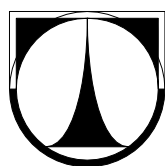


TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií



DIPLOMOVÁ PRÁCE

Liberec 2012

František Matyska

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: 3906T001 – Mechatronika

Řídicí jednotka s dálkovým ovladačem

Control unit with remote control unit

Diplomová práce

Autor: **Bc. František Matyska**

Vedoucí práce: **Ing. Zbyněk Mader, Ph.D.**

V Liberci 15. 3. 2012

Originál zadání

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum

Podpis

Poděkování

Chtěl bych poděkovat Ing. Zbyňku Maderovi, Ph.D., za jeho cenné rady, odborné znalosti při konzultacích a za vytvoření výborných podmínek pro realizaci mé diplomové práce.

Dále děkuji rodičům za jejich všeobecnou podporu během mého vysokoškolského studia.

Abstrakt

Diplomová práce se zabývá návrhem a realizací universální programovatelné řídicí jednotky a dálkového ovladače pro tuto jednotku. Systém je navržen tak, aby k němu bylo možno připojovat externí moduly, které budou s jednotkou komunikovat pomocí libovolné jednovodičové sériové komunikace. Systém má velmi široké možnosti využití, jelikož je možné k němu připojit modul s libovolnou funkcionalitou.

V první části nalezneme popis mikroprocesoru PIC16F887, který slouží jako řídicí prvek celého systému. Ten je možné programovat pomocí jazyka C a zapojení je navrženo pro programátor PicFlash programmer od firmy MikroElektronika. V druhé kapitole je popsána sériová komunikace SPI, která je využita pro komunikaci mezi řídicím mikroprocesorem a vysokofrekvenčními moduly RFM12B. Pomocí těchto modulů je realizována bezdrátová vysokofrekvenční komunikace mezi dálkovým ovladačem a řídicí jednotkou.

Hlavní část se zabývá návrhem zapojení pro řídicí jednotku a dálkový ovladač. Interakce s uživatelem je u řídicí jednotky zajištěna pomocí čtyř ovládacích tlačítek a dvouřádkového LCD displeje. Pro připojení externích modulů slouží šest konektorů RJ45. Jako zdroj energie je možné u řídicí jednotky využít buď externí stejnosměrný zdroj napájení, nebo napájení z uvedeného programátoru. Dálkový ovladač je také programovatelný a obsahuje dvanáct tlačítek. Napájen je lithiovým článkem CPAE-08/2005 a je zde integrovaný obvod MAX1555 pro dobíjení tohoto akumulátoru. Vše je řízeno stejně jako v případě řídicí jednotky mikroprocesorem PIC 16F887. Další část se zabývá inicializací a nastavením vysokofrekvenčního modulu RFM12B a jeho připojením k uvedenému mikroprocesoru. Pro tyto účely je zde navržena softwarová funkce SPI v jazyce C tak, aby byla plně duplexní a měla volitelnou šířku přenášeného slova.

V poslední části nalezneme návrh a realizaci plošných spojů a jejich implementaci do pouzder běžně užívaných ovladačů.

Klíčová slova

řídicí jednotka, dálkové ovládání, mikroprocesor, vysokofrekvenční přenos

Abstract

This diploma thesis deals with the concept and the realization of universal programmable control unit and the remote control for this unit. System is designed to be joined to external modules which will communicate with a unit via any single-wire serial communication. The system has a wide range of use because there is a possibility to connect a module with any functionality to it.

In the first part, there is a description of microprocessor PIC16F887 which serves as control component of the whole system. The microprocessor could be programmed by programming language C and is designed for PicFlash programmer from MicroElektronika Company. In the second chapter, there is described the serial communication SPI. This serial communication is used for communication between control microprocessor and high-frequency modules RFM12B. The high-frequency communication between remote control and control unit is realised via this modules.

The main part deals with the concept of connection for the control unit and the remote control. Interaction with a user is provided by four control buttons and two-line LCD display. There are six connectors RJ45 for connection of external modules. The source of energy of control unit could be external unidirectional power supply or power from mentioned programmer. The remote control is also programmable and it contains twelve buttons. It is charged by lithium cell CPAE-08/2005 and there is the integrated circuit MAX1555 for charging this accumulator. Everything is controlled by microprocessor PIC 16F887, same as in case of control unit. The next part is about initialization and setup of high-frequency module RFM12B and its connection to mentioned microprocessor. For these purposes, there is the software function SPI designed in programming language C to be duplex and to have selectable width of transported word.

In the last part, there is the concept of realization of printed circuit and its implementation to ordinary used cases of remotes.

Key words

control unit, remote control, microprocessor, high-frequency transmission

Obsah

Abstrakt.....	5
Abstract.....	6
Obsah.....	7
Seznam obrázků.....	9
Seznam použitých zkratk 10	10
Úvod.....	11
1 Mikroprocesor PIC 16F887	12
1.1 Piny mikroprocesoru	14
1.2 Programování mikroprocesoru PIC 16F887	16
2 HOPE – RF12B modul.....	20
3 SPI (Serial Peripheral Interface).....	22
3.1 Základní parametry	22
4 Řídicí jednotka	25
4.1 LCD displej.....	26
4.2 Tlačítka	29
4.3 Napájecí obvody.....	30
4.4 Programovací konektor.....	30
4.5 Komunikace s moduly	30
5 Dálkové ovládání	33
5.1 Zdroj napájení	33
5.2 Tlačítka	35
5.3 Signalizační LED diody.....	37
5.4 Programovací konektor.....	37
6 Komunikace mezi řídicí jednotkou a dálkovým ovladačem	38
6.1 Zapojení modulu	38
6.2 Nastavení modulu RFM12B.....	39
7 Návrh DPS.....	44
Závěr	46
Literatura	47

Příloha A – schéma zapojení dálkového ovladače	48
Příloha B – DPS dálkového ovladače	48
Příloha C – schéma zapojení řídicí jednotky	49
Příloha D – DPS řídicí jednotky	49
Příloha E – schéma zapojení řídicí jednotky s komunikací SPI	50
Příloha F – DPS řídicí jednotky s komunikací SPI	50

Seznam obrázků

Obrázek 1 Blokové schéma mikroprocesoru PIC16F887	12
Obrázek 2 Pouzdro TQFP 44DIP mikroprocesoru 16F887 [6].....	13
Obrázek 3 Nastavení funkce pinů pomocí registru TRIS.....	14
Obrázek 4 Přiřazení bitů registru PORTA a TRISA	15
Obrázek 5 Zapojení pull - up rezistorů na výstupy portu B	16
Obrázek 6 PicFlash programmer od firmy MikroElektronika [14]	17
Obrázek 7 Základní zapojení programovacího konektoru [14]	18
Obrázek 8 Použité schéma zapojení programovacího konektoru	19
Obrázek 9 Znázornění pouzdra modulu RFM12B s označenými vývody [1]	20
Obrázek 10 Základní schéma zapojení SPI	22
Obrázek 11 Propojení registrů obvodu master a slave	23
Obrázek 12 Příklad nastavení parametrů CPOL a CPHA	24
Obrázek 13 Řídící jednotka	25
Obrázek 14 Blokové schéma displeje	26
Obrázek 15 Vzájemné přiřazení pinů displeje a mikroprocesoru	27
Obrázek 16 Schéma zapojení displeje	27
Obrázek 17 Paměť DD ROM.....	28
Obrázek 18 Schéma zapojení tlačítek řídicí jednotky	29
Obrázek 19 Připojení tlačítek na piny mikroprocesoru	29
Obrázek 20 Schéma zapojení stabilizátorů	30
Obrázek 21 Schéma zapojení konektorů pro externí moduly	31
Obrázek 22 Blokové schéma nastavení přerušení na portu B	31
Obrázek 23 Znázornění speciálního funkčního registru INTCON	31
Obrázek 24 Znázornění speciálního funkčního registru IOCB	32
Obrázek 25 Blokové schéma obvodu Max1555 [13]	33
Obrázek 26 Závislost nabíjecího proudu na aktuálním napětí baterie [13]	34
Obrázek 27 Schéma zapojení zdroje napájení	34
Obrázek 28 Schéma zapojení tlačítek a jejich přiřazení k pinům mikroprocesoru.....	35
Obrázek 29 Schéma signalizační LED diody	37
Obrázek 30 Rozměry pouzdra použitého modulu [1].....	38
Obrázek 31 Popis pinů modulu RFM12B [1]	38
Obrázek 32 Zapojení pinů modulu RFM12B k pinům mikroprocesoru	39
Obrázek 33 VF modul - časový diagram pro zápis a čtení. [2].....	41
Obrázek 34 Power management command [2].....	43
Obrázek 35 Universální plošný spoj	44
Obrázek 36 Osazená DPS řídicí jednotky	44
Obrázek 37 DPS řídicí jednotky s komunikací SPI pro přídavné moduly	45
Obrázek 39 Dálkový ovladač	45
Obrázek 38 Náhled na DPS dálkového ovladače.	45

Seznam použitých zkratk

SPI	serial peripheral interface
PIC	pilot in command
LCD	liquid crystal display
DPS	deska plošných spojů
RISC	reduced instruction set computer
RAM	random access memory
ROM	read only memory
EEPROM	electrically erasable programmable read only memory
A/D	analog/digital
PC	personal computer
USB	universal serial bus
MCU	microcontroller unit
ICSP	in circuit serial programming
PGC	programmable clock
PGD	programmable data
MCLR	master clear
ISM	industrial, scientific and medical
FSK	frequency shift keying
PLL	phase locked loop
PA	power amplifier
LNA	low noise amplifier
SNR	signal to noise ratio
MOSI	master out, slave in
MISO	master in, slave out
CLK	clock
CS	chip select
MSSSI	master synchronous serial port

Úvod

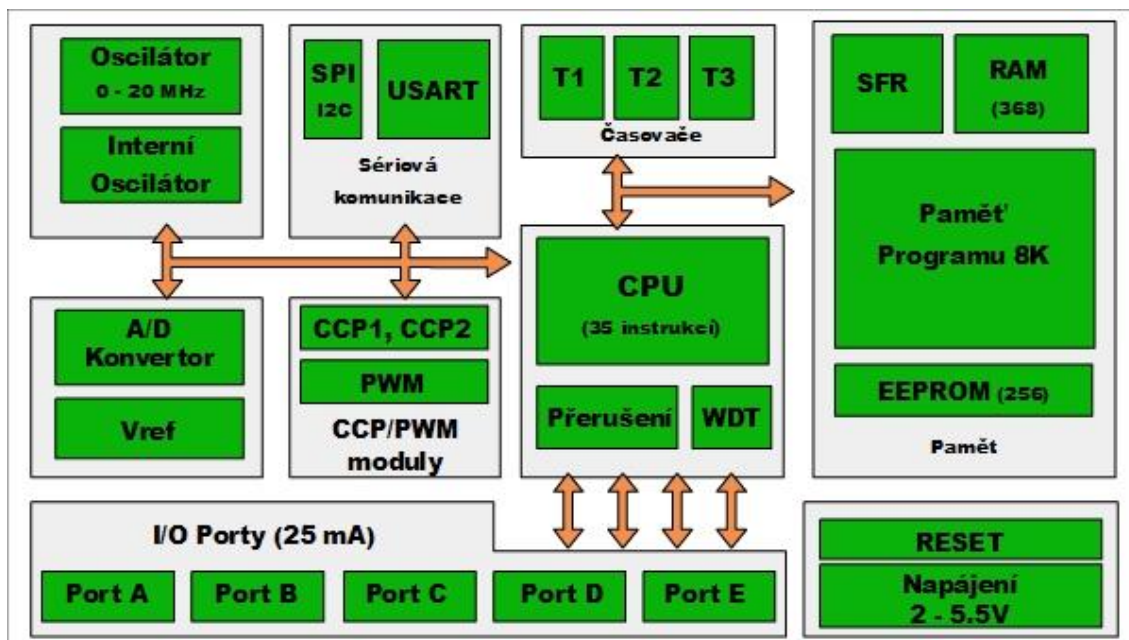
Řídicí jednotky jsou dnes k nalezení v mnoha zařízeních. Ve většině případů se jedná o jednotku, která je navržena pro předem známé komponenty, které bude obsluhovat. Ty pak nemáme možnost změnit a nemáme ani možnost měnit způsob jejich obsluhy. Cílem této diplomové práce je vytvoření univerzální programovatelné řídicí jednotky, ke které bude možné připojit další funkční moduly podle aktuální potřeby. Tedy takové řídicí jednotky, u které bude její funkcionality dána připojenými moduly a bude možné měnit i algoritmus pro ovládání těchto modulů. Tato jednotka bude sloužit při výuce na Ústavu informačních technologií na Fakultě mechatroniky, informatiky a mezioborových studií na Technické univerzitě v Liberci.

Jednotka má být ovládána pomocí tlačítek umístěných přímo na desce jednotky nebo pomocí dálkového ovladače komunikujícího s jednotkou na frekvenci 868 MHz. Komunikace mezi řídicí jednotkou a dálkovým ovladačem bude zajištěna pomocí vysokofrekvenčních modulů HOPE RFM12B, které jsou pro takovéto přenosy určeny. Jako hlavní řídicí obvod bude použit mikroprocesor PIC16F887, který bude jak v řídicí jednotce tak i v každém z připojených modulů. Pro vytvoření programu pro řídicí jednotku bude použit programovací jazyk C.

Práce si tedy klade za cíl vytvořit jádro celého systému, ke kterému budou jednotlivé moduly připojovány a vytvoření vzájemné komunikace mezi řídicí jednotkou a dálkovým ovladačem pomocí modulů RFM12B.

1 Mikroprocesor PIC 16F887

Tento mikroprocesor je produktem firmy Microchip. Má všechny obvyklé funkční komponenty, které moderní mikroprocesory v dnešní době mívají. Díky své nízké ceně a širokému rozsahu použití je velmi rozšířený a používaný v mnoha aplikacích, jako je řízení různých procesů v průmyslu, řízení strojů, měření elektrických i neelektrických veličin a podobně. Na obrázku 1 je znázorněné blokové schéma tohoto mikroprocesoru.



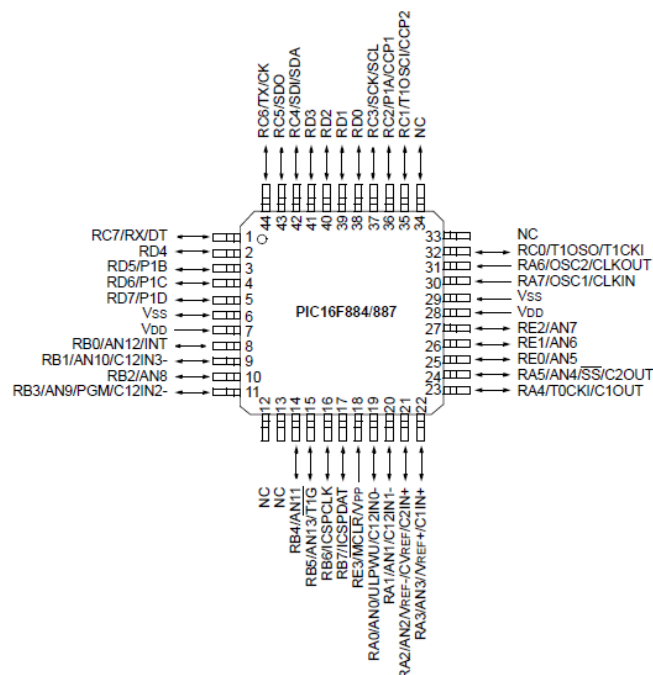
Obrázek 1 Blokové schéma mikroprocesoru PIC16F887

Základní parametry tohoto mikroprocesoru jsou následující. Architektura je typu RISC a využívá se tedy pouze 35 instrukcí. Všechny instrukce trvají čtyři hodinové cykly, kromě skoků, které jsou vykonány během dvou hodinových cyklů. Provozní frekvence oscilátoru je až 20 MHz. Procesor může pracovat při napájecím napětí od 2,0 do 5,5 V. Uváděná spotřeba je při 2,0 V a 4 MHz 220 μ A, při 2,0 V a 32 kHz 11 μ A a v režimu stand-by je uváděna spotřeba 50 nA.

Procesor je vybaven třemi typy pamětí, kde každá z nich má rozdílné vlastnosti a organizaci. Jsou to paměť typu Flash, RAM a EEPROM. Paměť typu Flash má kapacitu

8 kbitů a slouží pro uložení vykonávaného programu. Obsah paměti EEPROM může být, na rozdíl od obsahu paměti ROM, měněn i za běhu programu, a proto tato paměť slouží pro ukládání výsledků výpočtů, které jsou realizovány během vykonávání programu. Obsah této paměti zůstává nezměněn i po vypnutí napájecího napětí. Její kapacita je 256 bitů. Posledním typem paměti je RAM paměť. Ta se dělí na dvě části a to uživatelské registry, kam si můžeme ukládat dočasná data, která jsou potřebná při vykonávání programu a speciální funkční registry, které slouží pro nastavení integrovaných obvodů v čipu jako je například modul pro sériovou komunikaci, A/D převodníky, časovače a podobně. Speciální funkční registry i jejich jednotlivé bity mají svá jména, což výrazně zjednodušuje proces psaní kódu programu. Abychom mohli naplno využít této výhody, tak musíme použít takové vývojové prostředí, které zná přesné adresy takto pojmenovaných registrů. V našem případě je použito vývojové prostředí MicroPro C for PIC.

Mikroprocesor se vyrábí ve třech druzích pouzder. Jsou to 40PDIP, 44QFN a TQFP44. Pro naše účely bylo zvoleno pouzdro TQFP 44DIP, které je znázorněno na obrázku 2. Výhodou tohoto pouzdra oproti zbývajícím dvěma je to, že je možné jej ručně zapájet a při tom je poměrně malé.



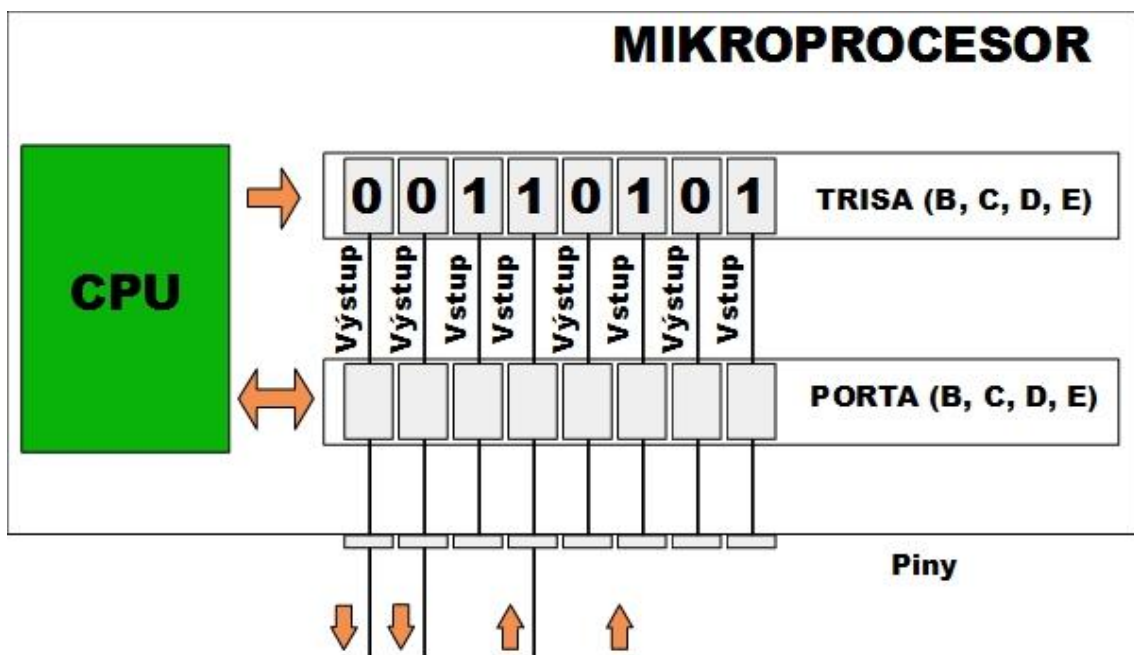
Obrázek 2 Pouzdro TQFP 44DIP mikroprocesoru 16F887 [6]

1.1 Piny mikroprocesoru

Všechny piny mikroprocesoru jsou rozděleny do pěti skupin - portů, které jsou pojmenovány A, B, C, D a E. Všechny tyto porty mají několik společných vlastností.

- Všechny piny jsou multifunkční.
- Každý port k sobě má přiřazený registr TRIS, který určuje, zda budou jednotlivé piny daného portu vstupem či výstupem.

Když nastavíme jakýkoli bit registru TRIS na logickou nulu, bude odpovídající pin nakonfigurován jako výstup, a opačně, když nastavíme jakýkoli bit registru TRIS na hodnotu logická jedna, bude odpovídající pin nakonfigurován jako vstup. Vše je znázorněno na následujícím obrázku.



Obrázek 3 Nastavení funkce pinů pomocí registru TRIS

Každý port má některé specifické vlastnosti, podle kterých je výhodné k nim připojovat jednotlivé periferie. Níže jsou popsány nejdůležitější parametry těchto portů.

Port A

Tento port má šířku 8 bitů. Nastavení funkce jeho pinů je realizováno pomocí registrů TRISA a ANSEL. Následující obrázek znázorňuje, jak jsou k sobě přiřazeny jednotlivé piny registru PORTA a bity registru TRISA. Pět pinů může pracovat nejen jako digitální vstupy či výstupy ale i jako analogové vstupy.

PORTA	R/W(x)	R/W(x)	R/W(x)	R/W(x)	R/W(x)	R/W(x)	R/W(x)	Název bitu
	RA7	RA6	RA5	RA4	RA3	RA2	RA1	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

TRISA	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	Název bitu
	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

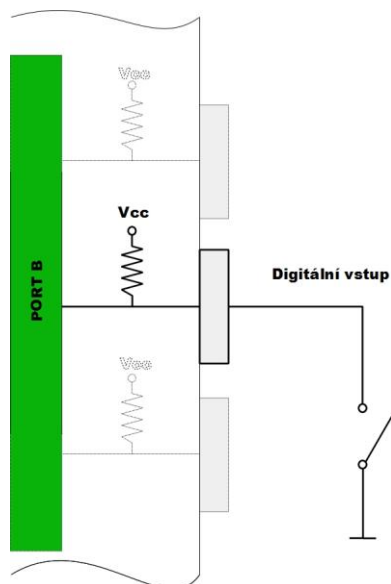
Obrázek 4 Přiřazení bitů registru PORTA a TRISA

Když nastavíme jakýkoli bit v registru ANSEL na hodnotu logická nula bude odpovídající pin nakonfigurován jako digitální vstup/výstup. Nastavíme-li však hodnotu logická jedna bude odpovídající pin nastaven jako analogový vstup a to již bez ohledu na nastavení registru TRISA.

PORT B

Tento port má také šířku osmi bitů a nastavování funguje obdobně jako u portu A. Nastavovací registry jsou však nazvány TRISB a ANSELH. Zde může být až 6 bitů použito jako analogový vstup.

Další funkcí, díky které je tento port velmi často využíván, je to, že všechny piny portu B v sobě mají integrovány pull-up rezistory, díky čemuž jsou velmi vhodné pro připojení tlačítek či celé klávesnice. Pro připojení těchto vnitřních pull-up rezistorů je zapotřebí nastavit příslušný bit v registru WPUB a nastavit bit RBPU v registru nazvaném OPTION_REG. Zapojení těchto rezistorů je znázorněno na obrázku 5.



Obrázek 5 Zapojení pull-up rezistorů na výstupy portu B

Další významnou funkcí portu B je možnost nastavení přerušení na změnu logické úrovně v případě, že je pin konfigurován jako digitální vstup. K tomuto nastavení je určen registr nesoucí název IOCB, dále je potřeba povolit přerušení na portu B pomocí bitu RBIE v registru INTCON a je potřeba povolit globální přerušení bitem GIE, který je v registru INTCON.

PORT C, D a E

Port C a D má šířku osm bitů a nastavení zda jsou jejich bity používány jako vstupy či výstupy se udává pomocí registru TRISC respektive TRISD. Port E má šířku čtyři bity. Tyto porty již nemají možnost využít některé piny jako analogové vstupy a nebudeme u nich využívat žádnou z jejich speciálních funkcí.

1.2 Programování mikroprocesoru PIC 16F887

Mikroprocesor vykonává program, který je uložen v jeho flash paměti a je dělen do slov, která mají šířku 14 bitů. Každé slovo představuje jeden příkaz, který má při spuštění programu mikroprocesor vykonat. U tohoto typu mikroprocesoru obsahuje instrukční sada celkem 35 různých příkazů.

Jako kompilátor byl použit program MIKROC PRO FOR PIC, který je určen pro psaní programů v jazyce C. Pro jednotlivé mikroprocesory obsahuje názvy registrů, jejich adresy, instrukční soubor a jména vývodů. Je tedy možné naplno využít všech výhod psaní programu ve vyšším programovacím jazyce.

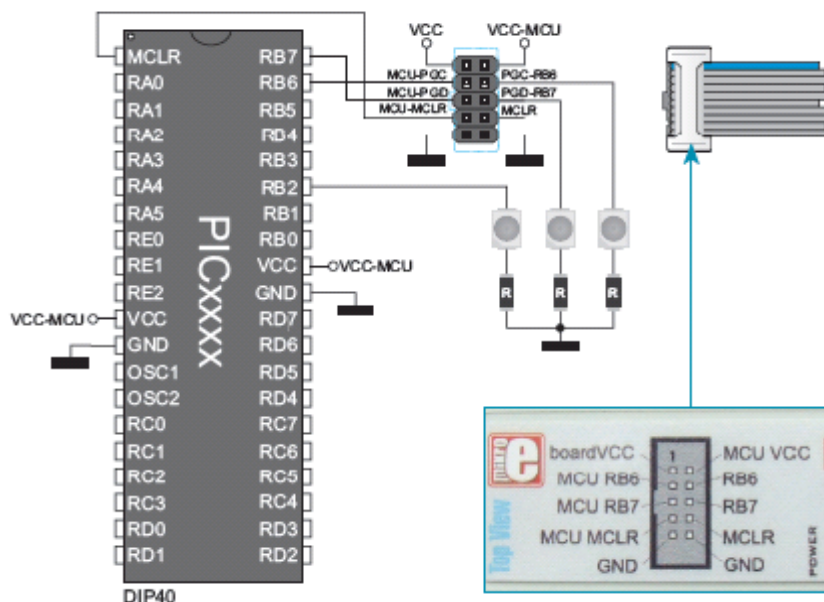
Jako programátor byl použit PicFlash programmer od společnosti MikroElektronika. Tento programátor komunikuje s PC pomocí protokolu USB, který je využit i pro jeho napájení. Aby bylo možné použít tento programátor, je nutné mít příslušný software mikroProg Suite pro PIC. Ten je dodáván při zakoupení programátoru nebo jej lze zdarma stáhnout na webových stránkách uvedené společnosti. V kombinaci s kompilátorem MikroC PRO for PIC poskytuje snadný způsob ladění a simulaci činnosti cílového zařízení. Ladicí program mikroICD je součástí programátoru. Ten umožňuje ladit program přímo na desce krok za krokem a sledovat stav všech registrů mikroprocesoru.



Obrázek 6 PicFlash programmer od firmy MikroElektronika [14]

Připojení programátoru

Programátor PICflash je připojen k mikroprocesoru pomocí kabelu lat, který je zakončen konektorem IDC10. Piny mikroprocesoru je tedy nutné pro programování připojit na pinovou lištu 2x5 tak, jak ukazuje následující obrázek.



Obrázek 7 Základní zapojení programovacího konektoru [14]

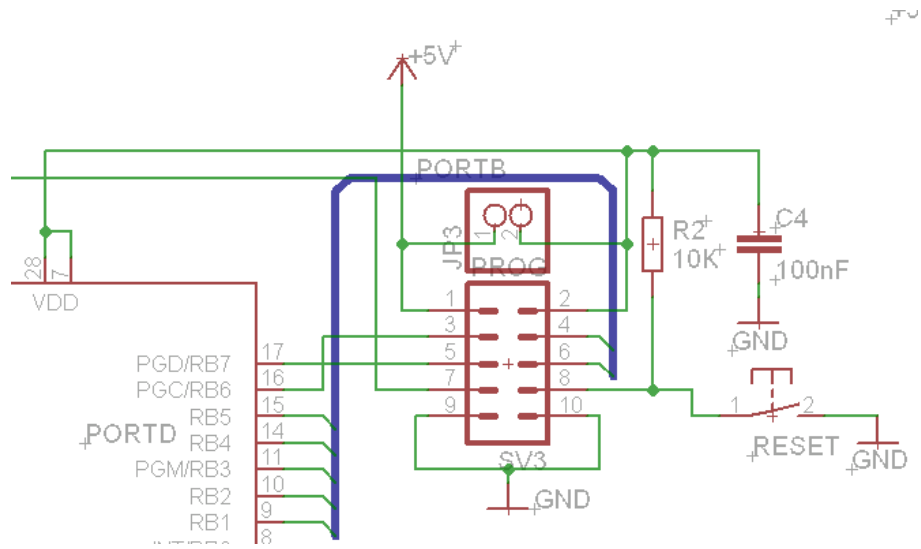
Programátor PICflash je možné napájet z PC prostřednictvím portu USB nebo z cílového zařízení. V tomto případě je využita první možnost, tedy napájení z portu USB. Konektor je vybaven klíčem, aby nemohlo dojít k jeho chybnému zapojení. Samotný programátor využívá pět pinů pro připojení k mikroprocesoru. Dva piny jsou použity pro napájení. Ostatní tři piny jsou používány pro přenos dat a pro přepínání mikroprocesoru do programovacího režimu.

Piny konektoru využité pro programování:

- MCU-VCC - napájení z PICflash programátoru
- MCU-PGC – hodinový vstup (In-Circuit Debugger a ICSP programování)
- MCU-PGD – datový vstup (In-Circuit Debugger a ICSP programování)
- MCU-MCLR/Vpp – reset obvodu a programovací napětí (Master Clear)

Po dokončení procesu programování, můžeme odstranit IDC konektor PICflash programátoru. Tím ale zůstanou odpojeny piny mikroprocesoru použité pro programování (RB6, RB7, MCLR). Aby bylo možné tyto piny využít, je nutné umístit propojky na konektor 2x5 tak, aby byly spojeny piny MCU-PGC s PGC-RB6, MCU-PGD

s PGC-RB7 a MCU-MCLR s MCLR. Na následujícím obrázku je znázorněno použité schéma zapojení programovacího konektoru.

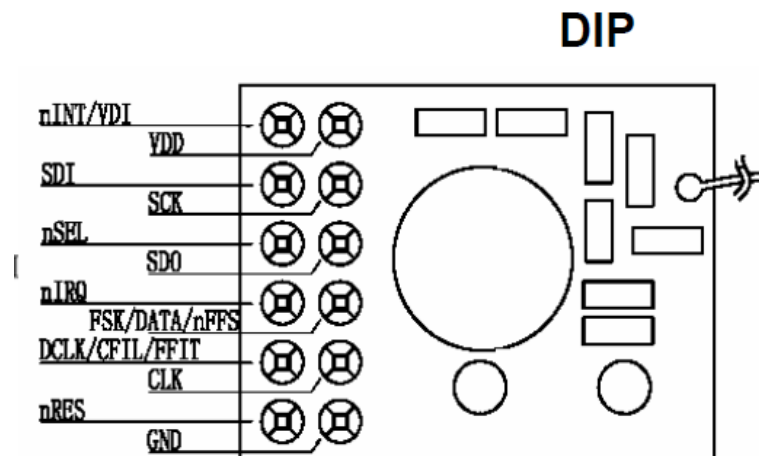


Obrázek 8 Použité schéma zapojení programovacího konektoru

2 HOPE – RF12B modul

Tento modul od firmy HOPE MICROELEKTRONICS byl vybrán především pro jeho dostupnost a snadné vytvoření bezdrátové komunikace. Komunikační frekvence těchto modulů je možné volit mezi 433 MHz, 868 MHz a 915 MHz. Při použití frekvence 868 MHz tedy vyhovíme požadavku zadání. Všechny tři frekvence spadají do bezlicenčního ISM pásma a vyhovují i předpisům Federal Communications Commission a European Telecommunications Standards Institute.

Transceiver je využíván pro komunikaci mezi dálkovým ovladačem a řídicí jednotkou. Jeho jádro tvoří integrovaný obvod RF12B využívající FSK modulaci. Jak pro přenos dat, tak pro přenos konfiguračních příkazů mezi mikroprocesorem a tímto modulem je využito rozhraní SPI. Napájecí napětí tohoto modulu je v rozmezí od 2,2 do 3,8 V. Modul je pro připojení do obvodu vybaven pinovou lištou DIP 2x6 pinů, která má rozteč pinů 2,00 mm. Popis funkce jednotlivých pinů modulu je na následujícím obrázku.



Obrázek 9 Znáornění pouzdra modulu RFM12B s označenými vývody [1]

Tento modul výrazně snižuje zatížení mikroprocesoru díky tomu, že má mnoho funkčních bloků integrovaných přímo v sobě. Patří mezi ně například filtr dat, integrovaný registr pro vysílání a přijímání dat, automatické řízení kmitočtu, které umožňuje použití méně přesných krystalů. Transceiver obsahuje vysílací registr sloužící

pro odesílání dat, který má šířku slova 8 bitů. Je zde také možnost využít krystal osazený přímo na modulu jako zdroj hodinového signálu pro připojený mikroprocesor. Pro další snížení spotřeby je možné využít integrovaný wake-up časovač. Dále tento modul obsahuje integrovaný multi-pásmový PLL syntezátor, výkonový zesilovač, nízkový zesilovač s nastavitelnou úrovní zisku (LNA) a další.

Programovatelný syntezátor PLL, neboli smyčka fázového závěsu, udržuje pracovní frekvenci s nastavenou přesností. Jako referenční kmitočet je brán signál generovaný krystalem, který je osazený přímo v modulu PLL. V obvodu je implementován výkonový zesilovač s otevřeným kolektorem. Má diferenciální výstup a může přímo ovládat anténu požadovaným výstupním výkonem. Velkou předností je také integrovaný obvod pro automatické ladění antény. LNA je zkratka z anglického Low noise amplifier. Jedná se o elektronický zesilovač používaný pro velmi slabé signály. Například signály zachycené anténou. Zesilovač je umístěný co nejbližší k anténě tak, aby byl zesílen přijímaný signál co nejdříve bez zbytečných ztrát na vedení. Tím je zajištěn co nejlepší poměr signál/šum (SNR).

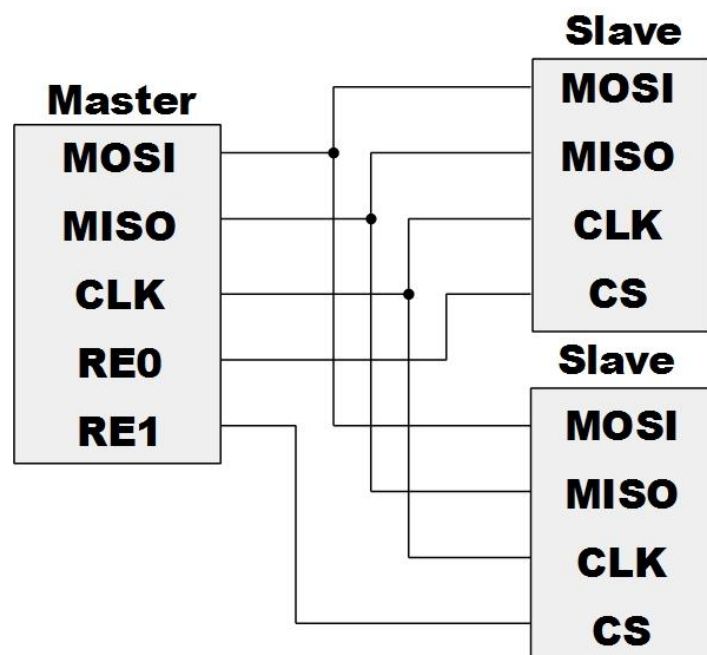
Abychom tento obvod mohli používat, je nutné nejprve provést jeho inicializaci. To je realizováno pomocí konfiguračních příkazů a popis této činnosti najdeme v šesté kapitole této diplomové práce.

3 SPI (Serial Peripheral Interface)

SPI je synchronní sériové rozhraní, které většinou slouží pro komunikaci mezi řídicím a periferními obvody nebo pro vzájemnou komunikaci dvou mikroprocesorů. V této diplomové práci je toto rozhraní využito pro komunikaci řídicího mikroprocesoru s modulem RFM12B.

3.1 Základní parametry

Rozhraní SPI může být jak jednosměrné tak obousměrné. Data jsou přenášena po dvou vodičích, po každém vodiči je možné komunikovat jen v jednom směru. Z tohoto důvodu jsou na plně duplexní komunikaci potřeba dva datové vodiče. Adresace je realizována adresovými vodiči. Pro každou periférii je potřeba samostatný adresový vodič. Realizace SPI vždy zahrnuje nejméně dva obvody, z nichž jeden, obvykle procesor, je typu master a ostatní jsou typu slave. Obvod master generuje hodinový signál pomocí kterého je řízena veškerá komunikace. Obvody slave pak, pokud jsou aktivovány příslušným adresovým signálem, v taktu tohoto hodinového signálu data buď přijímají či vysílají.



Obrázek 10 Základní schéma zapojení SPI

Základní schéma zapojení SPI je uvedeno na předchozím obrázku. K propojení jsou tedy využity tyto čtyři vodiče: MOSI, MISO, CLK a CS

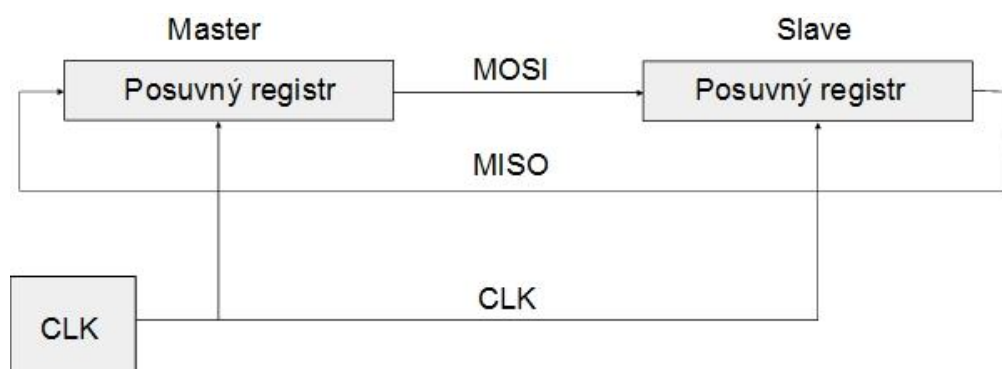
MOSI (master out, slave in) – datový výstup obvodu master je připojen na datový vstup obvodu slave.

MISO (master in, slave out) – datový výstup obvodu slave je připojen na datový vstup obvodu master.

CLK (clock) – hodinový výstup obvodu master je připojen na hodinové vstupy všech obvodu slave.

CS (chip select) – vstup CS, někdy také označován SS (slave select) nebo CE (chip enable), je připojen ke každému obvodu typu slave. Je-li CS v neaktivní úrovni, je daný obvod neaktivní a jeho výstup je ve vysoko-impedančním stavu. Vstupy CS jsou s obvodem master propojeny samostatnými vodiči. Je-li ve funkci obvodu master mikroprocesor, tak se tyto vodiče připojují k některým z jeho portů, čímž je zajištěn snadný výběr obvodu, s kterým se má komunikovat.

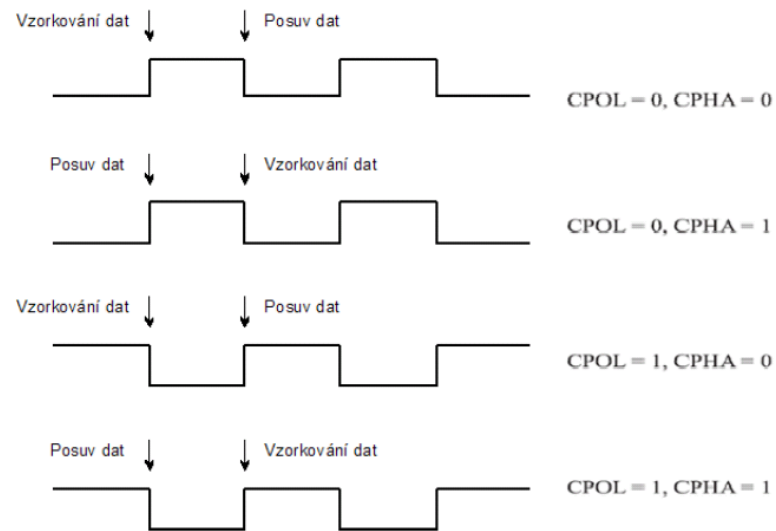
Při jakékoli komunikaci jsou data přenášena mezi obvodem master a některým z obvodů slave. Tyto obvody mají v době přenosu dat propojeny posuvné registry tak, jak je zobrazeno na následujícím obrázku.



Obrázek 11 Propojení registrů obvodu master a slave

Hodinový signál generovaný obvodem master řídí oba posuvné registry. Na obrázku 12 je zobrazen vztah mezi datovým a hodinovým signálem při různých nastaveních parametrů CPOL a CPHA. Tyto parametry se dají nastavovat při použití

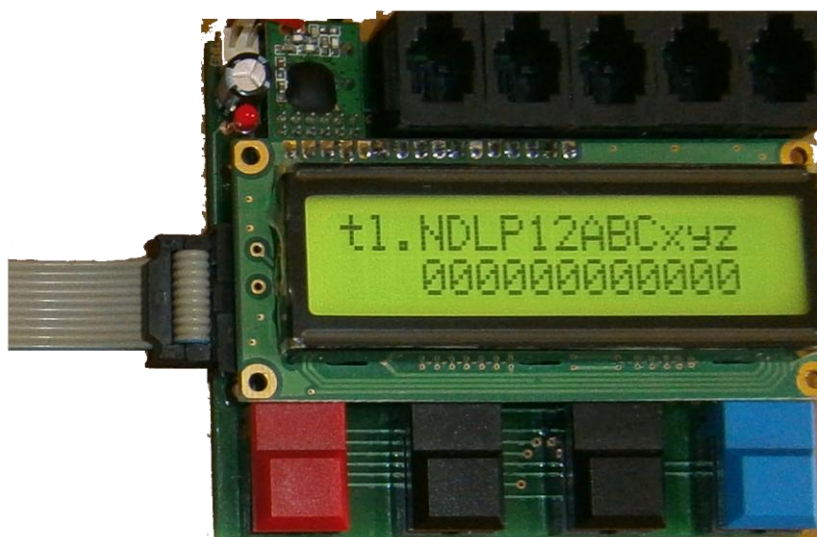
specializovaného řadiče SPI. Při programové realizaci se musí časování nastavit tak, aby byla vzorkována ustálená data.



Obrázek 12 Příklad nastavení parametrů CPOL a CPHA

4 Řídicí jednotka

Tato univerzální programovatelná řídicí jednotka je navržena tak, aby k ní mohly být připojovány externí moduly, které budou pomocí této jednotky řízeny. Hlavní předností této jednotky je možnost jejího programování pomocí jazyka C, díky čemuž získává velký rozsah použitelnosti.



Obrázek 13 Řídicí jednotka

Komunikace mezi uživatelem a řídicí jednotkou je řešena pomocí čtyř tlačítek a dvouřádkového LCD displeje. Řídicí jednotku je také možné ovládat pomocí dálkového ovladače. Bezdrátová komunikace je realizována pomocí transceiveru RFM12B, který je v jednotce implementován.

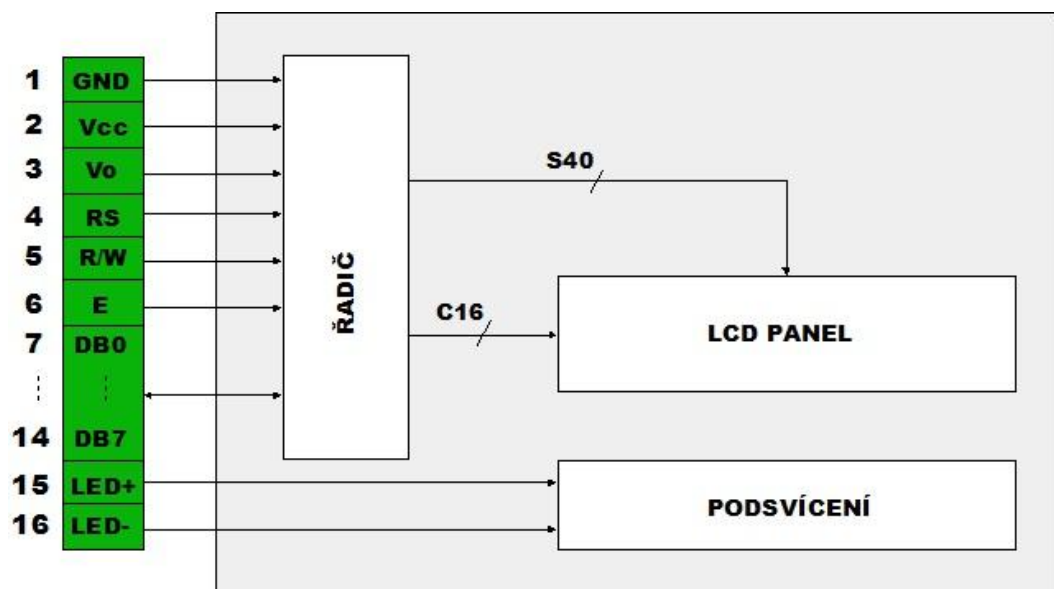
Návrh komunikace mezi jednotlivými moduly a řídicí jednotkou není předmětem této diplomové práce, avšak je zde hardwarová koncepce navržena tak, aby mezi sebou mohli komunikovat mikroprocesory PIC16F887, které jsou obsaženy jak v řídicí jednotce, tak v každém modulu. Použitý mikroprocesor je popsán v kapitole 1.

Jednotka obsahuje následující komponenty.

- LCD displej
- Tlačítka na ovládání ošetřená proti zákmitu
- Konektor pro programování mikroprocesoru
- Napájecí obvody
- Konektory pro připojení periférií

4.1 LCD displej

Pro zobrazování potřebných informací řídicí jednotky byl použit alfanumerický dvouřádkový LCD displej SIC1602A, který má integrovaný vlastní řadič.



Obrázek 14 Blokové schéma displeje

Na předchozím obrázku je znázorněno jeho blokové schéma. Pro připojení displeje slouží jeho šestnáct vývodů.

GND - napájecí zem

Vcc - napájecí napětí +5V

Vo - kontrast displeje (rozsah připojeného napětí je od -0,3 V do Vdd +0,3 V)

RS - ukazatel, zda jsou po vývodech DB0 až DB7 posílány řídicí příkazy či data, která se mají zobrazit na displeji.

R/W - nastavení zapisování či čtení do displeje respektive z displeje

E - enable displeje jak pro zapisování, tak pro čtení dat.

DB0 až DB7 - Datové piny. Jsou po nich posílány jak data, tak řídicí příkazy.

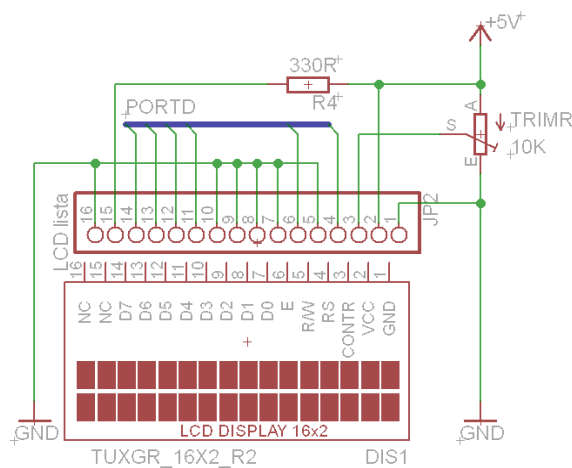
LED+ a LED- - Podsvícení displeje. Mají ale i funkci druhého enablu celého displeje. Není-li na ně přiloženo napájení, pak není možná komunikace s displejem.

Propojení jednotlivých pinů displeje a mikroprocesoru je následující.

pinová lišta displeje	Vss	Vdd	Vo	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	LED+	LED-
odpovídající piny mikroprocesoru	GND	5V	0-5V	RD5	GND	RD4	GND	GND	GND	GND	RD3	RD2	RD1	RD0	5V	GND

Obrázek 15 Vzájemné přiřazení pinů displeje a mikroprocesoru

Jak je vidět ze zapojení jednotlivých pinů, jsou využity pouze čtyři datové piny pro komunikaci s mikroprocesorem a to proto, aby připojení displeje nezabíralo zbytečně mnoho pinů použitého mikroprocesoru. Schéma zapojení je na následujícím obrázku.



Obrázek 16 Schéma zapojení displeje

Kontrast displeje je možné ovládat pomocí trimru, který je připojen na pin displeje nesoucí název VO. Čím větší napětí na tomto pinu bude, tím bude kontrast větší a naopak.

Inicializace

Abychom mohli displej používat je potřeba jej inicializovat, tedy nastavit zda bude displej nastaven jako jednořádkový či dvouřádkový, kolik bude použito datových vývodů, zda bude automaticky inkrementována adresa při příchodu hodinového signálu, nastavit zda má být kurzor vidět či nikoliv, nastavení počáteční pozice kurzoru a smazání obsahu displeje. Jednotlivé příkazy pro inicializaci je možné nalézt v technické dokumentaci tohoto displeje [12].

Zápis dat do displeje

Po inicializaci displeje je možné do něj zapisovat znaky. Každý znak, který může být zobrazen tímto displejem, je určen osmibitovou informací, která je zapsána v jeho řadiči. Podle toho zda je komunikace displeje a mikroprocesoru nastavena jako osmibitová nebo čtyřbitová tedy trvá i délka zápisu. Ta může trvat buď jeden hodinový impuls, nebo dva hodinové impulsy. V tomto případě trvá dva hodinové impulsy, jelikož je využito čtyřbitové komunikace. Každému bytu tedy odpovídá jeden konkrétní znak. Znakovou sadu pro použitý displej je možné nalézt v technické dokumentaci tohoto displeje [12].

Požadované znaky jsou zapisovány do paměti displeje DD ROM, jejíž obsah je zobrazován na samotném displeji. Schematické znázornění této paměti je na obrázku 16.

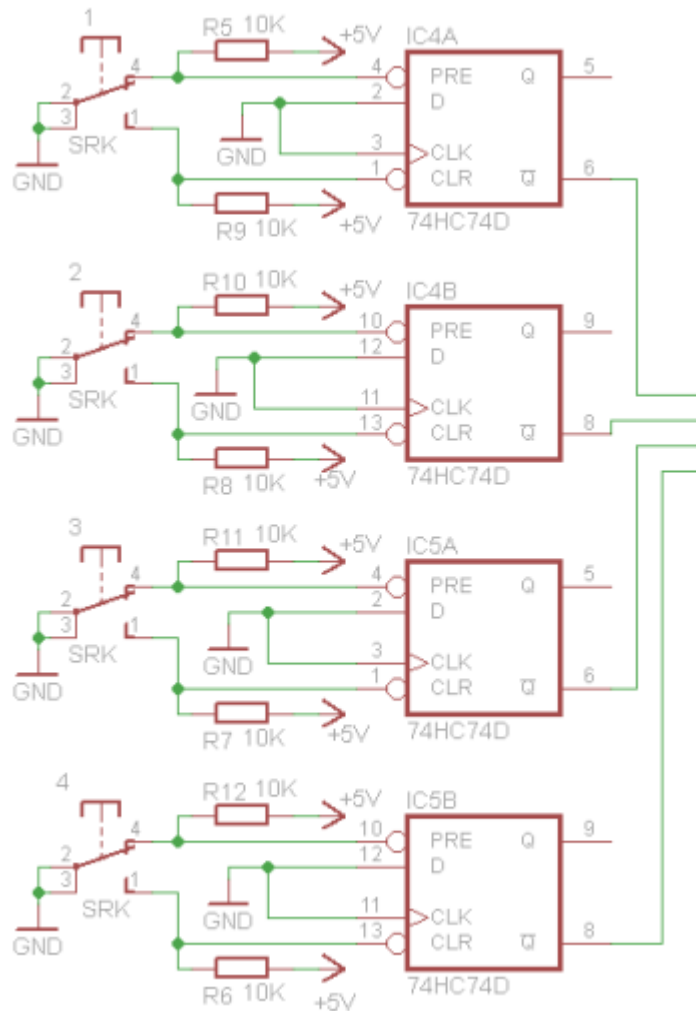
1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	...	27
2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	...	67
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	...	40

Obrázek 17 Paměť DD ROM

Při statickém režimu zobrazení jsou zobrazeny znaky uložené na prvních šestnácti pozicích v každém řádku. Tedy znaky uložené na adresách 00 až 0F respektive 40 až 4F. Pro zobrazení znaků uložených na adresách 10 až 27 respektive 50 až 67, je nutné displej rotovat o příslušný počet míst pomocí příkazu shift. Seznam všech možných příkazů pro ovládání displeje lze nalézt v jeho technické dokumentaci [12] i s dobou, která je pro vykonání daného příkazu potřebná.

4.2 Tlačítka

Řídicí jednotka je vybavena čtyřmi tlačítky P-B1710. Ty mohou být využity dle libosti programátora buď k nastavení řídicí jednotky, nebo k jejímu ovládání při běhu programu. Tlačítka jsou hardwarově ošetřena proti zákmitu pomocí klopných obvodů D. K tomuto byly použity integrované obvody 74HC74 SMD. K nastavení výstupní logické hodnoty těchto klopných obvodů jsou použity nastavovací vstupy reagující na hodnotu logická nula nazvané PRE a CLR. Tyto vstupy jsou připojeny přes pull-up rezistory. Hodinový vstup CLK a vstup D jsou



Obrázek 18 Schéma zapojení tlačítek řídicí jednotky

uzemněny. U těchto integrovaných obvodů byly využity negované výstupy z toho důvodu, aby při stisknutí tlačítka byla na příslušném vstupu mikroprocesoru hodnota logická nula a při nestisknutí tlačítka hodnota logická jedna. Tedy stejné hodnoty jako při použití pull-up rezistorů na portu B.

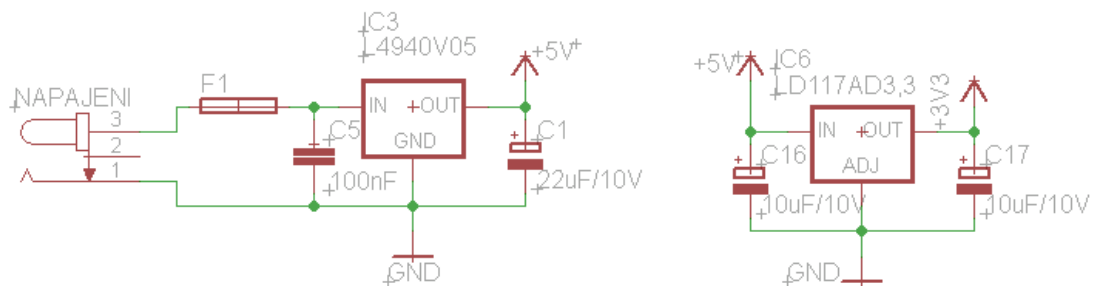
Na obrázku 18 je schéma zapojení těchto tlačítek a na obrázku 19 je přiřazení jednotlivých tlačítek k pinům použitého mikroprocesoru.

čísla přepínačů DPI lišty	1	2	3	4
odpovídající piny mikrokontroleru	RD7	RD6	RE1	RE2

Obrázek 19 Připojení tlačítek na piny mikroprocesoru

4.3 Napájecí obvody

Napájecí obvod je tvořen konektorem pro připojení externího zdroje napětí o jmenovité hodnotě 6 až 7 V a dvěma stabilizátory napětí. První stabilizátor je L4940V05, který stabilizuje napětí na hodnotu 5 V a zajišťuje potřebné napětí pro chod mikroprocesoru, tlačítek, displeje, signalizační LED diody a všech ostatních komponent kromě VF modulu RFM12B. Ten je potřeba napájet nižším napětím, 2,2 V až 3,8 V, a k tomu je zde určen druhý stabilizátor LD117AD3,3, který stabilizuje napětí na hodnotu 3,3 V. Schéma jejich zapojení je na následujícím obrázku.



Obrázek 20 Schéma zapojení stabilizátorů

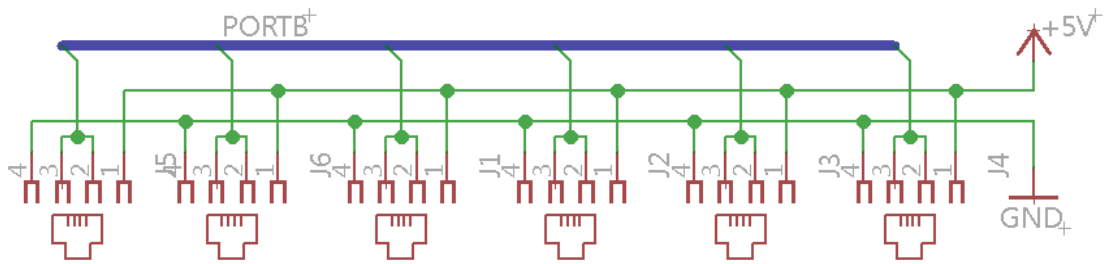
4.4 Programovací konektor

Dále je zde připojen programovací konektor, jehož zapojení je popsáno v kapitole 1.2. Díky tomu může být funkce celé řídicí jednotky měněna podle toho, jaké externí moduly k ní jsou připojovány či jaké funkcionality jsou požadovány. Jednotka díky tomu má velmi široký rozsah použití.

4.5 Komunikace s moduly

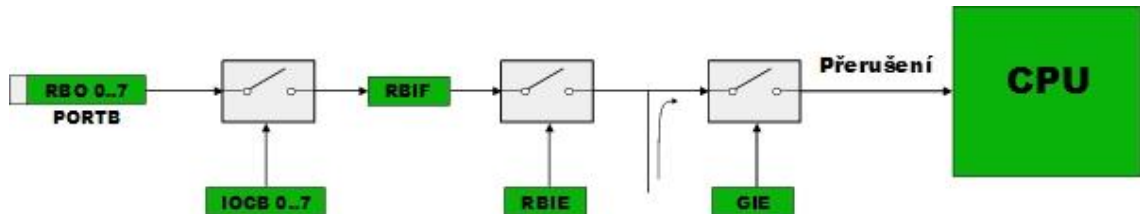
Řídicí jednotka obsahuje 6 konektorů RJ45 pro připojení externích modulů. Zapojení těchto konektorů je realizováno tak, aby každý připojený modul mohl komunikovat s řídicí jednotkou pomocí jednovodičové komunikace. Na jeden pin použitého konektoru je vyvedeno napájecí napětí 5 V a na jeden pin napájecí zem GND. U připojených modulů je tedy možné využít i centrálního napájení z řídicí jednotky. V tomto případě se ovšem musí brát v potaz maximální možný odebíraný

proud ze zdroje napájení a jeho maximální povolený výkon. Schéma zapojení je na následujícím obrázku.



Obrázek 21 Schéma zapojení konektorů pro externí moduly

Každý konektor je připojen samostatně k pinům použitého mikroprocesoru. Piny byly voleny tak, aby bylo možné obsluhovat vnější přerušení vyvolané na jakémkoli konektoru. Jak je popsáno v kapitole 1, k tomuto účelu jsou určeny piny portu B a pro povolení přerušení je nutné nastavit některé bity ve speciálních funkčních registrech.



Obrázek 22 Blokové schéma nastavení přerušení na portu B

Tyto bity mají ve svém názvu písmena IE z anglického interrupt enable. Nejprve je nutné nastavit bit GIE v registru INTCON na hodnotu logická jedna. Tím povolíme takzvané globální přerušení. Je-li tento bit nastaven na hodnotu logická nula, pak k žádnému přerušení nedojde, ať je zdroj přerušení jakýkoli. Pro povolení přerušení, které je vyvoláno změnou logického stavu na portu B, je nutné nastavit bit RBIE v registru INTCON na hodnotu logická jedna.

INTCON	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(x)	Název bitu
	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	

Obrázek 23 Znárodnění speciálního funkčního registru INTCON

Posledním registrem, který je nutné pro povolení přerušení na změnu logického stavu na portu B nastavit, je registr IOCB. Ten obsahuje osm bitů přiřazených k jednotlivým pinům portu B. Pro každý pin se zde nastavuje povolení přerušení samostatně.

IOCB	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(x)	Název bitu
	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	

Obrázek 24 Znárodnění speciálního funkčního registru IOCB

Dále jsou piny portu B spojeny s bitem nazvaným RBIF, který informuje o požadavku na přerušení bez ohledu na to, zda je přerušení povoleno či nikoli. Tento bit má ve svém názvu písmena IF z anglického interrupt flag. Po příchodu přerušení je nejprve nastaven bit IF. Není-li však odpovídající bit IOCB nastaven na hodnotu logická jedna, přerušení nenastane. V opačném případě přerušení nastane. Je-li povoleno více zdrojů přerušení najednou, pak je nutné zjistit od kterého zdroje je přerušení vyvoláno ještě před samotnou obsluhou přerušení. K tomuto slouží kontrolní bit Flag. Tento bit není při vyvolání přerušení automaticky nastavován zpět na hodnotu logická nula ale je nutné tak učinit softwarově při obsluze přerušení. Kdyby se tak nestalo, pak dojde po skončení obsluhy přerušení k dalšímu přerušení, bez toho aniž by nějaké opravdu nastalo.

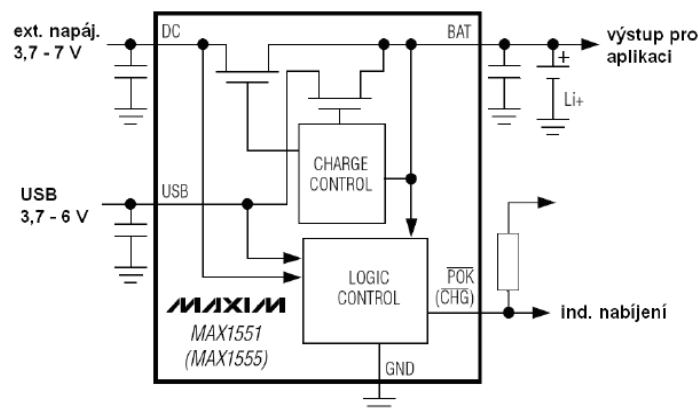
5 Dálkové ovládání

Dálkové ovládání je navrženo tak, aby jeho pomocí mohla být ovládána výše popsaná řídicí jednotka. Bezdrátový přenos je realizován pomocí modulu RFM12B. Samotný ovladač je vybaven dvanácti tlačítky. Napájení zajišťuje lithiový článek CPAE-08/2005 a je zde integrován obvod pro dobíjení akumulátoru. Vše je řízeno stejně jako v případě řídicí jednotky mikroprocesorem PIC 16F887.

5.1 Zdroj napájení

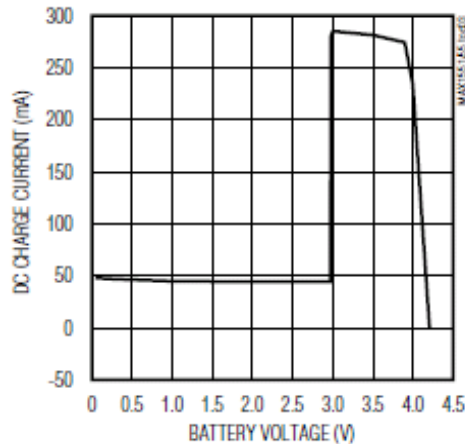
Jako zdroj napájení je v tomto případě použit lithiový článek CPAE-08/2005 o jmenovité hodnotě napětí 3,7 V s kapacitou 600 mAh. K tomuto článku je připojen obvod pro jeho nabíjení. Nabíjení je řízeno integrovaným obvodem MAX1555 od firmy Maxim.

Integrovaný obvod MAX1555 je možné napájet dvěma způsoby. Jak je vidět na následujícím obrázku, je zde vývod označený DC, ke kterému je možné připojit externí napájení v rozmezí od 3,7 V do 7 V a maximální nabíjecí proud je omezen na hodnotu 280 mA. Druhou možností je využít vývod označený USB, ke kterému je možné připojit USB kabel. Povolený rozsah napětí je zde od 3,7 V do 6 V a maximální nabíjecí proud je omezen na hodnotu 100 mA. V tomto případě je použita první možnost, tedy externí napájení připojené k vývodu DC.



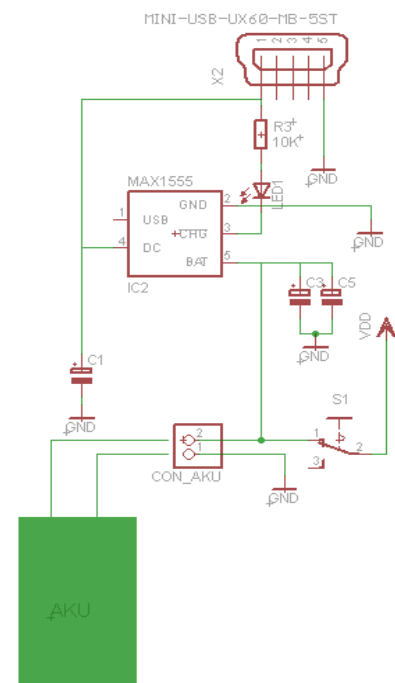
Obrázek 25 Blokové schéma obvodu Max1555 [13]

Další funkcí tohoto obvodu je snížení nabíjecího proudu na hodnotu 40 mA při vybití článku pod 3 V. Při hodnotě nad 3V je nabíjecí proud vyšší a koncová hodnota nabíjení je 4,25 V. Na obrázku 26 je tato skutečnost znázorněná v grafu závislosti nabíjecího proudu na aktuálním napětí baterie.



Obrázek 26 Závislost nabíjecího proudu na aktuálním napětí baterie [13]

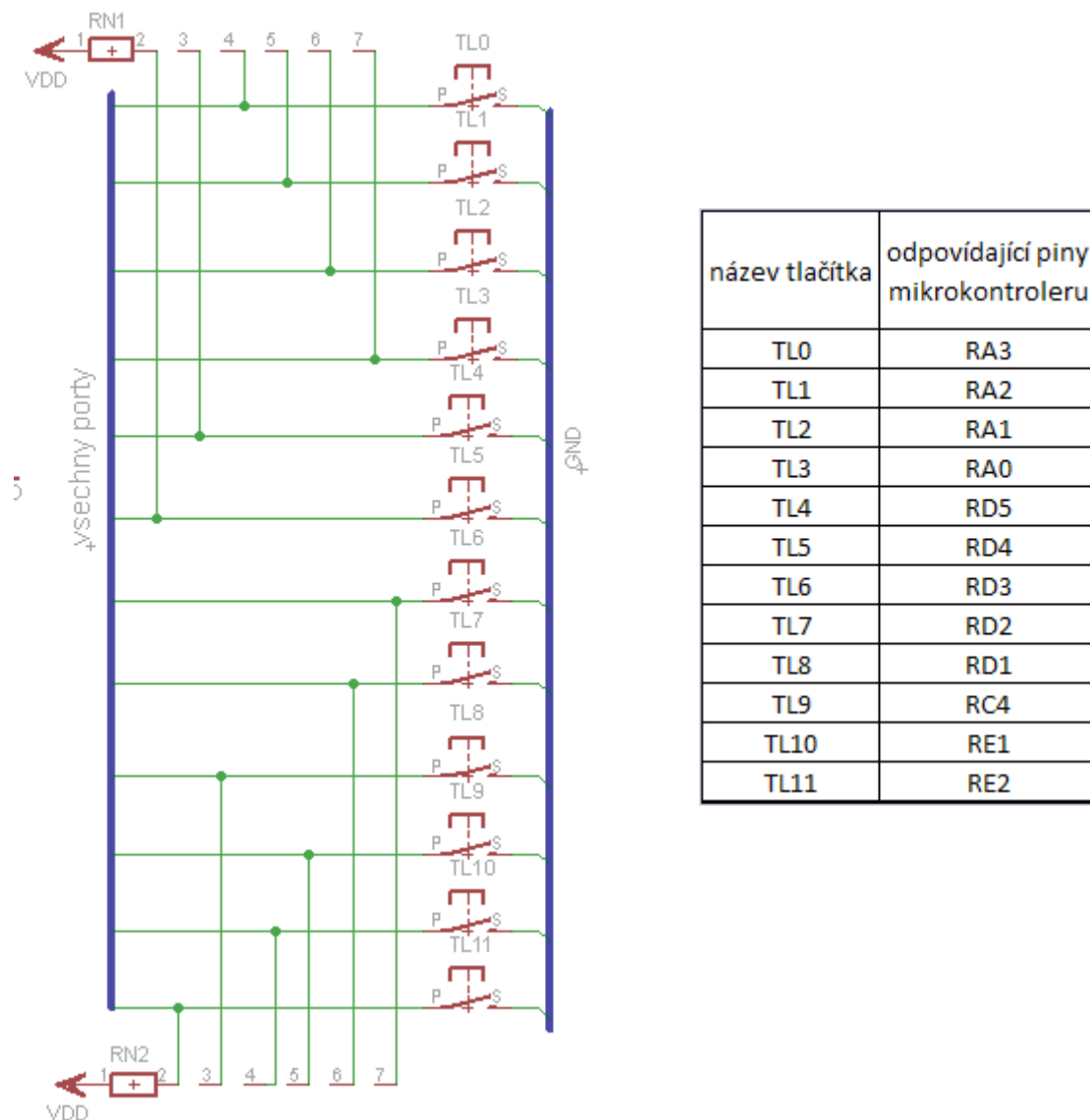
Ve stavu, kdy ani na jednom z napájecích pinů není přítomno napětí, je vnitřní spotřeba tohoto obvodu okolo 3 μ A. Z tohoto důvodu mezi tento integrovaný obvod a lithiový článek není nutné vkládat ochranou diodu. Vývod CHG slouží jako indikátor stavu nabíjení. Při nabíjení je na jeho výstupu hodnota logická nula a v opačném případě hodnota logická jedna. Další funkcí tohoto obvodu je tepelná ochrana při nabíjení z externího zdroje napájení. Při přesažení teploty 75°C při nabíjení se začne automaticky snižovat nabíjecí proud o 17 mA s každým dalším °C. K přerušení nabíjení dojde při dosažení kritické teploty, která má hodnotu 110 °C. Při nabíjení z USB tato ochrana díky nižšímu nabíjecímu proudu není aktivní.



Obrázek 27 Schéma zapojení zdroje napájení

Schéma zapojení zdroje napájení je na obrázku 27. Celý obvod napájení je možné odpojit přepínačem S1 od ostatních komponent. Odpojení je vhodné provést vždy, když programujeme mikroprocesor a také když nabíjíme akumulátor. Akumulátor se nabíjí z externího zdroje stejnosměrného napětí o jmenovité hodnotě 3,7 až 7 V.

5.2 Tlačítka



Obrázek 28 Schéma zapojení tlačítek a jejich přiřazení k pinům mikroprocesoru

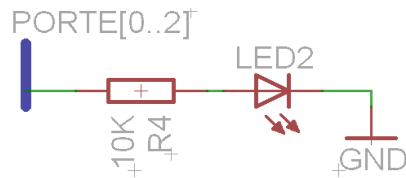
Ovladač je vybaven celkem dvanácti tlačítky, u kterých je možnost naprogramovat jejich funkci. Tlačítka jsou konstrukčně navržena pro softwarové ošetření. K tomu může být využita funkce Button. Tato funkce je v knihovně použitého vývojového prostředí. Kromě zjištění stavu kdy je tlačítko stisknuto, je určena i pro ošetření jeho zákmitu. U této funkce se zadávají čtyři parametry. Jsou to: název portu, na který je dané tlačítko připojeno, číslo pinu daného portu, na které je tlačítko připojeno, čas po kterém se má testovat stav tlačítka (udává se v milisekundách) a logická hodnota, při které má být stav tlačítka vyhodnocen jako aktivní. Pro ošetření zákmitu tlačítka je podstatný parametr čas. Je to čas, po kterém se opakovaně zjišťuje stav tlačítka. Odpovídají-li si dva stavy tlačítka ohraničené zadaným časem pak je tento stav považován za platný, tedy bez zákmitu. Tato funkce při vyhodnocení tlačítka jako aktivního vrátí hodnotu 255 a při vyhodnocení tlačítka jako neaktivního hodnotu 0. Schéma zapojení tlačítek a jejich přiřazení k pinům mikroprocesoru je na obrázku 28. Jak je vidět, tak tlačítka jsou připojena k mikroprocesoru pomocí pull-up rezistorů tak, aby při stisknutém tlačítku byla na vstupu mikroprocesoru logická nula a při nestisknutém tlačítku logická jednička.

Elektronický mechanismus tlačítek dálkového ovládání

Princip funkce tlačítek dálkového ovládání byl zachován stejný, jaký byl původně v použitém pouzdru ovladače. Jedná se o mechanické sepnutí elektronického obvodu, které se často nazývá „Vodivá guma“. Uprostřed spodní části klávesy je umístěn vodivý pruh gumy, který po stisku klávesy spojí kontakty na desce tištěného spoje (DPS). Kontakty na DPS jsou tvořeny dvěma vodivými cestami umístěnými vhodně vedle sebe tak, aby byly při stisku klávesy bezpečně spojeny. Jedna cesta nese potenciál kladného napětí a druhá má potenciál země. Návrhový mechanismus je v našem případě tvořen gumovým „kloboučkem“ jak tomu bývá například i u klávesnic Mitsumi.

5.3 Signalizační LED diody

Ovladač obsahuje LED diodu, která slouží jako informativní prvek o aktuálním stavu dálkového ovladače. LED dioda je připojena na port E k pinu RE0. V příložených programech je určena k vizualizaci stavu, kdy ovladač přijme data od řídicí jednotky. Schéma jejího zapojení je na následujícím obrázku.



Obrázek 29 Schéma signalizační LED diody

5.4 Programovací konektor

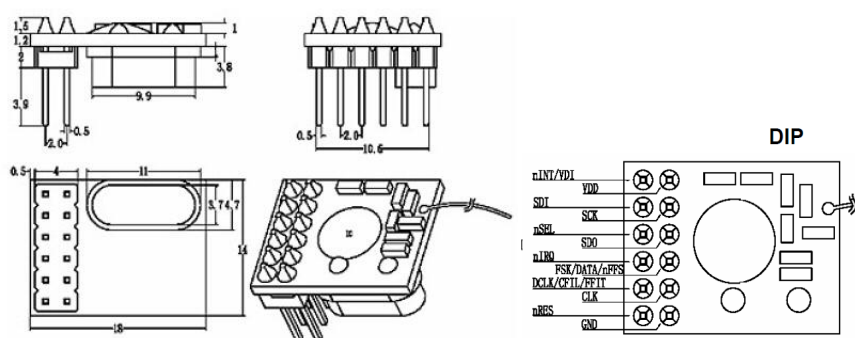
Poslední komponentou dálkového ovladače je programovací konektor. Díky tomu je dálkový ovladač velmi flexibilní, co se týče jeho funkce. Přímo v něm mohou být naprogramovány různé funkcionality, jako je například ošetření některých klávesových kombinací, vícenásobného zmáčknutí téhož tlačítka a podobně. Dále můžeme volit formát odesílaných dat podle aktuálních požadavků řídicí jednotky. Schéma zapojení a popis jednotlivých pinů je uveden v kapitole 1, kde je popsán i proces programování použitého mikroprocesoru PIC16F887.

6 Komunikace mezi řídicí jednotkou a dálkovým ovladačem

Kromě uvedených komponent obsahuje jak řídicí jednotka, tak dálkový ovladač ještě modul, který zajišťuje jejich vzájemnou komunikaci. Jedná se o VF modul RFM12B od firmy HOPE MICROELEKTRONICS. Jeho základní popis lze nalézt v kapitole 2.

6.1 Zapojení modulu

Modul RFM12B má své piny vyvedeny na lištu 2x6 pinů, která má rozteč 2 mm. Na tyto parametry je potřeba pamatovat při návrhu DPS a vytvořit si vlastní součástku s požadovanými rozměry pouzdra. Rozměry pouzdra použitého modulu jsou na následujícím obrázku.



Obrázek 30 Rozměry pouzdra použitého modulu [1]

Každý pin tohoto modulu má svoji funkci. Na následujícím obrázku je uveden jejich přehled.

nINT/VDI	Interrupt input (active low)/Valid data indicator
VDD	Positive power supply
SDI	SPI data input
SCK	SPI clock input
nSEL	Chip select (active low)
SDO	Serial data output with bus hold
nIRQ	Interrupts request output (active low)
FSK/DATA/nFFS	Transmit FSK data input/ Received data output/FIFO select
CLK/CFIL/FFIT	Clock output/ external filter capacitor/FIFO interrupts when FIFO level set to 1, FIFO empty interruption can be achieved
CLK	Clock output for external microcontroller
nRES	Reset output (active low)
GND	Power ground

Obrázek 31 Popis pinů modulu RFM12B [1]

Každý pin je připojen přímo na pin mikroprocesoru. Pin s názvem nIRQ je zdrojem signálu pro přerušení. Aby bylo možné toto přerušení v daném mikroprocesoru obsloužit, pak musí být tento pin připojen na jeden pin portu B mikroprocesoru, kde je možné toto přerušení obsloužit. V tomto případě je připojen na pin RBO. Připojení tohoto modulu k pinům mikroprocesoru se liší v zapojení pro dálkový ovladač od zapojení použitého pro řídicí jednotku. Jejich zapojení lze nalézt na následujícím obrázku.

pin modulu RFM12B	řídicí jednotka	dálkový ovladač
nRES	c6	c6
Ffit	c7	c7
FSK	c1	c1
nIRQ	b0	b0
SDO	c4	b2
nSEL	c2	c2
SCK	c3	b1
SDI	c5	c5
nINT	c0	c0

Obrázek 32 Zapojení pinů modulu RFM12B k pinům mikroprocesoru

Napájecí napětí tohoto modulu je v rozmezí od 2,2 do 3,8 V. U ovladače je využito napájení z lithiového akumulátoru. V zapojení řídicí jednotky je použito stabilizované napětí o hodnotě 3,3 V od stabilizátoru napětí LD117AD3,3.

6.2 Nastavení modulu RFM12B

Pro komunikaci mezi mikroprocesorem a modulem RFM12B je použit protokol SPI, který je podporován jak použitým mikroprocesorem PIC16F887 tak i modulem RFM12B. Modul RFM12B však vyžaduje přenos šestnáctibitového slova a plně duplexní přenos. Tedy vysílání i přijímání dat zároveň. To však nepodporuje modul pro sériovou komunikaci MSS1, který je integrovaný v mikroprocesoru PIC16F887. Tento modul podporuje odesílání osmi bitových slov a poloduplexní přenos, tedy přenos vždy jen v jednom směru. Z tohoto důvodu bylo potřeba vytvořit vlastní softwarovou funkci SPI.

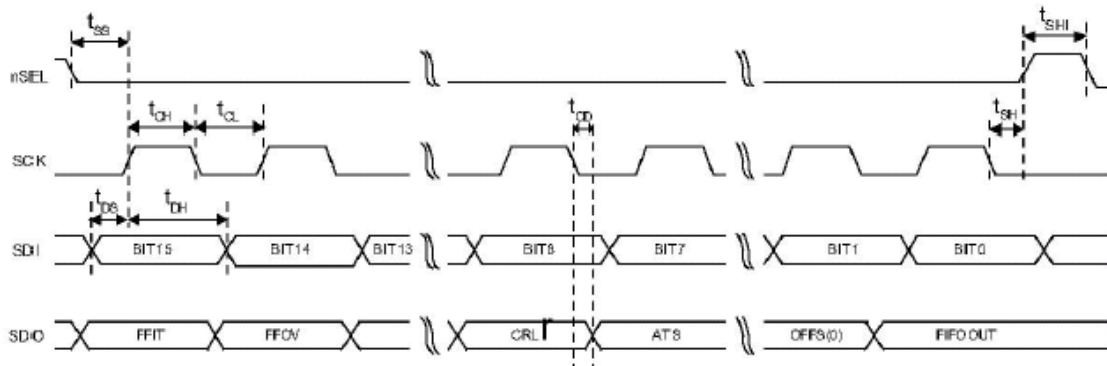
Realizace vlastní SPI

Vlastní funkce realizující SPI komunikaci se jmenuje `SPI_16` a přebírá parametr typu bezznaménkový integer nazvaný „`spi_data_out`“. V této proměnné je uloženo šestnáctibitové slovo, které chceme pomocí této funkce přenést. Následuje výpis kódu dané funkce.

```
unsigned int SPI_16(unsigned int spi_data_out){
unsigned int spi_data_in = 0;
    unsigned short N;
    CS = 0;
    CLK = 0;
    for(N=0;N<16;N++){
        if((spi_data_out & 0x8000)==0){SDO=0;}
        else{SDO=1;}
        spi_data_out = spi_data_out << 1;
        spi_data_in = spi_data_in << 1;
        if(SDI!=0){spi_data_in++;}
        CLK = 1;
        delay_us(5);
        CLK = 0;
    }
    CS = 1;
    return spi_data_in ;
}
```

Tato funkce začíná nastavením proměnné `CS` na hodnotu logická nula, čímž umožníme komunikaci s připojeným modulem. Proměnou `CLK` nastavíme na hodnotu logická nula. Poté do cyklu `for` vložíme funkci pro samotné zapisování a čtení dat. Počet opakování cyklu volíme tak, abychom přenesli požadovanou šířku slova. V našem případě tedy šestnáct opakování. Poté čteme hodnotu jednotlivých bitů vysílaného slova a bit `SDO` nastavujeme podle aktuální hodnoty. Postupné čtení bitů je realizováno rotací celého registru, kde máme uložena výstupní data, o jeden bit vlevo. Obdobně čteme hodnotu jednotlivých bitů přijímaného slova a nastavujeme dle aktuální hodnoty těchto bitů hodnotu proměnné `spi_data_in`. Opět je zde využito rotace o jeden bit vlevo, tentokrát však pro vstupní data. Po zjištění těchto hodnot nastavíme signál `CLK` na hodnotu logická jedna. Tím zapíšeme požadovaná data i v radiči SPI, který je implementovaný v modulu `RFM12B`. Po následném uplynutí času pět mikrosekund, který zajistí vzorkování ustálených dat, nastavíme signál `CLK` opět na

hodnotu logická nula. Po proběhnutí všech šestnácti opakování nastavíme proměnnou CS na hodnotu logická jedna, čímž ukončíme komunikaci s VF modulem a do programu vrátíme proměnnou obsahující přijatá data. Časový diagram pro zápis do VF modulu, respektive čtení z VF modulu, je na následujícím obrázku.



Obrázek 33 VF modul - časový diagram pro zápis a čtení. [2]

Inicializace

Aby bylo možné modul používat, je nutné jej nejprve inicializovat. To je realizováno zapsáním konfiguračních příkazů do paměti modulu. Jejich seznam je uveden v technické dokumentaci tohoto modulu [2]. Při inicializaci je možné nastavovat vysílací výkon, frekvenční deviace, bitovou rychlost, podobu synchronizačního bytu, wake-up časovač, detekci nízké napájecí úrovně a další funkcionality.

Data jsou odesílána pomocí výše popsané funkce SPI_16, jsou zapisována v hexadecimálním tvaru a mají následující tvar:

- tři synchronizační byty (0xAAAAAA)
- adresní byty (0x2D, 0xD4)
- datové byty
- kontrolní součet

V oficiálním datasheetu k tomuto modulu [1] jsou uvedeny základní hodnoty pro inicializaci. Níže jsou vypsány hodnoty, které jsou použity pro naše nastavení. Oproti

hodnotám uváděným v oficiální dokumentaci se v některých případech liší tak, aby odpovídali našim požadavkům.

Použité hodnoty inicializace

```
(0x80e7); // El,Ef,868Mhz,12pf
(0x82d9); // er,ebb,!Et,Es, Ex,!eb,!Ew,Dc
(0xa67c); // 868,3MHz(F=60)
(0xc647); // 4.8kbps(R=71)
(0x94a0); // Vdi,fast,134kHz,0db,-103dbm
(0xc2ac); // Al,!ml,Dig,Dqd4
(0xca81); // Fifo8,Sync,!ff,Dr
(0xc483); // @pwr,NoRstric,!st,!fi,Oe,En
(0x9854); // 90 kHz , power - 0 dB
(0xe000); // not used
(0xc800); // not used
(0xc000); // 1 MHz , 2.2V
```

Za lomítky jsou uvedeny názvy jednotlivých bitů, které jsou daným příkazem nastaveny buď na hodnotu logická jedna (název bez vykřičníku), nebo na hodnotu logická nula (název s vykřičníkem). Názvy jednotlivých bitů jsou převzaty z oficiální dokumentace [1].

Režimy transceiveru

Aby byla umožněna vzájemná komunikace těchto dvou transceiverů, jeden je součástí řídicí jednotky a druhý dálkového ovládání, pak musí být vždy jeden nastaven v režimu vysílače a druhý v režimu přijímače. Toto nastavení však není trvalé, ale je měněno podle toho, zda vysílá transceiver umístěný v dálkovém ovladači či transceiver umístěný v řídicí jednotce.

Přepínání mezi vysíláním a přijímáním je realizováno pomocí příkazu, který lze v oficiální dokumentaci nalézt pod názvem Power management command. Je to příkaz, který má při hexadecimálním zápisu podobu 0x82yy. Písmena yy v tomto případě znamenají vlastní nastavení, které požadujeme. Na následujícím obrázku je zobrazeno možné nastavení tohoto příkazu. Tento příkaz se tedy liší, chceme-li transceiver použít jako vysílač či jako přijímač. Pro vysílání je v našem případě použita hodnota 0x8238

a pro příjem hodnota 0x82c8. V programátorské příručce [2] se v tomto příkazu vyskytuje chyba, díky které není přijímač správně nastaven a komunikace pak nemůže být navázána.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	0	0	0	0	0	1	0	er	ebb	et	es	ex	eb	ew	dc

- er: Enable receiver
- ebb: Enable base band block
- et: Enable transmitter
- es: Enable synthesizer
- ex: Enable crystal oscillator
- eb: Enable low battery detector
- ew: Enable wake-up timer
- dc: Disable clock output of CLK pin

Obrázek 34 Power management command [2]

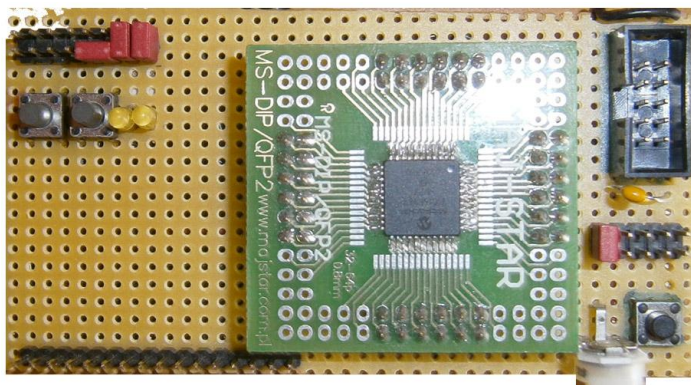
Tento příkaz je tedy použit při každém volání funkce pro odesílání dat, která se jmenuje Send_rfm12 i při volání funkce pro příjem dat nesoucí název Receive_rfm12. Obě tyto funkce lze nalézt ve zdrojových souborech na přiloženém CD.

Vysílání dat

Pro samotné vysílání dat je v transceiveru implementován registr nesoucí název Transmitter Register Write. Požadovaná data je tedy potřeba do tohoto registru zapsat. Registr má šířku slova 8 bitů, tedy jeden byte a jeho adresa je 0xB8xx, přičemž xx je právě vysílaný datový byte. Jako potvrzení o odvysílání dat slouží výstup nIRQ. Po odvysílání dat je na něm nastavena hodnota logická nula.

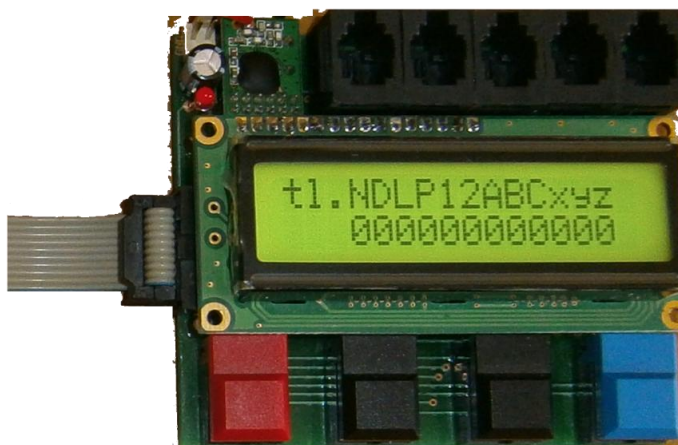
7 Návrh DPS

Návrh DPS a samotného zapojení probíhal v několika krocích. Nejprve byly navrženy a realizovány zapojení na univerzálních plošných spojích, na kterých byla zkoušena funkce jak samotného mikroprocesoru PIC16F887 tak použitých periférií. Na následujícím obrázku je fotografie jedné takové univerzální desky.



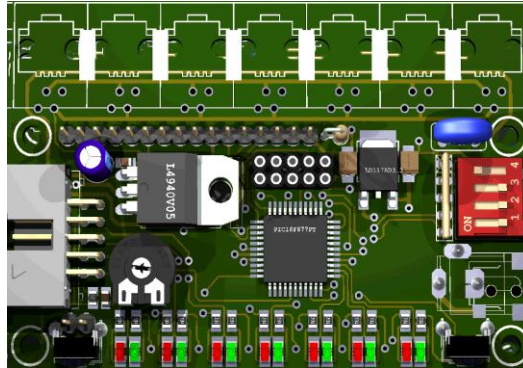
Obrázek 35 Univerzální plošný spoj

Při tom byla v návrhovém prostředí EAGLE vytvářena schémata zapojení pro různé možnosti zapojení jednotlivých komponent. Pro konečnou verzi řídicí jednotky byly vytvořeny návrhy DPS s vhodným rozmístěním součástek a ty byly poté vyrobeny a testovány. Na těchto deskách byla také vytvořena konečná koncepce navrženého softwaru. Fotografie této verze řídicí jednotky je na následujícím obrázku.



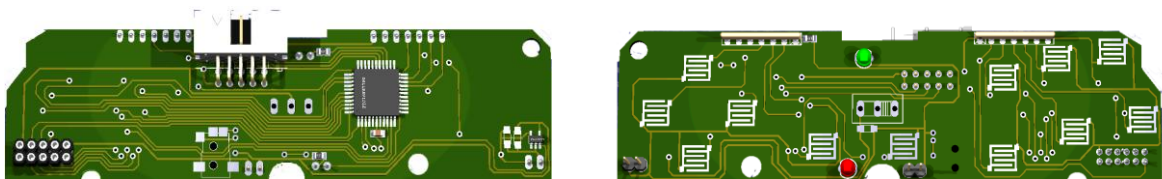
Obrázek 36 Osazená DPS řídicí jednotky

Pro řídicí jednotku bylo dále navrženo zapojení, kde se jednotlivé periferie připojují pomocí komunikačního protokolu I2C. Tento návrh se ale ukázal jako nevhodný. Schéma zapojení a návrh DPS pro tuto možnost je však přiložen na CD se zdrojovými soubory a je tak možné tento návrh dále rozvinout. Náhled na tuto DPS vygenerovaný programem Eagle je na následujícím obrázku.



Obrázek 37 DPS řídicí jednotky s komunikací SPI pro přídatné moduly

Jako poslední bylo navrženo schéma DPS pro samotný dálkový ovladač. Návrh desky byl uzpůsoben použitému pouzdru pro tento ovladač. Na následujícím obrázku je zobrazen dálkový ovladač a deska plošného spoje.



Obrázek 38 Náhled na DPS dálkového ovladače.



Obrázek 39 Dálkový ovladač

Závěr

Úkolem bylo navrhnout a realizovat univerzální programovatelnou řídicí jednotku, ke které se mají připojovat moduly s různou funkcionalitou a dále programovatelný dálkový ovladač pro tuto jednotku. K realizaci bylo potřeba získat teoretický přehled o bezdrátové komunikaci, komunikačních protokolech a o vlastnostech mikroprocesoru PIC16F887, který tvoří jádro obou zařízení.

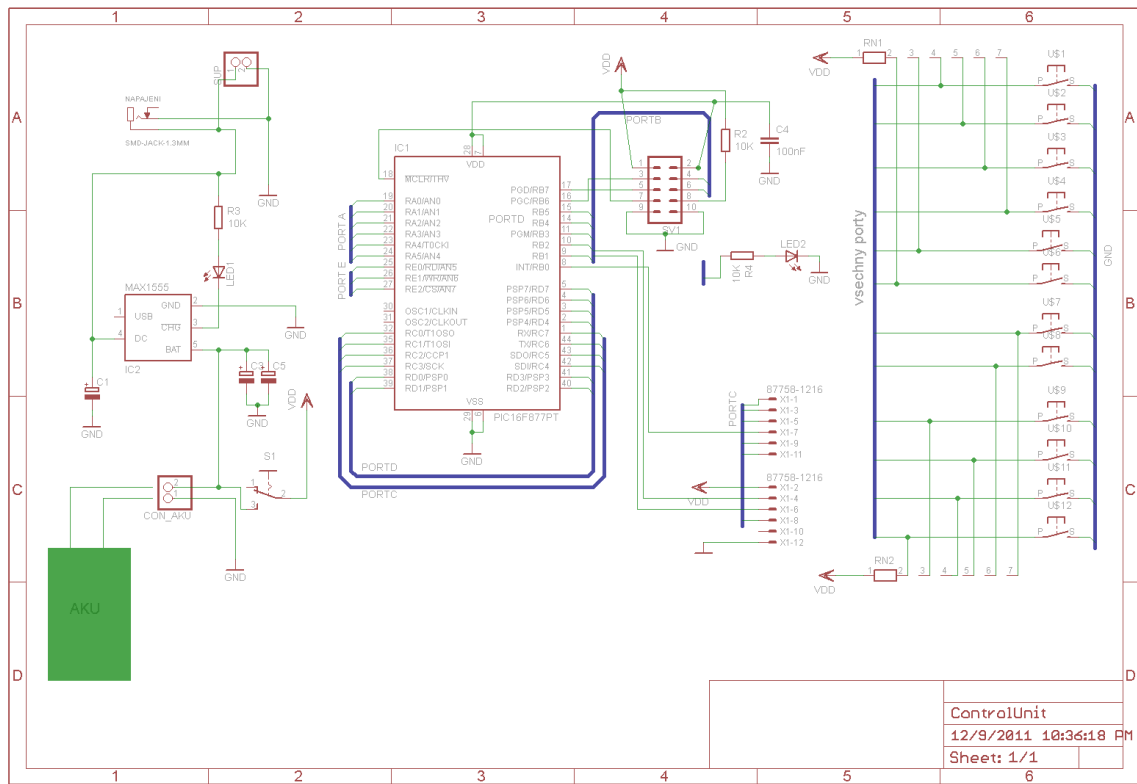
Programovatelnost je zajištěna tím, že je jak k řídicí jednotce, tak k dálkovému ovladači možno připojit programátor PicFlash programmer od firmy MikroElektronika. Jako programovací jazyk je použit jazyk C. K jednotce je možné připojit další moduly, které mohou s jednotkou komunikovat pomocí jakékoli jednovodičové komunikace. Moduly je možné připojit k jednotce pomocí konektoru RJ45 a je možné je i napájet ze zdroje energie řídicí jednotky. Vzájemná komunikace mezi řídicí jednotkou a dálkovým ovladačem je realizována pomocí vysokofrekvenčního modulu RFM12B. Ten umožňuje komunikaci na frekvenci 868 MHz, jak bylo požadováno v zadání. Přenos je plně duplexní a umožňuje tedy nejen předávání řídicích příkazů z dálkového ovladače do řídicí jednotky ale i přenos informací z řídicí jednotky do dálkového ovladače. Tím získává celý systém možnost automatického přepínání mezi funkcionalitou jednotlivých tlačítek dálkového ovladače podle aktuálního stavu řídicí jednotky. Řídicí jednotka je vybavena dvouřádkovým LCD displejem, který zajišťuje pohodlnou interakci mezi uživatelem a daným systémem. Dále řídicí jednotka obsahuje čtyři tlačítka ošetřená proti zákmitu, která je možné využít pro řízení jednotky bez nutnosti používat dálkový ovladač. Řídicí jednotka je navržena tak, aby ji bylo možné napájet externím zdrojem napájení nebo z programátoru. To umožňuje pohodlnější ladění celé jednotky při nahrávání nového softwaru a umožňuje to i využívání ladění celé jednotky pomocí mikroICD (In Circuit Debugger). Řídicí jednotka má dále vyvedeny nepoužité piny řídicího mikroprocesoru PIC16F887 tak, aby k nim mohli být připojeny další obvody a tím využity i speciální funkce jednotlivých pinů. Navržený dálkový ovladač obsahuje dvanáct tlačítek, je napájen lithiovým článkem o jmenovitém napětí 3,7 V a oproti požadavkům zadání v sobě má integrován nabíječku pro uvedený článek. Nabíjení je řízeno obvodem Max1555.

Návrhy DPS a programy s ovladači pro navrženou jednotku jsou na přiloženém CD.

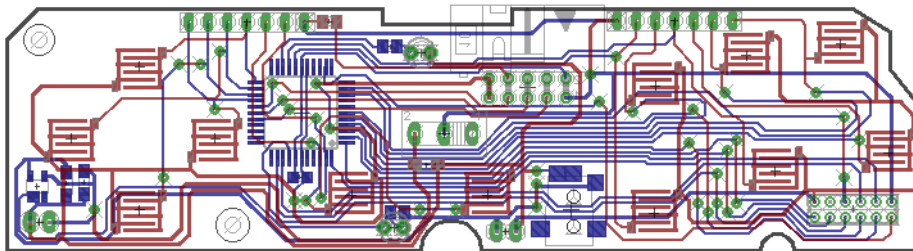
Literatura

- [1] **HopeRF.** *RFM12B datasheet* [online] 6. 10. 2011
URL:<<http://www.hoperf.com/upload/rf/RFM12B.pdf> >
- [2] **HopeRF.** *RF12B programming guide* [online] 6. 10. 2011
URL:<http://www.hoperf.com/upload/rf/RF12B_code.pdf>
- [3] **HopeRF.** *RF12B universal ISM band FSK transceiver* [online] 6. 10. 2011
URL:<<http://www.hoperf.com/upload/rf/RF12B.pdf>>
- [4] **Hrbáček, Jiří.** *Komunikace mikrokontroléru s okolím – 1. díl*, Praha: BEN – technická literatura, 1999. ISBN 80-86056-42-2
- [5] **Verle, Milan.** *PIC Microcontrollers - Programming in C*, mikroElektronika, 2009
ISBN13 978-86-84417-17-8
- [6] **Microchip.** *PIC16F887 Data Sheet* [online]. 3. 10. 2011
URL:<<http://ww1.microchip.com/downloads/en/DeviceDoc/41291F.pdf>>
- [7] **Herout, Pavel.** *Učebnice jazyka C, 1.díl*, Koop, 2002. ISBN 80-85828-21-9
- [8] **Herout, Pavel.** *Učebnice jazyka C, 2.díl*, Koop, 2002. ISBN 80-85828-50-2
- [9] **Plíva, Z.** *Eagle prakticky*, Praha: BEN – technická literatura, 2007.
ISBN 978-80-7300-227-5
- [10] **GM Elektronik.** *Elektronické součástky* [online]. 10. 10. 2011
URL:< <http://www.gme.cz/elektronicke-soucastky-material-a-zarizeni/> >
- [11] **Transfer multisort elektronik.** *Katalog on-line* [online]. 10. 10. 2011
URL:< <http://www.tme.eu/cz/katalog/#index> >
- [12] **Hitachi.** *HD44780U (CLD-II)* [online]. 10. 10. 2011
URL:<<http://pdf1.alldatasheet.com/datasheet-pdf/view/63673/HITACHI/HD44780.html>>
- [13] **Maxim.** *MAX1555 - Maxim Integrated Products* [online]. 10. 10. 2011
URL:<<http://pdf1.alldatasheet.com/datasheet-pdf/view/97001/MAXIM/MAX1555.html>>
- [14] **MikroElektronika.** *PICflash – User manual Programmer* [online]. 10. 10. 2011
URL:<http://www.mikroe.com/eng/downloads/get/10/picflashwicd_manual_v100.pdf>

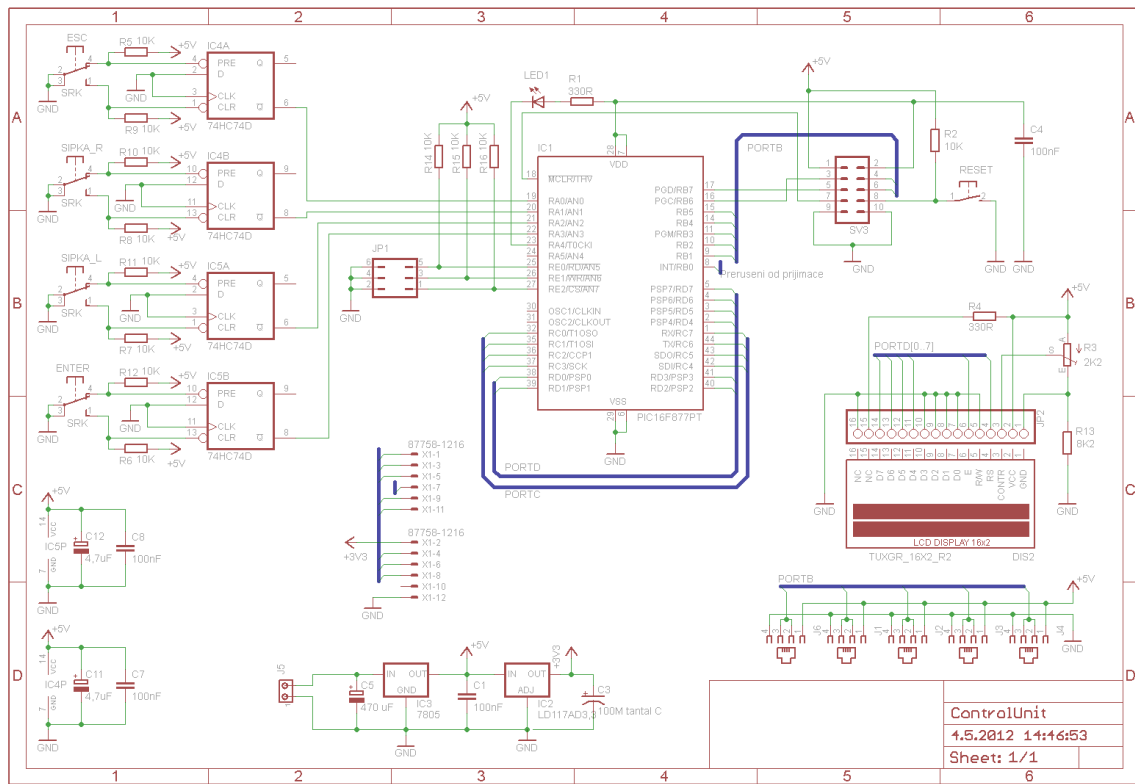
Příloha A – schéma zapojení dálkového ovladače



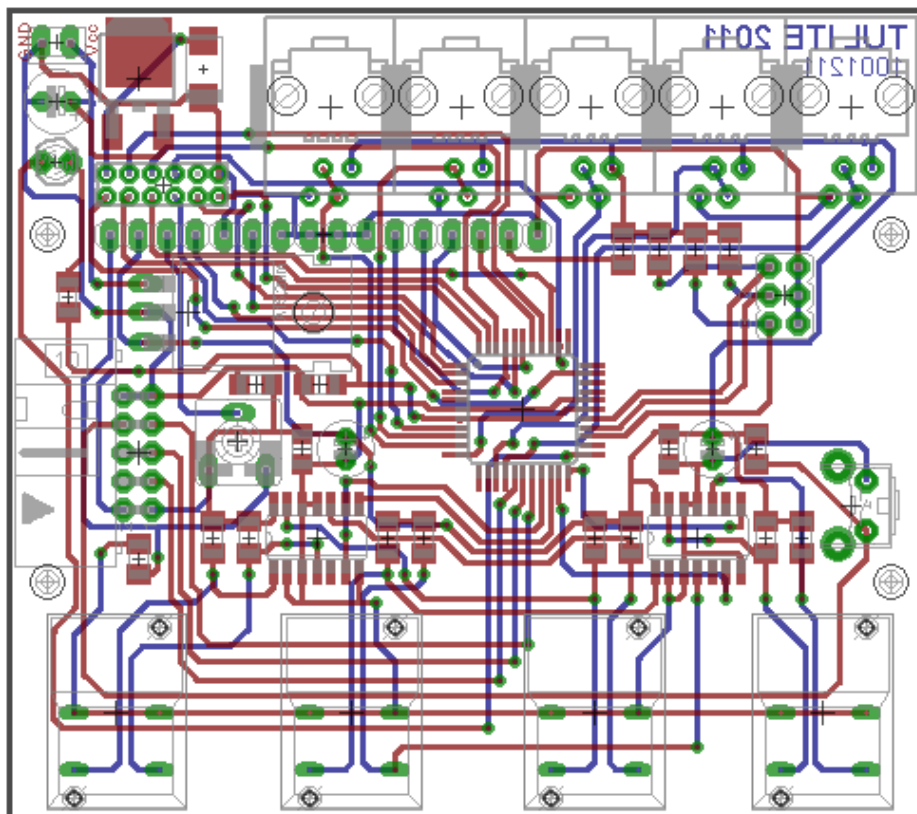
Příloha B – DPS dálkového ovladače



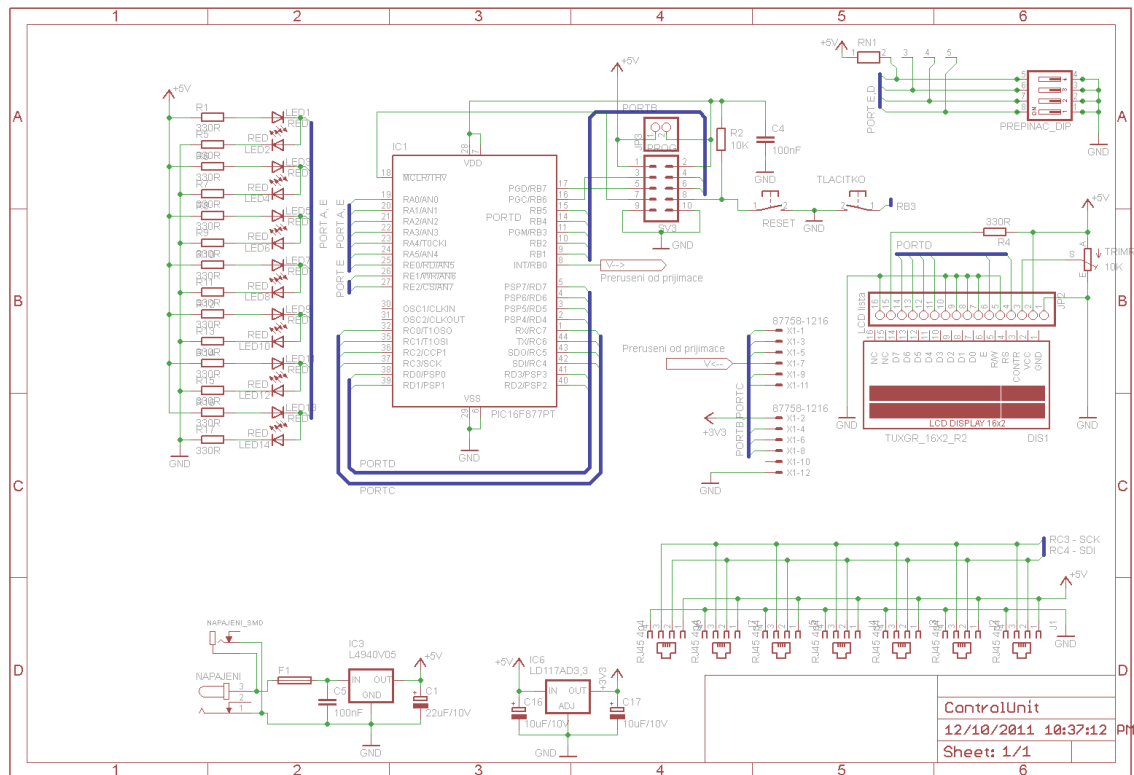
Příloha C – schéma zapojení řídicí jednotky



Příloha D – DPS řídicí jednotky



Příloha E – schéma zapojení řídicí jednotky s komunikací SPI



Příloha F – DPS řídicí jednotky s komunikací SPI

