hodnota. Tab. 5.3 ukazuje hodnotu výše vypočteného zbytku po dělení, pokud by se pro jeho získání využilo stejného postupu jako pro zjištění binární hodnoty polynomu 5.1.

Tabulka 5.3: Hodnota zbytku po dělení

Index	2	1	0
Hodnota	-3	-4	-1

Hned na první pohled je patrné, že hodnota uvedená v tab. 5.3 není binární a není ji tedy možné použít pro CRC. Z toho důvodu se ze zbytku po polynomiálním dělení nejprve vypočte modulo 2. To se provede tak, že se jednotlivé koeficienty podělí dvěma a zbytek po dělení pak zastoupí místo daného koeficientu. Vzhledem k tomu, že zbytek po dělení dvěma může být buď 1, nebo 0, je vždy výsledná hodnota binární.

Pro hodnotu -3 je modulo 2 rovno 1, pro -4 je rovno 0 a pro -1 je rovno 1. Po aplikaci modulo 2 na hodnotu v tab. 5.3 se získá binární hodnota  $(101)_2$ , kterou je již možné použít pro CRC.

Vypočtení zbytku po polynomiálním dělení je možné provést i v binární soustavě a získat tak přímo binární hodnotu CRC. V binární soustavě se pro tento účel využívá logická funkce XOR (Exclusive OR). Výstup funkce XOR pro dvě vstupní proměnné je roven log. hodnotě 0, pokud jsou hodnoty vstupních proměnných stejné. Pokud jsou hodnoty vstupních proměnných odlišné, je výstup roven log. hodnotě 1.

Tab. 5.4 ukazuje postup binárního dělení aplikovaný na stejné hodnoty použité v tab. 5.2. Dělenec má hodnotu  $0x350$  (848) a dělitel  $0xB$  (11).

Tabulka 5.4: Binární dělení

	$x^9$	$x^8$	$x^7$	$x^6$	$x^5$	$x^4$	$x^3$	$x^2$	$x^1$	$x^0$
0	1	1	0	1	0	1	0	0	0	0
	1	0	1	1						
1	0	1	1	0	0	1	0	0	0	0
		1	0	1	1					
2		0	1	1	1	1	0	0	0	0
			1	0	1	1				
3			0	1	0	0	0	0	0	0
				1	0	1	1			
4				0	0	1	1	0	0	0
						1	0	1	1	
5						0	1	1	1	0
							1	0	1	1
6							0	1	0	1

V obou případech se dojde ke stejnému zbytku po polynomiálním dělení v binární podobě  $(101)_2$ .



## 5.3 Algoritmus pro výpočet CRC

V dnešní době není většinou potřeba počítat CRC ručně. Výpočet se provádí automaticky přímo v technologii použité pro komunikaci na sběrnici CAN. Použitá technologie sama CRC počítá a kontroluje.

Pro výpočet CRC obsahuje příslušné zařízení v sobě výpočetní algoritmus. V praxi je snaha vymýšlet algoritmy, které jsou spolehlivé a zároveň i efektivní na výpočet. Proto tyto algoritmy nemusí zcela přesně odpovídat výše zmíněnému postupu. Níže vypsáný algoritmus je přepsanou verzí algoritmu uvedeném v [16] do programovacího jazyka C.

```
1 CRC_RG = 0;
2 do
3 {
4     CRCNXT = NXTBIT ^ (CRC_RG>>14 & 0x1);
5     CRC_RG = (CRC_RG <<1) & 0x7FFF;
6     if(CRCNXT)
7         CRC_RG = CRC_RG ^ 0x4599;
8 }
9 while(CRC SEQUENCE starts || ERROR);
```

Tento algoritmus využívá pro výpočet CRC 15bitový registr *CRC\_RG*. Na prvním řádku kódu je registr vynulován. Následující část kódu obsahuje smyčku, která se vykonává tak dlouho, dokud algoritmus neprojde všechna vstupní data nebo dokud se nevyskytne chyba.

Na čtvrtém řádku kódu se provádí logická operace XOR. Binární proměnná *NXTBIT* je hodnota jednotlivých bitů ve zprávě od bitu SOF až po poslední bit datového pole. Hodnota bitu *NXTBIT* se porovnává s hodnotou nejvyššího bitu registru *CRC\_RG*. Výsledek operace XOR je uložen do binární proměnné *CRCNXT*.

Řádek 5 pouze posouvá všechny bity registru *CRC\_RG* o jednu pozici doleva a přidá nulovou hodnotu na pozici 0.

Poslední část kódu obsahuje podmíněný příkaz *if*. Tato část se vykoná pokaždé, když je hodnota bitu *CRCNXT* rovna log. 1. V takovém případě se provede operace XOR mezi registrem *CRC\_RG* a hexadecimální hodnotou 0x4599 (17817). Hexadecimální hodnota se získá odebráním první log. 1 z klíče CRC 0xC599 (50585). Výsledek po operaci XOR se uloží do registru *CRC\_RG* a tím přepíše jeho hodnotu.

Ve chvíli, kdy dojde k ukončení smyčky, obsahuje registr *CRC\_RG* hodnotu CRC. Ta se ve zprávě zapíše do CRC sequence.

## 6 Lineární pohony

Lineární pohony jsou zařízení, jež vykonávají lineární (posuvný, přímočarý) pohyb. Transformaci rotačního pohybu na lineární je možné provést více způsoby. Podle způsobu provedení se tyto pohony dělí na mj. hydraulické, pneumatické, mechanické a elektromechanické. Elektromechanické lineární pohony využívají pro svoji činnost elektromotor generující rotační pohyb a mechanismy, které rotační pohyb převádí na pohyb lineární. Jedním z mechanismů, který dokáže docílit zmíněného převodu, je šroubový mechanismus. [23]

Elektromechanické lineární pohony jsou označovány též jako elektrické. V praxi je možné setkat se se všemi typy lineárních pohonů. Počínaje pneumatickými, které se používají např. v autobusech pro otvírání a zavírání dveří, konče hydraulickými, které jsou hojně využívány ve stavební technice.

Pro provoz hydraulického pohonu je potřeba komplexní systém, který zahrnuje nádrž na médium, čerpadlo, filtry a hadice. Takovýto systém vyžaduje pravidelnou údržbu, má vyšší nároky na spotřebu energie a existuje u něj riziko úniku média. Navíc pokud je požadováno snímání polohy pohonu, není možné použít integrovaný snímač polohy. Snímač musí být na pohon namontován, což přináší určité další komplikace. Oproti tomu elektrické lineární pohony jsou značně jednodušší. Nevyžadují žádnou údržbu, mají nižší spotřebu energie, je možné do nich integrovat snímač polohy a k provozu nevyužívají médium, které by mohlo potenciálně uniknout. Z těchto důvodů jsou v některých případech hydraulické pohony často nahrazovány elektrickými. [24]

Nejedna společnost se zabývá elektrickými lineárními pohony. Mezi takové společnosti se řadí i společnost LINAK C&S s.r.o. (dále jen LINAK). Ta se výrobě elektrických lineárních pohonů věnuje již od roku 1979 a drží se v čele odvětví, jež se zabývá lineárním pohybem. Jedním z produktů společnosti LINAK je pohon LA36. [25] Pohon LA36 byl vybrán pro projektu AnteTUL, aby zajistil směrové řízení kol. Pokud se tento pohon ukáže jako vhodný, stane se náhradou za stávající systém řízení EPAS.

Tato kapitola obsahuje popis pohonu LA36, jeho elektrické zapojení a způsob komunikace s řídicí jednotkou pohonu.

### 6.1 Pohon LA36

Pohon LA36 je jedním z nejvíce výkonných a odolných pohonů společnosti LINAK. Jedná se o pohon, který je navržený pro provoz v extrémních podmínkách s bezúdrž-

bovým provozem, dlouho životností a vysokým stupněm krytí IP66 (z ang. Ingress Protection). Díky všem těmto vlastnostem se stává silnou konkurencí hydraulickým systémům. [26] Zobrazení pohonu s popisky je možné vidět na obr. 6.1.

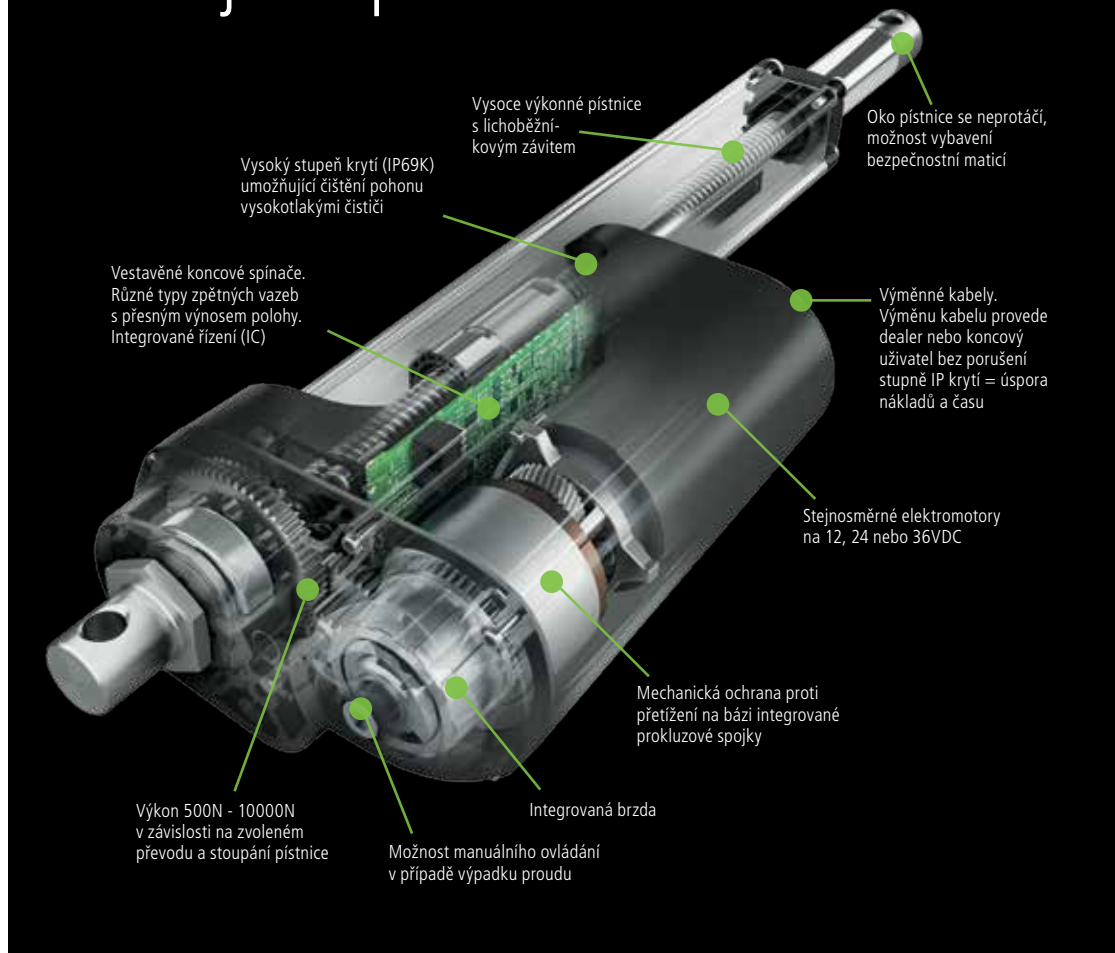
Hlavní roli pohonné jednotky u pohonu LA36 zastupuje stejnosměrný elektromotor s permanentními magnety. Ten je možné pořídit s napájecím napětím na 12, 24, 36 nebo 48 V<sup>7</sup>. V závislosti na převodovém poměru a stoupání pístnice dokáže tento pohon vyvinout sílu v tahu v rozmezí 500 - 6800 N. Podle zatížení pohonu a stoupání pístnice může společnost LINAK poskytnout pohon, který disponuje rychlostí až 160 mm/s. Přímo v pohonu je také integrovaná brzda, která zaručuje vysokou samosvornost. Aby se docílilo vysoké efektivity, je tato brzda v momentě spuštění pohonu deaktivována. Pohon je možné získat v délkách od 100 do 999 mm. Pracovní cyklus pohonu je závislý na aktuálním vysunutí pístnice. Pokud je vysunutí pístnice menší než 600 mm, je pracovní cyklus roven 20 % a pokud je vysunutí v rozsahu 601 - 999 mm, je možné pohon zatěžovat v cyklu 15 %. [26]

O řízení pohonu se stará integrovaná řídicí jednotka nebo externí řídicí systém. V případě použití integrované řídicí jednotky společnost LINAK nabízí několik variant pro komunikaci s jednotkou. Jedná se např. o sběrnici Modbus, LIN bus nebo CAN bus. O samotné řízení pohonu se stará integrovaná řídicí jednotka, která od nadřazeného systému získává pokyny prostřednictvím jedné z těchto sběrnic. V případě použití externího řídicího systému (bez integrované řídicí jednotky) je nutné připojit k pohonu řídicí systém, který zastoupí roli řídicí jednotky. Přesná regulace polohy pohonu je možná díky zpětné vazbě, kterou pohon disponuje. Ta může být digitální nebo analogová v závislosti na požadavcích zákazníka. [26]

---

<sup>7</sup>Všechny komponenty elektromobilu popisované v této práci jsou napájeny stejnosměrným napětím. Z toho důvodu hodnoty napětí (volty) nejsou doplněny o zkratku DC.

# Co je to pohon - LA36?



Obrázek 6.1: Zobrazení pohonu LA36 s popisky (zdroj [24])

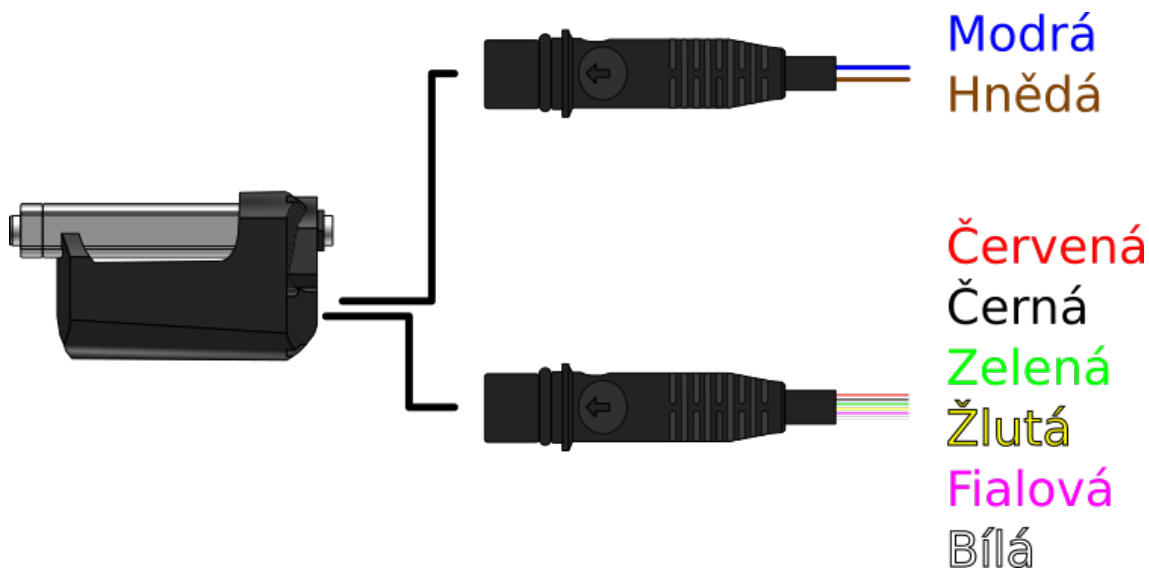
## 6.1.1 Zapojení a komunikace

Pro potřeby projektu AnteTUL byl zvolen pohon LA36 s možností vysunutí pístnice na  $150 \pm 6$  mm. Aby bylo možné pohon ihned testovat a nemusela se řešit polohová regulace, obsahuje vybraný pohon již integrovanou řídicí jednotku. Pro způsob komunikace byla určena sběrnice CAN s protokolem SAE J1939. Na sběrnici CAN komunikuje většina zařízení použitá v projektu AnteTUL. Nadřazený řídicí systém je tedy již připraven na tento typ komunikace, což umožňuje snadnější začlenění pohonu LA36.

Vybraný pohon také disponuje dvěma koncovými spínači pro určení nejvyššího zasunutí a vysunutí pístnice. Tyto spínače nejen, že zamezují pohybu pístnice mimo výrobní rozsah, ale slouží i pro kalibraci snímače polohy pístnice, jenž slouží jako zpětná vazba. Při každém dosažení jednoho z koncových spínačů dojde k přepočítání polohy pístnice. [26]

Zapojení pohonu LA36 spočívá ve správném připojení dvou kabelů. Jeden kabel

slouží jako napájecí a druhý je určený pro komunikaci a manuální řízení pohonu. Napájecí kabel obsahuje modrý a hnědý vodič. Druhý kabel obsahuje vodiče barvy červené, černé, zelené, žluté, fialové a bílé viz obr. 6.2. Účel jednotlivých vodičů je popsán v tab. 6.1. [27]



Obrázek 6.2: Kabely pohonu LA36 (zdroj autor)

Tabulka 6.1: Zapojení pohonu LA36 (podle [27])

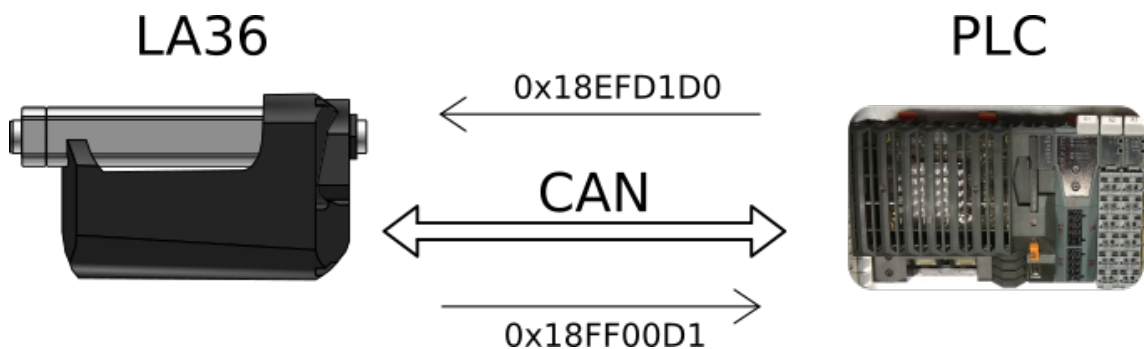
Barva vodiče	Popis
Modrá	Napájení (GND)
Hnědá	Napájení (24 V)
Červená	Příkaz k vysunutí pístnice
Černá	Příkaz k zasunutí pístnice
Zelená	CAN Low
Žlutá	CAN High
Fialová	Servisní rozhraní
Bílá	Servisní rozhraní (GND)

Komunikace s řídicí jednotkou pohonu je zajištěna protokolem SAE J1939. Celá komunikace je postavena na cyklickém posílání dvou zpráv mezi řídicí jednotkou pohonu a nadřazeným systémem. Nadřazený systém posílá řídicí jednotce zprávu s PGN 0x00EF00 (61184) s nejvyšší periodou 250 ms a naopak řídicí jednotka posílá nadřazenému systému zprávu s PGN 0x00FF00 (65280) s nejvyšší periodou 100 ms. [27]

Zpráva od nadřazeného systému je ve formátu PDU 1. Jak bylo již řečeno v kapitole 4, poslední bajt PGN zprávy ve formátu PDU 1 obsahuje adresu zařízení, kterému je zpráva určena. Tímto zařízením je pohon LA36. Tomu byla přidělena adresa 0xD1 (209). PGN zprávy určené pro tento pohon má tedy tvar 0x00EFD1 (61393).

V opačném případě, kdy zprávu posílá zařízení nadřazenému systému, je zpráva ve formátu PDU 2. Poslední bajt PGN zprávy ve formátu PDU 2 obsahuje rozšíření skupin parametrů. To je v případě pohonu LA36 rovno nulové hodnotě. Hodnota PGN zprávy od řídicí jednotky pohonu LA36 je rovna 0x00FF00 (65280).

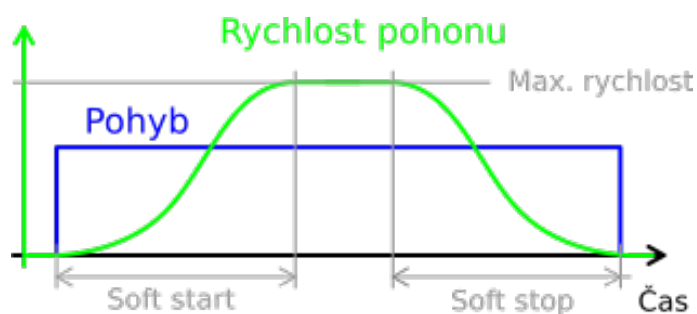
Aby byl identifikátor obou zpráv zcela kompletní, je potřeba k PGN ještě přidat prioritu zprávy a zdrojovou adresu. Oběma zprávám byla přidělena stejná priorita  $(110)_2$ . Nadřazený systém má adresu 0xD0 (208). Celý identifikátor zprávy od nadřazeného systému pro řídicí jednotku pohonu LA36 má pak tvar 0x18EFD1D0 (418370000). A identifikátor zprávy posílané řídicí jednotkou pohonu pro nadřazený systém je roven hodnotě 0x18FF00D1 (419365073).



Obrázek 6.3: Komunikace mezi pohonem LA36 a PLC (zdroj autor)

### Popis zprávy pro řídicí jednotku pohonu LA36

Ve zprávě pro řídicí jednotku pohonu je obsaženo celkem 8 bajtů dat. Tato data umožňují potvrdit chybu pohonu, zadat příkaz pro vysunutí nebo zasunutí pístnice na koncové spínače nebo dosažení požadované polohy vysunutí pístnice. Dále data umožňují nastavit rychlost pohybu pístnice, nejvyšší povolený odběr proudu a parametry Soft Start a Soft Stop. [27] Tyto parametry definují požadovaný čas pro rozběh a doběh pístnice v ms viz obr. 6.4.



Obrázek 6.4: Rychlost pohonu LA36 během pohybu pístnice (zdroj autor)

Pro jednodušší představu je struktura zprávy pro řídicí jednotku pohonu LA36 zobrazena v tab. 6.2.

Tabulka 6.2: Struktura zprávy pro řídicí jednotku (podle [27])

Číslo bajtu	Přenášená data
7	0xFF (255)
6	0xFF (255)
5	Soft Stop [ms*50]
4	Soft Start [ms*50]
3	Rychlost [%*0,5]
2	Proud [mA*250]
1	Poloha MSB [mm*0,1]
0	Poloha LSB [mm*0,1]

Bajty 7 a 6 obsahují konstantní hodnoty 0xFF (255). Tyto bajty jsou rezervovány pro možné budoucí použití. Pro komunikaci s řídicí jednotkou pohonu LA36 nejsou využity. [27]

Bajt 5 (popř. 4) umožňuje nastavit dobu pro doběhnutí (popř. rozběhnutí) pístnice. Hodnota obsažená v tomto bajtu je nejprve 50 krát vynásobena. Výsledná hodnota v desítkově soustavě udává čas v ms. Pokud je hodnota času v ms větší než nula, je zpomalení (popř. zrychlení) pístnice plynulé. Zadáním hodnoty 0x00 (0) dojde k nastavení tzv. Hard Stop (popř. Hard Start). Toto nastavení způsobí skokovou změnu zpomalení (popř. zrychlení) při doběhu (popř. rozběhu) pístnice, což způsobuje rázy. Hodnoty 0x01–0xFA (1–250) odpovídají časům od 50 ms do 12500 ms. Časy mezi 50 a 300 ms ale nejsou povoleny. Je to z důvodu minimalizování efektu zpětné EMF (z ang. Electromotive Force). Hodnoty 0xFB a 0xFF (251 a 255) odpovídají původní nastavené hodnotě z výroby (dále jen výchozí hodnota)<sup>8</sup>. Ta je rovna 300 ms. Zbylé hodnoty 0xFC–0xFE (252–254) nejsou použity. [27]

Nejvyšší rychlost pístnice je nastavována bajtem 3. Hodnota tohoto bajtu je nejprve vydělena dvěma. Vypočtená hodnota převedená do desítkové soustavy udává rychlost v procentech z nejvyšší rychlosti, kterou může pohon dosáhnout (32 mm/s). Hodnoty 0x00–0xC7 (0–199) udávají rozsah od 0 % do 99,5 %. Hodnotami v rozsahu 0xC8–0xFA (200 - 250) se zadá 100 %, což znamená nejvyšší možnou rychlost pohonu. Pro zadání výchozí hodnoty je potřeba, aby bajt 3 obsahoval hodnotu 0xFB (251). Hodnoty 0xFC–0xFF (252–255) nejsou použity. [27]

Bajtem číslo 2 se nastavuje nejvyšší dovolený proud, který může pohon při provozu odebírat. Opět je hodnota tohoto bajtu nejprve vynásobena. Násobek je roven 250 a dekadický výsledek je roven nejvyššímu dovolenému odběru proudu v mA. Pokud tento bajt obsahuje hodnoty od 0x00 (0) do 0xFA (250), je nejvyšší dovolený odběr proudu dán těmito hodnotami v rozmezí 0 A až 62,5 A. Hodnota 0xFB (251) odpovídá jako v předchozím případě výchozí hodnotě. Stejně tak i hodnoty 0xFC–0xFF (252–255) nejsou použity. [27]

Poslední dva bajty 1 a 0 slouží pro nastavení požadované polohy, na kterou má řídicí jednotka vysunout nebo zasunout pístnici. Celková požadovaná hodnota polohy je dána dvěma bajty.<sup>9</sup> Tu je možné určit tak, že se hodnota bajtu 1 (MSB z ang.

<sup>8</sup>V anglických textech se hodnota nastavená z výroby označuje jako výchozí (z ang. default).

<sup>9</sup>Proměnná, která obsahuje dva bajty dat, je označována jako typ word.

Most Significant Bit) přidá před hodnotu bajtu 0 (LSB z ang. Least Significant Bit). V předchozích případech byl rozsah parametrů omezen jen na hodnoty v rozmezí 0x00–0xFF (0–255). Použitím dvou bajtů dat je možné zadávat požadovanou hodnotu polohy v rozsahu 0x0000–0xFFFF (0–65535). Díky tak velkému rozsahu hodnot se požadovaná poloha může zadávat v desetínách mm. Hodnotám od 0x0000 (0) do 0xFAFF (64255) pak odpovídají hodnoty vysunutí pístnice od 0 mm do 6425,5 mm. Vzhledem k tomu, že vybraný pohon LA36 má omezené vysunutí pístnice na 150 mm, odpovídají hodnoty větší než 150 mm nejvyššímu možnému vysunutí pístnice. Pokud je zadána hodnota 0x0000 (0), dojde k úplnému zasunutí pístnice. Hodnoty od 0xFB04 (64260) po nejvyšší možnou hodnotu 0xFFFF (65535) nejsou použity. [27]

Výše bylo řečeno, že zpráva pro řídicí jednotku pohonu umožňuje posílat i některé příkazy. K tomu slouží zatím nezmíněné hodnoty bajtů 1 a 0. Pro potvrzení chyby pohonu musí být hodnota požadované polohy rovna 0xFB00 (64256). Bajt 1 tedy obsahuje hodnotu 0xFB (251) a bajt 0 obsahuje 0x00 (0). Po zadání hodnoty 0xFB01 (64257) dojde k vysouvání pohonu dokud nedojde k aktivaci koncového spínače nebo dokud není poslán příkaz pro zastavení pohybu pístnice. Pro opačný pohyb tedy zasunutí pístnice až na koncový spínač je nutné zadat hodnotu 0xFB02 (64258). Poslední hodnota 0xFB03 (64259) zadá příkaz pro okamžité zastavení pohybu pístnice. [27]

Následující tabulky ukazují příklady zpráv, které by mohl poslat nadřazený systém řídicí jednotce pohonu. Zpráva v tab. 6.3 obsahuje data pro nastavení požadovaných parametrů pohonu a příkaz pro vysunutí pístnice na 200 mm. A posláním zprávy v tab. 6.4 dojde k zasunutí pístnice na koncový spínač při parametrech, které byly nastaveny z výroby (výchozí hodnoty).

Tabulka 6.3: Zpráva pro dosažení požadované polohy vysunutí pístnice

Číslo bajtu	Přenášená data	Výsledný požadavek
7	0xFF (255)	-
6	0xFF (255)	-
5	0x1E (30)	Soft Stop 1500 ms
4	0x0A (10)	Soft Start 500 ms
3	0x64 (100)	Rychlost 50 % (15 mm/s)
2	0x08 (8)	Proud 2000 mA
1	0x07 (7)	Požadovaná poloha 200,0 mm 0x07D0 (2000)
0	0xD0 (208)	



Tabulka 6.4: Zpráva pro zasunutí pístnice

Číslo bajtu	Přenášená data	Výsledný požadavek
7	0xFF (255)	-
6	0xFF (255)	-
5	0xFF (255)	Soft Stop 300 ms
4	0xFF (255)	Soft Start 300 ms
3	0xFB (251)	Rychlost 100 % (30 mm/s)
2	0xFB (251)	Proud ... mA
1	0xFB (251)	Příkaz pro zasunutí pístnice 0xFB02 (64258)
0	0x02 (2)	

### Popis zprávy od řídicí jednotky pohonu LA36

Výše byl popsán způsob, jakým je možné posílat řídicí jednotce pohonu příkazy. Aby bylo řízení spolehlivé, posílá řídicí jednotka nadřazenému systému zpětnou vazbu ve formě další zprávy. Ta obsahuje údaje o aktuální rychlosti pohybu pístnice, proudovém odběru, poloze a stavu pohonu a v případě, že se pohon dostane do chybového stavu, tak i číslo chyby. Uspořádání těchto proměnných ve zprávě zobrazuje tab. 6.5. [27]

Tabulka 6.5: Struktura zprávy od řídicí jednotky (podle [27])

Číslo bajtu (bitů)	Přenášená data
7	Hodnoty vstupů
6 (4–7)	0xF (15)
6 (0–3)	Rychlost MSB [mm/s*0,1]
5	Rychlost LSB [mm/s*0,1]
4	Číslo chyby
3	Binární proměnné <sup>10</sup>
2	Proud [mA*250]
1	Poloha MSB [mm*0,1]
0	Poloha LSB [mm*0,1]

Pokud dojde k aktivaci příkazu pro vysunutí nebo zasunutí pístnice manuálním způsobem (přivedením napětí na příslušné signály), řídicí jednotka informuje nadřazený systém o této události příslušnými hodnotami v bajtu 7. [27]

Bajt 6 je rozdělen na dvě čtveřice bitů. Vyšší čtyři bity (4–7) obsahují neměnnou hodnotu 0xF (15). Nižší čtyři bity (0–3) spolu s bajtem číslo 5 obsahují údaj o aktuální rychlosti pohybu pístnice. Hodnota nižší čtveřice bitů bajtu 6 je přidána před 8bit hodnotu bajtu 5. Hodnota aktuální rychlosti je tedy udávána pomocí 12 bitů, jejichž dekadická podoba odpovídá rychlosti v desetínách mm/s. Možné hodnoty těchto bitů jsou v rozsahu 0x000–0xFAF (0–4015), které odpovídají hodnotám rychlosti v rozmezí 0–401,5 mm/s. Zbylé hodnoty 0xFB0–0xFFF (4016–4095) nejsou použity. [27]

Hodnota bajtu 4 umožňuje zjistit, zda se pohon nachází v chybovém stavu popř. jaká chyba v pohonu nastala. Ačkoliv pomocí jednoho bajtu je možné nastavit až 256 hodnot, pro účel tohoto bajtu bylo použito pouze 10 hodnot, jejichž desítková hodnota odpovídá číslu dané chyby viz tab. 6.6. [27]

Tabulka 6.6: Seznam chyb pohonu LA36 (podle [27])

Hodnota bajtu 4	Odpovídající chyba
0x00 (0)	Žádná chyba
0x01 (1)	Chyba hallovy sondy
0x02 (2)	Přepětí
0x03 (3)	Podpětí
0x04 (4)	Chyba komunikace na sběrnici CAN
0x05 (5)	Chyba koncového spínače
0x06 (6)	Power on block state
0x07 (7)	Teplotní chyba
0x08 (8)	Chyba Heart beat (vnitřní)
0x09 (9)	SMPS chyba (vnitřní)

Během provozu pohonu může nastat situace, kdy se objeví najednou více chyb. V takovém případě řídicí jednotka pohonu pošle po sběrnici pouze číslo chyby s nejvyšší prioritou. [27]

Bajt 3 obsahuje informace o stavu binárních proměnných. Aktuální hodnoty jednotlivých proměnných jsou definovány hodnotami jednotlivých bitů bajtu 3. Díky tomu jsou hodnoty binárních proměnných na sobě zcela nezávislé. Pokud je hodnota bitu rovna 1, je daná binární proměnná aktivní. Tab. 6.7 zobrazuje jednotlivé binární proměnné a jím odpovídající bit bajtu 3. [27]

Tabulka 6.7: Seznam binárních proměnných pohonu LA36 (podle [27])

Číslo bitu bajtu 3	Odpovídající binární proměnná
0	Koncový spínač úplného zasunutí pístnice
1	Koncový spínač úplného vysunutí pístnice
2	Nadproud
3	Vysouvání pístnice
4	Zasouvání pístnice
5–7	Není použito

Hodnotu aktuálně odebíraného proudu pohonem je možné zjistit pomocí bajtu 2. Obdobně jako ve zprávě pro řídicí jednotku je potřeba dekadickou hodnotu tohoto bajtu nejprve vynásobit 250. Výsledná hodnota pak odpovídá proudu v mA. Pokud se pohon nepohybuje, je hodnota bajtu 2 rovna 0x00 (0). Obsahuje-li tento bajt hodnoty od 0x01 (1) do 0xFA (250), je pohon v pohybu a odběr proudu je zobrazován v rozsahu 0,25–65,2 A. Hodnota 0xFE (254) indikuje, že v okruhu pro měření proudu nastala chyba. A zbylé hodnoty 0xFB–0xFD (251–253) a 0xFF (255)

nejsou použity. [27]

Poslední dva bajty 1 a 0 ve zprávě od řídicí jednotky udávají aktuální polohu pístnice. Celková hodnota polohy je opět dána bajty 1 a 0 a její dekadický ekvivalent odpovídá poloze pístnice v desetinách mm. Polohu je možné číst v rozsahu hodnot 0x0000 (0) až 0xFAFF (64255), které odpovídají hodnotám od 0 mm do 6425,5 mm. Pokud dojde k situaci, kdy řídicí jednotka pohonu ztratí údaj o poloze pístnice, nastaví hodnotu 0xFE00 (65024). Zbylé hodnoty opět nejsou použity. [27]

## 7 Ovladač IFM

Obsahem této kapitoly je popis ovladače IFM, jeho využití pro projekt AnteTUL, zapojení a způsob komunikace s nadřazeným systémem.

Označení Ovladač IFM je zjednodušený název ovladače, který poskytla společnost ifm electronic, spol. s r.o. (dále jen ifm). Skutečné označení ovladače je CR1301 a jedná se o tlačítkový modul s knoflíkem<sup>11</sup> s funkcí push a joystick ve dvou osách s podsvícenými klávesami. [28]



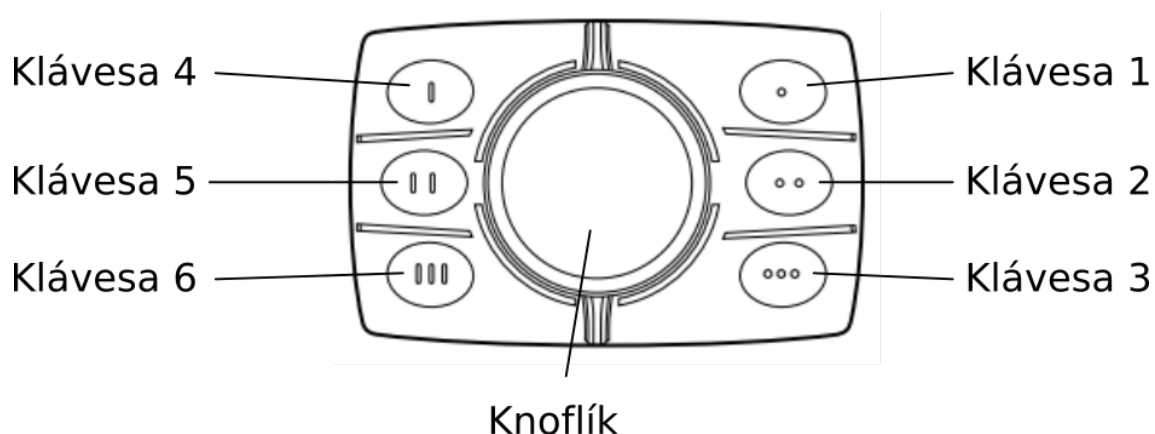
Obrázek 7.1: Ovladač IFM (CR1301) (zdroj [28])

Ovladač IFM disponuje šesti funkčními klávesami a knoflíkem s funkcí push (stlačení). Knoflík má v sobě integrované tlačítko, které dokáže zaznamenat polohy UP (nahore), DOWN (dole), RIGHT (vpravo) a LEFT (vlevo). Knoflíkem je možné dosáhnout i diagonální polohy, kdy se aktivují dvě polohy naráz např. UP a RIGHT (nahore vpravo). Celkem je tedy možné knoflíkem dosáhnout devíti poloh včetně klidové polohy, do které se knoflík sám vrací. V klidové poloze jsou všechny polohy neaktivní. Všechny tyto funkce slouží jako vstupy (příkazy) od uživatele, na které

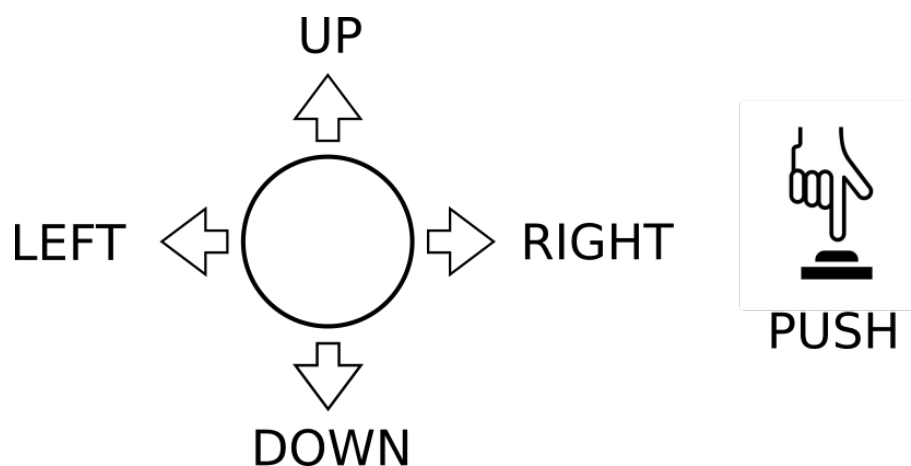
---

<sup>11</sup>Výraz knoflík využívá společnost ifm jako překlad ang. slova knob. Z toho důvodu je i v této práci používán.

může nadřazený systém reagovat.

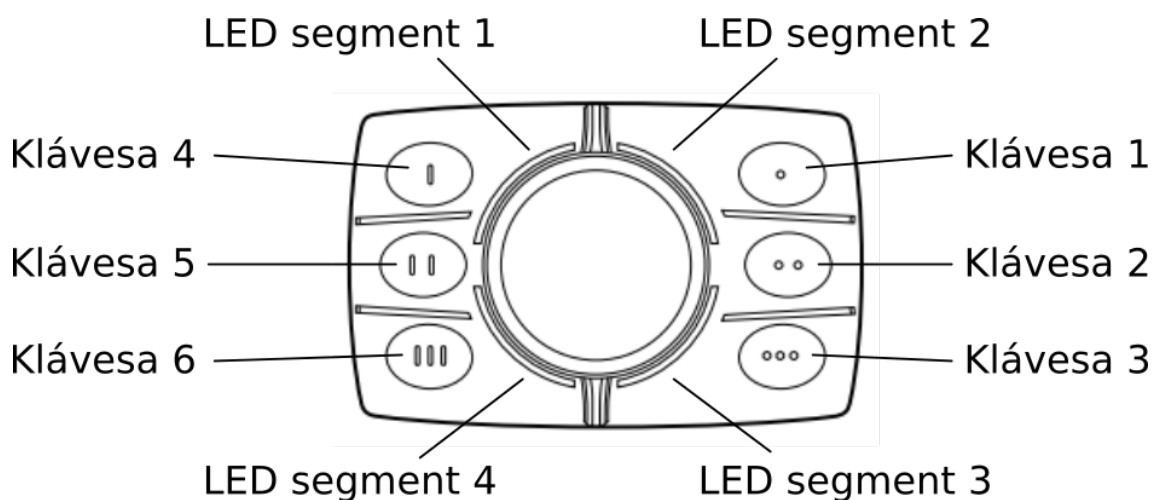


Obrázek 7.2: Označení prvků ovladače IFM (zdroj autor)



Obrázek 7.3: Polohy Knoflíku (zdroj autor)

Tento ovladač disponuje i výstupy (indikátory), které může využít nadřazený systém pro informování uživatele a usnadnit mu tak ovládání daného zařízení. Mezi tyto výstupy patří šest RGB LED diod a jeden segmentovaný LED kruh kolem Knoflíku, který je rozdělený celkem na 4 segmenty. Každá z RGB LED diod slouží k podsvícení jedné z funkčních kláves. Ovladač obsahuje tedy 10 indikátorů viz obr. 7.4. Každý z těchto indikátorů je možné nastavovat samostatně v požadované barvě.



Obrázek 7.4: Označení indikátorů ovladače IFM (zdroj autor)

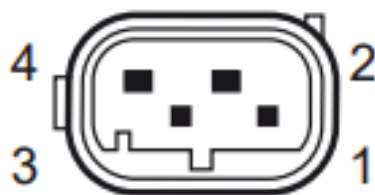
Barvu indikátoru je možné definovat pomocí 3 bajtů. Hodnoty těchto bajtů představují míru zastoupení jedné ze tří barevných složek. Jedná se o složku červenou (R), zelenou (G) a modrou (B), proto RGB LED dioda. Jak bylo již řečeno, jeden bajt může nabývat dekadické hodnoty v rozsahu 0 až 255. Hodnota 255 odpovídá plnému rozsvícení (zastoupení) dané barevné složky a hodnota 0 zhasnutí. Vzhledem k tomu, že každý indikátor obsahuje tři bajty hodnot, existuje až 16 milionů barevných kombinací, které je možné nastavit.

Komunikace s ovladačem je zprostředkována protokolem SAE J1939 na sběrnici CAN [28] s rozšířeným identifikátorem na 29 bitů. Díky všem výše zmíněným vlastnostem je možná jednoduchá integrace ovladače IFM (i ve stávajících aplikacích) a existuje velká míra možností, jak jej využít pro rozhraní mezi uživatelem a řídicím systémem. Z toho důvodu byl i ovladač IFM zvolen jako jeden ze způsobů ovládání vyvíjeného elektromobilu v rámci projektu AnteTUL.

Celkem se plánuje použití dvou ovladačů IFM, kde jeden bude umístěn na ovládacím panelu pro přímé ovládání elektromobilu a druhý bude součástí zařízení pro dálkové ovládání.

## 7.1 Zapojení a komunikace

Zapojení ovladače IFM je poměrně intuitivní. Pro připojení ovladače slouží 4pinový konektor. Ten je vyobrazen na obr. 7.5 a funkci jednotlivých pinů popisuje tab. 7.1. [29]



Obrázek 7.5: Konektor ovladače IFM (zdroj [29])

Tabulka 7.1: Zapojení ovladače IFM (podle [29])

Hlavní účel	Číslo pinu	Popis
Rozhraní CAN	1	CAN Low
	2	CAN High
Napájení	3	GND
	4	9–32 V

Dva piny tedy slouží pro komunikaci s řídicí jednotkou ovladače a dva pro napájení. Napájení ovladače je možné v rozsahu 9–32 V. Proudový odběr ovladače by při napájení 13,5 V neměl přesáhnout 400 mA. [28]

Stejně jako u pohonu LA36 je i komunikace s řídicí jednotkou ovladače IFM postavena na cyklickém posílání zpráv mezi ovladačem a nadřazeným systémem. Ovladač posílá nadřazenému systému zprávu s identifikátorem 0x10FF6487 (285172871). Kvůli vysokému počtu možných barevných kombinací, omezení počtu bajtů jedné zprávy a celkovému počtu indikátorů není možné, aby nastavení všech indikátorů bylo zahrnuto pouze v jedné zprávě. Z toho důvodu nadřazený systém musí posílat řídicí jednotce ovladače celkem 5 zpráv s identifikátory 0x18FF7587 (419394951), 0x18FF7687 (419395207), 0x18FF7787 (419395463), 0x18FF7887 (419395719) a 0x18FF7987 (419395975). [29]

Použitá přenosová rychlost komunikace je 250 kbit/s a všechny zprávy by se měly posílat v cyklech 30 ms. [29]

### 7.1.1 Popis zpráv pro řídicí jednotku ovladače IFM

Zprávy určené pro řídicí jednotku slouží pouze pro nastavování barev jednotlivých indikátorů. Struktura všech těchto zpráv je stejná. Každá zpráva umožňuje nastavení barevných kombinací v plném rozsahu pro dva indikátory. Indikátory příslušející jednotlivým zprávám jsou popsány v tab. 7.2.

Tabulka 7.2: Struktura zpráv pro ovladač IFM (podle [29])

ID zprávy	Číslo bajtu							
	0	1	2	3	4	5	6	7
0x18FF7587 (419394951)	Klávesa 1			Klávesa 2				
	R	G	B	R	G	B	0xFF (255)	0xFF (255)
0x18FF7687 (419395207)	Klávesa 3			Klávesa 4				
	R	G	B	R	G	B	0xFF (255)	0xFF (255)
0x18FF7787 (419395463)	Klávesa 5			Klávesa 6				
	R	G	B	R	G	B	0xFF (255)	0xFF (255)
0x18FF7887 (419395719)	LED segment 1			LED segment 2				
	R	G	B	R	G	B	0xFF (255)	0xFF (255)
0x18FF7987 (419395975)	LED segment 3			LED segment 4				
	R	G	B	R	G	B	0xFF (255)	0xFF (255)

Kvůli omezenému počtu osmi bajtů na jednu zprávu, obsahuje každá zpráva dva nepoužité bajty (bajt 6 a 7). Hodnota těchto bajtů je rovna 0xFF (255).

Zbylé bajty umožňují nastavovat jednotlivým indikátorům hodnotu dané barevné složky. Některé barevné kombinace jsou vyobrazeny v tab. 7.3.

Tabulka 7.3: Vybrané kombinace v barevném prostoru RGB

Hodnota barevné složky			Odpovídající barva
R	G	B	
0xFF (255)	0x00 (0)	0x00 (0)	Červená
0x00 (0)	0xFF (255)	0x00 (0)	Zelená
0x00 (0)	0x00 (0)	0xFF (255)	Modrá
0xFF (255)	0xFF (255)	0xFF (255)	
0x00 (0)	0x00 (0)	0x00 (0)	Černá
0xFF (255)	0xA5 (165)	0x00 (0)	Oranžová
0x8B (139)	0x00 (0)	0xFF (255)	Fialová

Obr. 7.6 pak znázorňuje příklad možného nastavení zpráv. Hodnoty poslaných zpráv jsou vypsány v tab. 7.4<sup>12</sup>. Pro tento příklad jsou Klávesy 1 a 4 nastaveny na červenou barvu, Klávesy 2 a 5 na zelenou a Klávesy 3 a 6 na modrou. Všechny segmenty LED kruhu obsahují hodnoty odpovídající bílé barvě.

<sup>12</sup>Aby tabulka byla čitelná a nemusela být umístěna na šířku stránky, jsou v tomto případě hexadecimální hodnoty jednotlivých bajtů vypsány bez jejich dekadických ekvivalentů v závorkách.





Obrázek 7.6: Příklad nastavení indikátorů ovladače IFM (zdroj autor)

Tabulka 7.4: Příklad zpráv pro ovladač IFM

ID zprávy	Číslo bajtu							
	0	1	2	3	4	5	6	7
0x18FF7587 (419394951)	Klávesa 1			Klávesa 2				
	0xFF	0x00	0x00	0x00	0xFF	0x00	0xFF	0xFF
0x18FF7687 (419395207)	Klávesa 3			Klávesa 4				
	0x00	0x00	0xFF	0xFF	0x00	0x00	0xFF	0xFF
0x18FF7787 (419395463)	Klávesa 5			Klávesa 6				
	0x00	0xFF	0x00	0x00	0x00	0xFF	0xFF	0xFF
0x18FF7887 (419395719)	LED segment 1			LED segment 2				
	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x18FF7987 (419395975)	LED segment 3			LED segment 4				
	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

## 7.1.2 Popis zprávy od řídicí jednotky ovladače IFM

Stisknutí jednotlivých kláves (i pohyb Knoflíku) je signalizováno binárními hodnotami (aktivní a neaktivní stav). Každý bajt obsahuje celkem 8 bitů. To znamená, že jedna zpráva obsahuje 64 bitů, které je možné použít pro signalizování binárních stavů. To zdaleka převyšuje potřebný počet pro definování stavů všech kláves a poloh Knoflíku nezávisle na sobě.

Kromě pohybu Knoflíku a stlačení je možné s ním i rotovat. Ovladač IFM obsahuje inkrementální enkodér s 24 inkrementy. Díky tomuto enkodéru je možné zjistit směr rotace Knoflíku a počet pootočení Knoflíku o 15°. Směr rotace je signalizován opět binární proměnnou. Pokud tato proměnná nabývá hodnoty 0, je Knoflíkem rotováno ve směru hodinových ručiček. Pokud proměnná nabude hodnoty 1, je směr

rotace proti směru hodinových ručiček. Pro zjištění počtu pootočení Knoflíku v rámci jedné otáčky byl použit jeden bajt. Ten může nabývat hodnot od 0 do 24, přičemž 0 odpovídá žádnému pootočení a hodnota 24 signalizuje pootočení o 360° tedy o jednu celou otáčku. [29]

Hodnoty bitů a bajtů, jenž odpovídají stavům a hodnotám příslušných proměnných, jsou zobrazeny v tab. 7.5. Pokud je daná klávesa stisknuta, je hodnota příslušného bitu rovna 1. V opačném případě je hodnota bitu 0. Bity a bajty, které nejsou vypsané v tab. 7.5, nejsou použity a nabývají nulových hodnot.

Tabulka 7.5: Struktura zprávy od řídicí jednotky (podle [29])

Číslo bajtu	Číslo bitu	Odpovídající proměnná
1	0	Stav Klávesy 1
	2	Stav Klávesy 2
	4	Stav Klávesy 3
	6	Stav Klávesy 4
2	0	Stav Klávesy 5
	2	Stav Klávesy 6
3	0–4	Počet pootočení Knoflíku
	5	Směr rotace Knoflíku
	6	Stisknutí Knoflíku (push)
4	0	Poloha Knoflíku LEFT (vlevo)
	2	Poloha Knoflíku RIGHT (vpravo)
	4	Poloha Knoflíku UP (nahore)
	6	Poloha Knoflíku DOWN (dole)

Knoflíkem je možné pohybovat, i když je stisknutý. Hodnoty uvedené v tab. 7.6 znamenají, že došlo k současnému stisknutí Klávesy 3 a 5 a Knoflík je stisknutý v poloze UP a RIGHT (nahore a vpravo).

Tabulka 7.6: Příklad zprávy od řídicí jednotky

Číslo bajtu	Hodnota bajtu	Hodnota bitu								Odpovídající stav
		7	6	5	4	3	2	1	0	
0	0x00 (0)	0	0	0	0	0	0	0	0	-
1	0x10 (16)	0	0	0	1	0	0	0	0	Stisknutá Klávesa 3
2	0x01 (1)	0	0	0	0	0	0	0	1	Stisknutá Klávesa 5
3	0x30 (64)	0	1	0	0	0	0	0	0	Stisknutý Knoflík
4	0x14 (20)	0	0	0	1	0	1	0	0	Poloha UP a RIGHT
5	0x00 (0)	0	0	0	0	0	0	0	0	-
6	0x00 (0)	0	0	0	0	0	0	0	0	-
7	0x00 (0)	0	0	0	0	0	0	0	0	-

## 8 Ovládání akčních členů

Dle zadání má být ovládání akčních členů umožněno jak pomocí vizualizace, tak pomocí ovladače IFM. Tato kapitola popisuje obě možnosti ovládání a též i zapojení aktuátorů k PLC.

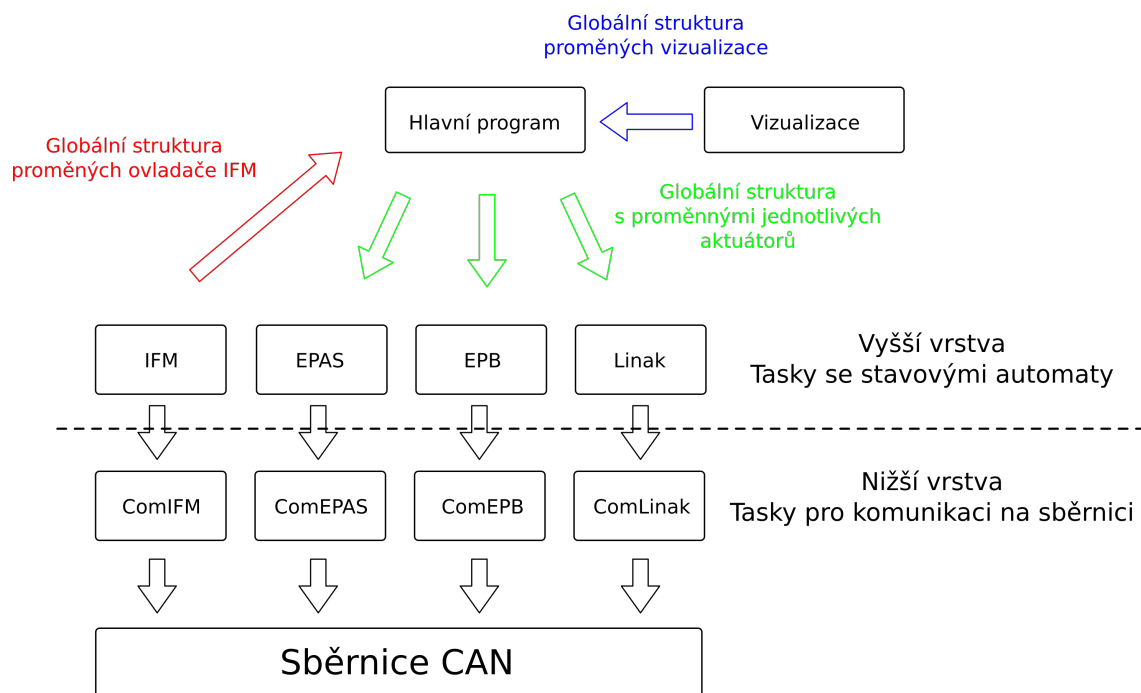
Obě možnosti ovládání mají umožňovat servisnímu technikovi nezávislé ovládání jednotlivých aktuátorů elektromobilu. Servisní technik by měl být schopen např. zabrzdít a odbrzdít parkovací brzdy EPB, libovolně pohybovat systémem EPAS a pohonem LA36 a identifikovat a odstranit případné chyby řídicích jednotek atd.

Veškerá komunikace s řídicími jednotkami aktuátorů je zprostředkována pomocí PLC (z ang. Programmable Logic Controller). PLC je programovatelný logický automat reálného času, kterému je možné nastavovat taskové třídy. Do taskových tříd se vkládají programy, které jsou vykonávány v cyklech daných nastavením taskové třídy. Operační systém použitého PLC je AR (z ang. Automation Runtime). Tomuto systému je možné nastavit celkem 8 taskových tříd. Programy v jedné taskové třídě jsou vykonávány postupně. Pokud dojde k požadavku k vykonání programů z více taskových tříd v jeden časový okamžik, jsou přednostně vykonány programy z taskové třídy s vyšší prioritou. První tasková třída má nejvyšší prioritu a z toho důvodu jsou programy v ní vykonávány bez přerušování. U první taskové třídy se většinou nastavuje nulová tolerance prodlevy při vykonání programů. Tím je zaručeno, že programy v první taskové třídě budou vždy vykonány přesně v požadovaných cyklech. Pokud dojde k prodlevě při vykonávání programu z první taskové třídy, přejde PLC do chybového stavu a je nutné jej restartovat. Do první taskové třídy se z toho důvodu vkládají programy, u nichž je kladen velký důraz na vykonání v požadovaném cyklu.

Každému aktuátoru byly přiděleny dva programy. První program se stará o komunikaci po sběrnici CAN s daným aktuátorem. Tento program nastavuje hodnoty proměnných, které se používají v PLC. A podle aktuálního stavu stavového automatu posílá dané řídicí jednotce příkazy. Při komunikaci po sběrnici CAN je u řídicích jednotek aktuátorů kladen velký důraz na přesnost posílání zpráv. Z toho důvodu jsou programy určené pro komunikaci na sběrnici CAN přiděleny do první taskové třídy. V rámci komunikace se dá tato část nazvat jako nižší vrstva systému. Vyšší vrstva systému je na úrovni druhého přiděleného programu. Tento program obsahuje stavový automat pro ovládání daného aktuátoru.

Aby bylo možné oddělit ovládání z vizualizace a ovladače IFM, byly vytvořeny celkem tři globální struktury proměnných. Pro ovládání aktuátorů slouží pouze jedna globální struktura. Zbylé dvě struktury byly použity pro nastavování proměnných vizualizace a ovladače IFM. To znamená, že proměnné nastavované z vizualizace jsou

v jiné struktuře proměnných než proměnné nastavované z ovladače IFM. Podle aktuálního režimu ovládání předává hlavní program hodnoty proměnných z vizualizace nebo ovladače IFM proměnným pro ovládání aktuátorů.



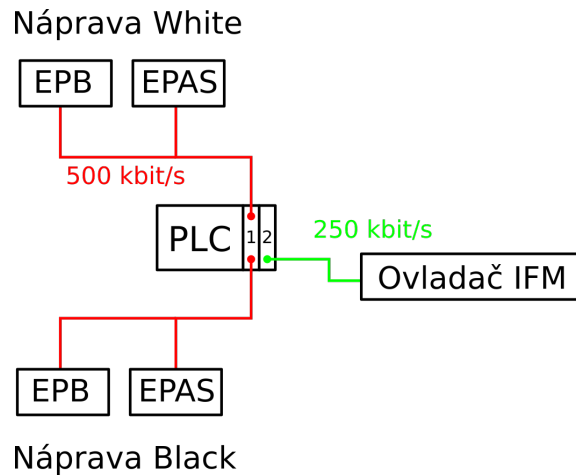
Obrázek 8.1: Struktura programu (zdroj autor)

Aby se zabránilo možné kolizi při současném použití vizualizace a ovladače IFM pro řízení elektromobilu, může být v jednu chvíli aktivovaná pouze jedna možnost ovládání. Vzhledem k tomu, že při použití vizualizace je nutné, aby bylo PLC připojené na síť a servisní technik měl k dispozici chytré zařízení viz níže, je při zapnutí PLC automaticky spuštěné ovládání elektromobilu z ovladače IFM.

## 8.1 Zapojení

Všechny systémy elektromobilu jsou připojeny k napájení přes spínač. Napájení je zprostředkováno pomocí dvou baterií na 12 a 24 V. Otočením spínače dojde k připojení systémů na příslušnou baterii a zároveň dojde i k posláni signálu o nastartování vozidla. Signál o nastartování vozidla je nutné poslat jednotkám EPAS a EPB. Tento signál je v automobilu standardně poslán při otočení klíčku zapalování.

Použité modulární PLC pro ovládání všech systémů je od společnosti B&R a ne-se označení X20CP1586. na obr. 8.2 je znázorněno připojení systémů EPB, EPAS a ovladače IFM k PLC. Systémy EPAS a EPB jsou připojeny na modul 1 a ovladač IFM na modul 2. Číslem 1 je nahrazen název modulu X20IF2772 a číslem 2 je nahrazen modul X20CS2770.



Obrázek 8.2: Schéma zapojení aktuátorů se systémy EPAS (zdroj autor)

Oba moduly obsahují dvě rozhraní, na které je možné připojit sběrnici CAN. Modul 1 umožňuje přesnější časování zpráv, které je kritické pro použití systému EPAS a EPB. Pokud dojde k určité prodlevě při poslání zprávy z PLC, vyhodnotí řídicí jednotky EPB a EPAS tento stav jako chybový a přestanou reagovat. Navíc je nutné, aby rychlost komunikace po sběrnici CAN se systémy EPAS a EPB byla 500 kbit/s. Tuto rychlost sice zvládne i modul 2, ale nedokáže potřebné zprávy po sběrnici posílat tak rychle, aby nedošlo k prodlevám. Proto byl modul 2 použit pro komunikaci s ovladačem IFM. Komunikace s ovladačem IFM probíhá při rychlosti 250 kbit/s, modul 2 posílá ovladači IFM méně zpráv než modul 1 a i perioda posílaných zpráv je značně pomalejší. Při tomto zapojení tak nedochází k chybovým stavům kvůli prodlevám mezi jednotlivými zprávami.



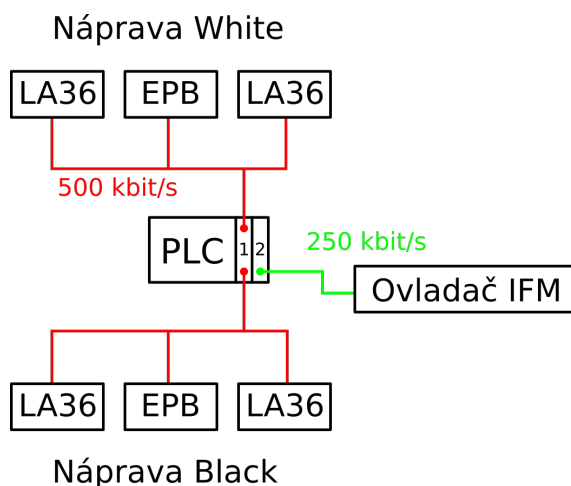
Obrázek 8.3: Modul X20IF2772



Obrázek 8.4: Modul X20CS2770

Na obr. 8.5 je zobrazeno zapojení se systémy EPB, ovladačem IFM a pohony

LA36. Ačkoliv pohony LA36 využívají pro komunikaci protokol SAE J1939, který má danou rychlost přenosu 250 kbit/s, je možné těmto pohonům nastavit i vyšší přenosovou rychlost. Díky tomu je možné pro připojení pohonů LA36 k PLC použít stejnou sběrnici CAN, na kterou byl připojen i systém EPAS a není tak nutné používat další modul pro sběrnici CAN.



Obrázek 8.5: Schéma zapojení aktuátorů s pohony LA36 (zdroj autor)

## 8.2 Vizualizace

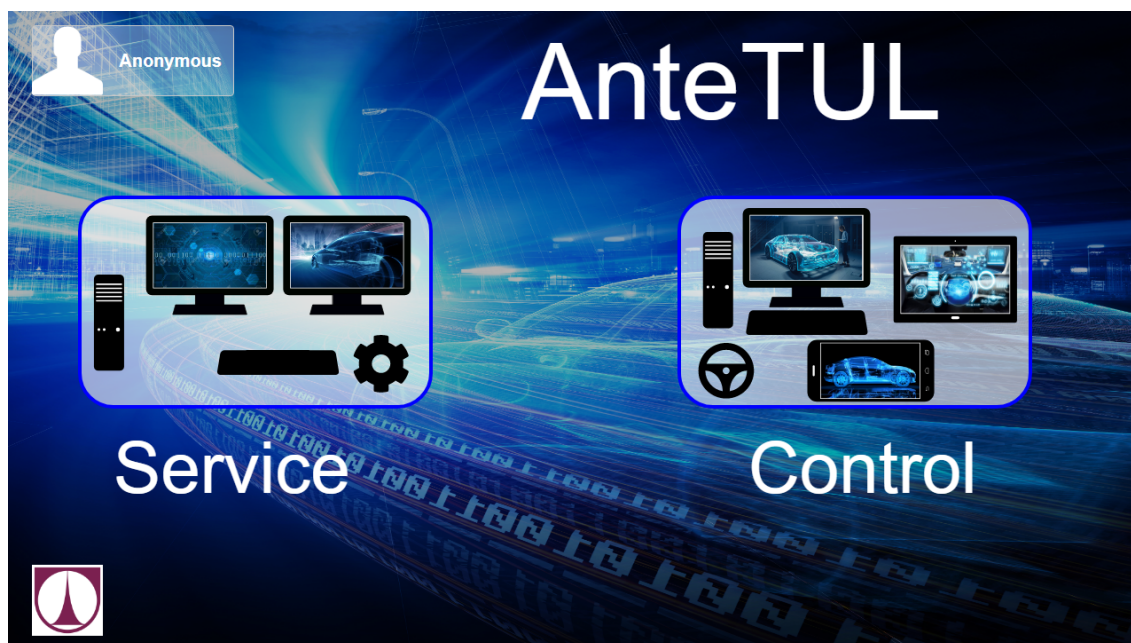
Ovládání pomocí vizualizace je zprostředkováno technologií mappView. Tato technologie umožňuje vytvořit webovou vizualizaci a je implementovaná v prostředí AS (z ang. Automation Studio). Vytvořená vizualizace je založena na webových technologiích jako je HTML5, CSS3 a JavaScript. Pro vytvoření webové vizualizace je standardně nutná znalost těchto technologií. S pomocí MappView je možné vytvořit vizualizaci, aniž by uživatel musel ovládat tyto technologie. Více o technologii MappView je možné se dočíst v bakalářské práci Grafické rozhraní servisního technika pro ovládání skladovacího robota [30].

K vytvořené vizualizaci je možné se připojit, jakýmkoliv chytrým zařízením (zařízením s přístupem na internet). Servisní technik má tak možnost pro připojení využít např. mobil, tablet nebo počítač. Stačí jen, aby zařízení servisního technika bylo připojené ke stejné síti jako je PLC a uživatel do internetového vyhledávače zadal příslušnou url adresu. Ta se skládá z IP (z ang. Internet Protocol) adresy PLC, portu a názvu vizualizace.

Po připojení k vizualizaci se nejprve uživateli zobrazí úvodní stránka. Tato stránka slouží pro zvolení režimu ovládání elektromobilu a pro přihlášení uživatele. Uživatel má možnost zvolit režim **Service**<sup>13</sup> nebo **Control**. Režim **Control** by měl v budoucnu umožnit jednodušší ovládání elektromobilu. V současné době tento režim není využit a jeho vyhotovení není ani součástí této práce, proto se o něm dále

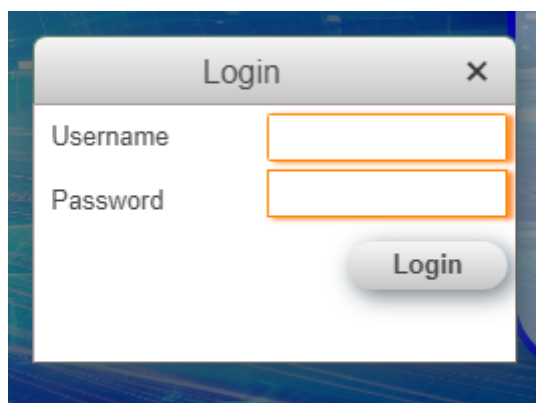
<sup>13</sup>V rámci této kapitoly tučně zvýrazněný text značí, že se jedná o prvky ve vizualizaci.

v této práci už nezmiňuje.

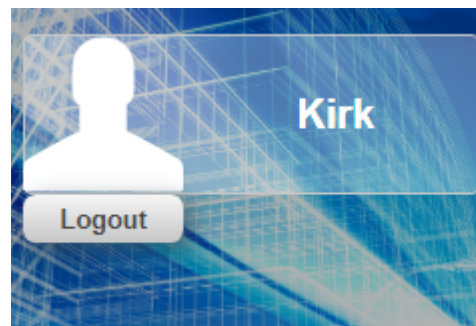


Obrázek 8.6: Úvodní stránka vizualizace (zdroj autor)

Režim **Service** zpřístupní uživateli servisní část vizualizace. Po kliknutí je od uživatele nejprve požadováno přihlášení, pokud se tak již nestalo.

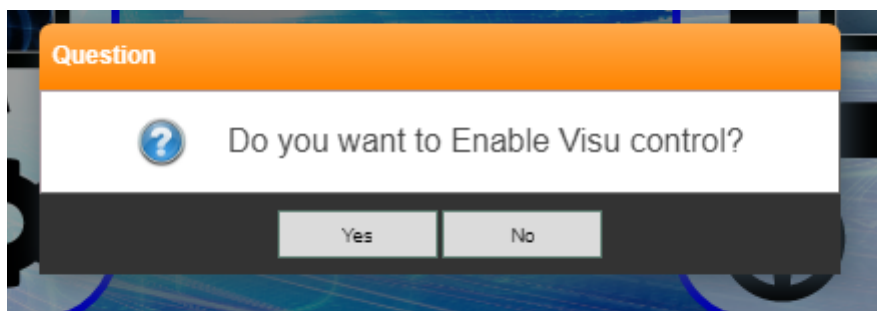


Obrázek 8.7: Přihlašovací okno



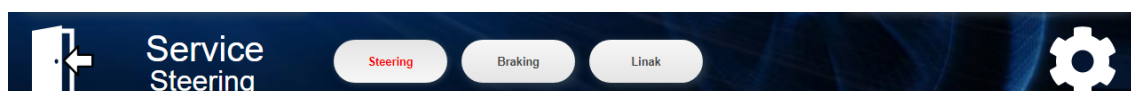
Obrázek 8.8: Přihlášený uživatel

Přihlášenému uživateli se následně zobrazí okno, ve kterém může aktivovat servisní režim z vizualizace. Pokud uživatel neaktivuje servisní režim z vizualizace, je možné elektromobil nadále ovládat jen pomocí ovladače IFM. V takovém případě se uživateli zobrazí servisní rozhraní se všemi údaji o jednotlivých aktuátorech, ale nemá možnost nastavovat jakékoliv parametry nebo zadávat příkazy řídicím jednotkám.



Obrázek 8.9: Spuštění ovládání z vizualizace (zdroj autor)

V případě aktivace servisního režimu z vizualizace dojde k přepnutí ovládání elektromobilu z ovladače IFM na vizualizaci a uživateli se zpřístupní všechny možnosti servisního rozhraní. Pro intuitivnější ovládání jsou jednotlivé aktuátory rozděleny na dílčí stránky. Ty je možné přepínat pomocí navigační lišty vizualizace viz obr. 8.10.



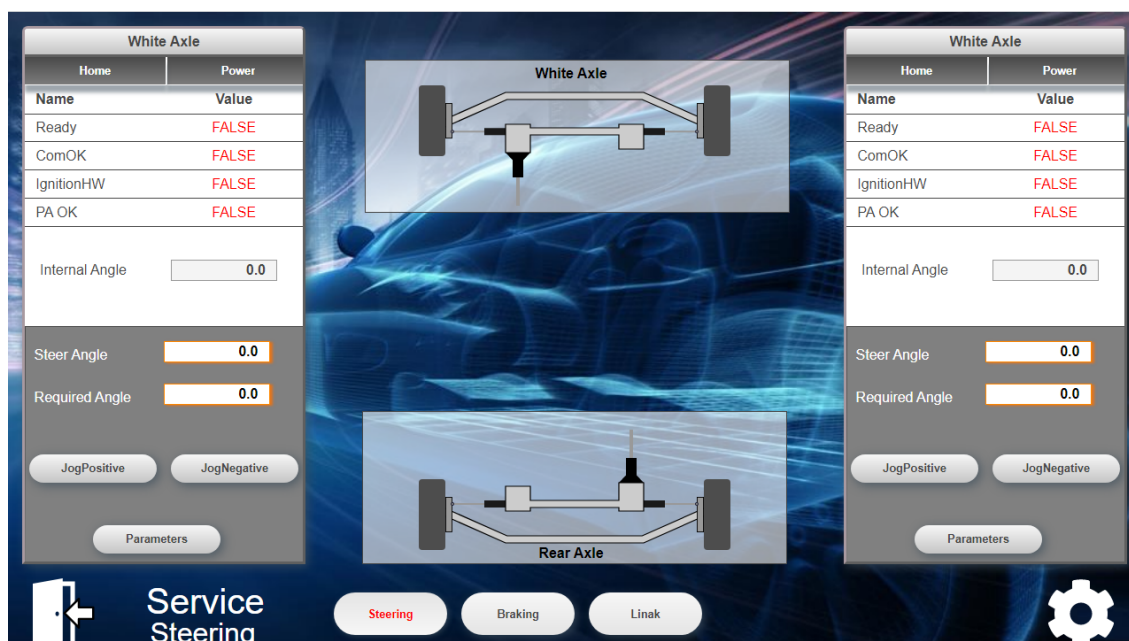
Obrázek 8.10: Navigační lišta (zdroj autor)

K opuštění servisního režimu slouží ikona dveří, která je umístěna v levé části navigační lišty. Po kliknutí dojde k návratu na úvodní stránku vizualizace a automaticky se ovládání servisního režimu předá zpátky ovladači IFM.

### 8.2.1 Stránka EPAS (vizualizace)

Jak název napovídá, stránka EPAS slouží k ovládání systému EPAS na nápravě White i Black. Ovládání pro obě nápravy je identické. Při zapnutí elektromobilu je komunikace s řídicími jednotkami prostřednictvím sběrnice CAN vypnuta. Komunikaci se systémy EPAS je možné zapnout stisknutím tlačítka **Power**. Úspěšné navázání komunikace s řídicí jednotkou systému a další stavy jsou indikovány binárními proměnnými. Dále je možné na stránce EPAS zjistit hodnotu natočení kol. Podle aktuální hodnoty natočení kol se natáčí i grafické zobrazení systému řízení uprostřed stránky. To je zprostředkováno pomocí obrázků SVG (z ang. Scalable Vector Graphics).



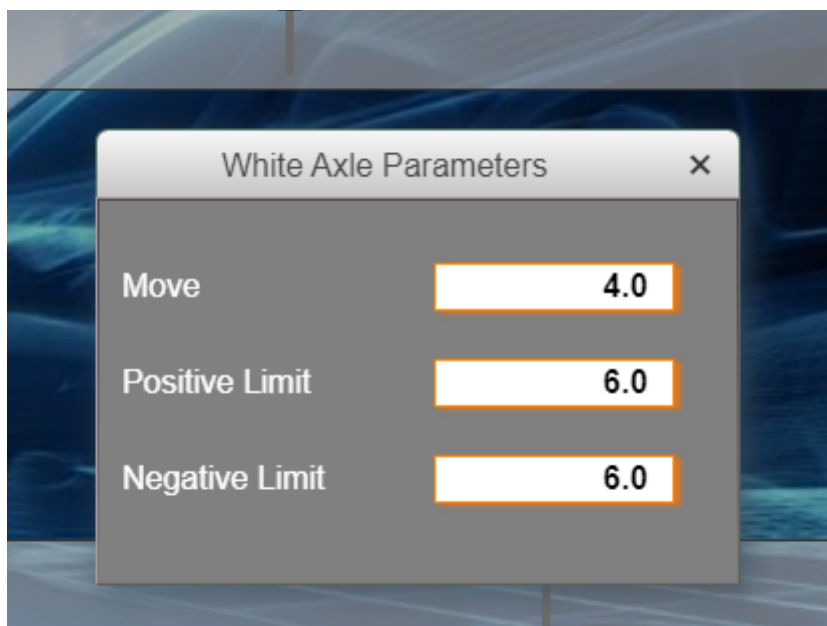


Obrázek 8.11: Stránka EPAS ve vizualizaci (zdroj autor)

Jednotkám EPAS je dále možné posílat příkazy k natáčení kol v pozitivním nebo negativním směru. Vysunutí jednotky EPAS je dáno úhlem natočení volantu. Tento úhel nabývá hodnot od  $-450^\circ$  do  $450^\circ$ . Natáčení kol v pozitivním směru znamená, že se úhel natočení volantu zvyšuje. Negativní směr natáčení pak způsobuje, že se úhel natočení volantu snižuje. Jakým směrem se budou kola natáčet vůči vozidlu, je závislé na mechanickém zapojení celého systému řízení. Není tedy možné říci, že pozitivní směr pohybu natočení kol způsobí stáčení vozidla doprava ve směru jízdy nebo naopak.

Příkaz pro natáčení kol v pozitivním směru je zprostředkován tlačítkem **JogPositive** a pro negativní směr slouží tlačítko **JogNegative**. Zadáním hodnoty natočení do pole **Required Angle** začne řídicí jednotka EPAS sama natáčet kola na požadovanou hodnotu.

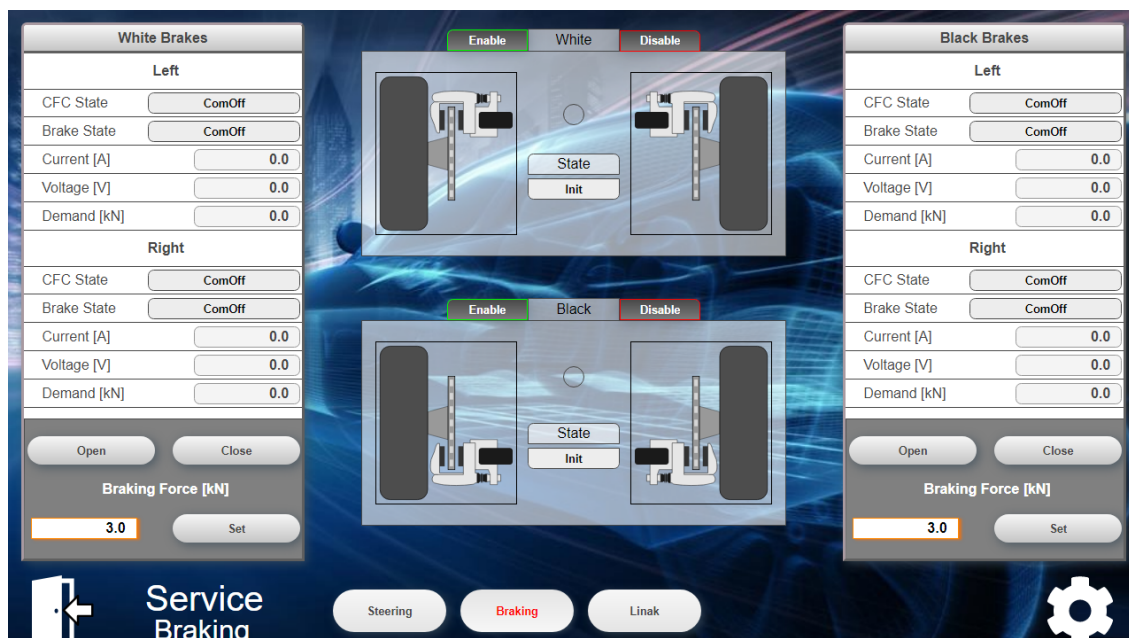
Kliknutím na tlačítko **Parameters** dojde k otevření okna pro nastavení parametrů jednotky EPAS. Tyto parametry umožňují měnit rychlost natáčení kol.



Obrázek 8.12: Nastavení parametrů jednotky EPAS (zdroj autor)

## 8.2.2 Stránka EPB (vizualizace)

Po kliknutí na tlačítko **Braking** v navigační liště dojde k přepnutí vizualizace na stránku EPB. Tato stránka umožňuje servisnímu technikovi aktivovat a deaktivovat komunikaci po sběrnici CAN s jednotkami EPB, posílat příkazy pro zabrzdění a odbrzdění, nastavovat velikost brzděné síly a číst některé stavové a procesní proměnné jednotek EPB.

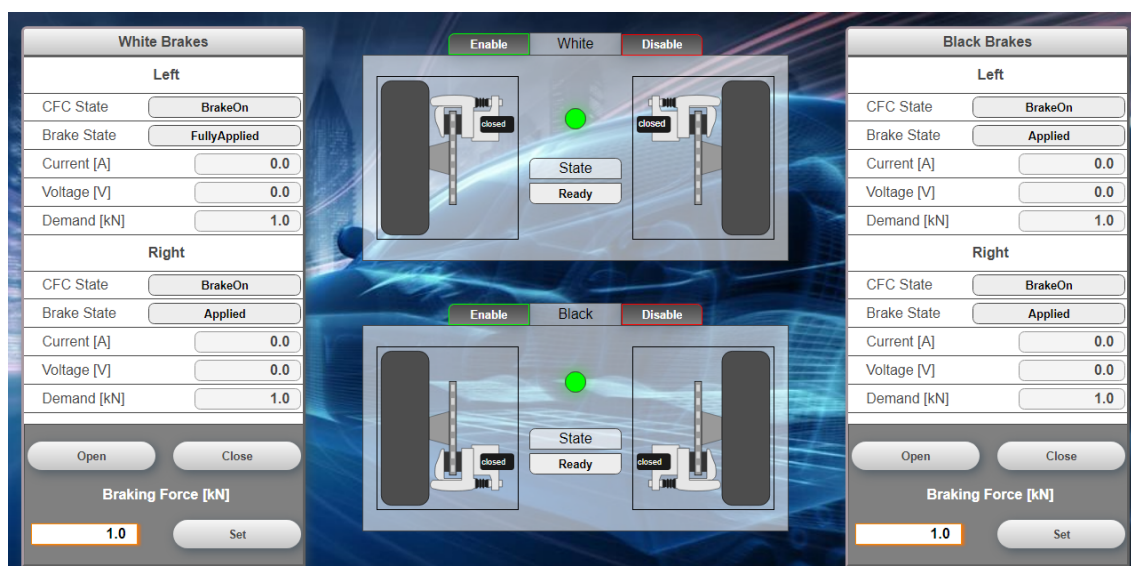


Obrázek 8.13: Stránka EPB ve vizualizaci (zdroj autor)

Pro obě nápravy je ovládání opět identické a je na krajích stránky. Tlačítka pro aktivaci komunikace jsou tentokrát umístěny uprostřed stránky nad grafickým duplikátem systému brzdění. Stisknutím tlačítka **Enable** dojde k aktivaci komunikace s jednotkou EPB a tlačítko **Disable** komunikaci deaktivuje.

Stav stavového automatu, který se stará o komunikaci s řídicí jednotkou EPB a posílání příkazů, je vypisován v textové podobě uprostřed stránky pro každou nápravu zvlášť (pod nápisem **State**). Pokud je s danou jednotkou navázána komunikace a jednotka je připravena poslouchat příkazy, je stavový automat ve stavu Ready a rozsvítí se zelené světýlko. Stavové a procesní proměnné jednotek EPB jsou zobrazovány v boxech<sup>14</sup> na krajích stránky. Grafické zobrazení čelistí systému EPB se pohybuje v závislosti na aktuálním stavu brzd. Aktuální stav každé brzdy je v grafickém zobrazení i textově vypsán nápisem **Closed** nebo **Opened**.

Prvky pro posílání příkazu pro zabrzdění (tlačítko **Close**), odbrzdění (tlačítko **Open**) a nastavení brzdné síly (**Braking Force**) se nachází ve spodní části boxu dané nápravy.



Obrázek 8.14: Zobrazení zabrzděného stavu ve vizualizaci (zdroj autor)

### 8.2.3 Stránka LINAK (vizualizace)

Pro ovládání pohonů LA36 slouží stránka LINAK. Každý z pohonů má na této stránce svůj vlastní grafický duplikát s tlačítky **Enable** a **Disable**, které slouží pro komunikaci s jednotkami pohonů. Pod grafickým duplikátem každého pohonu je vypsána aktuální hodnota vysunutí pístnice (**Position**) a stav obslužného stavového automatu (**State**). Hodnota vysunutí pístnice je zobrazována i polohou pístnice grafického duplikátu pohonu LA36.

<sup>14</sup>Box je označení prvku vizualizace, který obsahuje další libovolné vizuální prvky. Díky tomuto prvku je ovládání z vizualizace přehlednější a intuitivnější.

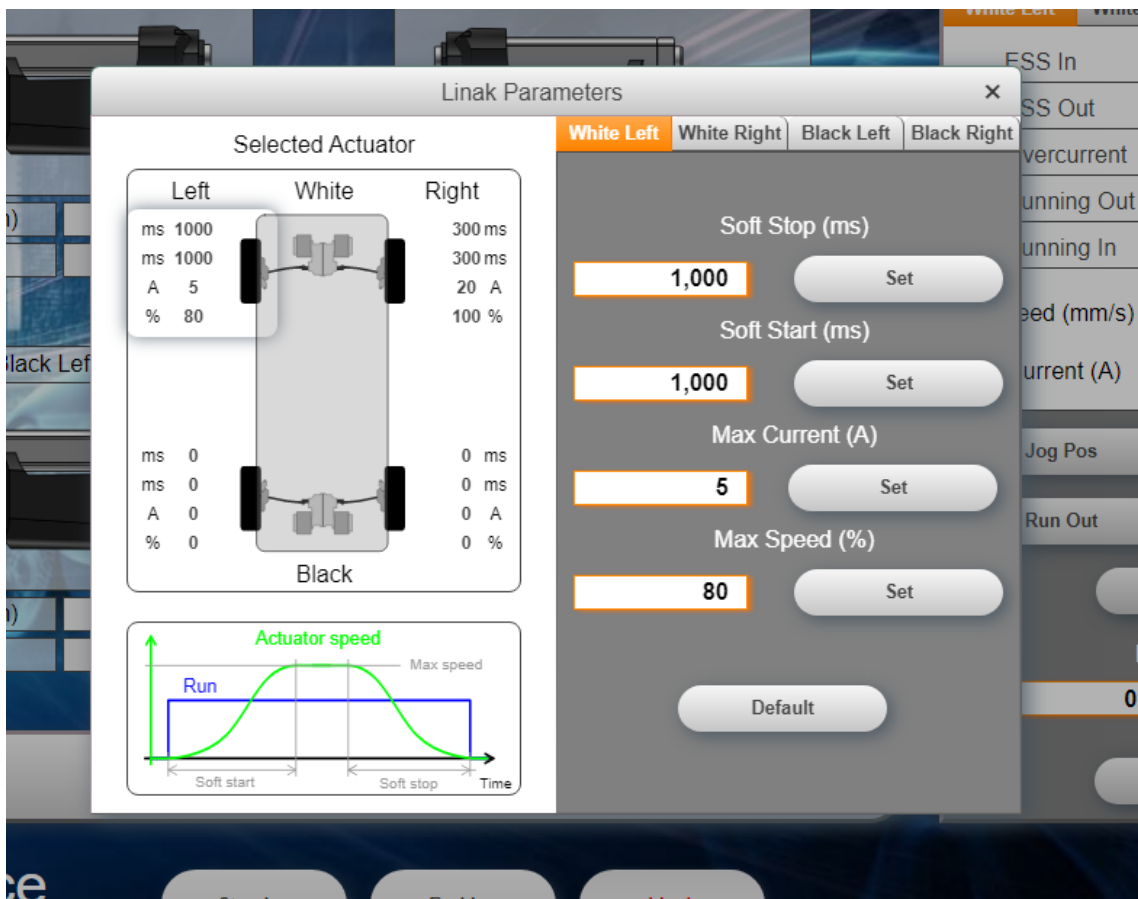


Obrázek 8.15: Stránka LINAK ve vizualizaci (zdroj autor)

Další stavové a procesní proměnné pohonu je možné zjistit pomocí boxu na pravé části stránky LINAK. V horní části tohoto boxu je umístěna lišta, která umožňuje přepínat zobrazení proměnných a ovládání jednotlivých pohonů LA36.

Servisní technik má možnost poslat řídicí jednotce pohonu hned několik příkazů. Jedná se o příkazy pro úplné vysunutí pístnice (tlačítko **Run Out**), úplné zasunutí pístnice (tlačítko **Run In**), vysouvání popř. zasouvání pístnice po dobu stisknutí tlačítka **Jog Pos** popř. **Jog Neg**, zadání požadované hodnoty vysunutí pístnice nebo může prostřednictvím tlačítka **Stop** poslat příkaz pro zastavení jakéhokoliv pohybu pístnice.

Jak bylo řečeno v podkapitole 6.1, pohon LA36 disponuje několika parametry, jenž je možné nastavovat. K nastavení těchto parametrů slouží dialogové okno. To se otevře po kliknutí na tlačítko **Parameters**.



Obrázek 8.16: Okno pro nastavení parametrů pohonů LA36 (zdroj autor)

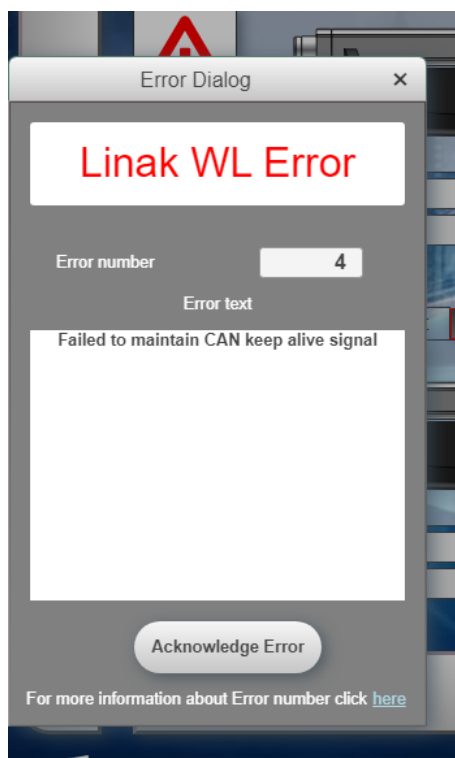
Servisní technik může prostřednictvím tohoto okna nastavit každému pohonu požadované parametry zvlášť. Aktuální nastavené hodnoty parametrů jsou vypisovány v levé části okna. Pravá část okna umožňuje přepínat mezi jednotlivými pohony, nastavovat požadované hodnoty parametrů nebo stisknutím tlačítka **Default** nastavit výchozí hodnoty parametrů z výroby.

Aktivní chybový stav pohonu LA36 je ve vizualizaci signalizován ikonkou vykřičníku. Pokud je alespoň jeden pohon v chybovém stavu, je vykřičník zobrazen i v navigační liště u tlačítka **Linak**.



Obrázek 8.17: Zobrazení aktivního chybového stavu pohonu LA36 ve vizualizaci (zdroj autor)

Po kliknutí na ikonu vykřičníku se otevře chybové okno daného pohonu. Toto okno vypisuje servisnímu technikovi číslo a textový popis chyby pohonu. Servisní technik má tak možnost přesně chybu identifikovat a odstranit její příčinu. Po odstranění příčiny může stisknout tlačítko **Acknowledge Error**, které pošle řídicí jednotce daného pohonu příkaz pro smazání chyby. Pokud byla příčina chyby úspěšně odstraněna, zmizí ikona vykřičníku a je možné pohon libovolně používat.



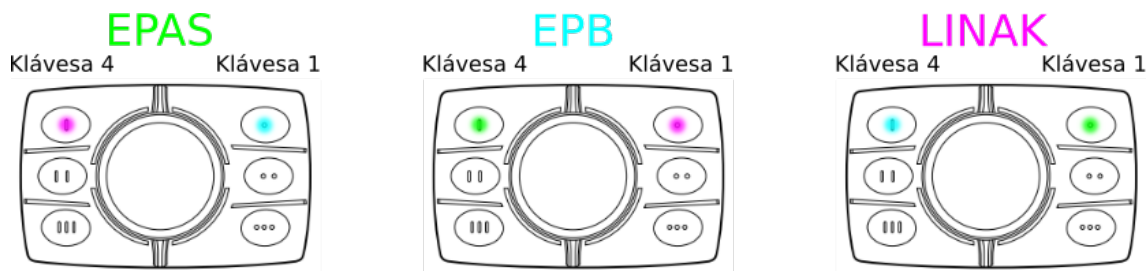
Obrázek 8.18: Zobrazení chybového okna pohonu LA36 ve vizualizaci (zdroj autor)

### 8.3 Ovládání prostřednictvím ovladače IFM

Ve chvíli, kdy dojde k navázání komunikace mezi PLC a ovladačem IFM prostřednictvím sběrnice CAN, rozsvítí se na ovladači všechny LED segmenty modrou barvou. Ostatní indikátory jsou zhasnuty. To signalizuje uživateli, že je ovladač připraven k použití. Pro ovládání servisního režimu z ovladače, je nutné ovladač nejprve aktivovat. To se provede současným stisknutím Kláves 1 a 4 po dobu 4 s.

Stejně jako u vizualizace je i ovládání jednotlivých aktuátorů z ovladače IFM rozděleno na stránky. Ovladač IFM nedisponuje tak rozsáhlým množstvím indikátorů a prvků jako vizualizace. Z toho důvodu je ovládání z ovladače trochu omezenější.

Jednotlivé stránky aktuátorů v ovladači IFM odpovídají vesměs stránkám ve vizualizaci. Opět je možné mezi těmito stránkami přepínat a ovládat vždy jen jeden typ aktuátoru. Pro jednoduché rozlišení, na které stránce se momentálně uživatel nachází, byly jednotlivým stránkám přiděleny barvy. Stránce EPAS byla přidělena zelená barva, stránce EPB tyrkysová a stránce pro řízení pohonů LA36 fialová. LED segmenty kruhu mají barvu aktuálně vybrané stránky. Pro přepínání stránek je možné použít Klávesu 1 nebo 4. Klávesa 1 přepne ovládání na následující stránku a Klávesa 4 na předchozí. Obě klávesy svítí barvou stránky, na kterou přepnou ovládání. Obr. 8.19 zobrazuje cyklus přepínání jednotlivých stránek. Pořadí stránek je EPAS, EPB a LINAK.



Obrázek 8.19: Přepínání stránek na ovladači IFM (zdroj autor)

Klávesy 1 a 4 tedy slouží k aktivaci ovladače a k přepínání stránek. Pokud je ovladač aktivní, je možné současným podržením Klávesy 1 a 4 ovladač i deaktivovat. Funkce Kláves 1 a 4 platí pro všechny stránky stejně. Funkce ostatních kláves a Knoflíku se liší podle aktuální stránky tedy aktuálně ovládaného aktuátoru.

### 8.3.1 Stránka EPAS

Pokud má servisní technik aktivovaný ovladač a je na stránce EPAS, dokáže prostřednictvím ovladače posílat řídicí jednotce systému EPAS příkazy pro změnu úhlu natočení kol a nastavovat rychlost natáčení.

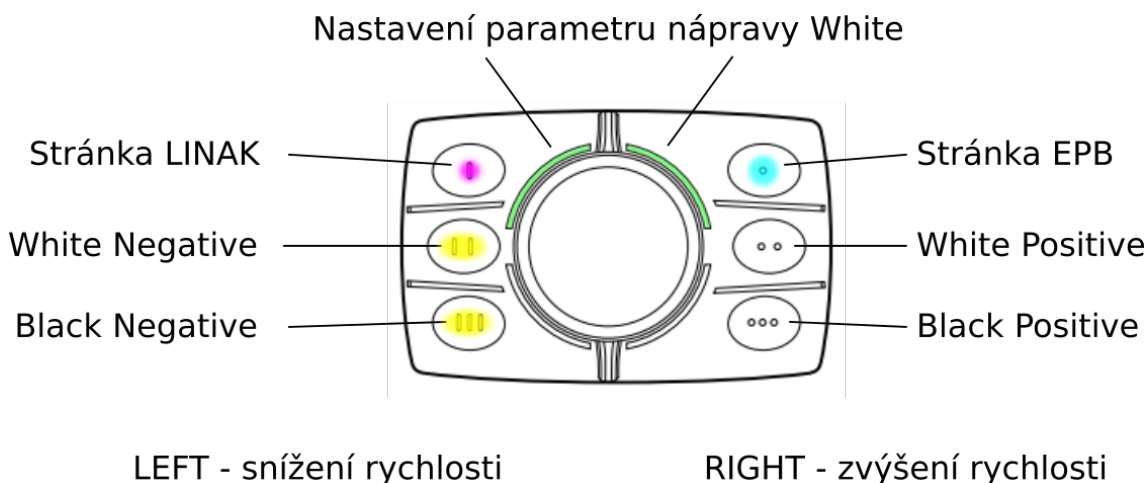
Natáčení kol je možné pomocí Kláves 2, 3, 5 a 6. Klávesy 2 a 5 slouží k natáčení kol na nápravě White a Klávesy 3 a 6 na nápravě Black. Stisknutím Klávesy 2 nebo 3 dojde k natáčení kol v pozitivním směru. Tyto dvě klávesy svítí bílou barvou. Stisknutím Klávesy 5 nebo 6 dojde k natáčení kol v negativním směru a klávesy svítí žlutou barvou. Poloha vysunutí je u jednotky EPAS dána úhlem natočení volantu. Ten je v rozsahu hodnot od  $-450^\circ$  do  $450^\circ$ . Pozitivní směr natáčení kol znamená, že se úhel natočení volantu zvyšuje. Negativní směr natáčení pak způsobuje, že se úhel natočení volantu snižuje. Jakým směrem se budou kola natáčet vůči vozidlu, je závislé na mechanickém zapojení celého systému řízení. Není tedy např. možné říci, že pozitivní směr pohybu natočení kol způsobí stáčení vozidla doprava ve směru jízdy.

Kola se natáčejí tak dlouho, dokud je některá klávesa stisknuta. Pokud dojde na jedné nápravě k požadavku na pohyb v pozitivním i negativním směru najednou (např. současné stisknutí Kláves 2 a 5), pošle se příkaz pro zastavení pohybu. Je ale možné současně stisknout dvě klávesy, z nichž každá ovládá jinou nápravu. Díky tomu je možné natáčet obě nápravy najednou ve shodném i opačném směru.

Rychlost pohybu natáčení kol se dá měnit pomocí Knoflíku. Při spuštění je nastavena stejná výchozí hodnota pro obě nápravy. Servisní technik má možnost nastavit každé jednotce jinou hodnotu rychlosti. Ke zvýšení rychlosti natáčení kol je potřeba posunout Knoflík do polohy RIGHT a pro snížení rychlosti do polohy LEFT. Aktuálně zvolená jednotka (náprava elektromobilu), u které dojde ke změně parametru, je indikována zelenou barvou LED segmentů. Pokud je zvolena jednotka nápravy White, svítí LED segmenty 1 a 2. Pokud je zvolena jednotka nápravy Black, svítí LED segmenty 3 a 4. Zvolenou nápravu je možné měnit pohybem Knoflíku. V případě, že je aktivováno řízení jednotky EPAS na nápravě White, musí servisní technik



posunout Knoflíkem do polohy DOWN, aby došlo k přepnutí na nápravu Black. V opačném případě je nutné posunout Knoflíkem do polohy UP. Aktuální hodnota rychlosti dané nápravy je dána jasem LED segmentů. Nastavováním vyšší hodnoty rychlosti se jas LED segmentů zvyšuje a nastavováním nižší hodnoty se jas zmenšuje.



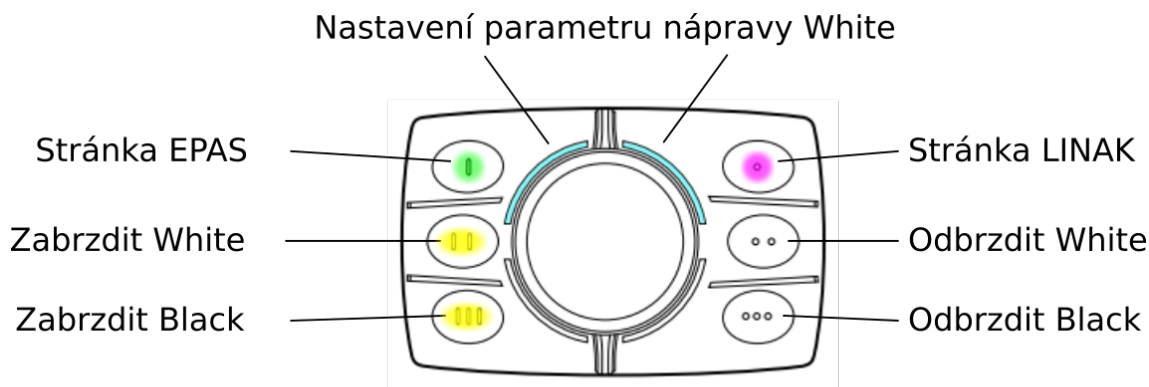
Obrázek 8.20: Stránka EPAS (zdroj autor)

### 8.3.2 Stránka EPB

Stránka EPB je velmi podobná stránce EPAS. Servisní technik může prostřednictvím této stránky poslat řídicí jednotce parkovacích brzd příkaz pro zabrzdění, odbrzdění a také nastavit sílu, kterou má jednotka vyvinout při brzdění.

Pro poslání základních příkazů slouží Klávesy 2, 3, 5 a 6. Klávesa 2 slouží pro odbrzdění nápravy White a Klávesa 5 pro zabrzdění. Klávesa 3 umožňuje odbrzdit nápravu Black a Klávesa 6 pošle příkaz k zabrzdění. Klávesy pro zabrzdění (5 a 6) svítí žlutou barvou a klávesy pro odbrzdění (2 a 3) svítí bílou barvou.

Nastavování požadované síly brzdění je opět možné pomocí Knoflíku. Stejně jako u stránky EPAS polohy Knoflíku UP a DOWN přepínají nápravu, na které dojde ke změně parametru (brzdné síly). Poloha Knoflíku RIGHT hodnotu brzdné síly zvyšuje a poloha LEFT brzdnou sílu snižuje. Velikost brzdné síly je opět signalizována jasem LED segmentů dané nápravy.



LEFT - snížení brzdné síly

RIGHT - zvýšení brzdné síly

Obrázek 8.21: Stránka EPB (zdroj autor)

### 8.3.3 Stránka LINAK

Přepnutí na stránku LINAK umožňuje servisnímu technikovi vysouvat a zasouvat jednotlivé pohony LA36 a nastavovat jim požadovanou hodnotu rychlosti pohybu pístnice.

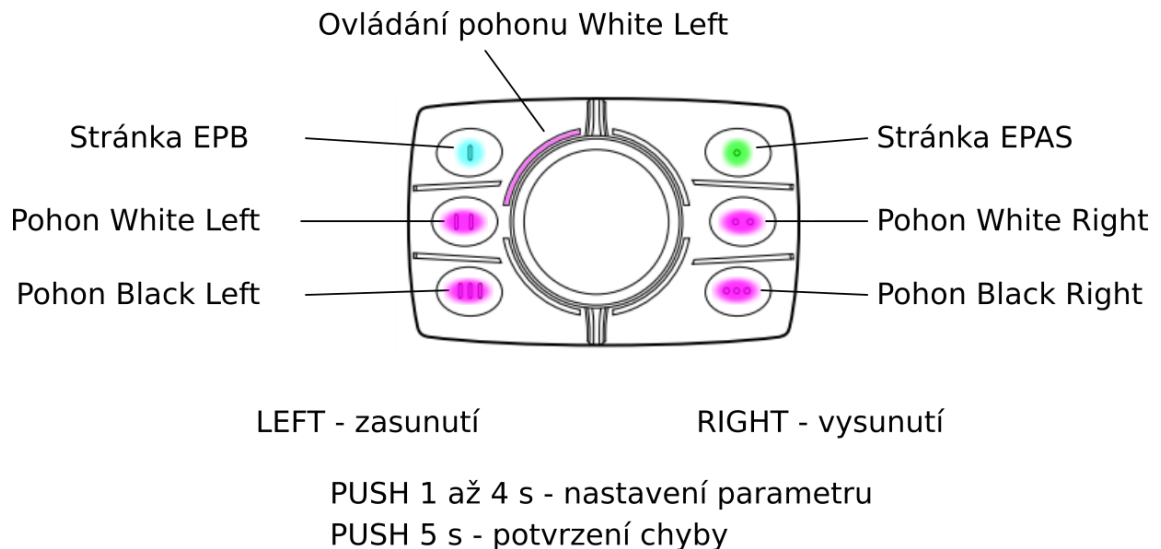
Aby bylo možné ovládat a nastavovat hodnotu rychlosti libovolnému pohonu LA36 i libovolné kombinaci pohonů, je nejprve nutné prostřednictvím Kláves 2, 3, 5 a 6 zvolit, jakému pohonu nebo kombinaci pohonů se budou posílat příkazy. Každá klávesa přidá nebo odebere pouze jeden pohon. Pohony, kterým se budou posílat příkazy, indikují LED segmenty. Rozsvícený LED segment znamená aktivní ovládání a zhasnutý neaktivní. Každý LED segment odpovídá právě jednomu pohonu. Pohonu, který je umístěn vlevo na nápravě White, byla přidělena Klávesa 5 a LED segment 1. Pohonu na druhé straně nápravy White byla přidělena Klávesa 2 a LED segment 2. Pohonu natáčející levé kolo na nápravě Black odpovídá Klávesa 6 a LED segment 4. A zbylá Klávesa 3 a LED segment 3 slouží pro přidání a odebrání pohonu na pravé straně nápravy Black.

Vybraným pohonům je možné poslat příkaz k vysunutí umístěním Knoflíku do polohy RIGHT a příkaz pro zasunutí posunutím Knoflíku do polohy LEFT.

K nastavení parametrů vybraných pohonů je potřeba stisknout Knoflík (PUSH) v intervalu od 1 s do 4 s. Pokud se Knoflík v tomto intervalu pustí, dojde k přepnutí stránky LINAK do režimu, který umožňuje nastavovat hodnotu nejvyšší rychlosti pohybu. Tento režim je signalizován periodickým blikáním LED segmentů vybraných pohonů. V tomto režimu se zcela změní funkce Knoflíku a není již možné posílat řídicím jednotkám pohonů příkazy pro vysunutí nebo zasunutí. Umístěním Knoflíku do polohy RIGHT dojde ke zvyšování hodnoty rychlosti pohybu pístnice vybraných pohonů a poloha LEFT tuto hodnotu snižuje. Nastavené hodnoty rychlosti jednotlivých pohonů jsou dány jasem kláves 2, 3, 5 a 6. Tyto klávesy odpovídají stejným pohonům jako v režimu pro ovládání pohonů. Pro potvrzení nastavených hodnot a opuštění režimu pro nastavování parametrů je potřeba stisknout Knoflík. Po uplynutí 1 s přestanou LED segmenty vybraných pohonů blikat a polohami Knoflíku

RIGHT a LEFT je opět možné posílat příkazy vybraným pohonům pro vysunutí a zasunutí.

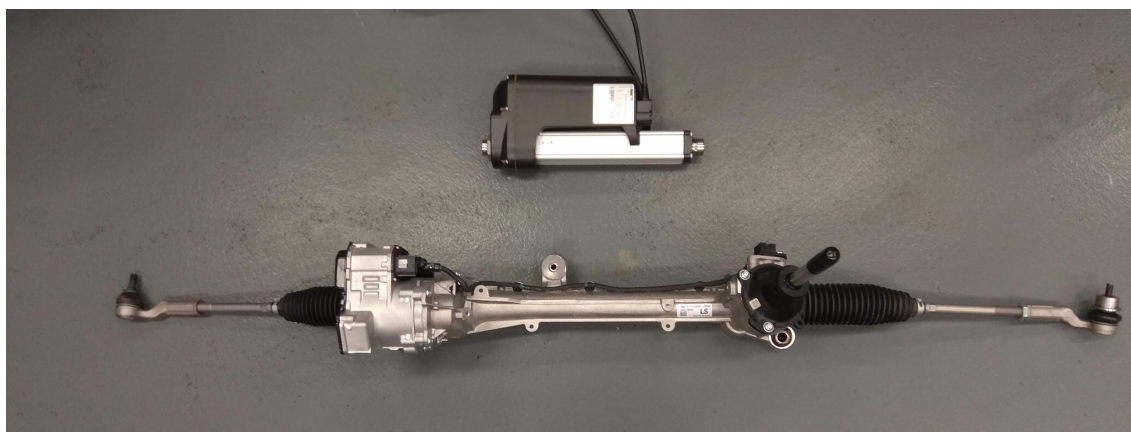
Pokud se v řídicí jednotce pohonu LA36 vyskytnou chyby, je možné je potvrdit stisknutím Knoflíku po dobu 5 s. Vzhledem k tomu, že může nastat několik různých chyb, je možné prostřednictvím ovladače IFM potvrdit jen ty méně závažné. Závažnější chyby je nutné potvrzovat pomocí vizualizace. Jak bylo řečeno výše, vizualizace umožňuje servisnímu technikovi zjistit, o jakou chybu se jedná. Servisní technik má tak možnost identifikovat příčinu chyby, tuto chybu odstranit a poté chybu z vizualizace potvrdit.



Obrázek 8.22: Stránka LINAK (zdroj autor)

## 9 Porovnání a testování systému EPAS a pohonu LA36

Systém EPAS i pohon LA36 mají umožnit u projektu AnteTUL směrové řízení kol. U projektu bude použita ale pouze jedna možnost ovládání. Tato kapitola se zabývá porovnáním systému EPAS s pohonem LA36 jak z hlediska mechanického, tak z hlediska komunikace prostřednictvím sběrnice CAN.



Obrázek 9.1: Porovnání systému EPAS a pohonu LA36 (zdroj autor)

### 9.1 Porovnání

#### Účel

- Systém EPAS je osvědčený systém, který se používá v řadě osobních automobilů jako posilovač řízení a parkovací asistent. Tento systém je přímo navrhnut pro natáčení kol vozidla.
- Oproti tomu pohon LA36 je lineární pohon, jehož využití není přesně specifikované a může být použit k různým účelům.

#### Samosvornost

- Systém EPAS se po zapnutí snaží neustále udržovat nastavený úhel natočení kol. Pokud bude na kola působit vnější síla, která se bude snažit natočit koly,

bude systém EPAS vyvíjet protisílu, aby tomuto pohybu zabránil. Pokud je ale systém EPAS vypnutý, nedochází k reakci na vnější sílu a je možné s koly pohybovat.

- U pohonu LA36 je zamezení nežádoucího pohybu řešeno pomocí integrované brzdy. Při výpadku energie nebo dosažení požadované polohy vysunutí pístnice dojde automaticky k aktivaci integrované brzdy, která brání v pohybu pístnice. Pokud je jednotce pohonu zadán požadavek na změnu polohy vysunutí pístnice, je brzda nejprve deaktivována.

### **Plynulost pohybu**

- Po zadání požadované hodnoty vysunutí pístnice pohonu LA36 se řídicí jednotka pohonu snaží dosáhnout této polohy s ohledem na zadané parametry. Při nastavení nenulových hodnot Soft Start a Soft Stop je možné docílit plynulého rozběhnutí a doběhnutí pístnice.
- Systému EPAS není možné zadávat jakékoliv parametry ovlivňující pohyb tyče řízení. Po zadání požadované hodnoty natočení kol se řídicí jednotka systému EPAS snaží dosáhnout požadované polohy co nejrychleji. To způsobuje značné rázy během pohybu tyče řízení. Navíc řídicí jednotce je možné zadávat požadovanou polohu jen v určitém rozmezí od aktuální polohy. Pokud bude požadovaná hodnota natočení kol mimo rozsah, dojde k chybovému stavu a musí se systém restartovat.

### **Komunikace po sběrnici CAN**

- Ačkoliv řídicí jednotky obou systémů komunikují na sběrnici CAN, je tato komunikace značně odlišná. Systém EPAS využívá pro komunikaci klasický CAN ve specifikaci 2.0A a pohon LA36 využívá protokol SAE J1939. Zapojení sběrnice CAN je pro oba systémy stejné. Data, která se po této sběrnici posílají, jsou zcela jiná.
- Systému EPAS je potřeba posílat několik zpráv (celkem deset) při rychlosti sběrnice 500 kbit/s. Tyto zprávy jsou posílány v pravidelných intervalech 10 nebo 20 ms v závislosti na dané zprávě. Každá zpráva obsahuje nejvyšší možný počet bajtů dat (tedy 8). Celkem je tedy zhruba každých 20 ms posláno po sběrnici 80 bajtů dat. Prostřednictvím těchto dat je možné řídicí jednotce systému EPAS zadat úhel natočení volantu, aktivovat posilovací nebo parkovací režim a zadávat požadovaný úhel natočení kol v parkovacím režimu. Všechny tyto údaje jsou po sběrnici posílány v zakódované podobě. Každý údaj je před posláním potřeba zakódovat pomocí vzorce daným výrobcem systému. Ve chvíli, kdy je nějaký údaj zakódován chybně nebo dojde k prodlevě v poslání některé ze zpráv, vyhodnotí tento stav řídicí jednotka jako chybový a je nutné ji restartovat.
- Řídicí jednotka systému EPAS naopak po sběrnici posílá jen jednu zprávu. Tato zpráva obsahuje zakódovaný údaj o natočení vnitřní hřídele systému. Jednotka tento údaj posílá pravidelně každých 20 ms.

- Jak bylo řečeno v kapitole 6.1, pohonu LA36 stačí posílat pouze jednu zprávu každých 250 ms nebo i častěji. Tato zpráva kromě nastavení požadované polohy vysunutí pístnice, umožňuje nastavení některých parametrů a dokáže posílat řídicí jednotce pohonu jednoduché příkazy. Rychlost sběrnice je u pohonu LA36 možné nastavit. Není tedy nutné použít pouze rychlost danou protokolem SAE J1939 (250 kbit/s).
- Řídicí jednotka pohonu, prostřednictvím zprávy na sběrnici CAN, informuje nadřazený systém o aktuální poloze vysunutí pístnice, rychlosti pohybu pístnice atd. každých 100 ms (viz kapitolu 6.1).
- K pohonu LA36 byla k dostání dokumentace popisující komunikaci po sběrnici CAN. U systému EPAS tato dokumentace k dispozici nebyla. Komunikace se systémem EPAS se postupně vytvářela pomocí poskytnutých programů, konzultací se zaměstnanci společnosti ZF Friedrichshafen AG a odposloucháváním sběrnice CAN při zapojeném systému EPAS.

### Omezení síly v tahu

- Systém EPAS neumožňuje prostřednictvím sběrnice CAN omezit sílu, kterou bude tyč řízení působit na mechanismus natáčení kol.
- U pohonu LA36 taktéž není možné nastavit omezení síly v tahu, ale je možné nastavit velikost proudu, kterou může pohon odebírat. U stejnosměrných elektromotorů je velikost momentu přímo úměrná velikosti odebíraného proudu. Omezením nejvyššího proudového odběru je tak možné omezit točivý moment elektromotoru a tím pádem i sílu v tahu při pohybu pístnice.

### Inicializace pohonu

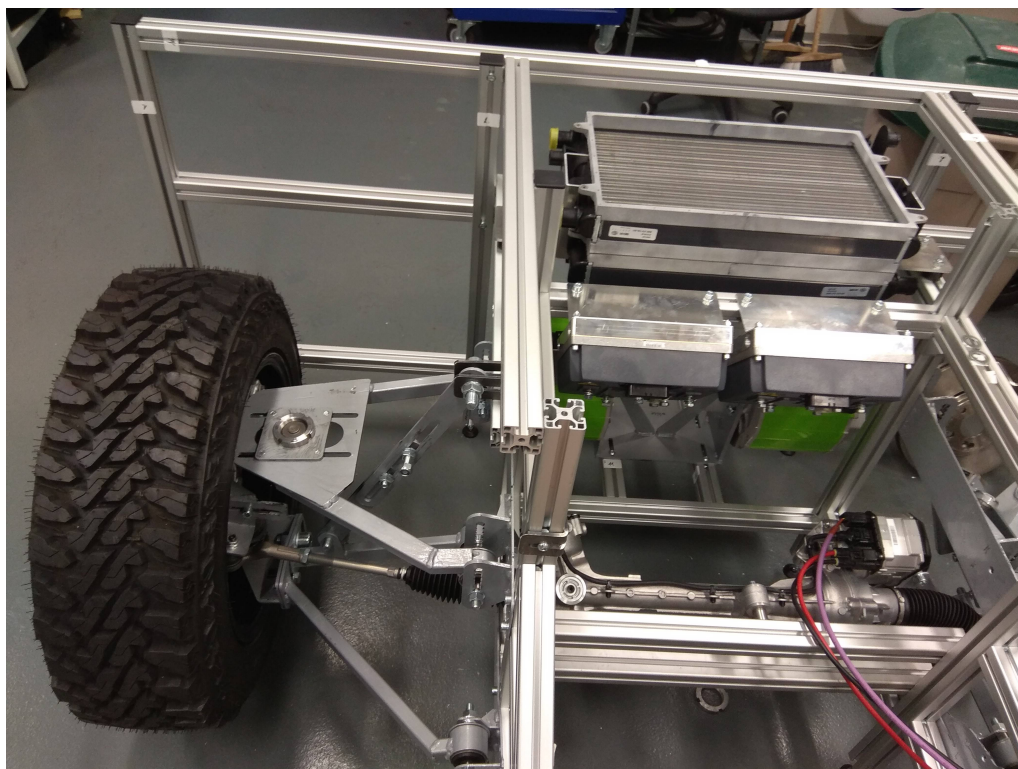
- Při zapnutí systému EPAS je nutné poslat řídicí jednotce systému aktuální úhel natočení volantu. Díky tomuto úhlu řídicí jednotka určí úhel natočení vnitřní hřídele a zinicilizuje se tak. Absence volantu tento údaj neumožňuje poslat a je tak nutné provést inicializaci jiným způsobem. Způsobů existuje více např. přidání absolutního enkodéru na vnitřní hřídel systému, přidání koncových snímačů detekující maximální vysunutí tyče řízení popř. natočení kol aj. Je také možné systém EPAS zinicilizovat pouze jednou a při vypnutí systému uložit aktuální hodnotu natočení kol. Při opětovném zapnutí by pak tento údaj byl poslán řídicí jednotce EPAS a došlo by k inicializaci. Tento způsob sice umožňuje použít systém EPAS bez nutnosti přidání senzoru, ale nevýhoda spočívá v možné ztrátě informace o úhlu natočení kol. K té může dojít v případě, kdy bude při vypnutém systému natáčeno koly vnější silou.
- Řídicí jednotka pohonu LA36 si neustále pamatuje poslední hodnotu vysunutí pístnice. Pokud dojde ke ztrátě této informace, uvědomí o tom jednotka nadřazený systém přechodem do chybového stavu. Opětovnou inicializaci je možné provést posláním příkazu pro úplné vysunutí nebo zasunutí pístnice. Při úplném vysunutí (zasunutí) pístnice dojde k aktivaci koncového snímače,

který je již od výroby integrován přímo v pohonu. Vzhledem k tomu, že jednotka přesně ví umístění koncového spínače, dokáže zjistit i polohu vysunutí pístnice a zinicizovat se tak.

## 9.2 Testování

Všechny použité aktuátory v této práci byly nejprve testovány na pracovišti. Aktuátory byly během testování připojeny k napájení a k PLC a podle poskytnutých informací byly vytvářeny části SW pro komunikaci a ovládání. Teprve ve chvíli, kdy se ověřila funkčnost ovládání a zapojení daného aktuátoru, byl aktuátor začleněn do projektu AnteTUL.

Během testování byl systém EPAS namontován na rám spolu s jednoduchým mechanismem pro natáčení kol viz obr. 9.2. Při pohybu tyče řízení tak docházelo i k natáčení kol. Rázy od tyče řízení se při pohybu přenášely na kola i celý rám. To způsobovalo vibrování rámu. Nutno dodat, že kola nebyla zatížena hmotností elektromobilu. Systém EPAS tak překonával pouze sílu potřebnou k natočení kol danou mechanismem řízení a hmotností kol. Při použití parkovacího asistenta u osobního automobilu je většinou pohyb natáčení kol i volantů plynulý. Je tedy možné, že rázy od tyče řízení budou utlumeny zvýšením potřebné síly pro natáčení kol.



Obrázek 9.2: Ukázka zapojení systému EPAS na pracovišti (zdroj autor)

Aby se ověřilo, zda bude možné použít pohon LA36 pro směrové řízení kol, byl vytvořen testovací rám a SW. Rám umožňoval na pohon zavěsit závaží, které simu-

lovalo sílu potřebnou k natočení kol viz obr. 9.3. Testovací SW pak cyklicky posílal pohonu příkazy pro vysunutí a zasunutí a neustále sledoval parametry pohonu. Celkem byly provedeny dva tříhodinové testy se závažím i bez. Ani jednou se v pohonu LA36 neobjevila chyba, která by signalizovala přetížení pohonu.



Obrázek 9.3: Testovací rám pohonu LA36 (zdroj autor)



## 10 Závěr

V diplomové práci se podařilo úspěšně realizovat všechny body zadání. V rámci diplomové práce byl vytvořený SW díky němuž je možné prostřednictvím ovladače IFM nebo webové vizualizace ovládat jednotlivé aktuátory vyvíjeného elektromobilu.

Během realizace této práce bylo nutné doplnění znalostí autora o sběrnici CAN. Především se jednalo o specifikaci CAN 2.0B a protokol SAE J1939, který v rámci magisterského projektu nebyl použit. Aby bylo možné prostřednictvím tohoto protokolu komunikovat s řídicími jednotkami aktuátorů, nestačilo pouze pochopit daný protokol, ale bylo nutné porozumět i použité technologii pro komunikaci po sběrnici CAN.

Během návrhu způsobu ovládání pomocí ovladače IFM i pomocí webové vizualizace byl kladen velký důraz na jednoduchost a intuitivnost ovládání. Ovládání bylo zároveň navrženo tak, aby servisní technik nebyl nijak omezen použitím jednoho nebo druhého způsobu ovládání. Splnění těchto podmínek během vytváření SW způsobovalo určitá úskalí. V některých případech byla i celá část ovládání jednoho z aktuátorů kompletně předělána, aby se docílilo lepších výsledků.

Ačkoliv se použití systému EPAS na začátku projektu AnteTUL ukazovalo jako nejlepší řešení, došlo nakonec k jeho nahrazení pohonem LA36. Důvodů bylo hned několik. Komunikace s řídicí jednotkou EPAS je oproti pohonům LA36 značně náročnější jak po výpočetní stránce, tak i po stránce zatížení sběrnice CAN. Při použití systému EPAS se počítá s připojením volantu vozidla včetně předávání řídicí jednotce EPAS informací o úhlu natočení volantu. Systém EPAS není stavěný na použití bez volantu, což přináší značné komplikace při jeho použití. I z hlediska strojního se ukázalo použití systému EPAS jako nevhodné.

Touto diplomovou prací práce autora na SW a projektu AnteTUL zdaleka nekončí. Vytvořený SW i způsob ovládání bude v nadcházející měsících testován na pracovišti a je možné, že v budoucnu bude potřeba provést ještě dodatečné úpravy. Zajisté bude potřeba doplnit ovládání o další aktuátory jako je např. hydraulika pro nastavování výšky podvozku a odpružení celého vozidla, pohonné jednotky elektromobilu, provozní brzdy aj. Tyto aktuátory v současnosti nejsou na pracovišti namontovány. Koncem roku 2022 by měl být prototyp Generace 1 už zcela sestaven. V tomto období podstoupí vytvořený SW nejtěžší zkoušku a ukáže se, jestli je vytvořené ovládání vhodné pro řízení pohybu elektromobilu.

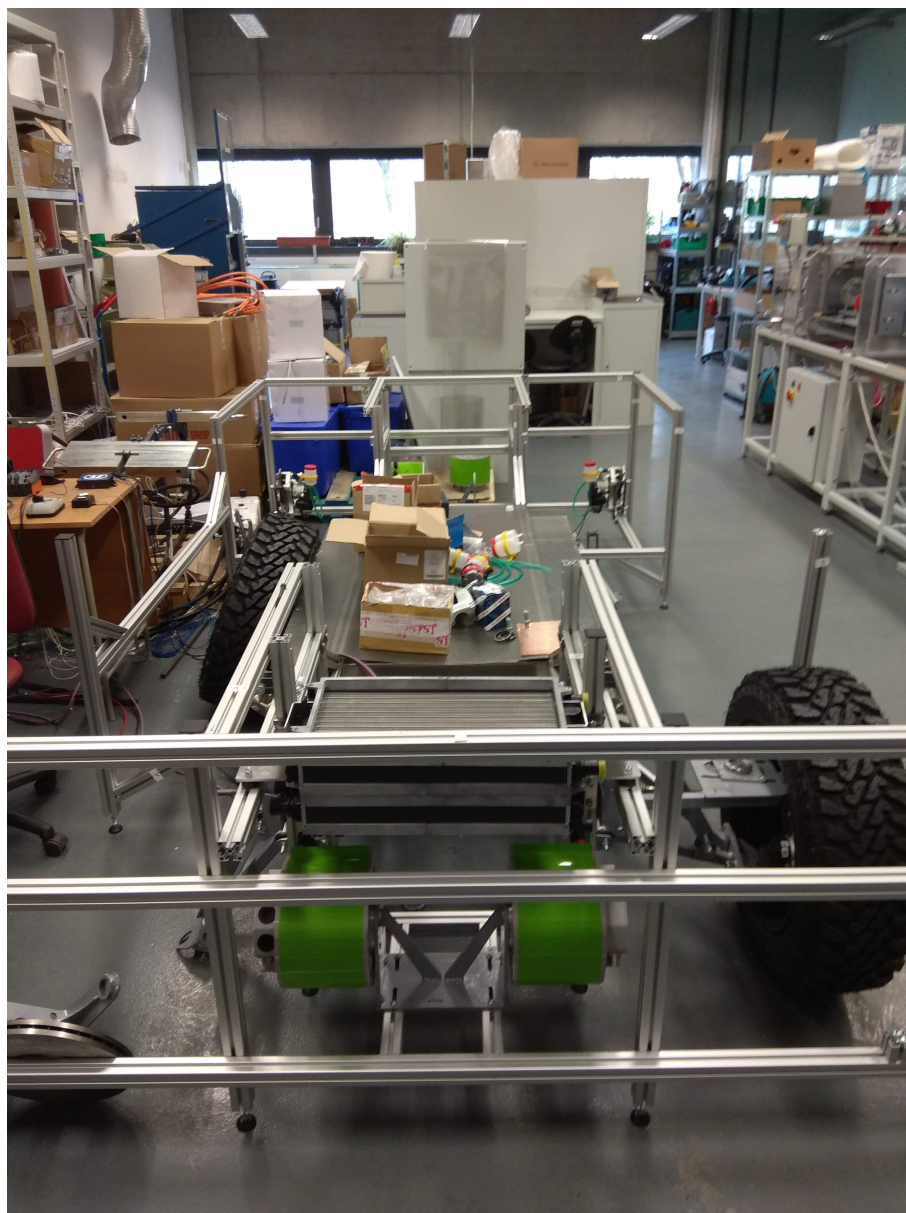
## Použitá literatura

- [1] KUBÍČEK, Tomáš. *Řízení akčních členů elektromobilu pomocí sběrnice CAN*. Liberec, 2021. Magisterský projekt. Technická univerzita v Liberci, Fakulta mechatroniky, informatiky a mezioborových studií.
- [2] ŠŤASTNÝ, Jiří; REMEK, Branko. *Autoelektrika a autoelektronika*. Vyd. 6. Praha: T. Malina, 2003. ISBN 80-862-9302-5.
- [3] Autonomní vozidlo z Liberce pomůže zvyšovat bezpečnost přepravy v průmyslových halách i v terénu. *Technický týdeník* [online]. [B.r.] [cit. 2022-05-12]. Dostupné z: [https://www.technickytydenik.cz/rubriky/archiv/autonomni-vozidlo-z-liberce-pomuze-zvysovat-bezpecnost-prepravy-v-prumyslovych-halach-i-v-terenu\\_54498.html](https://www.technickytydenik.cz/rubriky/archiv/autonomni-vozidlo-z-liberce-pomuze-zvysovat-bezpecnost-prepravy-v-prumyslovych-halach-i-v-terenu_54498.html).
- [4] *SURFACE VEHICLE RECOMMENDED PRACTICE: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. Warrendale, PA: SAE International, 2021. Č. J3016.
- [5] Pět stupňů k autonomnímu řízení. *ŠKODA Storyboard* [online]. [B.r.] [cit. 2022-05-12]. Dostupné z: <https://www.skoda-storyboard.com/cs/skoda-svet-cs/inovace-a-technologie/pet-stupnu-k-autonomnimu-rizeni/>.
- [6] *Honda Receives Type Designation for Level 3 Automated Driving in Japan* [online]. Tokio: Honda Motor Co., Ltd. [cit. 2022-05-12]. Dostupné z: <https://global.honda/newsroom/news/2020/4201111eng.html>.
- [7] *HONDA SENSING ELITE S AUTONOMNÍM ŘÍZENÍM TŘETÍ ÚROVNĚ* [online]. Praha: Honda Motor Europe Limited [cit. 2022-05-12]. Dostupné z: <https://www.honda.cz/cars/world-of-honda/experience-honda/news-events/honda-sensing-elite.html>.
- [8] MÁRA, Ondřej. *Mercedes předběhl Teslu, jeho autonomní řízení dostalo v Německu zelenou* [online]. Praha: CZECH NEWS CENTER a.s, c2001-2022 [cit. 2022-05-15]. Dostupné z: <https://www.auto.cz/mercedes-predbehl-teslu-jeho-autonomni-rizeni-dostalo-v-nemecku-zelenou-141945>.
- [9] MACKENZIE, Angus. *Mercedes-Benz Drive Pilot First Drive: It Actually Drives Itself* [online]. El Segundo, CA: MotorTrend [cit. 2022-05-12]. Dostupné z: <https://www.motortrend.com/reviews/mercedes-benz-drive-pilot-autonomous-first-drive-review/>.

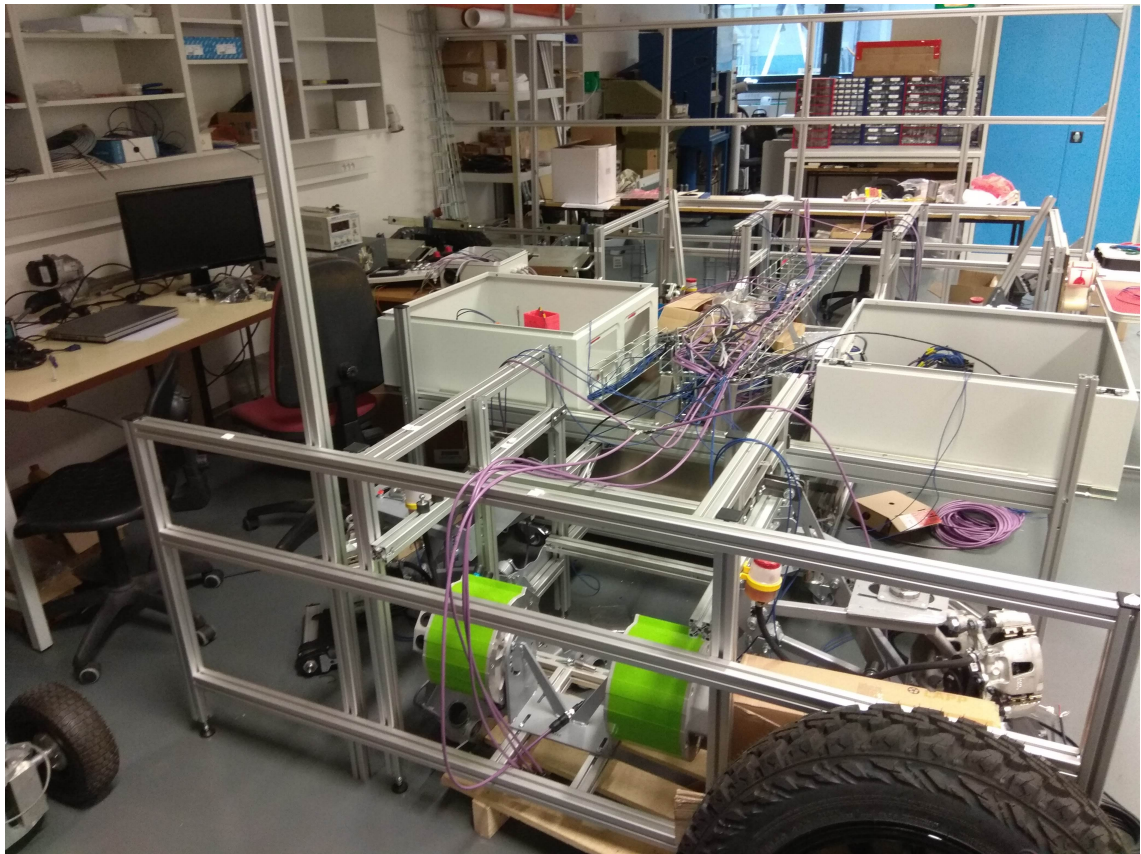
- [10] SRB, Luboš. *10 věcí, které rozhodně musíte vědět o značce Tesla* [online]. České Budějovice: ArteMan s.r.o. [cit. 2022-05-12]. Dostupné z: <https://elektrickevozy.cz/clanky/10-veci-ktere-rozhodne-musite-vedet-o-znacce-tesla>.
- [11] *Autopilot and Full Self-Driving Capability* [online]. Palo Alto, CA: Tesla [cit. 2022-05-12]. Dostupné z: <https://www.tesla.com/support/autopilot>.
- [12] TAUFIQ, Haque. *WAYMO VS TESLA: WHAT ARE THE DIFFERENCES?* [Online]. Sutton Coldfield: Motor Verso [cit. 2022-05-12]. Dostupné z: <https://www.motorverso.com/waymo-vs-tesla/>.
- [13] *Company* [online]. Mountain View, CA: Waymo LLC, c2019-2022 [cit. 2022-05-12]. Dostupné z: <https://waymo.com/company/>.
- [14] SRB, Luboš. *Tesla vs Waymo – Podívejte se na zřejmě první reálný souboj autonomních systémů* [online]. České Budějovice: ArteMan s.r.o. [cit. 2022-05-12]. Dostupné z: <https://elektrickevozy.cz/clanky/tesla-vs-waymo-podivejte-se-na-zrejme-prvni-realny-souboj-autonomnich-systemu>.
- [15] *CallasEV. Tesla FSD 10.4 VS Waymo* [online]. San Bruno, CA: YouTube [cit. 2022-05-12]. Dostupné z: <https://www.youtube.com/watch?v=1BsWFzgUBQY>.
- [16] *CAN Specification: Version 2.0*. Stuttgart: Robert Bosch GmbH, 1991.
- [17] *Classical Controller Area Network (CAN)* [online]. CAN in Automation (CiA), c1992-2022 [cit. 2022-05-12]. Dostupné z: <https://www.can-cia.org/can-knowledge/can/classical-can/>.
- [18] *Standardized higher-layer protocols* [online]. CAN in Automation (CiA), c1992-2022 [cit. 2022-05-12]. Dostupné z: <https://www.can-cia.org/can-knowledge/hlp/standardized-higher-layer-protocols/>.
- [19] *CAN FD - The basic idea* [online]. CAN in Automation (CiA), c1992-2022 [cit. 2022-05-12]. Dostupné z: <https://www.can-cia.org/can-knowledge/can/can-fd/>.
- [20] *SAE J1939 Know-how* [online]. Stuttgart: Vector Informatik GmbH, c2010-2022 [cit. 2022-05-12]. Dostupné z: <https://www.vector.com/at/en/know-how/protocols/sae-j1939///#>.
- [21] *J1939 Introduction* [online]. Kvaser, c2022 [cit. 2022-05-12]. Dostupné z: <https://www.kvaser.com/about-can/higher-layer-protocols/j1939-introduction/>.
- [22] *Cyclic redundancy check (CRC) in CAN frames* [online]. CAN in Automation (CiA), c1992-2022 [cit. 2022-05-12]. Dostupné z: <https://www.can-cia.org/can-knowledge/can/crc/>.
- [23] *Lineární pohon* [online]. San Francisco (CA): Wikimedia Foundation, c2021 [cit. 2022-05-12]. Dostupné z: [https://cs.wikipedia.org/w/index.php?title=Line%C3%A1rn%C3%AD\\_pohon&oldid=20027019](https://cs.wikipedia.org/w/index.php?title=Line%C3%A1rn%C3%AD_pohon&oldid=20027019).

- [24] Doba jde kupředu - přejděte na elektrické pohony. *LINAK* [online]. c2022 [cit. 2021-12-06]. Dostupné z: <https://cdn.linak.com/-/media/files/brochure-source/cz/techline-prejdete-na-elektriku-brozura-lcz.pdf>.
- [25] *Elektrické lineární pohony pro plynulé, inteligentní a výkonné polohování* [online]. Nordborg: LINAK C&S s.r.o., c2022 [cit. 2022-05-12]. Dostupné z: <https://www.linak.cz/>.
- [26] *LA 36 - technický list* [online]. Nordborg: LINAK C&S s.r.o., c2022 [cit. 2021-12-06]. Dostupné z: <https://cdn.linak.com/-/media/files/data-sheet-source/cz/pohon-la36-technicky-list-lcz.pdf>.
- [27] *SAE J1939 CAN bus: Version 3.1* [online]. Nordborg: LINAK C&S s.r.o., c2022 [cit. 2022-05-12]. Dostupné z: <https://cdn.linak.com/-/media/files/ic-and-bus-actuators/techline-can-bus-version-3-1-2-user-manual-eng.pdf>.
- [28] *CR1301 - tlačítkový modul s knoflíkem...* [Online]. Praha: Ifm electronic, c2022 [cit. 2022-05-13]. Dostupné z: <https://www.ifm.com/cz/cs/product/CR1301?tab=details>.
- [29] *Operating instructions ecomatPanel: CR1300, CR1301* [online]. Praha: Ifm electronic, c2022 [cit. 2022-05-13]. Dostupné z: <https://www.ifm.com/mounting/80296590UK.pdf>.
- [30] KUBÍČEK, Tomáš. *Grafické rozhraní servisního technika pro ovládání skladovacího robotu*. Liberec, 2020. Bakalářská práce. Technická univerzita v Liberci, Fakulta mechatroniky, informatiky a mezioborových studií.

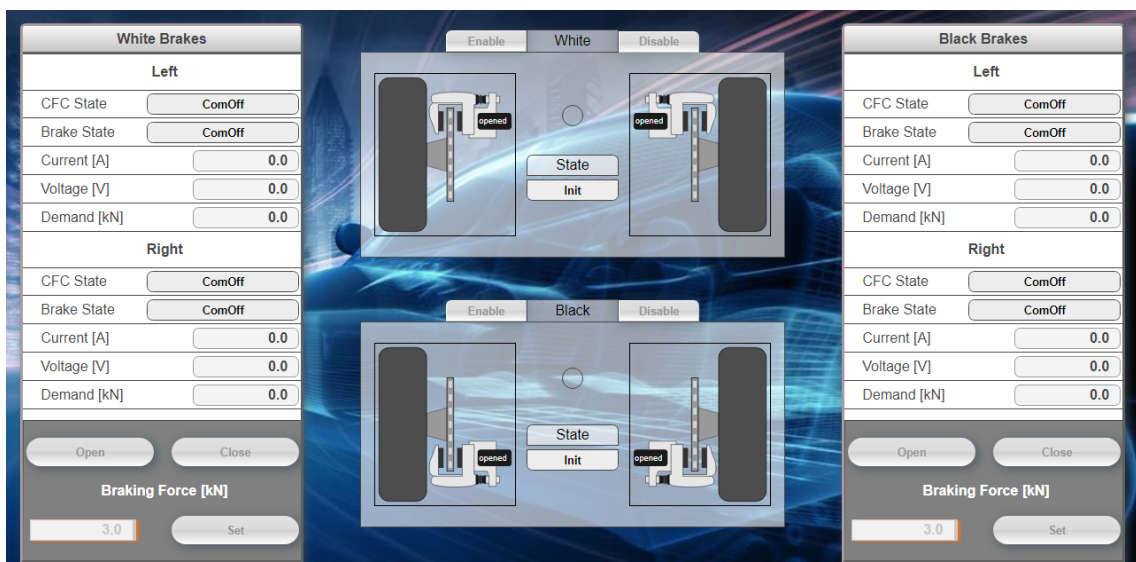
## A Doplnkové ilustrace



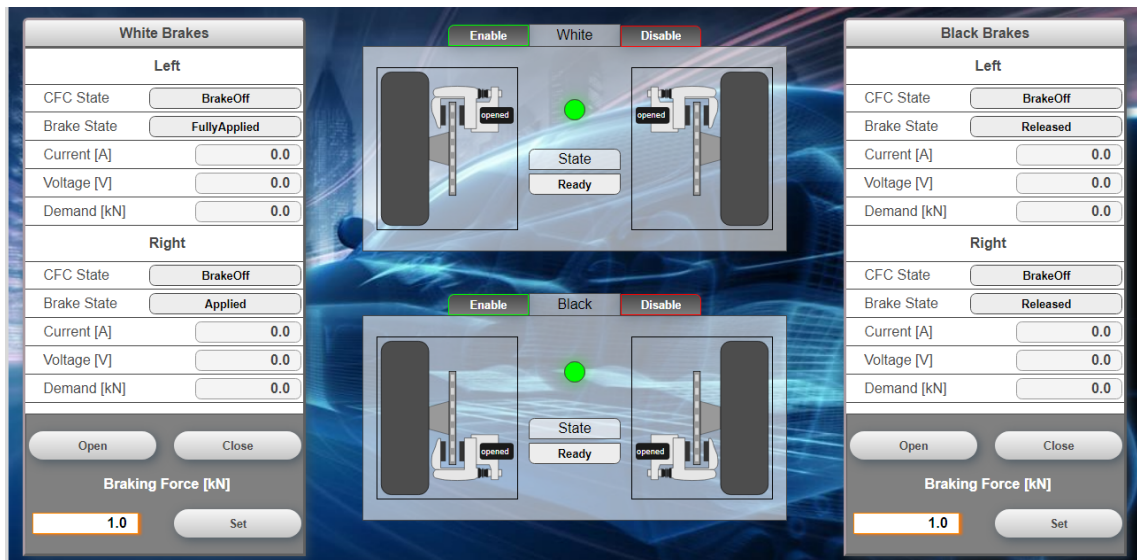
Obrázek A.1: Ukázka pracoviště (před instalací aktuátorů) (zdroj autor)



Obrázek A.2: Ukázka pracoviště (po instalaci aktuátorů) (zdroj autor)



Obrázek A.3: Zobrazení vypnutého ovládání z vizualizace (zdroj autor)



Obrázek A.4: Zobrazení odbrzděného stavu ve vizualizaci (zdroj autor)



Obrázek A.5: Zobrazení zasunutého pohonu LA36 ve vizualizaci (zdroj autor)



Obrázek A.6: Zobrazení vysunutého pohonu LA36 ve vizualizaci (zdroj autor)



Obrázek A.7: Ovládací panel servisního technika (zdroj autor)