

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

SENZOROVÉ MODULY PRO BEZDRÁTOVOU SÍŤ ZIGBEE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

TOMÁŠ OCHMANN

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

SENZOROVÉ MODULY PRO BEZDRÁTOVOU SÍŤ ZIGBEE

SENSOR MODULES FOR ZIGBEE WIRELESS NETWORK

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

VEDOUCÍ PRÁCE

SUPERVISOR

TOMÁŠ OCHMANN

Ing. VÁCLAV ŠIMEK

BRNO 2008

Abstrakt

Cílem této práce je navrhnout a implementovat moduly pro bezdrátovou sensorovou síť ZigBee. Tyto moduly mezi sebou budou schopny komunikovat a předávat si informace o naměřených hodnotách. Budou řízeny koordinátorem sítě, který bude také zpracovávat data sensorových modulů a rozhodovat o dalších krocích v síti.

Klíčová slova

ZigBee, IEEE 802.15.4, bezdrátové sensorové sítě, TinyOS

Abstract

The aim of this thesis is to design and implement modules for wireless sensor network ZigBee. These modules could communicate together and share information about measured values. The network will be managed by coordinator of network, which will process data from sensor modules and will decide about next steps around the network.

Keywords

ZigBee, IEEE 802.15.4, wireless sensor network, TinyOS

Citace

Tomáš Ochmann: Sensorové moduly pro bezdrátovou síť ZigBee, diplomová práce, Brno, FIT VUT v Brně, 2008

Senzorové moduly pro bezdrátovou síť ZigBee

Prohlášení

Prohlašuji, že jsem tuto semestrální práci vypracoval samostatně pod vedením pana Ing. Václava Šimka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Tomáš Ochmann
31. července 2008

Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Václavu Šimkovi za pomoc při řešení tohoto diplomového projektu. Dále bych chtěl poděkovat ústavu UPSY na FIT VUT v Brně a vedoucímu ústavu Doc. Ing. Zdeňku Kotáskovi, Csc za finanční podporu při realizaci projektu.

© Tomáš Ochmann, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	4
1.1	Slovo úvodem	4
1.2	Cíl projektu	4
1.3	Struktura projektu	5
1.4	Návaznost na předchozí projekty	5
2	Senzory	6
2.1	Důležité parametry senzorů	6
2.2	IEEE 1451 a inteligentní (smart) senzory	8
2.3	Přehled senzorů a principů	9
3	Bezdrátové senzorové sítě	10
3.1	Architektura Zig-Bee	10
3.2	Bezpečnost v ZigBee sítích	14
3.3	ZigBee vs. Bluetooth	19
4	Senzorové moduly	21
4.1	Návrh modulů	21
4.2	Konstrukce bezdrátových modulů	25
5	Programové vybavení	38
5.1	TinyOS	38
5.2	Programování modulů	39
5.3	Přenosový model koordinátor – PC	42
5.4	Návrh SW vybavení	42
5.5	Zpracování sensorových dat	43
6	Závěr	47

Seznam obrázků

3.1	Porovnání základních parametrů používaných standardů pro bezdrátovou komunikaci [18]	11
3.2	Referenční model ZigBee [10]	12
3.3	Síťová vrstva ZigBee [26]	13
3.4	Příklady topologií realizovatelné standardem ZigBee [22]	14
3.5	Model ZigBee architektury [5]	15
3.6	Model klíčů v residential a commercial režimech [13]	16
3.7	Rámec po přidání zabezpečení [12]	17
4.1	Blokové schéma MCU ATmega644 [3]	23
4.2	Blokové schéma modulu koordinátor	24
4.3	Blokové schéma modulu se senzorem	25
4.4	Blokové schéma modulu přepínače	26
4.5	Stavový diagram základního operačního režimu	29
4.6	Schematické zapojení krystalu	31
4.7	Schematické zapojení obvodu FT232	32
4.8	Referenční schéma zapojení SPI rozhraní transceiveru a mikrokontroleru	33
4.9	Zapojení pin headeru s rozhráním JTAG k ATmega1281	35
4.10	Referenční schéma zapojení napájení ze sběrnice	36
5.1	Ukázka vrstevného modelu TinyOS [11]	39
5.2	Stavový automat pro testování obvodu	41
5.3	Schematické znázornění funkce systému	43
5.4	Vnitřní blokové schéma senzoru světla	44
5.5	Formát ADC registru	44
5.6	Vnitřní blokové schéma senzoru SHT11	45
5.7	Datová sekvence měření senzoru	45
5.8	Vnitřní blokové schéma senzoru zrychlení	46
5.9	Schéma zapojení SSR relé a mikrokontroleru ATmega644	46
1	Schéma zapojení modulu koordinátoru	48
2	Rozmístění součástek modulu koordinátoru – horní vrstva	49
3	Rozmístění součástek modulu koordinátoru – spodní vrstva	49
4	Schéma zapojení modulu zrychlení	50
5	Rozmístění součástek modulu zrychlení – horní vrstva	51
6	Rozmístění součástek modulu zrychlení – spodní vrstva	51
7	Schéma zapojení modulu teploty a vlhkosti	52
8	Rozmístění součástek modulu teploty a vlhkosti – horní vrstva	53
9	Rozmístění součástek modulu teploty a vlhkosti – spodní vrstva	53

10	Schéma zapojení modulu osvětlení	54
11	Rozmístění součástek modulu osvětlení – horní vrstva	55
12	Rozmístění součástek modulu osvětlení – spodní vrstva	55
13	Schéma zapojení modulu přepínače	56
14	Rozmístění součástek přepínacího modulu – horní vrstva	57
15	Rozmístění součástek přepínacího modulu – spodní vrstva	57
16	Osazené moduly	58

Kapitola 1

Úvod

1.1 Slovo úvodem

Nebývalý růst Internetu a velký komerční úspěch bezdrátových sítí dává příležitosti využívat tyto technologie čím dál více pro zlepšování a zkvalitňování poskytovaných služeb v oborech a činnostech, ve kterých to ještě před několika málo lety bylo naprosto nepředstavitelné. Pevné, drátové sítě mají již pevně vybudovanou infrastrukturu a své aplikace. Naproti tomu ad-hoc sítě, které nejsou omezeny touto pevně danou infrastrukturou, otvírají šance novým aplikacím jako domácí sítě, senzorové sítě, lokalizaci pohřešovaných osob nebo předpověď počasí. Senzorové sítě jsou především určeny pro monitorování v průmyslu, armádě, jsou určeny k hlídání objektů, požárů, přírodních katastrof, ale mohou se také využít v domácnostech pro sledování a řízení klimatizačních jednotek.

Senzorová síť je síť tvořená z mnoha inteligentních zařízení, zvaných uzly, které jsou rozmístěny v prostoru a vykonávají úlohy dané aplikací. Primárním zařízením sítě je senzor, který zjišťuje stav základních fyzikálních veličin nebo proměnných jako jsou teplota, vlhkost, tlak, vibrace, pohyb atd. Každé toto zařízení je jednoduché, malé a levné, proto může být vyráběno ve velkých sériích. Typické důležité požadavky senzorových sítí jsou energetická účinnost, kapacita paměti a výpočetní výkon. Tato inteligentní zařízení dále obsahují mikrokontrolér, rádiový přijímač a zdroj energie.

Technologie materiálů, návrhu a výroby těchto součástí pokročila natolik, že je možné umísťovat tyto systémy ve velmi nedostupných lokacích a jsou schopny fungovat i za nepříznivých podmínek několik měsíců bez jakéhokoliv zásahu poháněné například solární energií.

1.2 Cíl projektu

Cílem tohoto diplomového projektu je navrhnout a vytvořit senzorové moduly a modul koordinátoru, který bude řídit celou senzorovou síť, zpracovávat data získaná ze senzorů a rozesílat požadavky podle získaných dat ostatním modulům systému. Tyto senzorové moduly budou měřit fyzikální veličiny jako teplotu, tlak, intenzitu světla a také reagovat na pohyb senzorového modulu. Koordinátor podle potřeby zašle data do ostatních modulů, které jsou schopny spustit výstražné zařízení nebo klimatizaci. Tedy aplikace je vhodná pro zabezpečení objektů nebo pro získávání měřených dat z nedostupných míst.

1.3 Struktura projektu

Tento projekt spojuje jak části návrhu a implementaci samotného hardware, tak i softwarový návrh a implementaci. Výsledkem budou funkční moduly po HW i SW části. Tedy v jednotlivých kapitolách tohoto projektu nalezneme základní informace o senzorech, bezdrátových sítích i způsobu programování tohoto HW. V následujícím přehledu jsou popsány jednotlivé kapitoly a stručný obsah.

V první kapitole čtenář nalezne úvod k problematice a myšlenky pro tvorbu projektu. Je zde uvedena celková struktura projektu a popsány jednotlivé kapitoly. Druhá kapitola nabízí pohled do světa senzorů a čidel. Jsou zde popsány důležité vlastnosti a principy nejběžněji dostupných senzorů. V kapitole bezdrátové sensorové sítě nalezneme detailnější seznámení se ZigBee IEEE 802.15.4 komunikačním standardem pro bezdrátové sítě. Kapitola obsahuje informace o vzniku, používání a také je zde věnována velká část o bezpečnosti sítě. Ve čtvrté kapitole jsou konkrétní návrhy a schémata jednotlivých modulů. Dále je zde popsáno konkrétní provedení jednotlivých modulů, popsány detailně použité součástky a jednotlivá komunikační rozhraní. V předposlední kapitole nalezneme zmínku o operačním systému TinyOS, dále možnosti programování modulů a modely softwarového vybavení. V závěrečné kapitole je shrnutí celé práce a možnosti dalšího pokračování na projektu.

1.4 Návaznost na předchozí projekty

Tato práce navazuje na dříve vypracovaný semestrální projekt, v němž jsem se věnoval teoretickému pohledu na senzory a bezdrátové sensorové sítě (kapitola 2 a 3). Dále jsem provedl teoretický návrh sensorových modulů (kapitola 4.1).

Kapitola 2

Senzory

Zkoumání přírodních, fyzikálních, chemických a jiných vlastností prostředí je jednou z hlavních činností člověka. Díky těmto poznatkům je schopen zkvalitnit a zpřesnit výrobní technologie. V posledních desetiletích probíhá prudký vývoj v oblasti elektroniky, která ovlivňuje všechny další obory a pomáhá člověku při své každodenní práci. Ke zjišťování informací o stavu okolního světa byly stanoveny různé fyzikální a chemické veličiny, které potřebujeme měřit. Tyto veličiny se obecně měří pomocí senzorů.

Existuje spousta možností jak definovat senzor. Ve své podstatě je to však zařízení, které slouží k přeměně fyzikální veličiny ze vstupu na měřící veličinu na výstupu, nejčastěji na veličinu elektrickou. Obecně je senzor ekvivalentní pojmem snímač, převodník nebo detektor. Technicky je senzor chápán jako zařízení, které snímá danou veličinu a dle určitého principu ji transformuje na měřící veličinu. Část senzoru, která snímá stav sledované veličiny se nazývá čidlo. Zpracování provádí vyhodnocovací obvod senzoru. Výstupní veličina je dále zpracována dalšími řídicími obvody jako analogově-digitální převodník nebo mikroprocesor.

2.1 Důležité parametry senzorů

Při výběru senzorů musí návrhář vzít v úvahu různé požadavky kladené na systém. Mezi nejčastěji řešené problémy patří kompromis mezi přesností a cenou. Dále musí vzít v úvahu mechanické a elektrické vlastnosti těchto čidel, spolehlivost a také velikost, která hraje v dnešní době velmi významnou roli. Rozhodujícím požadavkem však většinou bývá vhodná závislost vstupní veličiny na výstupní a vhodný průběh základních charakteristik. V následujícím přehledu se podíváme blíže na tyto vlastnosti.

Statické vlastnosti senzorů

Statické vlastnosti senzoru popisují jeho chování v časově ustáleném stavu.

- **Přenosová funkce** – Udává vztah mezi měřenou veličinou a výstupní veličinou a nejčastější zobrazení je pomocí grafu. Lineární závislost přenosové charakteristiky je většinou pro statická měření nejvýhodnější.
- **Citlivost** – Citlivost souvisí s přenosovou funkcí. Určuje jakým způsobem se změní výstupní veličina při změně měřené veličiny. V případě lineární přenosové funkce je citlivost konstantní.

- **Práh citlivosti** – Je nejnižší hodnota měřené veličiny, která může být senzorem detekována.
- **Dynamický rozsah** – Je dán intervalem přípustných hodnot snímané fyzikální veličiny, ohraničený práhem citlivosti a maximální hodnotou měřené veličiny.
- **Linearita** – Je přesnost reálné kalibrační křivky s ideální statickou přenosovou funkcí. Je vyjádřena v procentech horní hranice měřícího rozsahu a udává maximální odchylku kteréhokoliv kalibračního bodu od odpovídajícího bodu na ideální charakteristice.
- **Hystereze** – Je maximální rozdíl ve výstupu při jakékoliv hodnotě měřeného rozsahu, kdy hodnota je měřena nejdříve při zvyšování a poté při snižování měřené veličiny. Je vyjádřena v procentech horní hranice měřícího rozsahu.
- **Reprodukovatelnost** – Je dána odchylkou naměřených hodnot za stálé (neměnné) velikosti vstupní veličiny a rušivých vlivů okolí při krátkodobém časovém sledu měření.
- **Rozlišitelnost** – Je nejmenší inkrement výstupu senzoru, který senzor zaznamená při změně výstupu. Odpovídá absolutní nebo relativní chybě senzoru.
- **Přesnost** – Vyjadřujeme relativní chybou δ vztaženou k horní hranici měřeného rozsahu.

Dynamické vlastnosti senzorů

Měřená hodnota fyzikální nebo jiné veličiny se neustále mění s časem. Sensory zařazené v regulačních smyčkách nebo indikující mezní stavy procesu musí být navrženy tak, aby výstupní signál $y(t)$ sledoval s minimálním zkreslením vstupní signál $x(t)$. V dalším budeme vycházet z předpokladu, že dynamické chování senzoru lze popsat lineární diferenciální rovnicí s konstantními koeficienty. Pokud rovnice není lineární, je nutné ji po úsecích lineari- zovat a dynamické chování sledovat v daných úsecích.

Graficky se zobrazují dynamické vlastnosti dynamickými charakteristikami:

- **Přechodová charakteristika** – Odezva na skokovou změnu vstupní veličiny. Je popisována přechodovou funkcí.
- **Rychlostní charakteristika** – Odezva na vstupní veličinu měnící se konstantní rychlostí. Popisuje ji rychlostní funkce.
- **Impulsní charakteristika** – Odezva na změnu vstupní veličiny ve formě impulsu. Popisuje ji impulsní funkce.
- **Frekvenční charakteristika** – Vyjádření chování přístroje při harmonické změně vstupní veličiny. Popisuje ji frekvenční přenosová funkce.

Průběh libovolné dynamické charakteristiky lze určit experimentálně, nebo výpočtem.

2.2 IEEE 1451 a inteligentní (smart) senzory

V dnešní době existuje na trhu velké množství výrobců senzorů a průmyslových sítí. Pro výrobce je velmi nákladné vytvářet speciální snímače pro každou síť na trhu, když by různé komponenty vyráběné různými výrobci mohly být kompatibilní. Proto v roce 1993 IEEE a Mezinárodní institut standardů a technologií začali pracovat na standardu pro inteligentní senzorové sítě[23] (Smart sensor networks). Výsledkem byl standard IEEE 1451 pro inteligentní senzorové sítě. Cílem tohoto standardu je ulehčit různým výrobcům výrobcům vyvinout smart senzory a rozhraní těchto zařízení pro snadné zařazení do sítě způsobem *plug-and-play*.

Pod standardem IEEE 1451 působí několik skupin – 1451.0 – 1451.6

- **IEEE 1451.0** – Koordinuje činnost ostatních skupin pro zachování vzájemné kompatibility dílčích částí standardu IEEE 1451 a stanovuje základní množiny funkcí a komunikačních protokolů společných všem verzím standardu.
- **IEEE 1451.1** – Popisuje možnosti snadného připojení snímače k různým typům komunikačních sítí prostřednictvím definice obecného objektového modelu pro snímač a pro tzv. smart síťový snímač spolu se specifikacemi pro tento model a současně stanovuje základní metody, typy proměnných a způsob přístupu k nim.
- **IEEE 1451.2** – Určuje hardwarové rozhraní mezi snímačem a hardwarovým modulem (zpravidla obsahujícím mikroprocesor) zprostředkovávajícím komunikaci s nadřazenou sítí. Cílem návrhu bylo vybavit snímač se zavedenou úrovní P1451.2 schopností poskytovat standardním způsobem informace o sobě samém formou tzv. TEDS. Pracovní skupina posléze charakterizovala univerzální komunikační protokol, kterým lze tyto informace nejen získat, ale také zapsat.
- **IEEE 1451.3** – Řeší zapojení několika fyzicky nezávislých snímačů, jejich časovou synchronizaci (digitální sběrnici pro přenos dat), možnost výměny za provozu (tzv. hot-swap) a s tím související automatickou identifikaci a samonastavování snímačů. Sběrnice se používá především ke sběru dat na místech, kde je v malém prostoru instalován velký počet snímačů, a také tam, kde není výhodné vést řídicí sběrnici ke každému snímači.
- **IEEE 1451.4** – Navrhuje možnosti automatické identifikace snímačů typu *plug-and-play* a použití smíšených digitálně-analogových připojení (*mixed-mode connection*).
- **IEEE 1451.5** – Doplnuje do standardu problematiku bezdrátových snímačů s předpokládanou těsnou návazností na standardy řady IEEE 802 (především IEEE 802.11, IEEE 802.15 a IEEE 802.16).
- **IEEE 1451.6** – Je standard pro funkčnost v bezpečné navrstvené síťové struktuře s několika řadiči na každé úrovni. Jako síťová transportní vrstva je sériová datová sběrnice CAN, která bývá implementována v mikrokontrolérech i jako samostatné řadiče sběrnice.

2.3 Přehled senzorů a principů

V následující tabulce 2.3 jsou znázorněny jednotlivé typy veličin a princip měření pro různé typy senzorů. Tabulka převzata z [15]

	Měřená veličina	Princip
Fyzikální vlastnosti	Tlak Teplota Vlhkost Proud	piezorezistivní, kapacitní Termistor, termomechanické Odporové, kapacitní Změna tlaku, termistor
Pohyb	Pozice Rychlost Úhlová rychlost Zrychlení	E-mag, GPS, kontaktní senzor Dopplerův princip, Hallův jev, optoelektronický Optický kodér Piezorezistivní, piezoelektrický, optická vlákna
Dotykové vlastnosti	Pnutí Síla Kroutivá síla Vibrace	Piezorezistivní Piezoelektrický, piezorezistivní Piezorezistivní, optoelektronický Piezorezistivní, piezoelektrický, optická vlákna, zvuk, ultrazvuk
(Ne)Přítomnost	Kontakt Těsná blízkost Vzdálenost Pohyb	Kontaktní spínače, kapacitní Hallův efekt, kapacitní, magnetické, seismické, akustické, RF E-mag (Sonar, radar), magnetické E-mag, IR, akustické
Identifikační	Osobní rysy Identifikace osoby	Pohled Otisky prstů, skenování sítnice, hlas

Kapitola 3

Bezdrátové senzorové sítě

Bezdrátové technologie [21] se rychle stávají rozhodující součástí počítačových sítí. Staly se otevřeným řešením poskytující mobilitu pro základní síťové služby. Tyto technologie doplňují nebo nahrazují pevná, kabelová spojení. Díky tomuto můžeme mezi sebou propojit zařízení jako jsou mobilní telefony, pagery, kapesní počítače, které nám usnadňují práci.

V mnoha průmyslových odvětvích, ve kterých je třeba něco kontrolovat, měřit nebo řídit je tato činnost velmi ztížena negativními přírodními podmínkami nebo dostupností. V těchto případech je velmi vhodné použít právě bezdrátová řešení. Díky dostupnosti embedded procesorů, mikrosenzorů, snímačů a antén s velmi nízkou spotřebou energie je možné využívat těchto sítí velmi dlouhou dobu bez zásahu člověka na velmi odlehlých místech. Další významnou výhodou je mobilita, kdy pracovník není závislý na délce kabelu, což může hrát velmi významnou roli například v bezpečnosti. Lidé také mohou využít flexibilitu, kterou jim bezdrátové řešení poskytuje, kdy mohou zpracovávat informace u zákazníka, které právě získali. Vytvoření bezdrátové sítě zabere daleko méně času a finančních prostředků. Není nutné pokládat kabely, odpadají problémy s propojením poboček firmy přes fyzické překážky jako jsou dálnice či řeky.

Stručně do historie

Historie bezdrátové komunikace spadá do dob druhé světové války, kdy Americká armáda poprvé použila rádiový přijímač pro signalizaci. Vyvinuli technologii rádiového přenosu dat, které bylo velmi těžké rozkódovat. V roce 1971 se skupina vědců z University of Havai inspirovala touto technologií a vytvořila první paketově založenou komunikační síť. Pojmenovali ji ALHONET a byla to v podstatě první lokální bezdrátová síť známá pod označením WLAN (wireless local area network). První WLAN se skládala ze sedmi počítačů, které komunikovaly pomocí dvousměrné hvězdicové topologie. Tato síť propojovala čtyři Havajské ostrovy s jedním centrálním počítačem.

3.1 Architektura Zig-Bee

Vznik Zig-Bee

V roce 2002 vznikla ZigBee Alliance a sdružuje přes 150 nadnárodních firem a korporací, které spolupracují na spolehlivých, nízkonákladových a nízkopříkonových bezdrátově

propojených systémech. ZigBee bylo schválen jako mezinárodní standard standardizační organizací IEEE a je označován jako standard IEEE 802.15.4.

Základní informace

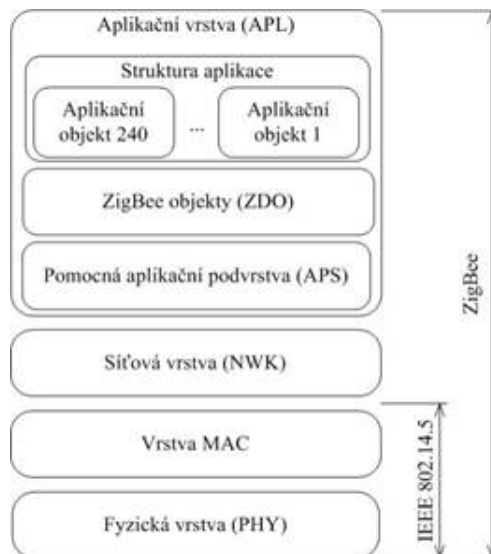
ZigBee [10] je jednoduchý bezdrátový komunikační standard, který je určen pro komunikaci mezi mnoha zařízeními na vzdálenosti desítek metrů. Díky nízkým nárokům na hardware, nízké spotřebě energie, spolehlivosti, jednoduchosti, ale také nízké ceně je vhodný pro uplatnění v průmyslu, při řízení budov, v oblasti spotřební elektroniky. Tato technologie se snaží vyplnit mezeru mezi nejrozšířenějšími technologiemi Bluetooth a WiFi, které se v řadě aplikací sice dají použít, ale nejsou vhodným řešením. ZigBee pracuje v bezlicenčních pásmech na frekvencích 858 MHz, 902-928 MHz a 2,4 GHz. Přenosová rychlost je 20, 40 nebo 250 kbit/s. Celý protokol je navržen jako jednoduchá, flexibilní technologie pro tvorbu i rozsáhlejších bezdrátových sítí u nichž není potřeba přenášet velké množství dat. Je vhodné použít ZigBee jako bezdrátovou náhradu sériového rozhraní RS-232 v průmyslu, kdy přenášíme pouze malé množství dat a tedy nám přenosová rychlost stačí a použití WiFi by bylo cenově nevýhodné. Další velmi významnou odlišností od ostatních typů bezdrátových sítí je různá topologie sítě, kterou může ZigBee vytvořit díky propracovanému způsobu adresování.

Obchodní jméno Standard	GPRS/GSM 1xRTT/CDMA	Wi-Fi™ 802.11b	Bluetooth™ 802.15.1	ZigBee™ 802.15.4
Aplikační zaměření	Široké oblasti Hlas & Data	Web, Email, Video	Náhrada za kabel	Monitorování & Řízení
Systémové zdroje (paměť)	16MB a více	1MB a více	250KB a více	4KB - 32KB
Životnost baterií (dny)	1-7	0.5 - 5	1 - 7	100 – 1 000 i více
Max. velikost sítě (počet uzlů/sít)	1	32	7	65 000 (příp. až 2 ⁶⁴)
Přenosová rychlost (Kb/s)	64 – 128	11 000	720	20 - 250
Komunikační dosah (m)	1 000 i více	1 - 100	1 - 10	1 – 100
Výhody	Dosažitelnost, Kvalita	Rychlost, Flexibilita	Cena, Jednoduchost	Spolehlivost, Výkon/Cena

Obrázek 3.1: Porovnání základních parametrů používaných standardů pro bezdrátovou komunikaci [18]

Referenční model ZigBee

Referenční model standardu ZigBee 3.2 vychází ze sedmivrstvého modelu ISO/OSI. Lze ho rozdělit však jen do třech základních bloků. Standard IEEE 802.15.4 definuje dvě nejnižší vrstvy - fyzickou (PHY) a podvrstvu MAC spojové vrstvy. Nad těmito vrstvami definuje ZigBee Alliance síťovou (NWK) a aplikační (APL) vrstvu.



Obrázek 3.2: Referenční model ZigBee [10]

Fyzická a MAC vrstva standardu IEEE 802.15.4

Fyzická vrstva IEEE 802.15.4 definuje několik základních rádiových pásem pro použití v různých zemích, kde jsou rozdílné národní standardy.

- pásmo ISM 2.4 GHz, 16 kanálů, přenosová rychlost 250kb/s, definováno celosvětově
- pásmo 915 MHz, 10 kanálů, přenosová rychlost 40kb/s, definováno pro americký kontinent a Austrálii
- pásmo 868 MHz, 1 kanál, přenosová rychlost 20kb/s, definováno pro Evropu

Celkový počet využitelných kanálů v těchto pásmech je 27. V pásmu 868 MHz je možné využít jeden kanál s přenosovou rychlostí 20 kb/s, v pásmu 915 MHz 10 kanálů s přenosovou rychlostí 40 kb/s a zbývajících 16 v pásmu 2,4 GHz s přenosovou rychlostí 250 kb/s

Pro přenos se datový signál moduluje metodou O-QPSK a vzduchem se přenáší prostřednictvím DSSS (Direct Sequence Spread Spectrum). Pro přístup na kanál se využívá metody CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance and optional time slotting).

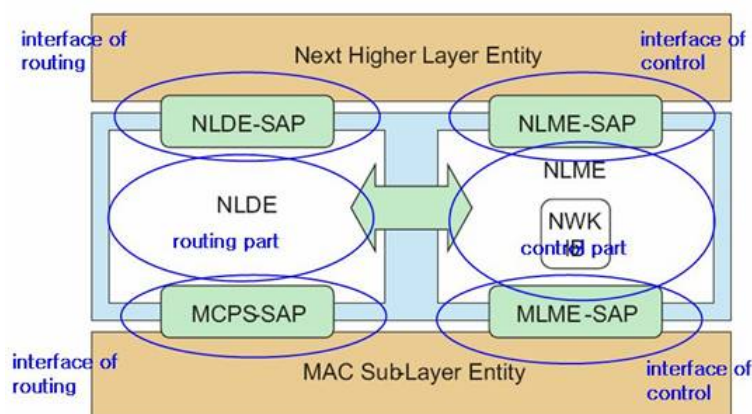
Vrstva MAC provádí synchronizaci, zabezpečuje přístup na rádiový kanál, ověřuje platnost rámce, potvrzuje příjem rámce, řídí spojení, generaci a rozpoznání adres. Jsou definovány čtyři typy komunikačních rámců využívané buď pro přenos užitečných datových informací, nebo k režijním účelům souvisejícím se sestavením, správou a řízením sítě.

- **MAC Command Frame** – slouží k nastavení a řízení klientských zařízení v síti.
- **Data Frame** – slouží pro všechny datové přenosy, je možné jej použít pro přenos až 104 byte užitečných dat
- **Acknowledgment Frame** – slouží pro potvrzení úspěšně přijatého rámce. Jsou vysílány okamžitě po příjmu datového rámce v čase mezi rámci (IFS – Interframe Space)

- **Beacon Frame** – rámeček používaný koordinátorem k vysílání tzv. beacons (používané pro uvádění klientských zařízení do spánkového režimu)

Síťová vrstva

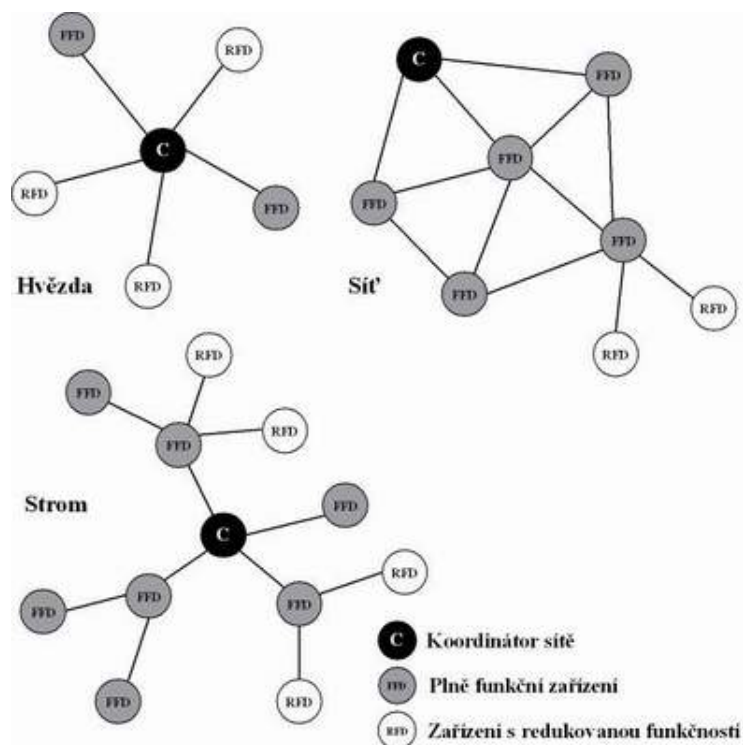
Síťová vrstva poskytuje dvě služby dostupné přes přístupové servisní body (service access point - SAP). Network data service, která podporuje přenos dat mezi ostatními zařízeními, a network management service, která podporuje služby řízení mezi zařízeními v rámci sítě. Tyto služby zajišťují rozhraní mezi fyzickou MAC vrstvou a aplikační vrstvou v rámci jednoho zařízení. Existuje také rozhraní mezi těmito službami na síťové vrstvě.



Obrázek 3.3: Síťová vrstva ZigBee [26]

Topologie sítě

Síťová vrstva ZigBee podporuje hvězdicovou, stromovou nebo mesh (sít') topologii 3.4. Při vytvoření hvězdicové topologie je síť řízena jedním zařízením, které se nazývá ZigBee koordinátor. Koordinátor je zodpovědný za vytvoření a udržování spojení mezi zařízeními v síti a také s ním všechny ostatní zařízení, známé jako koncové zařízení, přímo komunikují. V mesh a stromové topologii je ZigBee koordinátor zodpovědný za spouštění sítě a stanovuje parametry sítě. V síti však mohou být ještě ZigBee routery, které ji dále rozšiřují. Ve stromové topologii sítě se pro přenos dat a řídicích zpráv používá hierarchické směrování. Stromové sítě mohou také využívat *beacon* orientovanou komunikaci, která umožňuje synchronizaci sítě. Koordinátor pravidelně vysílá signál, který koncová zařízení využívají k připojení k síti a vlastní synchronizaci pro následný přenos dat. Pro navázání spojení se používají superrámce popsané v předchozí kapitole. V *non-beacon* sítích se také periodicky posílá signál, avšak ten slouží jenom k jeho vlastní identifikaci a koncovým zařízením k detekci. Koncová zařízení komunikují s koordinátorem pomocí požadavků na vysílání dat a potvrzovacích rámců. Musí být neustále připraveny podporovat komunikaci mezi rovnocennými uzly. Mesh sítě také mohou dovolovat plně peer-to-peer komunikaci. ZigBee routery však v těchto mesh sítích nevysílají běžné beacon rámce. Tato specifikace popisuje pouze intra-PAN sítě, což jsou sítě, ve kterých komunikace začíná a končí uvnitř, v rámci jedné sítě.



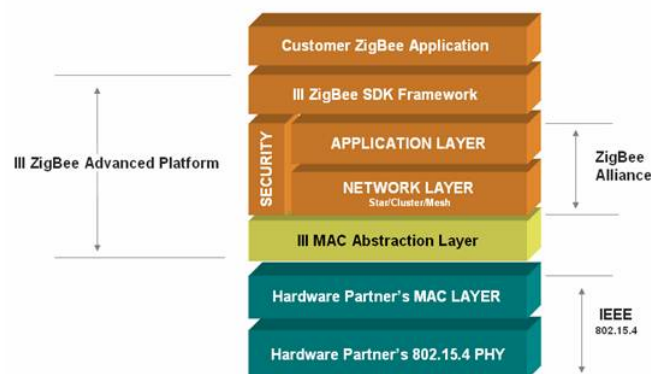
Obrázek 3.4: Příklady topologií realizovatelné standardem ZigBee [22]

3.2 Bezpečnost v ZigBee sítích

Bezpečnost a integrita dat jsou klíčovými výhodami technologie ZigBee. Jak se můžeme dočíst v ZigBee specifikaci [27], bezpečnost je přímou součástí této architektury. Bezpečnost má na starosti síťová a aplikační vrstva, což můžeme vidět na obrázku 3.5, který znázorňuje model architektury. V současné specifikaci je možné použít tři úrovně zabezpečení: žádné, seznam pro řízení přístupu (Access Control List) – ACL a šifrovací standard Hash Message Authentication Code – Advanced Encryption Standard (HMAC–AES). Je využíván koncept distribuce klíčů mezi jedním speciálním zařízením (trust center) a ostatními zařízeními. Můžeme použít linkové a síťové klíče. Jsou podporovány operace autentifikace a šifrování. Bezpečnost může být přizpůsobena podle aplikace a klíče mohou být pevně zadané v aplikaci.

Skupiny bezpečnostních specifikací

Jestliže zařízení pracuje v bezpečném režimu, je možné zařízení zabezpečit některým ze sady bezpečnostních specifikací [1]. Bezpečnostní sada obsahuje skupinu operací, které se provedou na MAC rámci a poskytnou dané zabezpečení. Název skupiny obsahuje název šifrovacího algoritmu, mód a počet bitů kontrolního součtu. Velikost kontrolního součtu je menší nebo rovna velikosti bloku symetrického šifrovacího algoritmu a určuje pravděpodobnost, že odhad integritního kódu je správný. Velikost kontrolního součtu však neudává sílu výchozího algoritmu. Pro všechny skupiny bezpečnostních specifikací v tomto standardu se používá šifrovací algoritmus Advanced Encryption Standard (AES). Každé zařízení, které pracuje v zabezpečeném módu musí provést AES-CCM-64 a buď žádnou nebo některou



Obrázek 3.5: Model ZigBee architektury [5]

Kód	Bezpečnostní skupina	Mechanismy zabezpečení			
		Kontrola přístupu	Šifrování	Integrita zpráv	Freshness check
0x00	Žádná				
0x01	AES-CTR	ANO	ANO		ANO
0x02	AES-CCM-128	ANO	ANO	ANO	ANO
0x03	AES-CCM-64	ANO	ANO	ANO	ANO
0x04	AES-CCM-32	ANO	ANO	ANO	ANO
0x05	AES-CBC-MAC-128	ANO		ANO	
0x06	AES-CBC-MAC-64	ANO			ANO
0x07	AES-CBC-MAC-32	ANO			ANO

Tabulka 3.1: Tabulka skupiny bezpečnostních specifikací

další ze skupiny bezpečnostních specifikací. Každá bezpečnostní specifikace je určena kódem, který ukazuje tabulka 3.1.

Mechanismy zabezpečení v ZigBee

Prvním způsobem ochrany je *Freshness check* [9]. Jedná se o ochranu před útoky založené na odpovědích od některého zařízení (reply attack). Útočník zašle zařízení nějakou zprávu a čeká na odpověď. Při vytvoření velké komunikace mezi zařízeními může útočník zjistit potřebný klíč k připojení do sítě. ZigBee zařízení si udávají počítadla příchozích a odchozích zpráv a v případě překročení limitů vytváří spojení s novým klíčem.

Integrita zpráv chrání před pozměňováním zpráv útočníkem během přenosu. Používá 0, 32, 64 nebo 128 bitovou kontrolu integrity zprávy CRC kontrolního součtu. Jako základní hodnota je nastaveno 64 bitové

Kontrola přístupu pomocí autentifikace poskytuje záruku o původci zprávy. Zamezuje útočníkovi vydávat se za jiné zařízení. Autentifikace je možná na síťové úrovni nebo na úrovni zařízení. Autentifikace na síťové úrovni je zajištěna pomocí tzv. síťových klíčů, které jsou popsány v následující podkapitole. Každé dvě zařízení, které spolu komunikují, mají

unikátní pár klíčů.

Šifrování slouží jako ochrana proti odposlechu zpráv. ZigBee používá 128 bitové AES šifrování. Šifrování je možné jak na síťové úrovni, tak přímo v zařízení.

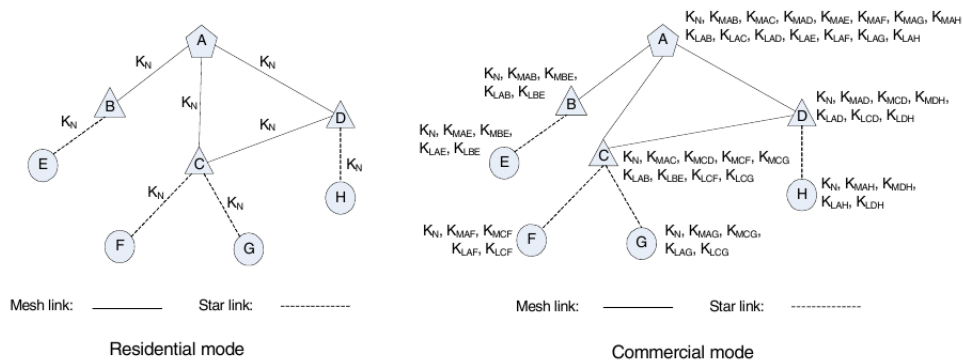
Architektura šifrovacích klíčů v ZigBee

Jsou definovány tři typy klíčů [13]. Master key je dlouhodobý bezpečnostní klíč, který je mezi dvěma zařízeními. Může být nastaven ručně, pevně při výrobě nebo si mohou tento klíč zařízení navzájem zaslat. Master key se používá pro předávání síťových a linkových šifrovacích klíčů. Linkové klíče slouží k zajištění bezpečnosti na určité lince mezi dvěma zařízeními. A poslední, síťové klíče slouží k zabezpečení sítě a chrání před útoky zvenčí. Linkový i síťový klíč může být také pevně nastaven při výrobě. Oba tyto klíče mohou být pravidelně aktualizovány. Při připojení nového zařízení do sítě je třeba nastavit tyto dva klíče.

Centrální prvek zabezpečení – trust centre je zařízení v ZigBee síti, které povoluje zařízením připojit se k síti a je zodpovědné za distribuci šifrovacích klíčů. Trust centre může pracovat ve dvou režimech. Residential a commercial režim. Na obrázku 3.6 je vidět struktura ZigBee sítě a rozdíl v jednotlivých režimech.

Residential režim je způsob zabezpečení, kdy je použitý pouze jeden síťový klíč K_N . Všechna zařízení v síti šifrují a dešifrují komunikaci pomocí toho sdíleného klíče. Výhodou tohoto přístupu jsou nižší náklady na komunikaci a nízké nároky na trust centre, avšak tento přístup je snadno napadnutelný útočníkem zvenku při připojování nového účastníka do sítě, kdy komunikace probíhá bez šifrování. Tento přístup je tedy vhodný v aplikacích, kde není požadována vysoká míra bezpečnosti.

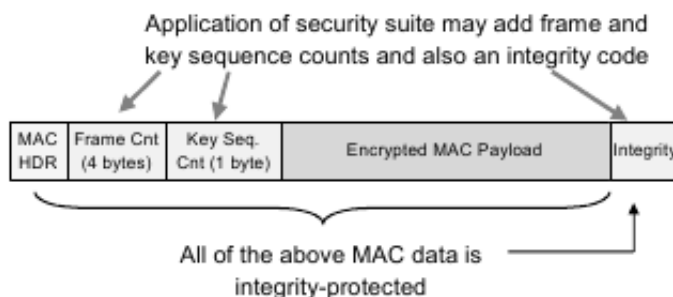
Commercial režim je oproti předchozímu vhodný pro aplikace, kde je kladen vysoký důraz na bezpečnost. V tomto řešení je zapotřebí více klíčů k ochraně komunikace. Také se však zvýší nároky na kapacitu přenosového kanálu. Jako u předchozího se používá jeden síťový klíč K_N . Dále se používají K_{MXY} master klíče, což je sdílený klíč mezi dvěma uzly a K_{LXY} , což je linkový klíč, který je třeba k šifrování linky mezi dvěma uzly.



Obrázek 3.6: Model klíčů v residential a commercial režimech [13]

Bezpečnostní služby jednotlivých vrstev

K ochraně spolehlivosti a odolnosti síťové infrastruktury je třeba zabezpečit zprávy na MAC vrstvě [12]. Základním zabezpečením dat a příkazů na MAC vrstvě je kontrola integrity rámce a ochrana proti reply útokům. Úroveň ochrany integrity se může lišit dle typu a účelu sítě. Počet integritních bitů je 32, 64 nebo 128. Dále, přijímací strana musí mít možnost ověřit identitu vysílající strany. Používá se Cipher Block Chaining (CBC) režim. Ne vždy je třeba zabezpečit samotné zprávy, avšak i toto lze nastavit. Doručena zpráva je akceptována přijímací stranou až poté, co úspěšně projde přes šifrování a je ověřena totožnost odesílatele. Jako šifrovací proces je použito sdíleného klíče mezi odesílatelem a příjemcem. Je použit standard Advanced Encryption Standard (AES) a bloková šifra využívající symetrické šifrování. Šifrování AES je v Counter (CTR) režimu. Ověření totožnosti odesílatele se ověřuje v seznamu zařízení odesílatelů, které se nazývá Access Control List (ACL). Jestliže MAC vrstva obdrží rámec, který je šifrovaný, podívá se na to, od koho rámec přišel a vybere klíč pro dešifrování z databáze. Zkontroluje integritu rámce, poté dešifruje data z rámce a informuje vyšší vrstvy o tom, že došel rámec a jak byl zabezpečený. Na obrázku 3.7 je vidět, jak se k rámcu přidávají kontrolní a zabezpečovací informace. Je vidět, že se šifrují jenom data a ne celý rámec, ale integrita se počítá pro celý rámec. Počítadlo rámců (Frame Counter) slouží k ochraně proti reply útokům. Vysílací strana pokaždé toto počítadlo zvýší, přijímací strana si pamatuje poslední stav počítadla a jestliže po zvýšení o jedničku tyto údaje nesouhlasí, je signalizována bezpečnostní chyba.



Obrázek 3.7: Rámec po přidání zabezpečení [12]

Možnosti zabezpečení síťové vrstvy jsou obdobné jako u MAC vrstvy. Protože zprávy síťové vrstvy mohou putovat po celé síti, než dorazí k adresátovi, je nutné opět šifrovat jenom data, aby bylo možné přistoupit k údajům v hlavičce nutným k předávání rámce po síti. Je tedy nutné aby zařízení důvěřovalo všem ostatním zařízením v celé síti. Jsou situace, kdy je vyžadováno také zabezpečení samotné aplikace, tedy na aplikační vrstvě. Samotná aplikace může využívat zabezpečení nižších vrstev nebo použít vlastního zabezpečeného aplikačního rámce, který má strukturu velmi podobnou jako nižší vrstvy. Je třeba také myslet na to, že rámec může putovat sítí, než dorazí ke koncovému zařízení, a proto se opět šifrují pouze data.

Protokoly založené na klíčích

V ZigBee se využívá dvou protokolů, symetrického klíče a veřejného klíče [12]. Toto dává na výběr při vytváření sítě podle požadavků na míru zabezpečení. Symetrické šifrování je jednodušší na implementaci a nemá takové výpočetní nároky, zatímco metody šifrování pomocí veřejného klíče jsou složitější na implementaci a potřebují také vyšší výpočetní výkon, avšak jsou zase více flexibilní a rozšiřitelné a lépe odolávají případným útokům. Hodí se na použití v rozsáhlých sítích a v sítích, kde je nutná velká míra zabezpečení. Šifrování pomocí symetrického klíče je vhodné do menších sítí, do domácností, kde není nutné takové zabezpečení.

Pro vytvoření linkových klíčů symetrického klíče je možné použít dvě metody. Symetrick – Key Key Establishment (SKKE) – metoda symetrického klíče a metoda nechráněného klíče – Unprotected Key Establishment (UKE). V prvním případě se ověřování děje pomocí sdílených, tajných a symetrických klíčů, které vychází z master klíče. Tento master klíč může být například nahraný při výrobě, může být vygenerovaný použitím některé z šifrovacích metod nebo se může vytvořit z uživatelských dat, např. zadáním PIN nebo hesla. Druhá metoda, UKE, používá jako master klíč standardní hodnotu. V tomto případě možný útočník může tento klíč zjistit. Ověřování se v tomto případě děje pomocí znalosti IEEE 64-bitové adresy druhého zařízení, která se získá při počáteční komunikaci. Tato metoda není příliš bezpečná, protože lze při počáteční komunikaci odchytil potřebné údaje k vytvoření klíčů. Tato metoda je vhodná pro aplikace, kde je malé riziko napadení sítě a v implementacích, kde nelze měnit master klíč, například v systémech bez přepisovacích pamětí Flash.

K vytváření klíčů protokolu veřejného klíče se nabízí také dvě metody. Metoda veřejného klíče – Public-Key Key Establishment (PKKE) a metoda založená na certifikátech – Certificate-Based Key Establishment (CBKE). První metoda se používá jako prevence proti pasivním útokům, které zahrnují odposlechy při vytváření klíčů. Důvěra mezi zařízeními se vytváří výměnou veřejných klíčů s odpovídajícím ID zařízení. V této metodě každý poskytne do sítě veřejný klíč s ID zařízení. Kdo bude chtít s tímto zařízením komunikovat, zašifruje tímto veřejným klíčem zprávu a pošle zařízení. Rozšifrovat tuto zprávu však může pouze zařízení, kterému byla zpráva poslána, protože jenom toto zařízení má privátní klíč. Tato metoda je obzvláště výhodná v případech, kdy není možnost použít certifikační autoritu – Certification Authority (CA). Metoda založená na certifikátech používá technologii veřejných klíčů se základními klíči (root key). Certifikát bývá většinou veřejný klíč s 64 – bitovou IEEE adresou zařízení podepsaný CA. Certifikáty poskytují způsoby kontroly komu patří který veřejný klíč a jestli je dané zařízení platným zařízením sítě. Pro obě metody vytváření klíčů, PKKE i CBKE, je doporučeno použití techniky 163-bitové Elliptic-Curve Cryptographic (ECC). Tato technika má nízké výpočetní nároky a klíče malé délky.

Zabezpečení sítě

Jelikož se jedná o bezdrátovou síť, tedy přenosové médium je vzduch, může docházet během přenosu signálu k chybám. Pro odhalení a opravu chyb se používá cyklického kódování (CRC nebo FCS). Při tomto kódování se ke každému rámcu přidá zbytek po dělení polynomem. Ve standardu ZigBee je použit polynom ve tvaru $G_{16}(x) = x^{16} + x^{12} + x^5 + 1$. K zabezpečení se používá celý rámec včetně jeho záhlaví.

Pro zvýšení spolehlivosti přenášených dat se vysílání provádí technologií DSSS (Di-

rect Sequence Spread Spectrum). Jednotlivé bity jsou nahrazeny početnější sekvencí bitů (chipů), které se pak vysílají. Signál je tak rozprostřen do větší části spektra a je více odolný vůči rušení. Uživatelům, kteří neznají mechanismus vytváření pseudonáhodné sekvence se přenášená data jeví jako šum.

Přenášená data mezi jednotlivými účastníky lze dále zabezpečit proti zcizení. Standard definuje tři režimy a to nezabezpečený přístup, přístup na základě práv a zabezpečený přístup. Je-li použit přístup na základě práv, pak síť odmítá rámce od neznámých zařízení. V zabezpečeném režimu mohou zařízení sítě využít další služby mezi které patří:

- přístup na základě práv
- šifrování dat pomocí AES 128 bit
- použití MIC (Message Integrity Code)
- odmítnutí opakujících se rámců (Sequential freshness)

Šifrovací standard AES (Advanced Encryption Standard) nahrazuje standard DES. Výhodou tohoto nového způsobu šifrování je, že nehrozí útok hrubou silou (tj. vyzkoušení všech možných klíčů).

MIC je kryptografický kontrolní součet, který je zahrnut do vysílaného rámce. Na přijímací straně se provádí stejná operace a hodnotu součtu porovnává s přijatou. Pokud se zpráva během přenosu změnila, budou se hodnoty lišit a rámec je odmítnut.

3.3 ZigBee vs. Bluetooth

Již delší dobu je technologie Bluetooth provozována také v nelicenčním kmitočtovém pásmu 2,4 GHz. Na první pohled to tedy vypadá, že ZigBee technologie je zbytečná. Mezi těmito technologiemi je však značný rozdíl a každá je dle svých atributů využívána v jiné oblasti.

Standard	Kmitočtové pásmo [MHz]	Přenosová rychlost [kb/s]	Typ použité modulace
Bluetooth IEEE 802.15.1	2400	$1 \cdot 10^3$	GFSK
ZigBee IEEE 802.15.4	868 (EU)	20	BPSK
	915 (USA)	40	BPSK
	2.400	250	O-QPSK

Tabulka 3.2: Porovnání vlastností ZigBee a Bluetooth

Technologie Bluetooth pracuje se sítěmi typu piconet, což umožňuje připojit k řídicí stanici pouze sedm aktivních podrízených stanic. Ostatní zařízení v síti mohou být, avšak jsou pouze v neaktivním stavu, tzv. *parked*. Další možnou topologií je scatternet, která sdružuje několik sítí piconet pomocí sdílených zařízení. ZigBee však může vytvořit topologii strom, a tím tedy výrazně zvětšit funkční dosah sítě. Řídicí stanici můžeme propojit s až 255ti zařízeními a lze komunikovat bez koordinátora.

Jiným, velmi významným rozdílem je přenosová rychlost Bluetooth, která je v řádu megabitů. Pro verzi 1.2 je rychlost 1Mb/s, verze 2.0 by měla podporovat rychlosti až 3Mb/s. U ZigBee jsou rychlosti 20, 40, resp. 250 kb/s. U Bluetooth je pro přenos použita modulace GFSK s technologií FHSS(Frequency Hopping Spread Spectrum) na rozdíl od ZigBee, kde je použita technologie DSSS.

Porovnání [10] technologie Bluetooth a ZigBee je uvedeno v tabulce 3.2. Technologie Bluetooth je vhodná např. pro přenosy multimediálních dat, kdy je potřeba využít kapacity kanálu a přenosové rychlosti v co největší míře. Není však příliš vhodná do aplikací, ve kterých je nutné tvořit rozsáhlé sítě, protože je zde omezení 7 aktivních zařízení. Používá se hlavně pro spojení dvou zařízení PEER-TO-PEER nebo Ad-Hoc. Kdežto standard ZigBee je díky svým vlastnostem vhodný pro použití v oblasti automatizace a průmyslu, kde není nutné přenášet tak velké množství dat.

Kapitola 4

Senzorové moduly

4.1 Návrh modulů

Analýza problému a specifikace požadavků

Vstupem tohoto systému budou naměřená data z okolního prostředí a výstupem bude vyhodnocení těchto dat a na jejich základě příslušná akce, například při snížení/zvýšení teploty se zapne topení/větrání. Celý systém se bude využívat jako systém pro automatickou kontrolu a zajištění obydlených i neobydlených objektů. Uživatel bude moci nastavit prahové hodnoty měřených veličin přes jednoduchou počítačovou aplikaci a také v této aplikaci sledovat měřená data. Přenášená data nejsou kritická na bezpečnost, tudíž není nutné toto uvažovat. Celý systém bude fungovat ve standardním rozložení teplot a nebudou kladeny zvýšené nároky na součástky pro práci v rizikovém prostředí. Chyby v systému budou vyhodnoceny a hlášeny uživatelské aplikaci jako nestandardní události.

Je nutné určit jaké veličiny by měly senzory měřit. Mezi základní a jednoduše dostupné senzory těchto veličin patří teplota, vlhkost, osvětlení nebo také senzor detekující pohyb dané součástky, tzv. accelerometer sensor. Jiné typy senzorů jsou buďto velmi drahé, hůře dostupné, mají složité zapojení nebo jsou rozměrné. Vybrané typy senzorů jsou dostupné u velkých distributorů, za poměrně nízkou cenu a schéma zapojení se skládá v podstatě jenom z daného senzoru připojeného na napájení a přes komunikační rozhraní nebo A/D převodník k mikrokontroléru.

Pro požadavek bezdrátového přenosu dat se přímo naskýtá řešení bezdrátové sítě ZigBee. Je to velmi jednoduchý a přitom mocný protokol pro přenos malého objemu dat při nízkých přenosových rychlostech. Jeho architektura naskýtá možnosti šetření energie kvůli možnostem usnutí celého systému a uvedení do provozu v intervalech od desítek ms do přibližně 15ti minut. Ostatní bezdrátové systémy jako wifi nebo bluetooth nejsou příliš vhodné pro tuto aplikaci, protože jsou to robustní systémy s vysokými požadavky na výpočetní výkon a tedy daleko vyššími nároky na spotřebu a velikosti MCU – hlavní části modulu. Dále je zbytečné používat tyto technologie, když velikost přenášených dat bude velmi malá a na vysoké přenosové rychlosti nám také nezáleží. Technologie ZigBee je výhodná také v tom, že dokáže vytvářet sítě typu strom s až 255 zařízeními oproti 7 zařízením v bluetooth sítích.

Jako řídicí jednotka celého systému se nejlépe hodí mikrokontrolér. Požadavek na jednoduchost a nízkou spotřebu vylučuje robustní 32bitové MCU s CISC architekturou. Daleko

vhodnější pro tuto aplikaci by byl MCU co nejmenší, 8bitový s RISC architekturou, který bude mít také daleko nižší spotřebu. Měl by mít také vhodné komunikační rozhraní pro práci se senzory a transceiverem. Je nutné vyřešit interakci s uživatelem, tedy sběr dat a přenos dat do osobního počítače. Pro tento účel je nejjednodušší a nejdostupnější rozhraní USB.

Celý systém bude poměrně komplikovaný a proto je v tomto případě velmi výhodné využití operačního systému vhodného pro tyto účely. Z pohledu MCU a dostupnosti OS se jeví jako vhodný kandidát TinyOS. Je konfigurovatelný pro dané účely a dokáže pracovat se ZigBee architekturou.

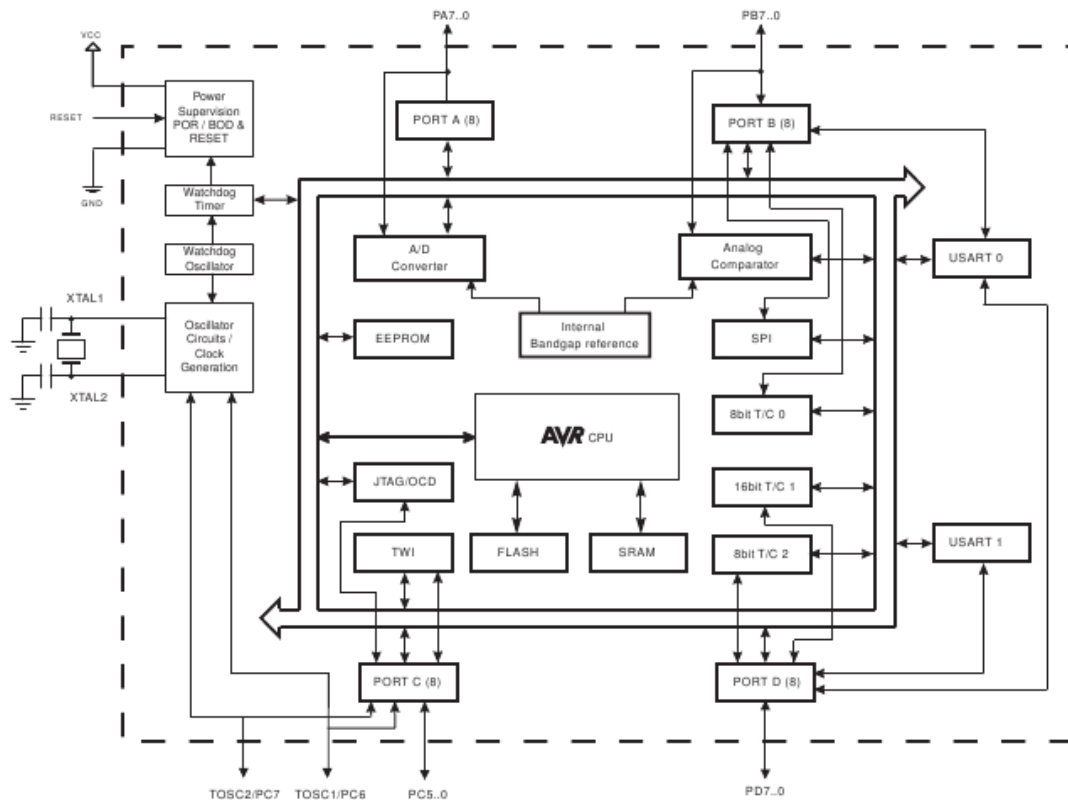
Při návrhu byl kladen důraz na jednoduchost a velikost jednotlivých modulů. Byly vybrány komponenty, které jsou dobře dostupné na trhu a také jejich cena je přijatelná. V následující kapitole je popsán souhrn nejdůležitějších součástí jednotlivých modulů.

Celý systém bude řízen operačním systémem TinyOS, který poběží na každém z jednotlivých modulů. Výhodou tohoto řešení je, že do TinyOS je možné zakomponovat ZigBee modul. ZigBee standard sice podporuje několik typů zařízení, které tvoří síť, jako jsou koncová zařízení, routery a koordinátory, ale ve skutečnosti toto řešení není zcela jednoduché. Ve větších sítích je třeba mít také routery, které musí dočasně ukládat data a poté přeposílat na jiná zařízení, která například v daný moment nejsou k dispozici. V některých případech je nutné provádět také různá zpracování dat a tím značně zatížit procesor. Dalo by se použít výkonnějších procesorů, avšak nasazením operačního systému docílíme velkého zjednodušení správy celého systému a sítě a zlepšení synchronizace jednotlivých součástí systému.

MCU

Pro tuto práci jsem vybral mikroprocesory od firmy Atmel[3] řady AVR. Jsou to 8–mi bitové mikroprocesory harwardského typu s jádrem RISC a vykonávají instrukci za jeden cyklus. Mají přesně stanovenou strukturu vstupně/výstupních operací, která omezuje potřebu externích komponent. AVR mikroprocesory obsahují všechny důležité součástky potřebné pro běh. Instrukční sada je velmi zúžena a nezáleží na tom, zda je kód psaný v assembleru nebo jazyce C. Na trhu jsou dostupné procesory v několika variantách velikosti paměti, napájení, rychlosti a pouzder. Byly vybrány procesory ATmega644 a ATmega1281. Základní parametry zvolených MCU jsou ukázány v následujícím přehledu.

- rychlost CPU 10MHz
- napájení 1.8V – 5.5V
- velikost vnitřní paměti FLASH 64KB resp. 128KB
- 32 pracovních registrů
- 6 režimů spánku
- dva 8–mi bitové a jeden 16–ti bitový čítač/časovač
- 8–mi kanálový, 10–ti bitový AD převodník
- rozhraní JTAG



Obrázek 4.1: Blokové schéma MCU ATmega644 [3]

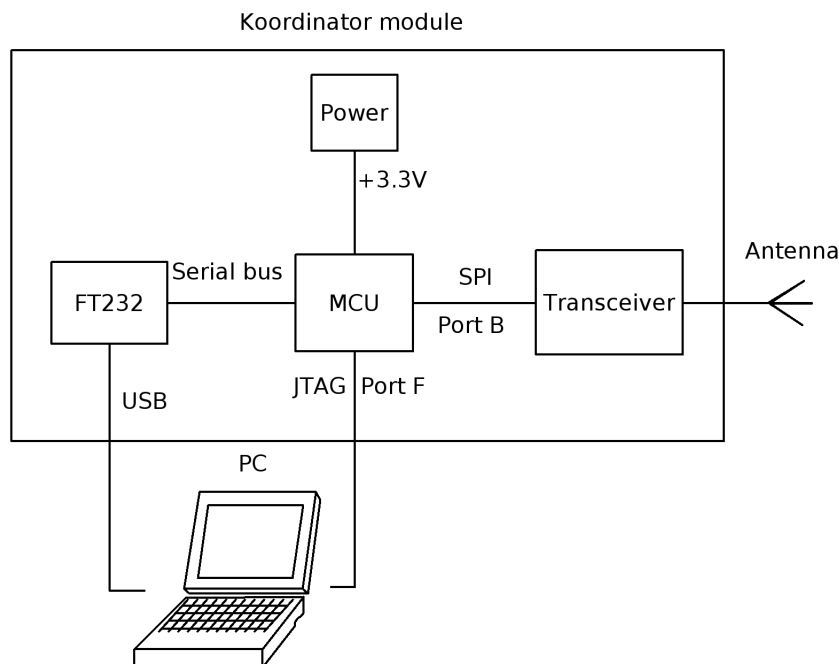
Transceiver

Pro ZigBee komunikaci slouží také čip od firmy Atmel, a to AT86RF230. Je to nízko-
příkonový transceiver pro frekvenci 2.4GHz speciálně vyráběný pro low-cost ZigBee/IEEE
802.15.4 aplikace. Čip obsahuje kompletní fyzickou(PHY) vrstvu ZigBee/IEEE 802.15.4,
IEEE standard pro tvorbu peer-to-peer, hvězda a mesh sítě. K čipu je nutné připojit
externí anténu, krystal a de-coupling kondenzátory(baloon). V kombinaci s Atmel AVR
mikroprocesory je vhodné jej použít v těchto aplikacích.

- bezdrátové sensorové sítě
- průmyslové řízení
- automatizace domácností a budov
- spotřební elektronika
- počítačové příslušenství

Modul koordinátoru

Na blokovém schématu 4.2 vidíme 4 základní bloky. Jsou to Samotné MCU – ATmega1281,
transceiver AT86RF230, modul rozhraní pro usb komunikaci RF232R a napájení 3.3V.



Obrázek 4.2: Blokové schéma modulu koordinátor

Tento modul by měl řídit celý systém, sbírat veškerá data ze sensorových modulů a rozesílat signály přepínacím modulům. Mikroprocesor pro tento modul bude použit ATmega1281 na rychlosti 8MHz a s kapacitou vnitřní paměti 128KB. Data se budou přenášet do počítače přes rozhraní USB. Pro programování a zjišťování stavu modulu slouží rozhraní JTAG. Jelikož se počítá se stálým připojením tohoto modulu, bude napájen ze sběrnice USB. Velikost tohoto modulu by měla být 5x5cm.

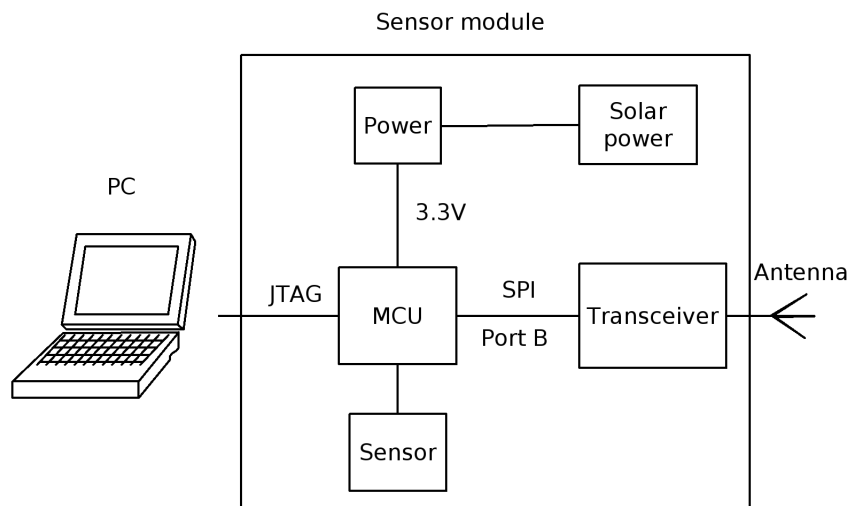
Senzorový modul

Následující blokové schéma 4.3 zobrazuje základní prvky modulu se senzorem. Základní součásti tohoto modulu jsou MCU ATmega644 na rychlosti 8MHz a s kapacitou vnitřní paměti 64KB, transceiver AT86RF230, senzor a případný solární panel pro dobíjení baterií. Pro programování slouží také rozhraní JTAG. Jestliže bude tento modul dobíjen ze solárního panelu, provoz by měl být i několik let. Velikost by měla být přibližně stejná jako u modulu koordinátoru, tedy 5x5cm.

Pro tuto práci jsou vybrány 3 typy senzorů. Bude možné měřit teplotu s vlhkostí, intenzitu světla a také senzor, který měří zrychlení. Tedy bude možné měřit například vibrace nebo detekovat pohyb modulu.

Teplota, vlhkost

Pro měření teploty a vlhkosti slouží senzor od firmy Sensirion. Výstup senzoru je digitální a je připojen k mikrokontroleru přes rozhraní TWI a poté zpracován do hodnoty relativní vlhkosti nebo teploty. Změna měřené fyzikální veličiny se provádí zapáním bitu do řídicího registru senzoru. Výstup relativní vlhkosti je téměř lineární a výstup teploty je zcela lineární.



Obrázek 4.3: Blokové schéma modulu se senzorem

Intenzita světla

Intenzita světla bude měřena senzorem firmy Taos. Tento senzor je složen z fotodiody a vysokoimpedančního zesilovače a tedy přímo mění intenzitu světla na napětí. Výstup senzoru je lineární a bude také přiveden rohraním TWI do mikrokontroleru.

Zrychlení

Modul, který bude měřit zrychlení bude osazen čipem od firmy Freescale, který dokáže detekovat zrychlení ve všech 3 osách. Senzor funguje na principu změny kapacit, kdy při pohybu se mění kapacity jednotlivých volných ploch. Výstup z této součástky je analogový, tedy je nutné jej přivést na AD převodník mikrokontroleru a zpracovat do digitální podoby.

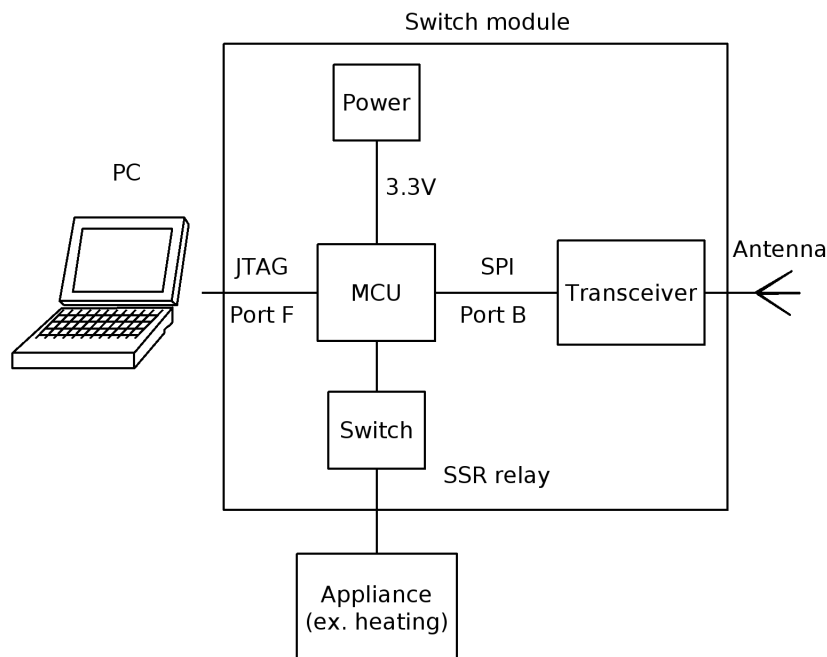
Přepínací modul

Základní bloky tohoto modulu 4.4 jsou také MCU ATmega644 a transceiver AT86RF230, jako u předchozího modulu. Předpokládá se stále připojení k napětí 220V, tedy je zbytečné bateriové napájení. Namísto senzoru jsem umístil relé, které dokáže spínat potřebný spotřebič při signálu z MCU.

Tento modul slouží pro spínání napětí 220V při potřebné události. Například, když senzorový modul detekuje snížení intenzity světla, tomuto modulu se pošle signál pro zapnutí osvětlení a pomocí relé sepne obvod se žárovkami.

4.2 Konstrukce bezdrátových modulů

K výrobě desek plošných spojů jednotlivých modulů bylo třeba vzít v úvahu několik základních požadavků. Velmi důležitý je výběr vývojového prostředí pro návrh DPS (desky plošného spoje). Jako vývojové prostředí jsem zvolil program Eagle verze 4.16, jelikož jsem s ním měl již dřívější zkušenosti. Dále je nutné zjistit dostupnost knihoven součástek potřebných pro DPS. Byla nutná kontrola pouzder a výroba knihovny se součástkami,



Obrázek 4.4: Blokové schéma modulu přepínače

kteří nebyly součástí standardní knihovny Eaglu. Po této přípravě softwarového vybavení následoval samotný návrh jednotlivých modulů. Bylo třeba zvážit vhodnost použití vícevrstevné desky plošného spoje pro výrobu. Rozhodl jsem se použít pouze oboustrannou desku, a to z důvodu jednoduššího návrhu a také nižší ceny. Následoval samotný návrh jednotlivých modulů. Vycházel jsem z referenčních zapojení jednotlivých součástek a snažil se je co nejlépe zkombinovat s ostatními schématy. Velkou roli při konstrukci hrála velikost DPS. Snažil jsem se o co největší využití volného místa a tedy o co nejmenší velikost DPS. Dále bylo třeba umístit komunikační rozhraní vzájemně propojených integrovaných obvodů co nejlépe k sobě a také s přibližně stejnou délkou spoje párových vodičů.

Kompletní obvodová schémata, desky plošných spojů a seznam použitých součástek jsou v dokumentu uvedeny v příloze. Bylo vyrobeno a osazeno pět modulů. Hlavní modul koordinátoru, moduly se senzorem teploty a vlhkosti, senzorem intenzity světla a senzorem zachycující zrychlení. Posledním modulem je přepínací modul, který obsahuje SSR relé pro spínání 220V. Velikost těchto modulů je mezi 3.5 – 4.5cm x 4 – 4.5 cm. Celková cena je přibližně 15000 Kč.

Mikrokontroler

Základním prvkem každého modulu je mikroprocesor firmy Atmel. Pro řídicí modul koordinátoru jsem vybral procesor ATmega1281 [3] v pouzdře TQFP64. Hodinový takt tohoto mikroprocesoru je 8MHz. Hodiny jsou přivedeny jako externí z transceiveru. Napájení je přivedeno z obvodu FT232RQ pro komunikaci přes USB sběrnici. Přes transformační a stabilizační obvody je z původního napětí 5V USB sběrnice vytvořeno napájení 3.3V. Pro ostatní moduly jsem vybral mikroprocesor stejné řady, ale nižší třídy ATmega644. Hlavní rozdíl oproti předchozímu je, že tento procesor má menší vnitřní paměť, je osazen v menším

Režim MCU	Aktivní hodinová základna			Oscilátory		Zdroje probuzení						
	clk _{IO}	clk _{ADC}	clk _{ASY}	Hlavní hodiny povoleny	Časovač povolen	Externí přerušeni	Záznam TWI adresy	Časovač 2	SPM/EEPROM ready	ADC	Přerušeni WTD	Jiné
Nečinný	X	X	X	X	X	X	X	X	X	X	X	X
ADC režim		X	X	X	X	X	X	X	X	X	X	
Vypnutý						X	X				X	
Úsporný			X		X	X	X	X			X	
Pohotovostní				X		X	X				X	
Ext. pohotovostní			X	X	X	X	X	X			X	

Tabulka 4.1: Tabulka režimů spánku MCU

SM2	SM1	SM0	Režim MCU
0	0	0	Nečinný
0	0	1	ADC režim
0	1	0	Vypnutý
0	1	1	Úsporný
1	0	0	rezervováno
1	0	1	rezervováno
1	1	0	Pohotovostní
1	1	1	Ext. pohotovostní

Tabulka 4.2: Tabulka SLEEP instrukcí

pouzdrů TQFP44, a tudíž má méně I/O portů. Je také řízen externími hodinami z transceiveru o kmitočtu 8MHz. Tyto procesory jsou napájeny baterií s napětím 9V transformovaným a stabilizovaným na 3.3V. Realizace jednotlivých částí modulu a komunikačních rozhraní jsou popsány níže.

Power management mikrokontroleru a režimy spánku

Díky režimům spánku, které dovolují aplikaci vypnout nepoužívané části MCU, dochází k výrazné úspoře energie. Mikrokontrolery použité v této práci dovolují hned několik těchto režimů spánku. Přehled režimů a zdroje probuzení jsou v tabulce 4.1.

Pro vstup do některého z těchto režimů je nutné zapsat logickou jedničku na bit SE registru SMCR a musí se provést instrukce SLEEP. Zápis hodnoty na bity SM0, SM1 a SM2 v tomto registru určuje, o jakou instrukci se jedná. V tabulce 4.2 můžeme vidět nastavení jednotlivých bitů pro aktivaci různých režimů.

V případě výskytu přerušeni v režimu spánku se MCU probudí, obslouží rutinu přerušeni

a pokračuje instrukcí, která následuje za instrukcí SLEEP. Po probuzení ze spánku se obsah instrukčního registru a paměti SRAM nezmění. Jestliže dojde k resetu během režimu spánku, MCU se probudí a vykoná podle vektoru resetu.

Z hlediska tohoto projektu je nejdůležitějším režimem spánku mikrokontroleru režim Vypnutý – Power–Down. V tomto režimu je celý mikrokontroler prakticky vypnutý, jsou zapnuté pouze detekce externích přerušení, rozhraní TWI, a watchdog. MCU lze také probudit jen událostí na těchto rozhraních.

Vypnutí systému – režim Power–down

Do tohoto stavu se mikrokontroler dostane po přepnutí transceiveru do RX_ON_NOCLOCK stavu. Po zadání příkazu pro přepnutí transceiveru do uvedeného stavu má mikrokontroler 35 cyklů na to, aby sám sebe přepnul do vypnutého stavu, protože po 35ti taktech se přeruší hodinový signál na pinu CLKM, což jsou externí hodiny mikrokontroleru. Transceiver tedy naslouchá okolnímu prostředí, a jak zachytí rámeček určený pro něj, vyvolá přerušení RX_START pro mikrokontroler na pinu IRQ. Dále se na pinu CLKM objeví hodinový signál pro mikrokontroler, a ten je schopný obnovit svoji činnost a přijmout rámeček od transceiveru. Celý tento systém je řízen koordinátorem, který v určitých periodách zasílá pakety s požadavky na sensorová data, a tím probouzí postupně jednotlivé moduly. Ty zpracují požadavek, naměřená data odešlou zpět koordinátoru a celý modul opět uspí.

Transceiver

Rádiovou část každého modulu obstarává transceiver ATRF230 firmy Atmel [3]. Kombinace těchto dvou součástek je přímo uvedena a doporučena výrobcem. Tento integrovaný obvod se dodává v pouzdře QFN32. Zapojení transceiveru je popsáno v kapitolách o hodinách 4.2, anténě 4.2 a rozhraní SPI 4.2. Podle referenčního zapojení jsem ještě připojil ke vstupům AVDD a DVDD kondenzátory o kapacitě $1\mu\text{F}$.

Transceiver může pracovat ve dvou režimech, v základním operačním režimu a v rozšířeném operačním režimu. Základní operační režim poskytuje základní funkcionalitu jako vypínání a zapínání obvodu a příjem a vysílání rámečků.

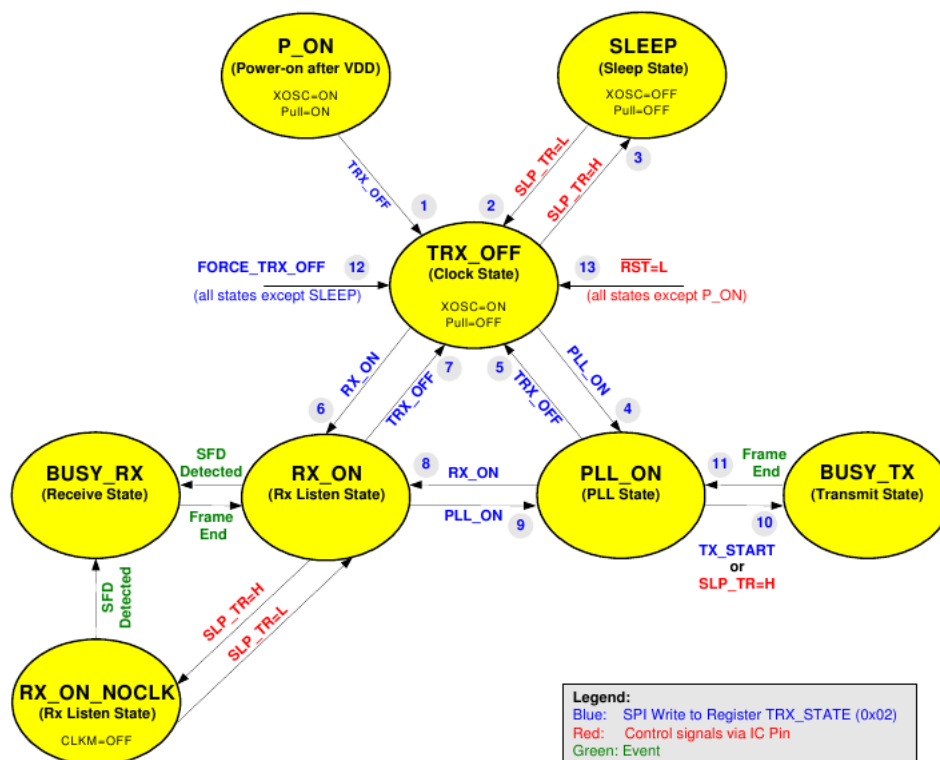
Stav transceiveru se řídí podle signály SLP_TR, RST a registrem TRX_STATE. Úspěšná změna stavu transceiveru je potvrzena v registru TRX_STATUS. Pomocí pinu SLP_TR a změny signálu náběžnou hranou se lze dostat do následujících stavů:

- TRX_OFF → SLEEP
- RX_ON → RX_ON_NOCLK
- PLL_ON → BUSY_TX

Pin RST způsobuje reset všech registrů a uvádí transceiver do stavu TRX_OFF. Zápisem hodnoty do registru TRX_STATE na pozici 4:0 se lze dostat do ostatních stavů základního režimu.

P_ON – zapnutí po připojení V_{DD}

Po připojení napětí se transceiver nachází v tomto stavu. Začne být aktivní oscilátor a hodiny na pinu CLKM slouží jako zdroj hodin mikrokontroleru. Dojde k vnitřnímu resetu



Obrázek 4.5: Stavový diagram základního operačního režimu

a obsah všech registrů se nastaví na jejich defaultní hodnoty. K opuštění tohoto stavu je nutné, aby mikrokontroler nastavil piny na standardní hodnoty SLP_TR = L a RST = H. Po ustálení napětí a ustanovení krystalu je možné se zapsáním do registru TRX_STATE hodnotami TRX_OFF nebo FORCE_TRX_OFF dostat ze stavu P_ON.

SLEEP – stav spánku

Ve stavu spánku, je celý transceiver vypnutý. Jestliže je povoleno CLKM, tedy transceiver je zdrojem hodin mikrokontroleru, stav spánku nastane po 35ti hodinových cyklech. Z důvodu uspání mikrokontroleru. Spotřeba je redukována pouze na svodový proud. Do tohoto stavu se lze dostat pouze ze stavu TRX_OFF nastavením pinu SLP_TR na hodnotu H. Nastavením SLP_TR na hodnotu L se transceiver dostane opět do stavu TRX_OFF.

TRX_OFF

V tomto stavu je povolené SPI rozhraní a oscilátor. Dále je povolen digitální napěťový regulátor(DVREG) a poskytuje 1.8V pro jádro, aby byl dostupný buffer rámce(Frame buffer). Mikrokontroler může přistupovat ke všem digitálním funkcím a na pinu CLKM je hodinový signál pro mikrokontroler. Piny SLP_TR a RST jsou povoleny pro řízení stavu.

PLL_ON

Vstupem do PLL_ON stavu z TRX_OFF stavu se nejprve povolí analogový napěťový regulátor(AVREG). Poté, co se ustaví napěťový regulátor, je povolený syntetizátor kmitočtů

PLL. Po nastavení frekvence PLL na přijímací frekvenci se provede zámknutí této frekvence, jejíž úspěch je indikován přerušáním PLL_LOCK. Jestliže přijde příkaz RX_ON v tomto stavu, je ihned povoleno přijímání. Stav PLL_ON odpovídá stavu TX_ON podle normy IEEE 802.15.4-2003.

RX_ON a BUSY_RX – naslouchání a příjem

Ve stavu RX_ON jsou povoleny všechny bloky pro příjem rámců a také syntetizátor kmitočtů PLL. Přejed z stavu TRX_OFF do stavu RX_ON se provádí zapsáním příkazu RX_ON do registru TRX_STATE. Režim příjmu se dělí na dva stavy: RX_ON a BUSY_RX. Rozdíl v těchto stavech je pouze ten, že ve stavu RX_ON běží pouze úvodní zpracování signálu, kdežto do stavu BUSY_RX se přejde detekcí začátku rámce(SFD).

RX_ON_NOCLOCK – stav naslouchání

Tento stav dovoluje naslouchání příchozích rámců, a v případě nečinnosti mikrokontroleru jej dokáže vypnout zastavením hodinového signálu na pinu CLKM. Detailněji je tento stav popsán v kapitole o SPI 4.2. V tomto stavu je spotřeba energie snížena na minimum.

BUSY_TX – stav vysílání

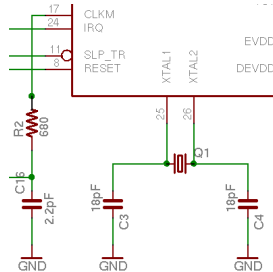
Vysílání je možné pouze začít pouze ve stavu PLL_ON. Existují dva způsoby, jak začít vysílat. První možností je s rostoucí hranou signálu SLP_TR, druhou možností je zápis příkazu TX_START do registru TRX_STATE. Během přechodu do tohoto stavu se nastaví syntetizátor kmitočtů na vysílací frekvenci. Vysílání prvních dat začíná po uplynutí 16 μ s, což je doba nastavení frekvence a zesilovače. Po ukončení vysílání se automaticky vypne zesilovač a vrací se do stavu PLL_ON.

Hodiny

Pro správnou funkci celého systému je třeba vybrat vhodný způsob generování hodinového kmitočtu. Každý modul obsahuje mikroprocesor a transceiver, tedy 2 obvody, které pro svou správnou funkci potřebují nějaký frekvenční zdroj. Dále je nutné tyto dvě součástky hodinově synchronizovat. Jako řešení jsem zvolil zapojení podle referenčního schématu transceiveru. Jedná se o připojení externího krystalu k vývodům XTAL1 a XTAL2 s příslušnými kondenzátory a propojení s mikroprocesorem přes signál CLKM. Dalšími řešeními by bylo pro každou součástku vlastní zdroj hodinového signálu nebo zapojení s interním oscilátorem. Tato řešení jsem však i po konzultaci s vedoucím zamítl.

Základním hodinovým taktem transceiveru je podle výrobce 16MHz. Transceiver je součástka velmi citlivá na přesnost, a proto se při výběru externího krystalu muselo dbát na nízké hodnotě frekvenční tolerance. Tato hodnota se udává v jednotkách ppm, která v tomto případě udává maximální počet chyb v hodinovém taktu na jeden milion taktů. V datasheetu výrobce se udává doporučená hodnota 40ppm. Dalším kritériem pro výběr byla vhodnost součástky do desky plošného spoje. Jelikož jsem se snažil o co nejmenší řešení, musel jsem také najít krystal malých rozměrů a také v SMD pouzdře. Krystal CS10 firmy Citizen [6] všechna tato kritéria splňuje. Kapacity připojených kondenzátorů 18pF doporučuje výrobce krystalu.

Hodiny pro mikroprocesor jsou tedy připojeny na pin XTAL1 externím hodinovým signálem CLKM s transceiveru. Dle referenčního schématu je nutné zapojit na tento signál filtr dolní propusti. Dle datasheetu výrobce jsou hodnoty kapacity kondenzátoru 2.2pF a hodnota odporu rezistoru 680Ω.



Obrázek 4.6: Schematické zapojení krystalu

Transceiver může pracovat ve 3 různých módech krystalu. Pro správnou funkci s externím krystalem je nutné nastavit pole XTAL_MODE registru XOSC_CTRL na hodnotu 100b. Hodinový takt pro mikroprocesor jsem zvolil 8MHz. Na tuto hodnotu je potřeba nastavit hodinový výstup transceiveru. Děje se tak nastavením pole CLKM_CTRL registru TRX_CTRL_0 na hodnotu 100b. Zdroj hodinového signálu pro mikrokontroler je třeba také správně nakonfigurovat nastavením pojistek v paměti flash. Celkem je k dispozici 6 různých režimů. Pro externí hodiny musí být nastavena hodnota CKSEL na 0000(external clock). Základní nastavení dodávané výrobcem je interní RC oscilátor o taktu 8MHz, který je však dělen na výsledný takt 1MHz.

Anténa

Standard ZigBee pracuje v nelicencovaných pásmech 2.4GHz, 915MHz a 868MHz. Rozdílné vlastnosti těchto pásem byly popsány v kapitole 3.1. Pro účel tohoto projektu bylo tedy nutné vybrat pásmo, ve kterém bude probíhat komunikace a tedy také příslušnou anténu. Jako nejdostupnější a z hlediska vlastností také nejvhodnější se jeví pásmo 2.4GHz. V tomto pásmu se pro přenos signálu využívá modulace QPSK a přenáší se prostřednictvím DSSS. Pro přístup k fyzickému médiu je použita metoda CSMA/CA. Přenosová rychlost v tomto pásmu je 250kbit/s. Je možné využít 16 kanálů o šířce pásma 5MHz.

Na trhu existuje několik druhů antén pro standard ZigBee. Většina projektů zabývajících se ZigBee využívá antén buďto leptaných přímo na desce plošného spoje nebo antén s SMA konektorem a externí anténou. U leptaných antén je nevýhoda ta, že je třeba znalostí technologie výroby této antény a dále speciální programy na otestování charakteristik antény. Dále tyto antény zabírají hodně místa na plošném spoji. SMA konektor by zvětšil velikost plošného spoje a anténa by zbytečně přečnickovala.

Posledním možným typem antény používané v těchto aplikacích jsou SMD antény. Tyto antény jsou velmi malých rozměrů a mají dostatečný dosah příjmu a vysílání signálu. Zvolil jsem tedy toto řešení. Anténa firmy Yageo [25] je v SMD pouzdře o velikosti 3.2mm x 1.6mm. V datasheetu výrobce je v referenčním schématu zapájená na ploše 18mm x 10mm bez mědi. Dále jsou její vlastnosti testovány při délce spoje od transceiveru k anténě o délce 30mm a impedanci 50Ω. Snažil jsem se tedy toto referenční schéma dodržet. Bohužel nemám

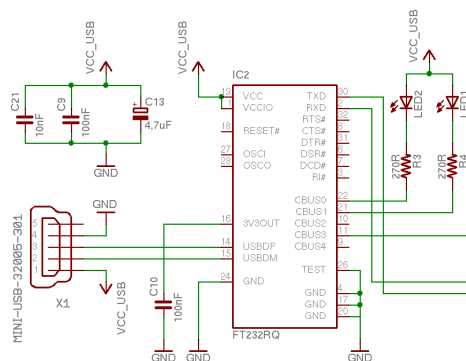
možnost otestovat, zda uvedené zapojení odpovídá charakteristikám uvedených v datasheetu výrobce.

Pro správnou funkčnost antény bylo třeba předřadit připojení k transceiveru balun. V tomto případě bylo třeba transformovat výstup z antény 50Ω na 100Ω na vstupu transceiveru. Výstupy z balunu jsou ještě připojené přes dva kondenzátory o kapacitě 22pF .

Rozhraní USB

Rozhraní USB pro komunikaci koordinátoru se stolním počítačem je navrženo pomocí čipu FT232RQ firmy FTDI [8]. Tento čip dokáže převést data ze sériového portu mikrokontroleru na data přenášená přes USB sběrnici do osobního počítače. Toto řešení jsem zvolil pouze u modulu koordinátoru, jehož funkcí je sbírat data od ostatních zařízení a předávat je k dalšímu zpracování. Tento modul měl být také napájen ze sběrnice USB. Tyto funkce však nejsou u modulu koordinátoru realizovány. Je to z důvodu nemožnosti zapájet součástku do desky plošného spoje.

Datovým vstupem tohoto obvodu jsou piny TX0 a RX0 z mikroprocesoru, které se nachází na portu E. Do obvodu jsou přivedeny inverzně, tedy TX0 na RXD a RX0 na TXD. Jedná se pouze o komunikaci sériovým rozhraním USART mezi mikroprocesorem a obvodem FT232, ten data po příjmu kovertuje na formát sběrnice USB a odešle. Jako indikaci komunikace s mikroprocesorem jsou na desce plošného spoje umístěny dvě zelené LED diody. Jsou připojeny k pinům CBUS0 pro odesílání dat a CBUS1 pro příjem dat. Jejich zapojení se řídí nastavením konfigurace EEPROM TXLED# a RXLED#. Ke konfiguraci lze použít program MPROG, který lze stáhnout ze stránek výrobce obvodu.



Obrázek 4.7: Schematické zapojení obvodu FT232

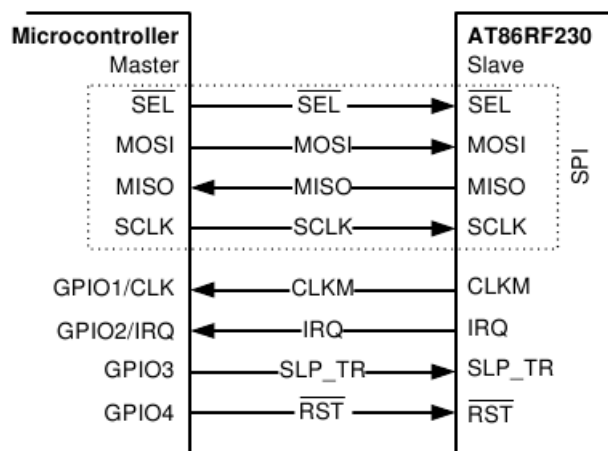
Při pájení nastal problém špatně vyrobené patice pro zapájení. Tento čip se vyrábí pouze v pouzdru QFN32. Podle označení a datasheetu výrobce jsou vývody tohoto pouzdra shodné s pouzdrům transceiveru. Při návrhu plošného spoje jsem tedy použil stejný vzor pro výrobu. Problém je však, že toto pouzdro je o něco větší než pouzdro transceiveru, pouzdro překrývá vývody a není ho možné ručně zapájet. Tuto skutečnost jsem při návrhu nevěděl, nebylo ani možné zkusit přeměřit velikost pouzdra a vývodů. Jediné možné řešení by bylo zkusit vyrobít masku pro automatické osazení, což by však trvalo delší dobu a navíc je výroba této masky značně nákladná v porovnání s cenou jednoho kusu výrobku.

Jako náhradní řešení jsem zvolil přenos dat z modulu přes druhé sériové rozhraní mikrokontroleru, které se nachází na portu D, který je vyvedený na pin header modulu. Napájení

celého modulu však není závislé na nezapájeném čipu, proto lze modul dát do provozu připojením ke sběrnici USB. Základní napětí sběrnice 5V je transformováno a stabilizováno na hodnotu 3.3V. Pro připojení modulu ke sběrnici jsem použil mini USB konektor typu B.

Rozhraní SPI

Rozhraní SPI [3] slouží ke komunikaci transceiveru a mikroprocesoru. Mikroprocesor může přes toto rozhraní přistupovat do frame bufferu a registrů. Základními signály SPI jsou SEL, MOSI, MISO a SCLK. Dále je transceiver s mikrokontrolerem spojen signálem CLKM, který slouží jako externí hodiny pro mikroprocesor, signál IRQ sloužící jako zdroj přerušení a signály SLP_TR a RST. V případě použití signálu CLKM pracuje rozhraní SPI v synchronním režimu. Frekvence, na které se v tomto režimu pracuje je 8MHz.



Obrázek 4.8: Referenční schéma zapojení SPI rozhraní transceiveru a mikrokontroleru

SPI je bajtově orientovaný protokol, komunikace je vždy obousměrná a používá se model master – slave. V tomto případě je master mikrokontroler a slave transceiver. K povolení komunikace je nutné nastavit signál SEL do logické nuly. Poté může začít master s přenosem. Generuje osm SPI hodinových cyklů pro přenos jednoho bajtu do transceiveru přes signál MOSI a v tom samém čase vysílá slave jeden bajt masterovi přes signál MISO. Každá přenosová sekvence začíná příkazovým bajtem ze strany mastera skrz signál MOSI. Tento příkazový bajt definuje přístupový režim a další informace závislé na režimu. Během přenosu příkazového bajtu transceiver vysílá zpět samé nuly.

Čtení resp. zápis z resp. do registru jsou dvoubajtové operace. První bajt je příkazový bajt s adresou registru a v druhém bajtu následují přečtená resp. zapisovaná data.

Operace příjmu/vysílání rámce slouží pro přenos dat z/do mikrokontroleru. Každý přenos začíná příkazovým bajtem, následuje délka rámce a samotný rámec s uživatelskými daty. Po skončení přenosu dat se ještě připojí LQI(Link quality indication) bajt, který nese informace o kvalitě linky a doručení paketu. Maximální délka dat jednoho paketu je 127 bajtů.

Signál SLP_TR – sleep/wake-up a transmit signal je multifunkční pin. Jeho funkce je závislá na stávajícím stavu transceiveru. Jestliže je použit jako signál pro usnutí, uvede

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Mód
1	0	Adresa registru [5:0]						Čtení z registru
1	1	Adresa registru [5:0]						Zápis do registru
0	0	1	Rezervováno					Příjem rámce
0	1	1	Rezervováno					Vysílání rámce
0	0	0	Rezervováno					Čtení z paměti
0	1	0	Rezervováno					Zápis do paměti

Tabulka 4.3: Tabulka SPI příkazového bajtu

transceiver do stavu TRX_OFF a povolí se hlavní hodiny. Když je použit jako signál pro začátek přenosu, s rostoucí hranou začíná přenos. Existují dva možné scénáře vypnutí podporované transceiverem s ohledem na SLP_TR pin.

- Mikrokontroler i transceiver je vypnutý.
- Transceiver čeká na příchozí rámec a mikrokontroler je vypnutý

U prvního modelu mikrokontroler uvede transceiver do stavu spánku nastavením SLP_TR na hodnotu H, když je transceiver ve stavu TRX_OFF. Hlavní hodiny pro mikrokontroler na pinu CLKM se vypnou po 35ti cyklech. Toto dovoluje mikrokontroleru dokončit vypnutí a zamezení případnému deadlocku. Transceiver je probuzen, když mikrokontroler uvolní pin SLP_TR. Tento koncept dovoluje nejnižší možný způsob spotřeby energie.

V druhém případě, jestliže se očekává příchod rámce, ale žádná jiná aplikace není v mikrokontroleru spuštěna, může být mikrokontroler vypnut bez ztráty příchozího rámce. Transceiver je ve stavu RX_ON, CLKM pin je vypnutý po 35ti cyklech po nastavení pinu SLP_TR na hodnotu H. Stav transceiveru se změní na stav RX_ON_NOCLOCK. Začátek příjmu rámce je indikován přerušáním RX_START a hodiny na pinu CLKM jsou opět spuštěny, tím pádem je obnovena funkce mikrokontroleru a ten přijme rámec.

Rozhraní JTAG

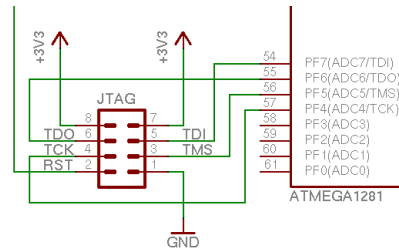
AVR IEEE 1149.1 JTAG [19] rozhraní může být použito pro následující úlohy:

- Testování obvodu pomocí JTAG boundary scan.
- Programování paměti flash(EEPROM), pojistek(fuses) a uzamčených bitů(lock bits)
- Nalézání chyb(debugging). Přístup ke všem vnitřním perifériím, k vnitřní i vnější RAM, k registrům, programovému čítači a paměti EEPROM

Rozhraní JTAG je přístupné přes tzv. Test Access Port(TAP) controller. Je to vlastně stavový automat ovládaný signály TMS a TCK. Test Mode Select(TMS) je signál, pomocí kterého je prováděna navigace ve stavovém automatu. Test Clock(TCK) udává synchronizaci s JTAG operací, tedy, zda se má načítat TMS se sestupnou nebo vzestupnou hranou. Navigací ve stavovém automatu lze pracovat buď s instrukčním registrem nebo jedním z datových registrů. Jako vstupy/výstupy do/z registrů slouží signály TDI a TDO. Nepovinným signálem JTAG rozhraní je RESET, který je však nutný pro reset obvodu při programování paměti. Pro správnou funkci rozhraní JTAG je nutné mít naprogramovanou

pojistku JTAGEN. Mikrokontroler je vyráběn s touto pojistkou naprogramovanou, tzn. bit JTAGEN = 1. Jestliže tato pojistka není naprogramovaná, jsou tyto piny normálními piny portu.

Volba tohoto programovacího rozhraní byla jediná možná. Pro programování mikrokontroleru bylo možné zvolit ještě rozhraní SPI, jenže toto rozhraní je použito pro komunikaci s transceiverem.



Obrázek 4.9: Zapojení pin headeru s rozhraním JTAG k ATMEGA1281

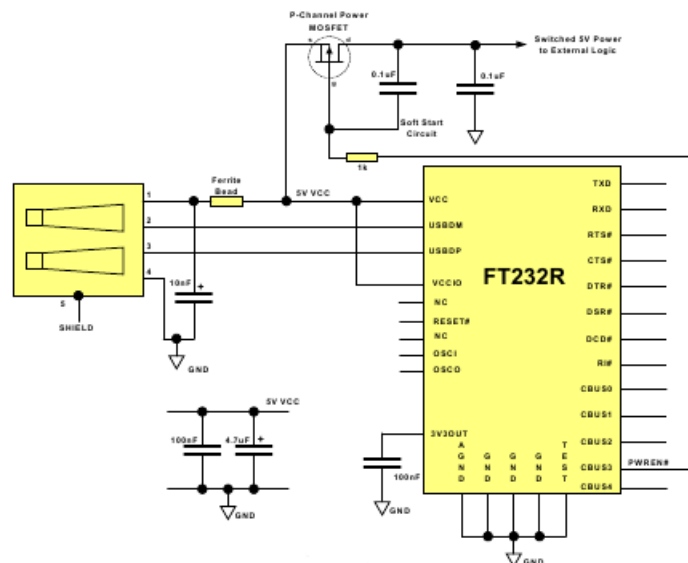
Každý modul je vybaven rozhraním JTAG, které je umístěné na 8mi pinovém pin headeru. Je to z důvodu čtyřech signálů samotného rozhraní, dále je nutný pro programování pin RESET a také je důležité mít vyvedené napětí a zem, které se používá buďto jako referenční napětí pro programátor nebo jako napájení programátoru. Na každém modulu je zapojení pin headerů jiné, a to z důvodu jiného rozmístění pin headeru a ostatních součástek na desce. Správné zapojení konektoru každého modulu lze najít v příloze, kde jsou uvedena schémata. Z důvodu nedostatku místa není toto označení uvedeno přímo na desce plošného spoje.

Napájení

Napájení modulů je jednou z klíčových otázek u všech modulů. Bylo třeba rozdělit způsob napájení na tři části. Napájení hlavního modulu koordinátoru, sensorových modulů a modulu přepínacího. Protože každá skupina modulů má, už ze své podstaty, jiný způsob napájení, bylo nutné zajistit stejné vstupní napětí mikroprocesoru, protože je na něm závislý takt procesoru, a tedy rychlost vykonávání instrukcí.

U hlavního modulu koordinátoru, jehož funkcí je předávat data přes USB sběrnici do počítače, jsem situaci vyřešil podle ukázkového zapojení v datasheetu k čipu FT232, které můžeme vidět na obrázku 4.10. Z USB sběrnice se odebírá napětí 5V, které je přivedeno na tranzistor, který je spínáný z výstupního pinu CBUS3 čipu FT232. U vstupu tranzistoru je zapojen polarizovaný kondenzátor $4.7\mu\text{F}$ a u vstupu do čipu je zapojen blokující kondenzátor 100nF . Dále je toto napětí transformováno a stabilizováno obvodem LP3990 na hodnotu 3.3V . V doporučeném zapojení jsou u vstupů tohoto obvodu zapojeny kondenzátory o kapacitě 100nF , které blokují před zákmity. Výstupní napětí je přivedeno na dva vstupy mikrokontroleru VCC a dále na vstup AVCC. K těmto vstupům jsou také připojeny blokující kondenzátory 100nF . Transceiver je napájen dvěma vstupy EVDD a DEVDD s blokujícími kondenzátory hodnoty $1\mu\text{F}$.

Funkce přepínacího modulu je spínat napětí 220V , tedy se předpokládá stálý zdroj elektrické energie. Jednou z variant napájení proto bylo vzít toto napětí, transformovat a stabilizovat na určených 3.3V . Toto jsem však neshledal jako vhodné řešení z důvodu



Obrázek 4.10: Referenční schéma zapojení napájení ze sběrnice

zvětšení plochy plošného spoje a relativně složitějšího zapojení. Dále mi přišlo zbytečné navrhovat zapojení a vytvářet samotný plošný spoj již hotového a vyzkoušeného řešení. Zvolil jsem proto napájení tohoto modulu z externího transformátoru zapojeného do zásuvky. V dnešní době se dají tyto adaptéry pořídit za velmi rozumné peníze, jsou odzkoušené a stabilní. Navíc může být zdroj schovaný a poměrně vzdálený od samotného modulu. Bylo tedy třeba pouze vhodného konektoru pro připojení.

Napájení pro sensorové moduly je z bateriového článku 9V. Tuto variantu jsem zvolil z důvodu jednoduchosti řešení a dostupnosti baterií. Napětí 9V je přivedeno přes klasický plochý bateriový konektor na obvod TPS76333, na jehož výstupu je napětí 3.3V. Tímto napětím jsou napájeny jak mikrokontroler s transceiverem, tak i obvody senzorů.

Většina součástek souvisejících s napájením celého modulu je z důvodu úspory místa na spodní straně plošného spoje. V případě nemožnosti napájení ze sběrnice USB a také kvůli programování je ještě signál pro napájení a zem přiveden na piny konektoru pro programování JTAG. Pro indikaci napájení a funkčnosti systému je umístěna na každém modulu jedna červená LED dioda.

Solární napájení

Zvláštní kapitolou napájení sensorových modulů je napájení solární energií. Myšlenka toho druhu energie byla v počátku tohoto projektu, avšak její realizace se nedostala ani k obvodovému návrhu. U tohoto typu napájení je totiž nutné vyřešit otázku člověkem neovlivnitelných změn v dodávce sluneční energie. Řešením by bylo například při poklesu dodávky proudu ze solárních panelů zapnout alternativní zdroj, tedy baterii. Dalším řešením je pouze solární energií nabíjet baterii. Bylo by možné použít například obvodu DS2714 firmy Dallas semiconductor distribuovaný firmou Maxim a malý solární panel, který je schopný dodávat napětí 6V a proud 50mA. Solární mini panely nabízí na svých stránkách firma Sundance Solar. Uvedené součástky by měly stačit pro funkci obvodu nabíjení. Z časových důvodů jsem však tento návrh nerealizoval. Dále by toto řešení bylo poměrně nákladné. Kromě

těchto dvou součástí, cena přibližně 80\$, by bylo nutné ještě objednat kvalitní baterii, která by zvládla přerušované nabíjení a s dostatečnou kapacitou. Cena takovýchto baterií se pohybuje kolem 20£.

Kapitola 5

Programové vybavení

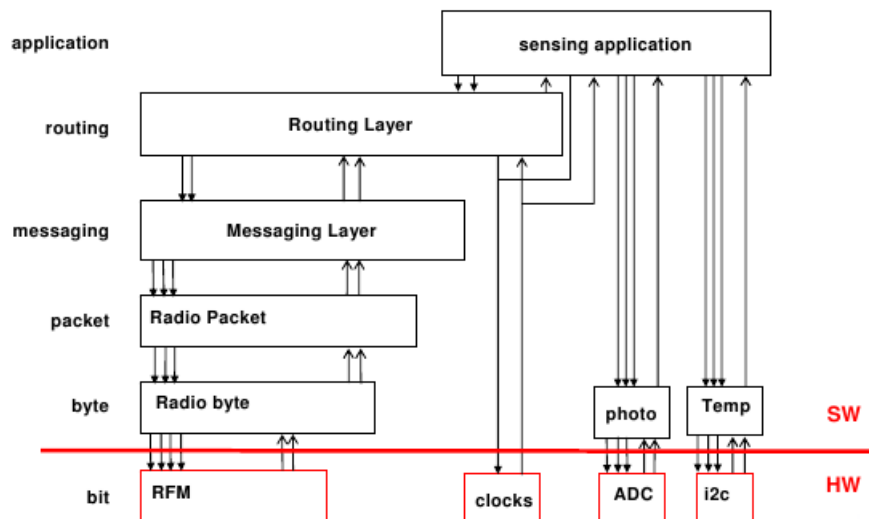
5.1 TinyOS

Jak bylo řečeno na začátku této kapitoly celý systém bude zpracovávat operační systém TinyOS [4]. Je to volně dostupný operační systém s otevřeným kódem a složený z jednotlivých komponent. Jeho platforma je typicky zaměřená na bezdrátové senzorové sítě. TinyOS je vestavěný operační systémem napsaný v jazyce NesC jako množina spolupracujících úloh a procesů. Předpokládá se, že bude začleněn do hypotetické sítě, která se bude skládat z malých bezdrátových mikroelektromechanických senzorových systémů, robotů nebo podobných zařízení. Tento systém je známý pod pojmem Smartdust [14]. Projekt TinyOS začal spoluprací mezi University of Carolina, Berkeley a Intel Research a stal se mezinárodním konzorciem, TinyOS Alliance.

TinyOS není operačním systémem v pravém slova smyslu. Je to pouze kostra určená k programování(programming framework) vestavěných systémů a soubor komponent, které umožňují sestavení specifického operačního systému pro danou aplikaci. Důvod pro nasazení tohoto operačního systému je zajistit, aby kód aplikace měl co nejmenší velikost. Navíc TinyOS je navržen tak, že nemá žádný souborový systém, podporuje pouze statickou alokaci paměti, implementuje jednoduchý model úloh a poskytuje minimální model abstrakce na úrovni zařízení a sítě.

Programovací model TinyOS je složen z několika částí – komponent. Tak jako u jiných operačních systémů jsou tyto komponenty děleny do vrstevového modelu. Nejnižší vrstva je nejbližší k hardware, vyšší vrstvy jsou blíže k aplikaci. Kompletní TinyOS aplikace si můžeme představit jako diagram komponent, jehož každá část pracuje nezávisle.

Jednotlivé komponenty můžeme rozdělit do tří výpočetních modelů: příkazy, události a úlohy. Příkazy a události jsou mechanismem pro komunikaci mezi jednotlivými komponentami, zatímco úlohy se používají pro komunikaci uvnitř jedné komponenty. Příkazem typicky žádáme komponentu o poskytnutí nějaké služby. Typickým příkladem je žádost nastartování čtení údaje ze senzoru. V porovnání s událostí, kterou signalizujeme ukončení této služby. Události mohou pracovat také asynchronně, v případě hardwarového přerušování nebo příchodu zprávy. Příkazy a události mohou vytvořit úlohu, což je funkce, kterou plánuje plánovač(scheduler) a operační systém ji vykoná v pozdějším čase. To dovoluje vykonávání příkazů a událostí okamžitě a přenechání objemnějších výpočtů úlohám. Úlohy reprezentují vnitřní souběžnost uvnitř komponenty a mají přístup pouze k informacím uvnitř komponenty. Vykonávání úloh je nepreemptivní a běží v pořadí FIFO. Tento jednoduchý model



Obrázek 5.1: Ukázka vrstevového modelu TinyOS [11]

typicky postačuje pro aplikace vstupně/výstupních operací, ale je velmi složité řídit aplikace, kde v operačním systému spolupracuje několik rozdílných vláken.

Knihovna komponent TinyOS obsahuje komunikační protokoly SPI, I2C, síťové protokoly (ZigBee), ovladače senzorů, distribuované služby a nástroje pro sběr dat. Všechny tyto části mohou být použity tak, jak jsou nebo mohou být dále upraveny pro jednotlivé aplikace. Událostmi řízený model vykonávání v TinyOS umožňuje jemnou správu napájení a poskytuje flexibilitu v plánování, která je nezbytná pro nepředvídatelnost v bezdrátových sítích a rozhraních pro reálný život. TinyOS je portován na více než deset platform a nespočetně senzorových desek. Široká komunita jej používá při simulacích ve vývoji a testuje různé algoritmy a protokoly.

Aplikace pro TinyOS [20] jsou napsané v jazyce NesC, což je odnož jazyka C optimalizovaná pro paměťová omezení v senzorových sítích. Doplňující nástroje jsou většinou ve formě skriptů v shellu nebo v Javě. Přidružené knihovny a nástroje, jako NesC překladač a AVR binutil toolchains jsou napsány v jazyce C. TinyOS je kompletně neblokující a má jediný zásobník. Tedy, všechny vstupně/výstupní operace, které trvají více jak několik mikrosekund jsou asynchronní a jsou odvolány. Aby překladač lépe optimalizoval hraniční volání, TinyOS používá vlastnost NesC, která spojuje tato volání, zvané události, staticky. Z důvodu neblokujícího systému je TinyOS s jedním zásobníkem velmi výkonný, ale to také tlačí programátory psát velmi komplexní logiku spojenou z mnoha nepatrných událostí.

Kód TinyOS je staticky linkován s kódem programu a kompilován do malého binárního souboru za použití GNU sady nástrojů potřebných pro překlad.

5.2 Programování modulů

K programování všech modulů slouží rozhraní JTAG. Každý modul má vyvedeny 4 piny tohoto rozhraní na pin header, dále je ještě na tento pin header vyveden pin s napájením, zemí a pin RESET. Zapojení je u každého modulu trochu jiné, proto je nutné zapojit rozhraní programátoru podle schmat, které jsou uvedena v příloze. Mikroprocesory ATmega

jsem vybral mimo jiné i proto, že je obecně známo jejich bezproblémové použití a snadné programování. S použitím jednotlivých programátorů a se samotným programováním jsem však měl velké problémy, které popíši v následujících podkapitolách.

Do termínu odevzdání práce jsem však stihl otestovat jednotlivé části HW vybavení modulů se zjištěním, že závada není v hardware. Dokázal jsem se se všemi procesory spojit a komunikovat vysíláním jednotlivých bitů na rozhraní JTAG a také získat data ze sériového rozhraní modulu koordinátoru.

Programátor PRESTO

Jednou z možností jak programovat procesory firmy Atmel přes rozhraní JTAG bylo použití programátoru a programového vybavení PRESTO [2] firmy ASIX. Tento programátor je připojen k osobnímu počítači rozhraním USB. Je nutné pouze nainstalovat ovladač obvodu FTDI. Po spuštění aplikace pro programování – UP je třeba vybrat typ procesoru, nastavit zdroj napájení programátoru a frekvenci krystalu pro rozhraní JTAG. Frekvence musí být nižší než frekvence krystalu mikrokontroleru. Je třeba nastavit f_j 750kHz, protože počáteční frekvence mikrokontroleru je dle datasheetu 1MHz. Poté je programátor připraven k připojení k MCU. Připojení se však nezdaří. Program končí čtením zprávou, že se nepodařilo navázat komunikaci s programovanou součástí.

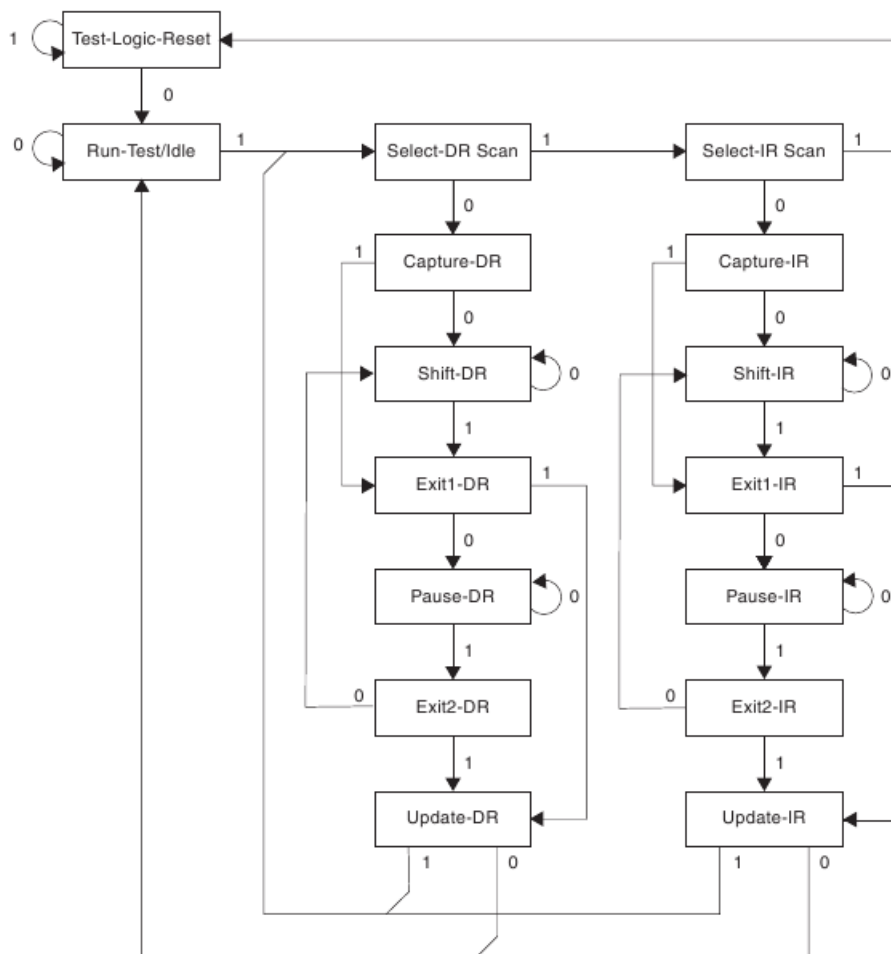
Program je odzkoušený na více počítačích i více operačních systémech a zkoušel jsem všech pět modulů, tedy dva typy mikrokontrolerů. Vždy však končí stejnou chybou. Testoval jsem správnost zapájení i zapojení. Dle datasheetu je vše v pořádku. Firma ASIX poskytuje také nástroj pro kontrolu programátoru a testování jednotlivých pinů rozhraní JTAG. Po spuštění tohoto programu se objeví logické hodnoty, které jsou na jednotlivých pinech rozhraní. Pomocí tohoto programu jsem zjistil, že na výstupech z mikrokontroleru jsou logické hodnoty, které by tam měly být. Tento software také umožňuje měnit hodnoty vstupních pinů rozhraní, tím lze docílit komunikace s mikrokontrolerem, není však možné do něj nějaký kód zapsat.

Programátor Xilinx

Jiným řešením bylo použití programátoru a vývojového prostředí Xilinx [24]. Tento programátor také komunikuje s počítačem přes rozhraní USB. Prostředí aplikace Impact balíku WebPack je primárně určeno pro vývoj na platformách Xilinx, avšak na stránkách atmelu jsou konfigurační soubory, se kterými lze v této aplikaci pracovat. Při detekci se zobrazí nově připojené zařízení k programátoru. Při inicializaci nově nadetekovaného zařízení a čtení konfiguračního souboru se však aplikace zhroutlá s chybou v grafickém rozhraní aplikace. Na stránkách podpory ani na Internetu se podobný problém nikomu neobjevil. Nelze tedy inicializovat mikroprocesor a tudíž nejdou provádět žádné operace. Jediné co funguje, stejně jako u předchozího programátoru, je zápis a čtení na jednotlivých pinech rozhraní JTAG. Tedy lze získat informace o stavu procesoru, výrobní číslo, číslo verze, atd... K této komunikaci je přímo v prostředí Impact stavový automat, pomocí kterého lze procházet paměť a provádět čtení a zápis jednotlivých bitů do registrů. Pro konfiguraci nebo zápis programového kódu je toto však absolutně nevhodné.

Boundary scan

Jediná operace, která se dá provádět je tedy procházení stavového automatu TAP (Test Access Port) Controller – obr. 5.2. Je to 16-ti stavový automat, pomocí kterého lze vykonávat operace testování obvodu, programování nebo ladění systému. Přechod z jednoho stavu do jiného závisí na hodnotě na pinu TMS a zapisí se s rostoucí hranou signálu TCK. Vstupy/výstup do/z registrů jsou na pinech TDI a TDO. Pomocí tohoto automatu jsem ověřil správnou funkčnost a reakci mikrokontroleru. Například výběrem čtení datového registru po počáteční inicializaci na výstupu TDO můžeme přečíst hodnoty 32-bitového registru, ve kterém je uloženo ID výrobce, výrobní číslo a číslo verze mikrokontroleru.



Obrázek 5.2: Stavový automat pro testování obvodu

Programatory Atmel a AVR studio

Jiným možným řešením problému komunikace modulů s programátory by bylo použití originálních programátorů a software přímo od firmy Atmel. Tyto programátory však škola nevlastní a ani je nebylo možné jiným způsobem získat. Problém s programátory by totiž mohl být způsobem použitím nových verzí procesorů s úsporným režimem. Použité procesory jsou označené písmenem "V" na konci číselného označení. Označení těchto procesorů

sorů není nikde v konfiguraci testovaných vývojových prostředků. Nikde jsem si však tuto domněnku nepotvrdil.

Jtag svf player

Po vyzkoušení nefunkčnosti programátorů ve vývojových prostředích jsem narazil na aplikaci Jtag svf player, kterou poskytuje firma Asix k programátoru PRESTO. Jedná se o aplikaci, která dokáže přes rozhraní programátoru nahrát svf soubor s kódem do mikrokontroleru. Svf soubor je v podstatě soubor obsahující čtecí operace a přechody mezi jednotlivými stavy automatu TAP popsaného v kapitole 5.2 na obrázku 5.2. Převod do tohoto formátu je možný z výsledného hex souboru přeloženého kódu. Firma Atmel přímo poskytuje tento nástroj – avrsvf.

Jediným řešením programování modulů, které jsem našel, je napsání kódu ve vývojovém prostředí, např. AVR studio, přeložení a vytvoření hex souboru, převod do svf formátu a nahrání do mikrokontroleru. Takovéto programování je však velmi neproduktivní, protože není možné vidět přímo obsahy pamětí, konfigurace a jiné vlastnosti mikrokontroleru, jako by to bylo v prostředí aplikace UP nebo IMPACT.

5.3 Přenosový model koordinátor – PC

Přenos dat naměřených sensorovými moduly obstarává modul koordinátoru. K této komunikaci měl původně sloužit obvod FT232, který měl být připojený ke sběrnici USB osobního počítače. Problém s výrobou popsaný v kapitole 4.2 však toto nedovoluje. Jediná možnost je využít druhého rozhraní USART, které se nachází na portu D a je vyvedeno na pin header modulu.

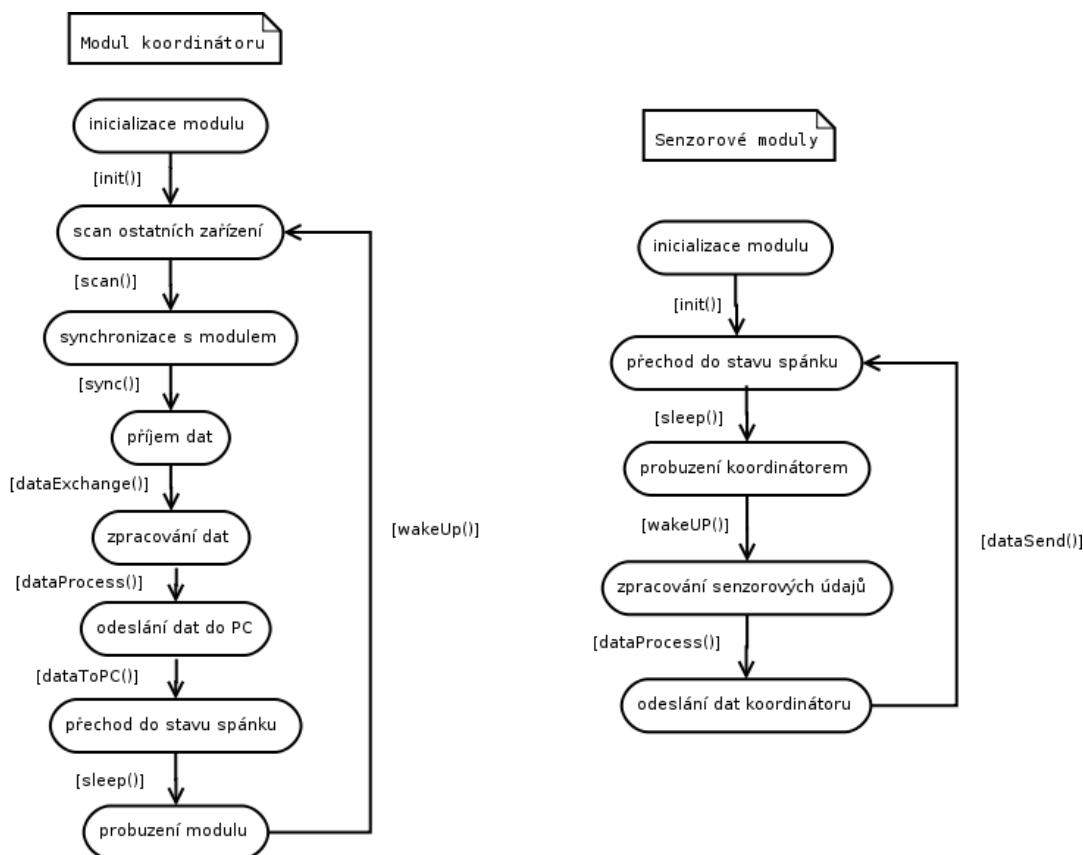
Modul koordinátoru tedy po příjmu dat od sensorového modulu provede odeslání dat přes sériovou linku na terminál počítače. Nejprve je třeba provést inicializaci rozhraní USART nastavením rychlosti přenosu (např. 9600 baudů), nastavením formátu rámce a povolením vysílání. Všechna tato nastavení se provádí zápisem do registru UBRR. Po nastavení může dojít k samotnému vysílání. Nejprve se musí počkat na uvolnění bufferu, což je signalizováno nastavením bitu UDRE1 registru UCSR1A. Po uvolnění bufferu stačí jenom zapsat data do registru UDR1 a data jsou poslána na terminál počítače. Posílá se celý řetězec s identifikací senzoru a naměřenou hodnotou. Tedy např.

```
temperature 22
humidity 76
:
```

Kvůli problémům uvedeným v předchozích kapitolách jsem však neměl možnost tuto funkci realizovat.

5.4 Návrh SW vybavení

Veškerá komunikace v síti jde přes modul koordinátoru. Ten si vyžádá data od sensorových modulů, v případě potřeby zašle informaci pro modul se spínačem a posílá data do osobního počítače. Tímto je řízena spotřeba sensorových modulů, které jsou ve stavu spánku a probudí se pouze při detekci rámce pro ně určené. Poté vykonají zpracování sensorového údaje a opět přejdou do stavu spánku. Schematické zobrazení funkce systému je na obrázku 5.3.



Obrázek 5.3: Schematické znázornění funkce systému

Tento model v době odevzdání textové části práce není vytvořený z důvodu zdlouhavého hledání způsobu komunikace s moduly přes různé druhy programátorů a aplikací, které jsem popsal v kapitole 5.2.

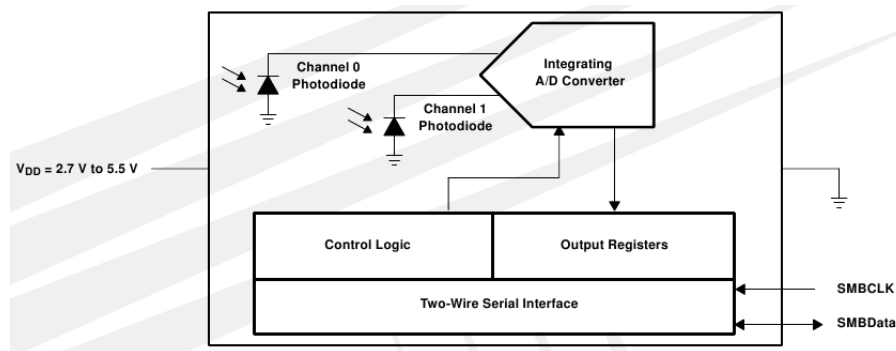
5.5 Zpracování senzorových dat

Senzor světla

Obvod TSL2550 [17] snímá intenzitu okolního světla a převádí ji na digitální signál na výstupu obvodu přes dvou-drátové(two-wire) rozhraní SMBus. Senzor je kombinací dvou fotodiód a analogově-digitálního převodníku. Tento obvod dokáže změřit intenzitu světla a vyjádřit ji v digitální podobě hodnotou s citlivostí podobnou lidskému oku.

Senzor je schopný snímat široké spektrum vlnových délek. První z fotodiód je citlivá na viditelné a infračervené světlo, zatímco druhá fotodióda je citlivá pouze na infračervené světlo. Dvě diody jsou zde z důvodu kompenzace infračervené složky okolního světla. Digitální výstup z tohoto senzoru je přibližná k lidskému vnímání osvětlení, které se udává v jednotkách Lux.

Komunikace s obvodem probíhá přes příkazový registr, kdy je k dispozici šest příkazů pro vypnutí/zapnutí obvodu, výběr režimu zpracování a pro čtení naměřených dat. Naměřená data jsou uložena v jednom ze dvou ADC registrů. Obsah tohoto registru je na obrázku



Obrázek 5.4: Vnitřní blokové schéma senzoru světla

5.5. Skládá se z jednoho Valid bitu, který určuje platnost výsledku v registru. A dále pak z CHORD bitů a STEP bitů. Výsledek převodu je poté určen vztahem

$$ADC = 16.5 * (2^C - 1) + (S * 2^C)$$

kde C je CHORD číslo(0..7) a S je STEP číslo(0..15).

VALID	CHORD BITS			STEP BITS			
B7	B6	B5	B4	B3	B2	B1	B0
VALID	C2	C1	C0	S3	S2	S1	S0

Obrázek 5.5: Formát ADC registru

Konečnou hodnotu intenzity v Luxech je potřeba spočítat z hodnot obou dvou fotodiód, kvůli kompenzaci infračervené složky.

$$Svetlo[Lux] = ADC_{D1} * 0.46 * e^{-3.13R}$$

kde

$$R = ADC_{D0}/ADC_{D1}$$

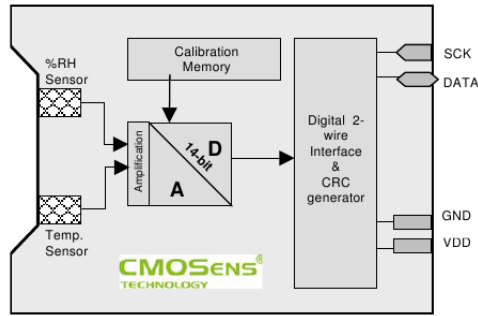
Senzor teploty a vlhkosti

Tento senzor [16] dokáže měřit okolní teplotu a relativní vlhkost. Tyto veličiny jsou změřeny a zpracovány analogově digitálním převodníkem. Obvod komunikuje přes dvou-drátové rozhraní SMBus, stejně jako senzor pro měření osvětlení.

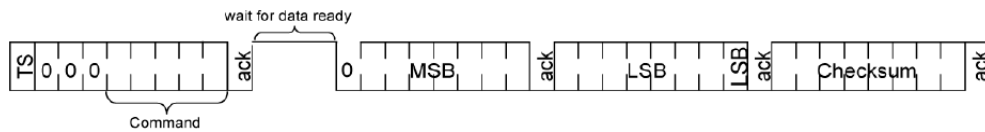
Při komunikaci s obvodem je nutné nastavit jakou veličinu budeme chtít změřit. Po inicializaci vysílání příkazu obvodu následuje samotný příkazový bajt, jehož bity 7:5 jsou adresové bity(vždy 0) a samotný příkaz, 00101 pro měření relativní vlhkosti nebo 00011 pro měření teploty. Následuje změřená veličina a převod do digitální podoby. Poté ihned následují změřená data na výstup. Po odeslání dat je sekvence ukončena ještě jedním bajtem kontrolního součtu CRC-8. Celá sekvence získání dat ze senzoru je na obrázku 5.7.

Výstup ze senzoru při měření relativní vlhkosti je téměř lineární. Výslednou hodnotu při použití 14b AD převodníku dostaneme přepočítáním ze vztahu:

$$RH_{linear}[\%] = -4 + 0.0405 * ADC_{RH} + -2.8 * 10^{-6} * ADC_{RH}^2$$



Obrázek 5.6: Vnitřní blokové schéma senzoru SHT11



Obrázek 5.7: Datová sekvence měření senzoru

kde ADC_{RH} je 2-bajtová hodnota relativní vlhkosti z AD převodníku.

Hodnotu teploty ve $^{\circ}C$ spočítáme jednodušeji, protože výstup ze senzoru je lineární. Při vstupním napětí 3.3V a použitím 14b převodníku spočítáme podle následujícího vztahu:

$$T[^{\circ}C] = -39.64 + 0.01 * ADC_T$$

kde ADC_T je 2-bajtová hodnota teploty z AD převodníku.

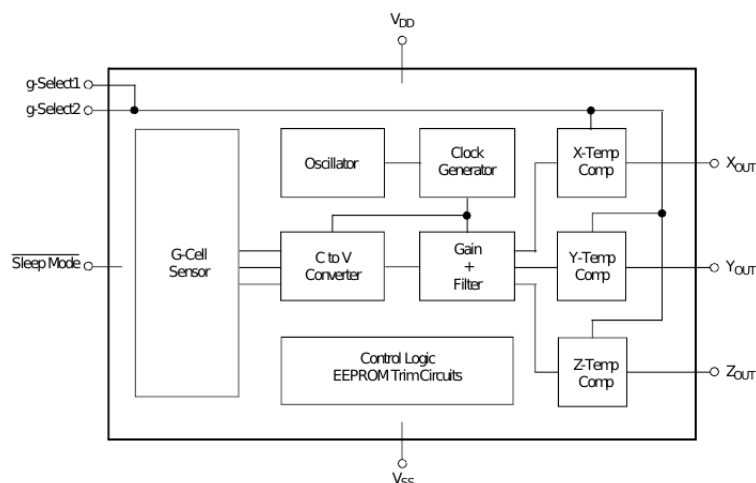
Odchytky od naměřených hodnot jsou u relativní vlhkosti $\pm 4.5\%$ a u teploty $\pm 0.4^{\circ}C$.

Senzor zrychlení

Obvod MMA7260QT [7] je kapacitním senzorem, který detekuje zrychlení ve všech třech osách. Výstup obvodu je analogový, pro každou osu zvlášť a je přiveden na vstupy A/D převodníku mikrokontroleru. Řídicím vstupem tohoto obvodu jsou piny g-Select1 a g-Select2, kterými se určuje citlivost senzoru. Signálem přivedeným na vstup pinu SLEEP lze obvod uspat pro snížení spotřeby.

Výstupní napětí v klidovém stavu při citlivosti 800mV/g je 1.65V. Při pohybu v kladném směru os X, Y a Z označovaných jako pohyb vlevo, dopředu a nahoru se hodnota napětí na výstupu zvyšuje, kdežto v opačných směrech pohybu po osách ozančovaných jako vpravo, dozadu a dolů se napětí na výstupu snižuje. Pohyb je brán z pohledu zhora na součástku. Pro detekci pohybu je tedy potřeba porovnávat výstupní napětí s hodnotou 1.65V.

Na straně mikrokontroleru je pro získání hodnoty napětí potřeba inicializovat převod. Nejprve se musí povolit převod zapsáním jedničky na bit ADEN registru ADCSRA. Pro signalizaci přerušení po konverzi se musí také povolit globální přerušení v SREG registru a zaplat jedničku na na bit ADIE registru ADCSRA. Po skončení konverze se nastaví flag ADIF. Dále je potřeba nastavit zdroj referenčního napětí. Nastavením bitů 7:6 registru ADMUX na hodnotu 11b zvolíme referenční napětí 2.56V. Výstupy ze senzoru jsou připojeny na port A na vstupy ADC5:ADC7. Volbu příslušného vstupu pro konverzi provedeme zapsáním hodnot 00101b:00111b na bity 4:0 registru ADMUX. Po tomto nastavení můžeme spustit samotnou konverzi nastavením jedničky bitu ADSC registru ADCSRA. Výsledek konverze



Obrázek 5.8: Vnitřní blokové schéma senzoru zrychlení

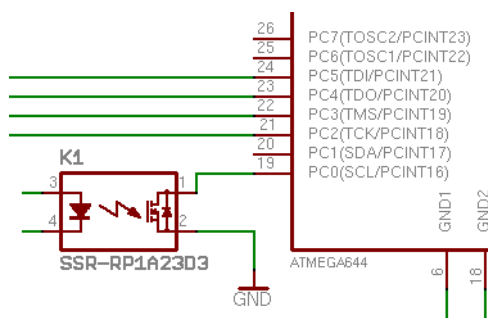
se objeví v registrech ADCL a ADCH. Výsledné napětí lze potom spočítat dle následujícího vzorce:

$$ADC = \frac{V_{IN} * 1024}{V_{REF}}$$

kde V_{IN} je vstupní napětí a V_{REF} je referenční napětí zvolené 2.56V.

Ovládání SSR relé

Modul sloužící ke spínání napětí 220V je osazen SSR relé, které spíná přivedením logické jedničky na pin 1. Tento pin je připojený k pinu PC0 portu C mikrokontroleru. K pinům 3 a 4 SSR relé je připojena fáze napětí 220V přes modrý konektor na modulu.



Obrázek 5.9: Schéma zapojení SSR relé a mikrokontroleru ATmega644

Kapitola 6

Závěr

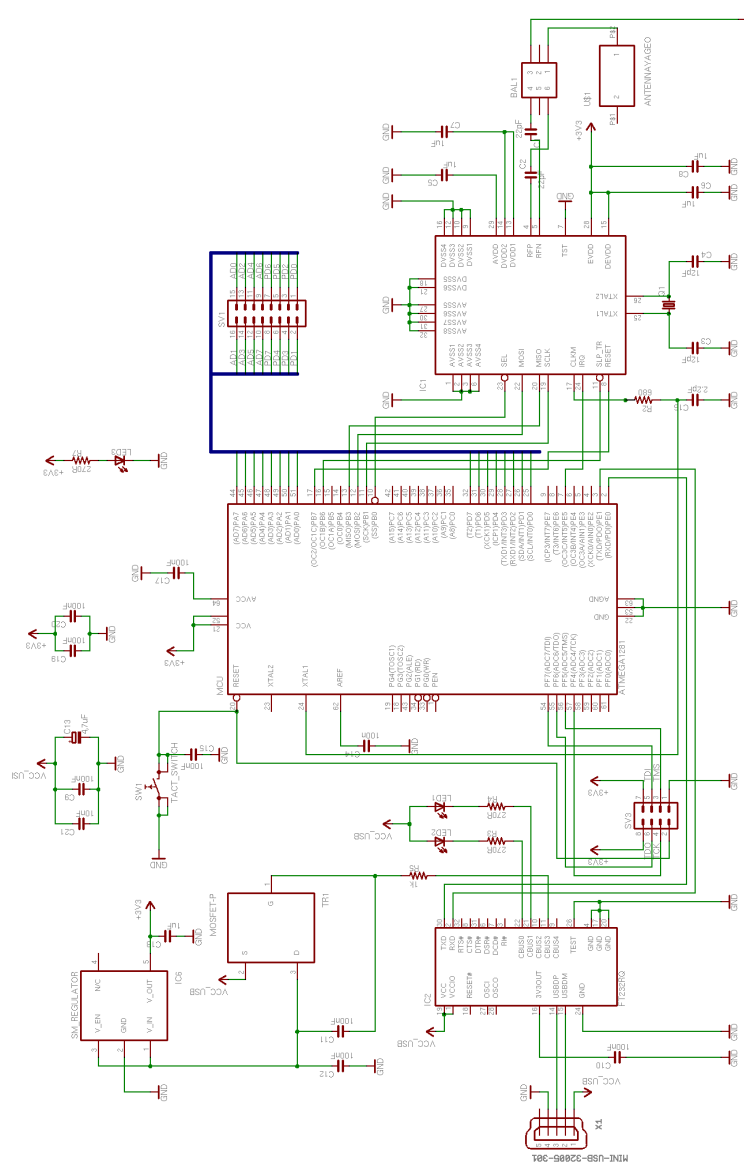
Cílem tohoto projektu bylo navrzení a praktická implementace bezdrátových sensorových modulů pracujících se standardem ZigBee. Během této práce jsem se seznámil s několika různými typy sensorů. V kapitole o senzorech je popsáno jejich základní rozdělení a charakteristiky. Dále jsem prostudoval bezdrátový síťový standard ZigBee a jeho možné použití v sensorových sítích. Popsal jsem základní vlastnosti tohoto standardu, možné topologie a detailně popsals zabezpečení této sítě. Navrhl jsem pět modulů pro práci v této síti a vybral vhodné součástky pro výrobu.

Po teoretické části tohoto projektu, popsané ve druhé a třetí kapitole a části čtvrté kapitoly, následovala samotná implementace. Tedy vytvořil jsem v návrhovém programu Eagle knihovnu všech součástek a dále samotná schémata. Velmi jsem se soustředil na samotný návrh desek plošných spojů, aby výsledný plošný spoj byl co nejmenší. Dále jsem obstaral potřebné součástky pro výrobu, nechal jsem vyrobit desky plošných spojů a posléze osadil. Následovalo oživení a testování jednotlivých modulů, o čemž je napsáno v páté kapitole.

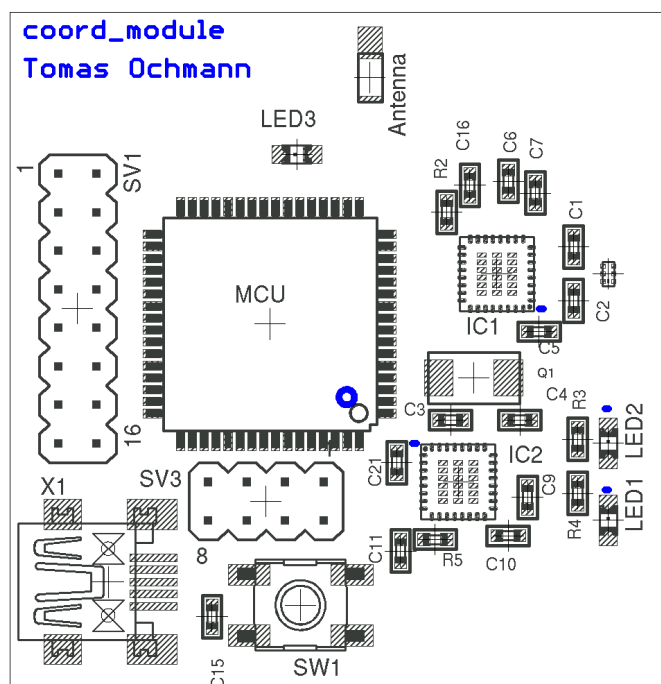
Podařilo se mi tedy navrhnout a prakticky realizovat moduly pro bezdrátovou sensorovou síť ZigBee. Vytvořil jsem hlavní modul koordinátoru celé sítě, tři moduly se senzorem teploty a vlhkosti, intenzity osvětlení a senzorem zrychlení. Dále jeden modul, který má na starosti spínání napětí 220V pro spotřebič (např. klimatizace, světlo). Všechny moduly jsou funkční z pohledu hardwarového. Pouze u hlavního modulu koordinátoru nastala chyba při výrobě plošného spoje, nelze tedy přímo komunikovat s počítačem přes rozhraní USB. Více je popsáno v kapitole 4.2. Z důvodu velmi zdlouhavého získávání součástek a praktického kompletního dokončení výroby těsně před odevzdáním práce nebylo možné realizovat softwarové vybavení jednotlivých modulů. Zdlouhavé získávání součástek bylo způsobeno také tím, že některé součástky jsou poměrně drahé, proto jsem usiloval o získání vzorků. Ty však nedošly. Dále jsem jednotlivé komponenty získával od distributora z Velké Británie, což také způsobilo nemalé zpoždění. Nakonec byly také problémy s osazením součástek, již dříve zmíněné.

Vytvořený projekt má význam v oblasti bezdrátových sensorových sítí, jelikož toto využití je zatím málo rozvinuté. Malé rozměry modulů, příznivá cena při průmyslové výrobě, ale hlavně jejich spolehlivost, jednoduchost a velmi nízká spotřeba způsobená použitím kombinace úsporného 8-mi bitového procesoru a transeiveru firmy Atmel dává možnost širokému využití tohoto projektu v praxi. Budoucnost tohoto projektu je v dopsání softwarové části a vytvoření dalších sensorových modulů, díky kterým by např. mohl být vytvořen kompletní zabezpečovací systém budov a veřejných prostranství.

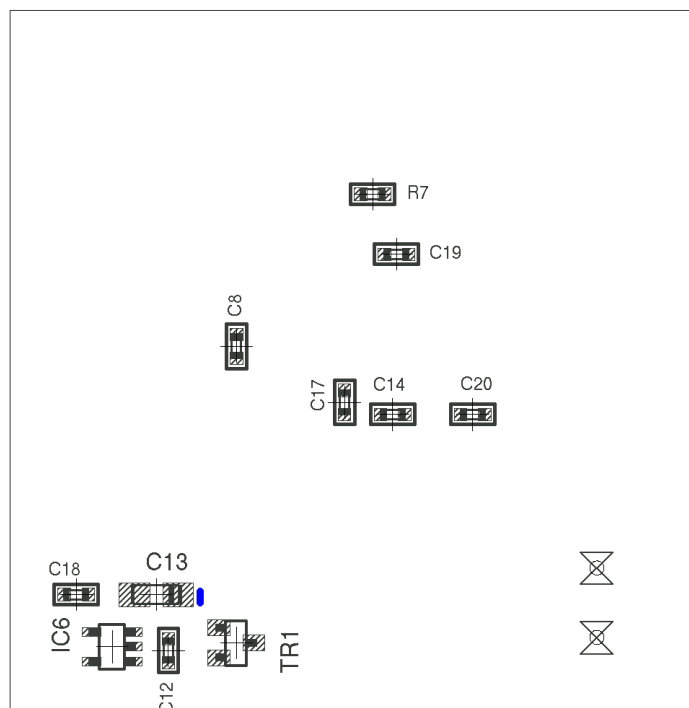
Obvodová schémata a rozmístění součástek



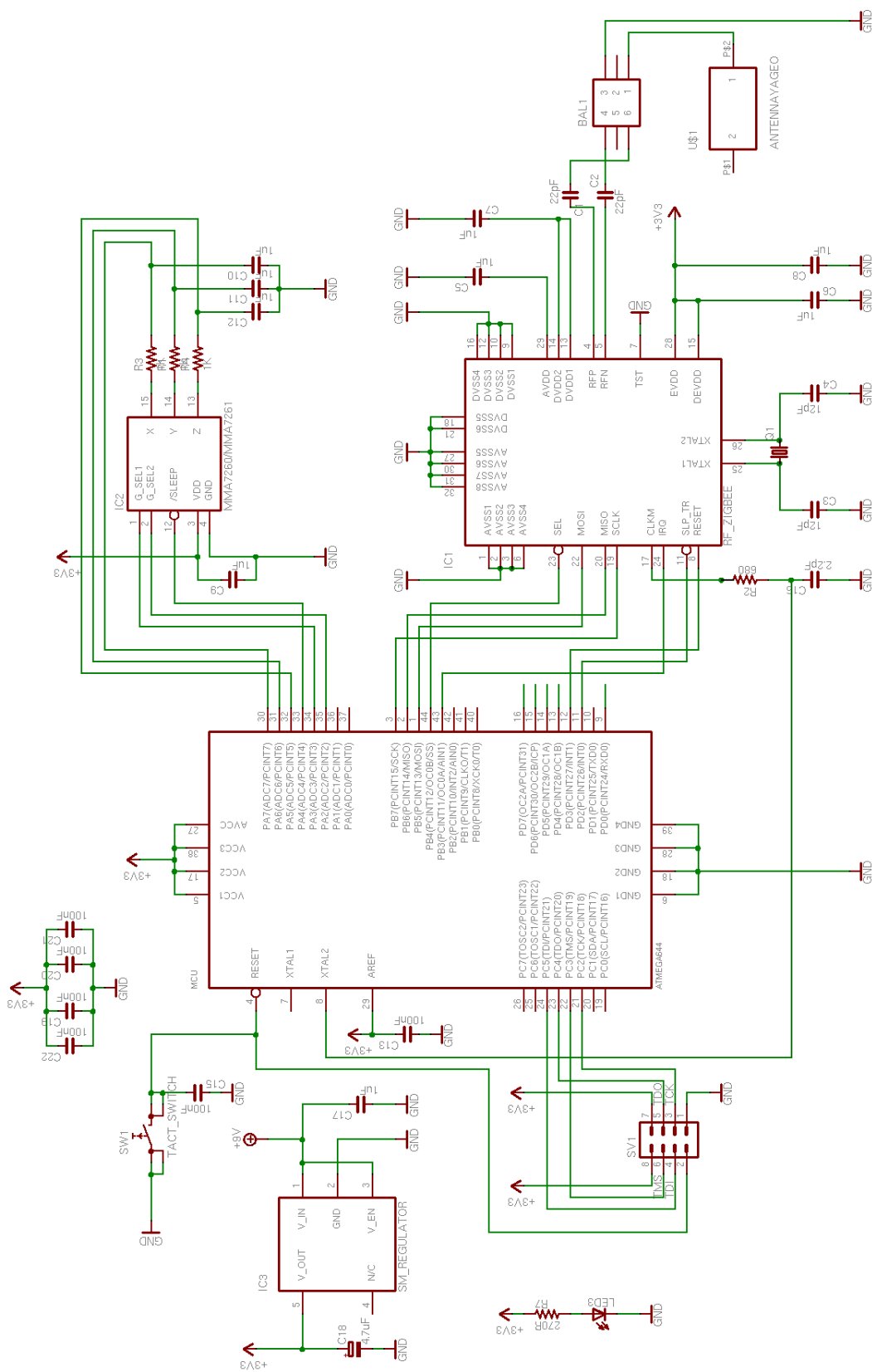
Obrázek 1: Schéma zapojení modulu koordinátoru



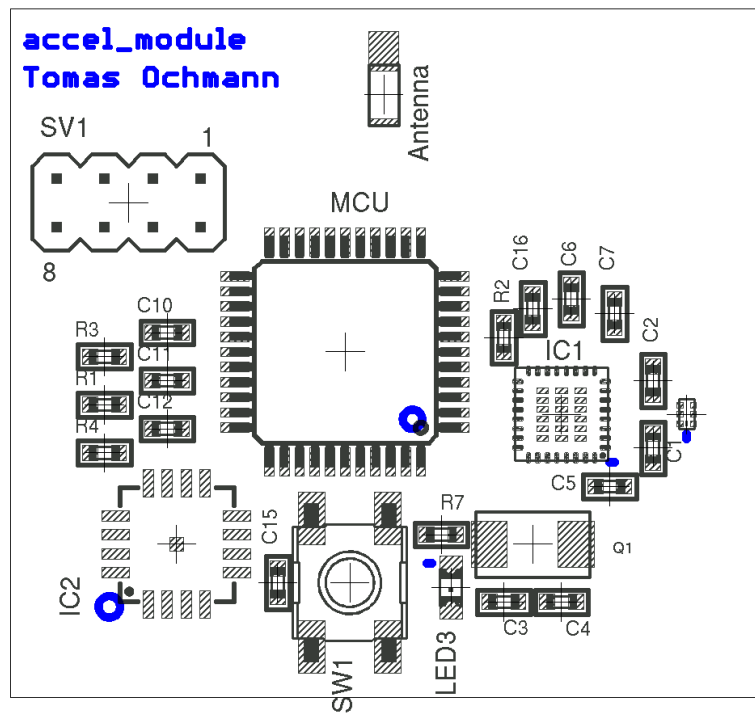
Obrázek 2: Rozmístění součástek modulu koordinátoru – horní vrstva



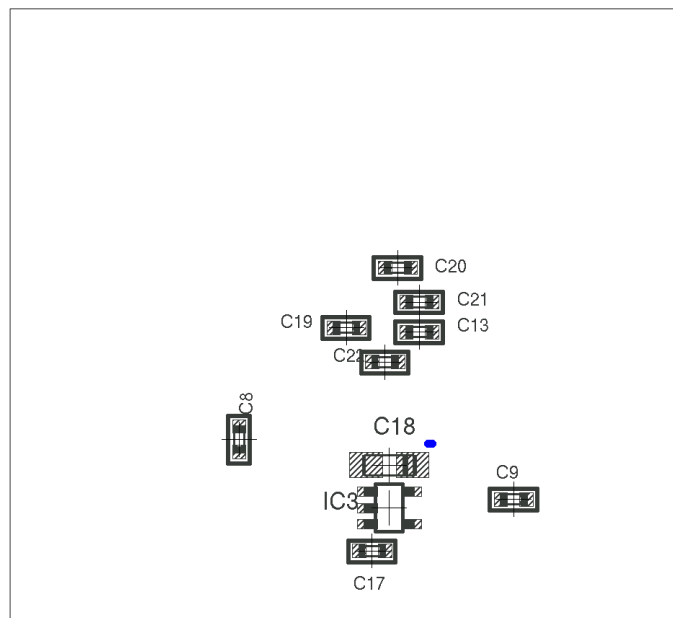
Obrázek 3: Rozmístění součástek modulu koordinátoru – spodní vrstva



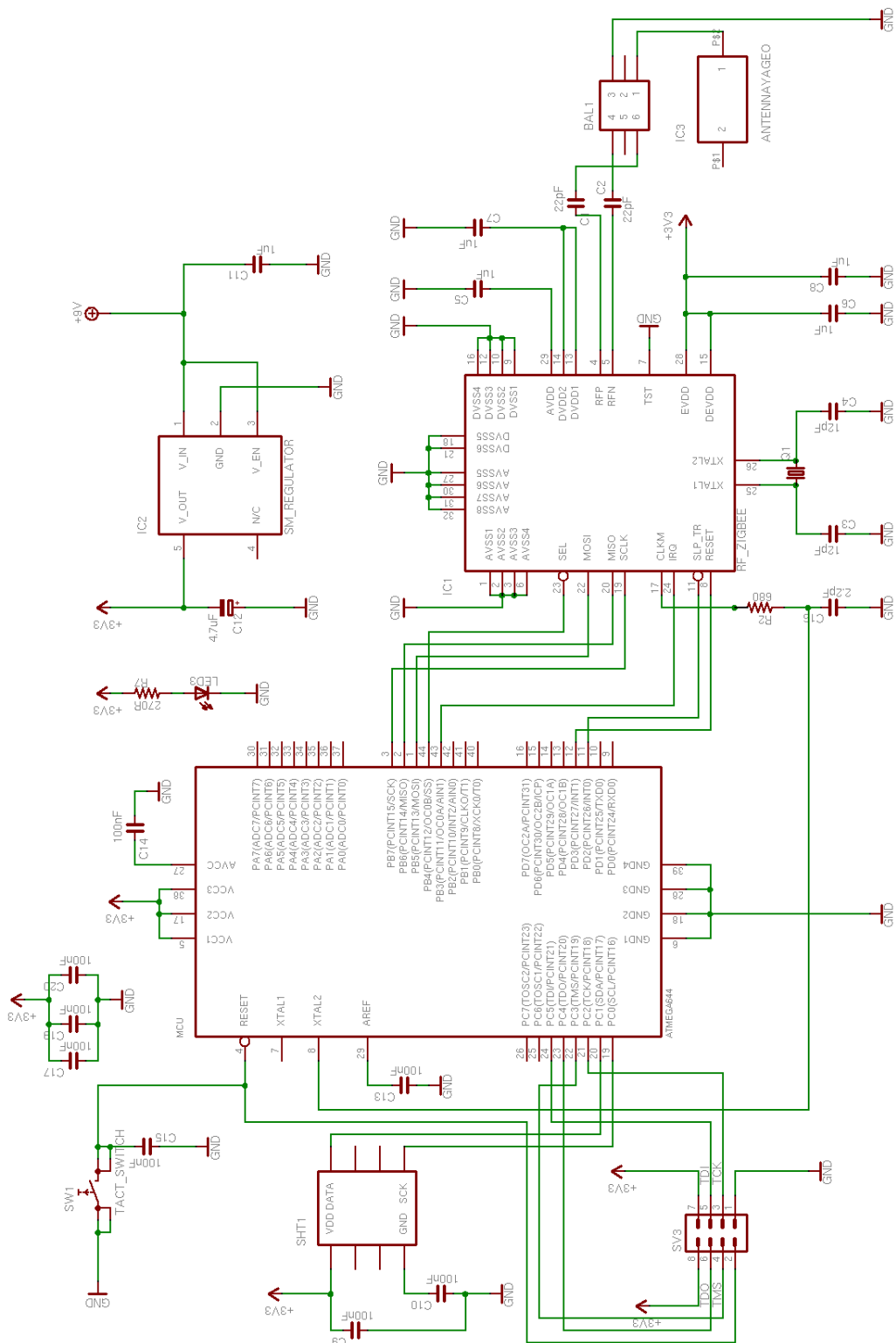
Obrázek 4: Schéma zapojení modulu zrychlení



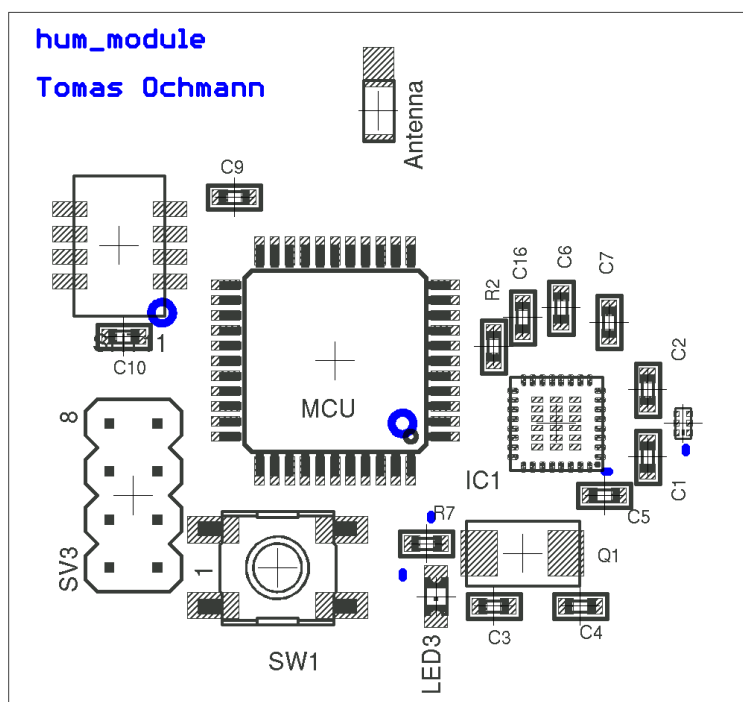
Obrázek 5: Rozmístění součástek modulu zrychlení – horní vrstva



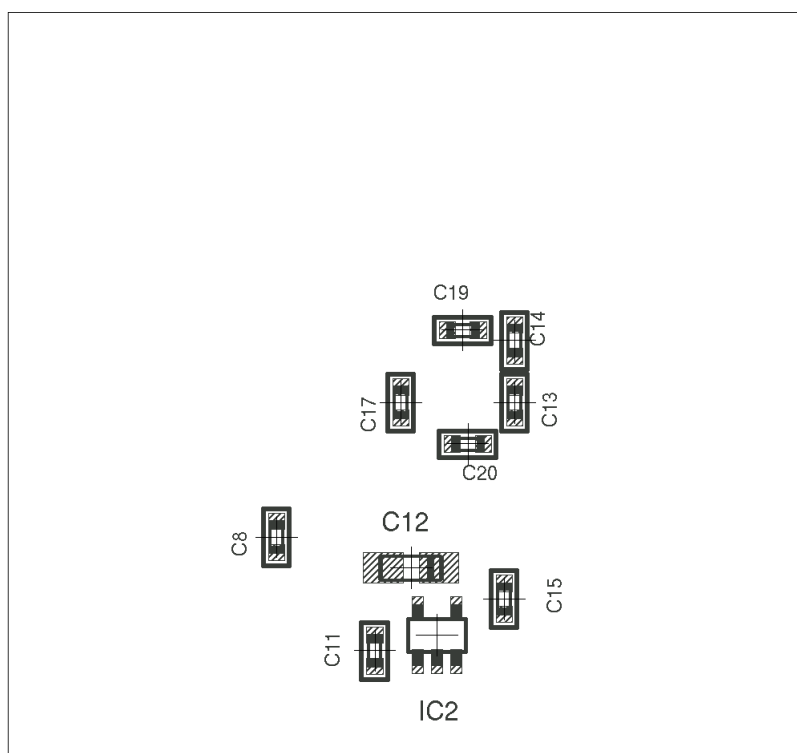
Obrázek 6: Rozmístění součástek modulu zrychlení – spodní vrstva



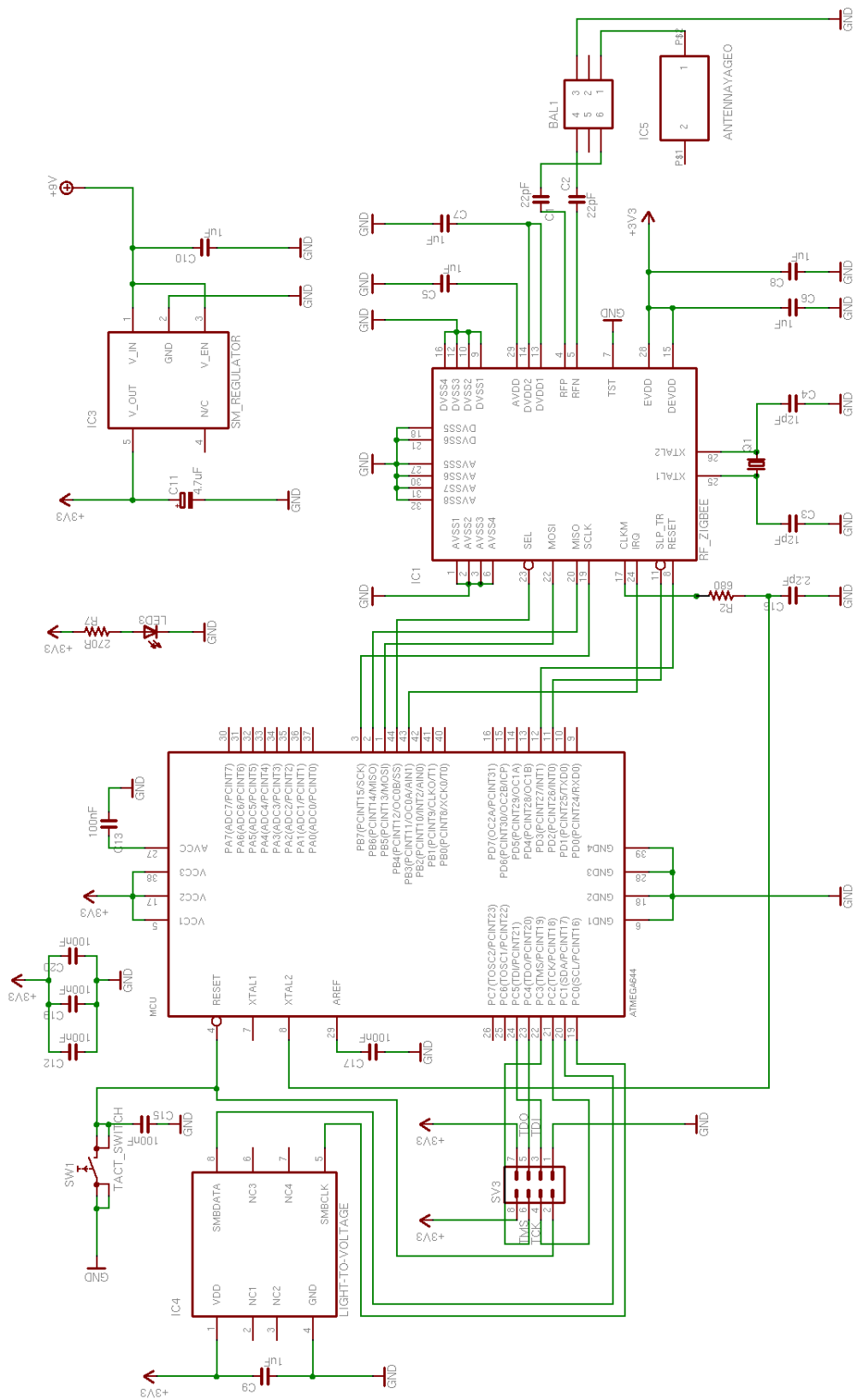
Obrázek 7: Schéma zapojení modulu teploty a vlhkosti



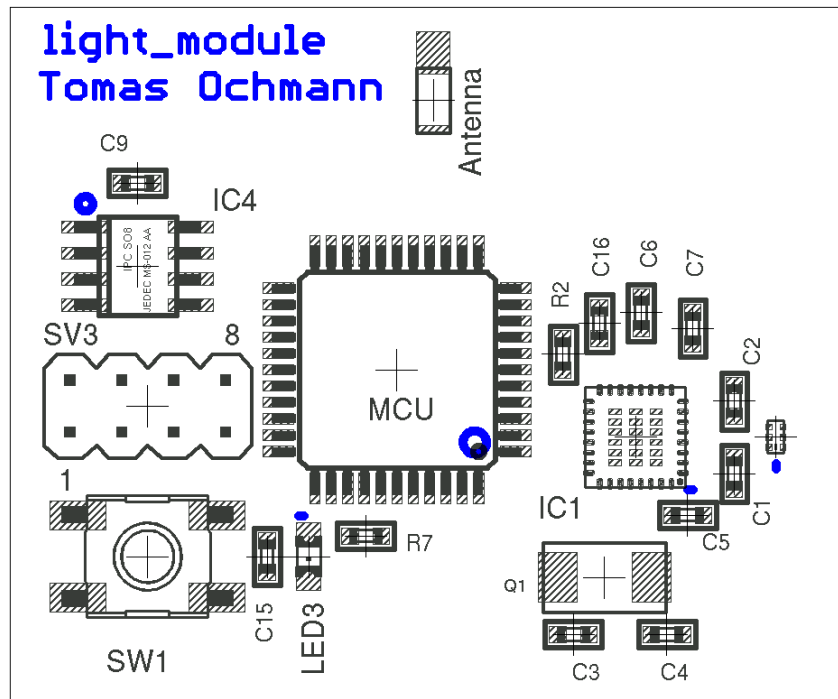
Obrázek 8: Rozmístění součástek modulu teploty a vlhkosti – horní vrstva



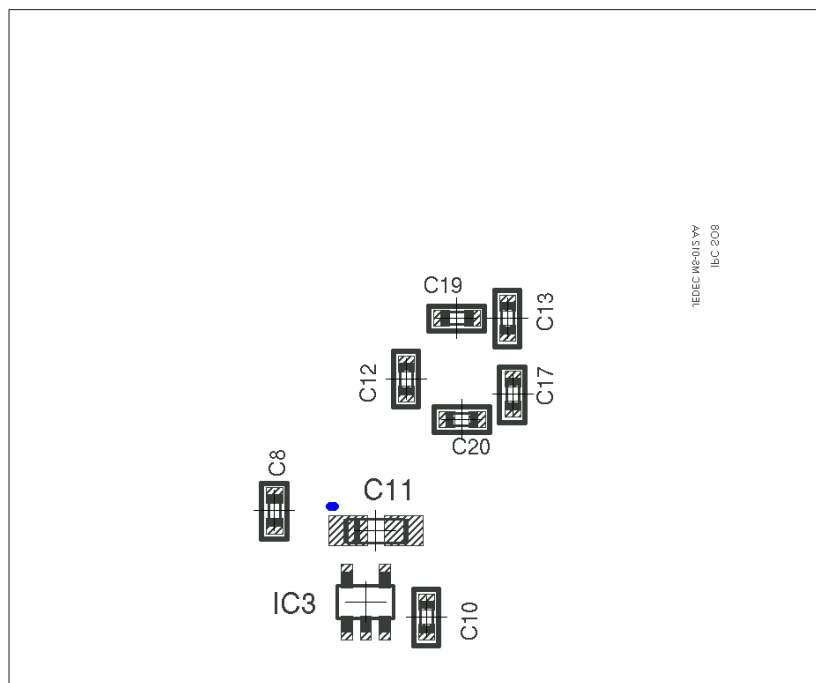
Obrázek 9: Rozmístění součástek modulu teploty a vlhkosti – spodní vrstva



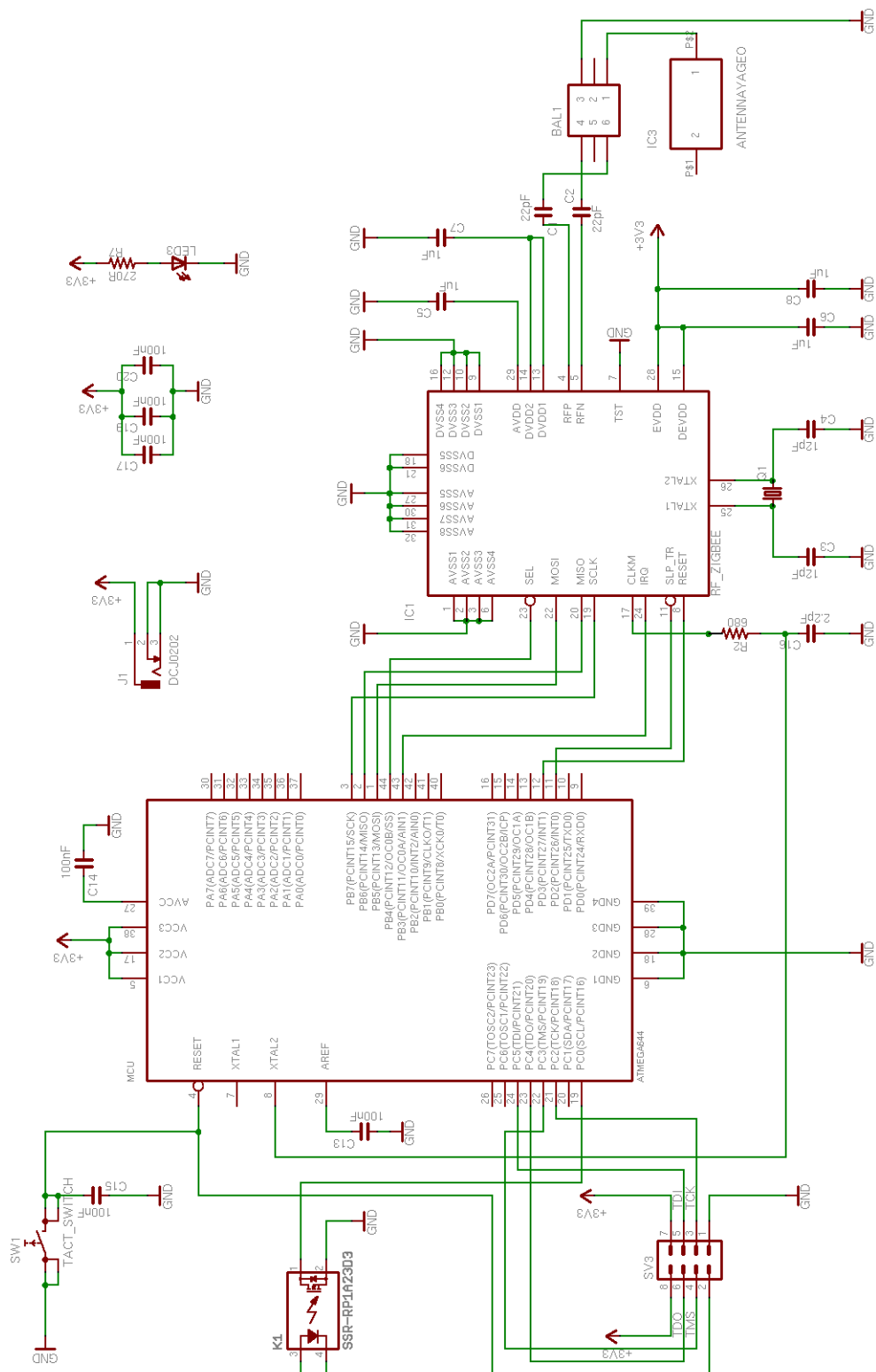
Obrázek 10: Schéma zapojení modulu osvětlení



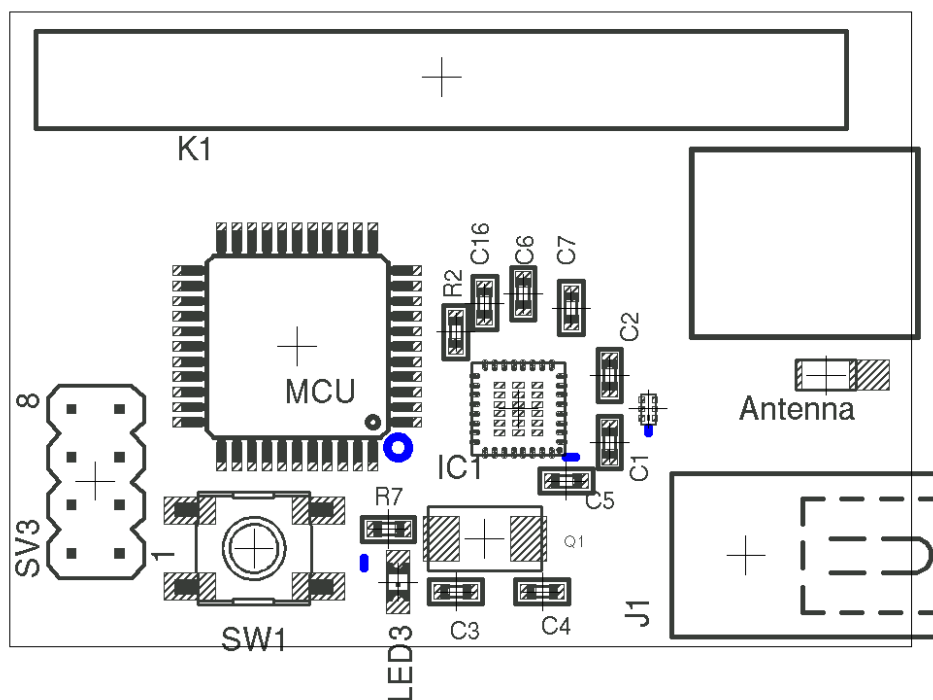
Obrázek 11: Rozmístění součástek modulu osvětlení – horní vrstva



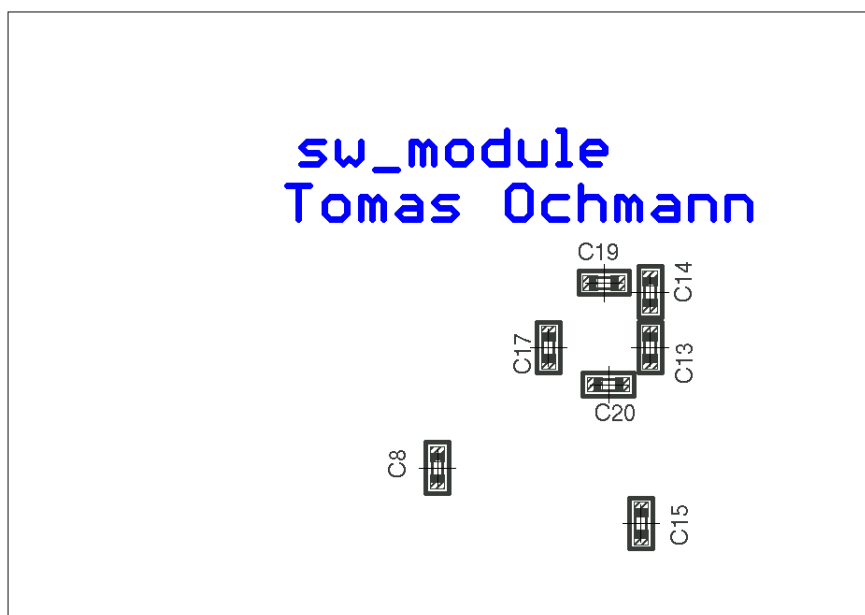
Obrázek 12: Rozmístění součástek modulu osvětlení – spodní vrstva



Obrázek 13: Schéma zapojení modulu přepínače

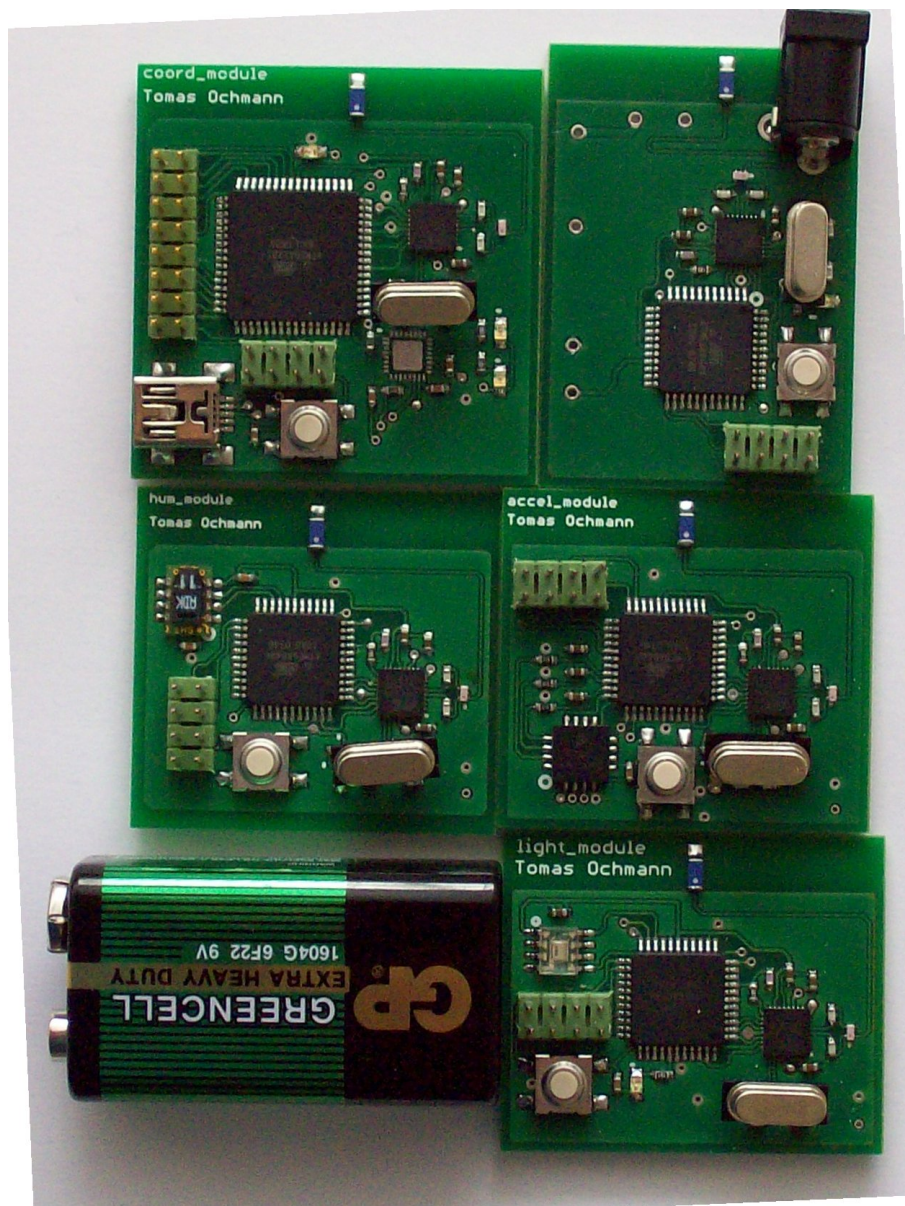


Obrázek 14: Rozmístění součástek přepínacího modulu – horní vrstva



Obrázek 15: Rozmístění součástek přepínacího modulu – spodní vrstva

Osazené moduly



Obrázek 16: Osazené moduly

Obsah přiloženého CD

- Data pro Eagle
- SW PRESTO – ovladače, program UP, jtag player
- WinAvr
- avrsvf
- zdrojové kódy tinyOS
- dokumentace k textové části DP
- seznamy součástek jednotlivých modulů
- datasheety použitých součástek

Literatura

- [1] Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). In *IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks Specific requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, New York, NY, USA: The Institute of Electrical and Electronics Engineers, Inc., 2003, ISBN 0-7381-3677-5 SS95127.
URL <http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf>
- [2] Asix: Programátor PRESTO. [online], poslední aktualizace 10. července 2008.
URL http://www.asix.cz/a6_presto.htm
- [3] ATMEL: Atmel Datasheets - ATmega640, ATmega1280, AT86RF230. [online].
URL <http://atmel.com/products/>
- [4] BERKELEY, U.: TinyOS. [online], poslední aktualizace 2004.
URL <http://www.tinyos.net/scoop/special/mission>
- [5] Center, A. M. T.: The features of III ZigBee Stack. online, 2007, [Online; navštíveno 7.4.2008].
URL <http://zigbee.iii.org.tw/EN/ZIGBEE.PHP>
- [6] Citizen: AT-CUT CRYSTAL UNIT. [online], poslední aktualizace 2007.
URL <http://www.citizencrystal.com/images/pdf/m-cs10.pdf>
- [7] Freescale: 1.5g - 6g Three Axis Low-g Micromachined Accelerometer. [online], poslední aktualizace březen 2008.
URL
http://www.freescale.com/files/sensors/doc/data_sheet/MMA7260QT.pdf
- [8] FTDI: Future Technology Devices International Ltd. FT232R USB UART IC. [online], poslední aktualizace 2008.
URL http://www.ftdichip.com/Documents/DataSheets/DS_FT232R.pdf
- [9] H. Labiod, H. A.; Santis, C. D.: Security in WLAN, WPAN, WSN and WMAN through Wi-FiTM, BluetoothTM, ZigBeeTM and WiMAXTM. In *WI-FI TM, BLUETOOTH TM, ZIGBEE TM AND WIMAX TM*, Springer Netherlands, 2007, ISBN 978-1-4020-5396-2 (Print) 978-1-4020-5397-9 (Online), s. 153–220.
URL <http://www.springerlink.com/content/v02mx8605rr7002w/>

- [10] KOTON, J.; ČÍKA, P.; KŘIVÁNEK, V.: Standard nízkorychlostní bezdrátové komunikace ZigBee. [online], poslední aktualizace 18.4. 2006.
URL <http://access.feld.cvut.cz/view.php?navezclanku=standard-nizkorychlostni-bezdratove-komunikace-zigbee&cislocclanku=2006032001>
- [11] Lee, J.: The Emergence of Networking Abstractions and Techniques in TinyOS. [online], poslední aktualizace 9. června 2004.
URL <http://kernel.kaist.ac.kr/~jinsoo/course/cs710/slides/TinyOS.pdf>
- [12] Messerges, T. S.; Cukier, J.; Kevenaar, T. A. M.; aj.: A security design for a general purpose, self-organizing, multihop ad hoc wireless network. In *SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, New York, NY, USA: ACM, 2003, ISBN 1-58113-783-4, s. 1–11.
URL <http://doi.acm.org/10.1145/986858.986860>
- [13] Nguyen, S. T.; Rong, C.: *Autonomic and Trusted Computing*. Springer Berlin / Heidelberg, 2007, ISBN 978-3-540-73546-5.
- [14] PISTER, K.: SMART DUST. [online].
URL <http://robotics.eecs.berkeley.edu/~pister/SmartDust/>
- [15] SARANGAPANI, J.: *Wireless Ad Hoc and Sensor Networks*. CRC Press, Taylor and Francis Group, 2007, ISBN 0-8247-2675-8.
- [16] Sensirion: Datasheet SHT1x. [online], poslední aktualizace červen 2008.
URL http://www.sensirion.com/en/pdf/product_information/Datasheet-humidity-sensor-SHT1x.pdf
- [17] Taos: Technical datasheet. [online], poslední aktualizace říjen 2007.
URL <http://www.taosinc.com/images/product/document/TSL2550-e75.pdf>
- [18] VOJÁČEK, A.: ZigBee - novinka na poli bezdrátové komunikace. [online], poslední aktualizace 8.6. 2005.
URL <http://hw.cz/Rozhrani/ART1299-ZigBee---novinka-na-poli-bezdratove-komunikace.html>
- [19] WIKIPEDIA: JTAG. [online], poslední aktualizace 28. července 2008.
URL <http://en.wikipedia.org/wiki/JTAG>
- [20] WIKIPEDIA: TinyOS. [online], poslední aktualizace 25. prosince 2007.
URL <http://en.wikipedia.org/wiki/TinyOS>
- [21] WIKIPEDIA: Wireless sensor network. [online], poslední aktualizace 2. prosince 2007.
URL http://en.wikipedia.org/wiki/Wireless_sensor_network
- [22] WIKIPEDIA: ZigBee. [online], poslední aktualizace 4.11. 2007.
URL <http://cs.wikipedia.org/wiki/ZigBee>
- [23] WOJCIAZSYK, P.: Inteligentní senzory a bezdrátové sítě. [online], poslední aktualizace 29.4. 2005.
URL www.fs.vsb.cz/akce/2005/asr2005/Proceedings/papers/519.pdf

- [24] Xilinx: WebPACK. [online], poslední aktualizace 2008.
URL http://www.xilinx.com/ise/logic_design_prod/webpack.htm
- [25] Yageo: Data sheet. [online], poslední aktualizace 2007.
URL <http://www.farnell.com/datasheets/86503.pdf>
- [26] ZIGBEE: ZigBee Specification. [online], poslední aktualizace 1.12. 2006.
URL <http://www.zigbee.org/en/index.asp>
- [27] ZigBee specification. In *ZigBee specification*, ZigBee Alliance, Říjen 2007, [Online; navštíveno 7.4.2008].
URL http://www.zigbee.org/en/spec_download/