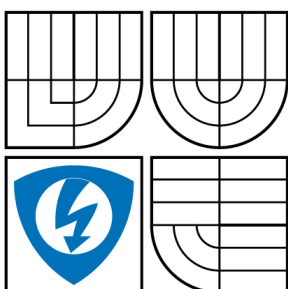


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ  
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF RADIO ELECTRONICS

# SYSTÉM PRO DIGITÁLNÍ BEZDRÁTOVOU KOMUNIKACI NA KRÁTKOU VZDÁLENOST

SYSTEM FOR DIGITAL SHORT-RANGE WIRELESS COMMUNICATION

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

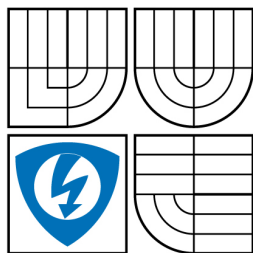
AUTOR PRÁCE  
AUTHOR

MIROSLAV RÉBL

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. PETR TOŠOVSKÝ

BRNO 2009



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav radioelektroniky

# Bakalářská práce

bakalářský studijní obor  
**Elektronika a sdělovací technika**

**Student:** Miroslav Rébl

**ID:** 72911

**Ročník:** 3

**Akademický rok:** 2008/2009

## NÁZEV TÉMATU:

**Systém pro digitální bezdrátovou komunikaci na krátkou vzdálenost**

## POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s možnostmi bezdrátové komunikace na krátké vzdálenosti pro účely předávání krátkých příkazů. Vyberte vhodné prostředky pro realizaci komunikačního systému a také vyberte metodu zabezpečení přenášených dat.

Realizujte komunikační systém s jedním vysílačem a více přijímači. Změřte hlavní parametry systému a pokuste se dosáhnout minimální spotřeby všech částí systému.

## DOPORUČENÁ LITERATURA:

[1] MARŠÁLEK, R. Teorie radiové komunikace. Elektronické skriptum. Brno: FEKT VUT v Brně, 2005.

**Termín zadání:** 9.2.2009

**Termín odevzdání:** 5.6.2009

**Vedoucí práce:** Ing. Petr Tošovský

**prof. Dr. Ing. Zbyněk Raida**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

## **ABSTRAKT**

Tato práce se zabývá návrhem komunikačního systému bezdrátového přenosu dat na krátkou vzdálenost. Rozebírá možné varianty komunikace mezi uzly sítě se zaměřením na specifický typ bezdrátové sítě tvořené jen jedním typem komunikujícího zařízení. Popisuje možnosti komunikace v systému, způsob ochrany proti chybovému příjmu dat. Následně uvádí vlastní realizaci části digitální bezdrátové sítě s vybranými funkcemi.

## **KLÍČOVÁ SLOVA**

Bezdrátový, přenos, komunikace, struktura, síť, topologie, uzel, MESH

## **ABSTRACT**

This project includes the design of the System for digital short-range wireless communication. The methodology of the communication between net only one device type based system is proposed. The thesis describes system communication rules and used ways for protection of a correct transmission. Furthermore, the project describes realization of the digital wireless communication with selected functions.

## **KEYWORDS**

Wireless, communication, structure, net, topology, device, MESH

Rébl, M. *Systém pro digitální bezdrátovou komunikaci na krátkou vzdálenost*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, 2009. 33 s., 5 s. příloh. Bakalářská práce. Vedoucí práce: Ing. Petr Tošovský.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma *Systém pro digitální bezdrátovou komunikaci na krátkou vzdálenost* jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce ing. Petru Tošovskému za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne .....

.....

(podpis autora)

# OBSAH

|                                             |            |
|---------------------------------------------|------------|
| <b>Seznam obrázků</b>                       | <b>vi</b>  |
| <b>Seznam tabulek</b>                       | <b>vii</b> |
| <b>Úvod</b>                                 | <b>1</b>   |
| <b>1 Návrh komunikace</b>                   | <b>2</b>   |
| 1.1 Obecné požadavky .....                  | 2          |
| 1.2 Komunikační moduly .....                | 2          |
| 1.2.1 Určení nosného kmitočtu .....         | 2          |
| 1.2.2 Volba specifických modulů .....       | 3          |
| 1.2.3 Vlastnosti modulů RFM0X.....          | 3          |
| 1.3 Struktura sítě .....                    | 4          |
| 1.3.1 Popis zvolené struktury.....          | 4          |
| 1.3.2 Blokové schéma uzlu .....             | 5          |
| 1.4 Možnosti komunikace .....               | 5          |
| 1.4.1 Vysílání uzel – uzel (unicast).....   | 5          |
| 1.4.2 Vysílání všesměrové (broadcast) ..... | 6          |
| 1.4.3 Mapování nové trasy .....             | 6          |
| 1.5 Kontrola chybovosti.....                | 6          |
| 1.5.1 Způsob ochrany přenosu .....          | 6          |
| 1.5.2 Popis použitého kódu.....             | 7          |
| 1.5.3 Kanálové kódování dat .....           | 7          |
| 1.6 Struktura komunikačního rámce .....     | 7          |
| 1.6.1 Přenosové rámce .....                 | 8          |
| <b>2 Realizace části komunikace</b>         | <b>11</b>  |
| 2.1 Obvodový návrh zařízení.....            | 11         |
| 2.1.1 Mikrokontrolér.....                   | 11         |
| 2.1.2 Napájení .....                        | 12         |
| 2.1.3 Komunikační rozhraní .....            | 12         |
| 2.1.4 Ovládací prvky a indikace + LCD ..... | 13         |

|          |                                                |           |
|----------|------------------------------------------------|-----------|
| 2.1.5    | Návrh DPS .....                                | 14        |
| 2.2      | Programové vybavení (firmware).....            | 14        |
| 2.2.1    | Programátor .....                              | 14        |
| 2.2.2    | Vývojové prostředí .....                       | 15        |
| 2.2.3    | Vlastní program pro realizaci komunikace ..... | 15        |
| <b>3</b> | <b>Naměřené parametry</b>                      | <b>21</b> |
| 3.1      | Rádiový dosah.....                             | 21        |
| 3.2      | Odběr .....                                    | 21        |
| 3.3      | Rychlost přenosu.....                          | 21        |
| <b>4</b> | <b>Závěr</b>                                   | <b>22</b> |
|          | <b>Literatura</b>                              | <b>23</b> |
|          | <b>Seznam symbolů, veličin a zkratk</b>        | <b>24</b> |
|          | <b>Seznam příloh</b>                           | <b>25</b> |

# SEZNAM OBRÁZKŮ

|           |                                           |    |
|-----------|-------------------------------------------|----|
| Obr. 1.1: | Struktura sítě (U-uzel).....              | 4  |
| Obr. 1.2: | Blokové schéma uzlu .....                 | 5  |
| Obr. 2.1: | Konfigurace pinů ATmega16 .....           | 11 |
| Obr. 2.2: | Zapojení napájecího konektoru.....        | 12 |
| Obr. 2.3: | Zapojení pinů JTAG konektoru .....        | 12 |
| Obr. 2.4: | Popis instrukcí LCD.....                  | 14 |
| Obr. 2.5: | Blokové schéma zapojení programátoru..... | 15 |
| Obr. 2.6: | Kontrola zvoleného mikrokontroléru.....   | 15 |
| Obr. 2.7: | Vývojový diagram .....                    | 16 |
| Obr. 2.8: | Časový diagram modulu .....               | 19 |

# SEZNAM TABULEK

|                                                              |    |
|--------------------------------------------------------------|----|
| Tab. 2.1: Parametry zvoleného ATmega16.....                  | 12 |
| Tab. 2.2: Označení vývodů programovacího rozhraní JTAG ..... | 12 |
| Tab. 2.3: Popis vývodů LCD.....                              | 13 |

# ÚVOD

Bezdrátová komunikace spočívá ve spojení dvou a více subjektů jinak, než mechanicky (tj. metalickým nebo optickým kabelem). Využívá se tak na místech, kde by bylo velmi obtížné až nemožné realizovat spojení kabelem nebo z hlediska prosté pohodlnosti řešení. Je tak snadno možné předejít stavebním úpravám budov a s tím spojenými nepříjemnostmi.

Mezi realizovatelné možnosti patří i komunikace prostřednictvím rádiových vln, infračerveného světla (IrDA), magnetické indukce či zvukových vln. Z těchto způsobů jsou v praxi rozšířenější pouze použití infračerveného světla a rádiových vln.

Tato práce se věnuje komunikaci pomocí vln rádiových. Nepotýkají se s takovými problémy, jako ostatní metody ohledně překážek v prostoru, které limitují využití sítě spíše na dvoubodovou komunikaci mezi infračervenými porty elektronických zařízení na velmi krátkou vzdálenost.

Bezdrátové sítě nabízejí v principu podobné služby jako sítě drátové avšak se zvýšenou flexibilitou. Je možné zapojit do nich servery a jejich klienty, ale také je možné v nich vytvářet spojení peer-to-peer. Z hlediska funkčnosti a výsledku jsou, odhlédneme-li od dosahovaných přenosových rychlostí, ekvivalentní k sítím drátovým.

# 1 NÁVRH KOMUNIKACE

Úmyslem je vytvořit komunikaci „inteligentního“ charakteru mezi několika zařízeními. Možnost vysílat data určené všem účastníkům v síti nebo pouze pro určeného příjemce. Schopnost zkontrolovat přijatá data z hlediska chybovosti a podle toho situaci následně řešit, popřípadě využít retranslace prostřednictvím jiného účastníka sítě.

## 1.1 Obecné požadavky

- V síti je vyžadována vzájemná komunikace všech uzlů
- Polo-duplexní komunikace
- Přenášenými informacemi by měly být krátké příkazy a data
- Předpokládaná vzdálenost uzlů - do 50 m
- Komunikace by měla být opatřena synchronizací, ošetřeným zahájením vysílání a kontrolou chybovosti
- Minimální spotřeba všech částí systému
- Přiměřená cena

## 1.2 Komunikační moduly

### 1.2.1 Určení nosného kmitočtu

Daná síť by měla být provozována ve volném ISM pásmu. Pod pojmem volné pásmo se rozumí pásmo kmitočtů, ve kterém je povolen radiový provoz bez licenčních poplatků držitelům homologovaných zařízení, přičemž jejich počet není předem omezen. Tito provozovatelé pak mohou sdílet při provozu celé vyčleněné pásmo, ovšem bez nároku na ochranu proti rušení.

V radiofrekvenčním spektru je takových volných pásem vyčleněno více, například pásma 433 MHz, 868 MHz a 2,4 GHz.

Volba kmitočtu byla provedena v souvislosti se zvolením specifických modulů k realizaci uzlů sítě, kde rozhodovalo, jestli se námi zvolený přijímač/vysílač v daném frekvenčním pásmu vyrábí. Dalším kritériem volby je zahuštěnost daného pásma. Příkladně 2,4 GHz pásmo je v nynější době poměrně dosti zahlceno ať už různými systémy podobného účelu, jako je navrhovaný nebo je možnost kolize s velmi využívaným bluetooth standardem. Jako možnost rušení lze uvést i mikrovlnná trouba. Aby ke kolizím docházelo, musel by se tento zdroj rušení nacházet v blízkosti zařízení sítě (přesněji v jeho „viditelnosti“). Po zvážení aktuálního využití pásma byla zvolena 868 MHz nosnou frekvencí pro přenos signálu.

Podmínky pro provoz datových a jiných zařízení ve volných pásmech jsou pro tuzemské prostředí stanoveny generálními licencemi (dříve generální povolení), které

vydal Český telekomunikační úřad pod číslem GL-12/R/2000, případně GL-30/R/2000 . Normy pro provoz bezdrátových zařízení s krátkým dosahem stanovuje [4].

### 1.2.2 Volba specifických modulů

Systém bude řešen pomocí již zhotovených modulů. Z toho důvodu je nutné vybrat přesný typ, který bude vyhovovat všem požadavkům na zhotovení sítě. Při záměru duplexní komunikace, je nutné, aby každý uzel obsahoval přijímací a zároveň vysílací prvek. Na trhu dnes existují moduly pro funkci vysílače, přijímače ale rovněž transceivery, které implementují funkci vysílače+přijímač do jedné konstrukce. Jsou však podstatně dražší.

Pro realizaci byl zvolen cenově výhodné a zároveň funkčně víc než dostačující miniaturní digitální RF vysílače RFM02/868D a přijímače RFM01/868D zvláště pro jejich širokou škálu různých nastavení. Důležitým kritériem při volbě zařízení je maximální možná vzdálenost mezi přijímačem. Předpokládaná podmínka pro vzdálenost uzlů (50m) by měla být splněna. Zvolené moduly by měly být schopny komunikovat ve vzdálenosti až 200m.

Dle [5] je RFM02 levný ISM pásmový vysílací modul implementován s unikátní PLL (phase locked loop) koncepcí, pracující s FSK modulovanými signálovými rozsahy z pásem 433/868/915MHz, splňující FCC a ETSI směrnice. SPI interface se používá ke komunikaci s mikrokontrolérem k nastavení parametrů modulu. RFM02 pracuje s RFM01 přijímacím modulem.

RFM01 je nízko ztrátový FSK přijímač, pracující se stejnou koncepcí, jako RFM02. Integruje všechny RF funkce do jediného čipu. Je zapotřebí pouze mikrokontroléru, krystalu a antény k vytvoření spolehlivého přijímače.

### 1.2.3 Vlastnosti modulů RFM0X

- Možné ladění během provozu
- FSK přenos
- Smyčka fázového závěsu (PLL)
- Krátký čas zavěšení PLL smyčky
- Krokování PLL po 2,5 kHz
- Nastavitelná výstupní úroveň
- Volitelný datový tok
- Symetrický výstup antény
- Automatické ladění antény
- SPI interface
- Výstup hodinového a reset signálu pro externí MCU použití
- 10 MHz krystal pro PLL referenci
- Časovač probuzení

- Detekce slabé baterie
- 2,2 – 5,4 V zdroj napětí
- Nízká spotřeba
- Přechod do spánkového režimu při proudu nižším než 0,3μA
- 16 bitový FIFO zásobník
- Vnitřní demodulátor přijímače

#### **Důležité údaje z datasheetu:**

Odběr proudu      10,5 mA

Nejvyšší nastavitelný datový tok      115,2 kbps

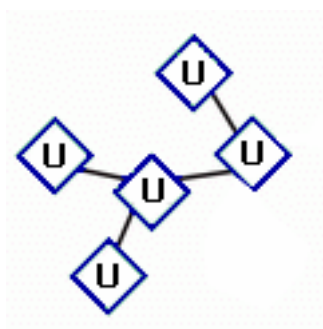
## **1.3 Struktura sítě**

Ve většině dosavadních sítí je zapotřebí různých druhů uzlových zařízení s odlišnými funkcemi a právy. Používají se koordinátory a směrovače, které mají za úkol řízení dat, zprostředkování domluvy mezi dvěma koncovými uzly nebo přiřazování adres nově přidaným zařízením. Užívá se tak topologií hvězdicových, stromových a smíšených.

### **1.3.1 Popis zvolené struktury**

Navržená síť je založena na struktuře peer-to-peer inspirací z technologie DigiMesh [2]. Což znamená, že komunikace probíhá přímo mezi dvěma účastníky sítě, bez nadřazeného prostředníka. Není přítomen žádný master – slave, všechna zařízení jsou si rovna a tudíž mají všechna stejná práva a možnosti.

Komunikační soustava se skládá pouze z jediného druhu zařízení, které budeme dále označovat pouze, jako uzel. Díky homogenní struktuře je tedy možné naprosto libovolně uspořádat propojení uzlů (viz. Obr. 1.1). Jde tedy o Ad hoc síť.



Obr. 1.1: Struktura sítě (U-uzel)

Výhody:

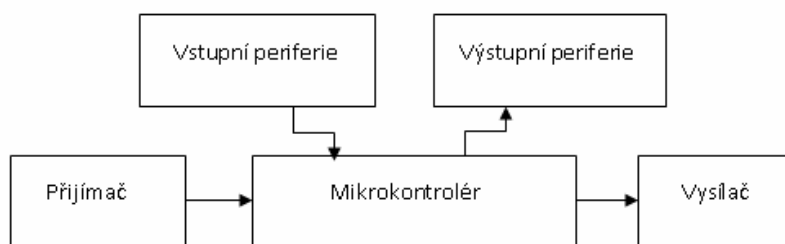
- Flexibilita
- Nezávislost na okolních uzlech
- Pouze jeden typ zařízení

Nevýhody:

- Složitější řešení směrování
- Kolize příjmů
- Možnost zahlcení sítě při broadcast vysílání

### 1.3.2 Blokové schéma uzlu

Na obrázku 1.2 je blokově znázorněno schéma jednoho uzlu. Mikrokontrolér zpracovává přijatá data, zobrazovaná prostřednictvím indikace výstupu. Pomocí vstupní periferie a vysílače data odesílá.



Obr. 1.2: Blokové schéma uzlu

## 1.4 Možnosti komunikace

U komunikace master zařízení s více slave zařízeními se využívá frekvenčního multiplexu, což znamená více kanálů (frekvenci vysílací a přijímací). Vzhledem k homogenitě navržené sítě probíhá komunikace pouze na jediné frekvenci. Zařízení ignoruje příjem vlastního vysílání.

### 1.4.1 Vysílání uzel – uzel (unicast)

Jde o přenos mezi dvěma specifickými uzly. Zdrojový uzel vyšle data pomocí směrovacího algoritmu a tabulky. Asociativní směrovací tabulka je soupis postupné cesty pro zadanou trasu uzel zdrojový -> uzel konečný. Díky této tabulce přesně víme, kudy požadovaná data poslat, aby se dostala k cílovému uzlu. Každý uzel ve zmapované trase potom přepośle data na přesně určený uzel další a ten tak učiní taktéž, než se data dostanou k cílovému zařízení.

Cílové zařízení po obdržení sobě adresované zprávy vyšle zpět ke zdrojovému uzlu paket s potvrzením o přijetí. Využije se tak obrácené trasy od předešlé. Až když zdrojový uzel přijme informaci o správném přijetí, potom vyšle následující paket ve

frontě. Pokud přijde hlášení o chybě v přenosu, daný paket se odvysílá znovu.

V případě, že se potvrzení přenosu vůbec nenavráť ke zdrojovému uzlu v čase rovnajícím se dvojnásobné době přenosu paketu sítí, provede se vysílací proces znovu. Opakování přenosu je možné až 3x, pokud stále není přítomna informace o přenosu, jde o chybu v cestě (nefunkční cílový uzel, chybná cesta k uzlu). Vyšle se tedy požadavek na získání nové trasy k cíli, opět s možným opakováním. Pokud hledání nové trasy vyjde negativně, je v síti rušení, vadný uzel, nutný pro komunikaci s cílovým uzlem nebo cílový uzel sám.

### **1.4.2 Vysílání všesměrové (broadcast)**

Tento druh vysílání je určen všem uzlům v síti, jako koncovým příjemcům. Zdrojový uzel vyšle do okolí zprávu určenou všem zařízením 4x po sobě. Zařízení, které zprávu obdrží, jí poté samy přeposílají do svého okolí taktéž 4x. Kvůli kolizím příjmu se při broadcast vysílání využije náhodně dlouhého zpoždění.

### **1.4.3 Mapování nové trasy**

Používá se k nalezení nové cesty ke zvolenému uzlu, pokud cestu neznáme nebo se jeví chybově. Zdrojový uzel vyšle všesměrově požadavek k nalezení trasy ke specifickému zařízení. Každý uzel, který zprávu obdrží a nenáleží mu, se stává přechodným uzlem. Takový uzel buď žádost ignoruje nebo zprávu přepošle dál. Pokud je nová cesta ke zdrojovému uzlu kratší než předešlá, zpráva bude aktualizována a všesměrově vyslána dále. Jestliže cílový uzel obdrží žádost o cestu, zašle odpověď zpět ke zdroji stejnou trasou, kudy k němu přišla a to bez ohledu na to kolik žádostí přijde. Zdroj má pak možnost z navrácených zpráv o cestě vybrat tu nejvýhodnější. Většinou tu s nejméně přeskoky. Ta se pak zapíše do tabulky a použije na směrování zpráv při každém požadavku o shodnou dvojici adres zdroj-cíl.

## **1.5 Kontrola chybovosti**

Pro částečnou kontrolu správnosti přijatých dat je vhodné přidat kontrolní kód, který může odhalit chybný přenos a popřípadě i přesně určit chybně přijaté bity.

### **1.5.1 Způsob ochrany přenosu**

V této komunikaci je použit blokový kód CRC-32bit. Z jehož výsledku se použijí pouze poslední 2 Bajty. Tato redukce výsledku je za účelem snížení celkové délky komunikačního rámce ovšem za cenu snížení šance k odhalení chybného příjmu. Ale vzhledem k tomu, že chceme přenášet krátké příkazy, by neměla tato úprava šanci k nalezení chyby kriticky snížit.

Zvolený kód sice neumožní nalézt špatně přijaté bity, ale při zjištění chybného přenosu vyšle žádost ke zdroji o opakování chybného paketu.

### 1.5.2 Popis použitého kódu

Postup při výpočtu použitého CRC:

K datům, která mají být přenesena se připojí stejný počet nul, jaká je délka generačního polynomu. Zde je to 32 bitů, použije se tedy 32 nul. Celá bitová posloupnost se poté pomocí funkce XOR sčítá s generačním polynomem, dokud nejsou místo přenášených dat přítomny pouze nuly. Za nimi potom místo přidaných 32 nul zůstane vypočtené CRC. Toto CRC se zkrátí o prvních 16 bitů a jako kontrolní součet se s daty pošle pouze zbylá polovina. Příjemce si provede shodnou úpravu přijatých dat a kontrolní součty se musí shodovat. Pokud tomu tak není, data byla poškozena.

Stanovený polynom pro výpočet CRC ze zasílaných dat :

1000 0001 1001 0001 0100 0000 1010 0011

Alternativou pro kontrolu podobného smyslu je například parita, samodetekční kódy...

### 1.5.3 Kanálové kódování dat

Je možné celou přenášenou zprávu kromě preamble překódovat prostřednictvím konvoluce na úrovni kanálového kódování. Zavede se tím předběžná korekce chyb, ovšem na úkor datového toku.

## 1.6 Struktura komunikačního rámce

Komunikační rámec by měl být co nejkratší, ale zároveň musí obsahovat všechny důležité informace o aktuálním přenosu dat. K čistým datům tak přidáváme další redundantní informace, které slouží k synchronizaci, směřování, kontrole či správné a snadnější orientaci v přenosu. Příjemce není fyzicky schopen zachytit a zpracovat první vyslaný bit, proto se před každým rámcem vyše ještě synchronizační slovo ve tvaru 10101010.

**Skladby použitých rámců:**

Vysvětlení zkratk:

|     |                             |
|-----|-----------------------------|
| Pre | preamble                    |
| TR  | typ rámce                   |
| AZ  | adresa zasilatele           |
| APP | adresa přechodného příjemce |
| AKP | adresa koncového příjemce   |
| DDR | délka datového rámce        |
| PNB | počet nahrazených bitů      |
| ČP  | číslo paketu                |
| D   | přenášená data              |
| CRC | kontrolní součet            |
| PS  | počet skoků                 |
| PB  | Potvrzovací byte            |

### 1.6.1 Přenosové rámce

#### Rámec pro přenos dat

|     |    |    |     |     |     |    |         |     |
|-----|----|----|-----|-----|-----|----|---------|-----|
| Pre | TR | AZ | AKP | DDR | PNB | ČP | D       | CRC |
| 8b  | 4b | 4b | 4b  | 5b  | 8b  | 8b | 0 – 32B | 16b |

Do připravené šablony rámce se vloží dané informace a rámec se zkontroluje, jestli se v něm nenachází fiktivní sekvence preamble. Pokud sekvenci nalezneme, nahradíme první bit ve fiktivním znaku jeho negací a zároveň navýšíme počet nahrazených bitů (PNB) o 1 a na konec dat přidáme byte s bitovou adresou nahrazeného bitu.

Po skončení kontroly celého rámce provedeme konečné CRC a připišeme na konec rámce posledních 16bitů výsledku CRC. Takovýto finální rámec se zařadí do fronty k vysílání.

Zaregistrováním preamble začne přijímač přijímat data. Přijme celý rámec a provede korekci změněných bitů, pokud se adresa zařízení shoduje s adresou přechodových příjemců v tabulce, rámec je tak jak byl přijat odvysílán aktuálním uzlem. Pokud zařízení nenalezne svou adresu v tabulce přechodových adres a zároveň není koncovým příjemcem, přijatý paket bude zahozen.

Takto paket putuje až k cílovému zařízení, které provede kontrolu CRC. Následně zkoriguje bity změněné kvůli zachování preamble, a pokud se CRC příchozího paketu shoduje s vypočteným, zašle potvrzovací rámec na zdrojovou adresu. Pokud se neshoduje, zašle požadavek o opakování přenosu daného paketu.

#### Rozbor jednotlivých částí:

##### Preamble (Pre):

Preamble je úvodním a jedinečným znakem, který se v celém rámci již nesmí znovu objevit.

Podoba preamble: 11100101

##### Typ rámce (TR):

Udává skladbu přijatého rámce.

TR – 0001: Rámec datový

0010: Rámec oznamující aktivitu zařízení

0100: Rámec pro hledání cesty

0110: Rámec s odpovědí o nalezené cestě

1000: Potvrzovací rámec

##### Adresa zasílatele (AZ):

Jedná se o 4-bitovou adresu zdrojového uzlu, uzly se adresují od čísla 0001. Adresy zařízení jsou pevné a je tedy možné do sítě připojit až 15 zařízení. Pro větší kapacitu sítě (více zařízení) by bylo nutné navýšit počet bitů adresného pole.

#### Adresa konečného příjemce (AKP):

Jde o 4-bitovou adresu cílového uzlu. Pokud je adresa nulová 0000 jedná se o zprávu broadcast.

#### Délka datového rámce (DDR):

Udává velikost datového bloku v bajtech.

#### Počet nahrazených bitů (PNB):

8mi bitová informace udávající počet bitů negovaných kvůli zachování jedinečné preamble. Prvotně nastaven na 00000000.

#### Číslo paketu (ČP):

Číslo právě zasílaného paketu

#### Přenášená data (D)

Souhrn informací, které chceme přenést

#### Kontrolní součet (CRC):

Kontrola chybovosti

#### **Rámec pro oznámení aktivity**

|     |    |    |     |
|-----|----|----|-----|
| Pre | TR | AZ | AKP |
| 8b  | 8b | 8b | 8b  |

Zařízení pravidelně ve stanoveném čase vysílají oznámení o vlastní aktivitě. Každé zařízení v jiném čase, stanoveném dle jeho osobní adresy + 5s. Časový úsek po jehož uplynutí se zařízení ohlásí, má název oznamovací doba, která se vypočítá takto

$$t_{\text{ozn}} = \text{čz} * 20 + 5000, \quad (1.1)$$

kde  $t_{\text{ozn}}$  je oznamovací doba a čz je číslo zařízení.

Např. zařízení s adresou 0010 = 2 -> 2\*20+5000 = 5040 ms.

#### **Rámec pro hledání trasy**

|     |    |    |     |     |    |       |     |
|-----|----|----|-----|-----|----|-------|-----|
| Pre | TR | AZ | APP | AKP | PS | Cesta | CRC |
| 8b  | 4b | 4b | 4b  | 4b  | 4b | PSx4b | 16b |

Při hledání trasy se jako adresa přechodného příjemce používá broadcast (0000). Zařízení zkontroluje z dříve došlých žádostí počet skoků a porovná ho s aktuálně došlým. Pokud je nový počet skoků nižší, zařízení připsá svou adresu do cesty a navýší počet skoků o 1. Následně upravený paket odovysílá.

#### Adresa přechodného příjemce (APP):

Jedná se o 4-bitovou adresu aktuálního příjemce

#### Počet skoků (PS):

4-bitová informace udávající počet provedených skoků.

#### **Rámec pro potvrzení nalezené trasy**

Používá se rámec stavbou shodný s rámcem k nalezení trasy, jen s jiným typový

označením, které nám říká, že rámec pouze přenáší informace o již nalezené trase. Zpráva se vrací k původnímu tazateli na cestu pozpátku. Jako přechodnou adresu vloží zařízení adresu před svou vlastní v zápisu cesty.

#### **Rámec s potvrzením příjmu**

|     |    |    |     |    |    |
|-----|----|----|-----|----|----|
| Pre | TR | AZ | AKP | ČP | PZ |
| 8b  | 4b | 4b | 4b  | 8b | 4b |

#### Potvrzovací znak (PZ):

Složení znaku nese informaci o správném či nesprávném příjmu

0110 = přijato, v pořádku

1001 = opakovat přenos daného paketu

## 2 REALIZACE ČÁSTI KOMUNIKACE

Tato kapitola se zabývá vlastním řešením části navrženého systému s vybranými funkcemi.

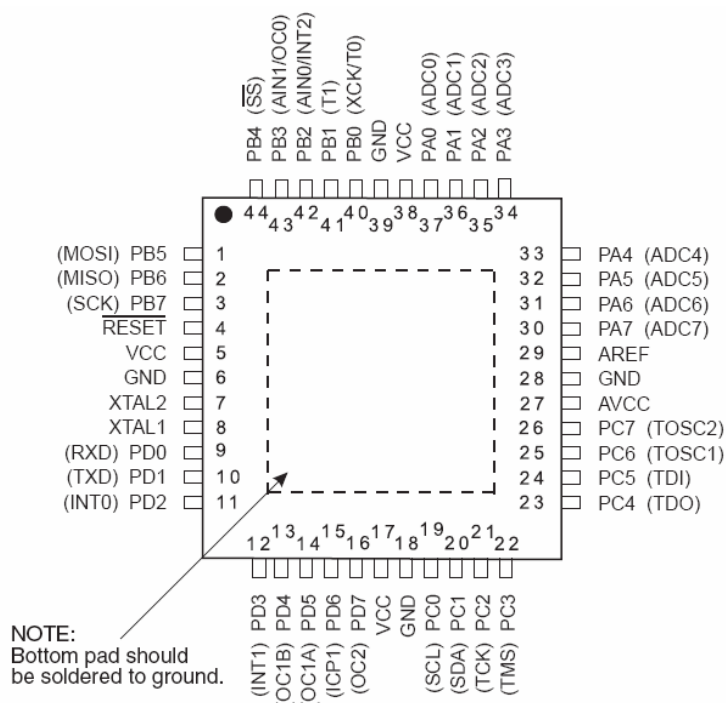
### 2.1 Obvodový návrh zařízení

V příloze A.1 je k dispozici kompletní obvodové schéma celého zařízení vytvořeného za účelem testování a demonstrace funkcí komunikace. Návrh byl vytvořen pomocí programu Eagle Layout Editor 5.0.0.

Většina pasivních součástek, diod a samotný mikrokontrolér byly obvodově navrženy v provedení SMD, z důvodu minimalizace a přehlednosti na úkor složitosti návrhu DPS. Klasický metalizovaný rezistor byl použit pouze u R12 navrženého k omezení podsvícení LCD, s ohledem na možné teplotní ztráty.

#### 2.1.1 Mikrokontrolér

Pro funkci mikrokontroléru byl zvolen ATmega16AU od firmy Atmel [10]. Pinová konfigurace daného integrovaného obvodu je znázorněna na obrázku (viz. Obr. 2.1). Volba byla provedena podle požadavku dostatečně rozsáhlé paměti, 8-bitové šířky registrů a rychlosti. Výpis vybraných parametrů zvoleného mikrokontroléru je uveden v tabulce (viz. Tab. 2.1).



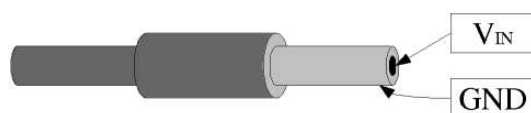
Obr. 2.1: Konfigurace pinů ATmega16

Tab. 2.1: Parametry zvoleného ATmega16

|                     |          |
|---------------------|----------|
| Pracovních registrů | 32x8     |
| Paměť flash         | 16kB     |
| Paměť EEPROM        | 512B     |
| Frekvence           | 16MHz    |
| Napájení            | 4,5-5,5V |
| Pouzdro             | TQFP44   |

### 2.1.2 Napájení

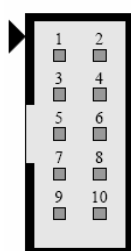
Z volby typu komunikačních modulů a mikrokontroléru vyplynula potřeba použití stabilizovaného zdroje napětí hodnoty 5V. Při úvaze použití 9V akumulátorů k napájení mobilního zařízení a předchozího požadavku byl zvolen stabilizátor napětí typu 7805. Připojení napájecího napětí je indikováno červenou LED diodou.



Obr. 2.2: Zapojení napájecího konektoru

### 2.1.3 Komunikační rozhraní

V obvodu zaimplementován převaděč MAX232 zajišťující komunikaci s osobním počítačem přes RS232 rozhraní. Toto spojení se využívá při identifikaci a koordinaci desky s programovacím softwarem. Pro programování samotné je na desce vyveden konektor JTAG-ice rozhraní 2x5 pinů pro programování a ladění přímo na desce. Zapojení pinů JTAG konektoru na desce bylo odvozeno z datasheetu ATmega16.



Obr. 2.3: Zapojení pinů JTAG konektoru

Tab. 2.2: Označení vývodů programovacího rozhraní JTAG

| Vývody | JTAG | Funkce                                |
|--------|------|---------------------------------------|
| 4,7    | +5V  | Napájecí napětí programovacího kabelu |
| 2,10   | GND  | Nulový potenciál                      |
| 1      | TCK  | Vstup hodinového signálu              |
| 3      | TDO  | Sériový datový výstup                 |

|   |      |                      |
|---|------|----------------------|
| 9 | TDI  | Sériový datový vstup |
| 5 | TMS  | Řídící signál        |
| 6 | RST\ | Resetovací signál    |

### 2.1.4 Ovládací prvky a indikace + LCD

U zařízení je požadováno použití vstupní periferie a indikace výstupu. Tyto funkce plní soustava 4 tlačítek pro ovládání příkazů a 4 informačních luminiscenčních diod.

K desce je též možné připojit zobrazovací displej typu 4x20 řádků pracující s řadičem HD44780. S ATmega16 je propojen 4-bitovou komunikací, vodičem na přepínání instrukcí/dat a hodinovým signálem. Vývod R/W je trvale uzemněn a displej tak může být využit pouze jako výstupní periferie. Podrobněji v [6].

Tab. 2.3: Popis vývodů LCD

| Vývod | Název | Funkce                              |
|-------|-------|-------------------------------------|
| 1     | Vss   | GND                                 |
| 2     | Vcc   | napájení +5V                        |
| 3     | V0    | nastavení kontrastu                 |
| 4     | RS    | volba mezi: instrukce – 0, data – 1 |
| 5     | R/W   | volba mezi: zápis – 0, čtení – 1    |
| 6     | E     | hodinový vstup                      |
| 7     | DB0   | data 0                              |
| 8     | DB1   | data 1                              |
| 9     | DB2   | data 2                              |
| 10    | DB3   | data 3                              |
| 11    | DB4   | data 4                              |
| 12    | DB5   | data 5                              |
| 13    | DB6   | data 6                              |
| 14    | DB7   | data 7                              |
| 15    | LED+  | anoda podsvětlení                   |
| 16    | LED-  | katoda podsvětlení                  |

Ovládání LCD je prováděno jak již bylo zmíněno 4-bitově pomocí instrukcí řadiče HD44780. Popis těchto instrukcí je vypsán na obrázku 2.4.

| Význam instrukce                                                                                                      | RS | RW | DB7  | DB6            | DB5            | DB4 | DB3 | DB2 | DB1 | DB0 | Délka   |
|-----------------------------------------------------------------------------------------------------------------------|----|----|------|----------------|----------------|-----|-----|-----|-----|-----|---------|
| smaže disp. a nastaví kurzor na začátek                                                                               | 0  | 0  | 0    | 0              | 0              | 0   | 0   | 0   | 0   | 1   | 1,64 ms |
| nastaví kurzor na začátek                                                                                             | 0  | 0  | 0    | 0              | 0              | 0   | 0   | 0   | 1   | x   | 1,64 ms |
| smer posuvu kurzoru I/D (0=vlevo, 1=vpravo), posuv textu S (0=ne, 1=ano)                                              | 0  | 0  | 0    | 0              | 0              | 0   | 0   | 1   | I/D | S   | 40 us   |
| D - zapne displej, C - zapne kurzor, B - zapne blikání kurzoru                                                        | 0  | 0  | 0    | 0              | 0              | 0   | 1   | D   | C   | B   | 40 us   |
| 1x posune (S/C=0 kurzor, S/C=1 text) smerem (R/L=0 vlevo, R/L=1 vpravo)                                               | 0  | 0  | 0    | 0              | 0              | 1   | S/C | R/L | x   | x   | 40 us   |
| inicializace:<br>DL=0 4-bit, DL=1 8-bit mód<br>N=0 jednořádkový, N=1 dvouřádkový disp.<br>F=0 font 5x8, F=1 font 5x10 | 0  | 0  | 0    | 0              | 1              | DL  | N   | F   | x   | x   | 40 us   |
| přepnutí na zápis do CGRAM                                                                                            | 0  | 0  | 0    | 1              | adresa v CGRAM |     |     |     |     |     | 40 us   |
| přepnutí na zápis do DDRAM                                                                                            | 0  | 0  | 1    | adresa v DDRAM |                |     |     |     |     |     | 40 us   |
| čtení příznaku BF (BF=0 příjem povolen, BF=1 řadič zaneprázdněn),<br>čtení adresy v DDRAM                             | 0  | 1  | BF   | adresa v DDRAM |                |     |     |     |     |     | 0 us    |
| zápis dat do CGRAM nebo DDRAM                                                                                         | 1  | 0  | data |                |                |     |     |     |     |     | 40 us   |
| čtení dat z CGRAM nebo DDRAM                                                                                          | 1  | 1  | data |                |                |     |     |     |     |     | 40 us   |

Obr. 2.4: Popis instrukcí LCD

### 2.1.5 Návrh DPS

K vytvoření návrhu byl taktéž použit software Eagle Layout Editor 5.0.0. Deska je vybavena v nepoužitých oblastech polygonem (tzv. rozlité měď), který je připojen k nulovému potenciálu. Tím je zajištěn komplexní rozvod zemnění. Polygon je odstraněn v oblasti s komunikačními moduly, která je vyčištěna od rozlité mědi i ostatních vodivých cest. Tato oblast je vytvořena z důvodu možného nežádoucího ovlivnění vysokofrekvenčních částí modulů s anténami.

Návrhy desky ze strany součástek i spojů jsou obsaženy v přílohách A.2 a A.3.

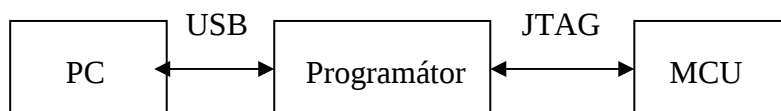
## 2.2 Programové vybavení (firmware)

Tato podkapitola přibližuje programovou část zařízení a součásti potřebné k jejímu vývoji.

### 2.2.1 Programátor

Programátor je zařízení prostřednictvím kterého je možné z vývojového prostředí převést program do mikrokontroléru. Zapojení mezi PC a rozhraní JTAG je blokově znázorněno na obrázku 2.4. JTAG rozhraní také umožňuje velice efektivní odladění vyvíjeného programu za pomoci HW debug jednotky použitého MCU.

V řešení byl použit UniProg-USB - univerzální programátor v1.0 od firmy PK-design, jehož parametry jsou upřesněny v manuálu [9]. Tato verze je uživatelsky příjemná z důvodu připojení k počítači přes USB rozhraní.



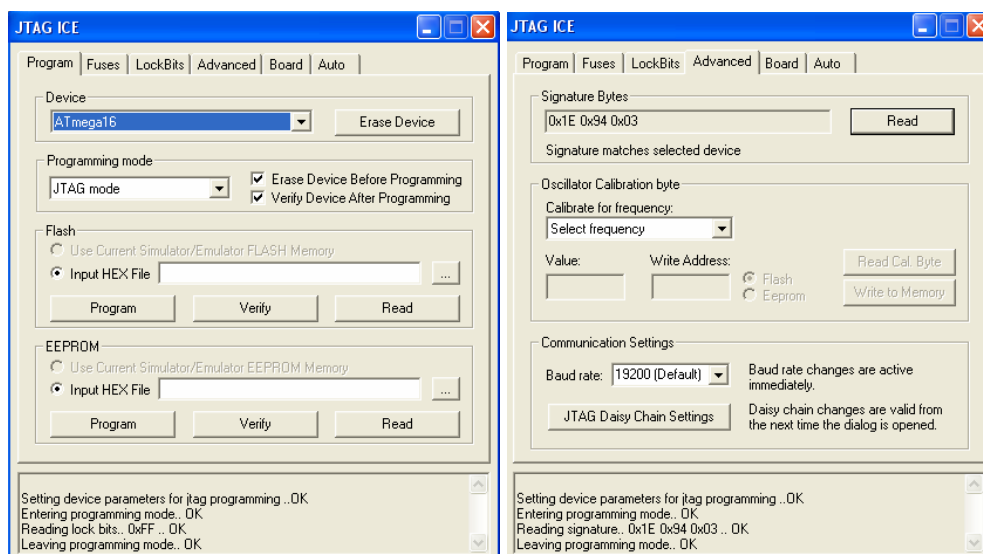
Obr. 2.5: Blokové schéma zapojení programátoru

## 2.2.2 Vývojové prostředí

Pro tvorbu firmwaru bylo využito AVR studio 4 (vývojové prostředí od firmy Atmel) spolu s kompilátorem AVR GCC z balíčku WinAVR. Společně fungují, jako veškerý software potřebný k tvorbě programu v syntaxi jazyka C pro MCU z rodiny AVR, který může být následovně naprogramován do mikrokontroléru.

Prostřednictvím AVR studia je možné nahrát zhotovený program do mikrokontroléru. Studio je také vybaveno nástrojem debugger, pomocí kterého je možné krokovat aplikaci v chodu a tím ji i ladit.

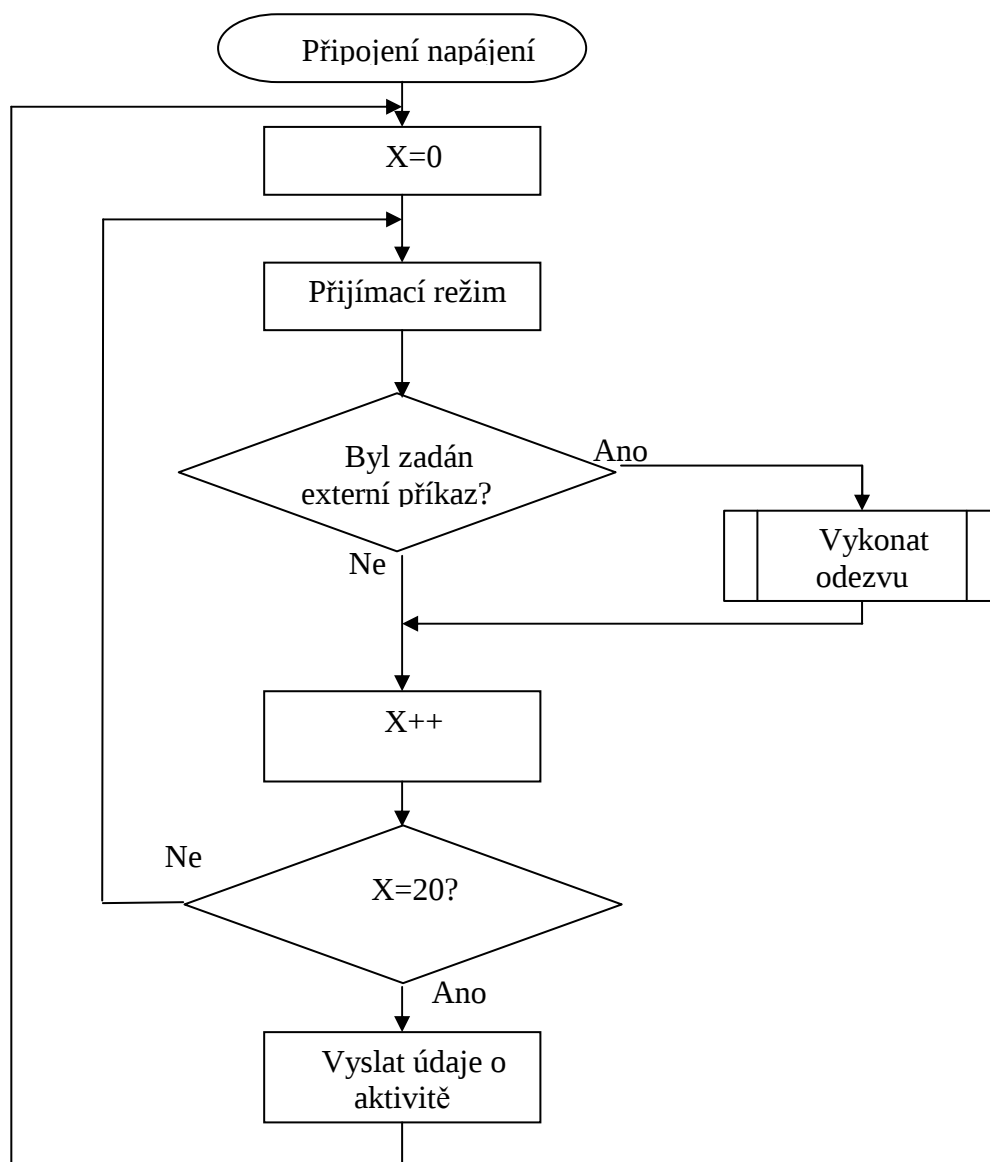
Po zprovoznění AVR studia, volbě mikrokontroléru a připojení programátoru ve správném módu bylo otestováno zda se na desce nachází výše zvolený typ mikrokontroléru vyčtením signatury ze známé lokace. Průběh kladně ohodnocené kontroly je viditelný na obrázku.



Obr. 2.6: Kontrola zvoleného mikrokontroléru

## 2.2.3 Vlastní program pro realizaci komunikace

Pomocí vývojového diagramu znázorněného na obrázku 2.7 je popsán přibližný chod základního cyklu.



Obr. 2.7: Vývojový diagram

Moduly je nutné před vlastním programem příkazově nastavit přesně do zvoleného režimu, aby tak bylo možné uskutečnit přenos mezi dvěma zařízeními. Nastavení přijímací strany:

```

iRFXX_WRT_CMD(0x0000);
iRFXX_WRT_CMD(0x918A); //860pásmo, +/-60khz
iRFXX_WRT_CMD(0xA640); //868MHz
iRFXX_WRT_CMD(0xC847); //4.8kbps
iRFXX_WRT_CMD(0xC6AB); //AFC nastavení
iRFXX_WRT_CMD(0xC42A); /*ruční nastavení clocku, digitalní
*/filt, DQD=4

iRFXX_WRT_CMD(0xC240); //vystup 2 MHz

iRFXX_WRT_CMD(0xC080); //zavření přijímače
iRFXX_WRT_CMD(0xCE88); //použit FIFO
  
```

```
iRFXWRT_CMD(0xCE8B);
iRFXWRT_CMD(0xC081); //otevreni prijimace
```

Nastavení vysílací strany:

```
oRFXWRT_CMD(0x9361); //860pásma, +/- 60kHz
oRFXWRT_CMD(0xA640); //868MHz
oRFXWRT_CMD(0xD040); //rate/2
oRFXWRT_CMD(0xC823); //4.8kbps
oRFXWRT_CMD(0xC220); //povoleni bit. synchronizace
oRFXWRT_CMD(0xC001); //vyrazeni TX
PORTB|=(1<<RFXWRT_DATA); //data na vystup
DDRB|=(1<<RFXWRT_DATA); //nastaveni portu
oRFXWRT_CMD(0xC039); //start
```

Po těchto nastaveních následuje soubor příkazů pro příjem/vysílání.

Zařízení nastaví přijímací režim, následně zkontroluje zda byl podán příkaz k přenosu dat, potom provede odezvu. Tato rutina je v chodu 3 sekundy a poté zařízení vyšle do okolí zprávu o své vlastní aktivitě. V programu taktéž proběhne aktualizace seznamu zařízení v dosahu. Hlavní rutina tedy vypadá takto:

```
void main(void)
{
    LCD_Init(); //inicializace displeje

    unsigned int i,j; //deklarace proměnných

    LED_OUTPUT(); //nastavení portu jako výstupního

    LED0_OFF();
    LED1_OFF();
    LED2_OFF();
    LED3_OFF();

    InitKbd(); //inicializace klaves

    int MyAddr=01;
    int HisAddr=02;
    BrdcstTimeOutReceived=65;
    BrdcstTimeOutTrasmit=20;

    int ctBroadLoop=0;
    while (1)
    {
        while (ctBroadLoop<BrdcstTimeOutTrasmit)
        {
            ReceiveEngine(); //naslouchani

            if (ReadKey()) //pokud byla stisknuta klavesa
            {
                LED3_OFF();
                int lKey=LastKey(); //do lKey posled. Stisk.
                switch (lKey) //rozhodovani podle typu klavesy
                {
                    case 1:
```

```

        if(SendSetLedPacket(MyAddr,HisAddr,0b00001000))
        {
            LED3_ON();
            BrdcstTimeOutReceived=65;
        }
        break;
        case 2:
        if
        (SendClearLedPacket(MyAddr,HisAddr,0b00001000))
        {
            LED3_ON();
            BrdcstTimeOutReceived=65;
        }
        break;

        default:
        break;
    }
}
ctBroadLoop+=1;

int i;
int wasRemoved=0;
for (i=0;i<DevCount;i++)
{
    DevToutCounter[i]--;
    if (DevToutCounter[i]<=0)
    {
        DevRemove(DevList[i]);
        wasRemoved=1;
    }
}
if (wasRemoved)
LCD_DispWriteDevs();//vypis aktual. Seznamu zarizeni

LED1_TRG();
}
ctBroadLoop=0;
SendBroadcast(MyAddr); //vyslani oznameni o aktivite
}

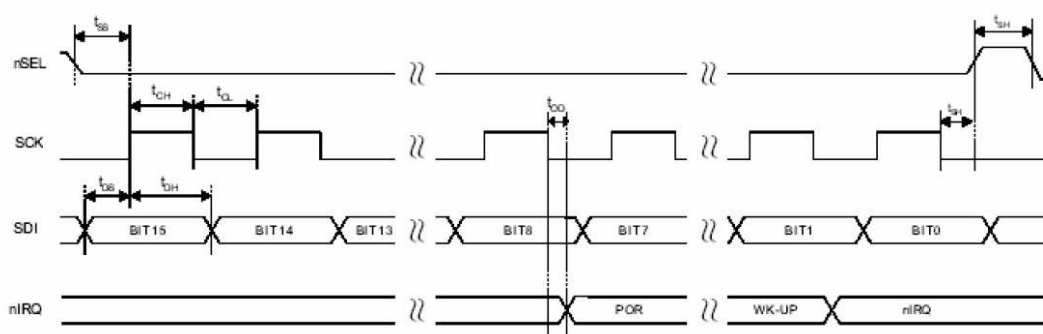
```

Při vysílání či příjmu je třeba mít inicializované moduly. Tuto rutinu je třeba provést v souladu s časovým diagramem modulu stanoveném v [5]. Inicializace vypadá takto:

```

void iRFXX_PORT_INIT(void)
{
    HI_SEL();
    HI_SDI();
    LOW_SCK();
    SEL_OUTPUT();
    SDI_OUTPUT();
    SDO_INPUT();
    SCK_OUTPUT();
}

```



Obr. 2.8: Časový diagram modulu

Vyčítání dat z FIFO paměti modulu probíhá po bitech a je vždy rozděleno po 8 bitech = 1 byte. První dva byty následující po preambuli jsou nutné pro zajištění správné komunikace mezi moduly a jsou stanovené pevně.

```

unsigned char RF01_RDFIFO(void)
{
    unsigned char i, Result;
    LOW_SCK();
    LOW_SDI();
    LOW_SEL();
    for(i=0; i<16; i++)
    { //preskakuje stavove bity
        HI_SCK();
        HI_SCK();
        LOW_SCK();
        LOW_SCK();
    }
    Result=0;
    for(i=0; i<8; i++)
    { //cteni dat z FIFO pameti po bitech
        Result<<=1;
        if(SDO_HI())
        {
            Result|=1;
        }
        HI_SCK();
        HI_SCK();
        LOW_SCK();
        LOW_SCK();
    };
    HI_SEL();
    return(Result);
}

```

Program provádí rutinu kontroly externích příkazů, aby mohl následně reagovat, případně jít dál. Proto je při stisku tlačítka zapamatována jeho adresa, od které se odvíjí další chod programu.

```

int ReadKey(void)
{
    if (_LastKey)
        return _LastKey;
}

```

```

if( bit_is_clear( PIND,4 ))
    _LastPushKey=1;
else
    if (_LastPushKey==1)
    {
        _LastKey=1;
        _LastPushKey=0;
        return 1;
    }
if( bit_is_clear( PIND,5 ))
    _LastPushKey=2;
else
    if (_LastPushKey==2)
    {
        _LastKey=2;
        _LastPushKey=0;
        return 2;
    }

```

Navrhovaná podoba preamble nemohla být použita při této realizaci komunikace z důvodu již předem pevně nastavené preamble v komunikačních modulech. Použitá preamble je tedy ve tvaru: 0xAA. Tento byte je před následujícími daty vyslán 3x.

Po vyslání zvoleného příkazu na druhé zařízení je očekáván příjem potvrzovacího rámce, který informuje o příchodu zprávy a její případné správnosti. Při přijmutí negativního potvrzovacího rámce je příslušný paket vyslán znovu. Pokud k přijmutí potvrzovacího rámce vůbec nedojde, bude vyslán tentýž příkaz ještě 3x. Příjem pozitivního potvrzovacího rámce je indikován čtvrtou informační LED diodou.

Soubory využívané pro ovládání LCD byly vytvořeny modifikací souborů užívaných k výuce předmětu Mikroprocesorová technika na FEKT.

## 3 NAMĚŘENÉ PARAMETRY

### 3.1 Rádiový dosah

Dle [5] by měl signál dosahovat do vzdálenosti 200m při zvolené nosné 860MHz. Tato skutečnost byla ověřena na volném prostranství. Zařízení č.1 bylo zanecháno na stabilním místě. Vzdalováním druhého zařízení a současnou kontrolou stavu komunikace byl hodnocen dosah.

Při vzdálenosti 150m mezi uzly komunikace byla komunikace stále stabilní, jako ve vzdálenosti bezprostřední blízkosti. Po prodloužení vzdálenosti o několik metrů (+10 až 30m) bylo možné v komunikaci pozorovat výpadky. Vyslané příkazy se občas nedostaly na stranu příjemce. Ve vzdálenosti okolí 200m byl pozorován stálý výpadek komunikace.

Rádiový dosah byl takto ověřen a odpovídá údajům předpokládaným. Pro navrhovanou komunikaci je víc než dostačující a je možné jej pomocí příkazu Receiver Setting Command omezit.

### 3.2 Odběr

Zařízení je možné provozovat v obyčejném módu bez LCD displeje. Odběr proudu v tomto režimu byl změřen na 29,7mA. Zatímco při použití LCD tedy v módu kontroly okolních zařízení byl naměřen více než dvojnásobný odběr proudu 70,9mA.

Příkon zařízení (se stabilizovaným zdrojem 7,93V) vypočtený dle vztahu

$$P = U \cdot I \quad , \quad (3.1)$$

tedy činí

bez LCD: 235,5mW

s LCD: 562,2mW

Z tohoto důvodu by měl být provoz LCD omezen pokud je využito omezeného zdroje napětí (akumulátor). Například vypnutým podsvětlením displeje a následnou aktivací na požadavek uživatele. Či možností uvést displej přímo do režimu spánku.

### 3.3 Rychlost přenosu

Rychlost přenosu je nastavitelná pomocí příkazu Data Rate Command. V uvedené verzi komunikace je rychlost nastavena na 4,8 kbps. Maximální dosažitelná rychlost s interním digitálním demodulátorem je stanovena na 115,2 kbps.

## 4 ZÁVĚR

Prostřednictvím této práce byl navržen systém pro bezdrátovou komunikaci se zadanými vlastnostmi a realizována jeho ukázka. Zpráva popisuje vybrané komunikační moduly použité k realizaci sítě, uspořádání sítě, komunikaci mezi jednotlivými uzly a návrh struktury komunikačních rámců.

Další částí práce byl schematický a hardwarový návrh uzlu sítě včetně výroby DPS, jejího osazení a oživení zařízení. Takto byly vyrobeny 2 uzly komunikace pro test části navržené realizace komunikace.

Následující část práce se soustředila na tvorbu programu ve vývojovém prostředí AVR studio v syntaxi jazyka C. Byla vytvořena aplikace realizující komunikační protokol mezi dvěma zařízeními pro přenos stanoveného příkazu prostřednictvím datového rámce a následnou kontrolu správného příjmu pomocí rámce potvrzovacího. Každé zařízení ve stanoveném cyklu vysílá informativní rámec o vlastní aktivitě na, který není odpovídáno potvrzením příjmu. Jedno ze zařízení bylo osazeno LCD zobrazovacím displejem, který informuje o aktivitě druhého zařízení v dosahu, na základě výše zmíněných ohlášení o vlastní aktivitě.

Bylo provedeno přibližné měření maximálního dosahu na volném prostranství mezi dvěma uzly, odběru zařízení v různém módu a nastavení rychlosti přenosu.

Dalšími kroky v postupu práce by mohla být konstrukce třetího uzlu sítě. Realizace adresování, trasování a vylepšení kontroly chybovosti v komunikačním protokolu. Stávající protokol je vytvořen k snadné modifikaci pro zprovoznění adresace. Nebyl dokončen z důvodu absence třetího uzlu komunikace a časové náročnosti.

# LITERATURA

- [1] MARŠÁLEK, R. *Teorie rádiové komunikace*. Elektronické skriptum. Brno: FEKT VUT v Brně, 2005
- [2] DIGI. Naverland 2, 13. *Wireless Mesh Networking, ZigBee vs. DigiMesh*. 2008. 6s.
- [3] THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC. New York, USA. *IEEE Standards 802.11.X*. 2007.
- [4] ČSN ETSI EN 300 220 (875015). *Elektromagnetická kompatibilita a rádiové spektrum (ERM) – Přístroje s krátkým dosahem (SRD)*, 2001
- [5] HOPES MICROELEKTRONICS, CO. *Data sheets RFM01, RFM02*. [online]. Dostupné na WWW: < <http://zefiryn.tme.pl/dok/wd1/rfm01.pdf> >
- [6] MASLAN, S. *Ovládání znakových LCD s řadičem HD44780*. [online], Hw.cz, 2009. Dostupné na www: <<http://hw.cz/novinky/art2727-ovladani-znakovych-lcd-s-radicem-hd44780.htm>>.
- [7] VÁŇA, V. *Atmel AVR programování v jazyce C*. BEN, 2003.
- [8] EZK, *E-katalog nabídky EZK* [online]. [cit. 20.5.2009], Dostupný z WWW: <[www.ezk.cz](http://www.ezk.cz)>.
- [9] PK-Design, *UniProg-USB v.1.0* [online]. 2007, 30s. Dostupný z WWW: <[http://www.pk-design.net/Datasheets/UniProgUSB\\_v10\\_doc\\_ver\\_v11\\_20070820.pdf](http://www.pk-design.net/Datasheets/UniProgUSB_v10_doc_ver_v11_20070820.pdf)>
- [10] ATMEL. San Jose, CA95131, USA. *ATmega16 datasheet*. 2007. 354s.

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

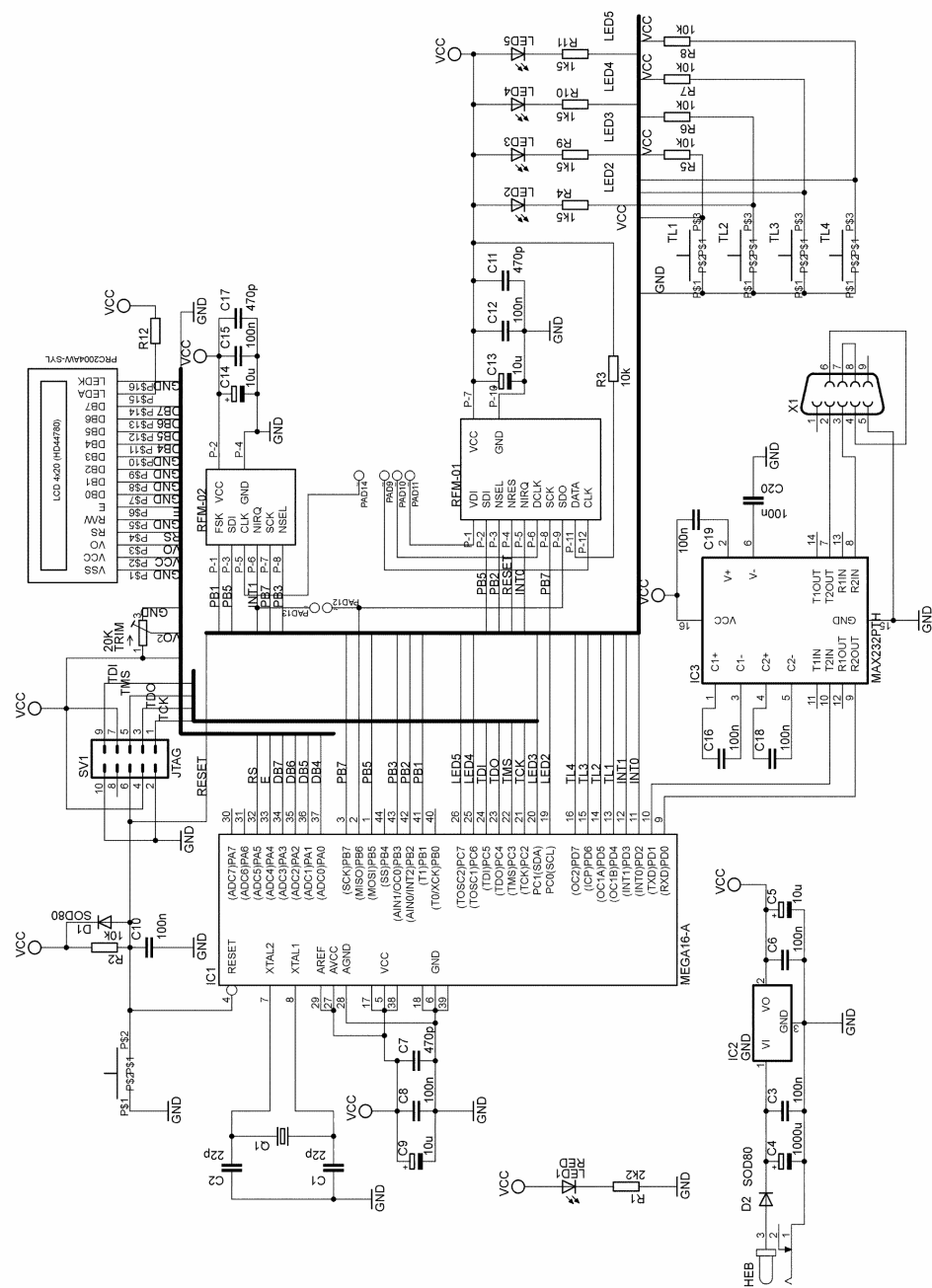
|             |                                                                                           |
|-------------|-------------------------------------------------------------------------------------------|
| <i>CRC</i>  | Cyclic redundancy check (kontrolní součet)                                                |
| <i>DPS</i>  | Deska plošných spojů                                                                      |
| <i>FSK</i>  | Frequency shift keying (frekvenční modulace)                                              |
| <i>GND</i>  | GrouND (zemní potenciál)                                                                  |
| <i>IrDA</i> | Infrared data association (konsorcium popisující komunikaci pomocí infračerveného světla) |
| <i>ISM</i>  | Industrial, scientific and medical (pásmo pro průmyslové, vědecké a zdravotnické využití) |
| <i>JTAG</i> | Joint test action group                                                                   |
| <i>LCD</i>  | Liquid Crystal Display (displej s kapalnými krystaly)                                     |
| <i>SMD</i>  | Surface Mounted Device (součástka pro povrchovou montáž)                                  |
| <i>SPI</i>  | Serial peripheral interface (sériové periferní rozhraní)                                  |

# SEZNAM PŘÍLOH

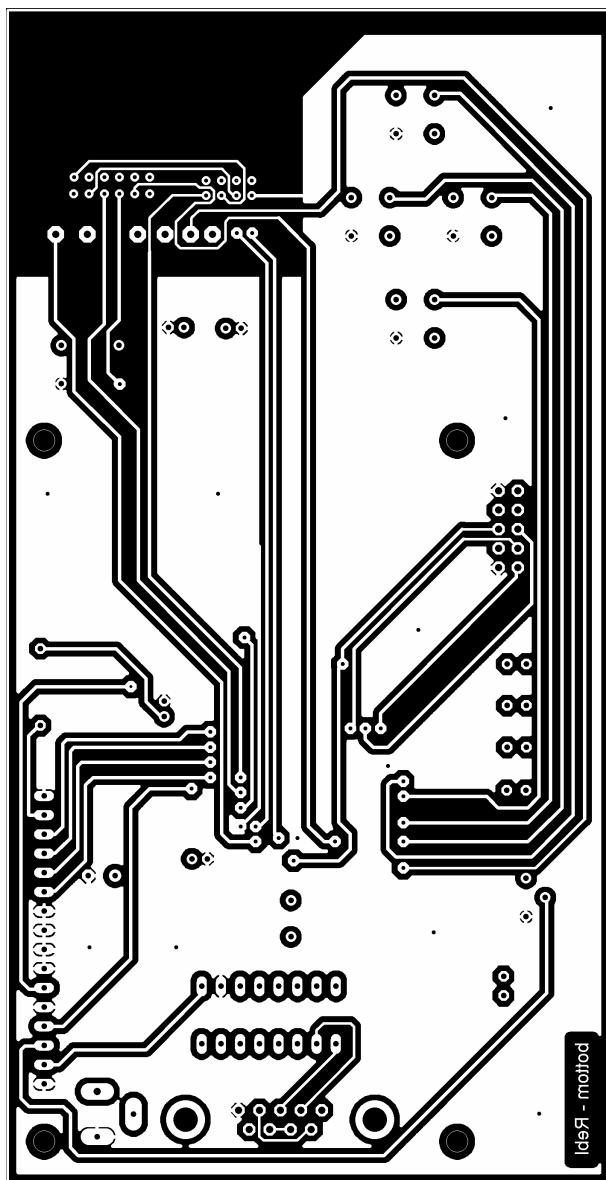
|          |                                                    |           |
|----------|----------------------------------------------------|-----------|
| <b>A</b> | <b>Dokumentace realizace zařízení</b>              | <b>26</b> |
| A.1      | Obvodové zapojení .....                            | 26        |
| A.2      | Deska plošného spoje – top (strana součástek)..... | 27        |
| A.3      | Deska plošného spoje – bottom (strana spojů) ..... | 28        |
| <b>B</b> | <b>Seznam součástek</b>                            | <b>29</b> |

# A DOKUMENTACE REALIZACE ZAŘÍZENÍ

## A.1 Obvodové zapojení

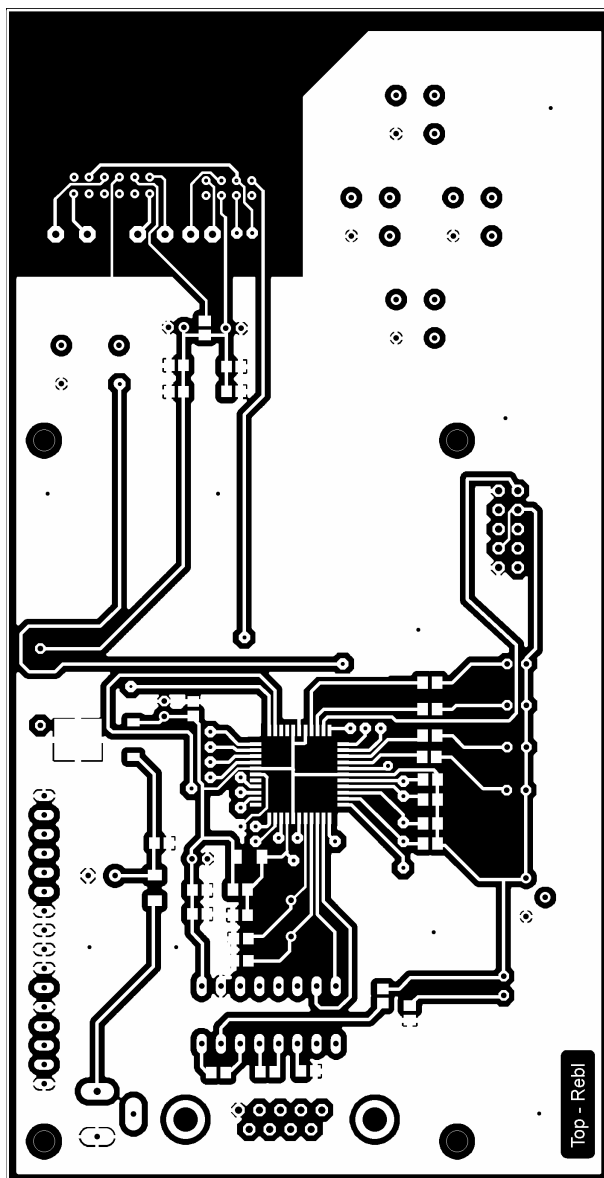


## A.2 Deska plošného spoje – top (strana součástek)



Rozměr desky 80 x 155 [mm], měřítko M1:1

### A.3 Deska plošného spoje – bottom (strana spojů)



Rozměr desky 80 x 155 [mm], měřítko M1:1

## B SEZNAM SOUČÁSTEK

| Označení | Hodnota           | Pouzdro     | Popis                      |
|----------|-------------------|-------------|----------------------------|
| C1       | 22p               | C0805       | Keramický kondenzátor SMD  |
| C2       | 22p               | C0805       | Keramický kondenzátor SMD  |
| C3       | 100n              | C0805       | Keramický kondenzátor SMD  |
| C4       | 1000u             | ELRA        | Elektrolytický kondenzátor |
| C5       | 10u               | ELRS        | Elektrolytický kondenzátor |
| C6       | 100n              | C0805       | Keramický kondenzátor SMD  |
| C7       | 470p              | C0805       | Keramický kondenzátor SMD  |
| C8       | 100n              | C0805       | Keramický kondenzátor SMD  |
| C9       | 10u               | ELRS        | Elektrolytický kondenzátor |
| C10      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C11      | 470p              | C0805       | Keramický kondenzátor SMD  |
| C12      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C13      | 10u               | ELRS        | Elektrolytický kondenzátor |
| C14      | 10u               | ELRS        | Elektrolytický kondenzátor |
| C15      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C16      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C17      | 470p              | C0805       | Keramický kondenzátor SMD  |
| C18      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C19      | 100n              | C0805       | Keramický kondenzátor SMD  |
| C20      | 100n              | C0805       | Keramický kondenzátor SMD  |
| D1       | TMM BAT42<br>SMD  | SOD 80C     | Usměrňovací dioda          |
| D2       | TMM BAT42<br>SMD  | SOD 80C     | Usměrňovací dioda          |
| HEB      |                   | NAZ 2.1 V P | Napájecí konektor          |
| IC1      | ATmega16AU        | TQFP44      | Mikrokontrolér             |
| IC2      | 7805              | TO252       | Stabilizátor napětí        |
| IC3      | MAX232N           | DIL16       | RS232/V 24                 |
| LED1     | RED               | L934ID      | LED dioda                  |
| LED2     | GREEN             | L934SGD     | LED dioda                  |
| LED3     | GREEN             | L934SGD     | LED dioda                  |
| LED4     | GREEN             | L934SGD     | LED dioda                  |
| LED5     | GREEN             | L934SGD     | LED dioda                  |
| PRC      | PRC2004AW-<br>SYL |             | LCD displej 4x20           |
| Q1       | 10MHz             | HC49/U      | Krystal                    |
| R1       | 2k2               | M0805       | SMD rezistor               |
| R2       | 10k               | M0805       | SMD rezistor               |
| R3       | 10k               | M0805       | SMD rezistor               |

|      |       |           |                       |
|------|-------|-----------|-----------------------|
| R4   | 1k5   | M0805     | SMD rezistor          |
| R5   | 10k   | M0805     | SMD rezistor          |
| R6   | 10k   | M0805     | SMD rezistor          |
| R7   | 10k   | M0805     | SMD rezistor          |
| R8   | 10k   | M0805     | SMD rezistor          |
| R9   | 1k5   | M0805     | SMD rezistor          |
| R10  | 1k5   | M0805     | SMD rezistor          |
| R11  | 1k5   | M0805     | SMD rezistor          |
| R12  | 22R   | 0207      | Metalizovaný rezistor |
| SV1  |       | S2G20     | 2x5 pinů              |
| TRIM | 20k   | T73YE     | Odporový trimr        |
| U1   |       | TM070     | Mikrospínač           |
| U2   | RFM02 | 868D      | Vysílací modul        |
| U3   |       | DT 6/8P   | Tlačítko              |
| U4   | RFM01 | 868D      | Přijímací modul       |
| U5   |       | DT 6/8P   | Tlačítko              |
| U6   |       | DT 6/8P   | Tlačítko              |
| U7   |       | DT 6/8P   | Tlačítko              |
| X1   | MALE  | CAN9 V 90 | Konektor RS232        |

---