

**ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ**

Katedra aplikované elektroniky a telekomunikací

DIPLOMOVÁ PRÁCE

**Ovládací jednotka pro vzdálené řízení audio systému
automobilu**

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta elektrotechnická

Akademický rok: 2019/2020

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Jakub JÁRA**
Osobní číslo: **E18N0024P**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Elektronika a aplikovaná informatika**
Téma práce: **Ovládací jednotka pro vzdálené řízení audio systému automobilu**
Zadávající katedra: **Katedra aplikované elektroniky a telekomunikací**

Zásady pro vypracování

Navrhněte koncepci distribuované ovládací jednotky pro ozvučovací systém v automobilu. Ovládací jednotka bude disponovat uživatelským rozhraním s dotykovým displejem a vhodným komunikačním rozhraním.

1. Seznamte se s funkcemi ovládací jednotky, zvolte vhodné komunikační rozhraní a specifikujte potřebné hardwarové signály a základní strukturu uživatelského ovládacího rozhraní s využitím dotykového displeje.
2. Na základě předchozí analýzy navrhněte blokovou strukturu zařízení a specifikujte vhodný řídicí mikrokontrolér a dotykový displej.
3. Proveďte návrh obvodového řešení zařízení. Zařízení realizujte a oživte.
4. Navrhněte a realizujte knihovnu funkcí pro zobrazení vhodných ovládacích prvků s volbou možnosti vysílání obecně definovaných zpráv pomocí zvoleného komunikačního rozhraní. Do použitého mikrokontroléru implementujte odpovídající firmware.
5. Diskutujte možnost uživatelské konfigurace a propojení zařízení s nadřazeným systémem.

Rozsah diplomové práce: **40 – 60 stran**
Rozsah grafických prací: **podle doporučení vedoucího**
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. www.symetrix.co/products/prism-0-0-prism-4-4-prism-8-8-prism-12-12-prism-16-16/
2. www.nxp.com
3. www.st.com/mcu
4. www.ti.com
5. www.microchip.com

Vedoucí diplomové práce: **Ing. Petr Weissar, Ph.D.**
Katedra aplikované elektroniky a telekomunikací

Datum zadání diplomové práce: **4. října 2019**
Termín odevzdání diplomové práce: **28. května 2020**


Prof. Ing. Zdeněk Peroutka, Ph.D.
děkan




Doc. Dr. Ing. Vjačeslav Georgiev
vedoucí katedry

V Plzni dne 4. října 2019

Abstrakt

Diplomová práce je zaměřena na vytvoření ovládací jednotky pro vzdálené řízení audio systému automobilu. Cílem je zhodnocení možností a vytvoření funkčního prototypu ovládací jednotky, která bude disponovat uživatelským rozhraním s dotykovým displejem a vhodným komunikačním rozhraním. Toho je docíleno pomocí vývojové desky STM32F746G-DISCOVERY. Jako komunikační rozhraní, které je přizpůsobeno ovládanému DSP Prism 8x8 od společnosti Symterix, je použit Ethernet a standard RS-485, pro který je navržena DPS s budičem a zdrojem pro celou ovládací jednotku.

Klíčová slova

Ovládací jednotka, dotykový displej, komunikační rozhraní, mikrokontrolér, STM32, grafické rozhraní, Ethernet, RS-485

Abstract

The diploma thesis is focused on the creation of a control unit for remote control of a car audio system. The aim is to evaluate the possibilities and create a functional prototype of the control unit, which will have a user interface with a touch screen and a suitable communication interface. This is achieved by using the STM32F746G-DISCOVERY development board. Ethernet and the RS-485 standard are used as a communication interface, which is adapted to the controlled Prism 8x8 DSP from Symterix. The PCB with a driver RS-485 and a power source for the entire control unit is designed.

Key words

Control unit, touch screen, communication interface, microcontroller, STM32, graphic interface, Ethernet, RS-485

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této diplomové práce, je legální.

.....
podpis

V Plzni dne 16.6.2020

Bc. Jakub Jára

Poděkování

Tímto bych rád poděkoval vedoucímu diplomové práce Ing. Petru Weissarovi, Ph.D. za cenné profesionální rady, připomínky a metodické vedení práce.

Obsah

OBSAH	8
SEZNAM OBRÁZKŮ	9
ÚVOD	10
SEZNAM SYMBOLŮ A ZKRATEK	11
1 OVLÁDACÍ JEDNOTKA	12
1.1 UŽIVATELSKÉ ROZHRAŇÍ.....	12
1.2 OVLÁDACÍ PRVKY.....	13
1.3 KOMUNIKAČNÍ ROZHRAŇÍ.....	15
1.4 REÁLNÉ APLIKACE.....	16
2 ANALÝZA MOŽNOSTÍ ŘEŠENÍ OVLÁDACÍ JEDNOTKY	18
2.1 POŽADOVANÉ VLASTNOSTI OVLÁDACÍ JEDNOTKY.....	18
2.2 VOLBA MIKROKONTROLÉRU.....	18
2.3 GRAFICKÉ ROZHRAŇÍ.....	19
2.3.1 <i>LittlevGL</i>	20
2.3.2 <i>Embedded wizard</i>	20
2.3.3 <i>TouchGFX</i>	21
2.4 KOMUNIKAČNÍ ROZHRAŇÍ.....	22
2.4.1 <i>Standard RS-485</i>	22
2.4.2 <i>Ethernet</i>	23
2.5 SD KARTA.....	25
3 REALIZACE	26
3.1 OVLÁDACÍ JEDNOTKA A JEJÍ ALGORITMUS.....	26
3.2 GRAFICKÉ ROZHRAŇÍ.....	30
3.2.1 <i>Algoritmus generování GUI</i>	31
3.3 KONFIGURAČNÍ SOUBORY.....	32
3.3.1 <i>Konfigurační soubor pro GUI a jeho čtení</i>	33
3.3.2 <i>Konfigurační soubor pro Ethernet</i>	38
3.4 VOLBA KOMUNIKAČNÍHO ROZHRAŇÍ.....	39
3.4.1 <i>Komunikační protokol</i>	40
3.5 NÁVRH DPS.....	42
3.6 UŽIVATELSKÁ KONFIGURACE A PROPOJENÍ S NADŘAZENÝM SYSTÉMEM.....	45
ZÁVĚR	47
SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ	49
PŘÍLOHY	1

Seznam Obrázků

Obrázek 1 Blokové schéma	12
Obrázek 2 Ukázka GUI pro audio systém [2]	13
Obrázek 3 Způsoby ovládání	14
Obrázek 4 BMW idrive ovladač [3].....	14
Obrázek 5 BMW řada 8 [4].....	16
Obrázek 6 Škoda Kodíaq [5]	16
Obrázek 7 Tesla mobilní aplikace [6].....	17
Obrázek 8 Tesla aplikace pro chytré hodinky [7].....	17
Obrázek 9 Vývojová deska STM32F746G-DISCOVERY [8]	19
Obrázek 10 Ukázka grafického rozhraní LittlevGL [9]	20
Obrázek 11 Ukázka grafického rozhraní Embedded Wizard u Porsche Panamera [10]	21
Obrázek 12 Ukázka grafického rozhraní TouchGFX [11]	21
Obrázek 13 Rámec standardu RS-485.....	23
Obrázek 14 Rámec TCP/IP [13].....	24
Obrázek 15 UDP rámec [13]	24
Obrázek 16 MicroSD karta pinout [14].....	25
Obrázek 17 Bloková struktura.....	26
Obrázek 18 Vývojový diagram ovládací jednotky	28
Obrázek 19 Obsluha přerušení ovládacího prvku	29
Obrázek 20 Ukázka LittlevGL kódu	30
Obrázek 21 Algoritmus generování GUI	31
Obrázek 22 Detekce vložení MicroSD karty.....	32
Obrázek 23 Nastavení názvů GUI menu	33
Obrázek 24 Vývojový diagram čtení dat z konfiguračního souboru	36
Obrázek 25 Příklad konfiguračního souboru pro GUI	37
Obrázek 26 GUI generované podle konfiguračního souboru	37
Obrázek 27 Ukázka konfiguračního souboru pro Ethernet	38
Obrázek 28 Volba komunikačního rozhraní.....	39
Obrázek 29 Detekce nepřipojeného kabelu	40
Obrázek 30 Komunikační protokol „ping-pong“	41
Obrázek 31 Schéma zapojení DPS	42
Obrázek 32 DPS pro ovládací jednotku	43
Obrázek 33 Rozmístění součástek na DPS	44
Obrázek 34 Ukázka jednoduchého presetu v programu Composer	46

Úvod

Tato diplomová práce je zaměřena na vytvoření ovládací jednotky pro vzdálené řízení audio systému automobilu.

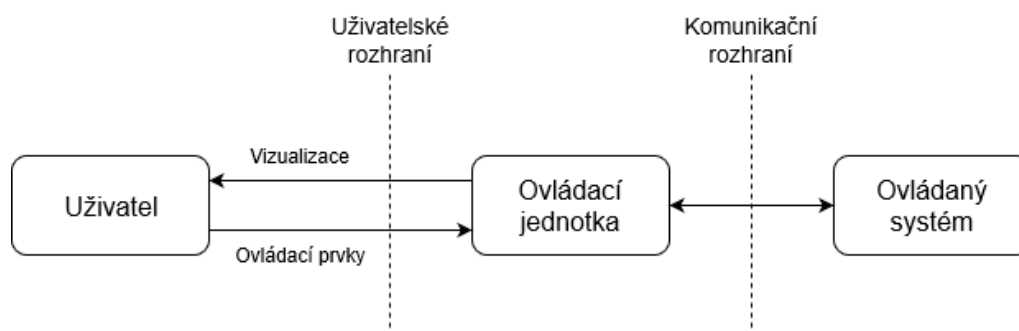
Obsah je rozdělen do tří hlavních částí: první se zabývá stručnou analýzou ovládací jednotky jako takové a ukázkou reálných aplikací, druhá uvádí analýzu možností řešení dle zadaných požadavků na ovládací jednotku pro DSP Prism 8x8 od společnosti Symterix. Třetí část popisuje konečnou realizaci ovládací jednotky pomocí vývojové desky STM32F746G-DISCOVERY s dotykovým displejem. Jako komunikační rozhraní, přizpůsobené DSP Prism 8x8, je použit Ethernet a standard RS-485. Dále je řešen hardware pro RS-485 a napájecí zdroj pro ovládací jednotku. V závěru jsou zhodnoceny dosažené výsledky práce.

Seznam symbolů a zkratek

STM32F746G-DISCOVERY.....	Vývojová deska od společnosti STMicroelectronics
GUI.....	Grafické uživatelské rozhraní
RS-485.....	Standard sériové komunikace
Rx.....	Signál přijatých dat
Tx.....	Signál odeslaných dat
DE.....	Signál řízení směru komunikace
TCP/IP.....	Komunikační protokol s garancí doručení dat
UDP.....	Komunikační protokol bez garance doručení dat
SD.....	Secure Digital – typ paměťové karty
DSP.....	Digitální signálový procesor
DPS.....	Deska plošných spojů
GND.....	Uzemnění
DC.....	Stejnoseměrné napětí

1 Ovládací jednotka

Ovládací jednotka je prostředník, který zprostředkovává ovládaní nad ovládaným systémem se zpětnou vizualizací. Tyto jednotky pro řízení audia v automobilech bývají většinou součástí většího multimediálního celku typu palubní počítač. Měly by disponovat určitým uživatelským komfortem, ať už je řeč o jednoduchosti ovládacího uživatelského rozhraní nebo o typu ovládaní samotném. Je vhodné zvolit snadno ovladatelné a intuitivní ovládací prvky, které ovšem záleží na zvoleném způsobu ovládaní jednotky. Způsoby ovládaní jde rozdělit zejména na tři kategorie, a to tlačítkové, dotykové a hlasové. Proto je vhodné zvolit způsob, který je pro danou specifickou aplikaci nejvhodnější. Dalším důležitým faktorem je vhodné zvolené komunikační rozhraní, které opět záleží na dané specifické aplikaci, má-li být například jednotka přenosná nebo pevně uložena. Nad těmito základními parametry je dobré se zamyslet před samotnou realizací.

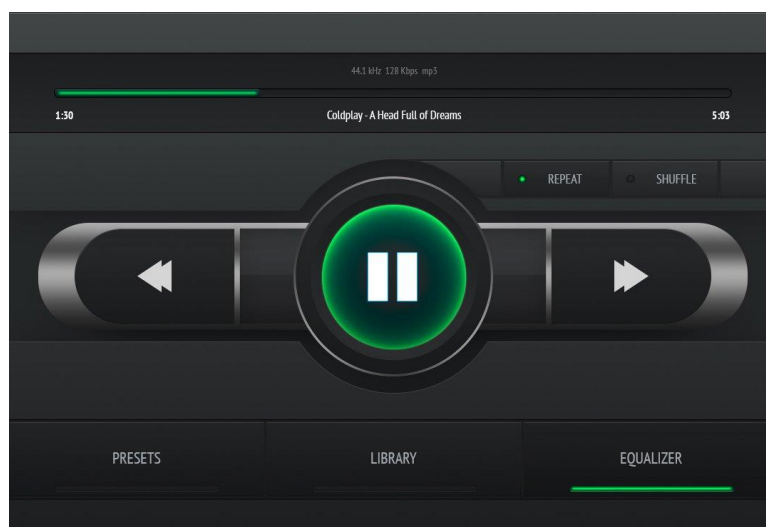


Obrázek 1 Blokové schéma

1.1 Uživatelské rozhraní

Uživatelské rozhraní UI (z angličtiny User Interface) je jedním z prvků informační a komunikační technologie. Jedná se o komunikační kanál mezi uživatelem a ovládaným systémem. Používá se tam, kde je požadována interakce mezi člověkem a strojem. Interakcí se zde rozumí zadávání požadavků člověkem, které poté stroj vyhodnocuje a prezentuje uživateli zpět přes uživatelské rozhraní.

Při zadávání požadavků stroji je preferován intuitivní způsob ovládání, který je více zaměřen na koncového uživatele. Je pro uživatele přívětivější a lépe se s ním pracuje. Komunikace s uživatelem probíhá na základě přímé manipulace s objekty. Jedním z nejpoužívanějších uživatelských intuitivních rozhraní je takové rozhraní, které lze ovládat pomocí hardwarových prvků (tlačítek) případně dotykem pomocí grafických prvků. Takovéto rozhraní se nazývá GUI (z angličtiny Graphic User Interface) neboli grafické uživatelské rozhraní. [1]



Obrázek 2 Ukázka GUI pro audio systém [2]

GUI může být tvořeno jednou obrazovkou nebo může obsahovat několik menu, mezi kterými lze navzájem přecházet, případně mohou být s různými stupni vnoření. Ukazovat by vždy mělo aktuálně ovládaný parametr a prvky sloužící k jeho ovládání.

1.2 Ovládací prvky

Ovládací prvek je zařízení, pomocí kterého může uživatel ovládat stroj. Volba ovládacích prvků je závislá na zvoleném způsobu ovládání, je-li tlačítkové, dotykové, hlasové nebo popřípadě hybridní. Je zde kladen důraz na jednoduchost, aby i běžný uživatel byl schopen dané zařízení snadno ovládat. Většinou jsou doprovázeny i vizualizací například pomocí displeje, který zobrazuje aktuálně vykonávanou činnost a hodnoty ovládaných parametrů.



Obrázek 3 Způsoby ovládání

- ***Tlačítkové prvky***

K těmto ovládacím prvkům se řadí klasická tlačítka, dvoustavové přepínače, ovládací kolečka nebo například joysticky. Existují speciální řídicí spínače, které kombinují více těchto prvků najednou.



Obrázek 4 BMW iDrive ovladač [3]

- ***Dotykové prvky***

Mezi tyto prvky, které jsou zobrazeny na dotykových displejích patří nejčastěji klasická tlačítka, posuvníky, zaškrtačací pole nebo přepínací tlačítka.

- ***Ovládání hlasem***

Jednotka je ovládána hlasovými povely, kdy příkazy většinou začínají klíčovým slovem, za kterým následuje daný příkaz.

- **Hybridní ovládání**

Je kombinace více zmíněných způsobů ovládání, může zahrnovat jak prvky tlačítkové, tak dotykové případy i ovládání hlasem.

1.3 Komunikační rozhraní

Komunikační rozhraní slouží ke komunikaci mezi ovládací jednotkou a ovládaným systémem. Zajišťuje správnou komunikaci a přenos dat mezi těmito prvky. Rozhraní můžeme obecně rozdělit na kabelové (zařízení jsou propojeny kabelem) a bezdrátové (komunikace probíhá vzduchem).

Při volbě komunikačního rozhraní pro ovládací jednotku je potřeba zohlednit její umístění. Jednotka může být přenosná nebo pevně uložená, případně může být přenosná jen její část. Podle těchto požadavků je pak nutné zvolit dané komunikační rozhraní. Zde je výčet několika používaných rozhraní:

- *CAN*
- *Ethernet*
- *FlexRay*
- *RS-485*
- *RS-232*
- *Bluetooth*
- *WLAN*

Dalším důležitým faktorem při volbě komunikačního rozhraní je pak zohlednění zabezpečení přenosu dat. Příkladem může být i zvolený komunikační protokol, jako je u Ethernetu volba mezi TCP/IP (garantuje spolehlivé doručení dat a ve správném pořadí), popřípadě UDP (bez záruky doručení dat) protokolem.

1.4 Reálné aplikace

Většina automobilek si vyvíjejí vlastní multimediální systémy, případně si je nechávají vyvinout na míru subdodavatelem. Součástí těchto multimediálních systémů je i ovládací jednotka pro řízení audia. Standardem bývá dotykový displej, který může být doplněn o několik málo tlačítek, což se týká pevně uložených systémů. Některé vyšší třídy automobilů mívají i hlasové ovládání. Pro příklad je zde zobrazen multimediální systém vozu BMW řady 8 a Škoda Kodiaq.

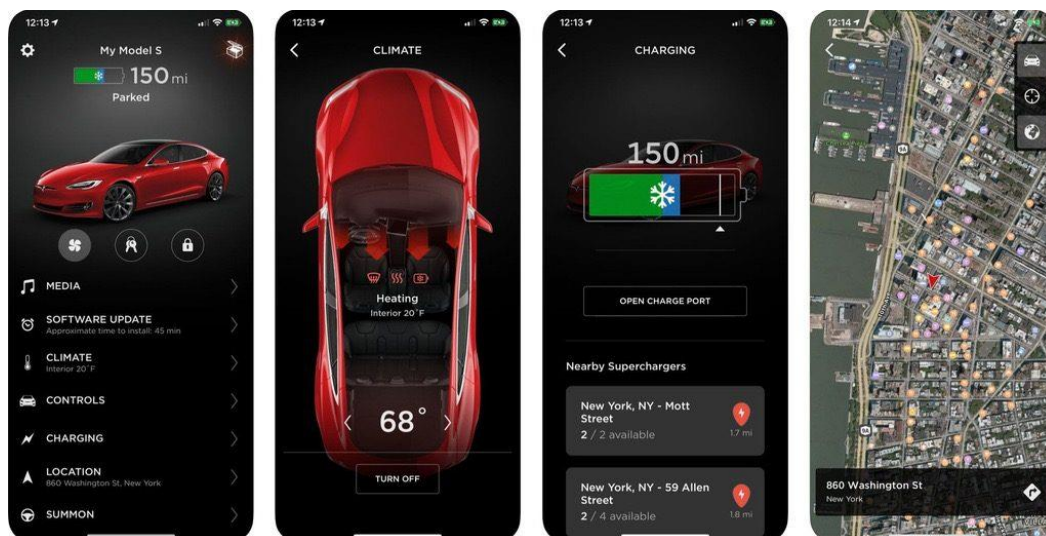


Obrázek 5 BMW řada 8 [4]



Obrázek 6 Škoda Kodiaq [5]

Určité vozy mají možnost propojení s mobilním telefonem a pomocí mobilní aplikace ovládat systém z povzdálí. Tuto možnost nabízejí například vozy automobilky Tesla, které nabízejí i ovládání pomocí chytrých hodinek.



Obrázek 7 Tesla mobilní aplikace [6]



Obrázek 8 Tesla aplikace pro chytré hodinky [7]

2 Analýza možností řešení ovládací jednotky

2.1 Požadované vlastnosti ovládací jednotky

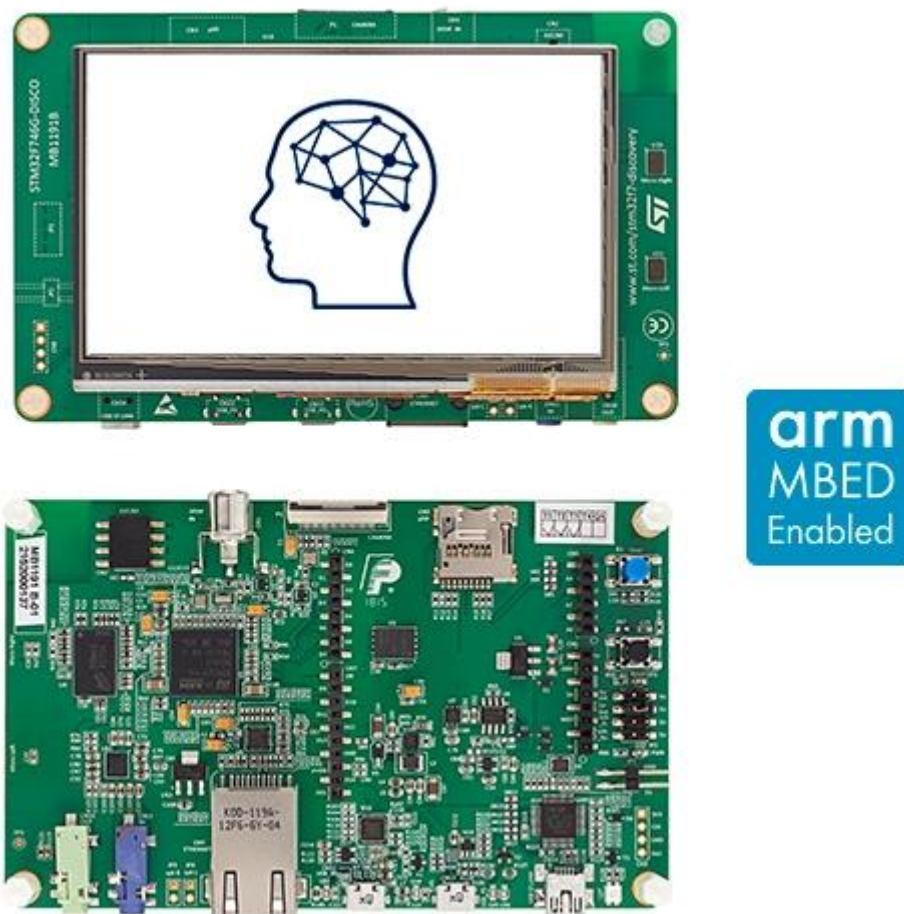
Byly zadány následující požadavky na ovládací jednotku pro audio systém v automobilu, v rámci kterých je audio systém zastoupen DSP Prism 8x8 od společnosti Symterix. Jednotka by měla obsahovat dotykový displej. Komunikace s DSP je možná pomocí RS-485 nebo přes Ethernet. Grafické menu by mělo být konfigurovatelné z vnějšku uživatelem, bez znalostí programování. Jeho konfigurace by měla být možná pomocí jednoduchého souboru, který bude uložen na externím médiu, kde si ho uživatel bude schopen rychle přenastavit, například pomocí PC. Grafické menu by mělo obsahovat jednoduché ovládací prvky typu tlačítko, posuvník a přepínač a měla by být možnost přepínání mezi několika menu. Těmto požadavkům byla podřízena realizace ovládací jednotky.

V následujících kapitolách je popsán výběr mikrokontroléru k realizaci ovládací jednotky, a k němu možnosti řešení pro výše zmíněné požadavky.

2.2 Volba mikrokontroléru

Podle výše zmíněných požadavků byl výběr zúžen na dva adepty, a to vývojovou desku od společnosti STMicroelectronics s názvem STM32F746G-DISCOVERY, nebo jednočipový počítač Raspberry Pi. Nakonec byl zvolen první zmíněný.

Vývojová deska STM32F746G-DISCOVERY má hned několik výhod, zejména již disponuje 4.3" RGB 480×272 LCD-TFT barevným kapacitním dotykovým displejem, dále obsahuje slot na SD kartu a dva micro USB sloty. Na DPS má také konektor RJ45 pro propojení přes Ethernet a několik GPIO portů. Srdcem vývojové desky je STM32F746NGH6 Arm® Cortex® -M7-core-based mikrokontrolér s 1 MB pamětí Flash a 340 kB RAM paměti. Také, na rozdíl od Raspberry Pi, není nutný operační systém, který by zpomaloval chod aplikace.



Obrázek 9 Vývojová deska STM32F746G-DISCOVERY [8]

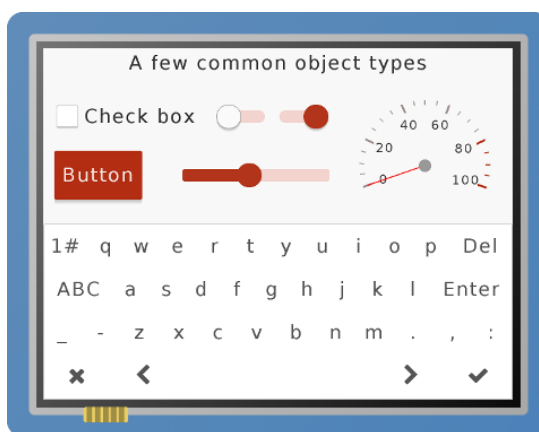
2.3 Grafické rozhraní

Pro danou vývojovou desku se vyskytuje hned několik variant pro návrh grafického rozhraní. Existují možnosti jako použít program pro návrh grafického prostředí, který následně vygeneruje kód, a ten je možno zaimplementovat do svého projektu. Další možností je použít software, kde se opět dá navrhnout grafické zobrazení, a pak vytvoří projekt, do kterého se vpisují úryvky vlastního kódu. Jinou variantou je použití prosté grafické knihovny, kterou je potřeba portovat do svého projektu nebo vytvoření vlastní grafické knihovny.

2.3.1 LittlevGL

LittlevGL je bezplatná grafická knihovna s otevřeným zdrojovým kódem, která poskytuje vše, co je potřeba pro vytvoření integrovaného grafického uživatelského rozhraní se snadno použitelnými grafickými prvky, vizuálními efekty a nízkou paměťovou náročností.

Lze vytvářet tlačítka, grafy, seznamy, posuvníky, obrázky a jiné. Pokročilá grafika s animacemi, vyhlazováním a plynulým posouváním. Z těchto prvků lze pak vytvářet i větší stavební bloky.



Obrázek 10 Ukázka grafického rozhraní LittlevGL [9]

2.3.2 Embedded wizard

Embedded Wizard je integrovaná GUI technologie TARA Systems, která umožňuje vytvářet grafická uživatelská rozhraní nezávislá na platformě a vysoce výkonná. Paměťově je více náročná. Pro použití je nutné stáhnout software, ve kterém se provádí grafický design. Následně je vygenerován projekt, do kterého se vpisuje vlastní kód na přesně daná místa. Celý projekt pak běží na operačním systému. Embedded Wizard je k dostání ve dvou variantách, a to placené verzi bez omezení, nebo verzi zdarma, kde jsou ovšem jistá omezení, co se týče počtu použitých prvků a celkového výkonu prostředí. Je zde ale znát vysoká úroveň. Technologii využívá například automobilka Porsche.



Obrázek 11 Ukázka grafického rozhraní Embedded Wizard u Porsche Panamera [10]

2.3.3 TouchGFX

TouchGFX je bezplatný grafický software optimalizovaný pro mikrokontroléry STM32. Umožňuje vytvářet pokročilejší grafická uživatelská rozhraní. Jeho základem je grafický designer, který na základě návrhu generuje kód. Je distribuován jako součást STM32CubeMX, a je tedy oficiálně doporučován společností STMicroelectronics pro vývoj grafických rozhraní.



Obrázek 12 Ukázka grafického rozhraní TouchGFX [11]

2.4 Komunikační rozhraní

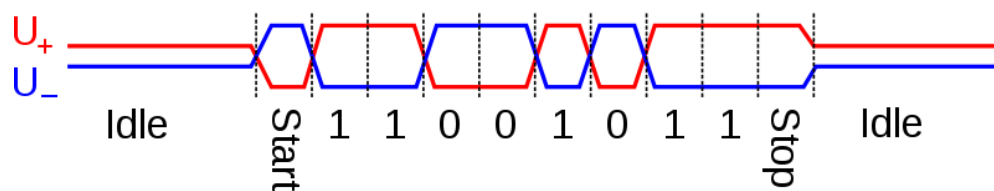
Prism 8x8 od společnosti Symterix disponuje komunikací přes standard RS-485 a Ethernet. Zejména u Ethernetu je vhodné se zamyslet nad použitým komunikačním protokolem. V následujících podkapitolách jsou stručně rozebrány tato komunikační rozhraní.

2.4.1 Standard RS-485

RS-485 je komunikační standard pro sériový přenos používaný především v průmyslovém prostředí. Je navržen tak, aby umožňoval vícebodové poloduplexní dvou vodičové sériové spojení. Vychází ze standardu RS-232, má ale jinak definované napěťové úrovně. Komunikace je možná až do vzdálenosti 1200 m a lze spojit až 32 zařízení. Je kompatibilní se standardem RS-232 pomocí jednoduchých převodníků úrovní.

Dvou vodičová komunikace, kde jsou tyto vodiče označeny písmeny A a B, je založena na rozdílovém napětí mezi těmito vodiči, které tvoří logické úrovně. Díky tomu je eliminován rušivý indukovaný signál, který se povětšinou přičítá k oběma vodičům stejně. I přesto, že RS-485 pracuje s rozdílovým napětím, je potřeba při větších vzdálenostech propojit i země. Ve vzdálených místech mohou totiž být podstatné rozdíly v potenciálech GND. Dvou vodičové spojení tedy bývá často spíše třívodičové.

Přenášená data mají buď 7 nebo 8 bitů se startbitem, 1 nebo více stopbity a může obsahovat i paritní bit. Startbit má podobu logické nuly a stopbit s neaktivním stavem logické jedničky. Protože je přenos poloduplexní, je zapotřebí zajistit řízení přenosu dat, které není součástí standardu RS-485. Nejednodušší variantou je, že jen jedna stanice je vysílací a zbytek jsou přijímací stanice [12].



Obrázek 13 Rámec standardu RS-485

2.4.2 Ethernet

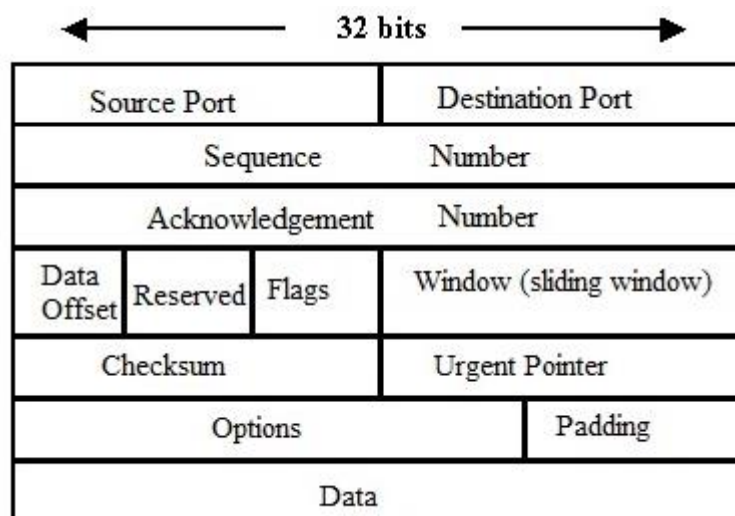
Ethernet je název souhrnu běžně používaných počítačových síťových technologií jako je lokální síť (LAN), metropolitní síť (MAN) a rozsáhlá síť (WAN). Přenosovým médiiem bývají kabely s kroucenou dvoulinkou nebo optické kabely.

Systémy komunikující přes Ethernet rozdělují tok dat na kratší části nazývané rámce. Každý rámec obsahuje zdrojovou a cílovou adresu a data pro kontrolu chyb, aby bylo možné detekovat a zahodit poškozené rámce, protokoly vyšší vrstvy nejčastěji spouštějí opakovaný přenos ztracených rámců.

Sítě Ethernet realizují fyzickou a linkovou vrstvu referenčního modelu OSI. Je možné po nich provozovat jeden nebo více protokolů síťové vrstvy, například protokoly IPv4 a IPv6, které ovšem přepravují data bez záruky. Negarantují doručení, ani zachování pořadí, či vyloučení duplicit. To je ponecháno vyšší vrstvě. Protokoly ve vyšší vrstvě, a to transportní vrstvě, jsou pak protokoly TCP/IP a UDP.

- **TCP/IP**

TCP (Transmission Control Protocol) je nejvyužívanějším protokolem transportní vrstvy. Tento protokol zaručuje jak spolehlivé doručování dat, tak i doručení ve správném pořadí. Další výhodou TCP je schopnost rozlišovat a rozdělovat data pro více aplikací, které běží na stejném počítači.



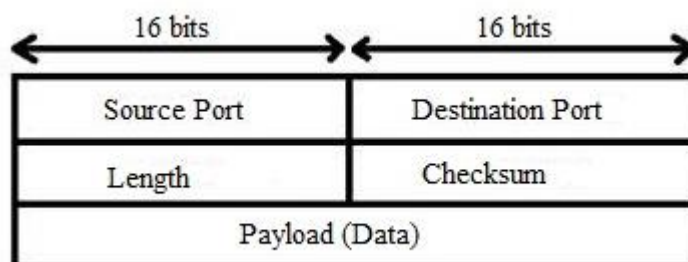
Obrázek 14 Rámec TCP/IP [13]

- **UDP**

UDP (User Datagram Protocol) je další ze sady protokolů. Protokol UDP nedává záruky na datagramy, které přenáší. Bývá označován jako nespolehlivý, ale správněji by mělo být bez záruky doručení, což je hlavní rozdíl proti protokolu TCP.

UDP protokol se uplatní tam, kde je zapotřebí jednoduchost, malá režie nebo v aplikacích, které pracují na principu otázka a odpověď.

Schopností UDP je rozřídování paketů UDP mezi více aplikací a také je přidán kontrolní součet.



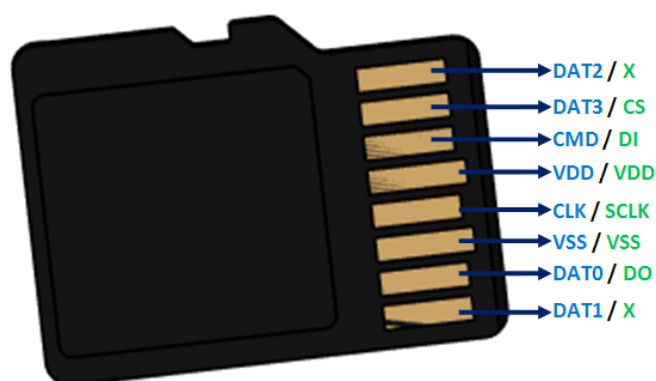
Obrázek 15 UDP rámec [13]

2.5 SD karta

SD karta neboli Secure Digital je paměťovým médiem, sloužícím k uložení elektronických dat. Formát SD je mezi paměťovými kartami tím nejpoužívanějším. Existují pak i upravené verze SD karty jako je třeba MicroSD karta, která je odvozena z původního standardu, ale má menší rozměry.

Funguje na bázi paměti typu flash. Je tedy elektricky programovatelná a vnitřně organizovaná do bloků. Každý blok lze samostatně programovat. Data zůstávají na paměťové kartě uložena i bez napájení.

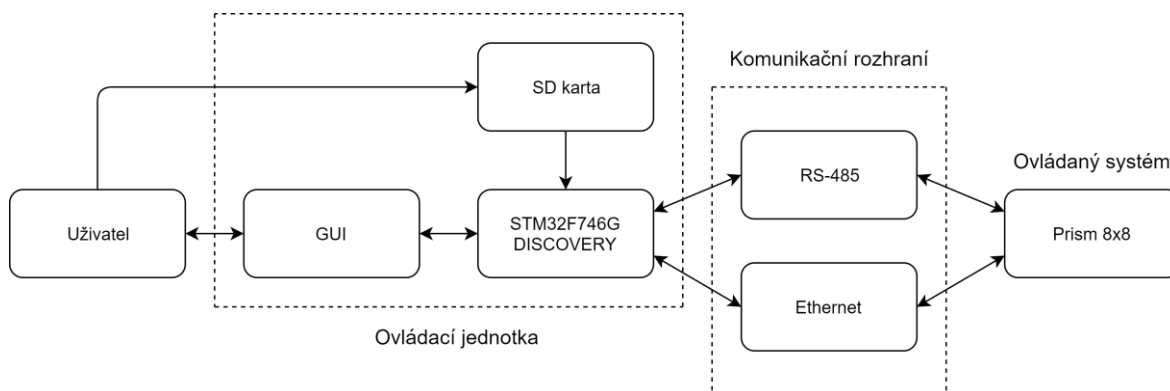
Životnost každé paměťové buňky celé flash paměti je omezena. Záleží na tom, kolikrát jsou jednotlivé buňky přepsány, obecně se udává, že buňka zvládne okolo 100 000 cyklů přepsání.



Obrázek 16 MicroSD karta pinout [14]

3 Realizace

V následujících podkapitolách je popsán princip ovládací jednotky a jsou zde rozebrány jednotlivé bloky, ze kterých se skládá, jak je zobrazeno na obrázku číslo 17. Je zde popsán princip vytváření GUI, možnost konfigurace pomocí SD karty, komunikace přes RS-485 a Ethernet a návrh DPS pro jednotku. Nakonec je rozebrána možnost propojení s nadřazeným systémem.



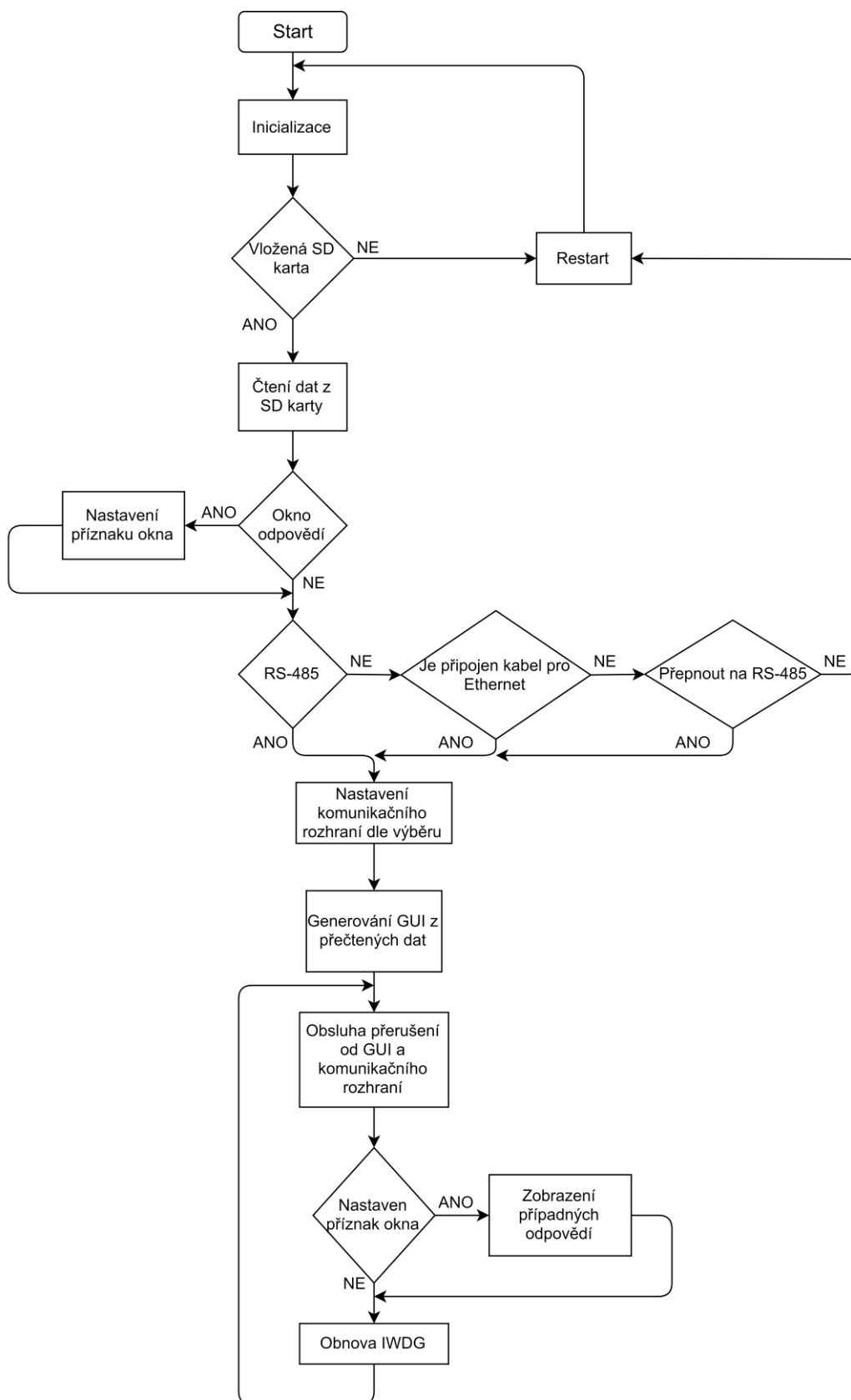
Obrázek 17 Bloková struktura

3.1 Ovládací jednotka a její algoritmus

Srdcem ovládací jednotky je vývojová deska STM32F746G-DISCOVERY, která disponuje 4,3palcovým dotykovým displejem pro grafické uživatelské rozhraní. Toto rozhraní je na volbě uživatele, který si jej může přizpůsobit k vlastním požadavkům pomocí konfiguračního souboru na MicroSD kartě. Ovládací jednotka disponuje detekcí vložení MicroSD karty. Uživatel může vytvořit až 30 ovládacích prvků a přiřadit je možno tyto prvky až do 4 nezávislých menu. Jako komunikační rozhraní lze zvolit buď standard RS-485 nebo Ethernet. U Ethernetu je implementována detekce připojeného kabelu a možnost konfigurace IP adres pomocí souboru na MicroSD kartě. Při výběru komunikačního rozhraní má uživatel dále možnost nastavení zobrazení okna odpovědí od

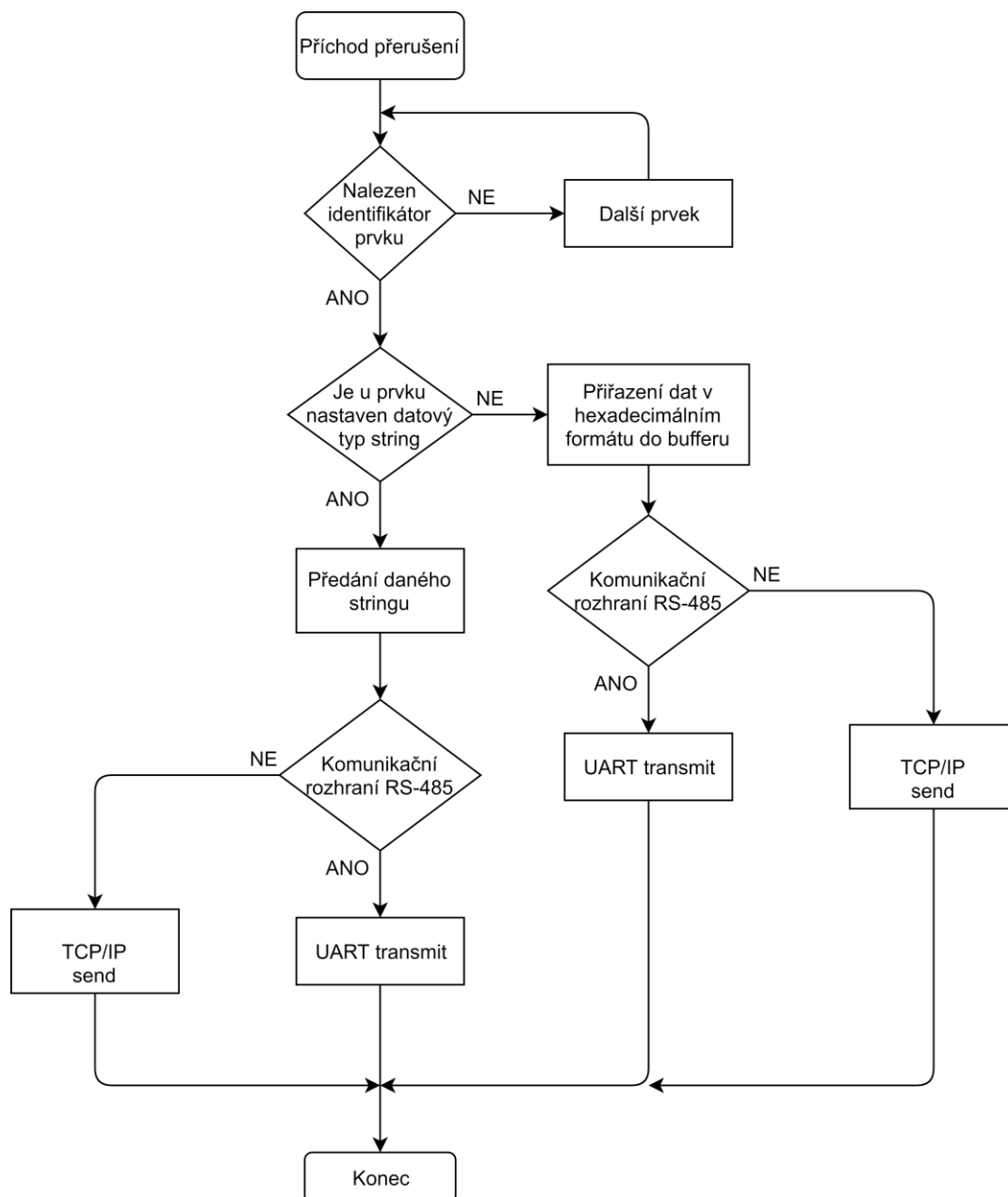
ovládaného systému, je-li to vyžadováno komunikačním protokolem, které je zobrazeno v jednotlivých menu. Po zvolení komunikačního rozhraní je zobrazeno GUI, kde v horní liště jsou obsažena jednotlivá menu, mezi kterými lze přepínat a pod touto lištou jsou pak zobrazeny ovládací prvky náležící do daného menu. V ovládací jednotce je implementován IWDG neboli nezávislý watchdog, který kontroluje průběh programů ovládací jednotky a v případě potíží jednotku restartuje. Pro jednotku je vytvořena deska plošných spojů disponující budičem RS-485 s galvanickým oddělením a vstupem pro 12 V DC napájení.

Vývojový diagram níže popisuje průběh programu ovládací jednotky. Po zapnutí jednotky probíhá inicializace. Před čtením dat z MicroSD karty je provedena detekce vložení karty. Pokud karta není vložena, je uživatel vyzván k jejímu vložení a restartu zařízení. Je-li karta vložena, proběhne čtení dat z konfiguračních souborů. Po jeho dokončení je zobrazeno menu s volbou komunikačního rozhraní a možností výběru zobrazení okna s odpověďmi od ovládaného systému. Zvolí-li uživatel Ethernet, je provedena detekce připojeného kabelu k jednotce. Pokud není kabel připojen, uživatel je vyzván k jeho připojení a následnému restartu zařízení, případně k volbě standardu RS-485. Po výběru komunikace proběhne generování grafického rozhraní podle přečtených dat a přiřazení obsluhy přerušení daným ovládacím prvkům. Program následně přejde do nekonečné smyčky, kde se kontroluje, zda přišlo přerušení od nějakého ovládacího prvku nebo v případě povolení okna odpovědi přišla odpověď od ovládaného systému. Během průchodu nekonečnou smyčkou je také pokaždé obnoven IWDG.



Obrázek 18 Vývojový diagram ovládací jednotky

Obsluha přerušení od ovládacích prvků je vždy tvořena tím, že po stisknutí prvku dochází k identifikaci zdroje přerušení a tím k nalezení příslušných dat, která mají být odeslána. Následuje zjišťování, zda se mají odeslat data zadaná v podobě textového řetězce, či data v hexadecimálním formátu. Tyto data jsou poté předána a odeslána komunikačním rozhraním, které bylo zvoleno při zapnutí jednotky. Postup obsluhy přerušení od tlačítka je zobrazen na následujícím vývojovém diagramu.



Obrázek 19 Obsluha přerušení ovládacího prvku

3.2 Grafické rozhraní

Pro vytvoření GUI bylo zvoleno LittlevGL. Díky faktu, že LittlevGL je prostá grafická knihovna, je nejvhodnější k automatickému generování prvků a přiřazování do jednotlivých menu dle požadavku uživatele. Toho je docíleno pomocí konfiguračního souboru obsaženém na MicroSD kartě, který je přečten při zapnutí ovládací jednotky. Uživatel si navolí menu a následně do nich přiřadí jednotlivé ovládací prvky s předem definovanými parametry. Jsou k dispozici základní ovládací prvky typu tlačítko, posuvník a zaškrťovací políčko (check-box).

Jako příklad, jak může vypadat kód pro vytvoření GUI pomocí LittlevGL, je na následující obrázku zobrazen výstřižek kódu pro vytvoření menu s volbou komunikačního rozhraní.

```
void Starting_menu(void)
{
    lv_obj_t *scr = lv_cont_create(NULL, NULL);
    lv_disp_load_scr(scr);

    lv_obj_t *label = lv_label_create(lv_scr_act(), NULL);
    lv_label_set_text(label, "Communications:");
    lv_obj_align(label, NULL, LV_ALIGN_CENTER, 0, -90);

    btn_tcp = lv_btn_create(lv_disp_get_scr_act(NULL), NULL);
    lv_obj_set_event_cb(btn_tcp, tcpip_event);
    lv_obj_t *label1 = lv_label_create(btn_tcp, NULL);
    lv_label_set_text(label1, "TCP/IP");
    lv_obj_align(btn_tcp, NULL, LV_ALIGN_CENTER, 0, 40);

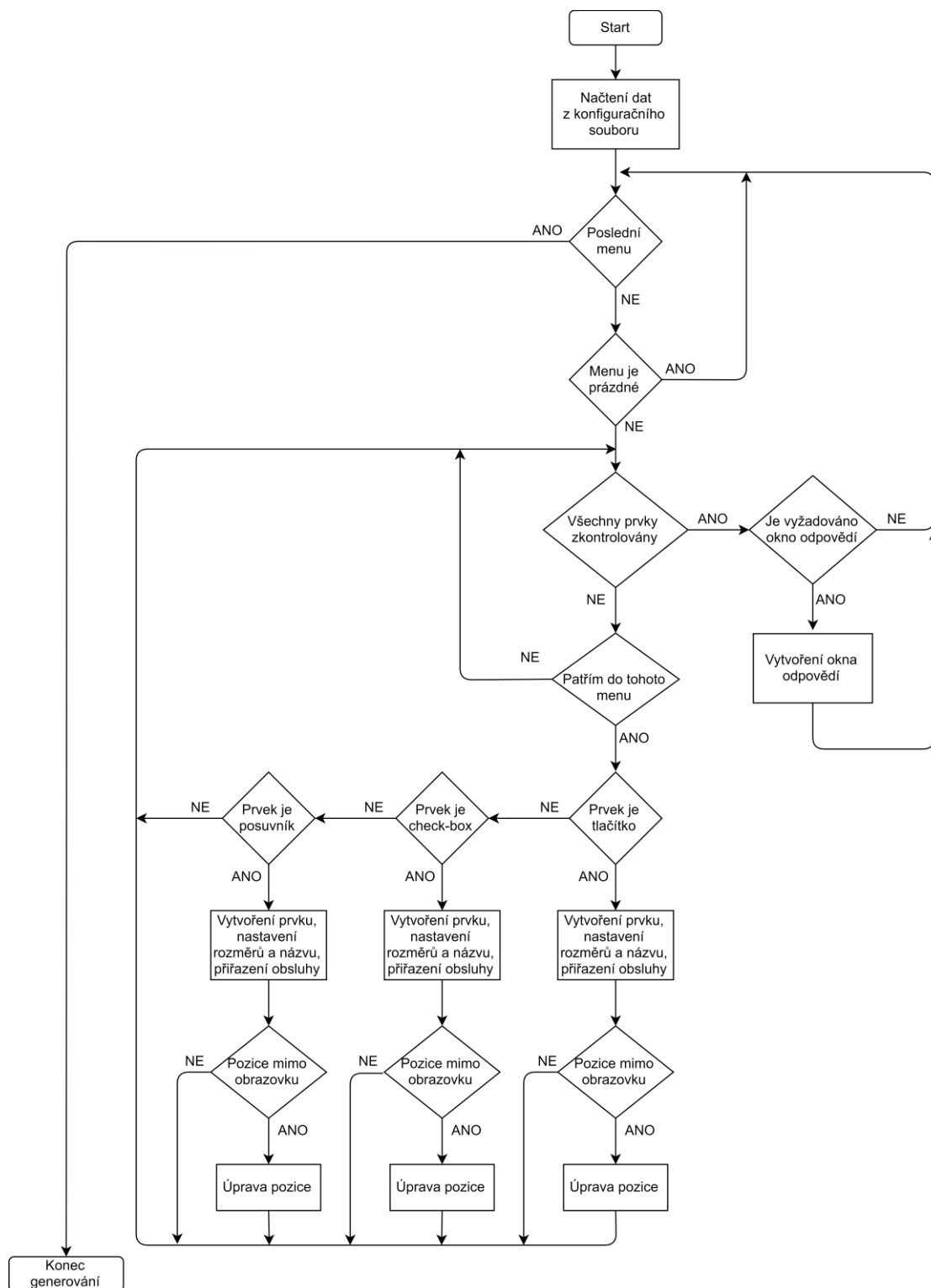
    lv_obj_t *btn2 = lv_btn_create(lv_disp_get_scr_act(NULL), NULL);
    lv_obj_set_event_cb(btn2, rs485_event);
    lv_obj_t *label2 = lv_label_create(btn2, NULL);
    lv_label_set_text(label2, "RS485");
    lv_obj_align(btn2, NULL, LV_ALIGN_CENTER, 0, -40);

    lv_obj_t *checkbox_win = lv_cb_create(lv_disp_get_scr_act(NULL), NULL);
    lv_obj_set_event_cb(checkbox_win, checkbox_win_event);
    lv_cb_set_checked(checkbox_win, false);
    lv_cb_set_text(checkbox_win, "Answer window");
    lv_obj_set_size(checkbox_win, 20, 20);
    lv_obj_align(checkbox_win, NULL, LV_ALIGN_IN_BOTTOM_LEFT, 0, 0);
} /* Starting_menu( ) */
```

Obrázek 20 Ukázka LittlevGL kódu

3.2.1 Algoritmus generování GUI

Na následujícím vývojovém diagramu je nastíněn algoritmus pro generování GUI pomocí dat obsažených v konfiguračním souboru uloženém na MicroSD kartě.

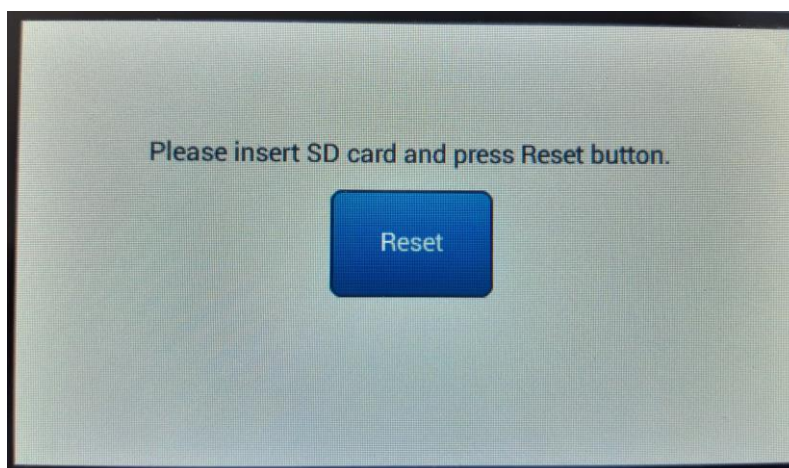


Obrázek 21 Algoritmus generování GUI

Při zapnutí ovládací jednotky jsou přečteny data z konfiguračního souboru na MicroSD kartě. Soubor obsahuje názvy jednotlivých menu a prvků v nich obsažených. Dále jejich parametry a data, která mají odesílat. Po načtení dat se prochází jednotlivá menu a případně prvky, které do nich náleží. Jsou-li menu prázdná, nebudou zobrazena. Náleží-li prvek do menu, detekuje se, o jaký druh ovládacího prvku se jedná. Následně je mu přiřazena obsluha a nastaví se jeho parametry jako je například velikost a pozice na obrazovce.

3.3 Konfigurační soubory

Pro ovládací jednotku jsou vytvořeny dva konfigurační soubory uložené na MicroSD kartě. První konfigurační soubor je pro vytváření GUI ovládací jednotky a druhý je pro nastavení komunikace pro Ethernet. Oba soubory jsou ve formátu CSV a jako oddělovač je použit středník (;). Pokud není vložena MicroSD karta při spouštění ovládací jednotky, je uživatel vyzván k jejímu vložení a následnému resetu systému. Bez vložené MicroSD karty pro korektní vytvoření GUI nelze jednotku spustit.



Obrázek 22 Detekce vložení MicroSD karty

3.3.1 Konfigurační soubor pro GUI a jeho čtení

Pro uživatelskou konfiguraci grafického rozhraní je uložen soubor na MicroSD kartě, pomocí kterého se nastavují jednotlivá menu a grafické dotykové ovládací prvky. Tento soubor je nazván "cfg_file.csv".

Soubor lze rozdělit na dvě sekce. V první sekci se nastavují názvy jednotlivým menu. Následně pak v druhé sekci se přidávají do jednotlivých menu ovládací prvky. Maximální počet menu je nastaven na čtyři. Lze nastavit i menší počet menu tím, že do nepoužívaného menu nepřidáme žádný ovládací prvek. Toto menu je následně ignorováno i přes to, že mu byl přiřazen název. Pro editaci menu je určena pouze 3. řádka souboru (viz obrázek 23).

	A	B	C	D	E
1	Name of menu:				
2	menu1	menu2	menu3	menu4	
3	Preset	Volume	Bas	Data	

Obrázek 23 Nastavení názvů GUI menu

Ve druhé sekci kromě přiřazování ovládacích prvků do příslušných menu se navíc těmto prvkům nastavují jejich parametry. Je nutné dodržovat přesně danou syntaxi, v opačném případě prvek nebude vytvořen nebo jeho funkce bude nesprávná. Celkový počet prvků k vytvoření je omezen na třicet. Jednotlivé parametry jsou zapisovány do následujících kolonek:

- **Object**

Tato kolonka určuje, o jaký ovládací prvek se jedná. Při zapsání "btn" je vytvořen ovládací prvek typu tlačítko, při "slider" je vytvořen posuvník a při "checkbox" je vytvořen check-box neboli zaškrťávací políčko.

- **Name**

Zapsaný text slouží jako název ovládacího prvku a bude zobrazen na obrazovce. Akceptovány jsou velká a malá písmena bez háčeků a čárek, případně číslice.

- **Menu**

Určuje příslušnost daného prvku do předem vytvořených menu. Při zapsání "menu1" je prvek přiřazen do prvního menu, "menu2" náleží druhému menu a takto to analogicky pokračuje až do menu čtyři.

- **Size X a Size Y**

Slouží k zadání rozměrů ovládacího prvku. Hodnoty musejí být kladné. Při zadání většího X než Y u posuvníku, bude posuvník orientován horizontálně a v opačném případě bude orientován vertikálně.

- **Position X a Position Y**

Kolonky určují souřadnice zobrazení prvku na obrazovce, kde nulový bod se nalézá v pravém horním rohu pod výběrem jednotlivých menu. Zadaný bod je souřadnice pravého horního rohu ovládacího prvku. Pokud je zadána souřadnice mimo rozsah displeje nebo by prvek svým rozměrem přesahoval, je prvek automaticky usazen na kraj této hranice.

- ***Data type***

Určuje, jaký datový typ se bude posílat při manipulaci s ovládacím prvkem. Pokud je zapsáno "string", bude se odesílat textový řetězec. Při zapsání "hex" se posílá pole bajtů v hexadecimálním formátu.

- ***String***

Text zapsaný v této kolonce se odesílá při manipulaci s ovládacím prvkem, pokud je nastaven Data type na "string". Je-li "hex" vyplňuje se "Null" a není použito. Text musí být přizpůsoben komunikačnímu protokolu s ovládaným systémem.

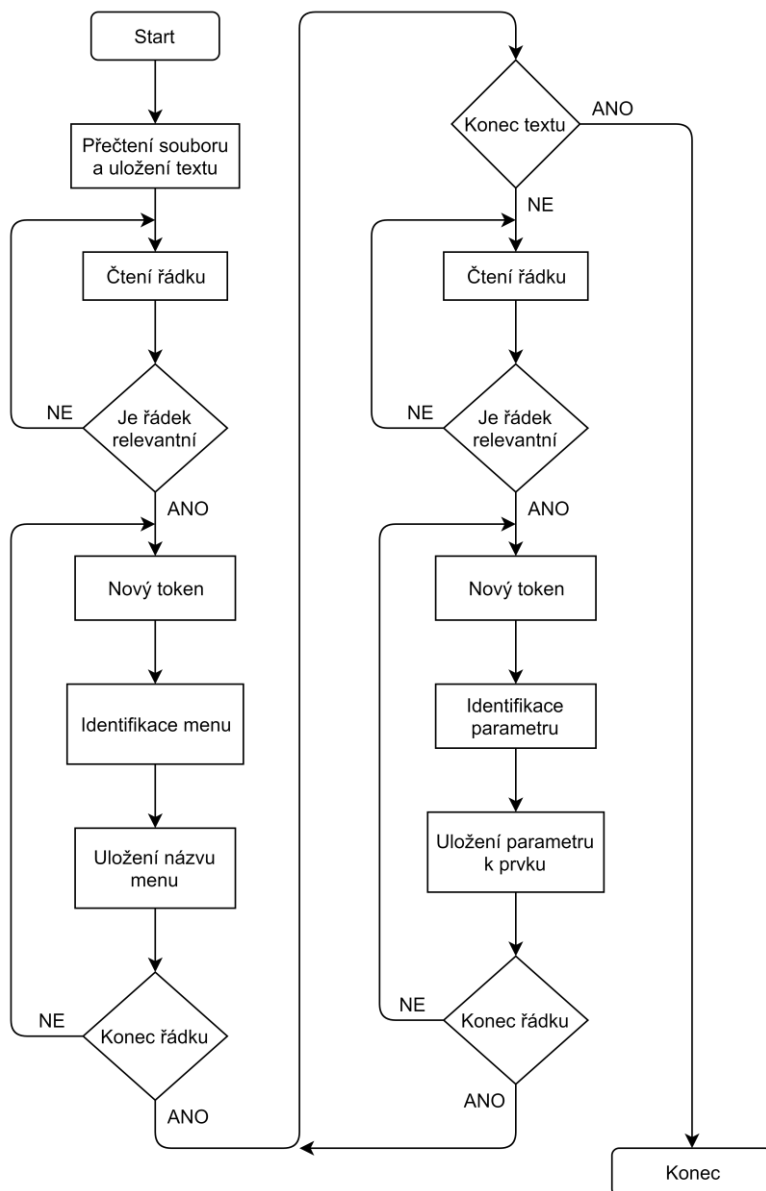
- ***Extension***

Toto pole je na rozdíl od ostatních více variabilní a je závislé na druhu ovládacího prvku k němuž je přiřazeno. Pro posuvník toto pole slouží jako zadání maximální a minimální hodnoty škály posuvníku. Zadává se ve formátu max. | min. škály ("100|0"). U check-boxu slouží jako druhé pole String. Tento řetězec se bude odesílat v případě, že se mění stav daného check-boxu ze zaškrtnutého na nezaškrtnutý. V případě tlačítka se toto pole vyplňuje "Null", protože není využito.

- ***Byte0 Byte1 Byte2 Byte3***

Tato pole se odesílají při manipulaci s ovládacím prvkem v případě, je-li Data type nastaven na "hex". Pokud je "string", vyplňují se "Null" a nejsou využita. Vyplňují se v hexadecimálním formátu. Data musí být přizpůsobena komunikačnímu protokolu s ovládaným systémem.

Na vývojovém diagramu je znázorněno čtení dat z konfiguračního souboru. Nejprve je celý soubor přečten a uložen. Poté jsou jednotlivé údaje roztrženy následujícím způsobem: Nejdříve je přečten řádek souboru a zjistí se, zda je relevantní a obsahuje data či je popisný. Z tohoto řádku je vybrán token, což je text mezi středníky. Ten je následně identifikován a uložen jako příslušný parametr. Po jeho uložení je přečten další token. Takto se postupuje po řádcích až do konce textu.

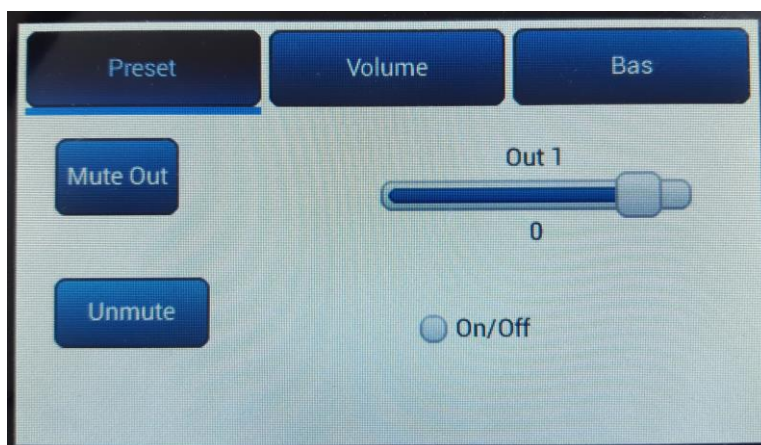


Obrázek 24 Vývojový diagram čtení dat z konfiguračního souboru

Na následujícím obrázku je uveden příklad konfiguračního souboru pro GUI a následně vygenerované grafické prostředí.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Name of menu:														
2	menu1	menu2	menu3	menu4											
3	Preset	Volume	Bas	Data											
4	Object	Name	Menu	Size X	Size Y	Position X	Position Y	Data type	String	Extension	Byte0	Byte1	Byte2	Byte3	
5	btn	Mute Out	menu1	80	50	20	10	string	CS 1 65535	Null	Null	Null	Null	Null	
6	slider	Out 1	menu1	200	30	230	15	string	CS 2	12 -72	Null	Null	Null	Null	
7	btn	Unmute	menu1	100	45	20	100	string	CS 1 0	Null	Null	Null	Null	Null	
8	btn	Set	menu2	60	40	200	100	string	CS 2 0	Null	Null	Null	Null	Null	
9	slider	Bas	menu3	300	40	70	40	hex	Null	100 0	0x26	0x46	Null	Null	
10	checkbox	Bas	menu3	40	40	150	120	string	BusOn	BasOff	Null	Null	Null	Null	
11	slider	Vertical	menu2	30	120	40	20	hex	Null	12 -72	0x43	0x56	0x20	0x32	
12	checkbox	On/Off	menu1	40	40	250	120	string	On	Off	Null	Null	Null	Null	
13															

Obrázek 25 Příklad konfiguračního souboru pro GUI



Obrázek 26 GUI generované podle konfiguračního souboru

3.3.2 Konfigurační soubor pro Ethernet

Tento konfigurační soubor slouží pro jednoduchou uživatelskou konfiguraci komunikace TCP/IP. Je uložen na MicroSD kartě a nese název "ip_cfg.csv". Lze nastavovat následující parametry:

- *Client IP*

V této kolonce se zadává IP adresa ovládací jednotky. Formát zadávání IP adresy je IPv4 v dekadických číslech oddělených tečkou.

- *Server IP*

Je určeno pro zadání IP adresy severu, tedy ovládaného systému. Formát IP adresy je opět zadáván dekadicky IPv4.

- *Server port*

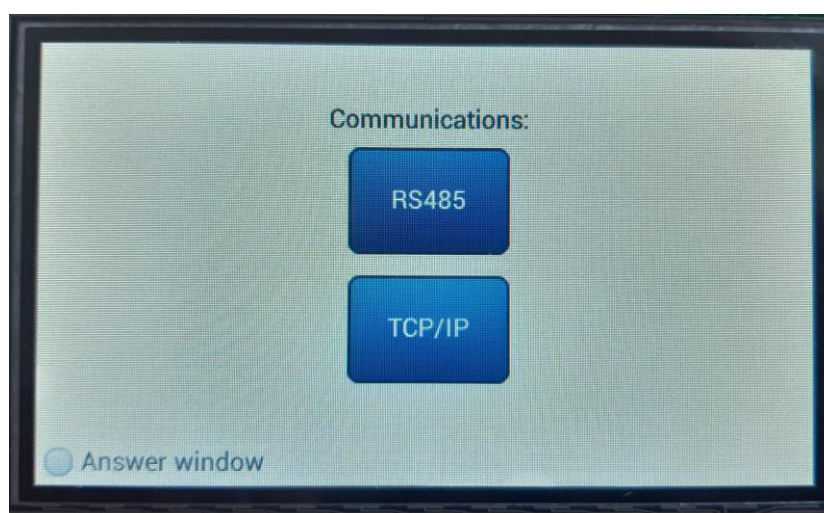
Slouží k nastavení portu, na kterém bude daná komunikace probíhat. Formát portu je dekadický.

	A	B
1	Client IP:	
2	192.168.10.249	
3	Server IP:	
4	192.168.10.250	
5	Server port:	
6	48631	
7		

Obrázek 27 Ukázka konfiguračního souboru pro Ethernet

3.4 Volba komunikačního rozhraní

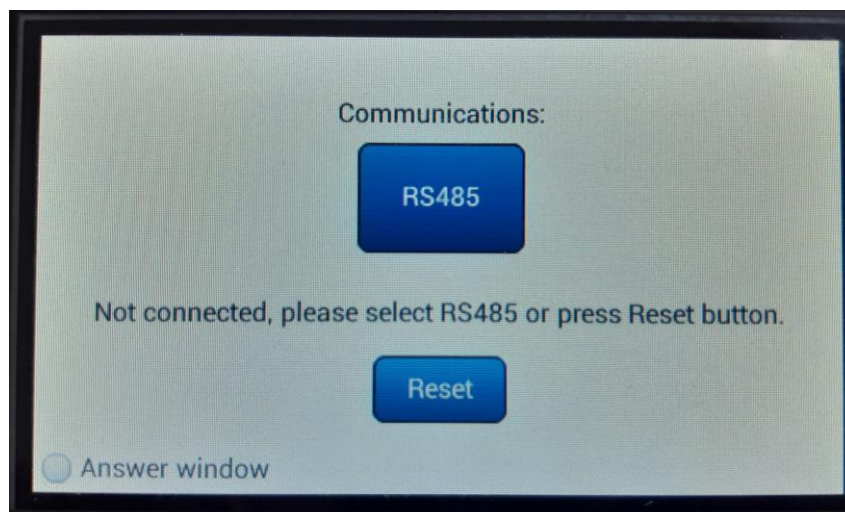
Do ovládací jednotky byla nakonec naimplementována obě dostupná komunikační rozhraní. Při zapnutí ovládací jednotky má uživatel možnost vybrat si z variant Ethernet nebo RS-485. Další možností volby je zobrazení okna s odpověďmi ovládaného systému, které je poté umístěno v levém spodním rohu jednotlivých menu. Okno zobrazuje poslední přijatou zprávu ve všech menu. Tato možnost je zařazena pro případ, kdy je nutná zpětná vazba při komunikačním protokolu s ovládaným systémem.



Obrázek 28 Volba komunikačního rozhraní

- ***Komunikace přes Ethernet***

Při volbě komunikačního rozhraní přes Ethernet je použit protokol TCP/IP, tedy se spolehlivým doručením dat ve správném pořadí. Implementován je lwIP raw TCP/IP klient, který je často používán v embedded systémech. IP adresa ovládací jednotky jako klienta i IP adresa ovládaného systému jako serveru je nastavena pomocí konfiguračního souboru obsaženém na MicroSD kartě. Pomocí tohoto souboru se také nastavuje port, na kterém bude komunikace probíhat. Tato možnost dále zahrnuje detekci, připojení UTP/STP kabelu. Pokud není kabel připojen, má uživatel na výběr přepnutí na rozhraní RS-485, případně musí připojit kabel a vybrat možnost Reset.



Obrázek 29 Detekce nepřipojeného kabelu

- **Komunikace přes RS-485**

Tato komunikace je tvořena na základě převodníku z UART na RS-485 úrovně, kdy je pro přenos dat použito 8 bitů, 1 stop bit a není využito parity. Rychlost přenosu je nastavena na 57600 baud. Z mikrokontroléru jsou vyvedeny signály Rx – příjem dat, Tx – odesílání dat a DE pro určení směru komunikace. Pro příjem dat je využit DMA kanál.

3.4.1 Komunikační protokol

Díky konfiguračnímu souboru má uživatel na výběr ze dvou možných komunikačních protokolů. První z nich je komunikace mezi systémy pomocí textových řetězců, kdy uživatel definuje příkaz jako textový řetězec v konfiguračním souboru v kolonce String. Tento řetězec je odeslán při užití daného ovládacího prvku. V případě posuvníku je za tento řetězec přidána ještě aktuální pozice posuvníku. Pro check-box, který má dva stavy je v kolonce Extension možnost nadefinovat textový řetězec, který bude odeslán při změně stavu ze zaškrtnutého na nezaškrtnutý.

Druhou možností je pak zvolení komunikace mezi systémy pomocí surových dat v hexadecimálním formátu. Pro tyto účely slouží v konfiguračním souboru kolonky Byte0 až Byte3. Zde může uživatel zakódovat příkazy do jednotlivých bajtů v hexadecimálním

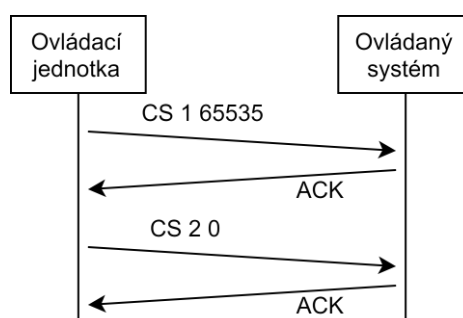
formátu. U posuvníku je za tato data připojena opět aktuální hodnota jeho pozice. V případě check-boxu jsou bajty 0 a 1 použity pro odeslání příkazu při změně stavu z nezaškrtnutého na zaškrtnutý a bajty 2 a 3 jsou použity pro příkaz změny stavu ze zaškrtnutého na nezaškrtnutý. Je-li vyžadována zpětná vazba od ovládaného systému, má uživatel při volbě komunikačního rozhraní možnost zvolit zobrazení okna s odpověďmi systému. Toto okno je poté umístěno ve všech menu v levém dolním rohu.

Ovládaný systém v podobě DSP Prism 8x8 komunikuje pomocí textových řetězců (řetězce ASCII). Obecný formát pro příkaz je:

<PŘÍKAZ> <PARAMETR1> <PARAMETR2> ... <CR>

Mezi příkaz a každý parametr musí být zahrnut znak mezery a na konci musí být přidán znak návratu <CR>. Například příkaz CS (Controller Set) je pro nastavení ovladače. Tento příkaz slouží k přesunutí pozice ovladače na novou absolutní hodnotu. Příkaz musí specifikovat číslo ovladače a jeho novou pozici. CS 1 0 <CR> nastaví ovladač číslo jedna na nulovou pozici.

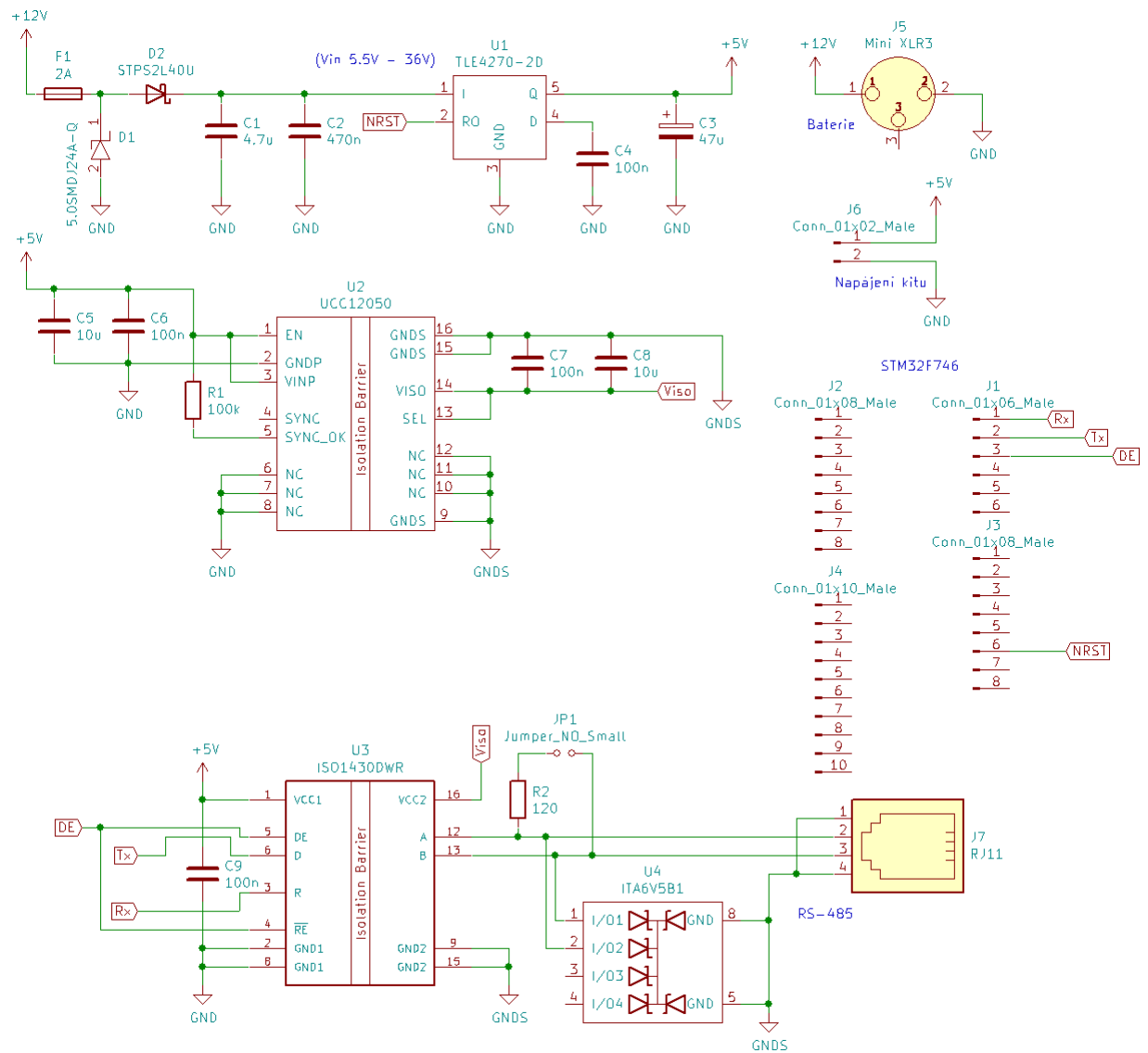
Pokud je příkaz přijat, ovládaný systém odpoví na každý příkaz pomocí potvrzovacího řetězce, jehož syntaxe se podle druhu příkazu může měnit. Ve většině případů je odpověď ve tvaru ACK, je-li příkaz přijat, a NAK, pokud příkaz z nějakého důvodu selhal. Komunikační protokol je tedy typu Acknowledgment Data Transfer neboli „ping-pong“. Více informací ke komunikačnímu protokolu pro Prism 8x8 je k nalezení na [15].



Obrázek 30 Komunikační protokol „ping-pong“

3.5 Návrh DPS

Pro ovládací jednotku je navržena deska plošných spojů s budičem pro RS-485 a s napájením pro vstupní napětí 12 V DC. Na následujícím obrázku je zobrazeno schéma zapojení DPS.

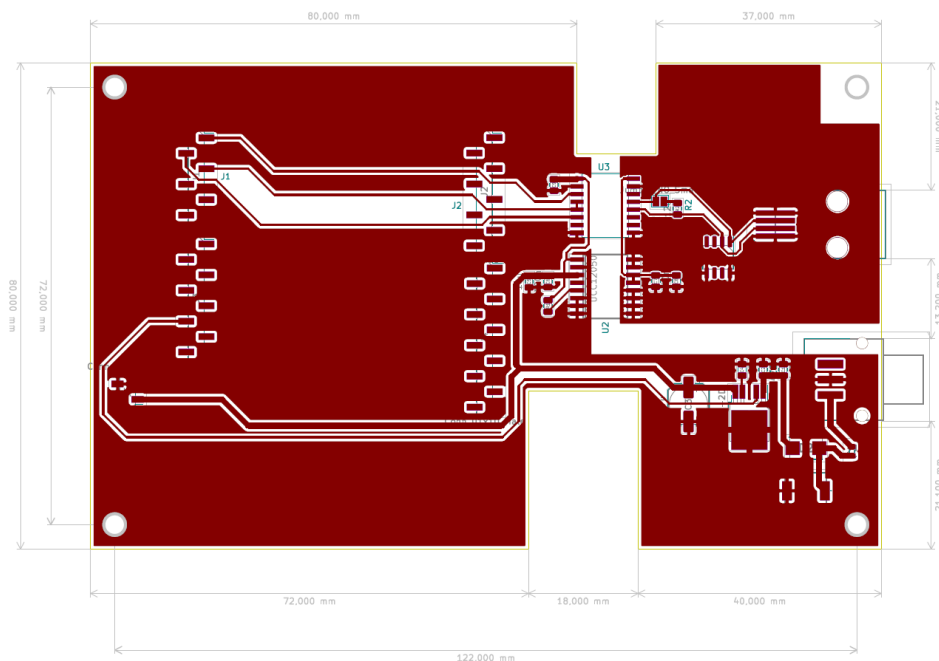


Obrázek 31 Schéma zapojení DPS

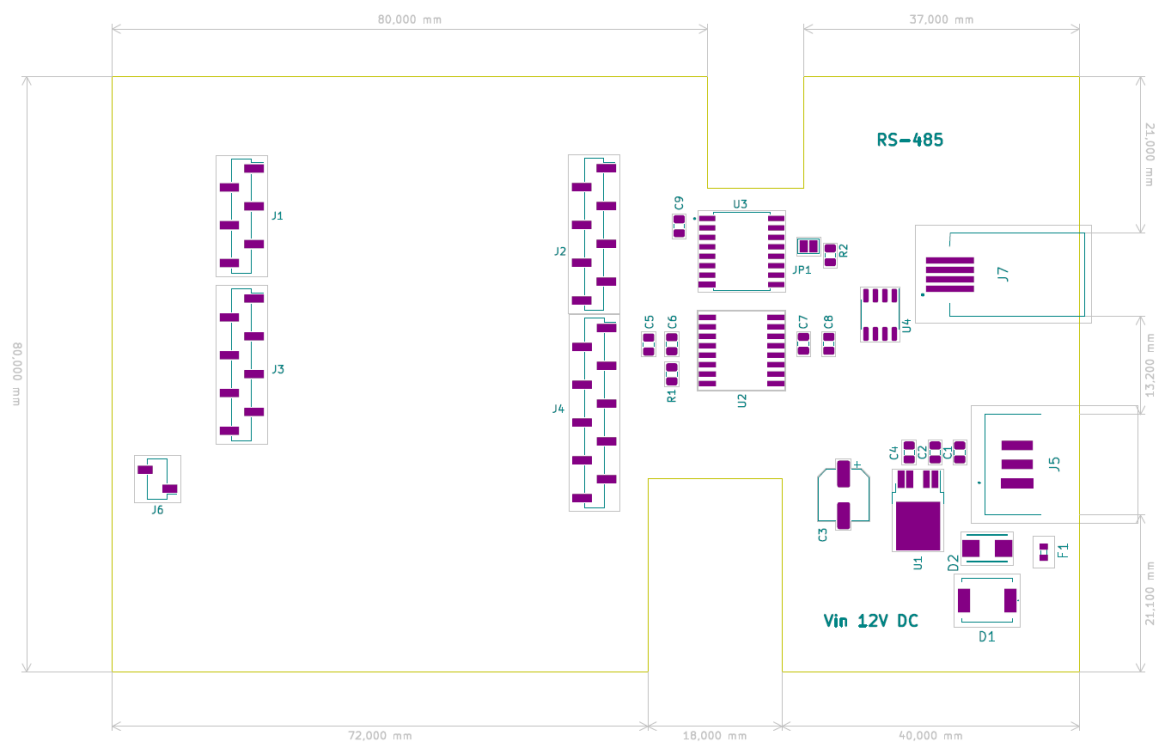
Pro 12 V DC vstup napájení do desky je použit konektor Mini XLR3 s mechanickým závitem. Na vstup jsou umístěny ochranné prostředky jako 2A pojistka nebo jednosměrný transil a Schottkyho dioda pro ochranu před přepětím a změnou polarit napájecího napětí. Jako napěťový regulátor je použit TLE4040-2D, který je určen pro použití v automotive prostředí a pro automotive elektroniku. Má v sobě integrované následující ochrany: ochranu proti přehřátí, ochranu proti změně polarit napájení a ochranu přepětí do 65 V DC. Vstupní napětí může dosahovat až 42 V a výstupní proud dosahuje 650 mA. Regulátor generuje signál pro reset, pokud jeho výstupní napětí klesne pod 4,5 V. Pro konstantní 5 V výstup by se mělo vstupní napětí pohybovat v rozmezí 5,5 až 36 V.

Jako budič RS-485 je použit ISO1430, který je vhodný do prostředí s větším rušením a má lepší EMC ochranu. ISO1430 je izolovaný budič s barierou 5 kV. Signály pro RS-485 jsou vyvedeny na konektor RJ11. Uživatel má možnost, je-li to vyžadováno, připojit terminační odpor 120 Ω mezi signály A a B. Pro ochranu RS-485 je také přidáno transilové pole ITA6V5B1. Pro vytvoření izolovaného napětí je použit izolovaný low-EMI DC-DC převodník. Jeho izolační bariéra je opět 5 kV.

Přes pin header je k DPS připojena vývojová deska STM32F746G-DISCOVERY, ze které jsou vyvedeny signály pro RS-485, a to Rx, Tx a DE. Do STM32 je přiveden signál NRST, který je generován při poklesu napětí pod 4,5 V a vyvolá restart ovládací jednotky.



Obrázek 32 DPS pro ovládací jednotku



Obrázek 33 Rozmístění součástek na DPS

Tvar desky plošných spojů je upraven podle rozměrů a tvaru vývojové desky STM32F746G-DISCOVERY. Rozmístění součástek a konektorů je přizpůsobeno možnostem desky, aby DPS šla přichytit na piny k vývojové desce od STM.

Deska je koncipovaná na sekce a jsou zde dva potenciály země navzájem izolované. Jeden pro napájení desky GND a druhý pro RS-485 GNDS. Země jsou odděleny izolační mezerou patrnou na obrázku číslo 32.

3.6 Uživatelská konfigurace a propojení s nadřazeným systémem

Ovládací jednotka je koncipovaná tak, aby měl uživatel širokou možnost volby. To platí zejména u generování grafického rozhraní, které je takřka celé na volbě uživatele. Jediným omezením je maximální počet menu, který je stanoven na 4 a mezi tyto menu lze rozdělit až 30 ovládacích prvků.

Uživatel má možnost volby i v použití komunikačního rozhraní, a to mezi standardem RS-485 a Ethernetem. Přes tyto komunikační rozhraní lze posílat obecně definované příkazy pomocí konfiguračního souboru uloženého na MicroSD kartě. Lze posílat příkazy ve formě textových řetězců nebo případně zakódovat příkazy do surových dat v hexadecimálním formátu.

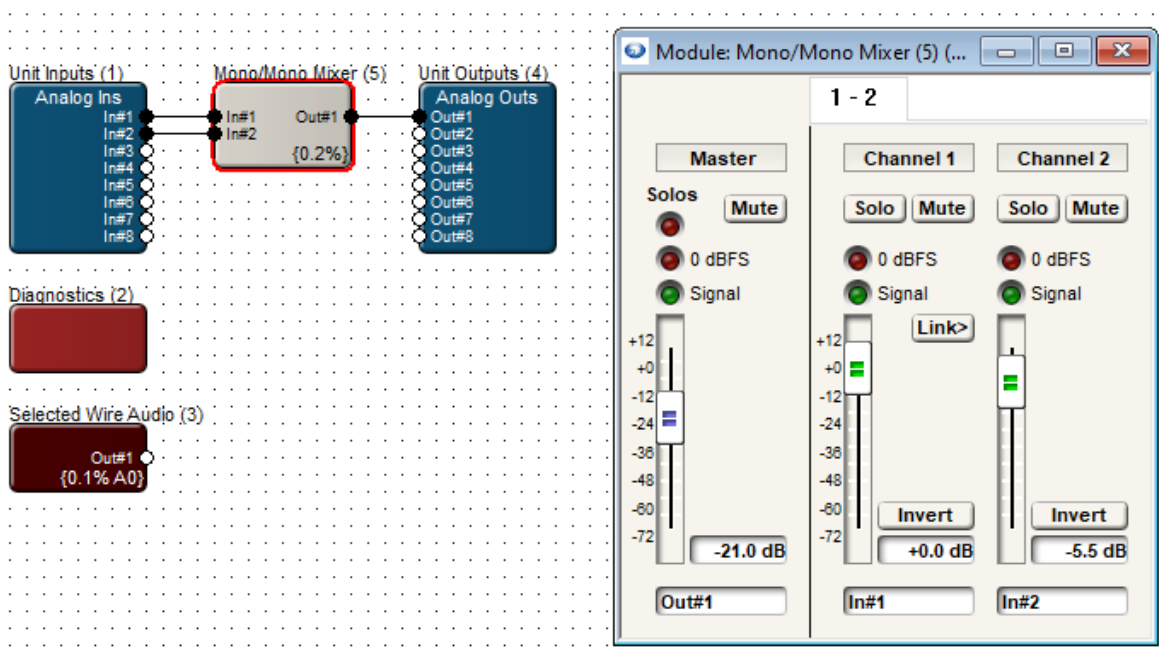
Podle zadaných požadavků je ovládací jednotka upravena pro použití s ovládaným systémem, a to DSP Prism 8x8 od společnosti Symterix. Úpravy jsou například v přepočtu hodnot pozice jednotlivých posuvníků na hodnoty rozsahu použitých v DSP. Rozsah nastavení pozice se pohybuje od 0 do 65535 a přepočítává se pomocí minimální a maximální hodnoty škály daného posuvníku a aktuální nastavené uživatelské hodnoty v dB. Pro nastavení pozice posuvníku na základě požadovaného uživatelského volume v dB je tedy hodnota vypočítávána takto:

$$\text{Pozice ovladače} = \frac{(\text{uživatelské volume v dB} - \text{minimální hodnota škály}) \times 65535}{(\text{maximální hodnota škály} - \text{minimální hodnota škály})} \quad (1)$$

Prism 8x8 komunikuje pomocí textových řetězců, je tedy i zde lehká úprava kódu pro odesílání příkazů, kdy je za každou zprávu přidán ukončovací znak <CR>. Komunikace přes RS-485 je také upravena k použití s DSP Prism 8x8. Pro přenos dat je použito 8 bitů, 1 stop bit a není využito parity. Rychlost přenosu je nastavena na 57600 baud. U komunikace přes Ethernet s Prism 8x8 je nutné zvolit komunikaci na portu číslo 48631.

Pomocí drobných úprav v kódu se může ovládací jednotka stát velice univerzální a lze ji snadno přenastavit na jiný ovládaný systém, který bude disponovat komunikačním rozhraním standardu RS-485 nebo Ethernetem.

Prism 8x8 je programovatelný pomocí programu Composer. Zde si uživatel znalý problematiky nastaví ovládací jednotky jednotlivých audio signálů, ovládacích prvků a vytvoří si tzv. preset neboli přednastavení, které následně nahraje do DSP. Pro tyto ovládací prvky poté vytvoří v konfiguračním souboru ovládací jednotky obdobné ovládací prvky odpovídající presetu. Těmto prvkům následně přiřadí příkazy, které bude ovládací jednotka odesílat a bude tak ovládat nastavení v Prism 8x8. Na obrázku 34 je ukázka, jak může vypadat jednoduchý preset v programu Composer.



Obrázek 34 Ukázka jednoduchého presetu v programu Composer

Závěr

Cílem práce bylo navrhnout a vytvořit ovládací jednotku pro vzdálené ovládaní audio systému automobilu. Byly stanoveny požadavky na danou jednotku, kterým byl podřízen následný vývoj.

Jako řídicí mikrokontrolér byla zvolena vývojová deska STM32F746G-DISCOVERY, která disponuje dotykovým displejem. Při návrhu grafického uživatelského prostředí byly vyzkoušeny nástroje, které jsou určeny pro danou vývojovou desku, jako je Embedded Wizard nebo Touch GFX. Ty se ovšem pro dané požadavky nakonec neosvědčily. Použita byla tedy prostá grafická knihovna LittlevGL, která je vodná pro generování kódu podle uživatelem zadaných parametrů.

Generování prostředí probíhá pomocí dat, která uživatel zadá do konfiguračního souboru na MicroSD kartě. Tyto data jsou přečtena při zapnutí jednotky a následně zpracována pro vytvoření GUI. V konfiguračním souboru lze přidávat ovládací prvky jako je tlačítko, posuvník, check-box a následně jim nastavit potřebné parametry.

Pro ovládací jednotku jsou použita dvě komunikační rozhraní, a to standard RS-485 a Ethernet. Do jednotky je implementován TCP/IP klient. Pro klienta byly vyzkoušeny varianty Socket a Netconn. Obě tyto varianty měly ale problémy. Jedním z problémů byla nutnost implementace RTOS pro tyto varianty, ovšem z dosud nezjištěných důvodů docházelo ke zpoždování a znovu vysílání paketů. Nakonec bylo tedy potřeba implementovat raw TCP/IP klienta, který nevyžaduje použití RTOS a mohl být tedy odebrán. Tím byly problémy následně odstraněny. Pro nastavení klienta je na MicroSD kartě vytvořen konfigurační soubor, kde může uživatel nastavit IP adresu jak klienta, tak serveru, se kterým bude komunikovat. Dále lze nastavit port na kterém komunikace probíhá. Pro standard RS-485 byla navržena DPS, jako převodník z UART na RS-485 úrovně. Z mikrokontroléru jsou pro komunikaci vyvedeny signály Rx, Tx a DE, který určuje směr komunikace pro poloduplexní přenos.

Navržená DPS neslouží jen pro komunikaci přes RS-485, ale je na ní i napájení pro ovládací jednotku. Vstupní napájení desky je 12 V DC, které je pomocí napěťového regulátoru upraveno na 5 V pro vývojovou desku. Budič RS-485 je izolovaný, proto je na desku přidán izolovaný DC-DC převodník. Jako konektor pro vstupní napájení je použit MiniXLR3 a pro RS-485 je použit RJ11.

Práce byla pojata tak, aby splňovala zadané požadavky na ovládací jednotku. Generování grafického uživatelského prostředí je snadné pomocí konfiguračního souboru uloženého na externím mediu. Komunikace přes Ethernet s ovládaným systémem DSP Prism 8x8 byla ověřena a je funkční. Ovládaný systém je tedy plně ovladatelný pomocí ovládací jednotky. Bohužel komunikace přes RS-485 nebyla prozatím ověřena z důvodu čekání na dodání DPS a součástek.

Seznam literatury a informačních zdrojů

- [1] *Uživatelské rozhraní (user interface)*. In: Wikisofia.cz [online]. [cit. 20-5-2020]. Dostupné z: [https://wikisofia.cz/wiki/Uživatelské_rozhraní_\(user_interface\)](https://wikisofia.cz/wiki/Uživatelské_rozhraní_(user_interface))
- [2] *Music Player Dark Mobile GUI*. In: Creativemarket.com [online]. [cit. 20-5-2020]. Dostupné z: <https://creativemarket.com/MarabuDesign/2103103-Music-Player-Dark-Mobile-GUI>
- [3] In: Ebay [online]. [cit. 15-5-2020]. Dostupné z: <https://www.ebay.co.uk/itm/BMW-CIC-idrive-controller-wheel-8-button-E60-E90-E81-E70-X5-X6-1-3-5-65829205177-/253045660441>
- [4] *BMW řada 8*. In: Autohled [online]. [cit. 16-5-2020]. Dostupné z: <https://www.autohled.cz/a/bmw/rada-8-cabrio>
- [5] *Škoda Kodiaq*. In: Automotive-ux.com [online] 28. 11. 2016. [cit. 16-5-2020]. Dostupné z: <http://www.automotive-ux.com/news/skoda-kodiaq/>
- [6] Fred Lambert. *Tesla releases a mobile app update with new charging feature*. In: Electrek.co [online] 06. 12. 2018. [cit. 16-5-2020]. Dostupné z: <https://electrek.co/2018/12/06/tesla-mobile-app-update-charging-feature/>
- [7] *Tesla Smartwatch with Camera 2019*. In: Blog.awok.com [online] 06. 12. 2018. [cit. 22-5-2020]. Dostupné z: <https://blog.awok.com/tesla-smartwatch/>
- [8] *32F746GDISCOVERY*. In: STMicroelectronics [online]. [cit. 17-5-2020]. Dostupné z: <https://www.st.com/en/evaluation-tools/32f746gdiscovery.html>
- [9] *Open-source Embedded GUI Library*. In: Littlevgl [online]. [cit. 21-5-2020]. Dostupné z: <https://littlevgl.com/>
- [10] *Embedded Wizard*. In: Embedded Wizard [online]. [cit. 21-5-2020]. Dostupné z: <https://www.embedded-wizard.de/>
- [11] *Touch GFX*. In: TouchGFX [online]. [cit. 21-5-2020]. Dostupné z: <https://www.touchgfx.com/>
- [12] Redakce HW serveru. *RS 485 & 422*. In: Vyvoj.hw.cz [online] 15. 01. 1998. [cit. 26-5-2020]. Dostupné z: <https://vyvoj.hw.cz//teorie-a-praxe/dokumentace/rs-485-422.html>
- [13] Fadil. *TCP and UDP frames*. In: Techsoftcenter.com [online] 13. 10. 2018. [cit. 26-5-2020]. Dostupné z: <https://techsoftcenter.com/tcp-and-udp-frames/>
- [14] *MicroSD Card*. In: Components101.com [online] 10. 06. 2018. [cit. 28-5-2020]. Dostupné z: <https://components101.com/misc/microsd-card-pinout-datasheet>
- [15] *Prism*. In: Symetrix [online]. [cit. 30-5-2020]. Dostupné z: <https://www.symetrix.co/products/prism-0-0-prism-4-4-prism-8-8-prism-12-12-prism-16-16/>

Přílohy

Obsah přiloženého CD:

- **Zdrojový kód**
- **Konfigurační soubory**
- **Výkresy DPS**
- **Podklady pro výrobu DPS**