

Prohlášení

Prohlašuji, že jsem svoji diplomovou práci na téma Komunikační systém pro nevidomé – přenosná část, vypracoval samostatně pod vedením vedoucího semestrálního projektu 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu 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 trestně právních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 30. 5. 2008

Podpis autora: _____

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Ing. Ivo Hermanovi CSc. za velmi užitečnou metodickou pomoc, cenné rady a v neposlední řadě za poskytnutí technického zázemí při zpracování diplomové práce. Také bych rád poděkoval panu Jaromíru Hamalovi za zapůjčení PDA a jinou technickou pomoc.

SEZNAM ZKRATEK

A/D	<i>Analogový/Digitální</i>
ALU	<i>Arithmetic Logic Unit</i>
ASK	<i>Amplitude Shift Keying</i>
AVR	<i>Advanced Virtual RISC</i>
CMOS	<i>Complementary Metal Oxide Semiconductor</i>
DSL	<i>Digital Subscriber Loop (Digital Subscriber Line)</i>
DSR	<i>Data Set Ready</i>
DTR	<i>Data Terminal Ready</i>
EEPROM	<i>Electrically Erasable Programmable Read-Only Memory</i>
GND	<i>Ground</i>
LPT	<i>Line Print Terminal</i>
PDA	<i>Personal Digital Assistant</i>
RISC	<i>Reduced Instruction Set Computing</i>
RTS	<i>Request To Send</i>
SRAM	<i>Static Random Access Memory</i>
SMD	<i>Surface Mounted Device</i>
TTL	<i>Transistor–Transistor Logic</i>
USART	<i>The Universal Synchronous and Asynchronous Serial Receiver</i>

SEZNAM OBRÁZKŮ

- Obr. 2.1: Blokové schéma komunikačního zařízení
- Obr. 3.1: Vnitřní uspořádání obvodu TDA5051
- Obr. 3.2: Vztah průběhů mezi výstupem a vstupem dle katalogového listu
- Obr. 3.3: Formát signálu pro obvod ATA5282
- Obr. 3.4: Zjednodušené vnitřní zapojení obvodu AD8018
- Obr. 3.5: Zapojení obvodu TDA5051 se zesilovačem
- Obr. 3.6: Naměřený vztah mezi vstupem a výstupem obvodu TDA5051
- Obr. 3.7: Zapojení hlavního konektoru modulu
- Obr. 3.8: Formát datového paketu
- Obr. 3.9: Vnitřní uspořádání procesoru ATmega8
- Obr. 3.10: Zapojení programátoru mikroprocesoru
- Obr. 3.11: Zapojení kontaktů bluetooth modulu
- Obr. 3.12: Aplikace Navigátor 1.0
- Obr. 4.1: Celkové schéma zapojení komunikačního zařízení
- Obr. 4.2: Deska plošných spojů, spodní strana
- Obr. 4.3: Vrchní strana desky plošných spojů s rozmístěním součástek

Obsah

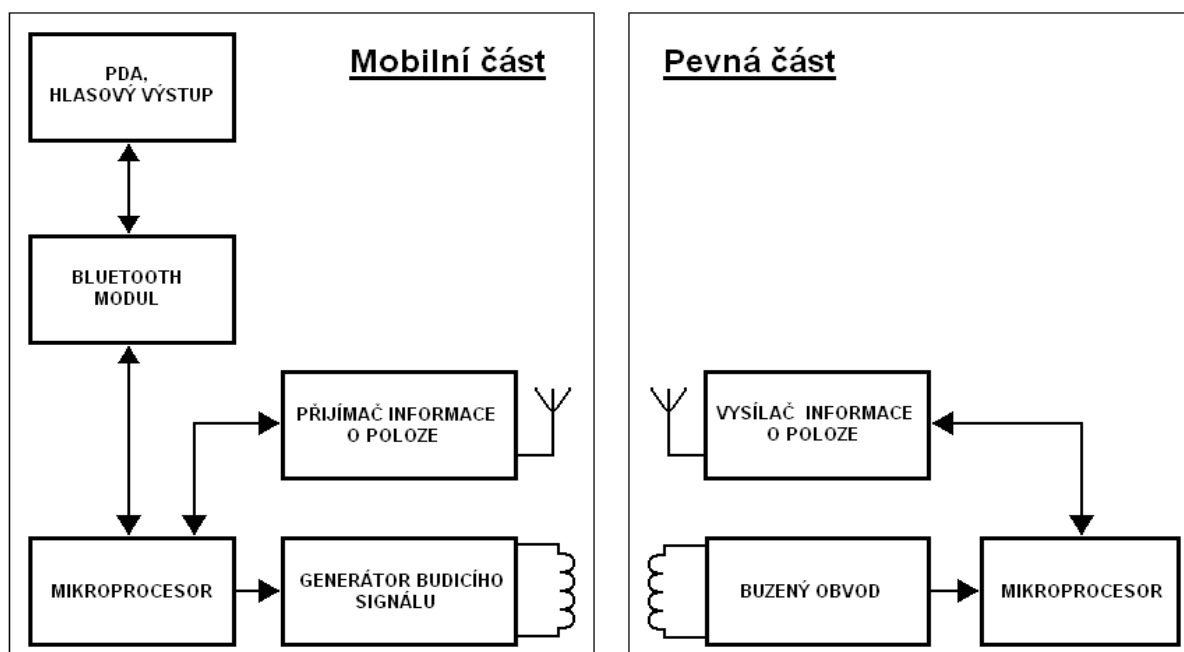
1 Úvod.....	11
2 Popis funkce zařízení.....	12
2.1 Popis jednotlivých bloků přenosné části systému.....	13
2.2 Popis jednotlivých bloků pevné části systému.....	14
3 Řešení jednotlivých bloků přenosné části.....	14
3.1 Návrh bloku generátoru budicího signálu.....	14
3.1.1 Popis obvodu TDA5051.....	15
3.1.2 Popis funkce obvodu.....	16
3.2 Přijímač vysílání o poloze.....	20
3.2.1 Radiokomunikační modul ARF10.....	20
3.2.2 Komunikační protokol.....	22
3.3 Mikroprocesor.....	23
3.3.1 Popis mikrokontroléru ATmega8.....	23
3.3.1.1 Modul časovače.....	25
3.3.1.2 Modul externího přerušení.....	25
3.3.1.3 Modul sériového rozhraní USART.....	26
3.3.1.4 Nastavení pojistek.....	28
3.3.1.5 Programování mikroprocesoru.....	29
3.3 Bluetooth modul.....	29
3.3.1 Popis modulu cB-0907-01	30
3.3.2 Připojení BT modulu.....	31
3.4 PDA, hlasový výstup.....	32
3.4.1 Vytvoření aplikace pro PDA.....	32
4 Propojení všech částí.....	35
5 Závěr.....	38
Seznam použité literatury.....	39

1 Úvod

Cílem této diplomové práce je návrh zařízení, které by usnadňovalo nevidomým orientaci v rámci veřejných budov, jako jsou například úřady, školy a další. Zařízení má za úkol v jednotlivých částech budovy oznamovat nevidomému jeho aktuální polohu. Úkolem zařízení však není snímat polohu nevidomého neustále a oznamovat mu tuto informaci v každém bodě trasy, nýbrž usnadňovat mu orientaci v objektu pomocí definovaných bodů. Těmito body budou místa důležitá z hlediska orientace, jako například vstupy (dveře) do jednotlivých místností budovy, místa, kde se rozděluje chodba na dvě části a podobně. Nevidomý tak může být informován například o čísle dveří, u kterých se právě nachází. Celý systém bude sestávat ze dvou částí, z nichž jedna bude umístěna pod povrchem podlahy a bude sloužit jako pevný, aktivní orientační bod, který na vyžádání vysílá informace o své poloze. Druhá část bude umístěna přímo na slepecké hůlce. Pokud nevidomý při chůzi přejede hůlkou přes vysílač v podlaze, vyšle vysílač do hůlky signál, který potom pomocí výstupního zařízení (např. PDA) informuje nevidomého o jeho poloze. Vedoucím diplomové práce mi bylo naznačeno přibližné řešení problematiky, byly mi taktéž doporučeny některé integrované obvody, které je vhodné použít. Protože se zabývám pouze návrhem přenosné části (návrh pevné části je úkolem jiného studenta), budu se zabývat popisem pevné části jen v té míře, aby mohlo být správně porozuměno funkci celého zařízení jako celku.

2 Popis funkce zařízení

Zařízení na hůlce nevidomého neustále vysílá datovou sekvenci, která obsahuje jednak budicí sekvenci pro obvod ATA5282, dále pak posloupnost bitů, které zajišťují identifikaci vysílače. Jednoznačná identifikace vysílače je nutná proto, aby bylo možné následné vysílání o poloze nevidomého adresovat přímo konkrétnímu vysílači (konkrétnímu nevidomému). V opačném případě by mohlo dojít k tomu, že vysílání o poloze by přijímal nevidomý, který se nachází v jiné části budovy. Je-li přijímačem v podlaze zachyceno vysílání obsahující budicí sekvenci, přepne se tento z klidového režimu do aktivního a zahájí se vysílání informací o poloze. Blokové schéma celého systému je na obr. 2.1.



Obr. 2.1: Blokové schéma komunikačního zařízení

2.1 Popis jednotlivých bloků přenosné části systému

Mikroprocesor - tvoří jádro celého zařízení. Jeho úkolem je vytvoření a odeslání dat tvořících budicí a identifikační sekvenci pro přijímací část pod podlahou, zpracování přijatých dat obdržených z části pod podlahou, jejich vyhodnocení a následné odeslání informace, která identifikuje polohu, do PDA přes rozhraní bluetooth. Procesor řídí veškerou komunikaci, v jeho paměti jsou uloženy konkrétní hodnoty, přiřazené konkrétním pevným bodům.

Generátor budicího signálu – plní v systému úlohu modulátoru a zároveň vysílače celé budicí a identifikační sekvence. Hlavním prvkem tohoto bloku bude integrovaný obvod DSL modemu TDA5051 od firmy Philips. Výstupní signál je navíc zesílen zesilovačem s integrovaným obvodem AD8018.

Přijímač vysílání o poloze - pomocí tohoto bloku je zajištěn příjem informací o poloze z části umístěné pod podlahou. Přijímaný signál je demodulován a předán mikroprocesoru k dalšímu zpracování. Tento blok je tvořen radiokomunikačním modulem ARF2104. Tento modul může pracovat jako přijímač nebo vysílač. Z toho důvodu je tento modul použit v pevné části zařízení.

Bluetooth modul – úkolem tohoto bloku je bezdrátově předávat data mezi částí umístěnou na hůlce nevidomého a jeho kapesním počítačem PDA. Blok je tvořen bluetooth modulem OEMSPA310.

PDA – kapesní počítač - jeho úkolem je zpracování dat o poloze, která jsou mu dodána z mikroprocesoru. Tato data jsou následně zpracována a pomocí hlasového výstupu je nevidomý informován o názvu bodu, u kterého se nachází.

2.2 Popis jednotlivých bloků pevné části systému

Budicí obvod – tento obvod má za úkol po zachycení signálu ve správném formátu uvést do chodu celé zařízení umístěné pod podlahou, které bude z důvodu úspory energie v úsporném režimu. Budicí obvod tedy uvede zařízení do bdělého stavu a zajistí vysílání signálu s informacemi o poloze. Funkci tohoto bloku zajišťuje integrovaný obvod ATA5283.

Mikroprocesor – jeho úkolem je zachytit identifikační sekvenci přenosné části zařízení a řídit vysílání informací o poloze. Vlastní informace o poloze konkrétního bodu (tedy o své poloze) je uložena v jeho paměti. Hlavní částí tohoto bloku je mikrokontrolér ATMEGA8.

Vysílač informací o poloze – na základě vstupních dat dodaných mikroprocesorem zajišťuje jejich modulaci a vysílání do přijímače umístěného na hůlce. Blok je tvořen modulem ARF2104.

3 Řešení jednotlivých bloků přenosné části

3.1 Návrh bloku generátoru budicího signálu

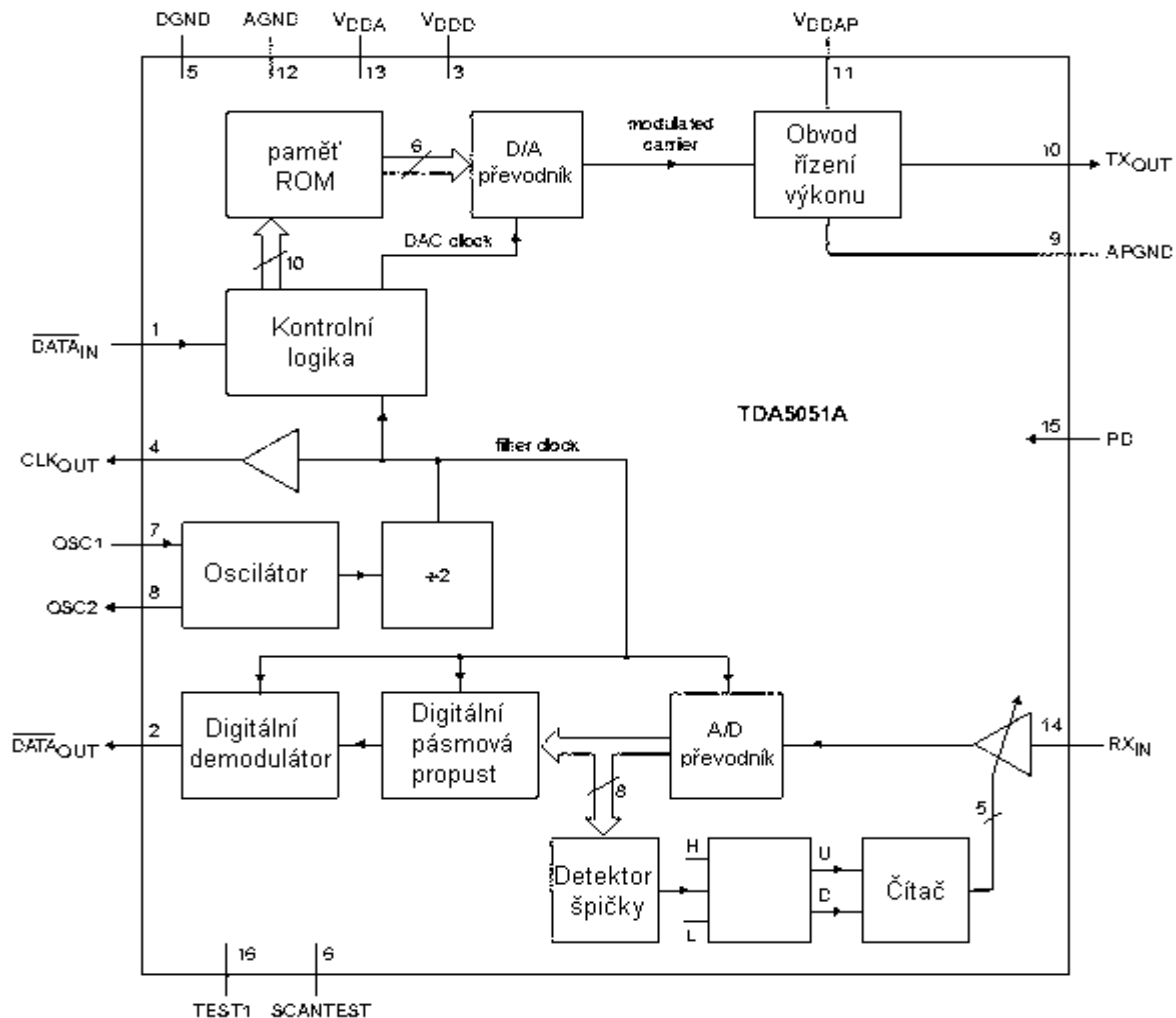
Tento blok byl od začátku navrhován tak, aby byl schopen spolupráce s integrovaným obvodem ATA5283, který zajišťuje zachycení budicí sekvence v části pod podlahou. Obvod je původně určen pro funkce monitorování tlaku v pneumatikách. Je navržen tak, že v režimu velmi nízké spotřeby (typicky 1mA) monitoruje stav na připojené snímací cívce (anténě). Teprve pokud detekuje signál modulovaný modulací ASK (amplitude shift keying) s kmitočtem nosné 125kHz, obsahující správný formát záhlaví, uvede se do bdělého stavu, čímž aktivuje blok zajišťující výstup přijatých dat a blok, pomocí kterého je možné uvést do bdělého stavu i ostatní části celého zařízení (např. mikroprocesor).

Vedoucí diplomové práce mi z vlastních zkušeností doporučil pro generaci potřebného signálu obvod TDA5051. Obvod je určen především pro komunikaci po domovní síti 220V, například pro ovládání klimatizace, osvětlení a podobně. Pro tento účel využívá obvod modulaci s klíčováním amplitudy o kmitočtu nosné 125kHz. Pokud jde o formát vysílaného signálu, odpovídá přesně požadavkům integrovaného obvodu ATA5283. Obvod TDA5051 bude tedy využit pro

generování budicí a identifikační sekvence z přenosné části směrem k části pod podlahou. Výstupní signál z obvodu bude navíc zesílen vhodným zesilovačem, aby se zvětšil dosah zařízení.

3.1.1 Popis obvodu TDA5051

Jedná se o obvod modemu, který pracuje s modulací ASK. Mezi vlastnosti obvodu patří například: plně digitální generování a tvarování nosné, frekvenci modulace a demodulace je možné určit buď pomocí vnějšího hodinového signálu (např. z mikrokontroléru) nebo pomocí vestavěného oscilátoru s připojeným krystalem, vysoký taktovací kmitočet 6-bitového A/D převodníku pro zamezení aliasingu a vestavěný výkonový stupeň s ochranou proti přetížení. Vnitřní blokové schéma obvodu je uvedeno na obr. 3.1.

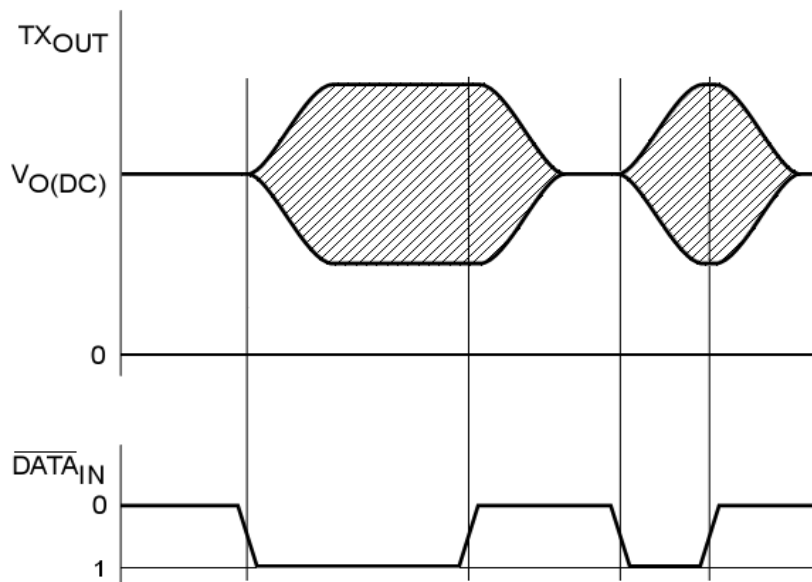


Obr. 3.1: Vnitřní uspořádání obvodu TDA5051

3.1.2 Popis funkce obvodu

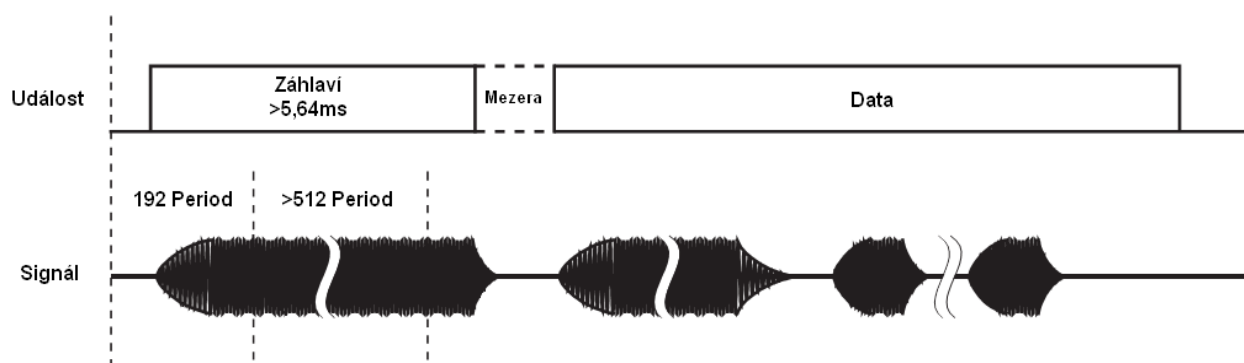
Na vstup DATAIN je přivedena datová sekvence (v našem případě z mikrokontroléru), určená pro vysílání. Tento logický vstup (jakožto i všechny ostatní logické vstupy a výstupy tohoto obvodu) je uzpůsoben pro příjem dat s typickými TTL/CMOS úrovněmi, je tedy možné použít přímo výstup mikrokontroléru. Vstup je uzpůsoben jako „active LOW“, tedy na výstupu TXOUT je vygenerován burst právě tehdy, je-li na vstupu DATAIN úroveň LOW. Signál ze vstupu je modulován modulací ASK, kmitočet nosné je nastaven buď vnitřním oscilátorem, nebo je řízen přímo mikrokontrolérem. Modulovaný signál je poté přiveden na výstup TXOUT. Obvod je uzpůsoben i pro funkci v opačném směru, tedy pro přijetí modulovaného signálu vstupem RXIN, následnou demodulaci a přivedení na výstup DATAOUT. Tuto funkci však v mém zapojení nevyužiji, neboť zpětná komunikace bude probíhat jiným kanálem.

Obvod je vybaven funkcí pro úsporu spotřeby energie v době, kdy není vysílač aktivní. Tato funkce je ovládána pomocí logického vstupu PD. Tento vstup je uzpůsoben jako „active HIGH“, tedy minimální spotřeba je v době, kdy je na vstup přivedena logická úroveň HIGH. V tom případě jsou vypnuty veškeré funkce s výjimkou generování taktovacího kmitočtu. Na obr. 3.2 je naznačen přibližný vztah průběhů mezi vstupem DATAIN a výstupem TXOUT.



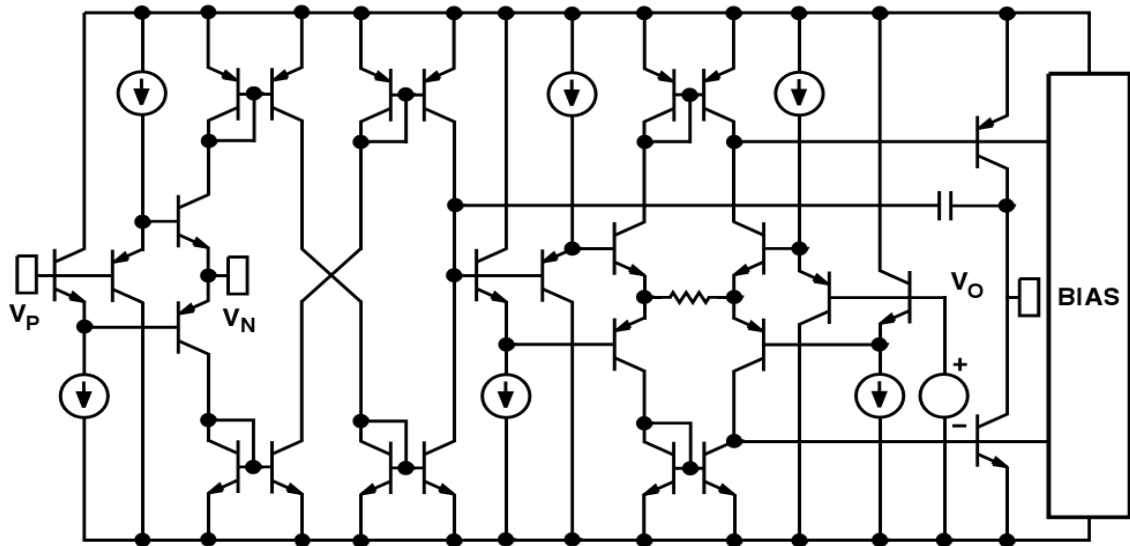
Obr. 3.2: Vztah průběhů mezi výstupem a vstupem dle katalogového listu

Sekvence vysílaná tímto obvodem musí být specifický tvar, který je nutný pro správné probuzení obvodu ATA5283 v pevné části systému. Tato sekvence musí sestávat ze záhlaví, po kterém je obvod teprve připraven pro příjem dat. Záhlaví musí mít formát 192 po sobě jdoucích period nosného signálu bez přerušení. Po příjmu hlavičky musí následovat alespoň 512 period nosného signálu, po kterých obvod aktivuje blok kontroly vstupního zisku, který je nezbytný pro demodulaci signálu. Po záhlaví následuje mezera v délce přibližně 100 period. Následuje blok vlastních dat. O následné převedení obvodu zpět do úsporného režimu se postará mikroprocesor pomocí vyslání reset signálu po správném přijetí dat. Přibližný tvar kompletní vysílací sekvence je na obr. 3.3.



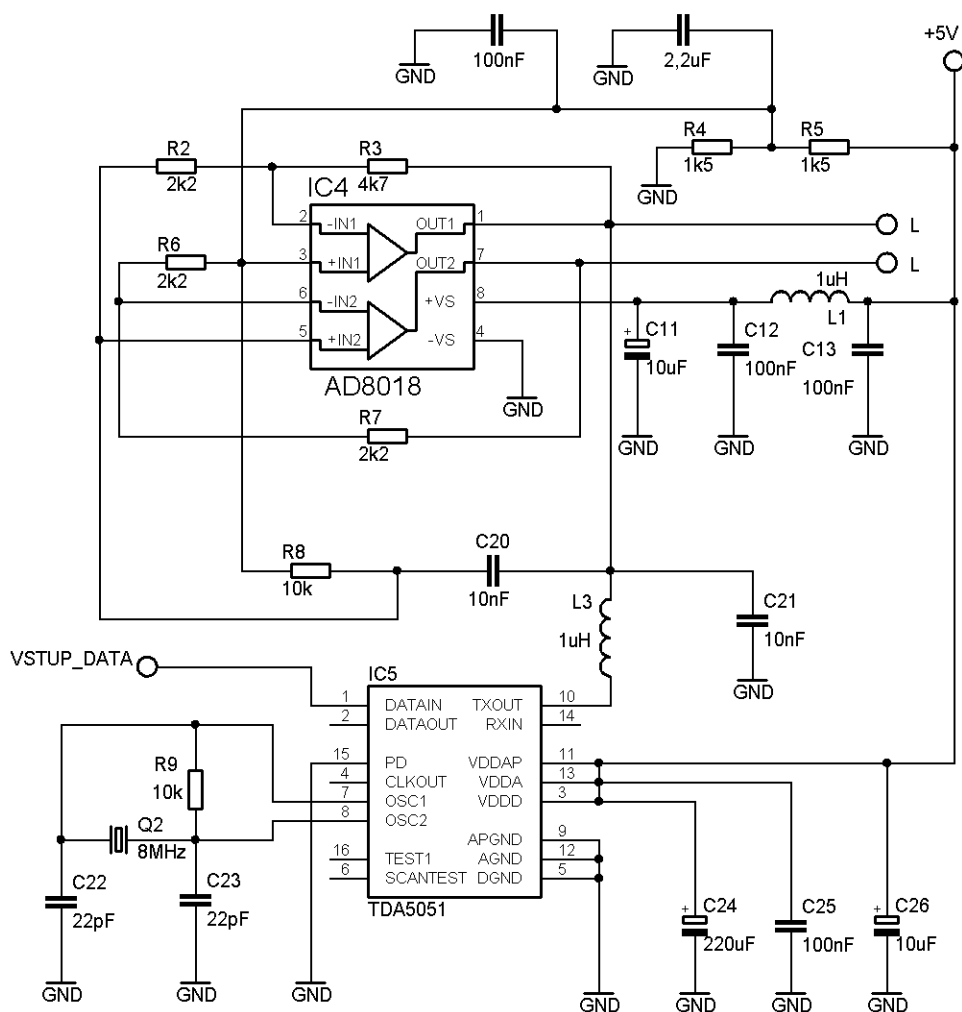
Obr. 3.3: Formát signálu pro obvod ATA5282

Z důvodu většího dosahu zařízení je na výstupu obvodu TDA5010 zařazen ještě zesilovací člen, tvořený integrovaným obvodem AD8018. Obvod je primárně určen do xDSL modemů. Při napájecím napětí 5V dosahuje zesilovač s tímto obvodem výstupního výkonu 16dBm. Na obr. 3.4 je uvedeno zjednodušené vnitřní zapojení integrovaného obvodu AD80180.



Obr. 3.4: Zjednodušené vnitřní zapojení obvodu AD8018

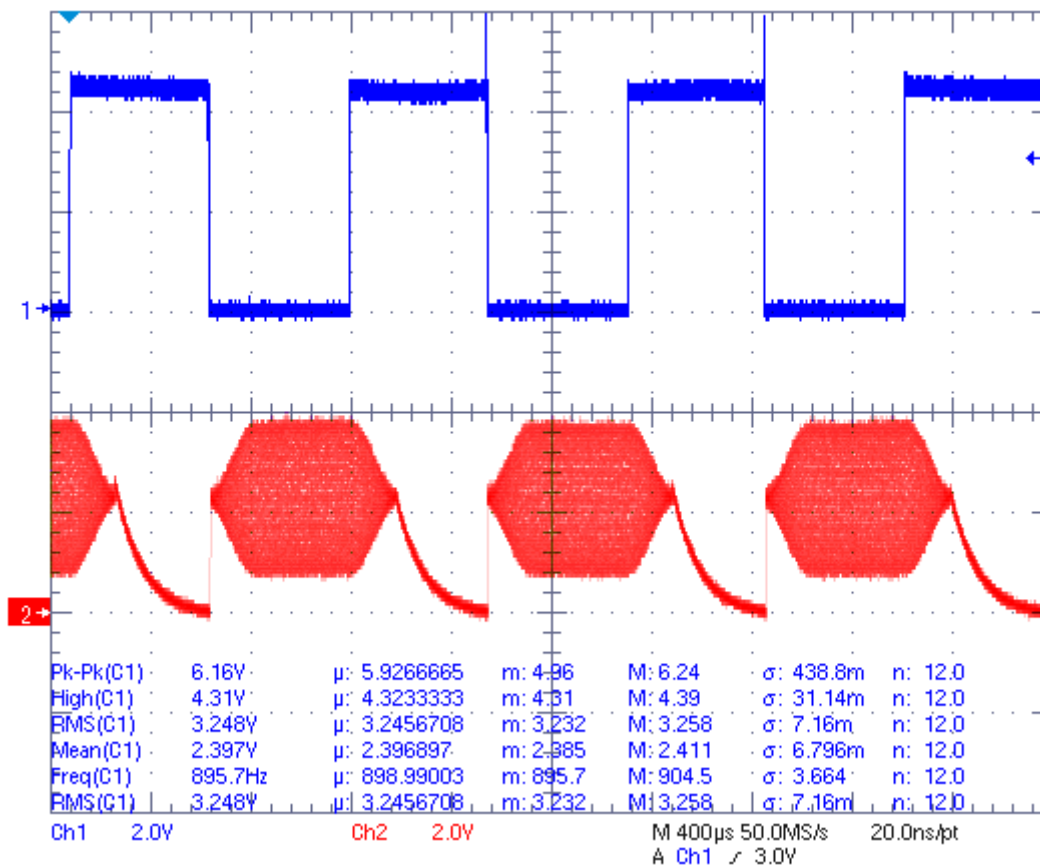
Zesilovač AD8018 je složen ze dvou zpětnovazebních proudových zesilovačů, schopných dodávat až 400mA výstupního proudu při velmi malých hodnotách zkreslení výstupního signálu. Obvod nabízí tři režimy úsporného režimu – plný režim pohotovostní režim a vypnuto. Na obr. 3.5 je zapojení obvodu TDA 5010 i se zesilovačem. Toto zapojení vychází z doporučení od vedoucího diplomové práce.



Obr. 3.5: Zapojení obvodu TDA5051 se zesilovačem

Vývody označené písmenem L slouží k připojení vysílací cívky (antény), vstup označený DATA_IN potom pro připojení datového výstupu z mikroprocesoru. Jako vysílací cívka je použita cívka z lakovaného drátu o průřezu $0,4\text{mm}^2$ se vzduchovým jádrem s naměřenou indukčností $169,45\mu\text{H}$.

Na obr. 3.6 je záznam osciloskopu z měření obvodu TDA 5051. Na jeho vstup byl přiváděn obdélníkový signál z mikroprocesoru (vytvořen pomocí časovače, během jednoho hodinového cyklu dojde ke změně stavu z 0 na 1 a naopak, registr TCCR0 v hodnotě 2, registr TCNT0 v hodnotě 0, viz. kapitola blabla.) o amplitudě přibližně 5V a frekvencí 895,7Hz. (průběh označený číslicí 1). Frekvence nosné signálu označeného číslicí 2 je 125kHz. Z průběhu je vidět, že naměřené hodnoty jsou velmi podobné těm katalogovým.



Obr. 3.6: Naměřený vztah mezi vstupem a výstupem obvodu TDA5051

3.2 Přijímač vysílání o poloze

Úkolem tohoto bloku je přijmout vysílání od vysílače umístěného pod podlahou, přijatý signál demodulovat a digitální data předat k dalšímu zpracování mikroprocesoru. Všechny tyto funkce obstarává radiokomunikační modul ARF2104. Stejný modul, pouze přepnutý na funkci vysílače, je použit v pevné části zařízení.

3.2.1 Radiokomunikační modul ARF10

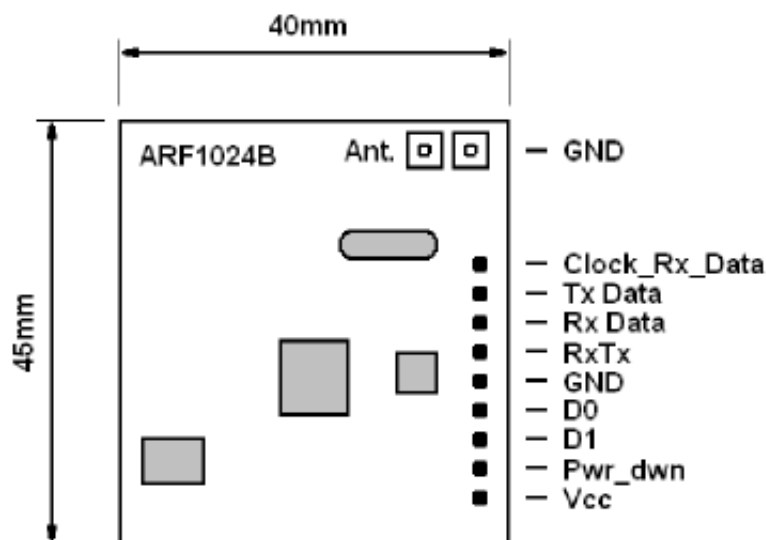
Jedná se o jednokanálový digitální přenosový systém, pracující na kmitočtu 433,9MHz. Tento kmitočet je určen oscilátorem s kalibrovaným SAW rezonátorem. Vysílací část má výkon 10mW (+10dBm) při impedanci 50 Ω . Přijímací část je tvořena homodynním přijímačem s citlivostí

lepší než -102dBm (1,8mV) pro 10⁻² BER. Modul nabízí volbu ze čtyř přenosových rychlostí: 4kbit/s, 16kbit/s, 32kbit/s a 64 kbit/s. Modul je určen výhradně pro přenos digitálních dat, je schopen pokročile zpracovávat přijímaný signál a obnovovat jeho hrany. Z důvodu usnadnění dalšího zpracování dat generuje modul hodinový signál pro synchronizaci (např. mikroprocesoru) na výstupu Clock_Rx_Data. Použitý komunikační protokol musí být vytvořen uživatelem, na míru konkrétním požadavkům, není v modulu integrován. Při sestavování spojení mezi vysílačem a přijímačem je především nutné nastavit hodnotu přenosové rychlosti. Nastavení se provádí pomocí pinů D0 a D1 na konektoru J1. Je důležité nastavit přenosovou rychlost u přijímače podle rychlosti, jakou vysílá vysílač. V opačném případě je generován chybný synchronizační signál. Před vlastním sestavením spojení je nutné moduly navzájem synchronizovat. Toho je možné dosáhnout vysláním synchronizačního rámce o délce 20 bitů obsahující sekvenci střídajících se nul a jedniček. Modul nabízí možnost omezení vlastní spotřeby v přijímacím režimu přepnutím do pohotovostního režimu. Je-li tento režim aktivován, je přijímač střídavě po dobu 1000ms v pohotovostním režimu a po dobu 35ms v aktivním režimu. V takovém případě je nutné vysílat synchronizační posloupnost po dobu delší, než je doba setrvání přijímače v pohotovostním režimu zvětšená o dobu trvání aktivního stavu a o dobu, kterou musí trvat vlastní synchronizační sekvence 20 bitů.

D1	D0	Přenosová rychlost
0	0	4kb/s
0	1	16kb/s
1	0	32kb/s
1	1	64kb/s

Tab. 1: Nastavení přenosové rychlosti

Modul je od výroby kompletně osazen a oživen. Jediná úprava spočívá v připájení pravouhlé pinové lišty, kterou bude modul připojen do hlavní desky plošných spojů a připojení antény. Na obr. 7 je znázorněno zapojení hlavního konektoru a rozměry modulu. Modul je uzpůsoben pro spolupráci s anténou typu „bič“ nebo helical.



Obr. 3.7.: Zapojení hlavního konektoru modulu

Popis vývodů modulu:

Ant – pin pro připojení antény

Clock_Rx_Data – na tomto pinu je generován synchronizační signál, platný na vzestupnou hranu signálu přijatého na pinu Rx Data

Tx Data – vstup pro data určená k vysílání

Rx Data – výstup přijatých dat

GND – zem obvodu

D0 – pin sloužící spolu s pinem D1 pro nastavení přenosové rychlosti (viz. tab. 1)

D1 - pin sloužící spolu s pinem D0 pro nastavení přenosové rychlosti (viz. tab. 1)

Pwr_dwn – pomocí tohoto pinu je možné přepnout modul do úsporného režimu

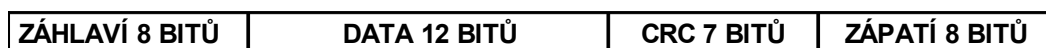
Vcc – připojení napájecího napětí 3V

3.2.2 Komunikační protokol

Vzhledem k tomu, že moduly ARF2104 nemají v sobě implementovaný komunikační protokol, bylo nutné tento navrhnout.

Mnou navržený komunikační protokol používá pakety o délce 35 bitů s ochranou proti

chybám CRC-7 (generující polynom 10001001). V případě, že je datová část paketu neprojde CRC kontrolou, je paket zahozen a přijat nový. 12 datových bitů je rozděleno na osm bitů, ve kterých je přenášena identifikace pevného bodu a na čtyři ve kterých je přenášena identifikace přijímače (slepecké hůlky). Zdrojový kód pro mikroprocesor, který obsahuje i vlastní protokol je v příloze této práce.



Obr. 3.8: Formát datového paketu

3.3 Mikroprocesor

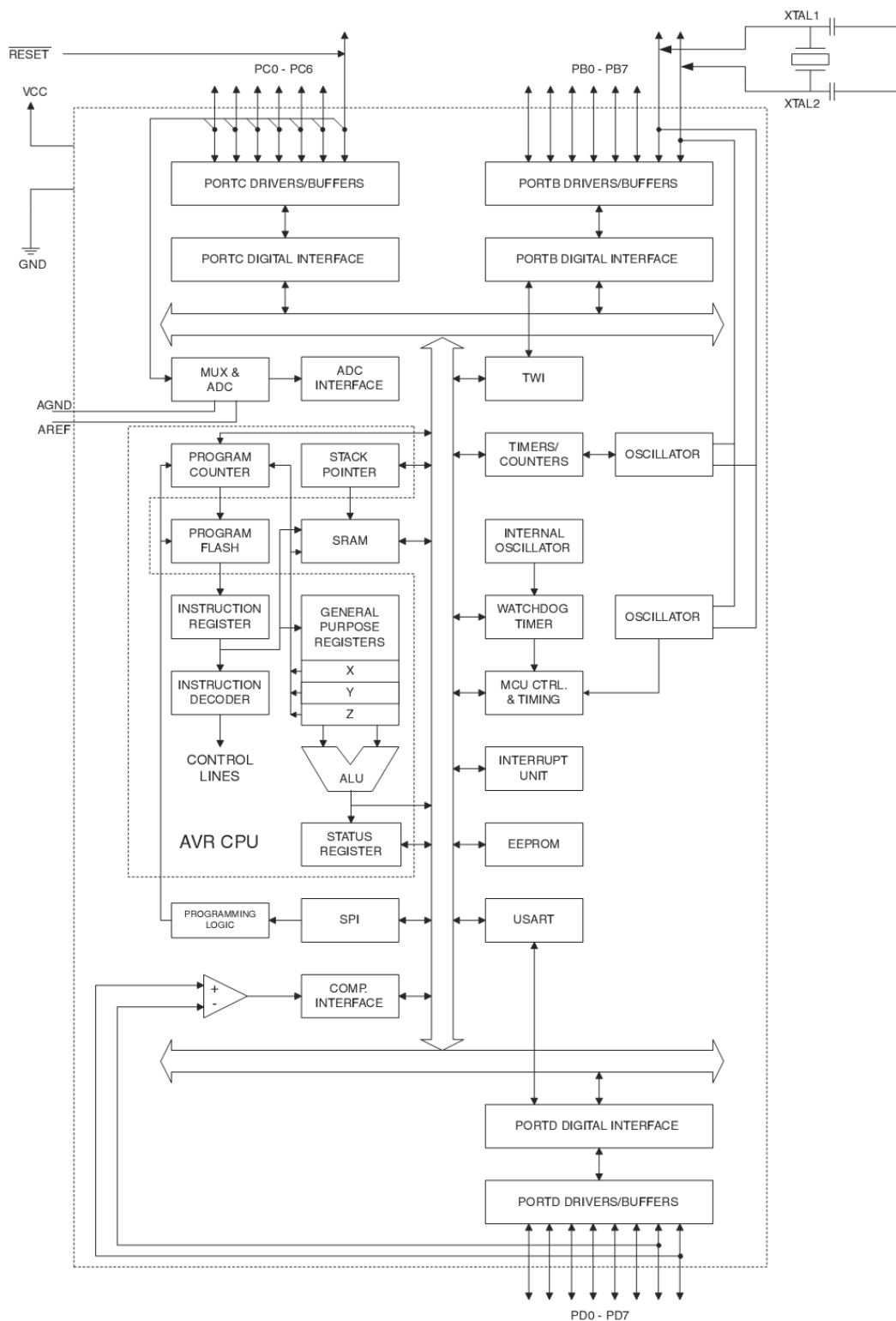
Mikroprocesor je řídicí jednotkou celého zařízení. Stará se o komunikaci všech částí zařízení, včetně komunikace se zařízením pod podlahou. V jeho paměti je uložen tvar budicí a identifikační sekvence, instrukce pro zpracovávání vysílaných a přijímaných dat pomocí komunikačních protokolů. Vzhledem k obrovské výkonnosti a dobré dostupnosti mikrokontrolérů dnešní doby, zvládne tyto úlohy se značnými rezervami plnit i procesor nejnižších výkonových řad. Vybral jsem pro tyto účely procesor od firmy Atmel ATmega8.

3.3.1 Popis mikrokontroléru ATmega8

Jedná se o nízkopříkonový 8 bitový CMOS jednočipový mikroprocesor založený na AVR RISC architektuře. RISC architektura (Reduced Instruction Aet Computer, tedy počítač s redukovanou instrukční sadou) vychází z harvardské architektury, tedy paměť programu a paměť dat jsou odděleny. Jádro procesoru se skládá z 32 totožných 8 bitových paměťových registrů, které mohou obsahovat jak data, tak adresy. Všechny 32 registrů je přímo připojeno do matematicko-logické jednotky (ALU), čímž je umožněno přistupovat ke dvěma nezávislým registrům během jedné instrukce, provedené za jeden hodinový cyklus.

Procesor disponuje 8kB vnitřní programovatelnou flash pamětí se schopností čtení během zápisu (Read While Write), 512B paměti EEPROM, 1kB paměti SRAM, 23 univerzálních I/O linek, 32 univerzálních pracovních registrů, třemi stavitelnými časovači/čítači, nabízí možnost vnitřních

a vnějších přerušení. Jeho součástí je také sériové programovatelné USART rozhraní, SPI sériové rozhraní, 6 kanálový A/D převodník, programovatelný „watchdog“ časovač s vnitřním oscilátorem a pět softwarově volitelných režimů úsporného režimu. Vnitřní blokové schéma procesoru je uvedeno na obrázku 3.6.



Obr. 3.9: Vnitřní uspořádání procesoru ATmega8

3.3.1.1 Modul časovače

Modul časovače (timer) slouží k vykonání operací, které je třeba v určitých časových intervalech opakovat, nebo pro operace, které po svém zavolání (např. přerušením) musí trvat jen určitou dobu. Časovač může být řízen pomocí interního oscilátoru, nebo pomocí externě připojeného krystalového oscilátoru. Princip časovače spočívá v neustálém přičítání jedničky do speciálního registru TCNT0, k tomuto přičtení dojde během každého hodinové cyklu, tedy s frekvencí f_{CLK} .

Velice často bývá časovač užíván ve funkci, kdy při přetečení hodnoty registru (maximální hodnota 8-bitového timeru je 255) dojde k obsluze přerušení. Během tohoto přerušení je potom volána funkce, která má být například periodicky vykonávána. Modul časovače obsahuje děličku (prescaler), pomocí které je možné měnit zvyšování hodnoty registru TCNT0. Nastavení děličky se pomocí zapisování hodnot do třech bitů (CS2, CS1 a CS0) registru TCCR0. Při nastavení hodnot bitů registru na hodnotu 001 je dělička vyřazena z provozu a přičítání do registru TCNT0 je řízeno aktuálním hodinovým cyklem procesoru. Další možnosti nastavení časovače pomocí děličky jsou uvedeny v tabulce BLABLA.

CS02	CS01	CS00	Stav časovače
0	0	0	časovač zastaven
0	0	1	dělička nastavena na hodnotu $f_{CLK}/1$ (tedy nepoužita)
0	1	0	dělička nastavena na hodnotu $f_{CLK}/8$
0	1	1	dělička nastavena na hodnotu $f_{CLK}/64$
1	0	0	dělička nastavena na hodnotu $f_{CLK}/256$
1	0	1	dělička nastavena na hodnotu $f_{CLK}/1024$
1	1	0	připojen externí oscilátor
1	1	1	připojen externí oscilátor

Tab. 2.: Nastavení časovače pomocí děličky

3.3.1.2 Modul externího přerušení

Externí přerušení (external interrupt) slouží k vykonání funkcí, jejichž zavolání je nutné v určitém okamžiku, který nemá žádnou souvislost s vnitřními událostmi v procesoru. Tedy podnět, při kterém je toto přerušení voláno přichází zvenčí. Tento modul umožňuje procesoru reagovat na události jako je zmáčknutí tlačítka a podobně.

Funkce externího přerušení je svázána s piny INT0 a INT1. Externí přerušení může být vyvoláno náběžnou hranou digitálního signálu, sestupnou hranou nebo může být vyvoláno při obou těchto událostech. K tomuto nastavení slouží dva bity ISC01 a ISC00 v registru MCUCR (MCU control register). Bity ISC01 a ISC00 slouží pro nastavení externího přerušení ve vztahu k pinu INT0. Pro pin INT1 slouží bity ISC11 a ISC01 téhož registru.

ISC01	ISC00	Nastavení externího přerušení
0	1	Stav 0 na pinu INT0 vyvolá přerušení
0	0	Jakákoli logická změna vyvolá přerušení
1	1	Přerušení vyvoláno při sestupné hraně signálu
1	0	Přerušení vyvoláno při náběžné hraně signálu

Tab.3: Nastavení externího přerušení

3.3.1.3 Modul sériového rozhraní USART

Sériové rozhraní USART (The Universal Synchronous and Asynchronous serial Receiver and Transmitter) tohoto procesoru je velmi výkonný komunikační nástroj, použitelný pro komunikaci s jinými zařízeními, jako například osobní počítač, stejně jako pro komunikaci dvou stejných mikroprocesorů. Mezi vlastnosti rozhraní patří plně duplexní režim, který umožňuje nezávislé registry pro vysílání a příjem, synchronní nebo asynchronní mód, podpora rámců o délce 5 až 9 datových bitů, generování sudé nebo liché parity, detekce chybných paketů, tři nezávislá přerušení propojená s událostmi na tomto rozhraní a další.

Výhoda modulu USART je v tom, že již obsahuje integrovaný komunikační protokol. Taktéž není zapotřebí dalšího vodiče pro synchronizaci dvou komunikujících zařízení, protože k synchronizaci dochází uvnitř protokolu. To může někdy vést k chybám v komunikaci z důvodů nesprávného nastavení generování hodinového signálu nebo kvůli jeho nestabilitě. Některé procesory z řady AVR proto umožňují použití externí synchronizace a díky tomu mohou pracovat

v synchronním módu.

Důležitou částí programování modulu USART je nastavení hodnoty registru generátoru modulační rychlosti (baud rate generator). Tento slouží pro nastavení přenosové rychlosti komunikace a je pevně svázán s hodnotou hodinového kmitočtu procesoru f_{CLK} . Generátor sestává z registru UBBR a s ním propojeného sestupného počítadla. Hodnotu, kterou je potřeba zapsat do registru UBBR při nastavování rozhraní USART je možné vypočítat podle vztahu

$$UBBR = ((f_{CLK} / (M \cdot 16)) - 1), \quad [5]$$

kde f_{CLK} je hodinový kmitočet procesoru a M modulační rychlost.

Pokud chceme v aplikaci rozhraní USART využít, je třeba pečlivě vybírat pracovní kmitočet procesoru f_{CLK} . Výsledná chybovost přenosu totiž přímo s tímto nastavením souvisí. Při nevhodně zvoleném pracovním kmitočtu může být chybovost tak velká, že naruší správné fungování rozhraní. Následuje příklad výpočtu pro kmitočet 4 MHz a modulační rychlost 19200 Baud/s. Hodnota zapsaná do registru UBBR bude tedy

$$UBBR = (4 \cdot 10^6 / (19200 \cdot 16) - 1) = 12,0208\bar{3}.$$

Jelikož víme, že hodnota zapsaná do registru UBBR musí být celé číslo, dostáváme tedy hodnotu 12. Upravíme původní rovnici tak, že modulační rychlost bude neznámá a dosadíme za UBBR hodnotu 12

$$M = (4 \cdot 10^6 / 16 \cdot (12 + 1)) = 19230,769 \text{ Baud/s.}$$

Z výsledku je patrné, že modulační rychlost zcela neodpovídá zvolené hodnotě modulační rychlosti. Výsledná chyba činí

$$d = (19200 / 19230,769 - 1) \cdot 100 = -0,16 \text{ .}$$

Pokud zvolíme hodnotu $f_{CLK} = 3,6864 \text{ MHz}$, pro hodnotu registru UBBR dostáváme

$$UBBR = (3,6864 \cdot 10^6 / (19200 \cdot 16) - 1) = 11,$$

nyňí jsme tedy obdrželi celé číslo. Aníž by bylo zapotřebí dalších výpočtů, víme, že výsledná chyba bude v tomto případě rovna nule.

I při pracovním kmitočtu procesoru 4MHz by přenos po rozhraní fungoval bezchybně, neboť k chybám v přenosu dochází až od hodnoty chyby větší než 0,5% [5]. Přesto jsem však při návrhu zařízení zvolil pracovní kmitočet procesoru 3,6864MHz.

3.3.1.4 Nastavení pojistek

Mikroprocesory této řady vyžadují pokročilejší nastavování vlastností procesoru přes tzv. pojistky. Mezi tato nastavování patří například změna kmitočtu vnitřního oscilátoru, přepnutí z vnitřního oscilátoru na vnější nebo vypnutí sériového programovacího rozhraní. Pojistky svou funkcí odpovídají funkci konfiguračního souboru procesoru. Při nastavování pojistek je na místě opatrnost, protože například při nechtěném vypnutí sériového programovacího rozhraní je pak možná náprava jedině pomocí jiného typu programátoru.

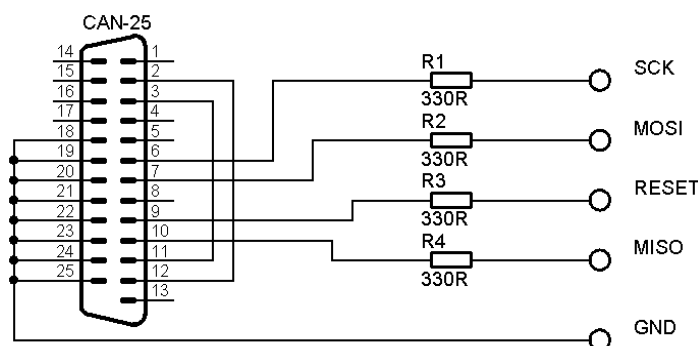
Pojistky tvoří dvě sady po osmi bitech. Pomocí nastavování těchto bitů lze měnit důležitá nastavení procesoru. K pojistkám se přistupuje pomocí programátoru. V následujícím příkladu vidíme způsob nastavení pojistek v příkazové řádce. Nastavení odpovídá mimo jiné použití vnitřního oscilátoru s frekvencí 8MHz, zapnutí sériového programovacího rozhraní a zapnutí watchdog časovače. Stk200 odpovídá typ programátoru, m8 potom typu procesoru. Zapisujeme do sady pojistek s označením low hodnotu 0xE4.

```
avrdude -c stk200 -p m8 -U lfuse:w:0xE4:m
```

Nastavení odpovídá mimo jiné použití vnitřního oscilátoru s frekvencí 8MHz, zapnutí sériového programovacího rozhraní a zapnutí watchdog časovače. Stk200 odpovídá typ programátoru, m8 potom typu procesoru. Zapisujeme do sady pojistek s označením low hodnotu 0xE4.

3.3.1.5 Programování mikroprocesoru

Pro programování procesoru ATmega8 není zapotřebí originálního programovacího kitu, jednoduchý programátor si lze poměrně snadno vyrobit. Schéma zapojení programátoru („dongle“) je na obr. 3.10. Zapojení sestává jen z konektoru pro paralelní port a čtyř rezistorů. Výstupní piny se připojí k pinům procesoru s odpovídajícím pojmenováním. Jedná se o programátor, využívající paralelní (LPT) rozhraní. Programování však probíhá sériově, přes vstupní sériové rozhraní procesoru. Dále již je potřeba jen vhodný kompilátor. Pro tyto účely byl zcela dostačující freeware program WinAVR:



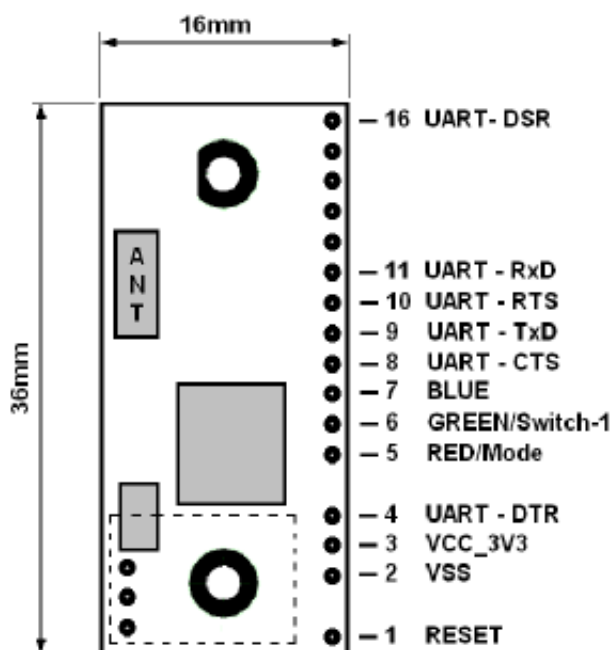
Obr. 3.10: Zapojení programátoru mikroprocesoru

3.3 Bluetooth modul

Tato část má za úkol vysílat bezdrátově data z přijímací části zařízení, umístěného na hůlce, do kapesního počítače PDA. Tento přenos je realizován bezdrátově z důvodu zvýšení komfortu při manipulaci se zařízením. Jádrem tohoto bloku je modul cB-0907-01. Jedná se bluetooth zařízení vývojové řady 2.0. Je osazen chipsetem BGB203 od firmy Philips. BGB203 obsahuje integrovanou paměť SRAM a FLASH. Výkonově modul odpovídá třídě class 2 s výstupním výkonem +3,5dBm. Modul je vybaven integrovanou anténou.

3.3.1 Popis modulu cB-0907-01

Zařízení je primárně určeno pro bezdrátový přenos sériové linky (např. RS232). Pro tento účel je vybaven všemi potřebnými vstupy. V případě používání modulu v tomto režimu dojde na cílovém zařízení (PC, PDA) k detekci tohoto modulu, následně je mu automaticky přiřazen jeden ze sériových portů (v tomto případě COM9). Následně je možné modul se všemi jeho vstupy používat jako rovnocennou náhradu „pevného“ sériového portu. Modul nabízí možnost změny vnitřního firmware přímo uživatelem, a tím i využití modulu v jiných aplikacích. Nahrání jiného firmware je možné buď přímo pomocí bluetooth rozhraní s použitím výrobcem dodaného software nebo pomocí připojení jeho vstupů na sériový port počítače. V takovém případě je však nutné zapojit do signálové cesty vhodný obvod, který by zajišťoval převod napětových úrovní mezi modulem, používajícím napětové UART úrovně 3V a počítačem, který používá napětové úrovně +- 3 až 15V (RS232). Na obr. 3.11 je znázorněno zapojení konektoru a rozměry modulu. Vzhledem k tomu, že modul disponuje poměrně velkým počtem vývodů, z nichž využívám jen zlomek, jsou v následujícím popisu uvedeny jen ty, které se vztahují k aplikaci v rámci diplomové práce.



Obr. 3.11: Zapojení kontaktů bluetooth modulu

Popis vývodů modulu:

VSS – uzemnění modulu

VCC_3V – připojení napájecího napětí 3V

UART-DTR – tímto signálem se oznamuje, že terminál je připraven komunikovat (*Data Terminal Ready*)

UART-CTS – tímto signálem terminál oznamuje, že komunikační cesta je volná (*Clear To Send*)

UART-TxD – vstup pro data určená k vysílání

UART-RTS – tímto signálem oznamuje terminál, že bude vysílat data (*Request To Send*)

UART-RxD – výstup přijatých dat

UART-DSR – tímto pinem je terminálu oznamováno, že zařízení na druhém konci signálové cesty je zapnuto a připraveno přijímat data (*Data Set Ready*)

3.3.2 Připojení BT modulu

Vzhledem k tomu, že rozhraní USART u mikrokontroléru AtMega8 tvoří pouze 4 vodiče (kladný pól napájení, záporný pól napájení, vstup dat k vysílání a výstup přijatých dat), nepoužívám v mém návrhu signálové vodiče jako například UART-CTS. Jejich použití není pro bezchybnou sériovou komunikaci nutné.

Modul je, pokud jde o mechanickou stránku, navržen pro připojení buď speciálním konektorem, který není v běžné maloobchodní síti dostupný nebo pomocí pájecích plošek. Tyto mají nestandardní rozteč 2mm, čímž znemožňují použití standardní pinové lišty. Připojení je tedy řešeno pomocí protikusů pinové lišty o délce 5 pinů, která je přilepena z rubu ve spodní části desky modulu (viz. čárkovaná čára ve vyobrazení modulu na obr BLABLA) a s pájecími ploškami je propojena pomocí tenkých vodičů. Díky tomu je možné modul kdykoli ze základní desky snadno vyjmout.

3.4 PDA, hlasový výstup

Kapesní počítač slouží pro konečné vyhodnocení polohy přenosného zařízení a porovnání této polohy s daty uloženými v jeho databázi. Na základě toho je nevidomému oznámena jeho přesná poloha v rámci objektu. PDA komunikuje s nevidomým pomocí hlasového výstupu. Vlastní hlasový výstup je ošetřen pomocí hlasových sekvencí zpracovaných ve formátu MP3.

Konkrétně je úkolem PDA přijímat data přes svůj sériový port (vytvořený virtuálně, přístup pomocí rozhraní bluetooth), tato data vyhodnotit a na jejich základě přehrát konkrétní zvukový soubor mp3, popisující aktuální polohu nevidomého. Pro tyto účely používám PDA od firmy Fujitsu Siemens Computers, typ Pocket LOOX N560 s procesorem pracujícím na kmitočtu 624MHz a operačním systémem Windows Mobile 6.

3.4.1 Vytvoření aplikace pro PDA

Aplikaci, plnící výše popsané úkoly, jsem programoval ve zkušební verzi vývojového Microsoft Visual Studio 2008 pomocí programovacího jazyku C Sharp. Tento vývojový software umožňuje při debugování, pokud je PDA zařízení připojeno k PC, přímé zobrazení právě debugované aplikace na obrazovce PDA (funkce deploy). Druhou možností je emulace zařízení PDA na obrazovce PC, kdy se zobrazí obrázek PDA i se všemi hardwarovými ovládacími prvky (tlačítka) a na jeho displeji debugovaná aplikace. Aplikace spuštěná v tomto režimu je vidět na obr. 3.12.

Úkolem aplikace Navigátor 1.0 je přijmout pomocí sériového portu (virtuální COM port) data, vysílaná mikroprocesorem do bluetooth modulu. Tato data obsahují sekvenci určující číslo v tu chvíli aktuálního orientačního bodu. Tato data se vyhodnotí a přiřadí se jim odpovídající zvuková nahrávka ve formátu mp3. Pro potřeby vyzkoušení v rámci diplomové práce má aplikace k dispozici jen pět souborů ve formátu mp3, které odpovídají číslům dveří 203, 205, 207, 209 a 211. Čísla dveří přímo odpovídají hodnotě v desítkové soustavě, která je posílána z mikroprocesoru. V případě, že dojde k příjmu jiné hodnoty, ke které není pořízena zvuková nahrávka, spustí se hlášení o chybném příjmu dat.

Aplikace po spuštění automaticky vyhledá dostupné sériové porty. Uživatel vybere správný port podle toho, jaký virtuální port přiřadil při vyhledávání dostupných bluetooth zařízení. V tomto zařízení použitý bluetooth modul se zobrazí pod názvem „oem310“. Po zvolení portu stačí stisknout

tlačítko „Spustit navigaci“ a tím spustit komunikaci s přijímačem umístěným na hůlce nevidomého. Spuštění a nastavení aplikace musí zajistit druhá osoba, například zaměstnanec konkrétního objektu, po kterém má být nevidomý navigován. Aplikace je nastavena tak, že v režimu vyhledávání je vždycky zvýrazněno tlačítko „opakovat hlášení“, čímž je kdykoli dostupné pomocí hardwarového tlačítka enter. Tlačítko pro opakování hlášení zabírá navíc velkou část spodní části displeje. Protože je displej dotykový, nevidomému stačí zmáčknout do spodní části displeje a dojde ke stlačení tlačítka.



Obr 3.12: Aplikace Navigátor 1.0

Popis prvků aplikace Navigátor 1.0

Menu „Zvol port“ - slouží pro výběr portu, na kterém je navázáno spojení s bluetooth modulem. Dostupné porty zjistí aplikace automaticky po startu.

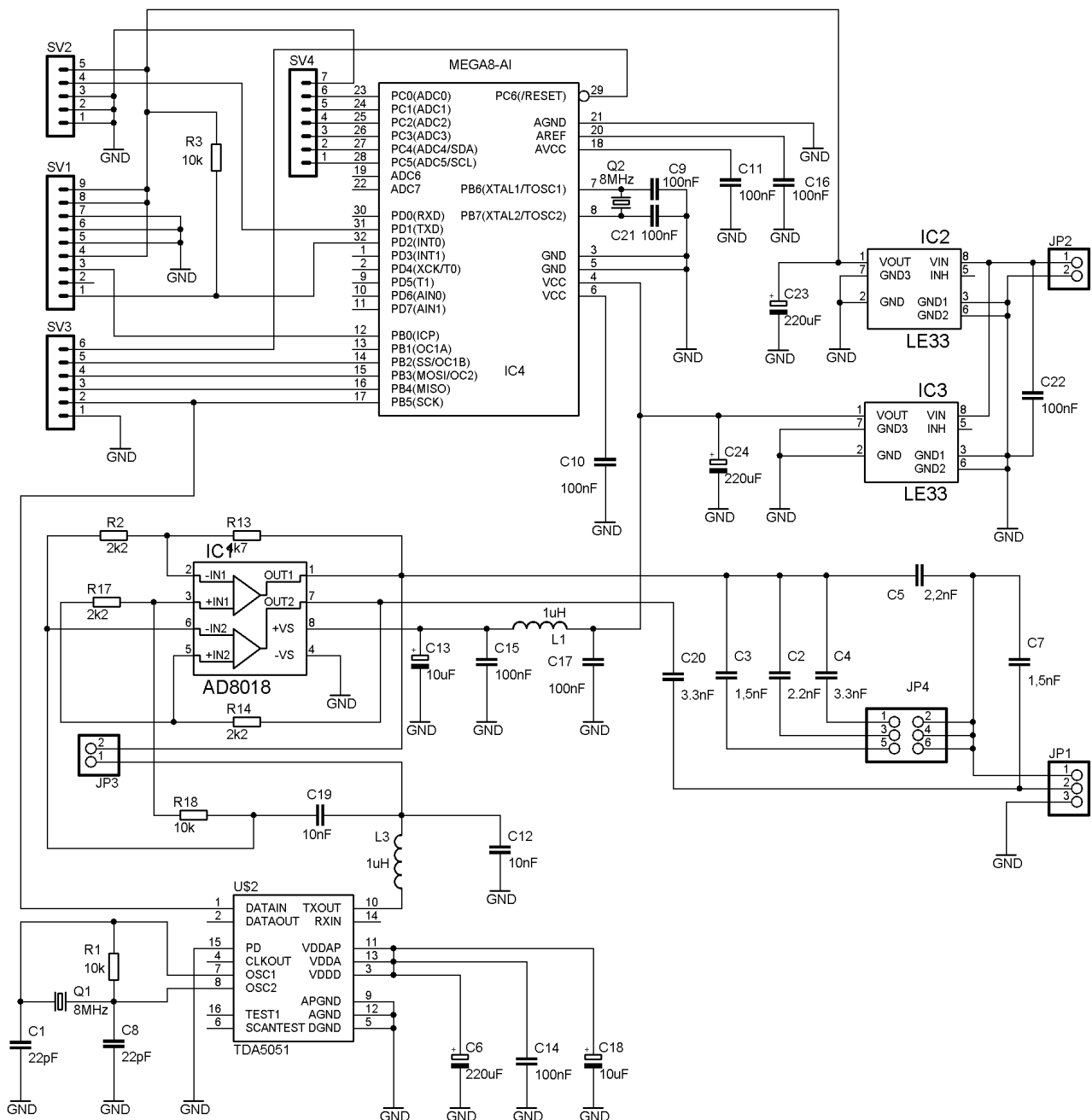
Tlačítko Reset – slouží pro uzavření aktuálně používaného portu a uvedení aplikace do výchozího stavu

Tlačítko „Spustit navigaci“ – po stisknutí se otevře zvolený port a zahájí se příjem dat z bluetooth modulu. V případě, že je v dosahu nějaký orientační bod, přehraje se příslušná zvuková nahrávka

Tlačítko „Opakovat hlášení“ – přehraje zvukovou nahrávku vztahující se k poslednímu orientačnímu bodu, který byl zařízením lokalizován.

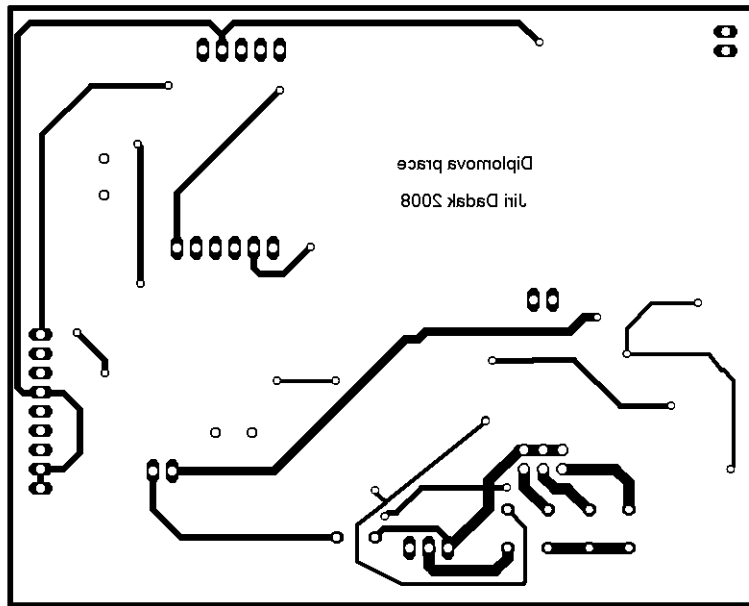
4 Propojení všech částí

Základem zařízení je základní deska plošných spojů, ve které jsou zasunuty i oba komunikační moduly. Připojení modulů je řešeno pomocí pinových lišt, moduly jsou vůči desce umístěné kolmo. Celkové schéma zapojení, včetně znázornění konektorů pro připojení modulů je na obr. 4.1.

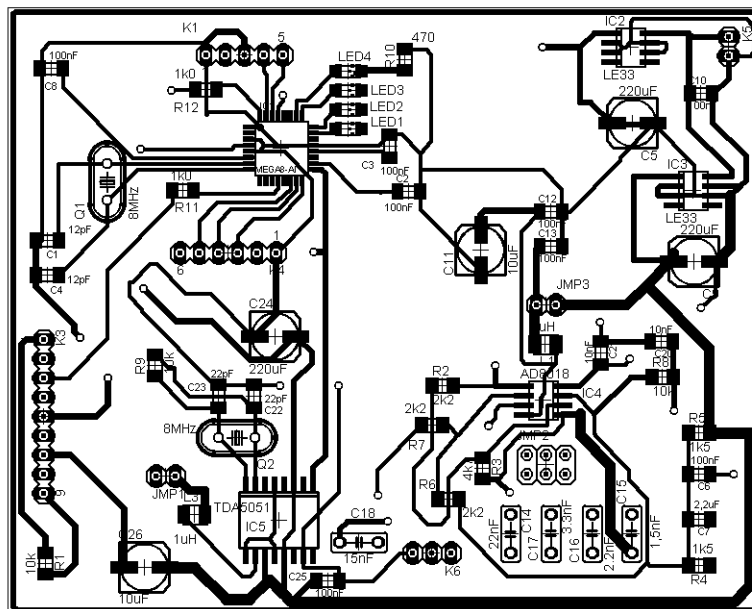


Obr. 4.1: Celkové schéma zapojení komunikačního zařízení

Zapojení je umístěno na dvouvrstvé desce plošných spojů. Tam, kde to je možné s přihlédnutím k výkonu součástky, je použito technologie povrchové montáže SMD. Tuto technologii nebylo možné použít u kondenzátorů ve výstupním obvodu cívky z důvodu většího napětového a proudového namáhání. Na obrázcích BLABLA a BLABAL je provedení desky plošných spojů, včetně způsobu rozmístění součástek.



Obr. 4.2: Deska plošných spojů, spodní strana



Obr. 4.3 Vrchní strana desky plošných spojů s rozmístěním součástek

5 Závěr

V projektu jsem navrhl a realizoval možné řešení komunikačního systému pro nevidomé s ohledem na co možná největší jednoduchost celého systému. Z toho důvodu byly také použity profesionální komunikační moduly pro bezdrátovou komunikaci, konkrétně modul ASK přijímače a bluetooth modul. Tím došlo k velkému zjednodušení projektu zejména v případě bezdrátového propojení zařízení s PDA. K úspěšnému splnění zadání bylo také nutné ovládnutí alespoň základních znalostí o programovacích jazycích C a C#. Dovoluji si tvrdit, že mnou naznačené řešení by bylo s dílčími úpravami možné použít i v praxi.

Seznam použité literatury

- [1] Philips Semiconductors: TDA5051 Home Automation Modem – Katalogový list
- [2] Analog Devices: AD8018 xDSL Line Drive Amplifier -Katalogový list
- [3] Adeunis: ARF2104 System user guide
- [4] Connct blue: OEM Serial Port Adapter™ cB-0907-01 – Katalogový list
- [5] Atmel: ATmega8 – Katalogový list
- [6] ST Microelectronics:Very low drop voltage regulators with inhibit LE33C – Katalogový list

SEZNAM PŘÍLOH

- 1.Zdrojový kód pro mikroprocesor**
- 2.Zdrojový kód aplikace Navigátor 1.0**