



Středoškolská technika 2014

Setkání a prezentace prací středoškolských studentů na ČVUT

Inteligentní domácnost

Michal Fuxa

Střední průmyslová škola elektrotechnická,
Praha 10, V Úžlabině 320

Anotace

Práce se zabývá návrhem a realizací systému pro řízení a kontrolu provozu rodinného domu (inteligentních budov). Návrh systému je proveden na základě analýzy dostupných systémů, mikrokontrolérů, komunikačních protokolů a sběrnic. Výstupem práce jsou prototypy hardware postavené na mikrokontrolérech Atmel AVR včetně veškeré dokumentace a také aplikační model rodinného domu. Implementovaným protokolem je protokol Spinel.

Obsah

Anotace.....	2
1. Teoretický rozbor	6
1.1. Inteligentní dům.....	6
1.1.1. Standard KNX.....	6
1.1.2. ABB i-bus® KNX.....	6
1.1.3. ABB Ego-n®.....	7
1.1.4. Loxone.....	7
1.2. Mikrokontroléry.....	8
1.2.1. Microchip PIC	8
1.2.2. Atmel AVR	9
1.3. Sériová komunikace (sběrnice)	9
1.3.1. Paralelní sběrnice	9
1.3.2. RS-232.....	10
1.3.3. RS-485/422.....	11
1.4. Komunikační protokol.....	11
1.4.1. KNX	12
1.4.2. Modbus.....	12
1.4.3. Profibus	12
1.4.4. Spinel.....	12
2. Idea vlastního řešení.....	12
2.1.1. Požadavky na systém, výběr technologií	13
2.1.2. Systém složený z modulů.....	13
2.1.3. Centralizovaný systém	14
2.1.4. Komunikace po lince RS-485	14
2.1.5. Výběr výpočetní platformy	15
3. Návrh a realizace hardwaru.....	15
3.1. Hardwarový základ pro moduly	16
3.1.1. Mikrokontrolér ATmega8	16
3.2. Vstupně výstupní moduly.....	17
3.2.1. Vstupy	18
3.2.2. Výstupy	19
3.3. Modul termostatu.....	20
3.4. Stavový displej	20
3.5. Modul teplotních čidel.....	21
3.6. Modul reálného času.....	21
3.7. Řídící jednotka.....	21
3.7.1. Mikrokontrolér ATmega162	22
3.7.2. MAX232.....	22
3.7.3. Konektorová výbava	22
3.8. Zkušební model budovy	23
4. Programová část	23
4.1. O protokolu Spinel.....	23
4.1.1. Základní vlastnosti protokolu.....	24
4.1.2. Formáty definované firmou Papouch.....	24
4.1.3. Datový paket formátu 97.....	24
4.2. Vývojové prostředí Bascom AVR.....	25

4.3. Implementace protokolu Spinel.....	25
4.3.1. Regulace vytápění	26
4.3.2. Ovládání světel.....	26
Seznam použité literatury a zdrojů.....	28
Použitý software.....	29
Seznam příloh.....	30
Příloha 1: Význam a zapojení pinů RS-232	31
Příloha 2: Schémata zapojení	32
Příloha 3: Plošné spoje, osazovací plán	36
Příloha 4: Zpracování paketu (Spinel)	40
Příloha 5: Ukázka zdrojového kódu	41
Příloha 6: Fotografie.....	43

Úvod

Již od samotného počátku se snaží člověk pro sebe najít to největší pohodlí a nebojí se investovat nemalé prostředky a úsilí. V posledních letech se čím dál tím více rozšiřují tzv. Inteligentní budovy a rodinné domy. Tento fakt mě přesvědčil se o tuto problematiku začít více zajímat a to hlavně z hlediska řídicí elektroniky a inteligentních elektroinstalací. Překvapila mě především cena těchto systémů, díky které je jejich použití především otázkou velkého luxusu. Hlavním cílem mé práce je vyvinout vlastní systém pro řízení inteligentního domu, navrhnout a vyrobit první prototypy vlastního hardware, naprogramovat prvotní verze firmware s implementovaným komunikačním protokolem a v neposlední řadě také vytvoření funkčního modelu inteligentního domu, který bude sloužit k ověření funkčnosti a propojení jednotlivých částí.

1. Teoretický rozbor

V této kapitole se budu zabývat obecnou rovinou problematiky inteligentních domů a elektroinstalací, analýzou některých dostupných systémů. Dále rozbor důležitých prvků celé práce, jako jsou například mikrokontrolér, sběrnice nebo komunikační protokol.

1.1. Inteligentní dům

Dům, který nabízí svému obyvateli optimální životní podmínky a pohodlí. Tento pojem zahrnuje celou řadu oblastí od samotné architektury domu, využívání efektivních materiálů, optimální ekonomické náročnosti, šetrnosti k životnímu prostředí až po inteligentní elektronické prvky (někdy také inteligentní elektroinstalace). Součástí inteligentních budov bývá i systém zabezpečené nebo chytré multimediální systémy. Tato práce se zabývá výhradně otázkou řídicí elektroniky. Příklady dostupných systémů jsou uvedeny v následujících podkapitolách.

1.1.1. Standard KNX

Standard vytvořený asociací KNX a je nástupcem systémů EIB, Batibus a EHS. V roce 2003 byl uznán jako zatím jediný celosvětový standard pro elektrotechnické systémy do domů a budov. Standard KNX definuje kompletně celou komunikaci řídicích systémů budov od vlastní svěrnice, přes mechanické způsoby propojení, definuje otevřený komunikační protokol a zabíhá do nejmenších detailů napříč celým modelem ISO/OSI.

1.1.2. ABB i-bus® KNX

I-bus® od firmy ABB je systém inteligentní elektroinstalace založený na komunikačním standardu KNX. Je určen pro nejluxusnější budovy a výstavbové projekty. Má velmi široký rozsah schopností jako např. běžné ovládání a stmívání osvětlení, vytápění, zabezpečení domu například pomocí dojmu přítomnosti obyvatel, spolupráce s dalšími zabezpečovacími systémy, protipožární signalizace, bezdrátové ovládání a vypínače. Kalkulace na tento systém pro dvoupatrový luxusní dům se pohybuje ve stovkách tisíc korun.

1.1.3. ABB Ego-n®

Na rozdíl od ABB i-bus® je Ego-n® určen spíše pro menší projekty a rodinné domky. Ego-n tvoří elektronické srdce domu (jeho centrální počítač), je pouze dobře promyšlenou inteligentní elektroinstalací, která přináší svému uživateli úsporu energie a nabízí další inovace. Výrobce mluví o své elektroinstalaci jako o „neviditelné“ pracující na pozadí běžného života. Ego-n® nabízí ovládání světel, řízení vytápění a hlídání teploty ovládání žaluzií apod., nedílnou součástí jsou i bezdrátové vypínače a ovládání z mobilní či počítačové aplikace.

1.1.4. Loxone

Firma Loxone nabízí komplexní řešení chytrého domu. Srdcem systému je Miniserver, malý počítač s vlastním operačním systémem (Loxone OS). Miniserver je základní stavební jednotkou bez nutnosti dalšího rozšiřování (ale tato možnost tu je). Nabízí základní počet analogových i digitálních vstupů a výstupů, webový server a konektivitu kompatibilní se systémy KNX a EIB, sériovou linku a LAN. To vše v kompaktních rozměrech s možností montáže na DIN lištu. Systém nabízí i řadu rozšíření v podobě dalších vstupů a výstupů (Loxone Extensions), modul s výkonovými relé, převodníky úrovní pro různé sběrnice, stmívače světel apod. Firma má v nabídce také řadu doplňků a komponent a také předplacené služby. Součástí systému jsou i aplikace pro různé platformy zdarma. Cena systému je přibližně 12tis. korun za Miniserver, ceny modulů se pohybují v řádu tisíců korun.

1.2. Mikrokontroléry

Mikrokontroléry (jinak také jednočipové počítače) představují v dnešní době důležitou součást řídicí a automatizační techniky. Mikrokontrolér je integrovaný obvod velmi vysoké integrace, který zahrnuje kompletní počítač, nejčastěji harvardské koncepce, umožňující použití v nesčetném množství aplikací. Dokáže nahradit velké množství logických a kombinačních obvodů. Kromě toho obsahují také řadu doplňujících obvodů. Mezi běžně integrované obvody na čip mikrokontroléru patří A/D i D/A převodníky, analogové komparátory, čítače/časovače, čítače s funkcí pulzně šířkové modulace (PWM), ze sběrnic je běžnou součástí asynchronní sériová sběrnice UART a synchronní sériová sběrnice SPI (někdy souhrnně označováno jako USART – Universal Synchronous / Asynchronous Receiver and Transmitter), multi-másterová sběrnice I²C firmy Philips, jednovodičová nízkorychlostní sběrnice 1-Wire od firmy DALLAS (např. číslicová teplotní čidla) atd. Použití mikrokontrolerů je výhodné hlavně z hlediska úspory součástek (nahradíme tak desítky různých typů integrovaných obvodů) a tím i úspory velikosti navražené elektroniky. Zjednoduší se i návrh zařízení. Snadná změna programu v již hotovém obvodu, to například s logickými obvody nelze. Mezi nejznámější společnosti vyrábějící mikrokontroléry patří firmy Atmel Corporation, Microchip Technology, Texas Instruments nebo Freescale Semiconductor. Na českém trhu jsou ve velké míře rozšířeny 8bitové mikrokontroléry firem Atmel a Microchip.

1.2.1. Microchip PIC

Mikrokontroléry Harvardské architektury s redukovanou instrukční sadou vyráběné firmou Microchip. Vyrábí se 8bitové (PIC10, 12, 14, 16, 17, 18), 16bitové (PIC24, dsPIC30, 33) a 32bitové (PIC32). Rozděleny jsou podle výkonnosti a počtu instrukcí. Nejnižší řady (PIC10-16) mají pouze 35 instrukcí a používají se na nejjednodušší aplikace psané obvykle v assembleru. Vyšší řady mají 75 instrukcí a více, mají vylepšenou podporu vyšších programovacích jazyků. Procesory PIC jsou jedny z nejstarších mikrokontrolerů na trhu a používají se napříč celým spektrem řídicí techniky od domácích spotřebičů po měřicí přístroje.

1.2.2. Atmel AVR

Řada AVR jsou 8bitové a 32bitové RISC mikrokontroléry Harvardské koncepce. Jsou nástupcem populární řady AT89, která vycházela z architektury 8051. Dělí se na čtyři základní skupiny. Asi nejznámější jsou ATtiny a ATmega. ATtiny jsou nejmenší mikrokontroléry z této řady, jsou 8bitové a vyrábějí se v provedení 6 až 32 pinů, hodí se pro nenáročnou aplikaci s minimem kódu. ATmega se vyrábí v pouzdrech od 28 do 100 pinů. Slouží pro středně náročnou aplikaci. Výkonově odpovídají ATtiny, ale mají větší SRAM i programovou paměť a více vstupně výstupních pinů. Nejvýkonnějšími zástupci jsou 8bitové ATxmega a 32bitové AT32. Mikrokontroléry AVR vynikají velkou vybaveností, nízkou cenou v poměru s výkonem a univerzálností použití.

1.3. Sériová komunikace (sběrnice)

Sériové sběrnice slouží ke komunikaci zařízení mezi zařízeními po dvojici (jeden signálový), případně i větším počtu vodičů. Jedná se o sekvenční komunikaci, kdy posíláme postupně každý bit odesílaného bajtu jeden za druhým. K řízení samotné sběrnice mohou být vedle standardních datových vodičů vedeny ještě další vodiče pro řídicí signály.

1.3.1. Paralelní sběrnice

Vedle sběrnice sériové stojí sběrnice paralelní, jedná se o skupinu signálových vodičů, kde jsou data posílány paralelně, tedy celý bajt (případně dvojici bajtů, čtveřici apod.) najednou. Dělí se na tři základní typy: Řídicí, adresová a datová sběrnice. Tento typ sběrnice však není pro tuto práci stěžejní.

1.3.2. RS-232

Sériová sběrnice (zvaná také sériový port, sériová linka) určená standardně pro komunikaci mezi dvěma zařízeními. Její zatím poslední specifikace (RS-232C) byla představena již v roce 1969. Z hlediska referenčního modelu ISO/OSI se jedná pouze o jeho fyzickou vrstvu. Standard definuje pouze způsob přenesení sekvence bitů, další zpracování dat již není součástí tohoto standardu. Pracovní vzdálenost je přibližně do 20m. Napěťové úrovně sériové linky se pohybují od -3V do -15V pro log. 1 a od 3V do 15V pro log. 0 pro datové signály (vodiče RxD a TxD) u řídicích signálů se rozhodovací úrovně pro log. 1 a 0 otáčí (vodiče označené jako RTS, CTS, DSR, DCD, DTR a RI). Hodnoty napětí jsou proti zemi značené jako GND. Při použití většího rozsahu napětí lze dosáhnout ještě vyšších vzdáleností. Je zde použit bipolární linkový kód NRZ (Non Return to Zero). Mezi -3V a 3V je tzv. zakázaná oblast, kde se nenachází žádná logická úroveň. Standardním konektorem pro sběrnici RS-232 je konektor D-Sub a to konkrétně v provedení DB-9 M i F a DB-25 M i F (starší provedení používané hlavně pro modemy, obsahuje stejný počet vodičů). Rozpis a funkce jednotlivých vodičů viz Příloha 1.

1.3.3. RS-485/422

Standard sériové sběrnice mající podobný základ jako sběrnice RS-232. RS-485 je dvou vodičová (značené jako A a B nebo + a -) poloduplexní (střídavě obousměrná) sběrnice, určená pro datovou a řídicí komunikaci až pro 32 zařízení (uzlů) na vzdálenost až 1,2km. Vedení by mělo být na obou stranách zakončeno zakončovacími odpory tzv. terminátory, které mají hodnotu 120Ω . Terminátory zabraňují odrazům na vedení a pomáhají omezit další rušivé elementy vznikající především na dlouhém vedení. Logické úrovně jsou zde reprezentovány rozdílovým napětím mezi oběma datovými vodiči. Pro rozdíl napětí na A a B větší než 200mV je detekována log. 0, při rozdílu A a B menším než -200mV je detekována log. 1. Naproti tomu RS-422 představuje plně-duplexní komunikaci (komunikace probíhá po dvou párech vodičů), vysílat i přijímat můžeme v jeden moment. Na rozdíl od RS-485 však RS-422 neumožňuje konstrukci vícebodové sítě. Na jeden vysílač však lze připojit až 10 přijímačů. Maximální délka linky je 1,5km a maximální datová propustnost do 12m je 10Mb/s. Maximální hodnoty rozdílového napětí jsou 6V/-6V. Oba standardy definují přenos pouze z elektrického hlediska (pravidla pro napěťové úrovně), Použité kabely nebo jejich koncovky musí definovat jiné standardy nebo systémy, které tyto sběrnice používají. Příkladem je například specifikace Profibus.

1.4. Komunikační protokol

Komunikační protokol je soubor pravidel nebo standart pro komunikaci mezi zařízeními, popřípadě pro přenos dat na příklad právě na sběrnici. Mezi náležitosti komunikačního protokolu patří většinou znaky pro zahájení nebo ukončení komunikace, celková podoba odesílané zprávy (její formát). Dále pak pravidla pro detekci spojení, detekci přerušení spojení apod. Protokol také může udávat pravidla pro zabezpečení správnosti dat. Mezi nejčastější způsoby ošetření správnosti a úplnosti informace patří například údaj o počtu bajtů packetu, kontrolní součet nebo CRC. S protokolem také úzce souvisí jeho úplná a přesná dokumentace, která umožní co nejjednodušší implementaci do dalších zařízení. Jednotkou přenášené informace je u podstatné části protokolů tzv. Paket, což je blok dat, jehož podoba je definována právě formátem použitého protokolu.

1.4.1. KNX

V souvislosti s inteligentními budovami je příkladem komunikačního protokolu i protokol standardu KNX, který je navržen pro použití výhradně v inteligentních systémech pro řízení budov.

1.4.2. Modbus

Otevřený komunikační protokol, jeden z prvních protokolů určený ke komunikaci typu master/slave, vytvořený již v roce 1979 firmou MODICON. Protokol pracuje v aplikační vrstvě referenčního modelu ISO/OSI. Modbus se podle typu kódování dělí na binární verzi RTU a verzi ASCII s kódování podle tabulky ASCII. Verze RTU je v automatizaci hojně rozšířená. Jeden byte packetu se přenáší jako binárních 8 bitů. Verze ASCII přenáší každý bajt packetu jako dva znaky, tedy 2x 8 bitů. Výhodou ASCII kódování je lepší čitelnost oproti verzi RTU, ale narůstá také doba odesílání packetu.

1.4.3. Profibus

Profibus je průmyslový uzavřený protokol, který je definován v aplikační vrstvě ISO/OSI modelu. Kromě komunikačního protokolu definuje Profibus také pravidla pro fyzickou vrstvu. Má tedy předem určenou použitou sběrnici (RS-485, optické vlákno nebo proudová smyčka), pravidla pro použitou kabeláž. Na linkové vrstvě definuje pravidla pro přístup účastníka na přenosové médium a tvorbu bitového řetězce.

1.4.4. Spinel

Protokol je jednoduchý, otevřený komunikační protokol české firmy Papouch. Protokol bude podrobněji popsán v kapitole 4.

2. Idea vlastního řešení

Na základě získaných informací jsem se začal zabývat podobou a návrhem vlastního systému. V první řadě jsem si musel stanovit požadavky na vlastnosti takového systému. Dalším bodem návrhu je sestavení orientačního blokového schématu.

2.1.1. Požadavky na systém, výběr technologií

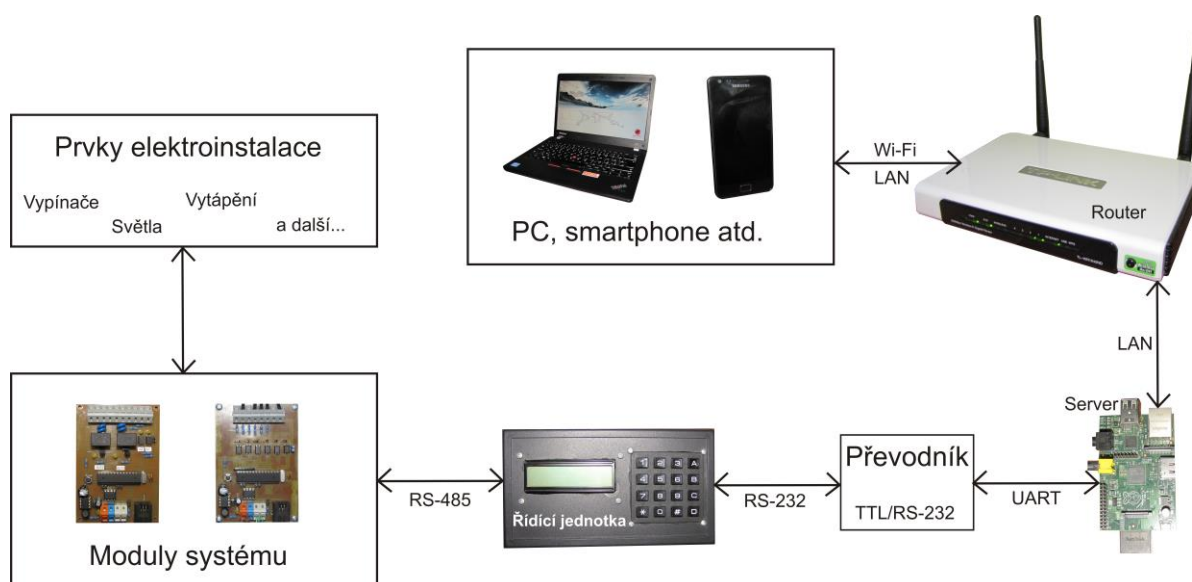
Od začátku se zabývám myšlenkou vytvořit co levný systém, který by svými funkcemi a možnostmi mohl konkurovat možnostem relativně drahých komerčních systémů. Požadavky na vlastnosti vycházejí z obecných požadavků pro tyto systémy. Můj systém by měl v budoucnu sloužit především pro amatérské konstruktéry a elektrotechnické nadšence

2.1.2. Systém složený z modulů

Jedním ze základních požadavků na budoucí hardware systému je jeho složení z jednotlivých modulů. Výstavbový systém je totiž velice výhodný a to z několika hledisek: Již nainstalovaný systém můžeme snadno rozšiřovat i v průběhu jeho používání, pouze ho připojíme k ostatním modulům a nastavíme (naprogramujeme) jeho funkci. Další výhodou je možnost vytváření dalších typů modulů a tím i rozšiřování možností systému, můžeme tak snadno vyhovět i požadavků, které vzniknou až po návrhu systému nebo reagovat na vznik nových technologií. V případě poruchy pak můžeme vyměnit pouze vadný modul, to nám šetří peníze a hlavně čas (nemusíme instalovat znovu celý systém).

2.1.3. Centralizovaný systém

Dalším požadavkem je systém centralizovaný pomocí řídicí jednotky. Řídicí jednotka bude navržena pro dvě sběrnice a bude mít úlohu posílání dotazů a vyhodnocení odpovědí nebo příkazů z obou sběrnic. První sběrnici bude průmyslová sběrnice RS-485 určená pro propojení všech modulů. Jako druhá sběrnice je použita RS-232 sloužící k propojení řídicí jednotky a domácího serveru, popř. inteligentního převodníku RS-232/LAN. Server, který je připojená k domácímu routeru pak umožňuje komunikaci s dalšími zařízeními jako je PC nebo smartphone, které slouží k externímu ovládní systému. Další možností využití připojení do domácí sítě je napojení na další systémy, jako je např. zabezpečení apod. Celý návrh je vidět na blokovém schématu níže.



Obr. 1 - Blokové schéma systému

2.1.4. Komunikace po lince RS-485

Komunikace na lince RS-485 bude specifikována jako komunikace typu master/slave. Řídicí jednotka, tedy master, bude vysílat postupně na všechny moduly dotazy. Když modul dostane dotaz, zpracuje ho a pošle odpověď. Tímto ošetřením se odstraňují případné kolize vzniklé neautorizovaným nebo náhodným vysíláním. Dalším nutným prvkem je použití komunikačního protokolu, který je vhodný pro komunikaci s více adresami. Já jsem zvolil protokol Spinel. Je přehledný a jeho implementace do zařízení je jedna z nejjednodušších oproti ostatním protokolům (např. Modbus).

2.1.5. Výběr výpočetní platformy

Při výběru vhodné jednočipové platformy připadaly v úvahu 8bit mikrokontroléry PIC od Microchipu a řada AVR od společnosti ATMEL. Mikrokontroléry PIC mají na českém trhu již tradiční postavení. Toto pevné postavení se však v budoucnu může změnit, konstrukcí s mikrokontroléry PIC v posledních ubývá. Některé firmy jako například ASIX již nenabízejí své vývojové desky pro tyto mikrokontroléry. Naopak mikrokontroléry AVR zažívají znatelný nástup. Vzniká spousta nových konstrukcí a vývojových prostředků pro tyto mikrokontroléry. Nemalý vliv na rozšíření jednočipů AVR má i použití v oblíbené platformě Arduino, která rovněž zažívá, především mezi amatéry, velice úspěšné období. Mikrokontroléry AVR mají modernější architekturu, která je přímo navržena pro použití s vyššími programovacími jazyky. Platforma AVR také nabízí větší počet jednoslovných instrukcí např. ATmega8 má instrukcí 130 (oproti tomu PIC16 pouze 35 a PIC18 75 instrukcí). To nabízí větší komfort i při psaní programu v jazyce symbolických adres. Pro platformu AVR hraje i fakt, že při stejné frekvenci externího krystalu je zhruba 4 krát rychlejší. U AVR totiž odpovídá jedna zpracovaná instrukce jednomu strojovému cyklu (1 MIPS na 1 MHz). I z těchto důvodů a také díky zkušenostem z předchozích projektů jsem nakonec zvolil platformu AVR.

3. Návrh a realizace hardwaru

Od teoretické přípravy a pouhé myšlenky vlastního řešení jsem přikročil k návrhu a realizaci konkrétních modulů a modulu řídicí jednotky. V době psaní této práce je zhotoveno celkem 8 různých prototypů hardwaru a v následujících podkapitolách se věnuji jejich popisu, popisu použitých součástek a výchozím zapojením. Schémata zapojení jednotlivých modulů a návrh DPS je umístěn v příloze.

3.1. Hardwarový základ pro moduly

Aby byl návrh modulů velice jednoduchý, rozhodl jsem se pro vytvoření jednotného hardwarového základu, ze kterého budou vycházet všechny navržené moduly. Výhody jednotného základu spočívají především v tom, že šetří čas při návrhu jednotlivých modulů. Vytvořený základ totiž nespočívá pouze v identickém schématu zapojení, ale také nákres jednotné rozmístění těchto základních součástek na plošném spoji. Při návrhu nového modulu (schématu nebo plošného spoje) pouze vložit již hotovou část a soustředit se pouze na řešení nového problému. Součástí hardwarového základu je stejný mikrokontrolér, převodník MAX485 pro připojení ke sběrnici RS-485, spínaný regulátor napětí +5V LM2574N-5, resetovací tlačítko, externí krystal 16MHz pro vnitřní krystalový oscilátor, svorkovnice Wago236 pro připojení sběrnice a napájení, volitelně i konektor RJ-11 pro testovací propojení po sběrnici RS-485.

3.1.1. Mikrokontrolér ATmega8

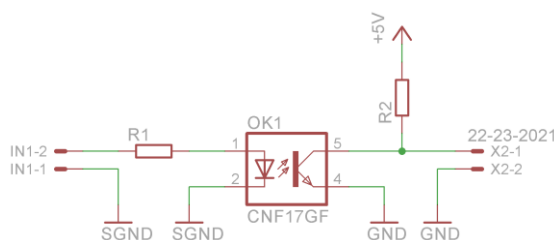
Základním stavebním prvkem a výpočetní jednotkou všech modulů je mikrokontrolér. Jak jsem si stanovil na začátku, výběr probíhal v řadě AVR. Mezi požadavky byla velká univerzálnost vybraného mikrokontroléru a střední počet vývodů. Konkrétně jsem vybral mikrokontrolér ATmega8 s 28 vývody. Výbava tohoto mikrokontroléru je následující: 8kB programové paměti FLASH, 1kB SRAM, 512B EEPROM pro uložení dat, 2x 8bit a 1x 16bit časovač, 3kanálové PWM, 4x 10bit a 2x 8bit A/D převodník, 1x UART, SPI, interní oscilátor. Maximální frekvence externího krystalu je 16MHz (16 MIPS).

3.2. Vstupně výstupní moduly

Vstupně výstupní moduly (jinak také I/O moduly) mohou být realizovány s různým počtem vstupů a výstupů, podle požadavků konkrétní aplikace. Mnou navržené a vyrobené moduly mají v prvním případě dva vstupy a dva výstupy. Druhý modul jsem navrhl se dvěma vstupy a čtyřmi výstupy. Konkrétně jsou pak výstupy vyhotoveny v podobě přepínacích kontaktů relé. Toto relé je spínáno stejnosměrnými 5V. Jeho kontakty může protékat až 3A. Tyto relé nejsou vhodné ke spínání síťového napětí, k tomu je nutné použití dalšího relé, které je umístěno např. V patici na lištu DIN. Relé nemůžeme spínat přímo mikrokontrolérem a to z několika důvodů. Proud nutný k sepnutí relé je větší, než maximální doporučené zatížení výstupního pinu mikrokontroléru (většinou 20mA). Indukční zátěž by také mohla mikrokontrolér poškodit. Já jsem pro spínání cívky relé použil tranzistor ve spínacím režimu. Vstupy jsou pak realizovány jako dvoustavové (zapnuto / vypnuto) a jsou galvanicky odděleny od mikrokontroléru optronem s bipolárním tranzistorem.

3.2.1. Vstupy

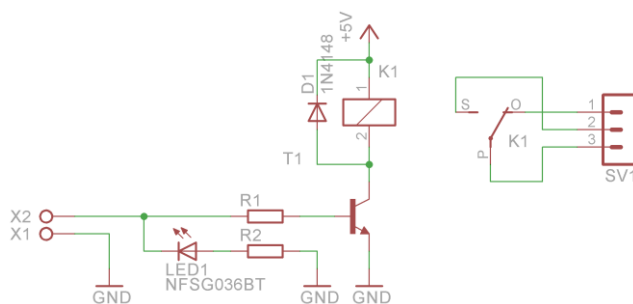
Jak už bylo řečeno, základním prvkem vstupů je optron. Optron se skládá ze neprůsvitného pouzdra, ve kterém je umístěna dioda LED, která svítí na světlocitlivý tranzistor. Princip funkce je tedy založen na rozsvícení svítivé diody a následném otevření fototranzistoru. Pro správnou funkci optronu musíme tedy vhodně nastavit pracovní bod svítivé diody. Vhodný proud LED diodou zjistíme nejlépe s katalogového listu námi zvoleného optronu. Tento proud se nastavuje odporem, který je k diodě připojen sériově (na obrázku označen R1). Anoda diody pak tvoří samotnou vstupní svorku, katoda je připojena na referenční svorku (zem), která je oddělena od země celého modulu. Výstup z optronu si můžeme představit jako tlačítko. Emitor tranzistoru je spojen se zemí napájení, kolektor připojíme na vstupní svorku mikrokontroléru. V klidovém stavu je na vstup mikrokontroléru nastavena log. 1 (přes rezistor R2 na +5V). Po sepnutí vstupu a otevření tranzistoru je hodnota vstupu změněna na log. 0.



Obr. 2 - Zapojení optronu

3.2.2. Výstupy

Zapojení výstupního relé je provedeno přes bipolární NPN tranzistor BC547B v nejzákladnějším spínacím režimu. Proud do báze tranzistoru je nastaven rezistorem R1, dioda LED1 indikuje sepnutí tranzistoru (popř. relé), její proud nastavuje rezistor R2. Cívka relé je umístěna v kolektoru tranzistoru T1 a napájena je napětím +5V. Emitor T1 je spojen se zemí. Kladná logická úroveň (log. 1) sepne tranzistor T1, ten se otevře a cívkou relé prochází proud. Relé se přepne z polohy P-O do polohy P-S. Změna logické úrovně na log. 0 uzavře tranzistor a rozepne kontakt relé. Dioda D1 slouží k ochraně tranzistoru T1.



Obr. 3 - Zapojení relé

3.3. Modul termostatu

Jedná se o nespojitý teplotní regulátor, jehož předností je především jeho variabilita. Termostat je osazen dvěma přepínacími kontakty v reléové podobě. Při použití jednoho relé můžeme realizovat klasický dvoustavový termostat pro vytápění (vypnuto a zapnuto), v případě použití dvou relé můžeme realizovat třístavový regulátor vytápění nebo kombinaci dvoustavového regulátoru vytápění a spínače kompresoru klimatizace (rychlost ventilátoru pak můžeme ovládat přidáním vstupně výstupního modulu). Dále je termostat osazen vstupem pro teplotní čidlo. Na současném prototypu má tento vstup tu vlastnost, že lze použít jak číslicové teplotní čidlo typu Dallas, tak jakékoliv analogové teplotní čidlo, jehož výstup je napětí v rozmezí 0-5V. Může se tedy jednat o standardní diskretní čidla (např. LM35, LM335 apod.) nebo o moduly teploměrů s výstupem, který splňuje uvedené vlastnosti. Pro zpracování teploty pak stačí znát pouze převodní konstantu, pomocí které převedeme změřené napětí na teplotu např. ve stupních Celsia. Z hardwarového hlediska volba analogového čidla nebo čidla Dallas otázka připojení rezistoru 4,7 k Ω mezi výstup čidla a napájecí napětí +5V. Volitelné připojení rezistoru je realizováno propojkou JP2 viz schéma termostatu v příloze č. 3. Při použití analogového čidla je možno použít volitelného kondenzátoru 100uF, zapojeného mezi signálový vodič čidla a zem. Zapojení kondenzátoru se provádí propojkou JP3. Připojení je pak provedeno zámkovým konektorem PSH02, ten však může být nahrazen svorkovnicí (např. typ Wago236).

3.4. Stavový displej

Modul stavového displeje je osazen 4řádkovým (16 znaků na řádek) inteligentním alfanumerickým displejem s řadičem kompatibilním s HD44780. Ten jsem zvolil z důvodu velkého rozšíření tohoto řadiče a velkého množství vyráběných displejů s tímto řadičem. Výroba desky elektroniky pro jinou velikost displeje pak obnáší pouze minimální úpravy a zapojení zůstává identické. Modul je dále osazen třemi tlačítky např. pro změnu údajů na displeji. Celý modul je připraven pro montáž do krabičky – místo svorkovnic wago236 je použita odpojitelná svorkovnice od společnosti PTR, která je vhodná pro vyvedení ven z krabičky. Jako tlačítka jsou použity kloboučkové spínače ZIPPY, rovněž vhodné pro vyvedení z krabičky.

3.5. Modul teplotních čidel

Tento modul jsem vyrobil především pro testovací účely. Je připraven pro připojení 4 analogových čidel LM355, ty jsou připojeny podobně jako u termostatu zámkovým konektorem PSH02. Čidlo LM335 je vlastně zenerova dioda a je určeno k měření teploty výhradně při napájení proudovým zdrojem, ten je zde vytvořen sériovým rezistorem od napájení +5V. Teplotu měříme jako napětí na diodě 10bit A/D převodníkem v uzlu mezi napájecím rezistorem a katodou diody viz schéma.

3.6. Modul reálného času

Modul reálného času slouží jako záloha a uchování hodin v době výpadku elektřiny nebo nemožnosti synchronizovat čas s internetem. Modul je osazen obvodem reálného času PCF8583 od firmy Philips. Tento obvod je vybaven hodinami a kalendářem a s procesorem komunikuje po sběrnici I²C. Modul je vybaven také držákem na lithiovou baterii CR2032 jako záložní napájení obvodu PCF8583.

3.7. Řídící jednotka

Řídící jednotka narozdíl od modulů nevychází přímo z hardwarového základu, avšak zachovává si některé jeho prvky. Zásadním rozdílem je použití jiného typu mikrokontroléru (rovněž z řady AVR), přidání sběrnice RS-232 ke stávající RS-485. Řídící jednotka je také vybavena 2řádkovým (16 znaků na řádek) inteligentním alfanumerickým displejem s řadičem kompatibilním s HD44780. Přítomna je také maticová hexadecimální klávesnice pro zadávání hodnot a ovládání jednotky. Řídící jednotka nemá žádné vstupy a výstupy ani žádné další funkce. Její úlohou je vysílat dotazy všem modulům a zpracovávat přijatá data (po sběrnici RS-485) a umožnit komunikaci s okolím po sběrnici RS-232. Celá řídící jednotka je zabudována do krabičky s vyvedenými důležitými konektory, displejem a klávesnicí. Velikost krabičky je 100x175mm (35mm hloubka). Součástí jsou i montážní úchyty, ale je zde i možnost přidělení držáku na lištu DIN.

3.7.1. Mikrokontrolér ATmega162

Při výběru vhodného mikrokontroléru byl kladen požadavek zejména na dostatečný počet pinů pro všechny periferie a přítomnost dvou hardwarových sériových linek UART. Z u nás běžně dostupných mikrokontroléru AVR splňovaly tyto požadavky ATmega644p a ATmega162. Z důvodu příznivější ceny a lepší dostupnosti jsem nakonec zvolil zmiňovaný ATmega162. Mikrokontrolér je integrován do pouzdra DIP40 (40 pinů). Vybaven je 16kB programové paměti FLASH, 1kB SRAM, 512B EEPROM pro data, analogovým komparátorem, 2x 8bit a 2x 16bit časovačem, čítačem, 6kanálovým PWM, 2 sériové linky, SPI, ladící rozhraní JTAG. Maximální frekvence krystalu je 16MHz (16 MIPS).

3.7.2. MAX232

K převodu napětí z úrovně TTL na napět'ovou úroveň RS-232 slouží v řídicí jednotce obvod MAX232 od firmy MAXIM. Jedná se o základní převodník/transceiver tohoto výrobce, je také velmi levný. Výhodou je, že pro výrobu napět'ových úrovní nepotřebuje zdroje +/- 15V, ale stačí mu pouze standardních +5V. Tento převodník je dvojitý, ale v této aplikaci je využita jen jeho polovina.

3.7.3. Konektorová výbava

Konektorová výbava řídicí jednotky je velice strohá. Vzhledem k předpokladu časté změny programu při vývoji, je uvnitř jednotky vyveden konektor ISP (In System Programing) pro sériové programování, bez nutnosti vyjmout mikrokontrolér z patice. Další konektory jsou vyvedeny ven z krabičky. Jako konektor pro napájení je použit napájecí jack s průměrem kolíku 2,1mm, ten slouží pro připojení napětí stejnosměrných 12-24V ze síťového adaptéru. Pro připojení ke sběrnici RS-485 slouží konektor RJ-11 (rozložení viz schéma zapojení v příloze č. 2). Připojeny jsou signály A, B a zem. Pro připojení protějšku na sběrnici RS-232 slouží konektor RJ-45. Jeho rozložení je kompatibilní s rozložením konfiguračního kabelu CISCO. Připojeny jsou vodič Rx, Tx a zem.

3.8. Zkušební model budovy

Pro realizaci zkušebních zapojení a ladění firmwaru byl realizován aplikační model budovy. Tento model obsahuje plánek rodinného domku s instalovanými světly v podobě modrých vysoce svítivých LED. Na plánu je v budoucnu možná realizace dalších zařízení nebo signalizačních prvků pro simulaci funkce těchto zařízení. Světla jsou připojena k I/O modulu se čtyřmi reléovými výstupy a dvěma vstupy. Na modelu se dále nachází Vstupní modul se šesti vstupy, na který jsou připojeny čtyři páčkové spínače (simulují použití klasických kolíbkových vypínačů). Na modelu je přítomen i termostat. Simulace vytápění probíhá na speciálním vytápěcím modulu, který je realizován čtyřmi zatíženími stabilizátory 7805. Teplotní čidlo je umístěno na kontaktní plošce pro umístění chladiče. Moduly jsou montovány na distančních sloupcích vedle sebe. Zbývající prostor mohou doplnit ostatní moduly, které zatím nejsou instalovány. Posledním instalovaným modulem je modul zobrazovací jednotky (stavový displej), který je umístěn nad předchozími moduly. Na modelu nechybí ani řídicí jednotka, ta je instalována vedle zobrazovacího modulu. Vše je propojeno po sběrnici RS-485. Veškeré napájení realizuje spínaný zdroj 12V, 2,1A, který je rovněž přimontován přímo na podkladové desce. Úplně poslední částí celého modelu je část obsahující domácí wi-fi router, malý server v podobě počítače Raspberry Pi a převodník TTL/RS-232 s napájecí částí pro server. Server je takto propojen s řídicí jednotkou.

4. Programová část

Vyrobené prototypy je potřeba ještě naprogramovat. Podstatnou část tvorby firmwaru tvoří implementace protokolu, to znamená vytvoření algoritmu zpracování datového paketu, implementace důležitých instrukcí protokolu apod. Následuje programování algoritmů řídicí jednotky, tzn. naprogramování všech požadovaných funkcí pro připojené moduly.

4.1. O protokolu Spinel

Protokol Spinel je standardizovaný protokol firmy Papouch. Protokol je plně otevřený a na webu je dostupná veškerá jeho dokumentace (obecný popis, postup implementace, příklady použití atd.). Je navržen pro bezproblémovou modifikaci a pro bezkolizní provoz více formátů tohoto protokolu na jedné lince.

4.1.1. Základní vlastnosti protokolu

Mezi hlavní přednosti protokolu spinel patří jeho snadná implementace do všech dostupných mikrokontroléru a dalších architektur včetně PC. Snadná modifikace protokolu vytvářením nových formátů (formát udává druhý byte odesílaného paketu, 0 - 96 pro ASCII kódování, 97 - 255 pro binární kódování). Definování vlastního formátu protokolu je velice jednoduché. Důležitou vlastností je také kompatibilita různých zařízení nezávisle na použité platformě (různé mikrokontroléry, PC nebo PLC). Vývoj zařízení s protokolem spinel je velice snadný, firma Papouch nabízí zdarma i jednoduchý Spinel terminál pro ladění aplikací.

4.1.2. Formáty definované firmou Papouch

Nejpoužívanějším formátem je binární formát 97, který je firmou Papouch doporučen pro většinu zařízení komunikujících na tomto protokolu. I ve svém projektu se snažím o implementaci tohoto binárního formátu. Dalším definovaným formátem je ASCII formát 65, ten je doporučen všude tam, kde je nutné použití formátu s ASCII kódováním. Posledním definovaným formátem je ASCII formát 66, tento formát je zjednodušeným formátem a slouží pouze pro nejjednodušší zařízení, jako jsou teplotní čidla.

4.1.3. Datový paket formátu 97

Datový paket binárního formátu 97 se skládá z tzv. prefixu (značen jako PRE), ten je pro všechny formáty stejný a vypadá takto „*“ (hexadecimálně 2Ah). Následuje znak FRM, který udává zvolený formát (jak už bylo řečeno, jedná se o desítkové číslo 97 nebo hexadecimální 61h). Pokračují dva bajty NUM, číslo, udávající délku řetězce. Specifickou součástí každého formátu je blok SDATA. Obsahem tohoto bloku je pro formát 97 adresa zařízení ADR (číslo 0 až 255), podpisový znak SIG, který slouží přiřazení odpovědi k odeslané otázce. Na následujícím místě se nachází v případě odesílané otázky znak zadané instrukce (INST). V případě odpovědi slave zařízení, se na tomto místě nachází znak ACK, který potvrzuje, zda byl dotaz proveden nebo zda nastala chyba a popř. jaká. Blok DATA, obsahující všechny bajty dat instrukce, určené ke zpracování. Následuje kontrolní součet SUMA a zakončovací znak CR (0Dh). Celý paket pak vypadá takto: „PRE; FRM; NUM; NUM; ADR; SIG; INST/ACK; DATA; SUMA; CR“. Kontrolní součet se vypočítá jako součet všech bajtů před SUMA a CR odečtených od 255. Tento způsob má tu nevýhodu, že obtížně odhalujeme chybu, protože stejný kontrolní součet může nastat v několika případech.

4.2. Vývojové prostředí Bascom AVR

BASCOM AVR je vývojové prostředí jazyka Basic pro mikrokontroléry Atmel AVR. Jedná se o integrované prostředí (IDE), zahrnuje tudíž vše potřebné pro napsání a editaci zdrojového kódu a jeho následného překladu do binární podoby souboru *.hex, který můžeme nahrát do mikrokontroléru pomocí integrovaného programovacího nástroje, nebo zvolit nějaký jiný. Jak už bylo řečeno syntaxe jazyka je velice podobná jazyku Basic, je tedy velice jednoduchá a i úplný začátečník je schopen během chvíle osvojit její základy.

4.3. Implementace protokolu Spinel

Implementace protokolu v prostředí jazyka Bascom probíhala nejprve realizací algoritmu pro zpracování jednoho datového paketu. Obrázek tohoto algoritmu je uveden v příloze společně s ukázkou zdrojového kódu. Potom následovala implementace samotných instrukcí protokolu pro jednotlivá zařízení. Vzhledem k rozsahu práce a velké časové náročnosti jsou prozatím implementovány pouze nejnütnější instrukce.

4.3.1. Regulace vytápění

Výhodou protokolu Spinel je to, že již obsahuje instrukce pro hlídání a regulaci teploty, dále také instrukce pro měření a čtení měřených hodnot. To velice usnadňuje softwarovou realizaci teplotního regulátoru. Vlastní měření teploty pak probíhá tak, že zadáme regulátoru horní a dolní mez regulované teploty, popř. jednu střední hodnotu. Tímto získáme překlápěcí úroveň regulátoru s hysterezí. Neustále pak porovnáváme měřenou teplotu, zda se nenechání v některé z překlápěcích úrovní. Naprogramovaný regulátor je pouze dvoustavový, tedy v horní překlápěcí úrovni vytápění vypínáme (dosáhli jsme maximální teploty) a v dolní překlápěcí úrovni vytápění opět zapínáme (dosáhli jsme teplotního minima). K měření teploty je použito analogové teplotní čidlo LM35, program tedy vyčítá hodnotu teploty z A/D převodníku. Vyčítání teploty je realizováno v přerušení, které je generováno časovačem, stane se tak přibližně každých 200ms.

4.3.2. Ovládání světel

Ovládání světel v domě je realizováno periodickým kontrolováním vypínačů připojených na I/O modulu se šesti vstupy. Řídící jednotka vyhodnotí zjištěné stavy a podle tohoto zjištění vysílá příkazy druhému vstupně výstupnímu modulu, na který jsou připojena světla. Světla lze ovládat i mimo hlavní smyčku po druhé sběrnici RS-232. Při použití klasických dvou polohových vypínačů je nutné zabezpečit, aby při vypnutí nebo zapnutí světel z externí sběrnice byl zaměněn význam poloh tohoto vypínače. Například pokud rozsvítím světlo vypínačem, který je připojen k modulu na sběrnici RS-485 a poté toto samé světlo vypnu jiným způsobem (např. pomocí aplikace běžící na serveru), nesmí mi toto světlo vypínač znovu rozsvít. Proto musí řídicí jednotka změnit význam polohy tohoto přepínače.

Závěr

Celou práci bych hodnotil velice kladně, protože mi dala spoustu cenných zkušeností z oblastí řídicí techniky a domácí automatizace. Počáteční cíl návrhu vlastního systému pro řízení rodinného domu jsem dovedl do stádia prvních prototypů a prvních funkčních verzí firmwaru. Náklady na výrobu jednotlivých modulů se pohybovali v cenovém rozmezí 500 – 1000 korun, tím jsem splnil i cíl relativně levných komponent. Do budoucna plánuji systém vhodně rozšířit o další prototypy a vylepšit některé staré. Dále plánuji použití výkonnějšího vývojového prostředí CodeVisionC AVR, které kompilátor jazyka C.

Seznam použité literatury a zdrojů

- [1] Wikipedia: RS-232[online]
Dostupný z WWW: < <http://cs.wikipedia.org/wiki/RS-232>>
- [2] Wikipedia: RS-485[online]
Dostupný z WWW: < <http://cs.wikipedia.org/wiki/RS-232>>
- [3] *Stránky LOXONE*[online]
Dostupný z WWW: < <http://www.loxone.com/cscz/start.html>>
- [4] *Stránky ABB*[online]
Dostupný z WWW: < <http://www.abb.cz/>>
- [5] Matoušek, David: *Práce s mikrokontroléry ATMEL AVR – ATmega16*
- [6] Váňa, Vladimír: *Mikrokontroléry ATMEL AVR – Programování v jazyce Bascom*
- [7] Datasheet ATmega8[online]
Dostupný z WWW: < http://www.atmel.com/images/atmel-2486-8-bit-avr-microcontroller-atmega8_1_datasheet.pdf>
- [8] Datasheet ATmega162[online]
Dostupný z WWW: < http://www.atmel.com/Images/Atmel-2513-8-bit-AVR-Microntroller-ATmega162_Datasheet.pdf>
- [9] Dokumentace protokolu spinel[online]
Dostupný z WWW: < <http://www.papouch.com/cz/website/mainmenu/spinel/>>
- [10] Datasheety dalších integrovaných obvodů [online]
Dostupný z WWW: < <http://www.gme.cz/>>

Použitý software

- [1] CadSoft EAGLE – kreslení elektronických schémat; freeware
- [2] Sprint-Layout 4 – návrh plošných spojů; školní licence
- [3] Bascom AVR – vývojové prostředí; demo, freeware

Seznam příloh

- [1] Význam a zapojení pinů RS-232
- [2] Schémata zapojení
- [3] Plošné spoje, osazovací plánky
- [4] Zpracování paketu (Spinel)
- [5] Ukázka zdrojového kódu (zpracování paketu)
- [6] Fotografie

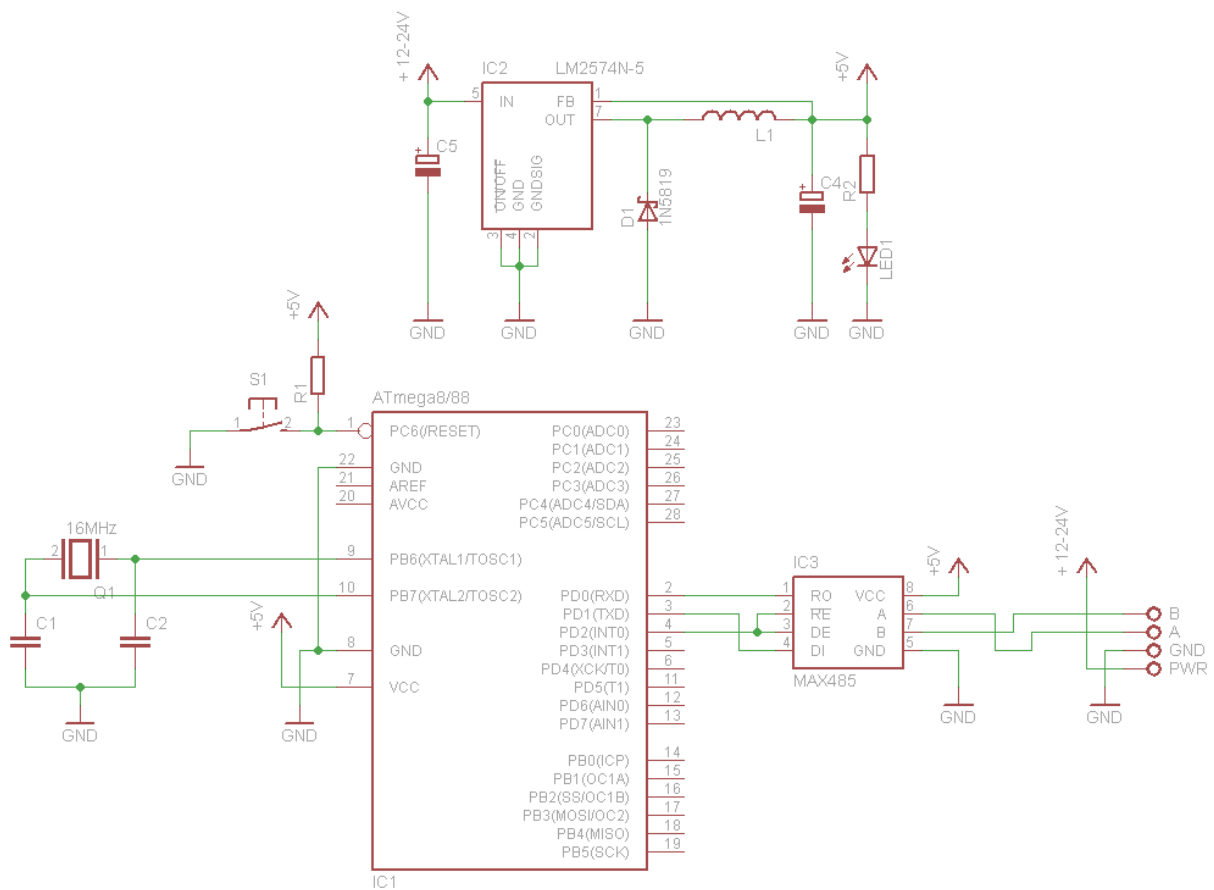
Příloha 1: Význam a zapojení pinů RS-232

Přehled zapojení a významu jednotlivých pinů na sběrnici RS-232. Tabulka je sestavena pro konektor DB-9, kdy protistrana představuje např. modem, který je připojený k PC

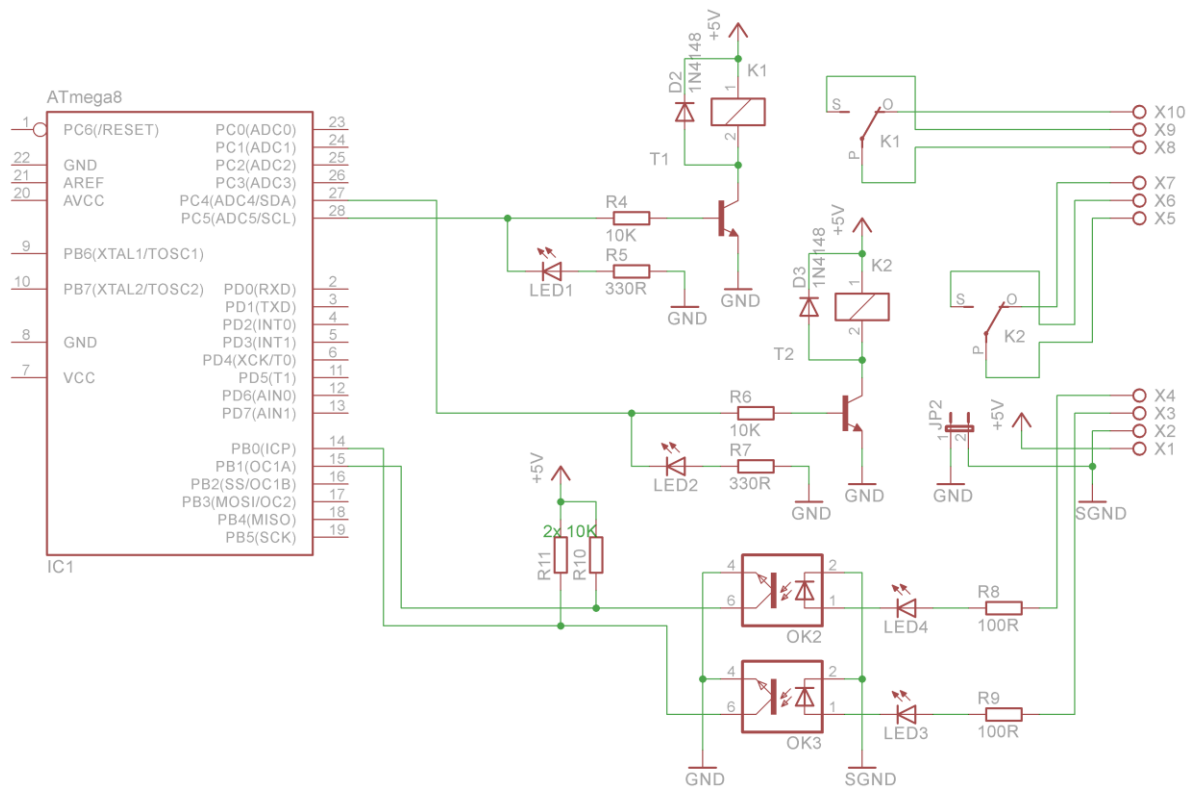
pin	Zkratka	Jméno pinu	Význam	
1	DCD	Data Carrier Detect	Log. 1 znamená, že protistrana detekovala nosný signál.	Vstup
2	RxD	Recieve Data	Data přijímané od protistrany	Vstup
3	TxD	Transmit Data	Data posílané protistraně	Výstup
4	DTR	Data Terminal Ready	Signalizace pro protistranu, že jsme připraveni.	Výstup
5	GND	Ground	Signálová zem.	-
6	DSR	Data Set Ready	Signalizace protistrany, že je připravena přijímat data.	Vstup
7	RTS	Request To Send	Zpráva protistraně, že chceme poslat data.	Výstup
8	CTS	Clear To Send	Povolení k vyslání. Protistrana signalizuje, že můžeme posílat data.	Vstup
9	RI	Ring Indicator	Signalizuje, že protistrana zaznamenala příchozí hovor (u modemu).	Vstup

Příloha 2: Schémata zapojení

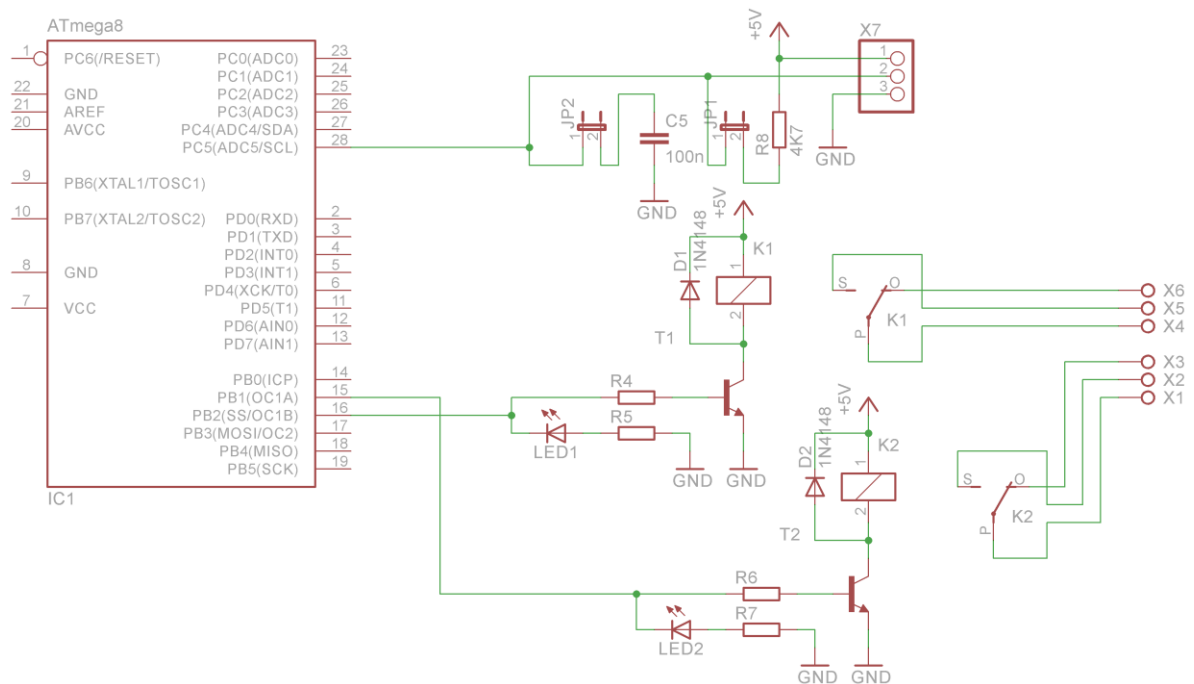
Schéma zapojení hardwarového základu, modulů z něj vycházejících a řídicí jednotky. U vstupně výstupních modulů je zobrazeno pouze schéma modulu 2/2, protože ostatní schémata jsou prakticky totožná, pouze mají jiný počet vstupů a výstupů.



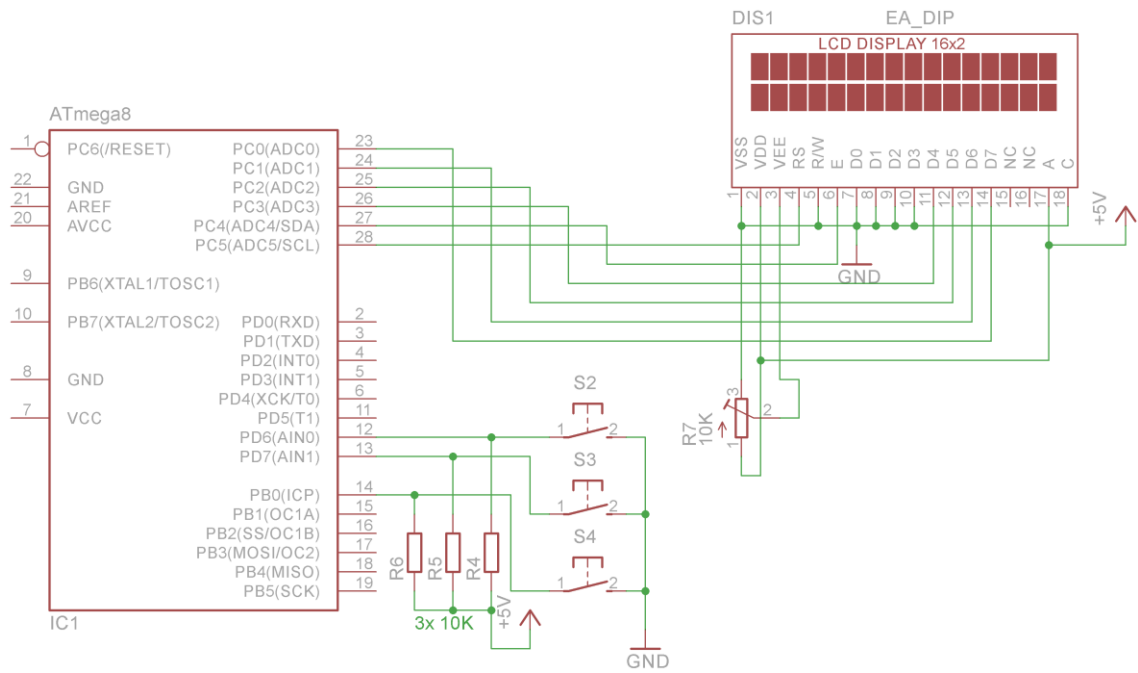
Obr. 4 - Schéma zapojení HW základu



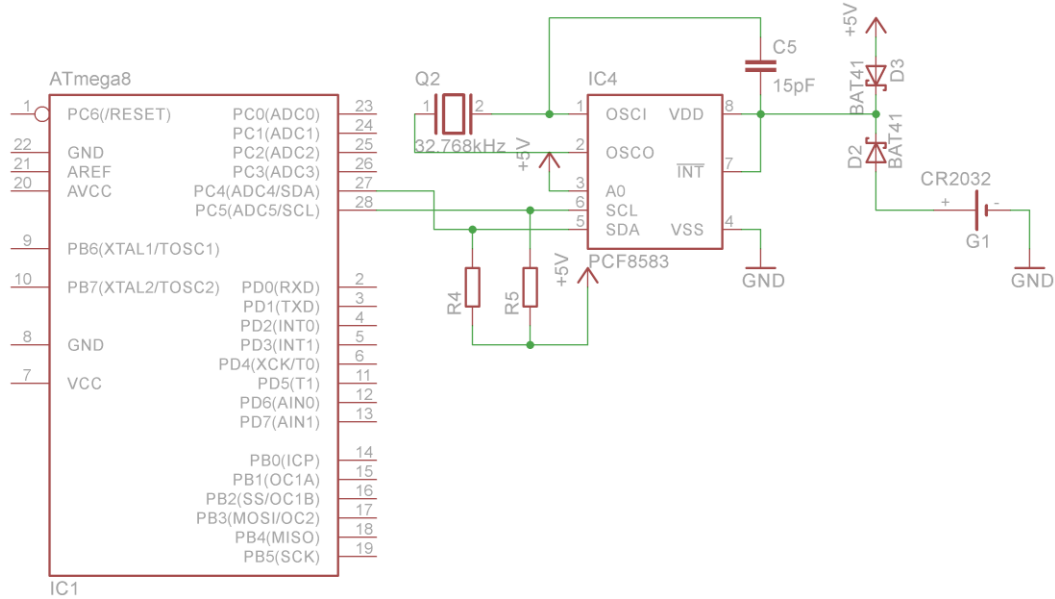
Obr. 5 - Zjednodušené schéma zapojení I/O modulu 2/2



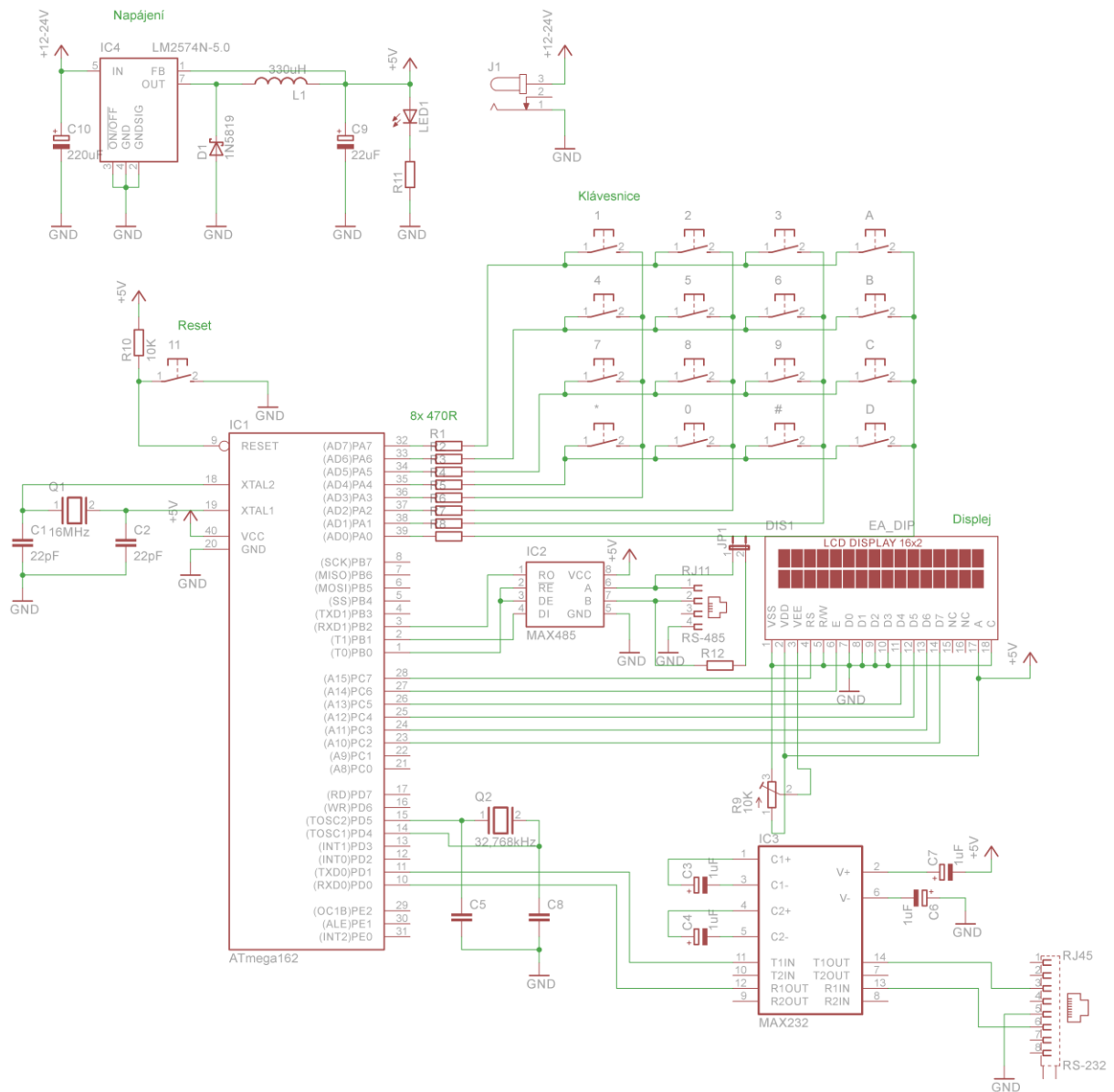
Obr. 6 - Zjednodušené schéma zapojení termostatu



Obr. 7 - Zjednodušené schéma zapojení stavového displeje

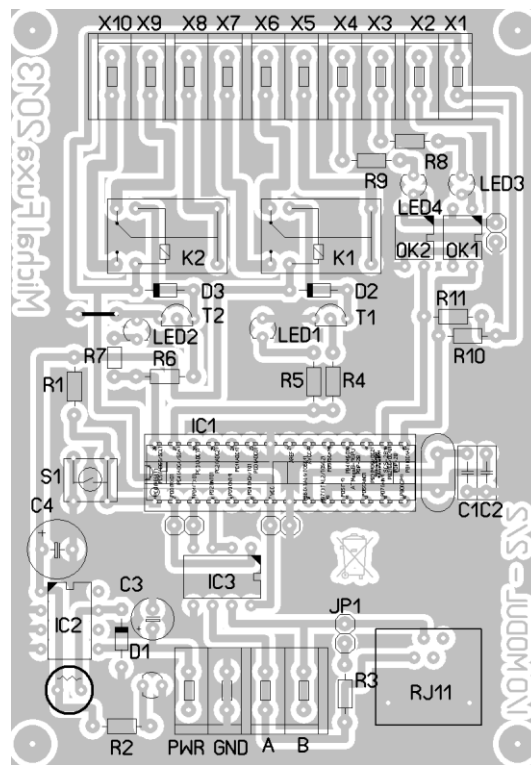


Obr. 8 - Zjednodušené schéma zapojení RTC modulu

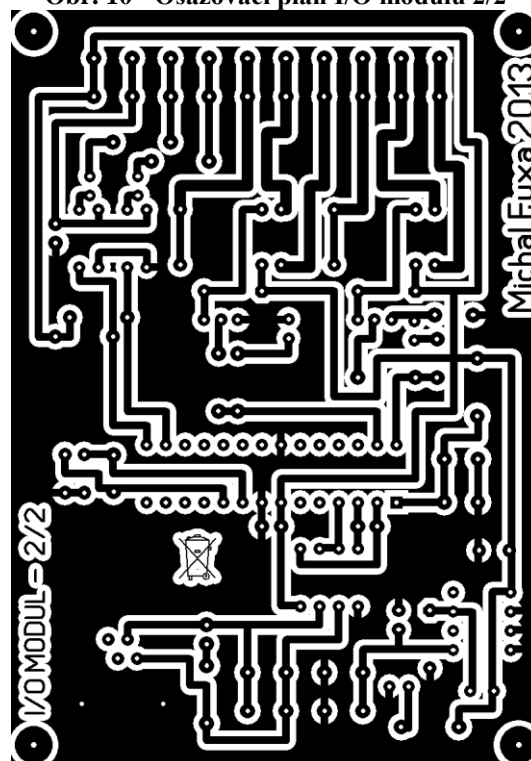


Obr. 9 - Schéma zapojení řídicí jednotky

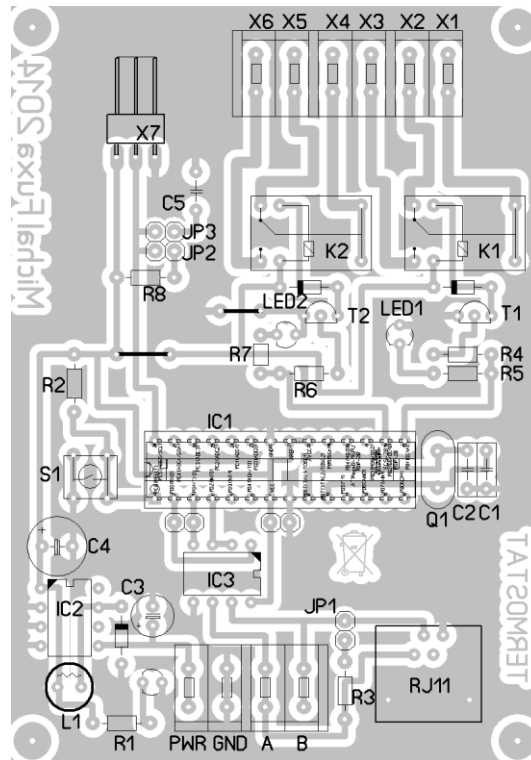
Příloha 3: Plošné spoje, osazovací plán



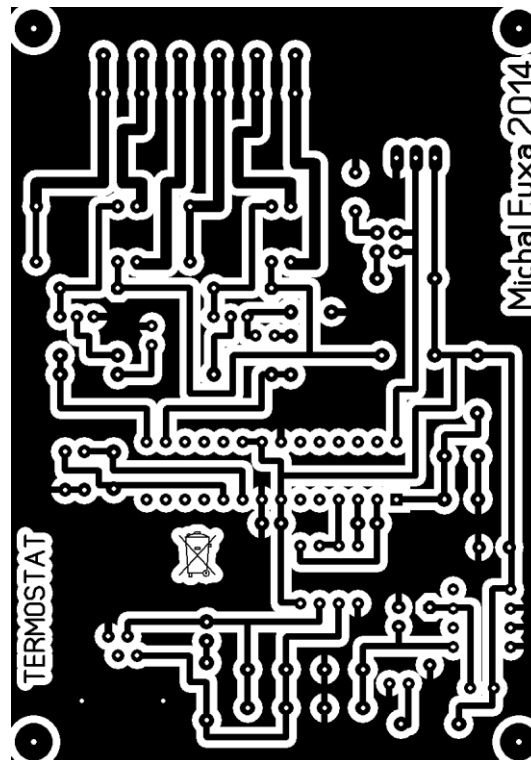
Obr. 10 - Osazovací plán I/O modulu 2/2



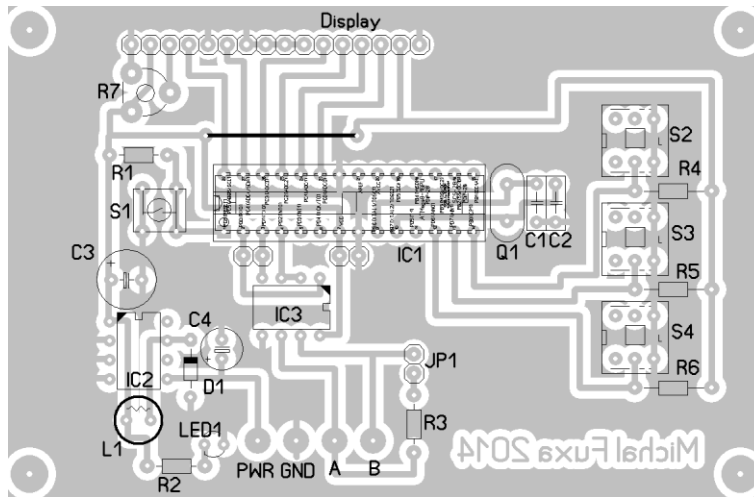
Obr. 11 - DPS I/O modul 2/2



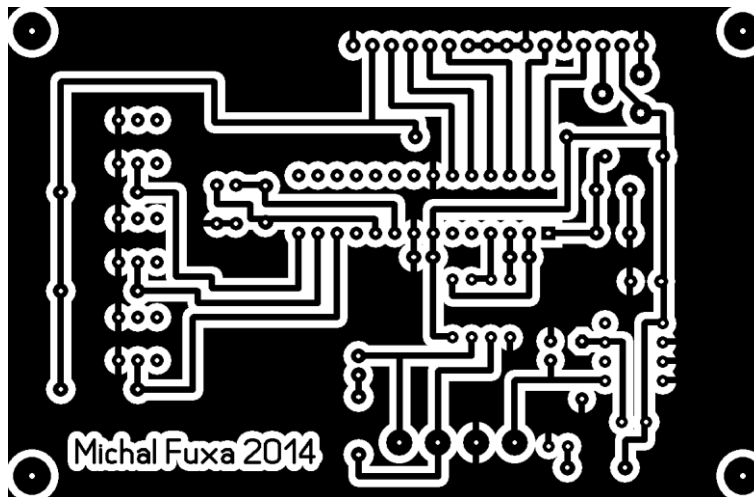
Obr. 12 - Osazovací plán termostatu



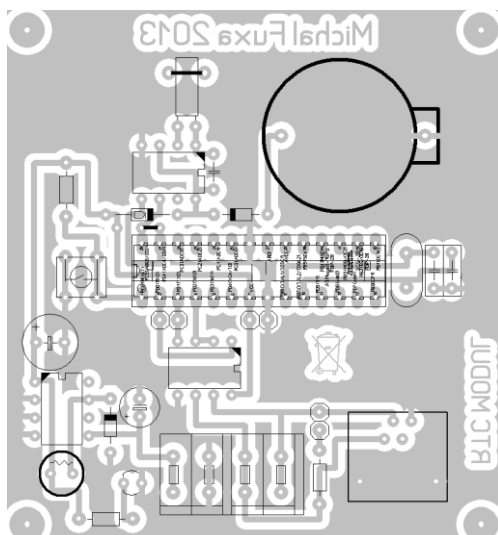
Obr. 13 - DPS termostat



Obr. 14 - Osazovací plán stavového displeje



Obr. 15 - DPS stavový displej

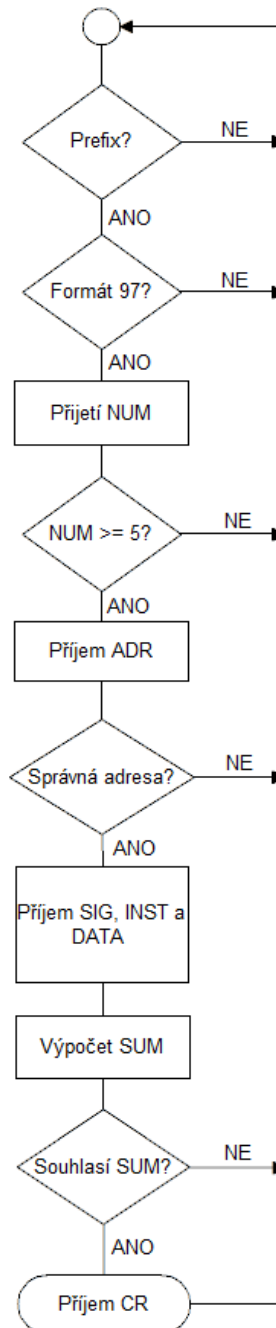


Obr. 16 - Osazovací plán RTC modulu



Obr. 17 - DPS RTC modul

Příloha 4: Zpracování paketu (Spinel)



Obr. 18 - Vývojový diagram zpracování paketu protokolu spinel

Příloha 5: Ukázka zdrojového kódu

```
Sub Spinelin
  Packet_sum = 0
  Packet = ""
  1byte = Inkey(#2)
  Packet_sum = Packet_sum + 1byte
  1byte_str = Hex(1byte)
  Packet = Packet + 1byte_str
  Do
    1byte = Inkey(#2)
    Packet_sum = Packet_sum + 1byte
    1byte_str = Hex(1byte)
    Packet = Packet + 1byte_str
  Waitms 1
  Loop Until 1byte = 13
znak
  Pre = Mid(packet , 1 , 2)
jednotlivé bajty paketu podle funkce
  Frm = Mid(packet , 3 , 2)
  Num_str = Mid(packet , 5 , 4)
  Num = Hexval(num_str)
  Ad = Mid(packet , 9 , 2)
  Sig_str = Mid(packet , 11 , 2)
  Sig = Hexval(sig_str)
  Ins = Mid(packet , 13 , 2)
  Dat_len = Len(packet) - 18
  If Dat_len > 0 Then
    Dat = Mid(packet , 15 , Dat_len)
  End If
  Position = 15 + Dat_len
  Sum_str = Mid(packet , Position , 2)
  Sum = Hexval(sum_str)
  Position = Position + 2
  Cr = Mid(packet , Position , 2)
  Packet_sum = Packet_sum - Sum
  Packet_sum = Packet_sum - 13
  Sum_aux = Packet_sum / 256
  Sum_aux = Sum_aux * 256
  Sum_calc1 = Packet_sum - Sum_aux
  Sum_calc1 = 255 - Sum_calc1
  If Pre = "2A" And Frm = "61" And Num >= 5 And Sum = Sum_calc1 Then
    'nyní
kontrolujeme zda paket splňuje všechny náležitosti
    If Ad = "FE" Or Ad = "01" Then
      If Ins = "00" And Sig_str = "01" Then
        'vyhodnocujeme přijatou instrikci/ack a
        Call Vypinace
      End If
    End If
  End If
End Sub
```

'Podprogram zpracování paketu
'Vynulování kontrolního součtu
'Vyprázdnění řetězce po předchozím paketu

'Čtení znaku ze vstupního bufferu
'Sčítání jednotlivých bajtů paketu
'převod bajtu na řetězec
'vytvoření řetězce paketu

'Toto se opakuje dokud nepřijde zakončovací

'nyní následuje postupné rozdělení řetězce na

'Odpočet delky bloku dat

'výpočet kontrolní sumy

'nyní
kontrolujeme zda paket splňuje všechny náležitosti

'vyhodnocujeme přijatou instrikci/ack a
podpisový znak sig

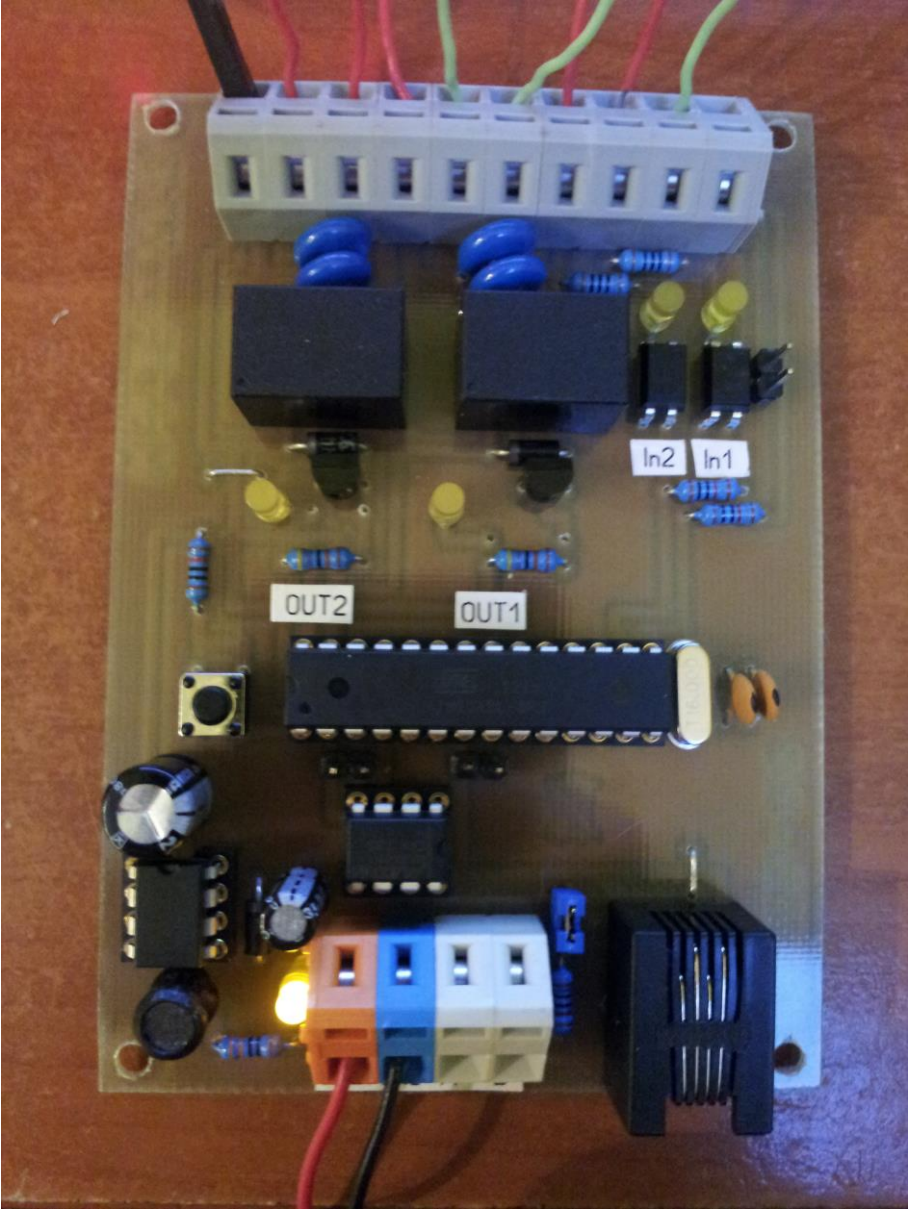
Call Vypinace

```
End If
If Ins = "00" And Sig_str = "02" Then

    Temp4 = Hexval(dat)
    Temp4_des = Temp4
    Temp4_des = Temp4_des / 10
    Temp4_str = Fusing(temp4_des , "##.#")

End If
End If
End If
End Sub
```

Příloha 6: Fotografie



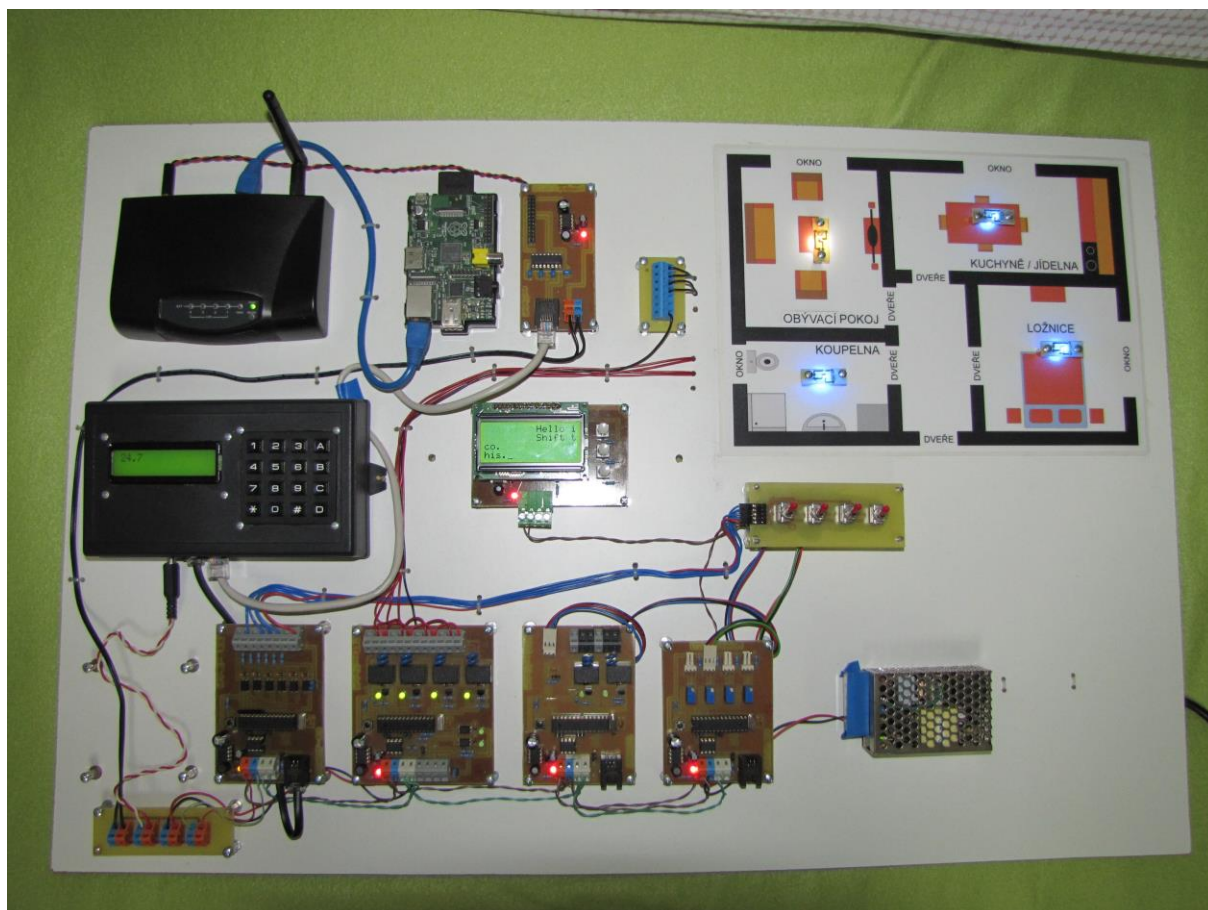
Obr. 19 – I/O modul 2/2



Obr. 20 – Řídicí jednotka



Obr. 21 – Řídicí jednotka, RTC modul, I/O modul 2/2



Obr. 22 – Model rodinného domu