

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra počítačů

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Radim Horák**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Výpočetní technika

Název tématu: **Řídicí deska pro ovládání modulů měřících teplotu, vlhkost a tlak**

Pokyny pro vypracování:

Navrhněte a zrealizujte řídicí desku vhodnou pro ovládání modulů měřících neelektrické veličiny jako je teplota, vlhkost a tlak. Řídicí deska bude s modulem komunikovat s pomocí sběrnice RS485. Řídicí deska bude umožňovat komunikaci s PC a zároveň zaznamenávat data na SD kartu. Pro vytvořenou řídicí desku napište obslužný program.

Seznam odborné literatury:

Dodá vedoucí práce

Vedoucí: Ing. Pavel Kubalík, Ph.D.

Platnost zadání: do konce zimního semestru 2012/2013

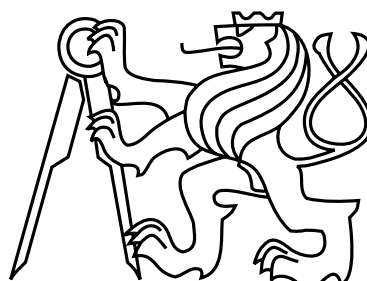
doc. Ing. Miroslav Šnorek, CSc.
vedoucí katedry



prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 14. 11. 2011

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačů



Bakalářská práce

**Řídící deska pro ovládání modulů měřících teplotu, vlhkost
a tlak**

Radim Horák

Vedoucí práce: Ing. Pavel Kubalík, Ph.D

Studijní program: Elektrotechnika a informatika, dobíhající, Bakalářský

Obor: Výpočetní technika

30. prosince 2011

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu. Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Mnichově Hradišti dne 30.12.2011.

.....

Poděkování

Rád bych zde poděkoval vedoucímu práce Ing. Pavlu Kubalíkovi PhD. za jeho ochotu okamžitě se mnou řešit problémy, které nastaly během tvorby této práce, jeho bleskovou odezvu při komunikaci a celkovou vstřícnost.

Zvláštní poděkování patří mým rodičům za jejich podporu po celou dobu mého studia.

Abstract

This bachelor thesis is focused on design and construction of development board for various data acquisition modules, measuring nonelectric variables like temperature, humidity and pressure. Development board will be designed to have low power consumption and small dimensions. And also has to be able to work with wide range of input voltage. All connected modules will use RS-485 bus to communicate with development board. All acquired data will be stored to SD card. It will be also possible to access all peripherals that microcontroller has to offer.

Abstrakt

Tato bakalářská práce se zabývá návrhem a realizací řídicí desky pro ovládání měřících modulů pro neelektrické veličiny jako jsou např. teplota, vlhkost a tlak. Řídicí deska bude vyvíjena tak, aby měla co nejmenší spotřebu a rozměry a zároveň velký rozsah napájecího napětí. S připojenými moduly bude komunikovat po sběrnici RS-485 a získaná data budou ukládána na SD kartu. Dále bude možno využít různé periférie, které vybraný mikrokontrolér nabízí.

Obsah

1	Úvod	1
2	Popis problému, specifikace cíle	3
3	Analýza a návrh řešení	5
3.1	Nabídka trhu	6
3.1.1	Konfigurovatelný řídicí modulární systém	6
3.1.2	Vývojový kit EvB 4.3	7
3.2	Výběr ovládaného modulu	7
3.3	Návrh desky	8
3.3.1	Mikrokontrolér	9
3.3.2	RTC	10
3.3.3	Ethernetový řadič	10
3.3.4	Převodník USART<->RS-232	11
3.3.5	Převodník USART<->RS-485	11
3.3.6	SD karta	12
3.3.7	Napájecí zdroj	12
3.3.7.1	Spotřeba obvodů napájených napětím 3,3V	12
3.3.7.2	Spotřeba obvodů napájených napětím 5V	12
3.3.8	Náklady na výrobu	12
3.3.8.1	Náklady na výrobu 1 kusu	13
3.3.8.2	Náklady na výrobu 100 kusů	13
4	HW Realizace	15
4.1	Mikrokontrolér	15
4.2	Ethernetový řadič	16
4.3	RTC	18
4.4	Převodník USART<->RS-232	19
4.5	Převodník USART<->RS-485	20
4.6	SD karta	21
4.7	LCD Displej	21
4.8	LED port	22
4.9	Rozšiřující konektory	23
4.10	Napájecí zdroj	24

5 SW Realizace	25
5.1 Vývojové prostředí	25
5.2 Obsluha sériové linky	25
5.3 Obsluha sběrnice RS-485	26
5.3.1 Komunikační protokol ovládaného modulu	26
5.4 Komunikace s RTC obvodem	28
5.5 Obsluha SD karty	29
5.6 Ovládací konzole	29
6 Testování a oživení	31
7 Závěr	33
Literatura	33
8 Seznam použitých zkratk	39
A Schéma desky	41
B DPS a rozmístění součástek	47
C Osazená deska	51
D Knihovny funkcí	53
D.1 485.c	53
D.1.1 Potřebné definice	53
D.1.2 Funkce	54
D.2 TWI.c	56
D.2.1 Potřebné definice	56
D.2.2 Funkce	56
E Obsah přiloženého CD	57

Seznam obrázků

3.1	Konfigurovatelný řídicí modulární systém[18]	6
3.2	Vývojový kit EvB 4.3[28]	7
3.3	Vybraný ovládaný modul	7
3.4	Blokové schéma řídicí desky	8
3.5	Atmel ATxmega128A1[29]	10
3.6	Epson RTC-8564[30]	10
3.7	Ethernetový řadič ENC424J600[31]	11
3.8	Převodník Max3232[32]	11
3.9	SN65HVD3082E[27]	11
4.1	Zapojení JTAG konektoru	15
4.2	Zapojení mikrokontroléru ATxmega128A1	16
4.3	Zapojení Ethernetového řadiče	17
4.4	Zapojení galvanického oddělení Ethernetového řadiče od zbytku sítě	18
4.5	Zapojení obvodu RTC	19
4.6	Zapojení převodníku MAX3232	19
4.7	Zapojení převodníku USART<->RS-485	20
4.8	Zapojení konektoru pro SD kartu	21
4.9	Zapojení konektoru pro LCD displej	22
4.10	Zapojení LED diod	23
4.11	Zapojení rozšiřujícího konektoru 1	23
4.12	Zapojení rozšiřujícího konektoru 2	23
4.13	Zapojení napájecího zdroje 5V	24
4.14	Zapojení napájecího zdroje 3,3V	24
5.1	Obecná struktura paketu	27
5.2	Screenshot ovládací konzole	29
A.1	Schéma zapojení mikrokontroléru, MAX3232, LED portu, LCD portu a konektorů	42
A.2	Schéma zapojení ethernetového řadiče	43
A.3	Schéma zapojení převodníku pro sběrnici RS-485 a obvodu RTC	44
A.4	Schéma zapojení SD karty a napájecích zdrojů	45
B.1	Vrstva TOP	48
B.2	Rozmístění součástek TOP	48

B.3	Vrstva BOT	49
B.4	Rozmístění součástek BOT	49
B.5	Neosazená deska vrstva TOP	50
B.6	Neosazená deska vrstva BOT	50
C.1	Osazená deska vrstva TOP	52
C.2	Osazená deska vrstva BOT	52

Seznam tabulek

3.1	Výběr mikrokontroléru	9
3.2	Spotřeba - napájecí napětí 3,3V	12
3.3	Spotřeba - napájecí napětí 5V	12
3.4	Náklady na výrobu jednoho kusu	13
3.5	Náklady na výrobu 100 kusů	13
4.1	Piny mikrokontroléru využité pro připojení Ethernetového řadiče k procesoru	17
4.2	Zapojení konektoru RJ45	17
4.3	Piny využité pro připojení RTC k procesoru	19
4.4	Piny využité pro připojení převodníku MAX3232 k procesoru	20
4.5	Piny využité pro připojení převodníku USART<->RS-485 k procesoru	20
4.6	Piny využité pro připojení SD karty k procesoru	21
4.7	Piny využité pro připojení LCD displeje k procesoru	22
4.8	Piny využité pro připojení LED diod k procesoru	22
5.1	Klíčová slova protokolu	27
5.2	Příkazy ovládací konzole	29

Kapitola 1

Úvod

Úkolem práce je navrhnout a zrealizovat vhodnou řídicí desku pro ovládání modulů měřících neelektrické veličiny jako je teplota, vlhkost a tlak. Řídicí deska bude navržena tak, aby měla co nejmenší spotřebu a měla velký rozsah napájecího napětí.

Na trhu je dnes mnoho modulů, které měří neelektrické veličiny a komunikují po sběrnici RS-485[21]. Setkat se s nimi můžeme v podobě malé meteostanice, nebo malého zařízení někde ve výrobní hale. Řídicí desky pro odběr a zároveň zpracování hodnot získaných z modulů mají na trhu mnohem menší zastoupení. Proto je často v pozadí připojen k modulu počítač, který data zpracovává. Takovéto řešení je hodně nákladné.

Navrhovaná deska bude hardwarově připravená pro komunikaci s moduly a načtená data bude ukládat na SD kartu. Komunikace mezi řídicí deskou a moduly bude probíhat po sběrnici RS-485. Řídicí desku bude možné ovládat pomocí jednoduché ovládací konzole, která bude dostupná přes sběrnici RS-232. Informace o stavu desky bude možné sledovat na LCD displeji. Řídicí deska bude navržena tak, aby byla využitelná i pro jiné aplikace. Vznikne tak široce využitelná univerzální vývojová deska.

Kapitola 2

Popis problému, specifikace cíle

Hlavním úkolem je vytvořit řídicí desku pro ovládání modulů měřících neelektrické veličiny, které komunikují po sběrnici RS-485[21].

- Navrhnout a zrealizovat řídicí desku.
- Naprogramovat firmware pro řídicí desku.

Kapitola 3

Analýza a návrh řešení

V dnešní době se setkáváme s různými systémy pro sběr dat z modulů, které měří neelektrické veličiny. Často se jedná o uzavřené systémy, které jsou vyvinuty pouze pro určité typy měřících modulů. Existují ale i moduly, které nejsou určeny výhradně pro uzavřený systém sběru dat. Takovéto moduly mají pouze nějakou výstupní sběrnici (RS-232, RS-485[21], Ethernet apod.) a uživatel si k nim systém pro odběr dat může sám doplnit, nebo dodatečně zakoupit. Pro uživatelské doplnění lze obvykle použít i moduly z uzavřených systémů.

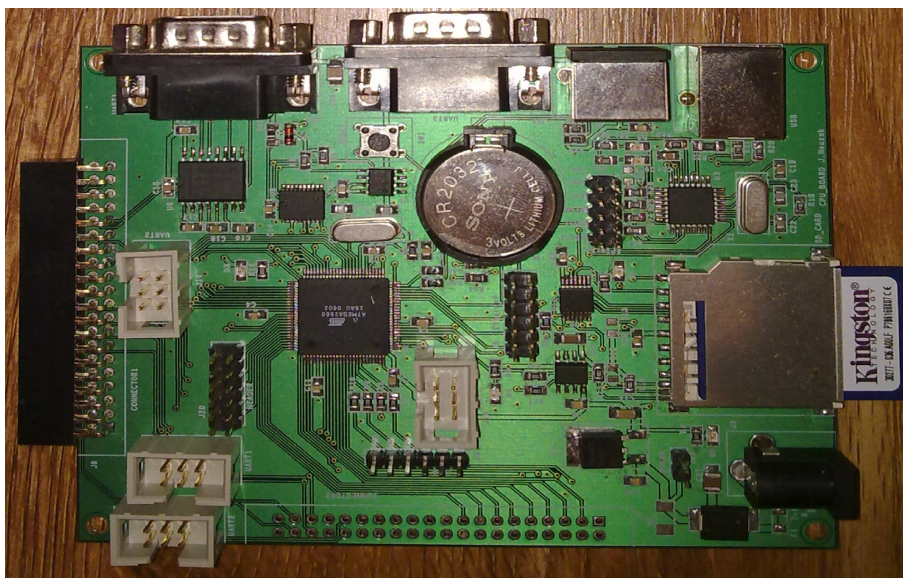
Existující uzavřené systémy jsou určeny pro sledování čidel např. v rodinném domě. V domě bývá umístěna centrální jednotka, většinou osazena velkým LCD displejem a základními ovládacími prvky. Tato jednotka je spojena s čidly, komunikace mezi nimi a systémem pak probíhá buď bezdrátově, nebo po nějaké datové sběrnici. S centrální jednotkou lze také často komunikovat přes internet za pomoci specializovaného programu v počítači, nebo vyčítáním stavových hodnot přes SNMP protokol v nějakém monitorovacím systému.

3.1 Nabídka trhu

V dnešní době je trh zahlcen velkým množstvím vývojových desek. Při hledání desky vhodné pro mou práci jsem se zaměřil výhradně na ty, které splňují požadované technické parametry (minimálně RS-485[21], RS-232[22] a SD karta). Z široké nabídky jsem vybral dvě, které jsou snadno dostupné nebo vyrobitebné. První z nich je Konfigurovatelný řídicí modulární systém, který navrhl ve své Diplomové práci Ing. Josef Nouzák[18]. Druhou vývojovou deskou je EVB 4.3[6] od firmy ANT-TECH.pl.

3.1.1 Konfigurovatelný řídicí modulární systém

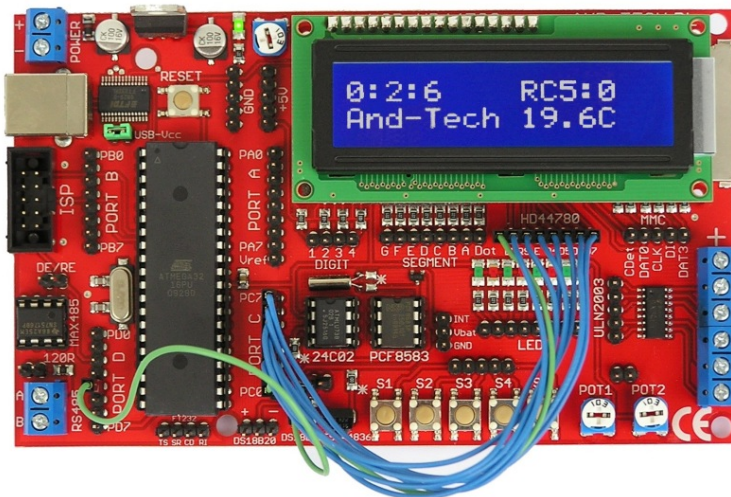
Konfigurovatelný řídicí modulární systém (obr.3.1) se skládá z procesorové desky a přídatných rozšiřujících modulů. Hlavní je mikrokontrolerová deska, která obsahuje všechny potřebné periférie (RS-232[22], SD kartu, RTC). Sběrnice RS-485[21] a Ethernet jsou přístupné po připojení jednotlivých rozšiřujících modulů. Náklady na pořízení desky se pohybují okolo částky 1640 Kč.



Obrázek 3.1: Konfigurovatelný řídicí modulární systém[18]

3.1.2 Vývojový kit EvB 4.3

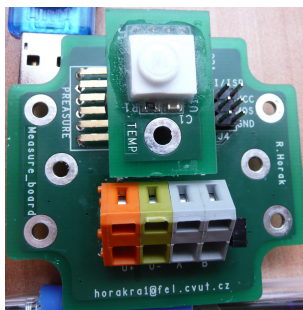
Vývojový kit EvB 4.3[6] (obr.3.2) od Polského výrobce AND-TECH.pl obsahuje také všechny potřebné periférie a ještě mnoho dalších. Až na Ethernet jsou všechny potřebné periférie součástí desky. Kit je dostupný ve třech verzích, které se liší osazenými procesory. Cena kitu s procesorem ATmega644P je 1644 Kč.



Obrázek 3.2: Vývojový kit EvB 4.3[28]

3.2 Výběr ovládaného modulu

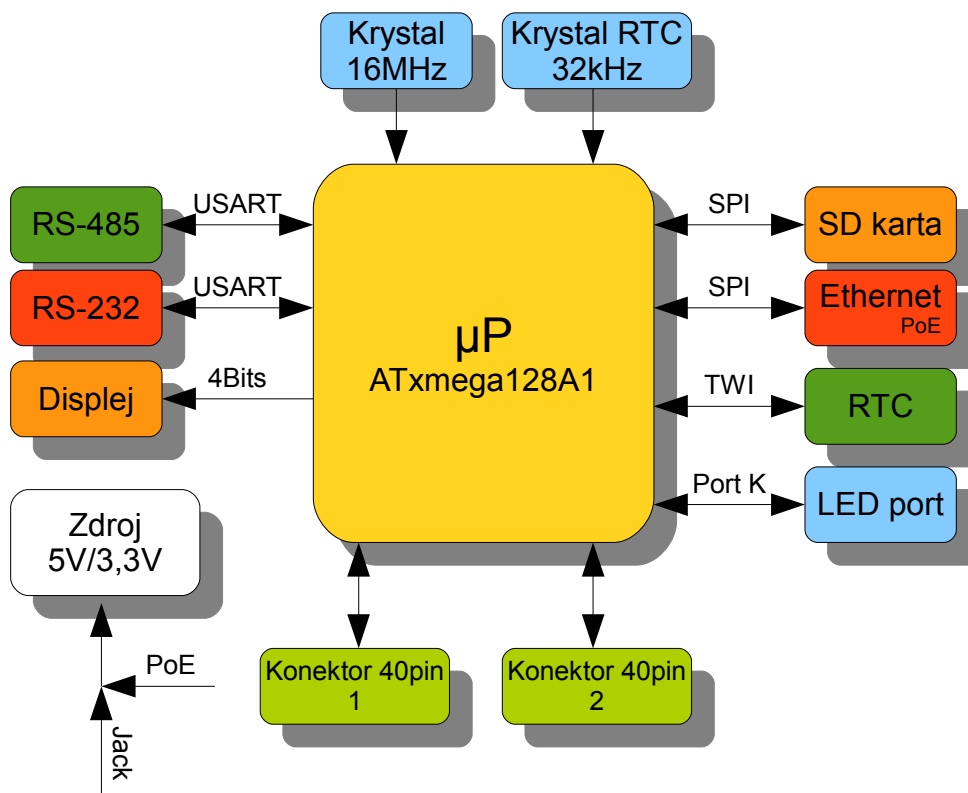
Řídicí deska bude po hardwarové stránce připravena komunikovat s jakýmkoliv modulem, po sběrnici RS-485[21]. Pro předvedení funkčnosti komunikace jsem jako ovládaný modul vybral pro mě snadno dostupný, který jsem sám navrhl a vytvořil (obr.3.3). Tento modul měří teplotu, vlhkost a tlak. Komunikuje pomocí vyžadované sběrnice RS-485[21] jednoduchým komunikačním protokolem. Rozsah napájecího napětí je 9-25V DC. Modul je osazen dvěma čidly. První je STH15[7] od firmy Sensirion, které měří teplotu a vlhkost. Druhé je čidlo MPX4115[8] pro měření tlaku od firmy Freescale Semiconductor.



Obrázek 3.3: Vybraný ovládaný modul

3.3 Návrh desky

Řídící deska bude navržena tak, aby se dala použít i k jiným aplikacím, než jen pro ovládání měřících modulů. z tohoto důvodu bude přidán ještě Ethernetový řadič, protože díky rozmachu internetu je dnes Ethernet velmi využíván. Blokové schéma je na následujícím obrázku (obr.3.4).



Obrázek 3.4: Blokové schéma řídicí desky

Deska bude mít dva rozšiřující 40-pinové konektory. Na konektory budou vyvedeny porty mikrokontroléru včetně napájecích napětí 3,3V a 5V. Jeden konektor bude obsahovat porty ABCD a druhý EFHJ. Použitý konektor bude shodný s konektorem, který se využívá pro připojení PATA disků v PC. K propojení tedy lze použít 40 žilový kabel a odpadá tak nutnost jeho výroby.

Aby bylo možné komunikovat mezi deskou a počítačem, bude na desce umístěn převodník USART<->RS-232. Pro komunikaci mezi deskou a ovládanými moduly bude na desce převodník USART<->RS-485.

Ethernetový řadič a SD karta budou připojeny pomocí SPi sběrnice. Aby bylo možné zaznamenávat naměřená data s časovou značkou, bude k mikrokontroléru přes TWI sběrnici připojen RTC obvod se zálohovaným napájením pomocí samostatné baterie.

Jako stavový zobrazovací prvek bude použit LCD displej, který bude možné připojit pomocí 10 pinového konektoru. Pro jednodušší zobrazení stavů bude možné využít i 8 LED diod, např. pro signalizaci chodu hodin.

Pro případné využití vnitřního RTC obvodu bude k mikrokontroléru připojen krystal 32,768kHz z důvodu lepší přesnosti. Z důvodu dřívější špatné zkušenosti s vnitřním RC oscilátorem, který vlivem teploty měnil svoji frekvenci kmitání, bude osazen ještě krystal 16MHz. Bude tak zaručena možnost využít desku i ve chladném prostředí.

Na desce se budou nacházet obvody, které potřebují napájení 3,3V, ale i 5V, proto budou na desce dva zdroje. Deska může být napájena buď pomocí adaptéru, nebo pomocí PoE[20] (Power over Ethernet).

3.3.1 Mikrokontrolér

Požadavky na mikrokontrolér jsou následující:

- USART
- SPI
- TWI
- Nízká spotřeba

Jako mikrokontrolér byl vybrán mikrokontrolér ATxmega128A1[9] od firmy Atmel. Jako druhá možnost byl mikrokontrolér ATmega2560 od stejné firmy. Srovnání obou mikrokontrolérů je v následující tabulce (tab.3.1).

	ATxmega128A1	ATmega2560
USART	8x	4x
SPi	4x	1x
TWi	4x	1x
Paměť	128kB	256kB
SRAM	32kB	8kB
EEPROM	4kB	4kB
Cena(bez DPH)	254Kč	303Kč

Tabulka 3.1: Výběr mikrokontroléru

Zvolený mikrokontrolér ATxmega128A1[9] (obr.3.5) má více periférií a tak má větší potenciál pro vývojovou desku. Je také novější a tak bude nejspíše déle dostupný na trhu.



Obrázek 3.5: Atmel ATxmega128A1[29]

3.3.2 RTC

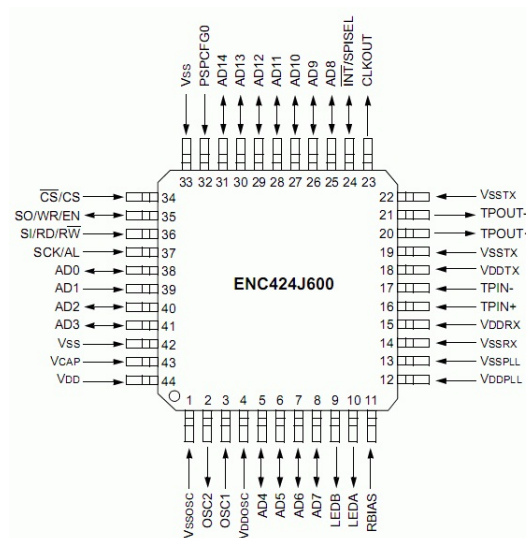
Obvod pro RTC by měl mít vnitřní krystal a možnost připojit bateriovou zálohu. Komunikace s procesorem buď po TWI sběrnici, nebo SPI. Požadavkům vyhovuje obvod RTC-8564[10] (obr.3.6) od firmy Epson. Obvod splňuje všechny požadavky a s jeho dostupností na trhu není problém.



Obrázek 3.6: Epson RTC-8564[30]

3.3.3 Ethernetový řadič

Vybraným obvodem pro ethernetový řadič je ENC424J600[11] (obr.3.7) od firmy Microchip. Umožňuje jednoduše vyřešit konektivitu pomocí rychlého Ethernetu. Komunikace s obvodem je možná po sériové SPi sběrnici (max. 14Mbit/s), ale i pomocí rychlé paralelní sběrnice, která nabízí rychlejší komunikaci (až 160Mbit/s). Řadič podporuje komunikaci standard 10Base-T i 100Base-T, dále splňuje specifikaci IEEE 802.3. Pro autentizaci je možné využít podpory SSL, TSL a některé další protokoly kryptografie. Mezi další podporované algoritmy patří podpora RSA, AES a SHA-1 algoritmů. V řadiči je předprogramována unikátní MAC adresa. Řadič má integrovanou 24kB RAM, která je použita pro Rx a Tx buffer. Lze ji použít i pro uložení libovolných dat z procesoru a rozšířit tak jeho paměť.



Obrázek 3.7: Ethernetový řadič ENC424J600[31]

3.3.4 Převodník USART<->RS-232

Pro převodník USART<->RS-232 je vhodné použít osvědčené převodníky od firmy Maxim. Vybral jsem převodník MAX3232[12] (obr.3.8) v pouzdře SO-16. tyto převodníky jsou běžně dostupné za přijatelnou cenu.



Obrázek 3.8: Převodník Max3232[32]

3.3.5 Převodník USART<->RS-485

Protože mikrokontrolér nepodporuje komunikaci po sběrnici RS-485[21], bude nutné připojit převodník USART<->RS-485. Zvolil jsem převodník SN65HVD3082ED[13] (obr.3.9) od firmy Texas Instruments. Tento převodník je použitý i u vybraného modulu. Pracovní rozsah teplot je -40°C až 85°C. Spotřeba je jen 0,3mA při práci a v režimu spánku dokonce jen 1nA.



Obrázek 3.9: SN65HVD3082E[27]

3.3.6 SD karta

Pro připojení SD karty k procesoru není potřeba žádný další obvod, stačí jen konektor pro SD kartu. Výběr konektoru není nijak složitý, podmínkou byl pouze kvalitní datasheet s popsánymi rozměry konektoru.

3.3.7 Napájecí zdroj

Pro hlavní napájení desky bude použit spínaný step-down regulátor LM2575-5[15], který je schopen dodat napětí 5V a proud 1A. Vstupní napětí v rozsahu 9-40V. Napájení 3,3V části bude zajištěno pomocí lineárního stabilizátoru LM1117DT-3,3[16] (3,3V/800mA), který bude napájen 5V z regulátoru LM2575-5. Napájecí Zdroj bude dostačovat i pro napájení přídatných modulů, které dohromady s deskou nepřesáhnou možnosti použitých obvodů.

3.3.7.1 Spotřeba obvodů napájených napětím 3,3V

obvod	spotřeba
ATxmega128A1	20mA
RTC	1mA
Ethernet	20mA
MAX3232	10mA
SD karta	70mA
8xLED	31mA
celkem	152mA

Tabulka 3.2: Spotřeba - napájecí napětí 3,3V

3.3.7.2 Spotřeba obvodů napájených napětím 5V

obvod	spotřeba
Displej + podsvícení	290mA
SN65HVD3082ED	0,3mA
celkem	290,3mA

Tabulka 3.3: Spotřeba - napájecí napětí 5V

3.3.8 Náklady na výrobu

Ceny jsou převzaty z obchodu TME[25], plošné spoje jsou vyrobené firmou Pragoboard s.r.o.[26]. Uvedené ceny jsou ceny za jednotlivé bloky součástek (např. Ethernet = Etherneťový řadič, konektor RJ45 atd.).

3.3.8.1 Náklady na výrobu 1 kusu

položka	Cena (bez DPH)
ATxmega128a1	212,15Kč
Ethernet	175,20Kč
RS-485	49,72Kč
RS-232	44,77Kč
SD karta	20,00Kč
Zdroje	130,50Kč
LED port	27,30Kč
RTC	88,81Kč
Drobné další součástky	20Kč
Celkem za PCB (bez filmů pro výrobu)	500,00Kč
CELKOVÁ CENA 1Ks	1268,00Kč

Tabulka 3.4: Náklady na výrobu jednoho kusu

3.3.8.2 Náklady na výrobu 100 kusů

položka	Cena (bez DPH)
ATxmega128a1	156,16Kč
Ethernet	120,22Kč
RS-485	30,19Kč
RS-232	23,18Kč
SD karta	20,00Kč
Zdroje	75,00Kč
LED port	25,30Kč
RTC	75,00Kč
Drobné další součástky	20Kč
Celkem za PCB (odhad ceny)	260,00Kč
CELKOVÁ CENA 1Ks	785,00Kč

Tabulka 3.5: Náklady na výrobu 100 kusů

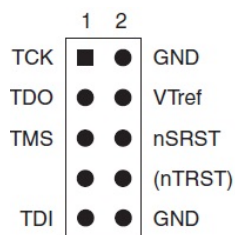
Kapitola 4

HW Realizace

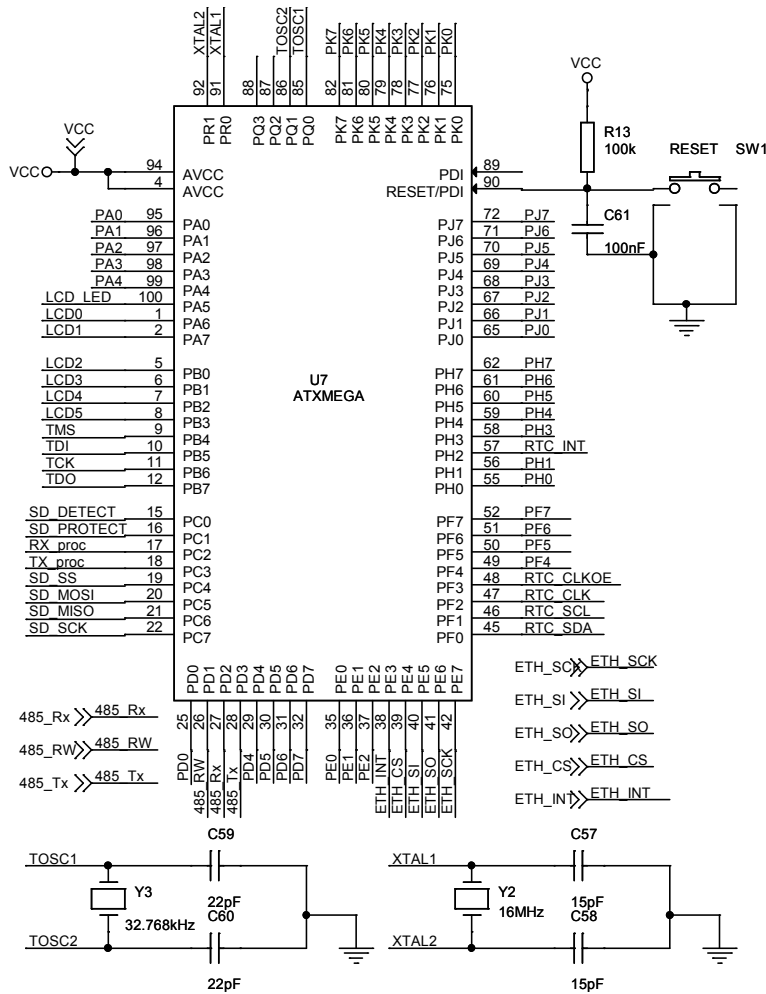
Schéma zapojení je v příloze pod písmenem A (A), vrstva TOP a BOTTOM společně s rozmístěním součástek je v příloze B (B), rozměry jsou v palcích (inch). Fotografie osazené desky je v příloze C (C). Řídící deska je navrhována tak, aby byla co nejmenší (maximálně 1 dm^2). Aby se řídící deska ze spodní strany nepoškodila bude přišroubována na plexisklo pomocí čtyř šroubů M3, mezera mezi plexisklem a deskou bude zajištěna pomocí distančních sloupků.

4.1 Mikrokontrolér

Srdcem desky je mikrokontrolér ATxmega128A1[9], který je zapojen dle doporučení výrobce (obr.4.2). K procesoru je připojen krystal 16MHz, aby byla zajištěna teplotní nezávislost v chladném prostředí. Druhým připojeným krystalem je krystal 32,768kHz, určený pro vnitřní RTC mikrokontroléru, který je možné využít např. pro Real-Time operace. Vnitřní RC oscilátor 32,768kHz nebyl natolik přesný, aby se dal využít. U každého napájecího pinu jsou osazeny blokovací kondenzátory (100nF a 820pF). Pro potřeby resetu mikrokontroléru je vyvedeno resetovací tlačítko. mikrokontrolér se programuje pomocí JTAG[24] rozhraní, které je vyvedeno na 10-pinový konektor (obr.4.1) a zapojeno dle doporučení výrobce.



Obrázek 4.1: Zapojení JTAG konektoru



Obrázek 4.2: Zapojení mikrokontroléru ATxmega128A1

4.2 Ethernetový řadič

Ethernetový řadič ENC424J600 (obr.3.7) je zapojen dle doporučení výrobce (obr.4.3). K procesoru je připojen (tab.4.1) pomocí SPI sběrnice (SPIE). Obvod U4 (obr.4.4) zajišťuje galvanické oddělení od sítě. Prodává se i zabudovaný v konektoru RJ45, ale nevyužité piny (4,5,7,8) bývají takřka vždy připojeny na stínění, proto by nebyl takový konektor vhodný pro potřeby využití pasivního PoE[20] (Power over Ethernet).

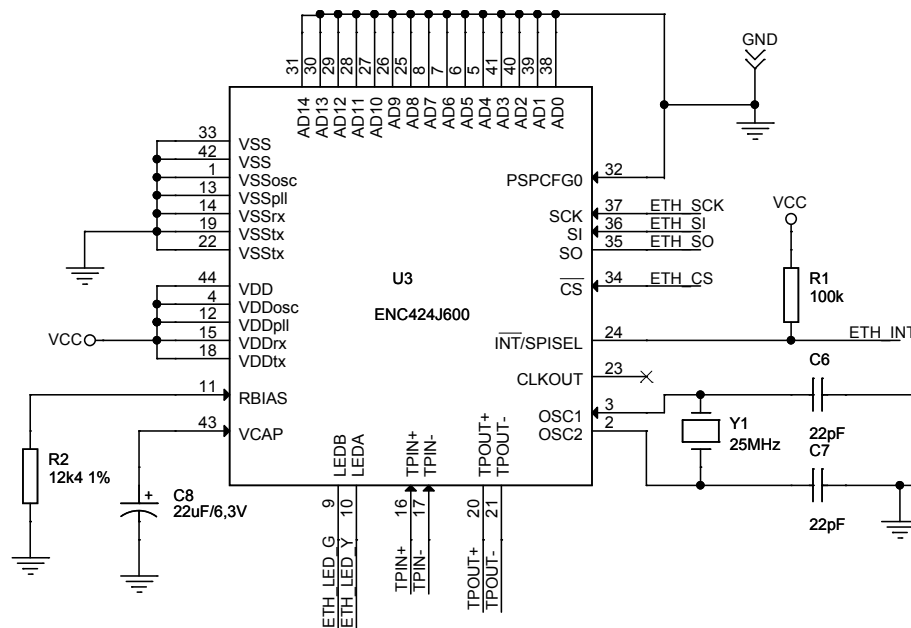
Pasivní PoE[20] využívá dvou volných párů a splňuje normu IEEE 802.3af[19]. Propojka J24 umožňuje zapnout/vypnout možnost napájení desky pomocí pasivního PoE. Dioda D5 zamezuje, aby vstupní napětí připojené napájecí konektor šlo dále po Ethernetovém kabelu. Naopak dioda D4 zamezuje, aby se napětí přivedené pomocí PoE[20] vyskytovalo na napájecím konektoru. Piny 4 a 5 konektoru RJ45 (tab.4.2) jsou využity pro připojení kladného pólu napětí PoE[20], piny 7 a 8 jsou využity pro záporný pól napětí PoE[20].

Signál	Pin_CPU
ETH_INT	PH3
ETH_CS	PH4
ETH_Si	PH5
ETH_SO	PH6
ETH_SCK	PH7

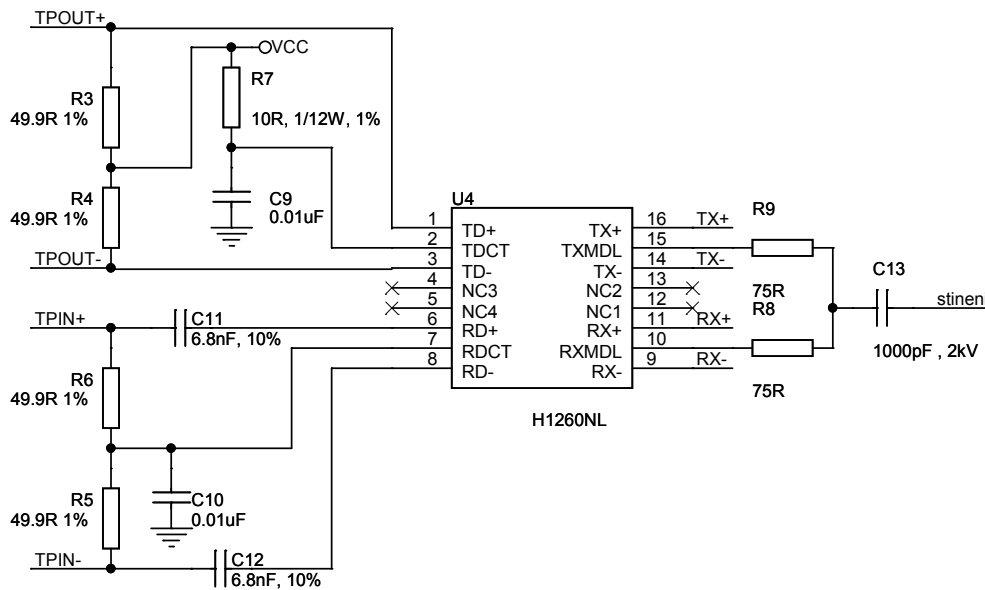
Tabulka 4.1: Piny mikrokontroléru využité pro připojení Ethernetového řadiče k procesoru

Signál	Pin_CPU
1	Tx+
2	Tx-
3	Rx+
4	PoE+
5	PoE+
6	Rx-
7	PoE-
8	PoE-

Tabulka 4.2: Zapojení konektoru RJ45



Obrázek 4.3: Zapojení Ethernetového řadiče



Obrázek 4.4: Zapojení galvanického oddělení Ethernetového řadiče od zbytku sítě

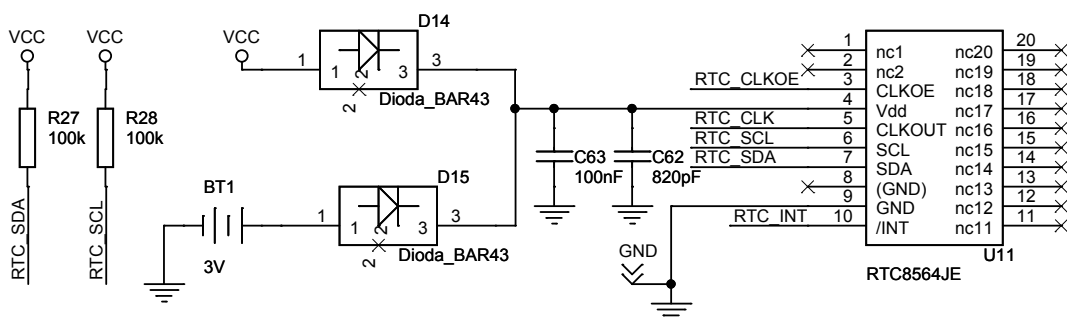
4.3 RTC

Použitý obvod RTC-8564 od firmy Epson má vestavěný kalibrovaný krystal a není proto třeba přidávat externí. Je možné nastavit výstup frekvence 1Hz, 32Hz, 1024Hz, nebo 32768kHz. Mezi další funkce RTC obvodu patří vnitřní kalendář a alarm. Výstup frekvence (RTC_CLK) a alarm (RTC_INT) generuje přerušení pro mikrokontrolér. Obvod (obr.4.5) je napájen napětím 3,3V a zároveň z baterie (3V), která je zapojena paralelně k napájení 3,3V. Baterie zajišťuje napájení RTC, když je deska odpojena od napájecího napětí. Dioda D14 zamezuje, aby se napětí z baterie šířilo dále po desce v případě, že deska není připojena na napájecí napětí. Naopak dioda D15 je zapojena tak, aby napětí 3,3V směřovalo do baterie. K napájecímu pinu jsou paralelně připojeny blokovací kondenzátory C63 (100nF) a C62 (820pF).

S RTC obvodem mikrokontrolér komunikuje po sběrnici TWI (RTC_SDA, RTC_SCL), využita je TWI sběrnice na portu F (TWIF). Adresa pro zápis je 0xA2 a 0xA3 pro čtení. Osazeny jsou pull-up rezistory R27 a R28. Aby bylo možné využít výstup frekvence, je třeba připojit RTC_CLK a RTC_CLKOE, kterým se povoluje výstup frekvence (log. 1 - výstup povolen). Pro využití přerušení generovaného alarmem je třeba připojit RTC_INT. Připojení jednotlivých signálů je uvedeno v následující tabulce (tab.4.3).

Signál	Pin_CPU
RTC_SDA	PF0
RTC_SCL	PF1
RTC_CLK	PF2
RTC_CLKOE	PF3
RTC_INT	PH2

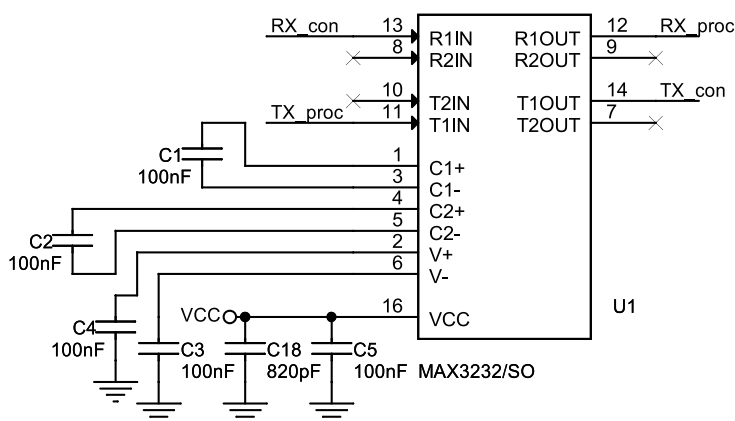
Tabulka 4.3: Piny využité pro připojení RTC k procesoru



Obrázek 4.5: Zapojení obvodu RTC

4.4 Převodník USART<->RS-232

Převodník MAX3232 (obr.4.6) je zapojen dle doporučení výrobce. Jako konektor je použit standardní 9-pinový CANNON (female). Pro připojení k PC je potřeba použít nekřížený kabel. Obvod je spojen s procesorem pomocí sériové linky USARTC0. Využité piny mikrokontroléru jsou v následující tabulce (tab.4.4).



Obrázek 4.6: Zapojení převodníku MAX3232

Signál	Pin_CPU
RX_proc	PC2
TX_proc	PC3

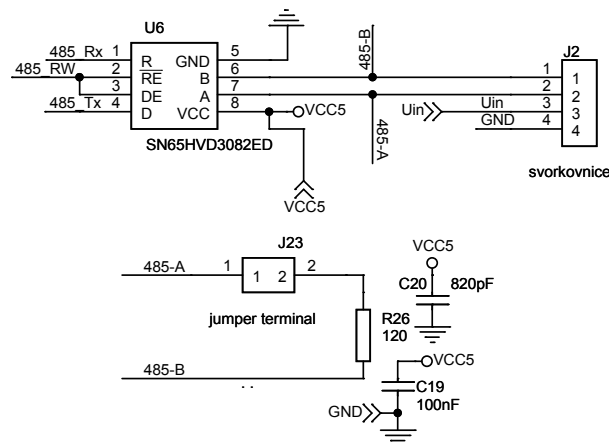
Tabulka 4.4: Piny využití pro připojení převodníku MAX3232 k procesoru

4.5 Převodník USART<->RS-485

Obvod převodníku je zapojen dle doporučení výrobce (obr.4.7). Napájecí napětí (5V) je přivedeno na pin č.8, včetně blokovacích kondenzátorů C19 (100nF) a C20 (820pF). Obvod je spojen s procesorem pomocí sériové linky USARTD0 (tab.4.5). Vysílání, nebo příjem dat po sběrnici RS-485[21] je řízeno signálem 485_RW (spojený pin č.2 a č.3), při log. 1 je zapnut mód vysílání. Rezistor R26 (120Ω) slouží jako zakončovací rezistor sběrnice RS-485[21]. Do činnosti se uvede propojením propojky J23. Zakončovací rezistor musí být na obou koncích sběrnice. Sběrnice je vyvedena na svorkovnici, kam je přivedeno i vstupní napětí, pomocí kterého tak lze napájet i připojený modul.

Signál	Pin_CPU
485_RW	PD1
485_Rx	PD2
485_Tx	PD3

Tabulka 4.5: Piny využití pro připojení převodníku USART<->RS-485 k procesoru



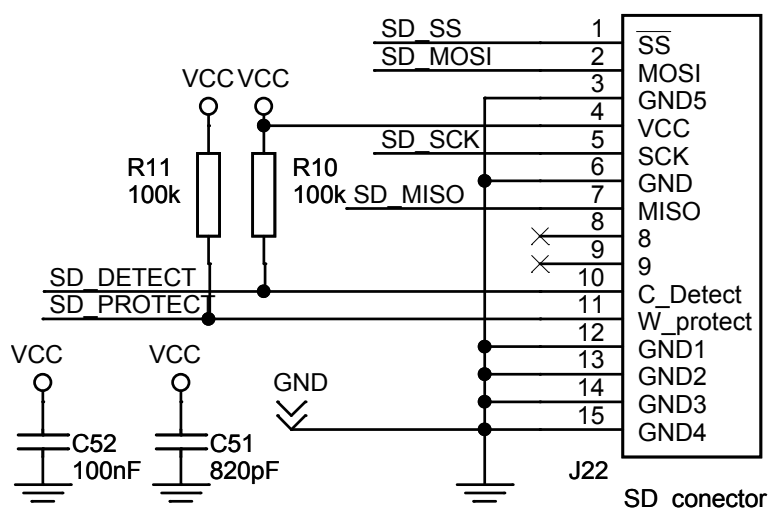
Obrázek 4.7: Zapojení převodníku USART<->RS-485

4.6 SD karta

SD karta je připojena (obr.4.8) k procesoru pomocí SPi sběrnice (SPiC). Pomocí pinu SD_DETECT lze detekovat, zda je karta připojena. Pin SD_PROTECT detekuje, zda je karta uzamčena proti zápisu. K napájecímu pinu jsou připojeny blokovací kondenzátory C52 a C51. Piny mikrokontroléru použité pro připojení SD karty jsou v následující tabulce (tab.4.6).

Signál	Pin_CPU
SD_DETECT	PC0
SD_PROTECT	PC1
SD_SS	PC4
SD_MOSi	PC5
SD_MISO	PC6
SD_SCK	PC7

Tabulka 4.6: Piny využívané pro připojení SD karty k procesoru



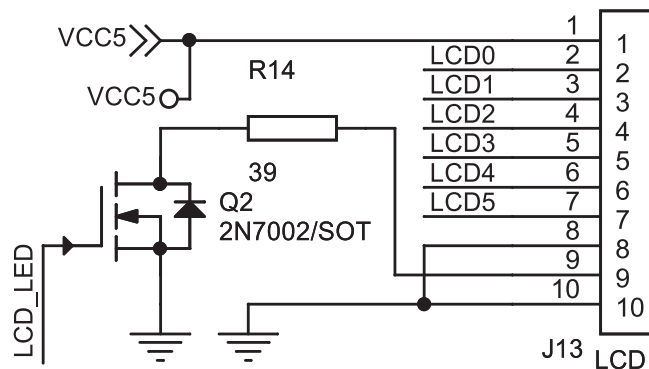
Obrázek 4.8: Zapojení konektoru pro SD kartu

4.7 LCD Displej

Displej se připojuje k desce pomocí 10-pinového konektoru (obr.4.9). Tranzistor Q2 umožňuje vypínat/zapínat podsvícení displeje. Intenzita podsvícení je nastavena rezistorem R14. Zapojení je připraveno pro displeje, které používají řadič HD44780. Tento řadič umožňuje 4 bitovou komunikaci s procesorem. Piny mikrokontroléru použité pro připojení SD karty jsou v následující tabulce (tab.4.7).

Signál	Pin_CPU
LCD_LED	PA5
LCD0	PA6
LCD1	PA7
LCD2	PB0
LCD3	PB1
LCD4	PB2
LCD4	PB3

Tabulka 4.7: Piny využívané pro připojení LCD displeje k procesoru



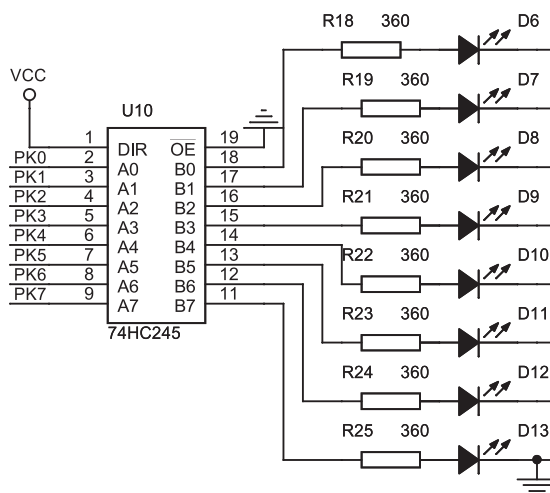
Obrázek 4.9: Zapojení konektoru pro LCD displej

4.8 LED port

Na desce je vyvedeno 8 LED diod, které jsou ovládány pomocí portu K procesoru. LED diody jsou připojeny k procesoru přes budič sběrnice 74HC245 (obr.4.10), aby bylo zamezeno větším proudovým špičkám a zbytečně nebyl zatěžován mikrokontrolér. LED dioda je aktivní v log. 1. Piny mikrokontroléru použité pro připojení budiče sběrnice jsou v následující tabulce (tab.4.8).

Signál	Pin_CPU
LED0	PK0
LED1	PK1
LED2	PK2
LED3	PK3
LED4	PK4
LED5	PK5
LED6	PK6
LED7	PK7

Tabulka 4.8: Piny využívané pro připojení LED diod k procesoru



Obrázek 4.10: Zapojení LED diod

4.9 Rozšiřující konektory

Řídící deska obsahuje dva rozšiřující 40-pinové konektory. Na konektor 1 (obr.4.11) jsou vyvedeny porty ABCD. Na konektor 2 (obr.4.12) jsou vyvedeny porty EFHJ. Každý konektor má vyvedeno jedno napětí 5V, dvakrát 3,3V a čtyřikrát GND. Pin č.19 není zapojen ani u jednoho konektoru.



Obrázek 4.11: Zapojení rozšiřujícího konektoru 1



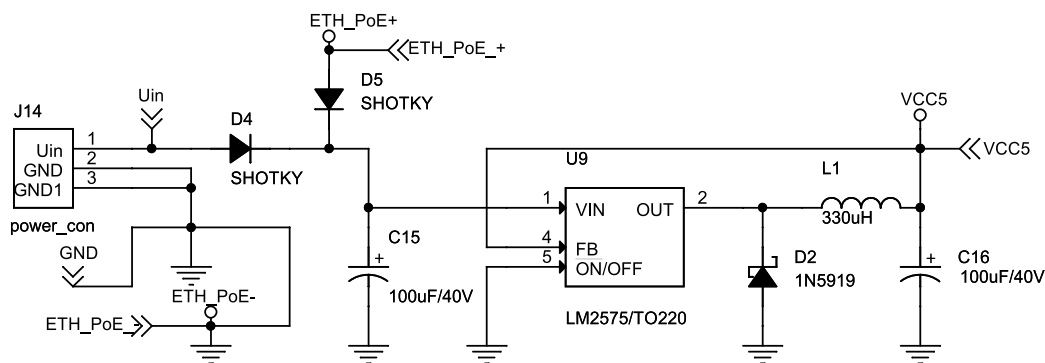
Obrázek 4.12: Zapojení rozšiřujícího konektoru 2

4.10 Napájecí zdroj

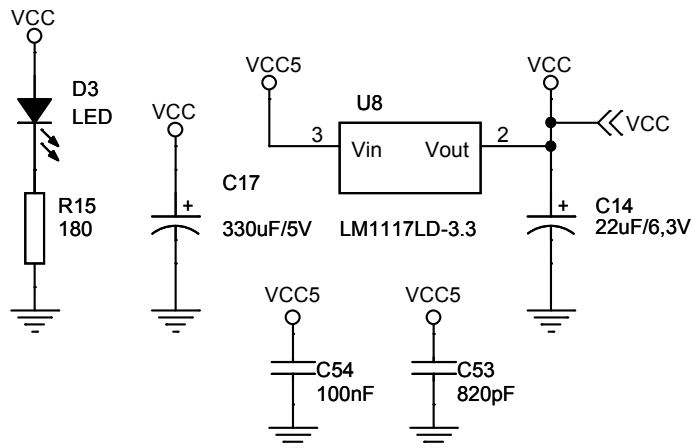
Deska obsahuje dva napájecí zdroje 5V (obr.4.13) a 3,3V (obr.4.14). Zdroj 5V je napájen vstupním napětím a zdroj 3,3V je napájen 5V ze zdroje 5V.

Zdroj napětí 5V je tvořen step-down regulátorem LM2575-5. Rozsah vstupního napětí je v rozmezí 9-40V. Vstupní napětí může být přivedeno napájecím konektorem, nebo pomocí pasivního PoE. Maximální odebíraný proud je 1A. Zdroj napájí LCD displej, převodník USART<->RS-485 a lineární stabilizátor LM1117.

Zdroj napětí 3,3V je tvořen lineárním stabilizátorem LM1117, který dá proud až 800mA. Zdroj napájí všechny zbylé obvody. LED dioda D3 slouží jako signalizace přítomnosti napětí na desce.



Obrázek 4.13: Zapojení napájecího zdroje 5V



Obrázek 4.14: Zapojení napájecího zdroje 3,3V

Kapitola 5

SW Realizace

5.1 Vývojové prostředí

Obslužný program řídicí desky byl napsán ve vývojovém prostředí AVRstudio4, které zdarma poskytuje výrobce mikrokontroléru firma Atmel po jednoduché registraci. Jako překladač jazyka C byl použit GNU GCC překladač WINAVR, se kterým je AVRstudio kompatibilní. WINAVR můžeme stáhnout ze stránek sourceforge.net[23]

Řídicí deska byla programována programátorem JTAGICE MKII, který umožňoval programovat přes JTAG[24] rozhraní. Rozhraní JTAG[24] umožňuje i debugování programu, které přináší velké usnadnění práce při vývoji programu.

5.2 Obsluha sériové linky

Aby mohl mikrokontrolér zpracovávat i ostatní úlohy, je komunikace po sériové lince a obsluha všech ostatních periférií řešena pomocí přerušení. U sériové linky se jedná o přerušení po přijetí znaku a o přerušení po odeslání znaku. Pokud je vyvoláno přerušení po přijetí jednoho znaku, je znak okamžitě uložen do příslušného bufferu. Vyvoláním přerušení po odeslání znaku je naopak zkontrolováno, zda se v bufferu ještě nenacházejí nějaké znaky k odeslání. Pokud buffer není prázdný, tak se odešle další znak a celá operace se opakuje. Využití přerušení a práce s buffery zjednoduší a hlavně zrychlí práci se sériovou linkou, mikrokontrolér mezitím může vykonávat jinou úlohu.

Listing 5.1: Obsluha přerušení od sériové linky.

```
1 ISR (USART_Rx_PC_INT)
2 {
3     LedUsartON;
4     byte = USART_ReadByte(&USARTC0);
5     //zpetny vypis znaku do konzole
6     USART_WriteByte(&USARTC0, byte);
7     //zapis do bufferu
8     bufferWrite_RxPC (byte);
9     LedUsartOFF;
10 }
11
12 ISR (USART_Tx_PC_INT)
13 {
14     LedUsartON;
15     bufferSend_TxPC ();
16     LedUsartOFF;
17 }
```

5.3 Obsluha sběrnice RS-485

Mikroprocesor komunikuje s ovládanými moduly pomocí sběrnice RS-485[21]. Jedná se v podstatě o half-duplexní obsluhu sériového portu, na který je připojen převodník na sběrnici RS-485. Dále je potřeba ovládat směr komunikace. Výchozí stav směru komunikace je příjem. Pro příjem a vysílání jsou využívána přerušování a buffery pro ukládání znaků. Zvolený obsluhovaný modul komunikuje rychlostí 19200 baud pomocí jednoduchého komunikačního protokolu, který je popsán níže.

V knihovně pro obsluhu sběrnice RS-485[21] jsou implementovány funkce pro základní komunikaci (odeslání a příjem znaku) a funkce komunikačního protokolu. V případě využití jiných modulů lze tuto knihovnu jednoduše upravit pro jiný komunikační protokol.

Listing 5.2: Funkce pro vytvoření paketu a jeho postupné nahrání do bufferu.

```

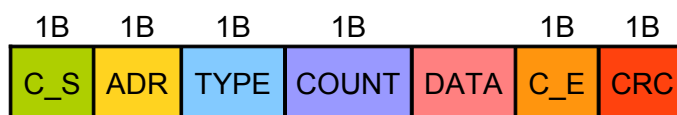
1 //vytvoreni~paketu
2 void createPaket (uint8_t addr, uint8_t len, uint8_t data, uint8_t packet_type)
3 /*
4 *Funkce vytvori~paket a~nahraje ho do bufferu pro odesilani~dat po lince RS-485\cite{RS485}.
5 */
6 {
7     uint8_t CRC = 0x00;
8     //start symbol
9     bufferWrite_Tx (C_S);
10    CRC ^= C_S;
11    //adresa
12    bufferWrite_Tx (addr);
13    CRC ^= addr;
14    //typ paketu
15    bufferWrite_Tx (packet_type);
16    CRC ^= packet_type;
17    //delka dat
18    bufferWrite_Tx (len);
19    CRC ^= len;
20    //data
21    if (len > 0)
22    {
23        bufferWrite_Tx (data);
24        CRC ^= data;
25    }
26    //konec paketu
27    bufferWrite_Tx (C_E);
28    //kontrolni~soucet
29    CRC ^= C_E;
30    bufferWrite_Tx (CRC);
31 }

```

5.3.1 Komunikační protokol ovládaného modulu

Zvolený ovládaný modul komunikuje pomocí jednoduchého komunikačního protokolu, který lze jednoduše rozšířit o další funkce. Data jsou odesílána v paketech a každý modul má svoji unikátní adresu. Řídící deska má vyhrazenou adresu 0x00, dále je pak vyhrazena adresa pro BROADCAST (0xFF).

Paket má svůj pevně daný tvar (obr.5.1). Začíná startovním znakem (C_S) a je zakončen znakem (C_E). Každý paket má na konci také kontrolní součet (CRC), aby bylo možné ověřit správnost přijatých dat. Kontrolní součet se vypočte ze všech odeslaných znaků pomocí funkce xor. Paket dále obsahuje adresu příjemce (ADR), typ o jaký paket se jedná (TYPE), počet bytu dat (COUNT) a odesílaná data (DATA).



Obrázek 5.1: Obecná struktura paketu

Typ	Hodnota	popis
C_S	0xF0	Startovní symbol
C_E	0xF1	Ukončovací symbol
C_TIME	0xF2	Nastavení intervalu měření
C_SET	0xF3	Nastavení adresy
C_PING	0xF4	Příkaz ping
C_TEMP	0xF5	Teplota
C_HUMi	0xF6	Vlhkost
C_PRESS	0xF7	Tlak

Tabulka 5.1: Klíčová slova protokolu

C_TIME - Nastavení intervalu měření Interval měření hodnot lze nastavit v rozsahu 1-60 min.

Odesílatel: TYPE: C_TIME, COUNT: 1, Data: nový interval.

Příjemce (odpověď): TYPE: C_TIME, COUNT: 1, Data: nový interval.

C_SET - Nastavení adresy Adresu na modulu jde nastavit v rozsahu 1-254.

Odesílatel: TYPE: C_SET, COUNT: 1, Data: nová adresa.

Příjemce (odpověď): TYPE: C_PING, COUNT: 1, Data: nastavená adresa.

C_PING - Příkaz ping Příkaz ping slouží k ověření, zda je modul funkční, můžeme ho přirovnat ke klasickému příkazu ping používaném v síťových aplikacích. Dále pak slouží ke zjištění adresy modulu pokud ji neznáme (využití Broadcastové adresy), hlavně pak pro přidání nového modulu do seznamu na řídicí desce.

Odesílatel: TYPE: C_PING, COUNT: 0, Data: žádná

Příjemce (odpověď): TYPE: C_PING, COUNT: 1, Data: adresa příjemce

Funkce dalších typů paketů jsou zřejmé z jejich názvu, proto je popisován pouze tvar odesílání a příjmu.

C_TEMP - Vyčtení teploty Modul odešle teplotu vynásobenou 10x, aby byla hodnota odeslána jako celé číslo. Datový typ **int16**.

Odesílatel: TYPE: C_TEMP, COUNT: 0, Data: žádná

Příjemce (odpověď): TYPE: C_TEMP, COUNT: 2, Data: hodnota teploty

Příklad odpovědi: F0 00 F5 02 4A 01 F1 BD. Paket je určen řídicí desce (adresa 0x00), typ C_TEMP (0xF5), délka dat je 2 (0x02), data jsou 0x4A a 0x01. První byte dat je nejnižší, proto když složíme číslo dohromady dostaneme 0x014A, což je 330 dec. Hodnotu je potřeba vydělit 10. Odesílaná hodnota teploty je 33°C. Obdobný postup je i u dalších typů komunikace.

C_HUMI - Vyčtení vlhkosti Hodnota má rozsah 1-100%. Datový typ **int**.

Odesílatel: TYPE: C_HUMI, COUNT: 0, Data: žádná

Příjemce (odpověď): TYPE: C_HUMI, COUNT: 1, Data: hodnota vlhkosti

C_PRESS - Vyčtení tlaku Modul odešle tlak vynásobený 100x, aby byla hodnota odeslána jako celé číslo. Datový typ **long int**.

Odesílatel: TYPE: C_PRESS, COUNT: 0, Data: žádná

Příjemce (odpověď): TYPE: C_PRESS, COUNT: 3, Data: hodnota tlaku

5.4 Komunikace s RTC obvodem

Pro komunikaci s RTC obvodem byla použita již hotová knihovna, kterou vytvořil Ing. Josef Nouzák[18]. Knihovnu bylo potřeba upravit pro mikrokontrolér ATxmega128. Jednalo se o změnu funkcí potřebných pro komunikaci po TWI sběrnici, pomocí které je RTC obvod připojen. Vytvořil jsem proto novou knihovnu pro komunikaci po TWI sběrnici. Implementovány jsou funkce pouze pro master komunikaci.

Listing 5.3: Funkce pro odeslání a přijetí znaku po TWI sběrnici.

```

1 unsigned char TWI_writeByte (TWI_t * twi, unsigned char data)
2 {
3     while (!(twi->MASTER.STATUS & TWLMaster_WIF_bm));
4     twi->MASTER.DATA = data;
5     //kontrola ACK
6     if (twi->MASTER.STATUS & TWLMaster_RXACK_bm)
7     {
8         return 1;
9     }
10    return 0;
11 }
12
13 unsigned char TWI_readByte (TWI_t * twi)
14 {
15     while (!(twi->MASTER.STATUS & TWLMaster_RIF_bm));
16     return twi->MASTER.DATA;
17 }

```

5.5 Obsluha SD karty

Pro obsluhu SD karty byla použita knihovna, kterou vytvořil Roland Riegel ([17]). Bohužel knihovna nepodporovala mikrokontroléry ATxmega, proto musela být upravena, aby jí bylo možné použít. Knihovna podporuje kompletní práci s SD kartou, jako je práce se souborovým systémem **FAT**, zápis a čtení souboru. Data vyčtená z ovládaného modulu jsou zapisována do souboru **data.csv**.

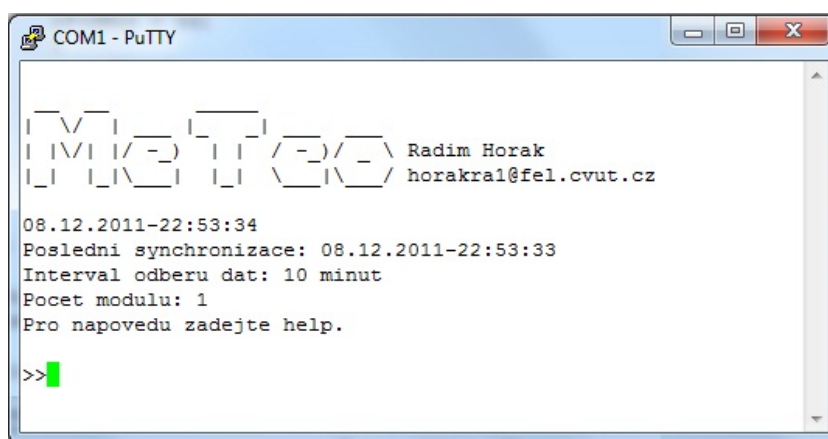
Formát ukládaných dat: datum-čas;jméno;vlhkost;teplota;tlak;

5.6 Ovládací konzole

Pro komunikaci mezi mikroprocesorem a počítačem je implementována jednoduchá ovládací konzole. Pomocí konzole lze nastavit vše potřebné pro obsluhu desky i modulů. Pro konzoli je využit RS-232[22] port řídicí desky, rychlost komunikace je 19200 baud. Příkazy se potvrzují klávesou ENTER. Implementované příkazy jsou v následující tabulce:

Příkaz	popis
add	přidání modulu
edit	editace modulu
del	smazání modulu
list	výpis modulů
help	nápověda
info	informace o stavu řídicí desky
date	nastavení datumu
time	nastavení času
interval	nastavení intervalu
ping	ping

Tabulka 5.2: Příkazy ovládací konzole



Obrázek 5.2: Screenshot ovládací konzole

add - Příkaz pro přidání modulu. Každý modul má své jméno (4 znaky), adresu a interval měření veličin.

edit - Příkaz pro editování záznamu o modulu v seznamu. Změny se projeví i v nastavení samotného modulu.

del - Smazání modulu ze seznamu stanic.

list - Vypis seznamu stanic.

help - příkaz vypíše seznam příkazů a jejich popis.

info - Příkaz info vypíše základní informace o stavu řídicí desky.

date - Příkaz pro nastavení datumu ve formátu dd/mm/rrrr.

time - Příkaz pro nastavení času ve formátu hh:mm:ss.

interval - Příkaz pro nastavení intervalu odběru dat z jednotlivých modulů.

ping - Příkaz, pomocí kterého můžeme zkontrolovat, zda modul komunikuje.

Listing 5.4: Dekódování příkazu pro konzoli

```
1 int dekodujPrikaz~(void)
2 {
3     //dekodovani~prikazu
4     if (!strncmp (cKonzolePrikaz,"add",3)) return 1;
5     if (!strncmp (cKonzolePrikaz,"edit",4)) return 2;
6     if (!strncmp (cKonzolePrikaz,"del",3)) return 3;
7     if (!strncmp (cKonzolePrikaz,"list",4)) return 4;
8     if (!strncmp (cKonzolePrikaz,"help",4)) return 5;
9     if (!strncmp (cKonzolePrikaz,"info",4)) return 6;
10    if (!strncmp (cKonzolePrikaz,"date",4)) return 7;
11    if (!strncmp (cKonzolePrikaz,"time",4)) return 8;
12    if (!strncmp (cKonzolePrikaz,"interval",8)) return 9;
13    if (!strncmp (cKonzolePrikaz,"ping",4)) return 11;
14    if (!strncmp (cKonzolePrikaz,"read",4)) return 12;
15    if (!strncmp (cKonzolePrikaz,"rtc",3)) return 13;
16    if (iKonzolePocetZnaku==0) return 10;
17
18    return 0;
19 }
```

Kapitola 6

Testování a oživení

Testování a oživování bylo prováděno po jednotlivých krocích. Nejprve byly osazeny napájecí zdroje 5V a 3,3V a ověřeno jejich výstupní napětí. V dalším kroku byl osazen mikrokontrolér včetně konektoru pro programování a odzkoušení komunikace mezi AVRstudiem a procesorem. Jako další část byl osazen ethernetový řadič, který je možné jednoduše odzkoušet připojením ethernetového kabelu. Po připojení kabelu začne řadič komunikovat. Komunikace je automaticky signalizována pomocí stavových diod na konektoru RJ45. Dalším osazeným odvodem byl RTC obvod, který byl odzkoušen pomocí převzaté knihovny, ve které musely být upraveny funkce pro komunikaci po TWI sběrnici. Po odzkoušení RTC obvodu byli osazeny obvody pro sběrnici RS-232 a RS-485, které byli odzkoušeny pomocí vlastních knihoven. Nakonec byla deska osazena LED diodami pro LED port, rozšiřujícími konektory a konektorem pro SD kartu. Komunikace s SD kartou byla odzkoušena pomocí převzaté knihovny.

Při testování a oživování nenastaly žádné vážné komplikace, které by nebylo možné rychle vyřešit. Byli zjištěny pouze menší chyby v návrhu desky. Jedna chyba byla v zapojení konektoru pro RS-232[22], kde nebyli překřížené vodiče Rx a Tx. Druhá chyba byla v zapojení konektoru RJ45 pro Ethernet, signalizační diody neměli katodu připojenou na zem, ale omylem spojenou se stíněním. Zapojení diody bylo jednoduše opraveno. Vodiče sběrnice RS-232[22] byli překříženy v konektoru. V příložených souborech jsou všechny nalezené chyby opraveny.

Kapitola 7

Závěr

V rámci práce se podařilo navrhnout a realizovat za poměrně krátkou dobu funkční řídicí desku s poměrně velkým rozsahem možného napájení (9-40V DC). Deska je po Hardwarové stránce připravena komunikovat s měřicími moduly, které měří různé neelektrické veličiny, nejen teplotu, vlhkost a tlak. Pro komunikaci s ovládanými moduly je určena sběrnice RS-485.

Na řídicí desce je vyvedeno mnoho užitečných periférií (RS-232, Ethernet, RS-485, SD karta, RTC, TWI, USART, SPI), které nabízejí využití ve mnoha různých aplikacích. Pomocí dvou rozšiřujících konektorů je možné připojit další přídatné moduly s dalšími perifériemi (např. modul pro bezdrátovou komunikaci). Připraven je i konektor pro připojení LCD displeje, který využívá řadič HD44780. Osazený Ethernetový řadič umožňuje jednoduše a rychle desku připojit k internetu a využít naplno jeho potenciál.

Knihovny funkcí, které byli převzaty a upraveny od jiných autorů, nebo byli nově napsané je možné dále využít při tvorbě jiné aplikace. Dojde tak k usnadnění a zrychlení vývoje.

Literatura

- [1] ZÁHLAVA, V. *OrCAD 10*. Grada Publishing, Praha 2004.
- [2] ZÁHLAVA, V. *Návrh a konstrukce desek plošných spojů*. Vydavatelství ČVUT, Praha 2005.
- [3] VOBECKÝ, J. a ZÁHLAVA, V. *Elektronika - součástky a obvody, principy a příklady*. Třetí rozšířené vydání Grada Publishing, Praha 2005.
- [4] *Latexdocweb - online manuál, 27.12.2011*.
<http://cstug.cz>.
- [5] *Info BP -K336 Info — pokyny pro psaní bakalářských prací, 27.12.2011*.
<http://info336.felk.cvut.cz>.
- [6] *Vývojový kit EvB 4.3, 27.12.2011*.
<http://www.and-tech.pl>.
- [7] *Popis čidla SHT15, 27.12.2011*.
<http://sensirion.com>.
- [8] *Popis čidla MPX4115A, 27.12.2011*.
<http://freescale.com>.
- [9] *Popis mikrokontroléru ATmega128A1, 27.12.2011*.
<http://www.atmel.com>.
- [10] *Popis obvodu RTC-8564, 27.12.2011*.
<http://www.epsontoyocom.co.jp>.
- [11] *Popis Ethernetového řadiče ENC424J600, 27.12.2011*.
<http://ww1.microchip.com>.
- [12] *Popis obvodu MAX3232, 27.12.2011*.
<http://www.maxim-ic.com>.
- [13] *Popis převodníku SN65HVD3082ED, 27.12.2011*.
<http://www.ti.com>.
- [14] *Popis obvodu 74HC245, 27.12.2011*.
<http://www.gme.cz>.

- [15] *Popis obvodu LM2574*, 27.12.2011.
<http://www.ti.com>.
- [16] *Popis obvodu LM1117DT*, 27.12.2011.
<http://www.ti.com>.
- [17] *SD/MMC knihovny pro AVR*, 27.12.2011.
<http://www.roland-riegel.de/sd-reader/>.
- [18] NOUZÁK, J. *Konfigurovatelný řídicí modulární systém*, 27.12.2011.
<http://dip.felk.cvut.cz>.
- [19] *Standart IEEE 802.3af*, 27.12.2011.
<http://www.ieee802.org>.
- [20] *Power over Ethernet*, 27.12.2011.
<http://en.wikipedia.org>.
- [21] *Sběrnice RS-485*, 27.12.2011.
<http://cs.wikipedia.org>.
- [22] *Sběrnice RS-232*, 27.12.2011.
<http://cs.wikipedia.org>.
- [23] *GNU GCC překladač WINAVR*, 27.12.2011.
<http://www.sourceforge.net/>.
- [24] *JTAG - popis*, 27.12.2011.
<http://cs.wikipedia.org>.
- [25] *TME, obchod s elektronickými součástkami*, 27.12.2011.
<http://www.tme.cz>.
- [26] *Pragoboard a.s., výrobce plošných spojů*, 27.12.2011.
<http://www.pragoboard.cz>.
- [27] *SN65HVD3082E obrázek*, 27.12.2011.
<http://www.ti.com>.
- [28] *EvB 4.3 obrázek*, 27.12.2011.
<http://shop.onpa.cz>.
- [29] *ATmega128A1 obrázek*, 27.12.2011.
<http://www.ulink-ele.com>.
- [30] *Epson RTC-8564 obrázek*, 27.12.2011.
<http://media.digikey.com>.
- [31] *Microchip ENC424J600 obrázek*, 27.12.2011.
<http://www.pandatron.cz>.

- [32] *MAX3232 obrázek, 27.12.2011.*
<http://www.electropark.pl>.
- [33] *JTAG pinout obrázek, 27.12.2011.*
<http://www.atmel.com>.

Kapitola 8

Seznam použitých zkratek

SD Secure Digital

USART Universal Synchronous / Asynchronous Receiver and Transmitter

SNMP Simple Network Management Protocol

PCB Printed Circuit Board

SPI Serial Peripheral Interface

ISP In System Programming

GCC GNU Compiler Collection

JTAG Joint Test Action Group

TWI Inter-IC-bus

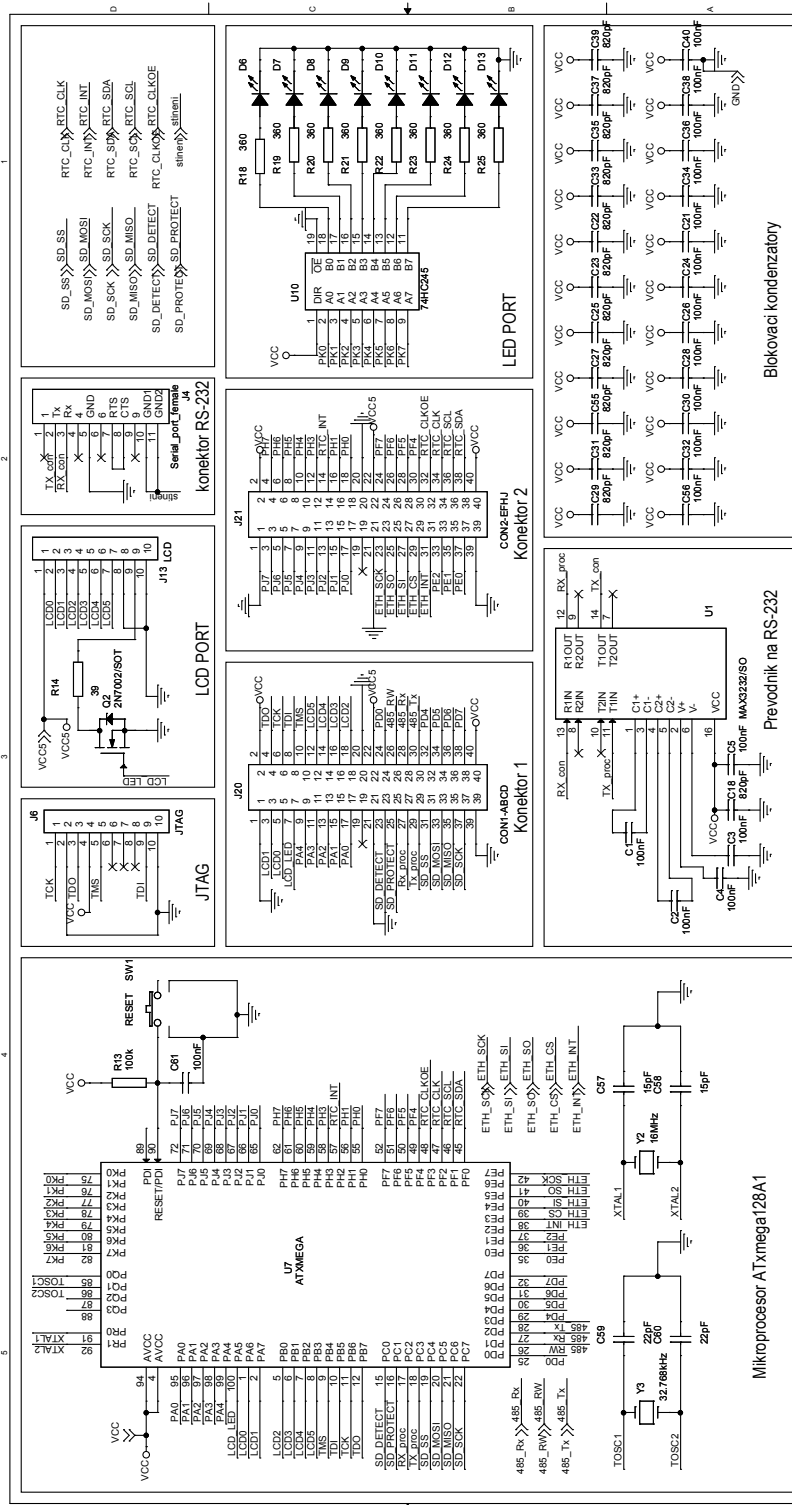
RTC Real-Time Clock

FAT File Allocation Table

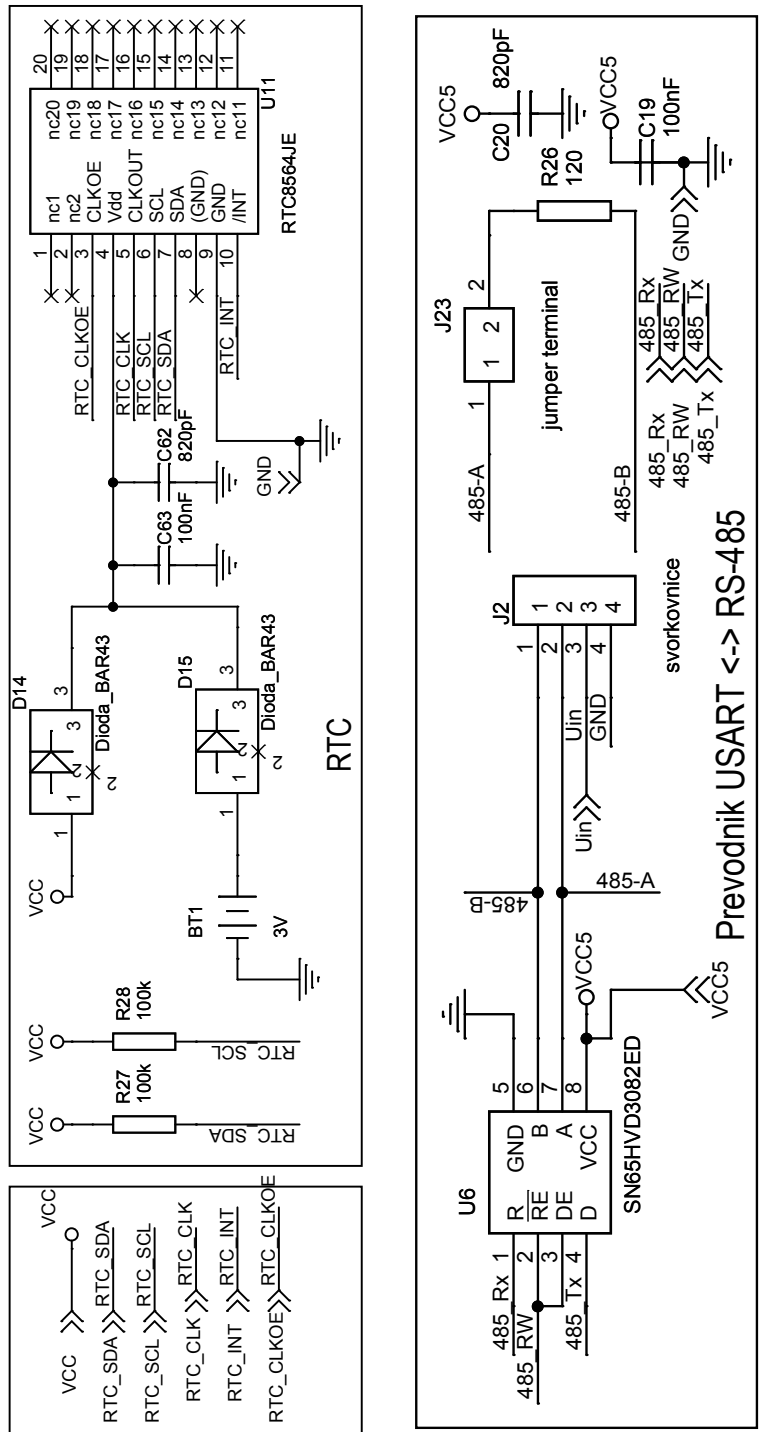
PoE Power over Ethernet

Příloha A

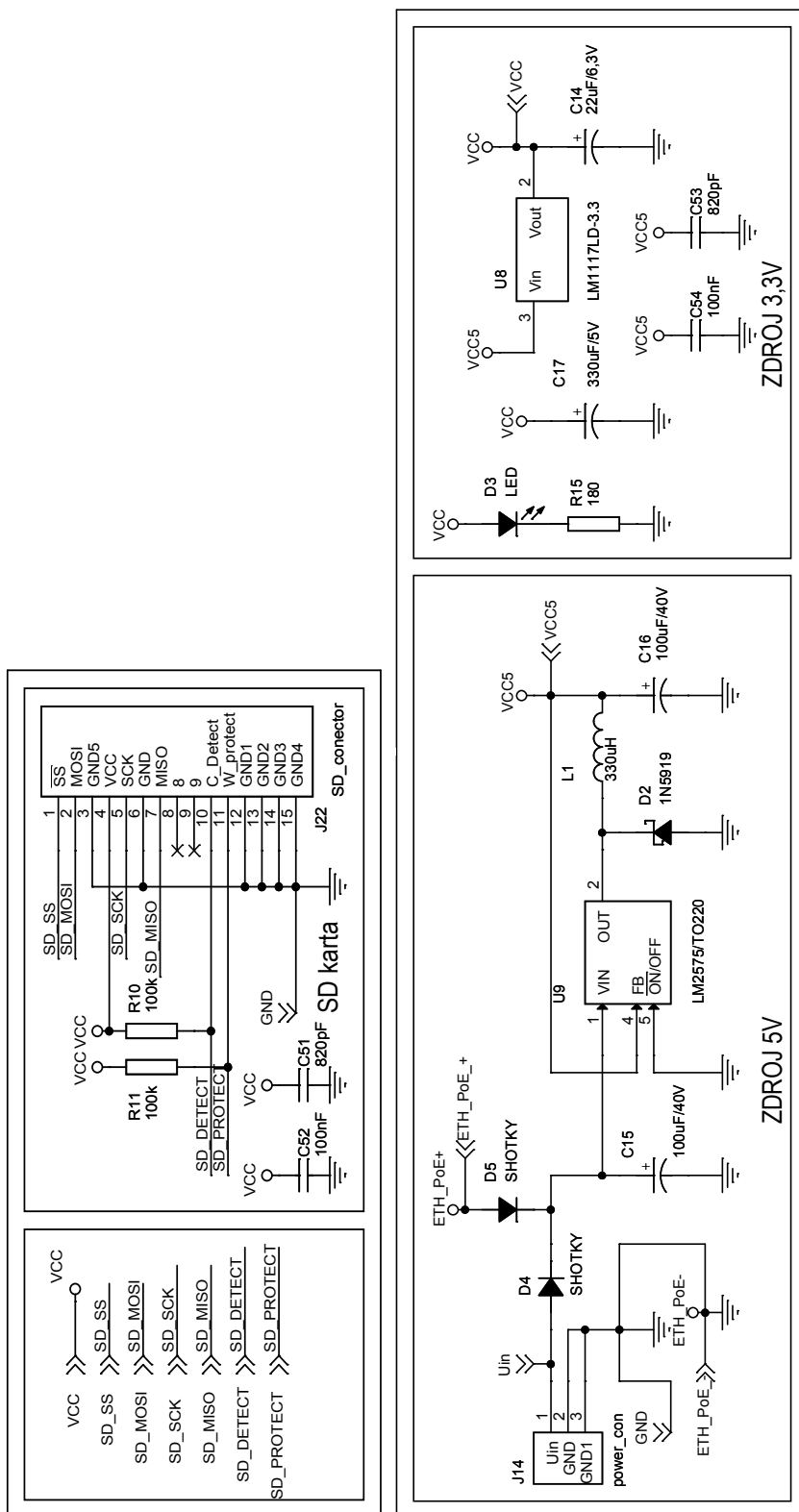
Schéma desky



Obrázek A.1: Schéma zapojení mikrokontroléru, MAX3232, LED portu, LCD portu a konektorů



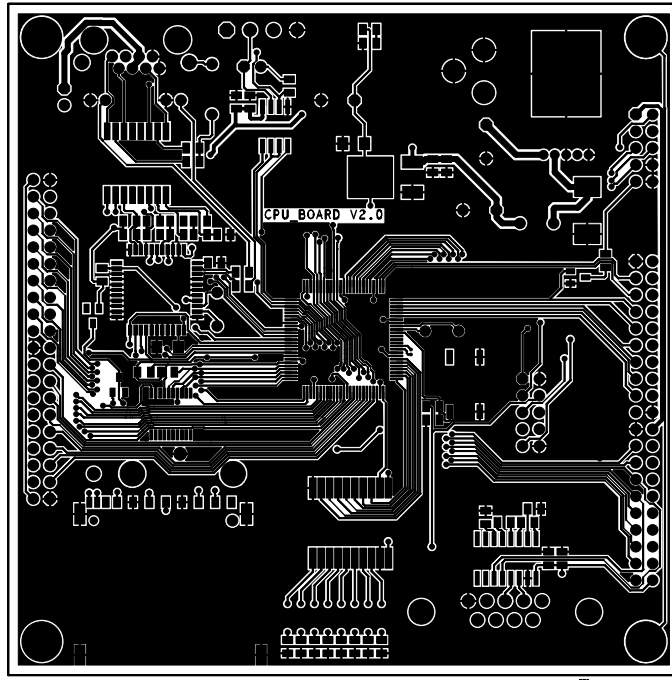
Obrázek A.3: Schéma zapojení převodníku pro sběrnici RS-485 a obvodu RTC



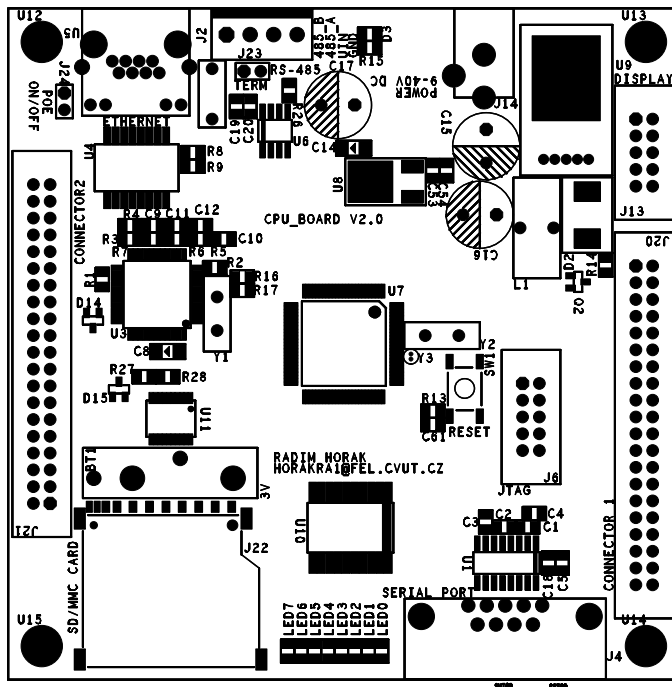
Obrázek A.4: Schéma zapojení SD karty a napájecích zdrojů

Příloha B

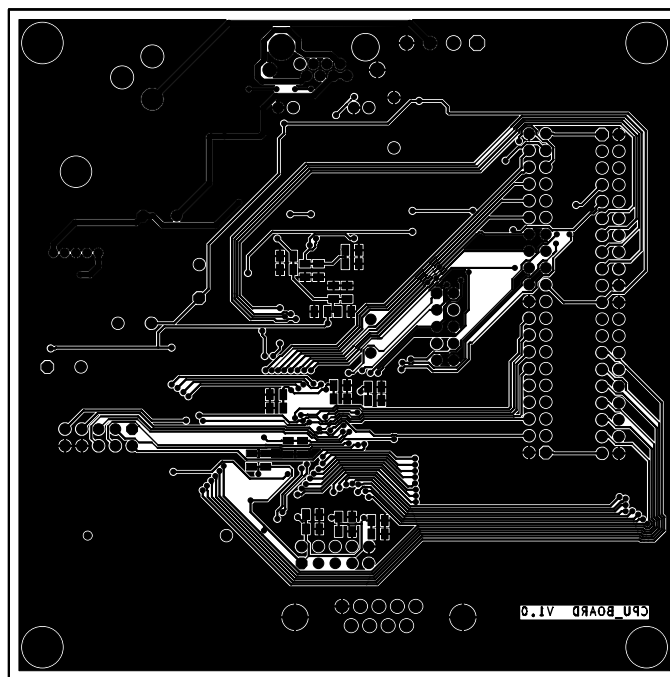
DPS a rozmístění součástí



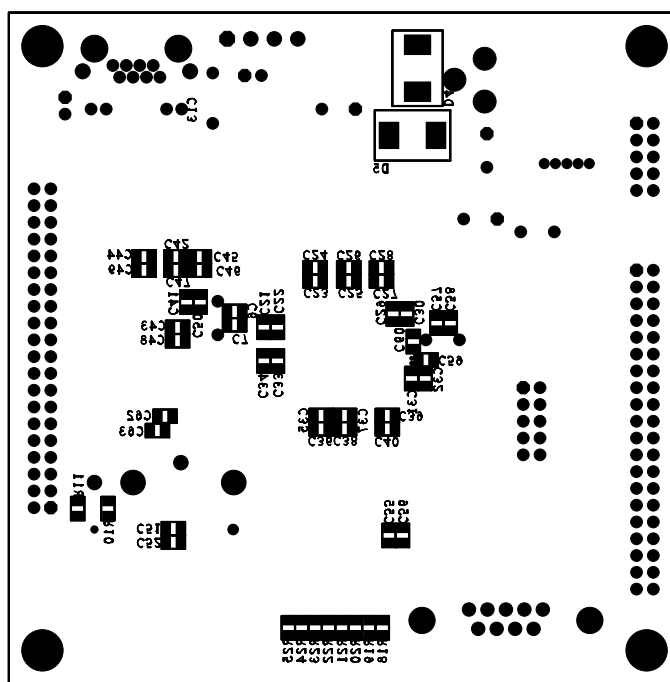
Obrázek B.1: Vrstva TOP



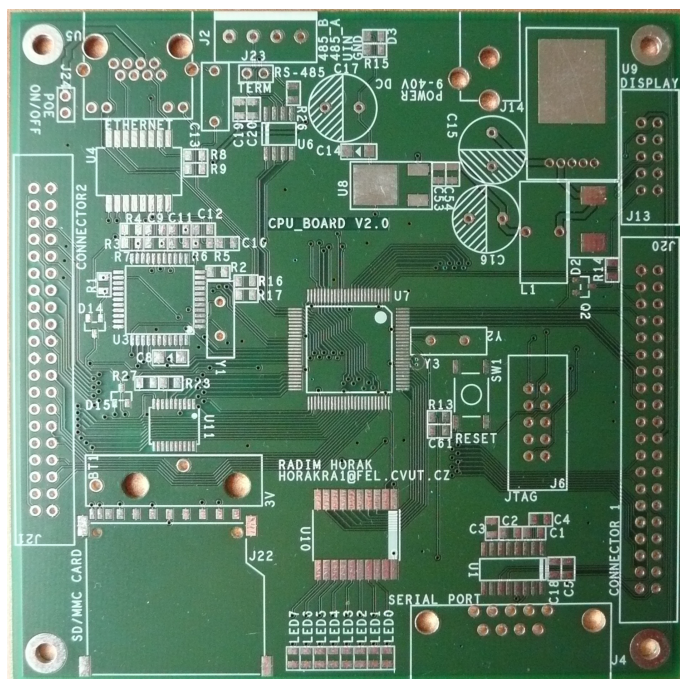
Obrázek B.2: Rozmístění součástek TOP



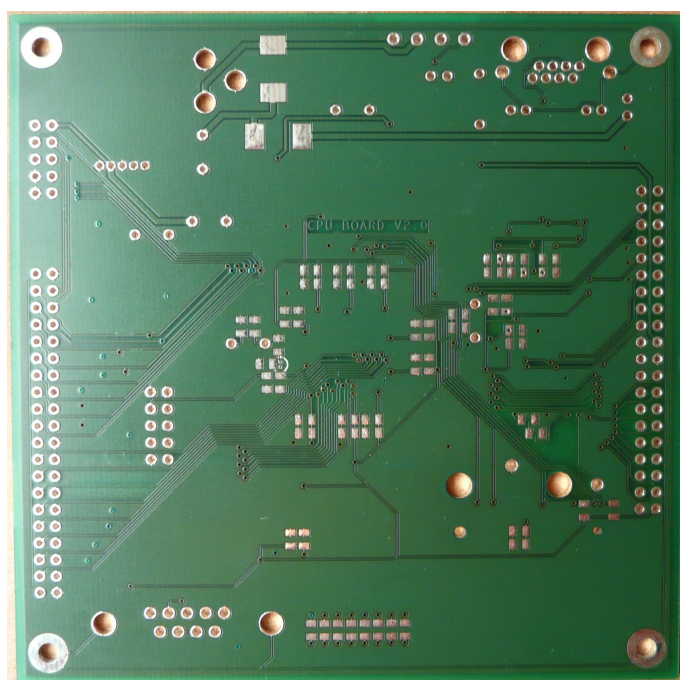
Obrázek B.3: Vrstva BOT



Obrázek B.4: Rozmístění součástek BOT



Obrázek B.5: Neosazená deska vrstva TOP



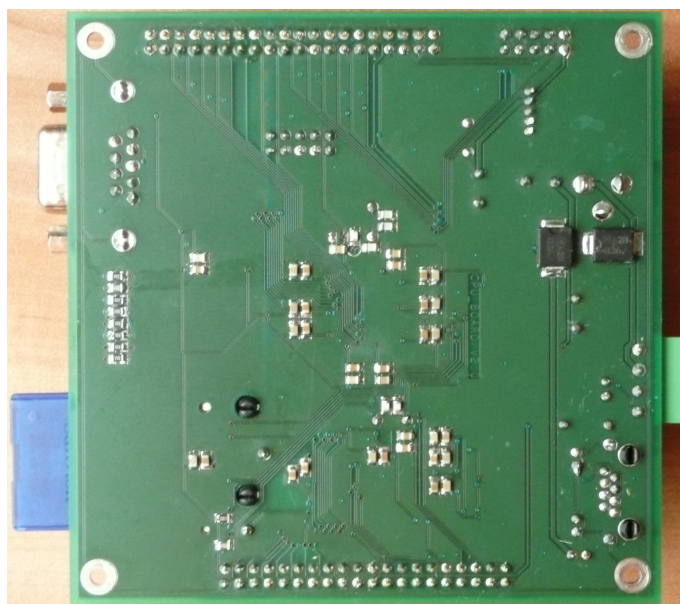
Obrázek B.6: Neosazená deska vrstva BOT

Příloha C

Osazená deska



Obrázek C.1: Osazená deska vrstva TOP



Obrázek C.2: Osazená deska vrstva BOT

Příloha D

Knihovny funkcí

V této příloze jsou popsány jednotlivé knihovny funkcí, které byly vytvořeny pro potřeby obsluhy periférií. Knihovny byly vytvořeny pro obsluhu sběrnice TWI, RS-485[21] a RS-232[22]. Knihovna pro obsluhu sběrnice RS-232[22] nebude popisována, protože knihovna pro ovládání sběrnice RS-485[21] obsahuje totožné funkce pro příjem a odeslání znaku.

D.1 485.c

D.1.1 Potřebné definice

#define USART_485 - Definice USART portu, který bude využíván pro komunikaci po sběrnici RS-485.

#define BaudRate_485 - Definování rychlosti komunikace po sběrnici RS-485.

#define SET_RIDICI_PIN - Definování řídicího pinu, konkrétně nastavení, že bude použit jako výstupní.

#define VYSILANI - Definice pinu pro vysílání, nastavení log. 1 na jeho výstupu.

#define PRIJEM - Definice pinu pro příjem, nastavení log. 0 na jeho výstupu.

#define Led485Up - Definice LED diody, která je využita pro signalizaci provozu na sběrnici RS-485, konkrétně log. 1.

#define Led485Down - Definice LED diody, která je využita pro signalizaci provozu na sběrnici RS-485, konkrétně log. 0.

#define BUFFER_SIZE_Rx - Nastavení velikosti bufferu pro příjem dat.

#define BUFFER_SIZE_Tx - Nastavení velikosti bufferu pro odeslání dat.

D.1.2 Funkce

void init485 (void) - Funkce inicializuje sériovou linku využitou pro komunikaci po sběrnici RS-485[21], nastaví řídicí pin jako výstupní.

uint8_t USART_485_Init (USART_t * usart, uint32_t ulBaudRate, uint32_t ulF_CPU) - Funkce inicializuje sériovou linku specifikovanou v parametru funkce (usart).

uint8_t USART_485_SetBaudRate (USART_t * usart, uint32_t ulBaudRate, uint32_t ulF_CPU) - Funkce nastaví baudrate pro komunikaci po sériové lince usart.

void USART_485_WriteByte (USART_t * usart, uint8_t bData) - Funkce odešle jeden znak bData po sériové lince usart.

void USART_485_WriteString (USART_t * usart, uint8_t * sData) - Funkce odešle řetězec znaků po sériové lince usart.

uint8_t USART_485_ReadByte (USART_t * usart) - Funkce přečte jeden znak přijatý po sériové lince usart a navrátí ho jako svoji návratovou hodnotu typu uint8_t.

void createPaket (uint8_t addr, uint8_t len, uint8_t data, uint8_t packet_type) - Funkce vytvoří paket ze vstupních parametrů a uloží jej do bufferu.

unsigned char receivePaket (paket * prijatyPaket) - Funkce přečte data z bufferu a vytvoří z nich paket. Pokud má vytvořený paket správný tvar vrátí 1.

void zpracujPaket (paket * prijatyPaket, char * data) - Funkce dekóduje přijatý Paket a zpracuje přijaté data, které jsou uloženy do data.

void bufferWrite_Rx (uint8_t c) - Funkce zapíše znak do bufferu pro příjem.

uint8_t bufferRead_Rx (void) - Funkce vrátí jeden znak z bufferu pro příjem dat.

void clearBuffer_Rx (void) - Funkce smaže buffer pro příjem dat.

void bufferWriteTx_Tx (uint8_t c) - Funkce zapíše znak do bufferu pro odesílání.

uint8_t bufferReadTx_Tx (void) - Funkce vrátí jeden znak z bufferu pro odeslání dat.

void bufferSend_Tx (void) - Funkce odešle jeden znak z bufferu pro odesílání dat.

char ping485 (uint8_t adr) - Funkce odešle paket typu C_PING na adresu adr. Dotazovaný modul s adresou adr odpoví stejným způsobem, ale přidá ještě jeden datový byte obsahující jeho adresu. Funkce vrátí 0x00 v případě, že vše proběhlo v pořádku, v případě chyby vrátí 0xFF. Pokud bude v parametru adr předána broadcastová adresa, bude návratová hodnota funkce rovna adrese zařízení, které odpovědělo.

char temp485 (uint8_t adr, char *cTeplota) - Funkce odešle paket typu C_TEMP na adresu adr. Vyčtená teplota bude uložena do cTeplota. Pokud vše proběhne v pořádku vrátí 0.

char humi485 (uint8_t adr, char *cHumi) - Funkce odešle paket typu C_HUMi na adresu adr. Vyčtená vlhkost bude uložena do cHumi. Pokud vše proběhlo v pořádku vrátí 0.

char press485 (uint8_t adr, char *cPress) - Funkce odešle paket typu C_PRESS na adresu adr. Vyčtený tlak bude uložen do cPress. Pokud vše proběhlo v pořádku vrátí 0.

char adresa485 (uint8_t adr, uint8_t iNewAdr) - Funkce nastaví novou adresu iNewAdr na modulu s adresou adr. Vrací 0 pokud vše proběhlo v pořádku.

char interval485 (uint8_t adr, uint8_t iInt) - Funkce pro nastavení intervalu odběru dat (iInt) na modulu s adresou adr. Vrací 0 pokud vše proběhlo v pořádku.

D.2 TWI.c

Knihovna podporuje komunikaci pouze jako Master.

D.2.1 Potřebné definice

```
#define TWI_BAUD (F_SYS, F_TWI) ( (F_SYS / (2 * F_TWI)) - 5) - Nastavení rychlosti na sběrnici twi.
```

D.2.2 Funkce

void TWI_init (TWI_t * twi, uint32_t TWI_BAUDRATE, uint32_t CPU_SPEED)
- Funkce inicializuje sběrnici TWI včetně rychlosti komunikace podle zadaných parametrů (TWI_BAUDRATE, CPU_SPEED).

unsigned char TWI_start (TWI_t * twi, uint8_t address) - Funkce odešle startovní byte po TWI sběrnici na adresu address. Tuto funkci lze využít i pro případný restart komunikace (často změna směru vysílání) Vrací 0, pokud je všechno v pořádku.

unsigned char TWI_writeByte (TWI_t * twi, unsigned char data) - Funkce odešle datat po sběrnici TWI. Vrací 0, pokud je vše v pořádku.

unsigned char TWI_readByte (TWI_t * twi) - Funkce přijme jeden byte po sběrnici twi. Návrátová hodnota této funkce je přijatý byte.

void TWI_stop (TWI_t * twi) - Funkce odešle stop byte a ukončí tím komunikaci po sběrnici TWI.

void TWI_sendACK (TWI_t * twi) - Funkce nastaví ACK (0), které Odesílateli udává, že byl byte úspěšně přijat.

void TWI_sendNACK (TWI_t * twi) - Funkce nastaví NACK (1) a dá tím najevo odesílateli, že došlo k chybě.

Příloha E

Obsah přiloženého CD

Na přiloženém CD jsou přiloženy zdrojové kódy, katalogové listy použitých obvodů a projekty z programu OrCad 16.3.

- Adresář Firmware
Adresář obsahuje kompletní zdrojové kódy.
- Adresář Datasheet
V adresáři jsou uloženy katalogové listy použitých obvodů.
- Adresář PCB
Adresář obsahuje dvě složky (Schema a PCB). Ve složce Schema je kompletní projekt se schématem z programu OrCAD-Capture 16.3. Složka PCB obsahuje kompletní projekt návrhu PCB včetně výrobních podkladů z programu OrCAD-PCBEditor 16.3.