



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

TRIGGEROVACÍ JEDNOTKA PRO SBĚRNICI CAN

TRIGGERING DEVICE FOR CANBUS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Vojtěch Piroch

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. František Burian, Ph.D.

BRNO 2023

Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Student: Vojtěch Piroch

ID: 221312

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Triggerovací jednotka pro sběrnici CAN

POKYNY PRO VYPRACOVÁNÍ:

Úkolem studenta je navrhnout, realizovat oživit a ověřit funkci jednoduché triggerovací jednotky pro sběrnici CAN pro snadné ladění zařízení se sběrnici CAN s osciloskopem, který tuto sběrnici nepodporuje. Při řešení by měl student klást důraz zejména na nízkou latenci a jitter výstupního signálu. Sběrníkový port by měl být galvanicky oddělený od zemí osciloskopu. Napájení primárně bateriové (uvnitř krabičky) s možností připojení externího zdroje. Součástí práce by měl být i návrh krabičky pro zařízení.

1. Seznamte se se sběrnici CAN a souvisejícími integrovanými obvody.
2. Navrhněte obvodové schéma prototypu realizující funkci.
3. Vyzkoušejte funkci obvodového schématu na prototypu (kontaktní pole, modulky, bez plošného spoje).
4. Navrhněte finální podobu schématu i plošného spoje realizovaného zařízení.
5. Realizujte plošný spoj zařízení.
6. Realizujte vhodný obal pro zařízení (např. 3D tisk).
7. Zařízení otestujte, a změřte skutečný jitter triggerovacího výstupu.

Pro úspěšné splnění semestrálního projektu je nutné splnění bodů 1-4

DOPORUČENÁ LITERATURA:

VOSS, Wilifried. A comprehensible guide to controller area network. Greenfield: Cooperhill Technologies Corporation, [2005]. ISBN 978-0976511601.

Termín zadání: 6.2.2023

Termín odevzdání: 22.5.2023

Vedoucí práce: Ing. František Burian, Ph.D.

doc. Ing. Václav Jirsík, CSc.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato bakalářská práce se zabývá návrhem triggerovací jednotky pro sběrnici CAN, která umožňuje připojení na triggerovací vstup osciloskopu a zobrazení signálových průběhů rámců sběrnice CAN. První část práce se zabývá teoretickým popisem samotné sběrnice CAN, jejím principem fungování, popisem rámců atd. Druhá část práce pojednává o výběru stěžejních součástí potřebných pro zapojení, návrhu finálního schématu, definování a výpočtu jednotlivých součástí v obvodu a způsobu ovládání. Třetí část je praktická, zabývá se testováním prototypu zařízení, návrhem plošného spoje a obalu pro zařízení. Následující úsek charakterizuje softwarovou část zařízení, konkrétně způsob, jak ovládat zařízení, popis programu a knihoven použitých při programování mikrokontroleru. Poslední část popisuje finální testování a měření parametrů, jako jsou jitter a zpoždění triggerovacího pulzu.

Klíčová slova

Sběrnice CAN, triggerovací jednotka, osciloskop, jitter

Abstract

This bachelor thesis deals with the design of a triggering device for the CAN bus that allows connection to the trigger input of an oscilloscope and display of the signal waveforms of the CAN bus frames. The first part of the thesis deals with the theoretical description of the CAN bus itself, its principle of operation, description of frames, etc. The second part of the thesis deals with the selection of the core components needed for the circuit, the design of the final schematic, the definition and calculation of the individual components in the circuit and the method of control. The third part is practical, it is dealing with testing the prototype device, designing the circuit board and packaging for the device. The following section describes the software part of the device, specifically how to control the device, a description of the program and libraries used in programming the microcontroller. The last section describes the final testing and measurement of parameters such as jitter and trigger pulse delay.

Keywords

CAN bus, triggering device, oscilloscope, jitter

Bibliografická citace

PIROCH, Vojtěch. Triggerovací jednotka pro sběrnici CAN. Brno, 2023. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/151290>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce František Burian.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta:	<i>Vojtěch Piroch</i>
VUT ID studenta:	<i>221312</i>
Typ práce:	<i>Bakalářská práce</i>
Akademický rok:	<i>2022/23</i>
Téma závěrečné práce:	<i>Triggerovací jednotka pro sběrnici CAN</i>

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 21. května 2023

podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Františku Burianovi, Ph.D. za odborné rady, a pomoc při zpracování mé bakalářské práce.

V Brně dne: 21. května 2023

podpis autora

Obsah

SEZNAM OBRÁZKŮ	9
SEZNAM TABULEK.....	10
ÚVOD	11
1. SBĚRNICE CAN.....	12
1.1 ZÁKLADNÍ POPIS.....	12
1.2 ARBITRÁŽ	13
1.3 RÁMCE.....	13
1.3.1 Datový rámeček.....	14
1.3.2 Vzdálený rámeček.....	15
1.3.3 Chybový rámeček.....	15
1.3.4 Rámeček přetížení.....	16
1.3.5 CAN FD rámeček	16
1.3.6 CAN XL rámeček	16
1.4 VYHODNOCOVÁNÍ CHYB	17
1.4.1 Čítače chyb	17
1.4.2 Chybové stavy.....	17
1.4.3 Ověřování přeneseného bitu	17
1.4.4 Vkládání bitů.....	17
1.4.5 Kontrola CRC	18
1.4.6 Kontrola předepsaných datových polí	18
1.4.7 Potvrzení zprávy (ACK)	18
1.5 ČASOVÁNÍ BITŮ.....	18
1.5.1 Časová kvanta.....	19
1.5.2 Vzorkovací bod.....	19
1.5.3 Šířka synchronizačního skoku.....	19
1.6 SYNCHRONIZACE BITOVÉ DOBY	19
1.6.1 Zpoždění šíření bitů.....	19
1.6.2 Synchronizace	20
1.7 PARAMETRY ZAŘÍZENÍ	22
2. NÁVRH OBVODU ZAŘÍZENÍ.....	24
2.1 VOLBA HW	24
2.1.1 CAN budič fyzické vrstvy.....	24
2.1.2 CAN kontrolér.....	25
2.1.3 Mikrokontroler.....	25
2.1.4 Periferie pro ovládání zařízení	26
2.2 NÁVRH SCHÉMATU.....	27
2.2.1 Blokové schéma.....	27
2.2.2 Napájecí obvod	27
2.2.3 Mikrokontroler.....	30
2.2.4 Měření napětí akumulátoru.....	30
2.2.5 CAN kontrolér.....	31
2.2.6 CAN budič a galvanické oddělení	31

2.2.7	<i>Ovládací a signalizační prvky</i>	31
3.	TESTOVÁNÍ NA PROTOTYPU	33
3.1	SCHÉMA ZAPOJENÍ PROTOTYPU	33
3.2	PROGRAM.....	33
3.3	TESTOVÁNÍ.....	33
4.	NÁVRH PLOŠNÉHO SPOJE	35
5.	NÁVRH OBALU	36
6.	SOFTWAREVÁ ČÁST	38
6.1	OVLÁDÁNÍ.....	38
6.2	PROGRAM PRO MIKROKONTROLER	39
6.2.1	<i>Popis funkce programu</i>	39
6.2.2	<i>Knihovny</i>	40
7.	FINÁLNÍ TESTOVÁNÍ	41
7.1	JITTER TRIGGEROVACÍHO PULZU	41
7.1.1	<i>Postup měření</i>	41
7.1.2	<i>Měřené hodnoty</i>	42
7.2	ZPOŽDĚNÍ TRIGGEROVACÍHO PULZU.....	43
8.	ZÁVĚR	47
	LITERATURA	49
	SEZNAM PŘÍLOH	51

SEZNAM OBRÁZKŮ

1.1	Zapojení sběrnice CAN [3].	12
1.2	Stavy sběrnice CAN [5]	13
1.3	Standardní rámec [6].	14
1.4	Rozšířený rámec [6].	15
1.5	Porovnání CAN FD a CAN2.0 [8]	16
1.6	Segmenty bitové doby [10].	19
1.7	Vzorkování při zpoždění jednoho uzlu [10].	20
1.8	Fázová chyba rovna nule [10]	21
1.9	Fázová chyba větší než nula [10]	21
1.10	Fázová chyba menší než nula [10]	21
1.11	Příklad generování pulzu při nastavení triggerování SOF [11]	22
1.12	Příklad generování pulzu při nastavení triggerování EOF [11]	23
1.13	Ilustrační obrázek generování pulzu při nastavení triggerování podle ID s číslem 2.	23
2.1	Modul OLED displeje 128x32 s řadičem SSD1306 [15]	27
2.2	Blokové schéma zařízení.	27
3.1	Triggerování osciloskopu pulzem SOF	34
5.1	3D model obalu	37
7.1	Blokové schéma zapojení pro měření jitteru a zpoždění pulzu	42
7.2	Měření jitteru na osciloskopu	43
7.3	Zpráva zachycená pomocí SOF pulzu	44
7.4	Zpráva zachycená pomocí EOF pulzu	45
7.5	Doba resetování EOF pulzu	46
A.1	Schéma hlavní desky strana 1	53
A.2	Schéma hlavní desky strana 2	54
A.3	Schéma ovládací desky	55
B.4	Schéma testovacího prototypu	56
C.5	Testovací pracoviště	57
C.6	Detail testovacího zapojení	58
D.7	Hlavní deska vrstva Top	59
D.8	Hlavní deska vrstva Bottom	59
D.9	Kontrolní deska vrstva Top	60
D.10	Kontrolní deska vrstva Bottom	60
E.11	Testování finálního zařízení	61

SEZNAM TABULEK

1.1	Nastavované parametry	23
2.1	Příklady možností výběru CAN budiče.....	24
2.2	Příklady možností výběru CAN kontroléru.....	25
2.3	Popis potřebných pinů mikrokontroleru	26
2.4	Změřený odběr jednotlivých součástek	28
7.1	Měření jitteru pro nastavení mód triggerování SOF.....	42
7.2	Měření jitteru pro mód triggerování EOF a ID.....	43
7.3	Porovnání počtu přijatých zpráv.....	46

ÚVOD

CAN je velice rozšířená sériová sběrnice, používaná primárně v automobilovém průmyslu, ale také v průmyslové automatizaci. Všechny běžně používané osciloskopy mají možnost triggerování při náběžné, či sestupné hraně, při pulzu o určité délce apod., vyšší a dražší kategorie osciloskopů mají také možnost triggerování a dekódování určitých sběrnic, jako jsou například I²C, SPI atd., což umožňuje jednoduché ladění zařízení, používající právě tyto sběrnice. Většina těchto osciloskopů však nemá možnost triggerování sběrnice CAN.

Cílem této bakalářské práce je tedy navrhnout triggerovací jednotku pro sběrnici CAN, kterou bude možno připojit na vstup externího triggerování osciloskopu a zajistit tak zobrazení průběhů signálů na sběrnici, což zjednoduší ladění jiného zařízení, využívajícího sběrnici CAN, například při jeho vývoji, testování či opravě. Uživatel bude mít taktéž možnost nastavení triggerovací jednotky pro jeho požadavky (jako jsou rychlost přenosu dat, bod vzorkování, mód triggerování atd.).

Bakalářská práce se dá rozdělit na dvě hlavní části – teoretickou a praktickou. Teoretická část práce se nejprve zabývá popisem samotné sběrnice CAN, jejími principy fungování, strukturou zpráv, způsoby synchronizace jednotlivých zařízení připojených k jedné sběrnici, metodami vyhodnocování chyb, časováním atd., dále se v teoretické části pojednává o výběru stěžejních součástek a následném návrhu elektrického obvodu zařízení, výpočtu a výběru jednotlivých součástek.

Praktická část práce začíná testováním zjednodušeného prototypu zařízení na nepájivém poli, testování proběhlo v rámci semestrální práce, na kterou tato bakalářská práce navazuje. Následuje návrh plošného spoje a všech náležitostí nutných k jeho mechanické montáži a oddělení ovládacích prvků na samostatný plošný spoj. Po návrhu plošného spoje je představen návrh obalu zařízení, programování ovládání a komunikace mikrokontroleru s použitým displejem a CAN kontrolérem.

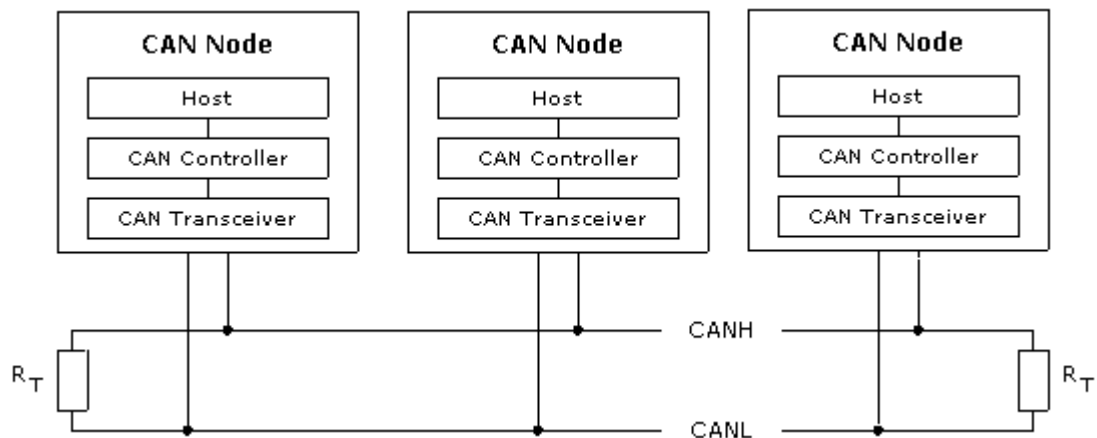
Poslední částí je finální testování a měření časových parametrů generovaného triggerovacího pulzu, při nastavení všech módů triggerování a všech možností bitrate. Kapitola ověřuje požadavky na minimální jitter a zpoždění triggerovacího pulzu.

1. SBĚRNICE CAN

Úvodní kapitola se zabývá teoretickým popisem sběrnice CAN a CAN parametry samotného zařízení.

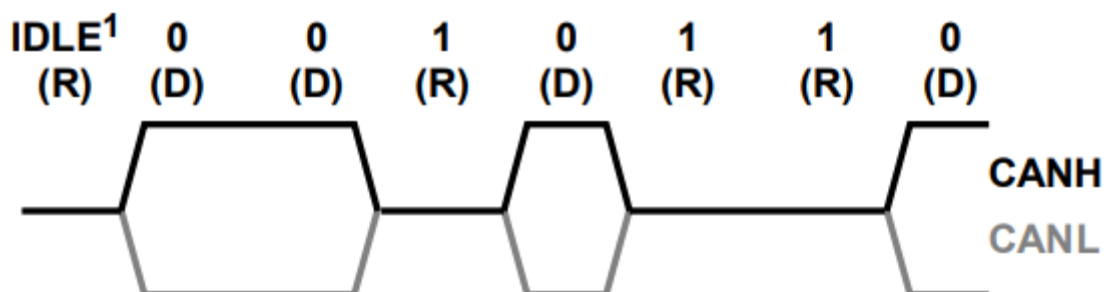
1.1 Základní popis

CAN (Control Area Network) je sběrnice představená společností Bosch v roce 1986, definována normou ISO 11898. Původně vyvinuta pro použití v automobilovém průmyslu, v této době se sběrnice CAN také hojně využívá v průmyslu a průmyslových aplikacích. Sběrnice CAN je odolná proti elektrickému rušení, je sama schopna diagnostikovat a opravit chyby a má jednoduché zapojení, je tedy vhodná pro aplikace požadující bezpečnost, rychlost a spolehlivost. [1]



Obrázek 1.1 Zapojení sběrnice CAN [3].

Všechna zařízení (uzly) komunikující přes sběrnici CAN jsou propojena jednou linkou, tvořenou jedním párem kroucených kabelů, označovaných jako CAN-High (CANH) a CAN-Low (CANL). Na obou koncích jsou kabely propojeny 120Ω rezistory. Sběrnice využívá kódování Non-return To Zero (NRZ) s Bit-stuffing (vkládání nevýznamných bitů), může být pouze ve dvou různých stavech, a to recesivní (logická 1) nebo dominantní (logická 0), tyto dva stavy jsou reprezentovány úrovněmi napětí na CANH a CANL, znázorněné na obrázku 2. Recesivní stav nastane, pokud je napěťová úroveň CANH i CANL stejná, tzn. jejich rozdíl je přibližně 0 V. Dominantní stav tedy nastane v případě, že se napětí na CANH zvýší a na CANL naopak sníží. Konkrétní napěťové úrovně záleží na použité fyzické vrstvě. [4]



Obrázek 1.2 Stavy sběrnice CAN [5]

1.2 Arbitráž

Sběrnice CAN je typu multi-master, tzn. pokud je sběrnice volná, tak každý uzel může poslat zprávu. V případě, že dva nebo více uzlů začne posílat zprávu současně, o jejich pořadí rozhodne nedestruktivní (uzel, který vyhraje arbitráž pokračuje v posílání zprávy a zpráva uzlu, který prohrál, se nezničí, ale odešle se následně po odeslání předchozí zprávy), bitová arbitráž. [1]

Priorita odeslání zprávy se rozhoduje podle čísla identifikátoru, čím je hodnota identifikátoru zprávy nižší, tím je jeho priorita vyšší. Identifikátor, který obsahuje pouze nuly, má nejvyšší prioritu zpráv, protože drží sběrnici nejdéle v dominantním stavu. Pokud tedy dva uzly začnou vysílat v jednu chvíli, první uzel, který odešle poslední bit identifikátoru logickou 0 (dominantní), mezitím co druhý uzel odešle logickou 1 (recesivní), tak prioritní zůstane první uzel a pokračuje v odesílání zprávy. Dominantní bit vždy přepisuje recesivní bit na sběrnici. Každý uzel, který odesílá zprávu, zároveň kontroluje, zda některý z dalších bitů nevyhrál arbitráž. [1][6]

1.3 Rámce

Struktura rámců (zpráv) je zásadní z důvodu dosažení robustní komunikace s detekcí chyb. Existují čtyři možné typy rámců, které mohou být odeslány po sběrnici CAN:

- Datový rámeček (Data frame)
- Vzdálený rámeček (Remote frame)
- Chybový rámeček (Error frame)
- Rámeček přetížení (Overload frame). [6]

Rozlišujeme dva typy rámců podle délky identifikátoru:

- Standardní rámeček (protokol CAN 2.0A) – obsahuje 11bitový identifikátor
- Rozšířený rámeček (protokol CAN 2.0B) – obsahuje 29bitový identifikátor. [1]

1.3.1 Datový rámeček

Datový rámeček je nejběžnější typ zprávy. Skládá se ze následujících částí:

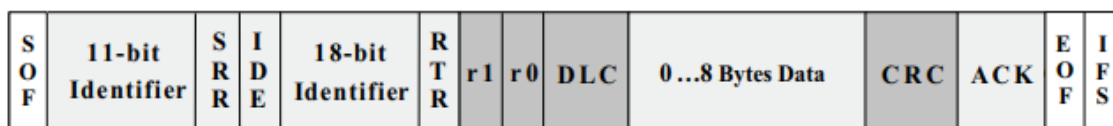
- Začátek rámečku (Start of frame – SOF) – jeden dominantní bit na začátku zprávy signalizuje začátek datového a vzdáleného rámečku, je také využíván k synchronizaci uzlů na sběrnici po nečinnosti.
- Arbitrážní pole (Arbitration field) – obsahuje identifikátor (11bitový, podle kterého se rozhoduje arbitráž) a RTR (Remote Transmission Request) bit, kde u datového rámečku je tento bit vždy dominantní.
- Řídicí pole (Control field) – obsahuje bit IDE (pokud je dominantní, tak označuje, že je vysílán standardní rámeček), bit r0 (rezervní bit pro případné využití v budoucnosti) a DLC (4bitová část, indikující počet bajtů posílaných v datovém poli).
- Datové pole (Data field) – obsahuje 0 až 8 bajtů dat.
- CRC pole (Cyclic redundancy check) – 15 bitů a 1 oddělovací bit (recesivní), je součástí detekce chyb. Obsahuje vypočítanou hodnotu z většiny rámečku.
- ACK pole (Acknowledgment) – obsahuje 1 oddělovací a 1 potvrzovací bit. Vysílač odesílající zprávu odešle 2 recesivní bity a přijímač, který tuto zprávu přijal, přepíše potvrzovací bit na dominantní a tím oznámí vysílači korektní přijetí zprávy.
- Konec rámečku (End of frame) – pole 7 recesivních bitů (tato sekvence na jiném místě ve zprávě nemůže nastat z důvodu „vkládání bitů“ (kapitola 1.4.4). [6]



Obrázek 1.3 Standardní rámeček [6].

Rozdíly u rozšířeného rámečku:

- SRR (Substitute remote request) bit – nahrazuje RTR bit, vždy recesivní.
- IDE bit – u rozšířeného formátu vždy recesivní. Indikuje, že následuje dalších 18 bitů identifikátoru.
- r1 bit – další rezervní bit, následuje po bitu RTR. [6]



Obrázek 1.4 Rozšířený rámeček [6].

1.3.2 Vzdálený rámeček

Účelem vzdáleného rámečku je žádat o přenos dat z jiného uzlu. Vzdálený rámeček je skoro stejný jako datový rámeček s důležitými rozdíly:

- RTR bit je vždy recesivní.
- Neobsahuje žádná data (na hodnotě DLC nezáleží). [6]

1.3.3 Chybový rámeček

Chybový rámeček obsahuje šest po sobě jdoucích bitů. Zda jsou tyto bity dominantní nebo recesivní záleží na stavu uzlu vysílajícího chybu. Každý uzel začne vysílat chybový rámeček ihned potom, co detekuje chybu a zapříčiní, že všechny ostatní uzly také detekují chybu a odešlou chybový rámeček, což má za následek superpozici více chybových rámečků. Každý uzel musí obsahovat dva chybové čítače, jeden pro odesílané chyby a jeden pro přijímané chyby. [5]

Chybový rámeček se skládá ze dvou částí:

- Chybový příznak – jsou možné 2 stavy:
 - Aktivní chybový příznak – obsahuje šest po sobě jdoucích dominantních bitů, jsou odeslány, pokud jeden z uzlů zaznamená chybu, což zapříčiní, že ostatní uzly detekují na sběrnici těchto šest dominantních bitů a také detekují chybu. Avšak aktivní chybový příznak může být dlouhý od šesti do dvanácti bitů z důvodu superpozice více chybových rámečků.
 - Pasivní chybový příznak – obsahuje šest po sobě jdoucích recesivních bitů, nastává v případě, že uzel detekuje chybu a zároveň je v pasivním stavu (z důvodu, že čítač překročil maximální hodnotu chyb pro přepnutí uzlu do pasivního stavu).
- Chybový oddělovač – obsahuje osm recesivních bitů. [1]

1.3.4 Rámec přetížení

Rámec přetížení se nevyužívá moc často, dnešní CAN kontroléry jsou dostatečně chytré, aby tento rámec nepoužívaly. Je podobný chybovému rámcu, ale je vysílán v případě, že je uzel příliš zaneprázdněný. [3]

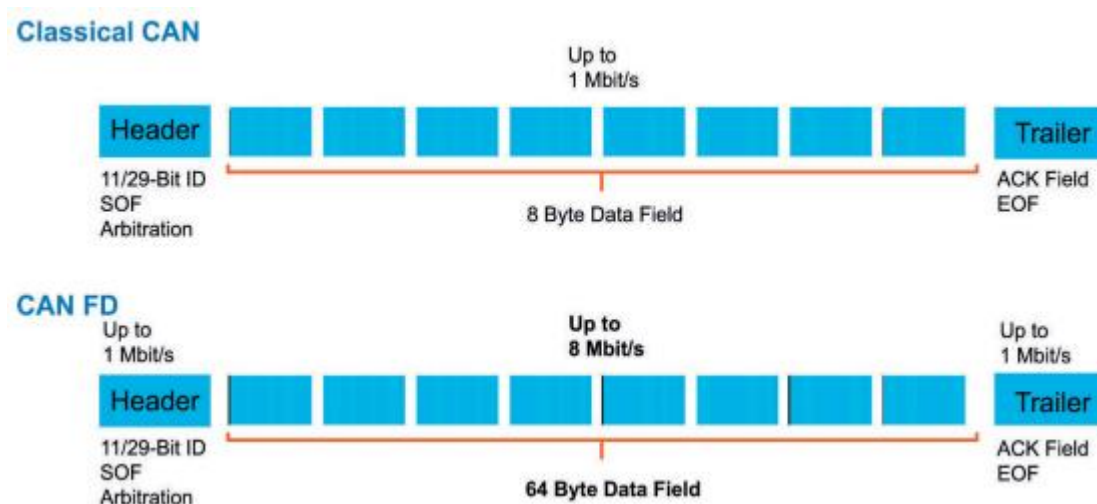
Rámec přetížení obsahuje dvě pole:

- Příznak přetížení – jsou možné dva stavy:
 - Uzel přijímající zprávu potřebuje zpoždění následujícího datového nebo vzdáleného rámce.
 - Detekce dominantního bitu během přerušování.
- Oddělovač přetížení – obsahuje osm recesivních bitů. [3]

1.3.5 CAN FD rámec

CAN FD (Flexible Data-Rate) je novější protokol, kompatibilní a podobný s CAN2.0. Byl vytvořen z důvodu přenosu více dat a zvýšení rychlosti jejich přenosu v jednom rámcu. [7]

Datový rámec CAN FD může být vysílán dvěma různými rychlostmi. Rychlost mimo datové pole je omezena na 1 Mbit/s a rychlost v rámci datového pole může být až 8 Mbit/s. Velikost datového pole je až 64 bajtů. [8]



Obrázek 1.5 Porovnání CAN FD a CAN2.0 [8]

1.3.6 CAN XL rámec

CAN XL (CAN Extra Long) je nejnovější protokol, kompatibilní s CAN FD i CAN 2.0. Je podobný CAN FD, s tím rozdílem, že rychlost datového pole může dosahovat až 10 Mbit/s a velikost dat je až 2048 bajtů. [9]

CAN XL protokol poskytuje také další možnosti konfigurace, týkající se vyšších vrstev ISO/OSI modelu, také umožňuje funkce jako vypnutí signalizování chyb a zapnutí PWM kódování místo NRZ. [9]

1.4 Vyhodnocování chyb

1.4.1 Čítače chyb

Každý uzel musí obsahovat dva čítače chyb, jeden pro přijímané chyby a druhý pro odesílané chyby. Čítače jsou inkrementovány na základě odeslání nebo přijetí chybového rámce. Dekrementace nastává v případě úspěšného odeslání nebo přijetí zprávy. [5]

1.4.2 Chybové stavy

Na základě čítačů mohou být uzly ve třech stavech:

- Aktivní chyba – v případě aktivní chyby může uzel komunikovat po sběrnici a posílat aktivní chybové příznaky. Čítač přijímaných i odesílaných chyb musí být na hodnotě menší než 127, aby mohl uzel zůstat v aktivním stavu.
- Pasivní chyba – v tomto případě uzel může komunikovat po sběrnici, ale může pouze odesílat pasivní chybové příznaky v případě chyby. Uzel se do tohoto stavu dostane v případě, že se alespoň jeden z čítačů dostane na hodnotu přes 127.
- Sběrnice vypnutá – v posledním případě má uzel zakázáno komunikovat po sběrnici, do tohoto stavu se uzel dostane, pokud se čítač odesílaných chyb dostane na hodnotu 256 nebo více. Poté, co bylo 128x detekováno jedenáct po sobě jdoucích recesivních bitů, mohou být čítače nastaveny na nulu a uzel se opět dostane do stavu aktivní chyby. [5]

1.4.3 Ověřování přeneseného bitu

Vysílající uzel monitoruje sběrnici ve chvíli, kdy vysílá, aby mohl ověřit, zda se každý bit objeví na sběrnici tak, jak má. Detekování špatného bitu se označuje jako bitová chyba, forma vysílací chyby. Výjimka platí pro pole identifikátoru, který se používá pro arbitráž a potvrzovací (ACK) bit, který vyžaduje přepsání recesivního bitu na dominantní. [5]

1.4.4 Vkládání bitů

Vždy, když jeden uzel odešle pět bitů stejné úrovně za sebou, dojde k tzv. vkládání bitů (bit stuffing), vložení doplňkového (nevýznamného) bitu za těchto pět stejných bitů. Tento proces se používá v celém datovém i vzdáleném rámci, s výjimkou bitových polí, které mají určenou sekvenci bitů, jako CRC pole, ACK pole, konec rámce, chybové rámce a rámce přetížení. Vkládání bitů zajišťují náběžné hrany pro probíhající synchronizaci

(kapitola 1.6.2), a také zajišťuje, že odesílané bity nejsou zaměněny za chybový rámeček nebo 7 rezesivních bitů signalizujících konec rámečku. Vložené bity jsou opět odstraněny přijímacím uzlem. [5]

1.4.5 Kontrola CRC

CRC je sekvence bitů vypočtených z první části (před datovým polem) rámečku. Odesílající uzel vypočítá tuto sekvenci dělením polynomu vytvořeného z dat a generátorového polynomu, zbytek po tomto dělení je zapsán do CRC pole. [5]

Přijímací uzel vydělí polynom vytvořený z dat a CRC sekvence s generátorovým polynomem. Pokud nenastaly žádné chyby, výsledek by měl být nula. [5]

1.4.6 Kontrola předepsaných datových polí

Některé bitové pole v rámci mají určené hodnoty, které musí vždy splňovat (CRC oddělovač, ACK pole, Konec rámečku). Přijímací uzel kontroluje, zda jsou tyto hodnoty správné. [5]

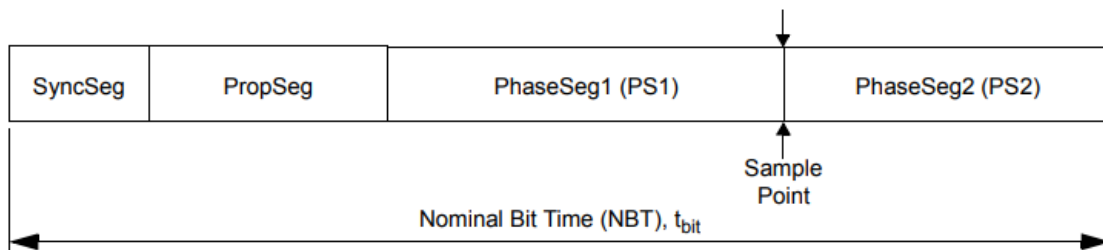
1.4.7 Potvrzení zprávy (ACK)

Vysílající uzel musí vždy detekovat dominantní ACK bit, který potvrzuje korektní přijetí zprávy od jiného uzlu [5].

1.5 Časování bitů

Bitová doba (Bit time) je doba, po kterou je jeden bit na sběrnici. Tato doba se skládá ze čtyř částí:

- Synchronizační segment (SyncSeg) – je první segment v bitové době, využíván k synchronizaci všech uzlů sběrnice. Očekávají se hrany bitů v tomto segmentu. Jeho délka je jedno časové kvantum (kapitola 0).
- Propagační segment (PropSeg) – následuje za synchronizačním segmentem, je zde z důvodu kompenzace fyzických zpoždění mezi uzly. Propagační zpoždění je definováno jako dvojnásobek součtu propagačních časů na sběrnici. Jeho délka může být od jednoho až osm časových kvant.
- Fázový segment 1 (PS1) a fázový segment 2 (PS2) – tyto dva segmenty kompenzují fázové chyby hran. PS1 může být prodloužen nebo PS2 může být zkrácen resynchronizací. Délka PS1 je jedna až osm časových kvant. PS2 je dlouhé dva až osm časových kvant. [10]



Obrázek 1.6 Segmenty bitové doby [10].

1.5.1 Časová kvanta

Každý segment bitové doby se skládá z celočíselných jednotek, tzv. časová kvanta (Time quantum – TQ). Délka časových kvant je určena z periody oscilátoru a je rovna dvojnásobku periody oscilátoru. [10]

1.5.2 Vzorkovací bod

Vzorkovací bod (Sample point) je bod v době bitu, ve kterém je hodnota bitu přečtena. Nachází se na konci fázového segmentu 1. Vzorkování může také být nakonfigurováno k vzorkování třikrát během jednoho bitu, v případě této konfigurace se jeden vzorek provádí opět na konci fázového segmentu 1, další dva se provedou v polovině časových kvant před koncem fázového segmentu 1 a výsledná hodnota je určena rozhodnutím většiny. [10]

1.5.3 Šířka synchronizačního skoku

Šířka synchronizačního skoku (synchronization jump width – SJW) nastavuje hodinový signál podle potřeby o jeden až čtyři časová kvanta, kvůli udržení synchronizace s odeslanou zprávou. [10]

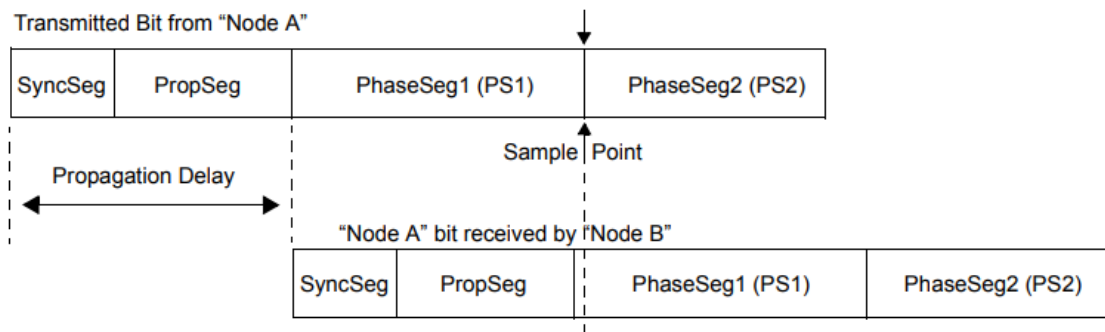
1.6 Synchronizace bitové doby

Všechny uzly na sběrnici musí mít stejnou jmenovitou přenosovou rychlost. Non-return to zero kódování nemá v sobě zároveň zakódovaný hodinový signál, ale přijímací uzly se potřebují synchronizovat s vysílanými daty, aby se zajistilo správné dekódování chyby. [10]

1.6.1 Zpoždění šíření bitů

Každý uzel, který je součástí arbitráže, musí vzorkovat každý bit ve stejné bitové době. Například pokud uzly na opačných koncích sběrnice začnou vysílat ve stejný čas, musí arbitráž rozhodnout o prioritě odesílání zprávy. Arbitráž je možná pouze pokud jsou oba uzly schopny vzorkovat ve stejnou bitovou dobu, z toho důvodu jsou délky sběrnice

omezeny při určitých rychlostech přenosu. [10]



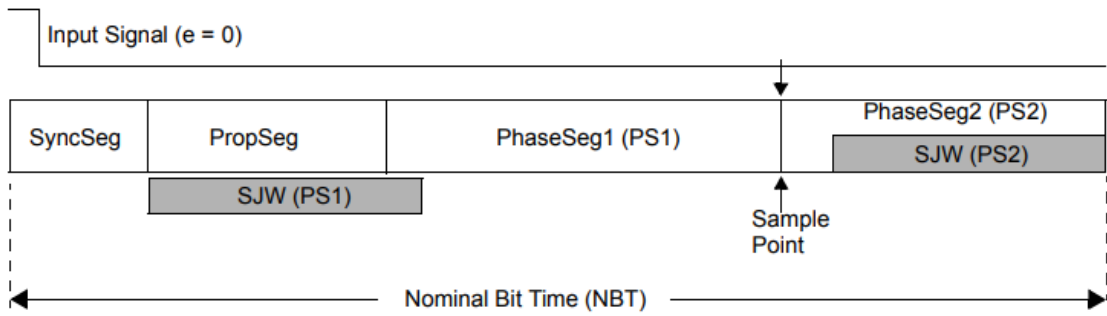
Obrázek 1.7 Vzorkování při zpoždění jednoho uzlu [10].

1.6.2 Synchronizace

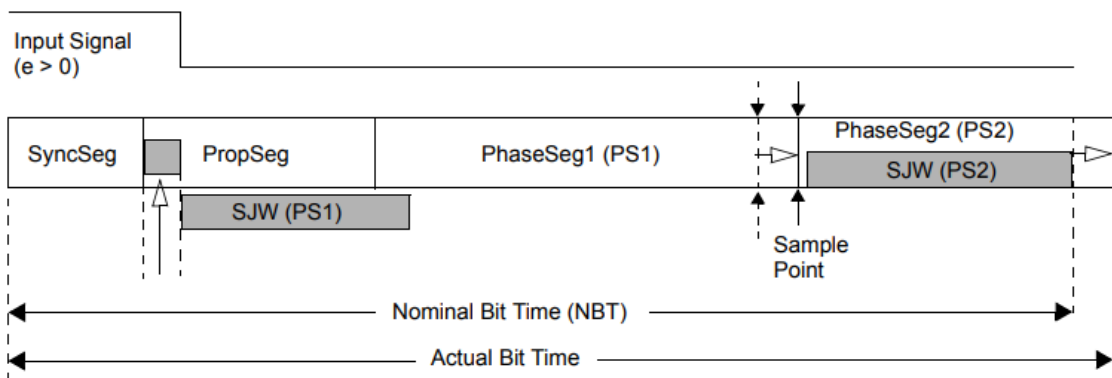
Existují dva druhy synchronizace:

- Tvrdá synchronizace (Hard synchronization) – nastává pouze jednou během každého rámce, při první hraně z recesivního na dominantní bit během nečinnosti sběrnice, která indikuje začátek rámce. Způsobuje, že se čítač časování bitů resetuje na synchronizační segment, poté se bude hrana nacházet v synchronizačním segmentu.
- Resynchronizace (Resynchronization) – provádí se pro zajištění udržení synchronizace provedené tvrdou synchronizací. Je dosažena využitím funkce Digital Phase Lock Loop (DPLL), která porovnává aktuální pozici hrany z recesivního na dominantní bit na sběrnici s pozicí očekávané hrany v synchronizačním segmentu. [10]

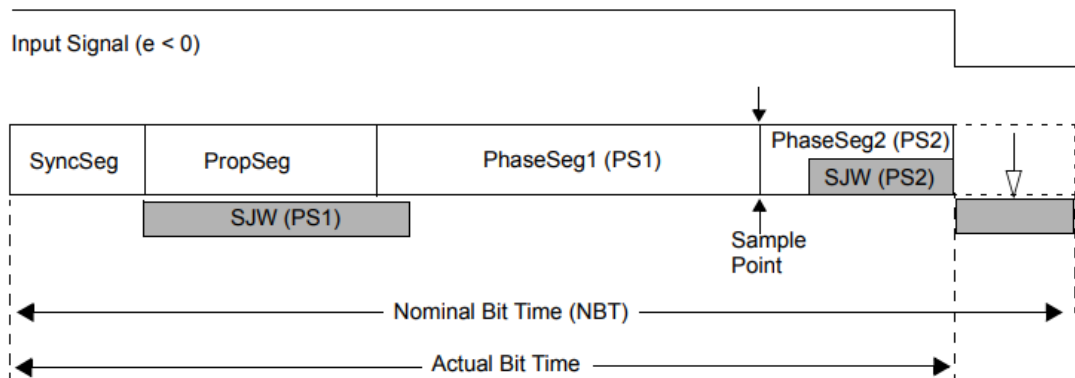
Fázová chyba bitu je dána posunem hrany vůči synchronizačnímu segmentu. Pokud je fázová chyba rovna nule, hrana leží v synchronizačním segmentu (obrázek 1.8). Pokud je fázová chyba větší než nula, hrana je posunuta za synchronizační segment a časová kvanta jsou přidána fázovému segmentu 1 (obrázek 1.9). Naopak, když je fázová chyba menší než nula, tak hrana leží ještě před synchronizačním segmentem, tzn. ve fázovém segmentu 2 předchozího bitu, časová kvanta jsou ubrána fázovému segmentu 2 (obrázek 1.10). [10]



Obrázek 1.8 Fázová chyba rovna nule [10]



Obrázek 1.9 Fázová chyba větší než nula [10]

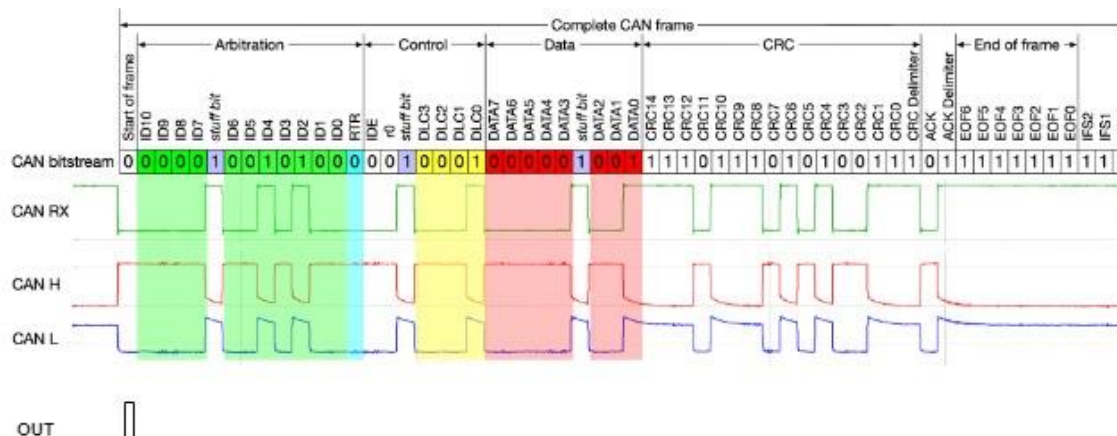


Obrázek 1.10 Fázová chyba menší než nula [10]

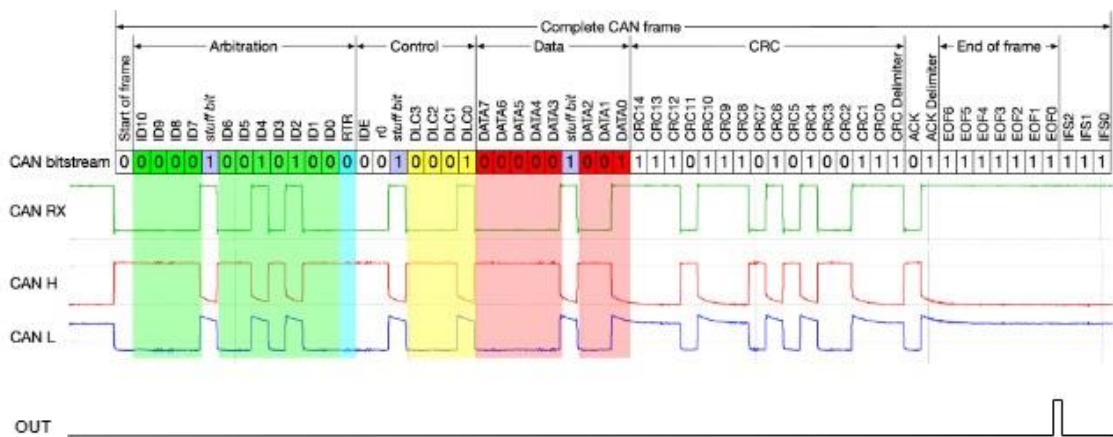
1.7 Parametry zařízení

Pro správné fungování zařízení je třeba nastavit jeho samotné parametry (uvedeny v tabulce 1.1). Bitrate se musí nastavit na stejnou rychlost jako ostatních zařízení připojených na sběrnici, typ rámce musí být taktéž stejný jako u ostatních uzlů sběrnice. Triggerovací mód a ID zprávy jsou parametry nastavované z důvodu správného generování požadovaného triggerovacího pulzu.

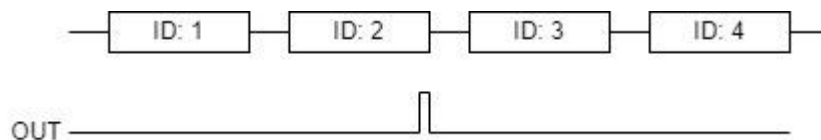
Triggerovací mód „SOF“ (Start of frame) zajistí generování triggerovacího pulzu při zachycení začátku jakéhokoliv rámce, příklad triggerování s převzatým průběhem je na obrázku 1.1. Při nastavení triggerovacího módu „EOF“ (End of frame) zařízení bude generovat pulz na konci jakéhokoliv přijatého rámce, příklad na obrázku 1.12. Nastavením možnosti triggerovacího módu podle konkrétního ID se otevře možnost dalšího nastavení, a to požadovaného ID rámce, při kterém se na konci rámce s nastaveným ID vygeneruje pulz. Zjednodušený obrázek 1.13 ukazuje příklad generování pulzu s nastaveným ID generování na 2.



Obrázek 1.11 Příklad generování pulzu při nastavení triggerování SOF [11]



Obrázek 1.12 Příklad generování pulzu při nastavení triggerování EOF [11]



Obrázek 1.13 Ilustrační obrázek generování pulzu při nastavení triggerování podle ID s číslem 2.

Tabulka 1.1 Nastavované parametry

Parametr	Možnosti
Bitrate [kbps]	250/500/800/1000
Sample point [%]	56,25/62,5/75/87,5
Triggerovací mód	SOF/EOF/dle ID
Typ rámce	standardní/rozšířený
ID zprávy	Konkrétní 11 nebo 29 bitů dlouhé ID

2. NÁVRH OBVODU ZAŘÍZENÍ

Tato kapitola se zabývá výběrem součástek a návrhem zapojení elektrického schématu jednotky.

2.1 Volba HW

Aby se mohlo pokročit k návrhu schématu a samotné realizaci zařízení, je třeba vybrat vhodné součástky pro zapojení. V této kapitole je popsán a zdůvodněn výběr hlavních součástek použitých v zapojení.

2.1.1 CAN budič fyzické vrstvy

CAN budič (přijímač/vysílač) zprostředkovává propojení mezi CAN kontrolérem a samotnou sběrnici (fyzickou vrstvou), převádí diferenční stavy sběrnice (dominantní a recesivní) na logické úrovni pro zpracování dále v kontroléru, a naopak z logických úrovní na stavy sběrnice CAN při odesílání zpráv. Na výběr samotného budiče existuje hned několik možností, viz tabulka 2.1.

Tabulka 2.1 Příklady možností výběru CAN budiče

Výrobce	Označení
Maxim Integrated	MAX13054
Microchip Technology	ATA6563
Microchip Technology	MCP2551
NXP	PCA82c250
NXP	TJA1050
Texas Instruments	SN65HVD23x
Texas Instruments	ISO1050DUB

Při výběru budiče hrála největší roli jejich dostupnost. Pro zapojení zařízení by bylo možné použít všechny uvedené budiče z tabulky 2.1, protože se v parametrech liší jen minimálně. U budiče SN65HVD23x je nevhodné napájecí napětí 3,3 V, které je rozdílné od napájecího napětí ostatních stěžejních částí obvodu a řešení obvodu by se zbytečně komplikovalo. MCP2551, jako jediný na trhu dostupný, by mohl být zvolen, ale nevyhovuje požadavku na galvanické oddělení sběrnice od zemí osciloskopu, a tedy je jako jediný možný pro řešení obvodu budič ISO1050DUB, dostupný pouze ze zásob.

ISO1050DUB je galvanicky izolovaný budič CAN, který splňuje specifikace normy ISO11898. Rozsah napájecího napětí umožňuje použití budiče s napájecím napětím 3,3 V i 5 V. [12]

2.1.2 CAN kontrolér

CAN kontrolér je část CAN uzlu mezi mikroprocesorem a budičem. Slouží k převádění dat přijatých z mikrokontroleru na datové rámce vhodné pro přenos po sběrnici CAN. Na výběr kontroléru je opět několik možností, viz tabulka 2.2.

Tabulka 2.2 Příklady možností výběru CAN kontroléru

Výrobce	Označení
Microchip Technology	MCP2510
Microchip Technology	MCP2515
Microchip Technology	MCP2517FD
Microchip Technology	MCP2518FD
Microchip Technology	MCP25625
Microchip Technology	MCP251863

Při výběru kontroléru stejně jako u budiče opět hrála nejvyšší roli dostupnost. První dva kontroléry v tabulce jsou velmi podobné, MCP2515 je pouze novější verzi MCP2510 s téměř stejnými elektrickými parametry, ale obsahuje některé vylepšení a přidání funkce. Bylo by také možné použít jeden z dalších dvou kontrolérů z tabulky 2.2 (MCP2517FD a MCP2518FD), které podporují i datové rámce FD (kapitola 1.3.5), a jednotka by mohla být rozšířena i o možnost triggerování FD rámců, ale nejsou dostupné na trhu.

Další možností by mohlo být využití posledních dvou kontrolérů z tabulky 2.2 (MCP25625 a MCP251863), jejichž velká výhodou je, že obsahují i budič v jednom integrovaném obvodu, díky čemuž je nižší cena a ušetření místa na výsledném plošném spoji. Z důvodu nedostupnosti většiny kontrolérů na trhu byl zvolen jako jediný možný MCP2515.

MCP2515 je samostatný CAN kontrolér schopný přijímat a vysílat standardní i rozšířené rámce zpráv. S mikrokontrolerem komunikuje pomocí rozhraní SPI. Jeho výhodou v této aplikaci je, že obsahuje výstupní pin CLKOUT/SOF (Start Of Frame), který při správném nastavení generuje pulz na začátku každého rámce zachyceného na vstupním pinu RXCAN, a je tedy možné jednoduché nastavení zařízení na triggerování při začátku rámce a připojení přímo na triggerovací vstup osciloskopu. Má možnost využití dvou přijímacích masek a šesti přijímacích filtrů, používané pro filtrování nevyžádaných zpráv. [12]

2.1.3 Mikrokontroler

Jako dostačující mikrokontroler byl zvolen ATmega8 od firmy Atmel [14]. Jedná se o 8bitový mikrokontroler s 8 kB programovou flash pamětí, 512 B EEPROM a 1 kB

SRAM. Jedním z požadavků na mikrokontroler je podpora SPI, pro možnost komunikace s CAN kontrolérem MCP2515, což ATmega8 splňuje. Výhodou vybraného mikrokontroleru je podpora I²C pro komunikaci s displejem (blíže popsány v kapitole 2.1.4).

Tabulka 2.3 Popis potřebných pinů mikrokontroleru

Pin	Typ	Input/Output	Popis
PD0	Digitální	Input	Ovládání pomocí tlačítek
PD1	Digitální	Input	
PD2	Digitální	Input	
PD3	Digitální	Input	
PD4	Digitální	Input	
PD5	Digitální	Output	Signalizační LED
PC0	Digitální	Input	Přerušování CAN kontroléru
PC2	Digitální	Output	Auto power off
SS	Digitální	Output	SPI komunikace s CAN kontrolérem a programátorem MCU
MOSI	Digitální	Output	
MISO	Digitální	Input	
SCK	Digitální	Output	
ADC3	Analogový	Input	Měření napětí baterie
SDA	Digitální	Input/Output	I ² C komunikace s displejem
SCL	Digitální	Output	

2.1.4 Periferie pro ovládání zařízení

Jednou z možností ovládání nastavení zařízení je použití DIP přepínačů, ale z důvodu možnosti nastavení triggerování dle určitého ID, by byl počet přepínačů příliš vysoký. Byla tedy zvolena možnost s displejem a ovládacími tlačítky. Displej sice ve finálním zapojení není požadován, ale dle mého uvážení se jedná o nejpřehlednější a nejjednodušší možnost nastavování parametrů z uživatelského hlediska. Konkrétně je použitý displej OLED s rozlišením 128x32 pixelů a řadičem SSD1306, komunikujícím pomocí sběrnice I²C.

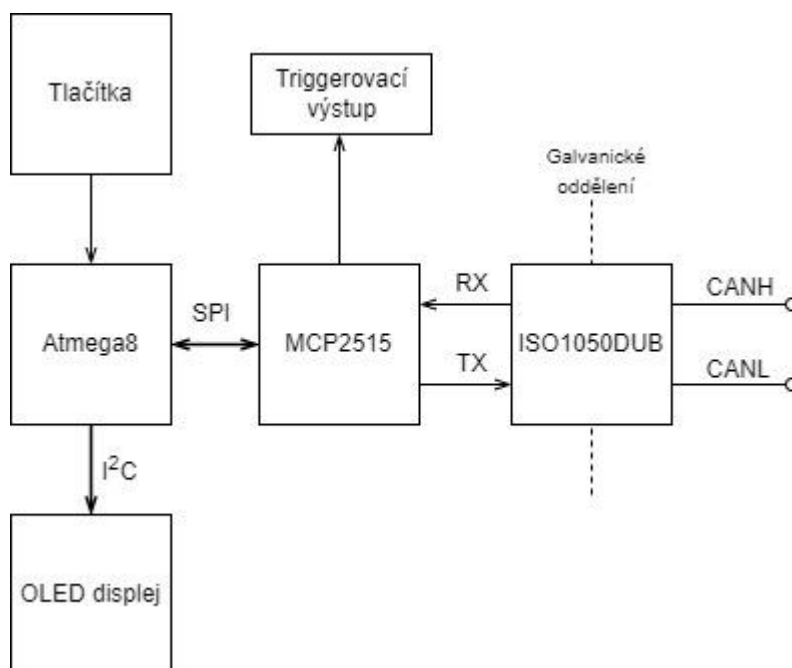


Obrázek 2.1 Modul OLED displeje 128x32 s řadičem SSD1306 [15]

2.2 Návrh schématu

2.2.1 Blokové schéma

Na obrázku 2.2 je blokové schéma, které zjednodušeně popisuje řešení výsledného zapojení zařízení.



Obrázek 2.2 Blokové schéma zařízení

2.2.2 Napájecí obvod

Dle zadání je požadováno bateriové napájení a zároveň možnost připojení i externího

napájení. Pro bateriové napájení je zvoleno použití dvou li-ion akumulátorů. Jmenovité napětí jednoho li-ion akumulátoru je 3,6 V, celkové napětí je tedy 7,2 V. Připojení externího napájení bude realizováno osazením napájecího konektoru na plošný spoj pro připojení externího zdroje se stejnosměrným výstupem.

Připojený napájecí zdroj musí mít aspoň 9 V na výstupu, aby při maximálním napětí akumulátorů a připojeném externím zdroji bylo na katodě diody D1 napětí větší alespoň o 0,7 V, a tedy dioda v uzavřeném stavu a zařízení se napájelo z externího zdroje. Maximální napětí externího zdroje je omezeno na 10 V pomocí transilu na vstupu.

Všechny části obvodu, které potřebují samostatné napájení, jsou zvoleny s napájecím napětím 5 V, které zajišťuje stabilizátor napětí LF50 s velmi malým úbytkem napětí.

Tabulka 2.4 ukazuje změřené změřený proudový odběr ($I = 67 \text{ mA}$) potřebný pro výpočet konkrétních hodnot součástek samo vypínacího obvodu, který je součástí navrženého zapojení.

Tabulka 2.4 Změřený odběr jednotlivých součástek

Součástka	Spotřeba [mA]	Poznámka
ATmega8	7	
MCP2515	3	
ISO1050DUB	2,3	Digitální strana
ISO1050DUB	19,5	Strana CAN (měřeno i s odběrem měniče)
Displej	5	Změřeno při textu přes celý displej
Signalizační LED	5	
Invertor	24	
OZ MCP602	0,4	
Celkový odběr	cca 66	

Hodnoty jsou vypočítané pro minimální možné napětí akumulátorů, tzn. 6,4 V. Dle katalogového listu PNP tranzistoru T1 [16] je jeho zesílení $h_{21E} = 60$. Napětí mezi bází a emitorem je typicky $U_{BE} = 0,7 \text{ V}$ a proud rezistorem R_9 je stejný jako proud bází T1 proto:

$$R_9 = \frac{U_{BE}}{I} = \frac{0,7}{\frac{66 \cdot 10^{-3}}{60}} = 636,4 \Omega. \quad (2.1)$$

Proud procházející rezistorem R_{10} je poté roven součtu proudu procházejícím rezistorem R_9 a bází tranzistoru T1 a hodnota R_{10} je tedy:

$$R_{10} = \frac{U_{BAT} - U_{D2} - U_{BE}}{I_{R9} + I_{B-T1}} = \frac{6,4 - 0,6 - 0,7}{\frac{66 \cdot 10^{-3}}{60} + \frac{66 \cdot 10^{-3}}{60}} = 2318,2 \Omega. \quad (2.2)$$

Dle katalogového listu NPN tranzistoru T2 [17] je zesílení $h_{21E} = 80$, proud do báze T2 musí být roven podílu proudu rezistorem R_{10} a jeho zesílení, proud rezistorem R_8 je stejný jako proud do báze T2, takže:

$$R_8 = \frac{U_{BE}}{\frac{I_{R10}}{h_{21E-T2}}} = \frac{0,7}{\frac{66 \cdot 10^{-3}}{60} \cdot 2} = 25\,454,5 \, \Omega. \quad (2.3)$$

Rezistorem R_7 tedy prochází proud roven součtu proudu rezistorem R_8 a proudem do báze T2, jeho hodnota je tedy:

$$R_7 = \frac{U_{Dig} - U_{D5} - U_{BE-T2}}{I_{R8} + I_{B-T2}} = \frac{5 - 0,6 - 0,7}{\frac{11}{400000} + \frac{11}{400000}} = 67\,272,7 \, \Omega. \quad (2.4)$$

kde: U_{Dig} ... digitální napětí výstupu mikrokontroleru [V]
Rezistor R_6 můžeme dopočítat jako dělič napětí z rovnice:

$$\frac{U_{BAT} - U_{D1}}{U_{Dig} - U_{D5}} = \frac{R_6 + R_7 + \frac{R_8}{2}}{R_7 + \frac{R_8}{2}}, \quad (2.5)$$

ze které dosazením a úpravou rovnice získáme:

$$R_6 = 25\,454,5 \, \Omega. \quad (2.6)$$

Konkrétní hodnoty ve schématu jsou zvoleny nejbližší hodnoty z řady k hodnotám vypočteným.

Rezistory R_7 a R_8 určují časovou konstantu vybíjení kondenzátoru C_{16} , a tedy odpojení obvodu od napájení po vypnutí digitálního napětí na diodě D_5 :

$$\tau_{OFF} = \left(R_7 + \frac{R_8}{2} \right) \cdot C_{16} = \left(67272,7 + \frac{25454,5}{2} \right) \cdot 220 \cdot 10^{-9} = 18 \, ms. \quad (2.7)$$

Doba pro odpojení obvodu je přibližně dvě časové konstanty τ , kdy napětí klesne a tranzistor T2 se vypne, tzn. cca 36 ms.

Podle Théveninovy věty určíme odpor definující časovou konstantu nabíjení kondenzátoru C_{16} . Věta nám říká, že při výpočtu můžeme nahradit zdroj napětí zkratem a zdroj proudu rozpojením, odpor definující nabíjecí konstantu kondenzátoru označený jako R_0 je tedy vypočten:

$$R_0 = R_6 \parallel \left(R_7 + \frac{R_8}{2} \right) = \frac{40228 \cdot \left(105714 + \frac{20000}{2} \right)}{40228 + \left(105714 + \frac{20000}{2} \right)} \quad (2.8)$$

$$= 19\,310,4 \, \Omega.$$

a tedy:

$$\tau_{ON} = R_0 \cdot C_{16} = 19\,310,4 \cdot 220 \cdot 10^{-9} = 4,2 \, ms. \quad (2.9)$$

Nabití kondenzátoru na napětí 4 V trvá přibližně jednu časovou konstantu τ (4,2 ms) při minimálním napětí akumulátorů ($U_{BAT} = 6,4 \, V$), to je doba nutná pro otevření tranzistoru T2, a tedy připojení napájecího napětí na obvod. Držení tlačítka pro spuštění však musí nutně být o něco delší, protože po připojení napájení na mikrokontroler trvá určitou dobu jeho spuštění.

2.2.3 Mikrokontroler

Mikrokontroler ATmega8 s napájecím napětím 5 V je opatřen třemi blokovacími kondenzátory s kapacitou 100 nF na všechny napájecí vstupy (V_{CC} je vstup pro napájení mikrokontroleru, AV_{CC} je vstup pro napájení A/D převodníku a A_{REF} je referenční napětí pro A/D převodník), které musí být na výsledném plošném spoji osazeny co nejbližně samotnému mikrokontroleru, kvůli filtraci šumu. Na pin RESET je připojen obvod s tlačítkem pro možnost resetování mikrokontroleru.

Nezbytnou součástí obvodu je taktéž možnost programování samotného mikrokontroleru. Programování probíhá pomocí programátoru pro mikrokontrolery z rodiny AVR, USBASP v2.0, proto jsou součástí obvodu také piny pro připojení programátoru.

2.2.4 Měření napětí akumulátoru

Protože je celý obvod napájen z li-ion akumulátorů, je důležité zajistit, že akumulátory nebudou podbity. Měření jejich napětí zprostředkovává napěťový sledovač připojený na vstup 10bitového A/D převodníku mikrokontroleru. Využití děliče napětí s operačním zesilovačem, jako napěťový sledovač, je zvoleno z důvodu jeho velmi vysokého vstupního odporu, a má tedy velmi malý vstupní proud, díky čemuž je docílena malá zátěž akumulátorů. Napětí akumulátorů musí být dělena dvěma, protože maximální možná výstupní hodnota A/D převodníku je rovna napětí na vstupním pinu referenčního napětí (A_{REF}), který je připojen na 5 V. Rozlišení A/D převodníku je:

$$ADC_{roz} = \frac{U_{MAX} - U_{MIN}}{2^n} = \frac{5 - 0}{2^{10}} = 4,88 \, mV, \quad (2.10)$$

n ... počet bitů A/D převodníku [-]

což je pro účely měření napětí na baterii naprosto dostačující.

2.2.5 CAN kontrolér

CAN kontrolér MCP2515 je taktéž jako mikrokontroler zaopatřen blokovacím kondenzátorem 100 nF na vstupu napájecího napětí a resetovacím obvodem na pinu RESET, který je nutný z důvodu softwarového resetu. Hodnota pull-up rezistoru R2 resetovacího obvodu je zvolena 10 k Ω , dle katalogového listu [13]. Hodnota kondenzátoru C8, který udrží napětí na RESET pinu po dostatečně dlouhou dobu, je zvolena podle většiny ostatních kondenzátorů v zapojení na 100 nF, pro udržení složitosti týkající se počtu různých součástek v zapojení co nejjednodušší. Dioda D3 zajišťuje rychlé vybití kondenzátoru v případě vypnutí napájení.

Krystal XTAL1 je externí oscilátor o frekvenci 16 MHz, připojený ke CAN kontroléru, generující stabilní a přesný signál.

Pin CLKOUT/SOF je jeden z triggerovacích výstupů, který při správném nastavení registrů generuje pulz na začátku přijetí rámce. SOF pin je připojen přes derivační RC článek na výstupní BNC konektor, který je možno propojit koaxiálním kabelem s osciloskopem.

2.2.6 CAN budič a galvanické oddělení

Jak je již v kapitole 2.1.1 popsáno, pro zapojení byl zvolen CAN budič ISO1050DUB, který jako jediný vyhovuje požadavku na galvanické oddělení. Pro plné galvanické oddělení je třeba zajistit i galvanické oddělení obou stran napájení budiče, k tomuto účelu slouží DC/DC měnič AM1/4S-0505 od firmy AIMTEC, který galvanicky oddělí 5V vstup s jeho 5V výstupem. Výstupní proud měniče je 50 mA, což pro napájení jedné strany budiče je postačující.

Budič je nutné vybavit možností připojení zakončovacího rezistoru (120 Ω , dle normy ISO 11898), to je zajištěno dvěma odpory 49,9 Ω , které sice v sériovém zapojení nesplňují konkrétní normovanou hodnotu, ale jsou zvoleny z důvodu udržení jednoduchosti schéma v počtu použití různých součástek. Mezi těmito odpory je jumper, který je možné v případě potřeby propojit, například zkratovací propojkou, a aktivovat tím zakončovací rezistor. Zakončovací rezistory zamezují odražení signálů odeslaných po sběrnici.

2.2.7 Ovládací a signalizační prvky

Jak je již v kapitole 2.1.4 popsáno, jako ovládací prvky zařízení jsou zvolena tlačítka s OLED displejem.

Zákmity tlačítek, ke kterým dochází při jejich stisku, jsou eliminovány pomocí jednoduchého RC obvodu.

Pro signalizaci zapnutí zařízení je v obvodu zapojena jedna červená LED, připojena přímo na 5V výstup stabilizátoru napětí a druhá červená LED na jeden z výstupů mikrokontroleru, která několikrát zabliká po jeho spuštění. Výpočet rezistoru omezující proud procházející LED:

$$R_{LED1} = \frac{U_{nap} - U_{LED}}{I_{LED}} = \frac{5 - 1,8}{0,005} = 640 \Omega, \quad (2.11)$$

kde: U_{nap} ... napájecí napětí [V]
 U_{LED} ... úbytek napětí na LED [V]
 I_{LED} ... proud procházející LED [A]

3. TESTOVÁNÍ NA PROTOTYPU

Před samotnou realizací plošného spoje je dobré vyzkoušet funkčnost zjednodušeného zapojení na kontaktním poli, tím se zabývá následující kapitola.

3.1 Schéma zapojení prototypu

Schéma zapojení prototypu na nepájivém poli se nachází v: Příloha B - Schéma zapojení prototypu. Jedná se o velice zjednodušené schéma pro testování na nepájivém kontaktním poli.

3.2 Program

Program testovaného prototypu je přiložen v samostatném souboru.

Základem komunikace mikrokontroleru ATmega8 a CAN kontroléru MCP2515 je rozhraní SPI, jehož funkce jsou implementovány v souborech SPI.c a SPI.h. Na těchto funkcích jsou pak dále postaveny jednotlivé instrukce pro ovládání MCP2515, jako instrukce pro resetování kontroléru, čtení registrů, zápis do registrů, čtení přijatých zpráv z bufferu, modifikace jednotlivých bitů v některých registrech, nastavení masek a filtrů atd. Popis ovládacích instrukcí je v katalogovém listu kontroléru [13], dle kterého byly implementovány. Potřebné instrukce se nachází v souborech MCP2515.h a MCP2515.c.

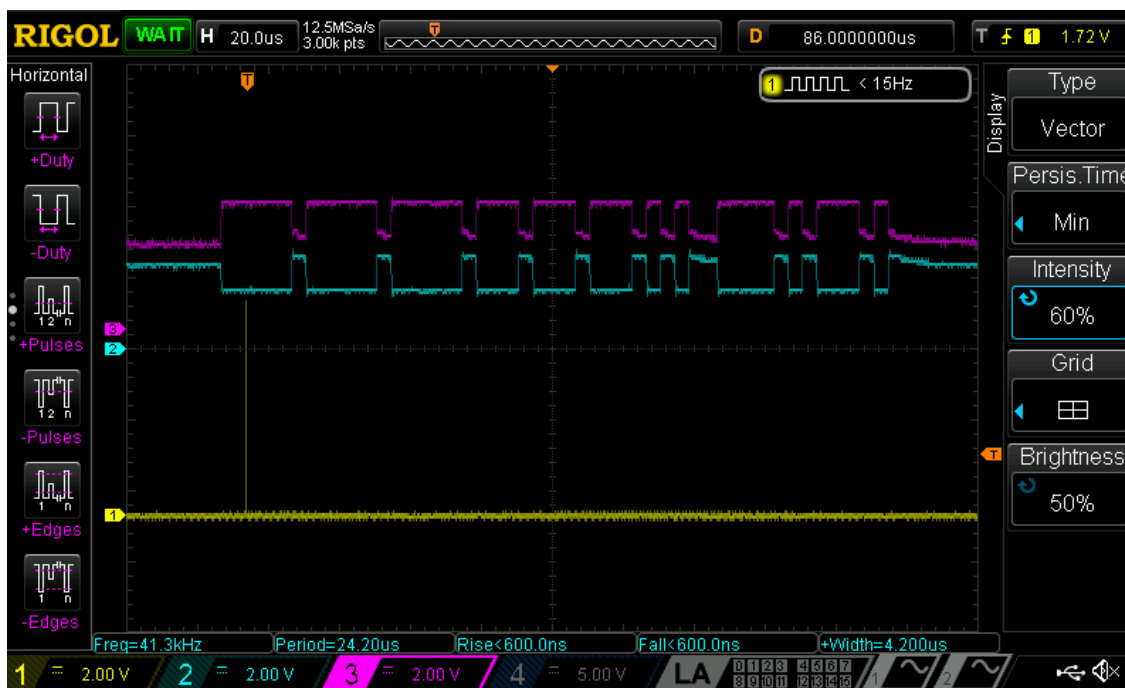
Funkce celého programu pro účely prototypu je velmi zjednodušená. V hlavním souboru programu main.c je pouze funkce, která nastaví potřebné parametry kontroléru, kde se jako první inicializuje samotné SPI, a poté se kontrolér resetuje, aby se dostal do základního nastavení a do konfiguračního režimu, ve kterém je možno přepisovat registry. V dalším kroku se přepíše kontrolní registr tak, aby byl aktivní výstupní pin CLKOUT/SOF, který může generovat hodinový signál nebo impulz na začátku rámce. Dále se nastaví konfigurační registry, aby zařízení fungovalo s požadovaným bitrate 250 kbps, vzorkovacím bodem v 87,5 % a výstupní pin CLKOUT/SOF generoval impulz se začátkem rámce. Nakonec se vynulují všechny masky a filtry pro příjem všech rámců a kontrolér se přepne do normálního operačního módu, ve kterém standardně přijímá zprávy.

3.3 Testování

Testování některých funkcí zařízení proběhlo s pomocí nepájivého kontaktního pole. Testovací zapojení obsahuje mikrokontroler ATmega8, programovaný pomocí USBasp programátoru a PC softwaru Microchip studio 7.0 v jazyce C, CAN kontrolér MCP2515 a budič ISO1050DUB. V rámci testovacího prototypu byla ověřena funkce výstupního pinu CLKOUT/SOF a jeho generování pulzu při zachycení rámce. Na obrázku 3.1 je vidět zobrazení standardního rámce na osciloskopu s nastavením triggerování při náběžné

hraně SOF pulzu, který je taktéž zobrazen na osciloskopu, při nastavení bitrate 250 kbps a sample pointu v 87,5 %.

Na obrázku 3.1 vidíme průběh rámce sběrnice a vygenerovaný triggerovací pulz, růžový průběh je rámec na CANH, modrý je rámec na CANL a žlutý je triggerovací impuls.



Obrázek 3.1 Triggerování osciloskopu pulzem SOF

Na obrázku C.5 v příloze je fotografie testovacího pracoviště a detailnější záběr testovacího zapojení je na obrázku C.6 v příloze.

4. NÁVRH PLOŠNÉHO SPOJE

Tato kapitola se zabývá návrhem a realizací plošného spoje zařízení.

Náhled plošného spoje je v příloze: Příloha D - Náhled plošných spojů. K návrhu plošného spoje je použit program Autodesk Eagle 9.6.2. Parametry plošného spoje jsou odvozeny od frézky, na které je vyroben:

1. Minimální šířka mezery – 0,3 mm.
2. Minimální vrták – 0,6 mm.
3. Další vrtáky odstupňované po 0,1 mm do 3 mm.

Výsledné zařízení se skládá ze dvou samostatných desek plošných spojů. Hlavní deska obsahuje většinu součástek a její výsledný rozměr je 73,5x87 mm se součástkami z vrchní i spodní strany. Zapojení je kombinované jak z dnes běžně používaných SMD součástek, tak i ze staršího typu THT součástek s kovovými vývody. SMD pouzdra běžných součástek, jako jsou rezistory a keramické kondenzátory, jsou zvolena většinou s velikostí 0805, z důvodu jejich jednoduchého ručního pájení. Mikrokontroler ATmega8 i CAN kontrolér MCP2515 jsou integrované obvody v pouzdře DIP, tedy typ THT a jsou zasazené do precizní patice, kterou je možno pájet z vrchní i spodní vrstvy. Konektory, pojistkový držák, DCDC měnič a elektrolytické kondenzátory jsou taktéž typu THT. Programovací piny jsou rozmístěny kolem mikrokontroleru tak, aby co nejméně komplikovaly samotný návrh plošného spoje a zabíraly co nejméně místa. Na hlavní desce je také konektor (piny 2x6), který umožňuje propojení s ovládací deskou zařízení.

Rozměr ovládací desky je 73,5x58 mm, rozměr je zvolen tak, aby její šířka byla stejná jako šířka hlavní desky, a bylo tak jednoduché jí mechanicky spojit s hlavní deskou, případně s krabičkou, za použití minima šroubů či distančních sloupků. Konektor pro spojení s hlavní deskou je umístěn tak, aby se zároveň přesně spasovaly díry pro mechanickou montáž desek. Ovládací deska je osamostatněna primárně pro zjednodušení přístupu k tlačítkům a displeji a možnosti jednoduššího zakomponování plošných spojů do krabičky.

Po celé ploše obou desek se nacházejí ručně pájené drátkové prokovy, sloužící jako propojení rozlité země na vrchní i spodní straně desek, některé však propojují i určité signály, které je potřebné přemostit na opačnou stranu plošného spoje. Obvod a konektor sloužící k připojení sběrnice CAN jsou galvanicky odděleny od zbytku celého obvodu.

Jak je již výše zmíněno, desky obsahují díry o průměru 3,2 mm pro M3 šrouby. Hlavní deska obsahuje šest děr, kde čtyři z nich slouží k mechanickému připojení ovládací desky a zároveň mechanickou montáž krabičky, a další dvě pouze pro montáž krabičky.

5. NÁVRH OBALU

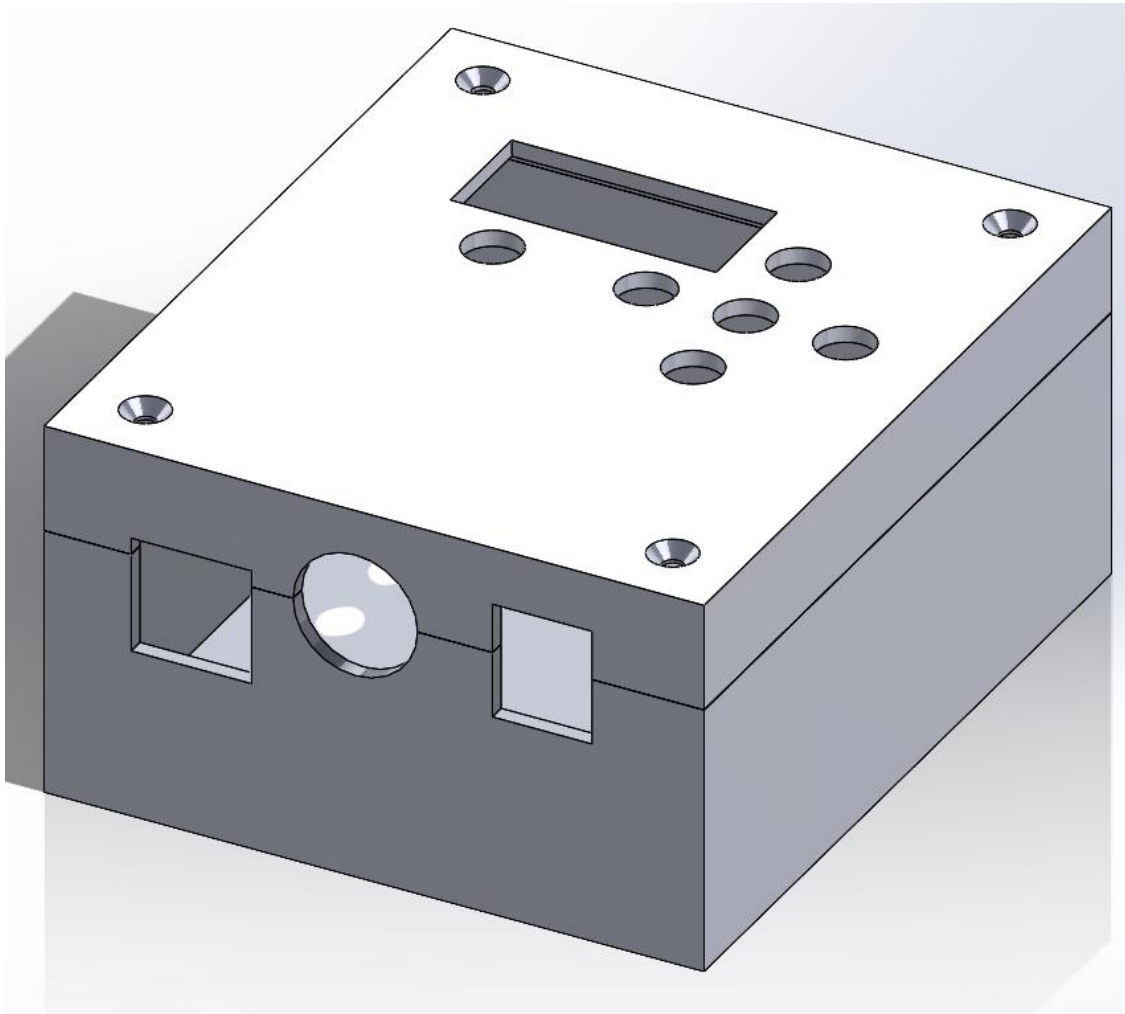
Jedním z bodů zadání práce je i návrh vhodného obalu pro zařízení, tím se zabývá následující kapitola.

Pro výrobu obalu zařízení bylo zvoleno vytisknutí na 3D tiskárně aditivní metodou FFF/FDM. 3D modely všech částí byly navrženy ve 3D CAD systému SolidWorks. Obal je navržený tak, aby plošný spoj do něho přesně pasoval s mezerou 1 mm mezi stěnou obalu a krajem plošného spoje na všechny strany. Sestava z vytvořených modelů je na obrázku 5.1.

Celý obal se skládá z několika částí, první ze dvou hlavních částí je spodní díl obalu s rozměry 93x79,5x32,5 mm. Přední strana tohoto dílu obsahuje tvarované výřezy pro možnost zasazení všech tří konektorů zařízení. Ze spodní strany jsou čtyři otvory, vzdálené od sebe stejně jako otvory na plošném spoji pro možnost mechanické montáže k plošným spojům a ostatním dílům obalu. Průměr těchto otvorů je 2,6 mm, díky zvolenému průměru a dostatečné měkkosti materiálu je možné šrouby se závitem M3 zašroubovat do těchto otvorů, čímž se v nich zároveň vytvoří závity, a je tak jednoduchá a pohodlná jejich montáž. Montážní otvory taktéž mají z vnější strany zkosené hrany tak, aby šrouby se zápustnou hlavou pasovaly do otvorů a byly tím zároveň zarovnané s plochou spodní strany obalu, docílí se tím i lepší estetický vzhled celého obalu. Ze zadní strany spodní části je i detail v podobě malého otvoru o průměru 1 mm pro možnost resetování mikrokontroleru stisknutím tlačítka tenkým nástrojem skrz tento otvor.

Druhou hlavní částí je vrchní díl obalu o rozměrech 93x79,5x13 mm. Rozměry šířky a délky této části jsou stejné jako u spodní části, aby na sebe díly přesně pasovaly, stejně tak otvory pro konektory mají rozměry stejné jako spodní díl, na vrchní části horního dílu jsou taktéž montážní otvory jako u spodního dílu pro pohodlnou montáž M3 šroubů se zápustnou hlavou. Důležitým prvkem této části obalu je obdélníkový otvor pro displej a otvory pro tlačítka kruhového tvaru. Pro tlačítka jsou také vyrobeny malé válečky, přilepené k vrškům tlačítek.

Součástí zařízení je i napájení z akumulátorů a je tedy nutné, aby obsahovalo držák na napájecí akumulátory. Samotný díl akumulátorového držáku je zakoupen již hotový k jeho upevnění, je vyroben díl se stejnými rozměry jako má hlavní deska plošného spoje (87x73,5 mm). Díl je možné opět pomocí šroubů či distančních sloupků se závitem M3 připevnit zesponu k plošnému spoji. Tento díl obsahuje dva otvory k upevnění zakoupeného akumulátorového držáku.



Obrázek 5.1 3D model obalu

6. SOFTWAREVÁ ČÁST

Softwarová část práce zajišťuje, aby uživatel mohl jednoduše a přehledně nastavit parametry zařízení pro triggerování připojené sběrnice, čímž se zabývá následující kapitola.

K návrhu programu pro mikrokontroler je použito vývojové prostředí Microchip studio, verze 7 od firmy Microchip Technology. Prostředí je určeno k vývoji aplikací pro mikrokontrolery z rodiny AVR a SAM. Spolu s vývojovým prostředím a programátorem USBASP v2.0 byl použit nástroj umožňující programování flash a EEPROM paměti mikrokontrolerů AVR zvaný AVRDUDE. Použitý programovací jazyk je C, stejně jako u programu pro prototyp zařízení (kapitola 3.2).

6.1 Ovládání

Základem celého ovládání je menu na displeji. Na prvním řádku se vždy zobrazuje název právě vybraného parametru, který uživatel nastavuje, a na druhém řádku jedna z možností pro daný parametr, kterou může uživatel potvrdit a nastavit tak zařízení s tímto parametrem nebo přepnout na další možnost. Konkrétní postup ovládání je popsán dále v této kapitole.

Po spuštění zařízení spínacím tlačítkem se na displeji zobrazí první položka v menu, a to nastavení bitrate, k dalším položkám menu může uživatel přejít pomocí tlačítek nahoru a dolů (jednotlivé položky a jejich možnosti jsou seřazeny dle tabulky 1.1.), položka nastavení konkrétního ID se zobrazí pouze v případě vybrání triggerování dle určitého ID a položka typ rámce pouze v případě vybrání triggerování dle ID nebo EOF. Při módu triggerování SOF je možné nastavit pouze bitrate a sample point. Konkrétní možnosti hodnot, u každé položky menu se vybírají tlačítka vlevo a vpravo. U možnosti nastavení požadovaného ID se hodnota inkrementuje a dekrementuje pomocí tlačítek vpravo a vlevo, ale protože standardní ID je (viz kapitola 1.3) dlouhé 11 bitů a rozšířené 29 bitů (tzn. maximální hodnota u standardního ID je $7FF_{\text{hex}}$ a u rozšířeného je $1FFFFFFF_{\text{hex}}$), nebylo by možné vždy dosáhnout požadovaného ID v přijatelném čase, proto je implementována možnost změny řádu hodnoty, přičítané nebo odčítané při změně ID. Změna přičítané hodnoty se provádí stiskem prostředního tlačítka zároveň s levým nebo pravým tlačítkem (stiskem prostředního zároveň s levým se zvětšuje a stiskem prostředního a pravého se zmenšuje). Přičítaná hodnota se zvětšuje a zmenšuje desetkrát a základem je číslo jedna, díky tomu je možné měnit vždy jen jednu libovolnou číslici celého ID. Pro potvrzení vybrané možnosti je nutné stisknout prostřední tlačítko, čímž se možnost uloží a okamžitě nastaví potřebné registry CAN kontroléru.

Zařízení má taktéž implementovány doplňkové funkce. První je možnost vypnutí zařízení stisknutím levého a pravého tlačítka současně po dobu tří sekund. Další funkce jsou spojené s měřením napětí napájecích akumulátorů, pokud při spuštění bude

naměřeno příliš nízké napětí na akumulátorech, na displej se vypíše zpráva „LOW BATTERY“ a uživateli nebude umožněno použití zařízení. Po krátké době se celé zařízení vypne. Měření napětí akumulátorů probíhá taktéž při běhu samotného programu, naměřením nízké hodnoty napětí na akumulátorech se v pravém horním rohu displeje zobrazí upozornění o vybité baterii.

6.2 Program pro mikrokontroler

Složka s celým projektem pro mikrokontroler je v příloze. Hlavním souborem programu je `main.c`, který obsahuje inicializaci všech potřebných komponent, jako je CAN kontrolér MCP2515, OLED displej, digitální vstupy a výstupy, AD převodník, a také povolení přerušování mikrokontroleru. Po inicializaci všech potřebných komponent následuje definice struktur pro jednotlivé položky menu, které se následně vypisují na displej. Mimo hlavního souboru jsou součástí projektu zmiňované knihovny, jako jsou `utils.h`, `MCP2515.h`, `lcd.h`, dále také knihovny starající se o samotnou komunikaci s CAN kontrolérem `SPI.h` a komunikaci s displejem `i2c.h`, nakonec také knihovna `macros.h`, obsahující jednoduchá makra pro nastavování vstupů a výstupů mikrokontroleru a knihovna `font.h` s fonty pro vypisování na displej. Tyto knihovny jsou dále detailněji popsány v kapitole 6.2.2.

6.2.1 Popis funkce programu

Hlavní částí celého programu je nekonečná smyčka, její většinou částí je příkaz `switch`, který rozhoduje na základě hodnoty vrácené funkcí `buttons_state`, vracející určitou celočíselnou hodnotu podle stisknutí ovládacích tlačítek. Každý `case` po stisknutí jednoho tlačítka obsahuje další příkaz `switch` rozhodující dále podle aktuálně zvoleného menu. Jak je popsáno v kapitole 6.1 tlačítka nahoru, dolů, doleva a doprava zajišťují pouze přecházení mezi samotnými položkami menu a změnou hodnoty proměnné „`select`“ ve struktuře dané položky. V případě stisknutí prostředního tlačítka se vybraná hodnota z proměnné „`select`“ vypsána na displeji запиše do proměnné „`current`“. Po zapsání proměnných následuje volání dvou funkcí `MCP2515_SetBtrRateAndSP` a `MCP2515_SetTrigger`, které, jak z jejich názvů vypovídá, nastavují požadované parametry CAN kontroléru MCP2515. Při volání těchto funkcí jsou za jejich parametry předány právě proměnné „`current`“ jednotlivých struktur.

Stejný `switch` obsahuje i implementaci pomocných funkcí zařízení při stisknutí více tlačítek, funkce vypínání zařízení a funkce nastavení konkrétní číslce ID, tyto funkce jsou popsány v kapitole 6.1.

Poslední funkcí volanou v rámci nekonečné smyčky je `MCP2515_IntReset`, jedná se o obsluhu přerušování od CAN kontroléru. Přerušování CAN kontroléru nastává v případě, že na sběrnici přijde zpráva, které se uloží v jeho bufferu a výstupní pin INT CAN kontroléru je nastaven na hodnotu logická 0, zároveň s ním se nastaví příslušný bit v registru na hodnotu logická 1, indikující důvod vzniku přerušování (v případě této práce nastává

přerušeni pouze při naplněním bufferu zprávou). Mikrokontroler po indikování logické 0 nastaví zmíněný bit registru v CAN kontroléru opět na hodnotu logická 0, a tím opět resetuje jeho přerušeni a je znovu možné přijmout zprávu do bufferu.

6.2.2 Knihovny

Program využívá hned několik různých knihoven, jako jsou standardní knihovny jazyka C, knihovny starající se o komunikaci s periferiemi (I²C pro displej, SPI pro CAN kontrolér), knihovnu s funkcemi pro ovládání displeje, knihovnu funkcí pro CAN kontrolér a knihovnu s pomocnými funkcemi, jako je inicializace vstupní/výstupních pinů mikrokontroleru. Základní knihovnou ovládání displeje je lcd.h, spolu s knihovnami font.h a i2c.h převzaté z [18]. Knihovna konkrétněji slouží k ovládání driveru SSD1306, umožňující zobrazování textu a grafických prvků jako jsou úsečky, obdélníky a kružnice. Zmíněné grafické prvky však v této práci nejsou použity. Ke komunikaci s driverem tato knihovna pak používá funkce z knihovny i2c.h, k hardwarovému I²C, které podporuje samotný mikrokontroler. Soubor font.h obsahuje všechny textové znaky, které je možno vypisovat na displej, knihovna umožňuje i přidání vlastních uživatelsky definovaných speciálních znaků.

V knihovně MCP2515.h jsou implementovány funkce potřebné pro ovládání CAN kontroléru v tomto zařízení. Jednoduché funkce MCP2515_Reset, MCP2515_Read, MCP2515_Write a MCP2515_BitModify jsou implementovány podle katalogového listu výrobce [13], kde jsou detailně popsány. Ostatní funkce této knihovny jsou vytvořeny pro účely této bakalářské práce. Ke komunikaci mikrokontroleru a CAN kontroléru je třeba obstarat komunikaci pomocí rozhraní SPI, o to se stará knihovna SPI.h, obsahující funkce inicializace SPI, přijetí zprávy, odeslání zprávy a funkci nastavující pin CS (chip select), který v SPI komunikaci slouží pro adresování, v tomto případě pouze s jedním podřízeným (slave) zařízením.

Poslední knihovnou je utils.h, obsahuje funkci pro inicializaci vstupních a výstupních pinů mikrokontroleru, jednoduchou funkci pro zablikání LED indikující funkčnost samotného zařízení i programu. Funkce buttons_state, která jak je již výše zmíněno, vrací hodnotu podle stisknutých tlačítek. Update_display je funkcí, která udává, jak jsou informace vypisovány na displeji, využívá funkce sprintf ze standardní knihovny jazyka C pro formátování stringů vypisovaných na displeji. Poslední funkcí této knihovny je timer_init, které inicializuje časovač, díky kterému každou sekundu inkrementuje hodnotu globální proměnné „counter“, po jedné minutě se hodnota dostane na 60 a proběhne změření napětí akumulátorů a proměnná se opět vynuluje, díky tomu měření napájecích akumulátorů proběhne vždy pouze při zapnutí zařízení a následně každou minutu, což je pro účel této práce dostačující.

7. FINÁLNÍ TESTOVÁNÍ

Posledním bodem zadání je měření skutečného jitteru triggerovacího výstupu a otestování zařízení, tím se zabývá tato poslední kapitola.

7.1 Jitter triggerovacího pulzu

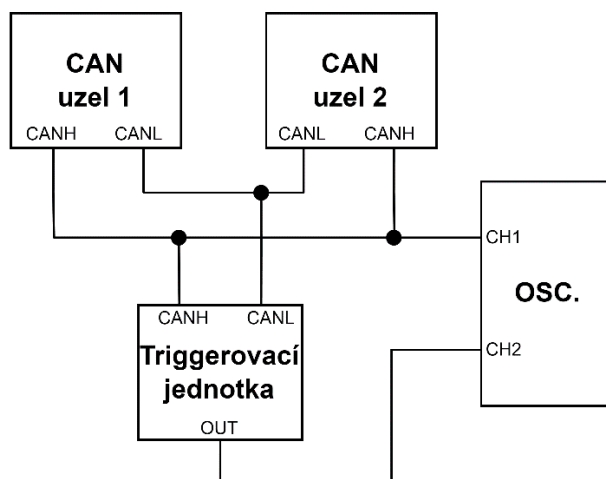
Pojem jitter označuje odchylku v časování generovaného impulzu. Konkrétní hodnota jitteru značí časový rozsah, ve kterém se s jistotou vygeneruje triggerovací impulz po přijetí zprávy na sběrnici.

Pro měření jitteru triggerovacího pulzu je použit čtyř kanálový osciloskop RIGOL MSO1074Z. Blokové schéma zapojení měření je na obrázku 7.1. Jako uzly jedna a dva slouží Raspberry Pi s přídatnou deskou obsahující dva uzly CAN s možností přijímání a odesílání zpráv na sběrnici. K jejich ovládání jsou použity příkazy v terminálu operačního systému Linux na samotném Raspberry Pi.

7.1.1 Postup měření

Jak je již výše zmíněno, pro měření je použit osciloskop RIGOL MSO1074Z, dle blokového schéma na obrázku 7.1 je na kanál číslo 1 připojena jedna z linek sběrnice CAN a na kanál číslo 2 triggerovací výstup, triggerování osciloskopu je nastaveno na náběžnou hranu kanálu 2. Následně je nutné změnit nastavení displeje osciloskopu „Persistence Time“ na nekonečno, což umožní zobrazení všech předchozích průběhů i po příchodu nového pulzu triggerovacího výstupu.

Po zapojení a přípravě měření se na sběrnici začnou posílat stejné zprávy se stejným ID ve velkém množství (například jedna zpráva každých 50 ms) po určitou dobu, tím se docílí „rozšíření“ hran CAN zpráv, způsobené právě nastavením „Persistence Time“ displeje osciloskopu. Pomocí kurzorů osciloskopu se poté změří šířka jedné z těchto „hran“, tato šířka odpovídá jitteru, triggerovacího pulzu. Na obrázku 7.2 je ukázka největšího naměřeného jitteru, blíže popsáno v kapitole 7.1.2.



Obrázek 7.1 Blokové schéma zapojení pro měření jitteru a zpoždění pulzu

7.1.2 Měřené hodnoty

V tabulce číslo 7.1 jsou změřené hodnoty jitteru pro všechny možnosti bitrate s módem triggerování na začátku přijaté zprávy. Jak je z tabulky patrné, největší jitter signálu je zaznamenán s nastavením bitrate na 250 kbps, proto bylo měření s tímto nastavením provedeno vícekrát. Hodnota jitter se průměrně pohybuje okolo hodnoty 165 ns. Při nastavení bitrate 250 kbps je doba jednoho bitu zprávy 4 μ s, zařízení tedy s nastavením triggerování SOF splňuje požadavek na co nejmenší jitter a hodnota je téměř zanedbatelná.

Tabulka 7.1 Měření jitteru pro nastavení mód triggerování SOF

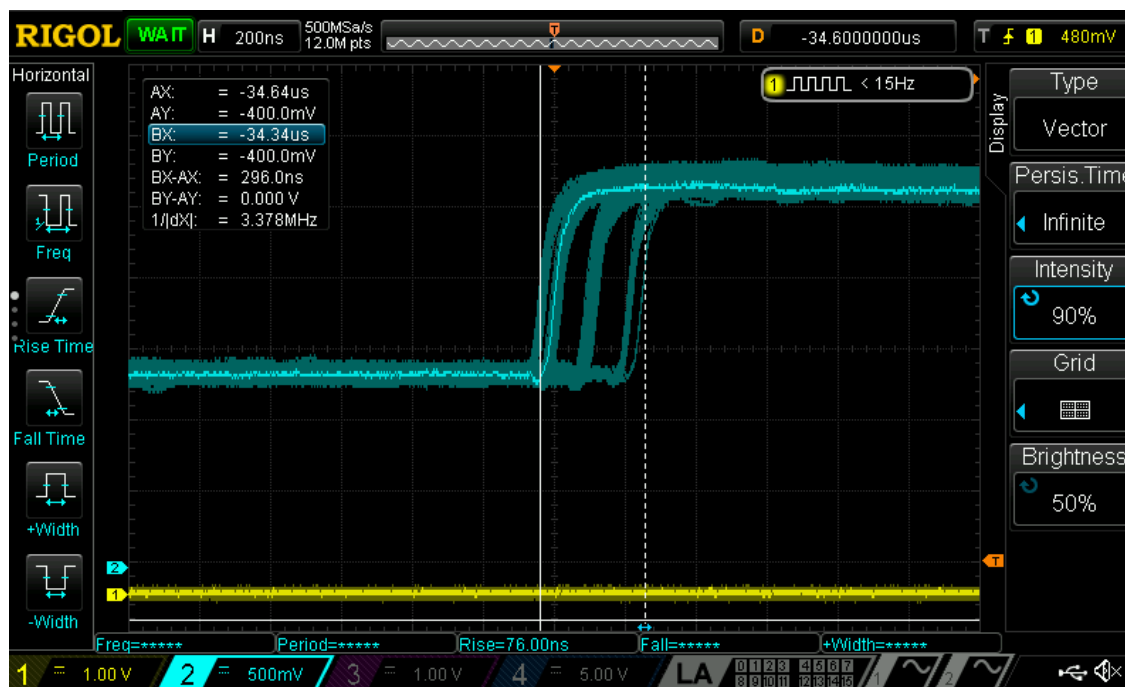
Bitrate [kbps]	1 000	800	500	250	250	250	250	250
Jitter [ns]	67	74	65	166	162	163	169	166

Tabulka číslo 7.2 ukazuje hodnoty jitteru naměřené pro všechny možnosti nastavení bitrate s módem generování triggerovacího pulzu na konci přijaté zprávy. Tabulka hodnot jitteru odpovídá módu jak pro triggerování všech zpráv na jejich konci, tak i triggerování zpráv s konkrétním ID, dle provedeného měření pro oba módy jsou jejich výsledky stejné.

Pro bitrate 250 kbps je naměřený jitter opět největší ze všech provedených měření. Jeho hodnota se průměrně pohybuje okolo 250 ns, což je opět velmi malá až zanedbatelná hodnota. Obrázek 7.2 ukazuje měření času největšího jitteru na osciloskopu s pomocí kurzorů.

Tabulka 7.2 Měření jitteru pro mód triggerování EOF a ID

Bitrate [kbps]	1 000	800	500	250	250	250	250	250
Jitter [ns]	150	151	93	202	290	212	296	252



Obrázek 7.2 Měření jitteru na osciloskopu

7.2 Zpoždění triggerovacího pulzu

Důležitým parametrem triggerovacího pulzu je i jeho zpoždění vůči samotné zprávě. U tohoto zařízení je zpoždění pulzu minimalizováno už jenom tím, že je generován hardwarově, a to samotným CAN kontrolérem.

Jak je z obrázku číslo 7.3 patrné, u pulzu SOF generovaného na začátku zprávy není žádné zpoždění. Dle měření je jasné, že SOF pulz je vygenerovaný zároveň s prvním bitem zprávy, konkrétně ve chvíli vzorkování, což odpovídá i popisu CAN kontroléru v katalogovém listu výrobce [13].



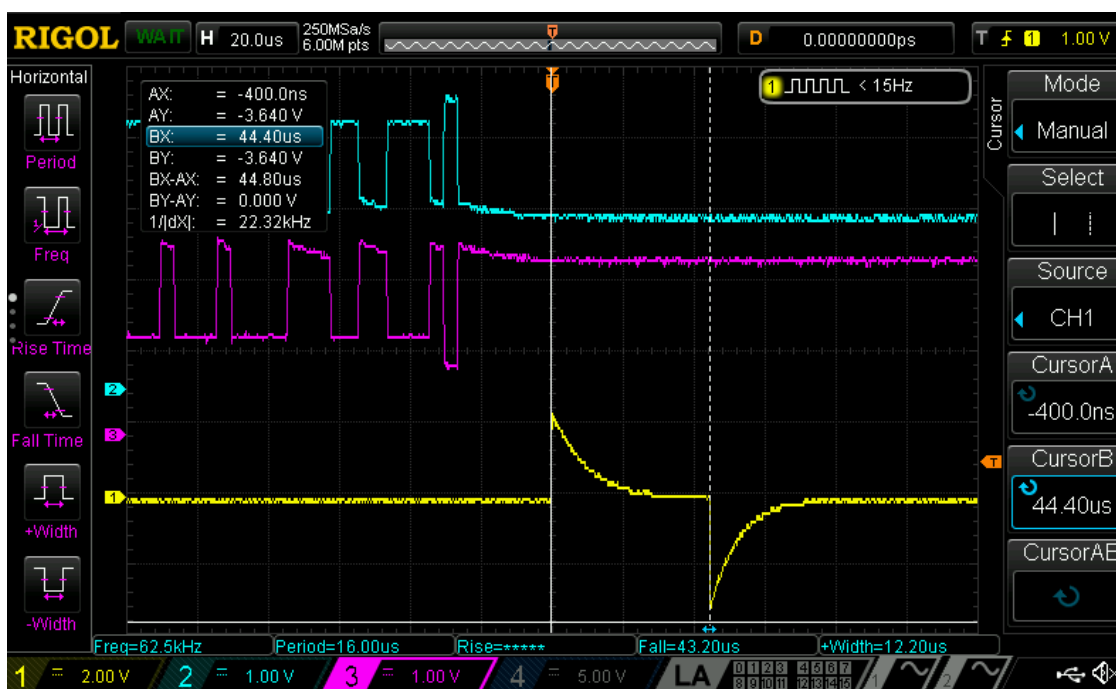
Obrázek 7.3 Zpráva zachycená pomocí SOF pulzu

Obrázek 7.4 ukazuje, že EOF pulz generovaný na konci zprávy nemá prakticky žádné zpoždění, i když to na první pohled není jasné. Poslední dominantní bit, který je na snímku zprávy, je bit ACK, za ním následuje 7 recesivních bitů indikujících konec zprávy (viz kapitola 1.3.1), tzn. že začátek EOF pulzu je vygenerovaný ve chvíli vzorkování posledního EOF bitu.



Obrázek 7.4 Zpráva zachycená pomocí EOF pulzu

U EOF pulzu nastává problém s tím, že je třeba ho resetovat pomocí mikrokontroleru, což trvá příliš dlouhou dobu (cca 45 μ s, obrázek 7.5.), princip resetování je popsán v kapitole 6.2.1. Minimální doba mezi koncem zprávy a začátkem další (Interframe space – IFS) je dle katalogového listu výrobce [13] 3 recesivní bity (tzn. u bitrate 250 kbps je 12 μ s). S délkou EOF pulzu 45 μ s tedy není možné zachytit, při plném zatížení sběrnice a vysokém bitrate, jednu následující zprávu po vygenerování triggerovacího pulzu. U vyšších bitrate je délka IFS kratší, tzn. s žádným nastavením není možné zachytit následující zprávu po vygenerování pulzu.



Obrázek 7.5 Doba resetování EOF pulzu

Tabulka č. 7.3 ukazuje porovnání počtu zpráv (pro jednotlivé bitrate) za sekundu, se standardním ID a délkou dat pouze jeden byte, který je možno zachytit při plném zatížení sběrnice s délkou IFS udávanou výrobcem a počet zpráv o stejné délce s potřebnou délkou IFS určenou délkou triggerovacího pulzu. Potřebná délka IFS je udávaná jako počet bitů potřebných mezi zprávami.

Tabulka 7.3 Porovnání počtu přijatých zpráv

Bitrate [kbps]	Počet zpráv/s (IFS = 3)	Potřebná délka IFS	Počet zpráv/s
1000	18182	12	3906
800	14545	23	3333
500	9091	36	2841
250	4545	45	2577

8. ZÁVĚR

Cílem této bakalářské práce byl návrh, realizace a ověření funkčnosti triggerovací jednotky pro sběrnici CAN, která je schopna generovat triggerovací pulz připojený na osciloskopu pro jednoduché ladění zařízení komunikující po sběrnici CAN i s osciloskopem, který nepodporuje synchronizaci se sběrnici CAN.

Úvodní část je zaměřena na teoretický úvod k samotné sběrnici CAN. Kapitola se zabývá principy fungování jako jsou arbitráž, časování bitů, vyhodnocování chyb a synchronizace uzlů připojených ke sběrnici. Dále je zde popsána struktura jednotlivých rámců (zpráv), které je možno odeslat a zachytit na sběrnici CAN.

Druhá kapitola se nejprve zabývá výběrem stěžejních součástek jako je mikrokontroler, CAN budič fyzické vrstvy, CAN kontrolér a periferie pro ovládání zařízení. Po výběru hlavních součástek obvodu následuje návrh konkrétních částí obvodu. Samotný obvod starající se o příjem zpráv a triggerování pulzu je doplněn o další části jako je obvod měřící napájecí akumulátory a obvod umožňující zapnutí a následně softwarově řešené vypnutí obvodu mikrokontrolerem.

Od třetí kapitoly začíná praktická část práce. Třetí kapitola popisuje návrh plošného spoje a všechny detaily nutné k mechanické montáži a oddělení ovládacích prvků na samostatný plošný spoj. Po návrhu plošného spoje následuje další kapitola o návrhu obalu pro zařízení. V šesté kapitole této práce je popis ovládání a popis programu pro mikrokontroler včetně použitých knihoven.

Ve finální fázi programování nastal problém s nedostatkem paměti mikrokontroleru pro potřebný program. Vybraný mikroprocesor ATmega8 obsahuje 8 kB programové paměti, samotný program pak má o téměř 1,8 kB více. Důvodů velikosti finálního programu je několik, jedním z nich je využívání funkcí ze standardní knihovny jazyka C, například funkce `sprintf` pro zápis stringů, nebo explicitní přetypování z datového typu `int` na datový typ `float` pro výpočet měřeného napětí akumulátorů. Protože je rozdíl velikosti programu a velikosti programové paměti mikrokontroleru dosti veliký, pravděpodobně již nebude možné program dostatečně optimalizovat pod velikost 8 kB a zachovat všechny naprogramované části. Pro měření a testování finálního zařízení byl použit velmi podobný mikroprocesor ATmega328P. Náhradní mikroprocesor se s vybraným shoduje v počtu i pořadí pinů a registry potřebné pro účely této práce jsou u obou mikroprocesorů shodné, důležitým rozdílem je pouze velikost programové paměti, kterou náhradní mikroprocesor má 32 kB, což je výrazně více než, původně vybraný mikroprocesor ATmega8.

Poslední kapitola se zabývá finálním testováním funkčnosti a měřením časových parametrů triggerovacího výstupu. Měření jitteru bylo provedeno pro všechny módy triggerování a pro všechny možnosti nastavení bitrate. Nejvyšší hodnota jitteru byla naměřena pro triggerovací mód EOF nebo ID, u obou jsou výsledky shodné, a to průměrně 250 ns, jak je v kapitole 7.1.2 popsáno, tato hodnota je velmi malá a splňuje

požadavek ze zadání na co nejmenší jitter. Dle měření není téměř žádné zpoždění příchodu triggerovacího pulzu u SOF módu ani u EOF a ID módu. Problém však nastává s resetováním triggerovacího pulzu v módu EOF a ID. Délka pulzu je přibližně 45 μ s, což při plném zatížení sběrnice znamená, že zařízení nezachytí jednu následující zprávu po vygenerování triggerovacího pulzu. Tento problém je primárně způsoben rychlostí přenosu rozhraní SPI, které je nastaveno na jednu polovinu hodinového signálu mikrokontroleru (mikrokontroler využívá vnitřní RC oscilátor o frekvenci 8 MHz), tedy 4 MHz. Pro resetování je nutné odeslat jedenáct 8bitových datových rámců přes SPI. Použitím externího 16 MHz krystalu místo interního RC oscilátoru mikrokontroleru by se dalo docílit lepších výsledků, zkrátit resetování pulzu na minimální dobu a umožnit triggerování více po sobě jdoucích zpráv při plném zatížení sběrnice.

LITERATURA

- [1] CAN Specification: Version 2.0 [online]. 1991, [cit. 2022-10-10]. Dostupné z: <http://esd.cs.ucr.edu/webres/can20.pdf>.
- [2] VAN HERREWEGE, Anthony a Ingrid VERBAUWHEDE. CANAuth - A Simple, Backward Compatible Broadcast Authentication Protocol for CAN bus [online]. [cit. 2022-10-14]. Dostupné z: <https://www.esat.kuleuven.be/cosic/publications/article-2086.pdf>.
- [3] The CAN Bus Protocol Tutorial [online]. Kvaser [cit. 2022-10-10]. Dostupné z: <https://www.kvaser.com/can-protocol-tutorial/##canBus>.
- [4] Hacking Your Car [online]. 2010 [cit. 2022-10-11]. Dostupné z: <https://marco.guardigli.it/2010/10/hacking-your-car.html>.
- [5] WATTERSON, Conal. Controller Area Network (CAN) Implementation Guide [online]. ©2012–2017 [cit. 2022-10-10]. Dostupné z: <https://www.analog.com/media/en/technical-documentation/application-notes/AN-1123.pdf>.
- [6] CORRIGAN, Steve. Introduction to the Controller Area Network (CAN) [online]. Texas Instruments, ©2018 [cit. 2022-10-11]. Dostupné z: <https://www.ti.com/lit/an/sloa101b/sloa101b.pdf>.
- [7] CAN FD - The basic idea [online]. [cit. 2022-10-13]. Dostupné z: <https://www.can-cia.org/can-knowledge/can/can-fd/>.
- [8] THIMM, Oliver. Migration to CAN FD [online]. 2021 [cit. 2022-10-13]. Dostupné z: <https://can-newsletter.org/uploads/media/raw/cd1e2ba4002ad9ed9dc316daaf017c9a.pdf>.
- [9] Controller Area Network Extra Long (CAN XL). Can-cia.org [online]. ©1992 - 2022 [cit. 2022-10-14]. Dostupné z: <https://www.can-cia.org/can-knowledge/can/can-xl/>.
- [10] RICHARDS, Pat. Understanding Microchip's CAN Module Bit Timing [online]. Microchip Technology, ©2001, 13 [cit. 2022-10-11]. Dostupné z: <https://ww1.microchip.com/downloads/en/Appnotes/00754.pdf>.
- [11] TINDELL, Ken. CAN bus. In: Wikipedia.org [online]. 2020 [cit. 2022-11-17]. Dostupné z: <https://commons.wikimedia.org/wiki/File:CAN-bus-frame-with-stuff-bit-and-correct-CRC.png>.
- [12] ISO1050 Isolated CAN Transceiver [online]. Texas Instruments, ©2022 [cit. 2022-12-05]. Dostupné z: https://www.ti.com/lit/ds/symlink/iso1050.pdf?HQS=dis-mous-null-mousermode-dsf-pf-null-ww&ts=1670221995452&ref_url=https%253A%252F%252Fwww.mouser.it%252F.
- [13] MCP2515 Stand-Alone CAN Controller with SPI Interface Data Sheet [online]. Microchip Technology, ©2003-2019 [cit. 2022-11-08]. Dostupné z:

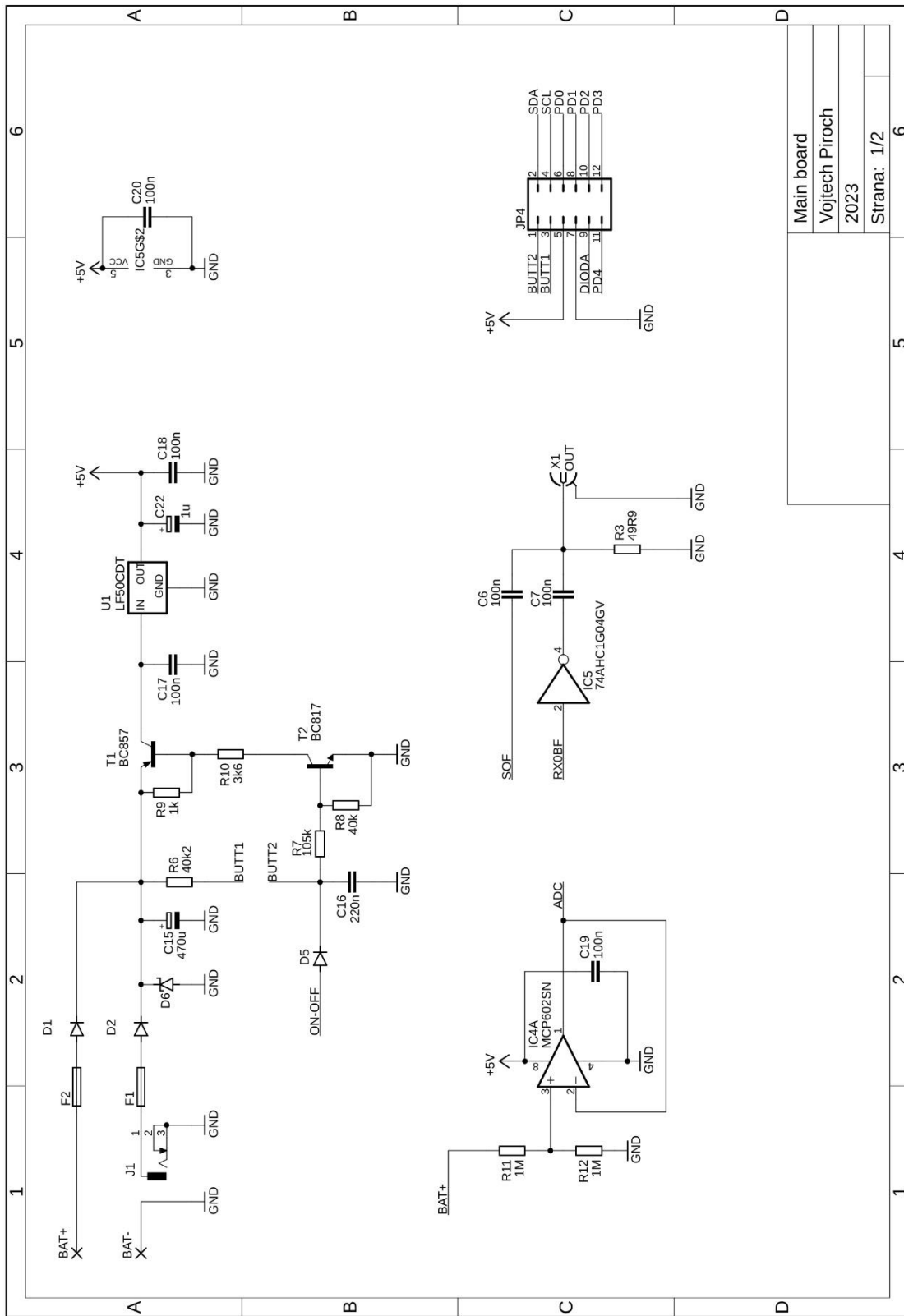
- <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP2515-Stand-Alone-CAN-Controller-with-SPI-20001801J.pdf>.
- [14] ATmega8 [online]. Atmel Corporation, 2013 [cit. 2022-12-01]. Dostupné z: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf.
- [15] OLED SSD1306 I2C Display(128x32) Interfacing with Arduino Uno [online]. Elementz [cit. 2022-12-08]. Dostupné z: OLED SSD1306 I2C Display(128x32) Interfacing with Arduino Uno.
- [16] 2N3906 [online]. Semiconductor Components Industries, ©2010 [cit. 2022-12-22]. Dostupné z: <https://www.onsemi.com/pdf/datasheet/2n3906-d.pdf>.
- [17] 2N3904 [online]. STMicroelectronics, ©2003 [cit. 2022-12-22]. Dostupné z: <https://www.sparkfun.com/datasheets/Components/2N3904.pdf>.
- [18] Oled-display [online]. 2018 [cit. 2023-05-02]. Dostupné z: <https://github.com/Sylaina/oled-display>.

SEZNAM PŘÍLOH

PŘÍLOHA A - FINÁLNÍ SCHÉMA ZAPOJENÍ	52
PŘÍLOHA B - SCHÉMA ZAPOJENÍ PROTOTYPU	56
PŘÍLOHA C - PRACOVÍŠTĚ S TESTOVACÍM ZAPOJENÍM	57
PŘÍLOHA D - NÁHLED PLOŠNÝCH SPOJŮ	59
PŘÍLOHA E - TESTOVACÍ PRACOVÍŠTĚ	61
PŘÍLOHA F - PROJEKT V EAGLE, NA PŘILOŽENÉM CD	62
PŘÍLOHA G - PROJEKT V MICROCHIP STUDIO, NA PŘILOŽENÉM CD	63
PŘÍLOHA H - MODEL KRABÍČKY, NA PŘILOŽENÉM CD	64

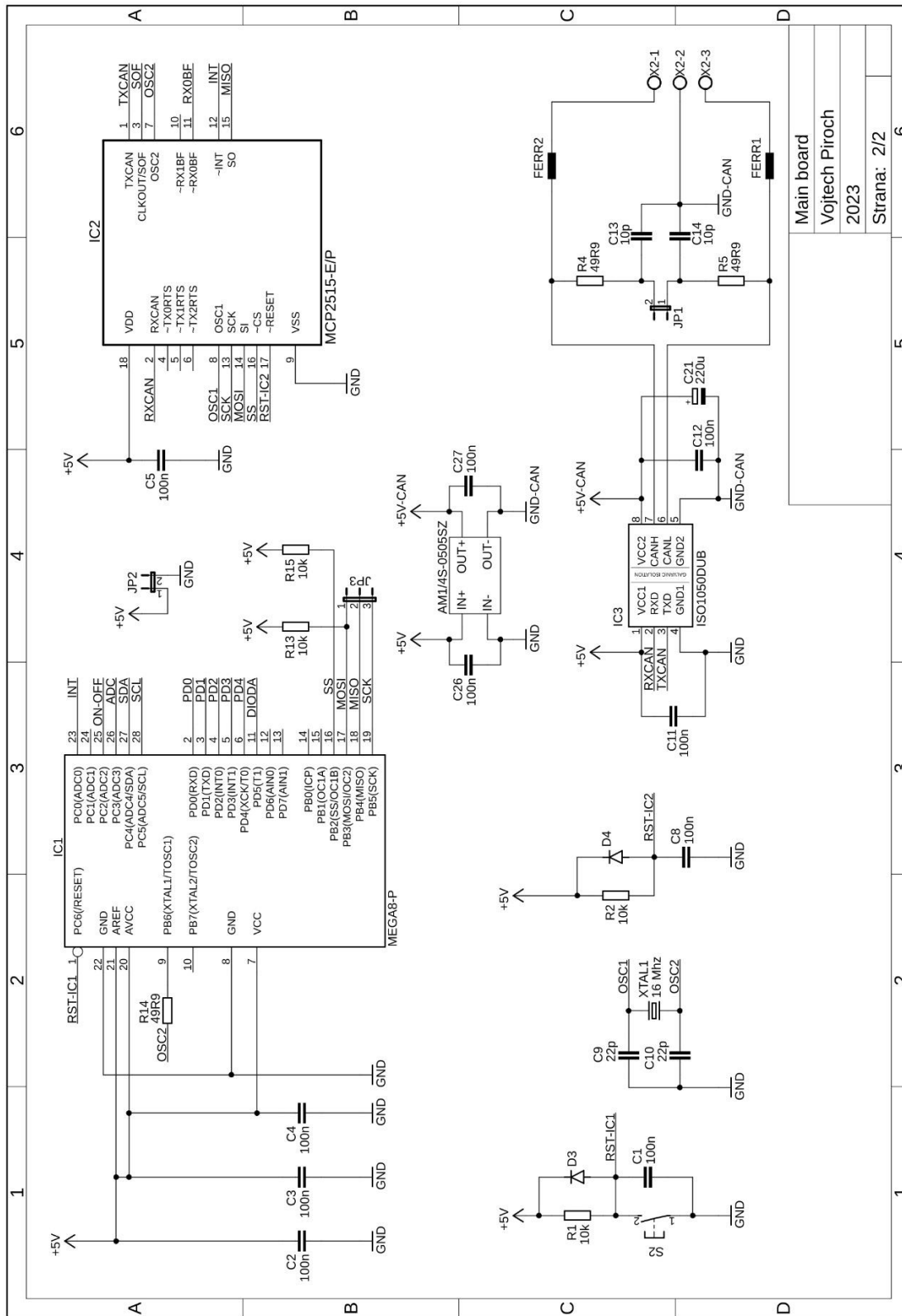
Příloha A - Finální schéma zapojení

1. Schéma hlavní desky 1
2. Schéma hlavní desky 2
3. Schéma ovládací desky

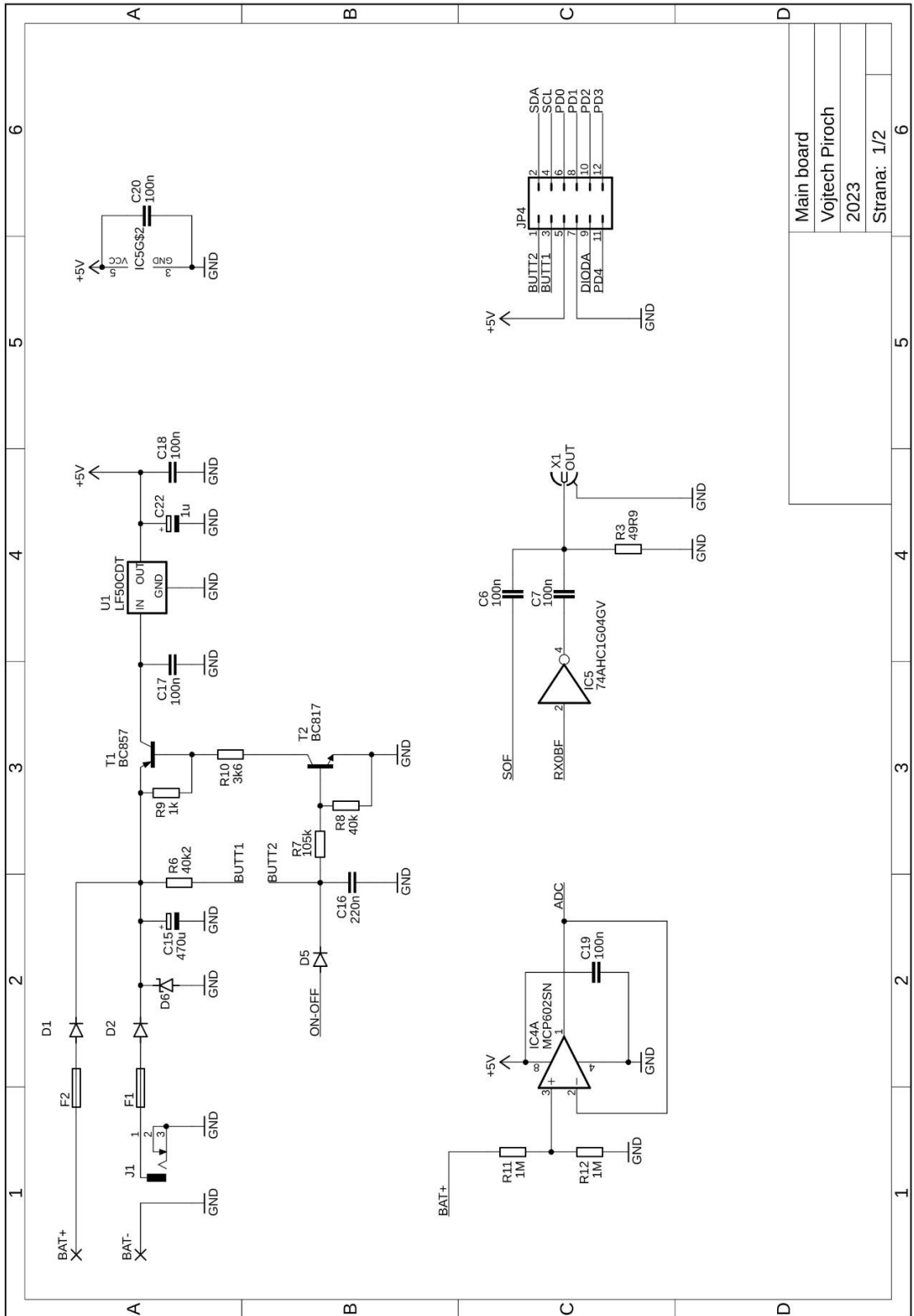


Main board	6
Vojtech Piroch	
2023	
Strana: 1/2	

Obrázek A.1 Schéma hlavní desky strana 1



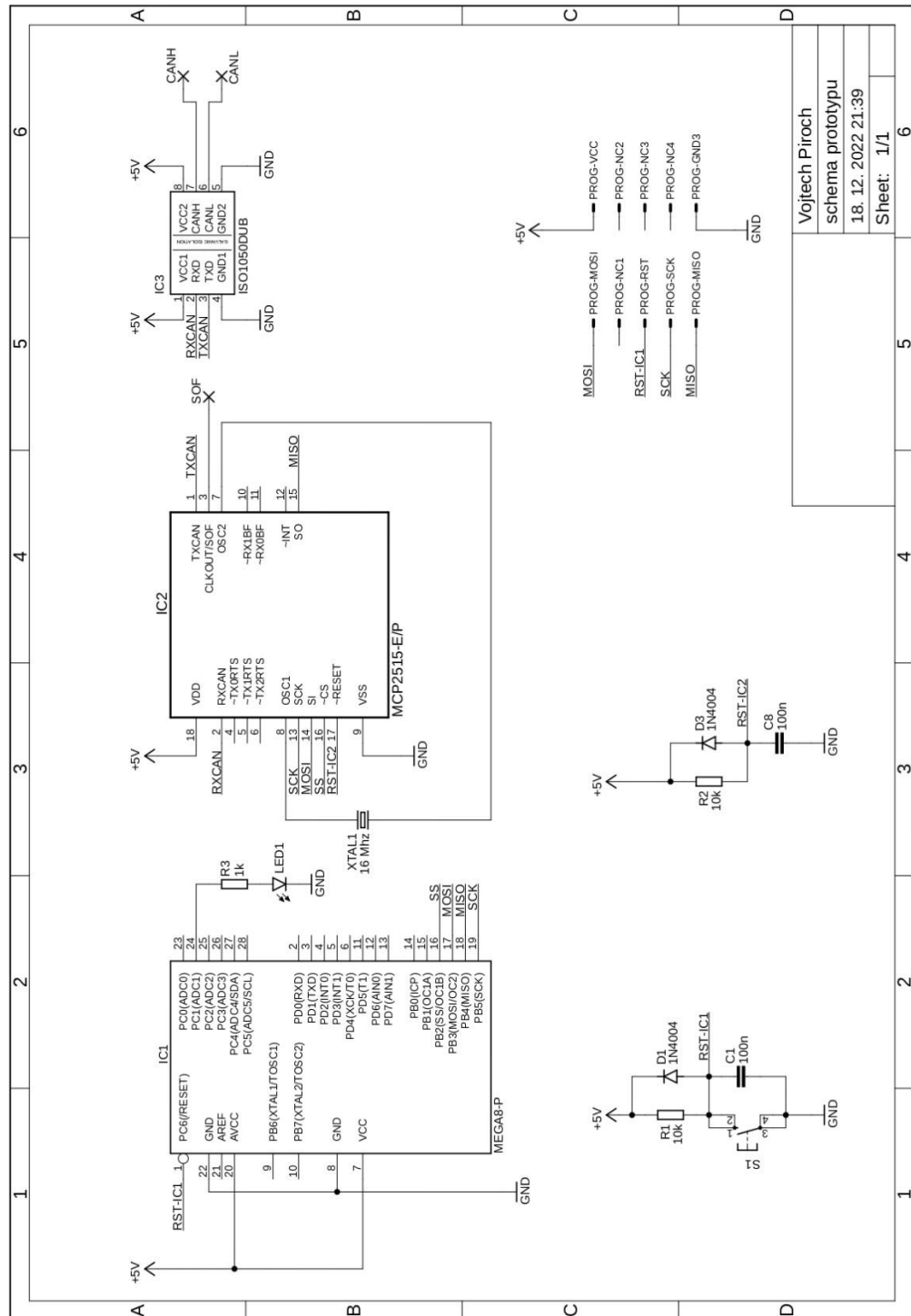
Obrázek A.2 Schéma hlavní desky strana 2



Main board
 Vojtech Piroch
 2023
 Strana: 1/2

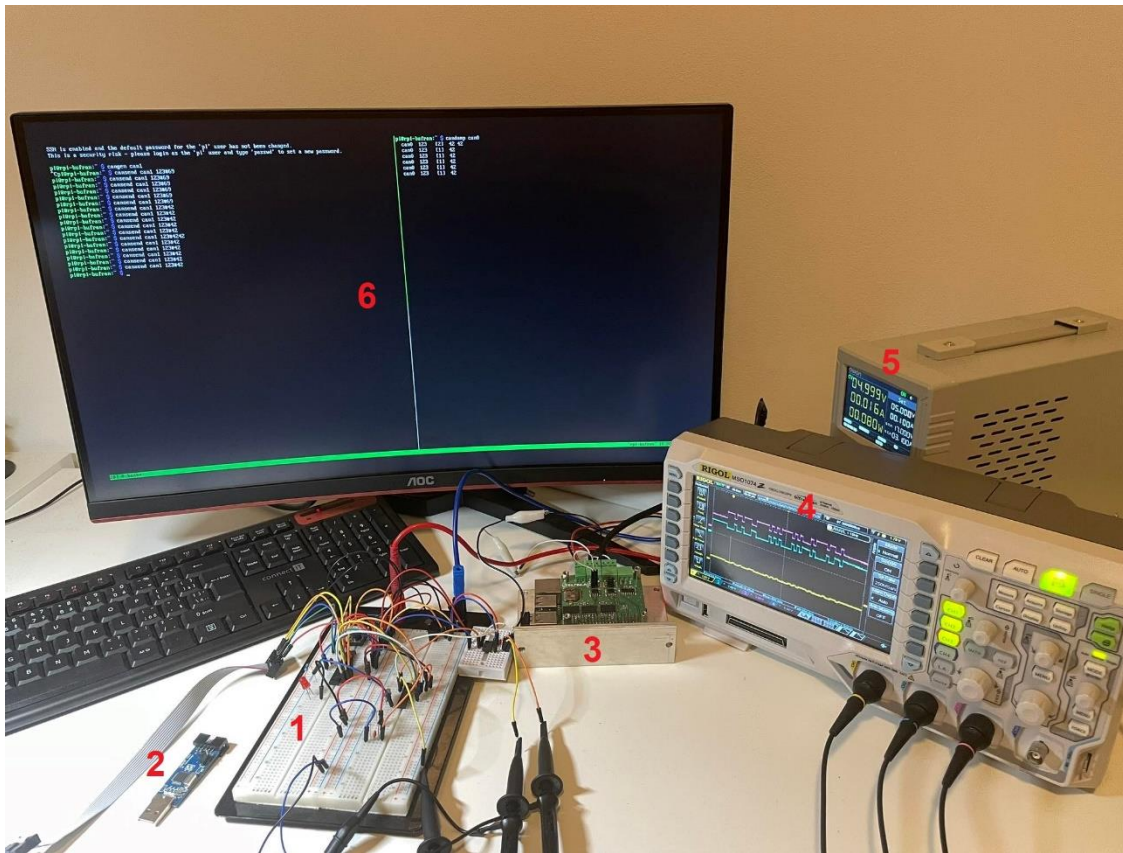
Obrázek A.3 Schéma ovládací desky

Příloha B - Schéma zapojení prototypu



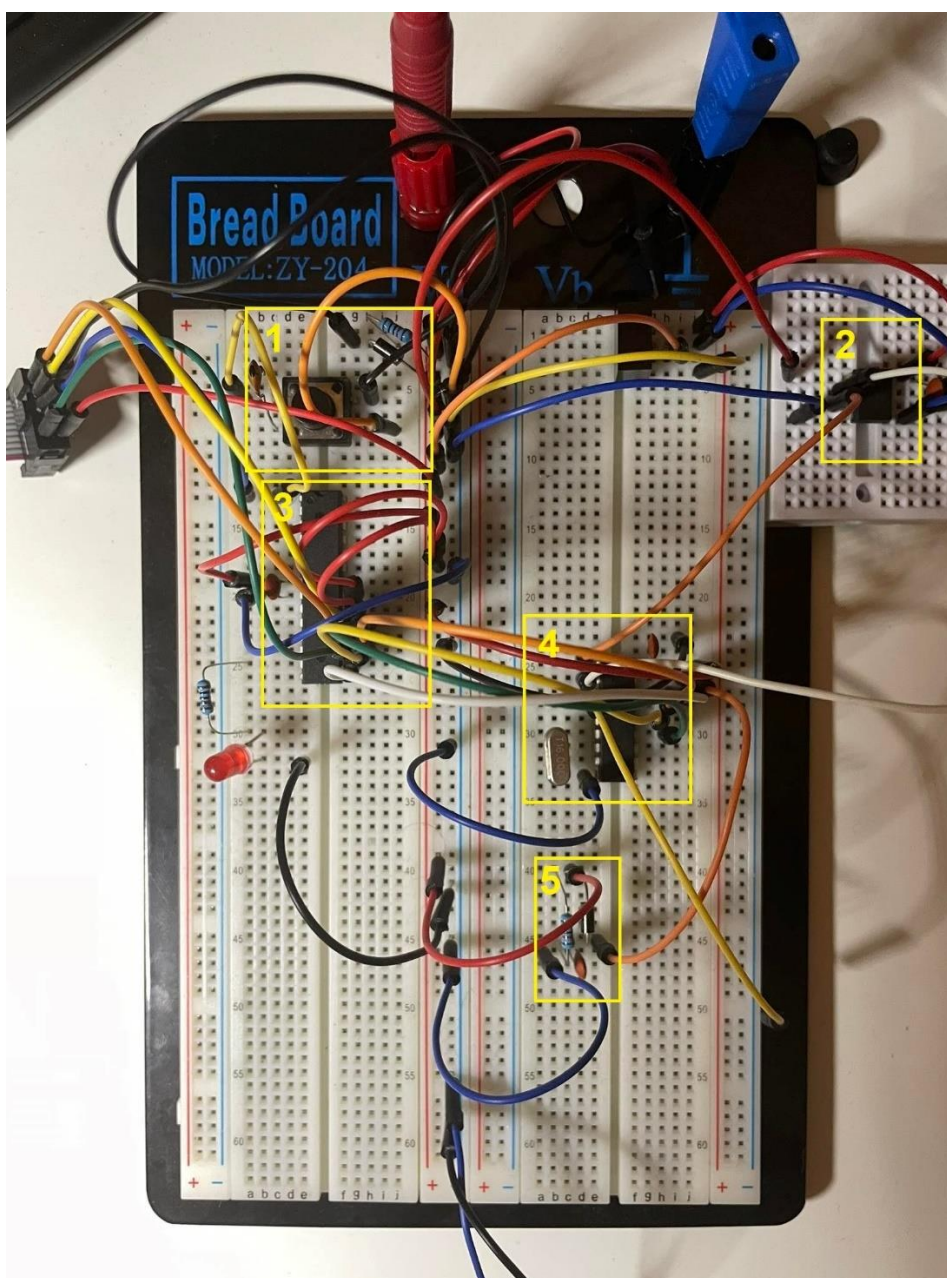
Obrázek B.4 Schéma testovacího prototypu

Příloha C - Pracoviště s testovacím zapojením



Obrázek C.5 Testovací pracoviště

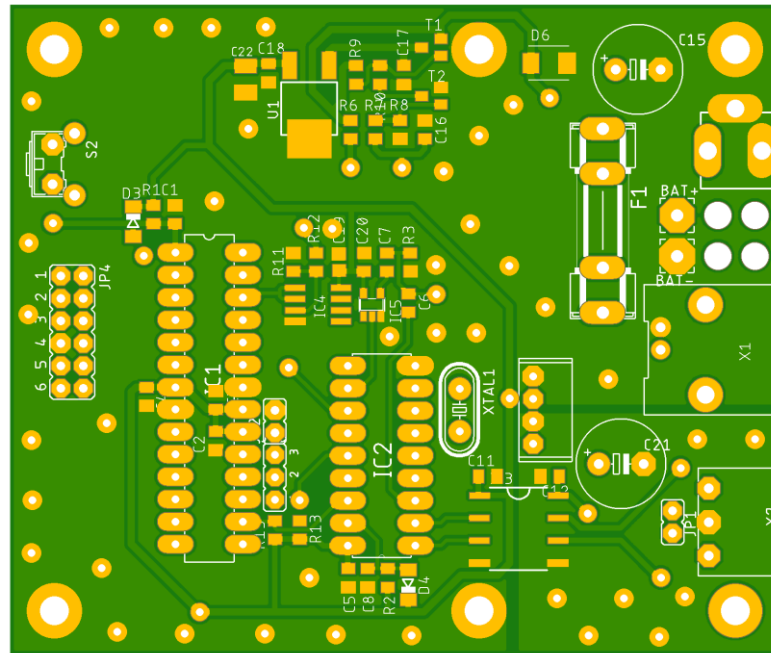
1. Testovací zapojení
2. Programátor USBASP v2.0
3. Raspberry Pi s deskou LTR, sloužící jako další uzel CAN, který odesílá zprávy
4. Osciloskop s vyobrazenou přijatou zprávou a triggerovacím impulzem SOF
5. Monitor připojený na Raspberry Pi pomocí HDMI



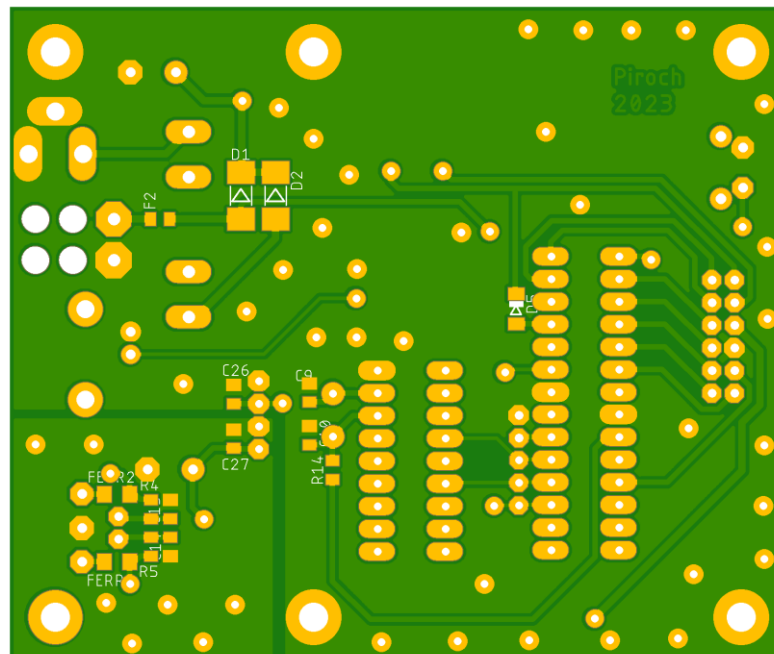
Obrázek C.6 Detail testovacího zapojení.

1. Resetovací obvod MCU
2. CAN budič ISO1050DUB
3. MCU ATmega8
4. CAN kontrolér MCP2515 s krystalem
5. Reset obvod CAN kontroléru

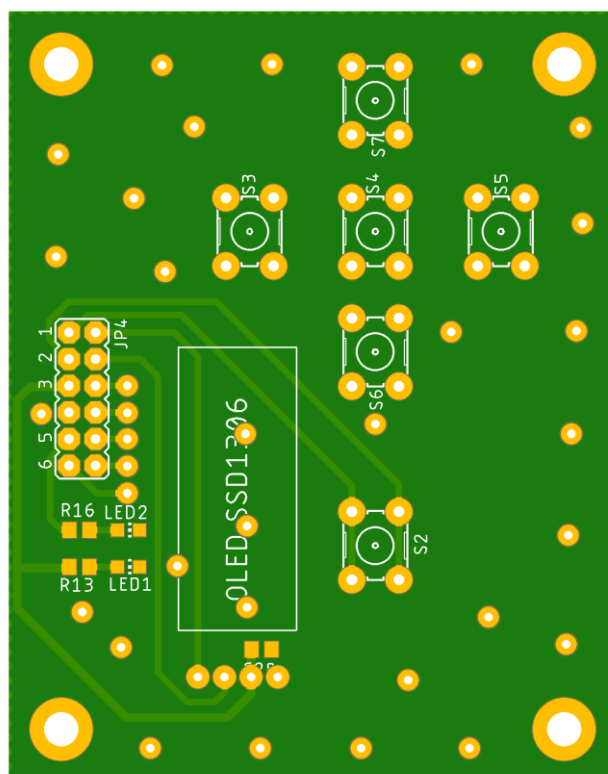
Příloha D - Náhled plošných spojů



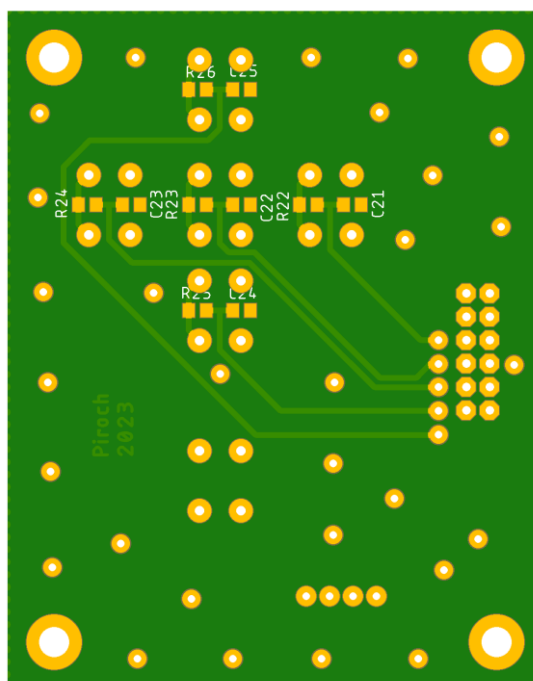
Obrázek D.7 Hlavní deska vrstva Top



Obrázek D.8 Hlavní deska vrstva Bottom

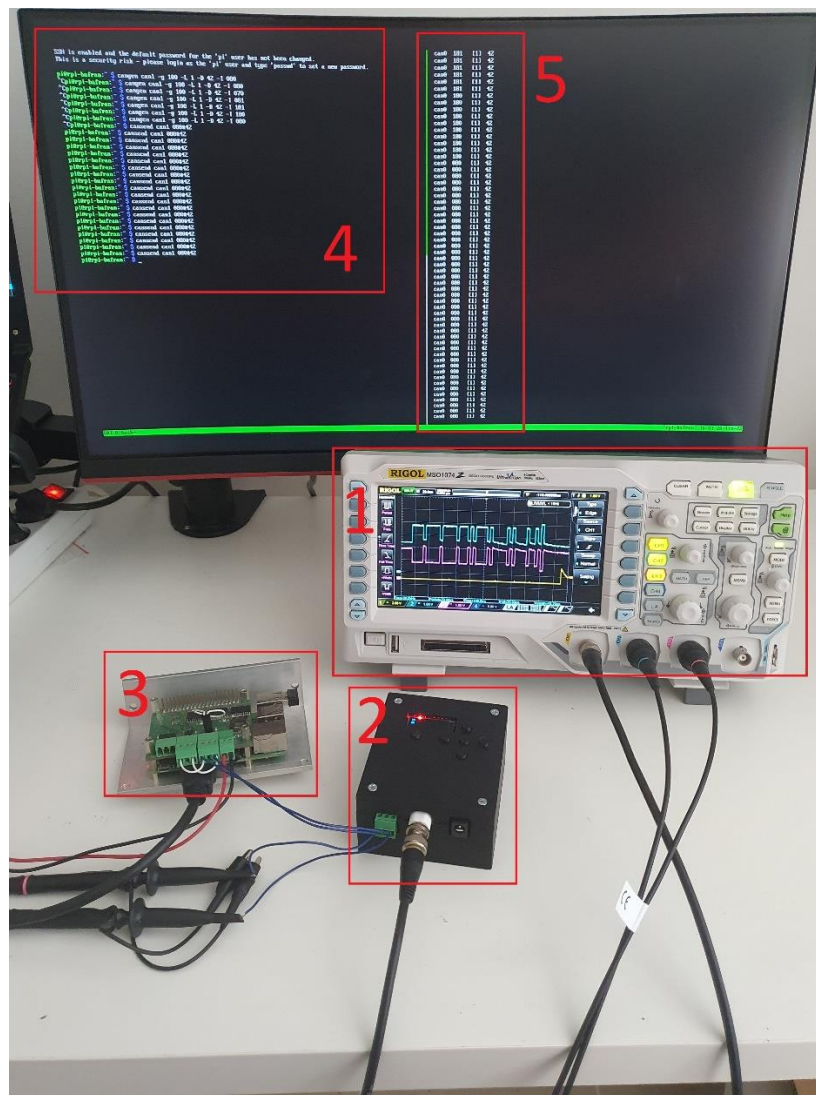


Obrázek D.9 Kontrolní deska vrstva Top



Obrázek D.10 Kontrolní deska vrstva Bottom

Příloha E - Testovací pracoviště



Obrázek E.11 Testování finálního zařízení.

1. Osciloskop se zachycenou zprávou
2. Triggerovací jednotka
3. Raspberry Pi odesílající a přijímající zprávy na sběrnici
4. Příkazy pro odesílání zpráv
5. Zprávy přijaté na sběrnici

Příloha F - Projekt v Eagle, na přiloženém CD

Příloha G - Projekt v Microchip studio, na přiloženém CD

Příloha H - Model krabičky, na přiloženém CD