



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

INTELIGENTNÍ GRAFICKÝ DISPLEJ PRO MĚŘICÍ ÚSTŘEDNU

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika

Studijní obor: 3906T001 – Mechatronika

Autor práce: **Bc. Petr Hoření**

Vedoucí práce: Ing. Lubomír Slavík, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

INTELLIGENT GRAPHIC DISPLAY FOR MEASURING MOTHERBOARD

Masters thesis

Study programme: N2612 – Electrical Engineering and Informatics

Study branch: 3906T001 – Mechatronics

Author: **Bc. Petr Hoření**

Supervisor: Ing. Lubomír Slavík, Ph.D.





Zadání diplomové práce

Inteligentní grafický displej pro měřicí ústřednu

<i>Jméno a příjmení:</i>	Bc. Petr Hoření
<i>Osobní číslo:</i>	M16000145
<i>Studijní program:</i>	N2612 Elektrotechnika a informatika
<i>Studijní obor:</i>	Mechatronika
<i>Zadávající katedra:</i>	Ústav mechatroniky a technické informatiky
<i>Akademický rok:</i>	2019/2020

Zásady pro vypracování:

1. Prostudujte strukturu procesorů ARM-Cortex M4.
2. Prostudujte dokumentaci řídicí ústředny GU100.
3. Navrhněte schéma a následně plošný spoj jednotky inteligentního displeje (zobrazení hodnot a ovládání pomocí klávesnice) pro ústřednu GU100 s důrazem na nízkou spotřebu.
4. Realizujte funkční model jednotky inteligentního displeje a navrhněte základní firmware.

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

dle potřeby dokumentace
40–50 stran
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] Praktické zkušenosti s procesory ARM Cortex M3, URL: <http://www.hw.cz>
- [2] Matoušek, D.: Práce s inteligentními displeji LCD, BEN Praha 2006, ISBN 80-7300-121-7.
- [3] Katalogové listy obvodů STM32F4xx a STM32L4xx.

Vedoucí práce:

Ing. Lubomír Slavík, Ph.D.
Ústav mechatroniky a technické informatiky

Konzultant práce:


Ing. Jaroslav Buchta
Ústav mechatroniky a technické informatiky

Datum zadání práce:

6. září 2019

Předpokládaný termín odevzdání:

18. května 2020


prof. Ing. Zdeněk Pliva, Ph.D.
děkan




doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 6. září 2019

Prohlášení

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

Jsem si vědom toho, že na mou bakalářskou/diplomovou/rigorózní/disertační práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské/diplomové/rigorózní/disertační práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou/diplomovou/rigorózní/disertační práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše. Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská/diplomová/rigorózní/disertační práce bude zveřejněna Technickou univerzitou v Liberci v souladu s §47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

Datum: 1.6.2020

Podpis: 

Declaration

I hereby certify, I, myself, have written my bachelor thesis/master thesis/rigorous thesis/dissertation as an original and primary work using the literature listed below and consulting it with my thesis supervisor and my thesis counsellor.

I acknowledge that my bachelor thesis/master thesis/rigorous thesis/dissertation is fully governed by Act No. 121/2000 Coll., the Copyright Act, in particular Article 60 – School Work. I acknowledge that the Technical University of Liberec does not infringe my copyrights by using my bachelor thesis/master thesis/rigorous thesis/dissertation for internal purposes of the Technical University of Liberec.

I am aware of my obligation to inform the Technical University of Liberec on having used or granted license to use the results of my bachelor thesis/master thesis/rigorous thesis/dissertation; in such a case the Technical University of Liberec may require reimbursement of the costs incurred for creating the result up to their actual amount.

At the same time, I honestly declare that the text of the printed version of my bachelor thesis/master thesis/rigorous thesis/dissertation is identical with the text of the electronic version uploaded into the IS STAG.

I acknowledge that the Technical University of Liberec will make my bachelor thesis/master thesis/rigorous thesis/dissertation public in accordance with paragraph 47b of Act No. 111/1998 Coll., on Higher Education Institutions and on Amendment to Other Acts (the Higher Education Act), as amended.

I am aware of the consequences which may under the Higher Education Act result from a breach of this declaration.

Date: 1.6.2020

Signature: 

Poděkování

Rád bych na tomto místě chtěl poděkovat všem, kteří se podíleli na vzniku této diplomové práce, zvláště vedoucímu práce Ing. Lubomíru Slavíkovi. Vedle jeho trpělivosti a vstřícného přístupu, také za přínosné informace z hlediska elektroniky.

Abstrakt

Tato diplomová práce se zabývá návrhem a realizací inteligentního displeje pro měřicí ústřednu, jehož klíčovou vlastností je nízká spotřeba. Teoretická pasáž čtenáře seznamuje se základními parametry měřicí ústředny pro sledování geofyzikálních veličin, jako je teplota nebo seismický otřes. Grafický displej je stěžejním prvkem této práce. Proto odstavce teoretické části obsahují popis různých technologií displejů a zhodnocení jejich výhod a nevýhod. Dále se věnuje procesoru typu ARM, jsou popsány jeho základní parametry a komunikace s okolními perifériemi, především s grafickými displeji.

Hlavním tématem praktické části je tvorba prototypu inteligentního zobrazovače s nízkou spotřebou elektrické energie. Nejdříve je naprogramována vývojová deska včetně komunikačních periférií, jako je zvolený displej, maticová klávesnice a komunikační rozhraní EIA-485. Celý program je vytvořen v softwaru Keil uVision, v jazyce C. Následně je navržena deska plošných spojů v prostředí Altium Designer. Nakonec je provedeno oživení vytvořeného prototypu a změřena jeho spotřeba.

Klíčová slova:

Měřicí ústředna, nízká spotřeba, displej, maticová klávesnice, rozhraní EIA-485, EPSNet

Abstract

This master's thesis deals with design and realization of the intelligent display for measuring unit, which key feature is low power consumption. The theoretic passage introduces main parameters of measuring motherboard for monitoring geophysical quantities like the temperature or seismic shock. The graphic display is the main topic of this work. Therefore, paragraphs of the theoretical part contain a description of various display technologies and evaluation of their advantages and disadvantages. In the last chapter ARM processor with his main parameters and peripheral communications, mainly graphics displays.

The main topic of the practical part is the design of an intelligent display prototype with low power consumption. Firstly, the development board is programmed, including communication peripherals such as the selected display, matrix keyboard, and RS485 communication interface. The whole program is created in Keil uVision software, in C language. Subsequently, the printed circuit board is designed in Altium Designer. Finally, the prototype is revived and its consumption measured.

Key words:

Measuring motherboard, low power consumption, display, matrix keyboard, EIA-485, EPSNet

OBSAH

Prohlášení.....	3
Declaration.....	4
Poděkování.....	5
Abstrakt.....	6
Abstract.....	7
Seznam obrázků.....	9
Seznam tabulek.....	11
Seznam zdrojových kódů.....	11
Seznam symbolů, termínů a zkratek.....	12
Úvod.....	13
1 Měřicí ústředna GU100.....	14
1.1 Motherboard.....	14
1.2 Měřicí moduly.....	14
1.3 Přehled na trhu.....	15
2 Uživatelské rozhraní.....	18
2.1 Klávesnice.....	18
2.1.1 Mechanická klávesnice.....	18
2.1.2 Membránová klávesnice.....	18
2.1.3 Kapacitní klávesnice.....	19
2.1.4 Piezoelektrické klávesnice.....	19
2.2 Dotykový panel.....	20
2.2.1 Rezistivní dotykový panel.....	20
2.2.2 Kapacitní dotykový panel.....	20
2.3 Displej.....	21
2.3.1 LCD.....	22
2.3.2 E – Ink.....	23
2.3.3 Memory LCD.....	24
2.4 Komunikace s mikroprocesorem.....	24
2.4.1 SPI.....	26
2.4.2 I2C.....	27
3 EPSNet.....	28
3.1 Fyzická vrstva.....	28

3.2	Linková vrstva a Aplikační vrstva	30
3.3	Porovnání s ostatními protokoly	32
3.3.1	CANopen a DeviceNet	32
3.3.2	Modbus (RTU/ASCII)	32
3.3.3	Profibus (DP/PA)	32
4	Řídicí procesor ARM	34
4.1	Popis architektury	34
4.1.1	Jádro	34
4.1.2	Instrukční sada	35
4.2	Procesory řady STM32	35
5	Praktická část	37
5.1	Program řídicí jednotky	37
5.1.1	Klávesnice	39
5.1.2	Displej	40
5.1.3	Grafické rozhraní	42
5.1.4	EPSNet	49
5.2	Návrh hardwaru	57
5.2.1	Výkonová část modulu	57
5.2.2	Řídicí jednotka	59
5.2.3	Modul sériové linky RS485	60
5.2.4	Zálohovaná paměť EEPROM	60
5.2.5	Klávesnice	61
5.2.6	Displej	62
5.2.7	Deska plošných spojů	64
5.3	Zhotovení a oživení prototypu	65
6	Závěr	69
	Seznam odborné literatury	70
A	Obsah příloženého CD	74
B	Tabulka rámcových znaků v síti EPSNet	75
C	Osazovací schéma	76
D	Seznam součástek	77
E	DPS – strana TOP	78
F	DPS – strana BOTTOM	79

Seznam obrázků

Obrázek 1.1: Měřicí ústředna ALMEMO 500 [2]	15
Obrázek 1.2: Datalogger COMET MS55D [3]	16
Obrázek 1.3: OMNILOG NI-2400 [4]	16
Obrázek 2.1: Schéma vyhodnocovacího obvodu pro kapacitní klávesu.....	19
Obrázek 2.2: Spotřeba různých technologií zobrazovačů [10]	22
Obrázek 2.3: Princip technologie E – Ink displeje [15]	23
Obrázek 2.4: Konstrukce COG a SMD čipu [17]	25
Obrázek 2.5: Připojení LCD panelu FPC konektorem [17]	25
Obrázek 2.6: Sběrnice SPI [18].....	26
Obrázek 2.7: Komunikace pomocí I2C [20]	27
Obrázek 3.1: Napět'ová úroveň sběrnice RS485	29
Obrázek 3.2: Oktet protokolu EPSNet	29
Obrázek 3.3: Průřez průmyslového kabelu kroucené dvojlinky [21]	29
Obrázek 4.1: Blokové schéma ARM jádra	34
Obrázek 4.2: Pracovní režimy procesoru STM32L496RGx [33]	36
Obrázek 4.3: Orientační odběr proudů [27]	36
Obrázek 5.1: Pinout přehled použitých pinů na vývojové desce NUCLO-L496ZG	37
Obrázek 5.2: Hardwarové prostředky řídicího programu	38
Obrázek 5.3: Časový diagram displeje Sharp [28].....	41
Obrázek 5.4: Část vývojového diagramu pro uživatelské rozhraní – část Úvodní zobrazení. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.....	43
Obrázek 5.5: Obrazovka Seznam modulů.....	43
Obrázek 5.6: Část vývojového diagramu pro uživatelské rozhraní – část Prohlížení seznamu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.....	44
Obrázek 5.7: Obrazovka Seznam modulů – následující osmice	44
Obrázek 5.8: Část vývojového diagramu pro uživatelské rozhraní – část Výběr modulu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.....	45
Obrázek 5.9: Obrazovka Výběr modulu	45
Obrázek 5.10: Část vývojového diagramu pro uživatelské rozhraní – část Menu modulu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.....	46
Obrázek 5.11: Menu vybraného měřicího modulu	46
Obrázek 5.12: Část vývojového diagramu pro uživatelské rozhraní – část Zobrazení hodnot. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.....	47
Obrázek 5.13: Obrazovka s naměřenými hodnotami	47
Obrázek 5.14: Nastavení jasu podsvícení displeje	48

Obrázek 5.15: Simulovaná komunikace modulu inteligentního displeje s motherboardem	49
Obrázek 5.16: Diagram komunikace modulu přes protokol EPSNet	50
Obrázek 5.17: Osciloskopem zaznamenaný průběh jednoho oktetu na lince A a B. 53	53
Obrázek 5.18: Stavový diagram přijímače	53
Obrázek 5.19: Blokové schéma modulu inteligentního displeje.....	57
Obrázek 5.20: Schéma zapojení řídicího obvodu vstupního napětí +12 V.....	58
Obrázek 5.21: Spínaný zdroj napětí +5 V a lineární zdroj napětí +3,3 V.....	58
Obrázek 5.22: První část zapojení řídicího procesoru typu ARM	59
Obrázek 5.23: Druhá část zapojení řídicího procesoru typu ARM.....	59
Obrázek 5.24: Zapojení sériové linky RS485	60
Obrázek 5.25: Zapojení EEPROM paměti.....	60
Obrázek 5.26: Membránová klávesnice	61
Obrázek 5.27: Ukázka ztráty životnosti použitého e-ink displeje.....	62
Obrázek 5.28: LED modul od Adafruit.....	63
Obrázek 5.29: Obvodové zapojení spínaných LED modulů.....	63
Obrázek 5.30: Osazovací schéma desky – strana BOTTOM.....	64
Obrázek 5.31: Podpěra pro klávesnici	65
Obrázek 5.32: Tělo čelního panelu	65
Obrázek 5.33: 3D sestava DPS a čelního panelu	66
Obrázek 5.34: Zhotovený čelní panel	66
Obrázek 5.35: Programování inteligentního displeje pomocí vývojové desky NUCLEO-L476RG. Prototyp je provizorně napájen z regulovatelného zdroje napětí EP-613, silnějším žlutým kabelem přiveden potenciál +12 V a silnějším modrým kabelem přiveden potenciál GND.	67
Obrázek 5.36: Záznam průběhů napětí během procesu vypnutí, kanál 1 – napětí +12 V, kanál 2 – napětí +5 V.	67

Seznam tabulek

Tabulka 1.1: Porovnání ústředny GU100 s trhem.....	17
Tabulka 2.1: Parametry studia elektrické spotřeby	21
Tabulka 2.2: Typické parametry technologie E-ink a Memory LCD	24
Tabulka 3.1: Základní rozdíl sériových standardů.....	28
Tabulka 3.2: Porovnání protokolů.....	33
Tabulka 5.1: Naměřené hodnoty odebíraného proudu displejem	68
Tabulka 5.2: Naměřené hodnoty proudu pro vyrobený prototyp.....	68

Seznam zdrojových kódů

Zdrojový kód: 5.1: Inicializace SPI1	40
Zdrojový kód: 5.2: Datová struktura reprezentující měřicí modul	42
Zdrojový kód: 5.3: Změna hodnoty podsvícení	48
Zdrojový kód: 5.4: Inicializace LPUART1	49
Zdrojový kód: 5.5: Ukázka některých symbolických konstanty pro EPSNet.....	51
Zdrojový kód 5.6: Ukázka části obslužného programu přerušení z obvodu UART. 54	
Zdrojový kód 5.7: Ukázka části obslužného programu přerušení z obvodu časovačů	55
Zdrojový kód 5.8: Ukázka převodů naměřené hodnoty nutné pro přenos	56

Seznam symbolů, termínů a zkratek

GU	Geofyzikální ústředna
ARM	Advanced RISC Machine
RS485	Časová konstanta
RISC	Emitorová plocha
MAR	Memory Address Register
MBR	Memory Buffer Register
MDR	Memory Data Register
FIQ	Fast Interrupt ReQuest
IRQ	Interrupt ReQuest
SPSR	Saved Program Status Register
CPSR	Current Processor Status Register
Ri	Registr i-tého pořadí
PC	Program counter
LR	Link Register
SP	Stack Pointer
Pipeline	Proudové zpracování instrukcí
RTC	Real Time Clock - Obvod reálného času
TCP/IP	Transmission Control Protocol/Internet Protocol - primární přenosový protokol/protokol síťové vrstvy
HMI	Human Machine Interface – rozhraní mezi člověkem a strojem
LCD	Liquid Crystal Display - displej z tekutých krystalů
COG	Chip On Glass
PLL	Phase Locked Loop – fázový závěs
LPUART	Low power Universal Synchronous/Asynchronous Receiver and Transmitter
LPTIM	Low power timer
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CAN	Controller Area Network
Profibus	Process Field Bus
DPI	Dots per inch
LED	Light-Emitting Diode

Úvod

Obsahem diplomové práce je navrhnout a vytvořit lokální komunikační rozhraní pro měřicí ústřednu (GU100), která slouží pro on-line sledování geofyzikálních procesů a jevů. Celá infrastruktura měření je složena ze sítě měřicích ústředn. Každá ústředna pak komunikuje s měřicími moduly různých geofyzikálních veličin. Součástí stávající ústředny je uživatelské rozhraní, které je tvořeno podsvíceným LCD displejem a kapacitní klávesnicí. Dále ústředna obsahuje záložní zdroj elektrické energie, a proto je kladen velký důraz na minimální spotřebu každého prvku měřicího systému. Je tedy nutné navrhnout nové uživatelské rozhraní pro dosažení úspornějšího provozu.

V první kapitole je popsána základní struktura měřicí ústředny a uveden výčet několika měřicích modulů. Kapitola druhá popisuje jednotlivé části HMI systému. Třetí kapitola čtenáře provádí se základním popisem komunikace pomocí protokolu EPSNet. Poslední kapitola teoretické části práce obsahuje popis řídicího procesoru ARM, který je součástí měřicí infrastruktury a také navrhovaného komunikačního rozhraní.

Praktická část práce obsahuje popis implementovaného řídicího softwaru, popis navrženého hardwaru inteligentního modulu displeje a závěrem jsou uvedeny výsledky naměřených proudů a napětí oživeného prototypu.

1 Měřicí ústředna GU100

1.1 Motherboard

Výpočet a přenos dat ústředny řídí mikroprocesor ARM (viz kapitola Řídicí procesor ARM), konkrétně typ Cortex M4 STM32F407. Jádro lze taktovat maximální frekvencí 168 MHz (po vynásobení pomocí PLL), zdrojem je krystal 25 MHz. Pro obvod reálného času (RTC) je použit krystal 32 kHz. Přijímat naměřená data z modulů lze přes dvě sběrnice EIA-485 (jedna interní – bez galvanického oddělení, druhá externí – s galvanickým oddělením). Procesor pak z těchto dat vytváří lokální databázi (data jsou uložena na MicroSD kartě) a zároveň lze databázi poskytnout nadřazenému systému pomocí rozhraní Ethernet (protokol TCP/IP).

Další úlohou procesoru je hlídání spotřeby energie celého zařízení. Motherboard obsahuje několik spínaných zdrojů (3,3 V, 12V a 24 V), které lze v případech úplného vypnutí měřicích modulů zapnout a vypnout na žádost procesoru. V opačném případě jsou procesory interních měřicích modulů v režimu standby a hlavním procesorem jsou probuzeny. Lze naprogramovat i obrácenou logiku probuzení tak, že teplotní modul probudí hlavní procesor v případě průběžného sledování teplot a překročením povolených hodnot se vyvolá alarm. Takto lze řídit a napájet pouze interní moduly, pro které jsou k dispozici čtyři pozice.

Ústřednu lze napájet z externího zdroje 230 V nebo 24 V, napětí je následně z těchto zdrojů převedeno na 13,6 V. Druhou možností je zařízení napájet z baterie LiFePO4 o napětí 12 V.

1.2 Měřicí moduly

V současné době lze zapojit následující moduly:

- 1) modul pro měření teplot,
- 2) modul pro měření proudové smyčky 4-20 mA,
- 3) modul pro měření napěťových signálů 0-10 V,
- 4) modul pro měření pulzních signálů,

Moduly jsou osazeny 32bitovým procesorem ARM Cortex M0 STM32F051 (pouze modul teplot má STM32L152), jejichž pracovní frekvence je 48 MHz.

1.3 Přehled na trhu

V následující podkapitole jsou pro srovnání s GU100 uvedeny měřicí ústředny, které lze na českém trhu zakoupit.

Na českém trhu je novinkou měřicí ústředna **ALMEMO 500** německého výrobce AHLBORN GmbH. Standardně je k dispozici 20 měřicích vstupů s možností navýšit tento počet o dalších 70 vstupů. Snímače se k ústředně připojují patentovanými konektory ALMEMO, z těchto konektorů přístroj zjistí typ snímače a následně přednastaví konkrétní veličinu a měřicí rozsah. Ústředna zaznamená až 600 milionů hodnot na interní SD kartě (4 GB), paměť je dále možné rozšířit pomocí sériového USB portu. Zdrojem může být střídavé napětí (100 až 240 V) nebo stejnosměrné napětí (12 V, 2A) ze dvou lithiových akumulátorů s celkovou kapacitou 13,8 Ah. Pro obsluhu a zobrazení hodnot je součástí nabídky tablet s předinstalovanou aplikací.

Výrobce nabízí přístroj ve dvou variantách

- ve stolním provedení
- v klasickém 19palovém rámu pro vestavbu do rozvaděče



Obrázek 1.1: Měřicí ústředna ALMEMO 500 [2]

Pro měření a záznam fyzikálních veličin lze použít i modulární měřicí ústřednu **MS55D** od výrobce COMET SYSTEM, s.r.o. Ústředna obsahuje 16 měřicích vstupů pro měření veličin (teploty, vlhkosti, tlaku, CO₂, napětí, proudu, ...). Teplota měřená termočlánkem má přesnost $\pm 0,3\%$ z měřené hodnoty a teplotním čidlem (Pt či Ni) má přesnost $\pm 0,2\%$ z měřené hodnoty. Pro stejnosměrný proud a napětí je přesnost měření $\pm 0,1\%$ z plného rozsahu. Pro uložení naměřených údajů (až 480 000 hodnot) slouží vnitřní zálohovaná SRAM paměť. Pro napájení záznamové ústředny je možné dokoupit zálohovaný zdroj a dva kusy olověných akumulátorů (12V/7Ah). Standardem je rozhraní RS232, RS485 a USB konektor, Ethernet modul je pouze na objednávku.



Obrázek 1.2: Datalogger COMET MS55D [3]

Datalogger **OMNILOG NI-2400** od produkce Next Industries disponuje 24 analogovými vstupy s 24 bitovým rozlišením a pomocí 16-ti kanálového multiplexování je možnost tento počet rozšířit až na 384 kanálů. Ústředna dokáže měřit teplotu, proudovou smyčku 0-20 mA, napětí 0-10 V, Wheatsonův můstek $\pm 10\text{MV/V}$ a odpor do hodnot $10\text{k}\Omega$. Naměřené hodnoty s přesností 0,01% z měřeného rozsahu jsou uložena na 2 GB interní SD kartu, data lze dále přenášet rozhraním Ethernet (10/100 Mbps), RS485, RS232 a USB. Pro základní nastavení obsluha pracuje s membránovou klávesnicí a podsvíceným LCD displejem (128x64 DPI).



Obrázek 1.3: OMNILOG NI-2400 [4]

Řídicím procesorem je ARM Cortex-M3 (120 MHz) se 1 MB zálohovanou RAM pamětí a 1 MB pamětí typu Flash. Elektronika má teplotní závislost $<10\text{ ppm}/^\circ\text{C}$ v rozsahu -30°C až $+70^\circ\text{C}$. Ústřednu lze napájet z externího zdroje napětí 10 až 30 V nebo z externí baterie 12 V. Uvnitř ústředny jsou navíc dvě Li/SOCI2 baterie o napětí 3,6 V.

Tabulka 1.1: Porovnání ústředny GU100 s trhem

Parametr	GU100	ALMEMO 500	MS55D	NI-2400
Počet kanálů	až 1008	až 200	až 16	až 384
Velikost paměti [MB]	2048 (lze rozšířit)	4096 (lze rozšířit)	2	2048 (lze rozšířit)
Počet záznamů	-	až 600 mil	až 480 tis	až 5 mil
Frekvence měření [kHz]	až 1	až 1	až 5	až 1
Komunikační rozhraní	Ethernet, RS485	USB, Ethernet	USB,(Ethernet), RS485,RS232,	Ethernet, USB, RS485, RS232
Napájecí napětí [VAC]	230	100 až 240	-	-
Napájecí napětí [VDC]	24 baterie: 12	12	9 až 30	10 až 30 ext. baterie: 12 int. baterie: 7,2
Proudový odběr [mA]	-	300	80	123 (při 12 V) 110 (při 7,2 V)
Provozní teplota [°C]	0 až + 50	-	0 až + 50	-30 až +70

2 Uživatelské rozhraní

Uživatelské rozhraní je soubor způsobů, jakým člověk (uživatel) ovlivňuje program mikroprocesorů, robotů, zařízení či komplexnějších systémů.

Způsob ovlivňování se dělí podle směru události:

- vstupy od uživatele, kterými se program řídí (klávesnice, ovládání řečí, pohybové gesty...),
- výstupy, které zobrazují výsledky uživatelských vstupů (displej, LED indikátor, zvuk z reproduktoru nebo buzzeru, ...)

Následující text neslouží pro vyčerpávající popis prvků uživatelského rozhraní, ale jsou uvedeny pouze některé technologie, které lze pro konstrukci inteligentního displeje použít.

2.1 Klávesnice

K základní vstupní periférii zařízení pro ovládání uživatelského programu a vkládání znaků do textového pole slouží klávesnice. Jednotkovým prvkem klávesnice je tlačítko, které nemusí být nutně jeho součástí.

2.1.1 Mechanická klávesnice

Jádrem klávesnice je kovový plát, nad nějž jsou upevněna mechanická tlačítka, kde pro návratový mechanismus je použita pružina a svorka. Výhodou této technologie je dlouhá životnost, kolik cyklů stisknutí dokáže tlačítko snést. Nejpoužívanější mechanická tlačítka dosahují až 20 milionů stisků. To je v porovnání s membránovou klávesou asi 10x více. Klávesnice je robustní a snadno opravitelná, ale i váhově těžká.

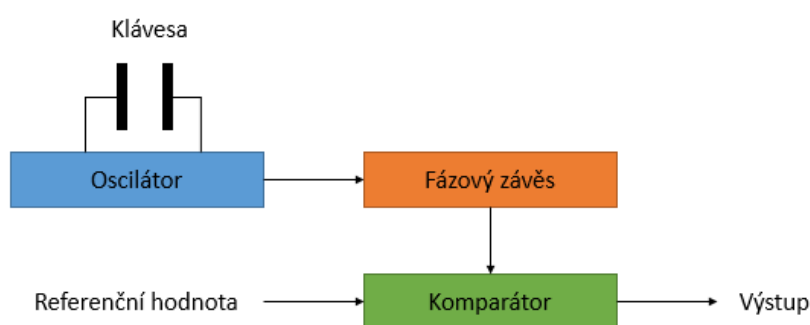
2.1.2 Membránová klávesnice

Mechanický spínač, u kterého je hmatová zpětná vazba nahrazena membránou (silikonovou) s bodovým uhlíkovým kontaktem na vrcholu. Výhodou je možnost použitím v agresivním prostředí a nízká cena. Nevýhodou je menší životnost (5 až 10 milionu cyklů) a horší samovolná návratová schopnost.

2.1.3 Kapacitní klávesnice

Kontaktně nemechanická klávesnice využívající elektrické kapacity mezi pevnou elektrodou a pružnou membránou. Je dražší než membránová a mechanická klávesnice. Protože se tento typ spínače nespolehá na kovové kontakty, je téměř imunní vůči korozi, nečistotám, má dlouhou životnost (až 30 milionů stisknutí) a navíc jsou velmi odolné vůči problémům s *odskokem*, který vede na několika násobné objevení znaku při jednom úderu tlačítka.

Pro vyhodnocení stavu stisknutí je potřebný řídicí obvod, který je zobrazen na obrázku Obrázek 2.1: Schéma vyhodnocovacího obvodu pro kapacitní klávesu.



Obrázek 2.1: Schéma vyhodnocovacího obvodu pro kapacitní klávesu

Kapacita kondenzátoru je dána vztahem:

$$C = \frac{\epsilon_0 \epsilon_r S}{d} \quad (4.1)$$

Při stisku klávesy dojde ke zmenšení vzdálenosti d a dojde ke zvýšení kapacity a tím ke snížení kapacitní reaktance:

$$X_c = \frac{1}{2\pi f C} \quad (4.2)$$

2.1.4 Piezoelektrické klávesnice

Piezoelektrická klávesnice je založená na přímém piezoelektrickém efektu. Při stisknutí tlačítka dochází k deformaci piezoelektrického prvku a na jeho elektrodách vzniká napětí dostatečné pro sepnutí tranzistorů. Uspořádání kladných a záporných iontů v krystalové mřížce piezomateriálu je takové, že se navzájem prostorově odpovídají a materiál je v klidovém stavu (bez deformací) elektricky neutrální. Klávesnice této technologie je elektricky nejučinnější a dosahují vysoké životnosti (až 50 milionů cyklů), neboť neobsahují žádné pohyblivé součásti. Nevýhodou je velká pořizovací cena.

2.2 Dotykový panel

S příchodem kapesních počítačů a následně smartphonů jsou klávesnice nahrazeny dotykovými obrazovkami. Navíc ceny dotykových obrazovek v poslední době klesají, stávají se dotykové obrazovky samozřejmou součástí každého moderního zařízení.

Mezi hlavní výhodu dotykového panelu patří lepší interakce s prostředím, protože uživatel se dotýká přímo toho, co je zobrazeno. Proto nedílnou součástí uživatelského rozhraní je i displej, který přesně určuje, kam má uživatel kliknout (např. zobrazením grafického tlačítka). [8]

2.2.1 Rezistivní dotykový panel

Rezistivní systém se skládá ze dvou tenkých vodivých vrstev, skleněného panelu s čistě vodivou vrstvou a odporovou vrstvou. Obě vrstvy jsou navzájem odděleny velmi pružnou membránou a tenkou vrstvou vzduchu. Pokud je displej pod napětím a uživatel zatlačí na jeho povrch, membrána se prohne a zatlačí na spodní vodivou vrstvu. Změnou elektrického pole v místě kontaktu je zařízením vyhodnocena souřadnice bodu dotyku.

U této technologie nezáleží na tom, čím je na displej tlačeno. Tlakový systém funguje stejně na prst i umělohmotné pero, nezáleží tedy na materiálu, ale pouze na tlaku dotyku. Nevýhodou je nedostatečná citlivost a přesnost dotyku.

2.2.2 Kapacitní dotykový panel

Kapacitní systém uchovává elektrický náboj, který je umístěn na neviditelné elektrostatické vrstvě. Při doteku obrazovky prstem se část náboje přenesou na uživatele, takže je tato vrstva narušena (kapacita vrstvy se sníží). Tento pokles se měří v rozích panelu a podle relativního rozdílu nábojů v každém rohu je určeno místo dotyku.

Kapacitní technologie je mnohem citlivější a přesnější než technologie rezistivní, protože nezáleží na velikosti mechanického tlaku. Dále má delší životnost a obraz displeje pod dotykovým panelem je při stejném podsvícení o něco jasnější. Jedinou nevýhodou je závislost na materiálu, nereagují na nevodivé předměty.

2.3 Displej

Jde o nejčastější výstupní periférii uživatelské rozhraní. Pokud je k zobrazení použita technologie LCD, pak je nutné k celkové spotřebě zobrazovacího panelu zahrnout i elektrický výkon zdroje světla. Proto je důležité si zvolit vhodnou technologii displeje.

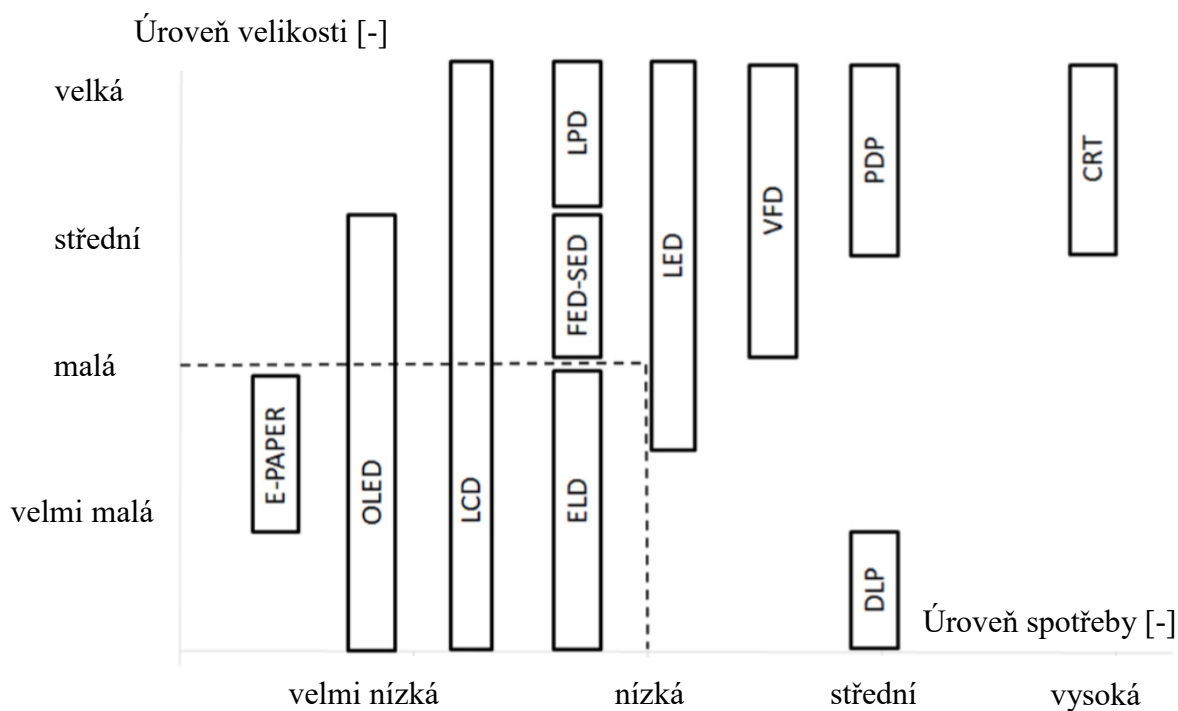
Dalším výrazným parametrem je velikost zobrazovací plochy, neboť některé displeje mají výrazný nárůst spotřeby energie s rostoucí úhlopříčkou. Podle tržního průzkumu [10], je během let 2010 až 2015 podíl displejů s menšími rozměry dvojnásobný a v následujících sedmi letech bude podíl pokračovat.

Staré CRT monitory vyžadují nejméně dvojnásobný výkon LCD panelů. Plazmové displeje (PDP) nejsou ekonomicky vhodné pro výrobu v malých rozměrech. Technologie DLP (Digital Light Processing) používána v digitálních projektorech má srovnatelný výkon s PDP technologií. Je nutné počítat se spotřebou DMD (Digital Mirror Device) čipu a světelného zdroje. LPD (Laser Phosphor Display) je technologie, kdy fosforová vrstva obrazovky je ostřelována laserovými paprsky. Protože je technologie ještě ve vývoji, nelze v provedené studii pro nedostatek informací o dostupných produktech uvádět spolehlivé hodnoty elektrické spotřeby.

Ve shrnutí jsou pro kvalifikaci displejů následující velikosti a výkony.

Tabulka 2.1: Parametry studia elektrické spotřeby

Úroveň velikosti	Úhlopříčka [palec]	Úroveň spotřeby	Osvětlení E [$\text{mW}\cdot\text{cm}^{-2}$]
velmi malá	< 1	velmi nízká	0 až 10
malá	1 až 7	nízká	10 až 100
střední	7 až 20	střední	100 až 250
velká	> 20	vysoká	> 250



Obrázek 2.2: Spotřeba různých technologií zobrazovačů [10]

Následujících podkapitolách popisují technologie displejů (zobrazovačů), které se vyrábějí v menších velikostech a mají nižší spotřebu elektrické energie.

2.3.1 LCD

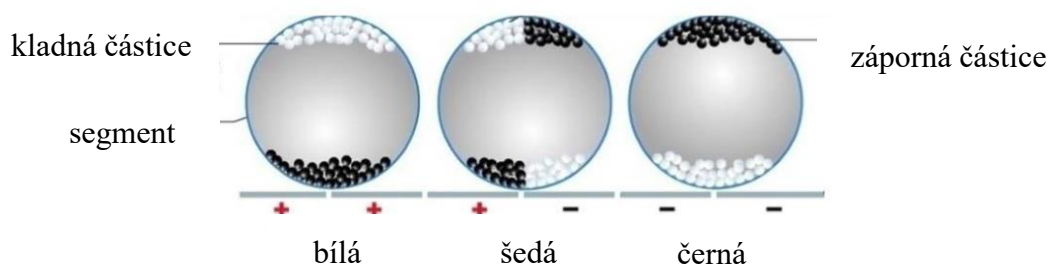
Jedná se o zobrazovací panel, který využívá vlastností kapalných krystalů (Liquid Crystal). Krystaly jsou tekuté jako kapalina, ale mají optické a elektromagnetické vlastnosti jako pevné látky. Propustnost a polarizace světla je řízena natočením těchto krystalů, které jsou vloženy mezi průhledné elektrody a polarizační filtry. Samotné natočení závisí na intenzitě elektrického pole. Větší část kapalných krystalů jsou organické sloučeniny, jejichž molekuly mají podlouhlý tvar.

Jednotka zobrazované plochy se nazývá pixel. Tento bod se skládá ze tří subpixelů, a to červeného, zeleného a modrého. Kombinací různých jasů jednotlivých subpixelů lze dosáhnout vykreslení všech barev spektra.

Elektrické napětí každého subpixel je aktivně řízeno nejméně jedním tenkým foliovým tranzistorem **TFT** (Thin Film Transistor).

2.3.2 E – Ink

Vysoká viskozita roztoku způsobuje, že mikročástice zůstávají ve své poloze bez změny prakticky natrvalo. Mít nabitou elektrodu je nutné pouze k překonání viskózních sil roztoku.



Obrázek 2.3: Princip technologie E – Ink displeje [15]

To znamená, že se jedná o ideální displej pro mobilní aplikace, protože elektrickou energii potřebuje pouze pro změnu zobrazeného obsahu. Displej odstraňuje nedostatky pozorovacích úhlů LCD technologie. [14]

Panel je reflexní, tj. nepotřebuje zdroj světla pro podsvícení, vlivem této vlastnosti má takovou malou (prakticky až žádnou) spotřebu elektrické energie. Problém nastává při špatných světelných podmínkách, v šeru a ve tmě. Tento nedostatek lze vyřešit nasvícením zobrazovací plochy. Na plochu je přiložena světlo vodivá vrstva z optického vlákna. Tato vrstva světlo jen rozvádí přes celou plochu panelu, zdrojem světla jsou LED diody, které jsou integrovány v rámečku kolem displeje. LED diody lze za nízké viditelnosti zapnout a také lze regulovat jejich výkon.

Displej je elektricky ovládán pouze za zlomek sekund, obvykle kolem (240 až 500 ms) při napětí elektrod 15 V nebo (720 až 2000 ms) při napětí 5 V. Jak již bylo zmíněno, displej vyžaduje proud pouze při obnovení obrazu. Pokud samotná aktualizace trvá dlouhou dobu (cca 1 až 2 s) nebo pokud frekvence aktualizací je příliš vysoká, může displej mít již nezanedbatelnou spotřebu.

Při aktualizaci okna vzniká tzv. ghosting. Tento výjev je účinkem zobrazení předchozího obrazu na displeji. Když je pixel aktualizován, může se stát, že se všechny částice nemohou přesunout na opačnou stranu segmentu. Tento problém lze vyřešit opakovaným zápisem stejného okna. Pravděpodobnost vzniku tohoto efektu se zvyšuje s klesajícím napětím připojeného k elektrodám a klesající teplotou pracovního prostředí.

Nemá-li zařízení používající tento displej externí zdroj vyššího napětí (cca 15 V), je nutné, aby řídicí procesor poskytl PWM signál mezi 100 až 300 kHz. Baterii nahradí nábojová pumpa, jejíž kondenzátory jsou nabíjeny zmíněným PWM signálem.

2.3.3 Memory LCD

Podobně jako elektroforetická technologie i tato technologie displejů kombinuje maticové uspořádání s jednobitovým uchováváním stavu jednotlivých pixelů. Displej je také dobře čitelný na přímém slunečním světle a pro práci v širém prostředí je potřeba zajistit podsvícení.

Výhodou této technologie je větší teplotní tolerance a rychlejší odezva zobrazení (při srovnatelné velikosti napětí) než u displejů s technologií E-ink. Navíc nedochází ke ghosting efektu, takže se nemusí provádět několikanásobný zápis stavu pixelů. Pokud obě technologie budou v pohotovostním a často v provozním režimu, má technologie Memory LCD menší spotřebu elektrické energie.

Tabulka 2.2: Typické parametry technologie E-ink a Memory LCD

Parametr	E-Ink	Memory LCD
Výrobce	Waveshare	Sharp
Model	2.9inch e-Paper	LS027B7DH01A
Rozlišení [pixel x pixel]	264x176	400 x 240
Statický příkon [μ W]	17	50
Dynamický příkon [μ W]	40 000	175
Rychlost obnovy [Hz]	-	až 20
Doba obnovy [ms]	600	30
Teplotní rozsah [°C]	0 až +60	-20 až +80
Napětí logiky [V]	3	3
Živosnost	1 mil cyklů	-

2.4 Komunikace s mikroprocesorem

Pro vykreslení obrazu musí procesor:

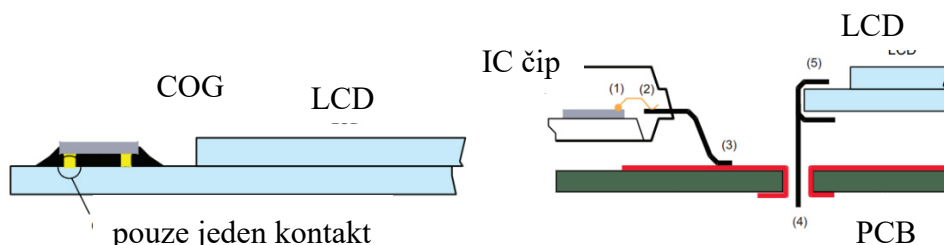
1. aktivovat a inicializovat COG,
2. zapsat obrazová data
3. vypnout řadič COG.

V první fázi komunikuje procesor s COG zasláním příkazů přes sběrnici SPI či I2C. Musí se přitom dodržet konkrétního časování, které je stanovené dokumentací COG.

Během aktualizace okna je obraz aktualizován ve čtyřech fázích:

1. vykreslující obraz je nejdříve invertován,
2. panel vykresluje prázdné okno (bílá plocha),
3. inverze obrazu (z bodu 1) je vykreslována,
4. nový snímek je zobrazen.

Dále je po dokončeném zápisu COG vypnut. I tento proces doprovází posloupnost příkazů a časování, které procesor musí dodržet. Po sekvenci vypnutí mohou být elektrody odpojeny od zdroje napětí a obraz bude nadále na panelu uložen.

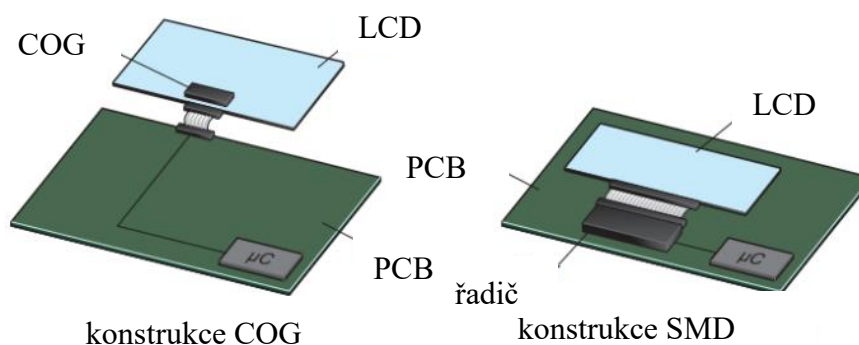


Obrázek 2.4: Konstrukce COG a SMD čipu [17]

Pro řídicí signál COG, který je integrován přímo na displeji, postačí pouze jeden kontakt. V konstrukci SMD displeje je řídicí signál veden až pěti kontakty. Označení kontaktů je následující:

- drátěné spojení z čipu (1) do olovněného rámu (2) jeho pouzdra,
- pájecí ploška DPS desky pro SMD pouzdro (3),
- THT spojení desky s panelem (4) a (5).

Při samotné aplikaci displeje je panel připojen k procesoru konektorem FPC (Flex Panel Connectors) jak je znázorněno na Obrázek 2.5: Připojení LCD panelu FPC konektorem [17] **Chyba! Nenalezen zdroj odkazů..**



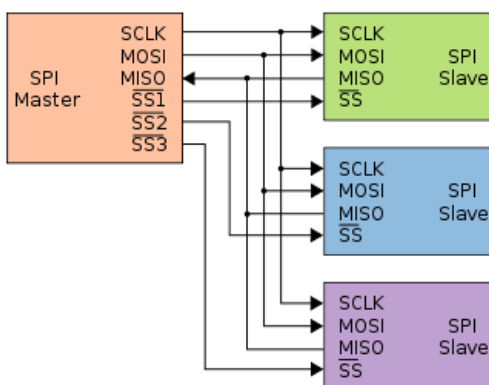
Obrázek 2.5: Připojení LCD panelu FPC konektorem [17]

COG očekává snímek ve zvláštním formátu. Každý pixel má maticovou souřadnici (X, Y), kde se počátek nachází v levém horním rohu displeje.

2.4.1 SPI

Sériové rozhraní používané pro komunikaci mezi procesorem a vnějším integrovaným obvodem (paměť EEPROM, A/D převodníky nebo displej). Může být také použit pro vzájemnou komunikaci mezi procesory nebo k programování vnitřní paměti Flash.

V systému (Obrázek 2.6: Sběrnice SPI [18]) je umožněna komunikace mezi dvěma nebo větším počtem připojených zařízení (uzlů), což je podstatný rozdíl oproti sériovému portu RS – 232C.



Obrázek 2.6: Sběrnice SPI [18]

Master je pouze jeden, uzlem tohoto typu obvykle bývá procesor (řídící člen). Komunikace je řízena pomocí hodinového signálu SCLK. Zbylé uzly typu Slave jsou adresovány pomocí vodiče (SS – Slave Select nebo CS – Chip Select).

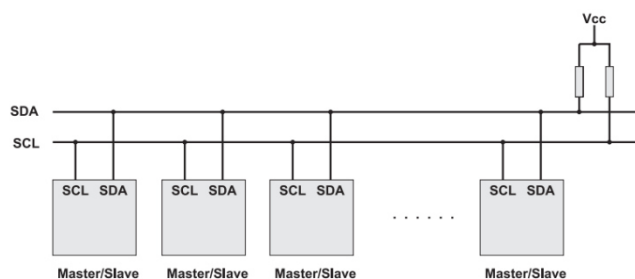
Pokud je zvolen některý z obvodů Slave, potom komunikace po sběrnici probíhá pouze mezi dvěma uzly. Směr datového toku je určen piny MOSI a MISO. Přičemž MOSI je vždy výstup z uzlu Master a Slave vstupem, podobně MISO (Master In Slave Out). Oba obvody obsahují posuvné registry, které jsou v okamžiku komunikace propojeny.

Posouvání obou registrů je řízen hodinovým signálem. Vztah mezi hodinovým a datovým signálem je dán parametry CPOL a CPHA.

2.4.2 I2C

Sběrnice je v některých vlastnostech (hodinový signál SCLK přiveden do více uzlů) podobná sběrnici SPI, ovšem jsou i odlišné. U SPI je umožněn obousměrný přenos vlivem vodičů MOSI a MISO, tato sběrnice je vybavena pouze jedním vodičem SDA. Poloduplexní přenos dat znamená složitější interní strukturu všech připojených uzlů, neboť pin musí zastávat funkci vstupu a výstupu. Adresování není realizováno výběrovým logickým signálem, ale jednotlivé uzly mají svou jednoznačnou adresu. Každá adresa má velikost sedmi nebo deseti bitů, která slouží k výběru i k arbitraci. Rozhraní tedy stanovuje nejen elektrické parametry, ale definuje i komunikační protokol.

Toto rozhraní je sice komplexní, ale flexibilitnost umožňuje komunikaci na delší vzdálenost. Na rozdíl od rozhraní SPI je I2C typu multimaster, tj. každý z uzlů je buďto typu Master nebo Slave.



Obrázek 2.7: Komunikace pomocí I2C [20]

3 EPSNet

Měřicí modul komunikuje s motherboardem pomocí protokolu EPSNet, který je především používán pro komunikaci komponentů (servomotor, PC ...) s PLC systémy od firmy Teco a.s. sídlící v Kolíně.

Jde o modifikaci protokolu **Profibus DP**, který dle **ISO/OSI** modelu definuje tři vrstvy (třetí až šestá vrstva je prázdná):

- první Fyzickou vrstvu,
- druhou Linkovou vrstvu,
- sedmou Aplikační vrstvu

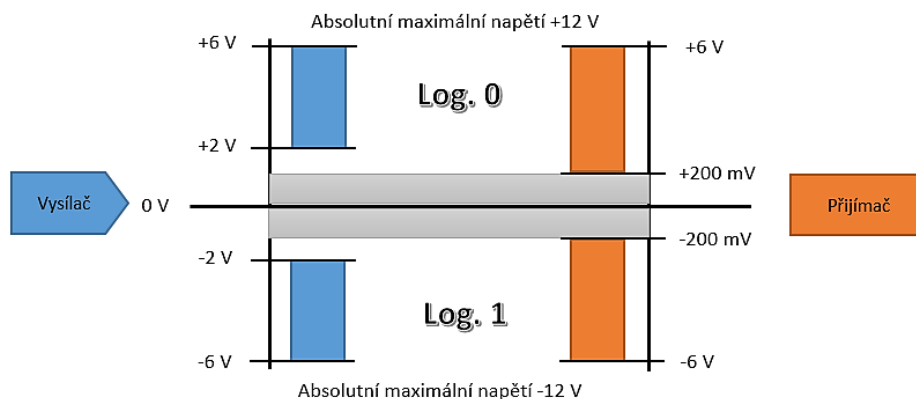
3.1 Fyzická vrstva

Specifikuje použití přenosového media (**kroucená dvojlinka**, optické vlákno, koaxiální kabel atd.), připojovací konektory a topologii sítě. Dále určuje kódování a modulaci přenášených dat včetně elektrických úrovní.

Data jsou po sériové lince v síti EPSNet digitálně přenášena pomocí rozhraní **RS485**. Vedle tohoto sériového standardu jsou definovány i standardy RS232 a RS422, ale v průmyslovém prostředí je nejčastěji používán, neboť využitím vícebodové topologie je umožněno zapojit více přijímačů i vysílačů. Vlivem diferenciálních vstupů a výstupů se do obou žil kroucené dvojlinky indukují rušivé napětí stejně, takže je možný přenos i při větším elektromagnetickém rušení a na větší vzdálenosti. Základní rozdíly těchto standardů jsou shrnuty v tabulce 3.1:

Tabulka 3.1: Základní rozdíl sériových standardů

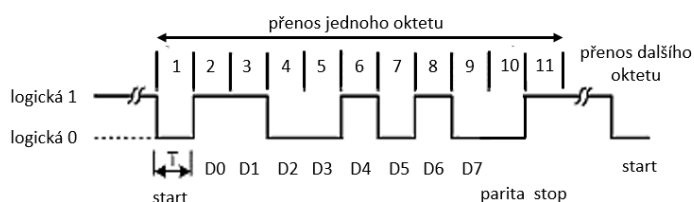
Název	RS232	RS422	RS485
Režim přenosu	poloduplex plný duplex	poloduplex (plný duplex)	poloduplex
Max. počet vysílačů	1	1	32 (bez opakovače)
Max. počet přijímačů	1	10	32 (bez opakovače)
Topologie sítě	dvoubodový	dvoubodový	vícebodový
Max. vzdálenost	15 m	1200 m	1200 m
Max. rychlost při 12 m	20 kb/s	10 Mb/s	35 Mb/s
Max. rychlost při 1200 m	1 kb/s	100 kb/s	100 kb/s



Obrázek 3.1: Napětová úroveň sběrnice RS485

K přenosu dat dochází v rozhraní RS485 v digitální podobě, za použití logických 0 a 1, které odpovídají určitým napětovým úrovním. Specifikace napětové úrovně vysílače a přijímače je znázorněno na obrázku 3.1.

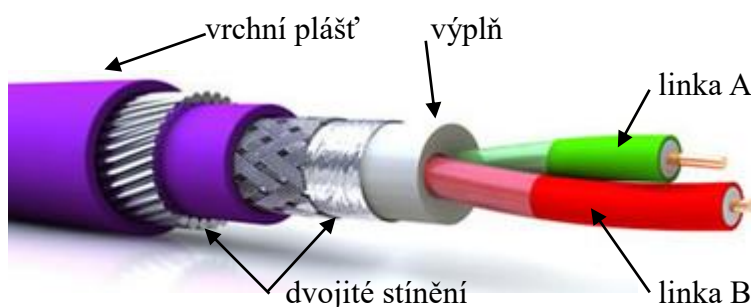
Zařízení RS485 (vysílač/přijímač) komunikuje s procesorem pomocí rozhraní USART připojením čtyř vodičů (TxD, RxD, DE a $\overline{\text{RE}}$). Data jsou přenášena rozhraním v oktetu, specifikace oktetu pro EPSNet je znázorněn na obrázku 3.2.



Obrázek 3.2: Oktet protokolu EPSNet

Synchronizace přenosu je zajištěna start bitem do logické 0, který předchází osm datových bitů D0 (LSB) až D7. Pro kontrolu správného přenosu se používá sudá parita. Oktet je ukončen jedním stop bitem, který má opačnou úroveň než start bit (logická 1).

Pro přenos se nejčastěji používá kroucená-stíněná dvojlinka, přičemž stínění slouží zároveň jako uzemnění. Kabel nesmí být natahován, stlačován ani výrazně ohýbán, neboť to může vést ke změnám struktury a následně k odrazu signálu. Existují dva druhy kabelu, linka A a linka B.



Obrázek 3.3: Průřez průmyslového kabelu kroucené dvojlinky [21]

Zařízení se ke sběrnici RS485 může připojit pomocí konektoru D-typu (D-Sub), v drsnějších průmyslových podmínkách pomocí zástrčky M12 a levnější zařízení přímo pomocí svorkovnice (např. modul k vývojové desce). Konektor D-typu, který je tradičně používán u sběrnice RS232, se doporučuje používat pro zařízení v rozvaděčích.

Každý moderní konektor má integrovaný přepínač pro zapnutí či vypnutí ukončovacího odporu.

3.2 Linková vrstva a Aplikační vrstva

Linková vrstva definuje způsob směrování a adresování zpráv v síti. Doplnuje data o záhlaví, adresaci a zabezpečení detekčním kódem. Při používání sítě se uživatel pohybuje v aplikační vrstvě, pomocí aplikací zpracovává a řídí přenos dat.

V síti EPSNet mohou komunikovat dva druhy stanic:

- **nadřazená stanice (master)** - aktivní účastník komunikace, kterou vyvolává a řídí
- **podřazená stanice (slave)** – pasivní účastník, který pouze reaguje na dotazy nadřazené stanice

Komunikace je inicializována na principu **dotaz – odpověď**, který umožňuje připojit větší počet stanic. Tento počet je omezen zpravidla přenosovým médiem na **maximálních 127**. Rozložení počtu nadřazených a počtu podřazených stanic, které sdílejí stejnou síť, je dáno konfigurací této sítě. [24]

Dotaz v síti **EPSNet monomaster** vysílá jediná nadřazená stanice a jedna ze 126-ti (možných) podřazených stanic musí na tento dotaz odpovědět. Pokud je požadavek na výměnu dat mezi dvěma podřazenými stanicemi, musí předávání dat řídit nadřazená stanice. Nadřazenou stanicí může být systém TECOMAT/ TECOREG se sériovým kanálem v režimu **MAS** či v režimu **MPC** s vypnutým přenosem zprávy token či PC s terminálem. Podřazené stanice mají sériový kanál v režimu **PC**. [24]

Konfigurace **multimaster** poskytuje připojení několika nadřazených stanic (master) a několik podřazených stanic (slave), kombinace stanic je však limitována na maximální počet 127. I v tomto případě síť řídí v daném okamžiku pouze jedna nadřazená stanice, která přijala zprávu token. Všechny ostatní nadřazené stanice se vůči této stanici chovají jako podřazené stanice. To umožňuje přenos dat mezi libovolnou stanicí (podřazenou i nadřazenou). Po vyřízení požadavků předává řídicí stanice zprávu token další nadřazené stanici a od tohoto momentu se chová jako podřazená stanice. [24]

Po přijetí řízení začíná nový proces komunikace pomocí zpráv a odpovědí, které mají následující strukturu [24]:

- 1) zpráva bez datového pole

SD1	DA	SA	FC	FCS	ED
------------	-----------	-----------	-----------	------------	-----------

- 2) zpráva s datovým polem

SD2	LE	LER	SD2R	DA	SA	FC	DATA	FCS	ED
------------	-----------	------------	-------------	-----------	-----------	-----------	-------------	------------	-----------

- 3) zpráva token – předání řízení sítě (bez odpovědi, odpovědí je převzetí řízení)

SD4	DA	SA
------------	-----------	-----------

A následně podřízená stanice (slave) reaguje odpovědí ve tvaru:

- 1) odpověď bez datového pole

SD1	DA	SA	FC	FCS	ED
------------	-----------	-----------	-----------	------------	-----------

- 2) odpověď s datovým polem

SD2	LE	LER	SD2R	DA	SA	FC	DATA	FCS	ED
------------	-----------	------------	-------------	-----------	-----------	-----------	-------------	------------	-----------

- 3) odpověď bez datového pole – krátké pozitivní potvrzení

SAC

Hodnoty konstant a význam znaků jsou uvedeny v příloze B.

Nadřízená stanice při komunikaci musí dodržet, že mezi vysílanými byty do stanice podřízené musí být kratší prodleva než doba potřebná pro vyslání 33 bitů. Dále mezi přijatou odpovědí a vyslanou další zprávou musí klid na lince delší než doba přenosu 33 bitů, tímto klidem se synchronizují přijímače všech stanic před zahájením nového přenosu. Pokud je síť v režimu multimaster platí navíc, že nadřazená stanice vysílá až po uplynutí nejdelšího timeoutu (doba mezi dotazem a odpovědí od podřízené stanice), který je závislý na adrese nadřazené stanice podle vzorce [24]:

$$\text{Doba klidu} = \text{timeout} + 500 + 10 \cdot \text{adresa} \text{ [ms]} \quad (3.1)$$

Během přenosu je zpráva i odpověď zabezpečena sudou paritou, jak už bylo uvedeno. Dále hodnotou kontrolního součtu FCS, která musí odpovídat skutečnosti a hodnoty znaků musejí dodržet správnou sekvenci. Pokud jsou data přenášena přes modemy, které nepodporují přenos parity, je umožněno paritu vypnout. K udržení bezpečnosti přenosu dat je provedeno doplňkové zabezpečení pomocí 16-ti bitovým **CRC** polynomem ve tvaru [24]:

$$X^{16} + X^{15} + X^2 + 1 \quad (3.2)$$

Pokud jsou data správně přenesena, pak po spočítání hodnoty CRC přes přenesena data včetně jejich hodnoty CRC (za datovým blokem), výsledek výpočtu je 0. Platnost přenesených dat a správné provedení služby znamená pozitivní odpověď. V opačném případě podržovaná stanice vysílá negativní odpověď⁴, která může být trojího druhu. Jestli řídicí jednotka stanice tuto komunikační službu nezná (FC = 2), nebo není služba v okamžiku komunikace aktivována (FC = 3), nebo požadovaná data nejsou ještě připravena na přenos (FC = 9) a nebo služba má chybné parametry, které nelze správně interpretovat (FC = 12). [24]

3.3 Porovnání s ostatními protokoly

3.3.1 CANopen a DeviceNet

Dva protokoly, které jsou standardizovány na sběrnici CAN. Původním účelem této sběrnice je zajistit komunikaci komponent v osobních automobilech, nyní vlivem firmy Bosh nachází uplatnění i mimo něj. Systém umožňuje sériovou komunikaci s vysokým stupněm spolehlivosti s rychlostí přenosu až 1 Mb/s. Protokol CANopen podporuje 127 komunikačních jednotek, zatímco DeviceNet umožňuje připojit 64 síťových uzlů.

3.3.2 Modbus (RTU/ASCII)

Dost rozšířený a jednoduchý protokol, který je postaven na lince RS232 nebo RS485. Pro komunikaci je nutné znát přenosovou rychlost, paritu, stop bit a adresy uzlů. Sběrnici může sdílet až 247 slavů, přičemž adresu 0 má master. Rychlost přenosu je až 115 kb/s.

Data v protokolu Modbus RTU jsou přenášena binárně a protokol Modbus ASCII kóduje data pro přenos do ASCII znaků.

3.3.3 Profibus (DP/PA)

Jak již bylo uvedeno, protokol **EPSNet** je modifikací protokolu Profibus (DP). Systém je vyvinut s cílem sjednotit dříve nekompatibilní přenosové technologie:

- symetrická sériová linka RS485, stíněný kroucený pár,
- upravená verze RS485-IS, čtyřvodičový přenos určený pro potenciálně výbušná prostředí,
- přenos dat optickým vláknem na velké vzdálenosti s malými útlumy přenosových cest

Profibus DP v průmyslové automatizaci připojuje řídicí automaty, vzdálené I/O jednotky, frekvenční měniče apod. Zařízení připojená k dvojvodičové RS485 mají svoje vlastní napájení, rychlost datového přenosu je různá (např. 500 kb/s, 1,5 Mb/s, 6 Mb/s).

V procesní automatizaci (především v petrochemických provozech) se používá Profibus PA. Data jsou přenášena na velké vzdálenosti pouze jednou rychlostí 31,25 kb/s a zařízení je napájeno pomocí vodičů komunikační sběrnice.

Tabulka 3.2: Porovnání protokolů

Parametry	CAN	Modbus	Profibus
Maximální počet síťových uzlů	127	247	127
Maximální rychlost přenosu [Mb/s]	1	0,112	12
Maximální vzdálenost [m]	5000 (10 kb/s)	1200 (93,75 kb/s)	1200 (93,75 kb/s)
Metoda přístupu	CSMA/CD	Master-Slave	Master-Slave
Pořadí bitů	MSb->LSb	LSb->MSb	LSb->MSb

Modbus je snadno implementovaný a volný protokol (open source), ale množstvím variací v samotném protokolu a jeho definici fyzické vrstvy způsobuje problémy v aplikacích s více dodavateli. Profibus velmi dobře pracuje s produkty od různých dodavatelů, s modemy a má podrobnou diagnostiku. Pokud tedy existuje vícebodové připojení nebo nebezpečné prostředí, je Profibus lepším řešením.

CAN umožňuje delší vzdálenost vedení, ale vlivem přístupu CSMA/CD má menší rychlost dat během komunikace než Profibus. V maximálním počtu připojitelných stanic jsou oba srovnatelné.

EPSNet používá pouze bitové (ne znakové) kódování přenosu, tím se zvyšuje kompatibilita komunikace mezi různými dodavateli. Výhodou je otevřenost protokolu, veřejně přístupný pro každého, bez nutnosti platit za jeho využívání nějaké poplatky a s metodou přístupu Master-Slave dosahuje velkých rychlostí přenosu. Nevýhodou je, že může existovat různých implementací.

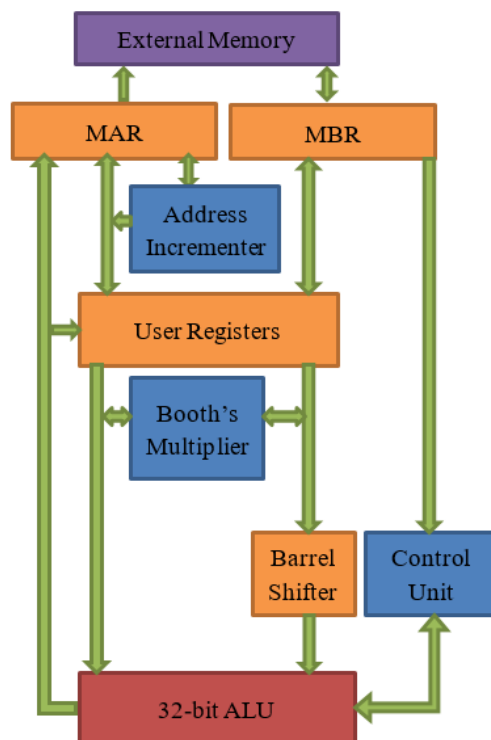
4 Řídicí procesor ARM

Procesor této konstrukce je znám již řadu let, ale až příchod současné generace mobilních zařízení (smartphone, tablet atp.) přinesla revoluci s jejich vyšším výkonem a nižší spotřebou. V následujících podkapitolách je popsána historie a struktura této architektury.

4.1 Popis architektury

4.1.1 Jádro

Jádro ARM procesorů je znázorněno na blokovém schématu (Obrázek 4.1: Blokové schéma ARM jádra). Blokem **External Memory** jsou shrnuty registry vnějších periférií nebo datové paměti. Registr **MAR** uchovává odkaz na adresu v datové paměti, kam následně budou informace uloženy nebo z paměti načteny. **MBR** nebo někdy MDR je registr pro data určená ke zpracování instrukce. Pokud jsou data načtena z paměti, ukládají se nejprve do tohoto registru a poté jsou následně zpracována. Nebo jsou data kopírována z jiného vnitřního registru (**User Registers**) do registru MBR a poté odeslána do vnější paměti. Registr působí jako vyrovnávací paměť, umožňující procesorovým a paměťovým jednotkám pracovat nezávisle, aniž by byly ovlivněny v rozdílných rychlostech.



Obrázek 4.1: Blokové schéma ARM jádra

4.1.2 Instrukční sada

Instrukce má velikost 32 bitů, což odpovídá jednomu slovu (word). Aby došlo k úspoře dat v programové paměti je zavedena (u ARM7 a vyšších) instrukční sada THUMB (16 bitů), která je podmnožinou klasické instrukční sady ARM a obsahuje nejčastější instrukce. Navíc použitím této sady instrukcí je docíleno úsporného režimu, jelikož se budí pouze 16 bitů sběrnice.

Architektura mikroprocesorů **RISC** je úzce svázána s architekturou load/store. U této architektury je instrukční sada redukována pouze na základní a nejčastěji používané instrukce. V programové paměti je zakódován menší počet instrukcí oproti architektuře **CISC**. Ale počet provedených instrukcí RISC procesorů může být paradoxně větší, jak se ukázalo na příkladu operace součtu. U CISC procesorů je operace součtu provedena jedinou instrukcí, ale jedná se o specifickou instrukci. Těchto instrukcí je nakonec mnoho a programová paměť zabírá větší prostor.

4.2 Procesory řady STM32

Procesory firmy STMicroelectronics jsou rozděleny do tří sérií:

- 1) Standardní série: STMF0, STM32F1 a STM32F3
- 2) Výkonová série: STM32F2, STM32F4 a STM32F7
- 3) Nízkopříkonová série: STML0, STM32L1, STM32L4 a STM32L5

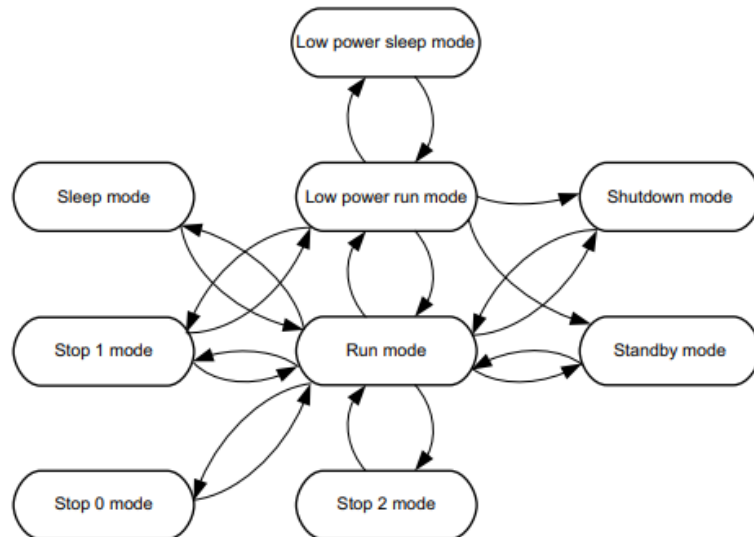
Jelikož je modul realizován s požadavkem na nízkou spotřebu, je použitý procesor STM32 série L4. Také proto následující text popisuje pouze tuto sérii.

Low-power série nabízí dynamicky stupňované napětí, periférie s nízkou spotřebou dostupné ve STOP režimu (např. LP-UART či LP-timer). Dále analogové periférie s nízkou spotřebou jako operační zesilovače, komparátory, 12-bitový DA převodník nebo 16-bitový AD převodník. [27]

LPUART má v porovnání s UART méně funkcí, je však schopen pracovat s nižším výkonem a má efektivnější využití LSE hodin. Mezi omezení patří absence Smartcard režimu, automatické detekce modulační rychlosti, LIN modu a IrDa režimu. Velké úspory lze dosáhnout, pokud je použit režim přímého vstupu do paměti DMA, procesor poté může část svého cyklu zůstat v režimu Sleep mode. Mnohdy nelze DMA režim kombinovat s režimem STOP, proto jsou před vstupem do tohoto režimu DMA kanály zakázány.

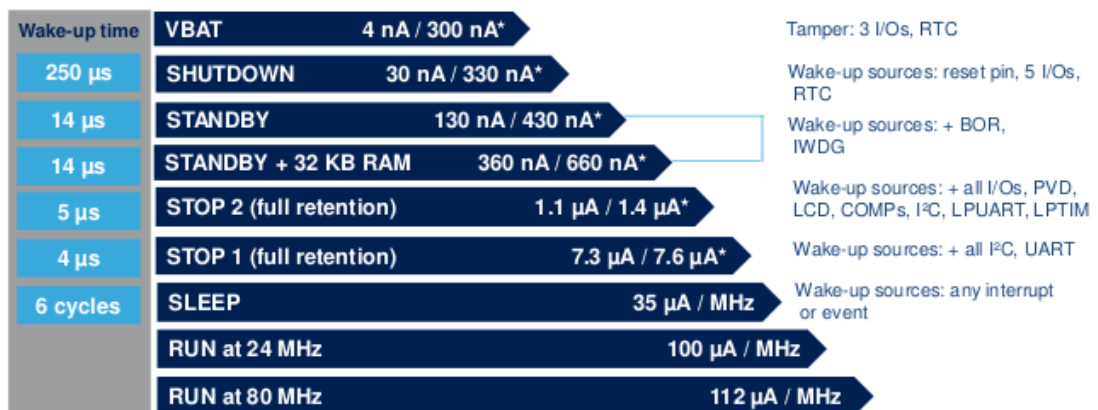
LP-timer je 16-ti bitový časovač, který běží ve všech režimech procesoru, kromě režimu Standby. Časovač lze použít pro čítání vnitřních nebo vnějších pulzů a pro časové probuzení z režimů nízké spotřeby.

Po resetu (zapnutí) je mikroprocesor v režimu Run, podle uživatelského nastavení lze následně vstoupit až do osmi režimů se sníženou spotřebou (Sleep, Low-power run, Low-power sleep, Stop 0, Stop 1, Stop 2, Shutdown a Standby)



Obrázek 4.2: Pracovní režimy procesoru STM32L496RGx [33]

Na níže uvedeném obrázku jsou zobrazeny přibližné hodnoty odebíraného proudu ve všech dostupných režimech procesoru v nízkopříkonové sérii L4.



Note : * without RTC / with RTC

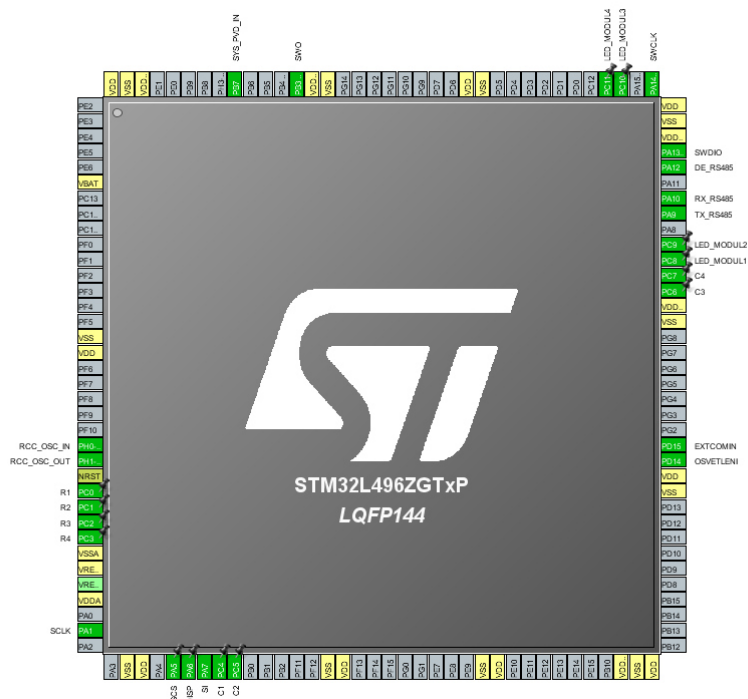
Obrázek 4.3: Orientační odběr proudů [27]

5 Praktická část

Cílem této práce je navrhnout a zhotovit prototyp inteligentního displeje k měřicí ústředně GU100. Modul displeje je napájen ze zdroje stejnosměrného napětí +12 V, který může být realizován baterií nebo spínaným zdrojem z motherboardu měřicí ústředny. Toto napětí je dále upraveno pomocí snížovacího (step-down) spínaného zdroje na +5 V, upravená hodnota napětí je určena pro napájení displeje a lineárního zdroje napětí +3,3 V. Výstup lineárního zdroje je přiveden na řídicí procesor a další pomocné obvody (např. EEPROM, modul RS485).

5.1 Program řídicí jednotky

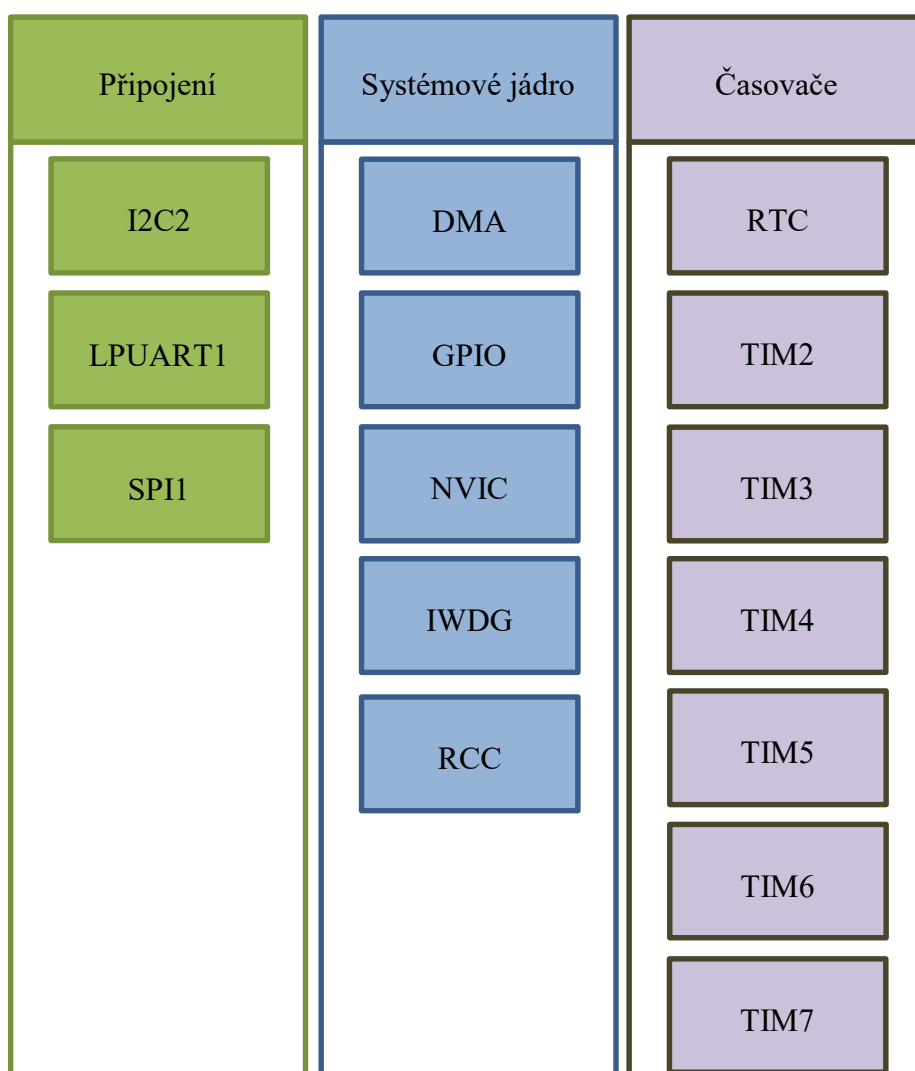
Pro vývoj programu jsem použil vývojový kit NUCLEO-L496ZG-P od firmy STMicroelectronics. Pro inicializaci firmwaru řídicí jednotky STM32L496 jsem použil program STM32CubeMX. Program umožňuje systematicky nastavit frekvenci hodinového signálu (*SYSCLOCK*) nebo základní parametry komunikačních kanálů (UART, SPI či I2C). Nakonec je možné pojmenovat, mapovat a nastavit funkce jednotlivých pinů všech paralelních portů.



Obrázek 5.1: Pinout přehled použitých pinů na vývojové desce NUCLEO-L496ZG

Hardwarové prostředky pro řídicí program lze rozdělit do funkčních celků, grafické schéma tohoto rozdělení je na obrázku Obrázek 5.2: Hardwarové prostředky řídicího programu. Rozhraní I2C2 slouží pro uložení dat do paměti EEPROM, LPUART1 komunikuje s modulem RS485 a přes SPI1 jsou přenášena data pro zobrazení na grafickém displeji. Snížené výpočetní náročnosti procesoru je dosaženo použitím obvodu DMA. Blok GPIO definuje vlastnosti použitých paralelních portů (přesněji jejich pinů) a NVIC definuje systémová a uživatelská přerušení.

Pro zajištění bezpečného chodu procesoru bez jeho zacyklení je použit systém IWDG. Časovač TIM4 slouží pro PWM modulaci a časovač TIM3 k vypnutí modulu. TIM2, TIM5, TIM6 i TIM7 k odpočítávání intervalů během komunikace přes protokol EPSnet. Obvod RTC aktualizuje reálný čas, který je při zapnutí modulu z motherboardu synchronizován. Pro práci s časem je nutné zajistit zdroj hodinového signálu, jehož vlastnosti specifikuje blok RCC.



Obrázek 5.2: Hardwarové prostředky řídicího programu

Dále v textu je řídicí firmware modulu rozdělen na popis komunikace uživatele s řídicím procesorem (pomocí klávesnice a displeje) a komunikace procesoru s motherboardem/měřicími moduly (pomocí rozhraní RS485).

Modul je obsluhou řízen pomocí maticové klávesnice 4x4 a výsledky interakce jsou zobrazeny na monochromatickém grafickém displeji.

5.1.1 Klávesnice

Piny *R1* až *R4* jsou nastaveny jako vstupy s funkcí *Pull-Up* (připojení přes odpor k napětí +3,3 V) a slouží k detekci stisknutého tlačítka. Událost stisknutí tlačítka je zachycena změnou napěťové úrovně, neboť klidový stav těchto vstupů je logická 1 (tj. +3,3 V) a stisknuté tlačítko přivede na příslušný vstup logickou 0 (tj. GND).

Piny *C1* až *C4* jsou použity jako výstupy s funkcí *Push-Pull*. Logická 0 (hladina *Low*) na výstupu je nastavena pomocí registru BSRR. Jde o 32-ti bitový registr, kde zápis logická 1 do prvních 16-ti bitů slouží pro nastavení hladiny *Low* výstupních pinů. Zápisem logická 1 do druhé poloviny registru je nastavena hladina *High* výstupních pinů stejného portu. Pro vyšší bezpečnost vyhodnocení stisknutého tlačítka je v jednom okamžiku nastavena hladina *Low* pouze jednoho výstupního pinu (např. *C1*) a zbylé tři piny (*C2*, *C3* a *C4*) jsou nastaveny na hladinu *High*.

Pro odfiltrování nežádoucích překmitů sepnutého kontaktu tlačítka slouží funkce *read_keypad()*. Pokud tato funkce zjistí, že se hodnota stlačeného tlačítka bez přerušení několikrát opakuje (dáno konstantou *FILTR_KONST*), pak tuto hodnotu tlačítka vrátí pouze jednou.

5.1.2 Displej

Specifikace vykreslování základních grafických objektů (geometrických obrazců či textových řetězců) a komunikace displeje po rozhraní SPI je implementována v hlavičkovém souboru *Sharp.h* (resp. ve zdrojovém souboru *Sharp.c*).

Jelikož naprogramované grafické rozhraní s displejem E-paper od Waveshare bylo skoro hotové a displej od Sharpu nemá od výrobce žádný oficiální firmware, proto je pro vykreslení grafických obrazců a původních fontů použita implementace od Waveshare [29].

Řídicí program pro komunikaci a zapnutí displeje je naprogramován dle datasheetu od výrobce [24]. Inicializace sběrnice SPI je přednastavena pomocí STM32CubeMX a uložena ve zdrojovém souboru *SPI.c*:

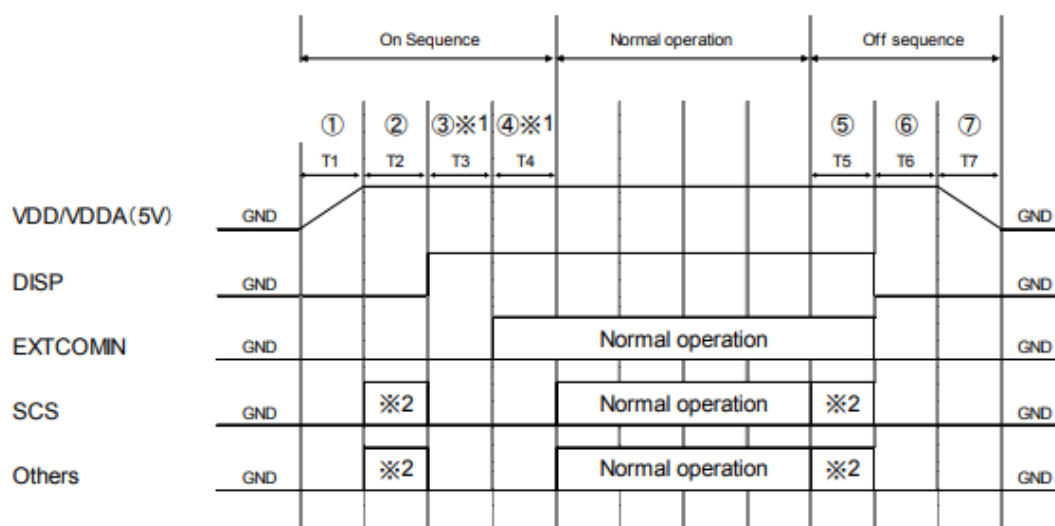
```
/* SPI1 init function */
void MX_SPI1_Init(void)
{
    hspi1.Instance = SPI1;
    hspi1.Init.Mode = SPI_MODE_MASTER;
    hspi1.Init.Direction = SPI_DIRECTION_1LINE;
    hspi1.Init.DataSize = SPI_DATASIZE_8BIT;
    hspi1.Init.CLKPolarity = SPI_POLARITY_LOW;
    hspi1.Init.CLKPhase = SPI_PHASE_1EDGE;
    hspi1.Init.NSS = SPI_NSS_SOFT;
    hspi1.Init.BaudRatePrescaler = SPI_BAUDRATEPRESCALER_8;
    hspi1.Init.FirstBit = SPI_FIRSTBIT_LSB;
    hspi1.Init.TIMode = SPI_TIMODE_DISABLE;
    hspi1.Init.CRCCalculation = SPI_CRCCALCULATION_DISABLE;
    hspi1.Init.CRCPolynomial = 10;
    if (HAL_SPI_Init(&hspi1) != HAL_OK)
    {
        Error_Handler();
    }
}
```

Zdrojový kód: 5.1: Inicializace SPI1

Během přenosu dat je nutné periodicky měnit polaritu tranzistorů, které jsou určeny pro změnu stavu obrazové buňky (tzn. pixelu). Moment změny polarity může být implementován v přenášených datech (softwarově) nebo může být určen pomocí signálu *EXTCOMIN* (hardwarově). Kvůli jednoduchosti přenosu dat je použit druhý způsob. Proto je dále implementován čtvrtý kanál timeru *TIM4*, jehož výstupem je periodický signál s frekvencí 1 Hz.

Před zahájením přenosu dat je povoleno časování čtvrtého kanálu timeru *TIM4*. Dále je povoleno časování sběrnice *SPI1* a nastaven výstup signálu *SCS* na logickou 1. Data grafického obsahu jsou přenášena z dvou rozměrného pole o velikosti 240 x 40 bytů

(řádky x sloupce), neboť velikost displeje je 240 x 320 pixelů (výška x šířka) a stav každého pixelu je zastoupen jednobitovou hodnotou (nejsou zobrazeny stupně šedi). Nakonec je přenos ukončen nastavením logické nuly na signálu *SCS*, navíc je zakázáno časování sběrnice *SPI1* a čtvrtého kanálu timeru *TIM4*. Tímto způsobem je možné ušetřit spotřebu elektrické energie v okamžiku, kdy data nejsou přenášena a je pouze potřeba na displeji zachovat zobrazený obsah.



Obrázek 5.3: Časový diagram displeje Sharp [28]

Proces zapnutí, vypnutí displeje a přenosu dat je zobrazeno na výše uvedeném časovém diagramu. Cyklus *T1* je náběh napájecího napětí +5 V a během cyklu *T2* dochází k inicializaci pixelů displeje (zápisem logických 0). Cyklus *T3* a *T4* je určen k inicializaci a generování periodického signálu (pokud je použitý hardwarový způsob inverze *TCOM*). Po ukončení zápisu (resp. celkové obsluhy) se provádí inicializace pixelů, stejným způsobem jako v cyklu *T2*. Nakonec před odpojením displeje od zdroje napětí (cyklus *T7*) je nutné během cyklu *T6* počkat, až signál *EXTCOMIN* odezní.

5.1.3 Grafické rozhraní

Toto rozhraní je implementováno v hlavičkovém souboru *GUI.h* a zdrojovém souboru *GUI.c*, ve kterém je definována podoba všech zobrazených oken. Samotná logika interakce s obsluhou se nachází v nekonečné smyčce *while()*, která je obsažena v hlavní funkci *main()*. Základem logiky je větvená podmínka *if()-else if()* a blok *switch()*. Nejprve se zjistí hodnota aktuální strany zobrazené na displeji a stav klávesnice. Podmínky kontrolují aktuální zobrazenou stranu a stisknutí tlačítka. Pokud je podmínka splněna, program vstupuje do bloku *switch()* s hodnotou znaku stisknuté klávesy. Stav grafického rozhraní je tedy reprezentován hodnotou aktuální zobrazené stránky a přechody do dalších stavů jsou podmíněny hodnotou znaku stisknutého tlačítka.

Tento modul slouží především pro zobrazení hodnot měřicích modulů, a proto se v celém programu pracuje s datovou strukturou *mericiModul*:

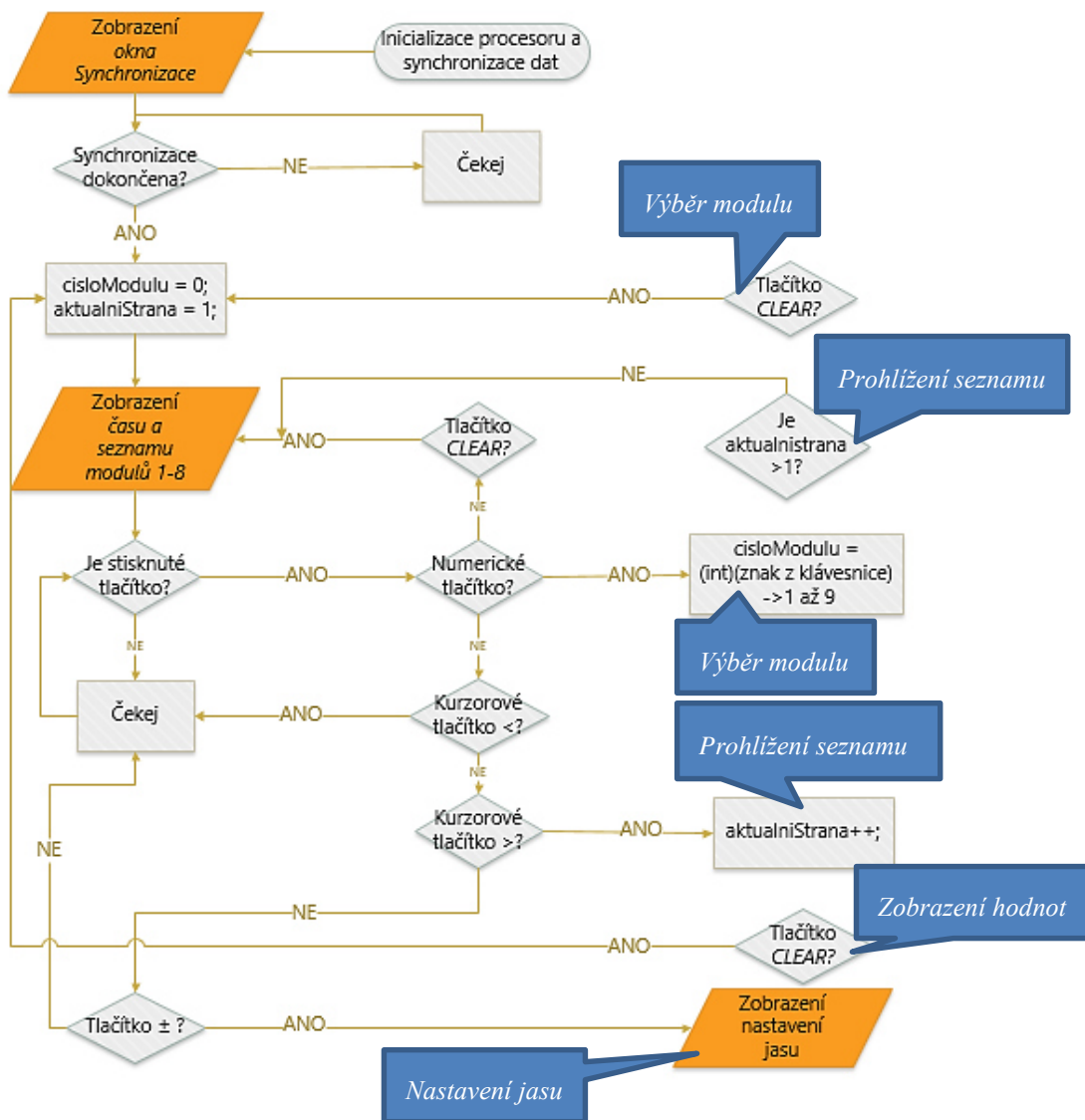
```
typedef struct {
    uint8_t adresa;
    char* nazev;
    char* jednotka;
    float
seznamHodnot[9];
    bool adresaObsazena;
} MericiModul_t;
MericiModul_t mericiModul;
```

Zdrojový kód: 5.2: Datová struktura reprezentující měřicí modul

Identifikátorem měřicího modulu v programu je dekadická hodnota adresy v proměnné *adresa*, pro zobrazení typu modulu slouží řetězec znaků *nazev* a pro zobrazení fyzikální jednotky řetězec znaků *jednotka*. Naměřené hodnoty jsou pak uloženy v poli *seznamHodnot*.

Stav inicializace je čekání, dokud nebude dokončena inicializace displeje a synchronizační komunikace modulu s motherboardem (synchronizace času, předání tokenu a seznamu zapojených měřicích modulů na jednotlivých adresách). Před zahájením synchronizace s motherboardem je název modulu inicializován hodnotou *Není k dispozici*. Tato hodnota se změní pouze, pokud je daná adresa obsazená a je pro danou adresu definován název a jednotka v podobě řetězce znaků.

Po inicializaci modulu se zobrazí obrazovka *Seznam modulů*, kde je uveden synchronizovaný čas a výčet prvních osmy definovaných/nedefinovaných měřicích modulů. Vlevo je pořadové číslo (= hodnota adresy) modulu a následně název definovaného měřicího modulu.

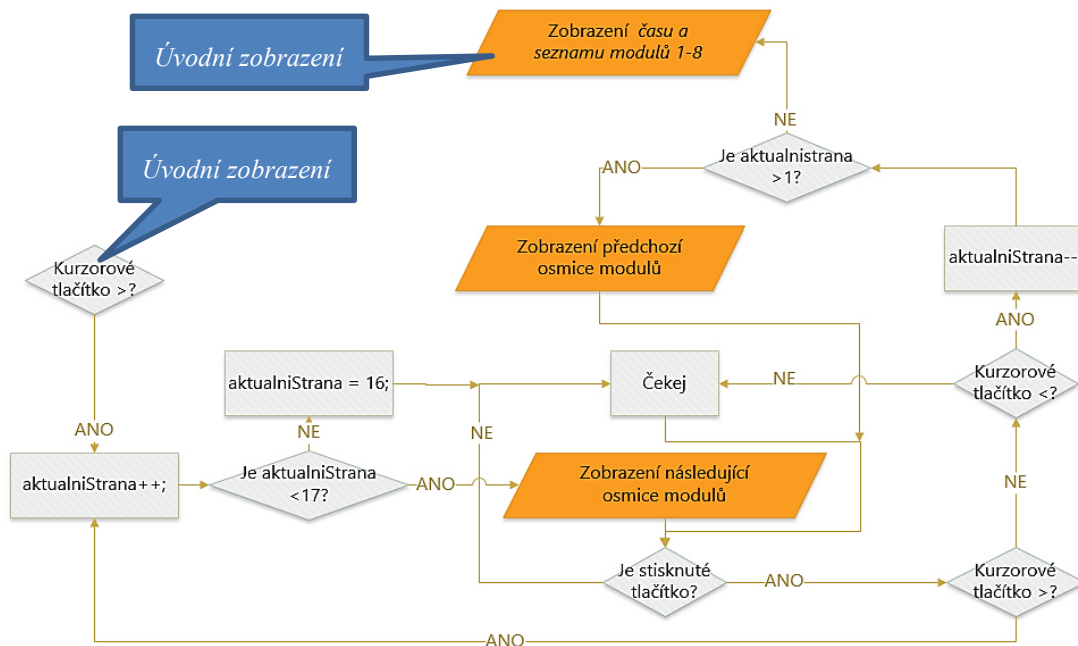


Obrázek 5.4: Část vývojového diagramu pro uživatelské rozhraní – část Úvodní zobrazení. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.

1	Teploty
2	Proudové smyčka 4-20 mA
3	Napeti 0-10 V
4	Impulzni signaly
5	Neni k dispozici
6	Neni k dispozici
7	Neni k dispozici
8	Neni k dispozici
09:12:00 12.04.20	

Obrázek 5.5: Obrazovka Seznam modulů

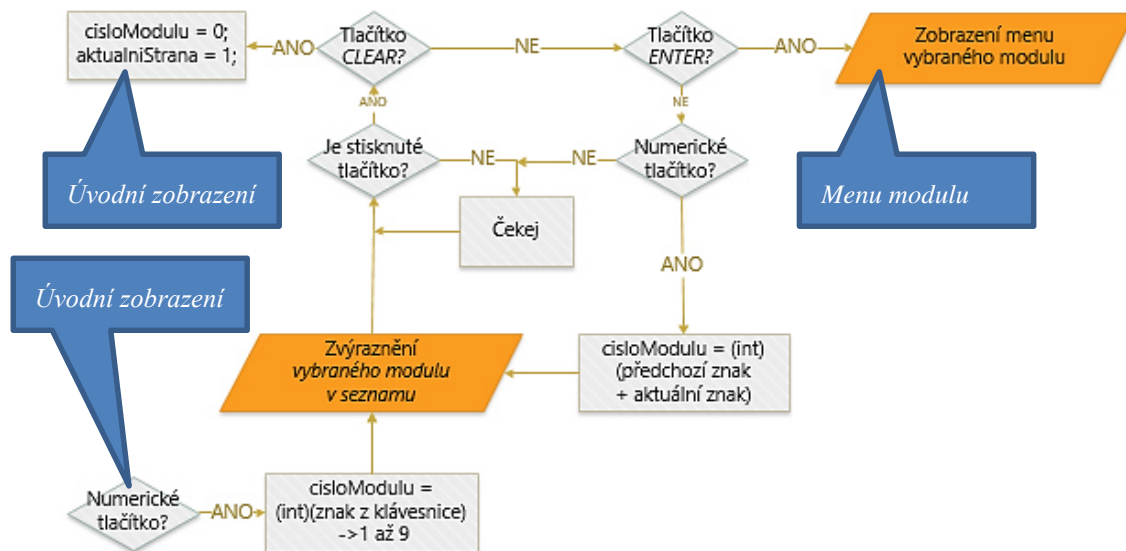
Stisknutím tlačítka </> se přechází na předchozí/další stranu s jinou osmicí modulů. Tato funkce slouží pouze k prohlížení celého seznamu bez volby modulu.



Obrázek 5.6: Část vývojového diagramu pro uživatelské rozhraní – část Prohlížení seznamu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.

9	Neni k dispozici
10	Neni k dispozici
11	Neni k dispozici
12	Neni k dispozici
13	Neni k dispozici
14	Neni k dispozici
15	Neni k dispozici
16	Neni k dispozici
09:14:03 12.04.20	
Tlacitko zpet 'CLEAR'	

Obrázek 5.7: Obrazovka Seznam modulů – následující osmice



Obrázek 5.8: Část vývojového diagramu pro uživatelské rozhraní – část Výběr modulu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.

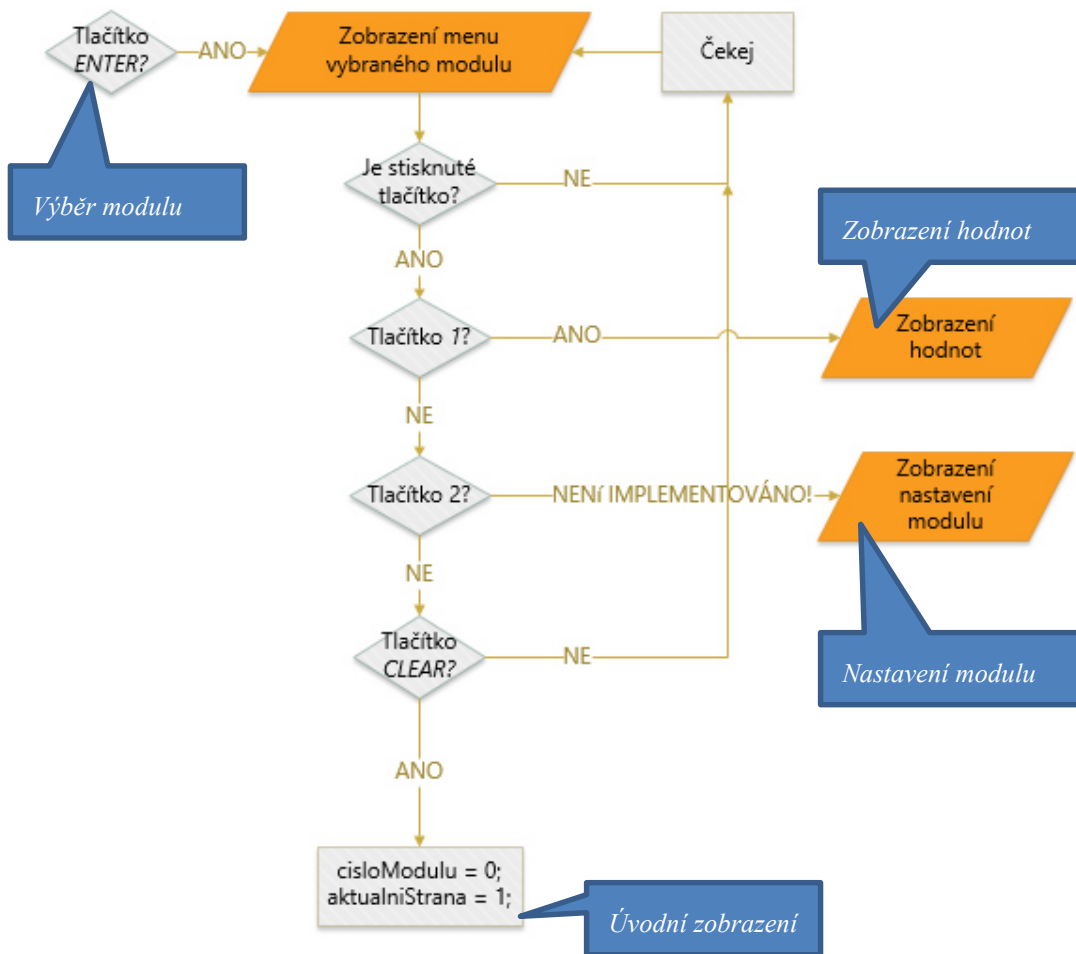
Modul se zvolí, pokud obsluha bude zadávat adresu modulu posloupností stisknutých čísel:

- pro adresu 1, stisknout 1
- pro adresu 10, stisknout 1 a následně 0
- pro adresu 100, stisknout 1, dále 0 a nakonec 0

1	Teploty
2	Proudové smyčka 4-20 mA
3	Napeti 0-10 V
4	Impulzní signaly
5	Není k dispozici
6	Není k dispozici
7	Není k dispozici
8	Není k dispozici
09:15:08 12.04.20	
Tlačítko zpět 'CLEAR'	

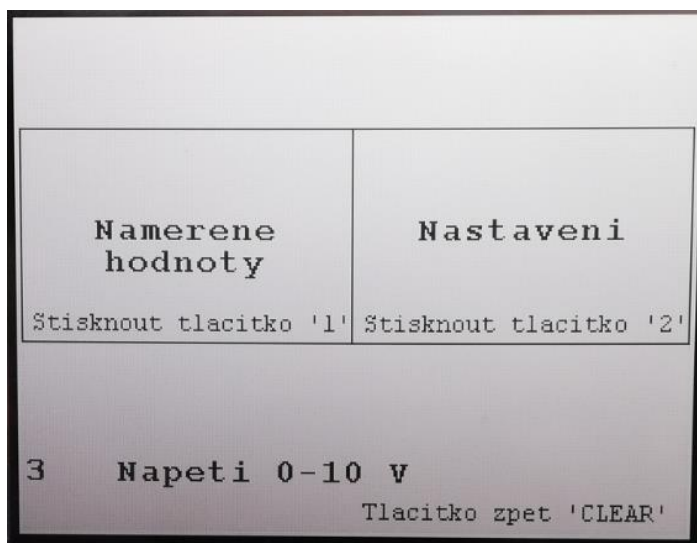
Obrázek 5.9: Obrazovka Výběr modulu

Aktuální hodnota zadávané adresy je ve zvýrazněném řádku. Obsluha potvrdí výběr stisknutím tlačítka *ENTER* a pro nové zadávání slouží tlačítko *CLEAR*.

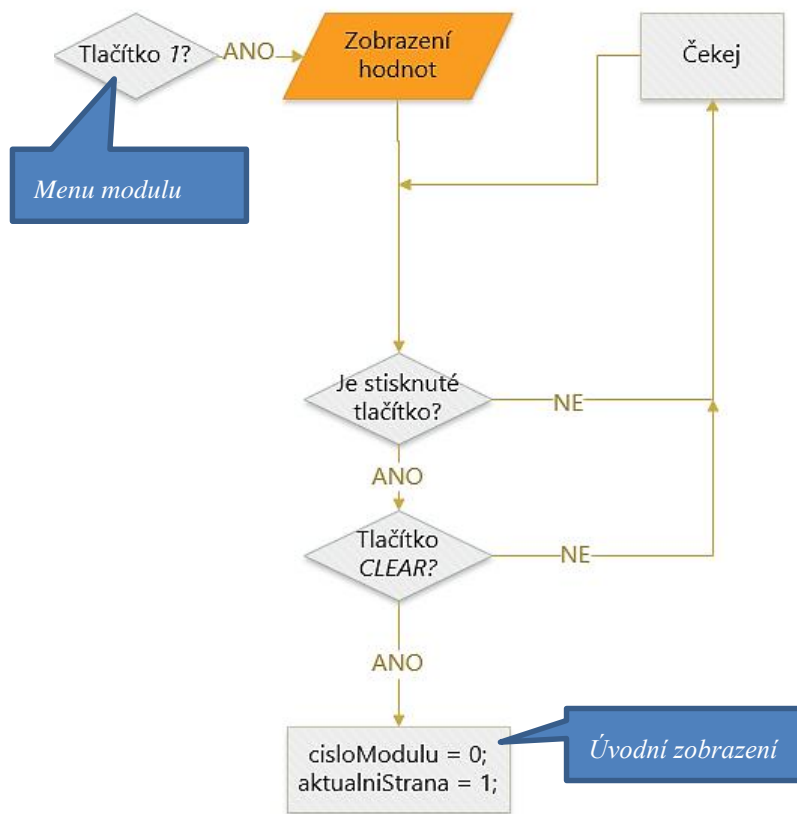


Obrázek 5.10: Část vývojového diagramu pro uživatelské rozhraní – část Menu modulu. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.

Po výběru zadané adresy a stisknutí tlačítka *ENTER* se zobrazí úvodní menu příslušného modulu:



Obrázek 5.11: Menu vybraného měřicího modulu



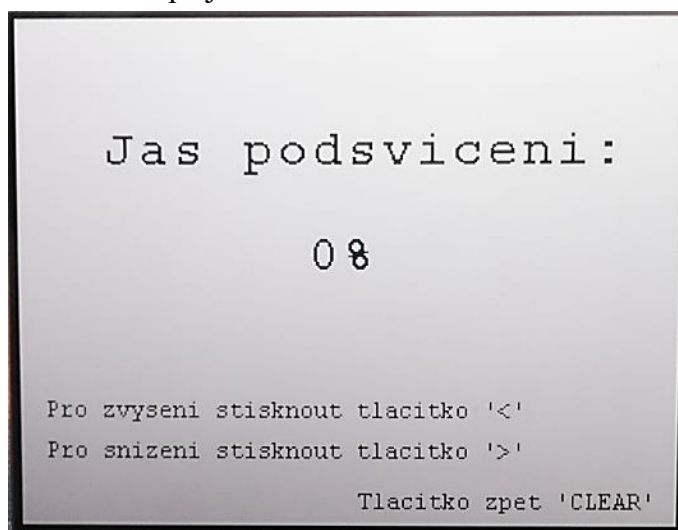
Obrázek 5.12: Část vývojového diagramu pro uživatelské rozhraní – část Zobrazení hodnot. V modré bublině je uvedeno, kde se blok nachází v kompletním vývojovém diagramu.

Stisknutím tlačítka 1 je zvoleno zobrazení hodnot senzorů, stisknutím tlačítka 2 nastavení modulu a stisknutím CLEAR se obsluha vrátí na obrazovku Seznam modulů.

1 1,1110 V	5 1,1110 V
2 1,1110 V	6 1,1110 V
3 1,1110 V	7 1,1110 V
4 1,1110 V	8 1,1110 V
3 Napeti 0-10 V	
Tlacitko zpet 'CLEAR'	

Obrázek 5.13: Obrazovka s naměřenými hodnotami

Pokud se obsluha nachází v *Seznamu modulu* a stiskne tlačítko *-/+*, tak se zobrazí *Nastavení jasu* podsvícení displeje:



Obrázek 5.14: *Nastavení jasu podsvícení displeje*

Tlačítkem *Směr nahoru* se zvyšuje šířka pulzu PWM modulace po hodnotách +100 a opačně, tlačítkem *Směr dolů* se snižuje šířka pulzu PWM modulace po hodnotách -100. Maximální šířka pulzu třetího kanálu časovače *TIM4* má hodnotu 999 (tj. 1s). Pro změnu hodnot jasu v procentech je maximální hodnota proměnné *velikostJasu* 1000, ale do registru časovače je skutečně zapsána hodnota 999. Pokud je procentuální hodnota nastavena na 0%, je použitý kanál pro PWM modulaci deaktivován.

```
void Sniz_HodnotuJasu()
{
    velikostJasu = velikostJasu - 100;
    if (velikostJasu <= 0) // velikostJasu je typu int16_t
    {
        velikostJasu = 0;
        HAL_TIM_PWM_Stop(&htim4, TIM_CHANNEL_3);
        //Zastavi casovani PWM signalu pro PODSVICENI
    }
}
void Zvys_HodnotuJasu()
{
    velikostJasu = velikostJasu + 100;
    if (velikostJasu >= 1000) //nez dojde k pretečení hodnoty,
    nastavuji maximalni hodnotu
    {
        velikostJasu = 1000;
    }
    if (velikostJasu == 100)
    {
        HAL_TIM_PWM_Start(&htim4, TIM_CHANNEL_3);
        //Aktivuje casovani PWM signalu pro PODSVICENI
    }
}
```

Zdrojový kód: 5.3: *Změna hodnoty podsvícení*

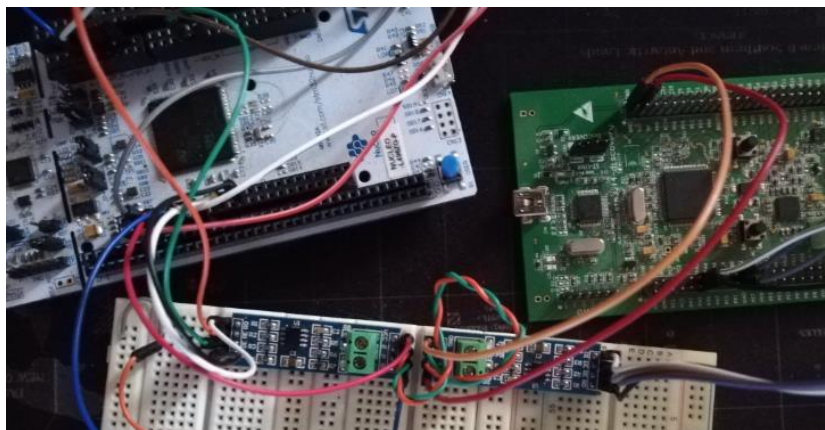
5.1.4 EPSNet

Modul inteligentního displeje komunikuje s motherboardem pomocí protokolu EPSNet na sériové lince RS485. Při vývoji programu je pro realizaci RS485 použit převodník TTL na RS485, který je připojen ke komunikačnímu kanálu *LPUART1*.

```
/* LPUSART1 init function */
void MX_LPUART1_UART_Init(void)
{
    hlpuart1.Instance = LPUART1;
    hlpuart1.Init.BaudRate = 9600;
    hlpuart1.Init.WordLength = UART_WORDLENGTH_9B;
    hlpuart1.Init.StopBits = UART_STOPBITS_1;
    hlpuart1.Init.Parity = UART_PARITY_EVEN;
    hlpuart1.Init.Mode = UART_MODE_TX_RX;
    hlpuart1.Init.HwFlowCtl = UART_HWCONTROL_NONE;
    hlpuart1.Init.OneBitSampling =
    UART_ONE_BIT_SAMPLE_DISABLE;
    hlpuart1.AdvancedInit.AdvFeatureInit =
    UART_ADVFEATURE_NO_INIT;
    if (HAL_UART_Init(&hlpuart1) != HAL_OK)
    {
        Error_Handler();
    }
}
```

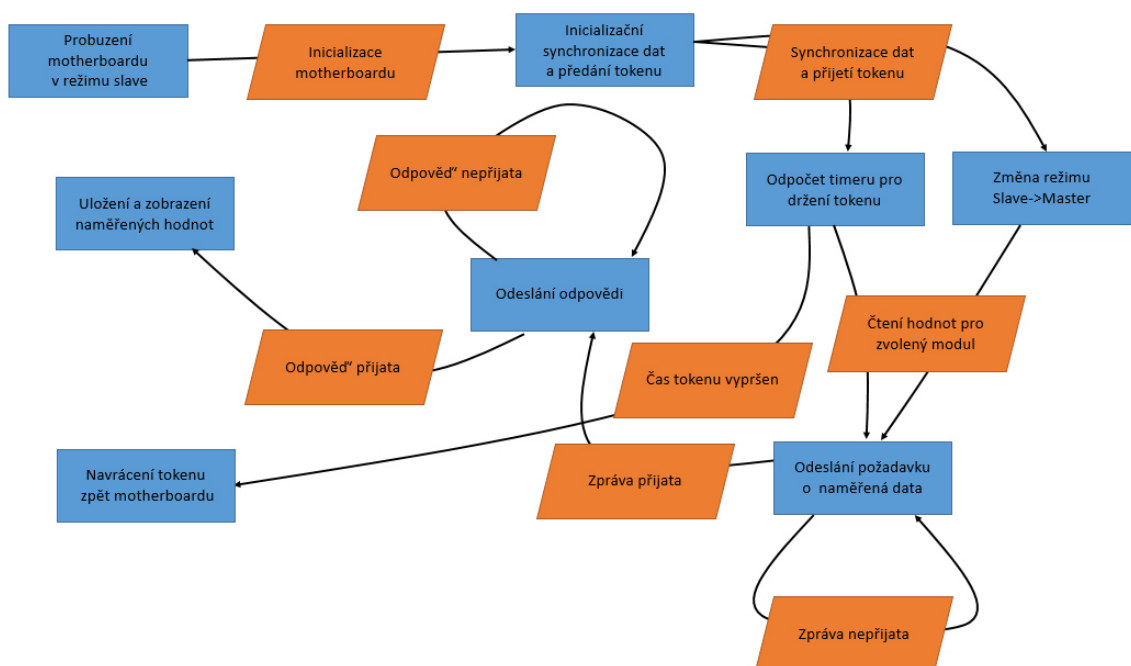
Zdrojový kód: 5.4: Inicializace LPUART1

Použitý protokol je implementovaný ve zdrojových souborech (*EPSNet_modul.c* a *EPSNet_motherboard.c*) a v hlavičkovém souboru (*EPSNet.h*). Ve zdrojovém souboru *EPSNet_motherboard.c* je implementován firmware motherboardu, který je zastoupen vývojovou deskou STM32F4-DISCOVERY) pouze pro test komunikace s modulem inteligentního displeje (vývojová deska NUCLEO-L496ZG-P), a proto veškerý popis programu v této práci je věnován pouze zmíněnému modulu. Testování se stávající měřicí ústřednou není možné, jelikož její řídicí software neobsahuje režim multimaster. Nutnost tohoto režimu je vysvětleno později.



Obrázek 5.15: Simulovaná komunikace modulu inteligentního displeje s motherboardem

Po zapnutí modulu a přetečení čítače v obvodu časovače *TIM5* je zahájen synchronizační proces. Během inicializační synchronizace je aktualizován čas vnitřního RTC obvodu, který řídící jednotka inteligentního modulu obsahuje. Dále je aktualizován seznam dostupných modulů pomocí sledu logických 0 a 1 (logická 0 – adresa není přiřazena, logická 1 – adresa je přiřazena) v datové oblasti přijaté zprávy. Pořadí bitů v *Stavové posloupnosti* určuje zároveň hodnotu adresy (např. 2. bit zleva = adresa 2). Typ modulu (a zároveň typ fyzikální jednotky naměřených dat) na konkrétní adrese je v programu (prozatím) pevně daný. V budoucnu lze typ modulu a fyzikální veličiny definovat například v menu nastavení.



Obrázek 5.16: Diagram komunikace modulu přes protokol EPSNet

Pro snížení objemu přenášených dat nejsou naměřená data všech dostupných modulů přenášena v jednom okamžiku (tj. během počáteční synchronizace), ale jsou přenášena pouze data pro vybraný modul v okamžiku jeho zvolení. Obsluha může tedy zobrazovat naměřená data pouze v režimu master. Od této chvíle se pod pojmem master uvažuje pouze jednotka s právem řídit komunikaci (tj. jednotka držící token). Jedinou výjimkou je okamžik, kdy se zapne modul inteligentního displeje a je nutné informovat motherboard o jeho zapnutí. Ve skutečnosti motherboard a modul inteligentního displeje jsou vždy v režimu master, jen nejsou popsány jako master s právem řídit komunikaci (držící token) a master bez práva řídit komunikaci. Toto chápání je nadále v programu i v popisu praktické části práce.

Původně se uvažoval dodatečný signál mezi měřicí ústřednou a modulem, který na straně motherboardu vyvolá přerušeni. Obslužný program zašle žádost inteligentnímu displeji o zvolený typ měřicího modulu, jehož naměřené hodnoty chce obsluha zobrazit. Po přijetí odpovědi měřicí ústředna následně zašle hodnoty zvoleného modulu. Komunikace se tímto způsobem zjednoduší, ale modularita měřicí ústředny se naruší. Nakonec je implementován režim multimaster, ve kterém dochází k předávání řízení pomocí tokenu.

Aby u měřicí ústředny nedošlo k značnému narušení procesu sběru dat, neboť měřicí ústředna může sbírat data pouze v režimu master (tj. při držení tokenu). Je tento token držen modulem jen po určité (přednastavenou) dobu. Po uplynutí doby (např. 30-ti sekundách) je token předán zpět měřicí ústředně a signálem *POWER SHUTDOWN* probíhá proces vypnutí, modul se odpojuje od zdroje napětí +12 V.

Pro snazší orientaci v programu jsou některé hodnoty pojmenovány slovně pomocí symbolických konstant. V hlavičkovém souboru jsou definovány symbolické konstanty pro hodnoty přenášených znaků a nutných časových intervalů.

```
// Hodnoty znaků ve zprávě/odpovědi pro
EPSNet////////////////////////////////////
#define SD1                0x10
// Úvodní znak 1 pro zprávu/odpověď' bez datového pole (delimiter 1)
#define SD2                0x68
// Úvodní znak 2 pro zprávu/odpověď' s datovým polem (1...246 Bytů)
#define SD4                0xDC //Úvodní znak 4 pro token zprávu
#define SC                 0xE5 // Krátké potvrzení (short
acknowledgment)
#define SAC                0xE5 // Krátké potvrzení (short
acknowledgment)
#define ED                 0x16 // Koncový znak (end delimiter)
```

Zdrojový kód: 5.5: Ukázka některých symbolických konstanty pro EPSNet

Mezi jednotlivými přenosy je nutné dodržet požadovaný klidový interval, který je závislý na počtu bitů v datovém oktetu a na modulační rychlosti přenosu dat. Pro modul inteligentního displeje je nutné pomocí časovače *TIM6* odpočítávat kratší časový interval, který je dán vztahem (5.1) a (5.3). Dosazením do vztahů (5.2) a (5.4) je zjištěna hodnota časovače *TIM7*, jenž je použita pro odpočítávání delšího časového intervalu. Skutečná zapsaná hodnota do registru je o jedničku menší.

Minimální časový interval klidového stavu pro přijetí nebo odeslání odpovědi je:

$$TIMEOUT_{1B_us} = \frac{\text{Počet bitů}}{\text{Modulační rychlost}} = \frac{11}{9600} \cong 1146 \text{ us} \quad (5.1)$$

Minimální časový interval klidového stavu pro přijetí nebo odeslání žádosti je:

$$TIMEOUT_{3B_us} = \frac{\text{Počet bitů}}{\text{Modulační rychlost}} = \frac{33}{9600} \cong 3438 \text{ us} \quad (5.2)$$

V případě taktu 16 MHz hodinového signálu *CLK* bude nutné pro interval *TIME_1B_us* vykonat následující počet inkrementů/dekrementů :

$$TIMEOUT_{1B} = \frac{TIMEOUT_{1B_us}}{CLK} = 1146 \cdot 10^{-6} \cdot 16 \cdot 10^6 = 18\,336 \quad (5.3)$$

Pro interval *TIME_3B_us* je počet inkrementů/dekrementů následující:

$$TIMEOUT_{3B} = \frac{TIMEOUT_{3B_us}}{CLK} = 3438 \cdot 10^{-6} \cdot 16 \cdot 10^6 = 55\,008 \quad (5.4)$$

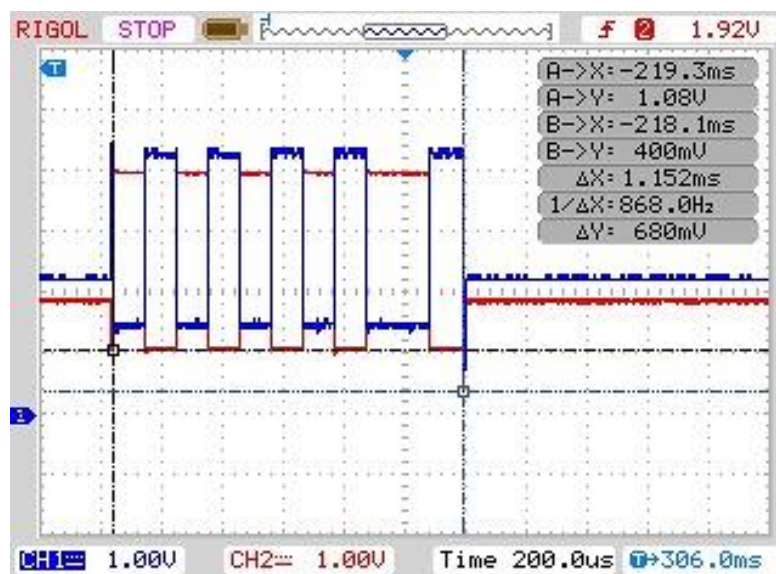
Ve stávajícím firmwaru měřicí ústředny je délka života odpovědi maximálně 750 ms. Se zachováním tohoto timoutu a pro adresu inteligentního displeje 127 je v režimu MULTIMASTER minimální časový interval dle (3.1) následující:

$$TIMEOUT_{MAX_ms} = 750 + 500 + 10 \cdot 127 = 2520 \text{ ms} \quad (5.5)$$

Za použití časovače *TIM5* s hodnotou předděliče *PSC* 15999 a s taktom hodinového signálu 16 MHz je nutné časovače přednastavit hodnotou:

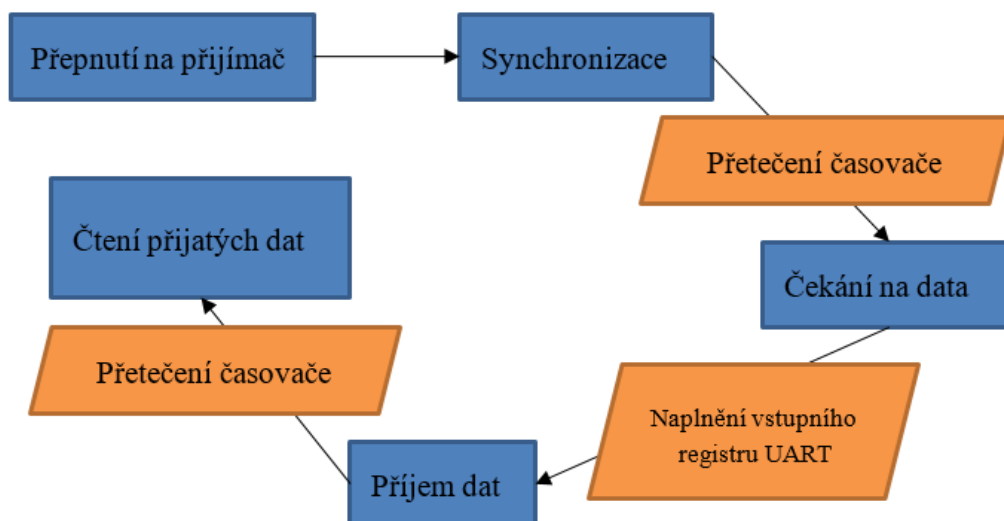
$$\begin{aligned} TIMEOUT_{MAX} &= \frac{TIMEOUT_{MAX_ms} \cdot CLK}{(PSC + 1)} = \\ &= \frac{2520 \cdot 10^{-3} \cdot 16 \cdot 10^6}{16\,000} = 2520 \end{aligned} \quad (5.6)$$

Pokud je modul v režimu slave (tj. během synchronizace dat a systémového času), používá se delší časový interval během přenosu přijaté zprávy a kratší časový interval pro kontrolu časování odesílané odpovědi a klidového stavu. V režimu master je delší interval použit pro dodržení klidového stavu před odesláním zprávy a pro splnění časového intervalu každého odeslaného bytu zprávy. Kratší interval se používá pro kontrolu klidového stavu po posledním odeslaném bytu zprávy a časování přijaté odpovědi. Ověření výpočtu časového intervalu získaného vztahem (5.1) je provedeno záznamem průběhu obou napětí na kroucené dvojlince během přenosu 11-ti bitů s modulační rychlostí 9600 Baudů.



Obrázek 5.17: Osciloskopem zaznamenaný průběh jednoho oktetu na lince A a B

Během naslouchání je modul v několika stavech, které jsou znázorněny na následujícím zjednodušeném stavovém diagramu.



Obrázek 5.18: Stavový diagram přijímače

V okamžiku přepnutí modulu RS485 do režimu přijímače (receiver) se vyčkává na cyklus synchronizace přetečením čítače daného časovače. Během synchronizace je aktivován stav *EPSNET_STAV_SYNCHRONIZACE*, během kterého jsou všechna příchozí data ignorována (nejsou zapsána) a začne nové čekání na klidový stav, kontrolováno funkcí *EPSNet_klidovyStavRS485()*. Obslužný program přerušení vyvolaného z obvodu časovače (viz Zdrojový kód 5.7) přepne modul do stavu *EPSNET_STAV_CEKANI* a následně první příchozí (zároveň zapsaný) byte přepne do následujícího stavu *EPSNET_STAV_PRIJEM* (viz Zdrojový kód 5.6).

```

/* UART in mode Receiver -----*/
if((isrflags & USART_SR_RXNE) != RESET) && ((crlits &
USART_CR1_RXNEIE) != RESET))
{
    if (EPSNET_rezim == EPSNET_REZIM_MPC) //rezim master
    {
        if ((EPSNET_stav == EPSNET_STAV_CEKANI) ||
(EPSNET_stav == EPSNET_STAV_PRIJEM))
        {
            cnt++;
            if (huart->RxISR != NULL)
            {
                huart->RxISR(huart);
            }
            EPSNET_stav = EPSNET_STAV_PRIJEM;
        }
        else
        {
            // je nutne vynulovat priznak RXNE bez zapisu
            uint16_t uhdata = (uint16_t) READ_REG(huart
->Instance->RDR);
        }
        __HAL_TIM_DISABLE_IT(&htim6, TIM_IT_UPDATE);
        HAL_TIM_GenerateEvent(&htim6, TIM_IT_UPDATE);
        __HAL_TIM_CLEAR_IT(&htim6, TIM_IT_UPDATE);
        __HAL_TIM_ENABLE_IT(&htim6, TIM_IT_UPDATE);
    }
}

```

Zdrojový kód 5.6: Ukázka části obslužného programu přerušení z obvodu UART

Nejprve se ověřuje naplnění přijímajícího registru RDR sběrnice UART (včetně povoleného přerušení) a nakonec se určuje, zda je modul v režimu master nebo slave. V obslužném programu přerušení vyvolaného přijetím dat se čítač resetuje. Celý kód lze nalézt ve zdrojovém souboru *stm32l4xx_hal_uart.c*.

Pokud ve stavu *EPSNET_STAV_PRIJEM* dojde k přetečení čítače, je ukončen příjem odpovědi funkcí *HAL_UART_RxDone(&hlpuart1)*. Následně obslužný program přerušení dokončeného přenosu volá funkci *EPSNet_cti_datagram()*.

```

if (EPSNET_rezim == EPSNET_REZIM_MPC) //rezim master
{
    if (htim->Instance == TIM6)
    {
        switch (EPSNET_stav)
        {
            case EPSNET_STAV_SYNCHRONIZACE:
                EPSNET_stav = EPSNET_STAV_CEKANI;
                cnt = 0;
                break;
            case EPSNET_STAV_CEKANI:
                break;
            case EPSNET_STAV_PRIJEM :
                EPSNET_stav = EPSNET_STAV_TIMEOUT;
                HAL_UART_RxDone(&hlpuart1);
                break;
            default:
                break;
        }
    }
}

```

Zdrojový kód 5.7: Ukázka části obslužného programu přerušení z obvodu časovačů

Při čtení datagramu je nejprve načten úvodní znak *SD* (první prvek z pole *uart_buffer_rx*) do proměnné *typDatagramu_SD*. Následně se funkce větví pomocí příkazu *switch(typDatagramu_SD)* na větve:

- *case SD1*
- *case SD2*
- *case SD4*
- *case SAC*

V odpovídajících větvích se kontroluje datagram na chybu kontrolního součtu (příznak *Err_FCS*), koncový znak *ED* (*Err_ED*), směr komunikace (*Err_FC_jednotka*) a znak funkce *FC* (*Err_FC_znak*). Směrem komunikace se rozumí, že je přenesen datagram se správnou cílovou adresou a s nesprávnou (nikoliv neznámou) hodnotou *FC*. Master jednotce je možné odeslat pouze odpověď od podřízené jednotky nebo token od nadřazené stanice. Pokud je přenos datagramu bez chyby a pokud je přenesený datagram určen modulu inteligentního displeje, zašle modul v režimu slave pozitivní odpověď nebo v režimu master zobrazí naměřené hodnoty. V opačném případě se zašle negativní odpověď nebo se opakovaně zašle poslední žádost. Počet opakování je určeno hodnotou v proměnné *pocet_Pokusu*.

Naměřené hodnoty jsou protokolem přenášeny ve formátu *malého endianitu* (1. bit je LSB) a mají datový typ *float*. Datová oblast odpovědi obsahuje 8 přepočtených hodnot (s fyzikální jednotkou modulu) a 8 skutečných hodnot (s fyzikální jednotkou použitého senzoru) naměřených dat. Na displeji jsou zobrazeny pouze přepočtené verze naměřených hodnot.

Pro přenos naměřené hodnoty je nutné tuto hodnotu převést na čtyř prvkové pole bytů a po přijetí následně převést zpět. K tomuto převodu je použita datová struktura *PrevodFloatHodnoty*. Zdrojový kód 5.8 znázorňuje použití této struktury při převodu v obou směrech.

```
union {
char floatHodnota_rozdelena[4];
float floatHodnota_komplet;
} PrevodFloatHodnoty;
//struktura pro prevod namerene hodnoty float -> pole bytu/pole
bytu -> float

// hodnota 1. cidla
PrevodFloatHodnoty.floatHodnota_komplet =
seznamModulu_test[cislo_VybranehoModulu].prepoctenaHodnota1;

uart_buffer_tx[7] = PrevodFloatHodnoty.floatHodnota_rozdelena[0];
uart_buffer_tx[8] = PrevodFloatHodnoty.floatHodnota_rozdelena[1];
uart_buffer_tx[9] = PrevodFloatHodnoty.floatHodnota_rozdelena[2];
uart_buffer_tx[10] = PrevodFloatHodnoty.floatHodnota_rozdelena[3];

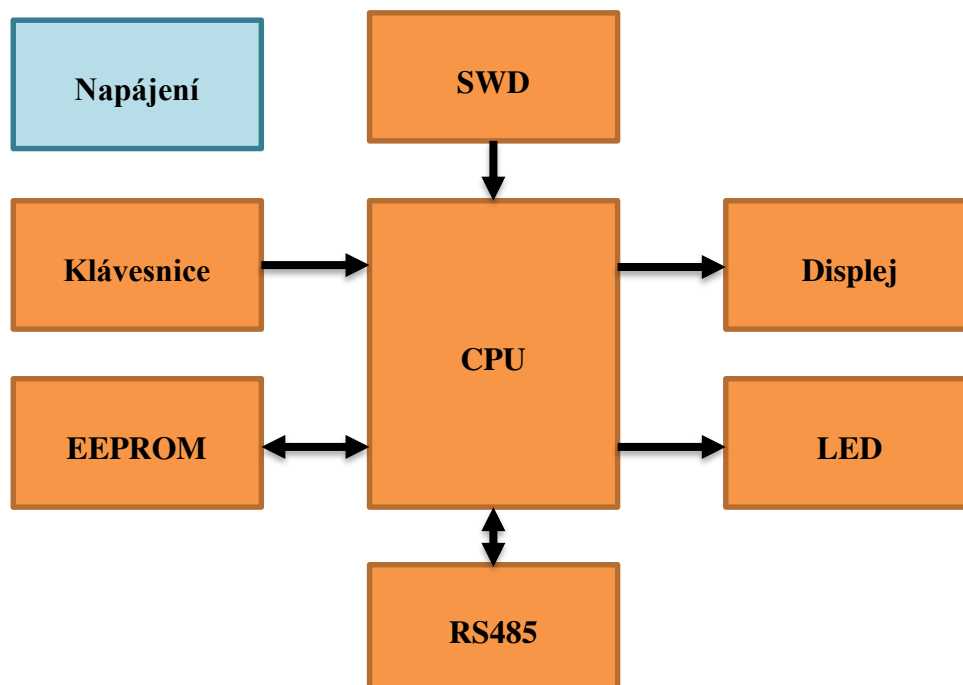
// hodnota 1. cidla
PrevodFloatHodnoty.floatHodnota_rozdelena[0] = uart_buffer_rx[7];
PrevodFloatHodnoty.floatHodnota_rozdelena[1] = uart_buffer_rx[8];
PrevodFloatHodnoty.floatHodnota_rozdelena[2] = uart_buffer_rx[9];
PrevodFloatHodnoty.floatHodnota_rozdelena[3] = uart_buffer_rx[10];

prepoctenaHodnota1_rx = PrevodFloatHodnoty.floatHodnota_komplet;
```

Zdrojový kód 5.8: Ukázka převodů naměřené hodnoty nutné pro přenos

5.2 Návrh hardwaru

Navržený prototyp lze rozdělit do několika funkčních bloků, která jsou v následujících podkapitolách detailněji popsána.

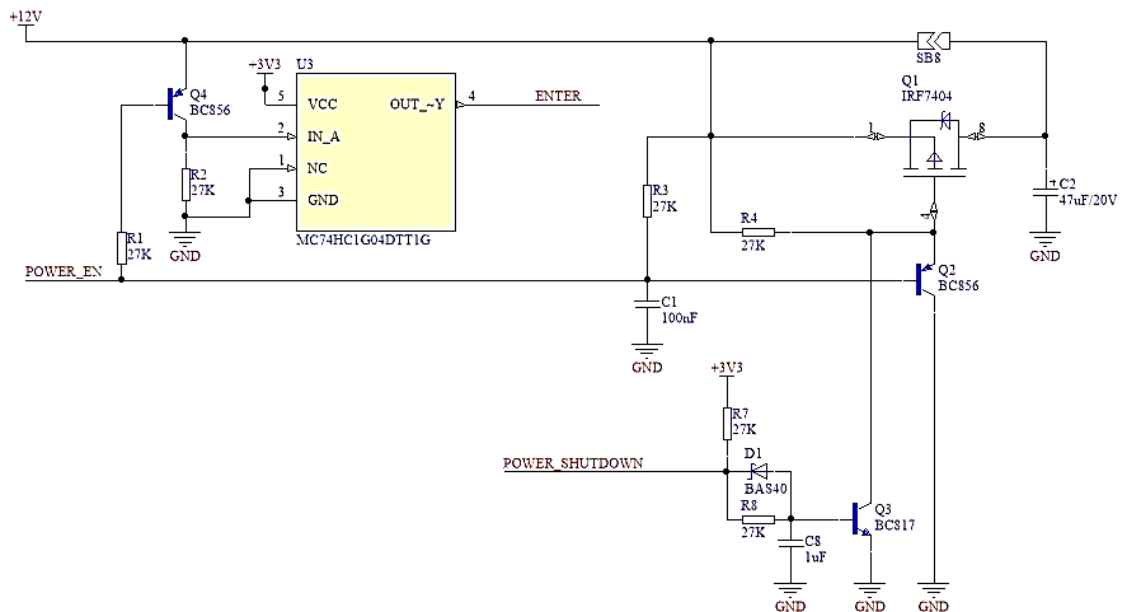


Obrázek 5.19: Blokové schéma modulu inteligentního displeje

5.2.1 Výkonová část modulu

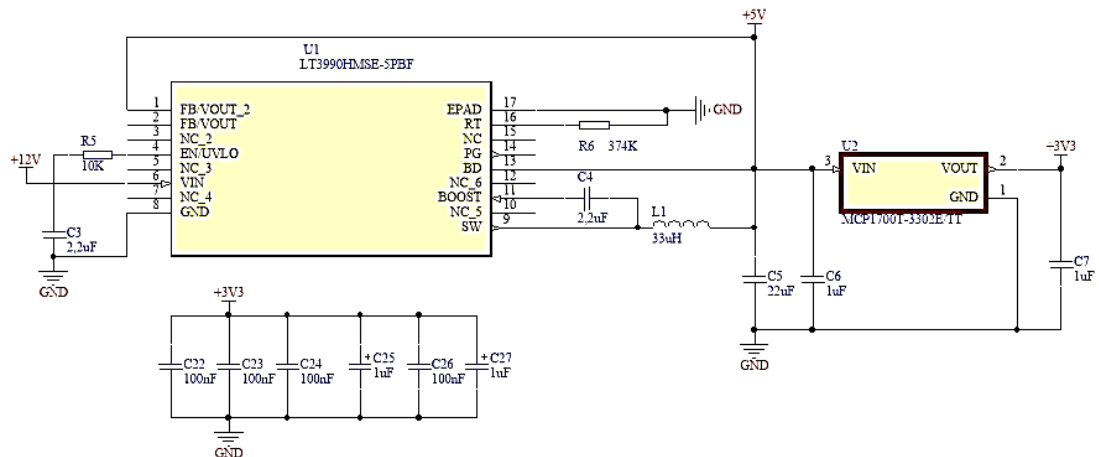
Napájení celého zařízení je možné pro vyšší úsporu připojit/odpojit od přivedeného zdroje napětí +12 V pomocí unipolárního tranzistoru $Q1$ (IRF7404). Stisknutím tlačítka *ENTER* je přiveden proud do báze bipolárního tranzistoru $Q2$ (BC856), otevřením tohoto tranzistoru vznikne úbytek napětí (průchodem proudu) na rezistoru $R4$ a k sepnutí přivedeného napětí +12 V tranzistorem $Q1$. Zároveň dojde k otevření bipolárního tranzistoru $Q4$ a průchodem proudu rezistorem $R2$ vznikne úbytek napětí, které je sledováno invertorem $U3$ (MC74HC1G04). Takto je přivedena informace o stisknutí tlačítka *ENTER* na vstupní pin procesoru. Po rozpojení tlačítka *ENTER* udržuje otevření tranzistoru $Q1$ nadále sepnutý bipolární tranzistor $Q3$ (BC817).

Po ukončení požadavků obsluhy je procesorem přiveden signál *POWER SHUTDOWN*, který tranzistor $Q3$ uzavře a rezistorem $R4$ již neteče dostatečný proud. To má za následek uzavření tranzistoru $Q1$ a odpojení modulu od zdroje napětí +12 V.



Obrázek 5.20: Schéma zapojení řídicího obvodu vstupního napětí +12 V

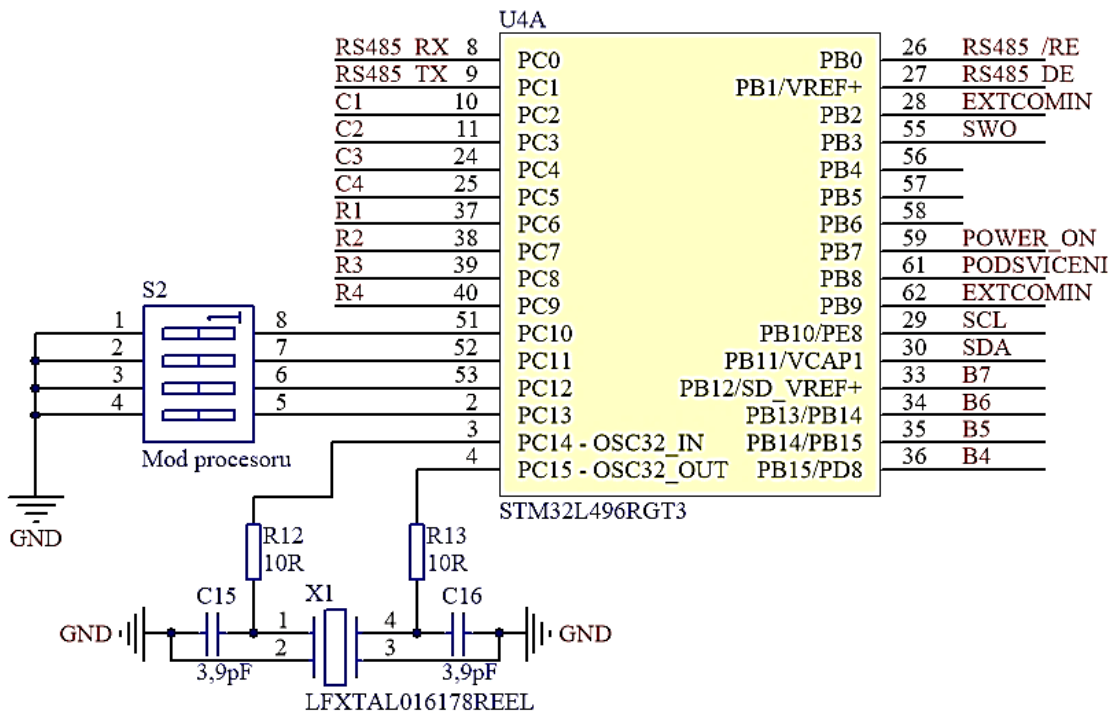
Vstupní napětí +12 V je sníženo spínaným zdrojem napětí $U1$ na +5 V, které napájejí použitý displej. Pro napájení řídicí a komunikační elektroniky modulu je dále toto napětí převedeno pomocí lineárního zdroje napětí $U2$ na +3,3 V.



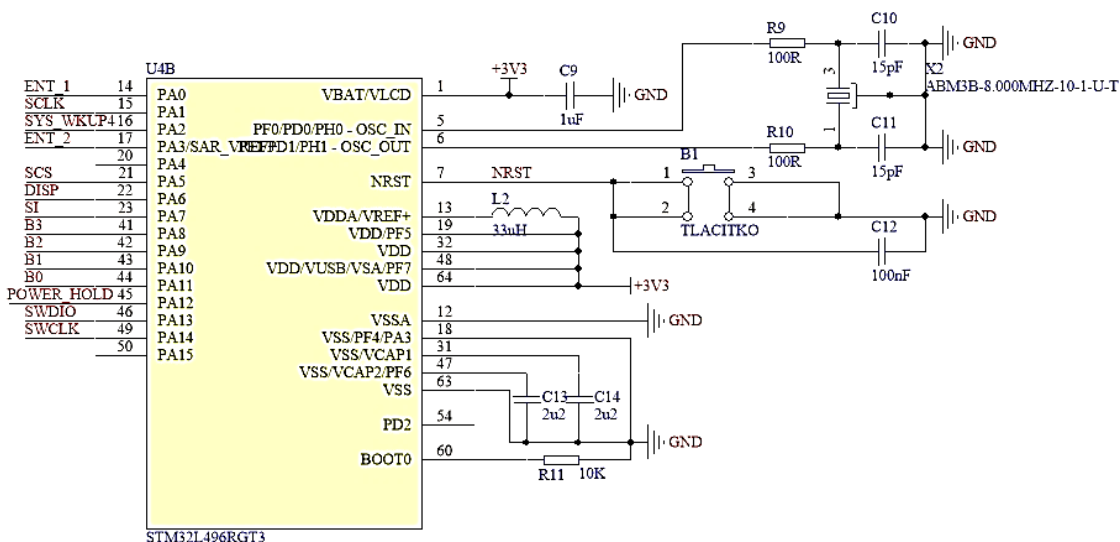
Obrázek 5.21: Spínaný zdroj napětí +5 V a lineární zdroj napětí +3,3 V

5.2.2 Řídicí jednotka

Řídicí jednotka *U4* je zastoupena procesorem STM32L496RGT3, jde o 32-bitový procesor typu ARM v nízko energetické třídě L4. Procesor má celkem 64 pinů a jejich zapojení je na obrázku 5.22 a 5.23.



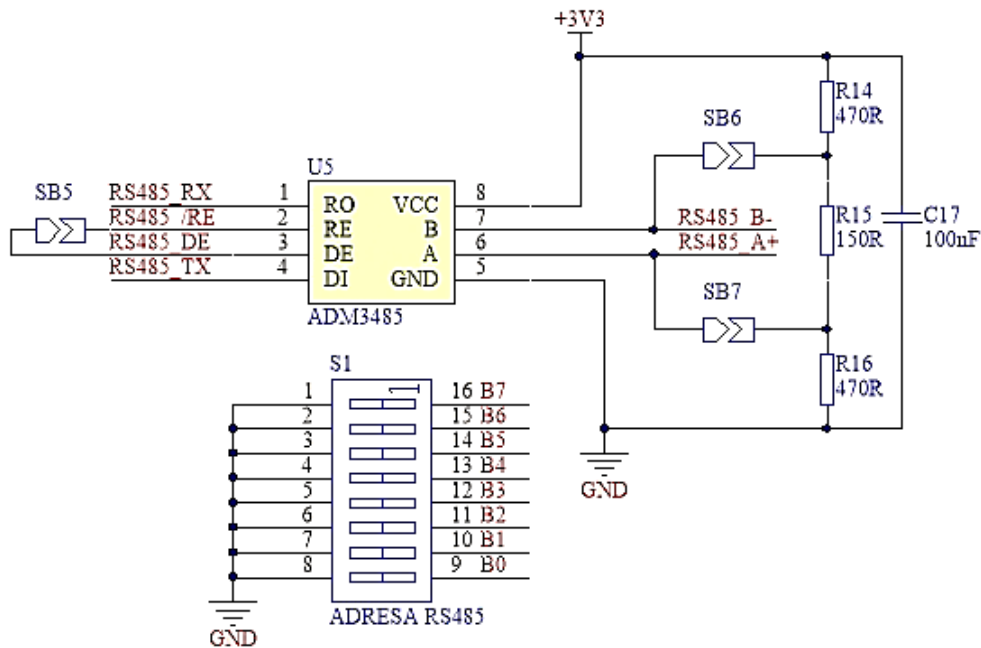
Obrázek 5.22: První část zapojení řídicího procesoru typu ARM



Obrázek 5.23: Druhá část zapojení řídicího procesoru typu ARM

5.2.3 Modul sériové linky RS485

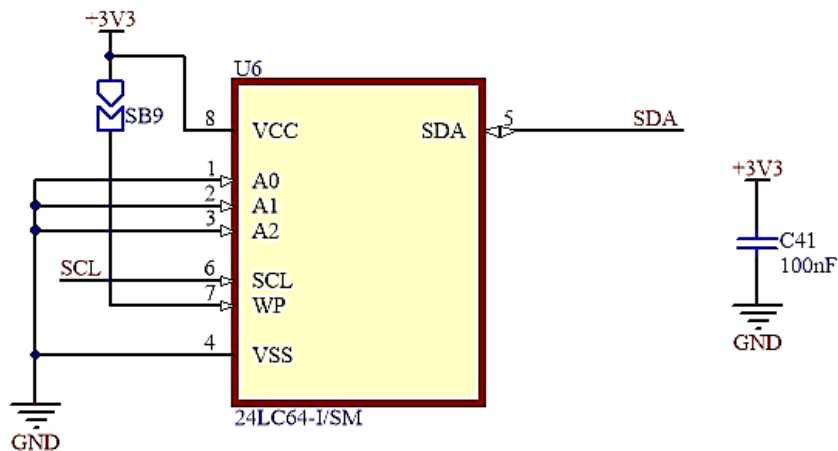
K použití sériové komunikační linky RS485 je nutné převést napěťovou úroveň diferenční dvojice signálu RS485_A+ a RS485_B-, k tomu slouží integrovaný obvod U5 (ADM3485). Jelikož modul inteligentního displeje může být koncovým členem na vedení této komunikační linky, je možné pomocí propojek SB6 a SB7 připojit terminátor (odpor R15). Adresa tohoto modulu je dána binární kombinací hodnot na DIP switchi S1.



Obrázek 5.24: Zapojení sériové linky RS485

5.2.4 Zálohovaná paměť EEPROM

Pro zálohu proměnných a konstant slouží externí paměť EEPROM U6, která s procesorem sdílí informace pomocí rozhraní I2C.



Obrázek 5.25: Zapojení EEPROM paměti

5.2.5 Klávesnice

Pro místní obsluhu je použita membránová maticová klávesnice o čtyřech řádcích a čtyřech sloupcích, přičemž tlačítko *ENTER* je vyvedeno zvlášť (tzn. oproti standardu má 10 vodičů). Její výhodou je snadné použití a nízká pořizovací cena. Nevýhodou je absence podsvícení, ale lze i pořídit klávesnici se zabudovanými LED diodami, která však má vyšší pořizovací cenu. Na zakázku lze objednat například zde: www.unipad.cz nebo www.ttklavesnice.cz.

Doba odezvy je 1 ms, kontaktní odpor 500 Ω a izolační odpor 100 M Ω . Klávesnici lze jednoduše nalepit a připojit pomocí 10 pinového konektoru. Kde první dva piny jsou pro tlačítko *ENTER*, první čtveřice pinů tvoří vývody řádků a poslední čtveřice pinů je pro zapojení vývodů sloupců.



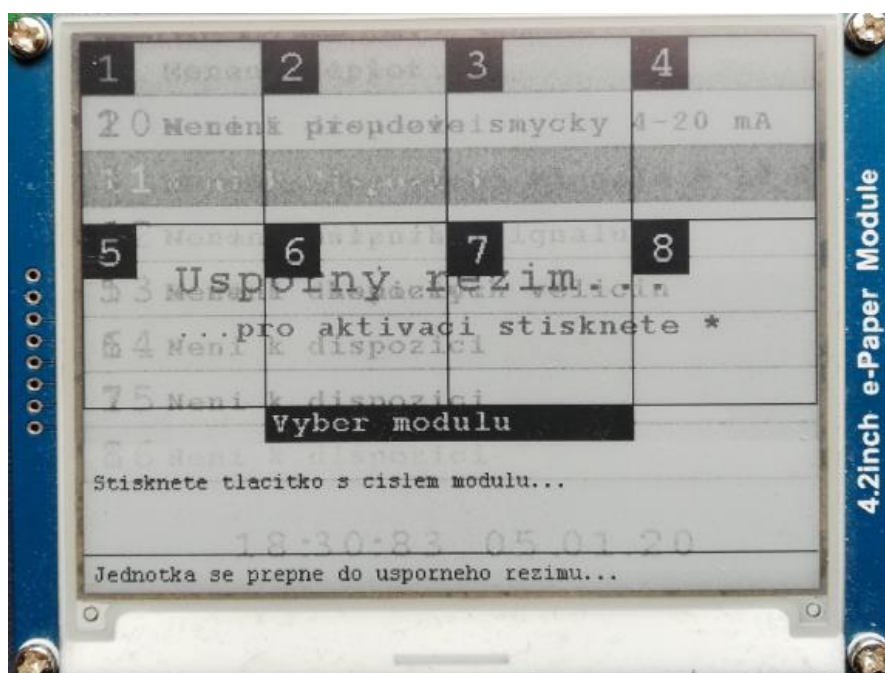
Obrázek 5.26: Membránová klávesnice

Pro dosažení vyšší úspory bylo uvažováno použít piezoelektrickou klávesnici, ale cena klávesnice 4x4 se pohybuje okolo 4 000 Kč. A vytvoření vlastní klávesnice pomocí piezoelektrických destiček, které se používají při výrobě bzučáků, vyžadují dodatečné zesilovací tranzistory a výroba je nákladná (časově i cenově).

5.2.6 Displej

Z hlediska úspory elektrické energie je prvním použitým displejem E-Paper 4,2inch modul od výrobce Waveshare. Tento displej má v klidovém stavu odběr proudu 5 uA a při vykreslování 8 mA. Požadovaný příkon displeje pro napětí +3,3 V je 16,5 uW (klidový stav) a 26,4 mW (při procesu vykreslování).

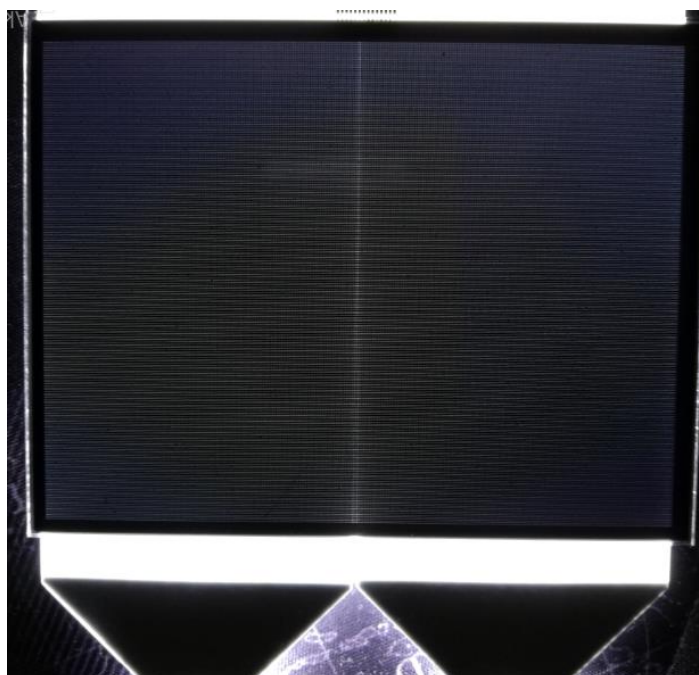
Při programování uživatelského rozhraní bylo dosaženo jeho živostnosti, neboť jeho životnost (1 mil cyklů nebo 5 let) je výrobcem garantována pouze, pokud je displej obnovován s minimálním intervalem 180 s. Výsledkem ztráty životnosti jsou trvalé stopy předešlých stránek.



Obrázek 5.27: Ukázka ztráty životnosti použitého e-ink displeje

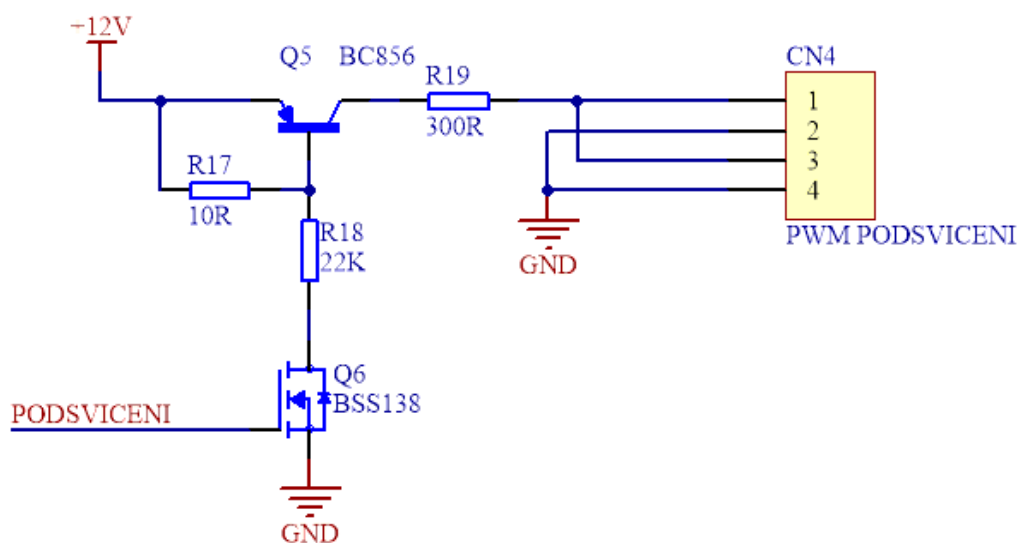
Dalším použitým displejem je Sharp Memory LCD, model LS044Q7DH01. Požadovaný příkon je při úsporném režimu 250 uW a při vykreslování paměti 650 uW. Tento displej komunikuje s procesorem pomocí sběrnice SPI (třívodičová varianta) a je připojen 10-ti pinovým FPC konektorem.

Jelikož se obsluha může nacházet i v širém prostředí, je displej podsvícen pomocí dvou LED modulů (45 mm x 86 mm) od výrobce Adafruit.



Obrázek 5.28: LED modul od Adafruit

V každém modulu je jedna výkonová LED dioda, která má maximální propustný proud 25 mA. Proto není přímo připojena na výstupní pin procesoru, ale je spínána pomocí PWM modulace přes tranzistor $Q5$ (BC856). Zdrojem spínaného napětí je hlavní přivedený zdroj napětí +12 V.

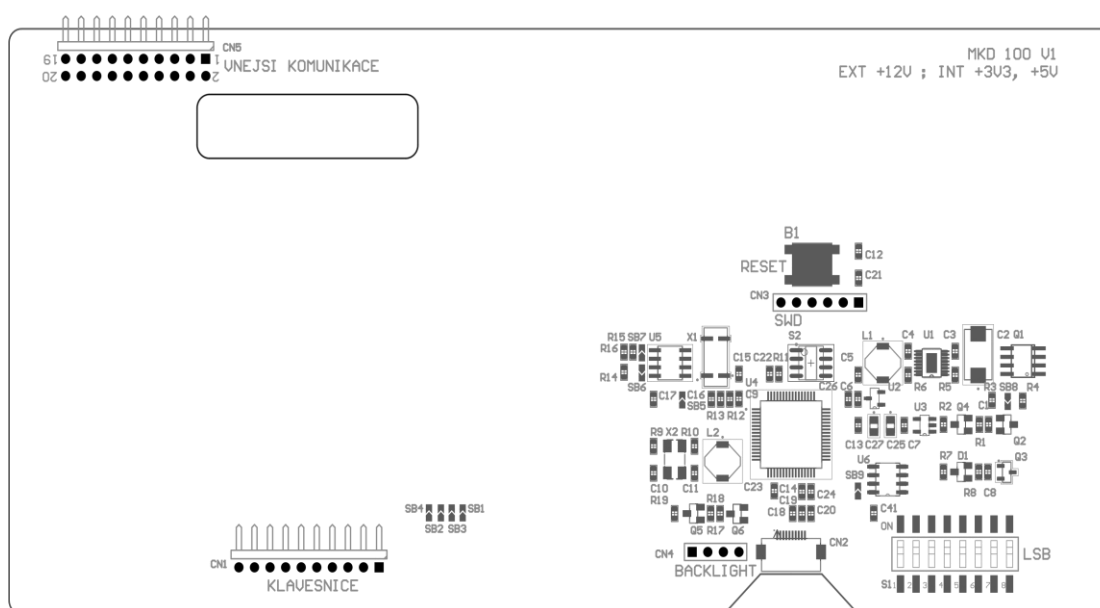


Obrázek 5.29: Obvodové zapojení spínaných LED modulů

5.2.7 Deska plošných spojů

Samotná deska (příloha E a F) je oboustranná, součástky jsou orientovány výhradně na spodní stranu (*BOTTOM*) a pomocí prokovů jsou použity i cesty na horní straně (*TOP*). Na desce jsou použity převážně součástky pro povrchovou montáž (SMD) a otvorové (THT) součástky, které jsou tvořeny pouze konektory s roztečí 2,54 mm. Rozlévaná měď na obou stranách má zemní potenciál. Napájecí spoje mají šířku 40 mil a signální cesty šířku 12 mil (v některých případech i 8 mil). Při rozmístování součástek je brán zřetel i na krátké vzdálenosti spojů. Především u krystalů, spínaného zdroje napětí a obvodu pro komunikaci s displejem, kde je frekvence signálů 2 MHz.

Pokud se podíváme ze strany *BOTTOM*, vpravo nahoře je hlavní 20-ti pinový konektor pro vnější komunikaci (RS485) a pro napájení +12 V. Pod tímto konektorem se nachází 10-ti pinový konektor pro připojení klávesnice. Levé spodní části dominuje řídicí procesor STM32L496RGT3. Vlevo od procesoru jsou dva krystaly X2 (8MHz) a X1 (32,768 kHz). Dále modul RS485 a podsvícení displeje (spínací obvod i konektor pro připojení). Vpravo od procesoru je celá výkonová část, paměť EEPROM a osmy úrovňový DIP switch pro nastavení adresy RS485. Nad procesorem je tlačítko pro *RESET*, SWD konektor a čtyř úrovňový DIP switch pro nastavení procesorového režimu. A nakonec pod procesorem se nachází ZIF konektor pro připojení displeje.



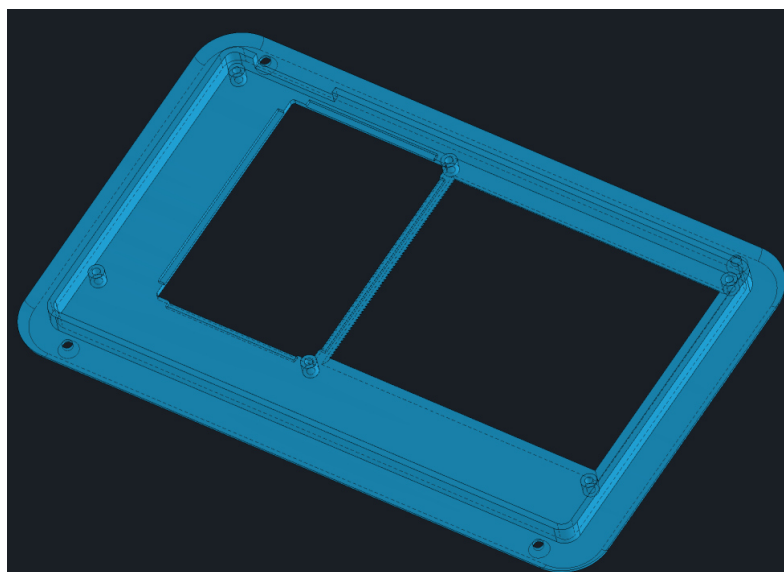
Obrázek 5.30: Osazovací schéma desky – strana *BOTTOM*

5.3 Zhotovení a oživení prototypu

Rozměr desky je 87,25 mm X 180,25 mm a obsahuje otvor pro přívodní kabel klávesnice. Celá deska je připevněna pomocí šesti šroubků k 3D výtisku. 3D model čelního panelu je vytvořen v programu SOLIDWORKS 2018. 3D tisk komplikovala prohlubeň, do které se vsadí klávesnice. Jde o rozsáhlou plochu, a proto čelní panel je rozdělen na dvě části. Pomocí dodatečných ploch podpěry jsou obě části následně přilepeny.

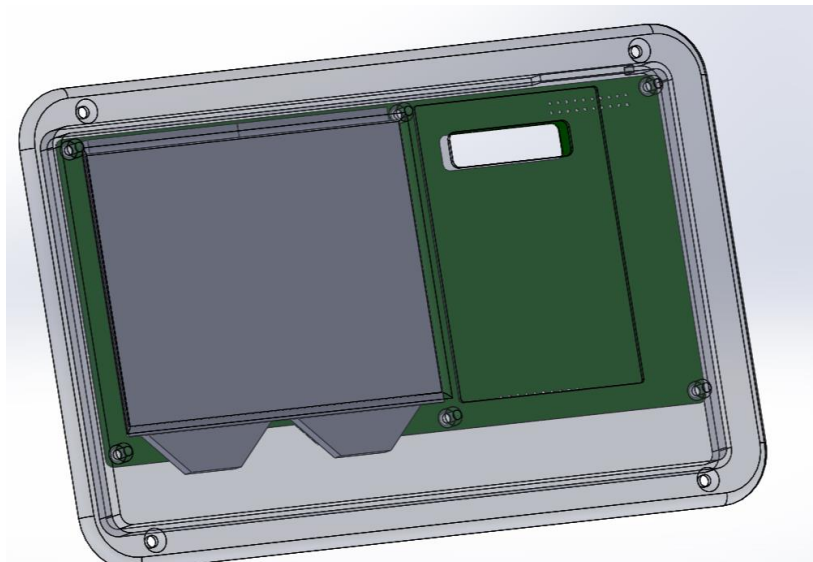


Obrázek 5.31: Podpěra pro klávesnici



Obrázek 5.32: Tělo čelního panelu

Na obrázku 5.33 je 3D sestava vyrobeného prototypu. Zde je vidět, že displej je k čelnímu panelu přitlačován deskou plošných spojů skrze LED moduly. Pro první vydání modulu inteligentního displeje bych displej doporučil vložit do rámu, který se připevní k DPS pomocí šroubků nebo zacvakávacích kolíků.



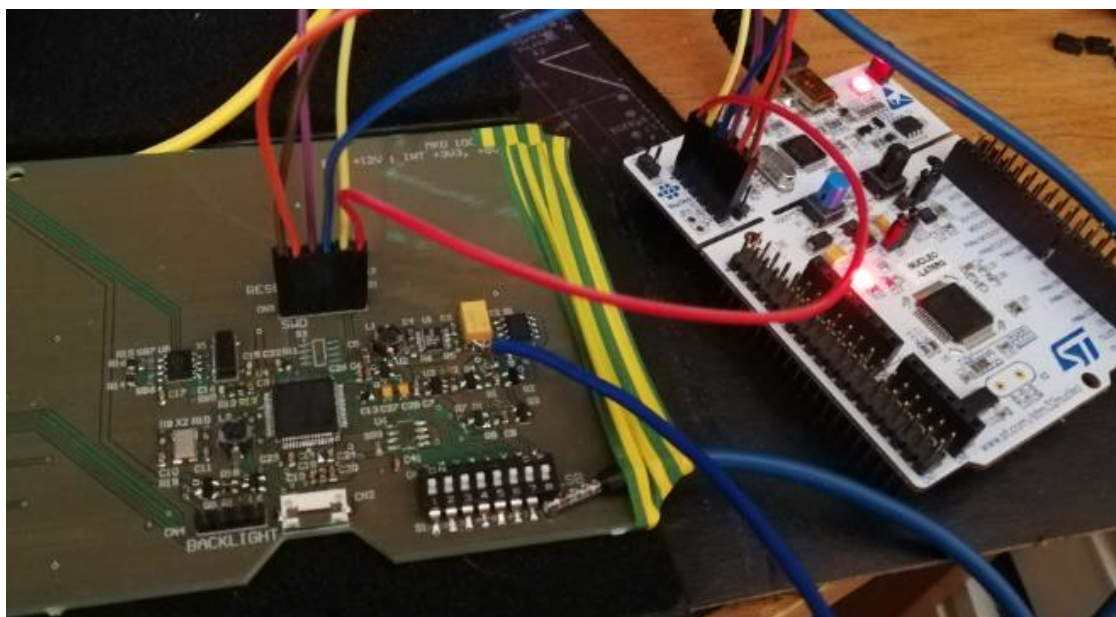
Obrázek 5.33: 3D sestava DPS a čelního panelu

3D výtisk je z PLA materiálu a má barvu Volcano grey. Z praktického důvodu je povrch hrubší povahy. Celý tento čelní panel je vložen do víka měřicí ústředny a zafixován čtyřmi šrouby se zápusťnou hlavou.



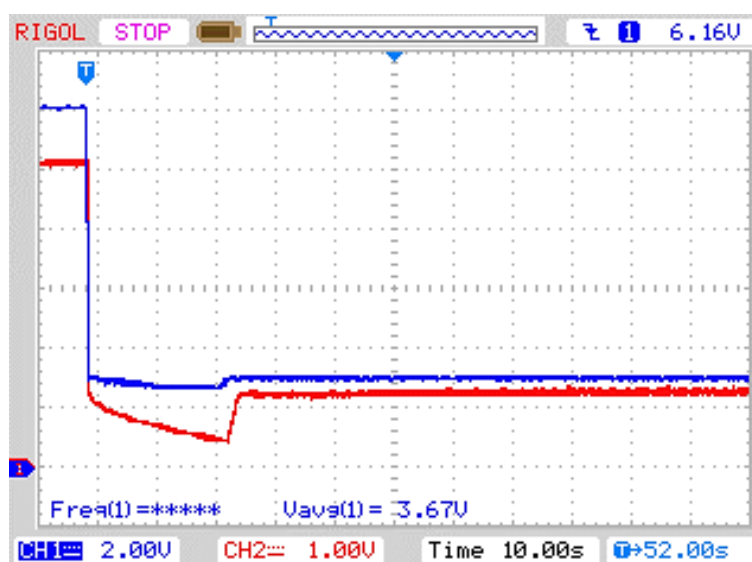
Obrázek 5.34: Zhotovený čelní panel

Po připájení integrovaných obvodů a ostatních komponentů k vyrobené desce plošného spoje je řídicí procesor naprogramován přes rozhraní SWD, který se nachází na vývojové desce NUCLEO-L476RG.



Obrázek 5.35: Programování inteligentního displeje pomocí vývojové desky NUCLEO-L476RG. Prototyp je provizorně napájen z regulovatelného zdroje napětí EP-613, silnějším žlutým kabelem přiveden potenciál +12 V a silnějším modrým kabelem přiveden potenciál GND.

Následně byl digitálním osciloskopem RIGOL DS1102D zaznamenán průběh vypnutí prototypu pomocí tranzistoru $Q1$ (IRF7404), odpojením od zdroje napětí +12 V.



Obrázek 5.36: Záznam průběhů napětí během procesu vypnutí, kanál 1 – napětí +12 V, kanál 2 – napětí +5 V.

Nakonec pomocí multimetru Agilent 34410A je změřen proud odběru celého modulu a samostatného displeje. Změřit odběr proudu displeje za pomocí osciloskopu a odporu se nepodařilo, neboť průběh napětí na odporu se ztrácel v rušivém signálu i za použití vstupního filtru.

Tabulka 5.1: Naměřené hodnoty odebíraného proudu displejem

	Střídavý proud [uA]	Stejnoseměrný proud [uA]
Zobrazení snímku	4, 183	1, 484
Vykreslení snímku	32, 677	4, 645

Tabulka 5.2: Naměřené hodnoty proudu pro vyrobený prototyp

	Stejnoseměrný proud [mA]
Statické zobrazení snímku a přenos dat	5, 806

6 Závěr

Cílem této diplomové práce bylo zhotovit modul inteligentního displeje k měřicí ústředně GU100. Zhotovení prototypu zahrnovalo implementaci uživatelského rozhraní, které tvořilo grafický displej a klávesnice. Pro komunikaci s měřicí ústřednou a přenos naměřených hodnot sloužil protokol EPSnet. Jelikož měřicí ústředna byla převážně napájena ze zálohovaného zdroje napětí (tj. z baterie), musel být důraz i na nízkou spotřebu realizovaného modulu.

Pro vývoj firmwaru prototypu inteligentního displeje byla použita vývojová deska NUCLEO-L496ZG-P. Pro inicializaci hardwarových prostředků jsem použil program STM32CubeMX v5, který umožňoval vygenerovat hlavičkové soubory a zdrojové soubory s HAL funkcemi v jazyce C. Software byl dále upravován v rozhraní Keil uVision v5, také v jazyce C. Finální program ve formátu BIN byl uploadován do flash paměti pomocí programu STM32 ST-LINK Utility v4.1.

Pro dodržení nízkého odběru proudu byl použit displej s technologií Memory LCD od výrobce Sharp. Displej vykazoval výborný kontrast na přímém slunečním svitu a zpětnou odezvu.

Aby se zachoval modulární koncept měřicí ústředny, byl inteligentní displej chápán jako další modul, který komunikuje přes protokol EPSNet.

Součástí praktické části byl dále návrh obvodového zapojení a realizace desky plošných spojů. Pro její návrh byl použit software Altium Designer v18. Byl zvolen procesor typu ARM od společnosti STMicroelectronics v nízkoenergetické třídě L4 s pracovním režimem, kdy se procesor po určité době nečinnosti obsluhy sám od napětí odpojí. Nakonec byl prototyp inteligentního modulu displeje oživen a nad rámec zadání DP vložen do čelního panelu z 3D výtisku, v tomto případě pro zabudování do krytu měřicí ústředny.

Seznam odborné literatury

- [1] *Řídící a komunikační software geofyzikální ústředny*. Liberec, 2014. Diplomová práce. Technická Univerzita v Liberci. Vedoucí práce Ing. Lubomír Slavík Ph.D
- [2] UNIVERZÁLNÍ MĚŘICÍ ÚSTŘEDNA - DATALOGGER - AHLBORN ALMEMO 500 - VÍCEKANÁLOVÁ. *AHLBORN.cz* [online]. [cit. 2020-01-29]. Dostupné z: <https://www.ahlborn.cz/pristroje/merici-ustredny/univerzalni-merici-ustredna-datalogger-ahlborn-almemo-500-vicekanalova.html>
- [3] Monitorovací systémy MS. *Cometsystem.cz* [online]. [cit. 2020-01-30]. Dostupné z: https://www.cometsystem.cz/userfiles/dokumenty_menu/3/cz_comet_monitoring_systems.pdf
- [4] Datasheet OMNIAlog NI-2400. *TR instruments spol. s.r.o.: Měřicí přístroje a systémy* [online]. , 6 [cit. 2020-01-30]. Dostupné z: <http://www.trinstruments.cz/data/files/ds-ni-2400-1720.pdf>
- [5] Piezoklávesnice. *Unipad.cz* [online]. [cit. 2020-02-01]. Dostupné z: <http://www.unipad.cz/index.php/cs/piezo-klavesnice>
- [6] Keyboard Technology. *Flylib.com* [online]. [cit. 2020-02-01]. Dostupné z: <https://flylib.com/books/en/4.52.1.92/1/>
- [7] Technologie klávesnic. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation [cit. 2020-02-01]. Dostupné z: https://cs.wikipedia.org/wiki/Technologie_klávesnic
- [8] Dotyková obrazovka. *Datona.cz* [online]. [cit. 2020-02-01]. Dostupné z: https://www.datona.cz/content/dotyková_obrazovka
- [9] Jak funguje dotykový displej. *Okey.cz* [online]. [cit. 2020-02-01]. Dostupné z: <https://www.okay.cz/jak-funguje-dotykovy-displej/>
- [10] FERNÁNDEZ, María, Eduardo CASANOVA a Ignacio ALONSO. Review of Display Technologies Focusing on Power Consumption. *Sustainability* [online]. 2015, 7(8), 10854-10875 [cit. 2020-01-15]. DOI: 10.3390/su70810854. ISSN 2071-1050. Dostupné z: <http://www.mdpi.com/2071-1050/7/8/10854>

- [11] Technologie LCD panelů. *SVĚT HARDWARE* [online]. 2013 [cit. 2020-01-15]. Dostupné z: <https://www.svethardware.cz/technologie-lcd-panelu/14465-2>
- [12] Průvodce plochými panely: TN, IPS, MVA, který je lepší? *PC TUNING* [online]. 2005 [cit. 2020-01-15]. Dostupné z: <https://pctuning.tyden.cz/component/content/4509?task=view>
- [13] Electronic Ink. *Eink.com* [online]. [cit. 2020-02-01]. Dostupné z: <https://www.eink.com/electronic-ink.html>
- [14] EPaper, e-Ink, ... máme pro vás displej, který nahradí i papír. *SOSelectronic.com* [online]. 20.02.2017 [cit. 2020-02-01]. Dostupné z: <https://www.soselectronic.cz/articles/thingwell/epaper-e-ink-mame-pro-vas-displej-ktery-nahradi-i-papir-1873>
- [15] How E Ink's Triton Color Displays Work, In E-Readers and Beyond. *WIRED.com* [online]. 12.09.2010 [cit. 2020-02-01]. Dostupné z: <https://www.wired.com/2010/11/how-e-inks-triton-color-displays-work-in-e-readers-and-beyond/>
- [16] Memory LCD Technology. *Sharpsma.com* [online]. [cit. 2020-02-01]. Dostupné z: <https://www.sharpsma.com/sharp-memory-lcd-technology>
- [17] *Chip-on-Glass LCD Driver Technology* [online]. Červenec 2012, , 9 [cit. 2020-01-15]. Dostupné z: https://www.nxp.com/docs/en/white-paper/LCD_WhitePaper_D3.pdf
- [18] Serial Peripheral Interface. In: *Wikipedia: the free encyclopedia* [online]. Wikimedia Foundation, 2017 [cit. 2020-1-16]. Dostupné z: https://cs.wikipedia.org/wiki/Serial_Peripheral_Interface
- [19] *Externí sériové sběrnice SPI a I²C* [online]. In: . 30. 12. 2008 [cit. 2020-01-25]. Dostupné z: <https://www.root.cz/clanky/externi-seriove-sbernice-spi-a-i2c/>
- [20] DUDÁČEK, K. *Sériová rozhraní SPI, Microwire, I²C a CAN* [online]. 2002 [cit. 2020-01-25]. Dostupné z: http://home.zcu.cz/~dudacek/NMS/Seriova_rozhrani.pdf

- [21] Profibus Cable. In: *Indiamart* [online]. 2020 [cit. 2020-01-25]. Dostupné z: <https://www.indiamart.com/proddetail/profibus-cable-13652138873.html>
- [22] ZÁKLADNÍ INFORMACE O PRŮMYSLOVÉ SBĚRNICI PROFIBUS – ČÁST II. *Foxon.cz* [online]. 2013 [cit. 2020-01-30]. Dostupné z: <https://www.foxon.cz/blog/prakticka-teorie/163-zakladni-informace-o-prumyslove-sbernici-profibus-cast-ii>
- [23] FELSER, Max. *PROFIBUS Manual: A collection of information explaining PROFIBUS networks*. Berlín: epubli, 2011. ISBN 978-3-8442-1435-2.
- [24] *SÉRIOVÁ KOMUNIKACE PROGRAMOVATELNÝCH AUTOMATŮ TECOMAT - MODEL 32 BITŮ* [online]. 20. vydání. Teco a. s, 2013, s. 184 [cit. 2020-01-30]. Dostupné z: https://www.tecomat.cz/modules/DownloadManager/download.php?alias=txv00403_01_comm_serial32_cz
- [25] *VÝVOJOVÝ MODUL S 32BITOVÝM PROCESOREM TYPU ARM*. BRNO, 2009. Bakalářská práce. Vysoké učení technické v Brně. Vedoucí práce Ing. Tomáš Frýza, Ph.D.
- [26] Mikroprocesory s architekturou ARM. *Root.cz* [online]. [cit. 2020-01-15]. Dostupné z: <https://www.root.cz/clanky/mikroprocesory-s-architekturou-arm/>
- [27] Product Focus: Introducing the STM32L4 series of Ultra-Low Power ARM MCUs from STMicroelectronics. *AngliaLive* [online]. [cit. 2020-05-23]. Dostupné z: <https://www.anglia-live.com/ProductFocus/ProductFocus.aspx?pf=125>
- [28] *White Backlight Displays - Technical Data Sheet* [online]. Lucky Light Electronics Co., 2006, s. 6 [cit. 2020-04-12]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/KWB-R8445W-1W.pdf>
- [29] *4,2" e-Paper specification datasheet* [online]. Waveshare Electronics, 2017, s. 45 [cit. 2020-04-12]. Dostupné z: <https://www.waveshare.com/w/upload/6/6a/4.2inch-e-paper-specification.pdf>

- [30] *Technical Literature for TFT-LCD Module LS044Q7DH01* [online]. SHARP CORPORATION, 2011, s. 29 [cit. 2020-04-12]. Dostupné z: <https://media.digikey.com/pdf/Data%20Sheets/Sharp%20PDFs/LS044Q7DH01.pdf>
- [31] *LT3990/LT3990-3.3/LT3990-5 datasheet* [online]. LINEAR TECHNOLOGY CORPORATION, 2010, s. 23 [cit. 2020-04-12]. Dostupné z: <https://cz.mouser.com/datasheet/2/609/3990fa-1270774.pdf>
- [32] *Low Quiescent Current LDO MCP1700 datasheet* [online]. Microchip Technology, 2018, s. 31 [cit. 2020-04-12]. Dostupné z: <https://cz.mouser.com/datasheet/2/268/MCP1700-Low-Quiescent-Current-LDO-20001826E-737536.pdf>
- [33] *STM32L496RGT3 datasheet* [online]. STMicroelectronics, 2020, s. 281 [cit. 2020-04-12]. Dostupné z: <https://www.st.com/resource/en/datasheet/stm32l496rg.pdf>
- [34] *STM32L4x5 and STM32L4x6 advanced Arm®-based 32-bit MCUs Reference manual* [online]. STMicroelectronics, 2018, s. 1881 [cit. 2020-04-12]. Dostupné z: https://www.st.com/resource/en/reference_manual/dm00083560-stm32l4x5-and-stm32l4x6-advanced-armbased-32bit-mcus-stmicroelectronics.pdf
- [35] *4.2inch e-Paper Module - User Manual* [online]. Waveshare Electronics, 2018, s. 25 [cit. 2020-04-12]. Dostupné z: <https://www.waveshare.com/w/upload/2/20/4.2inch-e-paper-module-user-manual-en.pdf>

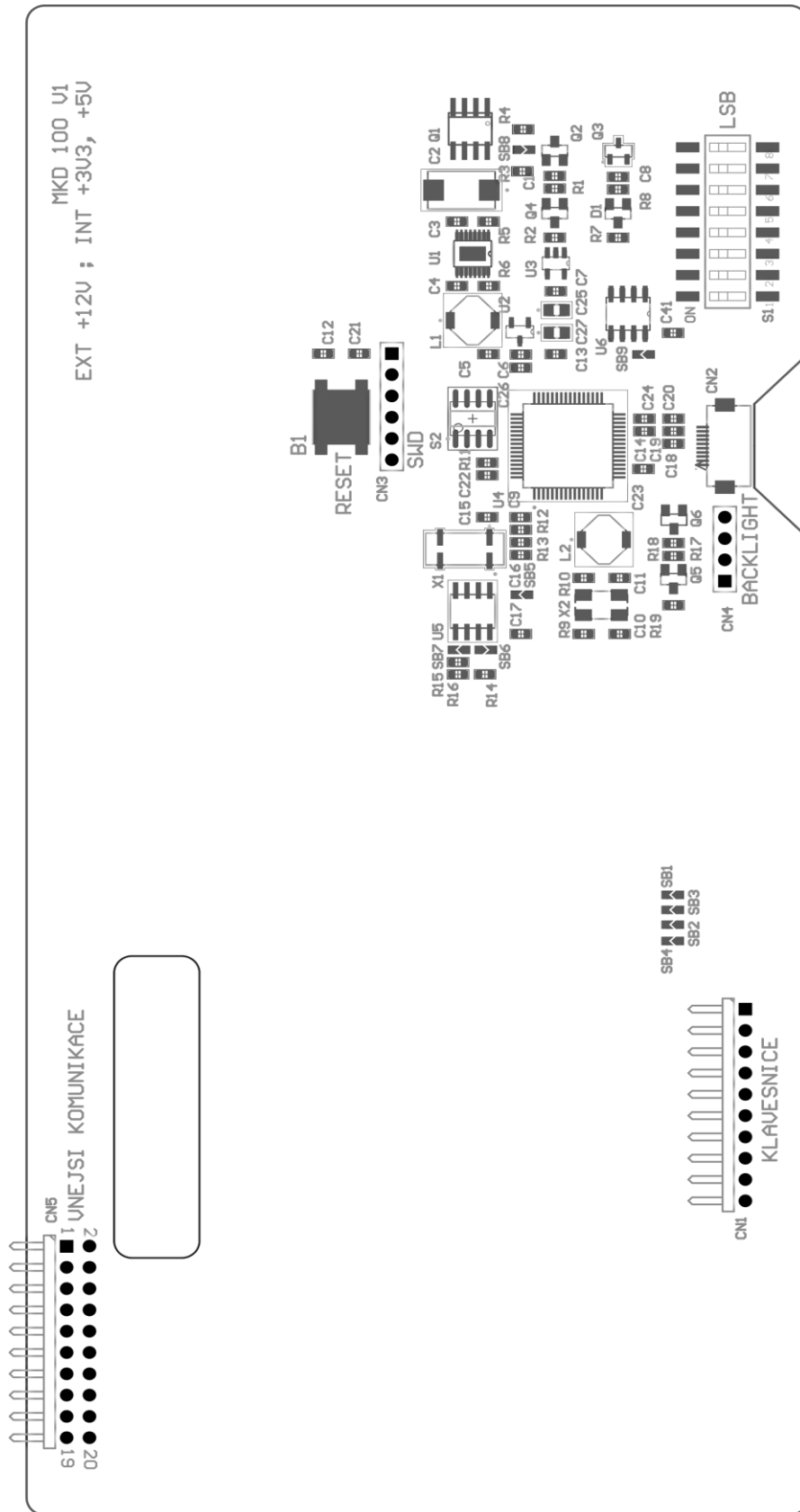
A Obsah příloženého CD

- Diplomová práce ve formátu PDF a DOCX
- Vytvořený program
- Vývojový diagram GUI ve formátu PDF a VSDX
- Katalogové listy
- Schéma navrženého obvodu
- Schéma desky plošného spoje
- Osazovací schéma
- 3D model čelního panelu

B Tabulka rámcových znaků v síti EPSNet

Znak	Význam znaku	Pevná hodnota
SD1	Úvodní znak 1 (start delimiter 1)	\$10
SD2	Úvodní znak 2 (start delimiter 2)	\$68
SD3	Definováno pouze v protokolu Profibus	
SD4	Úvodní znak 4 (start delimiter 4)	\$DC
LE	Délka dat (length), počet bytů položek DA+SA+FC+DATA (tj. 3...249)	
LER	Opakovaná délka dat (length repeat), počet bytů položek DA+SA+FC+DATA (tj. 3...249)	
SD2R	Opakovaný úvodní znak 2 (start delimiter 2 repeat)	\$68
DA	Cílová adresa (destination address), - dotazu pro podřízenou stanici 0...126 - v odpovědi pro nadřízenou stanici 0..126	
SA	Zdrojová adresa (source address), - dotazu pro nadřízenou stanici 0...126 - v odpovědi pro podřízenou stanici 0..126	
FC	Řídící byte rámce (frame control byte) - konkrétní hodnoty jsou uvedeny v citaci []	
DATA	Vlastní datové tělo zprávy (maximálně 246 bytů)	
FCS	Kontrolní součet (frame check sum) - bytový součet všech bytů položek DA+SA+FC+DATA	
ED	Koncový znak	\$16
SAC	Krátké potvrzení (short acknowledge)	\$E5

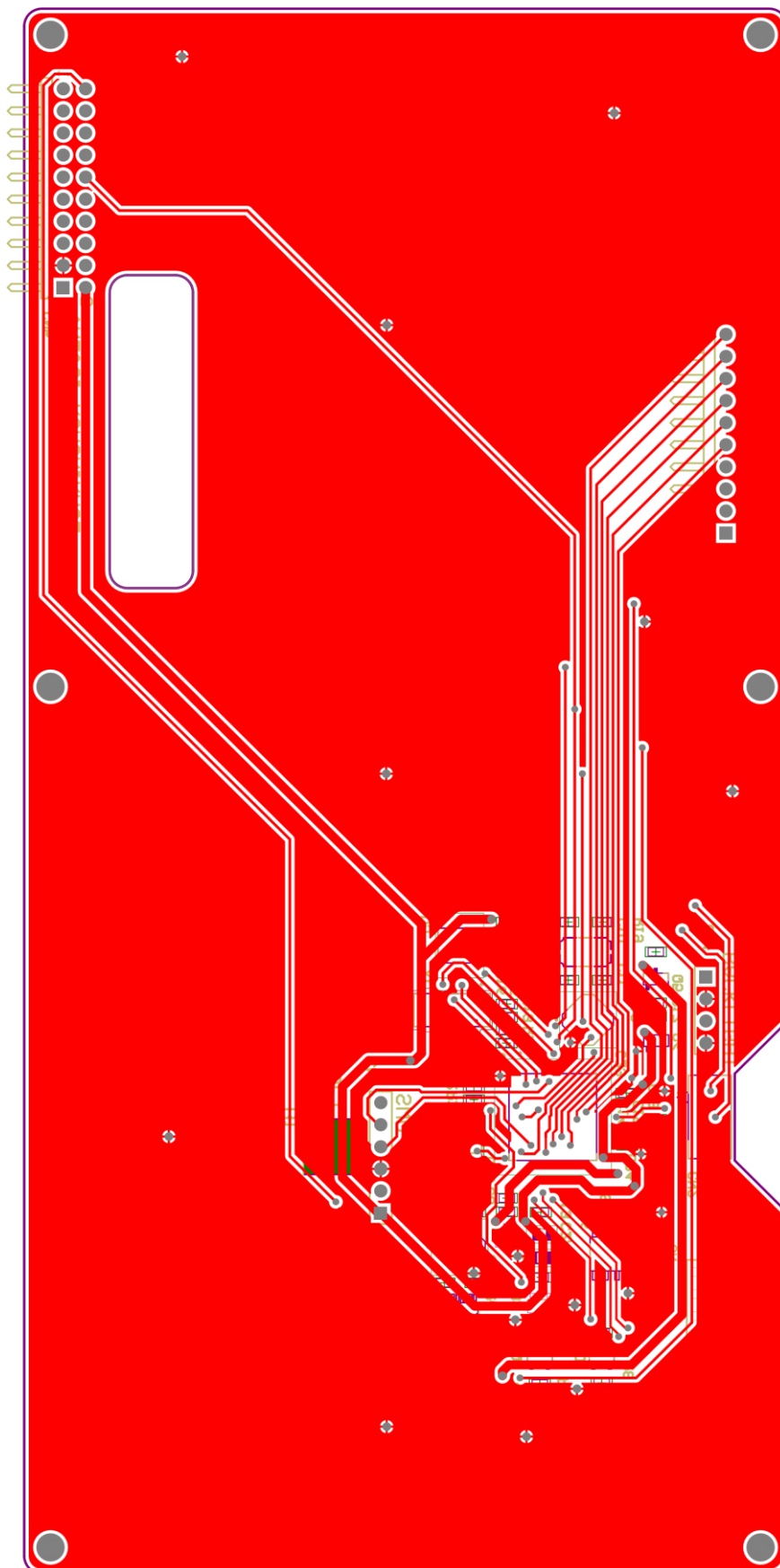
C Osazovací schéma



D Seznam součástek

Hodnota	Popis	Označení	Počet
RESET	SE PUSHBUTTON	B1	1
100nF	Capacitor	C1, C12, C17, C18, C21, C22, C23, C24, C26, C41	10
47uF/20V	Polarized Capacitor (Radial)	C2	1
2,2uF	Capacitor	C3, C4, C13, C14	4
22uF	Capacitor	C5	1
1uF	Capacitor	C6, C7, C8, C9, C19, C20	6
15uF	Capacitor	C10, C11	2
3,9pF	Capacitor	C15, C16	2
1uF	Polarized Capacitor (Radial) 0805	C25, C27	2
KLAVESNICE	Header, 10-Pin, Right Angle	CN1	1
SHARP_MEMORY	ZIF konektor, 10-Pin, female	CN2	1
SWD	Header, 6-Pin, Single row	CN3	1
PWM PODSVICENI	Header, 4-Pin	CN4	1
Header 10X2H	Header, 10-Pin, Dual row, Right Angle	CN5	1
BAS40	Diode Array 1 Pair	D1	1
33uH	Inductor	L1, L2	2
IRF7404	P-Channel 20V 6.7A	Q1	1
BC856	Bipolar _BJT_ Transistor PNP	Q2, Q4, Q5	3
BC817	Bipolar _BJT_ Transistor NPN	Q3	1
BSS138	N-Channel 50V 220mA	Q6	1
27K	Resistor	R1, R2, R3, R4, R7, R8	6
10K	Resistor	R5, R11	2
374K	Resistor	R6	1
100R	Resistor	R9, R10	2
10R	Resistor	R12, R13, R17	3
470R	Resistor	R14, R16	2
150R	Resistor	R15	1
22K	Resistor	R18	1
300R	Resistor	R19	1
ADRESA RS485	DIP Switch, 8 Position, SPST	S1	1
Mod procesoru	DIP Switch, 4 Position, SPST	S2	1
PROPOJKA		SB1, SB2, SB3, SB4, SB5, SB6, SB7, SB8, SB9	9
LT3990HMSE-5PBF		U1	1
MCP1700T-3302E/TT	Linear Voltage Regulator	U2	1
MC74HC1G04DTT1G	MC74HC1G04 Series 2 to 6 V	U3	1
STM32L496RGT3		U4	1
ADM3485		U5	1
24LC64-I/SM	24LC64 Series 64 Kb I2C 2	U6	1
LFX TAL016178REEL	Crystal	X1	1
ABM3B-8.000MHZ-10-1-U-T	8MHz $\hat{A}\pm 10$ ppm Crystal	X2	1

E DPS – strana TOP



F DPS – strana BOTTOM

