

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 10: Elektrotechnika, elektronika a telekomunikace

Volant, řídicí jednotka, spoiler a doprovodná aplikace pro elektroformuli AF-23

**Patrik Švoma, Jakub Aldorf
Kraj Vysočina**

Žďár nad Sázavou, 2025

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 10: Elektrotechnika, elektronika a telekomunikace

Volant, řídící jednotka, spoiler a doprovodná aplikace pro elektroformuli AF-23

Steering wheel, control unit, spoiler and a mobile app for the AF-23 electric vehicle

Autoři: Patrik Švoma, Jakub Aldorf

Škola: SPŠ Žďár nad Sázavou, Studentská 1, 591 01 Žďár nad Sázavou

Kraj: Kraj Vysočina

Konzultant: Ing. Jaroslav Příbramský

Žďár nad Sázavou, 2025

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval samostatně a použil jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.

Nemám závažný důvod proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších předpisů.

Ve Žďáře nad Sázavou dne 8.3.2025

Patrik Švoma,

Jakub Aldorf

Poděkování

Chtěli bychom poděkovat učitelům Jaroslavovi Příbramskému, Petrovi Šuhajovi a Alešovi Gregorovi za pomoc a dohled při práci ve školních dílnách.

Dále bychom chtěli poděkovat Tomášovi Jeřábkoví, Tomášovi Smolíkovi, Ondřeji Pancnerovi a Janovi Topinkovi za vytvoření formule.

Anotace

Tato práce se zabývá výrobou volantu, řídicí jednotky a spoileru pro elektroformuli AF-23. Dále se zabývá vývojem doprovodné mobilní aplikace pro práci s daty. Pro výrobu těchto komponent byly využity principy 3D tisku, návrhu a výroby plošných spojů a programování mikrokontrolerů. Díky řídicí jednotce je možné v reálném čase sbírat data z veškerých systémů formule a zároveň je ovládat.

Klíčová slova

Elektrické vozidlo; volant; řídicí jednotka; mobilní aplikace; spoiler

Annotation

The aim of this project is to create a steering wheel, a control unit and a spoiler for the AF-23 electric vehicle. It also deals with the development of a companion app, which can be used to look at data. The principles of 3D printing, design and production of printed circuit boards and microcontrollers were used to produce these components. Thanks to the control unit, it is possible to collect data from all systems of the formula in real time and control them.

Keywords

Electric vehicle; steering wheel; control unit; mobile app; spoiler

1	Úvod	8
2	Volant.....	9
2.1	Elektronika	9
2.1.1	Raspberry Pi Pico W	10
2.1.2	Displej.....	11
2.1.3	Tlačítka	11
2.1.4	Rotační enkodér	12
2.1.5	LED	13
2.1.6	Posuvné registry	14
2.1.7	Napájení a komunikace s řídicí jednotkou	15
2.1.8	Plošné spoje	16
2.2	Firmware	18
2.2.1	Architektura firmwaru.....	18
2.2.2	Menu.....	20
2.2.3	Dashboard.....	21
2.2.4	Monitorování baterií.....	23
2.2.5	Stopky.....	24
2.2.6	Monitorování teplot.....	25
2.2.7	Jízdní statistiky.....	26
2.2.8	Hra.....	27
2.3	Konstrukce.....	28
2.3.1	Plošný spoj.....	29
2.3.2	Záda volantu	30
2.3.3	Výztuha.....	31
2.3.4	Přední kryt	32
2.3.5	Držadla	33
2.3.6	Příruba pro rychloupínací náboj.....	34
2.3.7	Rychloupínací konektor.....	35
3	Řídicí jednotka	36
3.1	Elektronika	36
3.1.1	Napájení.....	37
3.1.2	Raspberry Pi Pico W	39
3.1.3	Komunikace s volantem	39

3.1.4	Komunikace s jednotkou motoru	39
3.1.5	Senzor tlaku brzd	43
3.1.6	Komunikace s BMS	44
3.1.7	Teplotní senzory.....	45
3.1.8	Flash pamět.....	46
3.1.9	Hodiny reálného času	47
3.1.10	Gyroskop a akcelerometr.....	48
3.1.11	LED pásek na spoileru	48
3.1.12	Ventilátor, rezervní vstupy a výstupy	49
3.1.13	Plošný spoj.....	51
3.2	Firmware	52
3.2.1	OTA aktualizace firmwaru	52
3.2.2	Ovládání motoru	53
3.2.3	Komunikace s BMS	54
3.2.4	Čtení hodnot ze senzoru tlaku brzd.....	55
3.2.5	Čtení teplotních senzorů.....	55
3.2.6	Flash pamět.....	55
3.2.7	RTC	56
3.2.8	Gyroskop	56
3.2.9	LED spoileru.....	56
3.2.10	Rezervní vstupy a výstupy.....	56
3.3	Pouzdro	57
4	Mobilní aplikace.....	59
4.1	Komunikace s jednotkou.....	59
4.2	Hlavní obrazovka.....	60
4.3	Konfigurace LED spoileru	62
4.4	Přehled přechozích jízd.....	63
4.5	Aktualizace řídicí jednotky	65
5	Spoiler.....	68
6	Výroba a osazování plošných spojů	70
6.1	Návrh a testování zapojení	70
6.2	Testovací software	72
6.3	Výroba finálních plošných spojů.....	73

7	Výroba a sestavení tištěných součástí.....	75
7.1	Volant.....	75
7.2	Řídící jednotka.....	77
7.3	Spoiler	78
	Závěr.....	79
	Použitá literatura	80
	Seznam obrázků	81
	Seznam rovnic.....	83
	Seznam tabulek	83
	Seznam příloh	83

1 ÚVOD

Naše práce se zabývá vylepšením elektroformule AF-23. Jedná se o elektrické vozidlo poháněné elektromotorem o výkonu 5 kW, kterému dodává energii šestnácti článkový akumulátor založený na technologii LiFePo₄ s celkovou kapacitou 164 Ah. Akumulátor je zapojen v sériovém zapojení a jeho stav hlídá systém BMS (battery management system). Rám vozidla je vytvořen ze svařených ocelových profilů obdélníkového průřezu. Kapotáž je vyrobena ze sklolaminátu a je vyztužena ocelí. Pro převod síly z motoru do zadních kol slouží diferenciál a řemenice, které jsou zároveň připojeny k hydraulickému brzdovému systému. Vozidlo vzniklo jako maturitní projekt čtyř bývalých studentů školy SPŠ Žďár nad Sázavou.

V současném stavu je formule ovládána pomocí dvou přepínačů, které ovlivňují směr jízdy dopředu/dozadu a nízkou/vysokou rychlost. Dále je k dispozici startovací tlačítko. Není jakákoliv možnost čtení dat o jízdě jako například rychlost, teploty komponent nebo otáčky motoru, s výjimkou stavu baterií, jelikož BMS podporuje čtení stavu pomocí zařízení připojeného přes Bluetooth.

Hlavním cílem naší práce bylo zlepšení možností ovládání formule a zajištění většího počtu dat o jízdě. Primárním prostředkem k ovládání a zobrazování dat je volant. Pro připojení k elektronice slouží řídící jednotka, jejíž provoz je řízen mikrokontrolerem a která zároveň disponuje možností připojení přes Bluetooth. Pro využití tohoto připojení jsme vyvinuli doprovodnou mobilní aplikaci, ve které lze taktéž zobrazovat údaje.

V poslední řadě jsme přepracovali design spoileru a přidali možnost zobrazovat světelné efekty signalizující například brždění nebo couvání.

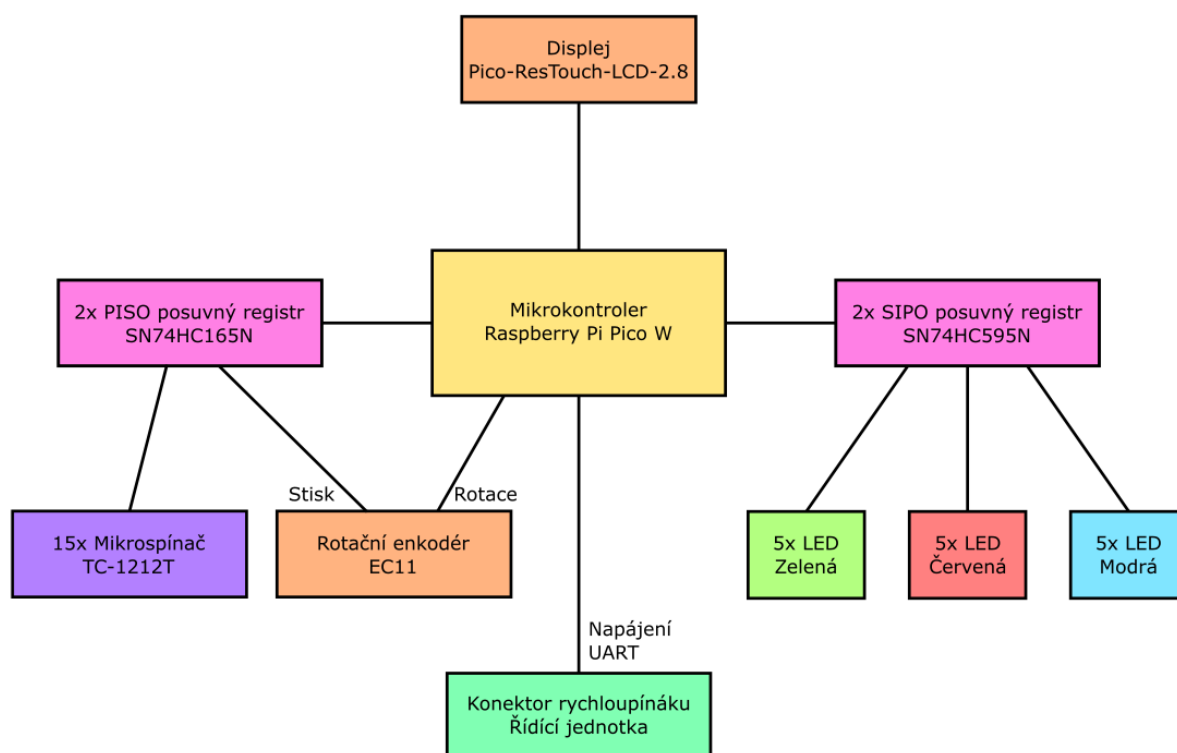
2 VOLANT

Nejviditelnější částí práce je volant, který slouží k ovládání veškerých funkcí formule. Zároveň zobrazuje informace o stavu vozidla, jako například rychlost, údaje z teplotních senzorů nebo kapacitu baterie.

2.1 Elektronika

Z hlediska elektroniky volant slouží pouze jako „ovladač“ řídicí jednotky. Sám o sobě nekomunikuje s ničím jiným než s řídicí jednotkou. Pro zobrazování informací a ovládání na volantu najdeme:

- 2,8palcový dotykový displej s rozlišením 320x240
- 15 LED (5x zelená, 5x červená, 5x modrá)
- 15 tlačítek
- Inkrementální rotační enkodér s tlačítkem



Obrázek 1 - Blokové schéma elektroniky volantu

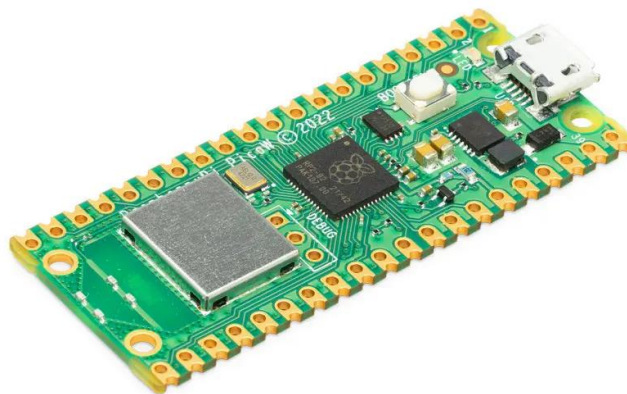
2.1.1 Raspberry Pi Pico W

Srdcem volantu je mikrokontroler **Raspberry Pi Pico W**, který zpracovává vstupy, ovládá displej a LED diody a komunikuje s řídicí jednotkou. Primárním důvodem, proč jsme zvolili tento mikrokontroler je velké množství pinů a rozhraní, které jsou k dispozici. Tento mikrokontroler jsme také používali v předchozích projektech, byli jsme s ním tedy dobře seznámeni. Původním záměrem bylo použít variantu bez bezdrátové konektivity, kvůli podpoře OTA aktualizací jsme se ale nakonec rozhodli pro tuto vylepšenou variantu mikrokontroleru.

Mikrokontroler využívá dvoujádrového procesoru RP2040, který běží na frekvenci 133 MHz a je doplněn 264 KB SRAM a 2 MB flash pamětí. Lze využít 26 GPIO pinů, na každém z nich je k dispozici jeden ze 16 PWM kanálů. 3 piny lze také využít jako 12bitový analogový převodník. Velkou výhodou je také všestrannost z hlediska komunikačních rozhraní. Kromě USB portu využívaného k programování, případně komunikaci jiným zařízením, se na Raspberry Pi Pico W nachází:

- 2x hardwarové SPI
- 2x hardwarové I²C
- 2x hardwarový UART
- 8x PIO (Programmable input output)
- Serial Wire Debug
- Wi-Fi 4 a Bluetooth 5.2

Komunikační rozhraní nejsou vázána pouze na jednu sadu pinů, pro každý pin komunikačního rozhraní je k dispozici alespoň jeden další alternativní pin. [1]



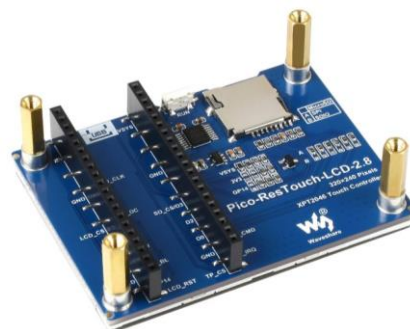
Obrázek 2 - Mikrokontroler Raspberry Pi Pico W. *Raspberry Pi Pico W*. Online. In: RPishop.cz. C2025. Dostupné z: <https://rpishop.cz/raspberry-pi-pico/5073-raspberry-pi-pico-w.html>. [cit. 2025-03-03].

2.1.2 Displej

Pro zobrazování informací řidičovi jsme zvolili modul displeje **Pico-ResTouch-LCD-2.8** od společnosti Waveshare. Jedná se o 2,8palcový dotykový IPS displej určený přímo pro použití s mikrokontrolerem Raspberry Pi Pico. Disponuje rozlišením 320x240 pixelů, 16bitovým nebo 18bitovým barevným rozlišením a komunikuje pomocí sběrnice SPI. [2] Na displeji se nachází také slot pro microSD kartu a resetovací tlačítko, v naší práci je ovšem nepoužíváme.



Obrázek 3 - Displej pro Raspberry Pi Pico. WAVESHARE. 2.8inch Touch Display Module for Raspberry Pi Pico, 262K Colors, 320×240, SPI. Online. Waveshare. C2025. Dostupné z: <https://www.waveshare.com/pico-restouch-lcd-2.8.htm>. [cit. 2025-02-24].



Obrázek 4 - Displej pro Raspberry Pi Pico. WAVESHARE. 2.8inch Touch Display Module for Raspberry Pi Pico, 262K Colors, 320×240, SPI. Online. Waveshare. C2025. Dostupné z: <https://www.waveshare.com/pico-restouch-lcd-2.8.htm>. [cit. 2025-02-24].

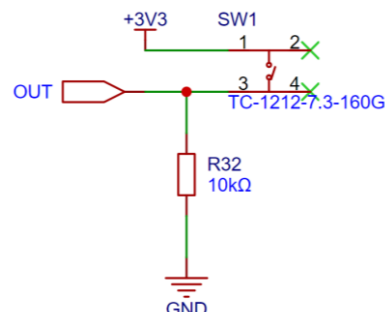
Na modulu displeje jsou čtyři otvory pro uchycení se závitem M2,5. Podsvícení displeje je nastavitelné za pomoci PWM signálu. Spotřeba displeje je závislá na zobrazeném obsahu, neměla by ale přesáhnout 140 mA. [2]

2.1.3 Tlačítka

Pro ovládání funkcí volantu jsme zvolili 15 běžných mikrospínačů **TC-1212T**. Na nich jsou nasazeny barevné kloboučky, které jsou barevně rozřazeny podle funkcí tlačítek.



Obrázek 5 - Mikrospínač TC-1212T. DRÁTEK.CZ. Mikrospínač TC-1212T - 12 x 12 x 7,3 mm SMD PCB DPS. Online. Drátek.cz. Dostupné z: <https://dratek.cz/arduino/1180-mikrospinač-tc-1212t-12x12x7.3mm-smd-pcb-dps.html>. [cit. 2025-02-24].

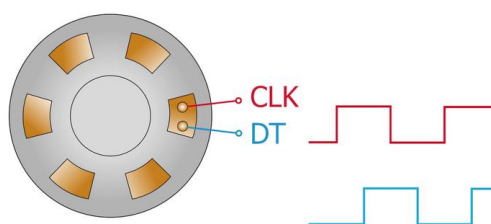


Obrázek 6 - Zapojení mikrospínače

2.1.4 Rotační enkodér

Dalším ovládacím prvkem na volantu je inkrementální rotační enkodér, využívaný vedlejší funkce, jako například pohyb v menu. Konkrétně se jedná o běžně používaný enkodér z rodiny **EC11**. Jedná se o variantu se 30 zarážkami a tlačítkem, které na volantu slouží jako 16. spínač.

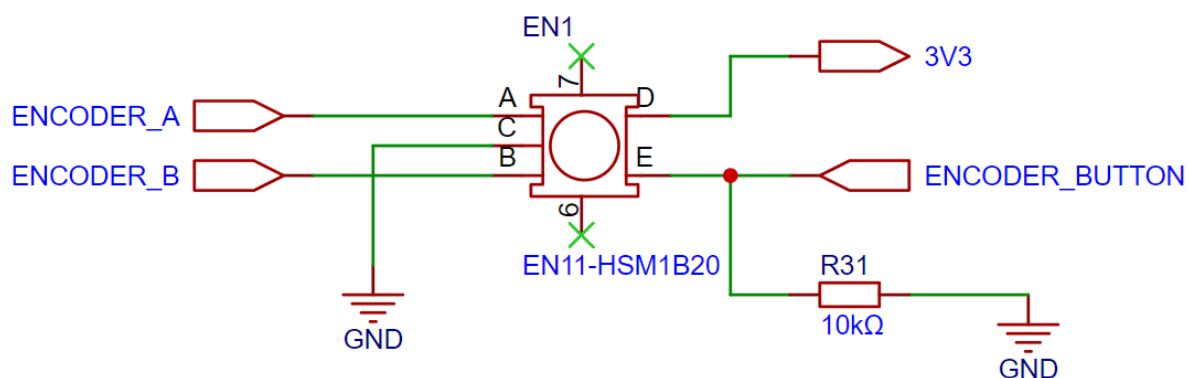
Zapojení tlačítka je stejné jako u ostatních spínačů. Enkodér má výstupy A a B, které se při otáčení spojují a odpojují od napájecího napětí. Vznikají tak 4 různé stavy, které může výstup enkodéru mít, podle kterých je možné určit směr rotace.



Obrázek 7 - Schéma funkce inkrementálního enkodéru. NEMATIC!. *How Rotary Encoder Works With Arduino!*. Online. Instructables. Dostupné z: <https://www.instructables.com/How-to-Use-an-Rotary-Encoder-With-Arduino/>. [cit. 2025-02-24].



Obrázek 8 - Rotační enkodér EC11. EC11E15204A3. Online. In: Alps Alpine. C2025. Dostupné z: <https://tech.alpsalpine.com/e/products/detail/EC11E15204A3/>. [cit. 2025-02-24].



Obrázek 9 - Zapojení rotačního enkodéru

2.1.5 LED

Nad displejem je umístěna řada 15 LED, které se používají jako další indikátor rychlosti. V reálných formulích se používají pro indikaci otáček motoru, jelikož naše elektroformule převodovku nemá, je počet otáček za minutu vždy přímo úměrný rychlosti. LED jsou rozděleny na tři barvy, zelenou, červenou a modrou.

Do série s každou diodou je třeba zapojit rezistor, jehož hodnota se počítá pomocí Ohmova zákona. Pro přesnější kontrolu nad jasnem a proudem jsou u každé LED místo jednoho rezistoru umístěny dva rezistory v paralelním zapojení. Toto zapojení zároveň zlepší možnosti zpětné úpravy hodnot rezistor bez změny samotného plošného spoje.

$$R = \frac{U}{I}$$

Rovnice 1 - Výpočet hodnoty rezistoru

Pro výpočet celkového odporu paralelně zapojených rezistorů platí:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$$

Rovnice 2 - Výpočet celkové hodnoty dvou paralelně zapojených rezistorů

U všech tří barev bylo cílem dosáhnout proudu přibližně 7 mA, aby svítily jasně, ale zároveň spotřeba LED nepřesahovala limity posuvného registru, jež je ovládá. Napětí zdroje je 3,3 V.

Barva	Napětí v propustném směru [V]	Hodnoty zvolených rezistorů [Ω] (Celkem)	Proud [mA]
Zelená	1,88	220, 2 200 (200)	7,1
Červená	1,79	220, 10 000 (215,26)	7,01
Modrá	2,61	100, 10 000 (99,01)	6,97

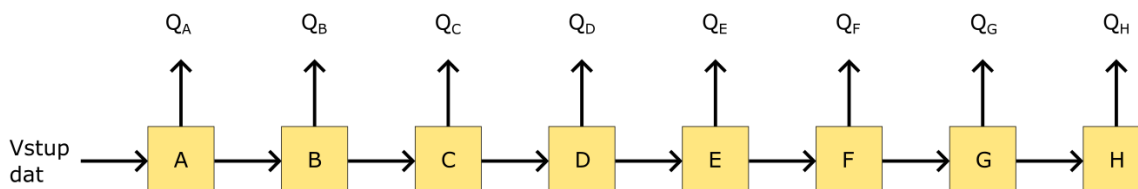
Tabulka 1 - Hodnoty rezistorů pro LED na volantu

2.1.6 Posuvné registry

Pro ovládání LED diod jsou použity tzv. **SIPO** (serial-in, parallel-out) posuvné registry, konkrétně SN74HC595N. Díky těmto registrům lze ovládat velké množství digitálních výstupů pomocí malého množství pinů. Pro ovládání SIPO registru se používají pouze 4 piny:

- SER – vstup pro data
- OE – zapnutí/vypnutí všech výstupů (pin je invertovaný)
- RCLK – tzv. latch pin, při každé změně na hodnotu HIGH se projeví změny dat
- SRCLK – tzv. clock pin, při každé změně na hodnotu HIGH se výstupy registrů posunou o jedno místo (změna se projeví až po sepnutí pinu RCLK)

Na jednom posuvném registru je k dispozici 8 výstupů. Pokud jich je potřeba více, dají se registry připojit za sebe, aniž by bylo potřeba využít více pinů na mikrokontroleru. Ve volantu jsou použity 2 registry pro ovládání 15 LED.

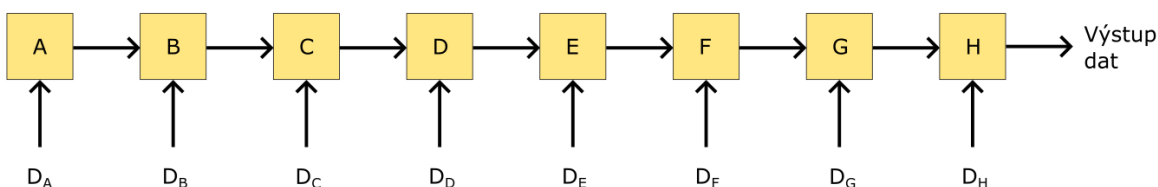


Obrázek 10 - Schéma fungování SIPO posuvného registru

Pro čtení stavu jednotlivých tlačítek jsou použity tzv. **PISO** (parallel-in, serial-out) registry, konkrétně SN74HC165N. S jejich pomocí lze číst stav všech 16 tlačítek s využitím pouze 3 pinů: Každý registr má 8 vstupů, stejně jako u SIPO registrů jich lze připojit více za sebou.

- PL – načtení dat z výstupů do registru
- CP – tzv. clock pin, posouvá hodnoty uvnitř registru o jedno místo
- QH – výstup dat

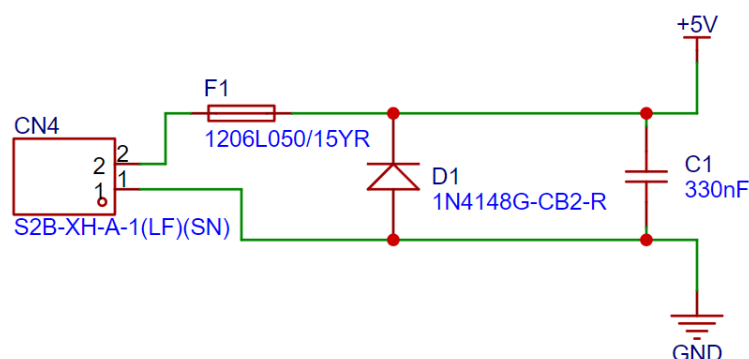
Nevýhodou tohoto řešení je, že pro detekci stisku nelze používat přerušování, je vždy potřeba přečíst stav všech tlačítek.



Obrázek 11 - Schéma fungování PISO posuvného registru

2.1.7 Napájení a komunikace s řídicí jednotkou

Volant je napájen 5 V, které poskytuje řídicí jednotka. Napájecí vstup je chráněn proti přepólování pomocí diody v paralelním zapojení. [3] Pokud je zdroj napájení připojen opačně, proud poteče skrz diodu do vratné pojistky, která obvod rozpojí. Vratná pojistka dimenzovaná na trvalý proud 500 mA zároveň chrání obvod proti zkratu.

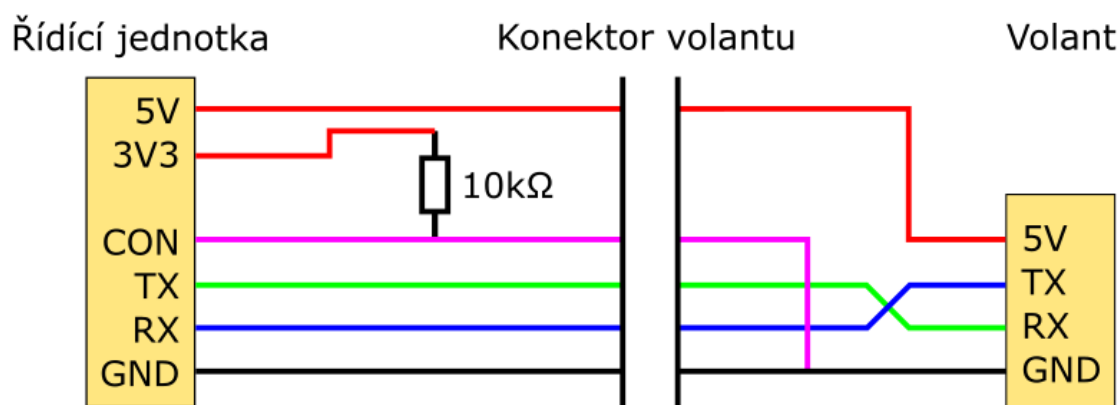


Obrázek 12 - Zapojení ochrany proti přepólování a zkratu na volantu

Protože mikrokontroler Raspberry Pi Pico pracuje s logickým napětím 3,3 V, komponenty jsou napájeny ze step-down měniče integrovaného na mikrokontroleru. Maximální zatížení na volantu by nemělo přesáhnout 150 mA, vzniká tak dostatečná rezerva k limitu 300 mA doporučenému výrobcem. [4]

Pro komunikaci s řídicí jednotkou volant využívá hardwarového UART rozhraní dostupného na mikrokontroleru. Pro tuto oboustrannou komunikaci jsou zapotřebí pouze dva další vodiče.

Další vodič se používá pro detekci připojeného volantu. Na straně jednotky se nachází pull up rezistor, po připojení volantu se vodič sepne se zemí, a změní se hodnota na vstupu.



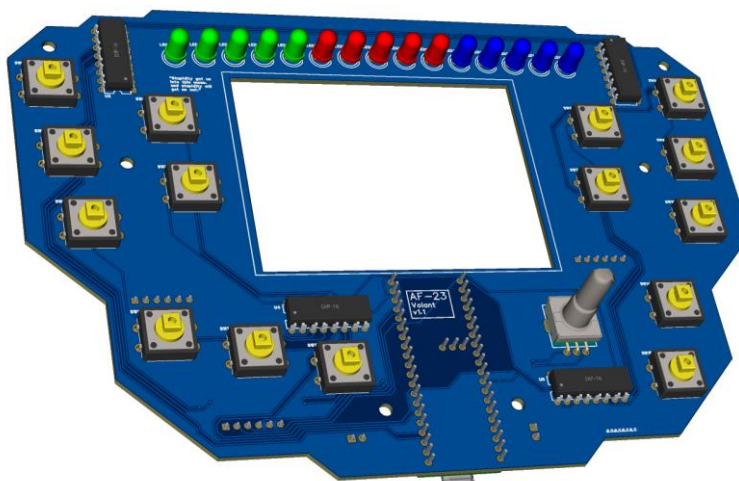
Obrázek 13 - Schéma připojení volantu k řídicí jednotce

2.1.8 Plošné spoje

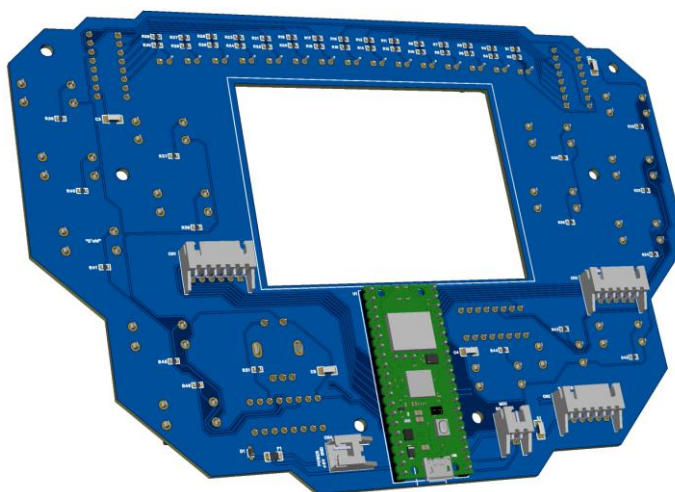
Pro spojení všech součástek jsme navrhli **dva** plošné spoje. Na hlavním plošném spoji neboli základní desce je umístěn mikrokontroler a všechny ostatní komponenty s výjimkou displeje. Ten je umístěn na samostatném plošném spoji. Oba plošné spoje jsou propojeny třemi 6 pinovými konektory řady JST-XH.

Toto řešení jsme zvolili, protože displej s připojovacími lištami je tlustý přes 14 mm. Díky oddělení plošného spoje tak displej může být v rovině se zbytkem volantu.

Na přední straně hlavního plošného spoje se nachází vstupní a výstupní periferie společně s posuvnými registry. Na zadní straně je umístěn mikrokontroler Raspberry Pi Pico, konektory a SMD součástky. Uprostřed dvouvrstvého plošného spoje je velký otvor, skrz který je umístěn displej. K upevnění je na plošném spoji 6 otvorů s průměrem 3 mm. Rozměry plošného spoje jsou **187,96 mm** na šířku a **132,08 mm** na výšku.

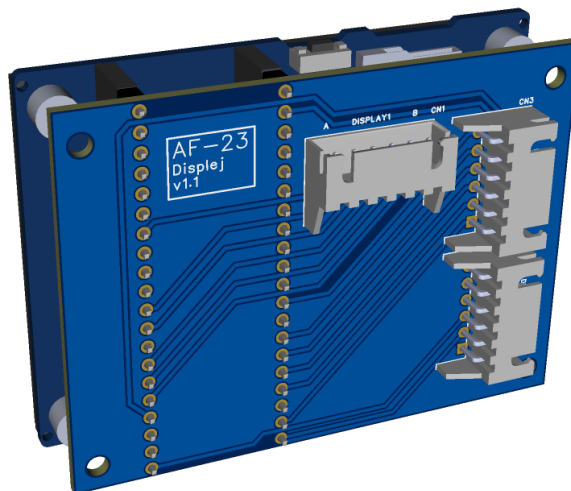


Obrázek 14 - 3D náhled přední strany hlavního plošného spoje volantu



Obrázek 15 - 3D náhled zadní strany hlavního plošného spoje volantu.

Druhý plošný spoj je určen čistě pro upevnění displeje, kromě konektorů a jednoho kondenzátoru se na něm tak nic dalšího nenachází. Pro montáž jsou určeny čtyři otvory s průměrem 2,8 mm, které mají stejnou rozteč jako závity na modulu displeje. Šířka plošného spoje je **68,58 mm** a výška **50,8 mm**.



Obrázek 16 - 3D náhled plošného spoje pro displej volantu

Návrh rozložení plošných spojů byl zdlouhavý zejména kvůli nutnosti návrhu zároveň s CAD modelem celého volantu. Cesty na plošných spojích jsou navrženy tak, aby jejich šířka odpovídala požadovaným proudům, a aby se příliš nepřibližovaly k dalším cestám. Po celých plošných spojích je také využito rozlití mědi pro zem, jak je u plošných spojů zvykem. Kolem montážních otvorů jsou dodrženy větší mezery od cest a komponent, aby se zamezilo případnému doteku hlavy šroubu a cesty nebo součástky.

Pomocí bílého potisku jsou na obou plošných spojích vyznačeny značky součástek, jejich obrysy, orientace pro pájení a u konektorů také označení stran „A“ a „B“ pro kontrolu zapojení.

2.2 Firmware

Firmware volantu jsme se snažili navrhnout jako spolehlivý a rozšiřitelný systém, který by dokázal efektivně komunikovat s řídicí jednotkou a současně zobrazovat důležité informace na displeji. Cílem bylo umožnit jezdcí přehled o klíčových hodnot jízdy a umožnit mu ovládat jednotlivé funkce formule.

Firmware je napsán v jazyce C++ a využívá framework arduino-pico, který poskytuje kompatibilní API s Arduino frameworkem, a to umožňuje využití existujících knihoven pro Arduino. Tento framework byl především zvolen z důvodu, že kvalitní knihovny pro nativní SDK často chybí.

2.2.1 Architektura firmwaru

Architektura firmwaru je navržena plně modulárně, kde jednotlivé komponenty fungují nezávisle na sobě. Základní architektura se skládá ze tří částí:

- Aplikace
- Ovladače hardwaru
- Hlavní smyčka

Funkčnost volantu je rozdělena do samostatných aplikací, z nichž každá se stará o konkrétní úlohu, například zobrazování stavu baterií, časomíru nebo menu.

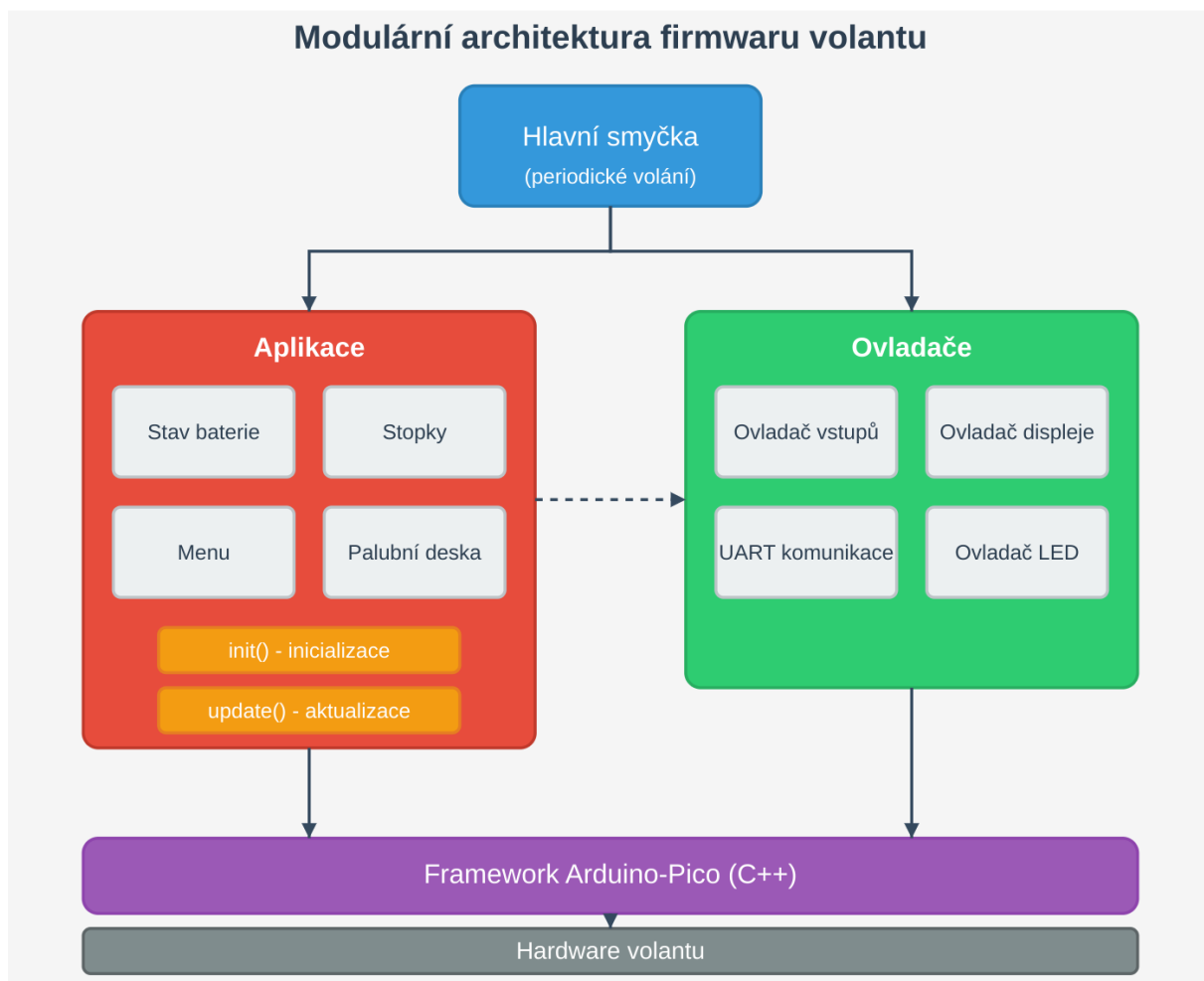
Každá aplikace je navržena tak, aby obsahovala dvě hlavní funkce:

- **Inicializační funkci**, která se spustí při načtení aplikace a provede potřebná nastavení.
- **Aktualizační funkci**, která je pravidelně volána v hlavní smyčce a zajišťuje průběžnou aktualizaci aplikace.

Díky této struktuře lze snadno přidávat nové funkce bez zásahu do hlavního kódu systému.

Další částí jsou ovladače, které jsou využívány pro různé hardwarové komponenty. Každý ovladač má jasně určené, kterou část hardwaru ovládá. U ovladačů již není požadováno striktně dodržovat strukturu, protože každý z nich má specifické požadavky na běh. Mezi nejdůležitější ovladače patří ovladač pro detekci stisku tlačítek (InputHandler), pro ovládání displeje (DisplayController) a také například ovladač pro komunikaci s jednotkou motoru.

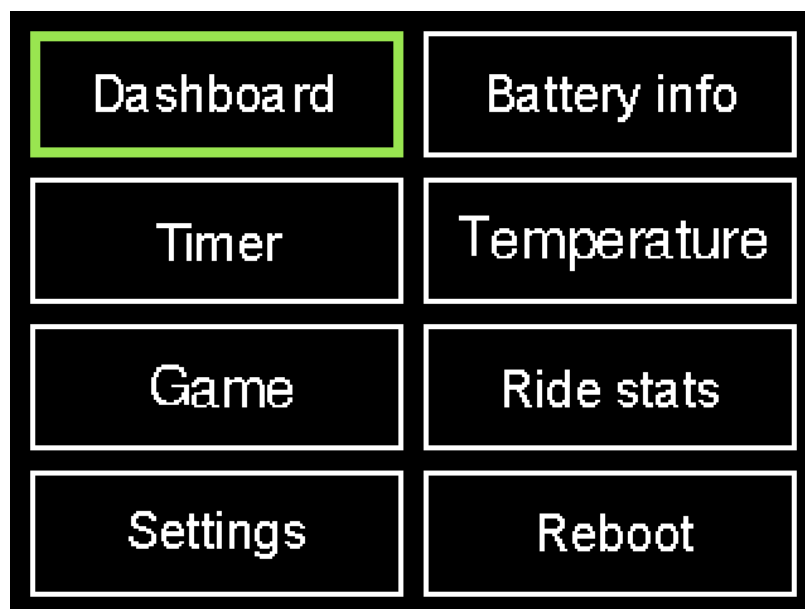
Všechno spojuje hlavní smyčka, která zajišťuje periodické volání aktualizace aktivní aplikace a potřebných ovladačů. V budoucnu je v plánu rozdělit jednotlivé části mezi více jader, což by mělo ještě více snížit odezvu.



Obrázek 17 - Architektura firmwaru volantu

2.2.2 Menu

Menu slouží jako hlavní vstupní bod k jednotlivým aplikacím volantu a umožňuje jezdcovi rychlý a přehledný přístup k důležitým funkcím volantu. Menu je navrženo tak, aby bylo co nejjednodušší na ovládání, a zároveň bylo přehledné.



Obrázek 18 - Návrh menu volantu

Díky modulární architektuře je možné jednoduše přidávat aplikace do menu bez jakéhokoliv zásahu do jeho základního kódu. Každá aplikace zapsaná do seznamu se automaticky objeví v nabídce.

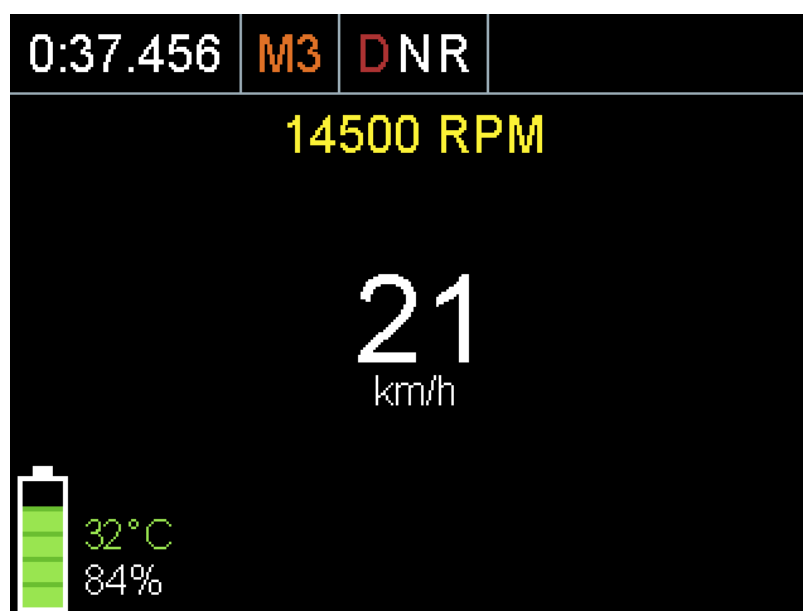
V menu se ovládáme pomocí rotačního enkodéru, kde otáčením se posouváme mezi jednotlivými položkami a jeho stisknutím potvrzuje výběr a otvíráme novou aplikaci.

2.2.3 Dashboard

Dashboard představuje hlavní aplikaci volantu, která poskytuje jezdcí všechny klíčové informace o jízdě v reálném čase. Tato aplikace navržena s důrazem na rychlou orientaci, čitelnost a efektivitu při jízdě.

Dashboard je rozdělen do několika sekcí, z nichž každá zobrazuje konkrétní informace:

1. Rychlost
2. Otáčky motoru
3. Teplotu motoru
4. Stav baterií
5. Aktivní rychlostní režim
6. Aktivní směr jízdy
7. Časovač – umožňuje sledovat čas kola



Obrázek 19 - Návrh hlavní obrazovky volantu

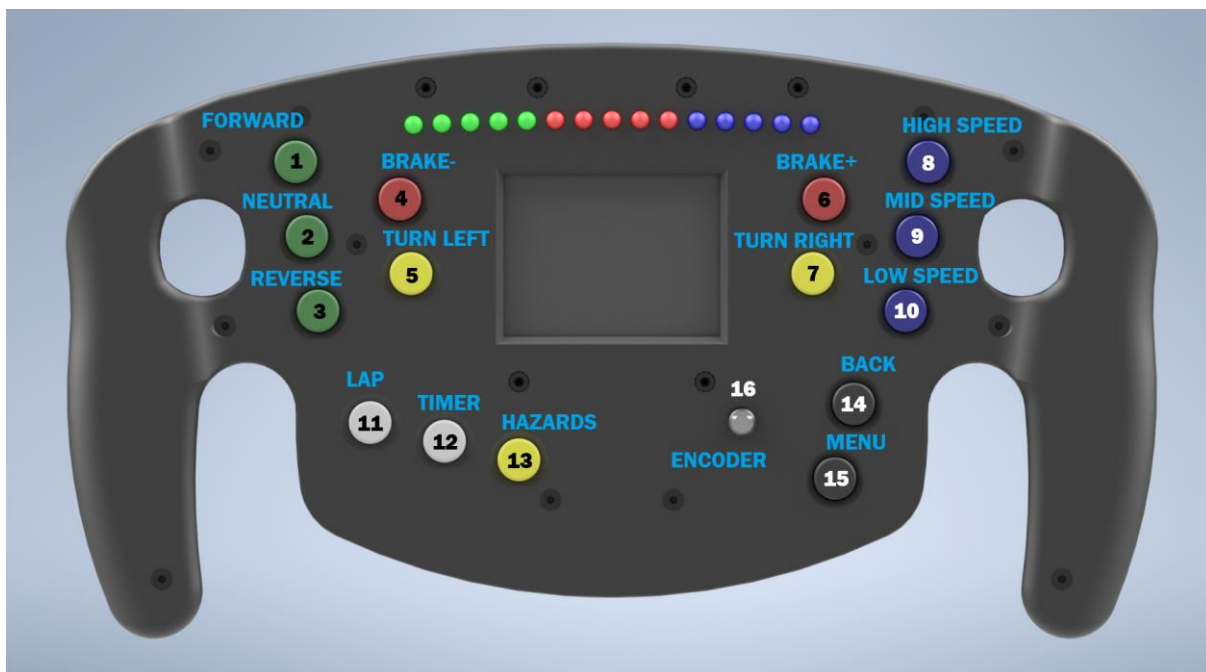
Dashboard je napojen na komunikační ovladač, který pravidelně přijímá data z řídicí jednotky přes sériovou linku a na základě nich se aktualizují zobrazené informace.

LED diody integrované ve volantu slouží jako další vizuální indikátor, který zobrazuje tyto možnosti:

- **Indikace rychlosti** – v závislosti na rychlosti se postupně rozvíjí jednotlivé LED diody.
- **Indikace brždění** – při prudkém brždění LED diody rychle zablikají.
- **Varování o nízké baterii** – pokud je stav baterií opravdu kritický, LED diody začnou blikat ve dvou vteřinovém intervalu.

Volant je vybaven 16 tlačítky, která umožňují jezdcovi snadno a rychle ovládat důležité funkce vozu, čímž zajišťují pohodlné a intuitivní ovládání během jízdy. Tato tlačítka slouží k ovládání následujících funkcí:

1. **Forward** – přepnutí směru jízdy dopředu
2. **Neutral** – vypne spínač pedálu a zamezí tak roztočení motoru
3. **Reverse** – přepnutí směru jízdy dozadu
4. **Brake-** - sníží účinek brždění motorem
5. **Turn left** – blink doleva na spoileru
6. **Brake+** - zvýší účinek brždění motorem
7. **Turn right** – blink doprava na spoileru
8. **High speed** – přepnutí na vysokou rychlost
9. **Mid speed** – přepnutí na střední rychlost
10. **Low speed** – přepnutí na nízkou rychlost
11. **Lap** – start nového kola na stopkách
12. **Timer** – spustí stopky
13. **Hazards** – výstražná světla na spoileru
14. **Back** – tlačítko zpět pro ovládání menu
15. **Menu** – vrátí se zpět do menu
16. **Encoder** – tlačítko enkodéru, slouží většinou jako OK



Obrázek 20 - Rozložení tlačítek volantu a jejich pořadová čísla

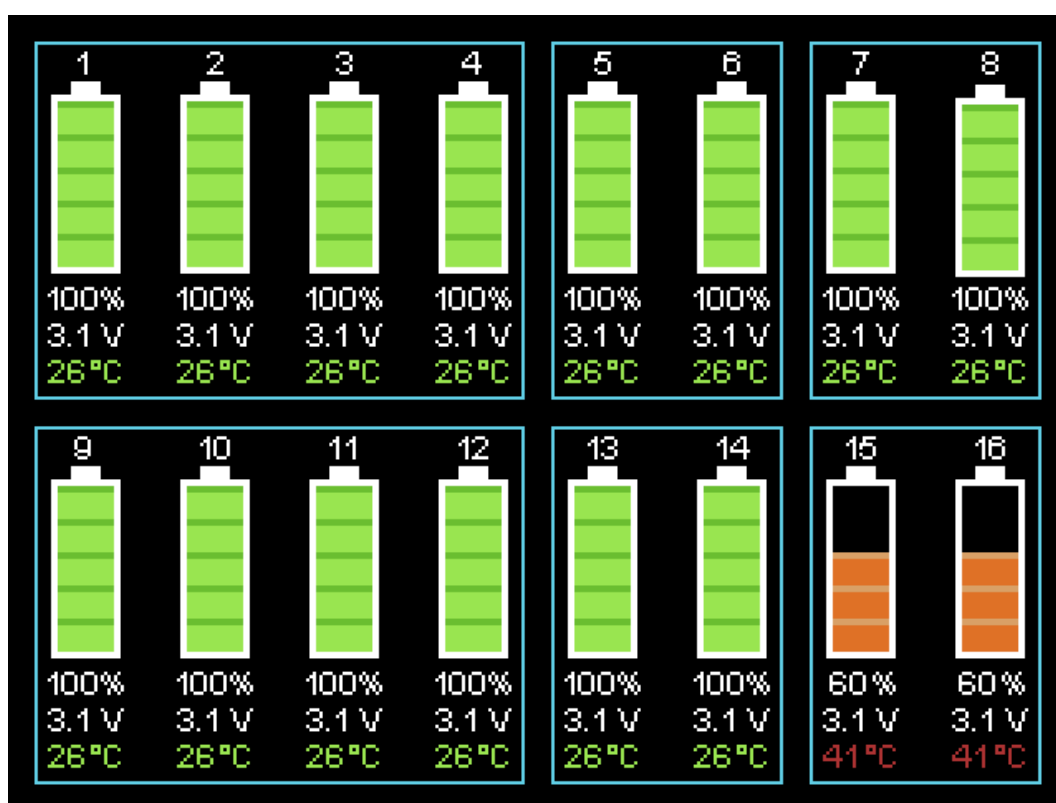
2.2.4 Monitorování baterií

Tato aplikace slouží k přehlednému zobrazení stavu baterií v reálném čase. Jezdec díky ní vidí detailní informace o jednotlivých článcích, což mu umožňuje sledovat, zda baterie fungují správně a jestli nehrozí nějaký problém.

Na obrazovce u každého článku vidíme tyto údaje:

- Úroveň napětí – Kolik procent zbývá v jednotlivém článku.
- Napětí článku – Pomáhá odhalit nerovnoměrné vybíjení
- Teplota článku – Důležitý údaj pro bezpečnost. Pokud by byl článek moc horký mohlo by to představovat riziko přehřátí nebo snížení životnosti baterie.

Data jsou neustále aktualizovaná z řídicí jednotky. Pokud některý z článků dosáhne kritické hodnoty, tak jeho ukazatel se přemění na červenou.



Obrázek 21 - Návrh obrazovky se stavem baterií

2.2.5 Stopky

Aplikace stopky umožňuje sledování a správu časů jednotlivých kol během jízdy. Zde si snadno můžeme měřit časy na kolo a porovnávat je.

Funkce stopky zahrnují:

- **Start / stop**
- **Zahájení nového kola** – Stiskem tlačítka se uloží čas a automaticky se spustí měření nového.
- **Reset** – Stopky lze vynulovat a začít měření od začátku.
- **Zobrazování přechozích časů** – Aplikace uchovává minulé kolo (Prev) a nejlepší dosažený čas (Best).



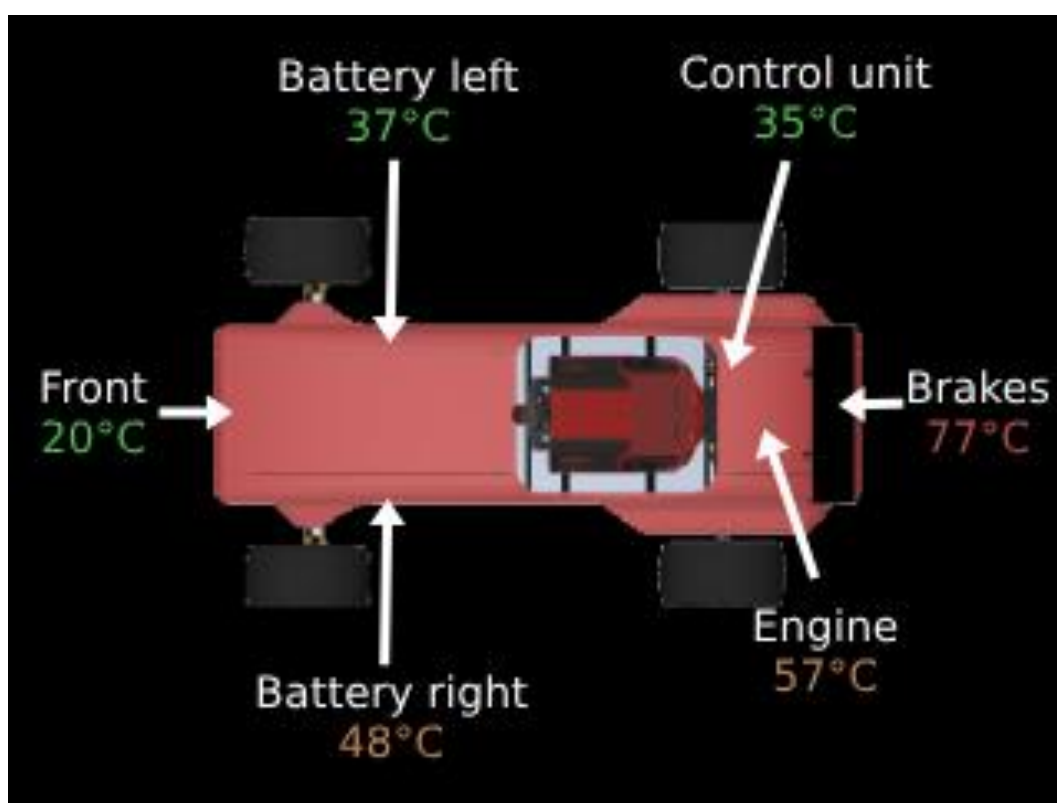
Obrázek 22 - Obrazovka s časovačem

2.2.6 Monitorování teplot

Aplikace pro monitorování teplot umožňuje sledovat teploty naměřené senzory rozmístěnými na různých místech formule. Cílem je zajistit, aby jezdec měl v reálném čase přehled o tepelném zatížení klíčových komponent, jako jsou motor, baterie, jednotka motoru nebo brzdy.

Po spuštění aplikace se na displeji zobrazí schéma formule, kde jsou jednotlivé senzory znázorněny pomocí šipek směřujících k jejich umístění. Nad každým senzorem je uvedena aktuální teplota doplněná o popisek odpovídající konkrétní součásti. Hodnoty jsou pravidelně aktualizovány.

Z technického hlediska aplikace při spuštění vyžádá aktuální teploty z řídicí jednotky, která zpracovává data ze senzorů a předává je volantu.



Obrázek 23 - Návrh obrazovky s teplotami komponent

2.2.7 Jízdní statistiky

Tato aplikace slouží ke zobrazení jednoduchých jízdních statistik, jako jsou maximální rychlost, maximální otáčky, maximální zrychlení rozděleno na úseky po 10 km/h, ujetá vzdálenost, celková ujetá vzdálenost. Více rozepsané statistiky o jízdě jsou rozepsány v aplikaci.

Tato aplikace poskytuje přehled základních statistik jízdy. Na displeji se zobrazují hodnoty, jako jsou:

- **Maximální rychlost** dosaženou během jízdy.
- **Maximální otáčky** motoru.
- **Nejvyšší naměřené zrychlení**, rozdělené do úseků po 10 km/h.
- **Ujetou vzdálenost** během aktuální jízdy.
- **Celkovou ujetou** vzdálenost ze všech jízd.

Pro detailnější analýzu jízdních dat slouží mobilní aplikace, kde se můžeme podívat na podrobnější statistiky jízd.



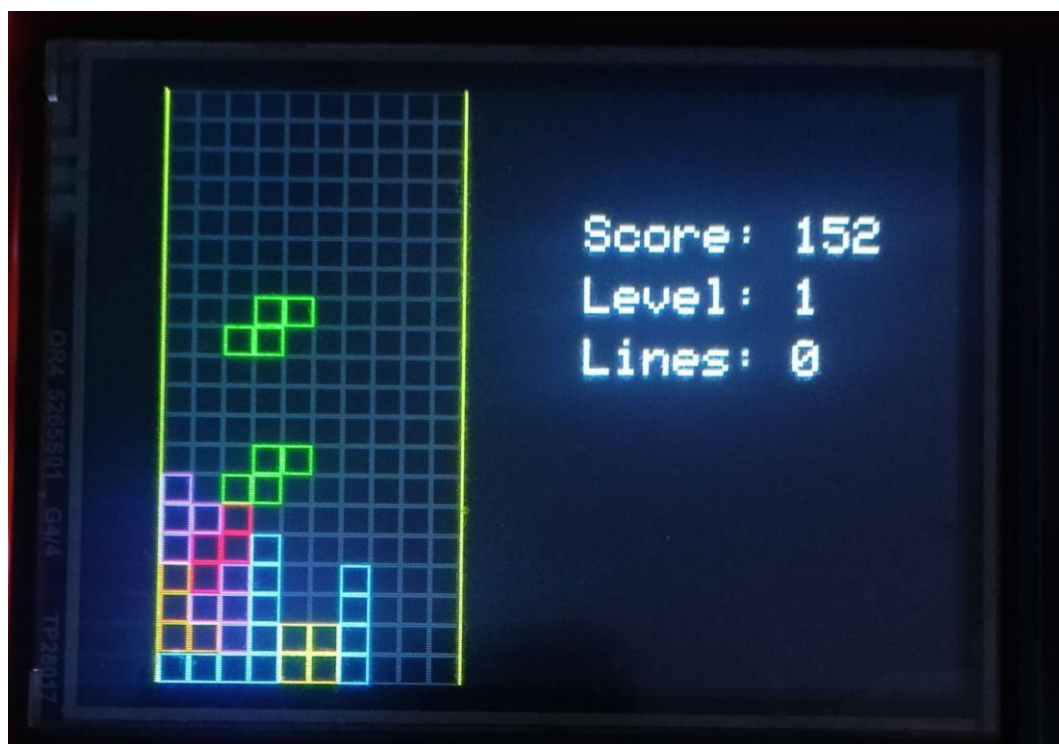
Obrázek 24 - Obrazovka s časy akcelerací a statistikami

2.2.8 Hra

Firmware volantu jsme neomezili jen na funkční aplikace, ale rozhodli jsme se přidat i jednoduchou hru inspirovanou známou hrou Tetris. Chtěli jsme tím demonstrovat možnosti displeje, ovládání i celkové rozšiřitelnosti systému.

Herní pole zabírá levou část obrazovky, kde postupně padají dílky složené ze čtyř čtverců. Hráč je může otáčet a posouvat pomocí tlačítek na volantu, aby je správně umístil a vyplnil řádky. Když se řádek kompletně zaplní, zmizí a hráč získává body. Na pravé straně obrazovky se zobrazuje aktuální skóre, počet zkompletovaných řádků a dosažená úroveň, která se zvyšuje s postupem hry.

Tato hra slouží nejen jako zábavná funkce, ale hlavně jako ukázka toho, jak lze na volantu programovat různé typy aplikací, včetně interaktivních a graficky náročnějších prvků.



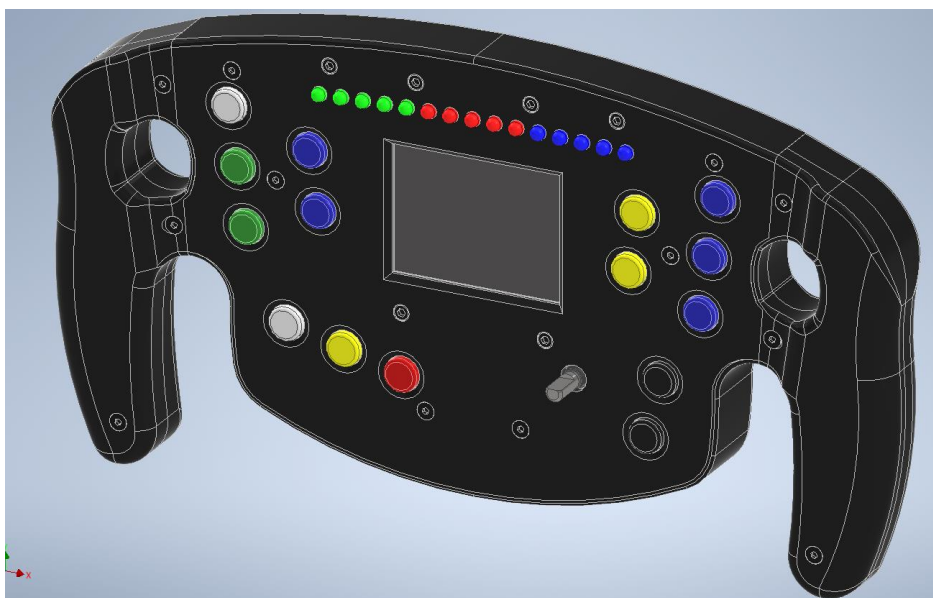
Obrázek 25 - Hra na volantu pro jeho demonstraci

2.3 Konstrukce

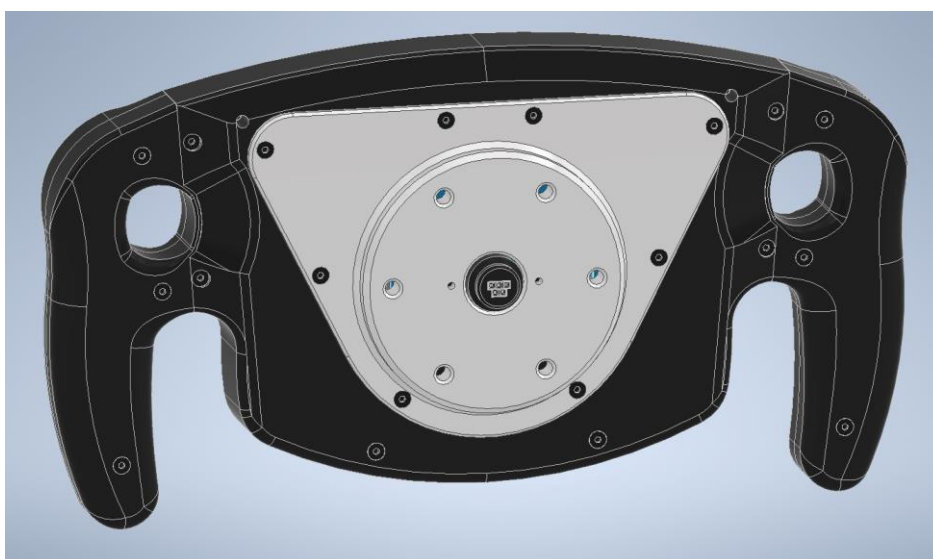
Většina volantu byla vyrobena pomocí technologie 3D tisku, nejprve bylo ovšem potřeba samotnou konstrukci navrhnout. Návrh probíhal v CAD softwaru Autodesk Inventor 2024. Pro převod dat na instrukce pro 3D tiskárnu byl poté použit PrusaSlicer.

Některé části volantu byly vyrobeny z kovu, aby došlo ke zpevnění celé konstrukce, která by ovšem i bez nich byla velmi robustní.

Celková šířka volantu dosahuje 293,6 mm, výška 160 mm a tloušťka bez rychloupínacího náboje 45,5 mm. Váží 1500 g a na jeho výrobu bylo přes 40 závitových vložek a šroubů.



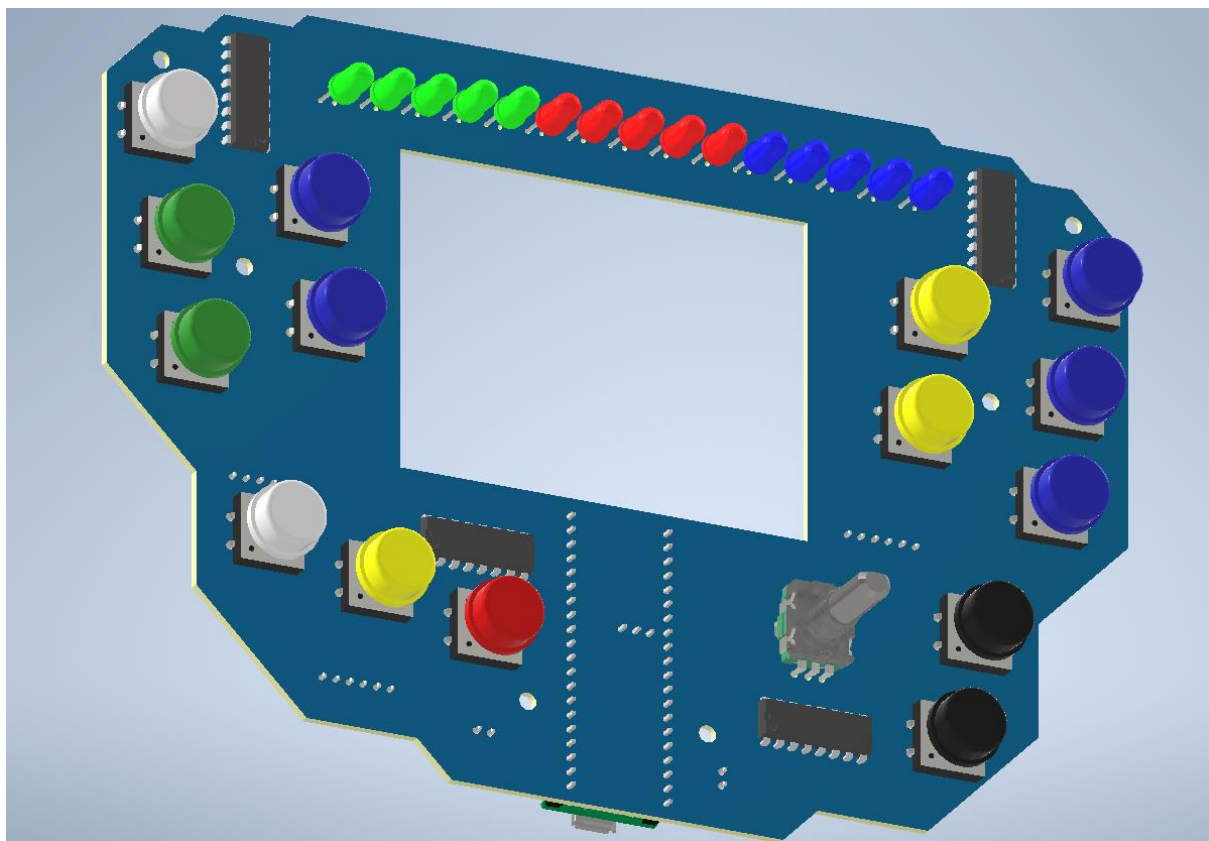
Obrázek 26 - Přední pohled na volant



Obrázek 27 - Zadní pohled na volant

2.3.1 Plošný spoj

Abychom mohli lépe navrhovat konstrukci kolem samotných plošných spojů, bylo nejprve potřeba vytvořit jejich přesné modely. Nejprve jsme tedy vyexportovali STEP objekty neosazených plošných spojů ze softwaru EasyEda Professional, převedli jsme je na nativní formát Inventoru, a nakonec k nim přidali jednotlivé součástky. Pro některé z těchto součástek, jako například displej, byl jejich model k dispozici od výrobců, ostatní jsme museli vymodelovat sami podle výkresů či měření.



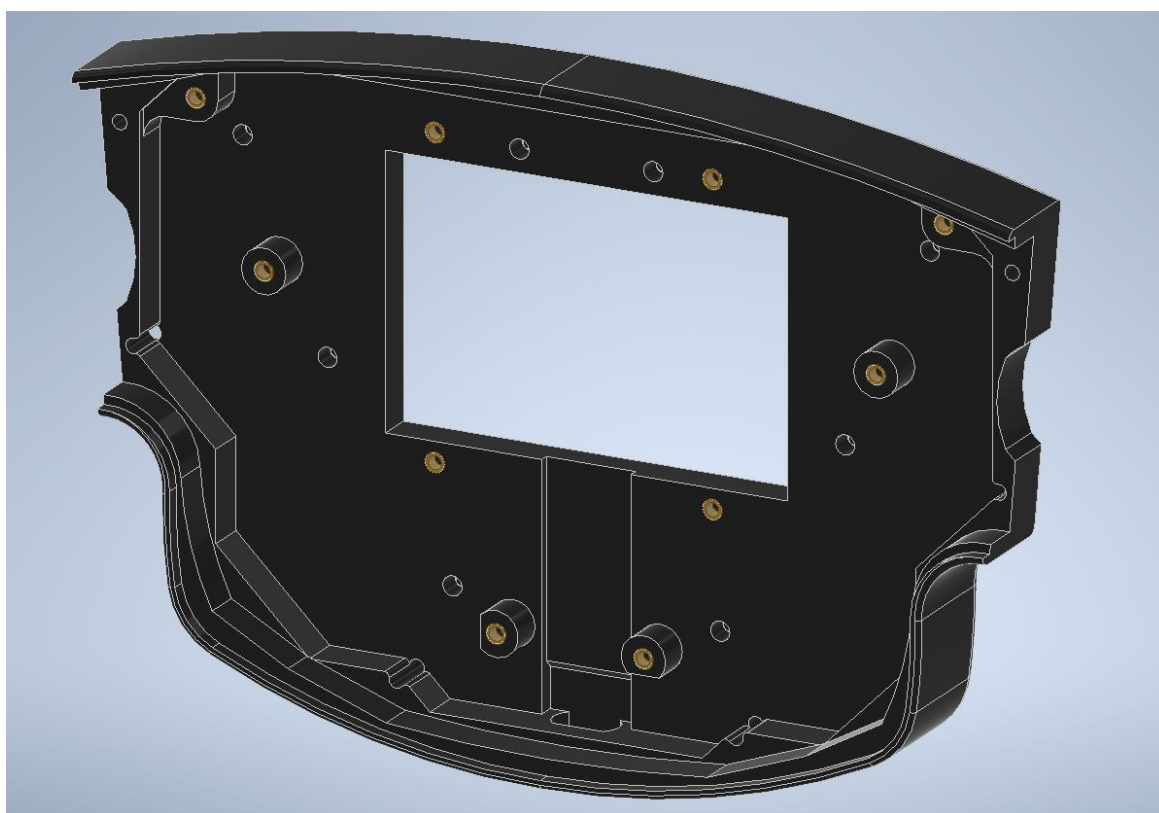
Obrázek 28 - CAD model základní desky volantu

2.3.2 Záda volantu

Záda volantu slouží k upevnění obou plošných spojů a zároveň také k připevnění střední výztuhy. Na zadní stranu je uchycena kovová příruba. Záda zároveň zakrývají strany volantu, s výjimkou otvorů pro výztuhu.

Základní deska je upevněna na sloupcích, které zaručují dostatek místa pro SMD součástky a nožky průchozích komponent. Pro mikrokontroler je udělán hlubší výřez a otvor pro Micro-USB konektor, sloužící pro programování volantu. Tento otvor lze zevnitř zakrýt záslepkou, aby se zamezilo zbytečnému vstupu prachu.

Pro všechny šroubové spoje jsou do vytisknuté součásti zataveny závitové vložky se závitem velikosti M3. Vložky umožňují získat ve vytisknuté součásti přesný a pevný závit.



Obrázek 29 - Záda volantu

2.3.3 Výztuha

Na plošném spoji je umístěna plastová výztuha. Je 8,5 mm tlustá, nachází se v ní otvory pro součástky a také otvory se závitovými vložkami pro upevnění nejen samotné výztuhy, ale také držadel.

Aby bylo výztuhu možné vyrobit na běžné 3D tiskárně, je navržena s možností rozdělení na dvě poloviny. Rozdělení na tyto dvě části by na celkovou pevnost volantu nemělo mít příliš vliv, jelikož je volant v horizontálním směru zpevněn také zády, předním krytem a kovovou přírubou. Navíc jsou poloviny spojeny dvěma kovovými výztuhami s tloušťkou 2 mm.



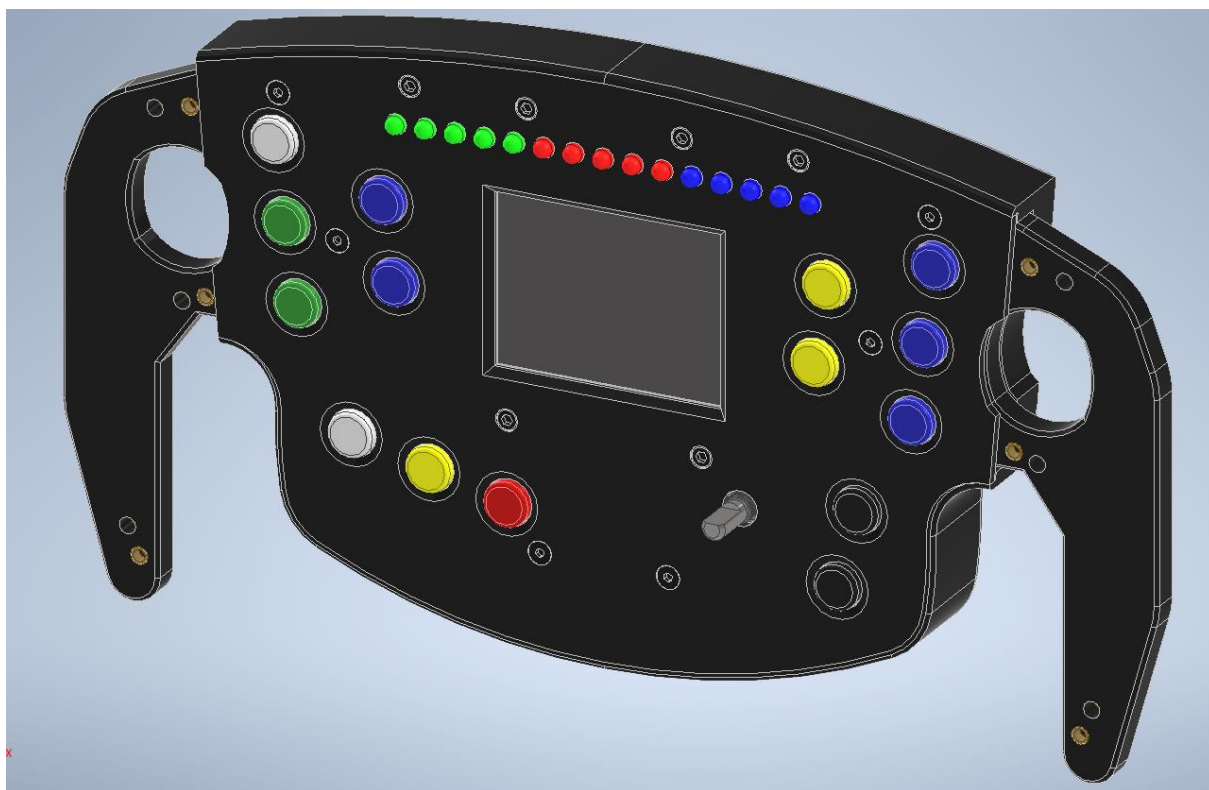
Obrázek 30 - CAD model volantu s výztuhou



Obrázek 31 - Pohled na zadní stranu výztuhy volantu

2.3.4 Přední kryt

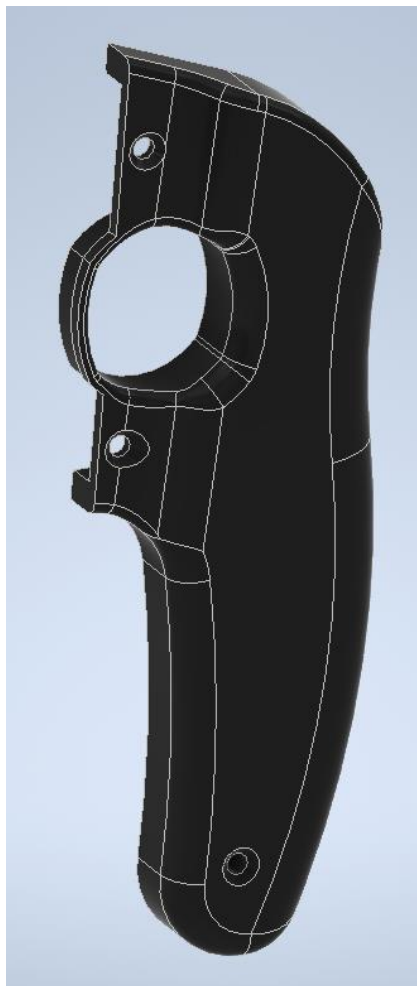
Aby volant zepředu vypadal pěkně, vytvořili jsme kryt, který zakrývá vnitřek konstrukce. Kryt je pouze 3 mm tlustý, jsou v něm otvory pro komponenty a také pro hlavy šroubů. Jsou použity zápusťné šrouby DIN 7991, aby byla přední strana plochá.



Obrázek 32 - CAD model volantu s předním krytem

2.3.5 Držadla

Na stranách výztuhy jsou připevněny symetrická držadla. Každé z nich je tvořeno přední a zadní částí. Každá strana držadla je přichycena třemi zápusťnými šrouby a je možné tak držadla jednoduše upravit a zaměnit za jiná.



Obrázek 33 - Přední strana držadla volantu

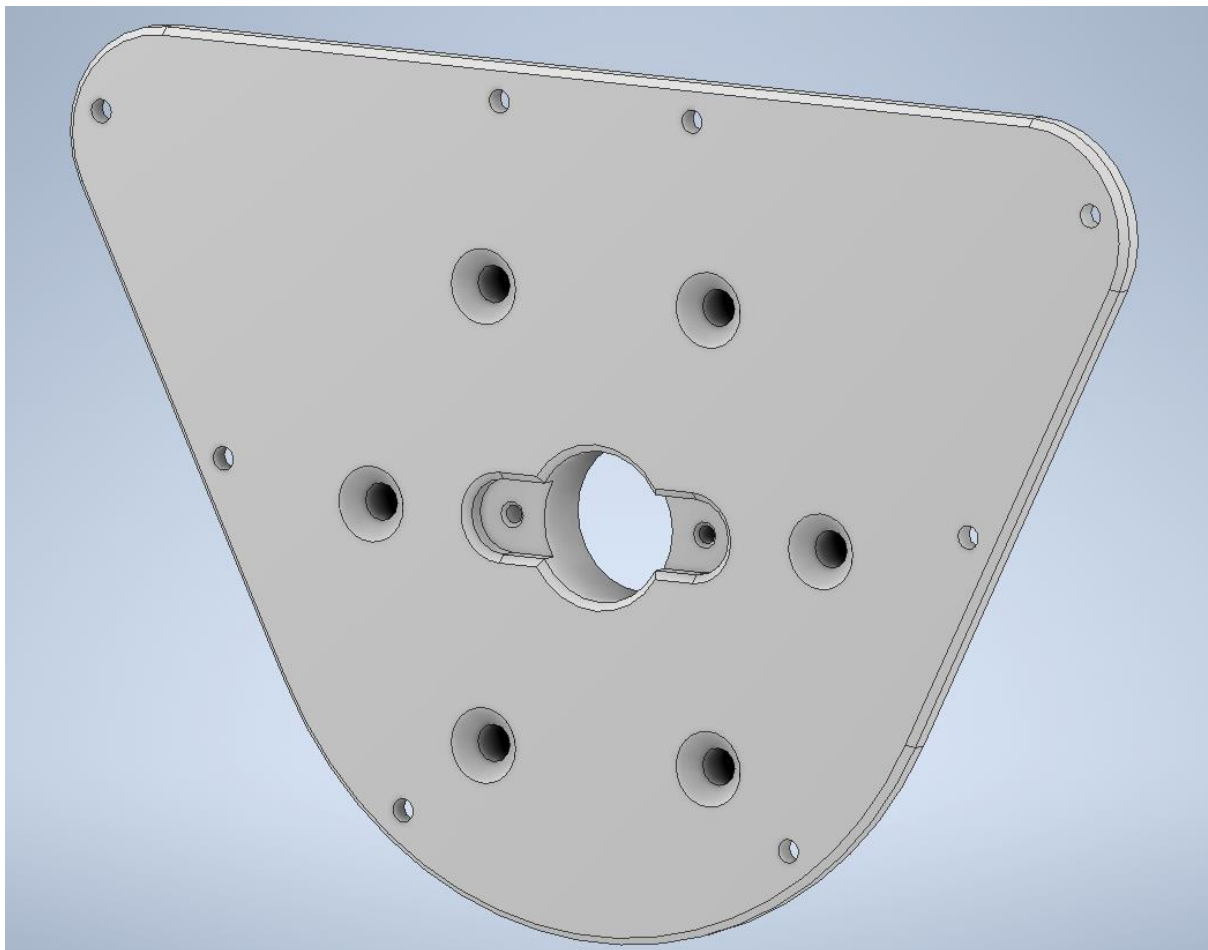


Obrázek 34 - Zadní strana držadla volantu

2.3.6 Příruba pro rychloupínací náboj

Pro uchycení volantu k rychloupínacímu náboji slouží kovová příruba. K zadům volantu je příruba připevněna osmi šrouby M3, k rychloupínacímu náboji potom pomocí šesti šroubů M5, které jsou zajištěny pojistnými maticemi.

Ve středu příruby se nachází otvor pro umístění 5 pinového konektoru.



Obrázek 35 - Příruba pro rychloupínací náboj

2.3.7 Rychloupínací konektor

Aby bylo do formule možné rychle nasedat a vysedat, je potřeba volant vyjmout. K tomu slouží rychloupínací náboj, jehož jedna část je připevněná k tyči řízení a druhá k volant. Abychom mohli volant připojit a zároveň zanechat možnost ho rychle odebrat, museli jsme navrhnout vlastní konektor, který je umístěn na rychloupínacím náboji.



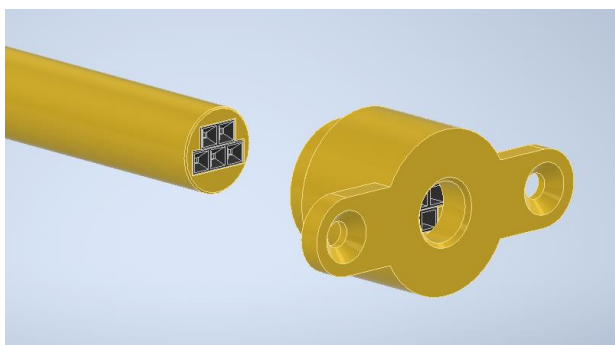
Obrázek 36 - Rychloupínací náboj

Výhodou tohoto náboje je, že velmi dobře zamezuje rotaci a vibracím mezi volantem a řídícím mechanismem. Můžeme tedy využít pevnosti náboje a při návrhu přidaného konektoru se nemusíme příliš zabývat odolností proti pohybu.

Jelikož volant není z hlediska spotřeby proudu příliš náročný, zvolili jsme řešení tzv. dupont konektorů vsazených do plastových dílů vyrobených na 3D tiskárně. Tyto díly jsou vloženy uvnitř rychloupínacího náboje. Díky tomu, že je konektor schován uvnitř náboje, je také lépe chráněn proti prachu.

Konektor má 5 pinů:

- Napájení 5 V
- Země
- 2x Sériová linka
- Pin pro rozpoznání připojeného volantu



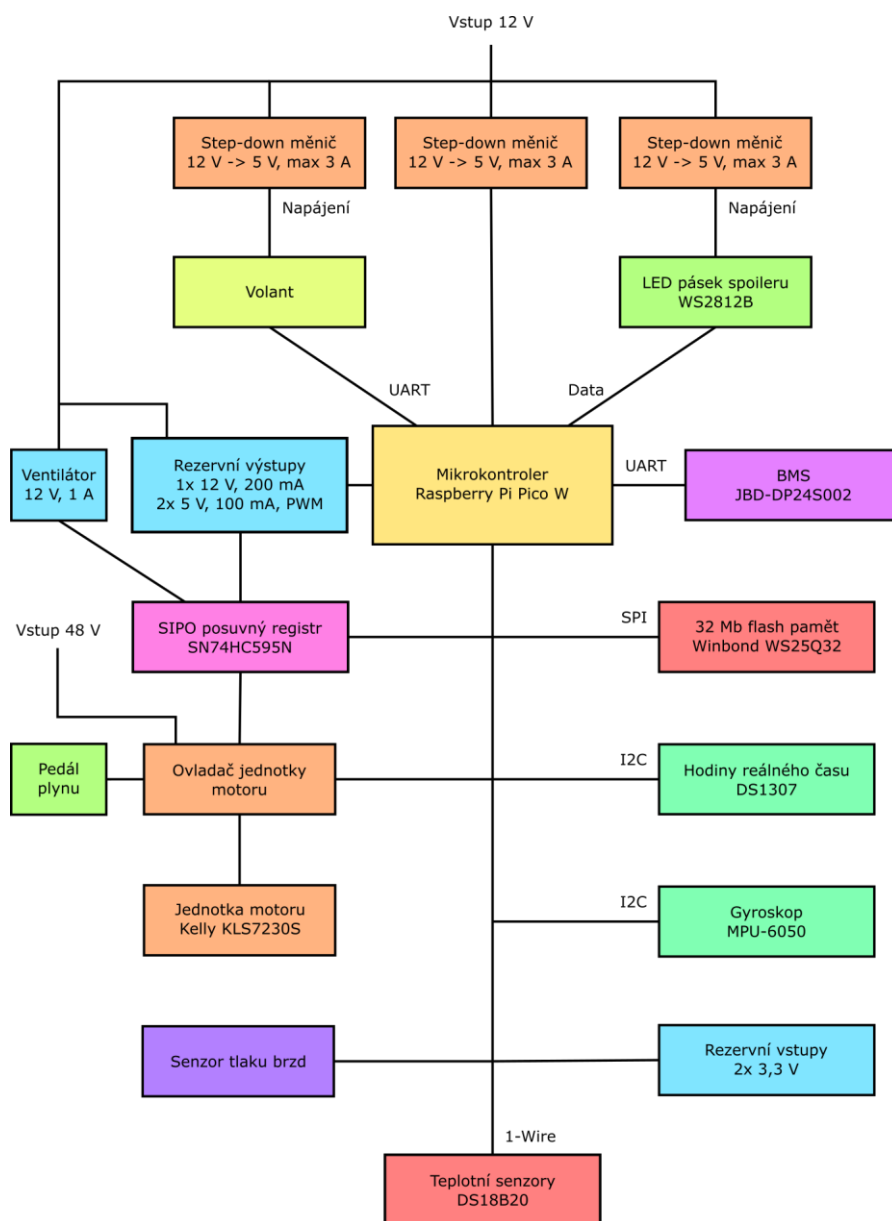
Obrázek 37 - Konektor do rychloupínacího náboje

3 ŘÍDÍCÍ JEDNOTKA

Řídicí jednotka je „srdcem“ celé práce. Jedná se o krabičku, která je umístěna v motorovém prostoru formule a slouží jako prostředník mezi volantem/aplikací a všemi elektronickými komponentami formule.

3.1 Elektronika

Všechny komponenty jsou svedeny do jednoho mikroprocesoru **Raspberry Pi Pico W**, některé přes jednoduché digitální nebo analogové signály, jiné přes sběrnice 1-Wire, UART, I2C nebo SPI.



Obrázek 38 - Blokové schéma elektroniky řídicí jednotky

Protože naprostá většina komponent vyžaduje napájení 5 V nebo 3,3 V, používáme pro převod step-down měniče. Rozhodli jsme se rozdělit napájení do tří větví:

- 5V – slouží pro napájení celé řídicí jednotky, včetně rezervních výstupů
- 5V_STEERING_WHEEL – určené pro napájení volantu
- 5V_SPOILER – napájení pro LED pásek na spoileru

Každá z těchto větví používá samostatný step-down měnič, schopný trvale dodávat výstupní proud až 3 A. [6] Protože například samotný LED pásek může při plném výkonu odebírat více než 2,5 A, jevílo se rozdělení jako ideální řešení.



Obrázek 40 - Použitý step-down měnič. *Mikro step-down měnič, nastavitelný*. Online. In: LáskaKit. C2025. Dostupné z: <https://www.laskakit.cz/mikro-step-down-menic--nastavitelny/>. [cit. 2025-03-05].

Pro komponenty, které jsou napájeny 3,3 V je využit step-down měnič integrovaný uvnitř mikrokontroleru Raspberry Pi Pico W, který je pro účely řídicí jednotky dostatečně výkonný.

3.1.2 Raspberry Pi Pico W

Stejně jako u volantu byl pro ovládání řídicí jednotky zvolen mikrokontroler Raspberry Pi Pico W. Bluetooth a Wi-Fi konektivita je u řídicí jednotky využita nejen pro OTA aktualizace, ale také pro připojení k mobilní aplikaci.

3.1.3 Komunikace s volantem

Pro komunikaci s volantem je využito jedno ze dvou hardwarových UART rozhraní dostupných na mikrokontroleru.

Pro detekci připojeného volantu je použit další pin spolu s externím pull up rezistorem.

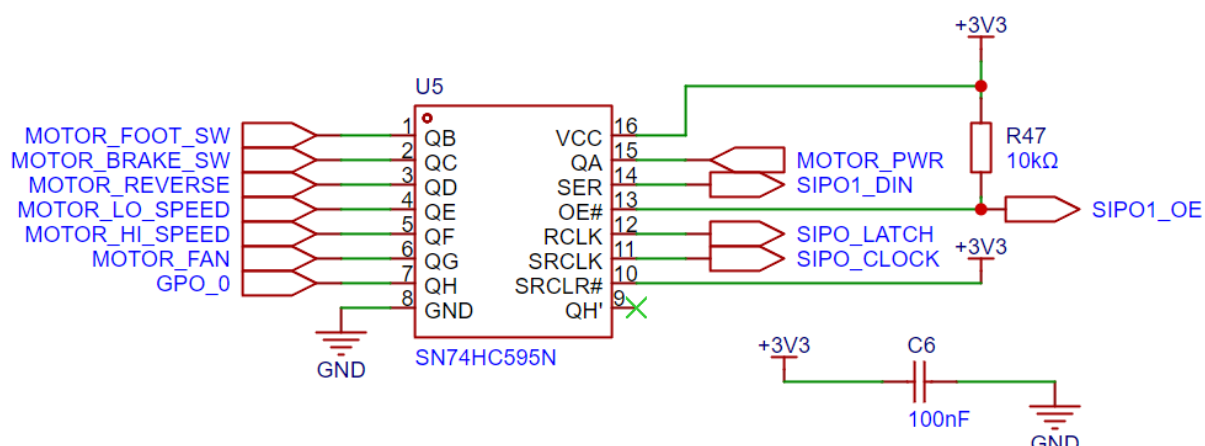
3.1.4 Komunikace s jednotkou motoru

Pro řízení běhu motoru se ve formuli používá jednotka KLS7230S od společnosti Kelly Controls.

Pro komunikaci s jednotkou jsou určeny dva konektory s dohromady 13 piny: [7]

- Konektor 1
 - **Zpátečka** – Slouží k přepínání směru motoru. Rozsah je 0 nebo 12 V.
 - **Země**
 - **Vysoká rychlost** – Přepínání na vysokou rychlost. Rozsah je 0 nebo 12 V.
 - **12 V** – Výstup napětí z jednotky motoru, zatížitelný maximálně 40 mA.
 - **Spínač brzdy** – Sepnutím dojde k zapnutí maximálního brždění motorem. Rozsah je 0 nebo 12 V.
 - **Nízká rychlost** – Přepínání na nízkou rychlost. Rozsah je 0 nebo 12 V.
 - **Spínač napájení** – Sepnutím s napětím baterií (48 V) dojde k zapnutí jednotky motoru.
- Konektor 2
 - **Spínač pedálu plynu** – Jednotka reaguje na pedál plynu pouze pokud je tento vstup sepnutý. Rozsah je 0 nebo 12 V.
 - **Pedál plynu** – Analogový vstup pro pedál plynu. Rozsah je 0-5 V.
 - **Země**
 - **Hallův senzor** – Výstup pro výpočet otáček motoru. Rozsah je 0 nebo 12 V.
 - **5 V** – Výstup napětí z jednotky motoru, zatížitelný maximálně 40 mA.
 - **Pedál brzdy** – Analogový vstup pro brždění motorem. Rozsah je 0-5 V.

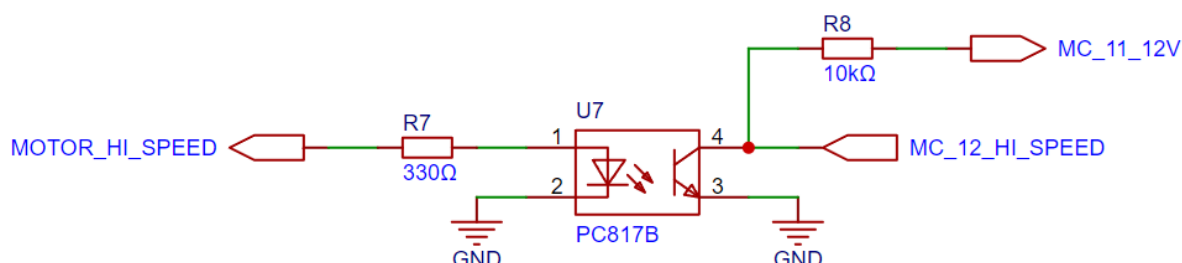
Abychom snížili počet pinů na mikrokontroleru využitých pro komunikaci, zvolili jsme možnost řízení digitálních vstupů pomocí SIPO posuvného registru **SN74HC595N**.



Obrázek 41 - Zapojení posuvného registru pro ovládání digitálních vstupů jednotky motoru

Pomocí čtyř pinů mikrokontroleru tedy lze ovládat šest digitálních vstupů jednotky motoru. Zbývající dva výstupy posuvného registru jsou použity pro ovládání výstupu pro ventilátor a 12 V rezervního výstupu.

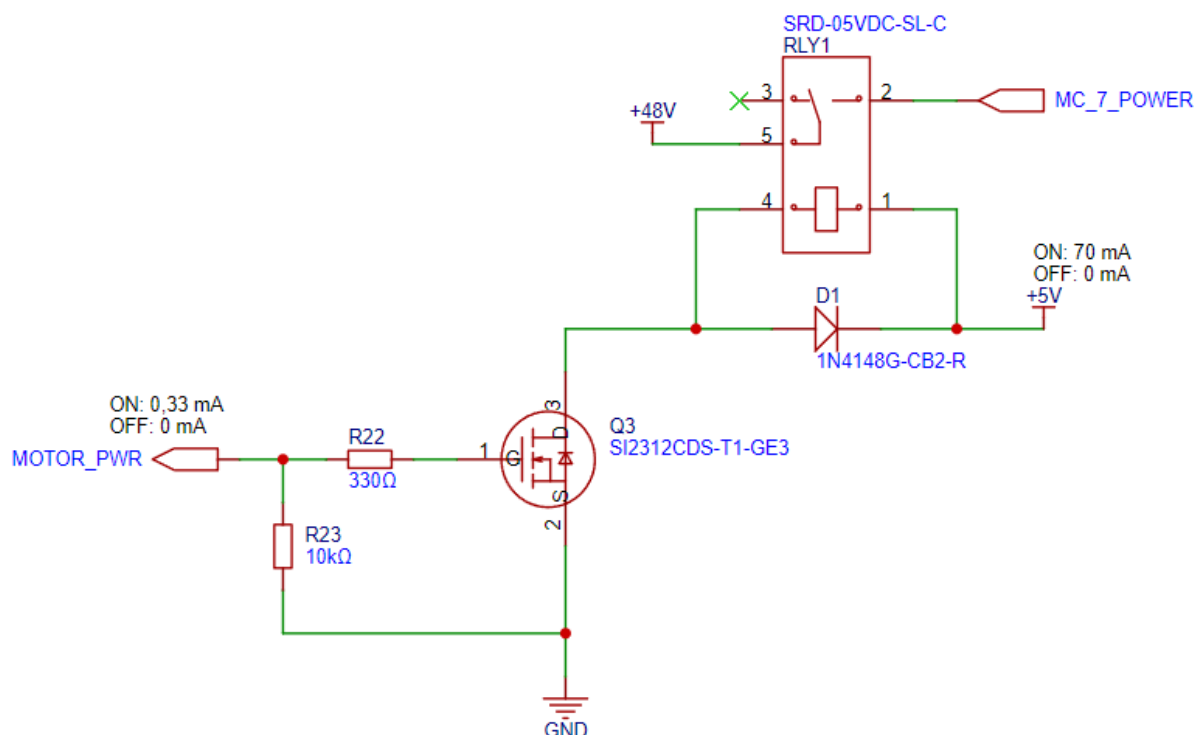
Protože výstupní napětí posuvného registru je pouze 3,3 V a vstupy jednotky by měly být spínány s 12 V, je pro každý vstup použit optočlen **PC817B**.



Obrázek 42 - Zapojení jednoho z digitálních vstupů jednotky motoru pomocí optočlenu

Pomocí tohoto zapojení s pull up rezistorem ovládáme vstup jednotky z jejího vlastního napájecího zdroje, tak jak je doporučeno výrobcem. [7] Signál je kvůli tomuto zapojení invertován, to lze ovšem jednoduše vyřešit ve firmwaru.

Výjimku z těchto signálů tvoří signál „MOTOR_PWR“, který slouží k zapnutí jednotky motoru. Tento 48 V signál by měl být spínán přímo s napětím baterií, je proto použito relé.



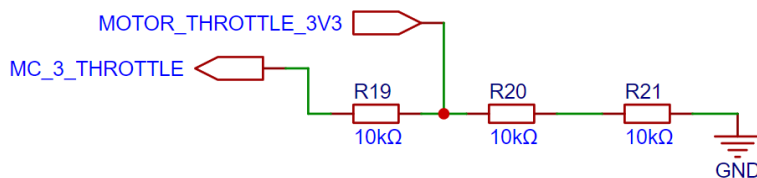
Obrázek 43 - Zapojení relé pro zapnutí jednotky motoru

Pro spínání relé, které v sepnutém stavu odebírá proud cca. 70 mA a vyžaduje napětí 5 V je použit MOSFET typu N, konkrétně **SI2312CDS-T1-GE3** od společnosti VBsemi. Toto zapojení dovoluje spínat proudově náročné relé pomocí výstupu z posuvného registru, jehož jeden pin by neměl být zatížen více než cca. 7 mA. [8] Důležité je zapojit diodu v závěrném směru paralelně s cívkou relé, abychom zamezili poškození komponent při vypnutí relé.



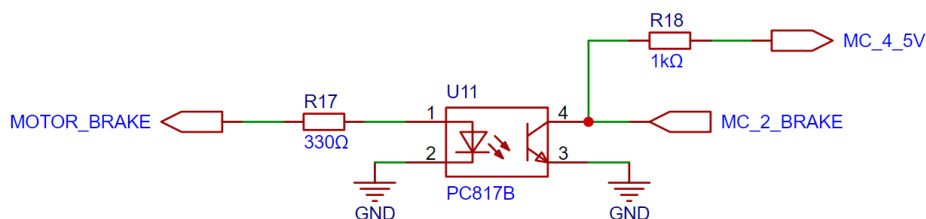
Obrázek 44 - Jednotka motoru Kelly KLS7230S. Kelly KLS7230S,24V-72V,300A, Sine Wave Brushless Motor Controller. Online. In: EV Components Store. C2025. Dostupné z: <https://www.evcomponents.com/kls7230s-300a-72v-sine-wave-brushless-controller.html>. [cit. 2025-03-06].

Signál pedálu plynu naše jednotka z bezpečnostních důvodů přímo neovládá, pouze čte jeho napětí, které vychází z pedálu plynu umístěného ve formuli. Jelikož je signál v rozsahu 0-5 V a analogové převodníky v mikrokontroleru pracují pouze v rozsahu 0-3,3 V, je signál upraven jednoduchým děličem napětí a následně čten jedním ze tří dostupných převodníků.



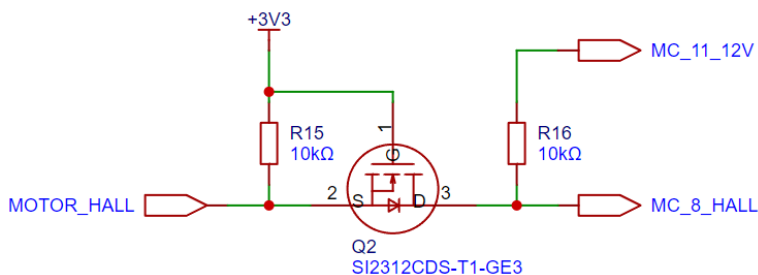
Obrázek 45 - Čtení napětí pedálu plynu pomocí děliče napětí.

Pro ovládání analogového vstupu brzdy je použito podobné zapojení s optočlenem jako pro digitální vstupy, ovšem s rozdílným pull up rezistorem. Místo 10kΩ rezistoru je zde použit rezistor hodnoty 1kΩ. Menší hodnota rezistoru umožňuje rychlejší náběžnou hranu signálu, vhodnou pro PWM výstup s frekvencí 1 kHz. Pull up rezistor je samozřejmě připojen k 5 V zdroji jednotky motoru.



Obrázek 46 - Zapojení pro ovládání analogového vstupu brzdy

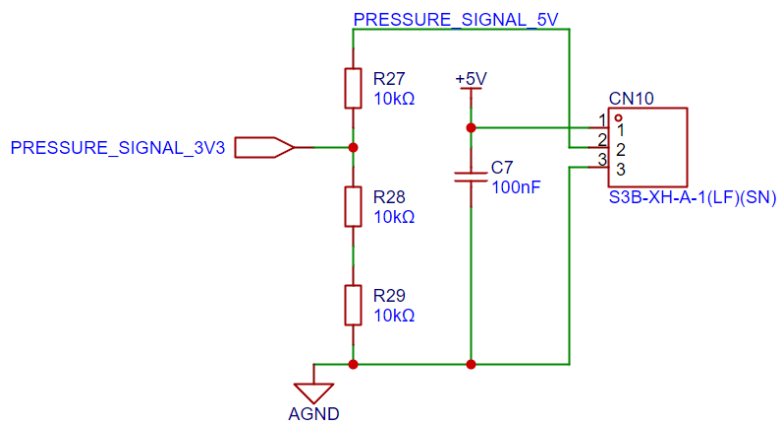
Posledním signálem je výstup z jednotky motoru pro čtení stavu hallových senzorů. Tento výstup z jednotky motoru kopíruje stav všech tří hallových senzorů, které jsou umístěny uvnitř motoru. Signál převádíme na 3,3 V pomocí MOSFET tranistoru typu N.



Obrázek 47 - Zapojení pro čtení stavu hallových senzorů

3.1.5 Senzor tlaku brzd

Měření tlaku v brzdovém systému je řešeno tlakovým senzorem s rozsahem 0-1,2 MPa. Výstup z tohoto senzoru je v podobě analogového napětí 0-5 V. Stejně jako u pedálu plynu toto napětí převádíme na 0-3,3 V pomocí jednoduchého děliče napětí a poté ho čteme jedním z analogových převodníků.



Obrázek 48 - Zapojení tlakového senzoru



Obrázek 49 - Tlakový senzor s rozsahem 0-1,2 MPa. *Tlakový senzor 0-1,2MPa, závit G1/4*. Online. In: Hadex. C2011-2025. Dostupné z: <https://www.hadex.cz/m553a-tlakovy-senzor-0-12mpa-zavit-g14/>. [cit. 2025-03-06].

3.1.6 Komunikace s BMS

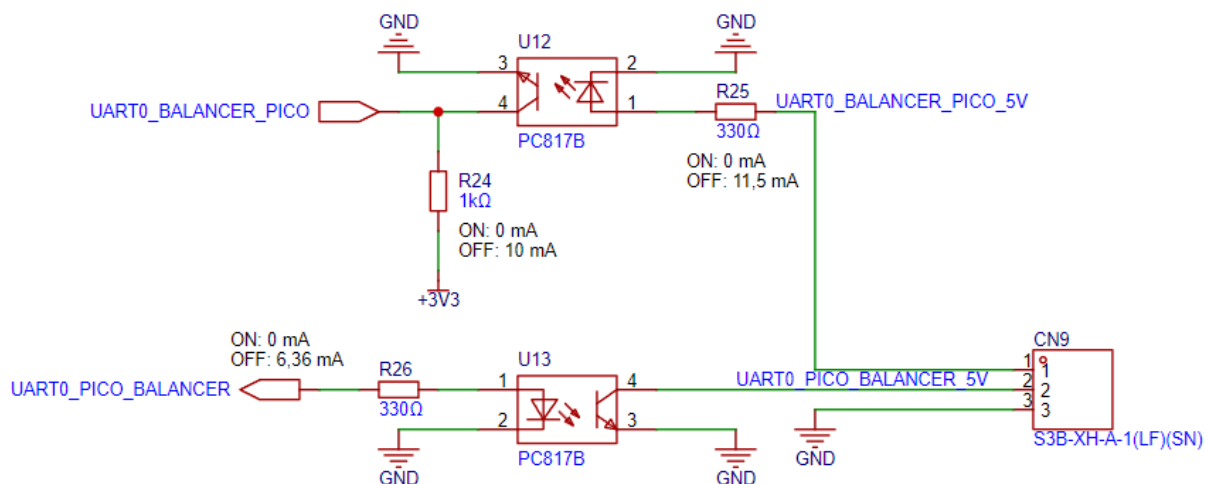
BMS (Battery management system) je zařízení, které se stará o stav akumulátoru ve formuli. V první řadě akumulátor chrání proti přebíjení, podbíjení, zkratu i přehřátí. Dále aktivně balancuje jednotlivé články, hlídá jednotlivá napětí a monitoruje celkový stav. Slouží také Použití článků bez BMS by bylo velice nebezpečné. [9]

Jedním z cílů řídicí jednotky je s BMS komunikovat, a číst tak data o akumulátoru. K tomu na BMS **JBD-DP24S002** slouží komunikační UART konektor. Na konektoru jsou k dispozici 3 piny:

- TX
- RX
- GND

Připojení datových linek a převod logických úrovní mezi 3,3 V (strana řídicí jednotky) a 5 V (strana BMS) je realizováno pomocí optočlenů. V tomto případě je naprosto zásadní galvanicky oddělit řídicí jednotku a BMS, protože ve stavu, kdy je výstup BMS vypnut, vzniká mezi zeměmi napětí.

Obvod invertuje logické signály, což lze jednoduše vyřešit ve firmwaru.

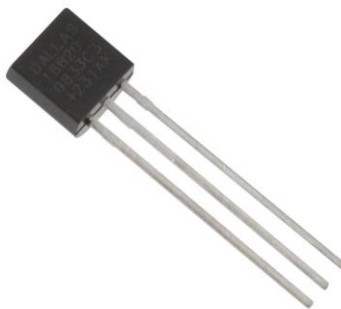


Obrázek 50 - Zapojení galvanicky odděleného převodníku napětí pro komunikaci s BMS

3.1.7 Teplotní senzory

Pro měření teplot v různých částech formule jsme využili digitální teplotní senzory **DS18B20** od společnosti Dallas. Každý senzor má 3 piny, **VCC**, **data** a **GND**. Se senzory se komunikuje pomocí sběrnice 1-Wire, která umožňuje pomocí jednoho vodiče komunikovat s teoreticky neomezeným počtem zařízení.

Mezi datovým vodičem a napájecí větví 3,3 V je třeba umístit 4,7k Ω pull up rezistor.



Obrázek 51 - Teplotní senzor DS18B20. *Digitální teplotní senzor DALLAS – DS18B20*. Online. In: Drátek.cz. Dostupné z: <https://dratek.cz/arduino/1187-teplotni-senzor-digitalni-dallas-ds18b20.html>. [cit. 2025-03-06].

Jeden ze senzorů je umístěn přímo uvnitř řídicí jednotky, další jsou umístěny na těchto místech:

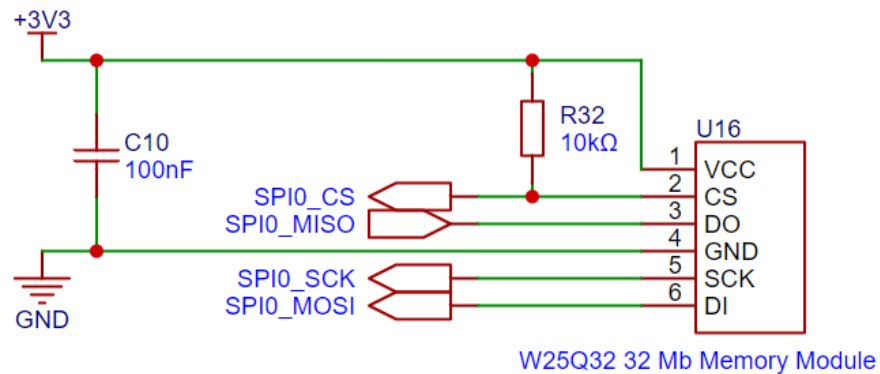
- Jednotka motoru
- Motor
- Diferenciál
- Brzda
- Senzor pro teplotu okolí

Pro připojení senzorů k řídicí jednotce jsou vyhrazeny dva konektory, čímž je umožněno případné přidání dalších senzorů.

Mimo tyto senzory je k dispozici také teplota baterií, kterou hlásí BMS.

3.1.8 Flash paměť

K trvalému ukládání dat slouží 32 Mb flash paměť **Winbond W25Q32**. Paměť je umístěna na samostatném modulu a komunikuje se s ní pomocí sběrnice **SPI**. Zapojení paměti je jednoduché, pro stabilizaci napájení je použit 100nF blokovací kondenzátor, a na pinu CS se nachází pull up rezistor pro zajištění správného stavu při startu jednotky.



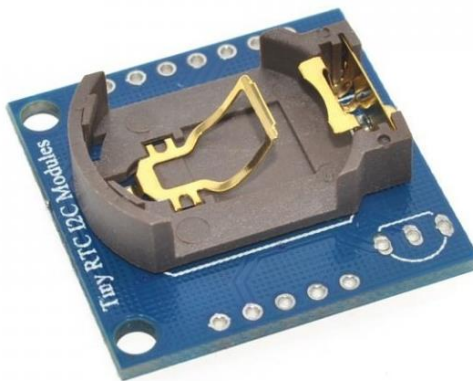
Obrázek 52 - Zapojení flash paměti



Obrázek 53 - Modul flash paměti. *Winbond W25Q32 Flash 32Mb SPI Modul*. Online. In: LáskaKit. C2025. Dostupné z: <https://www.laskakit.cz/winbond-w25q32-flash-32mb-spi-modul/>. [cit. 2025-03-06].

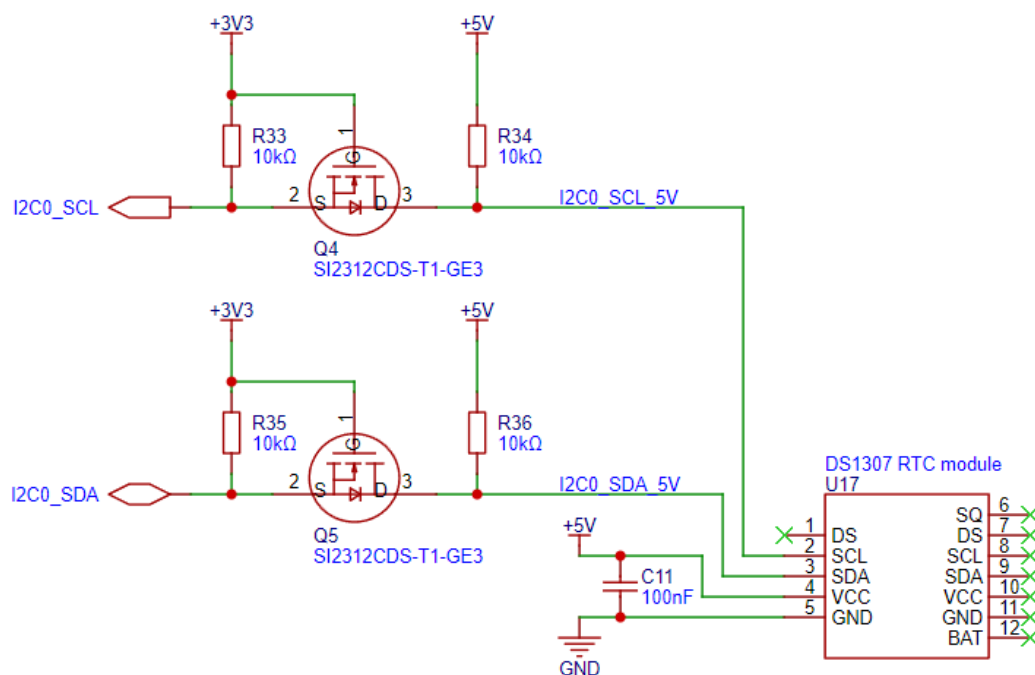
3.1.9 Hodiny reálného času

Hodiny reálného času slouží, jak název napovídá, k držení správného času i ve stavu, kdy řídicí jednotka není napájena. Použitý modul je založen na čipu **DS1307** a jsou napájeny baterií LIR2032. Komunikace s hodinami probíhá přes sběrnici **I²C**. Na modulu se zároveň nachází 32 kb EEPROM čip AT24C32, ten ovšem řídicí jednotka nepoužívá.



Obrázek 54 - RTC modul. *Mini RTC Hodiny reálného času DS1307*. Online. In: LáskaKit. C2025. Dostupné z: <https://www.laskakit.cz/arduino-mini-rtc-hodiny-realneho-casu-ds1307/>. [cit. 2025-03-06].

Modul je napájen a komunikuje na úrovni 5 V, pro komunikaci s mikrokontrolerem je tedy třeba napětí převést. K tomu slouží obvod logických převodníků, tvořený MOSFET tranzistory typu N a pull up rezistory. Pokud jakákoliv strana převodníku sepne signál k zemi, změna se projeví i na druhé straně.

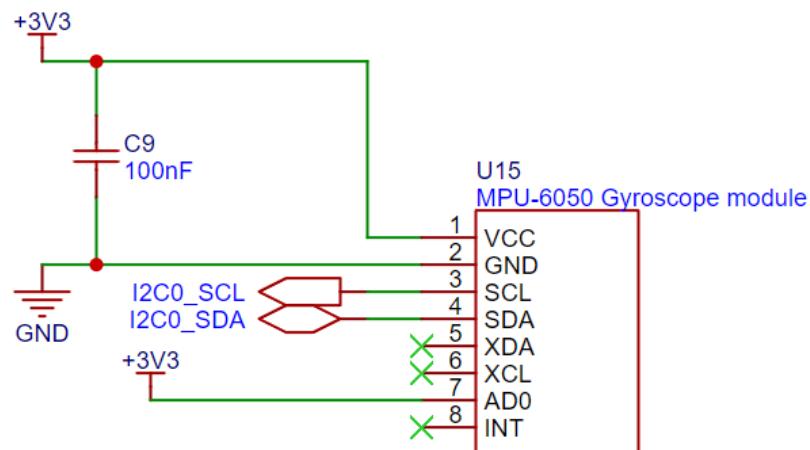


Obrázek 55 - Zapojení RTC modulu

3.1.10 Gyroskop a akcelerometr

Pro snímání dat o pohybu je použit modul gyroskopu a akcelerometru s čipem **MPU-6050**. Modul komunikuje pomocí sběrnice I²C na napětí 3,3 V.

Pro připojení k mikrokontroleru je třeba připojit linky SDA a SCL, modul je poté na stejné sběrnici jako hodiny reálného času. Pro změnu adresy je třeba spojit pin AD0 s napájecím napětím, v opačném případě by modul byl v konfliktu s hodinami.



Obrázek 56 - Zapojení akcelerometru a gyroskopu

3.1.11 LED páska na spoileru

Adresovatelný páska použitý na spoileru využívá LED s čipem **WS2812B**. Ovládání tohoto čipu je z hlediska zapojení velmi jednoduché, stačí připojit pin DIN k libovolnému pinu na mikrokontroleru.

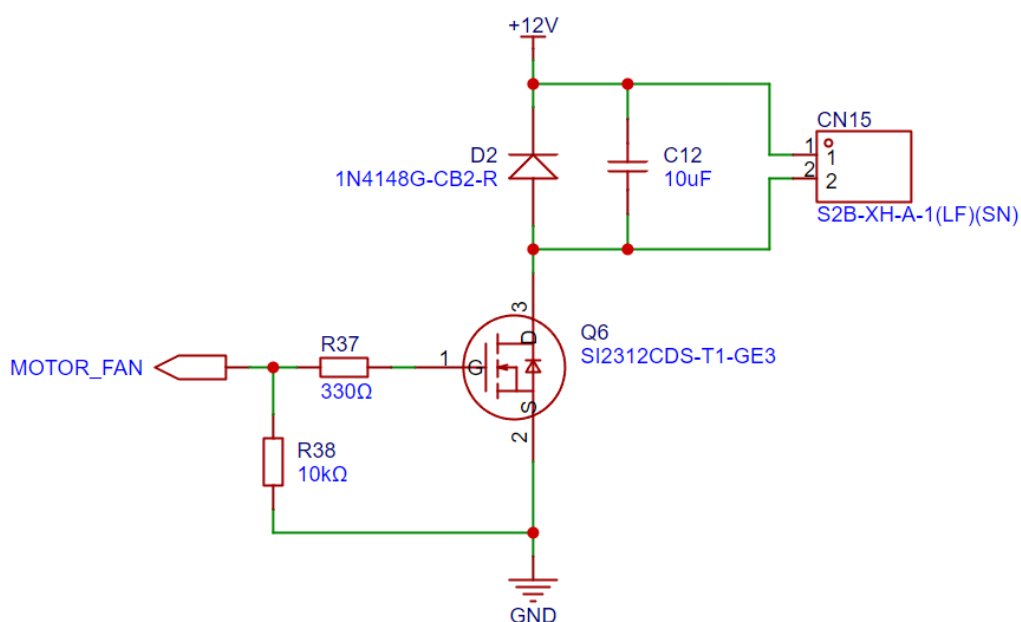
Napájení pásku probíhá z jeho vlastní napájecí větve 5 V. Protože výrobce udává výkon až 18 W/m, je důležité dodržet šířku cesty na plošném spoji, který s délkou 0,7 m může dosáhnout až 2,52 A. [10]

3.1.12 Ventilátor, rezervní vstupy a výstupy

Po návrhu zapojení všech komponent na řídicí jednotce zůstalo celkem 6 nevyužitých vstupů nebo výstupu. Rozhodli jsme se je využít jako **dva 12 V výstupy, dva 5 V výstupy a dva 3,3 V vstupy/výstupy**.

12 V výstupy jsou řízeny z výstupů posuvného registru, z toho důvodu u nich lze nastavovat pouze stavy zapnuto/vypnuto. 5 V výstupy jsou řízeny přímo z pinů mikrokontroleru a lze u nich tak využít PWM.

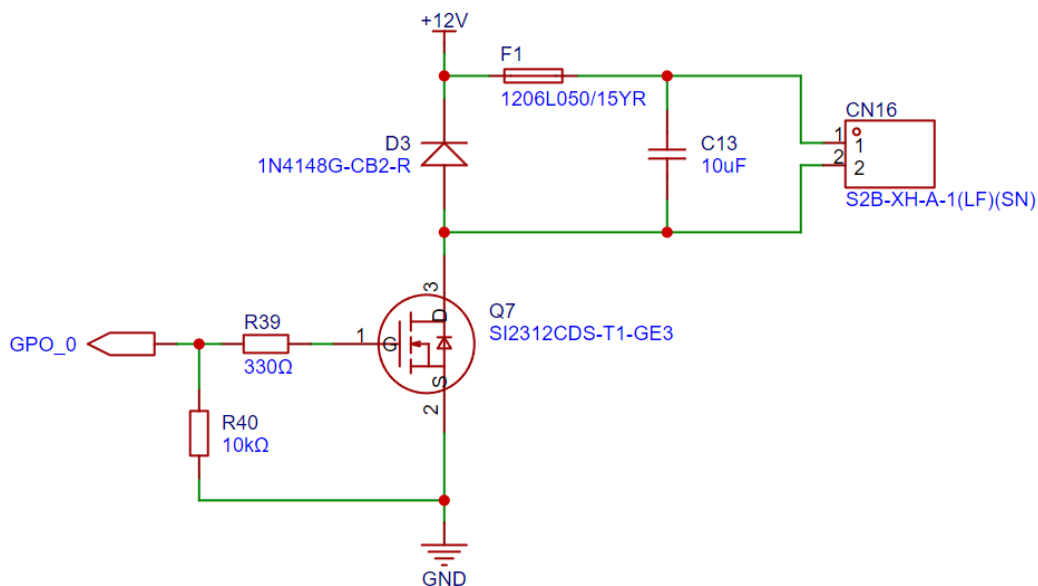
Jeden z 12 V výstupů je využit pro ventilátor jednotky motoru, která se při jízdě velmi zahřívá. Tento výstup je zatížitelný až 1 A a je spínáný pomocí MOSFET tranzistoru typu N. Je tedy možné ventilátor vypnout/zapnout.



Obrázek 57 - Zapojení výstupu pro ventilátor jednotky motoru

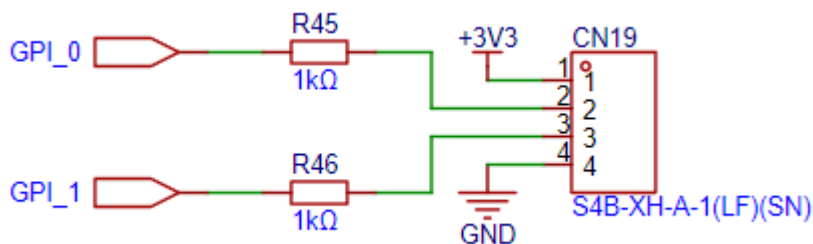
Zbylé 12 V a 5 V výstupy se od sebe liší pouze napětím, maximálním doporučeným proudem a možností využívat PWM. Každý výstup je ovládán MOSFET tranzistorem typu N. Protože jsou tyto výstupy myšleny primárně pro ovládání relé, obsahují tzv. flyback diodu pro ochranu řídicích obvodů před napětíovou špičkou při rozpojení relé.

12 V výstup je doporučeno zatížit maximálně 200 mA, 5 V výstupy maximálně 100 mA. Doporučená hodnota je samozřejmě určena s velkou rezervou, výstupy jsou proto jištěny vratnými pojistkami s maximálním trvalým proudem až 500 mA. Při jakémkoliv přetížení nebo zkratu těchto výstupů by tedy nemělo dojít k zahřívání nebo poškození řídicí jednotky.



Obrázek 58 - Zapojení jednoho z rezervních výstupů

Poslední dva piny jsou zamýšleny primárně jako digitální vstupy, pro ochranu pinů mikrokontroleru jsou v řadě zapojeny 1kΩ rezistory. Na pinu GPI_1 je k dispozici analogový převodník. Oba piny lze využít také jako digitální či PWM výstupy, stále ovšem s rezistory v řadě.



Obrázek 59 - Zapojení rezervních vstupů

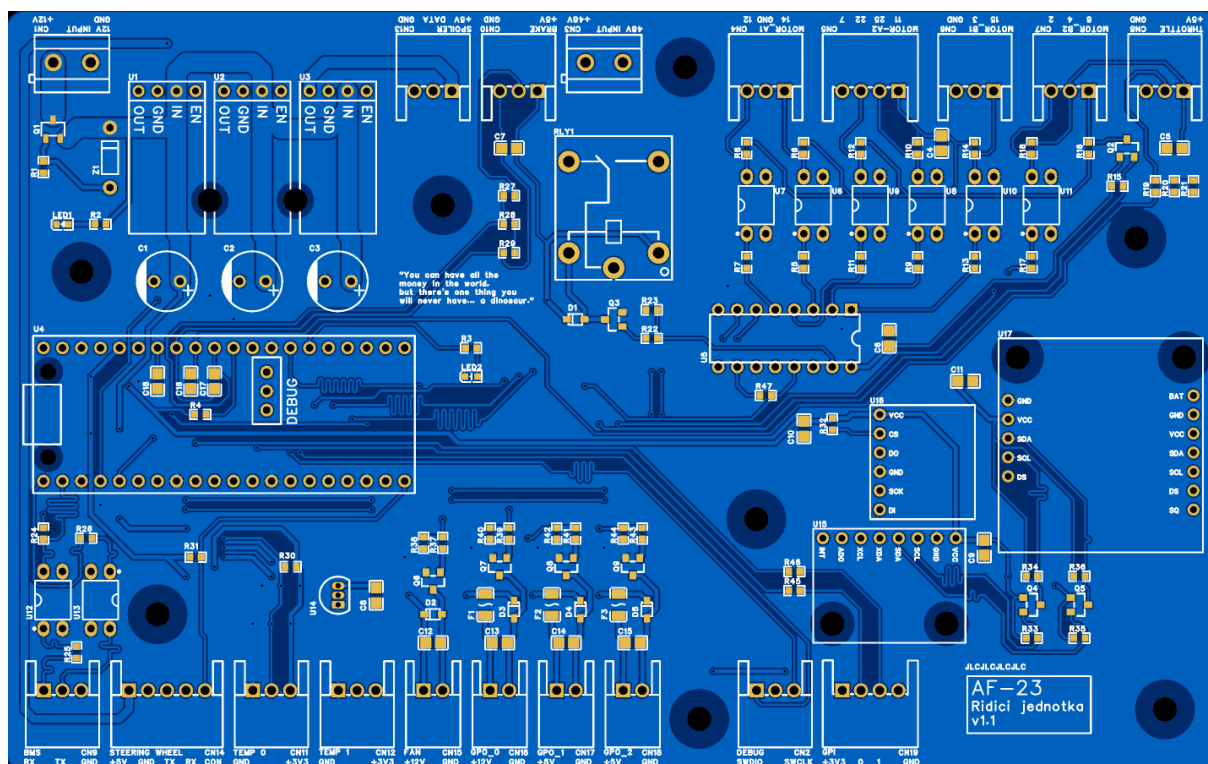
3.1.13 Plošný spoj

Pro řídicí jednotku jsme též vytvořili plošný spoj, na kterém jsou umístěny všechny součástky. Na dvouvrstvém plošném spoji jsou opět dodrženy dostatečné šířky cest pro silové obvody a obvody komunikačních rozhraní jsou navrženy se stejnou délkou cest.

Pro uchycení samotného plošného spoje slouží 8 děr s průměrem 3 mm. Další otvory slouží pro umístění podložek některých komponent, jako například step-down měničů, které jsou osazeny mimo samotný plošný spoj a mohl by u nich vzniknout problém kvůli vibracím. Kolem všech děr je dodržena volná zóna bez jakékoliv mědi.

Připojení ke zdrojům 12 V a 48 V je vyřešeno pomocí svorkovnic **KF301-5.0-2P**, které jsou vhodné pro vyšší proudová zatížení. Pro ostatní připojení jsou využity 2-5 pinové konektory z řady **JST-XH**.

Mikrokontroler Raspberry Pi Pico W je umístěn na dutinkových lištách a zajištěn dvěma šrouby M2. Tím je umožněno jeho vyjmutí bez nutnosti pájení, ale zároveň je zamezeno jeho nechtěnému odpojení.



Obrázek 60 - Vrchní strana plošného spoje řídicí jednotky

3.2 Firmware

Firmware řídicí jednotky je navržen tak, aby vycházel z podobného principu jako firmware volantu, s cílem zajistit optimální komunikaci a kontrolu všech klíčových komponent systému. I v tomto případě je použita modulární architektura, která však tentokrát zohledňuje specifické požadavky jednotlivých modulů zaměřených na řízení konkrétních částí systému, jako je motor, baterie, nebo brzdový systém. Každý modul je zodpovědný za obsluhu a kontrolu určitého zařízení, což umožňuje efektivní správu všech subsystémů bez vzájemného ovlivnění.

Řídicí jednotka průběžně zaznamenává data od většiny komponent v pravidelných intervalech a ukládá je do paměti. Tyto hodnoty slouží k monitorování stavu vozidla a k prediktivnímu řízení jeho chování. Na základě příkazů přijatých z volantu nebo aplikace jednotka zpracovává a reaguje na změny v systému.

3.2.1 OTA aktualizace firmwaru

OTA (Over-the-Air) aktualizace firmwaru na řídicí jednotce je navržena tak, aby umožnila bezdrátovou aktualizaci firmwaru bez potřeby fyzického připojení k jednotce. Tento proces probíhá v několika fázích a využívá specifický postup pro komunikaci mezi mobilním zařízením a řídicí jednotkou formou Wi-Fi hotspotu. Po zahájení aktualizace se řídicí jednotka přepne do "update módu", což zahrnuje aktivaci Wi-Fi hotspotu na jednotce, která následně čeká na připojení mobilního zařízení.

Jakmile je mobilní zařízení připojeno k Wi-Fi síti, která je vytvořena řídicí jednotkou, první krok zahrnuje odeslání základních informací o firmwaru, jako je velikost souboru a jeho MD5 hash pro kontrolu integrity. Tato data jsou přenášena prostřednictvím UDP protokolu. Pokud je kontrola integrity úspěšná, přechází se na samotný přenos firmwaru, který je prováděn pomocí TCP protokolu. Výběr TCP je motivován jeho spolehlivostí, která zajišťuje, že data budou přenesena bez chyby i při dlouhých nebo velkých souborech.

Přenos dat probíhá v několika etapách. Po zahájení přenosu firmwaru jsou soubory odesílány po malých paketových blocích. Pro každý přijatý blok je potvrzena jeho integrita a správnost pomocí MD5 kontroly. Jakmile je celý firmware úspěšně přenesen a ověřen, řídicí jednotka zahájí aktualizací proces, který zahrnuje zápis nového firmwaru do paměti zařízení. Tento proces je řízen pomocí knihovny **PicoOTA**.

Po úspěšném přenosu a zápisu nového firmwaru je zařízení restartováno, což umožňuje okamžitě spustit novou verzi firmwaru bez potřeby manuálního zásahu. Pokud by během procesu došlo k jakékoliv chybě, řídicí jednotka automaticky vrátí zařízení do původního stavu a zahájí opětovný pokus o aktualizaci.

3.2.2 Ovládání motoru

Tento modul slouží k ovládání motoru a sběru důležitých dat, která jsou potřebná pro správnou funkci motoru a jeho řízení. Má několik funkcí, které umožňují sledování otáček motoru, čtení stavu plynového pedálu a nastavení motorové brzdy.

Motorová brzda je řízena pomocí analogového signálu. Když chceme brzdít, posíláme hodnoty mezi 0 (maximální brzdění) a 255 (nejméně brzdění). Tento signál je vysílán do motoru, aby upravil jeho výkon a zpomalil jej podle potřeby.

Hallův senzor je používán k měření otáček motoru. Tento senzor detekuje, jak často se rotor motoru otáčí. Každý impuls, který senzor zaznamená, znamená jednu část otáčky. Modul měří čas mezi těmito impulzy, a na základě toho spočítá otáčky motoru za minutu. Tato hodnota je důležitá pro sledování výkonu motoru, aby bylo možné například upravit jeho rychlost nebo výkon v reálném čase. Pokud senzor po delší dobu (přes 250 milisekund) nezaznamená žádný impuls, znamená to, že motor zastavil, a otáčky jsou nastaveny na 0.

Plynový pedál je čten pomocí analogového signálu, který ukazuje, jak moc je pedál sešlápnutý. Tato hodnota se pohybuje od 0 (když je pedál úplně sešlápnutý) až po 4095 (když je pedál úplně uvolněný). Čtení této hodnoty umožňuje určit, jak silně je motor zatížen, a podle toho upravit jeho výkon.

Dále jsou další hodnoty, které ovlivňují chování motoru, nastavovány pomocí posuvného registru. Tento registr řídí funkce jako **zapnutí motoru, aktivace pedálového spínače, aktivace motorové brzdy, reverzní chod motoru, nastavení rychlostních režimů a ovládání ventilátoru motoru**. Každý z těchto parametrů je řízen pomocí signálů posílaných do motoru.

Kombinací těchto funkcí sledujeme výkon motoru, jeho otáčky a můžeme reagovat na změny v řízení, například když je pedál sešlápnutý nebo když je potřeba brzdít.

3.2.3 Komunikace s BMS

Komunikační modul BMS používá **UART** ke komunikaci s modulem **JBD-DP24S002**. Tento modul komunikuje pomocí protokolu definovaného v jeho dokumentaci, která popisuje strukturu a pravidla pro výměnu dat mezi BMS a dalším systémem.

Základní informace o BMS jsou získávány pomocí speciálního příkazu, který je odesílán přes UART. Tento příkaz vrací data, která jsou následně zpracována a převedena na lidsky čitelný formát. Mezi tato data patří informace o celkovém napětí baterie (v mV), aktuálním proudu (v mA), zbylé kapacitě (v mAh), nominální kapacitě, počtu cyklů, datumu výroby BMS a další technické parametry jako status ochrany nebo teploty senzorů baterie (ve °C).

Kontrola integrity dat je prováděna pomocí kontrolního součtu (checksum), který se ověřuje po každém přijetí dat. Pokud kontrolní součet souhlasí, data jsou považována za validní a jsou zpracována. Pokud kontrolní součet nesouhlasí, znamená to, že při přenosu došlo k chybě a data nejsou validní. Tato funkce zajišťuje, že přijaté informace jsou přesné a spolehlivé.

K dispozici jsou tři příkazy pro čtení dat: [11]

Příkaz	Popis	Data v odpovědi
0x03	Základní informace	Napětí akumulátoru Proud Zbývající kapacita Nominální kapacita Počet nabíjecích cyklů Datum výroby Vyváženost článků Stav ochrany Verze softwaru Počet baterií Počet teplotních senzorů Teploty senzorů
0x04	Napětí jednotlivých článků	Napětí článku 1 Napětí článku 2 Napětí článku 3 ...
0x05	Informace o BMS	Verze hardware

Tabulka 2 - Příkazy pro komunikaci s BMS

3.2.4 Čtení hodnot ze senzoru tlaku brzd

Modul brzdového senzoru měří tlak brzdové kapaliny pomocí analogového senzoru, který poskytuje napěťový signál odpovídající aktuálnímu tlaku. Tato hodnota je poté zpracována a převedena na tlak v jednotkách barů nebo megapascalů. Funkce modulu umožňují získat nejen surové hodnoty, ale také vypočítat tlak v bar nebo MPa, což nám usnadňuje sledování aktuálního stavu brzdového systému.

3.2.5 Čtení teplotních senzorů

Modul pro měření teploty využívá senzory DS18B20, které komunikují přes sběrnici OneWire. Při inicializaci jsou senzory detekovány a nastavena jejich rozlišovací schopnost. Funkce pro čtení teploty odečítá hodnotu přímo z konkrétního senzoru na základě jeho unikátní adresy. Teplota je následně vrácena v jednotkách Celsia. Senzory jsou schopny měřit teplotu s vysokou přesností a pokud není senzor připojen nebo došlo k problému při čtení, funkce vrátí chybu.

Pro čtení dat je využita knihovna **DallasTemperature**.

3.2.6 Flash paměť

Pro ukládání dat na flash paměť je využita kombinace dvou knihoven, **Adafruit SPIFlash** a **LittleFS**. První knihovna slouží ke komunikaci s pamětí na nízké úrovni. Dovoluje nám zapisovat data na určité místo v paměti, číst je nebo je mazat. Díky ní nemusíme psát vlastní metody pro komunikaci přes sběrnici SPI.

LittleFS je **souborový systém**, který umožňuje práci s pamětí podobně jako na počítači. Data můžeme zapisovat do souborů a řadit do složek, systém se zároveň stará o tzv. **wear leveling**, tedy rovnoměrné využívání části paměti pro zapisování. Tím zabráňuje předčasnému zničení určité části paměti kvůli častému přepisování.

Pro nastavení Adafruit SPIFlash je třeba nastavit pouze parametry paměti, o zbytek se stará knihovna. Pro LittleFS je třeba nastavit parametry systému, a definovat metody, které LittleFS volá při **zapisování/čtení/mazání** dat. V těchto metodách voláme funkce knihovny SPIFlash.

3.2.7 RTC

Hodiny reálného času komunikují pomocí sběrnice I2C a k jejich jednoduchému použití slouží knihovna **uRTCLib**. Knihovnu je třeba nejprve inicializovat a poté lze data číst několika jednoduchými metodami.

- `rtc.year()`
- `rtc.month()`
- `rtc.day()`
- `rtc.hour()`
- `rtc.minute()`
- `rtc.second()`
- `rtc.dayOfWeek()`

3.2.8 Gyroskop

Modul gyroskopu a akcelerometru také komunikuje přes sběrnici I2C. Pro sběr dat je využita knihovna **Adafruit_MPU6050**. Po inicializaci knihovny a nastavení parametrů můžeme jednoduše číst data o rotaci, akceleraci a teplotě čipu.

3.2.9 LED spoileru

Pro nastavování barev LED pásku slouží knihovna **FastLED**. Jedná se o velmi známou knihovnu, která je schopná ovládat mnoho druhů LED pásků. Po nastavení čipu použitého v LED pásku a maximálního výkonu můžeme samostatně nastavovat barvu každé LED. Po nastavení je zavolána funkce `FastLED.show()` a dojde k přenosu dat na LED pásek.

3.2.10 Rezervní vstupy a výstupy

Ovládání rezervních vstupů pro 12 V je spojeno s ovládáním výstupů pro jednotku motoru, jsou ovládány přes stejný **posuvný registr**.

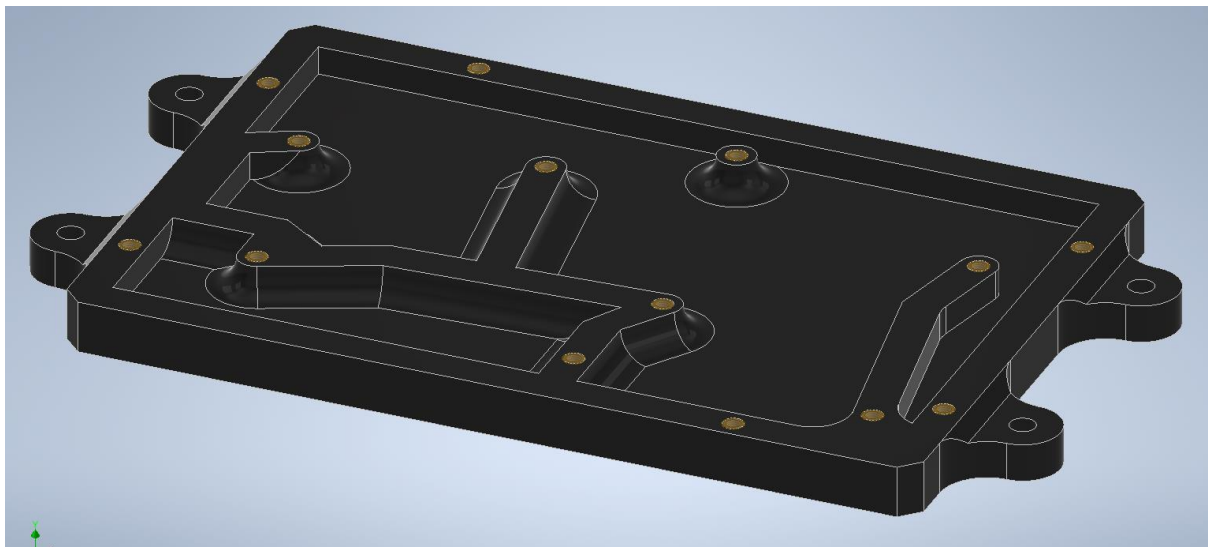
5 V výstupy jsou ovládány běžnými funkcemi **`digitalWrite()`** a **`analogWrite()`**.

Rezervní vstupy jsou čteny pomocí funkcí **`digitalRead()`**, případně **`analogRead()`**.

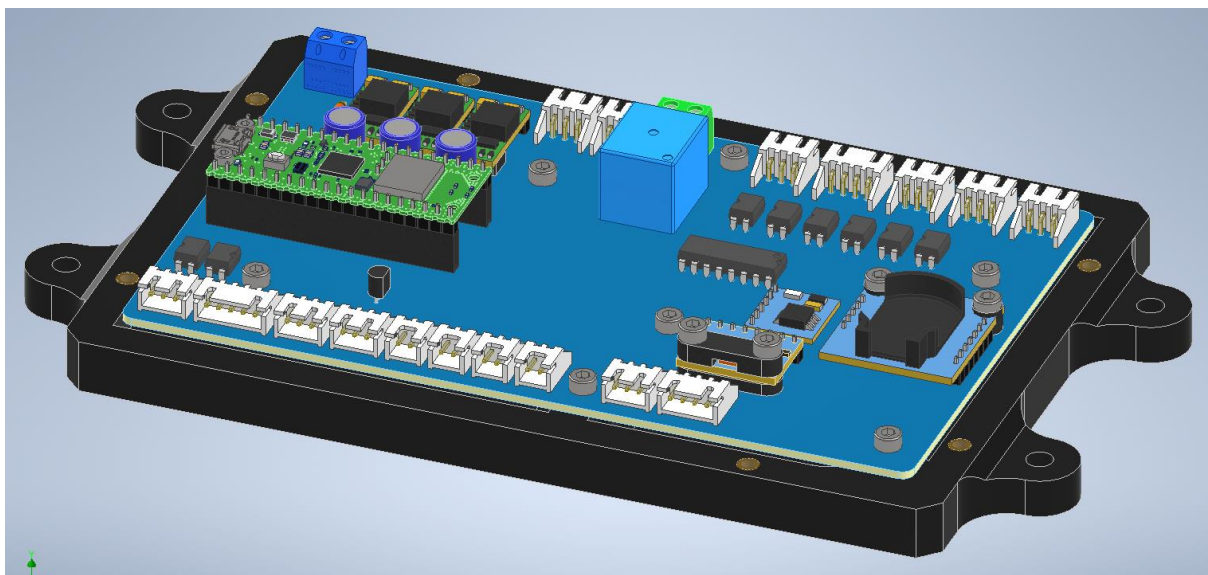
3.3 Pouzdro

Pouzdro řídicí jednotky je velmi jednoduché, je složené ze dvou částí vyrobených na 3D tiskárně.

Spodní část pouzdra obsahuje 8 vyvýšených závitových vložek pro upevnění plošného spoje. Pod plošným spojem tak vzniká dostatek místa pro nohy všech průchozích součástek. Po okrajích je umístěno 6 závitových vložek pro uchycení vrchní části pouzdra. Pro upevnění k desce s elektronikou jsou určeny čtyři otvory pro šrouby na stranách.

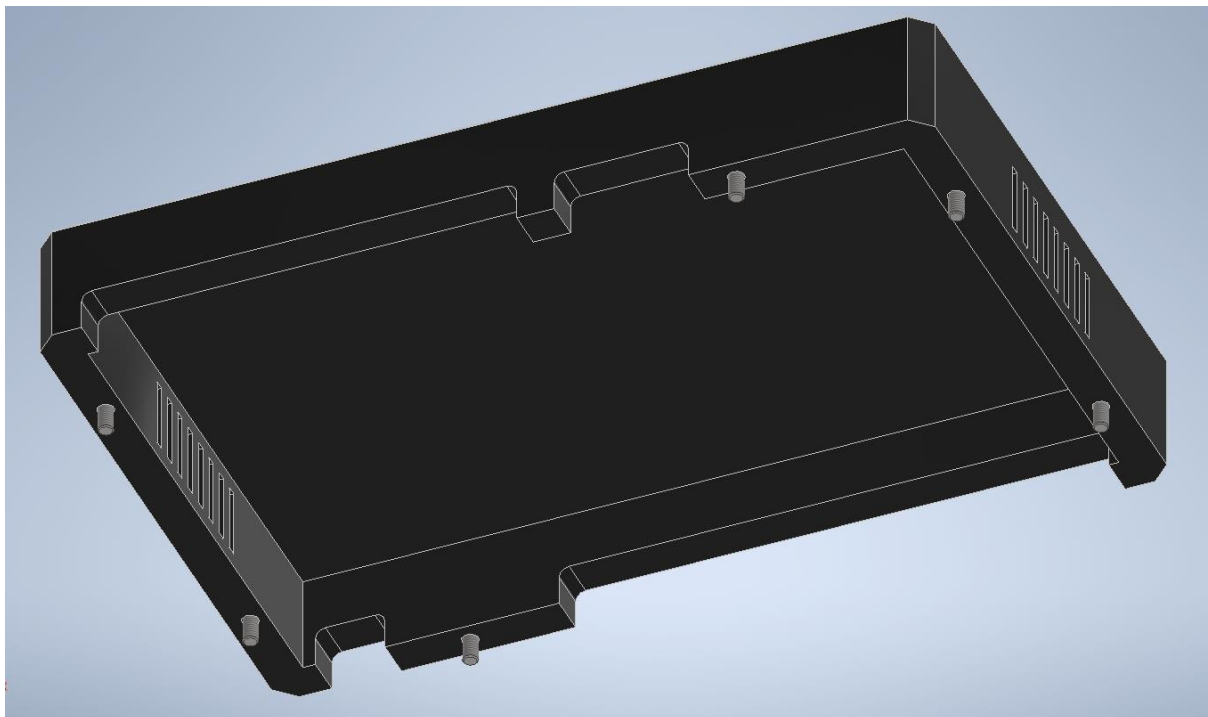


Obrázek 61 - Spodní část pouzdra řídicí jednotky

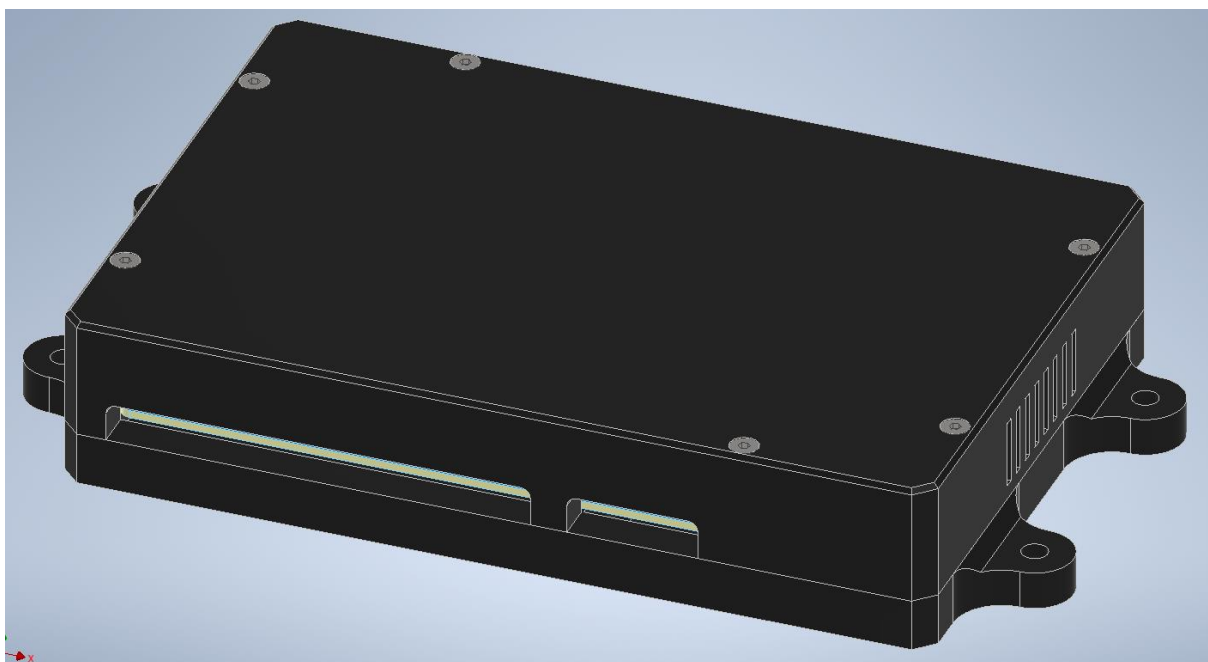


Obrázek 62 - Pouzdro řídicí jednotky s plošným spojem

Víko pouzdra obsahuje otvory pro vodiče připojených konektorů, tvar otvorů ovšem zamezuje vypojení konektorů, pokud je víko přišroubované.



Obrázek 63 - Víko pouzdra řídicí jednotky



Obrázek 64 - Zapouzdrěná řídicí jednotka

4 MOBILNÍ APLIKACE

Hlavním účelem mobilní aplikace v našem projektu je analýza jízdních dat a také možnost vzdálené aktualizace firmwaru řídicí jednotky (OTA). To je důležité, protože přístup k jednotce ve formuli je složitý, a tak je ideální mít způsob, jak ji aktualizovat bez nutnosti fyzického připojení.

Aplikace je napsaná ve frameworku **Flutter**. Tento framework byl zvolen, protože poskytuje efektivní způsob, jak pracovat s nativními funkcemi zařízení a zároveň zajišťuje plynulé uživatelské rozhraní. Flutter nám umožnil rychlý vývoj díky široké škále hotových knihoven, které jsme využili pro práci s Bluetooth a komunikaci s řídicí jednotkou. Také nám umožnil snadno implementovat vizuální prvky a animace, což pomohlo navrhnout aplikaci, která je přehledná a příjemná na používání. Tento framework také nabízí dobrou kompatibilitu s různými verzemi operačních systémů, což zjednodušuje údržbu aplikace.

4.1 Komunikace s jednotkou

Komunikace s řídicí jednotkou začíná stisknutím tlačítka na hlavní obrazovce aplikace, které slouží k navázání spojení. Jakmile uživatel klikne na toto tlačítko, aplikace požádá o oprávnění k používání Bluetooth. Po udělení potřebných práv začne aplikace vyhledávat dostupná zařízení v okolí. Ze seznamu zařízení hledá ta, jejichž název začíná předponou „AF-23“, což je identifikátor řídicí jednotky. Jakmile je zařízení nalezeno, aplikace se pokusí navázat spojení s tímto zařízením.

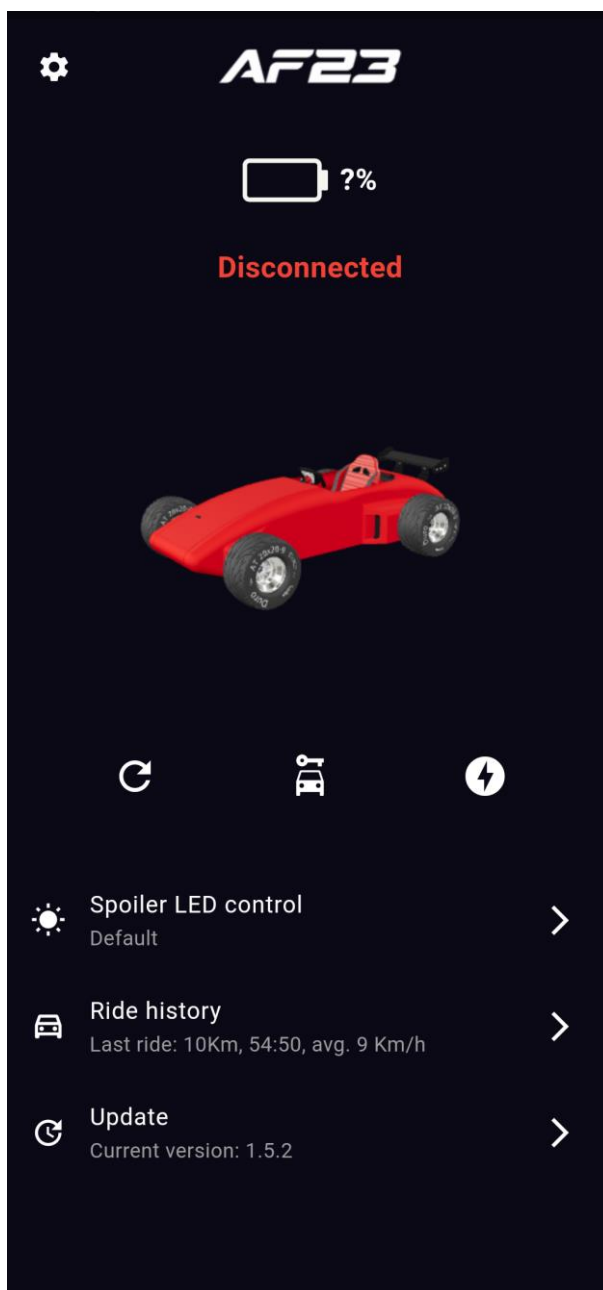
Po úspěšném připojení aplikace pošle HELLO paket, což je jednoduchý testovací signál, který slouží k ověření, že komunikace probíhá správně. Tento paket zajišťuje, že zařízení jsou schopná vzájemně komunikovat a že všechno funguje, jak má.

Aplikace nevyžaduje práci s daty v reálném čase, protože se neprovádí žádná složitá analýza nebo kontrola během jízdy. Když je potřeba získat konkrétní data, například jízdní statistiky, aplikace pošle požadavek ve formě paketu, který specifikuje, jaké údaje chce získat. Po odeslání tohoto požadavku aplikace čeká na odpověď, která obsahuje požadovaná data, například rychlost nebo stav baterie.

Díky tomuto způsobu komunikace je aplikace schopna efektivně a bezpečně získávat potřebné údaje z řídicí jednotky bez nutnosti neustálého připojení nebo zbytečného zatěžování systému.

4.2 Hlavní obrazovka

Hlavní obrazovka aplikace slouží jako rozcestník ke všem důležitým funkcím a zároveň poskytuje přehled o aktuálním stavu formule. Uživatel zde okamžitě vidí, zda je řídicí jednotka připojená, jaký je stav baterie a může provádět základní akce související s ovládáním formule.



Obrázek 65 - Hlavní obrazovka mobilní aplikace

V horní části obrazovky se nachází indikátor připojení k řídicí jednotce. Pokud je spojení navázáno, zobrazí se zelený text „**Connected**“, což signalizuje, že aplikace momentálně může komunikovat s formulí. Vedle toho je také ukazatel stavu baterie, který však zobrazí skutečnou hodnotu pouze v případě aktivního připojení k jednotce.

Uprostřed obrazovky je 3D model formule, který slouží jako vizuální reprezentace vozidla. Pod vizualizací se nachází tři hlavní ovládací tlačítka, která umožňují základní správu připojení a ovládání formule:

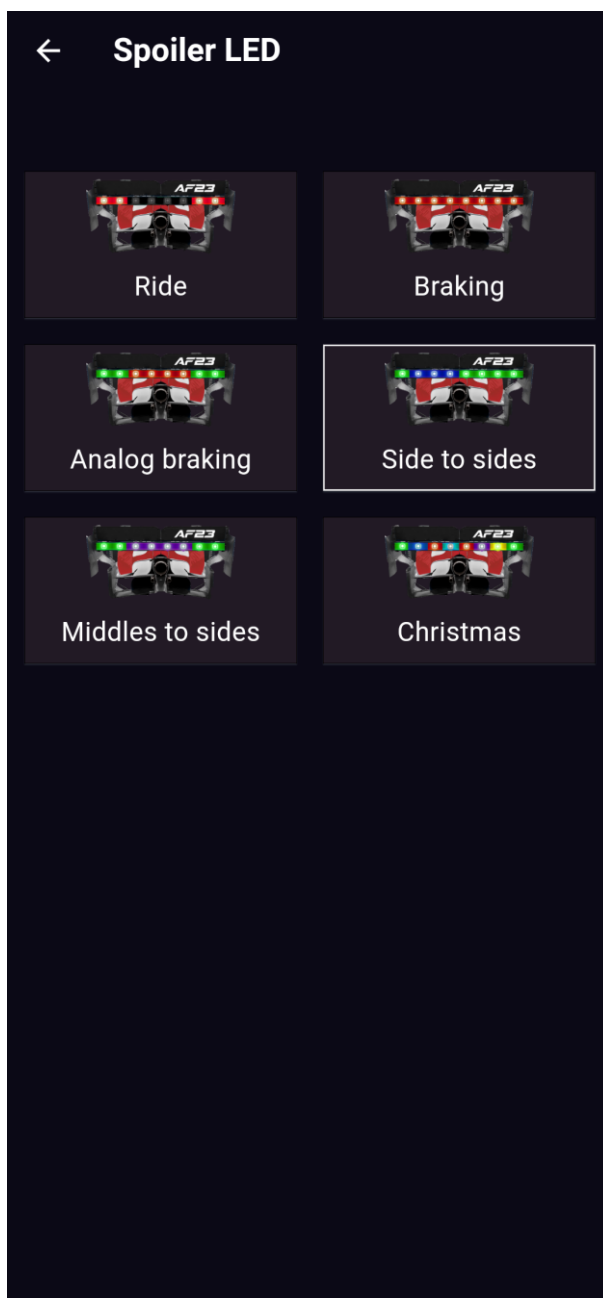
- **Obnovení připojení** – Tlačítko se symbolem kruhové šipky slouží k opětovnému navázání spojení s řídicí jednotkou. Po kliknutí aplikace začne vyhledávat dostupná zařízení a pokusí se k formuli připojit.
- **Zámek** – Symbol zámku představuje funkci deaktivace motoru.
- **Restart řídicí jednotky** – Poslední tlačítko, znázorněné bleskem, umožňuje softwarový restart řídicí jednotky. Tímto způsobem lze jednotku resetovat bez nutnosti fyzického odpojení nebo manuálního restartu přímo na vozidle.

V dolní části obrazovky jsou odkazy na další klíčové funkce aplikace, které umožňují uživateli přizpůsobit nastavení formule a zobrazit důležitá jízdní data. Funkce **nastavení LED spoileru** umožňuje změnit světelné režimy, což je užitečné například při prezentacích. **Historie jízd** uchovává statistiky poslední jízdy, jako jsou ujetá vzdálenost, doba trvání a průměrná rychlost, přičemž detailnější analýzu lze provést v samostatné sekci. **Aktualizace firmwaru** pak umožňuje provést vzdálenou OTA aktualizaci řídicí jednotky, čímž se eliminuje nutnost fyzického připojení. Hlavní obrazovka je navržena tak, aby uživatel měl všechny důležité informace a ovládací prvky na jednom místě.

4.3 Konfigurace LED spoileru

Tato obrazovka slouží ke konfiguraci světelného režimu na spoileru. Pokud chceme změnit režim, stačí kliknout na jeden z přednastavených módů. Každý režim je znázorněn v samostatném čtverečku, kde je jeho název a přibližná grafická reprezentace toho, jak bude spoiler svítit.

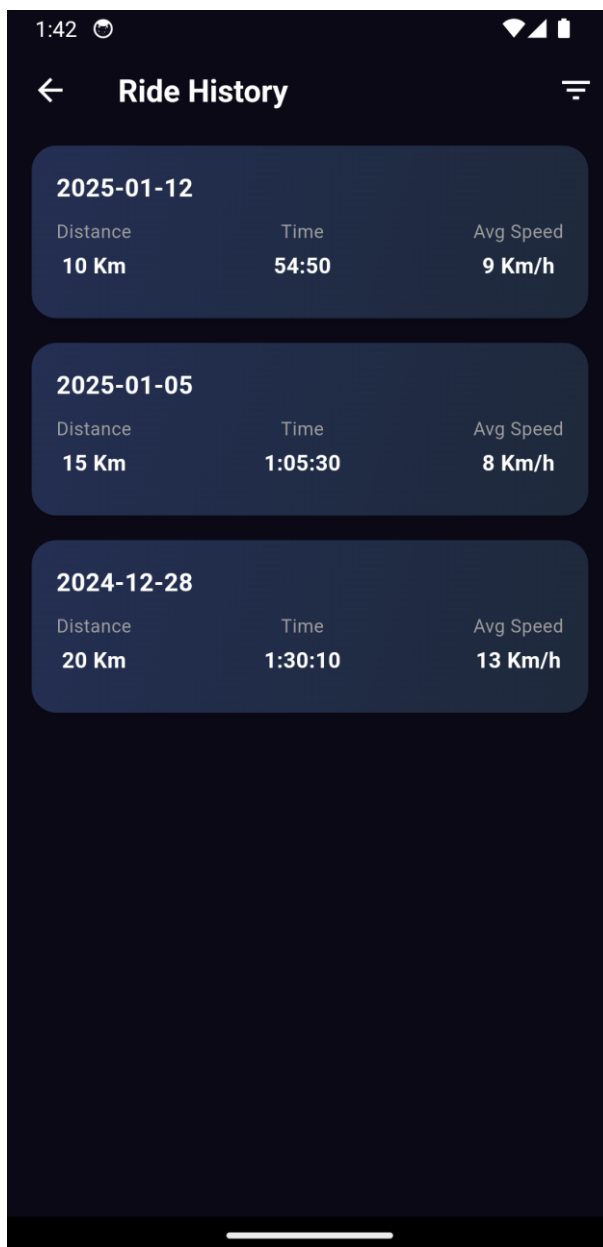
Tato funkce je užitečná při prezentacích nebo ukázkách formule, kdy chceme demonstrovat, co všechno spoiler dokáže.



Obrázek 66 - Obrazovka pro výběr světelných efektů

4.4 Přehled přechozích jízd

Funkce přehledu jízd slouží k uchovávání a analýze zaznamenaných jízd. Umožňuje uživateli rychle se podívat na základní statistiky jednotlivých jízd a zároveň usnadňuje jejich porovnání. Každý záznam obsahuje datum jízdy, ujetou vzdálenost, dobu trvání a průměrnou rychlost. V horní části obrazovky se nachází tlačítko pro filtrování záznamů – uživatel si může zobrazit pouze jízdy podle zvoleného časového období nebo délky trasy, což usnadňuje práci s větším množstvím dat. Po kliknutí na konkrétní jízdu se otevře její detailní analýza, kde se zobrazí další statistiky a grafická vizualizace průběhu jízdy.

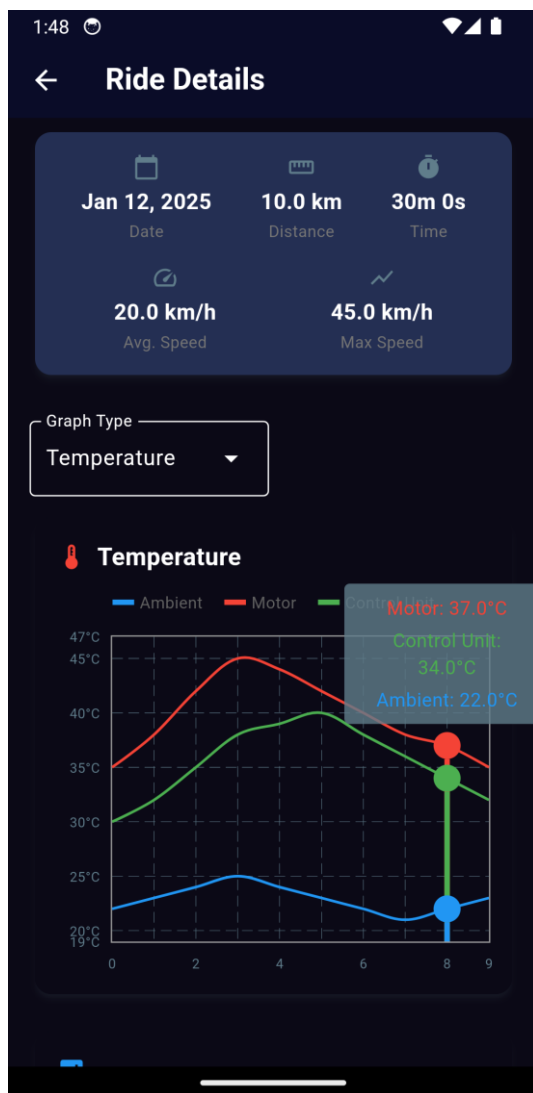


The screenshot shows a mobile application interface titled "Ride History". It features a dark theme with three distinct ride records, each enclosed in a rounded rectangle. Each record displays the date, distance, time, and average speed. The data is as follows:

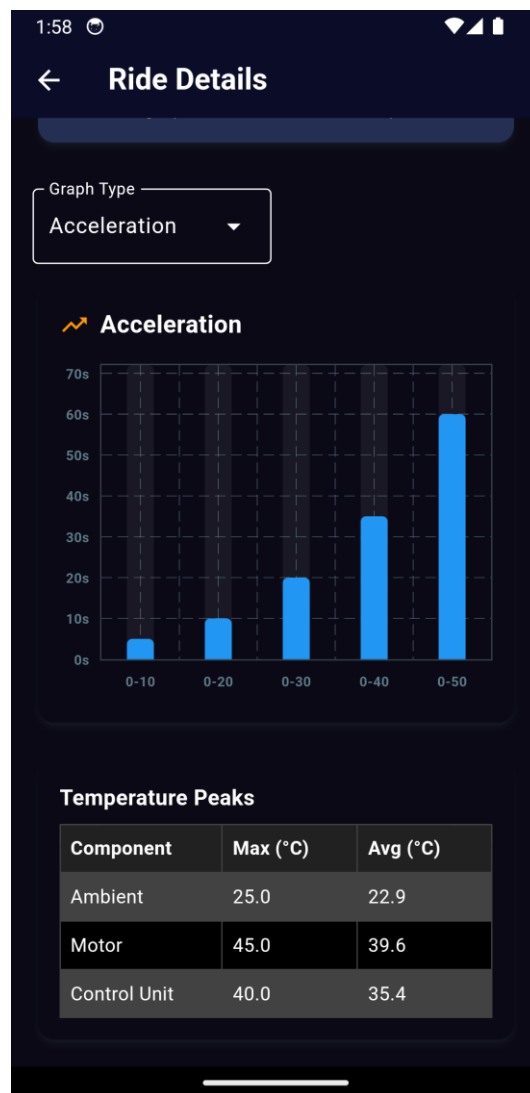
Date	Distance	Time	Avg Speed
2025-01-12	10 Km	54:50	9 Km/h
2025-01-05	15 Km	1:05:30	8 Km/h
2024-12-28	20 Km	1:30:10	13 Km/h

Obrázek 67 - Obrazovka s historií jízd formule

Na detailní obrazovce konkrétní jízdy se v horní části nachází souhrnný přehled s klíčovými údaji. Uživatel zde vidí datum jízdy, ujetou vzdálenost, délku trvání jízdy, průměrnou a maximální rychlost. Hlavní část této obrazovky tvoří grafická analýza průběhu jízdy. Aktuálně jsou dostupné tři hlavní grafy: **graf rychlosti** zobrazující změnu rychlosti v průběhu jízdy, což umožňuje analyzovat zrychlení a zpomalení, **graf teploty** umožňující sledovat teploty na různých částech formule, například motoru, řídicí jednotky a okolního prostředí, a **graf zrychlení** ukazující nejlepší dosažené zrychlení v různých rychlostních pásmech, například v intervalech po 10 km/h až do 50 km/h. Pod grafy se nachází ještě **tabulka teplot**, kde lze snadno vidět průměrné a maximální hodnoty naměřené jednotlivými senzory.



Obrázek 68 - Detaily jízdy v mobilní aplikaci



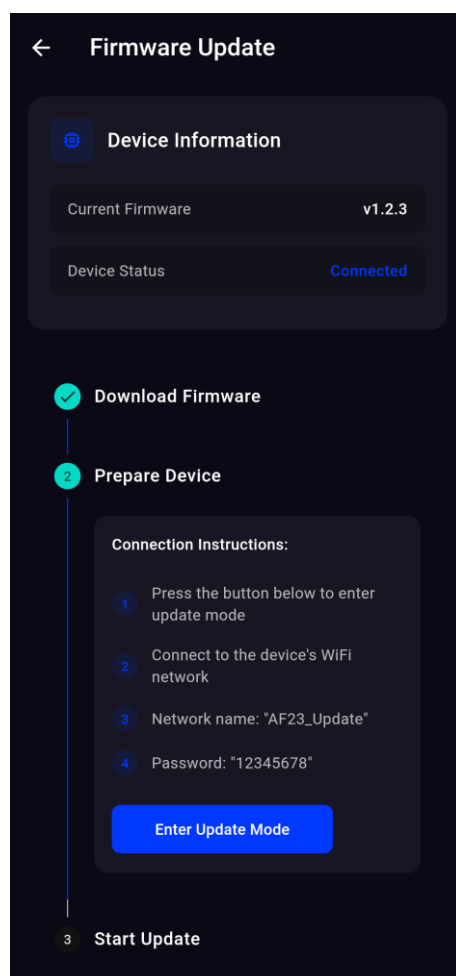
Obrázek 69 - Detaily jízdy v mobilní aplikaci

Systém analýz se bude snažit stále vylepšovat na základě budoucích reálných jízd. V budoucnu plánujeme přidat více možností filtrování, podrobnější grafy a analýzu dat z gyroskopu a akcelerometru.

4.5 Aktualizace řídicí jednotky

Jednou z nejdůležitějších funkcí naší aplikace je systém pro vzdálenou aktualizaci firmwaru řídicí jednotky. K řídicí jednotce se nelze dostat bez odmontování několika částí formule. Proto jsme navrhli systém, který umožňuje aktualizaci firmwaru na dálku, což značně zjednodušuje údržbu a umožňuje snadnou správu verzí softwaru.

Celý proces funguje následujícím způsobem: V aplikaci máme sekci, která je určená k aktualizacím firmwaru. Když ji otevřeme, zobrazí se nám možnost stáhnout a nainstalovat nový firmware. Po kliknutí na tlačítko „**Zkontrolovat aktualizace**“ aplikace provede kontrolu, zda je k dispozici nová verze firmwaru. Pokud ano, začne proces stahování a instalace.



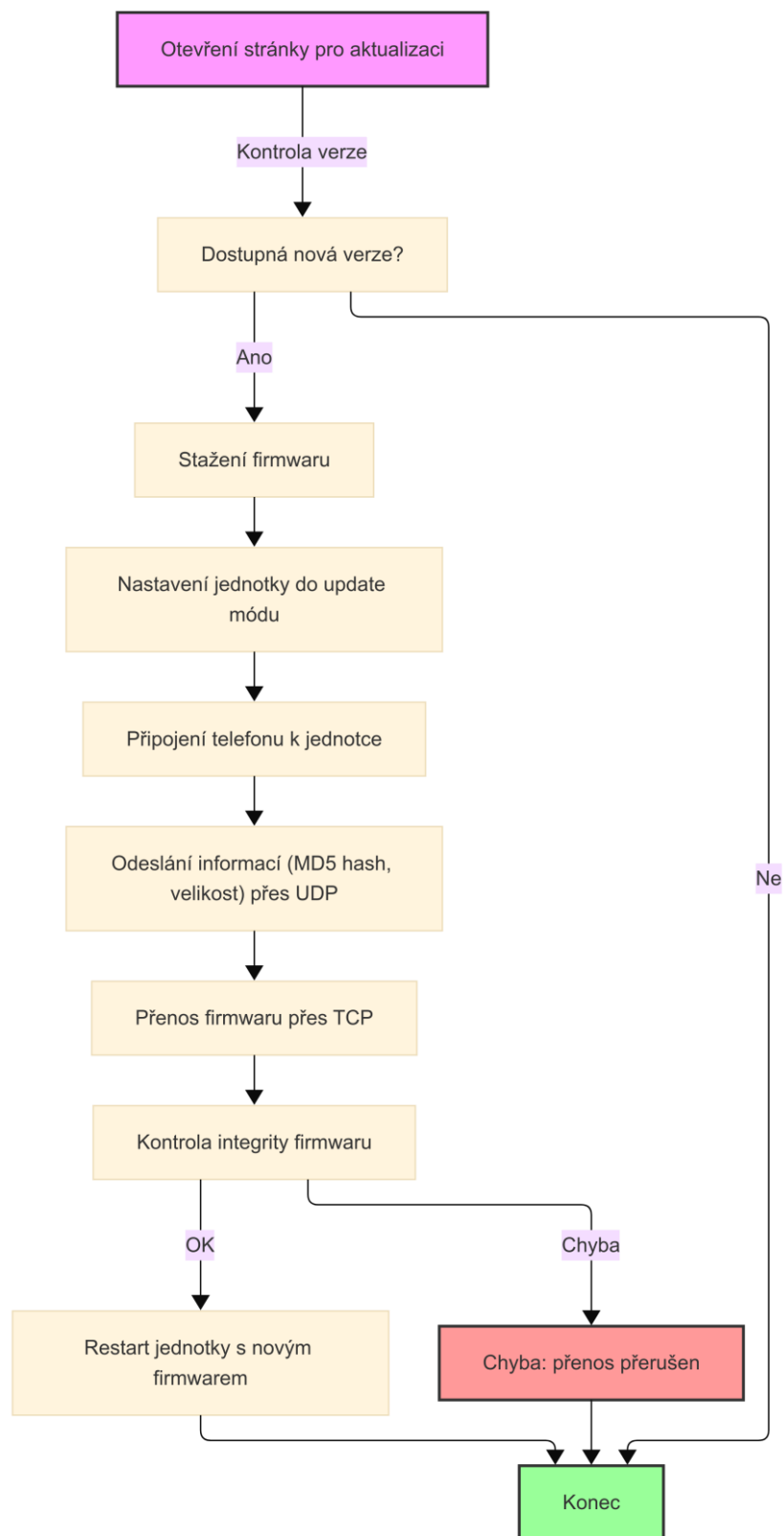
Obrázek 70 - Obrazovka pro aktualizaci řídicí jednotky

Firmware je uložen na backendu, který běží na open-source řešení PocketBase. Toto řešení nám umožňuje snadno spravovat a nahrávat nové verze firmwaru. Jakmile je nový firmware nahrán do backendu, můžeme ho stáhnout přímo v aplikaci a připravit na instalaci do řídicí jednotky.

Když uživatel zahájí aktualizaci, aplikace automaticky nastaví řídicí jednotku do tzv. "update módu". Tento režim aktivuje Wi-Fi hotspot na jednotce, který umožní připojení k zařízení bez potřeby připojení k veřejné síti. Poté nás aplikace provede kroky pro připojení telefonu k Wi-Fi síti, která je vytvořena řídicí jednotkou, pomocí instrukcí, které se zobrazí na obrazovce.

Prvním krokem po připojení je odeslání základních informací o firmwaru, jako je jeho velikost a MD5 hash pro kontrolu integrity souboru. Tato data jsou odeslána přes UDP protokol. Tento protokol byl zvolen na základě knihovny, kterou jsme použili při vývoji, protože nám poskytl jednoduchý a rychlý způsob pro tuto kontrolu. Po úspěšné validaci těchto informací je přenos firmwaru dokončen pomocí TCP protokolu. TCP byl zvolen pro přenos samotného souboru firmwaru, protože je spolehlivý a zajišťuje správné doručení dat.

Použití Wi-Fi hotspotu pro přenos dat bylo rozhodnuto z důvodu rychlosti a stability přenosu. Wi-Fi nabízí vyšší přenosovou rychlost než Bluetooth a zajišťuje stabilní připojení, což je pro tento typ aktualizace zásadní.



Obrázek 71 - Postup OTA aktualizace firmwaru

5 SPOILER

Spoiler jsme se rozhodli přepracovat, protože originální verze nebyla schopná vydržet silné vibrace způsobené faktem, že formule není odpružena. Při jízdě na nerovném povrchu proto praskal. Spoiler je hlavně estetickým doplňkem.



Obrázek 72 - Prasklý spoiler

V naší nové verzi jsme spoiler zesílili, hlavně v místech šroubových spojů, které byly problémové, a také jsme přidali adresovatelný RGB LED pásek, který je schopen zobrazovat jakékoliv světelné efekt. Ty můžou signalizovat například brždění, couvání, zatáčení, nebo různé animace které reagují na stav formule. Pásek je založený na LED **WS2812B**.

Křídlo spoileru je složeno ze čtyř velkých částí, které jsou ze stran zakryty bočnicemi. Všechny části spojují čtyři hliníkové výztuhy s vnějším průměrem 8 mm, které jsou na koncích opatřeny závity pro zašroubování špuntů. Pro provedení vodičů z LED pásku je k dispozici kanál uvnitř spoileru.



Obrázek 73 - Model nového spoileru

Ani pevnější plastové části ale vibracím neodolaly, byla proto vytvořena další verze zpevněná nerezovými výztuhami vypálenými na laserové řezačce.

6 VÝROBA A OSAZOVÁNÍ PLOŠNÝCH SPOJŮ

6.1 Návrh a testování zapojení

Prvním krokem výroby plošných spojů byl návrh zapojení. Po návrhu zapojení bylo potřeba je řádně otestovat. Pro volant bylo testování jednoduché, jednotlivé části obvodů (jako například řízení LED posuvnými registry) byly testovány na nepájivých polích a případně upraveny.

Návrh obvodů řídicí jednotky byl ovšem složitější záležitostí. Jednoduché obvody byly opět zapojeny nejprve na nepájivých polích, pro rozsáhlejší testování tento postup ovšem nebyl dostatečný. Ručně jsme proto vytvořili **7 plošných spojů**, každý z nich slouží k testování určitých částí řídicí jednotky. Tato metoda umožnila levnější a rychlejší výrobu plošných spojů pro prototypování oproti profesionální výrobě.

Pro ruční výrobu tištěných spojů byla využita metoda přenosu toneru (tzv. toner transfer). Na speciální papír se pomocí laserové tiskárny vytiskne pozitiv plošného spoje, ten je následně přezhelen na desku s měděným povrchem (tzv. cuprexitu). Po aplikaci tepla a tlaku se toner uvolní z papíru a přenesení se na plošný spoj.



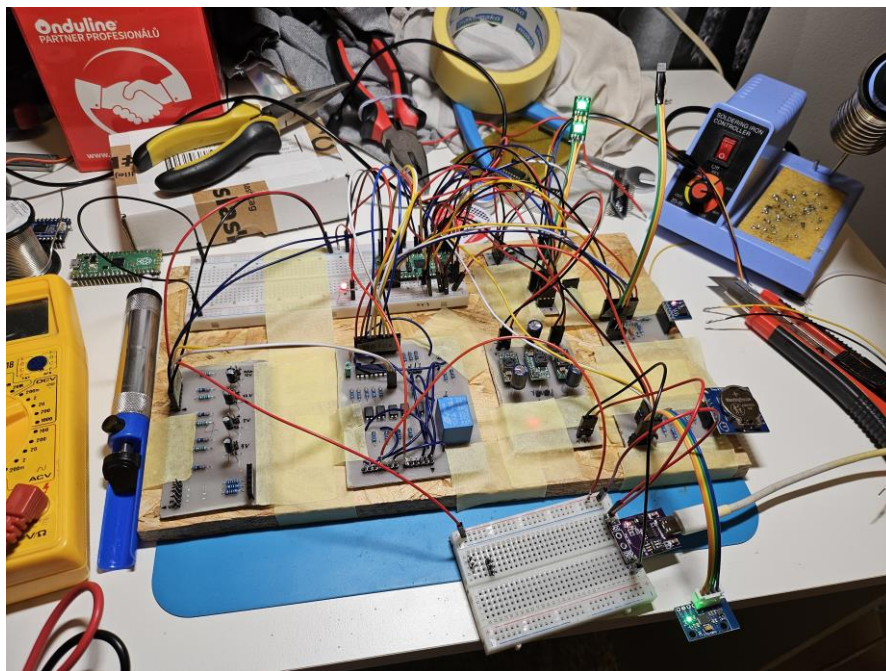
Obrázek 74 - Desky cuprexitu s nažehleným motivem

Tyto nažehlené spoje se následně vyleptají pomocí roztoku chloridu železitého, ten vyleptá všechnu měď, která není zakryta tonerem. Roztok je běžně dostupný, při práci s ním je ovšem třeba dodržovat bezpečnostní zásady. Toner z vyleptané desky se následně očistí acetonem, do plošného spoje se vyvrtají díry pro průchozí součástky a spoj je připravený k pájení.



Obrázek 75 - Plošný spoj po vyleptání chloridem železitým a očištění

Po výrobě a osazení všech testovacích plošných spojů byly propojeny k mikrokontroleru a prototyp byl připraven k testování.



Obrázek 76 - Prototyp řídicí jednotky

6.2 Testovací software

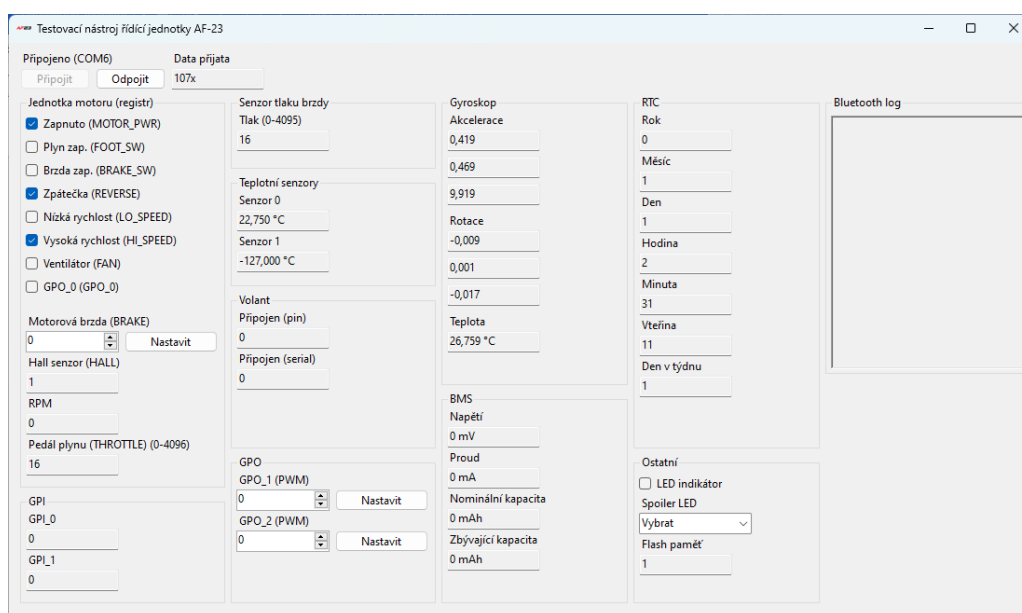
Pro otestování jednotlivých funkcí volantu byl vytvořen jednoduchý firmware, který rozsvítí displej, stav tlačítek a enkodéru odesílá přes USB sériovou linku do počítače a podle rotace enkodéru postupně rozsvěcuje jednotlivé LED. Tento firmware sloužil k otestování jednotlivých obvodů volantu, a poté i k ověření funkčnosti finálního plošného spoje.



Obrázek 77 - Prototyp volantu s testovacím firmwarem

Testování funkcí řídicí jednotky opět bylo složitější. Proto jsme vytvořili testovací firmware s ovladači jednotlivých obvodů jednotky. Tento firmware využívá Bluetooth pro komunikaci s počítačem a odesílá do něj hodnoty ze všech senzorů, které má řídicí jednotka k dispozici. Zároveň reaguje na příkazy z počítače a ovládá všechny funkce.

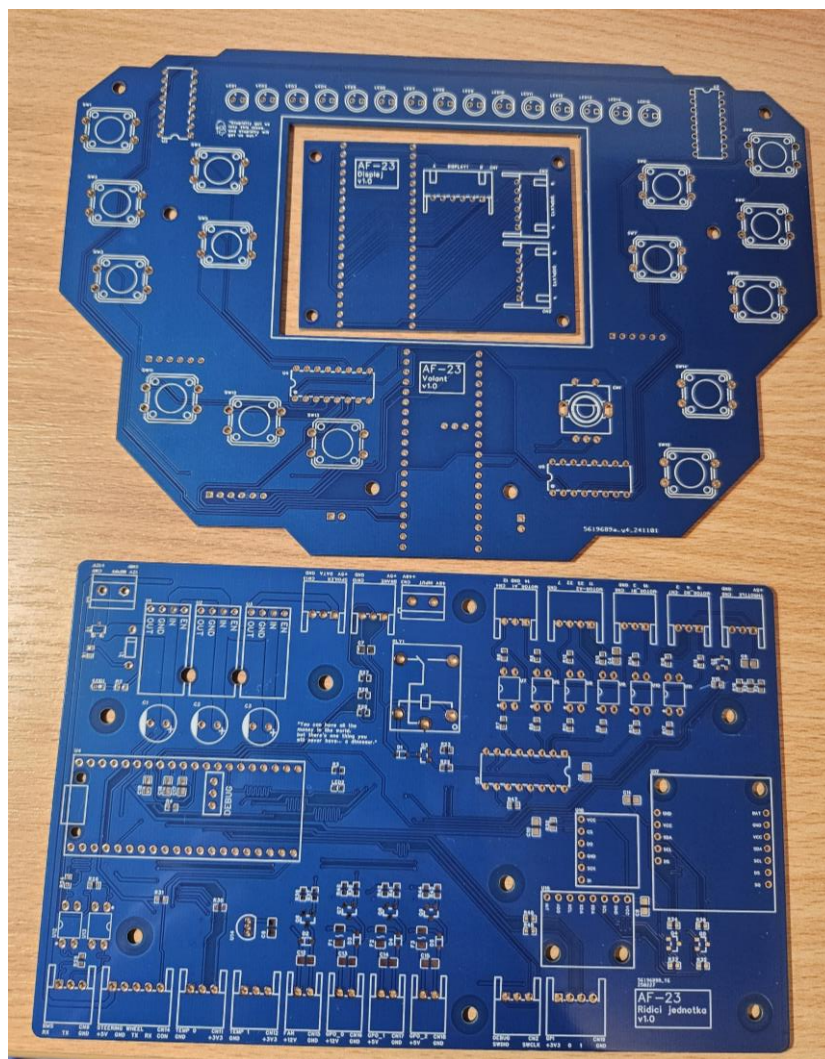
Pro komunikaci s tímto testovacím firmwarem byl vytvořen program pomocí jazyku C# a knihovny Windows Forms.



Obrázek 78 - Testovací nástroj řídicí jednotky

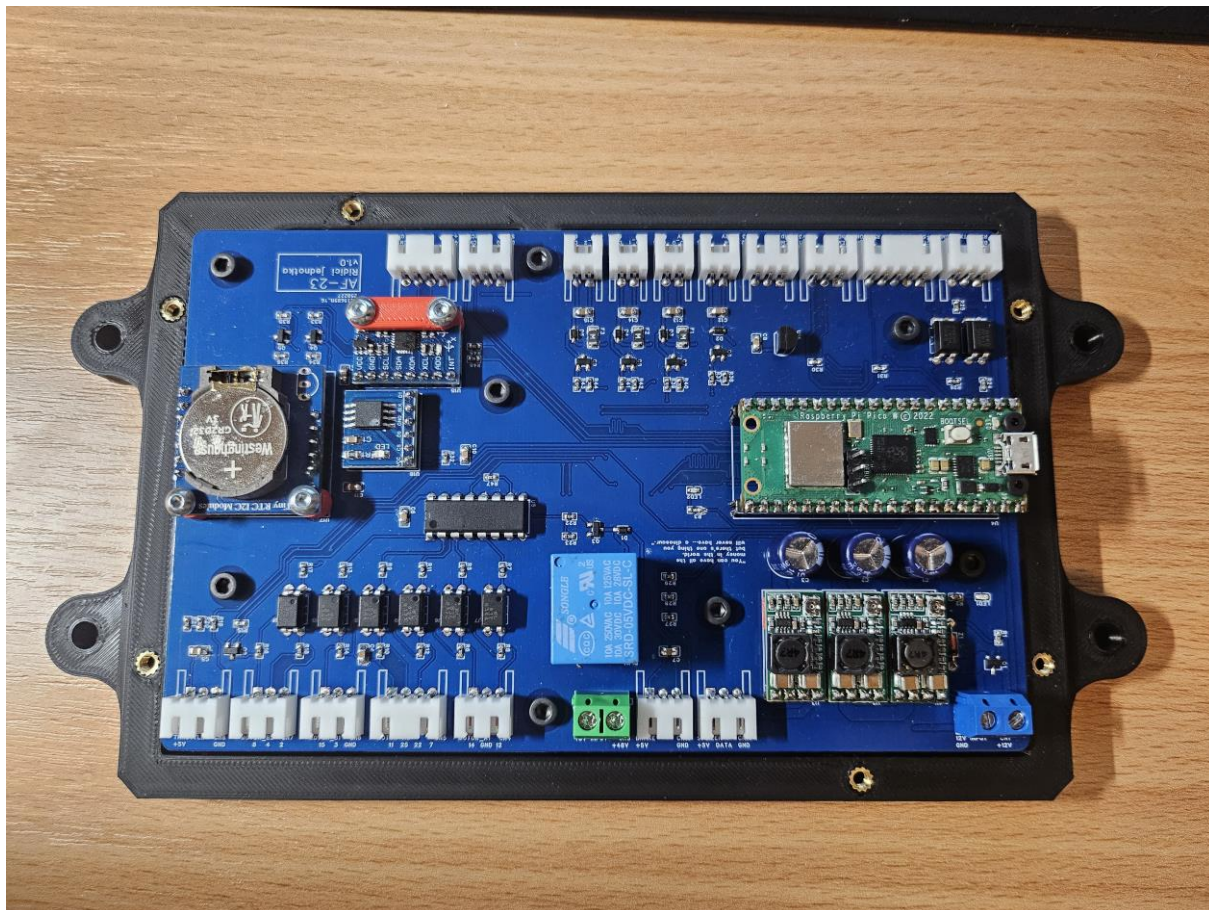
6.3 Výroba finálních plošných spojů

Po ověření všech zapojení jsme navrhli rozložení finálních plošných spojů, upravili vrstvu popisků tak, aby všechny konektory a součástky byly jasně popsány a data ve formátu gerber jsme zaslali do výroby. Pro výrobu jsme využili společnost JLCPCB, protože jsme s ní už byli seznámeni, a protože vyrobené spoje jsou kvalitní za přívětivou cenu.



Obrázek 79 - Vyrobené plošné spoje

Plošné spoje bylo potřeba osadit. Je dobré začít osazováním SMD součástek, dokud je spoj možné položit na rovný povrch. Čištění spoje od pájecí pasty je také jednodušší, pokud na spoji nejsou přítomny ostatní součástky. Po napájení SMD součástek jsme napájeli i ostatní součástky a spoje otestovali.

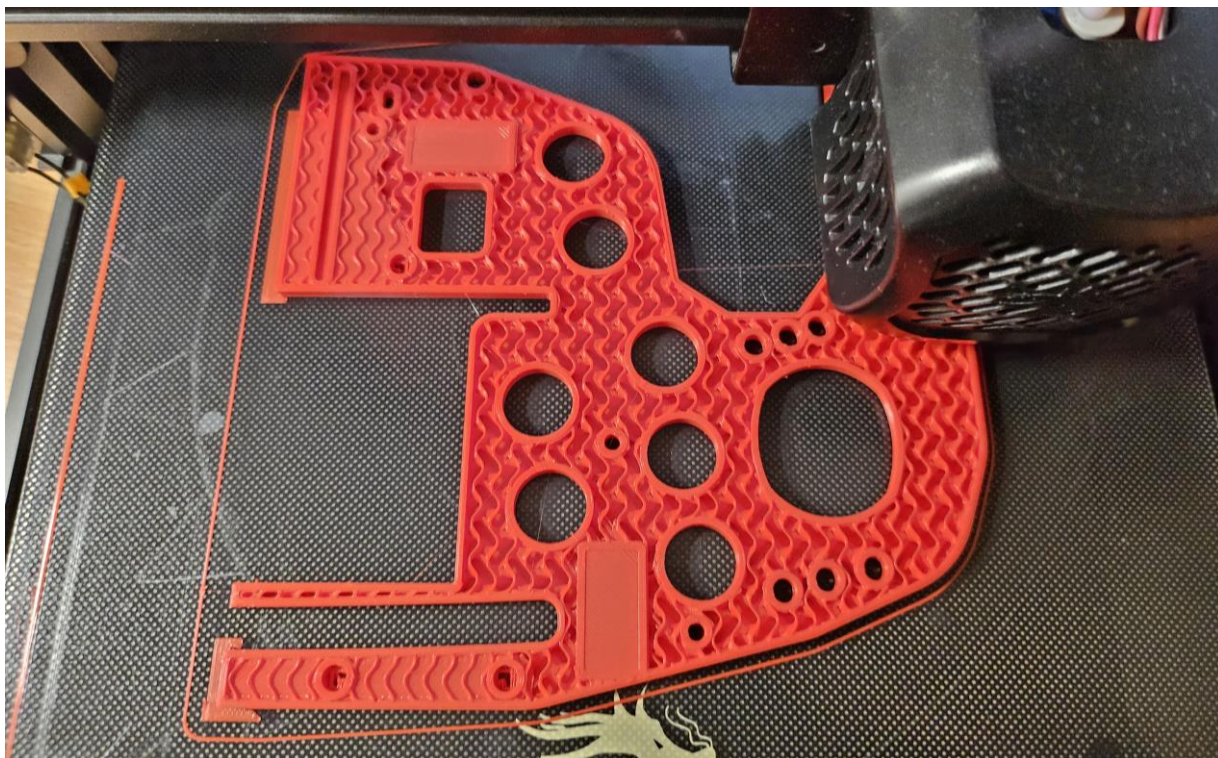


Obrázek 80 - Osazený plošný spoj řídicí jednotky

7 VÝROBA A SESTAVENÍ TIŠTĚNÝCH SOUČÁSTÍ

7.1 Volant

Nejprve bylo třeba vytisknout všechny součásti na 3D tiskárně. Volant se skládá pouze z devíti různých částí, některé z nich bylo samozřejmě třeba vytisknout na víc pokusů, kvůli chybám v návrhu nebo v tolerancích.



Obrázek 81 - Výztuha volantu při tisku

Po vytisknutí a odstranění podpěr ze všech dílů byly vloženy závitové vložky. Vložky se do plastových dílů natavují pomocí hrotu pájecí stanice, který nahřeje plast v okolí. Vytvoří se tak v dílu pevný závit.

Pro složení celého volantu jsme použili celkově:

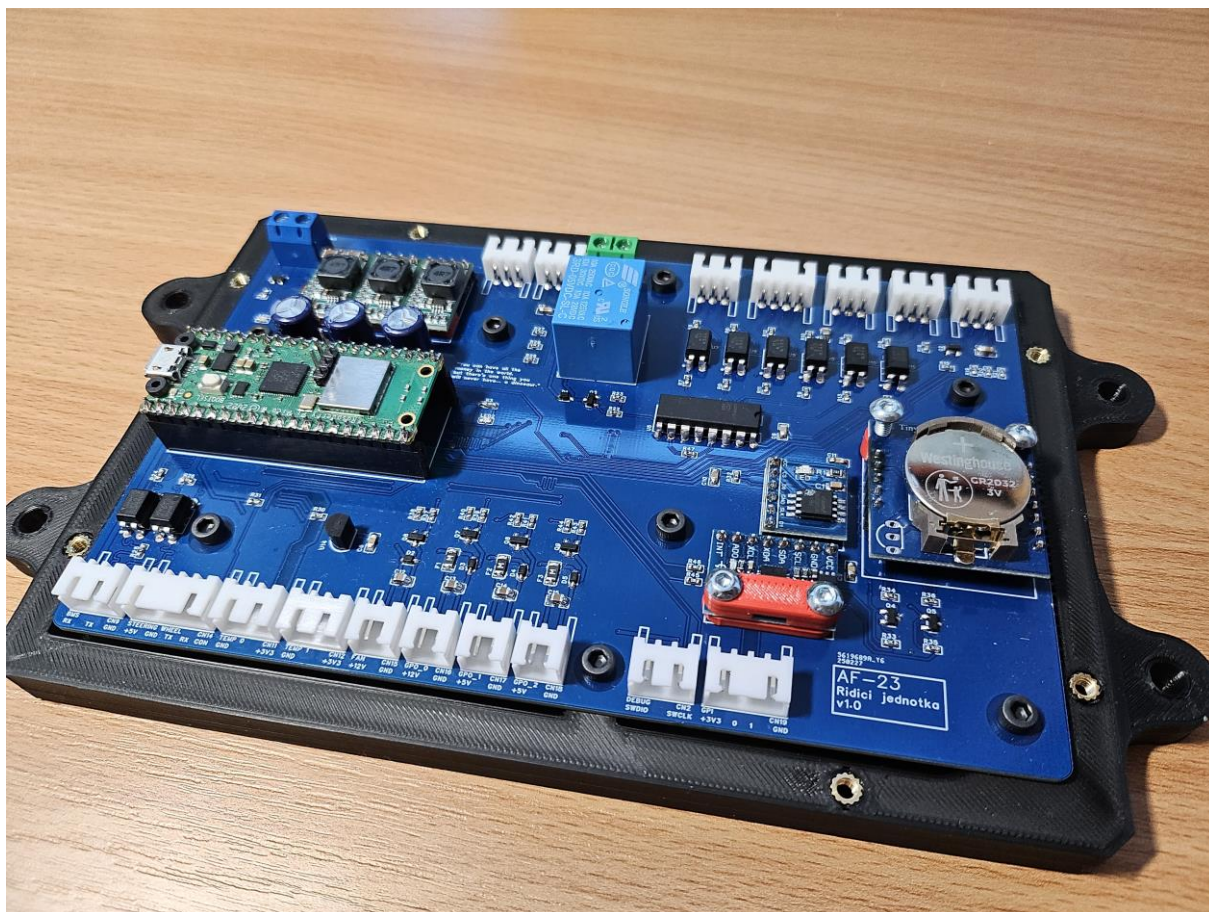
- 6x šroub DIN 912 M3x5
- 18x šroub DIN 7991 M3x8
- 8x šroub DIN 7991 M3x20
- 12x šroub DIN 7991 M3x16
- 42x závitová vložka M3
- 4x šroub DIN 912 M2,5x12



Obrázek 82 - Složený prototyp volantu

7.2 Řídící jednotka

Řídící jednotka se skládá pouze ze tří dílů, její složení je tak jednoduché. Po vytisknutí všech dílů a vložení závitových vložek stačí pouze přišroubovat plošný spoj ke spodní části. Po připojení konektorů se přišroubuje víko, to zároveň zajišťuje konektory.



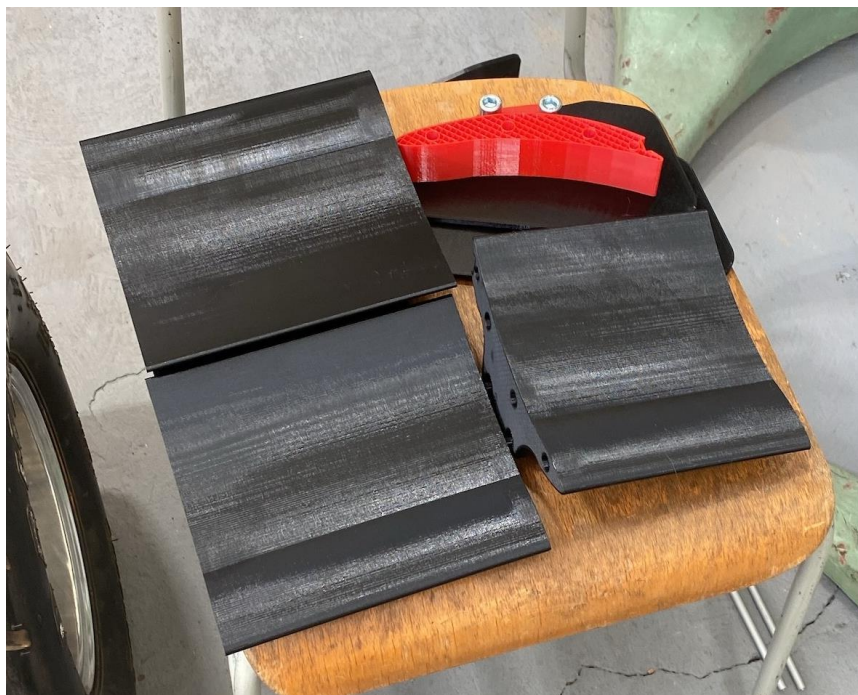
Obrázek 83 - Řídící jednotka bez víka

7.3 Spoiler

Prvním krokem při skládání spoileru je nasunutí všech částí na hliníkové výztuhy. Zároveň je třeba provést vodiče pro LED pásek skrz připravený kanál.

Po přišroubování špuntů na bocích spoileru jsme z LED pásku odloupili ochrannou vrstvu a přilepili ho ke spoileru.

Posledním krokem je přišroubování držáků ke spoileru a k rámu formule.



Obrázek 84 - Vytisknuté díly pro spoiler

ZÁVĚR

Všechny cíle naší práce se nám podařilo splnit. Vytvořili jsme tak lepší způsob ovládání elektroformule a zároveň umožnili sběr analytických dat o stavech jejích různých částí. Řidičovi jsou tak nyní v reálném čase k dispozici údaje jako teploty komponent, rychlost, výkon nebo stav baterií. Tyto údaje jsou dostupné k zobrazení nejen ve volantu, ale také v mobilní aplikaci pro další osoby.

Vzhled volantu byl inspirován reálnými volanty používanými ve vozech Formule 1. Jeho plošný spoj byl navržen tak, aby bylo možné jeho využití i pro jiné účely, například pro počítačové závodní simulátory. K tomu by samozřejmě bylo potřeba přidat základnu a upravit jeho firmware, tak aby se volant pro počítač tvářil jako herní ovladač. Fyzický design volantu je pro tento účel také použitelný.

Nejnáročnější částí práce byl vývoj řídicí jednotky, ve které je integrováno mnoho funkcí. Stejně jako při návrhu zapojení volantu jsme při vývoji začínali od nuly. Kvůli její složitosti bylo její testování náročné a museli jsme se naučit proces domácí výroby plošných spojů. Některé obvody, například připojení k BMS, trvalo déle vytvořit, protože jejich vnitřní fungování není známo. Řídicí jednotku jsme zároveň připravili pro případné budoucí rozšíření pomocí rezervních vstupů a výstupů

Při práci jsme výrazně rozšířili svoje obzory v oblastech embedded systémů a jejich programování, 3D tisku a vývoje mobilních aplikací. Práce na formuli také zahrnovala mnoho mechanické práce, údržby a lehké úpravy její konstrukce.

POUŽITÁ LITERATURA

- [1] RASPBERRY PI. *Pico-series Microcontrollers*. Online. Raspberry Pi Documentation. C2012-2025. Dostupné z: <https://www.raspberrypi.com/documentation/microcontrollers/pico-series.html>. [cit. 2025-02-23].
- [2] WAVESHARE. *Pico-ResTouch-LCD-2.8*. Online. Waveshare Wiki. Dostupné z: <https://www.waveshare.com/wiki/Pico-ResTouch-LCD-2.8>. [cit. 2025-02-24].
- [3] ČERNÝ, Michal. *Základní zapojení – Ochrana proti přepólování napájení*. Online. RoboDoupě. 2015. Dostupné z: <https://robodoupe.cz/2015/zakladni-zapojeni-ochrana-proti-prepolovani-napajeni/>. [cit. 2025-03-03].
- [4] RASPBERRY PI LTD. *Raspberry Pi Pico Datasheet*. Online. 15.10.2024. 2024. Dostupné z: <https://datasheets.raspberrypi.com/pico/pico-datasheet.pdf>. [cit. 2025-03-05].
- [5] VBSEMI. *SI2323CDS-T1-GE3-VB Datasheet*. Online. Dostupné z: <https://www.vbsemi.com/Package/SOT23-3/SI2323CDS-T1-GE3-VB.pdf>. [cit. 2025-03-05].
- [6] *Mikro step-down měnič, nastavitelný*. Online. Láskakit. C2025. Dostupné z: <https://www.laskakit.cz/mikro-step-down-menic--nastavitelny/>. [cit. 2025-03-05].
- [7] KELLY CONTROLS. *Kelly Cheetah series KLS-S Controllers User Manual*. Online. 2.12. 2025. Dostupné z: <https://media.kellycontroller.com/new/Kelly-Cheetah-series-KLS-NUserManualV2.12.pdf>. [cit. 2025-03-06].
- [8] TEXAS INSTRUMENTS. *SNx4HC165 8-Bit Parallel-Load Shift Registers*. Online. H. 2015. Dostupné z: <https://www.ti.com/lit/ds/symlink/sn74hc165.pdf>. [cit. 2025-03-06].
- [9] *Co je to BMS?* Online. ISwap. 2024. Dostupné z: <https://www.iswap.cz/nase-novinky/co-to-je-bms/>. [cit. 2025-03-06].
- [10] *LED pásek WS2812 (2811) chip 5050, 18W/m, RGB, 60 diod, 5V, 1 m, černý*. Online. SOH. C2016. Dostupné z: <http://eshop.soh.cz/led-pasky-5-v/i983-led-pasek-ws2812-2811-chip-5050,-18w/m,-rgb,-60-diod,-5v,-1-m,-cerny>. [cit. 2025-03-06].
- [11] *Software board general protocol V4. V3*. Dostupné také z: https://cdn.shopifycdn.net/s/files/1/0606/5199/5298/files/JDB_RS485-RS232-UART-Bluetooth-Communication_Protocol.pdf?v=1682577935.

SEZNAM OBRÁZKŮ

Obrázek 1 - Blokové schéma elektroniky volantu	9
Obrázek 2 - Mikrokontroler Raspberry Pi Pico W. <i>Raspberry Pi Pico W</i> . Online. In: RPishop.cz. C2025. Dostupné z: https://rpishop.cz/raspberry-pi-pico/5073-raspberry-pi-pico-w.html . [cit. 2025-03-03].....	10
Obrázek 3 - Displej pro Raspberry Pi Pico. WAVESHARE. <i>2.8inch Touch Display Module for Raspberry Pi Pico, 262K Colors, 320×240, SPI</i> . Online. Waveshare. C2025. Dostupné z: https://www.waveshare.com/pico-restouch-lcd-2.8.htm . [cit. 2025-02-24].....	11
Obrázek 4 - Displej pro Raspberry Pi Pico. WAVESHARE. <i>2.8inch Touch Display Module for Raspberry Pi Pico, 262K Colors, 320×240, SPI</i> . Online. Waveshare. C2025. Dostupné z: https://www.waveshare.com/pico-restouch-lcd-2.8.htm . [cit. 2025-02-24].....	11
Obrázek 5 - Mikrospínač TC-1212T. DRÁTEK.CZ. <i>Mikrospínač TC-1212T - 12 x 12 x 7,3 mm SMD PCB DPS</i> . Online. Drátek.cz. Dostupné z: https://dratek.cz/arduino/1180-mikrospinač-tc-1212t-12x12x7.3mm-smd-pcb-dps.html . [cit. 2025-02-24].	11
Obrázek 6 - Zapojení mikrospínače	11
Obrázek 7 - Schéma funkce inkrementálního enkodéru. NEMATIC!. <i>How Rotary Encoder Works With Arduino!</i> . Online. Instructables. Dostupné z: https://www.instructables.com/How-to-Use-an-Rotary-Encoder-With-Arduino/ . [cit. 2025-02-24].	12
Obrázek 8 - Rotační enkodér EC11. <i>EC11E15204A3</i> . Online. In: Alps Alpine. C2025. Dostupné z: https://tech.alpsalpine.com/e/products/detail/EC11E15204A3/ . [cit. 2025-02-24].	12
Obrázek 9 - Zapojení rotačního enkodéru	12
Obrázek 10 - Schéma fungování SIPO posuvného registru	14
Obrázek 11 - Schéma fungování PISO posuvného registru	14
Obrázek 12 - Zapojení ochrany proti přepólování a zkratu na volantu.....	15
Obrázek 13 - Schéma připojení volantu k řídicí jednotce	15
Obrázek 14 - 3D náhled přední strany hlavního plošného spoje volantu.....	16
Obrázek 15 - 3D náhled zadní strany hlavního plošného spoje volantu.	16
Obrázek 16 - 3D náhled plošného spoje pro displej volantu	17
Obrázek 17 - Architektura firmwaru volantu	19
Obrázek 18 - Návrh menu volantu	20
Obrázek 19 - Návrh hlavní obrazovky volantu.....	21
Obrázek 20 - Rozložení tlačítek volantu a jejich pořadová čísla.....	22
Obrázek 21 - Návrh obrazovky se stavem baterií	23
Obrázek 22 - Obrazovka s časovačem.....	24
Obrázek 23 - Návrh obrazovky s teplotami komponent.....	25
Obrázek 24 - Obrazovka s časy akcelerací a statistikami.....	26
Obrázek 25 - Hra na volantu pro jeho demonstraci	27
Obrázek 26 - Přední pohled na volant.....	28
Obrázek 27 - Zadní pohled na volant.....	28
Obrázek 28 - CAD model základní desky volantu.....	29
Obrázek 29 - Záda volantu	30

Obrázek 30 - CAD model volantu s výztuhou.....	31
Obrázek 31 - Pohled na zadní stranu výztuhy volantu.....	31
Obrázek 32 - CAD model volantu s předním krytem	32
Obrázek 33 - Přední strana držadla volantu.....	33
Obrázek 34 - Zadní strana držadla volantu.....	33
Obrázek 35 - Příruba pro rychloupínací náboj.....	34
Obrázek 36 - Rychloupínací náboj.....	35
Obrázek 37 - Konektor do rychloupínacího náboje	35
Obrázek 38 - Blokové schéma elektroniky řídicí jednotky	36
Obrázek 39 - Zapojení ochrany proti přepólování a LED indikátoru napájení	37
Obrázek 40 - Použitý step-down měnič. <i>Mikro step-down měnič, nastavitelný</i> . Online. In: Láskakit. C2025. Dostupné z: https://www.laskakit.cz/mikro-step-down-menic--nastavitelny/ . [cit. 2025-03-05].	38
Obrázek 41 - Zapojení posuvného registru pro ovládání digitálních vstupů jednotky motoru	40
Obrázek 42 - Zapojení jednoho z digitálních vstupů jednotky motoru pomocí optočlenu	40
Obrázek 43 - Zapojení relé pro zapnutí jednotky motoru	41
Obrázek 44 - Jednotka motoru Kelly KLS7230S. <i>Kelly KLS7230S,24V-72V,300A, Sine Wave Brushless Motor Controller</i> . Online. In: EV Components Store. C2025. Dostupné z: https://www.evcomponents.com/cls7230s-300a-72v-sine-wave-brushless-controller.html . [cit. 2025-03-06].	41
Obrázek 45 - Čtení napětí pedálu plynu pomocí děliče napětí.	42
Obrázek 46 - Zapojení pro ovládání analogového vstupu brzdy	42
Obrázek 47 - Zapojení pro čtení stavu hallových senzorů	42
Obrázek 48 - Zapojení tlakového senzoru	43
Obrázek 49 - Tlakový senzor s rozsahem 0-1,2 MPa. <i>Tlakový senzor 0-1,2MPa, závit G1/4</i> . Online. In: Hadex. C2011-2025. Dostupné z: https://www.hadex.cz/m553a-tlakovy-senzor-0-12mpa-zavit-g14/ . [cit. 2025-03-06].	43
Obrázek 50 - Zapojení galvanicky odděleného převodníku napětí pro komunikaci s BMS....	44
Obrázek 51 - Teplotní senzor DS18B20. <i>Digitální teplotní senzor DALLAS – DS18B20</i> . Online. In: Drátek.cz. Dostupné z: https://dratek.cz/arduino/1187-teplotni-senzor-digitalni-dallas-ds18b20.html . [cit. 2025-03-06].	45
Obrázek 52 - Zapojení flash paměti	46
Obrázek 53 - Modul flash paměti. <i>Winbond W25Q32 Flash 32Mb SPI Modul</i> . Online. In: Láskakit. C2025. Dostupné z: https://www.laskakit.cz/winbond-w25q32-flash-32mb-spi-modul/ . [cit. 2025-03-06].	46
Obrázek 54 - RTC modul. <i>Mini RTC Hodiny reálného času DS1307</i> . Online. In: Láskakit. C2025. Dostupné z: https://www.laskakit.cz/arduino-mini-rtc-hodiny-realneho-casu-ds1307/ . [cit. 2025-03-06].	47
Obrázek 55 - Zapojení RTC modulu.....	47
Obrázek 56 - Zapojení akcelerometru a gyroskopu	48
Obrázek 57 - Zapojení výstupu pro ventilátor jednotky motoru.....	49
Obrázek 58 - Zapojení jednoho z rezervních výstupů.....	50
Obrázek 59 - Zapojení rezervních vstupů.....	50
Obrázek 60 - Vrchní strana plošného spoje řídicí jednotky	51

Obrázek 61 - Spodní část pouzdra řídicí jednotky	57
Obrázek 62 - Pouzdro řídicí jednotky s plošným spojem.....	57
Obrázek 63 - Víko pouzdra řídicí jednotky	58
Obrázek 64 - Zapouzdřená řídicí jednotka	58
Obrázek 65 - Hlavní obrazovka mobilní aplikace.....	60
Obrázek 66 - Obrazovka pro výběr světelných efektů	62
Obrázek 67 - Obrazovka s historií jízd formule.....	63
Obrázek 68 - Detaily jízdy v mobilní aplikaci.....	64
Obrázek 69 - Detaily jízdy v mobilní aplikaci.....	64
Obrázek 70 - Obrazovka pro aktualizaci řídicí jednotky.....	65
Obrázek 71 - Postup OTA aktualizace firmwaru.....	67
Obrázek 72 - Prasklý spoiler.....	68
Obrázek 73 - Model nového spoileru.....	69
Obrázek 74 - Desky cuprexitu s nažehleným motivem.....	70
Obrázek 75 - Plošný spoj po vyleptání chloridem železitým a očištění.....	71
Obrázek 76 - Prototyp řídicí jednotky	71
Obrázek 77 - Prototyp volantu s testovacím firmwarem.....	72
Obrázek 78 - Testovací nástroj řídicí jednotky	72
Obrázek 79 - Vyrobené plošné spoje	73
Obrázek 80 - Osazený plošný spoj řídicí jednotky	74
Obrázek 81 - Výztuha volantu při tisku	75
Obrázek 82 - Složený prototyp volantu.....	76
Obrázek 83 - Řídicí jednotka bez víka	77
Obrázek 84 - Vytisknuté díly pro spoiler	78

SEZNAM ROVNIC

Rovnice 1 - Výpočet hodnoty rezistoru.....	13
Rovnice 2 - Výpočet celkové hodnoty dvou paralelně zapojených rezistorů	13

SEZNAM TABULEK

Tabulka 1 - Hodnoty rezistorů pro LED na volantu.....	13
Tabulka 2 - Příkazy pro komunikaci s BMS.....	54

SEZNAM PŘÍLOH

Příloha 1 - Schéma elektroniky plošného spoje základní desky volantu
Příloha 2 - Schéma elektroniky plošného spoje displeje volantu
Příloha 3 - Schéma elektroniky plošného spoje řídicí jednotky