

Vývoj softwaru zobrazovací jednotky motocyklu

Bakalářská práce

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: 2612R011 – Elektronické informační a řídicí systémy

Autor práce: Radim Luděk Šmíd

Vedoucí práce: Ing. Lukáš Krčmář





Motorcycle display unit software development

Bachelor thesis

Study programme: B2612 – Electrical Engineering and Informatics

Study branch: 2612R011 – Electronic Information and Control Systems

Author: Radim Luděk Šmíd

Supervisor: Ing. Lukáš Krčmář





TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Zadání bakalářské práce

Vývoj softwaru zobrazovací jednotky motocyklu

Jméno a příjmení: Radim Luděk Šmíd
Osobní číslo: M18000048
Studijní program: B2612 Elektrotechnika a informatika
Studijní obor: Elektronické informační a řídicí systémy
Zadávací katedra: Ústav mechatroniky a technické informatiky
Akademický rok: 2020/2021

Zásady pro vypracování:

1. Proveďte rešerši designu grafických zobrazovacích jednotek moderních jednostopých vozidel.
2. Vyberte vhodnou testovací platformu pro testování příslušné zobrazovací jednotky.
3. Navrhněte software zobrazovací jednotky.
4. Otestujte tuto zobrazovací jednotku v podmínkách provozu.

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

dle potřeby dokumentace
30–40
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] BANZI, Massimo a Michael SHILOH. *Getting started with Arduino*. Third edition. Sebastopol, CA: MakerMedia, [2015]. ISBN 1449363334.
- [2] UPTON, Eben a Gareth HALFACREE. *Raspberry Pi: uživatelská příručka. 2.*, aktualizované vydání. Přeložil Jakub GONER. Brno: Computer Press, 2016. ISBN 978-80-251-4819-8.
- [3] NORRIS, Donald. *Raspberry Pi: projekty*. Brno: Computer Press, 2015. ISBN 978-80-251-4346-9.

Vedoucí práce:

Ing. Lukáš Krčmář
Ústav mechatroniky a technické informatiky

Datum zadání práce:

9. října 2020

Předpokládaný termín odevzdání: 17. května 2021

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 9. října 2020

Prohlášení

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

Jsem si vědom toho, že na mou bakalářskou/diplomovou/rigorózní/disertační 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 bakalářské/diplomové/rigorózní/disertační práce pro vnitřní potřebu Technické univerzity v Liberci. Užiji-li bakalářskou/diplomovou/rigorózní/disertační 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 bakalářská/diplomová/rigorózní/disertační 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

Datum:

Podpis:

Abstrakt

Bakalářská práce popisuje realizaci zobrazovače přístrojové desky pro elektrický motocykl, který komunikuje s jednotlivými komponenty pomocí mikropočítače Raspberry Pi. Tato práce obsahuje několik následujících úkolů. První část práce se zaměřuje na rešerši jednotlivých grafických designů zobrazovacích jednotek moderních jednostopých vozidel. V druhé části práce autor volí příslušnou hardwarovou konfiguraci a popisuje veškeré použité komponenty. Následně autor vyvíjí software vlastního řešení celkové přístrojové desky a v závěrečné části zhodnocuje výsledná naměřená data z testovacích jízd.

Klíčová slova

motocykl, palubní počítač, zobrazovací jednotka, LCD, Raspberry Pi, Bluetooth, Python

Abstract

The bachelor's thesis describes the development of an displays dashboard for an electric motorcycle, which communicates with individual components using a microcomputer Raspberry Pi. This thesis contains several following tasks. The first part focuses on a research of individual graphic designs of display units of modern single-track vehicles. In the second part the author chooses the hardware configuration and describes all the components he used. Then the author develops the software of his own solution of the overall dashboard and finally he evaluates the resulting measured data from test drives.

Keywords

motorcycle, on-board computer, display unit, LCD, Raspberry Pi, Bluetooth, Python

Obsah

Předmluva	13
1 Rešerše zobrazovacích jednotek	14
1.1 Typy přístrojových desek jednostopých vozidel	14
1.1.1 Přístrojová deska se segmentovým LCD	14
1.1.2 Přístrojová deska s TFT digitálním displejem	15
1.2 Technologie displejů	15
1.3 Návrh designu v rámci bezpečnosti	16
2 Hardware	17
2.1 Výběr mikropočítače	17
2.1.1 Arduino	17
2.1.2 Raspberry Pi	18
2.1.3 Porovnání obou zařízení	19
2.2 Displej pro přístrojovou desku	20
2.3 Smart BMS 28s	21
2.4 Hallův senzor NJK-5002A	21
2.5 Ovládací panel Duke 790	22
2.6 Teplotní čidlo DS18B20	22
3 Komunikace Raspberry Pi s danými prvky	23
3.1 BMS komunikace	23
3.1.1 Protokol	23
3.1.2 Poskytnuté informace	24
3.1.3 Struktura programu komunikace	25
3.1.4 Kontrolní součet	26
3.2 Hallův senzor	27
3.2.1 Popis signálu	27
3.2.2 Popis programu na zaznamenávání rychlosti	28
4 Návrh palubního počítače	29
4.1 Konfigurace mikropočítače Raspberry Pi	29
4.2 Hlavní program	30
4.2.1 Databáze	31

4.2.2	Konfigurace GPIO pinů	32
4.2.3	Popis funkcí informačních ikon	32
4.3	Návrh grafického prostředí	34
4.4	Grafický design	35
4.4.1	Hlavní panel	36
4.4.2	Funkce panelu nabídky	38
4.4.3	Informační menu	40
5	Testování	41
5.1	Testování v rámci simulace	41
5.2	Porovnání realizovaného principu snímání rychlosti s komerčním tachometrem	42
6	Závěr	43
	Literatura	44

Seznam obrázků

1	Přístrojová deska motocyklu Honda Rebel 300	14
2	Přístrojová deska motocyklu Ducati Multistrada V4 model S	15
3	Deska Arduino UNO	17
4	Deska Raspberry Pi 3 model B V1.2	18
5	Deska PiCAN 2 Duo	20
6	Smart BMS	21
7	Hallův senzor NJK-5002A	21
8	Ovládací panel Duke 790 2018	22
9	Teplotní čidlo DS18B20 s prodlouženým kabelovým spojením	22
10	Diagram topologie digitální komunikace	23
11	Diagram programu Bluetooth komunikace s BMS jednotkou	25
12	Kontrolní součet	26
13	Detekce signálu přítomnosti magnetu Hallova senzoru	27
14	Diagram funkce hlavního programu	30
15	Schéma zapojení Raspberry Pi GPIO pinů s externími zařízeními	32
16	Vývoj ikony pro teplotu motoru v prostředí Inkscape	35
17	Hlavní panel	36
18	Výběrové menu informací	38
19	Funkce nabídky znázorněná pomocí matice	39
20	Informační menu	40
21	Testování v rámci simulace	41

Seznam tabulek

1	Hlavní rozdíly porovnávaných zařízení	19
2	Přijata data při požadavku o jednotlivých napětích na člancích	24
3	Přijata data při požadavku základních informací	24
4	Porovnání vytvořené práce s tachometrem Sigma	42

Seznam zdrojových kódu

1	Definování relevantních bajtů a hodnoty kontrolního součtu	26
2	Funkce pro kontrolu validního paketu	26
3	Přiřazení vlastností senzoru	28
4	Celková funkce „get_pulse“ na výpočet informací o pohybu vozidla	28

Seznam použitých zkratk a symbolů

1–Wire	technologie umožňující připojit více zařízení pomocí jednoho datového kabelu.
ABS	Anti Lock Brake System, protiblokovací systém vozidla
ARM	Acorn RISC Machine, označení architektury procesorů
Bluetooth LE	Bluetooth Low Energy, technologie Bluetooth, která se využívá pro nižší spotřebu provozovaného zařízení
BMS	Battery Management System, systém spravující akumulátor
C	programovací jazyk
C++	vyšší programovací jazyk, původně rozšířený jazyk C
CAN	Controller Area Network, komunikační protokol využívající v automobilech
CPU	Central Processing Unit, centrální procesorová jednotka
DPI	Display Parallel Interface, konektor zapojený paralelně, který se využívá na LCD a podobné zobrazovací technologie
DSI	Display Serial Interface, konektor zapojený sériově, který se využívá na LCD a podobné zobrazovací technologie
FPS	Frames Per Second, snímková frekvence
GATT SDK	Generic Attribute Profile Software Development Kit, protokol definující komunikační datové struktury
GND	Ground, uzemnění
GPIO	General Purpose Input/Output, vstupně výstupní ovladatelné piny mikropočítače
GPU	Graphics Processing Unit, grafická procesorová jednotka
GUI	Graphical User Interface, grafické uživatelské rozhraní
HDMI	High Definition Multi-media Interface, rozhraní, pomocí kterého se distribuuje nekomprimovaný signál v digitálním formátu
ICSP	In Circuit Serial Programming, protokol pro sériové programování mikrokontrolérů.
IDE	Integrated Development Environment, vývojové prostředí
LCD	Liquid Crystal Display, zobrazovací panel, který funguje na principu tekutých krystalů
MAC	Media Access Control, jedinečný identifikátor síťového hardwaru

MCU	Microcontroller
micro SD	Micro Secure Digital, malá paměťová karta používaná v přenosných zařízeních
NO	Normally Open, logika výstupu, která v normální stavu detekuje nulu.
OS	Operating System
RAM	Random Access Memory, operační paměť
RGB24	24bitový barevný formát
RS232	typ standardu sériové komunikace
RS485	typ standardu sériové komunikace
RTC	Real Time Clock
SMBus	System Management Bus, dvou vodičová sběrnice pro komunikaci
SQL	Structured Query Language, jazyk používaný pro práce s databázemi
TFT	Thin Film Transistor, typ LCD displeje
TN	Twisted Nematic, technologie LCD displeje
UART	Universal Asynchronous Receiver Transmitter, sběrnice sloužící k sériovému asynchronnímu přenosu dat
USB	Universal Serial Bus, univerzální sériová sběrnice používaná většinou u počítačů pro přenos dat

Předmluva

Tato bakalářská práce pojednává o návrhu a realizaci přístrojové desky ke školnímu projektu elektrické motorky. Motocykl bude komunikovat s mikropočítačem pomocí několika senzorů a desky Smart BMS, ze které získává základní data o akumulátoru. Veškeré údaje se zpracovávají pomocí programu a následně se zobrazují na displeji, díky vytvořenému grafickému rozhraní. Nejdůležitější informace se v průběhu jízdy budou ukládat do databáze a z těchto údajů budou stanovena příslušná měření pomocí praktického testování.

Bakalářská práce je rozdělena na jednotlivé úkoly. První zahrnuje rešerši moderních zobrazovacích jednotek jednostopých vozidel a následně porovnává jejich technologie. Též poukazuje na studie zabývající se bezpečností v ohledu návrhu designů přístrojových desek. V další části je charakterizován potřebný hardware a jednotlivá externí zařízení, která mikropočítač ke své funkci využívá. Následně je zde popsán celkový vytvořený software a stručně vysvětlené postupy při návrhu grafických prvků. Poslední kapitola směřuje k praktickému testování celého výrobku. Jsou zde popsány získané hodnoty a zhodnoceny mezi sebou.

1 Rešerše zobrazovacích jednotek

1.1 Typy přístrojových desek jednostopých vozidel

Dříve se jako zobrazovací jednotky využívaly převážně mechanické analogové zobrazovače, které zobrazovaly jezdcům jen ty nejdůležitější informace, jako je rychlost vozidla, otáčky motoru a celkový stav ujetých kilometrů. Použitelnost analogových zobrazovačů byla ale natolik spolehlivá a nenáročná, že se tato technologie u mnoha výrobců zachovala a přístrojové desky jsou v současnosti menšími LCD pouze doplňovány.

S cílem zobrazovat o vozidlu více, než pouhých pár základních informací, se vše začalo digitalizovat a přenášet na různé typy displejů. Nejprve se tento trend promítl do sekce automobilových jednotek, a následně se začal používat i pro jednostopá vozidla [7].

1.1.1 Přístrojová deska se segmentovým LCD

Drtivá většina novějších jednostopých vozidel, které nevyužívají analogové zobrazovače, používají již předem definované segmentové LCD. Tato technologie je velmi spolehlivá a energeticky vysoce nenáročná. Výhodou oproti analogovým zobrazovačům je možnost vyššího počtu zobrazovaných informací a pominutí veškerých mechanických součástí. Bohužel tento typ displeje má již předdefinované segmenty a ty jsou nadále neměnné, takže jeho možnosti zobrazování jsou v určitých mezích konečné a prakticky jsou obohaceny jen o pár informací navíc, nežli je tomu u těch analogových. Hrozí zde také vysoké množství méně čitelných prvků najednou, které mohou jezdce ohrozit v průběhu jízdy, kvůli zvýšení reakční doby při snaze přečíst danou informaci.

Tuto variantu zobrazování používá například Honda 2020 model Rebel 300, jejíž menší oválný displej je umístěn ve středu panelu a pod ním jsou zasazeny barevné LED diody signalizující jednotlivé kontrolky.



Obrázek 1: Přístrojová deska motocyklu Honda Rebel 300 [14]

1.1.2 Přístrojová deska s TFT digitálním displejem

Díky novému nástupu TFT displejů byl řidič schopný získávat data nejen při jízdě v průběhu provozu, ale také o aktuálním stavu celého motocyklu, například prostřednictvím servisního menu. Celková konstrukce více obrazovek na displeji také umožnila výrobci nesnažit se veškeré informace naskládat na jednu obrazovku, a tím také zvýšila bezpečnost daného jezdce. Rovněž je zde umožněno řidiči zvolit si libovolnou konfiguraci a rozložení jednotlivých prvků, kterou si přizpůsobí sám pro komfortnější styl jízdy.

Tyto novější displeje, které můžeme nalézt například u modelu Ducati Multistrada V4, mají oproti předešlým též pokročilejší funkce, jako je zabudovaný navigační systém nebo spojení s mobilním telefonem a zobrazování notifikací o příchozích hovorech a zprávách. V tomto daném případě motocykl zobrazuje veškeré informace pomocí až 6,5palcového displeje a již nepoužívá přídavné led signalizační technologie a implementuje vše na obrazovku [10].



Obrázek 2: Přístrojová deska motocyklu Ducati Multistrada V4 model S [2]

1.2 Technologie displejů

Při návrhu displeje pro motocykl zde nastává problém jeho zobrazování na přímém slunci, kdy například princip LCD technologie TN je velmi málo čitelný. Na rozdíl od automobilů, které mají svoji přístrojovou desku skrytou z větší části ve stínu díky střeše a dobře vytvarovanému krytí před slunečním svitem, by se běžnější LCD technologie zobrazovací jednotky motocyklů stávaly obětmi silného oslnění a špatné čitelnosti jednotlivých informací. Pro tyto účely se začaly používat takzvané TFT displeje.

TFT je technologie pro tenkovrstvý film z tranzistorů, což je varianta LCD nebo LED. Jejich největší výhodou pro zobrazování na denním světle je velmi jasné podsvícení a vysoká čitelnost jednotlivých prvků [15].

1.3 Návrh designu v rámci bezpečnosti

To, že displeje jednostopých vozidel nemají úplně propracovaný grafický vzhled, ale jsou navrhovány spíše minimalisticky a jednoduše, má své důvody. Většina výrobců klade při návrhu designu důraz na jednoduchost, co nejmenší zobrazování pomocí animací, a hlavně na vyobrazení pouze těch nejrelevantnějších informací na jízdní obrazovku. Zásadním důvodem je bezpečnost jezdce. Jízda na jednostopém vozidle je o mnohem náročnější v ohledu manévrovatelnosti, se kterou se snižuje řidičova bezpečnost než například v automobilu, kde je ještě řidič chráněn bezpečnostními pásy a spoustou dalších ochranných prvků. Řidiči motocyklů též často jezdí o mnoho vyšší rychlostí, než je rychlost povolená a bezpečná v daném úseku.

Studie lidského chování při jízdě ve vozidle zjistila, že maximální bezpečná doba fixace pohledu na displej, při rychlosti vozidla 30 km h^{-1} , je 4,1 sekundy [21]. Řidiči trvá přibližně 0,78 sekundy, aby správně vyčetl informaci z displeje, jako je například aktuální rychlost vozidla. Pro bezpečné čtení textové informace by měla obsahovat maximálně 11 znaků [22]. Za tyto krátké časové intervaly musí být jezdec schopný vyčíst požadovanou informaci z přístrojové desky, a čím je displej zřetelnější a neobsahuje pro jízdu nepodstatné prvky, které by ho rozptylovaly, tím kratší dobu může jezdec věnovat pozornost displeji a mít tak bezpečnější jízdu.

Jelikož je na Tchaj-wanu nejběžnějším vozidlem právě motocykl, tak tamní studie zabývající se vzhledem přístrojových desek motocyklů za účelem minimalizovat nehody řidičů, došla k následujícímu závěru o umístění jednotlivých informačních prvků na displeji. Hlavní prvek zobrazující rychlost vozidla, by měl být umístěn na straně levé. Stav paliva, eventuálně zbývající kapacity akumulátoru, napravo od rychloměru a signalizační světla a ikony by měly být umístěny po stranách těchto hlavních částí [18]. Z tohoto rozložení bylo vycházeno při návrhu vlastního řešení designu.

2 Hardware

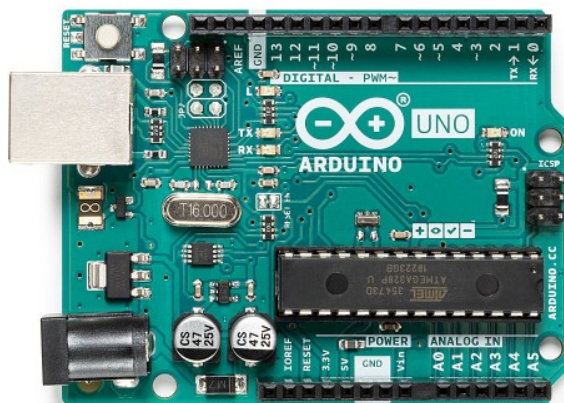
V následující části zhodnocuji výběr vhodných zařízení pro následné řešení práce. Porovnám jejich strukturu, využití a jejich výhody a nevýhody pro výrobu palubního počítače pro elektronické jednopólové vozidlo. Též popisují veškeré hardwarové komponenty, se kterými bude mikropočítač komunikovat.

2.1 Výběr mikropočítače

Pro tento projekt jsem volil ze dvou velmi známých programovatelných zařízení, která jsou snadně dostupná na dnešním trhu. Prvním je mikropočítač značky Raspberry Pi a druhým je mikrokontrolér Arduino. Obě tato zařízení byla vytvořena jako výukové pomůcky při tvorbě vlastního softwaru. I když jsou si velice podobná, liší se svým zpracováním a funkčním procesem v praxi. Hlavním rozdílem mezi nimi je fakt, že Arduino je deska mikrokontroléru a jeho hlavní funkce je zjednodušování hardwarových projektů. Zatímco jeho oponent Raspberry Pi je mikropočítač, který se spíše hodí na tvorbu softwarových aplikací.

2.1.1 Arduino

Arduino je mikrokontrolér skládající se ze tří hlavních periferií. První z nich je hardwarová prototypová platforma, druhou jazyk Arduino a v neposlední řadě rovněž integrované vývojové prostředí a knihovny. Deska je spíše mikrořadič než plnohodnotný počítač. Arduino nemá operační systém, ale kód lze využívat a spouštět tak, jak jej interpretuje jeho trvalý softwarový program. Vývojové desky v zásadě spouštějí jeden program iterativně.



Obrázek 3: Deska Arduino UNO [4]

Hardware se skládá z jednoduché základní desky s mikrokontrolérem a jeho podpůrnými obvody s konektory pro připojení k jednotlivým modulům a USB rozhraním pro komunikaci s počítačem na zápis vlastního kódu. Nejběžnějším z těchto zařízení je model Arduino Uno, který používá mikrokontrolér Atmel ATmega328P. Disponuje čtrnácti digitálními a šesti analogovými vstupy, resetovacím tlačítkem, ICSP piny a krystalovým oscilátorem. Tato deska je napájena pomocí 7–12 V.

Programy Arduino jsou psány pomocí aplikace Arduino IDE z vašeho počítače, což je speciální software spouštěný na daném operačním systému, který umožňuje psát kód pro desku. Kód Arduino se označuje jako sketch.

Projekty v Arduino se dají programovat pomocí upraveného jazyka C nebo C++, nicméně některé standartní knihovny z tohoto základního jazyka zde nefungují [5].

2.1.2 Raspberry Pi

Deska Raspberry Pi obsahuje mikroprocesor. Tento mikroprocesor má architekturu ARM a může provozovat kompletní operační systém Linux. Využívá takzvaný system-on-chip, který integruje CPU a GPU do jediného integrovaného obvodu.

Na konkrétní vybrané desce Raspberry Pi 3 model B+ 2017 můžeme najít čtyři USB vstupy, HDMI port na připojení externí obrazovky, DSI konektor pro připojení kamery a externího displeje, ethernetový port, napájení pomocí microUSB, 40 vstupně výstupních GPIO pinů a audio Jack 3,5mm konektor. Nemá integrované úložiště, místo toho má slot pro kartu microSD, na kterou se ukládá operační systém a výsledný program [20].



Obrázek 4: Deska Raspberry Pi 3 model B V1.2 [19]

Jelikož deska podporuje OS, můžeme pro vývoj programu využívat hned řadu programovacích jazyků. Pro vývoj projektu byl zvolen jazyk Python, s nímž mám základní zkušenosti a který je výhodný z hlediska snadného používání při vývoji grafických uživatelských rozhraní.

2.1.3 Porovnání obou zařízení

Z dostupných řešení a informací k nim jsem se rozhodl zvolit zařízení Raspberry Pi. Díky operačnímu systému se toto zařízení více hodí pro výrobu grafické aplikace z důvodu snazší dostupnosti příslušných knihoven a též vyššího výkonu zařízení. Dalším důvodem je fakt dostupnosti programových vláken, která jsou výhodou pro komunikaci různých přídavných komponent, a zajišťují tak plynulý průběh grafického rozhraní. Též je zde výhodně zabudovaný Bluetooth čip přímo v desce, díky kterému může Raspberry Pi komunikovat s BMS deskou bez dalších externích modulů.

Tabulka 1: Hlavní rozdíly porovnávaných zařízení

Parametr	Raspberry Pi 3 Model B+	Arduino Uno R3
Typ	Jednodeskový počítač	Mikrokontrolér
OS	Funguje na daném operačním systému	Funguje bez operačního systému
Wi-Fi/Bluetooth	Interně zabudováno	Pouze externě
Připojení externích zařízení	Na bázi PC (HDMI, USB, Ethernet... a GPIO piny)	Hardwarově orientovaná připojení, jako jsou GPIO piny, napájecí konektor, USB konektor
Provedení programu	Možnost spuštění více programů najednou	Spuštění pouze jednoho programu
MCU	Broadcom BCM2837B0	AVR ATmega328p
Frekvence MCU	1,4 GHz	16 MHz
Šířka registru	64 bit	8 bit
Paměť RAM	1 GB	2 kB
GPIO Porty	5 V, 3,3 V, GND, pouze digitální I/O piny	5 V, 3,3 V, GND, digitální i analogové I/O piny
GPIO piny	40	20
Spotřeba energie	1,5 W	175 mW
Cena	770 až 1200 Kč	250 až 400 Kč

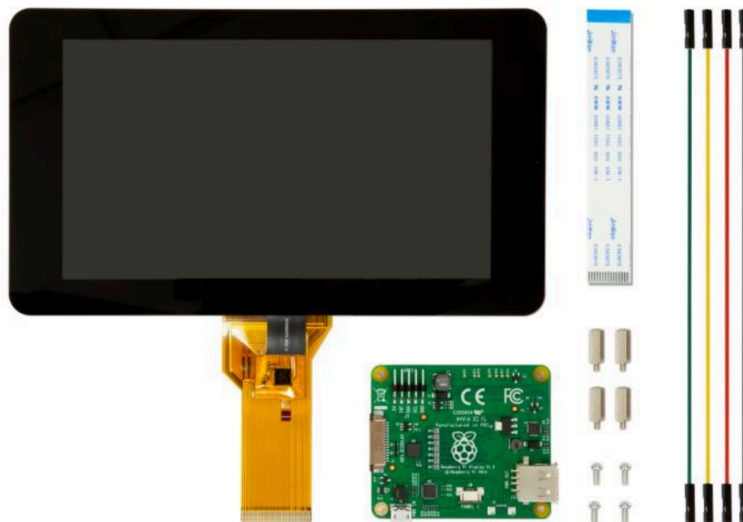
parametry tabulky jsou převzaty z rešerše zabývající se těmito zařízeními [1]

Kvůli vyššímu výkonu bude mít Raspberry Pi větší tendenci k zahřívání. Tento problém bude ale kompenzován jeho vhodným umístěním na vozidle. Celý mikropočítač i s displejem bude zabudovaný v krytu, který se bude nacházet na řídicích motocyklu, díky čemuž bude podléhat venkovním teplotám v situaci, kdy bude vozidlo v provozu, a tím bude dostatečně pasivně chlazen.

2.2 Displej pro přístrojovou desku

Pro zobrazování grafického designu jsme zvolili displej s rozlišením 800x480 pixelů. Pro projekt elektrické motorky jsme chtěli využít displej v paralelním DPI zapojení RGB24 připojením 8 bitů pro červenou, zelenou a modrou barvu. Výhodou pro toto zapojení je odstranění externího řadiče, a tím i snížení celkové spotřeby energie. Tento displej zároveň neobsahuje zbytečnou dotykovou vrstvu, jelikož se bude ovládat pomocí tlačítek upevněných na řídítkách motocyklu. Nevýhoda, kvůli které se pro testovací platformu tento displej nehodí, je fakt, že po zapojení displeje zaplní většinu nebo i všechny GPIO piny, které jsou důležité pro komunikační a ovládací komponenty.

Pro testování byl vybrán sedmipalcový displej, který disponuje stejným rozlišením, jako původní plánovaný. Připojuje se k mikropočítači pomocí základní desky, která je napájena vstupním napětím 5 V přivedeném z GPIO pinu Raspberry Pi. Deska zpracovává obrazový signál, který je přiveden pomocí plochého kabelu DSI.

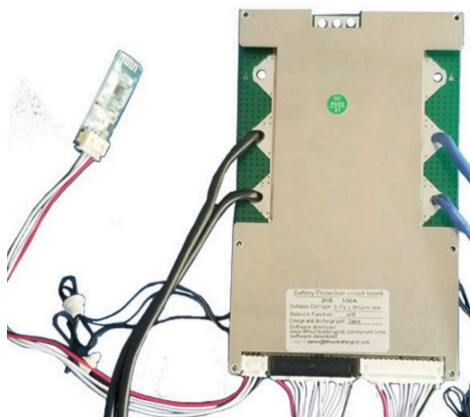


Obrázek 5: Deska PiCAN 2 Duo [17]

2.3 Smart BMS 28s

Tato Smart BMS je deska plošných spojů o sérii 28s, která řídí celkový akumulátor. Tento produkt vyrobený firmou LLT Power má čtyři teplotní senzory a řadu vestavěných softwarových funkcí. Uživatel má možnost nastavovat různé parametry a také číst informace o daném spravovaném akumulátoru, které budou v této práci využity. Deska je vybavena čipy Texas Instruments BQ76940 a Atmega328.

Smart BMS komunikuje prostřednictvím různých komunikačních protokolů jako jsou UART, CAN, SMBus, RS485 nebo RS232 a je dodávána s Bluetooth modulem, s kterým lze pomocí aplikace pro chytré telefony desku ovládat [11].



Obrázek 6: Smart BMS [11]

2.4 Hallův senzor NJK-5002A

Tento bezkontaktní typ senzoru funguje na principu Hallova jevu, kdy převádí detekci vnějšího magnetického pole na elektrický obdélníkový signál o napěťové úrovni signalizující logickou jedničku.

Konkrétní typ senzoru NJK-5002A je typ NO, který je schopný detekovat pole ve vzdálenosti 10 mm od detekčního magnetu při vstupním napětí v rozsahu 5–35 V [6].



Obrázek 7: Hallův senzor NJK-5002A [6]

2.5 Ovládací panel Duke 790

Panel z motocyklu Duke 790 z roku 2018 bude využíván na ovládání signalizačních světel jako jsou směrové ukazatele, přepínání mezi režimy denního, potkávacího a dálkového svícení a ovládání akustické houkačky. Těž obsahuje čtyři mikropřínače pro případné nastavování a interakci se softwarem displeje.



Obrázek 8: Ovládací panel Duke 790 2018

2.6 Teplotní čidlo DS18B20

Zvolený DS18B20 je typ senzoru, který je vyhotoven ve stylu vodotěsné sondy a je zapouzdřen spolu s prodlouženým kabelovým připojením pro snadné umístění senzoru do prostor motoriky. Teplotní senzor je poměrně přesný a ke své činnosti nevyžaduje žádné externí součásti. Může měřit teploty od -55 °C do $+125\text{ °C}$ s přesností $\pm 0,5\text{ °C}$, a je tak ideálním senzorem pro běžné měření okolních teploty.

Jednou z výhod DS18B20 je, že na stejné 1–Wire sběrnici může koexistovat více senzorů. Protože každý model má ve výrobě přidělen jedinečný 64bitový sériový kód, dají se tedy od sebe odlišit a používat funkci 1–Wire [9].



Obrázek 9: Teplotní čidlo DS18B20 s prodlouženým kabelovým spojením [9]

3 Komunikace Raspberry Pi s danými prvky

3.1 BMS komunikace

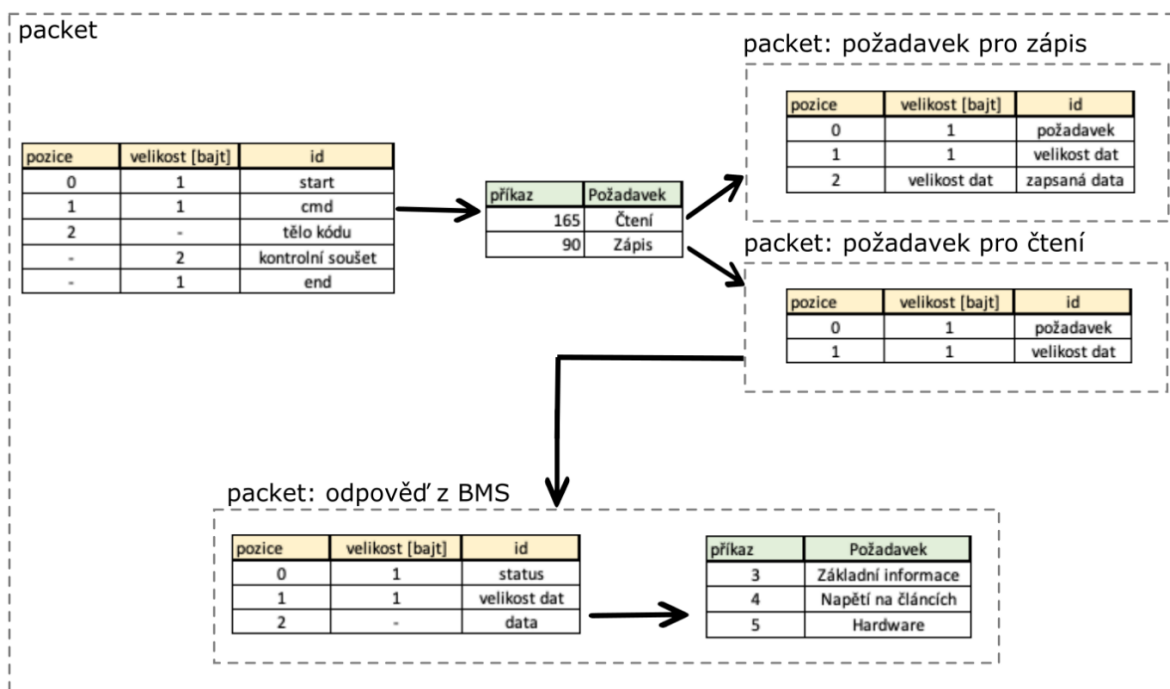
V této části se budu zabírat popisem komunikace BMS jednotky a mikropočítače, který následně se získanými daty pracuje.

Smart BMS používají k sériové komunikaci obecný protokol RS485, RS232, UART nebo lze pomocí protokolů komunikovat bezdrátově díky přídavnému Bluetooth modulu. Pro komunikaci byl zvolen již zmíněný Bluetooth modul, jelikož zde přišla vhod snadná manipulace, rychlé připojování a odpojování různých kontrolních zařízení k jednotce a hlavně galvanické oddělení mikropočítače od desky BMS.

3.1.1 Protokol

Veškerá komunikace probíhá v podobě hexadecimálního kódu. Každý přijatý, či odeslaný paket má počáteční bajt „DD“ a koncový bajt „77“.

Při zadávání požadavku po startovacím bajtu je tu možnost zápisu nebo čtení informací v desce. Bajt o hodnotě 165, („A5“) znamená, že host požaduje informace pro čtení, a bajt 90, („5A“) vyznačuje požadavek pro zápis. Poté zde existují bajty k identifikaci pole nebo typu a další data těla celého kódu.



Obrázek 10: Diagram topologie digitální komunikace

3.1.2 Poskytnuté informace

Pro identifikaci čtení jednotlivých polí o akumulátoru musí Raspberry Pi vyvolat požadavek, který sled informací má být přesně získán. Při zadání požadavku nám jednotka BMS odpoví kódem, u kterého jeho tělo obsahuje veškeré informace. Jsou zde tři možnosti o vyvolání kódu, lišící se následovně:

Požadavek „DD A5 05 00 FF FB 77“ nám pouze vypíše danou verzi použitého hardwaru.

Požadavek „DD A5 04 00 FF FC 77“ vypíše jednotlivá napětí na článcích. Velikost paketu se projevuje na maximálním počtu připojených článků, který podporuje daná verze desky BMS.

Tabulka 2: Přijatá data při požadavku o jednotlivých napětích na článcích

Informace	Velikost [bajt]	Popis	Jednotka
Velikost dat	2	Počet použitých bajtů (podle N počtu článků)	–
Napětí na článků 1	2	Napětí na daném 1. článku	mV
Napětí na článků N	2	Napětí na daném N článku	mV

Požadavek „DD A5 03 00 FF FD 77“ vypíše příslušné základní informace.

Tabulka 3: Přijatá data při požadavku základních informací

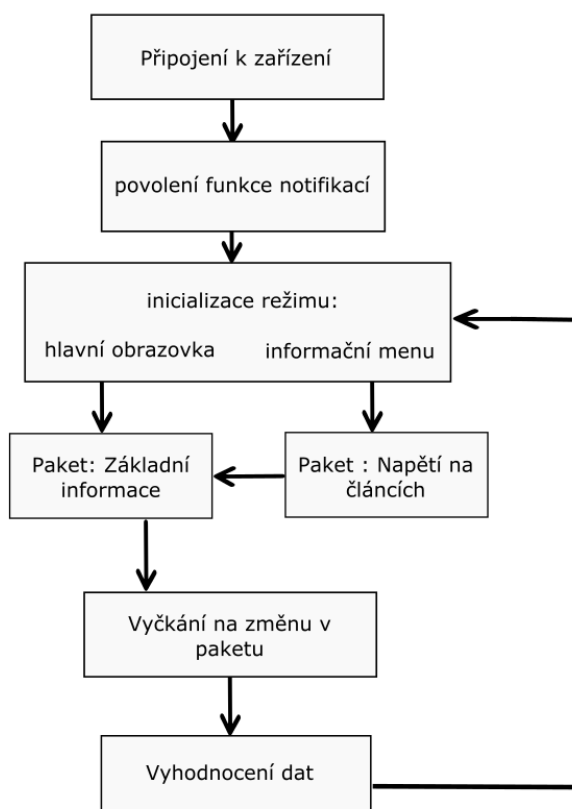
Informace	Velikost [bajt]	Popis	Jednotka
Celkové napětí	2	Celkové napětí na všech článcích	mV
Proud	2	Celkový odebíraný a dodávaný proud (odebíraný má zápornou hodnotu, dodávaný kladnou)	mA
Zbývající kapacita	2	Aktuální zbývající kapacita baterie	mAh
Nominální kapacita	2	Celková kapacita baterie	mAh
Počet cyklů baterie	2	Počet nabitých cyklů	–
Datum výroby	2	Datum výroby produktu	–
Status balancování 1	2	Balancování článků 0–16	–
Status balancování 2	2	Balancování článků 17–32	–
Stav ochrany	2	Chráněný stav celé BMS	–
Verze	1	Verze softwaru	–
SOC	1	Aktuální úroveň kapacity akumulátoru	%
FET Control Status	1	Status tranzistorů	–
Články v sérii	1	Počet zapojených článků v sérii	–
Počet teplotních čidel	1	Počet zapojených teplotních čidel	–
Teplotní čidla	2	Naměřené hodnoty jednotlivých čidel	K

3.1.3 Struktura programu komunikace

Na komunikaci byla použita knihovna Bluetooth GATT SDK. Tato knihovna pomáhá mikropočítači komunikovat se zařízením pomocí technologie Bluetooth LE, kterou Raspberry Pi 3 B+ podporuje. Tato technologie je navržena tak, aby při jejím používání docházelo k co nejmenší spotřebě energie zařízení, na kterém je provozována [8].

Tuto knihovnu jsem zvolil pro její snadnou manipulaci, vysokou spolehlivost a vhodnost využití pro projekt motocyklu. Raspberry Pi svoji komunikaci zahájí požadavkem o připojení k vybrané MAC adrese, která je speciálně přidělena ke každému Bluetooth přijímači u BMS jednotky, a je spuštěn manager funkcí. V programu používám funkci notifikací. Pomocí této funkce bude Raspberry Pi získávat data z desky pouze tehdy, pokud se v daném paketu bude vyskytovat změna v informaci.

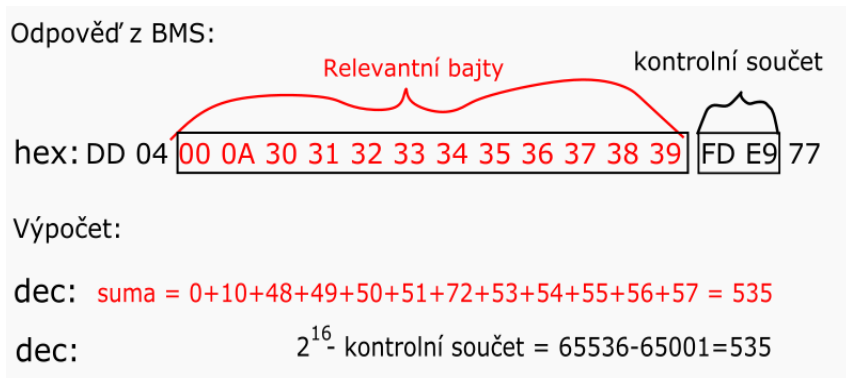
Volání jednotlivých požadavků si už program řídí podle toho, jestli se nachází v režimu informačního menu nebo na hlavní obrazovce, a to kvůli maximalizaci rychlosti přenosu relevantních dat.



Obrázek 11: Diagram programu Bluetooth komunikace s BMS jednotkou

3.1.4 Kontrolní součet

Protokol poskytuje též možnost kontrolního součtu, kterým se ověří správnost přenosu informace. Je to v podstatě součet za každý relevantní bajt. Tato suma se následně odečte od hodnoty 2^{16} (65 536) a konečný výsledek je roven dvěma posledním bajtům, před konečným bajtem „77“. Pokud se tyto dvě čísla rovnají, informace se přenesla v pořádku.



Obrázek 12: Kontrolní součet

Funkce přijme daný paket a vyhodnotí z něj relevantní bajty, které jsou od druhého počátečního, až do třetího posledního bajtu. Poté vyhodnotí kontrolní součet, který je v podobě třetího posledního až do předposledního bajtu.

```
def verify(packet):  
    data_packet = packet[2:-3]  
    check_sum = packet[-3:-1]
```

Zdrojový kód 1: Definování relevantních bajtů a hodnoty kontrolního součtu

For cyklus postupně od čísla 65 536 odčítá relevantní bajty a nakonec vrátí „TRUE“, pokud jsou obě hodnoty souhlasné.

```
crc=0x10000 #hodnota v dec. 65536  
for i in data_packet: #odečte od čísla 65536 relevantní bajty  
    crc = crc-i  
crc_b = crc.to_bytes(2, byteorder='big')  
return check_sum == crc_b #vrátí hodnotu True, pokud se shodují
```

Zdrojový kód 2: Funkce pro kontrolu validního paketu

3.2 Hallův senzor

Hallův senzor generuje na výstupu obdélníkový signál při každém otočení kola motocyklu. V tuto chvíli Raspberry Pi detekuje logickou jedničku a propočítává délku času mezi jednotlivými signály. Díky vypočtené době mezi otáčkami a znalosti průměru kola lze propočítávat aktuální rychlost vozidla, otáčky a ujetou vzdálenost.

3.2.1 Popis signálu

Čidlo při jedné otáčce generuje na výstupu celkově 2 krátké pulsy. Pokud je první signál kratší než ten druhý, znamená to, že se kolo točí ve směru jízdy a naopak. Pro motocykl je funkce o detekci směru otáčení irelevantní, tudíž tuto funkci eliminují pomocí vytvoření krátkého zpoždovacího času po detekci logické jedničky.



Obrázek 13: Detekce signálu přítomnosti magnetu Hallova senzoru

Z výstupu osciloskopu jsou zřejmé již zmíněné po sobě jdoucí pulsy detekující otočení kola a směr otáčení. Z naměřené frekvence 27,3704 Hz je patrná perioda signálu 36,5 ms. Rychlost motocyklu v tomto měřeném stavu je 117 km h⁻¹.

Pomocí stanovené maximální rychlosti 150 km h^{-1} , kterou je motocykl schopen dosáhnout, byl vypočítán čas, po který nebude detekován žádný nadcházející puls, a tím je eliminována chybovost čtení mezi dvěma po sobě jdoucími impulsy znázorňující směr otáčení kola.

$$T = \frac{3 \cdot \pi \cdot r \cdot 60}{v_{MAX} \cdot 25} = \frac{3 \cdot \pi \cdot 0.189 \cdot 60}{150 \cdot 25} = 28,5 \text{ ms}$$

Z daného vzorce bylo vypočítáno, že po 28,5 ms se bude opakovat další zaznamenaný puls při dosažení dané maximální rychlosti, takže při nastaveném času zpoždění 20 ms bude mít program na čtení následujícího rezervu 8,5 ms.

3.2.2 Popis programu na zaznamenávání rychlosti

Nejprve byl nastaven vstupní GPIO pin 14 na detekci náběžné hrany signálu při logické jedničce. Při každé detekci se volá funkce „get_pulse“, která následně propočítává další data. Nakonec je zde nastavený zpoždovací čas na eliminaci druhého pulzu o směru otáčení kola parametrem „bouncetime“.

```
hall= 14
GPIO.add_event_detect(hall,GPIO.RISING,callback = get_pulse,bouncetime=20)
```

Zdrojový kód 3: Přiřazení vlastností senzoru

Při každé detekci se zaznamená čas od posledního pulsu a z tohoto času se vypočítávají otáčky a rychlost vozidla. Při každém zaznamenaném pulsu program též inkrementuje hodnotu „distance“, která je vypočítávána ze vzorce pro obvod kola, a tím detekuje aktuální ujetou vzdálenost.

```
def get_pulse():
    cycle = 0
    pulse+=1
    if pulse > 0:
        elapse = time.time() - start
        pulse-=1
    if cycle > 0:
        distance += wheel_c/1000
        cycle-=1
    multiplier = 3600/elapse
    rpm = 1/elapse *60
    speed = (wheel_c*multiplier)/1000

    start = time.time()
```

Zdrojový kód 4: Celková funkce „get_pulse“ na výpočet informací o pohybu vozidla

4 Návrh palubního počítače

Navrhování počítače se dělí do několika odvětví. Jako první je příprava mikropočítače Raspberry Pi a jeho konfigurace. Dále se zaměřím na strukturalizaci hlavního programu a jeho členění. A následně popisuji jednotlivé funkce a grafický design panelů.

4.1 Konfigurace mikropočítače Raspberry Pi

Prvním úkolem bylo připravit si celkové prostředí systému mikropočítače.

- Nejprve byla nainstalována nejnovější verze operačního systému z oficiálních stránek Raspberry Pi [16].
- Poté byly zvoleny a nainstalovány příslušné Python knihovny.
- Dále byl vytvořen plně funkční software.
- Následně bylo v systému pozměněno původní bootování odstraněním příkazových řádků a notifikačních obrázků a nastaveno spouštěcí okno pro logo univerzity.
- Nakonec bylo nakonfigurováno automatické spouštění programu při startu zařízení.

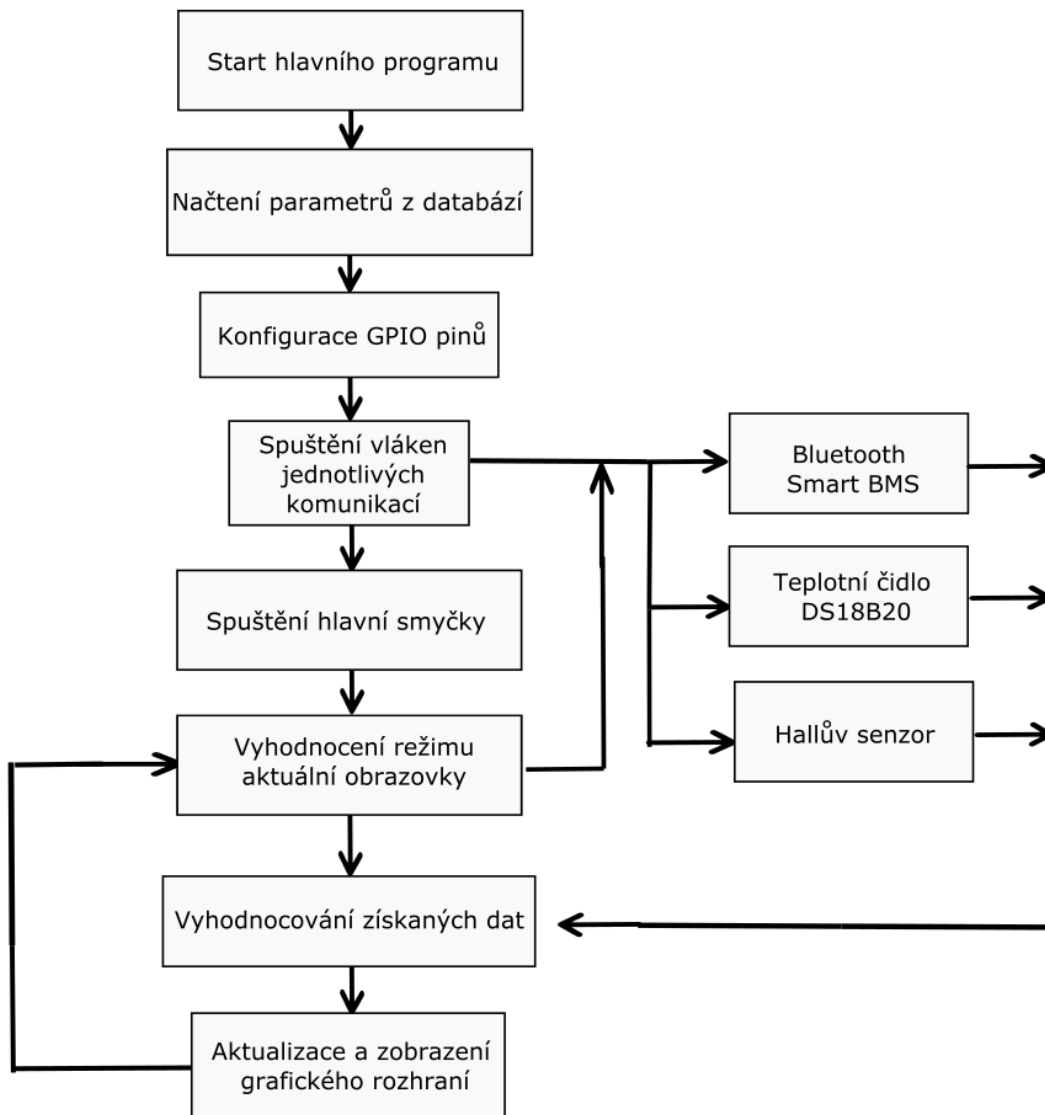
Režim automatického spouštění programu

Pro automatické spouštění programu byl vytvořen startovací soubor `.xsession` a byla do něj přiřazena cesta pro soubor `main.py`, který je hlavním článkem praktické práce. Poté byl nainstalován `window-matchbox nodm` pro systém Raspbian, který napomáhá zobrazování grafických aplikací bez nutnosti použití desktopového prostředí. Tato akce zakázala aktivaci mnoha nadbytečným funkcím pro vytvářený software, které se využívají za předpokladu plnohodnotného využívání systému. Tato operace výrazně snížila dobu trvání zapínání programu a zvýšila možný výkon spuštěné aplikace [12].

Jelikož používám pro zobrazování grafického obsahu modul Python Tkinter, tak se doba spuštění aplikace omezuje na určitý časový interval, který se prodlužuje načtením základního grafického rozhraní operačního systému, který je využíván. Celý systém je uložen na microSD kartě, která se svojí rychlostí čtení dokáže aplikaci plnohodnotně zobrazit ve 22 sekundách po přívodu napájení na desku Raspberry Pi.

4.2 Hlavní program

Struktura hlavního programu je postavená na vytvořeném grafickém designu, který se aktualizuje ve smyčce. Hlavní smyčka je nastavená na opakování se zpožděním 80 ms, tudíž snímková frekvence celkového rozhraní činí 12,5 FPS. Toto zobrazování je zcela dostatečné pro maximální rychlost čtení všech komunikací a prvků, aby žádný nebyl zbytečně zpoždován.



Obrázek 14: Diagram funkce hlavního programu

Při spuštění se nejprve načtou data z databáze. Dle příslušných dat se vyhodnotí rozložení posledních uložených parametrů a ty se projeví na grafické stránce programu. Následně program přidělí funkce pro jednotlivá tlačítka a spínače pro hlavní obrazovku. Vzápětí se spustí celkem tři vlákna, aby hlavní a grafický software nebyly závislé na zpoždění jednotlivých komunikací při jejich aktualizaci. Nakonec se budou vyhodnocovat data z vláken v hlavní smyčce a odesílat

pro další grafické zobrazování. Cyklus smyčky se bude opakovat do doby, než se odpojí mikro-počítač od napájení.

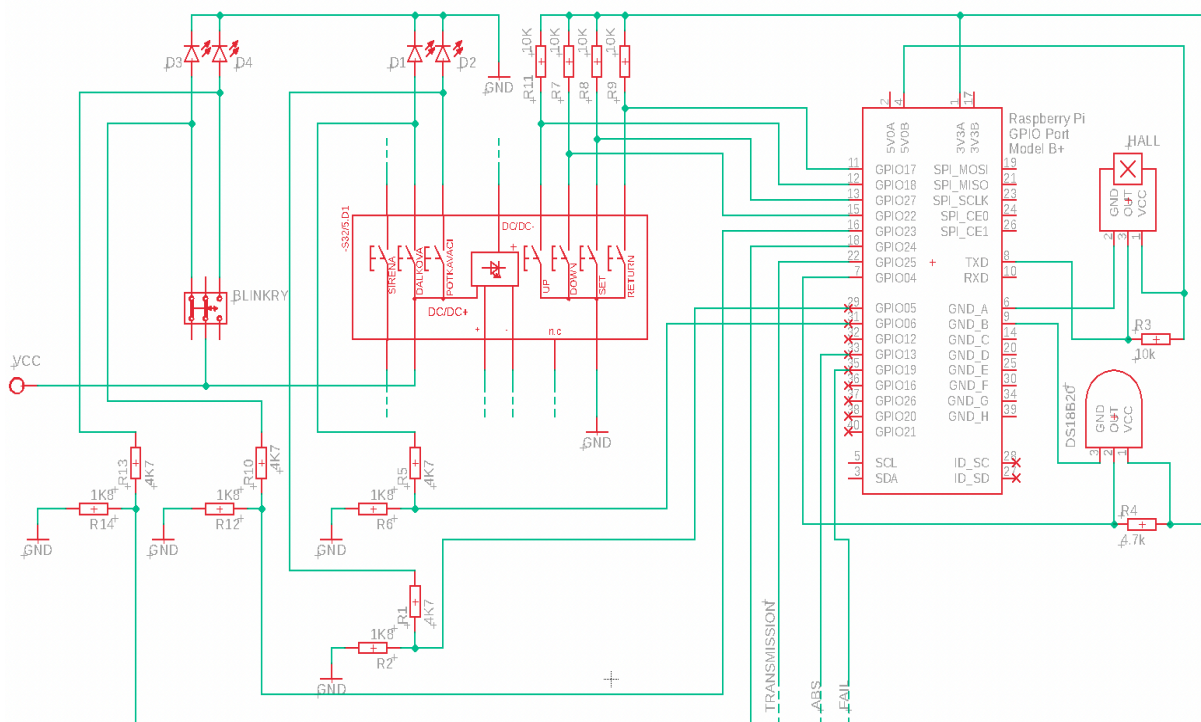
4.2.1 Databáze

Pro eliminaci ztrát informací za předpokladu vypnutí mikropočítače jsem zvolil ukládání dat do databáze. Jako konkrétní databáze byla zvolena SQLite. Tento typ jsem si zvolil, protože na rozdíl od ostatních nepoužívá externí systém, a tím pádem zapisuje svá data přímo do souboru nacházejícího se ve složce s programem a nespolečá se na funkci serveru jako například MySQL nebo MariaDB [23]. Jelikož nevyužívá žádný serverový proces, tak nezahluje výkon mikropočítače, pokud se tedy zrovna nevyužívá. S těmito vlastnostmi je tato databáze ideální pro tento projekt.

Program pracuje se dvěma soubory databází. Do první se ukládají pouze testovací data. Druhá obsahuje celkem dva tabely. První tabel obsahuje data o rozložení nabídky na hlavním panelu a druhý ukládá informace o celkové průměrné spotřebě a celkových ujetých kilometrech funkce „odometr“ a funkce „trip“.

Databáze je aktualizována v intervalu deseti vteřin. Též se do ní zapisují data pokaždé, když program detekuje spád aktuální rychlosti na hodnotu 0 km h^{-1} , aby nedocházelo ke ztrátě dat mezi desetisekundovým intervalem při okamžitém zastavení a vypnutím napájení mikropočítače.

4.2.2 Konfigurace GPIO pinů



Obrázek 15: Schéma zapojení Raspberry Pi GPIO pinů s externími zařízeními

Ovládací panel Duke 790 2018 má signálová tlačítka spojena se zemí, a proto byla zvolena v programu varianta „pull-down“, a tudíž po celou dobu GPIO piny detekují logickou jedničku a svůj stav změní při sestupné hraně. Naproti tomu tlačítka pro aktivaci světel potkávacích, dálkových a směrové ukazatele spínají napětíovou úroveň 12 V a jsou spojena s příslušnými GPIO piny pomocí odporového děliče snižujícího tuto úroveň na 3,3 V pro detekci logické jedničky. Tímto stylem jsou nakonfigurována v programu jako „pull-up“. Zbylé využití piny detekují aktuální stav zařazeného převodu motocyklu, funkci ABS a signalizaci FAIL kontrolky při některé z definovaných vad vozidla.

Hallův senzor je napájen 5 V a výstupní signál je přiveden do datového vstupu GPIO14. Teplotní čidlo DS18B20 využívá napájení 3,3 V a výstup čte pin GPIO04.

4.2.3 Popis funkcí informačních ikon

Přístrojová deska zobrazuje též vedlejší informační data, mezi kterými si bude moci uživatel následně vybrat pomocí ovládacího panelu. Program nabízí celkem sedm informačních ikon.

Teplota akumulátoru

Tato ikona nám zobrazuje aktuálně nejvyšší naměřenou teplotu akumulátoru z teplot, které jsou k dispozici ze čtyř čidel umístěných kolem článků. Tato data jsou získávána z jednotky BMS.

Venkovní teplota

Venkovní teplota nabývá svých hodnot díky teplotnímu čidlu DS18B20, které je umístěné v prostorách motorčky tak, aby měřilo pouze teplotu okolního prostředí.

Trip

Tento parametr nejprve načítá data z databáze programu. Po získání dat z databáze se v průběhu jízdy inkrementuje o hodnotu obvodu jedné otáčky kola, kdy je pomocí vzorce pro obvod kružnice, při znalosti průměru kola, vypočítávána ujetá vzdálenost. Uživatel má možnost pomocí podržení tlačítka „return“ po dobu 2 vteřin vynulovat tuto hodnotu pro případ měření určité vzdálenosti nadcházející jízdy. Tato kombinace též nuluje data pro hodnoty spotřeba a dojezd.

Spotřeba

Spotřeba je zde měřena jako celková, tudíž se po startu programu načte minulá naměřená hodnota z databáze a díky aktuálnímu proudu a napětí poskytovaného z komunikace jednotky BMS je propočítávána a graficky obnovována každých 10 vteřin.

Dojezd

Dojezd je získáván z průměrné spotřeby a z hodnoty zbývající kapacity, kterou poskytuje též BMS jednotka. Graficky je obnovována v intervalu stejném jako spotřeba.

Odometr

Funkce odomoter je téměř totožná funkci trip. Její odlišnost se nachází pouze v absenci kombinace nulování, kdy tento parametr bude ukazovat celkovou ujetou vzdálenost vozidla a nebude ho možné nijak uživatelsky vynulovat.

Teplota motoru

Tato informace by měla získávat hodnotu z čidla pro měření teploty motoru. Jelikož ale čidlo zatím v motorce neukazuje správná data, tak je zde ikona pouze připravena pro možné budoucí doplnění hodnoty.

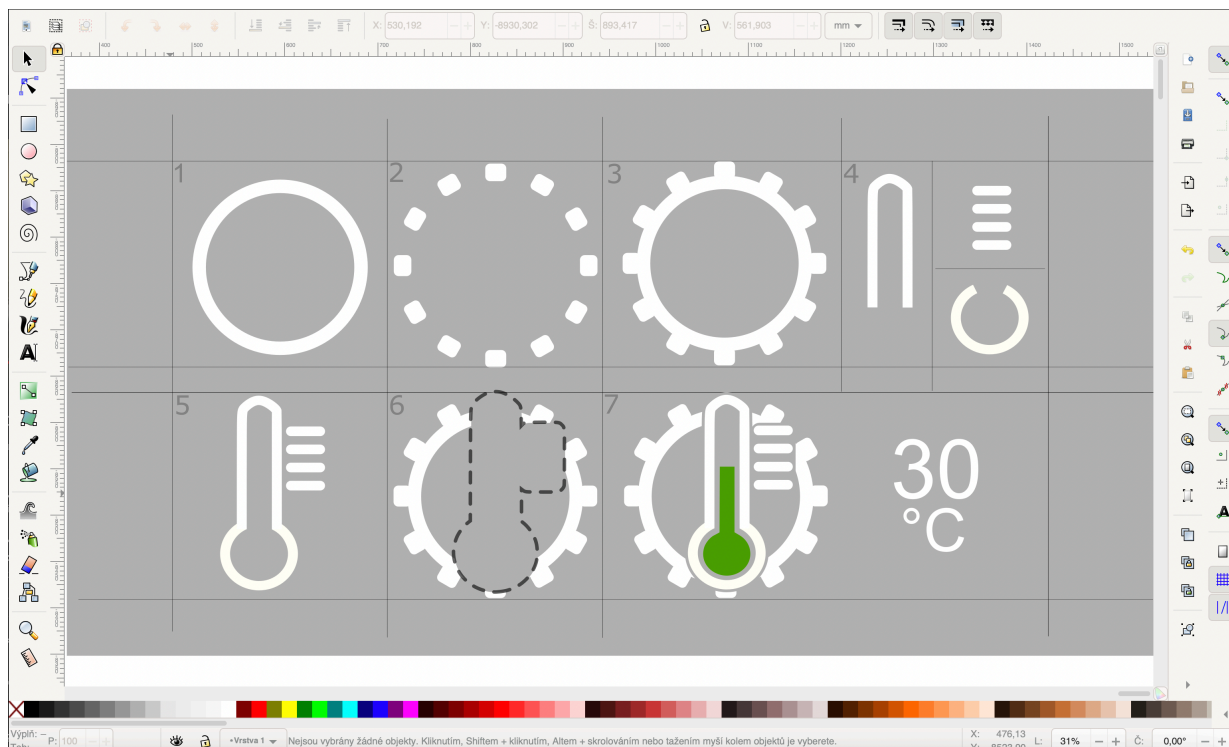
4.3 Návrh grafického prostředí

Pro zobrazení grafického prostředí v programu byla zvolena nejzákladnější forma přepisování grafických vektorů pomocí modulu Python Tkinter, který je zabudován do standardní knihovny jazyka Python. Jeho výhodou je, že je multiplatformní, tudíž vytvořený kód lze testovat na většině operačních systémů, jelikož je plně podporován vykreslováním nativních prvků pomocí daného systému, na kterém je aktuálně spuštěn. GUI vytvořené pomocí tohoto modulu mají ale nevýhodu zastaralejšího vzhledu a méně funkcionalit. Pro tento projekt je ale ideálním nástrojem díky jeho snadné manipulaci oproti ostatním modulům. Vzhled přístrojové desky motocyklu má být totiž jednoduchý, spolehlivý, a hlavně bez zbytečných animací, které by při jízdě u uživatele zbytečně upoutávaly pozornost [3] .

Software pro tvorbu grafického prostředí

Pro tvorbu veškerého grafického prostředí byl použit software Inkscape. Tento program se využívá výhradně pro vytváření a úpravy vektorové grafiky. Jeho hlavní výhodou je jeho multiplatformní dostupnost a fakt, že je zcela zdarma oproti například Adobe Ilustration. Je to ideální nástroj pro kreslení log, ikon a vytváření různých grafických obrazců dle potřeby vývojáře.

Při tvorbě jednotlivých panelů jsem si v softwaru vytvořil základní pole v příslušné barevné kombinaci o rozměrech rozlišení displeje. Tuto podstavu jsem si zvolil jako hlavní plátno a pomocí funkcí objektů a grafických nástrojů jsem postupně řadil do vrstev jmenovité bloky a objekty. Po vytvoření celkového plátna daného panelu byly exportovány jednotlivé grafické výrobky a rozřazeny do složek s ikonami k hlavnímu programu, kde jsou v průběhu volány.



Obrázek 16: Vývoj ikony pro teplotu motoru v prostředí Inkscape

Na obrázku výše je prezentován vzhled prostředí softwaru Inkscape. Pro příklad tvorby byla použita v tvarech komplexnější ikona, která ve výsledku bude značit a zobrazovat pomocí nevyplněného baru teploměru aktuální teplotu motoru. Jak je z obrázku zřejmé, v prvních krocích byla vytvořena hlavní kruhová základna, na kterou jsem symetricky přidával blokové prvky pro vyobrazení ozubeného kola představujícího motor. Z jednoduchých prvků v bodu 4 byla stvořena podoba teploměru, která bude ve své vnitřní části graficky zobrazovat hodnotu pomocí barevného ukazatele hladiny. Znásobený objekt v šířce a velikosti teploměru jsem odebral z bodu 3 ozubeného kola, abych zabránil nečitelnému překrytí dvou prvků a sloučil je do jedné celistvé ikony. V bodě 7 je znázorněna již celkově vypadající ikona, která bude zobrazena na plátně hlavní obrazovky.

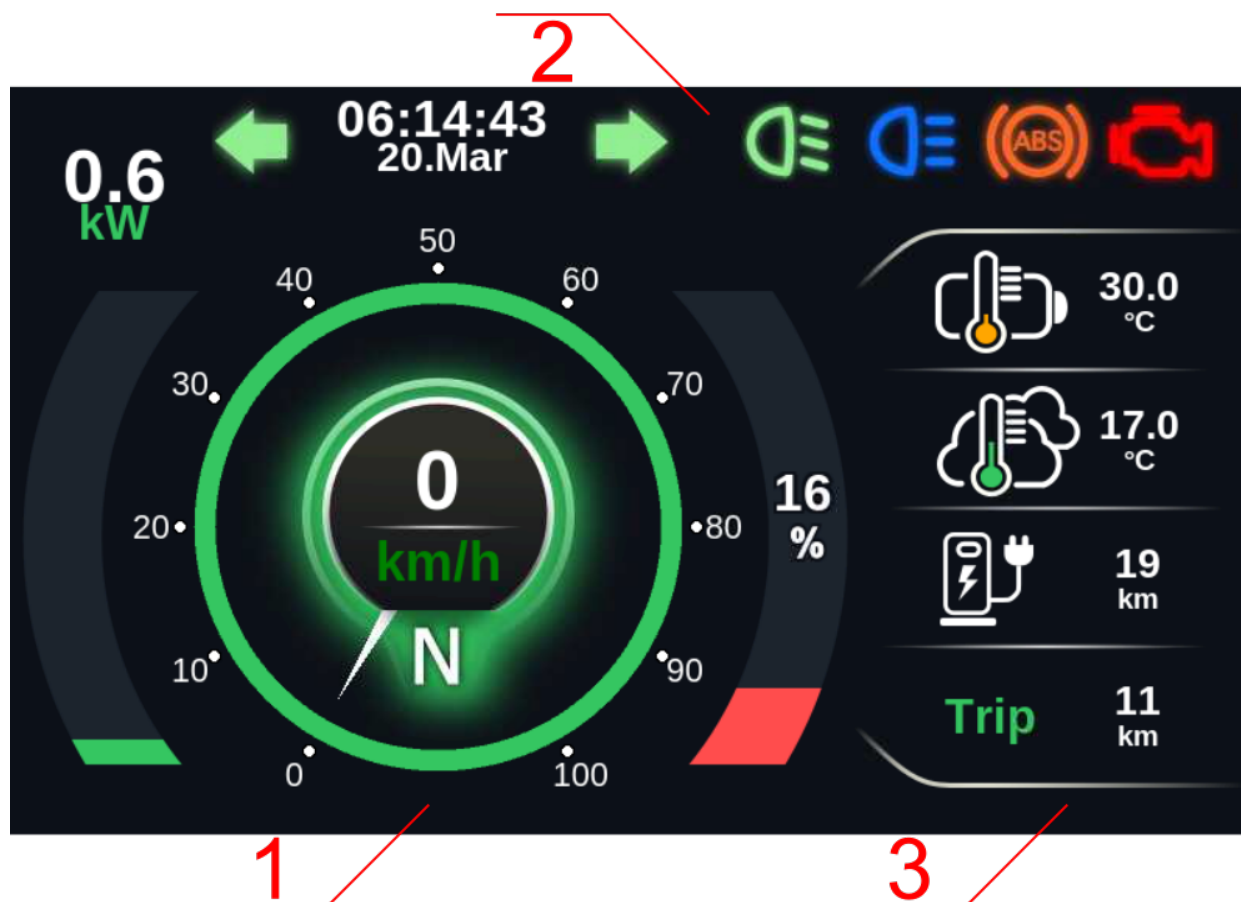
Pomocí těchto grafických interpretací byla stvořena veškerá grafika mého projektu.

4.4 Grafický design

Celkové grafické zobrazení je zhotovené pomocí tří panelů, které jsou přepínatelné pomocí ovládacích prvků na motocyklu. Každý panel zobrazuje specifické informace a v následujících krocích vysvětlím jejich funkčnost a schopnost nastavitelnosti a přizpůsobení vlastního vzhledu.

4.4.1 Hlavní panel

Hlavní obrazovka je nejdůležitější částí zobrazovacích funkcí displeje, jelikož bude primárně využívána při jízdě na motocyklu. Tento panel se nativně zobrazí při každém spuštění mikropočítače. Celý je rozdělený do tří hlavních částí.



Obrázek 17: Hlavní panel. Bod 1 znázorňuje nejdůležitější prvky obrazovky, bod 2 odkazuje na panel signalizačních prvků a bod 3 vyobrazuje aktuálně zvolené ikony v poli nabídky.

Hlavní informace

V první části se nachází nejdůležitější informace, které mají svoji váhu důležitosti upřednostněnou ve své velikosti zobrazených prvků.

V levé části se zde nachází grafický ukazatel hladiny aktuálního výkonu. Motocykl má funkci rekuperace, při které pracuje hnací elektromotor v režimu generátoru a vyrobená energie je zpět vrácena do akumulátoru. V tomto případě BMS jednotka vrací kladnou hodnotu proudu a zobrazovací hladina výkonu změní svoji barvu na červenou. Při odběru energie z akumulátoru je hodnota záporná a její barva je změněna zpět na barvu zelenou. Tato znaménková konvence je dána BMS jednotkou a nelze ji změnit. Dle normy by měla být kladná při odběru energie a záporná při jejím zpětném dodání. Textová přesná hodnota je zde nad grafickým ukazatelem ve formátu kW.

Podobný ukazatel jako je zobrazování výkonu se nachází napravo od zobrazovače rychlosti a definuje aktuální kapacitu baterie. Její hodnota se zde též zobrazuje pomocí hladiny grafického prvku a mění svoji barevnou kombinaci na základě aktuální hodnoty. Disponuje třemi typy barevného zobrazení. Zelená indikace se vyobrazuje při kapacitě akumulátoru od 100 % až do 50 %, oranžová od 50 % do 20 % a červená od 20 % do 0 %.

Posledním prvkem v této části je ukazatel aktuální rychlosti a vyznačení zařazeného převodu. Design má představovat klasický vzhled analogových budíků s ručičkou, které jsou doplněny o středový textový prvek. Zelený obrys pod číselníkem a vyznačené podsvícení na obrázku též mění své barevné kombinace v závislosti na dané rychlosti a přednastavených limitech. První barevná změna se provede při překročení rychlosti 70 km h^{-1} na barvu oranžovou a druhá při překročení rychlosti 90 km h^{-1} na barvu červenou. Tyto změny jsou vytvořeny v závislosti na krátkých časových intervalech plynulým přechodem mezi danými kombinacemi v rozsahu deseti mezibarev a nastavení určité hystereze, aby při přechodu mezi klíčovými rychlostmi barvy neustále nebalancovaly mezi sebou.

Kontrolky

V druhé části jsou zobrazeny především signalizační kontrolky. Jsou zde dvojí kontrolky směrových ukazatelů, mezi kterými se nachází aktuální čas s datem, kontrolky signalizující světla dálková a potkávací. Nakonec je zde systém ABS a kontrolka pro upozornění na určitou definovanou chybu.

Pole nabídky

V poslední třetí části napravo hlavní obrazovky se nachází pole ikon znázorňující dané informace spolu s textovou hodnotou a jejich jednotkami. Ikony, které zachycují teploty, zde zobrazují svoji hodnotu též graficky. Mají pomocí programu přednastaveny svoje maxima a minima pro své zobrazování a také svoji barvu při překročení určitých hodnot.

4.4.2 Funkce panelu nabídky

Při aktivaci ovládacího prvku „set“ po dobu 2 sekund se nám zobrazí panel výběrového menu.



Obrázek 18: Výběrové menu informací

Toto výběrové menu má v levé části obrazovky legendu veškerých kontrolků nacházejících se na displeji pro snazší orientaci. Hlavní funkcí tohoto menu je ale možnost vlastní nabídky jednotlivých dostupných ikon a jejich parametrů.

Pomocí nápovědy v levém horním rohu obrázku 19 můžeme při stisknutí tlačítek ovládacího panelu „up“ a „down“ pohybovat s šipkou nacházející se nalevo od nabídky. Tímto stylem výběru nasměrujeme šipku na určité pole, které budeme chtít pozměnit. Při opětovném stlačení ovládacího prvku „set“ pozměníme ikonu v daném poli za jednu ze sedmi dostupných ikon, která ještě v žádném poli není nevyužita, a původní se mezi aktuálně nevyužité zařadí pro další možný výběr. Tuto operaci můžeme zopakovat u dalších polí, dokud s výběrem všech položek nebudeme spokojeni. Stlačením prvku „return“ uložíme výběr a vrátíme se zpět na hlavní panel s již novou, námi zvolenou kombinací ikon.

Popis funkce

Při každém prvotním spuštění celého programu se rozkóduje informace z pole databáze na pole proměnné a odešle se požadavek o zobrazení informací do grafické části programu. Tam se zobrazí ikony, které obsahují z příslušné matice logické jedničky. Místa obsahující nuly jsou v danou chvíli skryta.

Po přidržení tlačítka „set“ po dobu dvou vteřin, se zobrazí možnost nabídky a spustí se funkce programu, která manipuluje s logickými hodnotami ikon. Pomocí tlačítek nahoru a dolů je uživatel schopen posouvat ve směru stlačení s šipkou, která vždy ukazuje na totožné pole, se kterým bude funkce aktuálně manipulovat. Při opětovném stlačení tlačítka „set“ budeme posouvat logickou jedničku mezi danými ikonami, které jsou aktuálně označeny logickou nulou. Směr posouvání jedničky je dán nejbližší hodnotou z matice možností obsahující nulu ve všech sloupcích, aby se informace neopakovala, jestliže je již přiřazena v jiném poli. Stará informace dostane vždy hodnotu logické nuly a zařadí se do fronty pro další možné přiřazení. Způsobem manipulace s těmito tlačítky si může uživatel navolit jakoukoliv kombinaci matice na daný panel nabídky. Pokud je s výběrem spokojený, aktivuje tlačítko „return“ a tím se matice zakóduje, uloží se do databáze a uživatel se též vrátí na hlavní panel.

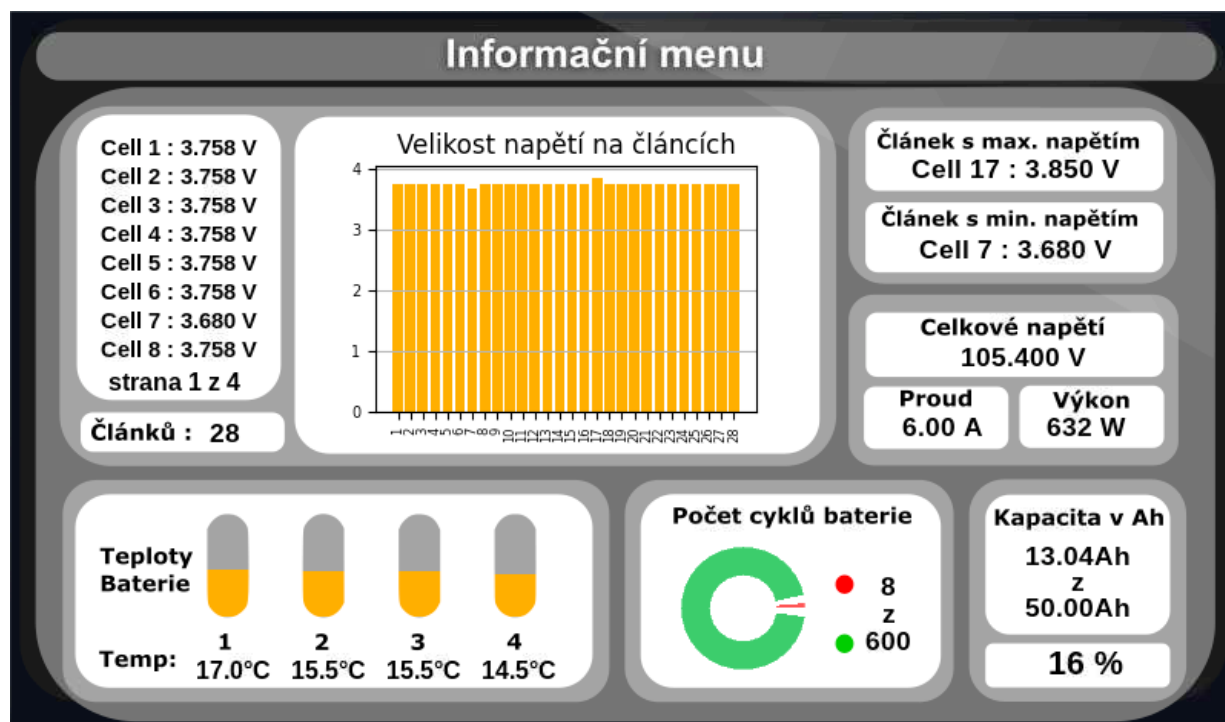
	teplota baterie	venkovní teplota	trip	spotřeba	dojezd	odometr	teplota motoru
Pole 1	1	0	0	0	0	0	0
Pole 2	0	0	0	0	1	0	0
Pole 3	0	0	0	0	0	1	0
Pole 4	0	0	0	1	0	0	0

Obrázek 19: Funkce nabídky znázorněná pomocí matice

Na obrázku je znázorněna matice, která odpovídá nastaveným parametrům obrázku 18.

4.4.3 Informační menu

Poslední panel nabízí důkladnější zjištění hlubších informací o motocyklu. Pro jeho zobrazení je nutná krátká aktivace tlačítka „set“ a pro návrat zpět na hlavní panel aktivace tlačítka „return“.



Obrázek 20: Informační menu. V horní části jsou vyobrazeny graficky i textově důležité informace o článcích akumulátoru. Vlevo dole jsou jednotlivé teploty, které jsou měřeny v prostorách akumulátoru. Koláčkový graf představuje počet cyklů a nakonec jsou zde textově zobrazena ostatní dostupná data z jednotky BMS.

V první části se nacházejí podrobné informace o akumulátoru. Při zahájení komunikace s BMS jednotkou se načtou data o počtu článků akumulátoru zapojených v sérii a program z této informace přizpůsobí v textové části o článcích počet možných stran pro zjištění jejich přesných hodnot napětí. Mezi těmito stranami lze přejíždět pomocí ovládacích prvků „up“ a „down“ umístěných na řídítkách motocyklu. Po zjištění dat o akumulátoru se též automaticky vygeneruje sloupcový graf, znázorňující jednotlivé články v závislosti na jejich napětí. Z tohoto grafu lze snadno a rychle vyčíst, který z článků už neposkytuje dostatečné napětí a který je tedy třeba vyměnit. V této části se rovněž napravo nachází textová informace o článku s maximálním a minimálním napětím.

Další část obsahuje měření teploty baterie. Jednotka BMS má kolem sebe umístěna teplotní čidla, díky kterým je schopna měřit teplotu baterie na čtyřech rozdílných místech. Tato informace je zde zobrazena textově a pomocí ukazatele hladiny. Na hlavním panelu v ikoně pro zobrazování stavu teploty baterie je zobrazována vždy nejvyšší hodnota z těchto teplot.

V tomto menu je dále vyobrazen počet nabitých cyklů v podobě koláčového grafu, který má přednastavený horní limit na hodnotu 600 cyklů.

V poslední části jsou vyobrazena ostatní textová data, která jsou vyhodnocována z BMS komunikace.

5 Testování

V této části zhodnocuji testování vytvořené přístrojové desky. Celkový zhotovený vzorek a jeho funkčnost byla předvedena vedoucímu bakalářské práce a byla schválena, jako plně funkční z hlediska všech jeho požadavků.

5.1 Testování v rámci simulace

Pro prvotní testování ovládacích prvků jsem mikropočítač připojil k běžným mikrospínačům a simuloval tak funkčnost zapojení na nepájivém poli.

Pro simulování získaných dat o otáčkách kola motocyklu byl připojen Hallův senzor menších rozměrů na testovací ventilátor. Do lopatky vrtule jsem připevnil permanentní magnet, který byl poté již zmíněným senzorem detekován. Tímto způsobem jsem byl schopen otestovat veškeré funkce programu, které byly závislé na měření otáček senzorem a simulovat tak motocykl v pohybu. Komunikace s BMS se nejprve prováděla na jednotce modelu 17s, která byla vyjmuta z funkčního vzorku elektrické koloběžky laboratoře emobility TUL. Desky jsou vytvářeny stejným výrobcem a jejich softwarové funkce jsou takřka stejné, tudíž lze software vytvořený v rámci bakalářské práce využívat naplno v odlišných verzích těchto BMS jednotek.



Obrázek 21: Testování měření parametrů získávaných z Hallova senzoru a funkčnost ovládacích tlačítek v rámci simulace

5.2 Porovnání realizovaného principu snímání rychlosti s komerčním tachometrem

Základní jízdní vlastnosti získané z databáze testování mého palubního počítače porovnávám s daty, které mi byly poskytnuty tachometrem značky Sigma, jež měřil totožný průběh celého měření [13]. Porovnání dat bylo uskutečněno na nosném podstavci, aby se dalo měřit na místě, jelikož motocykl spolu s přístrojovou deskou nebyl ještě dostatečně uzpůsobený k jízdě v běžném provozu.

Tabulka 4: Porovnání vytvořené práce s tachometrem Sigma

Společné parametry	Tachometr SIGMA	Vlastní řešení	Jednotky
Průměrná rychlost	38,8	38,77	[km h ⁻¹]
Ujetá vzdálenost	2,06	2,064	[km]
Maximální rychlost	78,8	79,17	[km h ⁻¹]
Minimální rychlost	6	0,712	[km h ⁻¹]
Čas měřeného úseku	3:08	3:08	[min]

Pomyslný měřený úsek činil 2 km. Ze získaných dat je zřejmé, že se hodnoty příliš neliší. Z tachometru byla pouze vizuálně vyčtena nejnižší rychlost 6 km h⁻¹, což je způsobeno nastavitelností od výrobce, a nikoliv chybou tachometru. Projekt z naměřené databáze vyčetl nejnižší rychlost v měřeném úseku 0,71 km h⁻¹. Nejvíce se liší nepatrným rozdílem hodnoty maximálních rychlostí. Tato chyba je nejspíše způsobena detekčními senzory rozdílných kvalit a jejich rychlostí reakce na magnetické pole. Tím pádem by se i aktuální vyšší hodnoty rychlosti lišily o nepatrný rozdíl. Tato chybovost by se projevila při nájezdu 10 000 km rozdílem 46 km za předpokladu počítaného rozdílu z naměřených maximálních rychlostí uvedených v tabulce 4.

6 Závěr

I přes nepříznivé podmínky nouzového stavu, který mě zpočátku omezoval v přístupu na univerzitu, se mi povedlo plně zhotovit funkční vzorek, kterým je přístrojová deska pro elektrické jednostopé vozidlo pomocí mikropočítače Raspberry Pi. Nedostatek času se odrazil na části testování, kdy motocykl s přístrojovou deskou nebyl schopen běžného provozu a nebyla tak možnost získávat reálná data k případnému zhodnocení. Veškeré testování probíhalo v rámci simulací nebo měření hodnot v prostorách laboratoře, kdy byl motocykl umístěn na nosném podstavci.

Při řešení praktické části jsem se potýkal s několika problémy. V části navrhování grafického designu jsem narazil na absenci důležité funkce antialiasingu při vykreslování grafických prvků přes modul Tkinter. Bez této funkce se oblé objekty jevíly poněkud nevyhlazeně a byl jsem nucen tento problém řešit překrytím dvou stejných grafických tvarů, přičemž prvek nacházející se v pozadí byl zvětšený o nepárné rozměry a jeho barevná kombinace byla nastavena na tmavší o 50 %, nežli původní. Tímto stylem jsem vyhladil veškeré tvary, které byly tvořeny graficky pomocí kódu a modulu Tkinter. Další problém se vyskytl při získávání reálných hodnot aktuálního času a data. Palubní počítač používá čas z operačního systému, a jelikož nedisponuje přídatnou baterií, která by čas udržovala aktuální, tak se vždy po odpojení počítače od napájení čas zastaví a naváže se až při budoucím zapnutí, čímž dojde k jeho celkovému zpoždění. Tento problém by se dal vyřešit přídatným zapojením RTC modulu pomocí GPIO pinů a jeho nakonfigurováním. Posledním větším problémem se jevílo vyobrazení sloupcového grafu článků závislejících na jejich napětí na informačním panelu. Pomocí zvoleného programovacího jazyka jsem nebyl schopen graf uzpůsobit tak, aby byl zároveň vystředěn v daném poli, v dostatečné velikosti a byl řádně popsán v oblasti jednotlivých os v tomto rozlišení.

Z hlediska vývoje celkové přístrojové desky se mi jevil jako nejobtížnější návrh grafického prostředí. Nejsložitější byl celkový design a umístění jednotlivých prvků, jakož i přiřazení veškerých funkcí a barev těmto prvkům. Pro tento projekt by se v budoucnu dala také vytvořit rozdílná barevná kombinace, která by byla využívána především v denním režimu. Pro její zobrazování by se jevílo jako vhodné využít výhradně světlé až bílé pozadí, na kterém by byly zobrazeny tmavě zbarvené prvky, které by byly lépe viditelné v případě většího oslnění displeje slunečním svitem.

Literatura

- [1] Adnan, Rukhiyah & Abd Samad, Nor Hafiza & Miserom, siti faizah. (2015). Raspberry Pi & Arduino What Makes Them Awesome!!!. https://www.researchgate.net/publication/287207118_Raspberry_Pi_Arduino_What_Makes_Them_Awesome/citation/download
- [2] ADVMOTO. Ducati Announces 2021 Multistrada V4 [online]. Legato Road, 2020 [cit. 2021-04-13]. Dostupné z: <https://adventuremotorcycle.com/news/2021-ducati-multistrada-v4>
- [3] AMOS, David. Real Python. Python GUI Programming With Tkinter [online]. [cit. 2021-04-13]. Dostupné z: <https://realpython.com/python-gui-tkinter/>
- [4] Arduino store. ARDUINO UNO REV3 [online]. [cit. 2021-04-13]. Dostupné z: <https://store.arduino.cc/arduino-uno-rev3>
- [5] BANZI, Massimo a Michael SHILOH. *Getting started with Arduino*. Third edition. Sebastopol, CA: MakerMedia, [2015]. ISBN 1449363334.
- [6] Finglai. NJK-5002D magnetic proximity sensor [online]. [cit. 2021-04-13]. Dostupné z: <https://www.finglai.com/products/sensors/magnetic-proximity-sensors/NJK-5002/NJK-5002D.html>
- [7] Gear patrol. TFT Displays Are the Next Motorcycle Technology Trend [online]. 2019 [cit. 2021-04-13]. Dostupné z: <https://www.gearpatrol.com/cars/motorcycles/a403115/tft-displays-are-the-next-motorcycle-technology-trend/>
- [8] Github. Bluetooth GATT SDK for Python [online]. 2017 [cit. 2021-04-13]. Dostupné z: <https://github.com/getsenic/gatt-python>
- [9] Last Minute engineers. Interfacing DS18B20 1-Wire Digital Temperature Sensor with Arduino [online]. [cit. 2021-04-13]. Dostupné z: <https://lastminuteengineers.com/ds18b20-arduino-tutorial/>
- [10] LIEBACK, Ron. Ultimate motorcycling. 2021 Ducati Multistrada V4 First Look [online]. 2020 [cit. 2021-04-13]. Dostupné z: <https://ultimatemotorcycling.com/2020/11/04/2021-ducati-multistrada-v4-v4-s-v4-sport-first-look-19-fast-facts/>
- [11] LLT POWER. 28S Li ion Battery BMS [online]. JingYe Industrial Park [cit. 2021-04-13]. Dostupné z: <https://www.lithiumbatterypcb.com/25s-to-32s-li-ion-or-lifepo4-battery-bms/>

- [12] Medium. Setting up Raspberry Pi to launch Python GUI app without Raspbian Desktop [online]. 2018 [cit. 2021-04-13]. Dostupné z: <https://medium.com/@daddycat/setting-up-raspberry-pi-to-launch-python-gui-app-without-raspbian-desktop-5022a90e5b63>
- [13] Moto technika. Digitální tachometr Sigma - MC 10 MOTO [online]. Gargulákova 34, Brno [cit. 2021-04-13]. Dostupné z: <https://www.mtt-brno.cz/tachometry-otackomery/2981-digitalni-tachometr-sigma-mc-10-moto.html>
- [14] Motorcycle.com. 2020 Honda Rebel 500 And 300 First Look [online]. Canadian corporation, 2020 [cit. 2021-04-13]. Dostupné z: <https://www.motorcycle.com/mini-features/2020-honda-rebel-500-and-300-first-look.html>
- [15] MURPHY, Kate. RideApart. 2020 TFT Screen Revolution [online]. 2019 [cit. 2021-04-13]. Dostupné z: <https://www.rideapart.com/features/386907/2020-motorcycle-tft-screen-revolution/>
- [16] Raspberry Pi. Raspberry Pi OS [online]. [cit. 2021-04-13]. Dostupné z: <https://www.raspberrypi.org/software/>
- [17] RPishop. Raspberry Pi 7" [online]. [cit. 2021-04-13]. Dostupné z: <https://rpishop.cz/displeje/243-7-oficialni-kapacitni-lcd-displej-800x480.html>
- [18] Shuo-Fang Liu, An-Sheng Lee, Yo-Wen Liang & Jenn-Yang Lin. A study on the influence of the designs and configurations of the displays on a motorcycle instrument panel on cognitive effects in Taiwan, *Journal of Information and Optimization Sciences*, 33:6, 685-700, DOI: 10.1080/02522667.2012.10700168
- [19] Svět počítačů. Raspberry Pi 3B+ 1GB RAM [online]. Přerov [cit. 2021-04-13]. Dostupné z: https://www.svetpocitacu.cz/imgcache/4/b/p669214-1436175-0a_-1_-1_98716.jpg
- [20] UPTON, Eben a Gareth HALFACREE. *Raspberry Pi: uživatelská příručka. 2.*, aktualizované vydání. Přeložil Jakub GONER. Brno: Computer Press, 2016. ISBN 978-80-251-4819-8.
- [21] VICS Promotion Council, VICS Promotion Council Safety Committee Report on Information Display. (Chapter 2 "Safety" Group).
- [22] Wierwille, W.W., Antin, J.F., Dingus, T.A., and Hulse, M.C. Visual Attentional Demand of an In-Car Navigation Display System in Vision in Vehicles II. Amsterdam: Elsevier Science B.V. (North-Holland) (1988), 307-316.
- [23] YOUNG, Emmet. PiMyLifeup. Setting up an SQLite Database on a Raspberry Pi [online]. 2021 [cit. 2021-04-13]. Dostupné z: <https://pimylifeup.com/raspberry-pi-sqlite/>