



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

Vytvoření firmware pro řídicí jednotku a HMI displej v elektrickém vozidle

Bakalářská práce

Studijní program: B2646 – Informační technologie
Studijní obor: 1802R007 – Informační technologie
Autor práce: **Jan Peča**
Vedoucí práce: Ing. Petr Bílek, Ph.D.
Konzultant: Ing. Lukáš Krčmář





Zadání bakalářské práce

Vytvoření firmware pro řídicí jednotku a HMI displej v elektrickém vozidle

Jméno a příjmení: Jan Peča
Osobní číslo: M15000049
Studijní program: B2646 Informační technologie
Studijní obor: Informační technologie
Zadávací katedra: Ústav mechatroniky a technické informatiky
Akademický rok: 2019/2020

Zásady pro vypracování:

1. Provedte rešerši již používaných zařízení, metod a postupů pro vývoj grafiky HMI displeje a seznamte se s programováním řídicí jednotky vybavené mikropočítačem s architekturou ARM.
2. Vyberte vhodný HMI displej s ohledem na použití v elektrickém vozidle (CAN sběrnice, velikost, cena). Připojte senzory (páčky blinkrů, plyn, brzda, světla a houkačka, teplota, vlhkost, rychlost) a akční členy (světla, houkačka) k řídicí jednotce a vše oživte.
3. Napište a odladte firmware pro řídicí jednotku v jazyce C. Vytvořte grafiku do vybraného displeje přes příslušný software. Zajistěte komunikaci řídicí jednotky po sběrnici CAN s ostatními jednotkami v elektrickém vozidle (snímač úhlu natočení volantu, HMI displej, řídicí jednotka motorů, správa baterie a nabíjení).
4. Otestujte na demo-boardu elektrického vozidla a zhodnoťte dosažené parametry.

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

dle potřeby dokumentace
30–40 stran
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] ŠTĚRBA, PAVEL. Elektrotechnika a elektronika automobilů. Praha: Computer press, 2004. 182 s. ISBN 80-251-0211-4.
- [2] ŽDÁNSKÝ, BRONISLAV. Elektrotechnika motorových vozidel II. Brno: Avid, 2006. 212 s. ISBN 978-80-87143-14-8.
- [3] Datasheety k vybraným jednotkám v elektrickém vozidle.

Vedoucí práce:

Ing. Petr Bílek, Ph.D.
Ústav mechatroniky a technické informatiky

Konzultant práce:

Ing. Lukáš Krčmář
Ústav mechatroniky a technické informatiky

Datum zadání práce:

10. října 2019

Předpokládaný termín odevzdání:

18. května 2020

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

Prohlášení

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

Jsem si vědom toho, že na mou bakalářskou 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 práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou 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 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.

1. června 2020

Jan Peča

Poděkování

Rád bych poděkoval panu Ing. Petru Bílkovi, Ph.D. za ochotu, cenné rady a vstřícnost při konzultacích a psaní rešerše této bakalářské práce.

Vytvoření firmwari pro řídicí jednotku a HMI displej v elektrickém vozidle

Abstrakt

Bakalářská práce se zabývá vývojem firmwaru pro řídicí jednotku a displej v elektrickém vozidle. Na vybrané vývojové desce byla realizována obsluha základních prvků v elektromobilu. Dále byl vtipován vhodný displej, který zde slouží jako palubní deska zobrazující informace o jízdě vozidla, stavu akumulátorů a elektromotoru. Pro tento displej bylo navrženo a vytvořeno grafické rozhraní. Následně byla vyzkoušena a realizována komunikace po sběrnici CAN.

Klíčová slova: elektromobilita, elektromobil, návrh grafiky displeje pro palubní desku, návrh grafiky, CAN sběrnice, mikropočítač ARM

Realization of firmware for control unit and HMI display in an electric vehicle

Abstract

The bachelor's thesis deals with the development of firmware for the control unit and display in an electric vehicle. The operation of the basic elements in the electric car was performed on a selected development board. Furthermore, a suitable display was selected, which serves as a dashboard showing information about the vehicle's driving, battery status and electric motor. A graphical interface was designed and created for this display. Subsequently, communication via the CAN bus was tested and implemented.

Keywords: electromobility, electromobile, graphics design for dashboard, graphics design. CAN bus, ARM microcontroller,

Obsah

Seznam zkratek	10
Úvod	11
1 Displeje a řídicí jednotka elektromobilu	12
1.1 Požadavky na řídicí jednotku	12
1.2 Komerční řídicí jednotky	12
1.3 Arudio Due	13
1.4 Požadavky na displej	13
1.5 Displeje dostupné v nabídce na trhu	14
1.5.1 Wecon PI 7 inch HMI : PI8070	17
1.5.2 Micromedia HMI 7	18
1.5.3 Weintek eMT3070B	19
1.5.4 Myir MY-EVC5100S-HMI	20
1.5.5 AMETEK VIS' C-COM 7GT	21
1.5.6 Opus A6 ECO BASIC (G2)	22
1.5.7 Riverdi RiTFT-50-IOT-CAP	23
1.5.8 4D systems GEN4-ULCD-50D-SB-AR	24
1.5.9 Grayhill 3D70	25
1.5.10 Nextion NX8048P070-011C-Y	26
1.5.11 STM32H747I-DISCO	27
1.6 Tipy pro vývoj grafiky HMI displejů	29
1.7 Osvědčené postupy při vývoji grafiky řídicího systému HMI	30
1.8 CAN Bus	31
1.9 Popis demo-boardu	32
1.9.1 Vstupy a výstupy	33
2 Realizace grafiky a firmwaru	34
2.1 Výběr a testování displejů	34
2.2 Návrh grafiky	36
2.3 Firmware řídicí jednotky	41
2.3.1 Ovládací prvky	42
2.3.2 Komunikace řídicí jednotky s displejem	43
3 Závěr	45
Seznam použité literatury	45

Seznam obrázků

1.1	Vývojová deska Arduino Due	13
1.2	Displej od společnosti Acewell	15
1.3	Displej od společnosti SiAECOSYS	16
1.4	Displej od společnosti QSMOTOR	16
1.5	Displej od společnosti Xingyuhe	16
1.6	Bafang 850C	17
1.7	Wecon PI 7 inch HMI : PI8070	18
1.8	Mikromedia HMI 7	19
1.9	Weintek eMT3070B	20
1.10	Myir MY-EVC5100S-HMI	21
1.11	AMETEK VIS' C-COM 7GT	22
1.12	Opus A6 ECO Basic	23
1.13	RiTFT-50-IOT-CAP	24
1.14	4D systems GEN4-ULCD-50D-SB-AR	25
1.15	Grayhill 3D70	26
1.16	Nextion NX8048P070-011C-Y	27
1.17	STM32H747I-DISCO	28
1.18	Winstar WF70A8SYAHLNN0	29
1.19	Blokové schéma demo-boardu	32
1.20	Demo-board	33
2.1	Návrh kruhového ukazatele	36
2.2	Návrh půlkruhového ukazatele	37
2.3	Vytváření půlkruhového ukazatele	37
2.4	Vytváření půlkruhového ukazatele II	38
2.5	Vytváření půlkruhového ukazatele III	38
2.6	Vytváření půlkruhového ukazatele IV	39
2.7	Uvažované zobrazení stavu akumulátoru	39
2.8	Finální návrh grafiky displeje	40
2.9	Zjednodušený vývojový digram pro řídicí jednotku	41
2.10	Zjednodušený vývojový digram pro displej	44

Seznam tabulek

1.1	Bafang 850C parametry	17
1.2	Wecon PI8070 parametry	18
1.3	Mikromedia HMI 7" parametry	19
1.4	Weintek eMT3070B parametry	20
1.5	Myir MY-EVC51000S-HMI parametry	20
1.6	Ametek parametry	21
1.7	Opus A6 parametry	22
1.8	RiTFT-50-IOT-CAP parametry	23
1.9	4D systems GEN4-ULCD-50D-SB-AR parametry	24
1.10	Grayhill 3D70 parametry	25
1.11	Nextion NX8048P070-011C-Y parametry	26
1.12	STM32H747I-DISCO parametry	27
1.13	Winstar WF70A8SYAHLNN0	28

Seznam zkratek

CAN	Controller Area Network
ECU	Electronic Control Unit
GUI	Graphical User Interface
HMI	Human Machine Interface
IOT	Internet of Things
IPS	In-plane Switching
LVDS	Low-Voltage Differential Signaling
MIPI DSI	MIPI Display Serial Interface
PLC	Programamable Logic Controller
PWM	Pulse-Width Modulator
USB	Universal Serial Bus
VCU	Vehicle Control Unit

Úvod

Jedním z cílů této práce bylo vytvoření firmwaru pro řídicí jednotku v elektromobilu. Řídicí jednotka poskytuje zastřešení nad komunikací, která probíhá v rámci automobilu, zaznamenává provozní data a případné chyby. Firmware vyvíjený v rámci této práce se stará o komunikaci se senzory a dalšími jednotkami ve vozidle. K řídicí jednotce jsou připojeny senzory od ovladačů směrovek, světel, zámků, plynového a brzdového pedálu. Dalšími jednotkami se pak rozumí displej, který plní funkci palubní desky, správa baterie a nabíjení a řídicí jednotka motorů. V rámci této práce byla řešena komunikace mezi řídicí jednotkou a displejem. Nicméně komunikace mezi řídicí jednotkou a dalšími jednotkami by probíhala na stejném principu, jelikož rovněž využívá sběrnici CAN.

Dalším cílem této bakalářské práce byl výběr displeje. Tento displej bude plnit funkci palubní desky v elektromobilu. Na tomto displeji bude zapotřebí zobrazovat informace týkající se vozidla. Mezi tyto informace patří: rychlost, spotřeba, dojezd, zařazený rychlostní stupeň a další. Tyto informace jsou z velké části totožné jako u klasických automobilů se spalovacím motorem. Rozdílem zde bude nutnost zobrazení informací o akumulátoru, který slouží pro pohon. Při výběru displeje bylo potřeba brát ohled na různé parametry, které by měl daný displej splňovat. Těmito parametry jsou svítivost, velikost, podporovaná rozhraní a cena. Po vybrání vhodného displeje bylo dalším úkolem navrhnout a zrealizovat grafiku pro palubní desku. Při návrhu grafického rozhraní bylo potřeba zakomponovat všechny potřebné informace, které by měly být uživateli (řidiči) zobrazeny. Tyto informace bylo potřeba zobrazit přehledně a pro řidiče co nejvíce užitečně.

Po dokončení obou předchozích částí, bylo dalším úkolem propojení těchto částí. To obnášelo fyzické připojení obou částí ke sběrnici CAN a následně bylo potřeba provést propojení po stránce programové.

1 Displeje a řídicí jednotka elektromobilu

1.1 Požadavky na řídicí jednotku

Rozhraní

Hlavním z požadovaných rozhraním, které musela řídicí jednotka resp. vybraná vývojová platforma poskytovat, bylo rozhraní sběrnice CAN.

Vstupy a výstupy

Dále bylo potřeba, aby tato platforma umožňovala připojení jak digitálních tak analogových vstupů. Výstupy zde stačily pouze digitální.

Cena

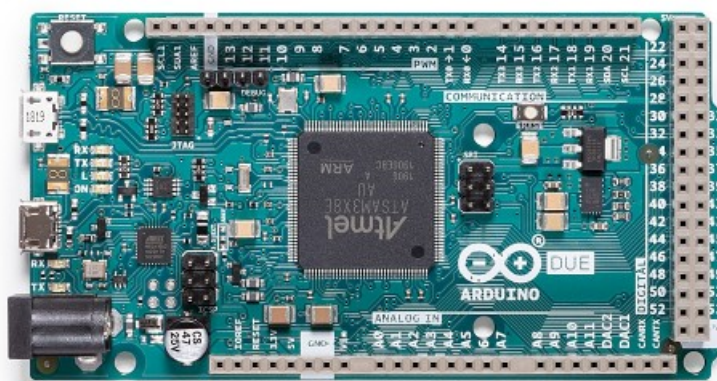
Bylo vhodné, aby se cena vybrané platformy pohybovala do 200 USD. Tato hranice vyřadila možnost využít již naprogramovaných jednotek.

1.2 Komerční řídicí jednotky

Na trhu je možné pořídit již naprogramované řídicí jednotky, které by šlo pro tuto práci jistě využít. V případě pořízení takovéto jednotky by ji stačilo připojit a podle potřeb nakonfigurovat. Toto řešení má ovšem velkou nevýhodu a to z pohledu jeho finanční náročnosti. Cena těchto jednotek se totiž pohybuje v rozmezí 1500–5000 USD, což značně převyšuje náklady na vytvoření vlastní řídicí jednotky. Pro porovnání zde bude uvedeno několik řídicích jednotek dostupných na trhu. Tyto jednotky jsou vybavené vlastním mikroprocesorem a jsou určené přímo pro oblast automotive. Většina z nich nabízí minimálně dvě CAN sběrnice. Zároveň tyto jednotky disponují analogovými i digitálními vstupy a výstupy. Jednotky nabízejí také PWM výstupy. Zde bude uvedeno jmenovitě několik VCU. Pi Innovo M110, M220, Continental gVCU, ECOTRONS ES1274A, SIGRATECH Electric Vehicle Powertrain Control Unit

1.3 Arudio Due

Zvolenou variantou pro vývoj firmwaru byla vývojová deska Arduino Due. Tato vývojová deska je založena na 32-bitovém ARM mikroprocesoru a nabízí 54 pinů pro vstup a výstup. Z těchto výstupů poskytuje 12 možností PWM výstupu. Dále poskytuje 12 analogových vstupů. Požadované rozhraní CAN bylo poskytováno rozšiřujícím modulem. Pro vývoj je k dispozici vývojové prostředí Arduino IDE. Tohoto prostředí bylo využito při vývoji firmwaru.



Obrázek 1.1: Vývojová deska Arduino Due
[33]

1.4 Požadavky na displej

Rozhraní

Jedním z prvních požadavků na displej bylo, aby podporoval komunikaci po sběrnici CAN. Toto vyplynulo již ze samotného zadání práce, resp. zamýšleného využití displeje. Po provedení počáteční rešerše se ukázalo, že možná bude potřeba z tohoto požadavku slevit. A to z důvodu, že se na trhu nenacházelo velké množství displejů, které by podporovaly komunikaci pomocí CAN protokolu a zároveň by splňovaly i další z požadavků. Také bylo usouzeno, že bude lepší se zaměřit nejdříve na kvalitu displeje jako takového a až následně řešit, zda mimo běžné komunikace po sériové lince podporuje i komunikaci po sběrnici CAN. A tak v případě nalezení dostatečně kvalitního a vyhovujícího displeje, který by neumožňoval komunikaci po CAN sběrnici, by byl tento nedostatek vyřešen vytvořením převodníku. Toto rozhodnutí značně rozšířilo množství displejů, ze kterých bylo následně možné vybírat.

Velikost

Výběr vhodné velikosti displeje byl také důležitý. Bylo potřeba vybrat optimální velikost, aby byla zajištěna dostatečná viditelnost a čitelnost údajů. Zároveň však nesměl být tento displej příliš velký. Jako vhodné se zde ukázaly displeje o úhlopříčkách 5" a 7" palců. V úvahu by také přicházely extra širokoúhlé displeje.

Odolnost

Vybraný displej musí být odolný vůči různým druhům rušení. Tento problém však vyřešilo již to, že výsledný displej byl vybírán z kategorie HMI displejů. Tyto displeje jsou určeny primárně pro použití v továrnách, kde se s možností rušení již počítá.

U displeje je také důležitá teplotní odolnost. Pro případ této práce by vyhovovala teplotní odolnost v rozsahu $-30-80^{\circ}\text{C}$.

Svítivost

Dalším důležitým parametrem byla svítivost displeje. Vybraný displej musel poskytovat dostatečnou míru svítivosti pro zajištění dobré čitelnosti i při vysokém okolním světle či přímém slunečním záření. Jako minimální hranice, zde byla určena svítivost 1000 cd/m^2 . Displej by měl dále umožňovat nastavení intenzity svítivosti. Například v noci či v prostředí tunelu tuto svítivost redukovat na pro řidiče příjemnou intenzitu. U displeje byl také požadován dobrý pozorovací úhel. Za displej takový by šlo v tomto případě označit každý s pozorovacím úhlem 120° a výše.

Cena

Poslední aspektem při výběru byla cena. Zde bylo vhodné udržet maximální cenu displeje do 200 USD. Vysoká cena by pak způsobila přílišné zvýšení nákladů na výrobu výsledného elektromobilu.

1.5 Displeje dostupné v nabídce na trhu

Jedním z úkolů této práce byl výběr displeje, který by byl vhodný pro využití v malém městském elektromobilu. Při výběru displeje bylo vhodné se pohybovat v oblasti HMI displejů. V této rešerši je uveden výčet displejů dostupných na trhu. Mezi těmito displeji je i několik pouze pro zajímavost a ukázkou, co vše je v této době dostupné. Některé z níže uvedených displejů nabízí i dotykové rozhraní, což ale nebylo vyžadováno.

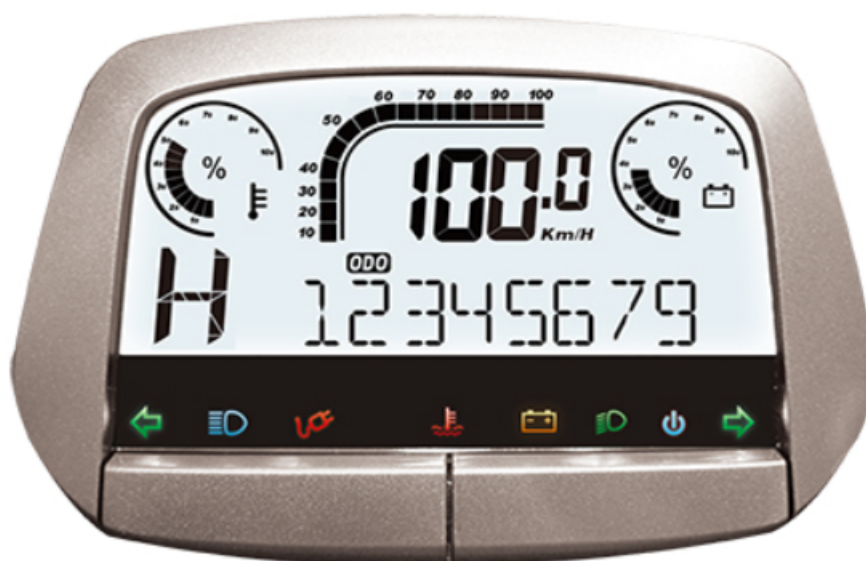
HMI

Zkratku HMI by šlo přeložit jako rozhraní mezi strojem a člověkem. HMI displeje poskytují textový či grafický pohled na stav systému a operací, které na něm probíhají. Pomocí HMI lze tak stroje (systémy) monitorovat, kontrolovat jejich stav

a mnoho dalších akcí. Tyto displeje jsou používány například ve výrobních halách jako grafické rozhraní jednotlivých strojů. [31]

Předdefinované displeje

Dalším druhem displejů jsou displeje, na kterých je forma grafického výstupu již předdefinována. A to ať už ve formě pevného textového či číselného výstupu, nebo ve formě kontrolky. U těchto displejů pak stačí pouze provést jeho správné připojení. Bohužel zde není možnost vlastní grafické úpravy. Tento typ displejů nebyl ani uvažován pro vývoj, slouží pouze jako ukázka. Takový typ by šlo například použít v případě, kdy výrobci automobilu postačí graficky jednodušší a zároveň levnější alternativa. Na následujících obrázcích budou zobrazeny některé z předdefinovaných displejů.



Obrázek 1.2: Displej od společnosti Acewell [16]



Obrázek 1.3: Displej od společnosti SiAECOSYS [17]



Obrázek 1.4: Displej od společnosti QSMOTOR [18]



Obrázek 1.5: Displej od společnosti Xingyuhe [19]

Bafang 850C

Hlavním zaměřením tohoto displeje jsou elektrická kola. Zde slouží pro zobrazení informací cyklistům. Displej disponuje IPS panelem, jehož úhlopříčka je 3,2", což je vzhledem k oblasti jeho využití vyhovující. Dále podporuje komunikaci pomocí CAN protokolu, vyhovuje certifikaci IP65 a jeho provozní teplota je výrobcem určena v rozmezí -20 °C - 70 °C. Cena tohoto displeje se pohybuje okolo 85 USD.

Tabulka 1.1: Bafang 850C parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
3,2"	-20°C až 70°C	CAN	—	85 USD



Obrázek 1.6: Bafang 850C
[3]

1.5.1 Wecon PI 7 inch HMI : PI8070

Tento HMI displej disponuje panelem s 16 miliony barev o úhlopříčce 7 palců. Zařízení umožňuje komunikaci pomocí sběrnice CAN. Mezi další rozhraní patří sériové COM porty, Ethernet a USB. Základní paměť typu FLASH o velikosti 2 GB je možné rozšířit pomocí SD karty až o 32 GB. Dále tento displej splňuje certifikaci IP65 ovšem pouze z přední strany.[5]

PIStudio

Toto vývojové prostředí je přímo od výrobce displeje Wecon. Nabízí také množství již předpřipravených prvků, které lze jednoduše použít. U některých je i možnost částečného přizpůsobení přímo z vývojového prostředí.

Tabulka 1.2: Wecon PI8070 parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-10°C až 60°C	CAN, COM, Ethernet, USB	250	220 USD



Obrázek 1.7: Wecon PI 7 inch HMI : PI8070 [5]

1.5.2 Micromedia HMI 7

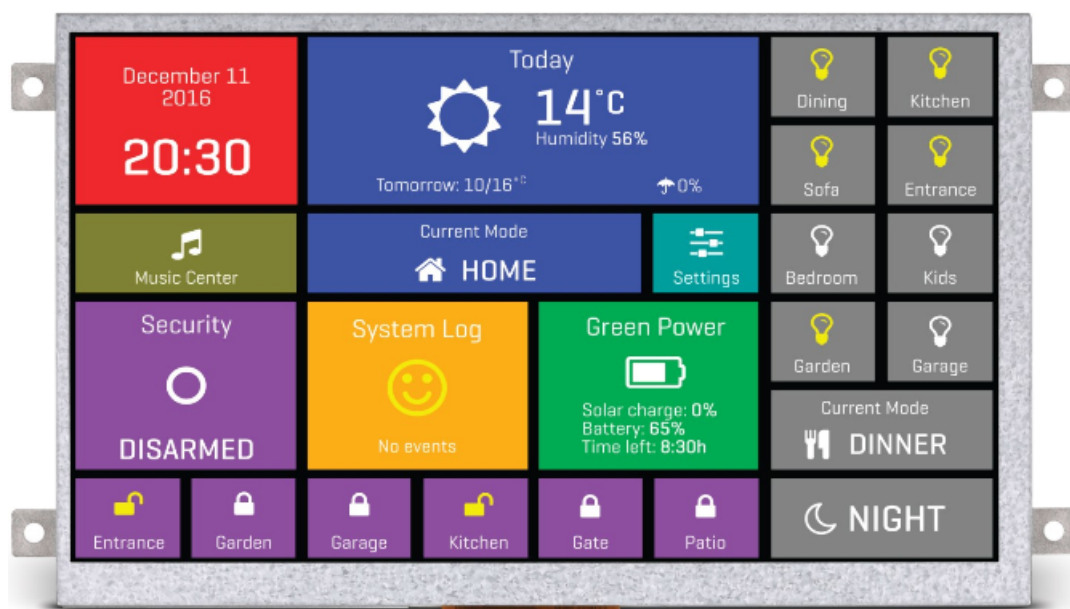
Jedná se o HMI displej s úhlopříčkou 7 palců vybavený panelem Riverdi s barevnou hloubkou 16,7 milionu. Tento displej je vyráběn srbskou společností Mikroee. Tato společnost se zabývá produkcí celé řady nástrojů pro všechny hlavní architektury mikrokontrolérů. Interní paměť 8 MB lze díky přítomnosti slotu na microSD kartu dále rozšířit. U tohoto konkrétního modelu se jedná o displej bez dotykového panelu, což je ale s ohledem na jeho plánované využití dostačující. V následující tabulce 1.3 jsou uvedeny parametry daného displeje. Jako vývojový nástroj pro displeje firmy Mikroee jsou k dispozici kompilátory pro programovací jazyky C, Pascal a Visual Basic. K využití všech periférií je zapotřebí rozšiřující deska, která obsahuje mimo jiné i porty microUSB a RJ-45. Pomocí této desky lze k displeji například připojit zařízení, které komunikuje po sběrnici CAN nebo po sériové lince.[1]

Visual TTF

Visual TTF je samostatná aplikace používána pro rychlý vývoj GUI (grafických uživatelských rozhraní) pro displeje. Tato aplikace generuje kód kompatibilní s kompilátory (microC, microBasic, microPascal). Software nabízí uživatelsky přívětivé a přehledné prostředí. Dále obsahuje mnoho drag-and-drop komponent, díky čemuž lze využít pro rychlé vytváření aplikací.[21]

Tabulka 1.3: Mikromedia HMI 7" parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-10°C až 60°C	CAN, COM	400	151 USD



Obrázek 1.8: Mikromedia HMI 7
[1]

1.5.3 Weintek eMT3070B

Společnost Weintek je předním světovým výrobcem HMI. Tento displej je vybaven panelem o úhlopříčce 7 palců s barevnou hloubkou 16,7 milionu. Dále tento displej disponuje certifikací IP66 z jeho přední strany. Interní paměť o velikosti 256 MB je možné rozšířit díky slotu na SD/SDHC kartu. Displej dále disponuje porty CAN, COM, Ethernet a USB. K vytváření aplikací pro tento displej je výrobce dodáváno vývojové prostředí Easy Builder PRO.[22]

Easy Builder PRO

Tento program nabízí přívětivé prostředí pro vývoj HMI aplikací. Mezi jeho přednosti patří například rychlá změna cílového HMI. Je tak možné jen díky pár krokům spustit projekt na různých HMI s různými velikostmi displeje. V neposlední řadě je možné provádět i simulace bez nutnosti připojení k HMI či PLC. A také toto prostředí umožňuje velké možnosti v přizpůsobení jednotlivých prvků přímo z vývojového prostředí bez potřeby psaní zdrojového kódu, či nutnosti vytváření vlastních grafických součástí.

Tabulka 1.4: Weintek eMT3070B parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-20°C až 50°C	CAN, COM, Ethernet	500	570 USD



Obrázek 1.9: Weintek eMT3070B
[2]

1.5.4 Myir MY-EVC5100S-HMI

MYIR Tech je globální poskytovatel ARM hardware a softwarových nástrojů. Tento displej je opatřen panelem o úhlopříčce 7 palců. Základní paměť 256 MB je možné dále rozšířit TF kartou. Díky tomu, že zařízení je označené jako Linux-ready je možné pro vývoj aplikací využít vývojové prostředí Qt. Displej disponuje množstvím rozhraní. Mezi tato rozhraní patří CAN, COM, Ethernet, USB, Wifi či bluetooth. Displej umožňuje také možné vložení SIM karty.[23]

Qt

Qt je uživatelsky přívětivé vývojové prostředí, které umožňuje tvorbu vlastních aplikací. Qt je kompletní sada nástrojů s rozsáhlými knihovnamy a uživatelsky přívětivými nástroji pro návrh designu. [11]

Tabulka 1.5: Myir MY-EVC5100S-HMI parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-20°C až 70°C	CAN, COM, Ethernet, 4G, WIFI, Bluetooth	1000	239 USD



Obrázek 1.10: Myir MY-EVC5100S-HMI
[10]

1.5.5 AMETEK VIS' C-COM 7GT

Tento displej je vybaven panelem o velikosti úhlopříčky 7" a barevnou hloubku 262 tisíc barev. Provozní i skladovací teplota tohoto displeje je v rozmezí -40 °C - 85 °C. Dále splňuje certifikaci IP67 a to jak z přední, tak i ze zadní strany displeje. Mezi rozhraní, která tento displej podporuje, patří CAN, USB, Ethernet.[24]

Qt

K vývoji aplikací pro tento displej je možné použít nástroj Qt. Tento nástroj je již stručně popsán výše.1.5.4

Tabulka 1.6: Ametek parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-40°C až 85°C	CAN, COM, Ethernet	800	— USD



Obrázek 1.11: AMETEK VIS' C-COM 7GT
[4]

1.5.6 Opus A6 ECO BASIC (G2)

Další displej s velikostí úhlopříčky 7". Provozní teplota tohoto displeje je od -30 °C do 65 °C. Tento displej nabízí USB rozhraní a vnitřní úložiště o velikosti 2 GB. Mezi další podporovaná rozhraní patří CAN, COM a Ethernet.[25]

Wachendorff projektor tool

V rámci tohoto nástroje je pro tento displej k dispozici množství widgetů přímo z oblasti automobilů.

Tabulka 1.7: Opus A6 parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-30°C až 65°C	CAN, COM, Ethernet	400	1315 USD



Obrázek 1.12: Opus A6 ECO Basic
[6]

1.5.7 Riverdi RiTFT-50-IOT-CAP

Tento displej od společnosti Riverdi je společností uváděn jako první na trhu s označením IoT. Panel má úhlopříčku 5". Mezi rozhraní, která tento displej podporuje, patří Wifi, bluetooth a USB. Displej je možné programovat v jazyce python a pro tento případ je k dispozici Zerynth studio, jehož integrovanou licenci již displej obsahuje.[26]

Zerynth studio

Zerynth studio je vývojové prostředí pro programování zařízení v programovacím jazyce python. Mezi výhody tohoto prostředí patří automatické rozeznání a nastavení zařízení, kde není potřeba žádné nastavování. Dále nabízí jednoduchou virtualizaci zařízení a nahrání programu. V tomto prostředí nejsou k dispozici předdefinované součásti a tak je potřeba realizovat grafiku jen z prostředí zdrojového kódu.

Tabulka 1.8: RiTFT-50-IOT-CAP parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
5"	-20°C - 70°C	Wifi, Bluetooth, USB	510	109 USD



Obrázek 1.13: RiTFT-50-IOT-CAP
[7]

1.5.8 4D systems GEN4-ULCD-50D-SB-AR

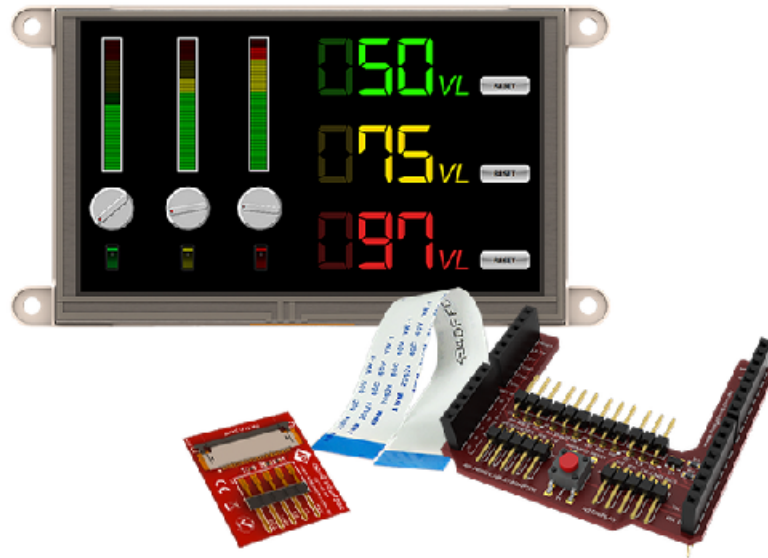
Displej od společnosti 4D systems je vybaven panelem od úhlopříčce 5". Výrobce poskytuje k tomuto displeji Arudiono Adaptor Shield, který nabízí rychlé a jednoduché rozhraní k různým typům desek Arduino. Tento displej je dále vybaven slotem na SD kartu pro rozšíření vnitřní paměti.[27]

4D Workshop4

Vývojové prostředí 4D Workshop je vyvíjené společností 4D systems. V rámci tohoto prostředí je k dispozici spousta předdefinovaných prvků pro snadné použití.

Tabulka 1.9: 4D systems GEN4-ULCD-50D-SB-AR parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
5"	-20°C až 70°C	I2C, SPI	820	129 USD



Obrázek 1.14: 4D systems GEN4-ULCD-50D-SB-AR
[8]

1.5.9 Grayhill 3D70

Úhlopříčka panelu displeje je 7". Předností tohoto panelu je zejména jeho vysoká svítivost. Za zmínění stojí také to, že tento displej splňuje specifikaci IP67 a to jak z přední tak i ze zadní strany. Mezi rozhraní, která jsou tímto displejem podporována, patří CAN, USB, Ethernet. Dále displej obsahuje 4 GB paměti. Pro vývoj jsou k dispozici různé aplikace mezi které například patří Codesys, VUI Builder a Qt. [28]

Qt

Toto vývojové prostředí je již stručně popsáno v dřívější části této rešerše.1.5.4

Tabulka 1.10: Grayhill 3D70 parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-30°C až 65°C	CAN, USB, COM, Ethernet	1000	560 USD



Obrázek 1.15: Grayhill 3D70
[9]

1.5.10 Nextion NX8048P070-011C-Y

Tento displej je vybaven dotykovým panelem o úhlopříčce 7" 65 tisíci barvami. Displej umožňuje komunikaci po sériové lince. Konkrétně tento displej pochází série označované výrobce jako Intelligent. Mimo této série jsou k dispozici série Basic a Enhanced. Tyto série jsou rozlišeny výkoností hardwaru. Displej je vybaven vnitřní pamětí o velikosti 120 MB. Dále je k dispozici slot pro SD kartu na případné rozšíření paměti.[29]

Nextion Editor

Nextion editor slouží pro vývoj grafických prostředí. I v rámci tohoto prostředí jsou k dispozici již předpřipravené widgety, které lze následně různě modifikovat.

Tabulka 1.11: Nextion NX8048P070-011C-Y parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-20°C až 75°C	COM	300	104 USD



Obrázek 1.16: Nextion NX8048P070-011C-Y
[20]

1.5.11 STM32H747I-DISCO

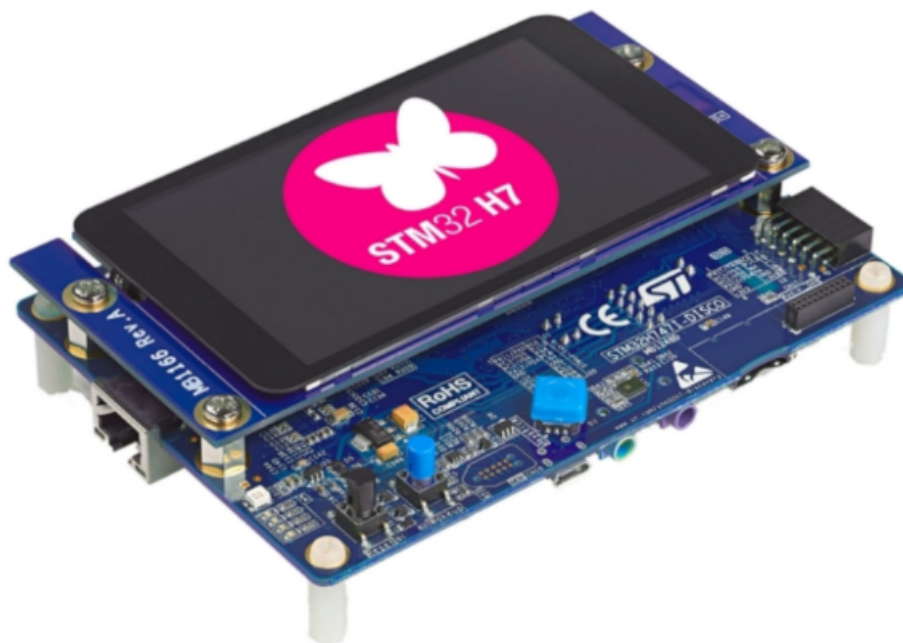
V základní verzi je tento kit vybaven displejem o velikosti 4". Tento kit nabízí širokou škálu podporovaných rozhraní. Mezi tato rozhraní patří CAN, Ethernet, slot pro SD kartu, microUSB a další. Nejdůležitější vlastností tohoto kitu je podpora komunikace po sběrnici CAN. Displej je v rámci tohoto kitu připojen pomocí MIPI[®] DSI rozhraní. Prostřednictvím tohoto rozhraní je možné připojení displejů až s rozlišením 720p. Pro tvorbu aplikací je dispozici celá řada vývojových nástrojů. Jedná se například o vývojová prostředí Keil uVision, TouchGFX Environment, Embedded Wizard studio.[30]

TouchGFX Designer

Vývojové prostředí TouchGFX Designer je jednoduché a uživatelsky přívětivé. V základu nabízí množství již definovaných widgetů, které lze použít, bez potřeby psaní kódu. Prostředí nabízí i jistou možnost přizpůsobení těchto widgetů, kdy stále není nutnost psaní jakékoliv kódu. Ale v případě náročnějších požadavků je již potřeba zásah i do oblasti kódu.

Tabulka 1.12: STM32H747I-DISCO parametry

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
4"	-40°C až 85°C	USB, COM, Ethernet	—	100 USD



Obrázek 1.17: STM32H747I-DISCO
[12]

Displej pro STM32H747I-DISCO

Jako vhodný displej, který by mohl sloužit po připojení k tomu kitu jako finální varianta se jevil displej od společnosti Winstar. Tento displej disponuje svítivostí 1100 cd/m². Dále výrobce uvádí, že displej je čitelný z každého úhlu. Teplotní odolnost displeje -30–80 °C. Displej je vybaven vstupem LVDS. Jelikož vývojářský kit poskytuje MIPI[®] DSI bude zapotřebí využití převodník mezi těmito rozhraními.[32]

Tabulka 1.13: Winstar WF70A8SYAHLNNO

Úhlopříčka	Provozní teplota	Rozhraní	Svítivost [cd/m ²]	Cena
7"	-30°C až 80°C	LVDS	1100	25 USD



Obrázek 1.18: Winstar WF70A88SYAHLNNO
[32]

1.6 Tipy pro vývoj grafiky HMI displejů

Udržovat jednoduchost

Pro vytváření vysoce výkonné HMI aplikace je důležité udržovat jednoduchost. To zahrnuje zobrazení pouze toho, co je pro uživatele důležité. Neočekávat, že uživatel bude vědět, co jednotlivé hodnoty v rámci procesu znamenají, na místo toho vhodně tyto hodnoty zobrazit do správně zvoleného rozmezí.[13]

Dodržet jednoduchost bude žádoucí i v rámci této práce. Nejdříve si zvolit nejdůležitější z prvků, které bude potřeba zobrazit pro uživatele. Poté provést zhodnocení přehlednosti.

Konzistentnost

Pokud jsou již například v továrně k dispozici jiná HMI, tak bude vhodné, aby nově vytvářený systém byl co nejvíce podobný těm stávajícím. Ať už se jedná o umístění ovládacích prvků, použité barevné schéma, a další. Pokud se jedná o vývoj úplně nové aplikace, je zde větší prostor pro kreativitu. Ale i v tomto

případě je potřeba dodržovat nějakou formu konzistence alespoň mezi jednotlivými obrazovkami.[13]

V případě této práce bude možné považovat za jistou formu inspirace zvyklosti z jiných automobilů a to, ať elektrických, tak klasických se spalovacím motorem. Nebylo by totiž rozumné vnášet nějakou vlastní formu rozložení, která se nebude alespoň částečně podobat zažitému standardu. Nebo například vymýšlet úplně vlastní reprezentaci různých prvků.

Použití přiměřeného počtu barev

Zvolit vhodný počet barev, aby bylo umožněno efektivně zachytit pozornost uživatele. Například barvy určené pro upozornění by měly být používány pouze za tímto účelem.[13]

Toto doporučení by se dalo převzít i pro případ této práce. Výsledná grafická podoba by tak měla využít přiměřeného počtu barev. Díky tomuto pak půjdu jasně a zřetelně upozornit uživatele, v tomto případě řidiče, například na nutnost nabití akumulátoru, či jiného technického problému.

1.7 Osvědčené postupy při vývoji grafiky řídicího systému HMI

Přesnost

Při vytváření grafiky je důležité začít s aktualizovanými požadavky a přesnými informacemi ohledně vytvářené aplikace. Vytváření neaktuální či nepřesné grafiky bude mít za následek nutnost přepracování a může vést i ke zmatku.[14]

Toto doporučení by mělo být snad nezbytností, platící obecně pro většinu projektů. V této práci bude vhodné začít nejdříve jen s náčrtem na papír, který by se následně mohl přenést do samotné grafiky displeje. Zde by se nejdříve vyřešilo, zda lze všechny požadované prvky umístit na obrazovku v rozumné velikosti. Pokud by se nepodařilo úspěšně umístit všechny požadované prvky, přišlo by na řadu nejdříve zmenšení v nějakém poměru, případně by pak muselo dojít k odebrání určitých prvků a jejich přesunutí na další obrazovku.

Vizuální logika

Jako další je potřeba promyslet nejlepší logické rozvržení procesů do grafiky. Zde se typicky používá rozdělení jednotlivého procesu do více oblastí. Tyto oblasti pak reprezentují jednotlivé části tohoto procesu. Grafika v celkovém náhledu by měla zobrazovat všechny interakce mezi všemi zařízeními a k těmto interakcím zobrazit pouze nejdůležitější informace. Z toho celkového zobrazení by měl být umožněn přístup k detailnějším informacím ohledně každé z interakcí.[14]

Použití animací

Animace obecně v grafice vždy zaujmou. V případě grafiky HMI by nemělo být dovoleno, aby tyto animace způsobovaly jakékoliv rozptýlení uživatele. Také by měly být zobrazeny pouze nezbytné informace. Uživatel se totiž potřebuje v první řadě zaměřit na sledování stavu procesu.[14]

Doporučení bude určitě vhodné dodržet i v rámci návrhu grafiky potřebného pro tuto práci. Jelikož uživatel, v tomto případě řidič, ocení více zobrazení stavu baterie a upozornění na nutnost dobytí, místo animace, zda je automobil v pohybu. Tato informace by byla na displeji na úkor viditelnost ostatních a pro řidiče důležitějších prvků.

1.8 CAN Bus

CAN (Control Area Network) je robustní standard sběrnice používaný ve vozidlech. Zjednodušeně by se dalo CAN sběrnici přirovnat k nervovému systému automobilu, který umožňuje komunikaci mezi všemi jeho částmi. V případě automobilu jsou těmito částmi jednotlivá ECU (Electronic Control Unit). ECU je zabudovaný systém v automobilové elektronice, který řídí jeden či více elektrických systémů nebo subsystémů ve vozidle. Například airbagy, audiosystém nebo management baterie. Moderní vozidla mohou mít až 80 takovýchto ECU. Protokol sběrnice CAN pak umožňuje komunikaci navzájem mezi jednotlivými systémy. Odpadá tak potřeba realizace vyhrazených linek mezi každým z těchto systémů. Zároveň je díky propojení pomocí této sběrnice umožněno, aby jednotlivá ECU používala informace z ostatních, což eliminuje potřeby instalace senzorů stejného zaměření do více zařízení. Zde budou uvedeny hlavní výhody a důvody, proč je tento standard využíván. [15]

Nízká cena

ECU komunikují pomocí jediného rozhraní CAN. Díky čemuž je zredukována výsledná cena, hmotnost i chybovosti.

Centralizace

Systém sběrnice CAN umožňuje centrální diagnostiku chyb a konfiguraci všech ECU.

Robustnost

Takovýto systém je odolný vůči elektrickým poruchám a elektromagnetickému rušení, což ho činí pro prostředí automobilů ideálním.

Efektivní

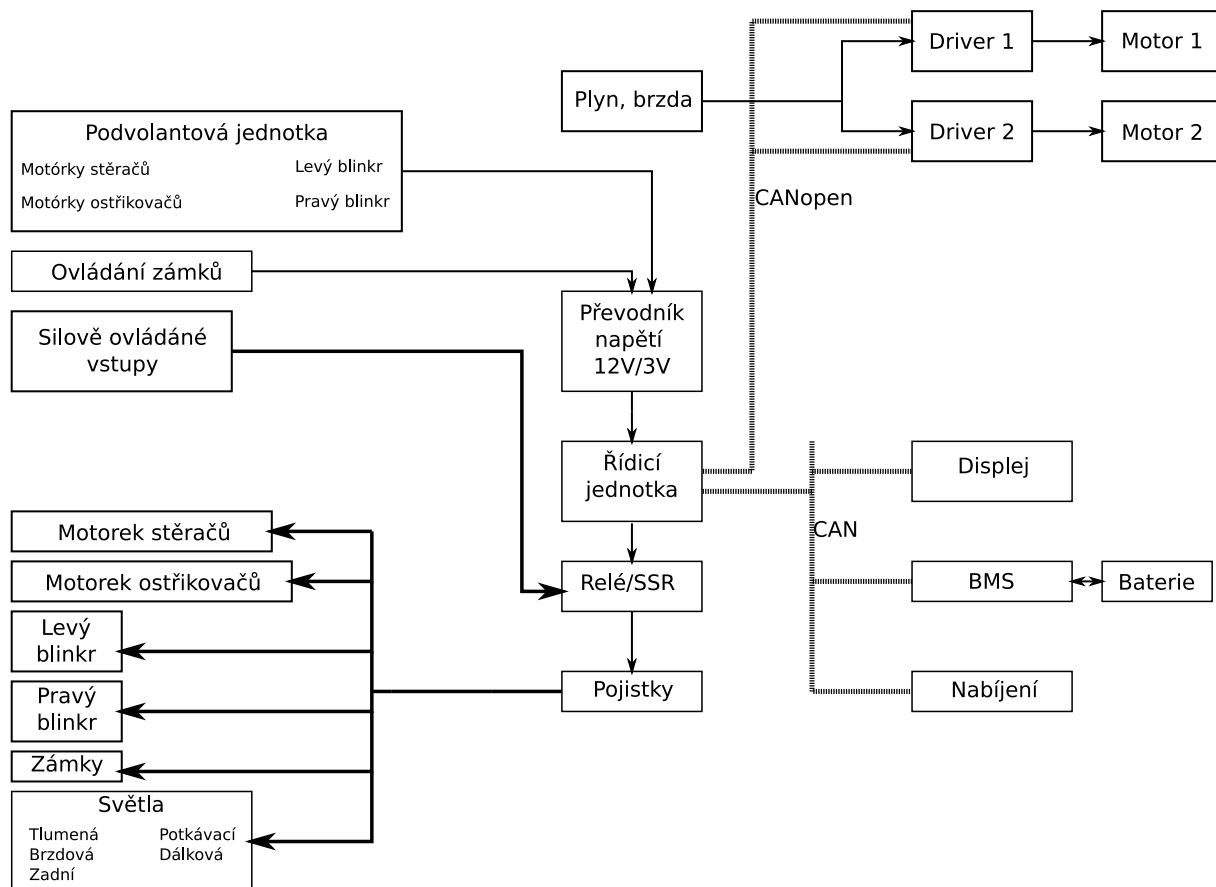
Priorita vysílaných CAN zpráv je rozlišena pomocí ID. Zprávy s nejvyšší prioritou získají přístup ke sběrnici přednostně.

Flexibilita

Každá jednotka ECU připojená k CAN sběrnici může přijímat všechny zprávy přenášené touto sběrnici. Pak se rozhoduje o relativnosti jednotlivých zpráv pro dané ECU a podle toho jsou data akceptována či ignorována.

1.9 Popis demo-boardu

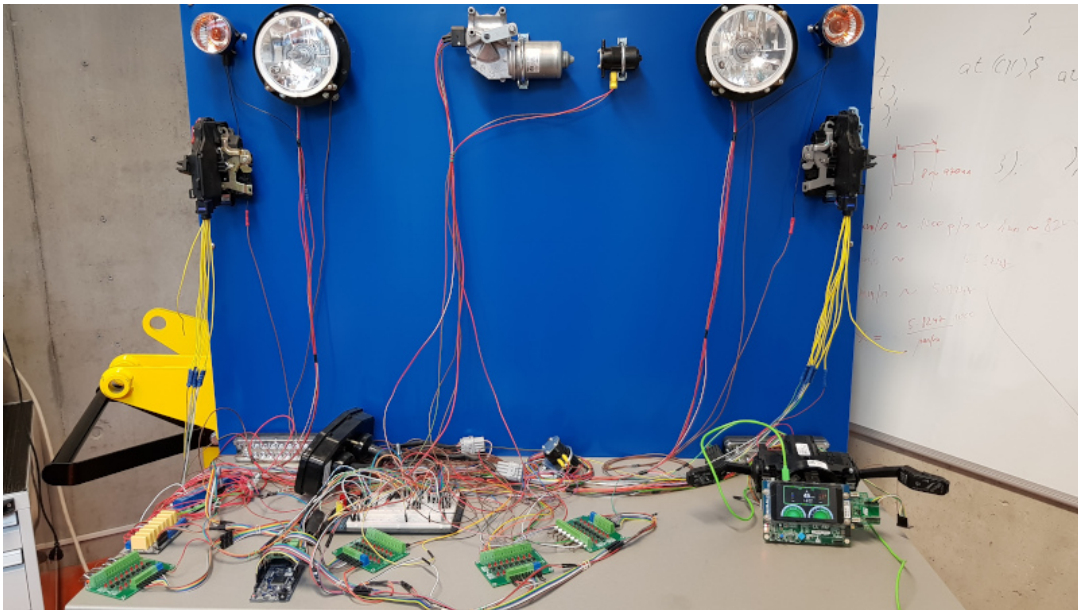
Pro případ této práce byl využit demo-board. Na obrázku 1.19 je vidět blokové schéma tohoto demo-boardu. V rámci této práce byla řešena funkcionality či komunikace pro následující části. Displeje (palubní desky), řídicí jednotky, ovládacích prvků podvolantové jednotky (páčky směrovek, páčky stěračů a ostřikovačů a ovládání mechanismu zámků).



Obrázek 1.19: Blokové schéma demo-boardu

1.9.1 Vstupy a výstupy

Mezi jeden druhu signálů, které bylo potřeba v tomto případě zpracovávat, patřily CAN zprávy. Tyto zprávy jsou posílány po sběrnici mezi řídicí jednotkou a displejem. Dalším druhem zpracovávaných signálů byly signály silové. Tyto signály sloužily například pro obsluhu světel pro denní svícení. Výstupy na tomto demo-boardu tvořily koncové prvky. Mezi tyto prvky patřila: světla směrovek, světla pro denní svícení, motorek stěračů, motorek ostřikovačů a zámky. Tyto výstupy byly ovládány za pomoci relé, případně „solid state relé“.



Obrázek 1.20: Demo-board

2 Realizace grafiky a firmwaru

2.1 Výběr a testování displejů

Mikroe HMI 7"

Po provedení počáteční rešerše displejů byl vybrán displej od společnosti Mikroelektronika. U tohoto displeje byla podle parametrů jediným možným problémem poměrně malá svítivost jeho panelu. Pro účel programování tohoto displeje byl k dispozici kompilátor „microC for FT90x“. Součástí tohoto kompilátoru bylo i prostředí pro vytváření grafického rozhraní. Prostředí jako celek se jevilo přehledně a přívětivě. Jako možný problém se zde ukázala jeho dokumentace. V dokumentaci knihovny s funkcemi potřebnými pro komunikaci pomocí CAN protokolu nebyl totiž u většiny z nich uveden konkrétní příklad jejich využití. Naštěstí zde bylo k dispozici několik příkladů od výrobce, které sloužily jako inspirace.

Nejdříve byla do displeje nahrána aplikace pro otestování. Toto proběhlo bez komplikací. Pro nahrávání zkompilovaného kódu zde sloužila aplikace USB HID Bootloader. Další částí bylo otestování komunikace po sběrnici CAN. V rámci seznámení se s displejem byla jako první otestována komunikace po sériovém portu. Toto testování proběhlo bez komplikací. Následně přišlo na řadu samotné otestování komunikace po sběrnici CAN, které ovšem dopadlo neúspěchem. Stále totiž nebylo možné v displeji přečíst přijaté CAN zprávy. Pro vysílání CAN zpráv bylo použito vývojové desky Arduino Due vybavené příslušným vysílačem. Zde bylo využito i podpory ze strany výrobce, ale ani s touto pomocí se CAN zprávy stále nedařilo zpracovat nebo i jen číst jejich jednotlivé části. Ohledně grafických možností byl tento displej prozkoumán jen zběžně. A ukázalo se, že pro požadovaný grafický design by bylo potřeba vytvoření či částečné upravení již dostupných widgetů. Dále testování tohoto displeje již neprobíhalo.

Weintek eMT3070B

HMI displej od společnosti Weintek byl dalším testovaným. Zde byla vcelku snadno vyzkoušena komunikace po sběrnici CAN, která na rozdíl od modelu uvedeného výše, fungovala bez problému. Vše ohledně nastavení a následného zobrazení dat z CAN zprávy, bylo velice intuitivní. V případě tohoto displeje tak stačilo provést jeho propojení s CAN vysílačem a ve vývojovém prostředí přidat nové lokální zařízení, které komunikuje po sběrnici CAN. Následně pak bylo možné, jednoduše číst požadovanou část CAN zprávy. Za zmínku dále stojí vývojové prostředí, které

je uživatelsky přívětivé a také obsahuje již spoustu předdefinovaných prvků. Tyto prvky dále nabízejí velkou variabilitu v možnostech jejich přizpůsobení. Výhodou tohoto displeje byla robustní konstrukce, jelikož displej byl umístěn v kovovém šasi. Jako nevýhoda se u tohoto displeje jeho poměrně vysoká cena, která by znesnadňovala jeho použití ve výsledném elektromobilu, a také jeho nízká svítivost.

4D systems GEN4-ULCD-50D-SB-AR

Jako další byl testován displej od společnosti 4D Systems. Tento displej nabízí pouze komunikaci pomocí sériové linky. Výrobce uvádí u tohoto displeje, že disponuje vysokou svítivostí, což by bylo pro případ této práce ideální. Bohužel se nepodařilo uvést do provozu ani jeden ze dvou kusů tohoto displeje, které byly k dispozici a tak nebylo možné ho nějakým způsobem dále otestovat. Displej po připojení napájení pouze problikával a tak byl hned ze začátku označen za nevhodný a byl zamítnut.

Riverdi RiTFT-50-IOT-CAP

Dalším z testovaných byl displej od společnosti Riverdi. Tento displej umožňoval vývoj pomocí různých vývojových prostředí. Zde bylo vyzkoušeno nahrání aplikace z vývojových prostředí Arduino IDE a také Zerynth Studio. V obou případech proběhlo vše bez komplikací. Tento displej vypadal jako přijatelná varianta pro realizaci výsledného řešení. Nedostatek tohoto displeje spočíval v nepřítomnosti rozhraní CAN a také v jeho nepřiliš vysoké svítivosti. Kombinace těchto dvou nedostatků vedla k hledání dalších variant.

STM32H747I-DISCO

Po zkušenostech z předchozích HMI displejů přišla na řadu tato varianta. V tomto případě se nejedná přímo o HMI displej, ale o vývojářský kit, který umožňuje připojení panelu. Jelikož tento kit disponuje rozhraním MIPI[®] DSI, bylo možné provést snadnou výměnu panelu na tomto kitu. Díky tomu bylo rozšířeno množství panelů, ze kterých bylo možné vybírat. Tento kit byl také nakonec vybrán jako varianta pro finální řešení s tím, že pro něj bude vybrán panel s dostatečnou svítivostí a pozorovacími úhly. Dalším důvodem proč byl tento kit nakonec vybrán, byla přítomnost CAN sběrnice.

Pro vývoj grafiky tohoto displeje bylo k dispozici hned několik vývojových prostředí. Jedna z variant, která byla k dispozici, bylo Embedded Wizard studio. Toto vývojové prostředí se z počátku jeví jako dobrá varianta, i když způsob vytváření aplikací zde byl mírně odlišný od většiny ostatních vývojových prostředí. Bylo zde ale k dispozici množství příkladů a ukázek. Jako další možnost pro vývoj grafiky byl nástroj TouchGFX Designer. Toto prostředí nabízelo klasický způsob vývoje, kde byla k dispozici část pro grafický návrh způsobem „drag and drop“ a k ní náležející zdrojový kód. Pro vytvoření uživatelského rozhraní bylo nakonec zvoleno druhé jmenované prostředí TouchGFX Designer.

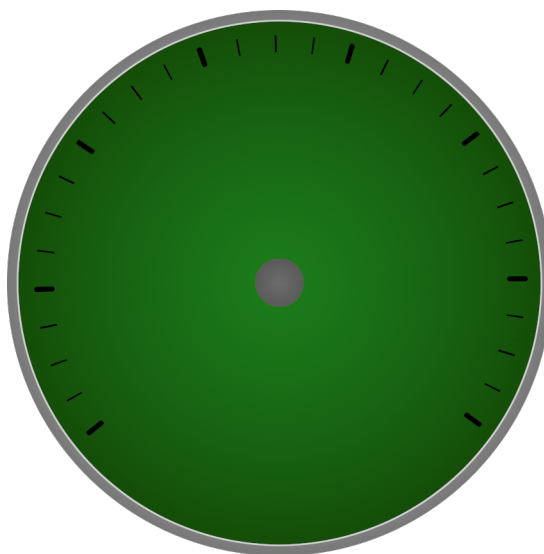
2.2 Návrh grafiky

Další částí této bakalářské práce bylo navržení grafického rozhraní pro vybraný displej. Už při původním seznamování se s displeji bylo jasné, že budou u většiny z nich možnosti předpřipravených widgetů nedostačující. Zejména pak z pohledu jejich grafické stránky. Většina z nich totiž obsahovala pouze widgety jednoduchého vzhledu. Pro vytváře podoby widgetů, která odpovídala požadavkům bylo využito aplikace GNU Image Manipulation Program, zkráceně GIMP.

Při vytváření grafického rozhraní bylo nejprve potřeba si ujasnit, co uživateli (řidiči) bude nutné zobrazovat. Jednalo se o aktuální rychlost vozidla, dojezd, průměrnou spotřebu, procenta zbývající baterie, aktuální využití motoru a pak různé signalizační kontrolky jako jsou směrovky, signalizace poruchy motoru či světel a další. Po ujasnění těchto požadavků bylo možné přistoupit k samotnému vytváření grafického rozhraní.

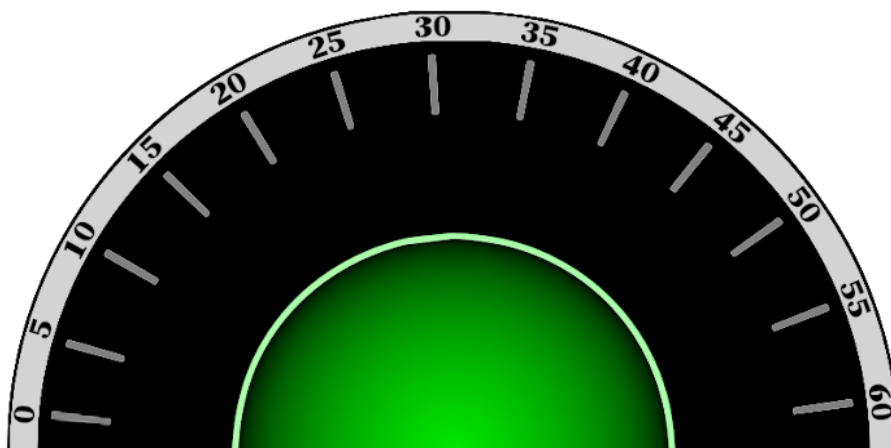
Návrh ukazatelů

Jako první z vyjmenovaných přišla na řadu realizace vizualizace využívaného výkonu a dojezdu. Zde bylo již od začátku uvažováno o vizualizaci ve formě, jakou je v automobilech reprezentován rychloměr a otáčkoměr. Prvním zhotoveným návrhem, byl tento kruhový analogový ukazatel na obrázku 2.1.



Obrázek 2.1: Návrh kruhového ukazatele

Dalším z vytvořených návrhů byl ukazatel půlkruhový. Ten na rozdíl od kruhové varianty poskytoval možnost využití i části prostoru uvnitř tohoto ukazatele. Na obrázku 2.2 se nachází jeho první návrh.



Obrázek 2.2: Návrh půlkruhového ukazatele

Následně bylo vhodné vybrat jeden z těchto dvou návrhů a touto cestou pak pokračovat. Jako varianta, která bude použita pro řešení a zároveň tak i pro další vývoj, byla zvolena varianta půlkruhového ukazatele. Na dalších obrázcích bude zobrazen postupný vývoj a úpravy tohoto ukazatele.

V dalším návrhu, který je zobrazen na obrázku 2.3, bylo provedeno drobné zmenšení celého ukazatele a zvětšení číslic. Tento ukazatel vypadal už o poznání lépe, jak varianta předchozí. Bohužel zde zvětšení číslic zlepšilo viditelnost ovšem na úkor přehlednosti. Ukazatel pak působil až moc přeplněným dojmem.



Obrázek 2.3: Vytváření půlkruhového ukazatele

Jako řešení tohoto nedostatku se nabízelo ponechání jen některých čísel. Po této úpravě vypadal ukazatel následovně 2.4. V tomto stavu již vypadá ukazatel o poznání přehledněji, než v předchozím případě. Tato varianta byla pak jistou dobu uvažována jako finální.



Obrázek 2.4: Vytváření půlkruhového ukazatele II

Následně byla provedena poslední změna v tomto ukazateli. Jelikož byl tento ukazatel uvažován pro vizualizaci dojezdové vzdálenosti automobilu, bylo potřeba provést drobné úpravy. Tyto úpravy byly provedeny zejména pro usnadnění čitelnosti a vylepšení přehlednosti tohoto ukazatele. Výsledná verze, která byla také nakonec použita pro finální realizaci je zobrazena na obrázku 2.5

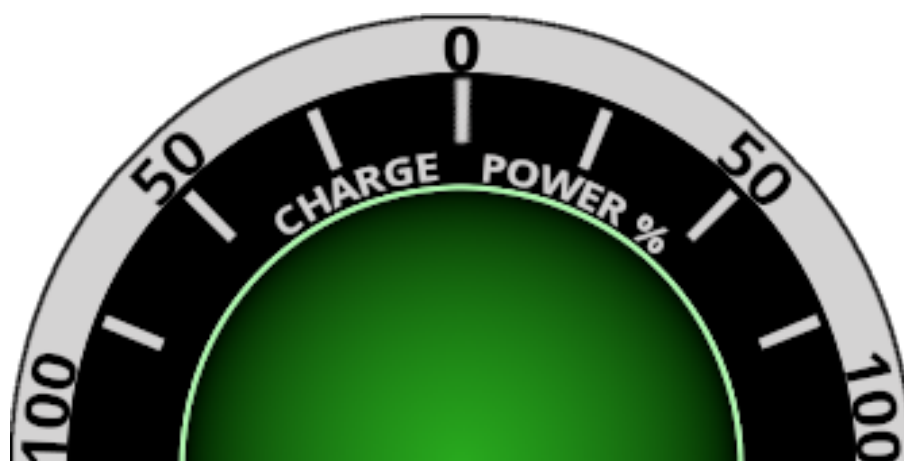


Obrázek 2.5: Vytváření půlkruhového ukazatele III

V tomto stavu vypadal tento ukazatel čitelně a přehledně, ale zároveň nepůsobil přeplněným dojmem. Do ukazatele byly ještě přidány popisky, aby bylo jasné, jaká data ukazatel zobrazuje. Vnitřní část ukazatele bylo využita pro zobrazení procent zbývajících baterie a vnější část, která byla vyplňována zeleně, sloužila pro zobrazení zbývajících dojezdu automobilu v kilometrech.

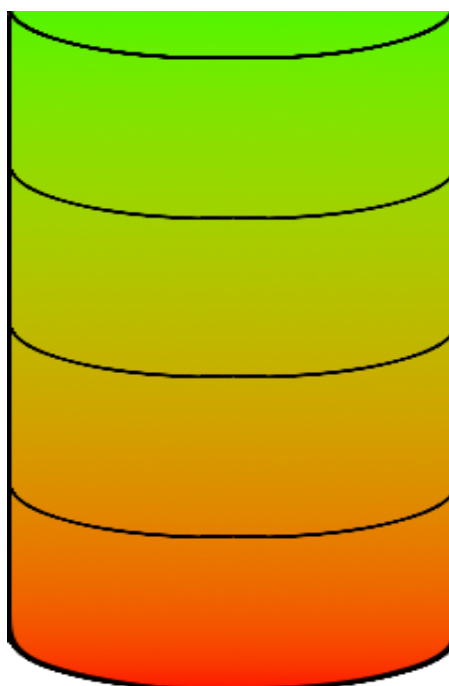
Tento návrh ukazatele posloužil i pro vytvoření další části grafiky displeje. Jednalo se o druhý ukazatel, který byl určen pro zobrazení aktuálního využití motoru a jeho druhá polovina byla určena pro zobrazení, jak se v době, kdy není využíván motor, dobíjí akumulátor. Například v době brzdění či jízdě z kopce. Tento ukazatel zobrazený na obrázku 2.6, byl až na jeho značení totožný s ukazatelem, který

zobrazoval dojezd automobilu.



Obrázek 2.6: Vytváření půlkruhového ukazatele IV

V průběhu vytváření bylo uvažováno o zobrazení stavu baterie v podobě ať už horizontální, či vertikálního baru. Jedním z možných uvažovaných zobrazení stavu akumulátoru byl například formát zobrazený na obrázku 2.7



Obrázek 2.7: Uvažované zobrazení stavu akumulátoru

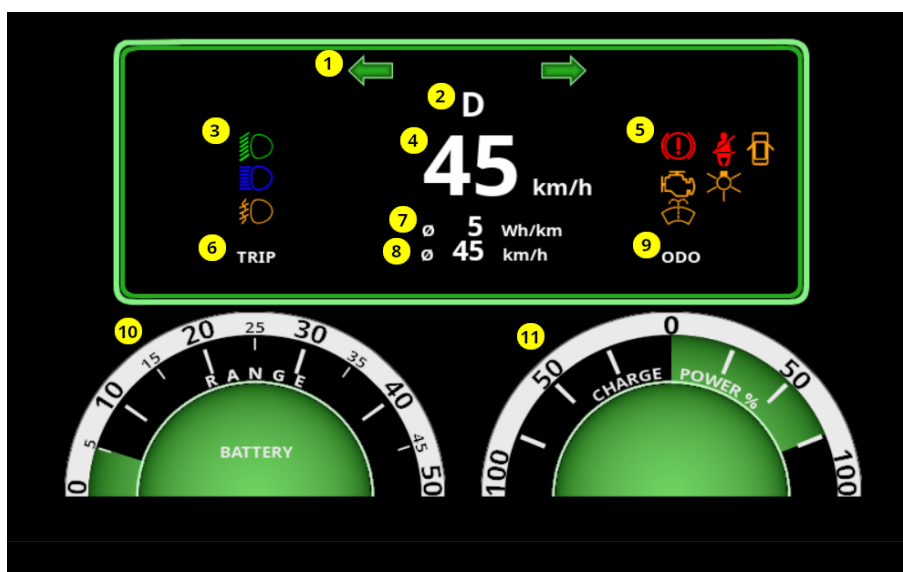
Nakonec bylo ovšem zvoleno zobrazení ve formě dvou stejně stylizovaných ukazatelů. Díky tomuto formátu bylo dosaženo stejnorodého vzhledu celé grafiky displeje, namísto více rozličně stylizovaných částí.

Kontrolky

U vytváření kontrolky nebyl přílišný prostor na kreativitu, jelikož jsou v tomto značení již zavedené nějaké standardy v oblasti automobilů a tak by bylo nežádoucí a nesmyslné provádět návrh jejich vlastního stylu. Jediné kontrolky, u kterých šlo použít alespoň z části vlastního návrhu, byly kontrolky směrovek.

Kompletní grafika displeje

Po vytvoření všech potřebných komponent, popsanych v předchozích bodech, mohlo přijít na řadu jejich spojení a realizace kompletní grafiky. Všechny komponenty spojené dohromady vypadaly pak následovně viz obr. 2.8



Obrázek 2.8: Finální návrh grafiky displeje

Zde bude následovat popis jednotlivých částí již finální verze grafiky displeje. Displej na obrázku je zobrazen ve stavu, kdy jsou všechny kontrolky rozsvícené.

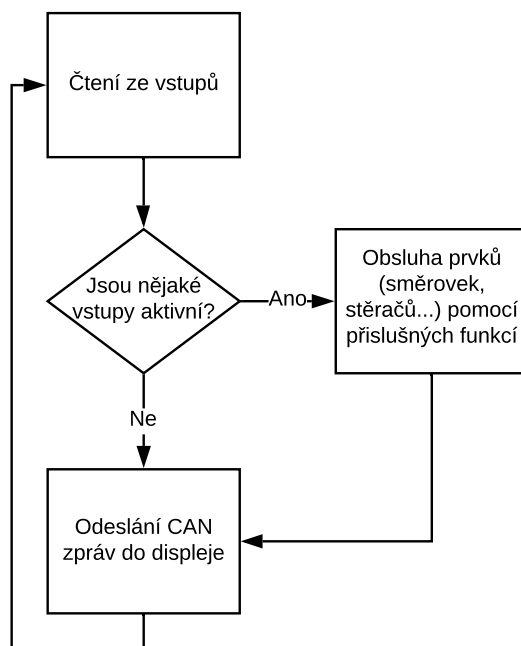
1. Zde jsou kontrolky pravé a levé směrovky
2. Zobrazení aktuálně zařazené rychlosti. Jelikož se jedná o displej do elektromobilu nejsou zde klasické rychlosti, ale pouze písmena D, R, P, N.
3. Kontrolky světel. První je kontrolka spuštěného denního svícení, druhá pak kontrolka světel dálkových a poslední je kontrolka mlhových světel.
4. Zde se zobrazuje aktuální rychlost vozidla.
5. Červené a žluté výstražné kontrolky. Mezi ně patří kontrolka zatažené ruční brzy, nezapnutých bezpečnostních pásů, poruchy motoru, světel či nedostatku kapaliny v ostřikovačích.
6. Zde je zobrazena krátkodobá ujetá vzdálenost.
7. Průměrná spotřeba elektřiny na kilometr.
8. Průměrná rychlost automobilu.
9. Zde je zobrazena celková ujetá vzdálenost u které řidič nemá možnost ji vynulovat.

10. Tento ukazatel bude zobrazovat dojezd v kilometrech v prostoru mezikruží. Uvnitř jsou pak zobrazena procenta zbývající baterie.

11. Tento ukazatel má dvě části. Levá část zobrazuje rekuperaci energie. Práva část pak zobrazuje využitý výkon motoru.

2.3 Firmware řídicí jednotky

Další částí bakalářské práce bylo vytvoření firmwaru, který se bude starat o řízení ovládacích prvků ve vozidle resp. na demo-boardu. Dále se bude tento firmware starat o komunikaci mezi ovládacími prvky a displejem. Mezi tyto prvky jmenovitě patří: směrovky, stěrače, zámky, ostřikovače. Po provedení počátečního průzkumu trhu, byla zvolena pro vývoj firmwaru vývojová deska Arduino Due. Tato deska je založená na architektuře ARM a je možné ji programovat pomocí programovacího jazyka C. Po úspěšném připojení této desky k demo-boardu bylo potřeba naprogramovat všechny vstupy a výstupy. Následně už bylo možné zahájit vývoj samotného firmwaru.



Obrázek 2.9: Zjednodušený vývojový digram pro řídicí jednotku

Na obrázku 2.9 je zobrazen digram řídicí jednotky. Jedná pouze o zjednodušenou verzi pro lepší představu, jak řídicí jednotka pracuje. Podrobnější popis jednotlivých funkcí, které slouží pro obsluhu ovládacích prvků ve vozidle, se nachází v kapitole 2.3.1.

2.3.1 Ovládací prvky

Zde následuje výčet jednotlivých prvků a stručný popis toho jaká funkcionalita k nim byla naprogramována a jak byla tato implementace řešena. Na obrázku je pak zobrazen vývojový diagram tohoto firmwaru.

Směrovky

Prvním z programovaných ovládacích prvků byly směrovky. Realizace ovládání směrovek přišla na řadu hned po prvotním seznámení s demo-boardem. Zde bylo základní požadavkem, aby byla část, která se bude starat o obsluhu směrovek vyřešena bez použití funkcí, které by způsobovaly zastavení celého procesoru. V tomto případě to znamenalo vyvarovat se použití řešení, která by byla založena na funkci „delay“. Tento požadavek se podařilo splnit za pomoci časovačů. Výsledná funkčnost směrovek po naprogramování pak zahrnovala jak klasické blikání, tak i funkci komfortního blikání.

Na demo-boardu bylo zapínání a vypínání světel směrovek řízeno pomocí „solid state relé“. V prostředí firmwaru pak vypadalo řešení veškeré logiky týkající se směrovek následovně. Nejdříve bylo potřeba vyřešit klasickou funkci směrovek. Tato funkce obnášela čtení ze správného vstupu a následné nastavení příslušné hodnoty na výstup. V případě realizace komfortního blikání musela být funkce starající se o ovládání směrovek, rozšířena a poupravena. Bylo zde provedeno přidání „counteru“ a následné kontroly, zda po projití požadovaného počtu iterací, je páčka ovládající směrovku stále ve stavu zapnuto. Díky tomuto „counteru“ bylo dosaženo požadovaného výsledku, kdy i po krátkém stisknutí páčky směrovek, byla zajištěna tři bliknutí.

Stěrače a ostříkovače

Stěrače byly na demo-boardu reprezentovány pouze jejich motorkem, což bylo ale pro potřeby programování dostačující. Pro stěrače byly naprogramovány klasické funkce, běžně známé z automobilů. Mezi tyto funkce patří rychlé stírání, pomalé stírání, intervalové stěrače (v tomto případě se 4 intervaly) a dále také funkce setření stěračů 1x a nakonec i spuštění ostříkovačů s následným setřením. Pro stěrače byly k dispozici vstupy, které signalizovaly spuštění daného režimu stěračů. Mimo těchto vstupů zde byl ještě vstup, který signalizoval nulovou pozici stěračů. Dále byl k dispozici výstup, který sloužil k zapínání a vypínání motoru stěračů.

Při realizaci funkcí starajících se o obsluhu stěračů bylo potřeba vyřešit vrácení stěračů do nulové pozice. Pokud by toto nebylo bráno v úvahu, pak by byly stěrače vypnuty ihned po nastavení páčky ovládání stěračů do defaultní pozice. Toto by ve výsledku znamenalo, že by bylo možné, aby se stěrače zastavily například i uprostřed čelního okna. Pro implementaci tohoto požadavku bylo zapotřebí využít vstup identifikující nulovou pozici stěračů. Ve výsledku bylo pak za pomoci tohoto vstupu zajištěno, že i po vypnutí stěračů pomocí páčky zůstal motorek, který se staral o ovládání stěračů, v běhu do doby, než byly stěrače zpět v nulové pozici.

Mezi další funkcionalitu, které byly implementovány, patřilo i intervalové stírání. Intervaly se přepínaly přepínačem, který umožňoval nastavení celkem čtyř intervalů. Na demo-boardu byl vstup z tohoto přepínače přiveden na analogový. Následně bylo provedeno rozdělení hodnot, které byly čteny z analogového vstupu, do čtyř příslušných intervalů. Hodnoty udávaly nominální hodnotu napětí na tomto vstupu.

V případě funkce, která se starala o spuštění ostříkovačů a následného setření, stačilo nastavit mírné zpoždění, než byly spuštěny stěrače. Toto bylo provedeno z důvodu, aby bylo zajištěno, že bude před případných setřením na okno nanesen čistící přípravek.

Zámky

Mezi další prvky, které byly k dispozici na demo-boardu, patřily elektronicky ovládané zámky. Pro zamčení, resp. odemčení, bylo potřeba nastavení dvou částí. První částí byl mechanismus samotného zámku, kde se provedlo jeho zamčení či odemčení. Druhou částí byl mechanismus, starající se o zajištění zámku po zamčení, případně odjštění před odemčením.

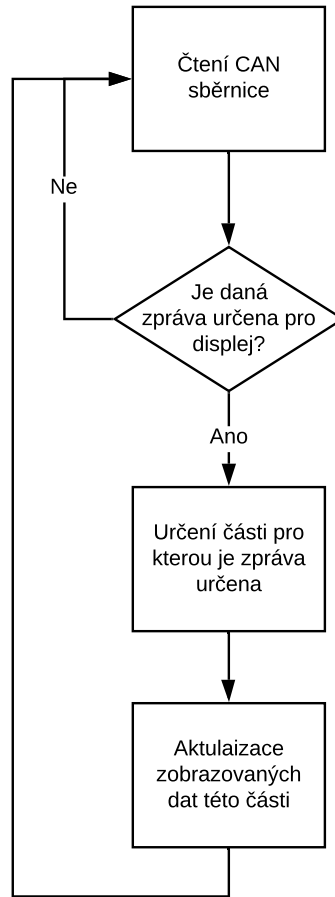
Pro ovládání zámků byla k dispozici na demo-boardu 3 relé, pomocí jejichž výstupů se zámky ovládaly. Samotné nastavení se pak provádělo nastavením příslušné kombinace na jejich výstupech. Kombinací se rozumí například nastavení relé 1 a 2 do stavu logické jedničky a relé 3 pak do stavu logické nuly. Po nastavení příslušné kombinace a vyčkání v řádu stovek milisekund, bylo zapotřebí všechna relé nastavit zpět do stavu logické nuly. A až následně mohl být nastaven další požadovaný stav.

2.3.2 Komunikace řídicí jednotky s displejem

Jak již bylo řečeno, starost o ovládání prvků ve vozidle nebyla jediná funkce programované řídicí jednotky. Jejím dalším úkolem bylo obstarání komunikace mezi touto jednotkou a displejem. Tato komunikace probíhala po sběrnici CAN. Hlavním úkolem této komunikace bylo předání informací o aktuálním stavu prvků na demo-boardu. Tento stav byl předávám pomocí CAN zpráv. Tyto zprávy byly pak v displeji zpracovány a podle jejich ID bylo rozhodnuto, ke které grafické části daná zpráva náleží. Následně pak mohl být aktualizován grafický výstup na displeji v závislosti na datech v dané CAN zprávě.

Identifikace CAN zpráv

Jelikož sběrnice CAN je určena pro vzájemnou komunikaci více jednotek v rámci vozidla, bylo vhodné odesílané CAN zprávy rozumně označit pro snadnější identifikaci. Jako nejjednodušší varianta se zde jevilo pomyslné rozdělení ID CAN zpráv. Jelikož CAN ID je nejběžněji reprezentováno 11 bity, bylo určeno logické rozdělení na dvě části. Jedna část se skládala z prvních tří bitů a druhá ze zbývajících 8. Díky tomuto rozdělení mohlo být v displeji snadno rozhodnuto, zda je tato zpráva určena pro něj či nikoli. V případě že nebyla daná zpráva pro displej určena tj. nezačínala bity 001 resp. 000 nebylo nutné číst její obsah. První tři bity 000 byly v tomto případě vyhrazeny pro případ broadcastové vysílání.



Obrázek 2.10: Zjednodušený vývojový digram pro displej

3 Závěr

V rámci této bakalářské práce byl vytvořen firmware pro řídicí jednotku v elektrickém automobilu. Tento firmware obstarává obsluhu prvků ve vozidle. Těmito prvky konkrétně jsou směrovky, stěrače, ostřikovače a zámky. Dalším úkolem tohoto firmwaru bylo obstarání komunikace mezi řídicí jednotkou a displejem. Tato komunikace probíhá prostřednictvím zpráv po sběrnici CAN. Celý vývoj tohoto firmwaru a testování probíhalo na demo-boardu.

Další částí této bakalářské práce byl výběr displeje a návrh jeho grafiky. Nejdříve bylo potřeba provést rešerši displejů dostupných na trhu. Po provedení této rešerše se ukázalo, že nebude snadné nalézt displej, který by splňoval úplně všechny požadavky. Pro testování pak bylo vybráno několik HMI displejů, které se mezi ostatními jeví jako nejlepší varianta. Ovšem i u těchto vybraných displejů se ukázaly jisté nedostatky. Jednalo se buď o nepřítomnost CAN sběrnice, vysokou pořizovací cenu nebo nízkou svítivost panelu. Po otestování těchto displejů byla zvolena varianta vývojářského kitu STM32H747I-DISCO vybaveného displejem. Displej byl k tomuto kitu připojen pomocí rozhraní MIPI[®] DSI. Díky přítomnosti tohoto rozhraní vzrostlo množství displejů, které by šlo pro tuto práci uplatnit. Pro finální řešení tak mohl být snadněji vybrán displej s velkou svítivostí a dobrými pozorovacími úhly. Jedinou podmínkou tohoto displeje pak bylo, aby podporoval MIPI[®] DSI rozhraní nebo rozhraní, které bude možné na MIPI[®] DSI převést. Jiné požadavky na displej již nebyly, jelikož požadované rozhraní CAN bylo poskytováno samotným kitem bez ohledu na připojený displej. Po dokončení obou těchto částí bylo provedeno propojení a otestování komunikace mezi řídicí jednotkou a displejem prostřednictvím sběrnice CAN.

Řešení vytvořené v rámci této práce nabízí možnost plné konfigurace podle vlastních požadavků. Další výhodou takto zpracovaného řešení je vzájemná nezávislost vybraných komponent. Díky tomu bylo umožněno snadné provedení výměny jednotlivých komponent. V případě takovéto změny bude zapotřebí provést konfiguraci a programování pouze v rámci nové komponenty.

Výhodou použití zmíněného kitu byla přítomnost výkonného mikroprocesoru, konkrétně se jedná STM32H747XIH6 Arm[®]-based microcontroller. To dává možnost při dalším vývoji, firmware naprogramovat přímo do tohoto kitu a odstranit nutnost použití platformy Arduino Due, která v tomto případě zastávala funkci řídicí jednotky.

Literatura

- [1] Mikromedia HMI 7 no touch [online]. [cit. 2020-02-23]. Dostupné z: <https://www.mikroe.com/mikromedia-hmi-70-no-touch>
- [2] Weintek eMT3070B [online]. [cit. 2020-02-23]. Dostupné z: https://www.weintek.com/globalw/Product/Product_spec.aspx?Series=eMT
- [3] Bafang 850C [online]. [cit. 2020-02-23]. Dostupné z: <https://www.dhgate.com/product/bafang-850c-full-color-lcd-8fun-dp-c14-lcd/410123561.html>
- [4] AMETEK VIS' C-COM 7GT [online]. [cit. 2020-02-23]. Dostupné z: <https://www.ametekvis.com/-/media/ametekvis/download/new%20downloads/170047-c-com-7g.pdf>
- [5] Wecon PI 7 inch HMI : PI8070 [online]. [cit. 2020-02-23]. Dostupné z: http://we-con.com.cn/en/productsinfo_3035.html
- [6] Opus A6 ECO BASIC [online]. [cit. 2020-02-23]. Dostupné z: https://www.motronica.com/wp-content/uploads/2019/08/TDS_A6e_BASIC.pdf
- [7] RiTFT-50-IOT-CAP [online]. [cit. 2020-02-23]. Dostupné z: <https://riverdi.com/product/ritft50iotcap/>
- [8] 4D systems GEN4-ULCD-50D-SB-AR [online]. [cit. 2020-02-23]. Dostupné z: <https://4dsystems.com.au/gen4-ulcd-50d-sb-ar>
- [9] Grayhill 3D70 [online]. [cit. 2020-02-23]. Dostupné z: <http://www.grayhill.com/assets/1/7/3D70-datasheet.pdf>
- [10] MY-EVC5100S-HMI [online]. [cit. 2020-02-23]. Dostupné z: <http://www.myirtech.com/list.asp?id=571>
- [11] What is Qt for Automotive? [online]. [cit. 2020-03-02]. Dostupné z: <https://www.qt.io/qt-in-automotive/#why>
- [12] STM32H747I-DISCO [online]. [cit. 2020-04-14]. Dostupné z: <https://www.st.com/en/evaluation-tools/stm32h747i-disco.html>

- [13] Best practices control system hmi graphics development [online]. [cit. 2020-05-14]. Dostupné z: <https://www.crossco.com/blog/best-practices-control-system-hmi-graphic-development/>
- [14] 5 tips for HMI development [online]. [cit. 2020-05-14]. Dostupné z: <https://huffmaneng.com/5-tips-hmi-development/>
- [15] Simple intro can bus [online]. [cit. 2020-05-14]. Dostupné z: <https://www.csselectronics.com/screen/page/simple-intro-to-can-bus/language/en>
- [16] Digital LCD Display for Multi-function EV/LEV Speedometer [online]. [cit. 2020-05-17]. Dostupné z: <http://www.acewell-meter.com/portfolio/ace-5000ec-canbus-sereis-speedometer-for-lev-digital-lcd-display/>
- [17] CT-22 48v-144v universal digital programmable Electric electronic Motorcycle Speedometer[online]. [cit. 2020-05-17]. Dostupné z: <https://www.aliexpress.com/item/4000018301688.html?spm=2114.12010612.8148356.2.50101848qoihJ6>
- [18] QSMOTOR Electric Car Speedometer 48V-96V Rated Programmable Automobile Instrument [online]. [cit. 2020-05-17]. Dostupné z: <https://www.amazon.com/dp/B07D11TSFZ?tag=duckduckgo-ffab-20&linkCode=ogi&th=1&psc=1>
- [19] Can bus LCD display 3.3V [online]. [cit. 2020-05-17]. Dostupné z: https://www.alibaba.com/product-detail/can-bus-lcd-display-3-3v_1166035352.html?spm=a2700.details.maylikeexp.13.5d0d58fa61NIRb
- [20] Nextion NX8048P070-011C-Y [online]. [cit. 2020-05-17]. Dostupné z: <https://nextion.tech/datasheets/nx8048p070-011c-y/>
- [21] Visual TFT [online]. [cit. 2020-05-17]. Dostupné z: <https://www.mikroe.com/visual-tft>
- [22] Datasheet Weintek eMT3070B [online]. [cit. 2020-05-17]. Dostupné z: https://dl.weintek.com/public/eMT3000/eng/Datasheet/eMT3070B1_Datasheet_ENG.pdf
- [23] Datasheet MY-EVC5100S-HMI Display Panel [online]. [cit. 2020-05-17]. Dostupné z: <http://www.myirtech.com/download/HMI/MY-EVC5100S-HMI.pdf>
- [24] Datasheet AMETEK VIS' C-COM 7GT [online]. [cit. 2020-05-17]. Dostupné z: <https://www.ametekvis.com/-/media/ametekvis/download/new%20downloads/170047-c-com-7g.pdf>

- [25] Datasheet Opus A6 ECO BASIC (G2) [online]. [cit. 2020-05-17].
Dostupné z: https://www.motronica.com/wp-content/uploads/2019/08/TDS_A6e_BASIC.pdf
- [26] Datasheet RiTFT-50-IOT-CAP [online]. [cit. 2020-05-17]. Dostupné z:
<https://riverdi.com/docs/datasheets/iot/iot-5/>
- [27] Datasheet 4D systems GEN4-ULCD-50D-SB-AR [online]. [cit. 2020-05-17]. Dostupné z:
<https://4dsystems.com.au/mwdownloads/download/link/id/109/>
- [28] Datasheet Grayhill 3D70 [online]. [cit. 2020-05-17]. Dostupné z:
<http://www.grayhill.com/assets/1/7/3D70-datasheet.pdf>
- [29] Datasheet NX8048P070-011C-Y [online]. [cit. 2020-05-17]. Dostupné z:
<https://nextion.tech/datasheets/nx8048p070-011c-y/>
- [30] Datasheet STM32H747I-DISCO [online]. [cit. 2020-05-17]. Dostupné z:
https://www.st.com/resource/en/data_brief/stm32h747i-disco.pdf
- [31] What is HMI? [online]. [cit. 2020-05-17]. Dostupné z:
<https://inductiveautomation.com/resources/article/what-is-hmi>
- [32] Winstar WF70A8SYAHLNN0 [online]. [cit. 2020-05-17]. Dostupné z:
<https://www.unisystem-displays.com/en/products/lcd-tft/standard/wf70a8syahlenn0.html>
- [33] Arduino Due [online]. [cit. 2020-05-17]. Dostupné z:
<https://store.arduino.cc/arduino-due>