

Výukový modul pro předmět Programování mikropočítačů: 2,4 GHz komunikační modul

Adam Mynařík

Bakalářská práce
2018

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2017/2018

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Adam Mynařík
Osobní číslo: A15059
Studijní program: B3902 Inženýrská informatika
Studijní obor: Informační a řídicí technologie
Forma studia: prezenční

Téma práce: Výukový modul pro předmět Programování mikropočítačů: 2,4 GHz komunikační modul

Téma anglicky: An Educational Module for the Microcontrollers Programming Course: 2.4 GHz Transceiver Modules

Zásady pro vypracování:

1. Zpracujte literární rešerši na téma bezdrátové komunikace embedded systémů.
2. Navrhněte způsob propojení komunikačního modulu s výukovým vývojovým kitem.
3. Realizujte mikropočítačový modul.
4. Vytvořte podpůrné programové vybavení modulu a ukázkovou aplikaci.
5. Vypracujte výukovou prezentaci k vytvořenému komunikačnímu modulu.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **NORDIC Semiconductor. nRF24L01+ Single Chip 2.4GHz Transceiver Product Specification v1.0, [online]. 2008 [cit. 2017-11-20]. Dostupné z: <http://www.nordicsemi.no>**
2. **NXP Semiconductors. Kinetis KL25 Sub-Family 48 MHz Cortex-M0+ Based Microcontroller with USB, [online]. 2014 [cit. 2017-11-20]. Dostupné z: <http://www.nxp.com>**
3. **NXP Semiconductors. KL25 Sub-Family Reference Manual, [online]. 2014 [cit. 2017-11-20]. Dostupné z: <http://www.nxp.com>**
4. **PINKER, Jiří. Mikroprocesory a mikropočítače. 1. vyd. Praha: BEN – technická literatura, 2004, 159 s. ISBN 80-7300-110-1.**
5. **VÁŇA, Vladimír. ARM pro začátečníky. 1. vyd. Praha: BEN – technická literatura, 2009, 195 s. ISBN 978-80-7300-246-6.**

Vedoucí bakalářské práce:

Ing. Petr Dostálek, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

15. prosince 2017

Termín odevzdání bakalářské práce:

25. května 2018

Ve Zlíně dne 15. prosince 2017



doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu


Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užit své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 21.5.2018


.....
podpis diplomanta

ABSTRAKT

První část bakalářské práce obsahuje rešerši na téma bezdrátové komunikace embedded systémů. Je zaměřena na zejména technologie ZigBee, Bluetooth Low Energy a Wi-Fi. Dále je popsán bezdrátový modul, který je použit v praktické části. Na závěr teoretické části jsou popsány technické parametry školního vývojového kitu s mikroprocesorem KL25Z. V druhé části je navržen způsob propojení mezi vývojovým kitem a bezdrátovým modulem NRF24L01+. Dále je vypracovaná podpůrná programová knihovna, která obsahuje funkce pro obsluhu komunikačního modulu. V další části je ukázka demo aplikace, která využívá danou knihovnu. Závěrem je vytvořena výuková prezentace pro studenty pro seznámení s programovou knihovnou.

Klíčová slova: NRF24L01+, KL25Z, Bezdrátový komunikační modul, Mikroprocesor, Programová knihovna

ABSTRACT

The first part of the bachelor thesis contains research on the wireless communication of embedded systems. It focuses on ZigBee, Bluetooth Low Energy and Wi-Fi technologies. Next, a wireless module is described, which is used in the practical part. At the end of the theoretical part are described the technical parameters of the school development kit with microcontroller KL25Z. The second part proposes a connection between the development kit and the NRF24L01 + wireless module. In addition, a support program library is created, which contains functions for the operation of the communication module. The next section is a demo application that uses the library. Finally, a tutorial presentation is created for students to get acquainted with the program library.

Keywords: NRF24L01+, KL25Z, Wireless communication module, Microcontroller, Program library

Děkuji panu vedoucímu mé bakalářské práce, Ing. Petrovi Dostálkovi, Ph.D., za cenné rady a trpělivost během vypracovávání mé bakalářské práce. A také rodičům a přítelkyni za podporu.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 BEZDRÁTOVÁ KOMUNIKACE EMBEDDED SYSTÉMŮ	11
1.1 HISTORIE BEZDRÁTOVÉ KOMUNIKACE.....	11
1.2 DRUHY BEZDRÁTOVÉ KOMUNIKACE.....	11
1.3 BEZDRÁTOVÉ TECHNOLOGIE PRO EMBEDDED SYSTÉMY	14
1.3.1 ZigBee	14
1.3.2 Bluetooth low energy	16
1.3.3 WiFi.....	19
2 BEZDRÁTOVÝ MODUL NRF24L01+	23
2.1 FREKVENČNÍ PÁSMO.....	23
2.2 TOPOLOGIE SÍTĚ	24
2.3 VYSÍLACÍ VÝKON	25
2.4 SPOTŘEBA MODULU.....	25
2.5 REŽIMY MODULU	26
2.6 TECHNICKÉ PARAMETRY	26
2.7 PROPOJENÍ BEZDRÁTOVÉHO MODULU S MCU.....	27
2.8 TVAR PAKETU ENHANCED SHOCKBURST	28
2.9 GFSK MODULACE	29
3 VÝVOJOVÝ KIT S KL25Z	31
II PRAKTICKÁ ČÁST	33
4 PROPOJENÍ KL25Z S NRF24L01+	34
5 PROGRAMOVÁ KNIHOVNA PRO OBSLUHU MODULU	37

5.1	FUNKCE VOID SPI_INIT(VOID)	37
5.2	FUNKCE STATIC UINT8_T SPI_SEND(UINT8_T VALUE).....	38
5.3	FUNKCE VOID WRITEREGISTER(UINT8_T REG, UINT8_T VALUE).....	39
5.4	FUNKCE VOID WRITEREGISTERDATA(UINT8_T REG, UINT8_T *BUFFER, UINT8_T BUFFERSIZE)	40
5.5	FUNKCE VOID READREGISTERDATA(UINT8_T REG, UINT8_T *BUFFER, UINT8_T BUFFERSIZE)	40
5.6	FUNKCE VOID WRITEREAD(UINT8_T VALUE).....	41
5.7	FUNKCE VOID WRITE(UINT8_T VALUE).....	41
5.8	FUNKCE UINT8_T GETSTATUS(VOID).....	42
5.9	FUNKCE RESETSTATUS(VOID).....	42
5.10	FUNKCE VOID TRANSMIT_PAYLOAD(UINT8_T *PAYLOAD, UINT8_T PAYLOADSIZE)	43
5.11	FUNKCE VOID RECEIVE_PAYLOAD(UINT8_T *PAYLOAD, UINT8_T PAYLOADSIZE)	43
5.12	FUNKCE PRO NASTAVOVANÍ VÝSTUPNÍCH LOGICKÝCH HODNOT	44
6	TVORBA UKÁZKOVÉHO APLIKACE	45
6.1	PROGRAM PRO VYSÍLAČ	46
6.2	PROGRAM PRO PŘIJÍMAČ.....	48
6.3	OVĚŘENÍ FUNKCE MODULŮ	ERROR! BOOKMARK NOT DEFINED.
7	PREZENTACE PRO PODPORU VÝUKY	52
	ZÁVĚR	53
	SEZNAM POUŽITÉ LITERATURY.....	54
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	56
	SEZNAM OBRÁZKŮ	58
	SEZNAM TABULEK.....	59
	SEZNAM PŘÍLOH.....	60

ÚVOD

Stále širší využití IoT (Internet of Things), s čím se zvětšuje i trh s těmito prostředky, je umožněno technickým pokrokem v různých oborech. Jako například vývoj v oblasti rádio frekvenčních přenosů nízkého výkonu.

Bezdrátová zařízení, aplikace a služby již radikálně změnily způsob jakým žijeme, pracujeme a jak se socializujeme. Existující koncepce Internetu Věcí dále slibuje konektivitu mezi prakticky jakýmkoli objekty v prostředí, které realizují komunikaci mezi strojem a člověkem. To by dále změnilo způsob komunikace s fyzickým světem a bezdrátová komunikace se stane integrovanou součástí lidského života.

V průmyslu mohou IoT zařízení sloužit pro sledování a optimalizaci procesů, jejímž výsledkem může být zvýšení produktivity výroby. Bezdrátová vestavěná zařízení se setkávají s výzvami z hlediska nákladů, fyzických rozměrů a spotřeby energie.

Tato bakalářská práce obsahuje stručný pohled do historie bezdrátové komunikace a její základní dělení. Dále se zabývá bezdrátovými technologiemi pro embedded systémy a to ZigBee, Bluetooth Low Energy a Wi-Fi.

Práce popisuje bezdrátový modul NRF24L01+ a vývojový školní kit s mikropočítačem. V práci se také zabýváme jak vhodně bezdrátový modul s vývojovým kitem propojit.

Výstupem praktické části je podpůrná programová knihovna, která umožňuje komunikaci po SPI sběrnici mezi mikropočítačem a modulem a také posílat či přijímat data. K ukázce funkčnosti programové knihovny byla vytvořena aplikace, která na jednom vývojovém kitu měří teplotu a vlhkost. Na druhém vývojovém kitu jsou naměřená data zobrazována na LCD displeji. Jestli byla data odeslána či přijata nám indikují dvě LED.

I. TEORETICKÁ ČÁST

1 BEZDRÁTOVÁ KOMUNIKACE EMBEDDED SYSTÉMŮ

Bezdrátová komunikace zajišťuje přenos dat mezi dvěma a více zařízeními bez použití spojovacího kabelu. Data mohou být v prostoru přenášena na vzdálenosti několika metrů až milióny kilometrů.

1.1 Historie bezdrátové komunikace

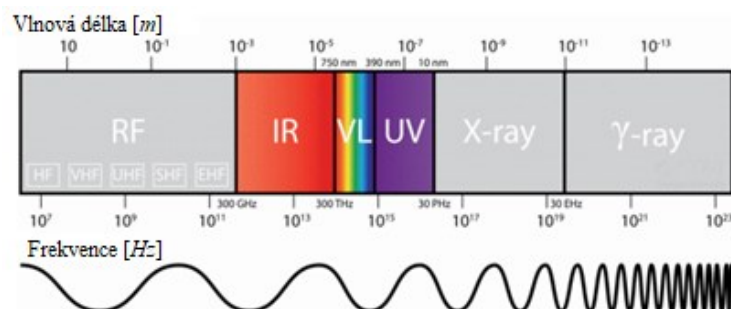
Kouřové signály jenž byly používány Indiánskými kmeny, by mohly být považovány za začátky bezdrátových komunikačních systémů. Mezi roky 1861 a 1862 byla formulována matematická teorie elektromagnetických vln Jamesem Clerkem Maxwellem a později roku 1887 demonstroval existenci těchto vln Heinrich Hertz. Nakonec Guglielmo Marconi roku 1901 provedl první transatlantické bezdrátové spojení. Tyto události se staly mezníkem bezdrátové komunikace a po dalších sto let byly vynalézány další metody přenosu, až po zařízení pro bezdrátový přenos jako známe dnes.[1]

1.2 Druhy bezdrátové komunikace

Máme tři základní typy bezdrátové komunikace:

- Optická komunikace

Informace se přenáší pomocí infračerveného, viditelného a ultrafialového světla. Optické komunikační systémy se snaží řešit omezení rádiové komunikace. Optická bezdrátová komunikace má ve srovnání s vysokofrekvenčními protějšky vynikající vlastnosti, jako je extrémně vysoká šířka pásma, odolnost vůči elektromagnetickému rušení, vysoký stupeň prostorového omezení, který přináší prakticky neomezené použití a fyzickou bezpečnost. Navíc to, že působí v neregulovaném spektru, nevyžaduje žádný licenční poplatek.[2]

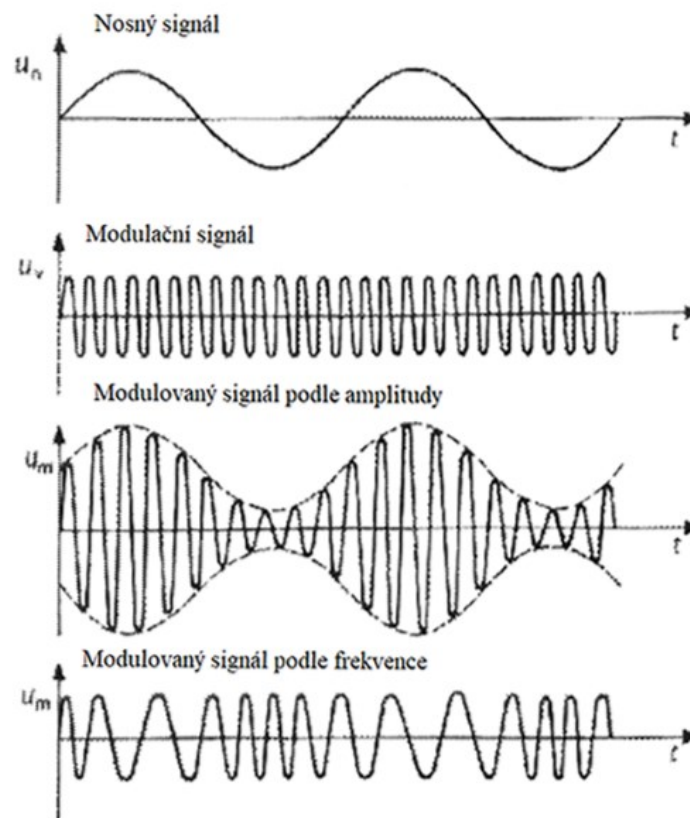


Obr. 1. Elektromagnetické spektrum [2]

Systémy optické bezdrátové komunikace pracující ve viditelném pásmu, využívají světelné diody, které mohou pulzovat při velmi vysokých rychlostech, aniž by to lidské oko stihlo zaregistrovat, používají se pro osvětlovací a komunikační účely a mají energetický potenciál v tom, jak využíváme světlo. Pozemní systémy typu point-to-point, pracující s blízkými IR frekvencemi. Tyto systémy typicky používají laserové vysílače s vysokými přenosovými rychlostmi. Nedávný pokrok v optických zdrojích a detektorech, zvedl zájem o ultrafialovou komunikaci. V takzvaném hlubokém UV pásmu je sluneční záření zemi zanedbatelné, což umožňuje konstrukci detektorů, které dokáží rozlišovat jednotlivá kvanta elektromagnetického záření, tedy fotony. Takové konstrukce podporují nízkou spotřebu UVC krátkého dosahu, například v bezdrátových sensorových sítí.[2]

- **Rádiová komunikace.**

Informace přenáší rádiové vlny, pomocí modulace vlastností vln elektromagnetické energie přenášeným prostorem, jako je amplituda, kmitočet, fáze nebo šířka pulzu. Rádiová vlna je generována vysílačem, který moduluje charakter nosného signálu pomocí modulujícího signálu. Detekovaná signálů probíhá pomocí přijímače. Toto je zajištěno díky anténám, které je používána pro vysílání a přijímání rádiových signálů. Antény jsou tedy určeny k přeměně elektrické energie na energii elektromagnetických vln a naopak. Rozsah rádiového spektra se pohybuje v rozmezí od 3 kHz do 300 GHz.[3]



Obr. 2. Ukázka amplitudové a frekvenční modulace [4]

- Sonická komunikace

Informace se přenáší zvukem, k šíření zvuku je potřeba látkového prostředí, všechny zvuky jsou vibracemi, které procházejí vzduchem jako zvukové vlny. Zvukové vlny jsou způsobeny vibracemi objektů a ty vyzařují ze svého zdroje všemi směry. Vibrační předmět tlačuje okolní molekuly vzduchu (více se k sobě přiblíží) a pak je odděluje (odtáhne je dále od sebe). V různých prostředí mají zvukové vlny různou vlnovou délku a rychlost šíření, frekvence zůstává stejná. Zvuk se ve své cestě odráží od objektů a dochází k pohlcování zvuku, amplituda tedy klesá. Frekvenční rozsah, který člověk může vnímat je od 16 Hz do 20 kHz. Frekvence menší než 16 Hz se nazývá infrazvuk, vyšší než 20 KHz ultrazvuk.

Dále se bude tato práce zabývat rádiovou komunikací pro embedded systémy.

1.3 Bezdrátové technologie pro embedded systémy

Bezdrátová zařízení, aplikace a služby již radikálně změnilы způsob jakým žijeme, pracujeme a jak se socializujeme. Existující koncepce Internetu Věcí dále slibuje konektivitu mezi prakticky jakýmkoli objekty v prostředí, které realizují komunikaci mezi strojem a člověkem. To by dále změnilo způsob komunikace s fyzickým světem a bezdrátová komunikace se stane integrovanou součástí lidského života. V následujících kapitolách jsou uvedeny tři bezdrátové technologie pro embedded systémy.

1.3.1 ZigBee

ZigBee je standard který definuje sadu komunikačních protokolů pro bezdrátovou síť s malým datovým tokem a krátkým dosahem. Je postavena na standardu IEEE 802.15.4 a patří do skupiny bezdrátových sítí PAN (Personal Area Network). Na trhu je od roku 2004, kdy byl vydán první ZigBee standard.[5]



Obr. 3. Ukázka ZigBee modulu XBCAWIT-001 [6]

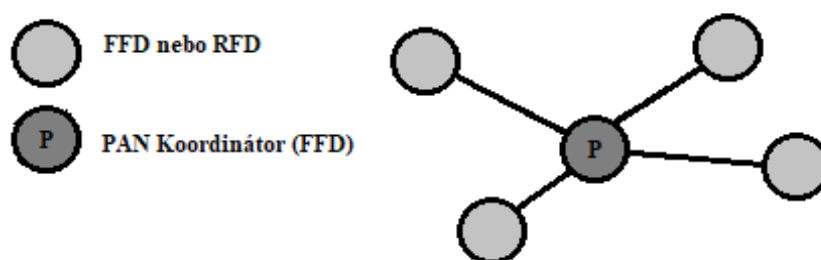
Frekvenční pásma

Globálně pracuje ve frekvenčním pásmu 2,4 GHz s 16 kanály a maximální přenosovou rychlostí 250 kbit/s. V Americe je to pásmo 915 MHz a 10 kanálů a nakonec v Evropě je pásmo 868 MHz s jedním kanálem. Rozsah přenosu je od 10 metrů do 100 metrů v závislosti na jeho nízkém výkonu.[7]

Topologie sítě

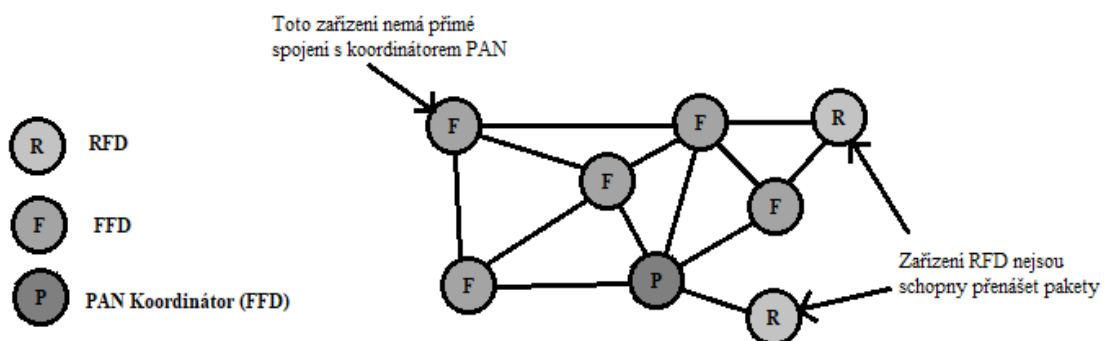
Technologie je postavena na dvou topologiích specifikovaných v IEEE 802.15.4, tedy topologie hvězdicová a peer to peer. ZigBee standard rozděluje zařízení na zařízení s plnou funkcí a zařízení s omezenou funkcí.

V hvězdicové topologii může každé zařízení v síti komunikovat jen s PAN koordinátorem. Například FFD (Full Functional Device), který je naprogramován jako PAN koordinátor, je aktivován a začíná vytvářet svou síť. První úkol tohoto PAN koordinátoru je vybrat identifikátor, který není používán žádnou jinou blízkou sítí.[5]



Obr. 4. Hvězdicová topologie u ZigBee [5]

V peer-to-peer topologii, může každé zařízení komunikovat přímo s kterýmkoli jiným zařízením, pokud jsou zařízení umístěna dostatečně blízko k vytvoření úspěšného komunikačního spojení. Každé FFD v této síti může být v roli PAN koordinátora. Všechna zařízení, která se podílejí na přenosu zpráv jsou FFD, protože RFD (Reduced Functionality Device) nejsou schopny předávat zprávy. RFD však může být součástí sítě a komunikovat pouze s jedním konkrétním zařízením v síti. Tato topologie může mít různé tvary a omezení daných zařízení, které spolu mohou komunikovat, pokud neexistuje žádné omezení, tak se topologie nazývá mesh.[5]



Obr. 5. Mesh topologie u Zigbee [5]

Další forma sítě peer-to-peer je stromová topologie. V tomto případě koordinátor vytvoří počáteční síť. ZigBee směrovače řídí kam budou jednotlivé zprávy směrovány. Směrovače ZigBee mohou rozšiřovat síť nad rámec počáteční sítě, kterou zřídil koordinátor.[5]

Technické parametry

Zde jsou rozepsány jednotlivé parametry technologie ZigBee:

- Frekvenční pásmo: 2,4 GHz, 915 MHz, 868 MHz
- Maximální přenosová rychlost: 250 kbit/s
- Maximální dosah: 100 m
- Typ modulace: BPSK, OQPSK
- Maximální počet uzlů: 65536
- Šifrování: 128-bit AES
- Topologie: Hvězda, Mesh, Strom

1.3.2 Bluetooth low energy

Tato technologie je jedna z nejnovějších technologií Bluetooth, která byla poprvé přidána jako součást specifikace Bluetooth 4.0. Jak název napovídá, tato technologie je zaměřena na zařízení s velmi nízkou spotřebou energie. Dříve tato technologie byla známa jako Wibree a Ultra Low Power. Předpokládá se tedy, že zařízení vyhovující tomuto standardu, mají nízkou spotřebu energie, takže mohou pracovat měsíce, dokonce i roky na menších knoflíkových bateriích, aniž by je bylo nutné dobít či vyměňovat. Datová komunikace je obecně v krátkých úsecích, které nemusí být časté. Je tedy nejvhodnější pro zařízení, které nevyžadují vysoký výkon a velké datové přenosy.[8]



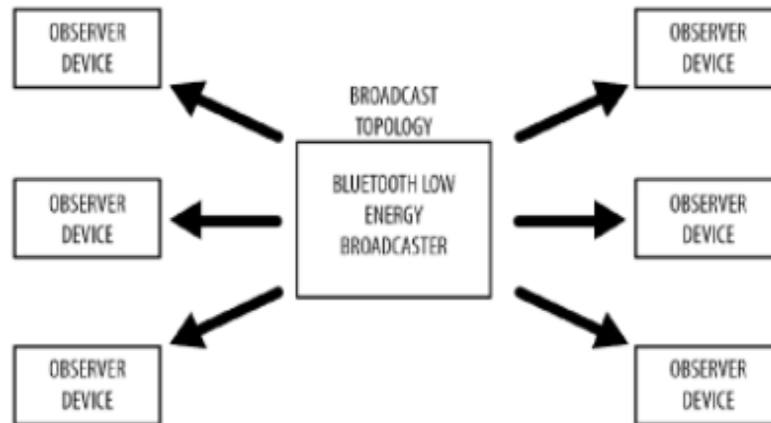
Obr. 6. Ukázka BLE modulu EY-08 [9]

Frekvenční pásma

Tato technologie pracuje v bezlicenčním ISM pásmu. Pro zajištění spolehlivého provozu v kmitočtovém pásmu 2,4 GHz přesněji v rozmezí 2400-2483,5 MHz. Využívá přístupu Adaptive Frequency Hopping což je adaptivní forma přeskokování frekvence, která se snaží snížit rušení přizpůsobením sekvence skoku, aby se zařízení vyhnuly kanálům používaným jinými zařízeními pracujícími ve stejném pásmu ISM. Adaptivní tedy znamená, že zařízení Bluetooth kontroluje, které kanály jsou již hodně používány a vyhýbají se takovým kanálům v sekvenci skoku, díky tomu se přenáší data přes 40 kanálů.[10]

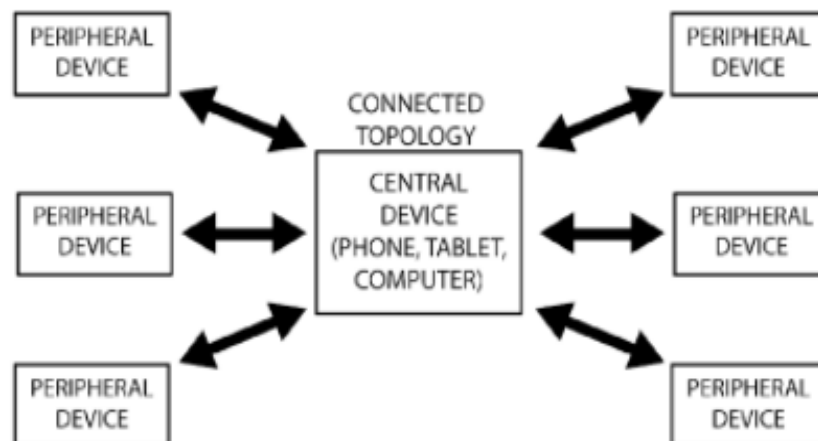
Topologie sítě

Zařízení můžeme rozdělit do dvou rolí a to Vysílač, ten pravidelně posílá pakety komukoli kdo je v dosahu a ochotně je přijímá. A dále Přijímač, ten kontroluje všechny přednastavené frekvence a přijímá všechny pakety, které jsou právě v tom okamžiku vysílány. Broadcast je důležitý, protože umožňuje přenášet data na více než jedno zařízení v čase. Standardní paket obsahuje 31 bajtů, které se používají pro popsání vysílatele a jeho vlastností, ale může obsahovat i vlastní informace, které je potřeba poslat do jiného zařízení. Pokud bude potřeba poslat více dat, BLE podporuje i druhý paket s 31 bajty, tedy celkem až 62 bajtů. Broadcasting je tedy rychlé a snadné a je vhodné pro přenášení menšího množství dat na více zařízení.[11]



Obr. 7. Broadcast topologie u BLE [11]

Druhý způsob se nazývá Připojení, pokud je potřeba přenášet data v obou směrech a nebo je potřeba přenést větší množství dat. Připojení je trvalá periodická výměna dat mezi dvěma zařízeními. Spojení zařízení zahrnují dvě samostatné role: Centrální (master), ten opakovaně prohledává přednastavené frekvence pro pakety a pokud je vhodná, iniciuje připojení. Jakmile je spojení vytvořeno, master spravuje periodické výměny dat. A dále Periferní (slave), tedy zařízení, které pravidelně odesílá připojitelné pakety a přijímá příchozí spojení. Jakmile je aktivní spojení, periferní zařízení sleduje časování a pravidelně s ním vyměňuje data.[10]



Obr. 8. Connected topologie u BLE [11]

Technologie BLE tedy podporuje tři síťové topologie podle potřeby bezdrátového připojení.

Patří zde:

- Point-to-point

Používá se pro komunikaci zařízení v poměru 1:1.

- Broadcast

Tato topologie se využívá pro komunikaci mezi zařízeními v poměru 1:m. Data se mohou odesílat do libovolného přijímacího zařízení v dosahu.

- Mesh

Tato síťová topologie umožňuje více zařízením propojit se a vzájemně komunikovat prostřednictvím ostatních zařízení v síti. Využívá se tedy pro komunikaci v poměru m:m. Mesh nebyla dříve součástí technologie Bluetooth, to se změnilo až po vydání BLE.

Technické parametry

- Frekvenční pásmo: 2,4 GHz
- Maximální přenosová rychlost: 2 Mb/s
- Maximální dosah: více než 400 m
- Typ modulace: GFSK
- Šifrování: 128-bit AES
- Topologie: Point-to-Point, Broadcast, Mesh

1.3.3 WiFi

Tuto technologii někteří považují za jednu z nejvýznamnějších technologických inovací od doby, co se rozšířil do světa Internet. Předchůdce dnešní WiFi byl na počátku devadesátých let vybudován v Nizozemsku. Nazýval se WaveLan a byl určen pro použití v pokladnách. První protokol ze standardu IEEE 802.11, byl uveden v roce 1997. O dva roky později byla tato verze rozšířena o nové specifikace a byla také založena WiFi Alliance, sdružení několika firem, která drží ochrannou známku WiFi.[12]



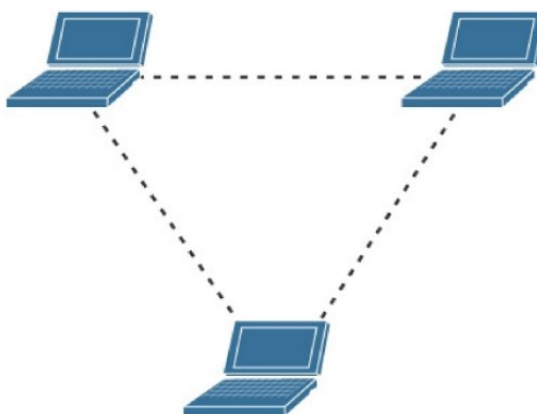
Obr. 9. Ukázka WiFi modulu ESP-07 ESP8266 [13]

Frekvenční pásma

Tato technologie funguje na několika standardech 802.11 ve frekvenčních bezlicenčních pásmech ISM při frekvencích 2,4 GHz a 5 GHz. Pásmo 2,4 GHz jenž přesněji pracuje od frekvence 2400 MHz až po 2500 MHz, je to nejrozšířenější z dostupných pásem pro Wi-Fi. Používá se pro 802.11 b/g/n. Pro 802.11g a 802.11n se šířkou pásma 20 MHz a pro starší 802.11b s šířkou pásma kanálu 22MHz. Může nést maximálně tři nepřekrývající se kanály. Pásmo 5 GHz. Může být použito pro 802.11 a/n a také pro 802.11ac. Může nést až 24 nepřekrývajících se kanálů.[14]

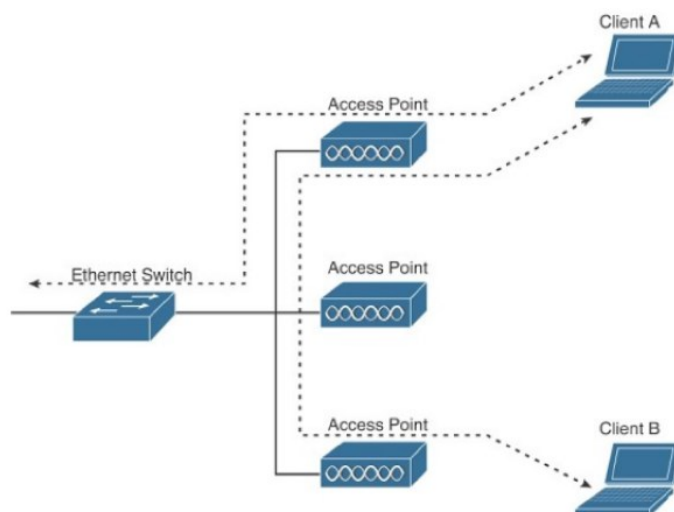
Topologie sítě

V závislosti na požadavcích systému je možné konfigurovat síť WLAN do různých architektur. První je ad-hoc, ta se skládá pouze z klientských stanic a je bez přístupových bodů. Klientské stanice komunikují přímo mezi sebou což se nazývá komunikace peer to peer. Hlavním smyslem ad hoc je zabezpečit komunikaci pouze mezi jednotlivými stanicemi. Nevýhoda je, že se může komunikovat pouze se zařízeními, které jsou právě v jeho dosahu.[15]



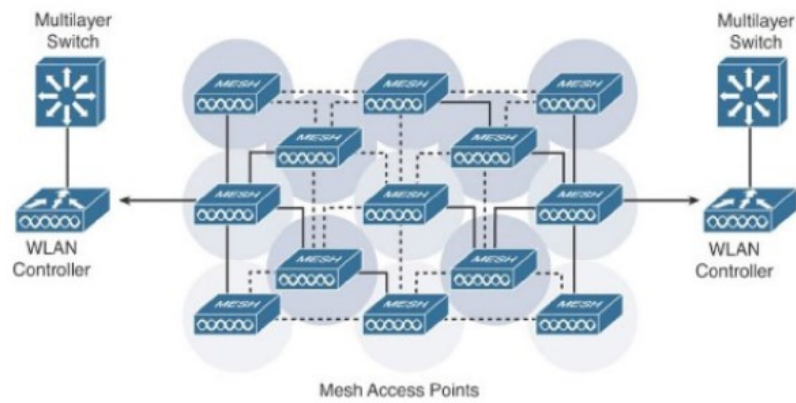
Obr. 10. Ad-hoc topologie u Wi-fi [15]

V infrastrukturní síti mezi sebou mohou radiopřijímače komunikovat tak, že máme jeden přístupový bod s jedním nebo více klientskými stanicemi. Tyto klientské stanice se připojují k přístupovému bodu a prostřednictvím jeho komunikují. Tohle zapojení se může rozšířit pomocí distribučního média, které slouží pro připojení přístupových bodů nejběžněji kabelem.[14]



Obr. 11. Infrastrukturní síť u Wi-fi [15]

V mesh síti je jeden nebo více přístupových bodů připojen kabelem k infrastruktuře a jiné přístupové body, které nejsou připojené k drátové síti, vytvoří bezdrátové páteřní připojení ke konkrétnímu přístupovému bodu.[15]



Obr. 12. Mesh topologie u Wi-fi [15]

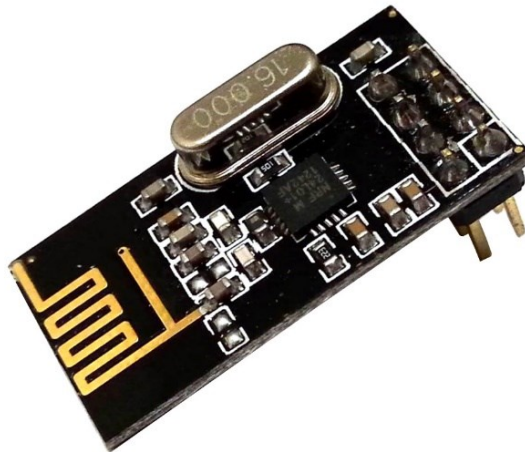
Technické parametry

Pro standard 802.11 b/g/n

- Frekvenční pásmo: 2,4 GHz, 5 GHz
- Maximální přenosová rychlost: 600 Mb/s
- Maximální dosah: 100m
- Typ modulace: OFMD
- Šifrování: 256-bit QAM
- Topologie: Ad hoc, Infrastruktura, Mesh

2 BEZDRÁTOVÝ MODUL NRF24L01+

Je jednočipový vysílač s přijímačem od firmy Nordic Semiconductors pracující na frekvenci 2,4 GHz s vestavěným protokolem nazvaným Enhanced ShockBurst, vhodným pro bezdrátové aplikace s velmi nízkou spotřebou. Tento modul je určen pro provoz v celosvětovém kmitočtovém pásmu ISM od 2,4 do 2,4835 GHz. Tento čip lze ovládat a konfigurovat přes rozhraní SPI. Protokol je založen na komunikaci prostřednictvím paketů a podporuje různé režimy. Má uživatelsky konfigurovatelné parametry, jako je frekvenční kanál, výstupní výkon a rychlost přenosu dat.[16]



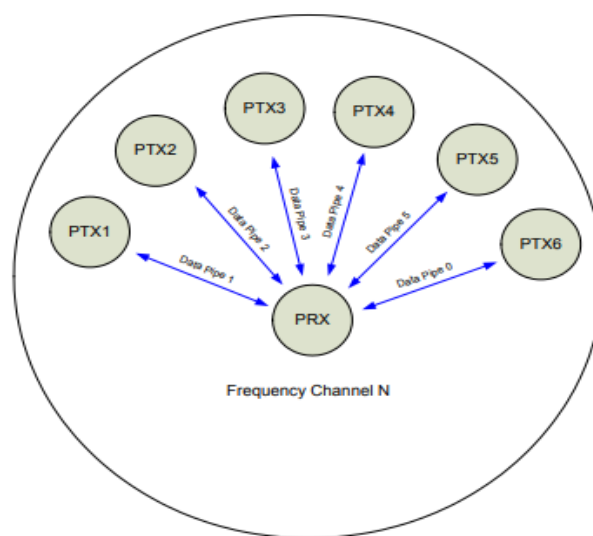
Obr. 13. Bezdrátový modul nRF24L01+ [17]

2.1 Frekvenční pásma

Kanál zaujímá šířku pásma menší než 1 MHz při rychlosti 250 kb/s a 1 Mb/s. Šířku pásma menší než 2 MHz při rychlosti 2 Mb/s. Nrf24l01+ může pracovat na frekvencích od 2,4 GHz do 2,525 GHz. Programové rozlišení nastavení frekvence RF kanálu je 1 MHz. Při rychlosti 2Mbps kanál obsadí šířku pásma, která je širší než rozlišení frekvenčního kanálu RF. Pro zajištění nepřekrývajících se kanálů v režimu 2 Mb/s musí být vzdálenost kanálu 2MHz nebo vyšší. Při rychlostech 1 Mb/s a 250 kb/s je šířka pásma kanálu stejná nebo nižší než rozlišení RF frekvence. Nastavená frekvence musí být stejná pro přijímač i vysílač, aby spolu mohli komunikovat.[16]

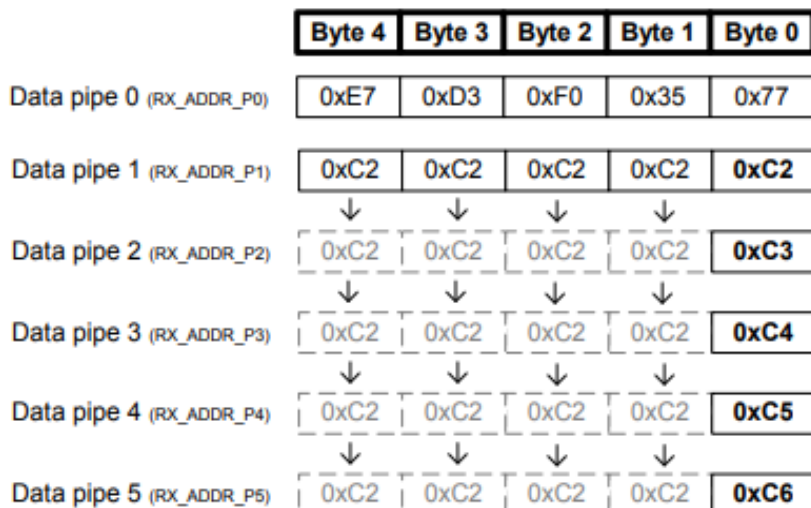
2.2 Topologie sítě

Tento modul obsahuje funkci nazvanou Multiceiver, která je používána v režimu RX (tedy „čtení“) a obsahuje až 6 paralelních datových trubic, které mají vlastní dekódování fyzických adres v nRF24l01. Modul konfigurovaný jako PRX tedy primární přijímač, může přijímat data pro šest různých datových trubic v jednom frekvenčním kanálu. Každá datová trubice má svou vlastní jedinečnou adresu a může být konfigurováno individuálně. Až šest nRF24l01 modulů konfigurovaných jak PTX, neboli primární odesílatel, může komunikovat s jedním modulem konfigurovaným jako PRX.[16]



Obr. 14. Multiceiver u nRF24L01+ [16]

Každá datová trubice může mít konfigurovatelnou adresu až do 5 bajtů. Datová trubka s indexem 0, má jedinečnou 5-ti bajtovou adresu. Datové trubky 1-5 sdílejí nejvýznamnější bajty adres. LSB musí být jedinečné pro všech 6 trubek. PRX přijímá pakety z více než jednoho PTX. Aby se zajistilo, že paket ACK z PRX, který obsahuje informaci, že data přišli v pořádku od PTX, PRX převezme adresu datové trubky, odkud přijala paket s daty od PTX a použije jej jako adresu TX při přenosu paketu ACK. Žádná jiná datová trubka nemůže přijímat data, dokud nedojde k přijetí kompletního paketu datovým potrubím, které zjistilo jeho adresu. Po dokončení tohoto procesu už může více PTX vysílat do jednoho PRX.[16]



Obr. 15. Adresování datových trubíc [16]

2.3 Vysílací výkon

Vysílací výkon jde programově nastavit ve čtyřech variantách, viz tabulka č. 1.

Tab. 1. Spotřeba modulu podle vysílacího výkonu

Vysílací výkon	Proudová spotřeba
0 dBm	11,3 mA
-6 dBm	9 mA
-12 dBm	7,5 mA
-18 dBm	7 mA

2.4 Spotřeba modulu

Spotřeba je různá od stavu ve kterém se modul nachází, viz. tabulka č. 2.

Tab. 2. Spotřeba modulu podle jeho stavu

Stav	Proudová spotřeba
Vypnutý	900 nA
Pohotovostní režim	26 μ A
Odesílání dat (při 0 dBm)	11,3 mA
Příjem dat (při 2 Mb/s)	13,5 mA

2.5 Režimy modulu

Modul NRF24L01+ může být konfigurován do několika režimů, které jsou popsány níže:

- Režim vypnutí – Power Down Mode

V tomto režimu je minimální spotřeba proudu, což umožňuje udržovat všechny hodnoty registru dostupné, SPI (Serial Peripheral Interface) je aktivní, takže je možná změna konfigurace registrů.[16]

- Pohotovostní režim – Standby mode

Tento režim může mít dva typy, první pohotovostní režim, který se nastaví, pokud bude na CE (Chip Enable), logická jednička. Modul se také nastavuje do pohotovostního režimu jedna, když bude CE logická nula a jakmile skončí režim TX nebo RX. Do pohotovostního režimu dva vstupuje zařízení, pokud je CE logická jednička a zařízení je nastaveno jako primární vysílač s prázdnou frontou FIFO.[16]

- Přijímací režim – RX režim

V tomto režimu přijímač demoduluje signály z RF kanálu, neustále je vyhledáván platný paket (s platným CRC a odpovídající adresou), jakmile je nalezen uloží se do RX FIFO fronty. Ta pokud je plná, tak je přijatý paket vyřazen.[16]

- Vysílací režim – TX režim

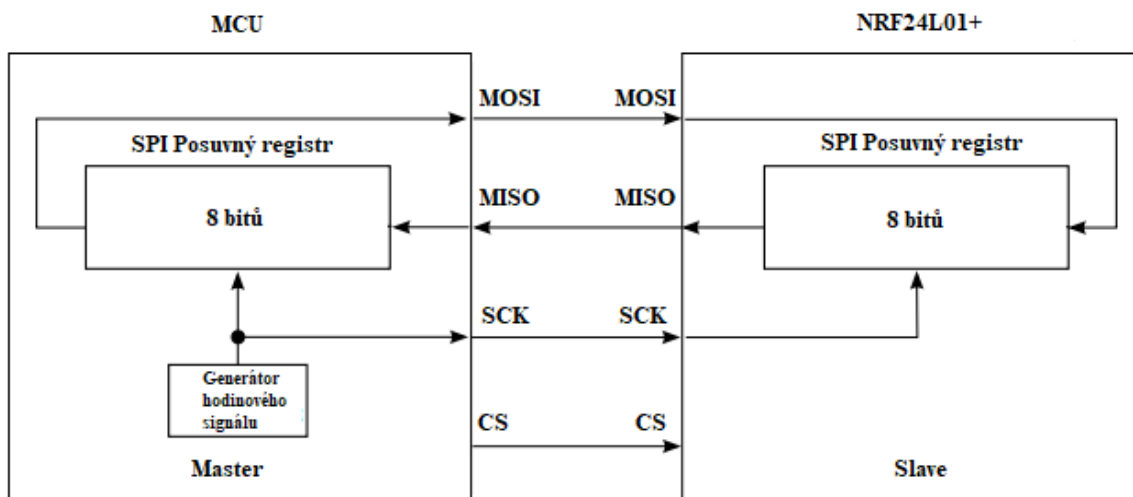
Tento režim se stane aktivní pro přenos paketů. Pokud TX FIFO fronta není prázdná, modul zůstává v tomto režimu, než budou poslány všechny pakety. Jakmile jsou, modul se přepne do pohotovostního režimu jedna.[16]

2.6 Technické parametry

- Frekvenční pásmo: 2,4 GHz
- Maximální přenosová rychlost: 2 Mb/s
- Maximální dosah: 100m
- Typ modulace: GFSK
- Topologie: 1:6 Hvězda

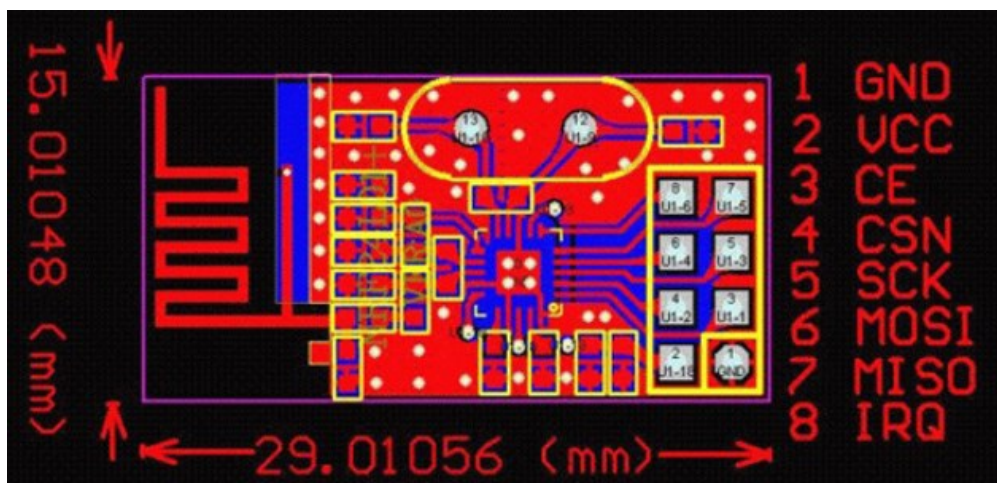
2.7 Propojení bezdrátového modulu s MCU

K tomu, aby modul mohl fungovat je potřeba ho propojit s MCU jednotkou.



Obr. 16. Blokové schéma komunikace mezi Master a Slave zařízením

Modul s MCU komunikuje přes SPI rozhraní tedy sériové periferní rozhraní. To funguje tak, že master nastaví logickou nulu na pinu CS (Chip Select) tomu zařízení, se kterým bude komunikovat a bude jeho slave obvod. Dále začne master generovat hodinový signál a při každé periodě se začnou přenášet data, tedy jeden bit z pinu MOSI (Master Out Slave In) na podřízené zařízení a ve stejný okamžik také bit z pinu MISO (Master In Slave Out). Pokud jsou data poslána a komunikace již nemusí probíhat, master přestane vysílat hodinový signál a nastaví na pinu CS logickou jedničku.[18]



Obr. 17. Osazení pinů NRF24L01+ [19]

Bezdrátový modul NRF24L01+ má na konektoru vyvedeny následující signály:

- GND
Napájecí a signálová zem
- VCC
K napájení modulu je potřeba 3,3 V.
- CE (Chip Enable)
Pokud je nastaven na logickou jedničku, modul buď vysílá, nebo přijímá, podle toho zda je nastaven jako přijímač či vysílač.
- CSN (Chip Select Not)
Pokud je nastaven na logickou nulu, modul reaguje na SPI příkazy.
- MOSI (Master Out Slave In)
Posílá data z mikropočítače do zařízení.
- MISO (Master In Slave Out)
Posílá data ze zařízení do mikropočítače.
- IRQ (Interrupt ReQuest)
Tento signál se nastaví do logické jedničky, když některá ze tří masek přerušení bude logická jednička

2.8 Tvar paketu Enhanced ShockBurst

Paket se skládá z těchto pěti částí:

- Preamble
Je to bitová sekvence používaná pro synchronizaci demodulátoru přijímače s příchozím datovým tokem a má velikost 1 bajt.[16]
- Adresa
Toto je adresa pro přijímač. Adresa zajišťuje, že je paket detekován a přijat správným přijímačem, což zabraňuje náhodné komunikaci mezi vícero zařízeními nRF24L01+.[16]

- PCF

Skládá se z 9 bitů, kde 6 bitů zahrnuje velikost užitečných dat a 2 bity PID (Process identifier), které je ke zjištění, zda přijatý paket je nový nebo opakovaný přenos. A poslední bit patří NAF, tento příznak se používá pouze při použití funkce automatického potvrzování. Nastavení příznaku na úroveň high, pověří přijímač, aby paket nebyl automaticky potvrzen.[16]

- Užitečná data

Je to uživatelsky definovaný obsah paketu. Může mít maximální velikost až 32 bajtů. Jsou dvě možnosti pro manipulaci s užitečnými daty: statické a dynamické[16]

- CRC

Je povinný mechanismus pro zjišťování chyb v paketu. Velikost je buď 1 nebo 2 bajty. Jeho výpočet se provádí přes adresu, PCF a užitečná data.[16]



Obr. 18. Tvar paketu Enhanced ShockBurst [16]

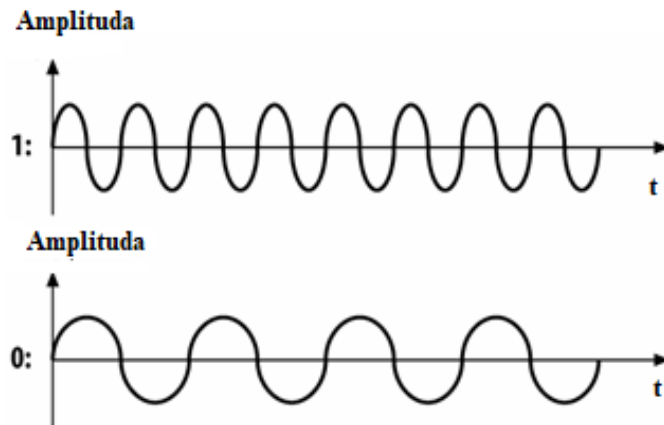
2.9 GFSK modulace

Tento typ modulace je použit v bezdrátovém modulu NRF24L01+, je to frekvenční binární modulace, je to forma FSK (Klíčování frekvenčním posunem) modulace, kde se přenáší digitální informace pomocí diskretních změn frekvence nosné vlny. Modulátor GFSK (Gaussovské klíčování posunem frekvencí) se liší od jednoduchého modulátoru s frekvenčním posunem, že prochází Gaussovým filtrem, aby se přechody uhladily a omezila se spektrální šířka.[20]

Dělí se na:

- 2-stupňová GFSK (2GFSK)

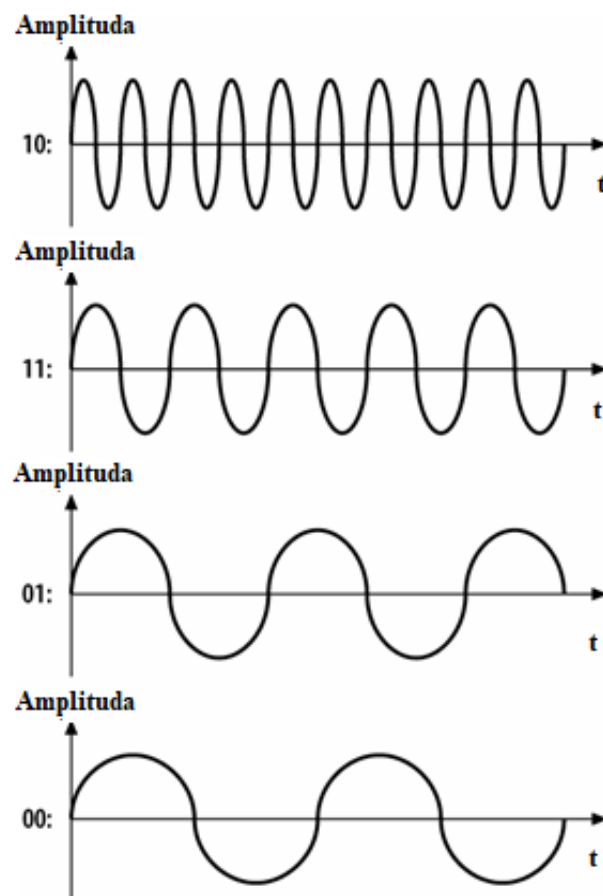
Používají se dvě různé frekvence v závislosti na tom zda přenášený bit je logická nula nebo jednička. Při vysílání signálu logické jedničky se zvýší nosná frekvence o určitou hodnotu.[20]



Obr. 19. Rozdíl frekvencí mezi logickou 1 a 0 [20]

- 4-stupňová GFSK (4GFSK)

Funguje na stejném principu jako 2GFSK, ale se čtyřmi symboly místo dvou. Každý ze čtyř symbolů (00, 01, 10, 11) odpovídají jiné disktrétní frekvenci a proto 4GFSK vysílá dvakrát tolik dat za stejnou dobu, naproti tomu však vyžaduje složitější vysílače a přijímače.[20]



Obr. 20. Rozdíl frekvencí mezi symboly [20]

3 VÝVOJOVÝ KIT S KL25Z

Školní vývojový kit použitý v praktické části, byl vytvořen pro mikropočítač FDRM-KL25Z vyráběný firmou NPX.[21]

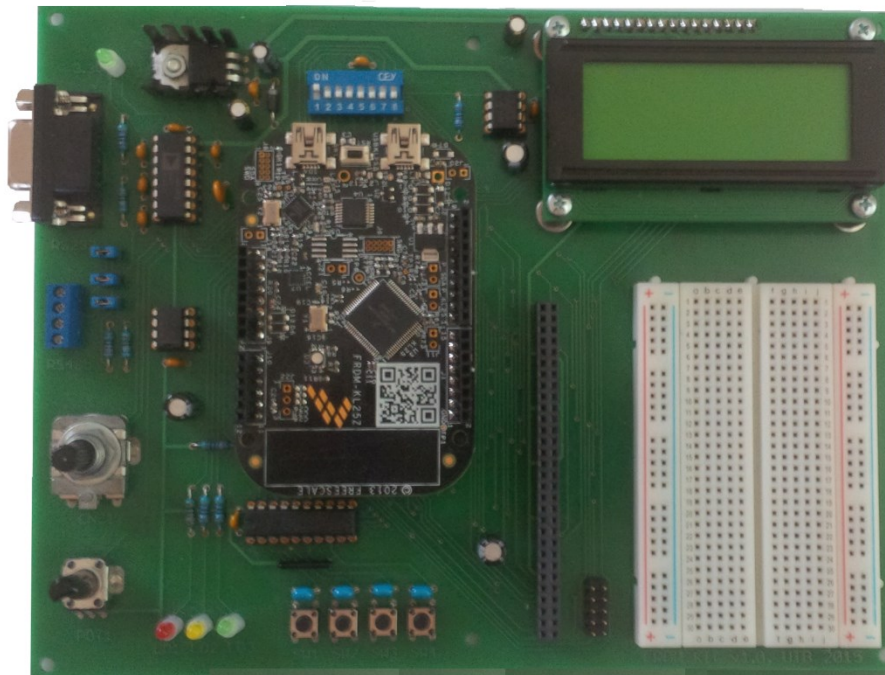
Mikropočítač má tyto následující parametry:

- 32-bitový MCU s jádrem Cortex-M0
- 128 kB Flash paměti a 16 kB paměti SRAM
- 16-bitový A/D převodník a 12-bitový D/A převodník
- Generátor hodinového signálu FLL a PLL
- Šestkrát časovač TMP 1 a dvakrát časovač TMP 2
- Systémový časovač, časovač s nízkou spotřebou
- Dvě 8-bitové sériové periferní rozhraní
- Jeden nízkopříkonový modul UART a dva standardní UART moduly
- Dva I2C moduly
- Akcelerometr MMA8451Q
- RGB LED
- Teplotní senzor LM75AD
- 48 MHz frekvence CPU

Školní vývojový kit obsahuje:

- Tři LED
- Čtyři tlačítka
- Potenciometr
- Rotační enkodér
- RS485 rozhraní
- RS232 rozhraní
- LCD displej

- Nepájivé pole
- Obvod reálného času
- EEPROM



Obr. 21. Školní vývojový kit s mikropočítačem KL25Z

II. PRAKTICKÁ ČÁST

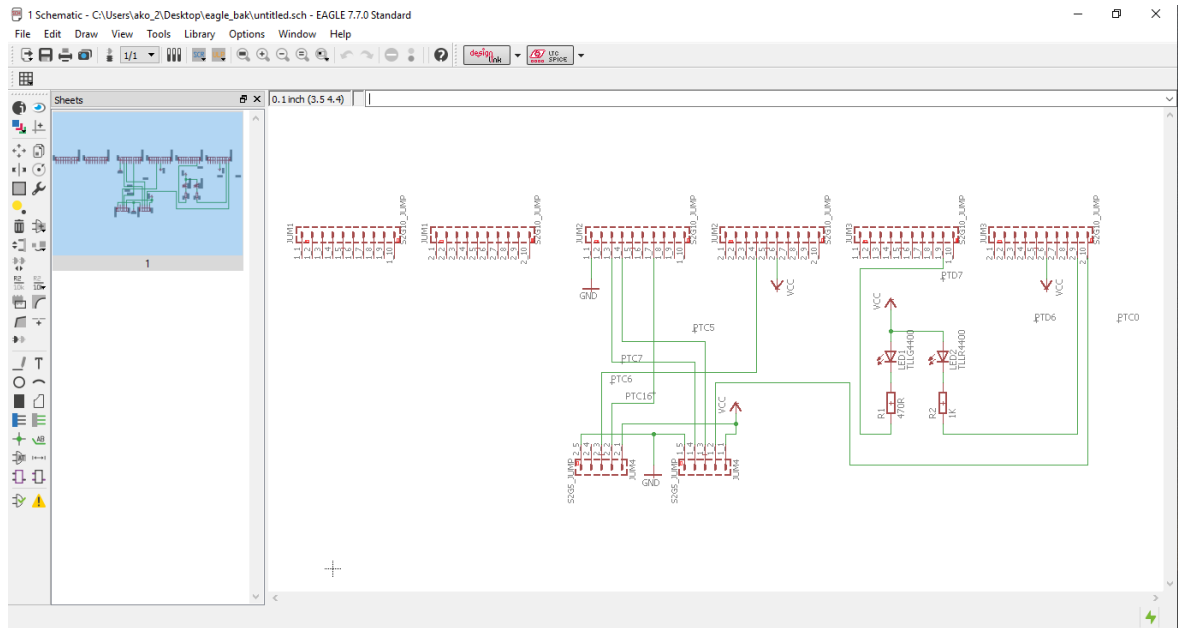
4 PROPOJENÍ KL25Z S NRF24L01+

Aby mohla probíhat komunikace mezi MCU a bezdrátovým modulem, bylo potřeba propojit jednotlivé piny modulu s vhodnými piny na MCU portu. Viz Tabulka 3.

Tab. 3. Propojení pinů MCU portu a bezdrátového modulu

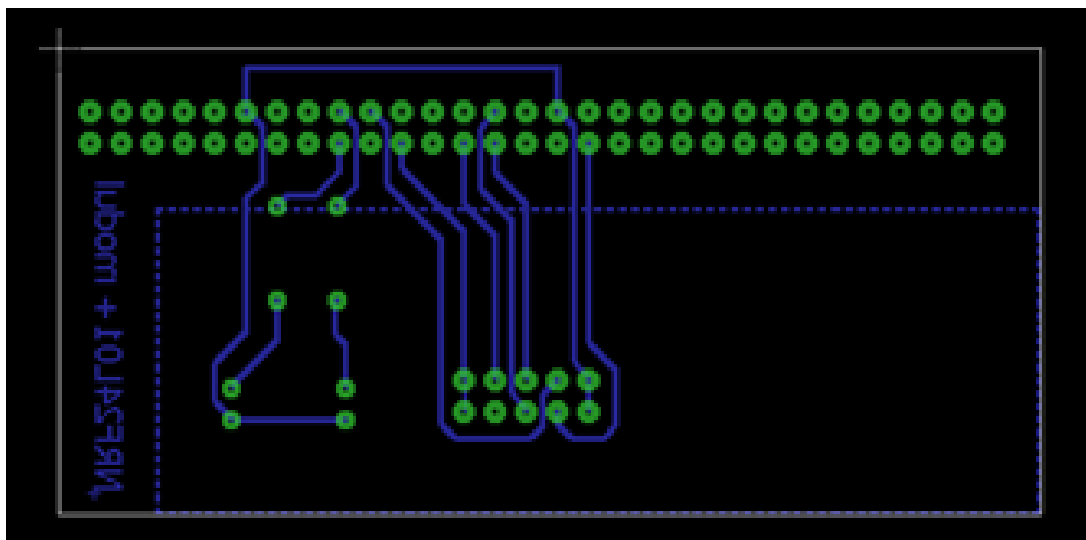
MCU signál	Pin	NRF24L01+ signál	Pin
GND	30	GND	1
3V	29, 49	VCC	2
PTC16	28	CSN	3
PTC17	41	CE	4
PTC5/SCK	34	SCK	5
PTC6/MOSI	33	MOSI	6
PTC7/MISO	36	MISO	7

Pro vytvoření desky plošných spojů byl použit editor plošných spojů Eagle 7.7.0, tento program lze zdarma stáhnout z Internetu, existují i zpoplatněné verze, které například umožňují neomezenou plochu desky plošných spojů. V programu Eagle se vytvoří nový projekt, do kterého se přidá nové schéma pomocí File->New->Schematic. Před samotným návrhem desky plošných spojů je potřeba navrhnout schéma zapojení zahrnující všechny potřebné komponenty.

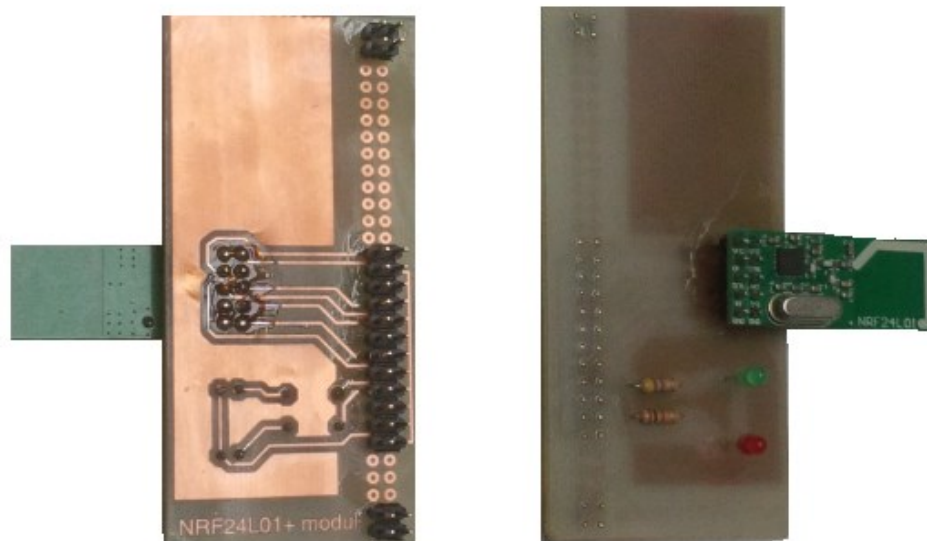


Obr. 22. Schéma propojení modulu s kitem v programu Eagle

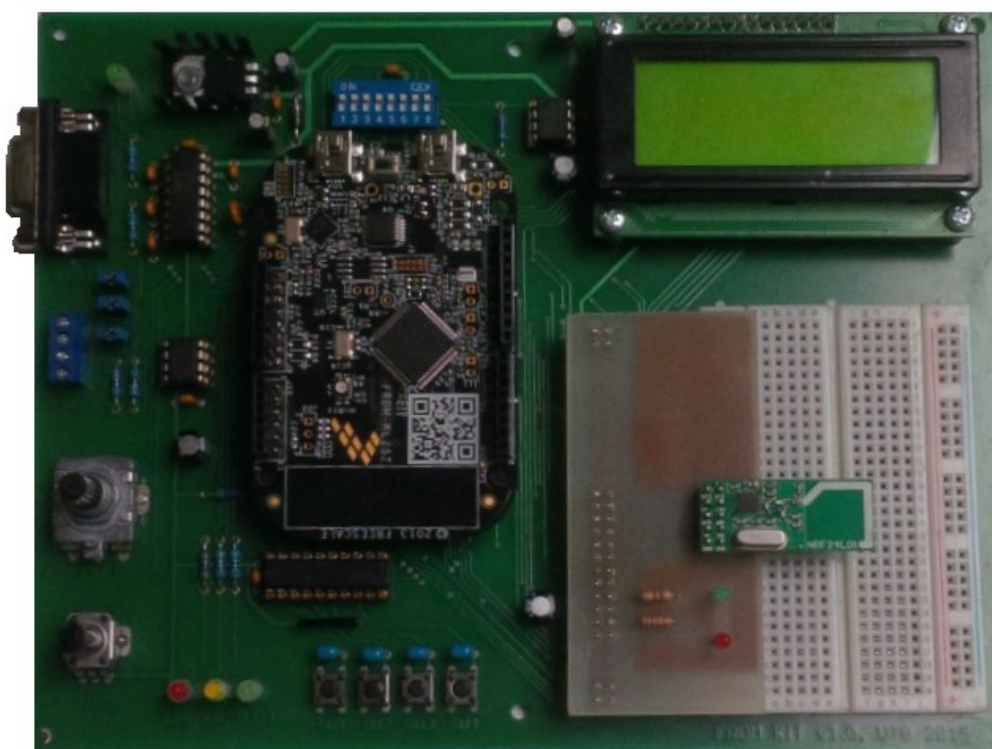
Bylo potřeba importovat knihovnu PaJa_konektory.lbr, která obsahovala řadové kolíky dvojitě 2x10, ty jsou použity třikrát, aby student mohl zapojit desku plošných spojů do MCU portu jen jedním způsobem. Dále byly z knihovny použity řadové kolíky dvojitě 2x5, do kterých se zasune NRF24L01+ modul. Nakonec schéma obsahuje dvě LED pro indikaci spolu s předřadnými rezistory. Napájení a zem u NRF24L01+ u obou propojíme mezi dva piny.



Obr. 23. Navrhnutá deska plošných spojů



Obr. 24. Plošný spoj s NRF24L01+



Obr. 25 Vývojový kit s vyrobeným bezdrátovým modulem

5 PROGRAMOVÁ KNIHOVNA PRO OBSLUHU MODULU

Podpůrná programová knihovna byla vytvořena v integrovaném vývojovém prostředí Kinetic Design Studio. Knihovna obsahuje dva soubory. V hlavičkovém souboru se nachází prototypy funkcí a mapování jednotlivých registrů, druhý soubor obsahuje funkce k obsluze bezdrátového komunikačního modulu.

Knihovna obsahuje funkce:

- 1) void SPI_Init(void)
- 2) static uint8_t SPI_send(uint8_t value)
- 3) void WriteRegister(uint8_t reg, uint8_t value)
- 4) void WriteRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)
- 5) void ReadRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)
- 6) void WriteRead(uint8_t value)
- 7) void Write(uint8_t value)
- 8) uint8_t GetStatus(void)
- 9) void ResetStatus(uint8_t flags)
- 10) void Transmit_Payload(uint8_t *payload, uint8_t payloadSize)
- 11) void Receive_Payload(uint8_t *payload, uint8_t payloadSize)

5.1 Funkce void SPI_Init(void)

Prototyp funkce:

```
void SPI_Init(void)
```

Funkci není předán žádný parametr:

Tato funkce nejprve povolí hodinový signál pro SPI0 a pro PORT C. Dále se nastaví piny MCU portu na specifické funkce pinů podle datového listu pro mikropočítač KL25Z. Piny pro signál CE a CSN jsou také nastaveny jako výstupní piny a inicializovány. Pro inicializaci SPI komunikace je potřeba nastavit tyto registry:

- SPI control register 1 (SPIx_C1)

Kde je nejdříve vynulován třetí bit a tím se nastaví CPHA na hodnotu 0, tedy hodnota MISO a MOSI bude vzorkována při každé náběžné hraně hodinového signálu. CPOL je defaultně nastaven na hodnotu 0, tedy klidová úroveň hodinového signálu je logická 0. Dále nastavíme SPI modul jako master zařízení a povolíme SPI systém. Sériové odesílání dat začíná s nejvíce významným bitem.

- SPI baud rate register (SPI_xBR)

V tomto registru nastavíme rychlost přenosu dat po sběrnici a to na 1 Mb/s.

```
void SPI_Init(void) {
```

```
    SIM_SCGC4 |= SIM_SCGC4_SPI0_MASK;
    SIM_SCGC5 |= SIM_SCGC5_PORTC_MASK|SIM_SCGC5_PORTD_MASK;
```

```
    PORTC->PCR[CSN] = PORT_PCR_MUX(1);
    PORTC->PCR[CE] = PORT_PCR_MUX(1);
    PORTC->PCR[SCK] = PORT_PCR_MUX(2);
    PORTC->PCR[MOSI] = PORT_PCR_MUX(2);
    PORTC->PCR[MISO] = PORT_PCR_MUX(2);
```

```
    PTC->PDDR |= (1 << CSN);
    PTC->PDDR |= (1 << CE);
```

```
    SPI0_C1 &= ~SPI_C1_CPHA_MASK;
    SPI0_C1 = SPI_C1_MSTR_MASK ;
    SPI0_BR = (SPI_BR_SPPR(0x02) | SPI_BR_SPR(0x02));
    SPI0_C1 |= SPI_C1_SPE_MASK;
```

```
    CSN_HIGH();
    CE_LOW();
```

```
}
```

5.2 Funkce static uint8_t SPI_send(uint8_t value)

Prototyp funkce:

```
static uint8_t SPI_send(uint8_t value)
```

Funkci je předán jeden parametr:

- 1) Celé kladné 8-bitové číslo

Funkce vrací kladné 8-bitové číslo, které bylo přečteno z SPI datového registru.

Tato funkce posílá bajt po SPI sběrnici, kde první while smyčka zajišťuje, že se vyprázdní data v transmit bufferu, druhá while smyčka neskončí, dokud data nebudou dostupná ve receive bufferu.

```
static uint8_t SPI_send(uint8_t value) {  
  
    uint8_t ch;  
    while(!(SPI0_S & SPI_S_SPTEF_MASK ));  
    SPI0_D = value;  
  
    while(!(SPI0_S & SPI_S_SPRF_MASK ));  
    ch = SPI0_D;  
  
    return ch;  
  
}
```

5.3 Funkce void WriteRegister(uint8_t reg, uint8_t value)

Prototyp funkce:

```
void WriteRegister(uint8_t reg, uint8_t value)
```

Funkci jsou předávány dva parametry:

- 1) Celé kladné 8-bitové číslo představující registr, který chceme využít
- 2) Celé kladné 8-bitové číslo představující hodnotu, jakou chceme nastavit

Nejdříve se nastaví CSN na logickou nulu, aby mohly zařízení, komunikovat po sběrnici. Tato funkce tedy přistupuje k danému registru a nastavuje potřebné funkce. Po dokončení komunikace po sběrnici se signál CSN může vrátit do původního stavu, nakonec zpoždění 10 mikrosekund.

```
void WriteRegister(uint8_t reg, uint8_t val) {  
    CSN_LOW();  
    SPI_send(W_REGISTER|reg);  
    SPI_send(val);  
    CSN_HIGH();  
    delay_us(10);  
}
```

5.4 Funkce void WriteRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)

Prototyp funkce:

```
void WriteRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)
```

Funkci jsou předávány tři parametry:

- 1) Celé kladné 8 bitové číslo představující registr, který chceme použít
- 2) Ukazatel na pole bufferu
- 3) Velikost bufferu v bajtech

Tato funkce slouží k zapsání více bajtů do SPI sběrnice, kdy signál CSN je uveden do stavu, že probíhá komunikace po sběrnici. Cyklem for je obstaráno posílání bajtů po sběrnici, podle velikosti bufferu.

```
void WriteRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize) {  
    CSN_LOW();  
    SPI_send(W_REGISTER|reg);  
    for(int i=0;i<bufferSize;i++) {  
        SPI_send(buffer[i]);  
    }  
    CSN_HIGH();  
    delay_us(10);  
}
```

5.5 Funkce void ReadRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)

Prototyp funkce:

```
void ReadRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize)
```

Funkci jsou předávány tři parametry:

- 1) Celé kladné 8bitové číslo představující registr, který chceme číst
- 2) Ukazatel na pole bufferu
- 3) Velikost bufferu v bajtech

Tato funkce slouží ke čtení více bajtů ze sběrnice a to tak, že cyklem for posíláme bajty a hned si je ukládáme.


```
void ReadRegisterData(uint8_t reg, uint8_t *buffer, uint8_t bufferSize) {
    CSN_LOW();
    SPI_send(R_REGISTER|reg);
    for(int i=0;i<bufferSize;i++) {
        buffer[i] = SPI_send(buffer[i]);
    }
    CSN_HIGH();
    delay_us(10);
}
```

5.6 Funkce void WriteRead(uint8_t value)

Prototyp funkce:

```
void WriteRead(uint8_t value)
```

Funkci je předáván jeden parametry:

- 1) Celé kladné 8 bitové číslo představující hodnotu k zápisu na sběrnici

Tato funkce slouží k zápisu, do posuvného SPI registru a jeho načtení zpět z posuvného registru. Načtenou hodnotu, pak vrací.

```
uint8_t WriteRead(uint8_t value) {
    CSN_LOW();
    val = SPI_send(val);
    CSN_HIGH();
    delay_us(10);
    return val;
}
```

5.7 Funkce void Write(uint8_t value)

Prototyp funkce:

```
void Write(uint8_t value)
```

Funkci je předáván jeden parametr:

- 1) Celé kladné 8 bitové číslo představující hodnotu k zápisu na sběrnici

Tato funkce zapíše bajt do sběrnice, ale už ho nenačítá zpět.

```
void Write(uint8_t value) {
    CSN_LOW();
    SPI_send(value);
}
```

```
    CSN_HIGH();
    delay_us(10);
    return value;
}
```

5.8 Funkce uint8_t GetStatus(void)

Prototyp funkce:

```
uint8_t GetStatus(void)
```

Funkci není předán, žádný parametr.

Využívá funkce WriteRead a zapisuje do sběrnice příkaz NOP, se kterým nám zařízení vrátí hodnotu status registru.

```
uint8_t GetStatus(void) {
    return WriteRead(NOP);
}
```

5.9 Funkce void ResetStatus(uint8_t flags)

Prototyp funkce:

```
void ResetStatus(uint8_t flags)
```

Funkci je předán jeden parametr:

- 1) Celé kladné 8 bitové číslo představující hodnotu příznaků

Tahle funkce zpracovává, příznaky, které mohou nastat při přijímání nebo odesílání dat. A to příznak, že byla přijata data přijímačem, dále když vysílač přijme ACK paket od přijímače a příznak překročení maximálního počtu přeposlání vysílačem.

```
void ResetStatus(uint8_t flags) {
    delay_us(10);
    CSN_LOW();
    delay_us(10);
    WriteRegister(RF_STATUS, flags);
    delay_us(10);
    CSN_HIGH();
    delay_us(10);
}
```

5.10 Funkce void Transmit_Payload(uint8_t *payload, uint8_t payloadSize)

Prototyp funkce:

```
void Transmit_Payload(uint8_t *payload, uint8_t payloadSize)
```

Funkci jsou předány dva parametry:

- 1) Ukazatel na pole payload
- 2) Velikost payload v bajtech

Tato funkce nejprve vyhodí stará data a pošle payload do Tx FIFO fronty k poslání. Signál CE se nastaví do logické jedničky a během čekání po dobu dvaceti mikrosekund se data pošlou.

```
void Transmit_Payload(uint8_t *payload, uint8_t payloadSize) {  
    Write(FLUSH_TX);  
    WriteRegisterData(W_TX_PAYLOAD, payload, payloadSize);  
    CE_HIGH();  
    delay_us(20);  
    CE_LOW();  
}
```

5.11 Funkce void Receive_Payload(uint8_t *payload, uint8_t payloadSize)

Prototyp funkce:

```
void Receive_Payload(uint8_t *payload, uint8_t payloadSize)
```

Funkci jsou předány dva parametry:

- 1) Ukazatel na pole payload
- 2) Velikost payload v bajtech

Signál CE musí být nastaven na logickou nulu, aby mohl další příkaz přijmout do Rx FIFO fronty payload, který uloží do bufferu.

```
void Receive_Payload(uint8_t *payload, uint8_t payloadSize) {  
    CE_LOW();  
    ReadRegisterData(R_RX_PAYLOAD, payload, payloadSize);  
    CE_HIGH();  
}
```

5.12 Funkce pro nastavování výstupních logických hodnot

Nakonec jsou tu jednoduché funkce, které buď nastavují logickou nulu, nebo jedničku na piny CE a CSN. Kde jsou využívány registry k ovládní pinů GPIO, tedy PCOR (Port Clear Register) a PSOR (Port Set Register).

```
void CSN_LOW(void) {  
    PTC->PCOR |= (1 << CSN);  
}  
  
void CSN_HIGH(void) {  
    PTC->PSOR |= (1 << CSN);  
}  
  
void CE_LOW(void) {  
    PTC->PCOR |= (1 << CSN);  
}  
  
void CE_HIGH(void) {  
    PTC->PSOR |= (1 << CSN);  
}
```

6 TVORBA UKÁZKOVÉHO APLIKACE

Ukázková aplikace slouží k ukázce funkčnosti vytvořené programové knihovny. Jsou použity dva školní vývojové kity a dva NRF24L01+ moduly. Moduly spolu komunikují, což indikují dvě LED, dále na LCD obrazovce vývojového kitu, kde je přijímač je zobrazována teplota a vlhkost naměřená z vývojového kitu kde je vysílač.

Inicializační kód je stejný pro přijímač a vysílač. Můžeme nastavit číslo komunikačního kanálu, těch může být až 126. Dále je potřeba nastavit velikost užitečných dat, tedy dat, které chceme přenášet a to od 0 do 32 bajtů. Stejná musí být i adresa zařízení. Aby nebylo potřeba vypojovat a zapojovat vývojový kit do počítače, když dojde k přerušení spojení a je ho potřeba resetovat pomocí tlačítka na vývojovém kitu, dojde k vymazání starých dat.

```
Write(FLUSH_TX);  
Write(FLUSH_RX);
```

Dále se nastaví v výstupní vysílací výkon a přenosová rychlost. Ve RX FIFO frontě datové trubice 0, se vytvoří počet bajtů podle zadané velikosti užitečných dat a také se nastaví číslo kanálu.

```
WriteRegister(RF_RF_SETUP, SETUP_RF_PWR_0 | SETUP_RF_DR_1000);  
WriteRegister(RX_PW_P0, PAYLOAD_SIZE);  
WriteRegister(RF_RF_CH, CHANNEL_NUMBER);
```

Je potřeba také nastavit vysílací a přijímací adresu a ty se musí rovnat, povolí se datová trubice 0. Jako poslední se smažou příznaky.

```
WriteRegisterData(RX_ADDR_P0, (uint8_t*)address, sizeof(address));  
WriteRegisterData(TX_ADDR, (uint8_t*)address, sizeof(address));  
WriteRegister(RF_EN_RXADDR, EN_RXADDR_ERX_P0)
```

```
ResetStatus(STATUS_RX_DR | STATUS_TX_DS | STATUS_MAX_RT);
```

V dalších podkapitolách bude popsán program pro vysílač a přijímač

6.1 Program pro vysílač

Nejdříve se povolí automatické potvrzení ACK. Časové rozmezí a počet opakování zaslání paketu dat, pokud po určitém počtu posláních, není přijat ACK paket. Dále se zapne zařízení a nastaví se jako primární vysílač PTX. Pro detekci chyb při přenosu je nastaveno dvou bajtové CRC.

```
WriteRegister(RF_EN_AA, EN_AA_ENAA_P0);
WriteRegister(RF_SETUP_RETR, SETUP_RETR_ARC_15);
WriteRegister(RF_CONFIG, EN_CRC|CRCO|PWR_UP|PRIM_TX);
CE_LOW();
```

Dále už následuje nekonečná smyčka, kde blikáním červené diody indikujeme, že se data posílají. Jsou využity funkce, které snímají teplotu a vlhkost a vrací jejich hodnotu jako celé kladné 32 bitové číslo.

```
temperature = MeasureTemperature();
humidity = MeasureHumidity();
```

SPI sběrnice přenáší data po jednu bajtu, takže bylo potřeba ho rozdělit na čtyři kladná 8 bitová čísla pomocí bitového posunu.

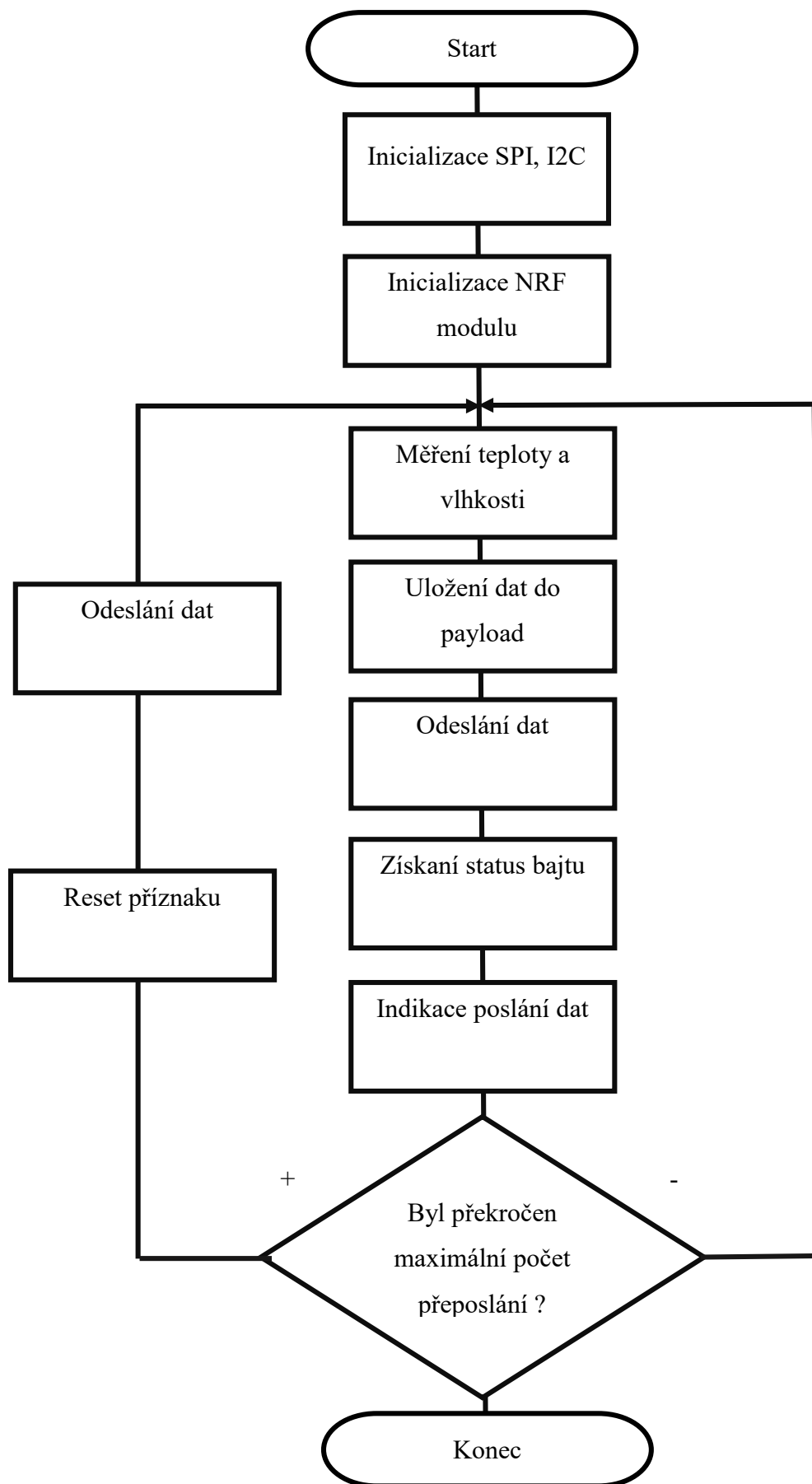
```
payload[0] = temperature>>24;
payload[1] = temperature>>16;
payload[2] = temperature>>8;
payload[3] = temperature;
```

```
payload[4] = humidity>>24;
payload[5] = humidity>>16;
payload[6] = humidity>>8;
payload[7] = humidity;
```

Nakonec využijeme funkci, která vrací stav v jakém se zařízení nachází. Pomocí podmínky testujeme, zda se nachází ve stavu překročení počtu přeposlaných dat, když ano tak je potřeba status vymazat, aby zařízení mohlo dál posílat data.

```
statusRF = GetStatus();

if (statusRF & (STATUS_MAX_RT)) {
    ResetStatus(STATUS_MAX_RT);
    Transmit_Payload(payload, sizeof(payload));
}
```



Obr. 26. Zjednodušený vývojový diagram pro vysílací cyklus

6.2 Program pro přijímač

Nejdříve se zapne zařízení, nastaví se jako primární přijímač PRX, pro detekci chyb je použit dvou bajtový CRC. Dále už následuje nekonečná smyčka, kde se na začátku vrací stavový bajt, jenž poskytuje informaci v jakém stavu se zařízení nachází, pokud bude status bajt roven příznaku, že byly přijaty data do RX FIFO fronty. Funkce přijme RX payload do bufferu. Pak se provede reset všech příznaků.

```
Receive_Payload(payload,sizeof(payload));  
ResetStatus(STATUS_RX_DR|STATUS_TX_DS|STATUS_MAX_RT);
```

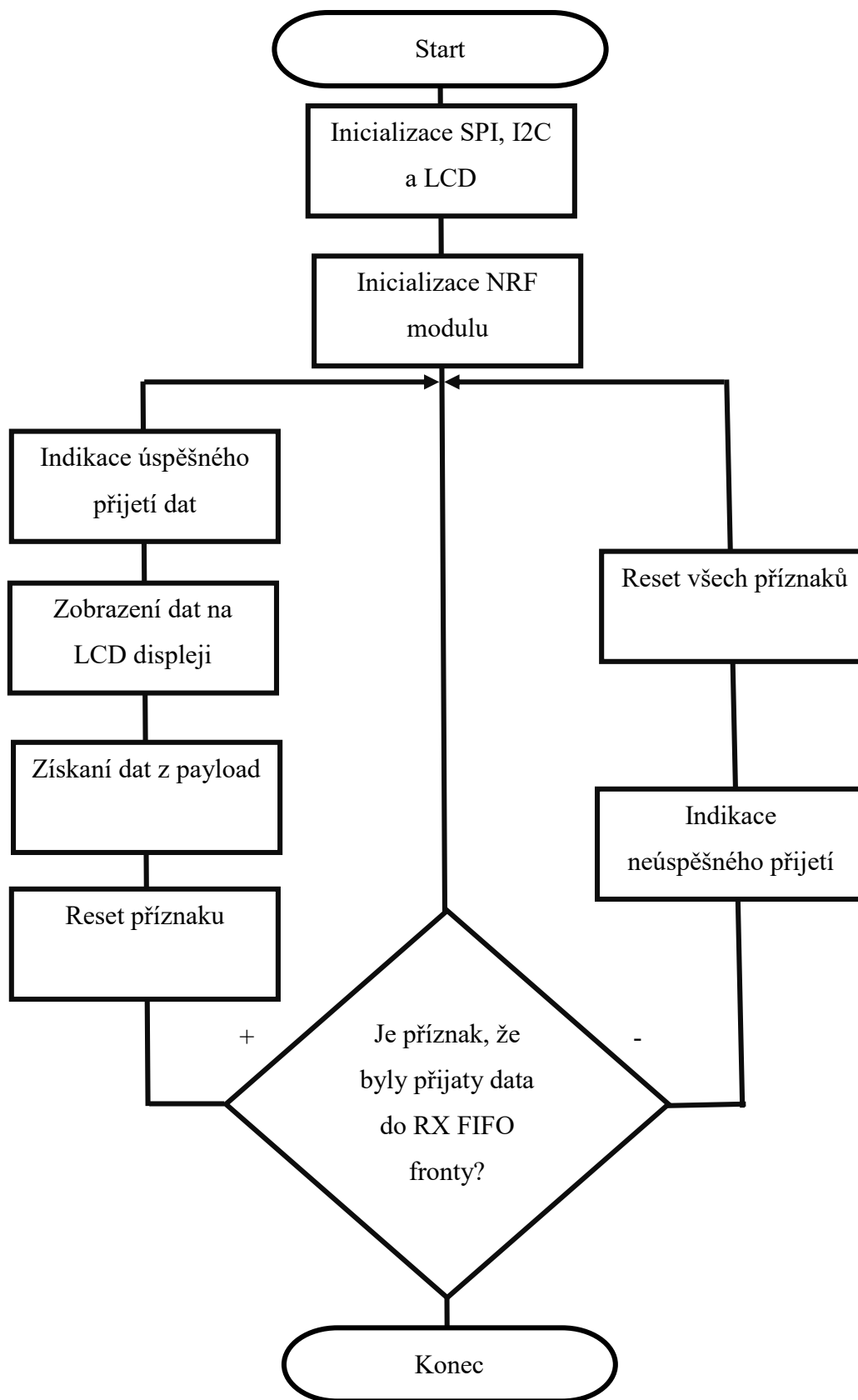
Přijátá data je potřeba ze čtyř kladných 8 bitových čísel převést znovu na 32 bitové, aby tyto naměřené hodnoty šly zobrazit na LCD displeji.

```
uint32_t vysledek_teploata = payload[0]<<24 | payload[1]<<16 | payload[2]<<8 |  
payload[3];  
uint32_t vysledek_vlhkost = payload[4]<<24 | payload[5]<<16 | payload[6]<<8 |  
payload[7];
```

Dále je potřeba zobrazit naměřená data z vysílače. Proběhne reset displeje, je potřeba funkce printf, která zapisuje formátovaný výstup stejně jako funkce printf() akorát výstup jde do paměti kam ukazuje str. A funkce LCD_puts vypíše hodnoty uložené v bufferu.

```
LCD_clear();  
printf(buffer, "H: %d %%", vysledek_vlhkost);  
LCD_puts(buffer);  
printf(buffer, "T: %d.%d C", vysledek_teploata/10, vysledek_teploata%10);  
LCD_set_cursor(2,1);  
LCD_puts(buffer);
```

Nakonec indikujeme zelenou LED, že data byla přijata. Pokud status bajt, nebude roven danému příznaku, že byla data přijata do RX FIFO fronty, proběhne indikace červenou LED a reset všech příznaků.



Obr. 27. Zjednodušený vývojový diagram pro přijímací cyklus

6.3 Ověření funkce modulů

Na obou modulech byl nastaven maximální vysílací výkon 0 dBm a přenosová rychlost 1 Mb/s a bylo zjištěno, že moduly komunikují:

- přes dvě zděné zdi
- přes patro domu (výškový rozdíl 4 metry)
- stojí-li v cestě člověk
- z jednoho domu do druhého (cca 30 metrů)
- ve volném prostranství maximální dosah cca 80 metrů

Na obou modulech byl nastaven vysílací výkon a to -6 dBm a přenosová rychlost 1 Mb/s a bylo zjištěno, že moduly komunikují:

- přes dvě zděné zdi
- přes patro domu spojení vypadává
- stojí-li v cestě člověk
- ve volném prostranství maximální dosah cca 60 metrů

Nekomunikuje:

- mezi dvěma domy spojení vypadává

Na obou modulech byl nastaven vysílací výkon a to -12 dBm a přenosová rychlost 1 Mb/s a bylo zjištěno, že moduly komunikují:

- přes dvě zděné zdi
- přes patro domu
- stojí-li v cestě člověk
- ve volném prostranství maximální dosah cca 40 metrů

Nekomunikuje:

- mezi dvěma domy

Na obou modulech byl nastaven nejmenší vysílací výkon a to -18 dBm a bylo zjištěno, že moduly komunikují:

- přes dvě zděné zdi
- stojí-li v cestě živý předmět
- ve volném prostranství maximální dosah cca 25 metrů

Moduly nekomunikují:


- mezi dvěma domy
- přes patro domu spojení vypadává

7 PREZENTACE PRO PODPORU VÝUKY

Výuková prezentace byla vytvořena pro studenty, aby po jejím nastudování, byli schopni naprogramovat svou vlastní aplikaci a zvládnout obsluhu bezdrátového komunikačního modulu. Prezentace byla vytvořena v programu Microsoft Office PowerPoint 2016. Kompletní vytvořená prezentace je součástí elektronické přílohy práce.

Jak přidat programovou knihovnu do projektu

- Je potřeba zkopírovat soubor nRF24I01.C do Vašeho projektu
- Pro přidání knihovny do Vašeho projektu vložte hlavičkový soubor
`#include "nRF24I01.H"`
- Knihovna obsahuje nízkourovňové a vysokoúrovňové, které slouží k obsluze bezdrátového modulu
- Prototypy funkcí a namapované registry jsou uvedeny v souboru nRF24I01.H
- Kód k jednotlivým funkcím v souboru nRF24I01.C



Obr. 28. Ukázkový snímek prezentace

Prezentace obsahuje následující kapitoly:

- 1) Stručný popis funkce NRF24L01+ modulu
- 2) Jak přidat programovou knihovnu do projektu
- 3) Prototypy jednotlivých funkcí
- 4) Popis jednotlivých funkcí
- 5) Popis ukázkové aplikace
- 6) Vývojové diagramy k ukázkové aplikaci

ZÁVĚR

Teoretická část bakalářské práce obsahuje stručný pohled do historie bezdrátové komunikace a její základní dělení. Dále byly popsány tři bezdrátové technologie pro embedded systémy a to ZigBee, Bluetooth Low Energy a Wi-Fi. Ke každé technologii jsou uvedené určité technické vlastnosti a rozdílnosti. Také je zde podrobněji popsán bezdrátový modul, který je použit v praktické části a to NRF24L01+. Ke konci teoretické části jsou uvedeny technické parametry mikropočítače, který byl v práci použit jako součást vývojového kitu.

V praktické části se zabýváme návrhem propojení bezdrátového komunikačního modulu s výukovým vývojovým kitem s výběrem vhodných pinů. Dále navrhnutím daného schématu a vytvoření desky plošných spojů, na to byl použit program Eagle 7.7.0.

V další části této práce byla potřeba vytvořit podpůrné programové vybavení modulu. Tedy jednak funkce pro inicializaci a komunikaci přes SPI sběrnici a také, aby mohl bezdrátový modul pomocí funkcí přijímat a odesílat data.

Ukázková aplikace u vysílače snímá teplotu a vlhkost, data jsou upravována pro přenos po SPI sběrnici a uživatele informuje indikací LED o tom, že data jsou posílána. Přijímač tyto data zpátky upraví do formy, aby bylo možno je zobrazit na LCD displeji druhého vývojového kitu a také probíhá indikace pomocí LED, zda jsou data přijímána či ne.

Součástí práce bylo také testování dosahu bezdrátového modulu při různých vysílacích výkonech a při různých situacích. Při nastavení největšího možného vysílacího výkonu moduly komunikovaly na dálku cca 80 metrů, pro moduly nebyl problém komunikovat přes zdi i při výškovém rozdílu a dokonce i mezi dvěma domy cca 30 metrů.

Nakonec práce obsahuje prezentaci vytvořenou v programu Microsoft Office PowerPoint 2016 o dané programové knihovně pro předmět Programování mikropočítačů: 2,4 GHz komunikační modul. Po jejímž nastudování by měl být student schopen tento modul ovládat.

SEZNAM POUŽITÉ LITERATURY

- [1] KIM, Haesik. *Wireless communications systems design*. Chichester, West Sussex, United Kingdom: Wiley, 2015. ISBN 9781118610152.
- [2] UYSAL, Murat a Hatef NOURI. *Optical wireless communications — An emerging technology* [online]. Graz, Austria: IEEE, 2014 [cit. 2018-05-17]. 978-1-4799-5601-2. Dostupné z: <https://ieeexplore-ieee-org.proxy.k.utb.cz/document/6876267/>
- [3] *What are radio waves?* [online]. 2012 [cit. 2018-05-17]. Dostupné z: https://www.nasa.gov/directorates/heo/scan/communications/outreach/funfacts/txt_radio_spectrum.html
- [4] *RÁDIO A TELEVÍZNA TECHNIKA* [online]. [cit. 2018-05-17]. Dostupné z: <http://eln.szm.com/pages/rtt.htm>
- [5] FARAHANI, Shahin. *ZigBee wireless networks and transceivers*. Boston: Newnes/Elsevier, c2008. ISBN 0750683937.
- [6] *XBee Pro 60mw Wire Antenna-Series 1* [online]. [cit. 2018-05-17]. Dostupné z: https://www.rhydolabz.com/wireless-zigbee-xbee-c-130_133/xbee-pro-60mw-wire-antennaseries-1-p-564.html
- [7] *ZigBee Specification FAQ* [online]. 2013 [cit. 2018-05-15]. Dostupné z: <https://web.archive.org/web/20130627172453/http://www.zigbee.org/Specifications/ZigBee/FAQ.aspx>.
- [8] GUPTA, Naresh. *Inside Bluetooth Low Energy*. Second Edition. Boston: Artech House, 2016. ISBN 9781630810894.
- [9] *EY-08 BLE Bluetooth 4.0 UART Modul, JDY-08 CC2541* [online]. [cit. 2018-05-16]. Dostupné z: <https://laskarduino.cz/prenos-dat-bezdratovy/230396-ey-08-ble-bluetooth-40-uart-modul-jdy-08-cc2541.html>
- [10] *What is Bluetooth Adaptive Frequency Hopping (AFH)?* [online]. 2015 [cit. 2018-05-16]. Dostupné z: <https://support.honeywellaidc.com/s/article/What-is-Bluetooth-Adaptive-Frequency-Hopping-AFH>
- [11] TOWNSEND, Kevin., Robert. DAVIDSON, AKIBA. a Carles. CUFÍ. *Getting started with Bluetooth low energy: tools and techniques for low-power networking*. Revised First Edition. Sebastopol, CA: O'Reilly, 2014. ISBN 9781491949511.

- [12] BORNSTEIN, Michelle. *Wifi Hotspot: Advantages and Disadvantages of Wifi Hotspots* [online]. Lulu Press, 2015 [cit. 2018-05-17]. ISBN 9781329543874. Dostupné z: https://books.google.cz/books?id=XDmaCgAAQBAJ&printsec=frontcover&hl=cs&source=gbs_atb#v=onepage&q&f=false
- [13] *ESP-07 ESP8266 serial Wi-Fi bezdrátový vysílač modul pro Arduino* [online]. [cit. 2018-05-17]. Dostupné z: <http://www.dx.com/cs/p/esp-07-esp8266-serial-wi-f-wireless-module-w-built-in-antenna-compatible-with-3-3v-5v-for-arduino-400559>
- [14] SHI, Gaotao a Li KEQIU. *Signal interference in WiFi and ZigBee networks*. New York, NY: Springer Berlin Heidelberg, 2016. ISBN 9783319478050.
- [15] GEIER, James T. *Designing and deploying 802.11 wireless networks: a practical guide to implementing 802.11n and 802.11ac wireless networks for enterprise-based applications*. Second edition. Indianapolis, IN: Cisco Press, 2015. Cisco Press networking technology series. ISBN 1587144301.
- [16] *Nordic Semiconductor nRF24L01+* [online]. 2008 [cit. 2018-05-16]. Dostupné z: <https://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P>
- [17] *Arduino 2,4 GHz modul NRF24L01* [online]. [cit. 2018-05-17]. Dostupné z: https://arduino-shop.cz/arduino/892-arduino-wifi-modul-nrf24l01-1420663506.html?gclid=EA1aIQobChMI8sjL5qaM2wIVyU0YCh3tHakvEAQYASABEgI0MfD_BwE
- [18] PINKER, Jiří. *Mikroprocesory a mikropočítače*. Praha: BEN - technická literatura, 2004. ISBN 80-7300-110-1.
- [19] *Raspberry Pi 2 Model B NRF24L01 Pin Connections* [online]. [cit. 2018-05-17]. Dostupné z: <https://raspberrypi.stackexchange.com/questions/48312/raspberry-pi-2-model-b-nrf24l01-pin-connections>
- [20] GAST, Matthew. *802.11 wireless networks: the definitive guide*. 2nd ed. Farnham: O'Reilly, 2005. ISBN 0596100523.
- [21] *NXP Semiconductors. KL25 Sub-Family Reference Manual* [online]. 2014 [cit. 2018-05-17]. Dostupné z: <http://www.nxp.com>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

IOT	Internet of things
LCD	Liquid Crystal Display
IR	Infrared radiation
UV	Ultraviolet
PAN	Personal Area Network
WLAN	Wireless Local Area Network
IEEE	Institute of Electrical and Electronics Engineers
FFD	Full Functional Device
RFD	Reduced Functional Device
BPSK	Binary-Phase Shift Keying
OQPSK	Offset Quadrature Phase Shift Keying
GFSK	Gaussian Frequency Shift Keying
OFDM	Orthogonal Frequency-Division Multiplexing
QAM	Quadrature amplitude modulation
AES	Advanced Encryption Standard
ISM	Industrial, Scientific and Medical
BLE	Bluetooth Low Energy
PTX	Primární vysílač
PRX	Primární přijímač
RX	Režim příjmu
PLL	Phase Locked Loop
FLL	Frequency Locked Loop
TPM	Timer and Pulse Width Modulation
SPI	Serial Peripheral Interface

UART	Universal Synchronous / Asynchronous Receiver and Transmitter
I2C	Two Wire Interface
PLL	Phase Locked Loop
FLL	Frequency Locked Loop
CPU	Central Processing Unit
MCU	MicroController Unit
CE	Chip Enable
CSN	Chip Select Not
SCK	Seriál Clock
MISO	Master In Slave Out
MOSI	Master Out Slave In
IRQ	Interrupt ReQuest
CRC	Cyclic Redundancy Code
EAGLE	Easily Applicable Graphical Layout Editor EAGLE
FIFO	First In First Out
LED	Light Emmiting Diode

SEZNAM OBRÁZKŮ

<i>Obr. 1. Elektromagnetické spektrum [2]</i>	11
<i>Obr. 2. Ukázka amplitudové a frekvenční modulace [4]</i>	13
<i>Obr. 3. Ukázka ZigBee modulu XBCAWIT-001 [6]</i>	14
<i>Obr. 4. Hvězdicová topologie u ZigBee [5]</i>	15
<i>Obr. 5. Mesh topologie u Zigbee [5]</i>	15
<i>Obr. 6. Ukázka BLE modulu EY-08</i>	17
<i>Obr. 7. Broadcast topologie u BLE [10]</i>	18
<i>Obr. 8. Connected topologie u BLE</i>	18
<i>Obr. 9. Ukázka WiFi modulu ESP-07 ESP8266 [12]</i>	20
<i>Obr. 10. Ad-hoc topologie u Wi-fi [14]</i>	21
<i>Obr. 11. Infrastrukturní síť u Wi-fi [14]</i>	21
<i>Obr. 12. Mesh topologie u Wi-fi [14]</i>	22
<i>Obr. 13. Bezdrátový modul nRF24L01+ [16]</i>	23
<i>Obr. 14. Multiceiver u nRF24L01+ [15]</i>	24
<i>Obr. 15. Adresování datových trubic [15]</i>	25
<i>Obr. 16. Blokové schéma komunikace mezi Master a Slave zařízením</i>	27
<i>Obr. 17. Osazení pinů NRF24L01+ [18]</i>	27
<i>Obr. 18. Tvar paketu Enhanced ShockBurst [15]</i>	29
<i>Obr. 19. Rozdíl frekvencí mezi logickou 1 a 0 [19]</i>	30
<i>Obr. 20. Rozdíl frekvencí mezi symboly [19]</i>	30
<i>Obr. 21. Školní vývojový kit s mikropočítačem KL25Z</i>	32
<i>Obr. 22. Schéma propojení modulu s kitem v programu Eagle</i>	35
<i>Obr. 23. Navrhnutá deska plošných spojů</i>	35
<i>Obr. 24. Plošný spoj s NRF24L01+ </i>	36
<i>Obr. 25 Vývojový kit s vyrobeným bezdrátovým modulem</i>	36
<i>Obr. 26. Zjednodušený vývojový diagram pro vysílací cyklus</i>	47
<i>Obr. 27. Zjednodušený vývojový diagram pro přijímací cyklus</i>	49
<i>Obr. 28. Ukázkový snímek prezentace</i>	52

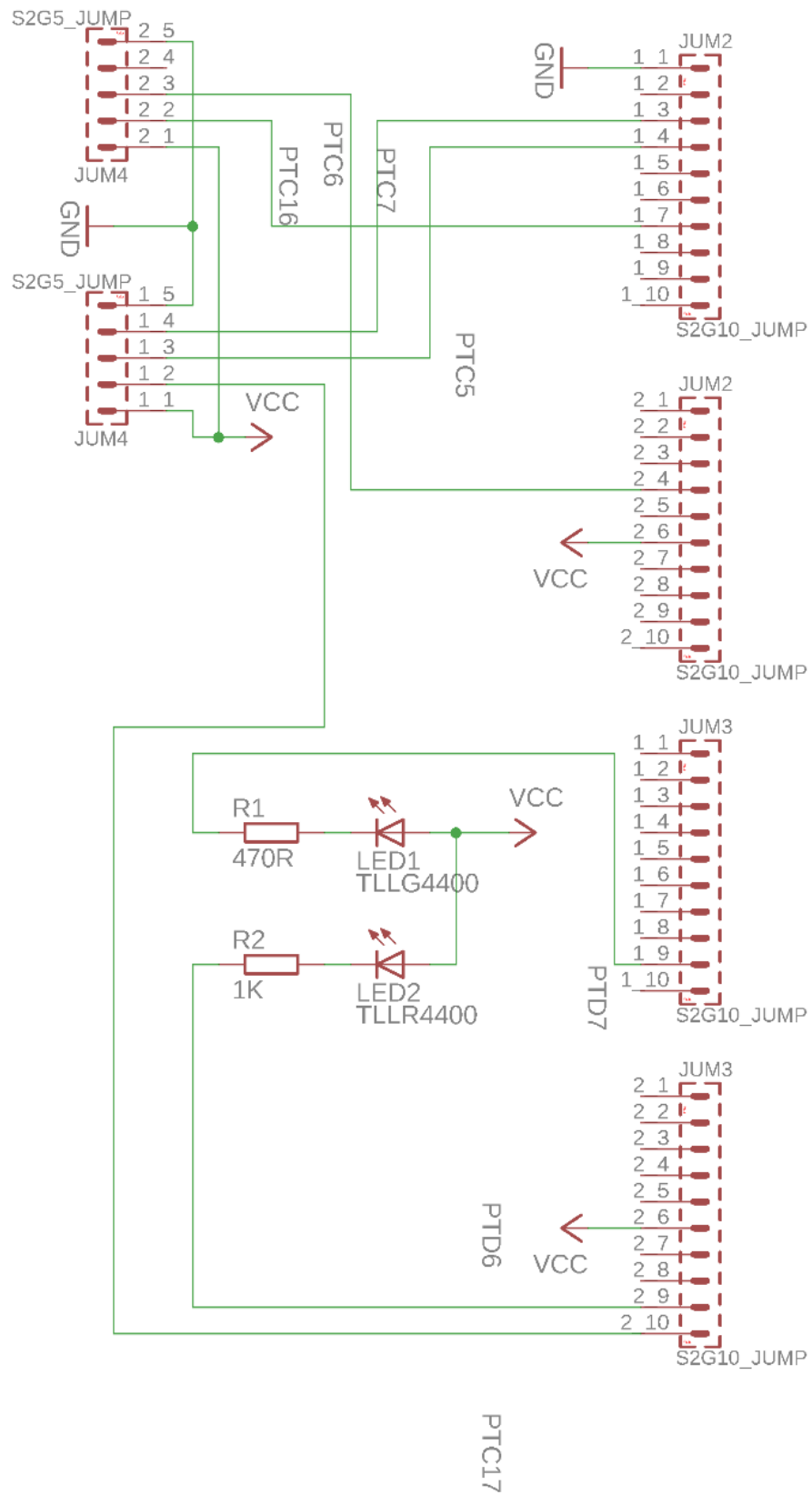
SEZNAM TABULEK

<i>Tab. 1. Spotřeba modulu podle vysílacího výkonu</i>	<i>25</i>
<i>Tab. 2. Spotřeba modulu podle jeho stavu.....</i>	<i>25</i>
<i>Tab. 3. Propojení pinů MCU portu a bezdrátového modulu</i>	<i>34</i>

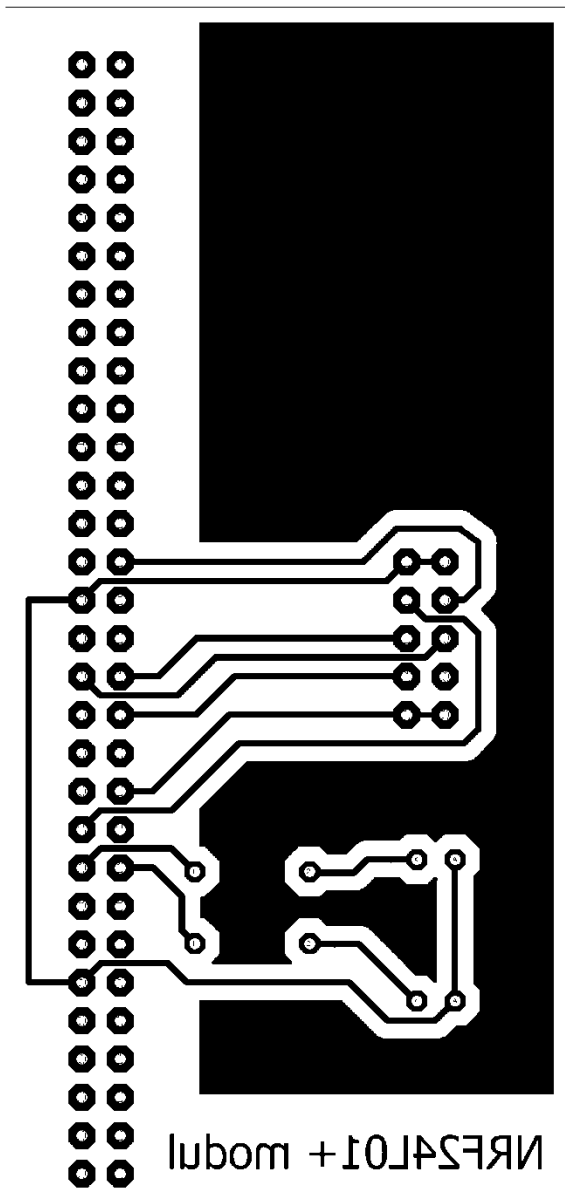
SEZNAM PŘÍLOH

- P I Schéma zapojení
- P II Motiv plošného spoje
- P III Osazovací plán
- P IV CD s podpůrnou programovou knihovnou, ukázkovou aplikací a prezentací pro předmět Programování Mikropočítačů a bakalářskou práci

PŘÍLOHA P I: SCHÉMA ZAPOJENÍ



PŘÍLOHA P II: MOTIV PLOŠNÉHO SPOJE



PŘÍLOHA P III: OSAZOVACÍ PLÁN

