

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

BEZDRÁTOVÝ PŘENOS DAT V ISM PÁSMU POMOCÍ
MANCHESTER KÓDOVÁNÍ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

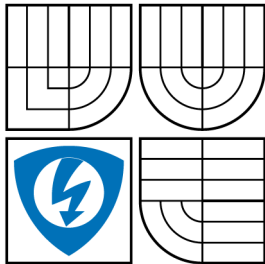
AUTOR PRÁCE
AUTHOR

ZDENĚK ZADĚRA

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

BEZDRÁTOVÝ PŘENOS DAT V ISM PÁSMU POMOCÍ MANCHESTER KÓDOVÁNÍ

WIRELESS DATA TRANSMISSION IN ISM FREQUENCY BAND USING
MANCHESTER ENCODING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

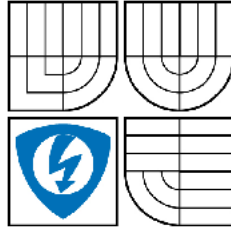
AUTOR PRÁCE
AUTHOR

ZDENĚK ZÁDĚRA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. ONDŘEJ PAVELKA

BRNO 2009



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Zdeněk Záděra
Ročník: 3

ID: 70256
Akademický rok: 2008/2009

NÁZEV TÉMATU:

Bezdrátový přenos dat v ISM pásmu pomocí Manchester kódování

POKYNY PRO VYPRACOVÁNÍ:

Navrhněte a realizujte moduly pro jednosměrný přenos telemetrických dat v ISM (433MHz, popř. 866MHz) pomocí Manchester kódování. Zařízení bude osazeno mikrokontrolérem ATmega, obslužný program bude napsán v jazyce C. Projekt řešte ve spolupráci s pracovníky vývojového centra Honeywell v Brně.

DOPORUČENÁ LITERATURA:

- [1] MANN, B. : C pro mikrokontroléry. Praha: BEN, 2003. ISBN 80-7300-077-6
- [2] Kainka, B. :Měření, řízení a regulace pomocí PC, BEN, 2003. ISBN 80-7300-089-X

Termín zadání: 9.2.2009

Termín odevzdání: 2.6.2009

Vedoucí práce: doc. Ing. Václav Zeman, Ph.D.

prof. Ing. Kamil Vrba, CSc.
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 věnuje návrhu a realizaci modulů pro obousměrný přenos dat v pásmu ISM (Industrial, Scientific and Medical), použitelných pro soukromé účely, např. přenášení telemetrických dat mezi mobilními zařízeními v terénu. Realizace je řešena pomocí RF modulů RFM12BP o maximálním výkonu 500mW, tyto inteligentní moduly jsou ovládnuty pomocí mikroprocesoru ATtiny2313 firmy Atmel. Mikroprocesor je použit také pro převod mezi sériovým rozhraním počítače a rozhraním RF modulu. Obslužný program pro mikrokontroléry je napsán v jazyce C ve vývojovém prostředí CodevisionAVR. Projekt je řešen ve spolupráci s pracovníky vývojového centra Honeywell v Brně.

KLÍČOVÁ SLOVA

ISM pásmo, bezdrátový přenos dat, RF modul, RFM12BP, ATtiny2313, CodevisionAVR

ABSTRACT

This work is dedicated to the design and implementation of modules for the bidirectional transmission of data in the ISM band (Industrial, Scientific and Medical), used for private purposes, such as the transmission of telemetry data between mobile devices in the field. Delivery is solved by using RF modules RFM12BP the maximum power 500mW, these intelligent modules are controlled with microprocessor companies ATtiny2313 Atmel. Microprocessor is also used for the transfer of serial interface between computers and RF interface module. Utility for microcontrollers is written in C language development in middle CodevisionAVR. The project is designed in cooperation with the staff development center in Brno Honeywell.

KEYWORDS

ISM band, wireless data transmission, RF module, RFM12BP, ATtiny2313, CodevisionAVR

ZDE VLOŽIT PRVNÍ LIST LICENČNÍ
SMOUVY

ZDE VLOŽIT DRUHÝ LIST LICENČNÍ
SMOUVY

ZÁDĚRA, Z. Bezdrátový přenos dat v ISM pásmu pomocí Manchester kódování. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 48 s. Vedoucí bakalářské práce Ing. Ondřej Pavelka

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Bezdrátový přenos dat v ISM pásmu pomocí Manchester kódování“ 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. Ondřeji Pavelkovi za cenné rady, připomínky a odborné vedení práce. Dále děkuji za konzultace doc. Ing. Václavu Zemanovi Ph.D.

V Brně dne

.....

(podpis autora)

OBSAH

1 Úvod	12
2 Bezdrátová komunikace	13
2.1 Typy bezdrátové komunikace	13
2.2 Historie a standardy rádiové komunikace	13
2.3 Radiotechnická spektra v ČR a jejich využití	13
2.3.1 ISM pásmo	14
2.3.2 Pásmo 433,05-434,79MHz	15
2.4 Rádiové (hybridní) moduly	16
2.5 Manchester kódování	16
3 Sériový přenos dat	17
3.1 Forma přenášených dat	17
3.2 Sériové rozhraní RS-232C	18
3.3 Sériové rozhraní UART	19
4 Návrh komunikačního modulu	20
4.1 Blokové schéma	20
4.2 Výběr mikroprocesoru	21
4.3 Výběr RF modulu	22
4.4 Vývojové prostředí CodevisionAVR	24
5 Realizace komunikačního modulu	26
5.1 Schéma zapojení	26
5.2 Komunikace mikrokontroléru s modulem	27
5.3 Přenosový diagram	30
5.4 Obslužný program	32
5.4.1 Vývojový diagram	32
5.4.2 Popis algoritmu	35
5.4.3 Zdrojový kód	37
6 Závěr	39
Literatura	40
Seznam symbolů, veličin a zkratk	42
Seznam příloh	43

A Schéma	44
A.1 Schéma zapojení	44
A.2 Schéma převodníku	45
B DPS	46
B.1 Desky plošných spojů ze strany součástek	46
B.2 Osazení desek plošných spojů	47

SEZNAM OBRÁZKŮ

2.1	Časový průběh Manchester kódování	16
3.1	Časový průběh vyslaného znaku M (ASCII 77) po sériové lince RS232	18
3.2	Časový průběh jednoho znaku, vyslaného přes UART	19
4.1	Blokové schéma komunikace	20
4.2	Blokové schéma navrhovaného modulu	21
4.3	Rozmístění pinů pouzdra PDIP mikrokontroléru ATtiny2313	21
4.4	Ukázka práce ve vývojovém prostředí CodeVisionAVR	25
4.5	Interní programátor nabízí spoustu programovacích protokolů	25
5.1	Vnitřní zapojení obvodu MAX3232	27
5.2	Propojení mikrokontroléru s RF modulem	28
5.3	Časový průběh SPI komunikace	29
5.4	Přenosový diagram komunikace mezi moduly	31
5.5	Vývojový diagram hlavní funkce main	32
5.6	Vývojový diagram funkce vysilaniRF	33
5.7	Vývojový diagram funkce prijemRF	34
A.1	Schéma zapojení modulu s mikroprocesorem	44
A.2	Zapojení převodníku sériového rozhraní na úroveň TTL s obvodem MAX3232	45
B.1	Deska plošných spojů ze strany součástek	46
B.2	Deska plošných spojů převodníku ze strany součástek	46
B.3	Osazení desky plošných spojů	47
B.4	Osazení desky plošných spojů převodníku	47

SEZNAM TABULEK

2.1	Srovnání bezdrátových standardů a jejich požadavků na licencovaná pásma	14
2.2	ISM pásmo, definované podle ITU-R	14
3.1	popis pinů konektoru CANNON 9	19

1 ÚVOD

Se vznikem prvních počítačů vznikla také potřeba přenositelnosti dat mezi jednotlivými počítači. V původní podobě byla data přenášena na pevných médiích, jako například diskety nebo děrné pásy. Později se tento přenos usnadnil propojováním počítačů kabely. Jsou však situace, kdy nelze z různých důvodů, nebo pro přílišnou složitost použít fyzické spojení kabely. Typický příklad je sběr dat v terénu, kdy se jeden z počítačů pohybuje a nemůže být omezen pevným kabelem.

V tomto případě je jediné řešení použití bezdrátového přenosu dat. Dnes existuje spousta komerčních technologií a standardů v pásmu ISM (např. Wifi, Bluetooth, GSM, UMTS ap.), nabízí se však řešení použití modulů pro přenos dat v tomto pásmu na tzv. volných frekvencích např. 434, 868 či 915 MHz bez potřebné licence. Pro tyto volná pásma je dnes na trhu k dostání spousta rádiových (RF) modulů, což jsou vlastně vysílače a přijímače, které využívají frekvenční kódování (FSK) k bezdrátovému přenášení dat. Použití těchto bezlicenčních pásem se liší podle geografické oblasti. Toto rozdělení stanovuje Mezinárodní telekomunikační úřad (viz dále).

V rámci této práce bude vybrán konkrétní RF modul a mikrokontrolér k jeho ovládání. Dále je třeba navrhnout algoritmus pro komunikaci mikrokontroléru na jedné straně se zdrojem dat (v této práci sériové rozhraní počítače RS232) a na straně druhé s modulem. Tento algoritmus by měl zařídit správný převod dat mezi oběma rozhraními. Použitý mikroprocesor bude tedy sloužit jako převodník mezi sériovou komunikací RS232 a rádiovým modulem a současně bude tento modul ovládat.

Součástí práce bude dále návrh schématu zapojení a desky plošných spojů, aby bylo možno vyrobit funkční komunikační modul. Výsledné moduly budou navrženy s přihlédnutím k minimální proudové spotřebě, moduly budou indikovat překročení maximální vzdálenosti. Práce bude řešena ve spolupráci s pracovníky vývojového centra Honeywell v Brně.

2 BEZDRÁTOVÁ KOMUNIKACE

2.1 Typy bezdrátové komunikace

Bezdrátovou komunikaci dělíme podle způsobu přenosu do tří základních skupin:

- **Optická** - přenos pomocí světelných vln (infračervené, laserové, řeč těla),
- **Rádiová** - přenos pomocí rádiových vln, dnes zřejmě nejpoužívanější forma bezdrátového přenosu,
- **Sonická** - využívá zvukové vlny (ponorky, verbální komunikace).

Tato práce se dále bude zabývat rádiovým přenosem, který bude také použit pro praktickou realizaci.

2.2 Historie a standardy rádiové komunikace

Průkopníkem bezdrátové komunikace byl podle [17] italský vědec Guglielimo Marconi. V roce 1895 úspěšně přenesl informaci do vzdálenost dvou kilometrů. V roce 1901 došlo k prvnímu bezdrátovému přenosu dat přes moře. V počátcích bezdrátové komunikace se data přenášela pomocí Morseovy abecedy. Později, v roce 1904, byl zaznamenán první přenos lidského hlasu. Po delší době v roce 1934 vznikla frekvenční modulace (FM), která byla kvalitnější, než starší amplitudová modulace (AM).

V roce 1947 se začaly objevovat první buňkové rádiové systémy, podobné dnešní síti GSM. K pokrytí používaly tzv. základnové stanice. Ty byly zatím analogové, ale v osmdesátých letech 20. století se začaly objevovat první digitální radiotelefonní systémy. Jedním z nich byl dnes známý systém GSM, který využívají mobilní operátoři. Ke konci 20. století se vyvíjely už jen digitální systémy. V budoucnu se očekává masivní rozvoj bezdrátových technologií a nahrazování kabelových spojení. Dnes najdeme bezdrátovou komunikaci v mobilních telefonech, GPS navigacích, satelitním i pozemním vysílání televize či rádia.

V tab. 2.1 je srovnání bezdrátových standardů, tato tabulka je převzata z [17].

2.3 Radiotechnická spektra v ČR a jejich využití

Využití rádiového spektra v České Republice řídí předpisy Českého telekomunikačního úřadu (ČTÚ) v souladu §56 a §95 zákona č. 151/2000Sb. Předpisy jsou vypracovány v souladu s Mezinárodní telekomunikační unií (ITU).

Tab. 2.1: Srovnání bezdrátových standardů a jejich požadavků na licencovaná pásma

Standard	licencované pásmo	nelicencované pásmo
Bluetooth	ne	ano
WLAN (obecné)	ano	ano
IEEE 802.11 (Wi-Fi)	ne	ano
GSM-GPRS	ano	ne
GSM-HSCSD	ano	ne
UMTS	ano	ne
Prioprietální řešení	ano	ano

2.3.1 ISM pásmo

Pásmo ISM (industrial, scientific and medical) bylo původně určeno pro potřeby průmyslové a vědecké pro účely jiné, než komunikační [16]. Toto frekvenční pásmo je řízeno Radiokomunikačním řádem ¹. Uživatel musí brát v úvahu určité interference mezi zařízeními v tomto frekvenčním pásmu. Pásmo má rozsah od desítek MHz až po stovky GHz, jak je znázorněno v tab. 2.2.

Tab. 2.2: ISM pásmo, definované podle ITU-R

Frekvenční rozsah	Základní frekvence	Dostupnost
6.765–6.795 MHz	6.780 MHz	potřeba povolení
13.553–13.567 MHz	13.560 MHz	
26.957–27.283 MHz	27.120 MHz	
40.66–40.70 MHz	40.68 MHz	
433.05–434.79 MHz	433.92 MHz	jen oblast 1
902–928 MHz	915 MHz	jen oblast 2
2.400–2.500 GHz	2.450 GHz	
5.725–5.875 GHz	5.800 GHz	
24–24.25 GHz	24.125 GHz	
61–61.5 GHz	61.25 GHz	potřeba povolení
122–123 GHz	122.5 GHz	potřeba povolení
244–246 GHz	245 GHz	potřeba povolení

Do tohoto pásma patří také známé standardy bezdrátového přenosu informací,

¹(Radiokomunikační řád, Mezinárodní telekomunikační unie, Ženeva, 2004 [Radio Regulations, International Telecommunication Union, Geneva, 2007])

například technologie Wifi či Bluetooth, obě pracující na frekvenci 2,450GHz, dále se tu nachází standard GSM, nebo bezdrátové telefony (tab. 2.1). Frekvence určené pro volné použití jsou 433,92MHz či 915MHz. Mezinárodní telekomunikační úřad rozděluje použití těchto volných pásem do tří oblastí podle [16]:

- Oblast 1 - pro Evropu, Afriku, Irák a Sovětský svaz
- Oblast 2 - pro Ameriku, Grónsko a některé Pacifické ostrovy
- Oblast 3 - pro asijské státy kromě Sovětského svazu, Irán a Oceánii.

Toto rozdělení lze nalézt také v [11]. Z uvedeného rozdělení plyne, že pásmo 915MHz lze využít jen v Americe a v Evropě nemůže být použito. Pro realizaci v naší geografické oblasti tedy připadá frekvence 433,92MHz a nabízí se i frekvence 868MHz. Výhoda těchto pásem je, že nejsou zpoplatněna, naopak nevýhoda je, že se smí používat jen zařízení s omezeným vysílacím výkonem, aby nedocházelo k interferencím s jinými zařízeními, používající stejnou frekvenci. I přes výhody použití frekvence 868MHz, jako jsou:

- menší provoz na této frekvenci, tedy menší interferenční rušení,
- větší dosah, při použití směrové antény až 3km,
- anténa menší velikosti,

má však provoz na této frekvenci velkou nevýhodu, a to horší přenos přes překážky. Čím více je překážek v přenosové cestě (budova, zeď, kopec, stromy ap.), tím více se zhoršuje kvalita přenosu a snižuje se vzdálenost, na kterou lze data přenášet. Proto je z hlediska použitelnosti i v zastavěné oblasti pro realizaci výsledného modulu použita frekvence 433,92MHz.

2.3.2 Pásmo 433,05-434,79MHz

V ČR je pásmo 433,05-434,79 MHz využíváno pro průmyslové, vědecké a lékařské účely (industrial, scientific and medical - ISM) [12]. Toto pásmo je určeno pro amatérské aplikace k osobnímu využití a pro nespécifikované stanice krátkého dosahu (SRD). Provoz je možný na základě všeobecného oprávnění. Škodlivé rušení, které vzniká provozem těchto aplikací, musí být omezeno na minimum. Na provoz a koordinaci v tomto pásmu se vztahují ustanovení Radiokomunikačního řádu a ustanovení dohody HCM ²

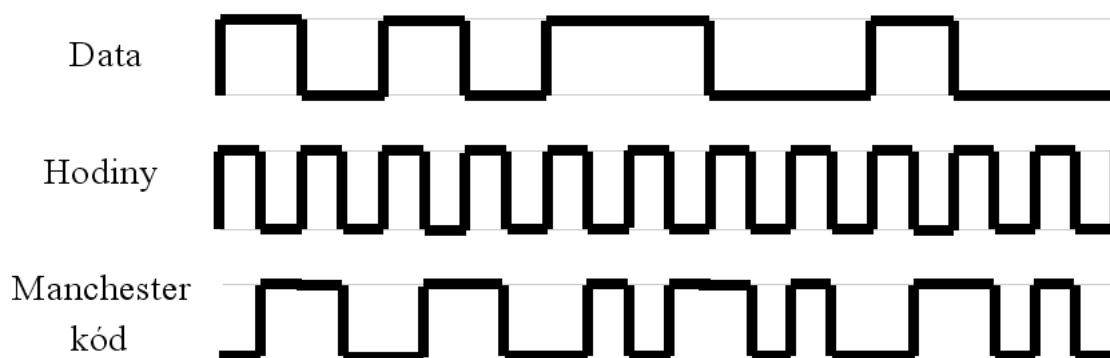
² Dohoda HCM – Dohoda mezi správami Rakouska, Belgie, České republiky, Německa, Francie, Maďarska, Nizozemí, Chorvatska, Itálie, Lichtenštejnska, Litvy, Lucemburska, Polska, Rumunska, Slovenska, Slovinska a Švýcarska o koordinaci kmitočtů mezi 29,7 MHz a 39,5 GHz pro pevnou službu a pozemní pohyblivou službu, Vilnius, 2005

2.4 Rádiové (hybridní) moduly

Též nazývané z angličtiny RF (Radio frequency) moduly. Zastávají funkci samotného vysílače, přijímače, či obou funkcí dohromady (tzv. transceiver). Na trhu je spousta RF modulů počínaje od jednoduchých s pár vývody, které pouze namodulují přijatý signál na nosnou vlnu, až po inteligentní moduly s desítkami vývodů, obsahujících vyrovnávací paměť typu FIFO, výkonový zesilovač, komunikaci s mikrokontrolérem po sériovém rozhraní, detekci slabého signálu, detekci slabé baterie ap. Ve většině případů tyto moduly používají frekvenční modulaci nebo také frekvenční kódování (FSK). Tento způsob kódování je založen na posunu vysílané frekvence o velikost dat od základní nosné frekvence. Tento rozdíl je pak na straně přijímače vyhodnocen jako data k přijetí a dále je s nimi pracováno.

2.5 Manchester kódování

Pro bezdrátový přenos je vhodné použít Manchester kód, který slučuje datový signál s hodinovým. Kódování Manchester je způsob zakódování dat, který se využívá pro přenos např. lokální počítačovou sítí (Ethernetem). Tento kód spojuje datový signál se synchronizačním signálem tak, že uprostřed délky každého bitu je hrana, která umožňuje synchronizaci na straně příjemce a současně tato hrana představuje hodnotu bitu: nástupná hrana znamená log. 1, sestupná hrana reprezentuje log. 0. Časový průběh tohoto kódování je naznačen na obr. 2.1. Nevýhoda manchester kódování je, že snižuje přenosovou rychlost na polovinu, protože pro přenos jednoho bitu jsou potřeba dva takty hodinového signálu.



Obr. 2.1: Časový průběh Manchester kódování

3 SÉRIOVÝ PŘENOS DAT

3.1 Forma přenášených dat

Přenos datového signálu rozdělujeme na dva základní druhy [2] [8]:

1. **Paralelní přenos** - data jsou přenášena po více bitech najednou (nejčastěji 8). K tomu je ovšem zapotřebí příslušný počet souběžných (paralelních) vodičů, to lze použít jen na krátké vzdálenosti (typicky 20 metrů). Paralelní přenos se používá např. při komunikaci mezi počítačem a tiskárnou přes paralelní rozhraní.
2. **Sériový přenos** - data jsou přenášena postupně bit po bitu. V převážné většině dnešních sítí je přenos dat sériový. Data jsou nejčastěji rozdělena po 8 bitech, tento shluk se pak nazývá znak (character). Tento způsob přenosu se dále rozděluje na:
 - **Asynchronní** - jednotlivé znaky mohou být posílány s libovolnými časovými odstupy, příjemce pak musí být informován o začátku jednotlivých znaků, k tomu slouží tzv. start-bit (též rozběhový prvek). Příklad start-bitu je pro příjemce současně i okamžikem pro správně nastavení vnitřních hodin, aby byly správně vyhodnoceny následující datové bity. Po těch následuje paritní bit (viz dále) a nakonec tzv. stop-bit (též závěrný prvek), který označuje konec vysílaného znaku.
 - **Synchronní** přenos - bity jednotlivých znaků následují těsně po sobě, nejsou prokládány žádnými start-bity či stop-bity (jsou však doplněny paritním bitem). Každý začátek bloku je indikován jedním nebo několika synchronizačními znaky, jejichž hlavním smyslem je zajistit časovou synchronizaci odesílatele i příjemce.

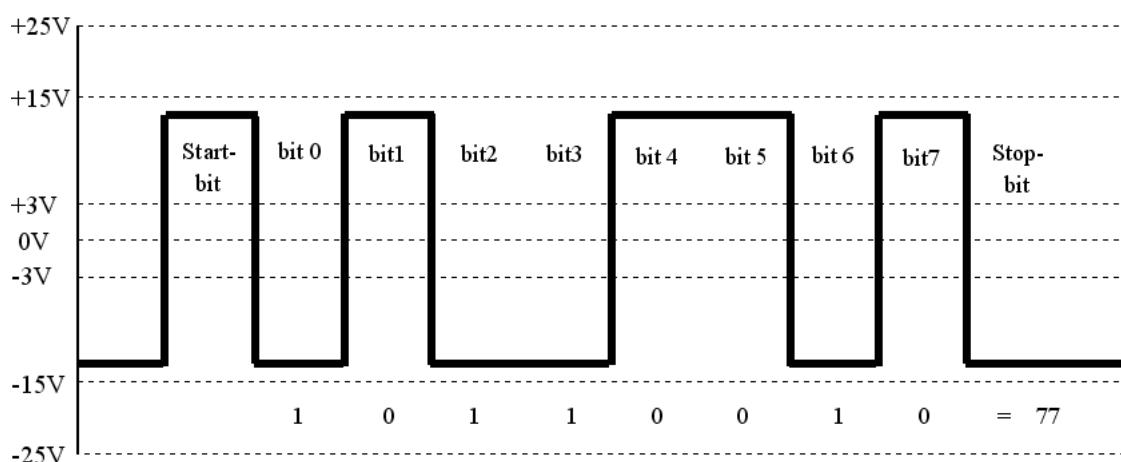
Paritní bit je nejjednodušší způsob zabezpečení jednotlivých znaků proti chybám. Datové bity jsou doplněny dalším bitem tak, aby celkový počet jedniček ve znaku byl lichý (pokud jde o lichou paritu), nebo sudý (pokud jde o sudou paritu). Příjemce musí vědět, zda jsou data zabezpečena lichou, nebo sudou paritou. Pokud má přijatý znak očekávanou paritu, není to ještě zárukou jeho bezchybnosti, nelze rozpoznat chyby v sudém počtu bitů. Více v [2].

3.2 Sériové rozhraní RS-232C

Data uvnitř počítače jsou přenášena paralelně, což je ale velmi nákladně při přenosu na větší vzdálenosti. Proto k propojení počítače s externí periferií se používá sériový přenos dat. Sériové rozhraní (RS232, V.24) [2] [8] [13] , bylo v roce 1962 vyvinuto kvůli potřebě komunikace dvou zařízení na krátkou vzdálenost do 20m. Přenos probíhá asynchronně, kdy synchronizace se provádí pomocí sestupné hrany startovacího impulsu. Obě strany musí mít stejně nastavenou rychlost přenosu aby si rozuměly. Tuto rychlost lze snížit a tím lze snížit chybovost na větší vzdálenosti.

Protokol RS232 používá dvě napěťové úrovně, odpovídající log. 1 a log. 0. Log. 1 je reprezentována -12 V a log. 0 je přenášena +12 V. Na přijímací straně je však rozsah napětí větší. Pro log. 1 je to -3 V až -25 V a pro log. 0 je to +3 V to +25 V. Je to z důvodu větší odolnost proti rušení, kdy může napětí na větší vzdálenosti klesnout i na polovinu. Časový průběh i s napěťovými úrovněmi je znázorněn na obr. 3.1. Jde o jeden znak, vyslaný sériovou linkou. Napěťový stav -3 V až +3 V není definován a přijímací strana ho ignoruje.

Pro přenos přes rozhraní RS232 se používá nejčastěji konektor CANNON 9. Najdeme ho dnes v každém počítači i jednodušších zařízeních. Tento konektor má 9 pinů, jejichž orientace je znázorněna. Popis těchto pinů naleznete v tab. 3.1. Nejdůležitější jsou piny č. 2 (RXD - Receive Data) a 3 (TXD - Transmit Data), přes které je zprostředkován vlastní přenos dat.



Obr. 3.1: Časový průběh vyslaného znaku M (ASCII 77) po sériové lince RS232

Tab. 3.1: popis pinů konektoru CANNON 9

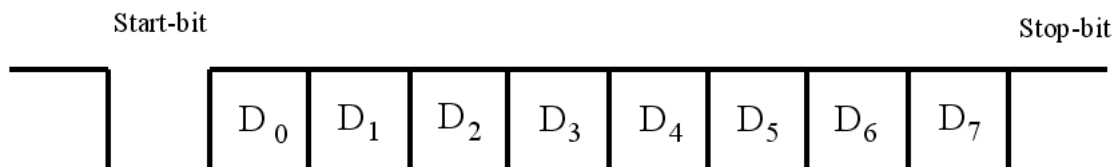
Číslo pinu	Název	Popis
1	CD	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	System Ground
6	DSR	Data Set Ready
7	RTS	Request to Send
8	CTS	Clear to Send
9	RI	Ring Indicator

3.3 Sériové rozhraní UART

Využívá stejný přenosový protokol, jako RS232, ale je určeno pro implementaci v mikroprocesorových obvodech pro komunikaci mikroprocesoru s okolím. Tento univerzální sériový přijímač/vysílač (UART) [6] [4] pracuje v asynchronním režimu. Podporuje duplexní provoz. Pro pohodlnější práci a komunikaci obsahuje tři typy vektorů přerušení:

- Vysílání dokončeno
- Vysílací vyrovnávací registr prázdný
- Příjem dokončen

Toto rozhraní je běžnou výbavou mikrokontrolérů Atmel AVR. Používá se převážně ke komunikaci s počítačem, který má sériové rozhraní RS232. Časový průběh je na obr. 3.2.

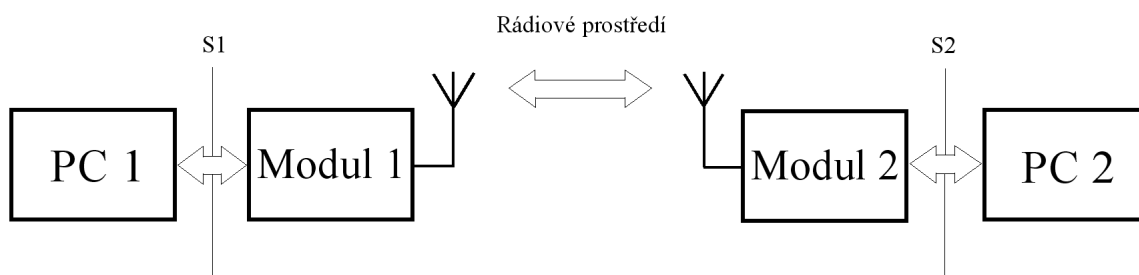


Obr. 3.2: Časový průběh jednoho znaku, vyslaného přes UART

4 NÁVRH KOMUNIKAČNÍHO MODULU

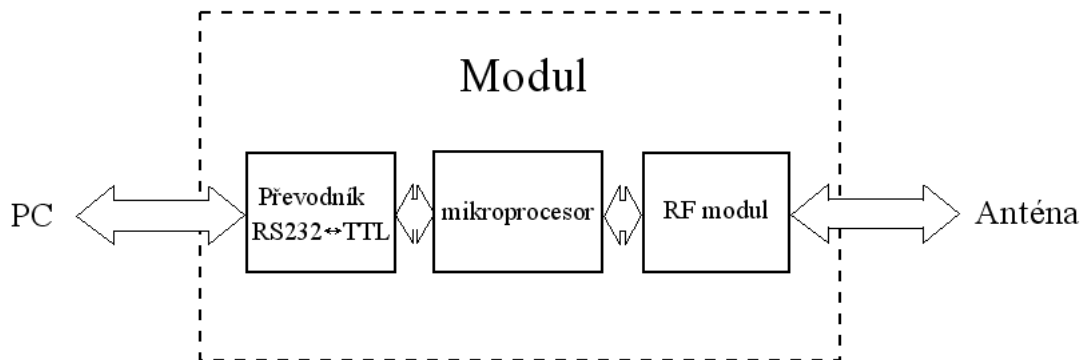
4.1 Blokové schéma

Cílem této práce je vytvořit funkční moduly, které budou zprostředkovávat bezdrátovou komunikaci a přenos dat. Tyto moduly budou tvořit rozhraní mezi bezdrátovým (rádiovým) prostředím a sériovým rozhraním počítače. Blokové schéma navrhované komunikace je znázorněno na obr. 4.1. Data, která přijme modul č.1 přes rozhraní S1 z PC1, zpracuje, aby byl možný jejich přenos bezdrátově, a pošle přes rádiové prostředí do modulu č.2, ten opět data převede, aby bylo možno je přenést přes rozhraní S2 do PC2. Komunikace bude obousměrná, proto moduly 1 i 2 budou obsahovat RF transceiver a mohou zároveň vysílat i přijímat data.



Obr. 4.1: Blokové schéma komunikace

Každý modul se bude skládat z převodníku, mikroprocesoru a RF modulu pro samotné vysílání a přijímání bezdrátových dat. Toto je znázorněno na obr. 4.2. Obvod mikroprocesoru bude vybaven resetovacím obvodem a LED diodami pro indikaci stavu. Zelená led dioda bude indikovat správný přenos dat a červená dioda bude oznamovat slabý signál nebo překročení maximální vzdálenosti dosahu.



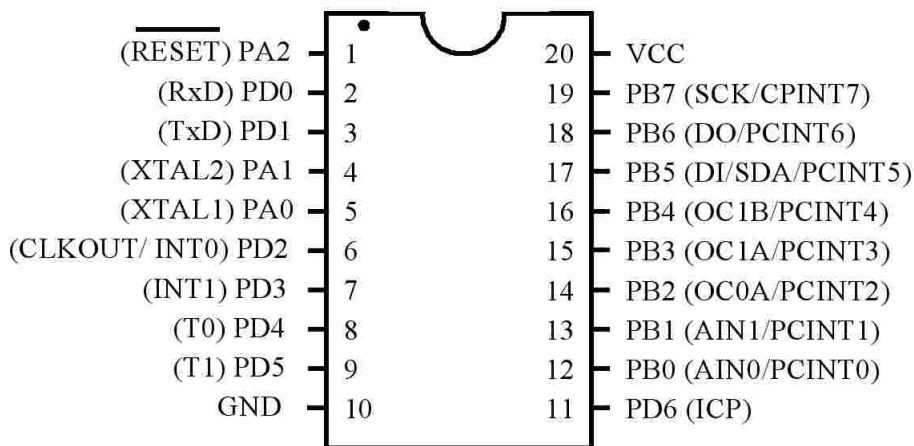
Obr. 4.2: Blokové schéma navrhovaného modulu

4.2 Výběr mikroprocesoru

Dle zadání bude použit mikrokontrolér AVR firmy Atmel [3] [5] [6]. Mikrokontroléry této rodiny se rozdělují na:

1. ATmega, které se vyznačují velkým počtem pinů a velkou pamětí
2. ATtiny, obecně mají od 8 do 20 pinů.

Tyto 8-bitové mikrořadiče jsou vybaveny RISC architekturou (zajistí vysoký výpočetní výkon, jelikož každá instrukce je prováděna v jednom taktu).



Obr. 4.3: Rozmístění pinů pouzdra PDIP mikrokontroléru ATtiny2313

V projektu je použit typ ATtiny2313 [14] v 20-ti pinovém pouzdře, které je vyobrazeno na obr. 4.3. Tento obvod má nízkou spotřebou, což je výhodné, protože zařízení budou napájena bateriově. Tento mikroprocesor má paměť programu

FLASH o velikosti 2kB, 128B paměti SRAM a 128B EEPROM. Instrukční soubor se skládá z 118 instrukcí [4]. V [4] je tento mikroprocesor pod označením AT90S2313, ten se již ale nevyrábí a je nahrazen kompatibilním obvodem ATtiny2313. Obsahuje sériové rozhraní UART, které je využito ke komunikaci přes rozhraní RS232 s přídatným zařízením. Obslužný program proto mikroprocesor je napsán v jazyce C, nejpoužívanějším programovacím jazyku k programování mikrořadičů Atmel AVR. Pro psaní kódu je využito vývojové prostředí CodevisionAVR (viz výše), které přeloží kód zapsaný v jazyce C a převede do hexadecimální formy, srozumitelné pro mikrokontrolér. Navíc lze přímo z tohoto prostředí přes hardwarový programátor mikrořadič naprogramovat. K tomu slouží protokol ISP (In system programming - programování přímo v zapojení). Tento protokol využívá piny mikrořadiče: /RESET, MOSI, MISO, SCK a pro napájení: VCC a GND viz obr. 4.3. Ve výsledné desce bude tedy vývod pro ISP konektor, který umožní sériové programování bez vyjmutí mikrokontroléru. MOSI, MISO a SCK jsou piny, které patří pod sériové rozhraní SPI:

SPI - serial peripheral interface je synchronní sériový port, který obsahuje většina mikrokontrolérů Atmel. Při komunikaci lze nastavit Master, či Slave zařízení, pořadí bitů. Přenos probíhá po dvou vodičích a další vodič je využit k přenosu hodin. Přenos je plně duplexní a lze současně posílat i přijímat znak. Více informací lze získat na [4].

4.3 Výběr RF modulu

Nyní máme na výběr, zda použijeme ve spojení s mikroprocesorem jednoduchý RF modul, který jen odesílá data na datovém vstupu, nebo použijeme jeden ze složitějších modulů, který obsahuje zásobník, obsahuje obvody k oznamování odeslání a přijetí znaku, slabého signálu ap. Tyto inteligentní moduly lze nastavovat přímo za provozu z mikroprocesoru většinou přes SPI rozhraní.

Pro výběr jednoduchého modulu hovoří nízká pořizovací cena, jednoduchá práce, není potřeba žádné nastavování a modul pracuje hned po zapojení. Stačí propojit přes jedno vedení datový pin a vysílat do modulu signál, který bude hned odeslán. Je však potřeba vytvořit datový tok pomocí mikroprocesoru a poté ho posílat do modulu. Pro vytvoření tohoto linkového protokolu je výhodné použít Manchester kódování, které se používá pro přenos, kde datový i hodinový signál je potřeba spojit do jednoho. Toto řešení je vhodné pro jednoduchá zařízení, kdy stačí jednoduché povely pro zapnutí či vypnutí zařízení na dálku. Všechna režie ovládání přenosu je na

programátorovi a algoritmu ovládacího programu mikroprocesoru a tím se zvyšuje náročnost realizace.

Výhoda inteligentního modulu je v tom, že obsahuje obvody pro automatické připravení dat k přenosu, obsahuje zásobník, a další obvody, které některé funkce, které by jinak musel vykonávat mikroprocesor a ovládací program tak bude jednodušší. Takovýto modul lze připojit k mikroprocesoru přes SPI port a celá komunikace probíhá přes tento port. Mikroprocesor může kontrolovat, zda je silný signál nebo zda jsou v zásobníku přijatá data atd. Tento modul je nutno před samotnou komunikací inicializovat a zapsat potřebné hodnoty do vnitřních registrů.

Nakonec byl k realizaci vybrán RF modul RFM12BP [15]. Jedná se o cenově dostupný transceiver s maximálním výstupním výkonem 500mW. V [15] lze najít seznam nejdůležitějších vlastností modulu. Tento modul je programovatelný přes rozhraní SPI (viz výše). Lze nastavit nosnou frekvenci, na kterou se modulují data (433, 868, 915MHz), dále lze nastavit frekvenční rozsah pro kódování, detekci slabého signálu a další. V tomto případě bude modul nastaven do přenosového pásma 433MHz. Přes SPI rozhraní se modul nastavuje na požadované hodnoty, zapnutí/vypnutí napájení obvodu transceiveru či receiveru a probíhá přes něj přenos znaku k bezdrátovému odeslání. Modul bude napájen napětím 3,3V, čemuž je potřeba upravit schéma zapojení a použít obvody, určené pro toto napětí. Modul navíc umožňuje měřit intenzitu přijímaného signálu, což lze využít pro indikaci překročení maximální vzdálenosti navrhovaných zařízení.

4.4 Vývojové prostředí CodevisionAVR

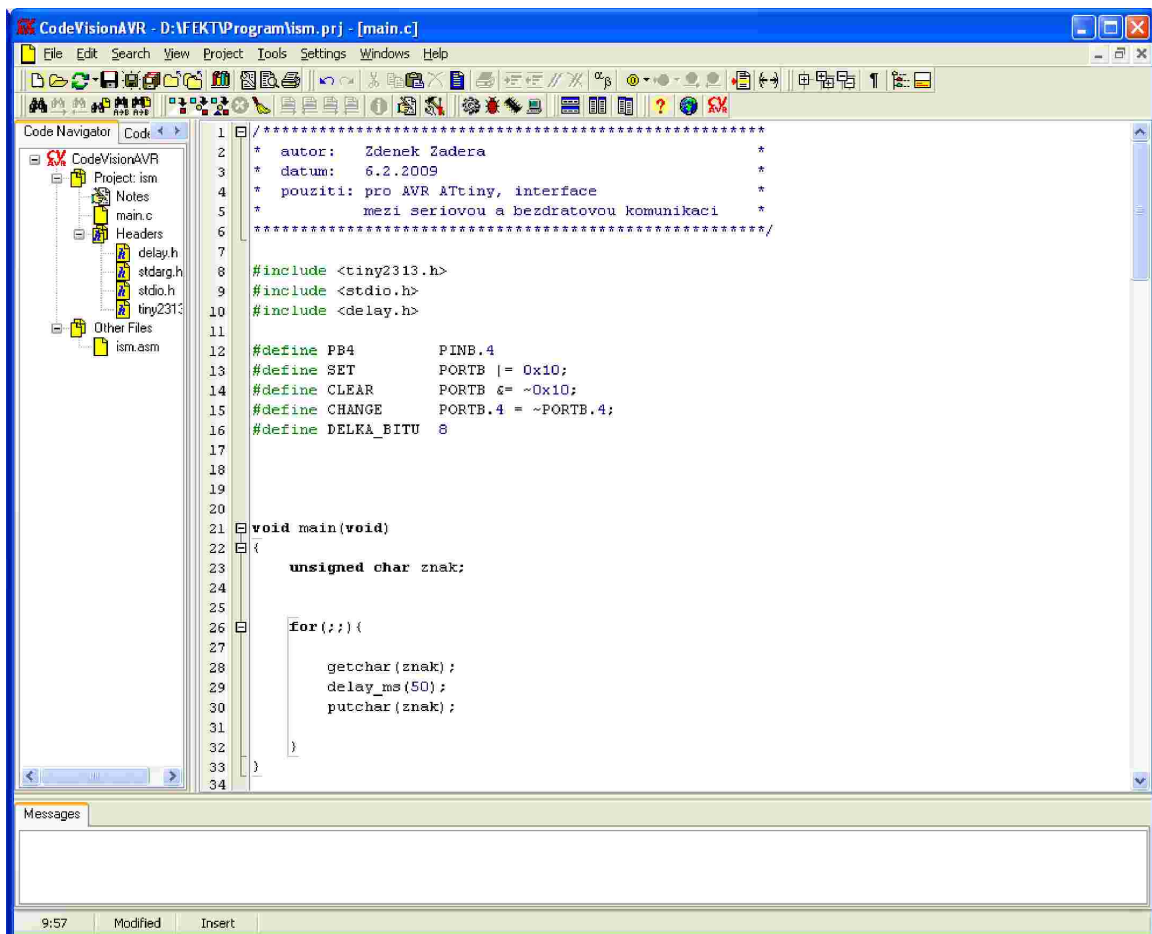
Ovládací program pro mikroprocesor bude napsán ve vývojovém prostředí CodeVisionAVR [3], je to výkonný kompilátor C pro rodinu mikrokontrolérů Atmel AVR. Obsahuje překladač jazyka C, integrovanéh vývojové prostředí, ke kterému lze připojit In-System programátor a průvodce pro rychlé vytvoření zdrojového souboru, bez znalostí programování. Jedná se o plnohodnotný vývojový systém pro mikroprocesory AtmelAVR, který vyhovuje ANSI-C. Obsahuje editor s automatickým odrážkováním a zvýrazněním syntaxe, umožňuje vkládání assembleru do zdrojového kódu C. Vestavěný sériový komunikační terminál RS232, RS422, RS485 umožňuje ladění aplikací. Obsahuje vestavěný programátor In-System pro AVR s automatickým programováním. Zdrojový kód je přeložen do .HEX souboru který je možné naprogramovat do mikroprocesoru i extrním programátorem. Kromě základních knihoven obsahují toto prostředí i knihovny jazyka C pro práci s:

- sběrnicí SPI,
- sběrnicí IIC,
- časovými prodlevami,
- nastavení I/O portů.

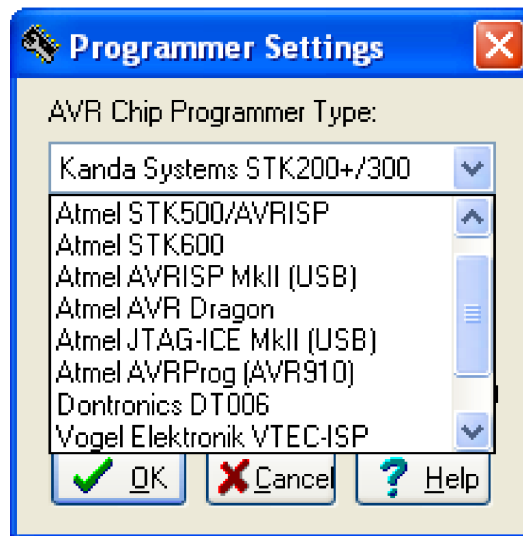
Vývojové prostředí CodevisionAVR vlastní firma HP InfoTech s.r.o. Nabízí dvě verze tohoto prostředí:

- CodeVisionAVR Tiny - komerční verze,
- CodeVisionAVR Evaluation - volně ke stažení, omezená velikostí kódu.

Verze CodeVisionAVR Evaluation je dostupná ke stažení z domovské stránky projektu (<http://www.hpinfotech.ro>) pro operační systémy Windows 95, Windows 98, Windows XP a 2000, tato verze je omezená velikostí kódu, který je možné kompilovat, na 2kB. Toto vývojové prostředí poskytuje uživatelský komfort s intuitivním ovládáním.



Obr. 4.4: Ukázka práce ve vývojovém prostředí CodevisionAVR



Obr. 4.5: Interní programátor nabízí spoustu programovacích protokolů

5 REALIZACE KOMUNIKAČNÍHO MODULU

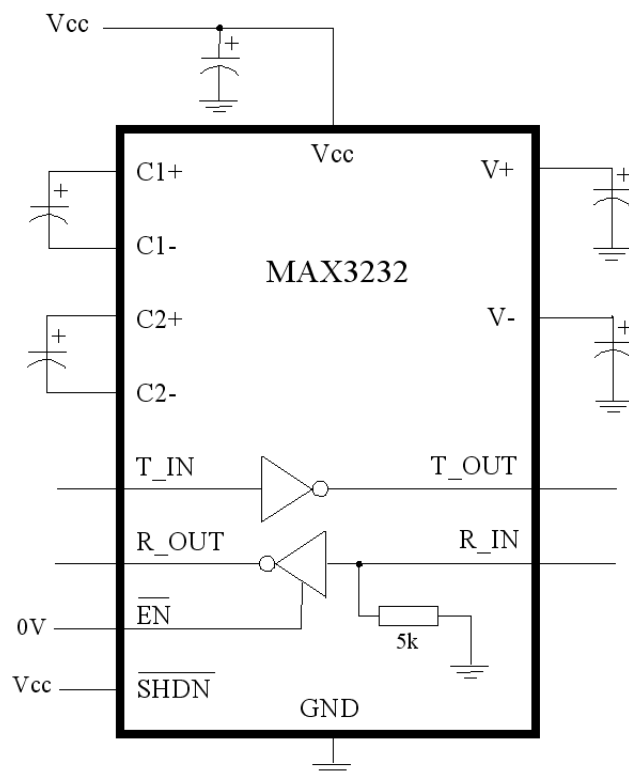
5.1 Schéma zapojení

Na obr. A.1 je navržené schéma zapojení. Toto schéma obsahuje mikrokontrolér, hybridní modul transceiveru RFM12BP, konektor ISP k připojení programátoru. RF modul je spojen s mikroprocesorem pomocí rozhraní SPI (MISO, MOSI, SCK), pro datovou komunikaci a dále piny PB3, PB4, které přenášejí signalizaci mezi těmito obvody.

Celý obvod je napájen napětím 3,3V, získaným pomocí napěťového stabilizátoru LE33CZ. Na jeho vstup se připojuje baterie, která má napětí větší, než zmíněné 3,3V plus úbytek napětí na stabilizátoru, v tomto případě 1,5V. Na vstupní svorky je tedy třeba přivést minimálně 4V. Stabilizátor má maximální výstupní proud 150mA. Při rozsvícení obou diod a zapnutém RX módu modulu je odběr asi 70mA. Při zapnutém TX módu je tento odběr větší, ale pro napájení výkonového zesilovače je použito přímo napětí baterie, takže stabilizátor není namáhán proudem výkonového zesilovače.

Obvod obsahuje dvě LED diody, připojené k pinům PB0 a PB1 mikrokontroléru. Tyto svítivé diody slouží k indikaci stavu k uživateli. Zelená dioda, připojená k PB1, signalizuje správný chod zařízení a svítí, když jsou přenášena data. Červená dioda na PB0 se rozsvítí při slabém nebo žádném přijímaném signálu a začne blikat při vybití baterii.

Dále obvod obsahuje čtyřpinový konektor k připojení převodníku mezi napěťovými úrovněmi mikroprocesoru a sériové linky RS232. Přes tento konektor se připojuje napájecí a zemnicí vodič, aby měl převodník stejné napětí jako mikroprocesor a dále obsahuje datové vodiče RXD a TXD, které jsou napojeny přímo k mikroprocesoru na piny č. 2 a č. 3. Obvod může tedy komunikovat s počítačem pomocí protokolu, určeného k tomuto rozhraní. Je tu však problém s nekompatibilitou napěťových úrovní. Napěťové úrovně na pinech Tx, Rx mikrokontroléru mohou nabývat hodnot 0 až 5V (TTL úrovně), napěťové úrovně na sériovém rozhraní počítače nabývají hodnot pro log 0 = +3 V až +25 V, pro log1 = -3 V až -25 V.



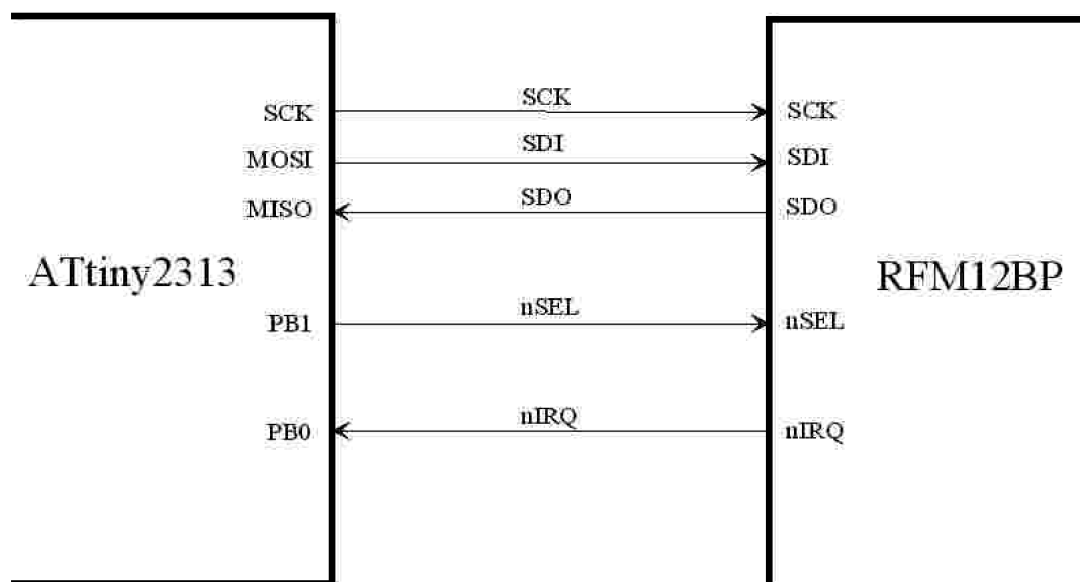
Obr. 5.1: Vnitřní zapojení obvodu MAX3232

Pro přizpůsobení těchto napěťových úrovní lze s úspěchem použít obvod MAX3232 firmy MAXIM. Je to obdoba známého obvodu MAX232, ale tato varianta je pro napájecí napětí od 3V. Zapojení tohoto obvodu je na obr. A.2. K obvodu jsou použity externí kondenzátory, jak je vidět na vnitřním zapojení obvodu na obr. 5.1. Nejdůležitější částí jsou vnitřní převodníky napěťových úrovní tj. $\pm 15V$ na 3V a naopak. Pomocí vstupu \overline{EN} lze řídit činnost obvodu, většinou se připojuje na 0V, to znamená, že obvod je pořád aktivní. Vstup \overline{SHDN} se používá k snížení potřeby tím, že se odpojí napájecí napětí, v tomto případě je však napevno připojen k napájecímu napětí.

5.2 Komunikace mikrokontroléru s modulem

Připojený RF modul je potřeba ovládat pomocí příkazů, posílaných z mikroprocesoru do tohoto modulu. Pomocí příkazů se zapisují informace do paměti modulu a tím dojde k nastavení požadované vlastnosti modulu. Komunikace mezi mikrokontrolérem a RF modulem probíhá přes standardní SPI rozhraní, toto rozhraní využívá tři vodiče, a dále pomocí dalších dvou vodičů, jak je znázorněno na obr. 5.2. SCK

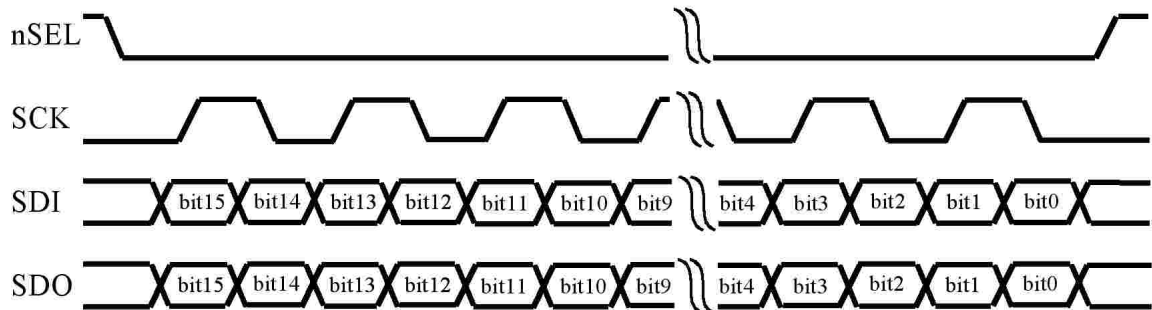
je hodinový signál a slouží k synchronizaci komunikace po linkách MOSI (SDI) a MISO (SDO). Po těchto linkách jsou přenášena 16-ti bitová slova pro posílání příkazů směrem k modulu a posílání hodnot zpět do mikrokontroléru. Linku nSEL ovládá mikroprocesor a sestupná hrana na této lince značí začátek posílání jednoho 16-ti bitového slova a tento stav trvá do odeslání celého slova. Linka nIRQ má opačný směr, tedy od modulu k mikroprocesoru a signalizuje pomocí vysoké logické úrovně (3V), že buffer na straně modulu je prázdný a mikrokontrolér může poslat další příkaz. Tímto se předchází tomu, aby mikrokontrolér přepsal předchozí příkaz, aniž by se stihl v modulu vykonat.



Obr. 5.2: Propojení mikrokontroléru s RF modulem

Jak je vidět na časovém průběhu na obr. 5.3, komunikaci začíná mikroprocesor sestupnou hranou linky nSEL. Popis signálů je uveden z pohledu RF modulu, SDI je přenášen směrem od mikroprocesoru do modulu a signál SDO naopak. Při nástupné hraně hodinového signálu SCK, dojde na straně modulu k přečtení hodnoty bitu SDI a v tutéž dobu dojde na straně mikroprocesoru k přečtení hodnoty bitu na lince SDO. Přenos probíhá sestupně od bitu 15 k bitu 0. Na konci přenosu 0. bitu mikroprocesor opět uvede linku nSEL do vysoké úrovně a tím naznačí modulu, že skončil přenos jednoho slova. Takto dojde k současnému přenosu 16-ti bitového slova na obě strany. Komunikace probíhá formou otázka-odpověď. Mikroprocesor vyšle příkaz a čeká na

odpověď od modulu.



Obr. 5.3: Časový průběh SPI komunikace

Po připojení napájení, nebo po resetu je potřeba modul inicializovat, to prování funkce `initRF` v obslužném programu. Obsahuje vlastně jen funkce, posílající šestnáctibitový příkaz do modulu. Pomocí těchto příkazů se nastaví správné parametry modulu. Nejdůležitější jsou například zapnutí vysílacího obvodu (TX), nebo zapnutí přijímacího obvodu (RX), dále je třeba nastavit parametry modulace a rozsah frekvenční modulace a další, více [15]. Po zavolání této funkce je modul připraven k práci, můžeme tedy posílat znaky k vysílání. Před každým vysláním dalšího znaku je však potřeba zkontrolovat, jestli předchozí znak byl odeslán. To značí nízká úroveň na lince NIRQ. Teprve poté může mikroprocesor do modulu poslat další znak. Po odeslání osmi bajtů je třeba poslat synchronizační dva bajty (2DD4), tím se započne vyslání dalšího rámce a přijímač se pomocí tohoto synchronizačního slova nastaví do režimu čekání na dalších osm bajtů.

Osmý byte v sobě nese informace o kontrolním součtu. Vysílač sečte jednotlivé předchozí byty rámce a tento součet pošle jako osmý byte. Na druhé straně přijímač také postupně sečítá přijaté byty a výsledek porovná s osmým bytem. Pokud srovnání nesedí, dojde k zahazení rámce.

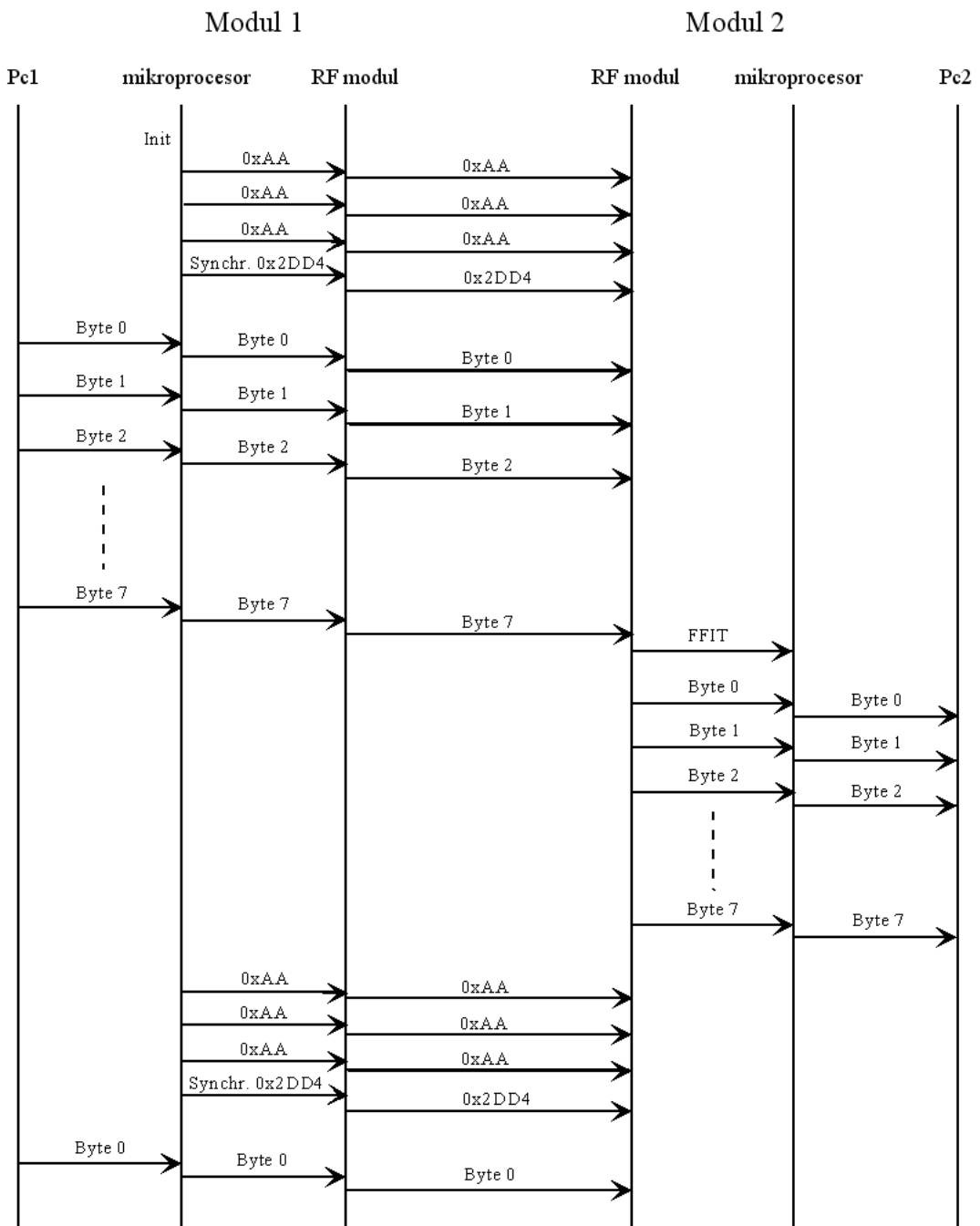
5.3 Přenosový diagram

Na linkové úrovni bude přenos probíhat po jednotlivých bytech (znacích). Rozhraní RS232 je standardizované a má předepsný přenosový protokol. Každý znak je uvozen startbitem a končí stopbitem, jak je popsáno výše. Na obr. 5.4 je zobrazen návrh přenosu v přenosovém diagramu. Pro zjednodušení je přenos uveden v jednom směru z PC1 do PC2.

Po zapnutí a inicializaci mikroprocesoru i modulu pošle mikroprocesor do modulu třikrát znak 0xAA jako preamble a po něm synchronizační slovo 0x2DD4, tyto znaky modul odešle. Poté čeká mikroprocesor na příjem znaku z PC1. Jakmile mikroprocesor přijme celý znak, zavolá se podprogram s obsluhou sériového kanálu a tento znak pošle do RF modulu, který je připojený na SPI (Serial peripheral interface - viz výše) port mikroprocesoru a modul tento znak odešle rádiovým prostředím.

Takto mikroprocesor vyšle postupně osm znaků přes SPI port do RF modulu a poté vyšle opět tři znaky 0xAA a synchronizační slovo, aby se mohl modul na příjímací straně správně nastavit na příjem dalších osmi znaků. Takto přenos probíhá po jednotlivých rámcích, každý rámeček obsahuje osm znaků.

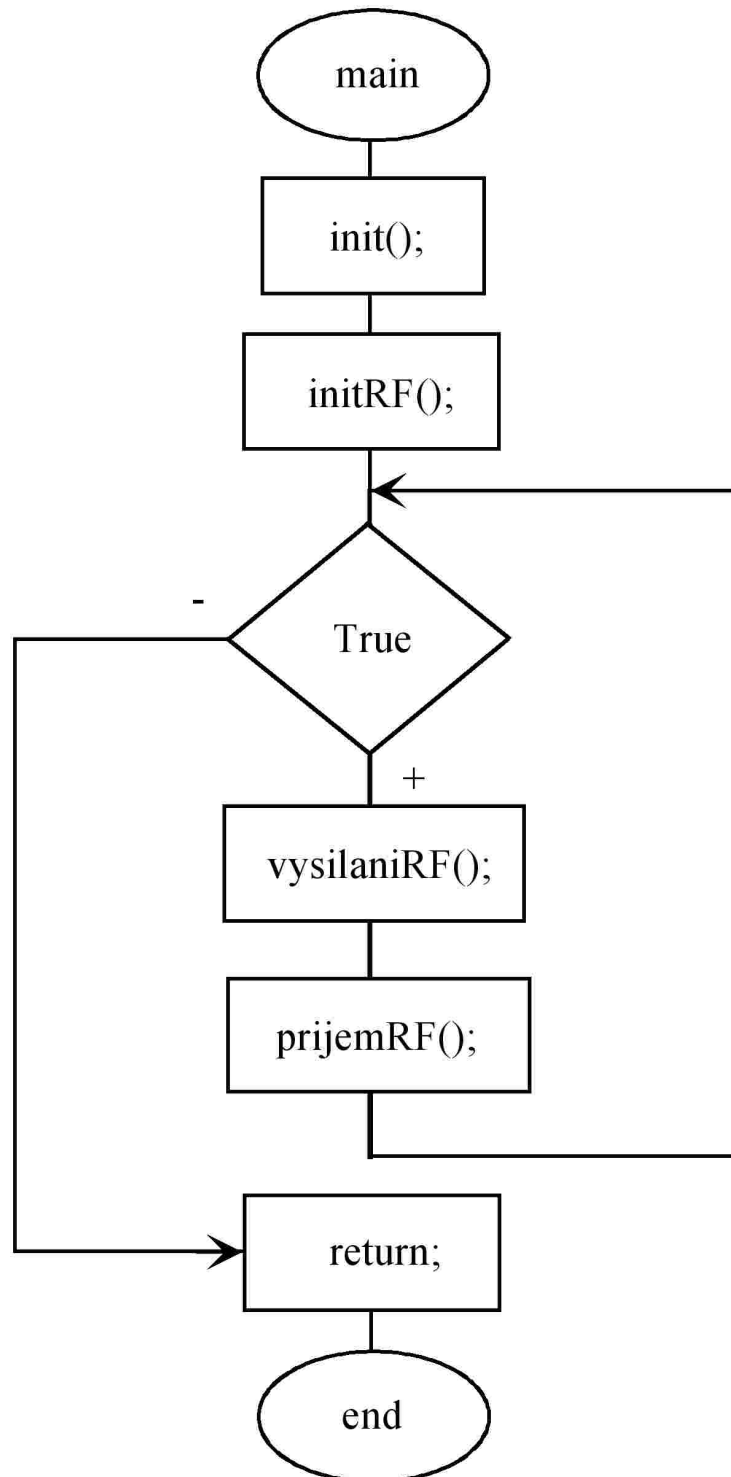
Na druhé straně mikroprocesor čeká na příjem rámce. Jakmile se naplní zásobník v RF modulu osmým znakem, modul nastaví příznak FFIT na log. 1, což značí, že zásobník se naplnil a mikroprocesor začne postupně číst ze zásobníku znaky. Po přečtení osmi znaků vymaže zásobník v RF modulu a čeká na příjem dalšího rámce. Toto se opakuje v nekonečném cyklu.



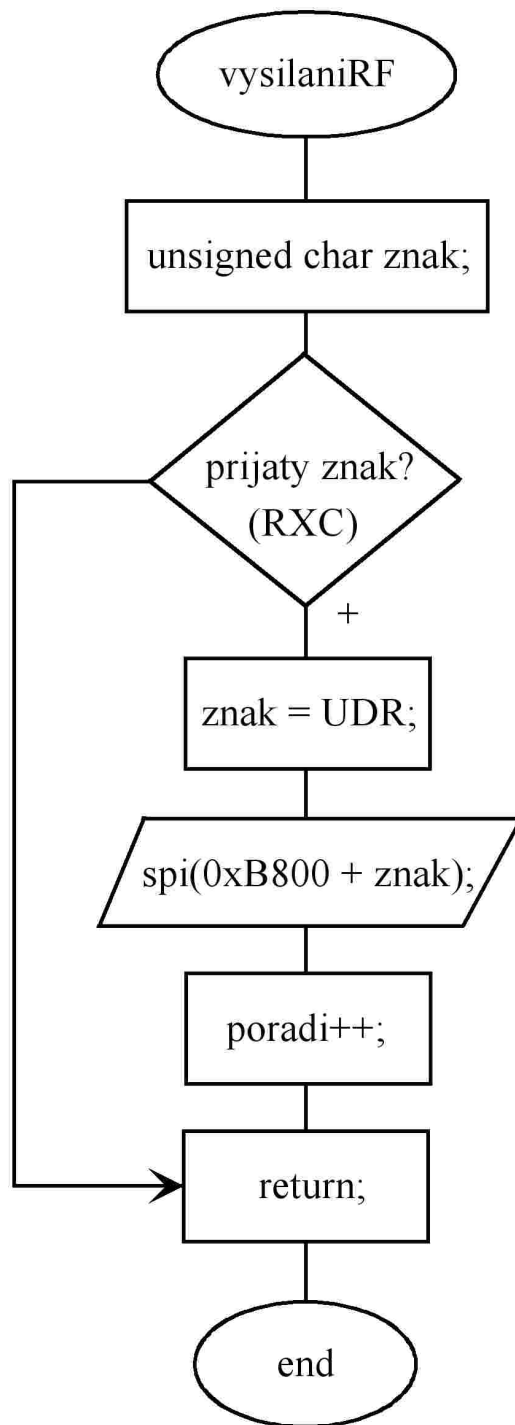
Obr. 5.4: Přenosový diagram komunikace mezi moduly

5.4 Obslužný program

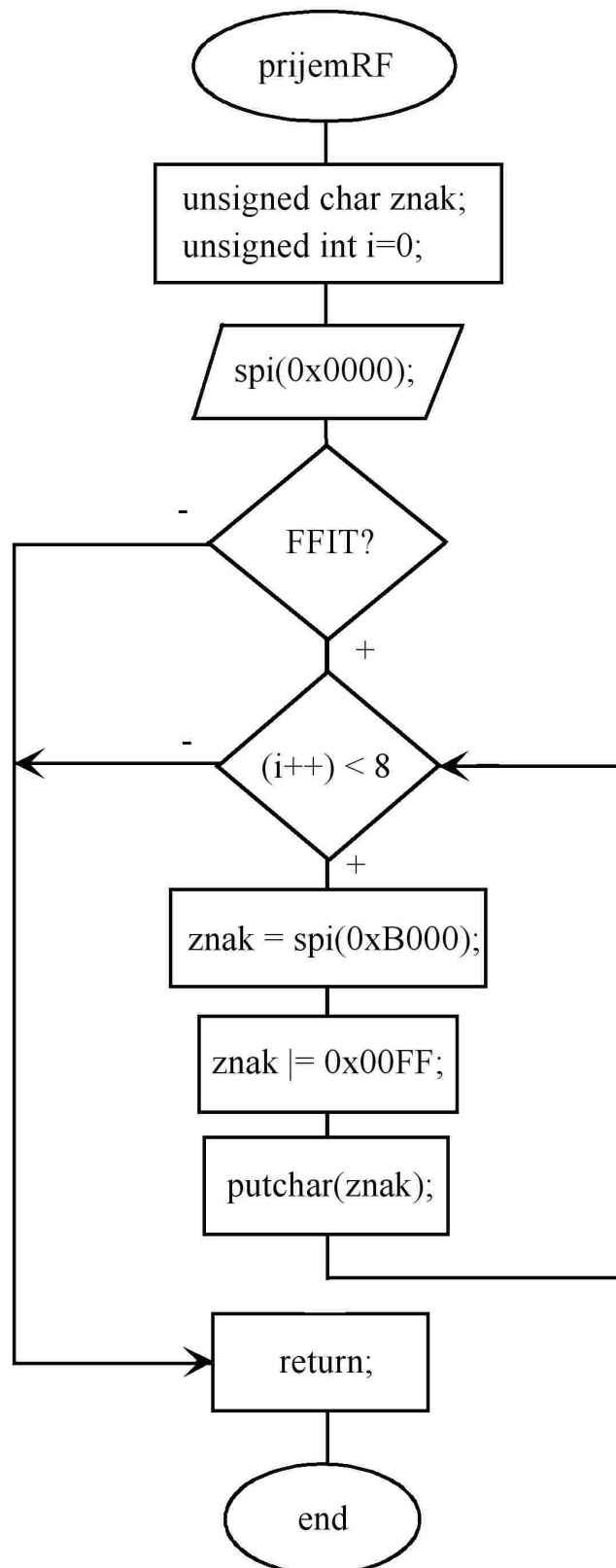
5.4.1 Vývojový diagram



Obr. 5.5: Vývojový diagram hlavní funkce main



Obr. 5.6: Vývojový diagram funkce vysilaniRF



Obr. 5.7: Vývojový diagram funkce prijemRF

5.4.2 Popis algoritmu

Obslužný program mikroprocesoru je napsán v jazyce C pomocí vývojového prostředí CodevisionAVR. Na začátku programu je třeba vložení hlavičkových souborů, potřebných pro práci s mikroprocesorem ATtiny2313 soubor `tiny2313.h` a pro práci se standardním rozhraním UART soubor `stdio.h`.

Poté je třeba nedefinovat makra pro práci s programem, tato makra zpřehledňují výsledná zdrojový kód a patří k dobrému programátorskému stylu. Makra `NASTAV` a `VYNULUJ` slouží k nastavení logické úrovně na daný výstupní pin. Makro `CEKEJ_NIRQ` obsahuje smyčku `while` s podmínkou na začátku, která se vykonává tak dlouho, dokud na lince `nIRQ` je vysoká logická úroveň. Toho je později v programu využito pro čekání, dokud nebude znak modulem odeslán, což je signalizováno nízkou úrovní na lince `nIRQ`.

Následuje deklarace globálních proměnných, které jsou využívány jednotlivými funkcemi a je do nich zapisováno, kolikrát která funkce proběhla a také kontrolní součty přijatých a odeslaných znaků.

Dále už jsou deklarace hlaviček použitých funkcí. Jako první je funkce `init()`, tato funkce zajišťuje nastavení všech parametrů mikroprocesoru, nastavení vstupně výstupních pinů, aby bylo možno komunikovat s okolím. Také je potřeba nastavit vlastnosti sériového rozhraní UART, jako rychlost přenosu, jestli bude použito paritního bitu, či kolik bitů bude použito pro komunikaci. Poté co je nastaven mikroprocesor, je potřeba inicializovat modul RF, aby pracoval v požadované frekvenční oblasti. Toto nastavení obsahuje funkce `initRF()`, která dále nastavuje parametry modulu, jako modulační frekvenční rozsah, velikost FIFO zásobníku, různá přerušení, rychlost komunikace modulu s mikroprocesorem, použití vnitřního krystalu a další.

Funkce `posliPrikaz()` slouží k odeslání 16-ti bitového slova do registru modulu přes SPI linku a vrací slovo přijaté z modulu. Další funkce `vysliZnak()` využívá předchozí funkci k zapsání jednoho znaku, určeného k vyslání, na adresu `0xB800`, čímž modul tento znak odešle.

Funkce `vysilaniRF` volá předchozí funkci, pokud je přijat znak na lince UART, a po odeslání osmi znaků pošle synchronizační slovo, aby se mohl přijímač opět správně nastavit. A toto se děje stále dokola, pokud jsou data k odeslání.

Naopak funkce `prijemRF()` kontroluje, jestli je nastaven příznak `FFIT` ve stavo-

vém slově, přijatém z modulu, jako odpověď na příkaz 0x0000 . Pokud je příznak FFIT nastaven, tak mikroprocesor si přečte přijatá data, uložená v FIFO zásobníku. To provede pomocí zaslání příkazu 0xB000 a modul na tento příkaz odpoví zasláním znaku z vrchu FIFO zásobníku a tak dále, dokud není přečteno všech osm znaků. Pokud není nastaven příznak FFIT, pak mikroprocesor zkontroluje další příznak tohoto stavového slova, který modul nastaví, pokud je slabý přijímaný signál.

Na konec následuje hlavní funkce *main()*. Tato funkce je vždy v jazyce C volána jako první a je vykonávána až do ukončení programu. Z této funkce jsou volány všechny předešlé obslužné funkce. Na začátku jsou nadefinovány proměnné a po nich příkazem *asm(cli)* je zakázáno přerušování. Poté je zavolána funkce *init()*, kdy dojde k nastavení mikrokontroléru, aby se následně mohl nastavit i modul zavoláním funkce *initRF()*. Další kód se stará o indikaci zapnutí uživateli pomocí rozsvícení obou diod na dobu danou počtem opakování do sebe vnořených dvou cyklů *for()*. Po tomto kroku je třeba vynulovat a povolit FIFO zásobník modulu, aby mohla být do něho ukládána data přijatá RF modulem. Poté následuje nekonečná smyčka, která je vytvořena pomocí cyklu *for()* bez parametrů, v této smyčce jsou volány funkce *vysilaniRF()* a *prijemRF()*. Tato smyčka se opakuje, dokud není modul vypnut nebo není aktivován odvod resetu tlačítkem T1, kdy se program začne vykonávat opět od začátku.

5.4.3 Zdrojový kód

```
/*
 * autor: Zdenek Zadera
 * datum: 20.5.2009
 * pouziti: pro AVR ATtiny, interface
 *          mezi rozhraním RS232 a RFM12BP
 */
*****/

#include <tiny2313.h>
#include <stdio.h>

#define RED          0
#define GREEN        1
#define SEL          3
#define NIRQ         4
#define SDI          5
#define SDO          6
#define SCK          7
#define RXC          7

#define NASTAV(pin)  PORTB |= (1<<(pin))
#define VYNULUJ(pin) PORTB &= ~(1<<(pin))
#define CEKEJ_NIRQ  while(PINB&(1<<NIRQ))

unsigned char soucet_tx=0;
unsigned char soucet_rx=0;
unsigned char poradi=0;
unsigned char pamet[8];

//_____ hlavicky pouzitych funkci _____

void init();
void initRF();
unsigned int posliPrikaz(unsigned int prikaz);
void vysliZnak(unsigned char znak);
unsigned int prijmiZnak();
void vysilaniRF();
void prijemRF();
```

```

//----- funkce main() -----
void main(void)
{
    unsigned int i,j;

    #asm("cli")

    init();
    initRF();

    NASTAV(RED); //
    NASTAV(GREEN); //
    for(j=0;j<16;j++) // indikace zapnuti
        for(i=0;i<30000;i++); //
    VYNULUJ(RED); //
    VYNULUJ(GREEN); //

    posliPrikaz(0xCA81); // nastavit FIFO
    posliPrikaz(0xCA83); // povolit FIFO

    for(;;){

        vysilaniRF();
        prijemRF();
    }
}

```

6 ZÁVĚR

Výsledkem této práce je návrh a realizace komunikačních modulů pro obousměrnou vzájemnou komunikaci v pásmu ISM (Industrial, Scientific and Medical). Je použito frekvenční pásmo 433,92MHz, toto pásmo je méně náchylné na překážky v rádiovém prostředí a výhoda je také snadnější dostupnost modulů. Pro samotný přenos je použita dvojice transceiverů RFM12BP, tyto transceivery jsou ovládány mikrokontroléry ATtiny2313, které také slouží ke zpracovávání přenášených dat a upozorňování uživatele o funkci celého modulu pomocí svítivých LED diod. Frekvence mikrokontrolérů je určena použitým krystalem 3,578MHz. Mikrokontroléry jsou vybaveny obslužným programem, napsaným v jazyce C v prostředí Codevisio-nAVR.

Oba moduly jsou identické, mohou vysílat i přijímat data, proto je jejich návrh popsán jedním postupem. Dosah modulů ve volném prostředí bez překážek může být až 1km při použití kvalitní antény délky 17cm. Pokud je místo antény použit pouze vodič dané délky, tak se dosažená vzdálenost snižuje. V první části přílohy této práce je navrhované schéma zapojení a v druhé části přílohy je návrh desky plošných spojů i s osazením součástek.

Schéma je navrženo s možností programování pomocí ISP (In System Programming), aniž by byl vyjmut mikrokontrolér z desky. Modul umožňuje komunikovat přes rozhraní RS232 s nadřazeným počítačem. K tomu je vyroben přídatný převodník pro komunikaci mikroprocesoru s počítačem, jehož návrh je součástí práce.

Ve schématu je ponechána možnost komunikovat přes jiné rozhraní, než použité RS232, stačí jen nahradit navrhnutý převodník za převodník např. modernějšího rozhraní USB. Dále je možnost doplnit zařízení o počítačový program, se kterým by tyto moduly mohly efektivněji spolupracovat.

LITERATURA

- [1] MANN, B., *C pro mikrokontroléry*, Praha: BEN, 2003. ISBN 80-7300-077-6
- [2] Kainka, B., *Měření, řízení a regulace pomocí PC*, BEN, 2003. ISBN 80-7300-089-X
- [3] Váňa, V., *Mikrokontroléry ATMEL AVR - Programování v jazyce C : popis a práce ve vývojovém prostředí CodeVisionAVR*, Praha: BEN, 2003. ISBN 80-7300-102-0
- [4] Váňa, V., *Mikrokontroléry ATMEL AVR - Popis procesorů a instrukční soubor*, Praha: BEN, 2003. ISBN 80-7300-083-0
- [5] Matoušek, D., *C pro mikrokontroléry ATMEL AT89S52*, Praha: BEN, 2007. ISBN 978-80-7300-215-2
- [6] Matoušek, D., *Práce s mikrokontroléry ATMEL. 3. díl, AVR AT90S*, Praha: BEN, 2003. ISBN 80-7300-088-1
- [7] Šubrt, V., *Mikrokontroléry ATMEL AVR : vývoj aplikací*, Praha: BEN, 2002. ISBN 80-7300-055-5
- [8] Vlach, J., *Počítačová rozhraní - přenos dat a řídicí systémy*, Praha: BEN, 2000. ISBN 80-7300-010-5
- [9] Hrbáček, J., *Komunikace mikrokontroléru s okolím 1*, Praha: BEN, 2000. ISBN 80-86056-36-8
- [10] Hrbáček, J., *Komunikace mikrokontroléru s okolím - 2.díl*, Praha: BEN, 2002. ISBN 80-86056-73-2
- [11] Hanus, S., *Bezdrátové a mobilní komunikace*, Ústav radioelektroniky FEKT VUT, Brno 2002
- [12] ČTÚ, *Telekomunikační věstník* [online], dostupný z WWW: http://www.ctu.cz/1/download/telekomunikacni_vestnik_09_2006_1145352180.pdf?PHPSESSID=7 , 10.2.2009
- [13] *HW internetový server: sériová linka RS-232* [online], dostupný z WWW: <http://hw.cz/rs-232> , 4.3.2009
- [14] *Datasheet ATtiny2313* [online], dostupný z WWW: <http://pdf1.alldatasheet.com/datasheet-pdf/view/80317/ATMEL/ATTINY2313.html> , 2.5.2009

- [15] *Datasheet RFM12BP firmy HopeRF* [online],
dostupný z WWW: <http://www.hoperf.com/pdf/rfm12bp.pdf> , 20.5.2009
- [16] *Wikipedia.org - ISM Band* [online],
Dostupný z WWW: http://en.wikipedia.org/wiki/ISM_band
- [17] AUTOMA, *Bezdrátové komunikace v automatizační praxi I: historie a současnost* [online], dostupný z WWW:
http://www.odbornecasopisy.cz/index.php?id_document=28818 ,
15.5.2009

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

ITU Internatinoal Telcommunication Union

ITU-R ITU Radiocommunication Sector

ČTÚ Český telekomunikační úřad

ISM Industrial, Scientific and Medical - frekvenční pásmo pro soukromý bezdrátový přenos

SPI Serial peripheral interface

UART Universal asynchronous receiver/transceiver

IEEE Institute of Electrical and Electronics Engineers

LAN Local Area Network - je tvořena např. firemní sítí krátkého dosahu

WAN Wide Area Network - pokrývá rozsáhlou plochu (překračuje hranice města, regionu nebo státu)

WAN Personal Area Network

Wifi Wireless Fidelity

FM Frequency Modulation

AM Amplitude Modulation

GSM Global System for Mobile Comunication

GPRS General Packet Radio Service

HSCSD High Speed Circuit Switched Data

UMTS Universal Mobile Telephone System

FSK Frequency Shift Keying

RF Radion frequency

FSK Frequency Shift Keying

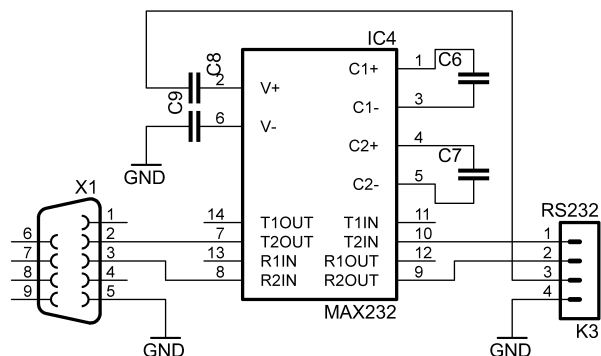
AVR Advanced Virtual RISC - označení mikroprocesorů firmy Atmel

MCU Machine Control Unit

SEZNAM PŘÍLOH

A Schéma	44
A.1 Schéma zapojení	44
A.2 Schéma převodníku	45
B DPS	46
B.1 Desky plošných spojů ze strany součástek	46
B.2 Osazení desek plošných spojů	47

A.2 Schéma převodníku



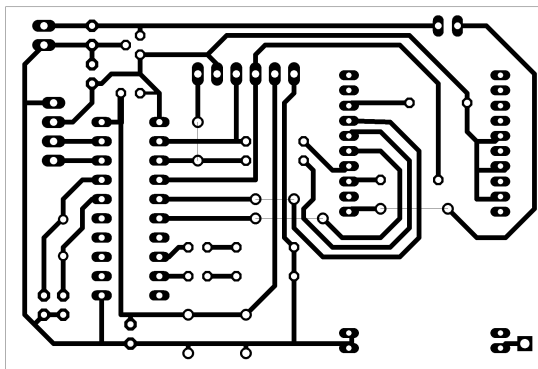
Obr. A.2: Zapojení převodníku sériového rozhraní na úrovni TTL s obvodem MAX3232

Seznam součástek

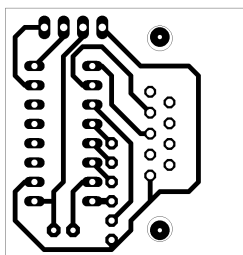
R1,R2	70 Ω
R3	4.7 k Ω
R4	4.7 k Ω
R5	4.7 k Ω
R6	10 k Ω
R7	4.7 k Ω
C1	100 nF
C2	22 pF
C3	22 pF
C4	10 μ F
C5	100 nF
C6,C7,C8,C9	10 μ F/tantalový
LED1	LED dioda, 3mm, červená
LED2	LED dioda, 3mm, zelená
IC1	AVR ATtiny2313
IC22	RFM12BP
IC3	LE33CZ
IC4	MAX3232
Q1	krystal 3.578 MHz
JP1	jumper, černý
K1	kolíková lišta 6 pinů
K2	kolíková lišta 4 piny
K3	konektor 2-pinový
X1	CANNON 9

B DPS

B.1 Desky plošných spojů ze strany součástek

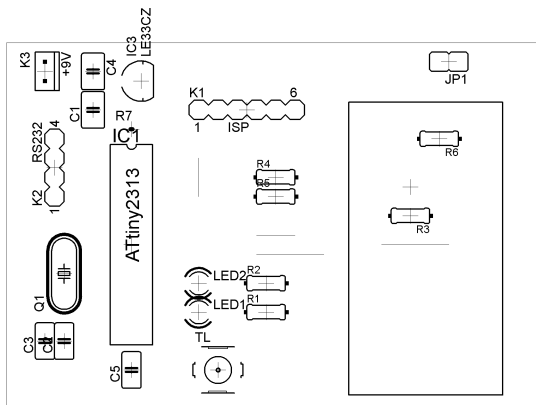


Obr. B.1: Deska plošných spojů ze strany součástek

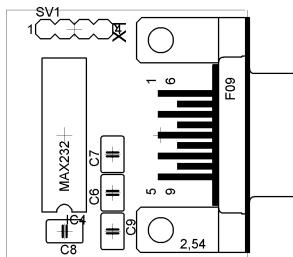


Obr. B.2: Deska plošných spojů převodníku ze strany součástek

B.2 Osazení desek plošných spojů



Obr. B.3: Osazení desky plošných spojů



Obr. B.4: Osazení desky plošných spojů převodníku