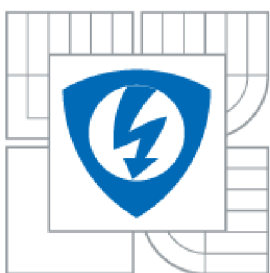




**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**  
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT RADIO ELECTRONICS

## **BEZDRÁTOVÁ SENZOROVÁ SÍŤ**

WIRELESS SENSOR NETWORK

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

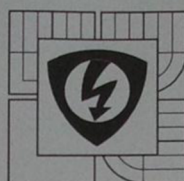
**AUTOR PRÁCE**  
AUTHOR

**Martin Ševčík**

**VEDOUcí PRÁCE**  
SUPERVISOR

ing. Jan Prokopec, Ph.D.

BRNO 2011



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav radioelektroniky

## Bakalářská práce

bakalářský studijní obor  
Elektronika a sdělovací technika

**Student:** Martin Ševčík

**Ročník:** 3

**ID:** 78410

**Akademický rok:** 2010/11

**NÁZEV TÉMATU:**

**Bezdrátová senzorová síť**

### POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s technologií ZigBee pro vytváření bezdrátových sítí s malou přenosovou rychlostí. Navrhněte jednoduchou senzorovou síť, která bude zjišťovat teplotu, otevřené dveře a další parametry v několika místnostech.

Vytvořte bezdrátovou síť pomocí dostupných ZigBee modulů a senzorů. Vytvořte řídicí software pro předávání dat mezi senzory a bezdrátovými moduly. Naprogramujte software pro zobrazení dat ze senzorů na PC.

### DOPORUČENÁ LITERATURA:

[1] 802.15.4 IEEE Standard: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANS), IEEE New York 2003.

**Termín zadání:** 7.2.2011

**Termín odevzdání:** 27.5.2011

**Vedoucí práce:** Ing. Jan Prokopec, Ph.D.

**Konzultanti bakalářské práce:**

prof. Dr. Ing. Zbyněk Raida  
předseda oborové rady



### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

## **Abstract**

This bachelor thesis deal with the ZigBee wireless technology and design of a simple sensor network on its principle. The network is designed for a smaller apartment. As the temperature sensor is used DS1631 thermometer. Communication of the coordinator with the computer ensures XPort®. Used modules from the Meshnetics company integrates the AVR processor and the radio transmitter / receiver at 2.4 GHz operating frequency.

## **Keywords**

ZigBee, ZigBit, topology, DS1631, XPort®, coordrinator, sensor, network, technologies, wireless.

## **Abstrakt**

Tato bakalářská práce se zabývá bezdrátovou technologií ZigBee a návrhem jednoduché senzorové sítě na jejím principu. Síť je navrhnutá na menší byt. Jako teplotní čidlo je použito teploměru DS1631. Komunikaci koordinátoru s počítačem zajišťuje XPort®. Použité moduly společnosti Meshnetics v sobě integrují AVR procesor a radiový vysílač/přijímač na pracovní frekvenci 2,4 GHz.

## **Klíčová slova**

ZigBee, ZigBit, topologie, DS1631, XPort®, koodrinátor, senzor, síť, technologie, bezdrátové

ŠEVČÍK, M. *Bezdrátová senzorová síť*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 62 s. Vedoucí bakalářské práce Ing. Jan Prokopec, Ph.D..

## Prohlášení

Jako autor bakalářské práce na téma Bezdrátová senzorová síť dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne.....

.....  
podpis autora

## Poděkování

Děkuji vedoucímu bakalářské práce Ing. Janu Prokopcovi, Ph.D., za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne.....

.....  
podpis autora

# Obsah

<b>1</b>	<b>Úvod</b>	<b>11</b>
<b>2</b>	<b>Bezdrátové technologie</b>	<b>12</b>
2.1	Bluetooth.....	12
2.2	Wi-Fi .....	12
2.2.1	Způsoby zapojení do sítě Wi-Fi .....	13
	Ad – hoc sítě .....	13
	Infrastrukturní sítě.....	13
2.3	ZigBee.....	14
2.4	Standard 802.15.4.....	15
2.4.1	Specifikace radiové čisti.....	16
2.4.2	Referenční síťový model .....	17
2.4.3	Topologie sítě .....	20
2.4.4	Vytváření nové sítě.....	22
2.4.5	Zabezpečení komunikace.....	22
2.4.6	Synchronizace zařízení ZigBee .....	23
2.4.7	Spotřeba zařízení .....	23
<b>3</b>	<b>Návrh bezdrátové sítě</b>	<b>24</b>
3.1	ZigBit modul modul ZDM-A1281-A2 .....	25
3.1.1	Klíčové vlastnosti: .....	25
3.1.2	Základní specifikace.....	26
3.2	Digitální teploměr DS1631.....	27
3.2.1	Základní vlastnosti:.....	27
3.2.2	Funkční diagram.....	28
3.2.3	Funkce termostatu:.....	29
3.2.4	Rychlost odezvy v závislosti na bitovém rozsahu .....	29
3.2.5	Dvou vodičová seriová datová sběrnice.....	30
3.3	XPort®.....	31
3.3.1	Charakteristika.....	32

---

3.3.2	Popis sériového rozhraní .....	32
3.3.3	Popis rozhraní ethernetu .....	34
<b>4</b>	<b>Realizace hardwaru</b>	<b>36</b>
4.1	Koordinátor.....	36
4.2	Koncové zařízení – Teploměr .....	38
<b>5</b>	<b>Softwarové vybavení</b>	<b>40</b>
5.1	Zobrazení dat v počítači.....	40
5.1.1	Testování aplikace.....	41
5.2	Program pro koordinátor .....	43
5.3	Program pro teploměr .....	44
<b>6</b>	<b>Závěr</b>	<b>46</b>
<b>7</b>	<b>Literatura</b>	<b>47</b>
<b>A</b>	<b>Schémata</b>	<b>50</b>
<b>B</b>	<b>Seznam použitých zkratk</b>	<b>52</b>
<b>C</b>	<b>Zdrojový kód PC aplikace</b>	<b>53</b>
<b>D</b>	<b>Vývojové diagramy</b>	<b>57</b>
<b>E</b>	<b>Seznam součástí</b>	<b>61</b>

## Seznam obrázků

<b>Obr. 1</b>	<b>Logo Bluetooth</b>	<b>12</b>
<b>Obr. 2</b>	<b>Logo Wi-Fi</b>	<b>13</b>
<b>Obr. 3</b>	<b>OSI model komunikačního protokolu ZigBee [3]</b>	<b>16</b>
<b>Obr. 4</b>	<b>Referenční model ZigBee [6]</b>	<b>18</b>
<b>Obr. 5</b>	<b>Struktury rámců MAC vrstvy [5]</b>	<b>19</b>
<b>Obr. 6</b>	<b>Příklad struktury sruperrámce [5]</b>	<b>19</b>
<b>Obr. 7</b>	<b>Možné topologie systému ZigBee [7]</b>	<b>21</b>
<b>Obr. 8</b>	<b>Návrh rozmístění několika senzorů v bytě</b>	<b>24</b>
<b>Obr. 9</b>	<b>Porovnání velikosti ZigBit™ modulu s mincí</b>	<b>26</b>
<b>Obr. 10</b>	<b>Rozložení pinů DS1631 [9]</b>	<b>27</b>
<b>Obr. 11</b>	<b>Blokový diagram teploměru DS1631 [9]</b>	<b>29</b>
<b>Obr. 12</b>	<b>Porovnání velikosti teploměru DS1631 s mincí</b>	<b>31</b>
<b>Obr. 13</b>	<b>Vnitřní zapojení XPortu® [11]</b>	<b>32</b>
<b>Obr. 14</b>	<b>Zapojení pinů XPortu při pohledu zespod [11]</b>	<b>34</b>
<b>Obr. 15</b>	<b>Fotografie XPort®</b>	<b>35</b>
<b>Obr. 16</b>	<b>Plošný spor redukce</b>	<b>36</b>
<b>Obr. 17</b>	<b>Redukce DIL8 na DIP8</b>	<b>36</b>
<b>Obr. 18</b>	<b>Plošný spoj koordinátoru</b>	<b>37</b>
<b>Obr. 19</b>	<b>Osazovací výkres koordinátoru</b>	<b>38</b>
<b>Obr. 20</b>	<b>Plošný spoj koncového zařízení – teploměru</b>	<b>39</b>
<b>Obr. 21</b>	<b>Osazovací výkres desky teploměru</b>	<b>39</b>
<b>Obr. 22</b>	<b>Obrázek okna programu pro čtení dat ze senzorů</b>	<b>40</b>



---

<b>Obr. 23</b>	<b>Obrázek z aplikace comocom</b>	<b>42</b>
<b>Obr. 24</b>	<b>Obrázek aplikace terminal</b>	<b>43</b>
<b>Obr. 25</b>	<b>Výpis programu DeviceInstaller</b>	<b>44</b>
<b>Obr. 26</b>	<b>Schéma koordinátoru</b>	<b>51</b>
<b>Obr. 27</b>	<b>Schéma teploměru</b>	<b>51</b>
<b>Obr. 28</b>	<b>Vývojový diagram koordinátoru</b>	<b>57</b>
<b>Obr. 29</b>	<b>Inicializace portu v PC programu</b>	<b>58</b>
<b>Obr. 30</b>	<b>Zpracování paketů PC programem</b>	<b>59</b>
<b>Obr. 31</b>	<b>Vývojový diagram koncového zařízení - teploměru</b>	<b>60</b>

## Seznam tabulek

<b>Tab. 1</b>	<b>Tabulka standardů Wi-Fi vč. frekvencí a přenosových rychlostí</b>	<b>14</b>
<b>Tab. 2</b>	<b>Porovnání základních parametrů bezdrátových standardů [3]</b>	<b>15</b>
<b>Tab. 3</b>	<b>Adresové pole MAC rámce [5]</b>	<b>20</b>
<b>Tab. 4</b>	<b>Příklady zobrazených teplot [9]</b>	<b>28</b>
<b>Tab. 5</b>	<b>Rychlost odezvy v závislosti na bitovém rozsahu [9]</b>	<b>30</b>
<b>Tab. 6</b>	<b>Tabulka kontrolních bitů [9]</b>	<b>30</b>
<b>Tab. 7</b>	<b>Popis pinů na sériové straně rozhraní XPortu [11]</b>	<b>33</b>
<b>Tab. 8</b>	<b>Popis pinů na straně rozhraní ethernetu. [11]</b>	<b>35</b>

# 1 Úvod

Tato práce si bere za téma bezdrátovou senzorovou síť postavenou na technologii ZigBee. Cílem práce je seznámit se s bezdrátovými technologiemi používanými na přenos dat. Podrobnější popis vybrané technologie ZigBee. Dále je cílem navrhnout a nakonec realizovat takovou síť.

## 2 Bezdrátové technologie

Na začátek si uvedeme krátký popis technologií pro přenos převážně datové informace a provedeme prorovnáání účelu jejich použití. Také zde budou uvedeny důvody, proč je pro tuto práci nejvhodnější použít zrovna technologii ZigBee.

### 2.1 Bluetooth

Bezdrátová technologie slouží převážně k propojení dvou mobilních zařízení. Používá se také pro propojení mobilního a pevného zařízení. Použití pro počítačové sítě se nevyužívá z důvodu malého dosahu.

Technologie Bluetooth je definovaná standardem IEEE 802.15.1. Spadá do kategorie počítačových sítí PAN.

Bluetooth pracuje v bezlicenčním pásmu na frekvenci 2,4 GHz jako konkurenční technologie uvedené v tomto přehledu. Komunikaci umožňuje přibližně na vzdálenost 10 – 100 metrů. Tyto hodnoty jsou teoretické a „laboratorní“ tzn. platí pouze ve volném terénu za ideálních podmínek. U mobilních zařízení z důvodu nižšího výkonu bývá dosah právě kolem deseti metrů. Nižšího výkonu se využívá z důvodů nároků na zdroj a u mobilních zařízení je nutná dlouhá výdrž na baterie. Pro technologii Bluetooth je definováno několik výkonových úrovní 2,5 mW, 10 mW, 100 mW. [1]



Obr. 1 Logo Bluetooth

### 2.2 Wi-Fi

Jedná se o standard bezdrátové lokální sítě Wireless LAN (WLAN). Standard je schválen pod označením IEEE 802.11 – který zaručuje její kompatibilitu. Tato technologie byla určena pro bezdrátové připojení do místních sítí. Dnes je hojně využívána i pro připojení k internetu. Svůj vliv na velkém rozšíření této technologie má provoz v bezlicenčním pásmu a nízká cena komponent. Výhoda velkého rozšíření se zároveň stává nevýhodou, protože v místech s velkou hustotou Wi-Fi sítí dochází k rušení a ovlivňování provozů těchto sítí, které tak nemusí být zcela stabilní. V poslední době se rozšiřuje kromě pásma 2,4 GHz provoz v pásmu 5 GHz, který umožňuje rychlejší přenos a

také podstatně menší rušení v důsledku většího počtu a odstupu přenosových kanálů. Výhodou je také v současnosti menší nasazení provozu na této frekvenci, tím i rušení od okolních vysílačů. Obě použité frekvence pracují v bezlicenčním pásmu.



Obr. 2 Logo Wi-Fi

### 2.2.1 Způsoby zapojení do sítě Wi-Fi

#### Ad – hoc sítě

Jedná se o přímé propojení dvou klientů, kteří jsou si rovnocenní. Tato pozice se označuje jako peer to peer. Tato síť se používá převážně příležitostně. Vzájemná komunikace takto propojených klientů probíhá pomocí SSID

#### Infrastrukturní sítě

Pro technologii Wi-Fi typická struktura. Taková síť se skládá z jednoho nebo více přístupových bodů označovaných jako AP (Access Point). V takovém případě vysílá každý přístupový bod svůj vlastní identifikátor SSID uživatel si vybere do které sítě se připojí, lze tak jednoduše zvolit která síť nabízí kvalitnější signál nebo ta do které má uživatel povolen přístup. [2]

Standard	Pásmo [GHz]	Maximální rychlost [Mbit/s]
IEEE 802.11	2,4	2
IEEE 802.11a	5	54
IEEE 802.11b	2,4	11
IEEE 802.11g	2,4	54
IEEE 802.11n	2,4 nebo 5	600

Tab. 1 Tabulka standardů Wi-Fi vč. frekvencí a přenosových rychlostí

## 2.3 ZigBee

Této technologii bude věnována samostatná kapitola, zde je uveden jen krátký přehled pro srovnání.

Snaha ZigBee není pokus o konkurenci již používaným a rozšířeným Wi-Fi a Bluetooth, ale nasazení v aplikacích, kde uvedené technologie nepodávají ideální výsledky nebo jsou z nějakého důvodu nevhodné, přesto je ve většině případů jejich použití možné.

Zařízení standardu ZigBee je tedy vhodné použít pro aplikace s bateriovým napájením díky velmi malé spotřebě a tím velké životnosti baterií. Tato vlastnost je mimo jiné vykoupena nižší přenosovou rychlostí, srovnatelnou se sériovým portem RS-232. Tato rychlost však pro většinu užívaných aplikací zcela dostačuje.

ZigBee ostatní technologie snadno předčí v topologii sítě a přesnému propracovanému adresování, díky kterému je možné snadno vytvořit složitou fungující síť.

Tyto vlastnosti předurčují ZigBee použití v aplikacích jako jsou bezdrátové senzory, na které se také zaměřuje tato práce tak i dálkové spínání a ovládání spotřebičů.

### Další příklady aplikací:

- Řízení budov - řízení vstupu, osvětlení, klimatizace, rolet apod.
- Automatizace - bezdrátová komunikace senzorů, řízení motorů a regulátorů
- Zdravotní péče - monitorování pacienta
- Sport - monitorování tělesných funkcí za pohybu (teplota, tep, tlak apod.)
- Spotřební elektronika - automatické dálkové ovládání domácích spotřebičů
- Bezdrátová komunikace počítačových periférií - klávesnice, myš apod.
- Zabezpečovací systémy - čipové přístupové karty, bezdrátové klávesnice, senzory apod. [3]

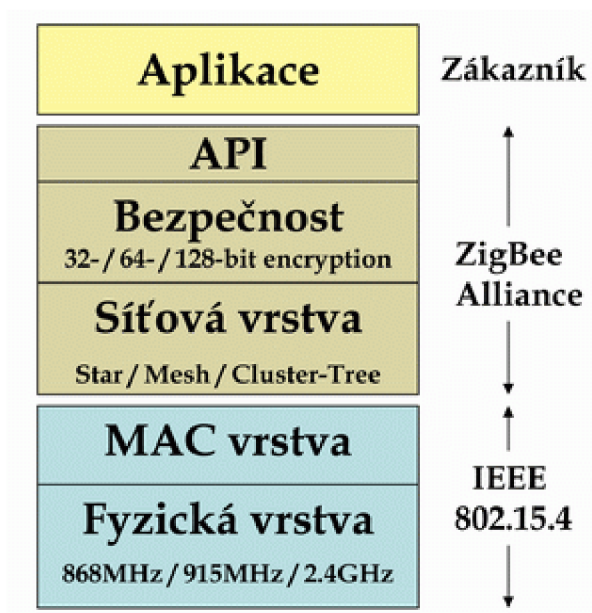
Obchodní jméno Standard	GPRS/GSM 1xRTT/CDMA	Wi-Fi™ 802.11b	Bluetooth™ 802.15.1	ZigBee™ 802.15.4
Aplikační zaměření	Široké oblasti Hlas & Data	Web, Email, Video	Náhrada za kabel	Monitorování & Řízení
Systémové zdroje (paměť)	16MB a více	1MB a více	250KB a více	4KB - 32KB
Životnost baterií (dny)	1-7	0.5 - 5	1 - 7	100 – 1 000 i více
Max. velikost sítě (počet uzlů/sít')	1	32	7	65 000 ( příp. až 2 <sup>64</sup> )
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

Tab. 2 Porovnání základních parametrů bezdrátových standardů [3]

## 2.4 Standard 802.15.4

Stejně jako každý jiný komunikační standard i ZigBee lze popsat OSI modelem. Ten lze rozdělit do třech základních bloků podle toho kým jsou definovány (viz. Obrázek .) :

[1]



Obr. 3 OSI model komunikačního protokolu ZigBee [3]

#### 2.4.1 Specifikace radiové čísti

Jak je již uvedeno v úvodu, ZigBee pracuje v bezlicenčním pásmu. Tyto pásma jsou definovány tři z důvodu odlišných podmínek pro některé země.

- Pásmo 915 MHz, 10 kanálů, přenosová rychlost 40 kb/s, definováno pro Ameriku a Austrálii
- Pásmo 868 MHz, 1 kanál, přenosová rychlost 20 kb/s, definováno pro Evropu
- Pásmo ISM 2,4 GHz, 16 kanálů, přenosová rychlost 250 kb/s, definováno celosvětově [3]

Signál je modulovaný metodou O-QPSK což je offsetová kvadrurní fázová modulace.

Přenos éterem realizuje metoda DSSS. Jde o techniku rozprostřeného spektra kde se každý bit určený k přenosu je nahrazen určitou pseudonáhodnou početnější sekvencí bitů. Tato nově vytvořená bitová sekvence se moduluje na nosnou a přenáší v radiovém spektru. Je sice zavedena umělá datová nadbytečnost. Výhodou a také důvodem proč se tato metoda používá, je menší náchylnost na rušení. Tuto vlastnost způsobí ono rozprostření do větší části radiového spektra. Pokud nezná přijímač kódovací mechanismus, jeví se mu takto přijímaný signál jako náhodný šum. Je zde tedy i jistá forma zabezpečení.

[5]



Použitá přístupová metoda je CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance and optional time slotting). Jde o metodu mnohonásobného přístupu s nasloucháním nosné. Tato metoda je charakteristická tím, že před vysláním paketu chvíli naslouchá, zda je přenosové médium volné, pokud ano zahájí vyslání potřebované informace. Jinak čeká náhodnou dobu před opakováním předešlé operace. Není zde detekce kolizí jako u systému CDSMA/CD. Kolizím se zabráňuje pouze tak, že vysílací stanice oznámí ostatním, že bude vysílat a tím si rezervuje prostor pro odeslání datového rámce. Tato metoda je schopná zabránit kolizím, protože ostatní účastníci vědí, že dojde k vysílání informace ještě před jejím začátkem. Bohužel, tato nezabrání případným kolizím se 100% jistotou. Proto se používá hlavně u polo-duplexního přenosu, kdy stanice vysílá nebo přijímá ne, však současně.

[4]

### 2.4.2 Referenční síťový model

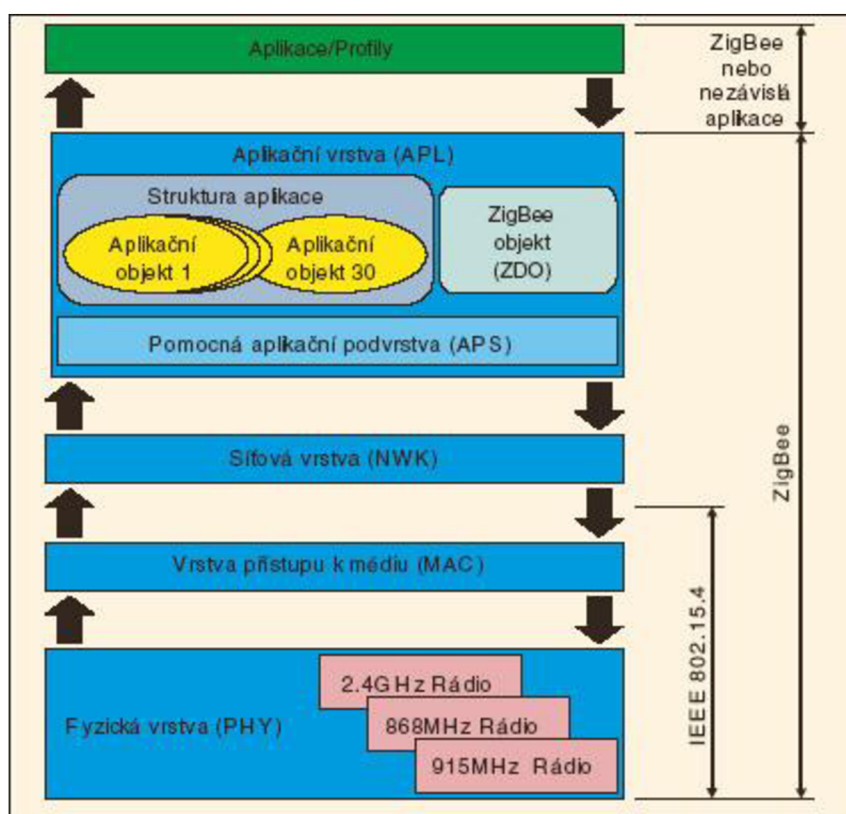
Referenční síťový model ZigBee vychází z klasického sedmivrstvového modelu ISO/OSI. Používá ale pouze vrstvy, které jsou důležité v konkrétní oblasti použití. Standard IEEE 802.15.4 definuje pouze dvě nejnižší vrstvy, fyzickou a vrstvu MAC. Vyšší vrstvy již definuje přímo ZigBee Alliance tyto vrstvy jsou síťová a aplikační.

Aplikační vrstva (APL) se skládá z pomocné aplikační podvrstvy (APS), z objektů ZigBee (ZDO) a z aplikačních objektů definovaných výrobcem.

Pomocná aplikační podvrstva má za úkol udržovat vazební tabulky, které umožňují propojení dvou zařízení na základě jejich potřeb. Také přeposílá zprávy mezi vzájemně vázanými zařízeními.

Objekt ZigBee – definuje, zda zařízení bude koordinátorem nebo koncovým zařízením, zařizuje zavádění spojení nebo odpovídá na požadavky na navázání spojení. Také zajišťuje hledání zařízení v síti a zjišťuje jimi poskytované služby. [5]

Síťová vrstva (NWK) zajišťuje zabezpečení rámců a jejich směrování k cílovým uzlům. Hledá přímé (one-hop) sousední uzly a ukládá o nich informace. Tato vrstva se vykytuje však pouze u plně funkčních zařízení, což bude rozvedeno u topologie sítě. [5]



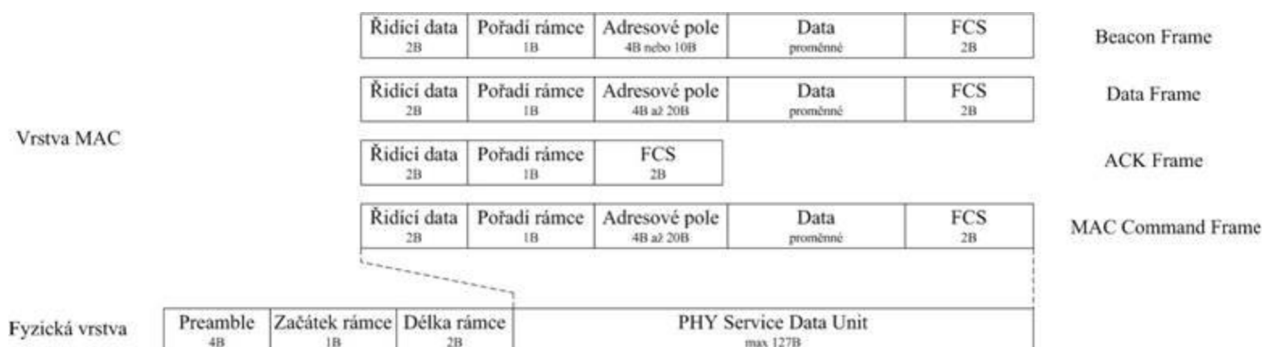
Obr. 4 Referenční model ZigBee [6]

Linková vrstva MAC – tuto vrstvu již definuje standard IEEE 802.15.4. Úkolem této vrstvy je synchronizace, zajištění přístupu na radiový kanál. Ověřuje platnost rámce, řídí spojení, generaci a rozpoznání adres. [5]

Definovány jsou čtyři typy komunikačních rámců, které jsou používané pro přenos informací užitečných datových informací nebo pro režijní účely, správu a řízení sítě. [3]

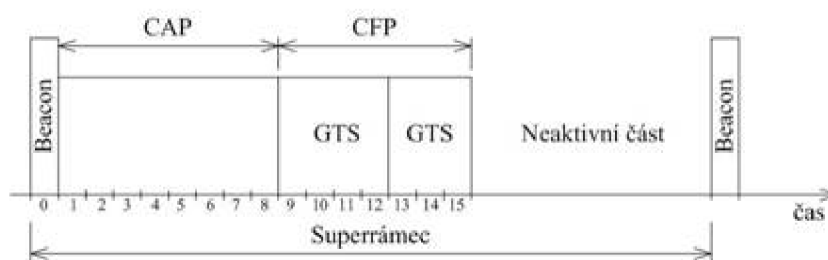
- **Data Frame** – rámeček pro přenos užitečné informace pro všechny datové přenosy má délku až 104 bytů užitečných dat.
- **Acknowledgement Frame** – tento rámeček slouží k potvrzování informace o úspěšně přijatém rámečku. Je odeslán ihned po přijetí rámečku v čase mezi rámečky.
- **MAC Command Frame** – slouží k centralizovanému konfigurování, nastavování a řízení klientských zařízení v ZigBee síti.
- **Beacon Frame** – slouží k synchronizaci zařízení v síti, je využíván především s povoleným režimem spánku koncových zařízení, které mají v takovém režimu extrémně malou spotřebu. Využívá je koordinátor. [3],[5]

Na obrázku 5 jsou zobrazeny struktury jednotlivých rámců používaných v MAC vrstvě. Maximální délka je 127 bytů.



Obr. 5 Struktury rámců MAC vrstvy [5]

Ve standardu IEEE 802.14.5 je možné použít superrámec. Jeho strukturu definuje koordinátor sítě. Tento superrámec je ohraničený beacon rámcem a dělí se na 16 stejných slotů. V prvním slotu je vyslán beacon rámec, ten je určen pro synchronizaci, identifikaci sítě PAN a popisuje strukturu superrámce. Ve zbývajícím čase může jakékoli zařízení na základě výše popsané přístupové metody CSMA-CA komunikovat. Každý superrámec může mít aktivní a neaktivní část. Během aktivní části koordinátor komunikuje s příslušnou PAN a v neaktivní části může přejít do režimu spánku (*low-power mode*). Aktivní část lze dále rozdělit na oblast CAP (*Contention Access Period*) a CFP (*Contention Free Period*). V době trvání CAP probíhá komunikace na základě CSMA-CA. Část CFP je složena z několika GTS (*Guaranteed Time Slot*), které jsou vyhrazeny pro pomalá (*low-latency*) a prioritní zařízení. Tento popis je znázorněn na následujícím obrázku. [5]



Obr. 6 Příklad struktury superrámce [5]

Nejnižší vrstvou, OSI modelu definovanou standardem IEEE 802.15.4 je Fyzická vrstva. Tato vrstva zajišťuje samotnou radiovou komunikaci. Její popis je uveden v kapitole specifikace radiové části. Nejvíce se využívá z uvedených kmitočtů frekvence 2,4 GHz, díky svým vlastnostem množství kanálů.

Výhodou je že kanál 11 až 26 má vzdálenost 5 MHz a jednotlivé kanály se nepřekrývají, což velmi redukuje rušení.

Velkým bonusem návrhu IEEE 802.15.4 je malá bitová chybovost u zařízení které mají velký šum.

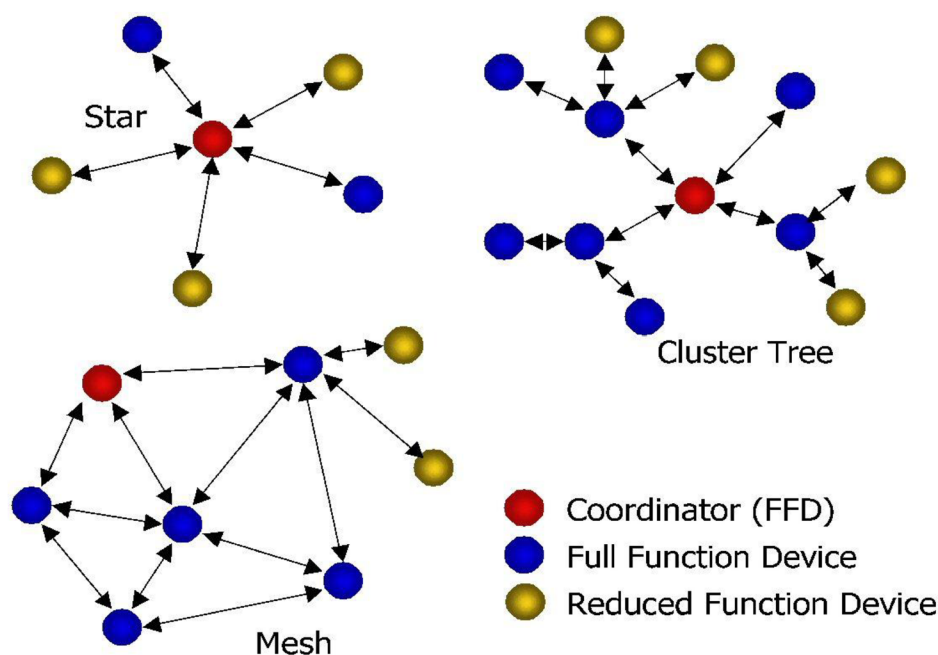
### 2.4.3 Topologie sítě

Adresace jednotlivých zařízení probíhá buď pomocí dlouhého adresovacího kódu (64 bit) nebo pomocí zkráceného kódu (16 bit) viz tabulka 3. Lokální zkrácená adresa adresovacího kódu umožňuje v síti adresovat maximálně 65 535 zařízení. Každá síť je jednoznačně určena pomocí 16 bitového identifikátoru PAN ID, ten se používá v případě, kdy je v blízkosti provozováno více sítí standardu IEEE 802.15.4. [3]

PAN ID zdroje 0/2 byte	Adresa zdroje 0/2/8 byte	PAN ID cíle 0/2 byte	Adresa cíle 0/2/8 byte
Adresové pole			

Tab. 3 Adresové pole MAC rámce [5]

Síť se může skládat ze tří způsobů síťové topologie. Jak znázorňuje obrázek 7 je možné propojení typu hvězda (star), strom (cluster tree) a síť (mesh) ta je kombinací předchozích dvou variant.



Obr. 7 Možné topologie systému ZigBee [7]

Jednotlivé prvky sítě jsou složeny z plně funkčních zařízení FFD (full function device) a prvky s omezenou funkcí RFD (reduced function device). FFD může zajišťovat funkci koordinátoru, opakovače nebo koncového zařízení. V případech kdy zařízení slouží jako koordinátor nebo opakovač pro prodloužení dosahu sítě je vhodné použít síťové napájení a ne bateriové, aby v případě vybití baterie nedošlo k rozpadu celé sítě, případně na daném prvku závislé části sítě. Koordinátor je zařízení, které řídí celou síť. Zařízení s redukovanou funkcí nemá schopnost řídit síť, pouze může zasílat nějaká data ze senzorů, obvykle bývá napájen bateriově.

V topologii hvězda komunikují všechna zařízení přímo s koordinátorem, v tomto režimu se všechna zařízení chovají jako koncová.

V topologii strom mohou zařízení komunikovat přímo s koordinátorem stejně jako u topologie hvězda, navíc však mohou být využity plně funkční zařízení jako opakovače. Síť může být tedy podstatně složitější a rozsáhlejší, nevýhodou je, že v případě výpadku takového opakovače přestane fungovat na něm závislá část sítě.

Topologie síť proti typu strom umožňuje navíc přímou komunikaci (peer-to-peer) [5]

Síť lze dále rozšířit, podle toho zda používají beacon rámeček nebo ne. Potom se dělí na síť „beacon-enabled“ a „non-beacon“.

V první uvedené „beacon-enabled“ koordinátor vysílá pravidelně signál, ten koncová zařízení využívají k připojení se k síti, synchronizaci pro přenos dat. V tomto případě je tedy k přenosu použit již výše popsáný superrámeček.

Ve druhé jmenované síti „non-beacon“ je z koordinátoru také vysílán periodicky signál, ten však slouží pouze k jeho vlastní identifikaci a k detekci koncovým zařízením. Koncová zařízení tak komunikují s koordinátorem pomocí požadavků k vysílání dat a potvrzovacích rámců. Musí však být neustále připraveny podporovat komunikaci s rovnocennými uzly.

#### 2.4.4 Vytváření nové sítě

Novou síť může založit pouze plně funkční zařízení FFD, které se stane koordinátorem. Ve chvíli vytváření nesmí být členem nějaké jiné sítě. Před vytvořením sítě podvrstva MAC změří energie jednotlivých kanálů a vytvoří seznam vhodných kanálů. Následně je vybrán kanál, který je zcela volný nebo na něm je nejmenší počet sítí. Tím se zabrání nebo omezí možnost interferencí. Dané nově vytvořené síti se přiřadí identifikátor (PAN ID), takový, aby nebyl shodný s jiným v dosahu, a vytvoří se 16 bitová síťová adresa.

Po vykonání uvedených procesů je možné, aby se další prvky připojily do vytvořené sítě. O připojení zařízení rozhoduje koordinátor nebo opakovač. [8]

#### 2.4.5 Zabezpečení komunikace

Při přenosu může docházet k chybám. K odhalení těchto chyb je využíváno cyklického kódu (CRC nebo FCS), kdy je každý rámeček doplněn o zbytek dělení polynomem. V ZigBee standardu je použit tvar polynomu (1) [5]

$$G^{16} = x^{16} + x^{12} + x^5 + 1 \quad (1)$$

Rámeček je zabezpečovaný celý a to včetně jeho záhlaví.

Data jsou pro zvýšení spolehlivosti přenosu zabezpečena technologií DSSS která je popsána v kapitole specifikace radiové části.

Dále lze data chránit i proti zneužití. Standardem jsou definovány tři režimy zabezpečení. Jedná se o nezabezpečený režim, přístup na základě práv a zabezpečený přístup. V režimu omezení přístupu jsou koordinátorem odmítány rámce od neznámých zařízení. Zabezpečený režim umožňuje použití služeb:

- šifrování dat pomocí AES 128 bit
- použití MIC (*Message Integrity Code*)
- odmítnutí opakujících se rámců (*Sequential freshness*) [5]

Šifrování metodou AES nahrazuje starší DES, jeho výhodou je, že nehrozí útok způsobem vyzkoušení všech možných klíčů tzv. hrubou silou.

MIC je kryptografický kontrolní součet, je zahrnut do vysílaného rámce. V přijímači se provede inverzní operace a ověřuje se hodnota kontrolního součtu. Pokud hodnota neodpovídá kontrolnímu součtu, celý rámec je odmítnutý.

#### **2.4.6 Synchronizace zařízení ZigBee**

Synchronizace jednotlivých prvků v síti, přesněji řečeno koncových prvků s koordinátorem je realizována pomocí beacon rámců. Koordinátor jako synchronizační autorita vysílá v daných okamžicích synchronizační sekvence – beacon. Tím se s koordinátorem synchronizují ostatní zařízení v síti. Tento postup umožňuje koncová zařízení uspat i na dlouhou předem definovanou dobu a tím velmi snížit jejich spotřebu a tím zvýšit životnost baterií. Doba vysílání synchronizačního impulsu může být přibližně 15 ms až 15 minut. Je možné využít super-rámce, ten začíná právě sekvencí beacon pak následuje sekvence CFP, kdy jednotlivé prvky soutěží o možnost přístupu k médiu. Ten může být následován intervalem s rezervovanými časovými sloty pro prioritní přenosy GTS. Koordinátor vysílá v beacon rámci informace o tom zda jsou pro dané zařízení přístupná nějaká data, pokud ano, dojde k jejich vyžádání koncovým zařízením. Jejich přenos se provádí v rezervovaných slotech.

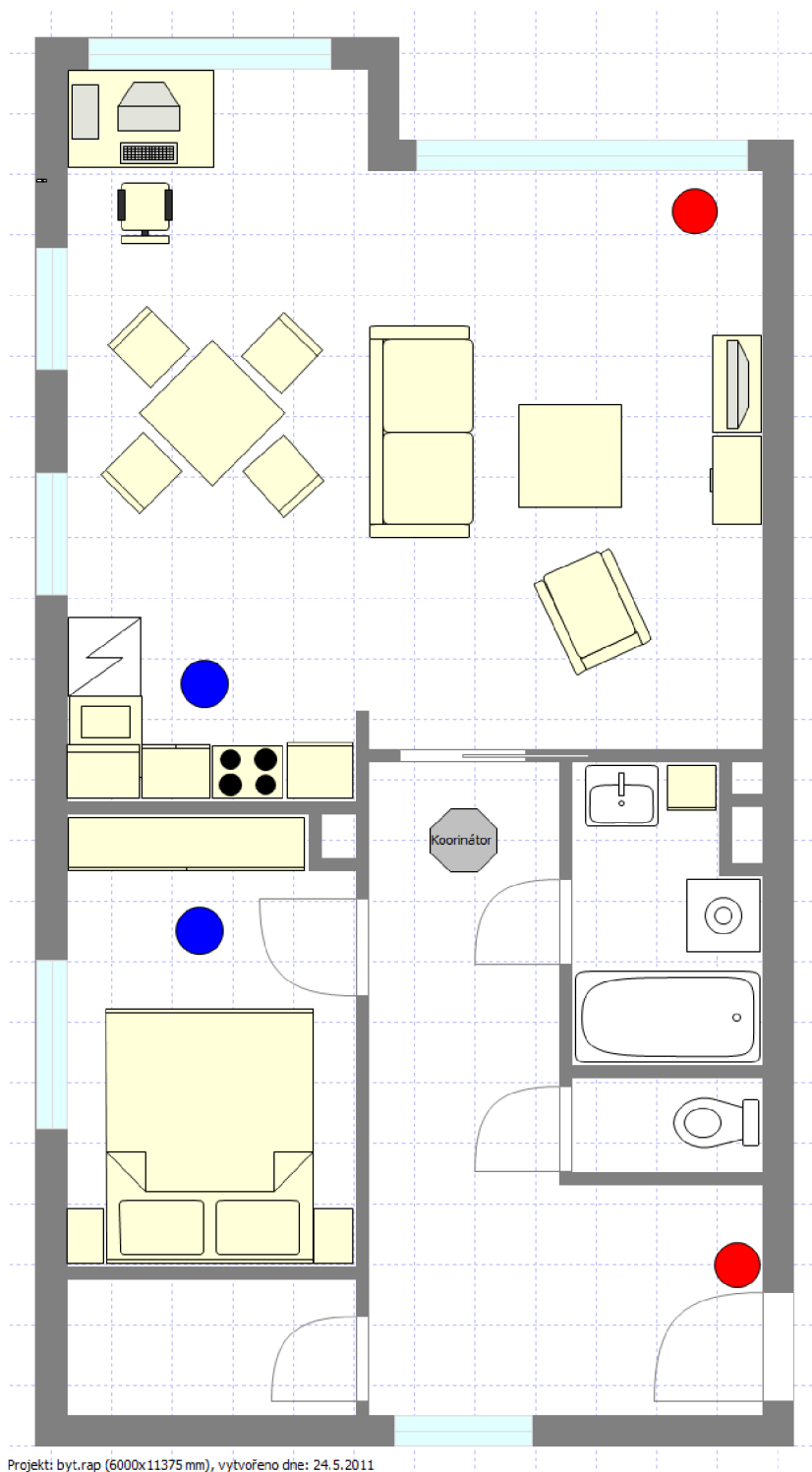
Pokud probíhá komunikace bez beacon rámců, musí se koncová zařízení periodicky dotazovat koordinátora na dostupnost dat. V takovém případě komunikace probíhá bez vyhrazených slotů. [6]

#### **2.4.7 Spotřeba zařízení**

ZigBee standard byl od počátku navrhován s požadavkem na extrémně nízkou spotřebu, naopak vzhledem k plánovanému užití nejsou nutné velké datové přenosy. Z tohoto důvodu jsou protokoly navrženy tak, aby spotřeba koncových stupňů byla co nejmenší. Koncové stupně jsou obvykle napájeny z baterií, narozdíl od koordinátoru, který je zvykem napájet ze sítě, protože na něm závisí funkčnost celé sítě.

Například při použití nejjednodušší topologie hvězda a využití beacon rámců se koncové zařízení aktivuje po přijetí beacon sekvence odešle svá data. Příjemce – koordinátor data přijme a uloží do paměti. Při přijetí další sekvence beacon koordinátor indikuje cílovému zařízení, že pro něj má data. Až si koncové zařízení data vyžádá, jsou mu koordinátorem vyslána. Díky tomuto způsobu je možné mít koncová zařízení většinu času přepnuta do režimu spánku, kde mají téměř nulovou spotřebu. Na koordinátor jsou kladeny největší nároky, protože musí obsloužit všechny zařízení v síti a ukládat jejich data do paměti. [6]

### 3 Návrh bezdrátové sítě



Obr. 8 Návrh rozmístění několika senzorů v bytě



Vysvětlivky:

- červená tečka označuje senzor otevření dveří
- modrá tečka označuje senzor teploty
- šedý osmiúhelník označuje koordinátor

Koordinátor je LAN kabelem připojen k počítači a napájen ze sítě.

Senzory otevřených dveří jsou použity pouze dva u vchodových dveří a balkonových dveří. Z důvodů lokace bytu ve třetím nadzemním podlaží není nutno hlídat otevřená okna.

Teplotní senzory jsou umístěny v ložnici například pro hlídání optimální teploty ke spánku. Druhý teplotní senzor, který je umístěn v hlavní obytné místnosti, je záměrně blízko kuchyňské linky. Lze jej tak použít i jako senzor požáru, na který je v bytě kuchyně asi nejvíce náchylná (zapomenutý zapnutý vařič). Čidlo teploměru DS 1631 není sice primárně určeno pro detekci požáru lze však programově nastavit, že nějaká vysoká teplota, která není běžná v obytných místnostech způsobí poplach.

Koordinátor je záměrně umístěn uprostřed bytu ve stropním podhledu, aby při použité topologii hvězda měly všechny senzory dobrý signál. Jak je popsáno dále v textu, využívá se zde výhody XPortu tak, že koordinátor nemusí být umístěn přímo u počítače. V tomto rozložení, se neočekávají problémy s dosahem signálu ke všem sensorům a lze použít základní topologii hvězda a koncová zařízení s omezenou funkčností.

### 3.1 ZigBit modul modul ZDM-A1281-A2

ZigBit modul vyrábí firma Meshnetics jedná se o nízkonapěťový, vysoce citlivý modul standartu IEEE802.15.4 s obchodním jménem ZigBee. Je určen pro bezdrátové snímání, ovládání a získávání dat aplikací. Jedná se o velmi kompaktní modul, pro názornost na obr 8. vedle mince.

#### 3.1.1 Klíčové vlastnosti:

- kompaktní rozměry 24 x 13,5 mm
- inovativní patentovaná anténa se ziskem až 0 dBi
- velmi nízká spotřeba energie (<6uA v režimu spánku)
- dostatek paměti (128 bajtů FLASH, 8 kB RAM, 4 kB EEPROM)
- vysoká citlivost RX (-101 dB)
- široké spektrum dostupných rozhraní
- 9 vstupně výstupních pinů (GPIO)
  - JTAG
  - UART
  - I2C
  - SPI
- pracovní pásmo 2,4 Ghz

- rozsah pracovních teplot  $-20^{\circ}\text{C}$  až  $70^{\circ}\text{C}$
- relativní vlhkost by neměla překročit 80 %

Tento modul obsahuje integrovaný mikroprocesor Atmega1281V, není tedy nutné pro základní aplikace přidávat další pomocné procesory. Tato vlastnost ve spojení s GPIO piny umožňuje napojení sezrožů přímo na tento modul, tím odpadá nutnost používat další převodníky či procesory.

Modul lze dále ovládat a nastavovat také dále pomocí AT příkazů, což zjednodušuje ladění a testování sítě.

### 3.1.2 Základní specifikace

- napájecí napětí 1,8 až 3,6 V
- spotřeba proudu RX režim 19 mA
- spotřeba proudu TX režim 18 mA
- spotřeba proudu při vypnuté radiové části procesor aktivní na 50% 14 mA
- spotřeba proudu v usporném režimu pouhých 6  $\mu\text{A}$

Všechny piny jsou orientované proti zemi, pouze s výjimkou RESETu který je negovaný.



Obr. 9 Porovnání velikosti ZigBit™ modulu s mincí

## 3.2 Digitální teploměr DS1631

Tuto součástku vyrábí společnost DALLAS Semiconductor, jedná se o jeden z nejpoužívanějších a nejoblíbenějších součástek svého typu. Oblíbený byl i v jeho starších variantách. Funkce tohoto teploměru jsou kromě přenosu hodnoty teploty v digitální podobě je i spínací funkce v závislosti na nastavené teplotě.

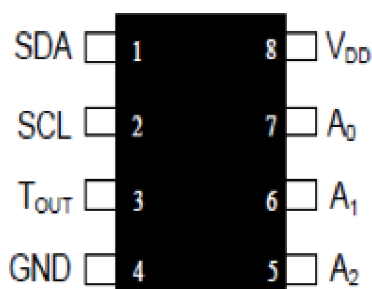
Vzhledem k pokročilým technologiím společnosti DALLAS Semiconductor není nutná žádná další korekce změřené teploty a jiné obslužné součástky.

Spínací hodnoty a komunikace s teploměrem probíhá pomocí dvou vodičové SDI sériové sběrnice.

Součástka je dostupná jak v klasické podobě tak i v SMD variantě pro povrchovou montáž. Varianta SMD je zachycena na obrázku 11 v porovnání s mincí.

### 3.2.1 Základní vlastnosti:

- Teplotní rozsah  $-55^{\circ}\text{C}$  až  $+125^{\circ}\text{C}$
- přesnost měření při teplotách  $0^{\circ}\text{C}$  až  $70^{\circ}\text{C}$  je  $\pm 0,5^{\circ}\text{C}$
- napájecí napětí v rozsahu 2,7 V a 5,5 V
- Přenos dat po dvoudrátové sběrnici
- vestavěný termostat



Obr. 10 Rozložení pinů DS1631 [9]

Popis jednotlivých pinů:

- SDA – datový vstup a výstup pro dvou vodičovou komunikaci – komunikační port
- SCL – hodinový pin pro dvou vodičovou komunikaci
- GND – Zemnicí pin
- A<sub>1</sub> – adresní vstupní pin
- A<sub>2</sub> – adresní vstupní pin
- A<sub>3</sub> – adresní vstupní pin
- V<sub>DD</sub> – Napájecí pin pro kladné napětí v rozsahu 2,7 V až 5,5 V

Teploměr DS1631 poskytuje teplotu v digitálním formátu reprezentovaným v 9, 10, 11 nebo 12 – ti bitovém čísle, které je uživatelsky volitelné. Teplotu je možné získávat ve °F tak i °C – které použijeme my v evropských podmínkách. Komunikace mezi procesorem a teploměrem probíhá podle předepsaných pravidel po sběrnici TWI.

Ukázka 12 – ti bitové komunikace je v následující tabulce. Tabulka zobrazuje příklady určitých teplot reprezentovaných v 12 – ti bitovém čísle a to ve formátu binárním a hexadecimálním.

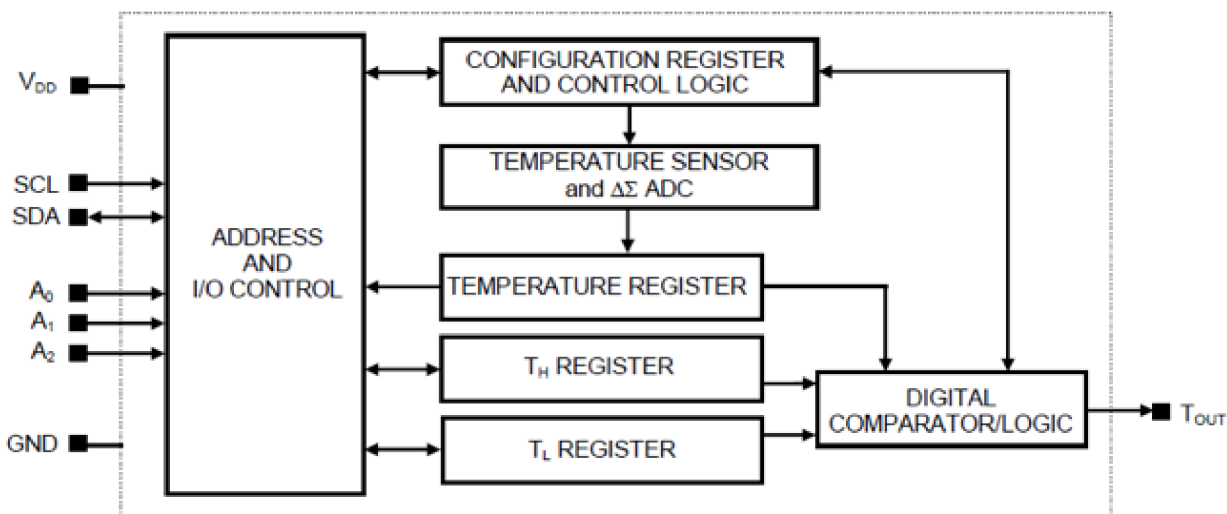
TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0111 1101 0000 0000	7D00h
+25.0625	0001 1001 0001 0000	1910h
+10.125	0000 1010 0010 0000	0A20h
+0.5	0000 0000 1000 0000	0080h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1000 0000	FF80h
-10.125	1111 0101 1110 0000	F5E0h
-25.0625	1110 0110 1111 0000	E6F0h
-55	1100 1001 0000 0000	C900h

Tab. 4 Příklady zobrazených teplot [9]

### 3.2.2 Funkční diagram

Bloky ze kterých se skládá.:

- Adresování a ovládání vstupně výstupních pinů
- konfigurační registr a kontrolní logika
- teplotní senzor
- teplotní registr
- Th registr
- Tl registr
- digitální komparátor



Obr. 11 Blokový diagram teploměru DS1631 [9]

### 3.2.3 Funkce termostatu:

Termostat je realizován výstupem T<sub>out</sub>. Je aktualizován při každém převodu teploty, na základě srovnání naměřené teploty s teplotou která byla uživatelsky definována. Jestliže je teplota vyšší nebo stejná, jako uživatelem nastavená hodnota, zapíše se do registru T<sub>H</sub>. Výstup T<sub>out</sub> se stane aktivní a aktivní zůstává až do doby než naměřená hodnota klesne na hodnotu uloženou v registru T<sub>L</sub>. Tento rozsah umožňuje nastavit určitou míru hysterze na výstupu termostatu.

Funkce termostatu teploměru DS1631 nebude u této práce využita, proto si vystačíme s takto stručným popis této funkce.

### 3.2.4 Rychlost odezvy v závislosti na bitovém rozsahu

Doba odezvy je závislá na uživatelem nastaveném bitovém rozsahu, tyto hodnoty jsou uvedeny v následující tabulce.

R1	R0	RESOLUTION (BIT)	CONVERSION TIME (MAX)
0	0	9	93.75ms
0	1	10	187.5ms
1	0	11	375ms
1	1	12	750ms

Tab. 5 Rychlost odezvy v závislosti na bitovém rozsahu [9]

### 3.2.5 Dvou vodičová seriová datová sběrnice

Teploměr DS1631 komunikuje pomocí obousměrné dvou vodičové sériové sběrnice, která se skládá se sériových hodin (SCL) a signálových dat na signálním vodiči (SDA)

U dvou vodičové komunikace se používá následující terminologie:

*Master (nadřazené) zařízení* – Tímto nadřazeným zařízením je mikroprocesor se kterým teploměr komunikuje. Generuje hodinový signál SCL a určuje start a stop podmínky.

*Slave (podřazené) zařízení* – do této skupiny patří všechna zařízení, která jsou připojena na sběrnici mimo vedoucího (master) zařízení. V našem případě tedy teploměr.

*Vysílač* - je zařízení, které zrovna po sběrnici odesílá data může jít o nadřazené i podřazené zařízení

*Přijímač* - je zařízení, které zrovna přijímá data které přichází ze sběrnice zase se může jednat o nadřazené i podřazené zařízení.

*Start stav* – Procesor (master) vyšle generovaný signál označující začátek komunikace po sběrnici. Start signál se vytváří změnou na SDA vodiči z vysokého stavu (log 1) na nízký (log 0), zatímco SCL je na log 1. Někdy se takto sekvence opakuje a má se za signál stop.

*Stop stav* – procesor (master) oznámí konec vysílání přenosu dat po sběrnici tím, že změni stav na SDA sběrnici z log 0 na log 1. SCL sběrnice zůstává na log 1. Tímto je ukončena komunikace po sběrnici, ta je uvolněna a přechází do klidového stavu.

*Potvrzení (ACK)* – Pokud se zařízení chová jako přijímač, je nutné aby generovalo potvrzovací (ACK) impuls po obdržení každého bitu. Během tohoto procesu musí vysílací zařízení uvolnit SDA sběrnici.

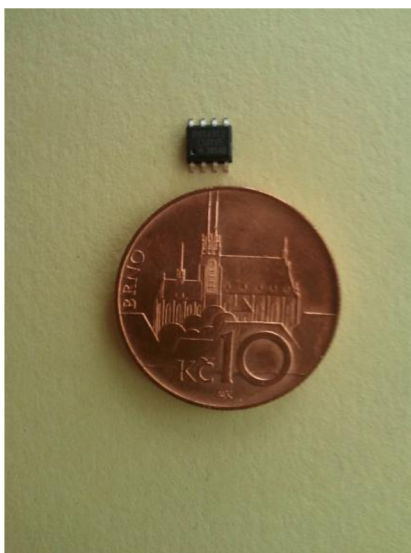
*Kontrolní bity* – kontrolní bit vysílá master a skládá se ze 7 bitové slave adresy a čtecích a zapisovacích bitů (read/write = R/W) . Jestliže master přijímá ze slave zařízení R/W = 1, pokud bude master zapisovat data na slave zařízení pak R/W = 0. Viz následující tabulka.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	0	0	1	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	R/ $\overline{W}$

Tab. 6 Tabulka kontrolních bitů [9]

*Příkazové bity* – příkazový byt může být tvořen některým z protokolů COMMAND SET tyto příkazy jsou:

- Start přenosu T [51h] – Iniciuje teplotní konverzi. Je zvolen jedno přenosový režim (1 SHOT = 1), je proveden pouze jeden přenos. V režimu kontinuálním (1 SHOT = 0), přenos teploty je prováděn průběžně, až do příchodu Stop přenosu T.
- Stop přenosu T [22h] – Zastaví se konverze teploty, pokud je přístroj v kontinuálním režimu (1 shoot = 0)
- Čtení teploty [AAh] – čte poslední konvertovanou teplotu z dvoubitového teplotního registru.
- Přístup TH [A1h] – čte nebo zapisuje do dvoubitového Th registru
- Přístup TH [A2h] – čte nebo zapisuje do dvoubitového Tl registru
- Přístup ke konfiguraci [ACh] – čte nebo zapisuje do jednobitového konfiguračního registru.
- Software (programový) POR [54H] - Zahájí program Power On Reset (POR). Tento program zastaví teplotní konverzi a obnoví všechny registry a logiku do stavu po zapnutí. Tento program umožňuje simulaci cyklu vypnutí napájení.



Obr. 12 Porovnání velikosti teploměru DS1631 s mincí

### 3.3 XPort®

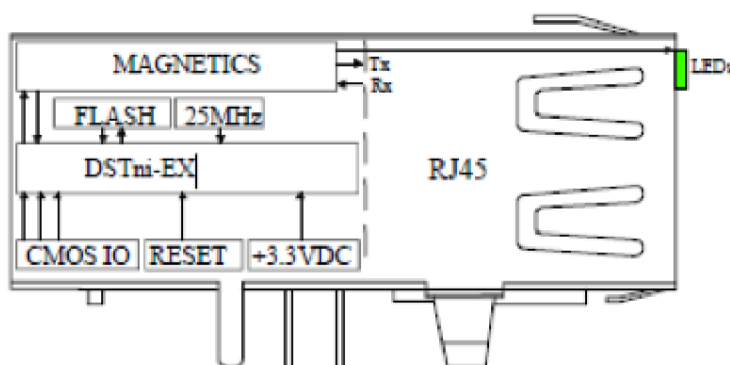
Pro komunikaci koordinátoru s počítačem je využita inteligentní a kompaktní součástka XPort®, křerý vyrábí společnost Lantronix. Využijeme tím výhody

umístit koordinátor v podstatě na libovolné místo a jeho spojení s počítačem pomocí UTP kabelu v na libovolnou vzdálenost kterou umožňují lan sítě což by v případě použití klasické sériové linky RS-232 (COM port) nebylo možné.

XPort® obsahuje integrovaný procesor k zajištění síťové nebo webové konektivity na vnitřní straně má běžné sériové rozhraní. Zjednodušeně řečeno navenek se chová jako klasický lan port a pro zařízení jako sériový port.

### 3.3.1 Charakteristika

- převod ze sériového rozhraní na ethernet
- vestavěný webový server
- snadná konfigurace přes webové rozhraní
- napájecí napětí 3,3 V
- aktualizace firmwaru XPortu přes síť
- některé verze XPortu umožňují i šifrování komunikace
- rozsah pracovních teplot  $-40^{\circ}\text{C}$  až  $+85^{\circ}\text{C}$
- automatická volba 10/100 Mb rozhraní intranet



Obr. 13 Vnitřní zapojení XPortu® [11]

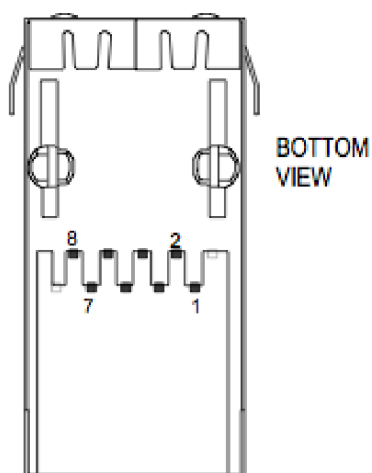
### 3.3.2 Popis sériového rozhraní

Obsahuje 8 pinů. Signály jsou v hodnotě napájecího napětí 3,3 V CMOS. Dva piny slouží jako signální, tři pro řízení toku dat, jeden pro reset, dále pak napájení 3,3V a signálová zem. Popis pinů je v následující tabulce.



Signal Name	Pin	Function
GND	1	Circuit Ground
Vcc	2	+3.3V Power In
Reset (In)	3	External Reset In
Data OUT	4	Serial Data Out
Data IN	5	Serial Data In
CP1	6	<p>CP1 can be configured as follows:</p> <p><b>Flow control:</b> RTS (Request to Send) <b>output</b> driven by DSTni's built-in UART for connection to CTS of attached device.</p> <p><b>Programmable input/output:</b> CP1 can be driven or read through software control, independent of serial port activity.</p>
CP2	7	<p>CP2 can be configured as follows:</p> <p><b>Modem control:</b> DTR (Data Terminal Ready) <b>output</b> driven by DSTni's built-in UART for connection to DCD of attached device.</p> <p><b>Programmable input/output:</b> CP2 can be driven or read through software control, independent of serial port activity.</p>
CP3	8	<p>CP3 can be configured as follows:</p> <p><b>Flow control:</b> CTS (Clear to Send) <b>input</b> read by DSTni's built-in UART for connection to RTS of attached device.</p> <p><b>Modem control:</b> DCD (Data Carrier Detect) <b>input</b> read by DSTni's built-in UART for connection to DTR of attached device.</p> <p><b>Programmable input/output:</b> CP3 can be driven or read through software control, independent of serial port activity.</p>

Tab. 7 Popis pinů na sériové straně rozhraní XPortu [11]



Obr. 14 Zapojení pinů XPortu při pohledu zespod [11]

### 3.3.3 Popis rozhraní ethernetu

Obsahuje konektor RJ45 a signalizační dvě signalizační led diody. Tyto diody svou barvou signalizují následující:

- Linková dioda (na levé straně)
  - Nesvítí – není připojena linka
  - Jantarová – připojeno k síti 10 Mbps
  - Zelená - připojeno k síti 100 Mbps
- Signalizace aktivity (na pravé straně)
  - Nesvítí – není žádná síťová aktivita
  - Jantarová – polo-duplexní přenos
  - Zelená – plně-duplexní přenos

Signal Name	DIR	Contact	Primary Function
TX+	Out	1	Transmit Data +
TX-	Out	2	Transmit Data –
RX+	In	3	Receive Data +
RX-	In	6	Receive Data –
Not Used		4	Terminated
Not Used		5	Terminated
Not Used		7	Terminated
Not Used		8	Terminated
SHIELD			Chassis Ground

Tab. 8 Popis pinů na straně rozhraní ethernetu. [11]



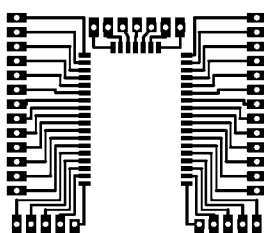
Obr. 15 Fotografie XPort®

## 4 Realizace hardwaru

Celý návrh je realizován pomocí vývojového prostředí Eagle. Jedná se návrhový program pro tvorbu schémat a plošných spojů.

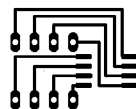
Pro realizaci jsou použity jako hlavní prvek čipy ZigBit™ společnosti Meshnetics jsou již popsány v samostatné kapitole.

Oba plošné spoje jsou navrhnutý jako vývojová deska proto jsou čipy ZigBit™ napájeny na redukci, aby při jejich případném přesunutí na jinou desku nedošlo ke zničení.



Obr. 16 Plošný spor redukce

Redukce je použita také z pouzdra DIL8 na DIP8 na které je napájen čip teplotoměru



Obr. 17 Redukce DIL8 na DIP8

### 4.1 Koordinátor

Návrh koordinátoru je proveden s ohledem na požadavky, které jsou na něj kladeny. Základní požadavky jsou:

- Síťové napájení
- Propojení s počítačem

Pro napájení je použit síťový adaptér, který se přes konektor připojuje do desky. Napájení desky požaduje hodnotu napájecího napětí na hodnotě 3,3 V (+/-) 5%, proto je použit třístupňový stabilizátor LM1084, který stabilizuje, napětí právě na hodnotu 3,3 V. Napájecí napětí je možno volit od 4,75 V až do hodnoty 27 V. Na stabilizátoru je umístěn chladič, ale z důvodů menšího zahřívání je lepší volit hodnotu napájecího napětí okolo 5 – 5,5 V, kdy je zahřívání minimální.

Pro komunikaci s počítačem je použit, také již popsáný XPort® společnosti Lantronix. Jeho výhodou je, že se pro procesor obsažený v modulu ZigBit chová

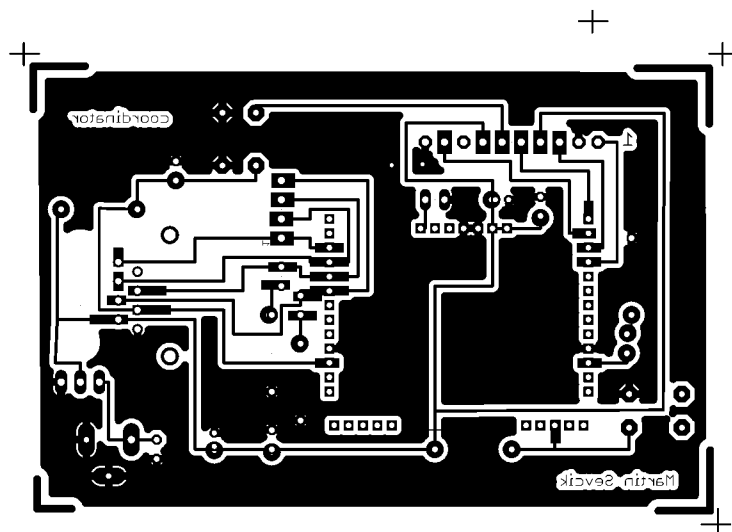
jako standardní sériový port. Naopak na straně počítače jako počítač umístěný někde v síti.

Toto řešení v sobě nese výhodu také toho, že na rozdíl od použití sériového portu, nemusí být zařízení komunikátoru umístěno v blízkosti počítače, na kterém se získaná data zpracovávají, ale může být umístěn kdekoli v dosahu sítě LAN. Takže je možno dosáhnout hodnoty řádově kolem sta metrů.

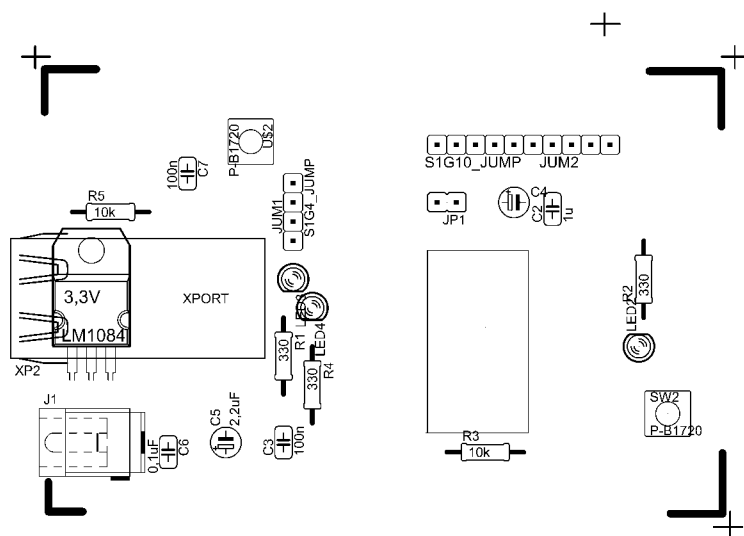
Na desce plošných spojů je dále umístěno několik signalizačních diod zelená a žlutá signalizují vysílání a přijímání dat při komunikaci procesoru s XPortem. Jedna červená dioda pro libovolný účel daný programem. Jumper a tlačítko také pro libovolný účel, který lze nastavit programem. Poslední tlačítko je RESET, které jak název napovídá, slouží ke znovunastartování zařízení v případě výpadku nebo nějakých problémů.

Pro možnost naprogramovat procesor je použito rozhraní JTAG. Pro případné rozšíření je přístupné i rozhraní UART, pomocí kterého probíhá i již zmíněná komunikace procesoru a XPortu.

Dále deska již obsahuje jen několik nutných rezistorů a kondenzátorů.



Obr. 18 Plošný spoj koordinátoru



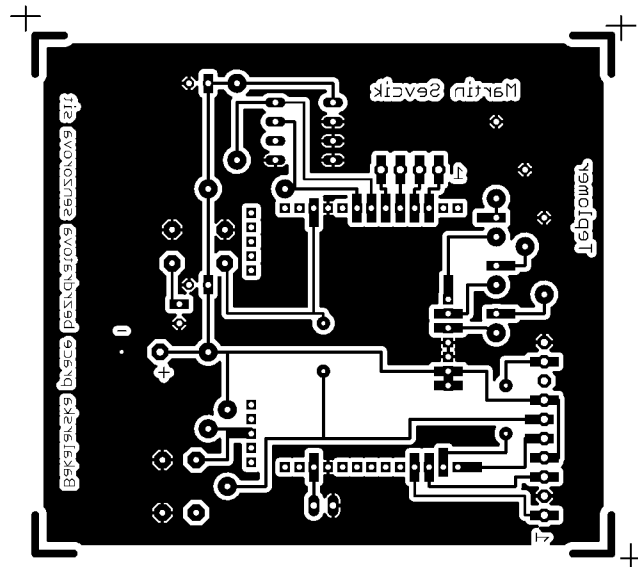
Obr. 19 Osazovací výkres koordinátoru

## 4.2 Koncové zařízení – Teploměr

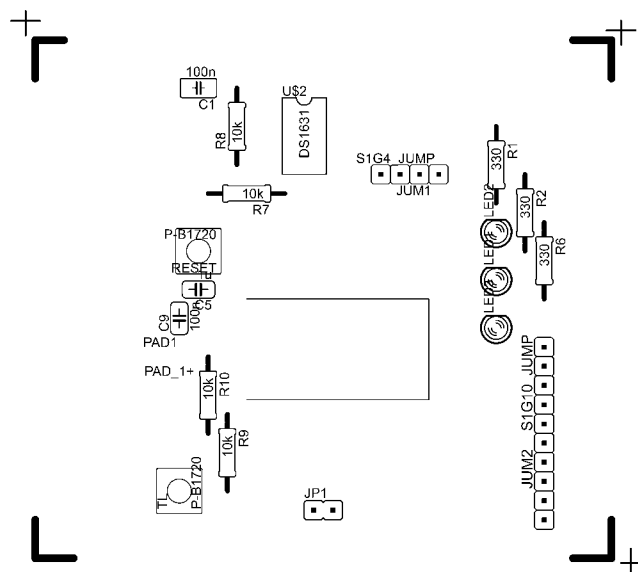
Požadavek kladený na koncové zařízení je odlišný od požadavků kladených na koordinátor. Je zde použito bateriové napájení, vzhledem k tomu, že zde součástky mají poměrně velkou toleranci na napájecí napětí, je použito napájení pomocí tří nabíjecích mikrotužkových baterií (AAA). Jedná se Ni-Mh články s napětím 1,2 V kus. Dohromady dostaneme 3,6 V. Tyto baterie lze ve finálním výrobku nahradit lithiovou specializovanou baterií o téměř napětí. Taková baterie je však finančně nákladnější, proto není u této vývojové desky použita. Velkou výhodou kterou přinese je minimální míra samovybíjení, takže její výměna v zařízení v průběhu provozu bude potřeba jen jednou za rok, možná i více. V závislosti na četnosti komunikace a měření daného zařízení. Navíc tato baterie může být malá 1/2 délky klasické tužkové baterie AA. Z toho plyne také výhoda možnosti kompaktních rozměrů celého zařízení.

Navržené koncové zařízení plní funkci teploměru. Samotné čidlo teploměru obstarává v samostatné kapitole popsaná součástka DS1631 výrobce Dallas semiconductor.

Dále deska kromě tlačítka RESET, obsahuje rozhraní UART pro možnost připojení dalšího jiného zařízení, rozhraní JTAG pro možnost programování procesoru. Tlačítko a jumper, které mohou plnit programem danou funkci. Jsou obsaženy také tři signalizační led diody barev červená, žlutá, zelená. Jejich funkce je zase jen na programovém vybavení procesoru. Samozřejmě je použito několik nutných kondenzátorů a rezistorů.



Obr. 20 Plošný spoj koncového zařízení – teploměru



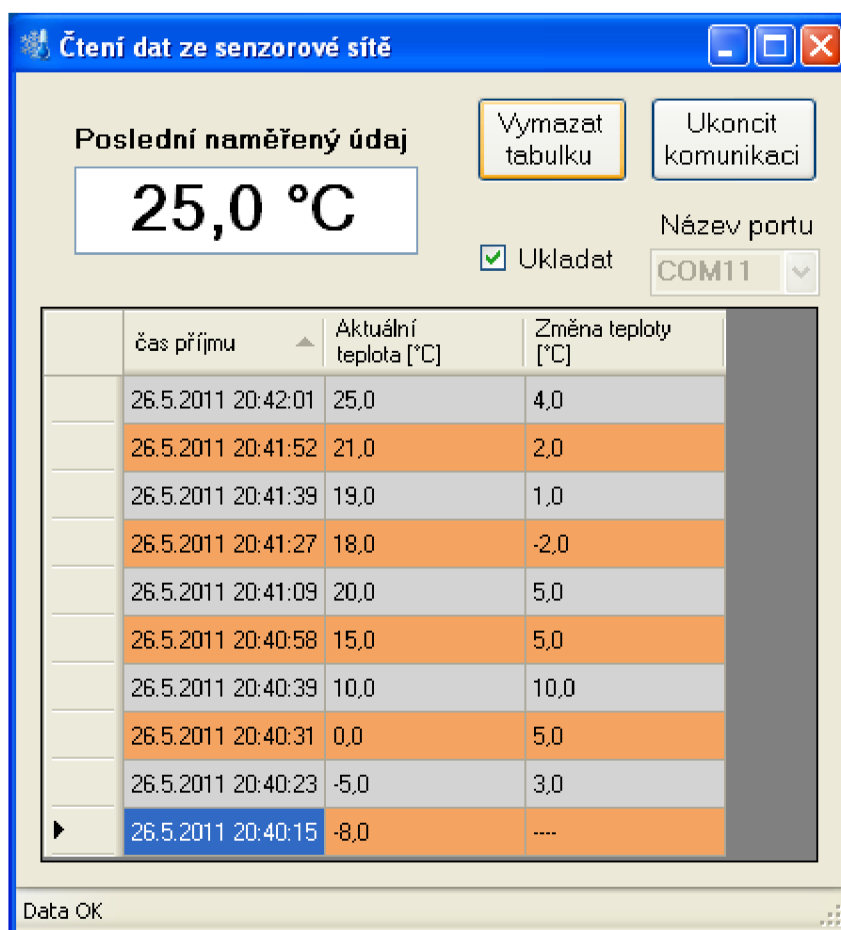
Obr. 21 Osazovací výkres desky teploměru

## 5 Softwarové vybavení

### 5.1 Zobrazení dat v počítači

Program pro zobrazení dat v počítači byl vytvořen ve vývojovém prostředí společnosti Microsoft Visual Studio 2008, které lze zdarma získat v rámci MSDN AA programu do kterého je naše fakulta zapojena. Použitý programovací jazyk C#.

Program je napsaný pro zobrazení a zpracování změřené teploty. Je vybaven instalačním souborem, který nainstaluje potřebné knihovny, aby aplikace fungovala i na starších Windows. Instalační balík vytvoří zástupce na ploše a aplikaci nainstaluje běžně do program files mezi ostatní programy. Toto lze samozřejmě změnit.



Obr. 22 Obrázek okna programu pro čtení dat ze senzorů

Software je z uživatelského hlediska navrhnutý tak, že umožňuje volbu portu COM ke kterému je připojen koordinátor, který data zasílá. Je na výběr z COM1



až COM15. Tento port je lepší volit vyšší, nižší čísla mohou být zabrána systémem a mohlo by docházet ke kolizím.

Po stisku tlačítka „Spustit komunikaci“ dojde k poslouchání na vybraném COM portu na příchozí data. Proti přijetí náhodných dat je požadovaná hodnota ošetřena start (S) a cíl (C) znakem mezi kterými je hodnota o teplotě například tvar S-10,5C.

Poslední změřená teplota se zobrazuje velkým písmem v okně aplikace, starší změřené hodnoty se ukládají do tabulky, kde je uveden čas příjmu teploty, změřená hodnota v daný čas a rozdíl proti předchozí hodnotě. Znaménko plus není zobrazeno, záporné hodnoty jsou označovány znaménkem mínus. Ve sloupci změna teploty při oteplení je hodnota bez znaménka, jestliže došlo k ochlazení, je před hodnotou znaménko mínus. Jednotlivé řádky v tabulce jsou pro lepší přehlednost barevně odlišeny.

Nad tabulkou je tlačítko „Vymazat tabulku“ které jak název vypovídá, vymaže obsah tabulky.

Jestliže je zatrhnuté zatrhávací políčko „Ukládat“ budou změřené hodnoty zapisovány do souboru s příponou CSV, který je umístěn v domovském adresáři aplikace. Tento formát lze otevřít mimo programů Poznámkový blok, TexEdit a jiné také v MS Excel, který data zobrazí v přehledné tabulce. Každé měření je odděleno hlavičku. Takto seřazená data je možné libovolně zpracovávat a uživatel si může snadno vytvářet grafy a jiné statistiky.

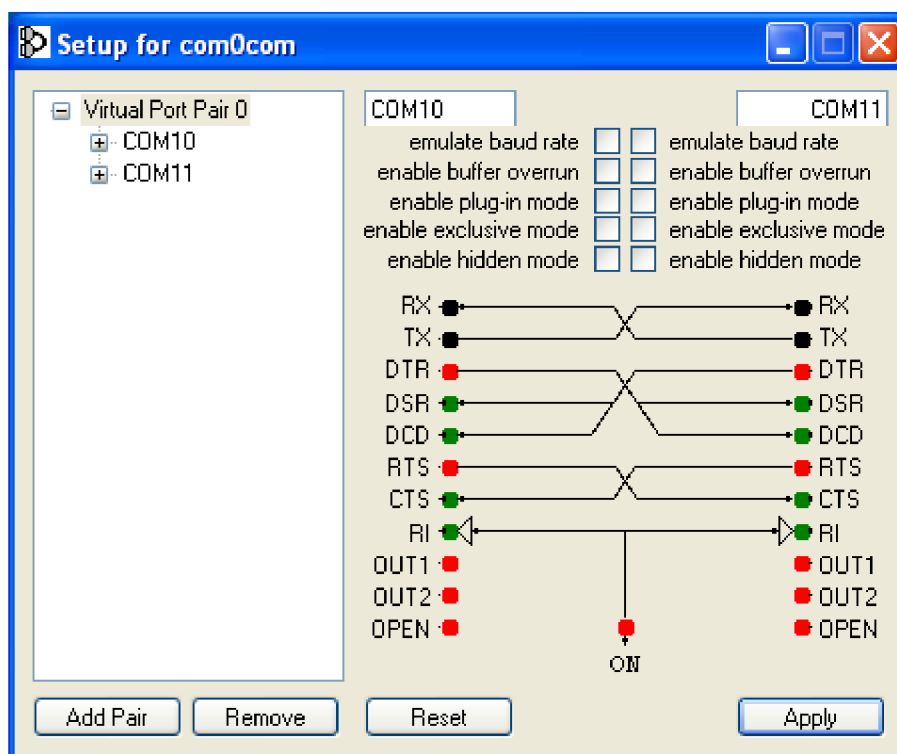
Pro ukončení komunikace slouží tlačítko „Ukončit komunikaci“ které je na stejném místě jako bylo „Spustit komunikaci“ mění význam podle toho, jakou funkci má plnit.

### 5.1.1 Testování aplikace

Tato aplikace byla vytvářena v předstihu před programováním čipů, proto ji bylo nutné testovat, k tomu bylo použito dvou pomocných programů.

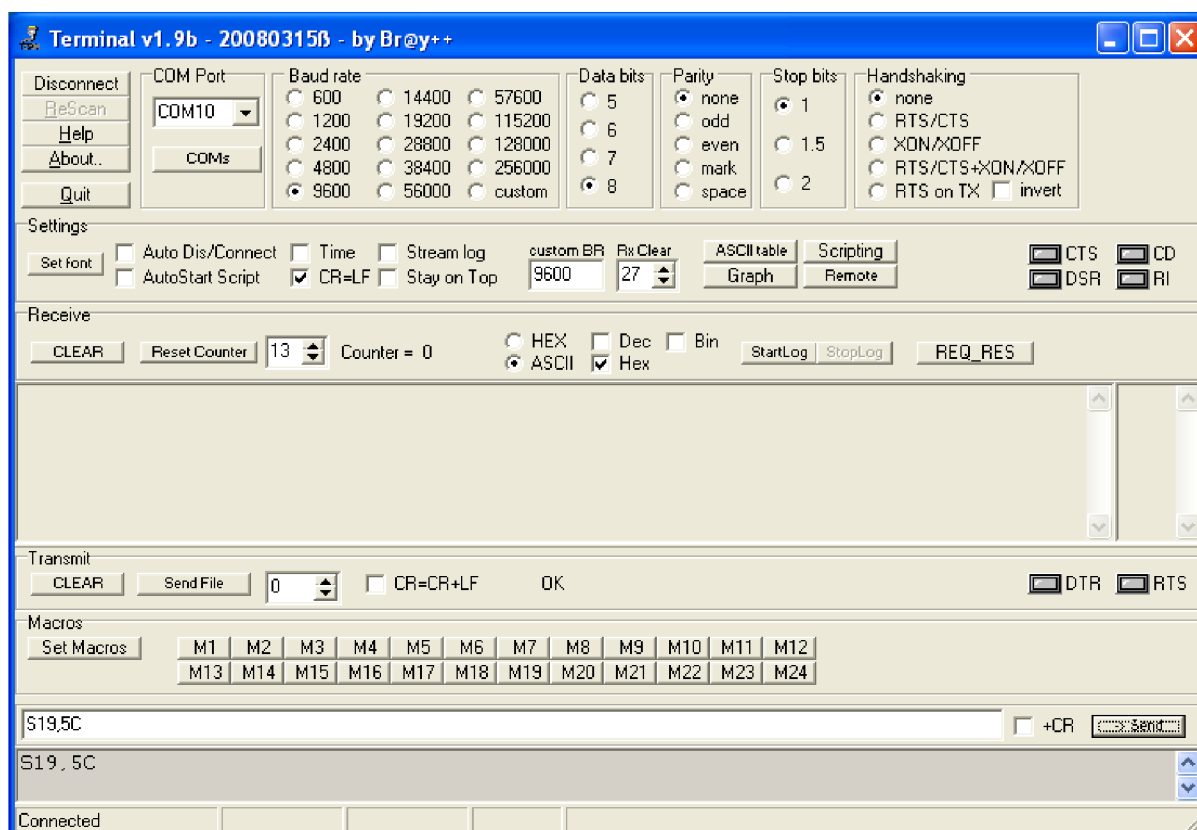
Prvním z nich je program na vytvoření virtuálních COM portů, po kterých bude aplikace komunikovat. K tomuto účelu posloužil freewarový program comocom.

Tímto programem jsou vytvořeny dva virtuální porty pro aplikaci na zobrazení dat je použit COM11, pro program hodnoty vysílající virtuální port COM10. Takto jsou zvoleny, lze volit i jinak.



Obr. 23 Obrázek z aplikace comocom

Pro zasílání dat a testování funkčnosti naprogramované aplikace je použito programu Terminal v1.9b autora Br@y. Pomocí tohoto programu lze ověřit, zda zaslouhou hodnotu, kterou bude zasílat v praxi koordinátor, přijme a správně zpracuje naprogramovaná aplikace.

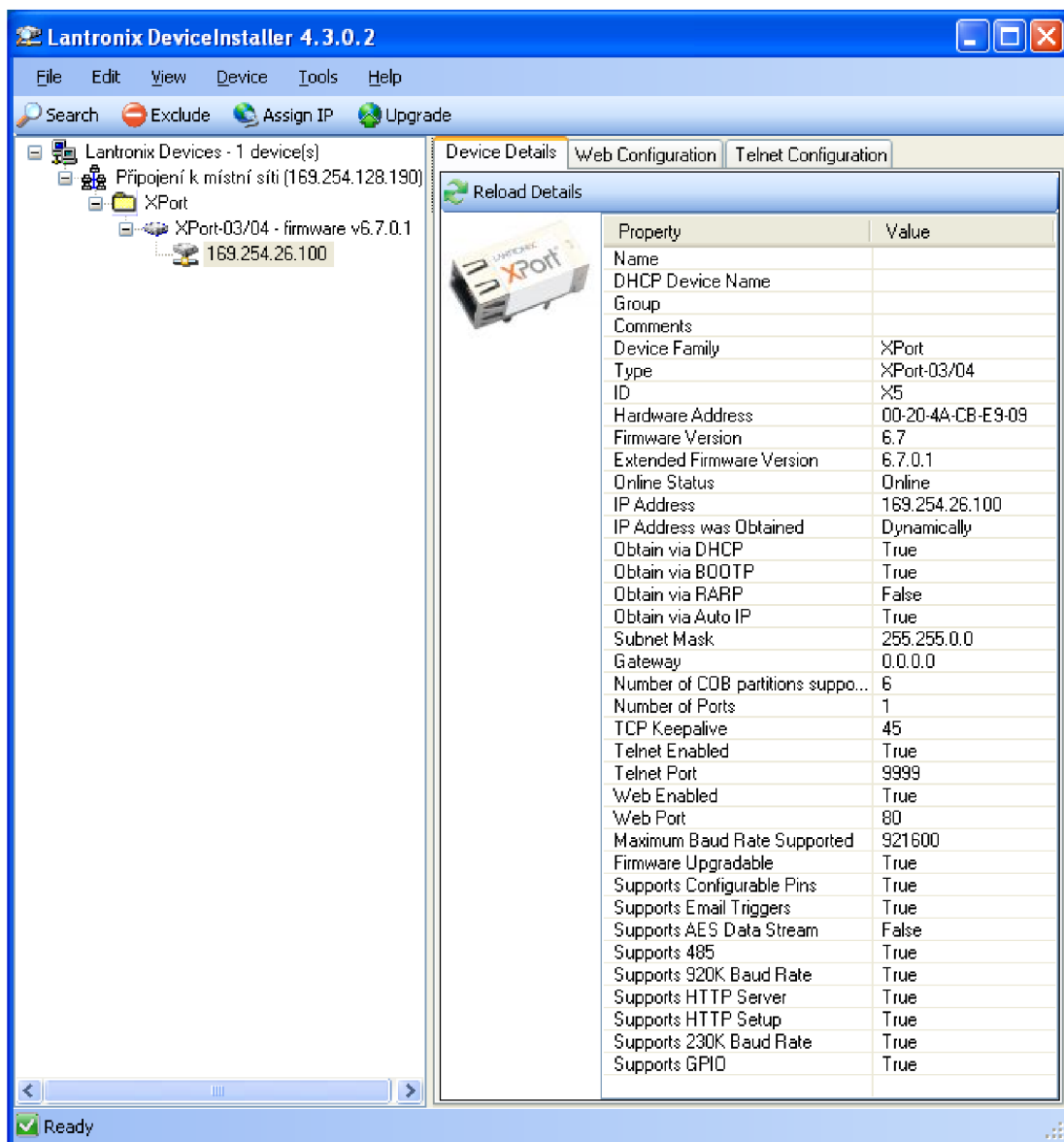


Obr. 24 Obrázek aplikace terminal

## 5.2 Program pro koordinátor

Program pro koordinátor komunikuje s procesorem pomocí Xportu, ten se tváří pro počítač jako virtuální COM port. Toto je však nutno nastavit.

Nejprve se pomocí programu Lantronix DeviceInstaller získá informace o XPortu a přidělená IP adresa.



Obr. 25 Výpis programu DeviceInstaller

Získaná IP adresa se zadá do internetového prohlížeče (Microsoft IE, Google Chrome...) a dostaneme se na webový server pro správu a nastavení x protu, kde je možné nastavit virtuální port COM pod kterým se bude hlásit v systému.

Pro komunikaci s koordinátorem po ZigBee se použije předprogramovaný open MAC pouze je třeba doplnit, aby se data dále odesílali na UART.

### 5.3 Program pro teploměr

Program pro teploměr je složitější, protože provádí více operací. Zase se využije předprogramovaného open MAC tentokrát pro koncové zařízení.

Je zde potřeba inicializovat komunikaci po I2C sběrnici s využitím knihovny TWI master. Autorem knihovny I2C Master je Peter Fleury.

## 6 Závěr

V této práci jsem se podrobně seznámil s technologií ZigBee. Prostudoval její vlastnosti a utvrdil se, že parametricky je pro tuto práci nejvhodnější.

Také jsem prostudoval možnost použití senzorů, z nichž jsem pro demonstraci použití této technologie použil senzor teploty v podobě teploměru DS1631.

Pro komunikaci s počítačem a získání změřených dat přenesených do koordinátoru jsem použil modul XPortu, který se pro tento účel ukázal jako velmi vhodný z důvodu popsaných v textu práce.

Naprogramoval jsem aplikaci na zobrazení dat v počítači, která umožňuje ukládat data do souboru pro případné pozdější zpracování.

Při vytváření programů pro procesor jsem narazil na problém s nahráním dat do procesorů pomocí rozhraní JTAG. Procesory se chovaly jako uzamknuté. Proto jejich program nebylo možné odladit ani vyzkoušet, je proto popsán pouze teoreticky.

## 7 Literatura

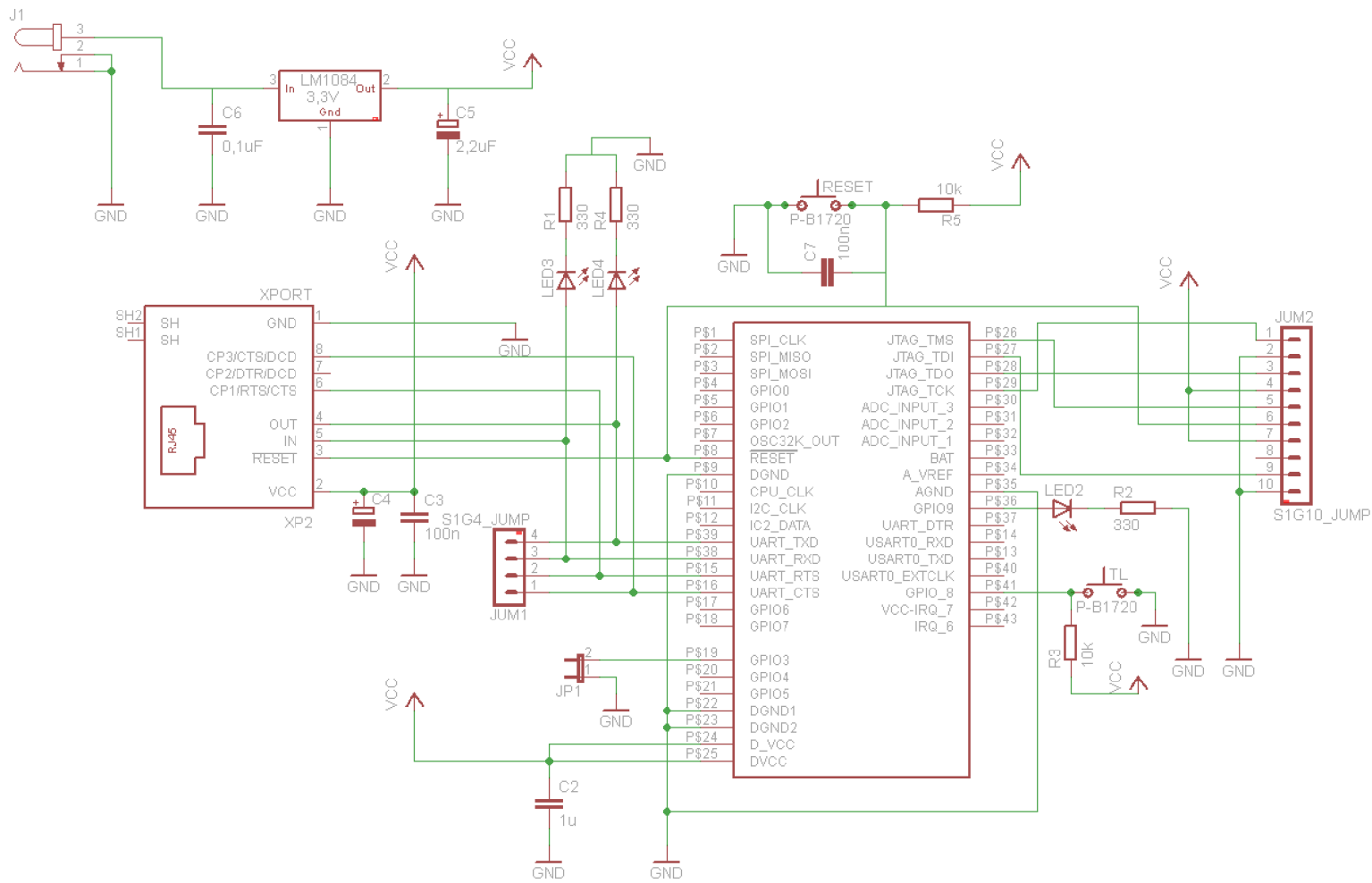
- [1] Otevřená encyklopedie wikipedie: <http://cs.wikipedia.org/wiki/Bluetooth>
- [2] Otevřená encyklopedie wikipedie <http://cs.wikipedia.org/wiki/Wi-Fi>
- [3] VOJÁČEK, ANTONÍN. *ZigBee - novinka na poli bezdrátové komunikace*. In Hw.cz [online], 8. červen 2005 [cit. 2011-04-21]. Dostupné z WWW: <http://hw.cz/Rozhrani/ART1299-ZigBee---novinka-na-poli-bezdratove-komunikace.html>
- [4] Otevřená encyklopedie wikipedie <http://cs.wikipedia.org/wiki/CSMA/CA>
- [5] *Standard nízkorychlostní bezdrátové komunikace ZigBee*. [online], 18. duben 2006 [cit. 2011-04-21]. Dostupné z WWW: <http://access.feld.cvut.cz/view.php?cisloclanku=2006032001>
- [6] BRADÁČ, ZDENĚK. *Bezdrátový komunikační standard ZigBee*. Automatizace. Duben 2005, 48, 4, s. 261-263. Dostupný také z WWW: <http://www.automatizace.cz/article.php?a=638>
- [7] KHANH, TUAN LE. *ZigBee SoCs provide cost-effective solutions*. In RFIC system architect [online]. 8. listopad 2005 [cit. 2011-04-24]. Dostupné z WWW: <http://www.eetimes.com/design/industrial-control/4012593/ZigBee-SoCs-provide-cost-effective-solutions>.
- [8] *802.15.4 & ZigBee*  
[Http://noel.feld.cvut.cz/vyu/scs/prezentace2007/MC1321x\\_ZigBee/802.15.4.htm](http://noel.feld.cvut.cz/vyu/scs/prezentace2007/MC1321x_ZigBee/802.15.4.htm)
- [9] Datasheet DS1631
- [10] Datasheet ZigBit
- [11] Datasheet XPort®
- [12] VIRIUS, MIROSLAV. *C# pro zelenáče*. Praha : Neocortex, 2002. 255 s. ISBN 80-86330-11-7
- [13] VIRIUS, MIROSLAV. *1001 tipů a triků pro C++*. Brno : Computer Press, 2011. 451 s. ISBN 978-80-251-2941-8.

[14] Webová prezentace ZigBee alliance <http://www.zigbee.org/>

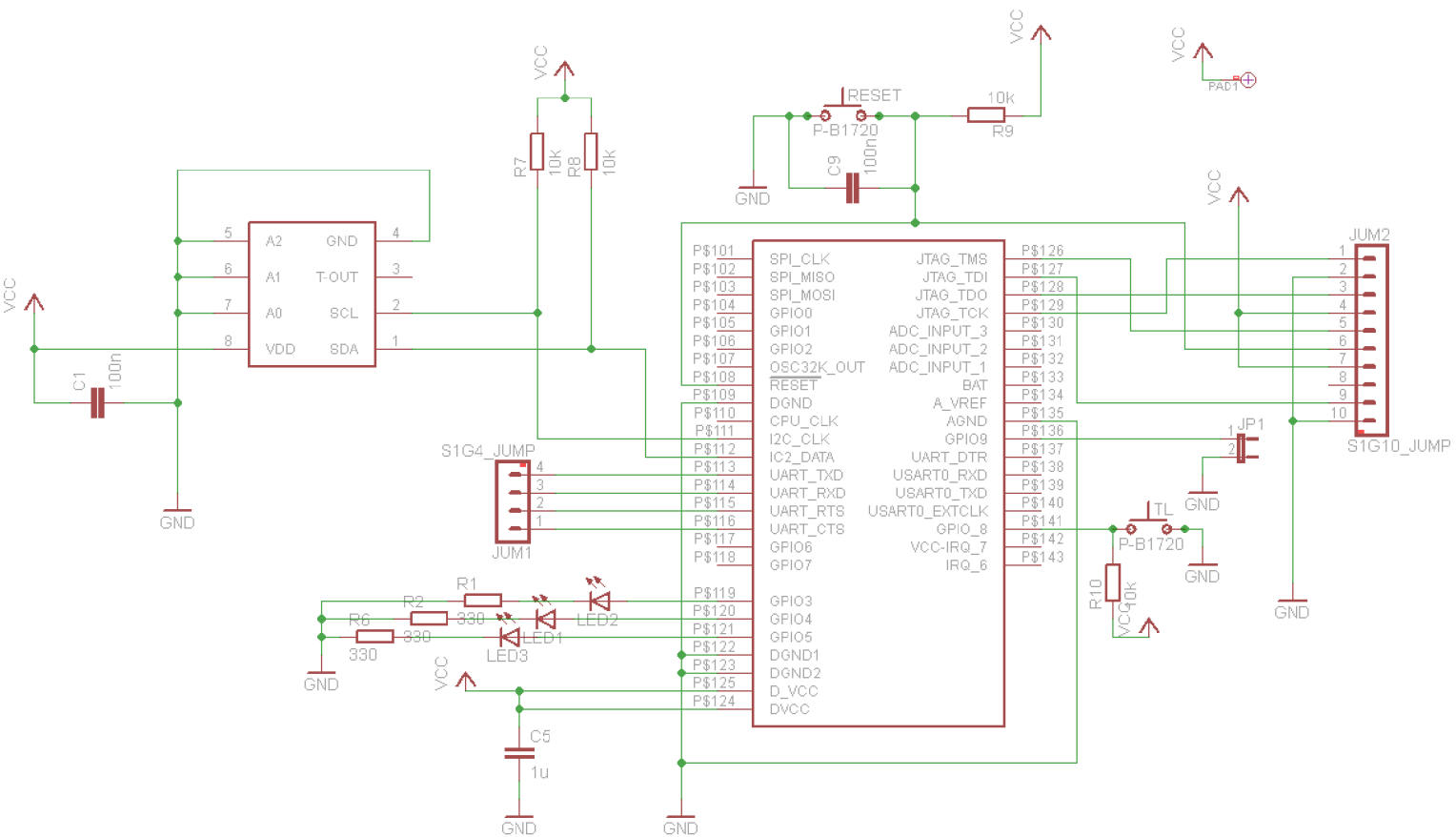


# **Přílohy**

# A Schémata



Obř. 26 Schéma koordinátoru



Obř. 27 Schéma teploměru

## B Seznam použitých zkratk

AES	Advanced Encryption Standard
AP	Access Point – přístupový bod
APL	Application Layer – aplikační vrstva prokolu ZigBee
APS	Application support sub-layer - Pomocná aplikační podvrstva
CFP	Contention Free Period - interval ve kterém volně soutěží zařízení zigbee o přístup k médiu
CRC	Cyclic Redundancy Check
COM	Sériový port, fyzický i virtuální
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance - Metoda s rozprostřeným spektrem s detekcí kolizí
CSMA/CD	Carrier Sense Multiple Access with Collision Detection - Metoda s rozprostřeným spektrem bez detekcí kolizí
CSV	Comma-separated values – hodnoty oddělené čárkami
DSSS	Direct Sequence Spread Spectrum
FFD	Full Function Device – plně funkční zařízení
GTS	Guaranteed Time Slot – interval s rezervovanými časovými sloty pro prioritní přenosy
IEEE	Institute of Electrical and Electronics - Institut elektronických a elektrotechnických inženýrů
LAN	Local Area Network – lokální síť
MAC	Medium Access Control - úroveň pro potvrzovanou komunikaci
MIC	Message Integrity Code – kryptografický kontrolní součet
MSDN AA	Microsoft Developer Network - Academic Alliance – Software spo- lečnosti Microsoft zdarma dostupný studentům zapojených universit.
O-QPSK	Offset Quadrature Phase Shift Keying - ofsetová kvadraturní fázo- vá modulace
PAN	Personal Area Network – osobní síť
PAN ID	Personal Area Network ID – identifikátor osobní sítě
RFD	Reduced Function Device – zařízení s omezenou funkčností
RS-232	Sériová linka
SSID	Service Set Identifier – servisní identifikátor sítě
UART	Universal asynchronous receiver/transmitter
WLAN	Wireless Local Area Network – bezdrátová lokální síť
ZDO	ZigBee device object -objekt ZigBee

## C Zdrojový kód PC aplikace

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.IO;
using System.Windows.Forms;

//vytvoreno: MARTIN ŠEVČÍK, UREL @ 2011

namespace SenzorickaSit
{
    public partial class cteniDat : Form
    {
        bool Snalezeno = false; // indikace startovního znaku 'S'
        char[] data = new char[5]; // přijata data
        float teplotaPom = 0; // promena pro uložení aktuální teploty
        string aktCas = "\0"; //retezec pro vložení aktuálního času
        int radku = -1, radkuSoub = -1; // pomocne promene pro určení velikosti pole a
        poradi dat v souboru
        float LastTemp = 0; // teplota minuleho mereni

        FileInfo fi = new FileInfo(Path.Combine(Environment.CurrentDirectory,
        "Data_teplomeru.csv"));

        public cteniDat()
        {
            InitializeComponent(); // inicializace formulare -- dela visual studio
        }

        // event pri stisku tlacitka spustit komunikaci
        private void KomTlac_Click(object sender, EventArgs e)
        {
            if (KomTlac.Text == "Spustit komunikaci") // znamena tlacitko
                spustit?
            {
                try // testovani komunikacniho portu
                {
                    // spravny pruchod
                    PortShift.Enabled = false; //zakazani formulare
                    KomTlac.Text = "Ukoncit komunikaci"; // prepsani tlacika na
                    druhou funkci
                    serialPort1.PortName = PortShift.Text; //zadani cisla portu

                    // dalsi vlastnosti seriove linky (nastaveno v designu):
                    // Baudrate = 38400
                    // Stopbits = 2
                    // Parity = none
                    // Event - kazdy prichodi bajty

                    if (!serialPort1.IsOpen)
                    {
                        serialPort1.Open(); //v pripade ze seriova linka neni
                        otevrena ji otevre
                        StatusProg.Text = "Komunikace zahajena"; //status
                    }
                    else
                    {
                        //jiz je otevrena --
                        StatusProg.Text = "Linka byla jiz otevrena"; //status programu
                    }
                }
            }
        }
    }
}
```

```

    }

    }
    catch // chybovy stav link
    {
        serialPort1.Close(); //zavre port
        StatusProg.Text = "Chyba komunikace - port není dostupný";
        PortShift.Enabled = true; //povoli prepínání portu
        KomTlac.Text = "Spustit komunikaci"; //tlacitko do stavu pro
start
    }
}
else
{
    serialPort1.Close(); // chceme program ukončit // zavre port
    KomTlac.Text = "Spustit komunikaci"; //tlacitko se zmeni na
startovaci
    PortShift.Enabled = true; //povoli prepínání portu
    StatusProg.Text = "Komunikace ukončena"; //status
}

// nastavení odlišného zbarvení jednotlivých radku
DataTepl.RowsDefaultCellStyle.BackColor = Color.LightGray;
DataTepl.AlternatingRowsDefaultCellStyle.BackColor = Color.SandyBrown;
}

// event při příjmu paketu
private void serialPort1_DataReceived(object sender,
System.IO.Ports.SerialDataReceivedEventArgs e)
{
    if ((serialPort1.BytesToRead > 6) && (!Snalezeno)) //je více než 6 znaků
prijato
    {
        while (serialPort1.ReadChar() != 'S') //hledání úvodního znaku "S"
        {
            if (serialPort1.BytesToRead == 0) //pokud není kde hledat stop a
vypsání hlásky
            {
                StatusProg.Text = "chyba rámce"; //status
                break; //vyskočí ze smyčky
            }
            Snalezeno = true; //nalezeno je true
        }
        if (serialPort1.BytesToRead > 5) //zbyvá víc jak 5 znaků?
        {
            serialPort1.Read(data, 0, 5); //nactení dat teploty 5 byte - znaků
            if (serialPort1.ReadChar() == 'C') //kontrola posledního znaku (zde se
i cte)
            {
                if ((data[3] == '.' || (data[3] == ',')) //desetina na třetí
pozici
                {
                    teplotaPom = (float)(data[1] - '0') * 10 + (float)(data[2] -
'0') + (float)(data[4] - '0') / 10; //převod na číslo
                    if ((data[4] == '.' || (data[4] == ',')) // čtvrtý znak je
tečka nebo čárka
                    {
                        teplotaPom = (float)(data[1] - '0') * 100 + (float)(data[2] -
'0') * 10 + (float)(data[3] - '0'); //převod na číslo
                        if (data[0] == '-') //je teplota
zaporná?? - přidá znaménko
                        {
                            teplotaPom *= -1; //nasobí se minus jedničkou
                        }
                    }
                }
                StatusProg.Text = "Data OK"; //stavový řádek
            }
        }
    }
}

```

```

        aktCas = DateTime.Now.ToString(); //aktualni cas a datum prijmu

        zapisTeplotu(teplotaPom.ToString("N1") + " °C"); //prida stupne a
zobrazí hodnotu ve formuláři (temp okno)

        if (++radku == 0) //prvni radek v tabulce
            novyRadekTab(aktCas, teplotaPom.ToString("N1"), "----", radku);
// -- funkce pro pridani radku do tabulky
        else //dalsi radky v tabulce
            novyRadekTab(aktCas, teplotaPom.ToString("N1"), (teplotaPom -
LastTemp).ToString("N1"), radku); // -- funkce pro pridani radku do tabulky

        if (ulozCheck.Checked) //je-li povolenou ulozi hodnotu
        {
            if (++radkuSoub == 0) //prvni radek v souboru??
            {
                Ulozit(DateTime.Now.Date.ToString() + "\n poradi mereni ;
datum a cas ; teplota [°C]; zmena od posledni hodnoty [°C]"); //informacni radek
                Ulozit((radku+1).ToString() + ";" + aktCas + ";" +
teplotaPom.ToString("N1") + ";" + "-----"); //funkce ulozeni - prvni radek dat
            }
            else //dalsi radky v souboru
            {
                Ulozit((radku + 1).ToString() + ";" + aktCas + ";" +
teplotaPom.ToString("N1") + ";" + (teplotaPom - LastTemp).ToString("N1")); //funkce
ulozeni - dalsi radky dat
            }
        }

        LastTemp = teplotaPom; //ulozi aktualni hodnotu teploty
    }
    else // pokud nenajde ukoncovaci znak
    {
        StatusProg.Text = "Chyba ukoncovaciho znaku"; //hlaska + data se
neukladaji ani nezobrazuji
    }
    Snalezeno = false; //prepne do hledani 'S'
}
}

//funkce pro zapis teploty do formulare
void zapisTeplotu(string data)
{
    if (TempBox.InvokeRequired) //zjistuje pristup k prvku
    {
        TempBox.Invoke(new MethodInvoker(delegate()
        {
            zapisTeplotu(data);
        }));
        return;
    }
    TempBox.Text = data; //vlastni zapis na prvek
}

//funkce pro zapis dat do tabulky
void novyRadekTab(string AktuCas, string AktHodnota, string Delta, int PocRadku)
{
    string[] row = new string[3]; //vytvoreni pole stringu (pro zapis
do tabulky)

    if (DataTepl.InvokeRequired) //zjistuje pristup k prvku
    {
        DataTepl.Invoke(new MethodInvoker(delegate()
        {
            novyRadekTab(AktuCas, AktHodnota, Delta, PocRadku);
        }));
        return;
    }
}

```

```
        row = new string[] { AktuCas, AktHodnota, Delta }; // sjednoceni dat do pole
retezcu

        if (PocRadku == 0) //prvni radek
        {
            DataTepl.Rows.Add(row); //pridani prvnioho radku
            DataTepl.AutoSizeColumn(0); //automaticke roztazeni prvnioho sloupce
        }
        else //dalsi radky
            DataTepl.Rows.Insert(0, row); //vlozi radek na prvni pozici pred
stavajici radky ktere se posunou dolu

        if (PocRadku < 0) // podminka pro mazani
            DataTepl.Rows.Clear(); //smaze vsechny radky

    }

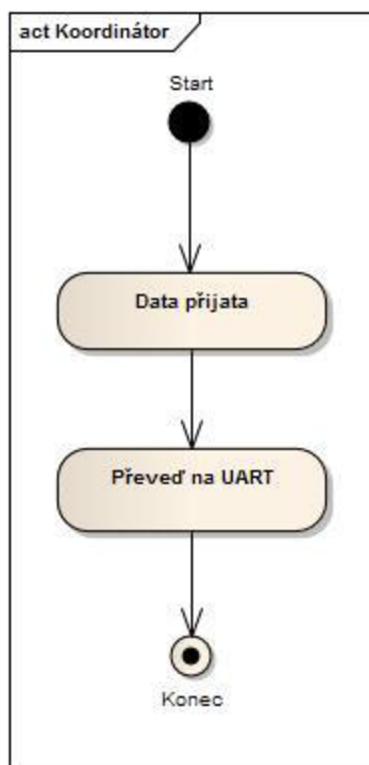
    //funkce pro ulozeni retezce do souboru
    public void Ulozit(string retezec)
    {
        StreamWriter sw = fi.AppendText(); //otevre

        sw.WriteLine(retezec); //ulozi radek do souboru
        sw.Close(); //zavre
        fi.Refresh(); //obnovis soubor
    }

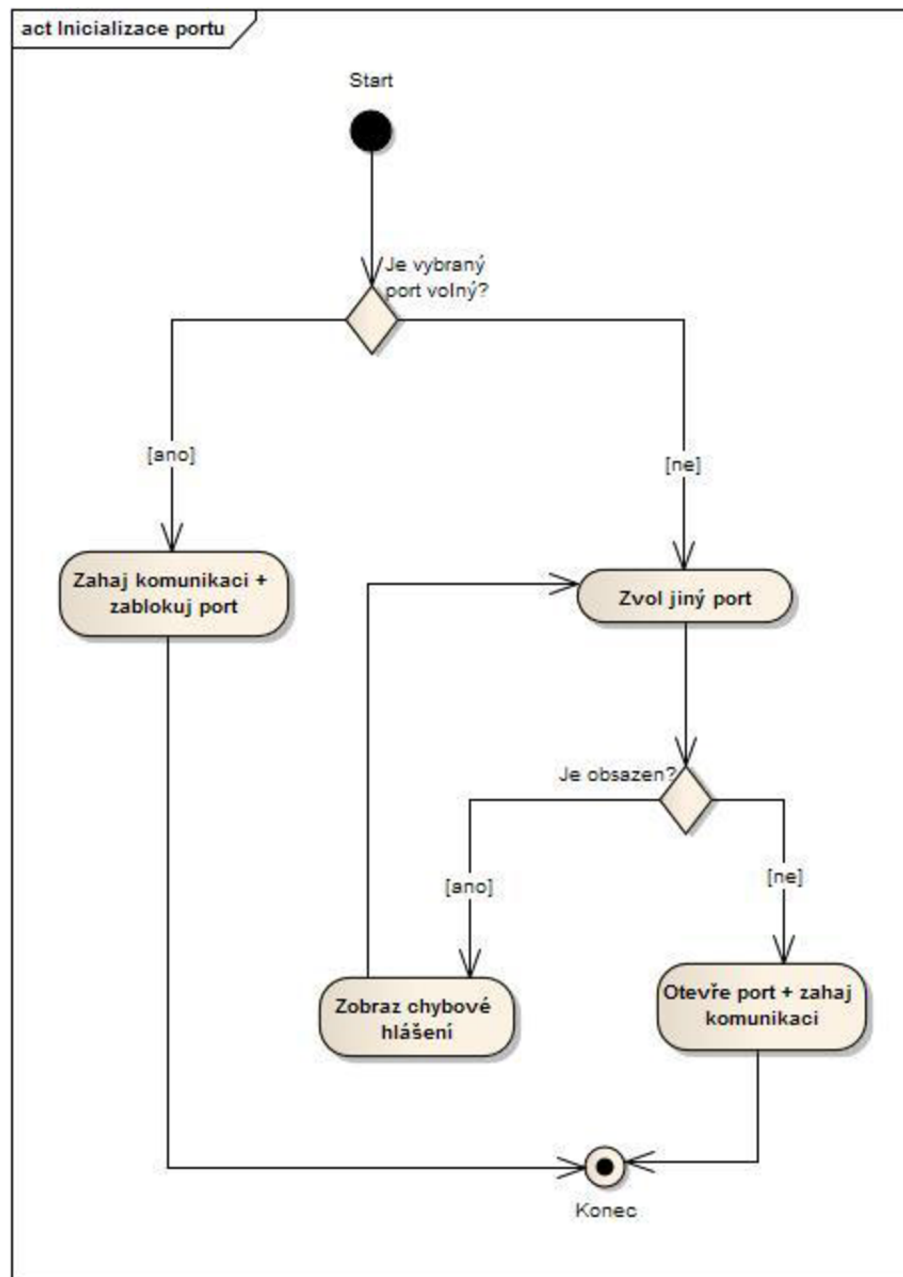
    //event pri stisku tlacitka smazat tabulku
    private void smazBut_Click(object sender, EventArgs e)
    {
        novyRadekTab(" - ", " - ", " - ", radku = -1); // mazani tabulky (radku je
zaporne cislo)
    }
}
}
```



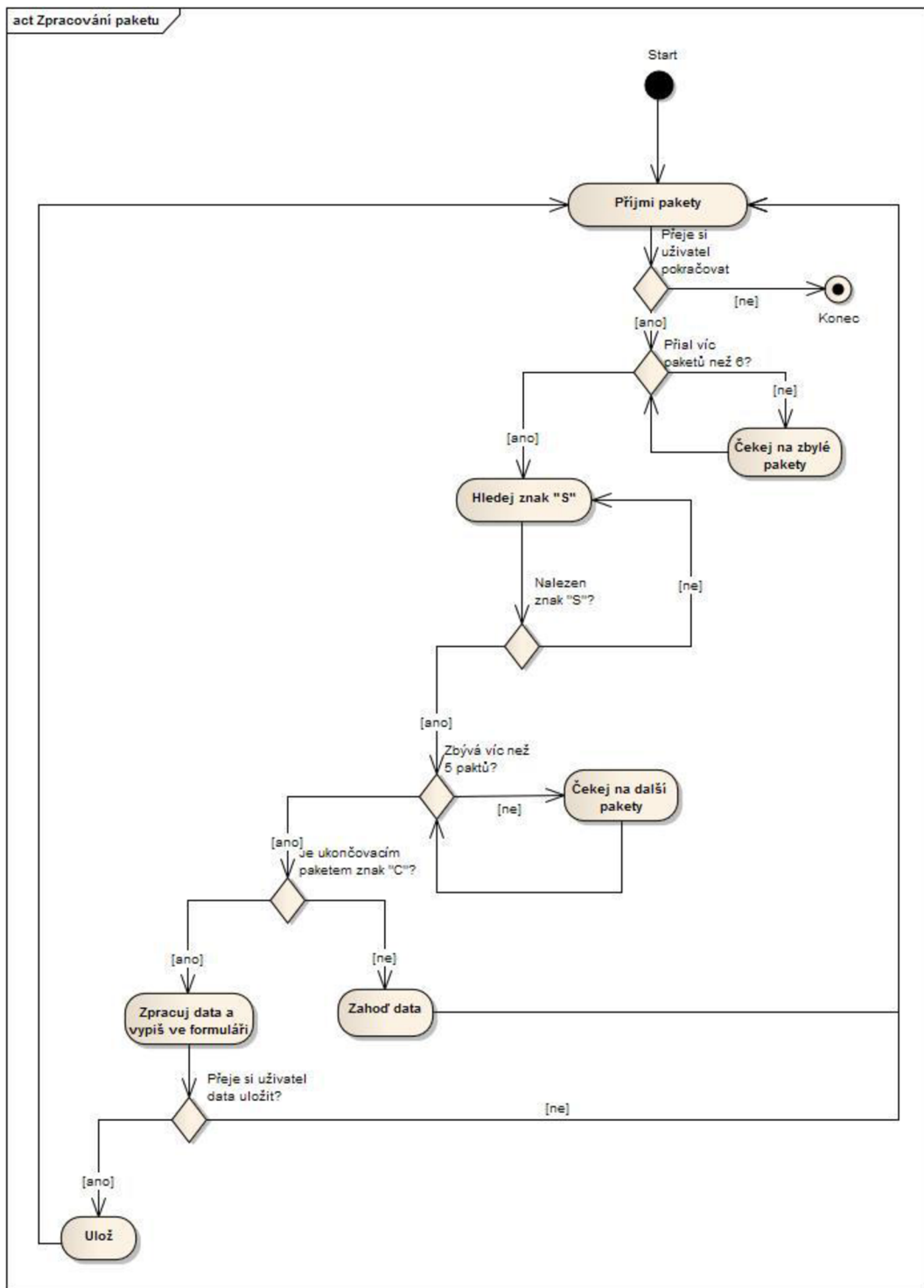
## D Vývojové diagramy



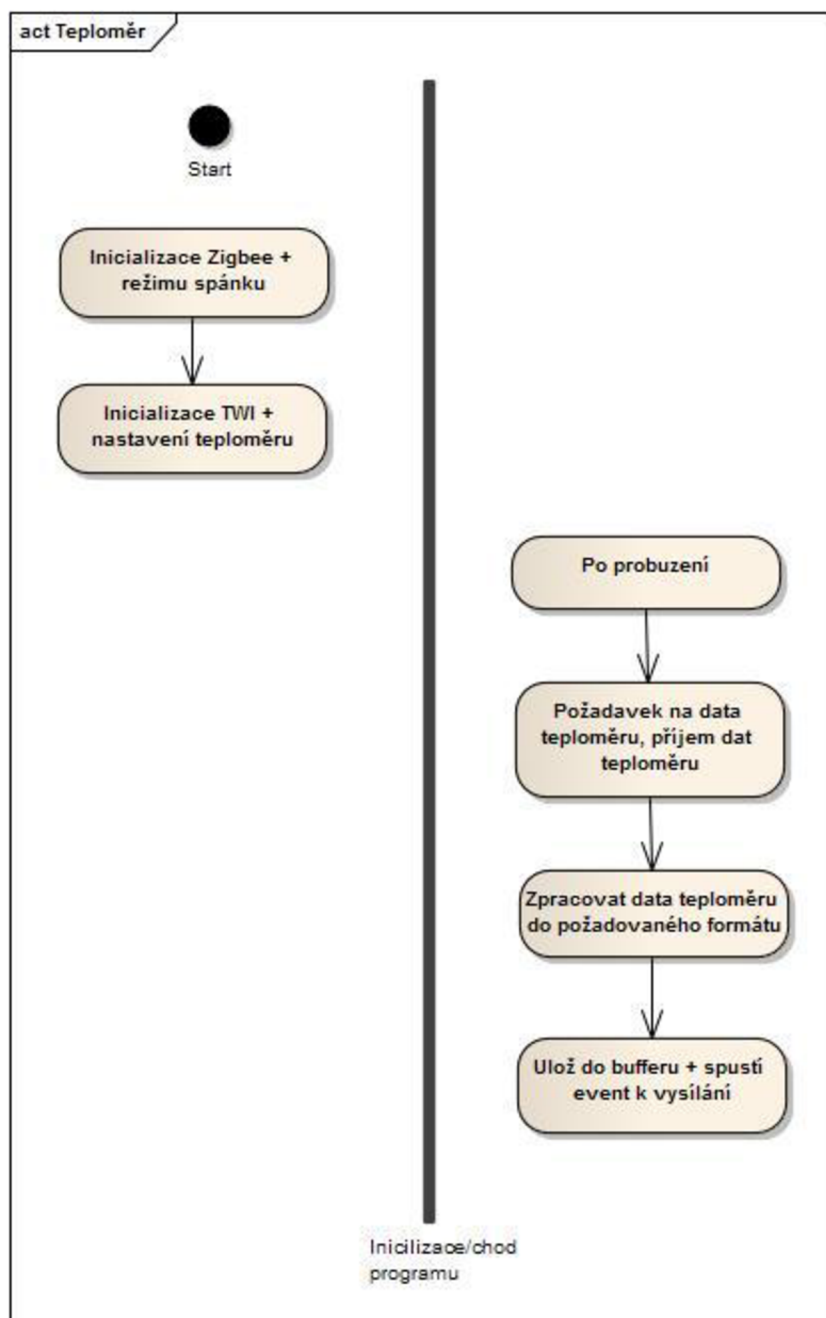
Obr. 28 Vývojový diagram koordinátoru



Obr. 29 Inicializace portu v PC programu



Obr. 30 Zpracování paketů PC programem



Obr. 31 Vývojový diagram koncového zařízení - teploměru

## E Seznam součástek

Koordinátor

Part	Value	Device	Package
3,3V	LM1084	LM1084	TO-220
C2	1u	C-EU025-024X044	C025-024X044
C3	100n	C-EU025-024X044	C025-024X044
C4		CPOL-EUE1.8-4	E1,8-4
C5	2,2uF	CPOL-EUE1.8-4	E1,8-4
C6	0,1uF	C-EU025-024X044	C025-024X044
C7	100n	C-EU025-024X044	C025-024X044
J1		JACK-PLUG1	SPC4078
JP1		JP1E	JP1
JUM1		S1G4_JUMP	S1G4_JUM
JUM2		S1G10_JUMP	S1G10_JUMP
LED2		LED3MM	LED3MM
LED3		LED3MM	LED3MM
LED4		LED3MM	LED3MM
R1	330	R-EU_0207/10	0207/10
R2	330	R-EU_0207/10	0207/10
R3	10k	R-EU_0207/10	0207/10
R4	33	R-EU_0207/10	0207/10
R5	10k	R-EU_0207/10	0207/10
RESET	P-B1720	P-B1720	P-B1720
TL	P-B1720	P-B1720	P-B1720
U\$1	REDUKCE2	REDUKCE2	REDUKCE2
XP2	XPORT	XPORT	XPORT

Teploměr

Part	Value	Device	Package
C1	100n	C-EU025-025X050	C025-025X050
C5	1u	C-EU025-024X044	C025-024X044
C9	100n	C-EU025-024X044	C025-024X0441
JP1		JP1	jumper
JUM1		S1G4_JUMP	S1G4_JUMP
JUM2		S1G10_JUMP	S1G10_JUMP
LED1		LED3MM	LED3MM
LED2		LED3MM	LED3MM
LED3		LED3MM	LED3MM

---

PAD1	PAD_1+	PAD_1+	PAD_1
R1	330	R-EU_0207/10	0207/10
R2	330	R-EU_0207/10	0207/10
R6	330	R-EU_0207/10	0207/10
R7	10k	R-EU_0207/10	0207/10
R8	10k	R-EU_0207/10	0207/10
R9	10k	R-EU_0207/10	0207/10
R10	10k	R-EU_0207/10	0207/10
RESET	P-B1720	P-B1720	P-B1720
TL	P-B1720	P-B1720	P-B1720
U\$2	DS1631	DS1631	DS1631
U\$3	REDUKCE2	REDUKCE2	