

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2018

Jiří Ježek



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

NÁVRH IOT ZAŘÍZENÍ PRO BEZDRÁTOVOU TECHNOLOGII NB-IOT

DEVELOPMENT OF AN IOT DEVICE FOR NB-IOT TECHNOLOGY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jiří Ježek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jiří Pokorný

BRNO 2018



Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Jiří Ježek

ID: 186523

Ročník: 3

Akademický rok: 2017/18

NÁZEV TÉMATU:

Návrh IoT zařízení pro bezdrátovou technologii NB-IoT

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce bude návrh zařízení využívající technologii NB-IoT (Narrow Band Internet of Things). V teoretické části budou porovnány LPWAN technologie se zaměřením na NB-IoT. V praktické části bude navrženo schéma a DPS (deska plošného spoje). Dále bude zařízení vyrobeno a připraveno pro účely měření. V poslední části student navrhne scénáře pro měření přenosu dat ze zařízení. Scénář pro využití (např.: teplotní senzor, vzdálený odečet spotřeby energií, zařízení pro sledování zásilek), si student zvolí sám. Dle zvoleného scénáře se bude odvíjet vlastní návrh zařízení. K dispozici bude NB-IoT modul od firmy u-Blox, ke kterému bude nutné připojit mikrokontroler, který se bude starat o zpracování dat a o komunikaci s modulem. Je preferován mikrokontroler s architekturou ARM z důvodu opětovného využití návrhu pro budoucí zařízení. Napájení zařízení bude z baterie, a proto je nutné se zaměřit na vhodný způsob nabíjení baterie a napájení všech dílčích součástí.

DOPORUČENÁ LITERATURA:

[1] RATASUK, Rapeepat, et al. NB-IoT system for M2M communication. In: Wireless Communications and Networking Conference (WCNC), 2016 IEEE. IEEE, 2016. p. 1-5.

[2] ADHIKARY, Ansuman; LIN, Xingqin; WANG, Y.-P. Eric. Performance evaluation of NB-IoT coverage. In: Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th. IEEE, 2016. p. 1-5.

Termín zadání: 5.2.2018

Termín odevzdání: 29.5.2018

Vedoucí práce: Ing. Jiří Pokorný

Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

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

ABSTRAKT

Bakalářská práce se zabývá návrhem IoT senzoru komunikující pomocí moderní technologie NB-IoT. V teoretické části bakalářské práce jsou popsány vlastnosti vybraných LPWAN sítí (nizkopříkonové rozsáhlé sítě) určených pro komunikaci mezi zařízeními na dlouhé vzdálenosti. Bezdrátové sítě jsou porovnány a zhodnoceny na základě frekvenčního pásma, výkonu a citlivosti, přenosových rychlostí, typů modulace a ceny. Následuje Průzkum trhu s IoT zařízeními komunikující přes LPWAN sítě. Praktická část se zabývá návrhem zařízení umožňujícího měření vybraných parametrů kvality vody a ovzduší komunikující přes bezdrátovou technologii NB-IoT, programovým řešením zařízení a testováním komunikace mezi zařízením a serverem. Výstupem praktické části je navržené schéma zapojení, deska plošných spojů a programové řešení napsané v jazyce C++ pomocí online vývojového prostředí Mbed.

KLÍČOVÁ SLOVA

Internet věcí, NB-IoT, LPWAN, měření kvality vody, měření kvality ovzduší

ABSTRACT

The bachelor thesis deals with the design of the IoT sensor communicating with the modern technologies NB-IoT. In the theoretical part of the bachelor thesis are described properties of selected LPWAN networks for communication between long distance devices. Wireless networks are compared and evaluated based on bandwidth, power and sensitivity, transmission rates, types of modulation, and pricing. It is followed by Market Research with IoT devices communicating over LPWAN networks. The practical part deals with the design of equipment for the measurement of selected parameters of water and air quality communicating through NB-IoT wireless technology, software solution of the device and testing communication between the device and the server. The output of the practical part is the proposed circuit diagram, printed circuit board and C++ written solution using the on-line development environment Mbed.

KEYWORDS

Internet of Things, NB-IoT, LPWAN, measuring water quality, measuring air quality

JEŽEK, Jiří. *Návrh IoT zařízení pro bezdrátovou technologii NB-IoT*. Brno, 2018, 65 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Jiří Pokorný

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Návrh IoT zařízení pro bezdrátovou technologii NB-IoT“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Jiřímu Pokornému, za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora



Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
<http://www.six.feec.vutbr.cz>

PODĚKOVÁNÍ

Výzkum popsany v této bakalářské práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno

.....

podpis autora



EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI



OBSAH

Úvod	12
1 Internet věcí a LPWAN sítě	13
1.1 Internet věcí	13
1.2 LPWAN sítě a jejich porovnání	13
1.2.1 NB-IoT	14
1.2.2 Lora	15
1.2.3 Wi-Fi HaLow	15
1.2.4 Sigfox	15
1.2.5 Porovnání	15
1.2.6 Shrnutí	18
2 Výběr scénáře pro použití zařízení	20
2.1 Inteligentní odpadkový koš	21
2.2 Zařízení, které přečte pomocí NFC data z mobilního telefonu	22
2.3 Měření kvality vody a ovzduší	22
2.3.1 Měření kvality vody	22
2.3.2 Měření kvality ovzduší	24
3 Návrh zařízení	26
3.1 Specifikace zařízení	26
3.2 Jednotlivé komponenty	26
3.3 Výběr krabičky	29
3.4 Návrh schématu	30
3.5 Návrh desky plošných spojů	32
3.6 Programování zařízení	33
3.6.1 Vývojové platformy	34
3.6.2 Vývojové desky	35
3.6.3 AT příkazy	36
3.6.4 Popis programového řešení	39
4 Závěr	45
Literatura	46
Seznam symbolů, veličin a zkratk	49
Seznam příloh	51

A Schéma zapojení	52
B Zdrojové kódy	54

SEZNAM OBRÁZKŮ

1.1	Topologie LPWAN sítě.	14
3.1	3D model navržené DPS – rozmístění součástek.	33
3.2	Vývojový diagram obsluhy NB-IoT modulu	39
3.3	Topologie testovacího scénáře	42

SEZNAM TABULEK

1.1	Porovnání vybraných LPWAN sítí.	19
3.1	Tabulka stupně krytí IP.	30
3.2	Tabulka vytvořených metod v knihovně sara	40
3.3	Tabulka vytvořených metod v knihovně senzory	41
3.4	Specifikace NB-IoT modulu	41
3.5	Specifikace serveru	41

SEZNAM VÝPISŮ

3.1	Výpis na straně NB-IoT modulu	44
3.2	Výpis na straně serveru	44

ÚVOD

V dnešní době internet stále více proniká do našich životů, ať už jde o vyhledávání informací, prodej a nákup zboží, komunikaci přes sociální sítě, zábavu v různé podobě, usnadnění administrativy atd.

V posledních dvou desetiletích přestává být internet vnímán pouze jako propojení počítačů. Začalo to umožněním připojení k internetu telefonům, kdy najednou nebyl člověk vázán pouze na svůj počítač a mohl se připojit na internet teoreticky odkudkoli. Stále se však jednalo o komunikaci člověka se strojem, až v poslední době se začíná objevovat i komunikace stroje se strojem, která je definována jako datová komunikace napříč zařízeními bez potřeby interakce člověka. Pro taková zařízení vznikl termín Internet Věcí (Internet of Things nebo také IoT). Propojení zařízení bývá převážně realizováno bezdrátově.

Součástí internetu věcí jsou i LPWAN sítě (Low-Power Wide Area Network), které byly vytvořeny právě pro přenos informací mezi zařízeními v internetu věcí. Jak už napovídá název, LPWAN sítě jsou konstruovány tak, aby byly možné přenosy na velkou vzdálenost, s nízkou energetickou náročností s čímž souvisí i nízký objem přenášených dat [1]. Existuje velké množství LPWAN technologií a platforem, nejznámější jsou: Technologie LoRa (Long Range WAN nebo LoRaWAN), Technologie Sigfox a NB-IoT (NarrowBand Internet of Things), na kterou se tato práce soustředí. Samozřejmě každá LPWAN síť má své výhody a nevýhody, z toho vyplývá i různost použití LPWAN sítí pro jednotlivé aplikace.

Pro bakalářskou práci byla vybrána LPWAN technologie NB-IoT, která nabízí velké množství využití. Úkolem bakalářské práce bylo mimo jiné, porovnat možnosti použití této sítě a následná specifikace konkrétního zařízení. V praktické části se bakalářská práce zabývá návrhem schéma zapojení zvoleného zařízení, návrhem desky plošných spojů a výběrem krabičky splňující provozní požadavky zařízení.

Po finálním návrhu konstrukčního řešení zařízení bylo nutné napsat program ovládající zařízení. Program je napsán v jazyce C++ a využívá AT příkazů pro komunikaci MCU s NB-IoT modulem. Na závěr bylo provedeno testování komunikace zařízení se vzdáleným serverem.

1 INTERNET VĚCÍ A LPWAN SÍTĚ

Obsahem následující kapitoly je stručný popis IoT, jak funguje a také informace o LPWAN sítích včetně porovnání vlastností čtyřech vybraných.

1.1 Internet věcí

Termín Internet Věcí odkazuje na spojení mezi různými zařízeními, které mezi sebou mohou komunikovat (vyměňovat si data) s minimálním dohledem člověka. Díky obrovskému rozvoji je internet věcí využíván v mnoha různých odvětvích např. v oblasti bezpečnosti, vzdálené sledování, měření spotřeby (voda, plyn, elektřina) a chytré sítě (v angličtině smart grid). Počet připojených zařízení exponenciálně stoupá, v roce 2020 je odhadováno, že bude připojeno 50 miliard zařízení [1].

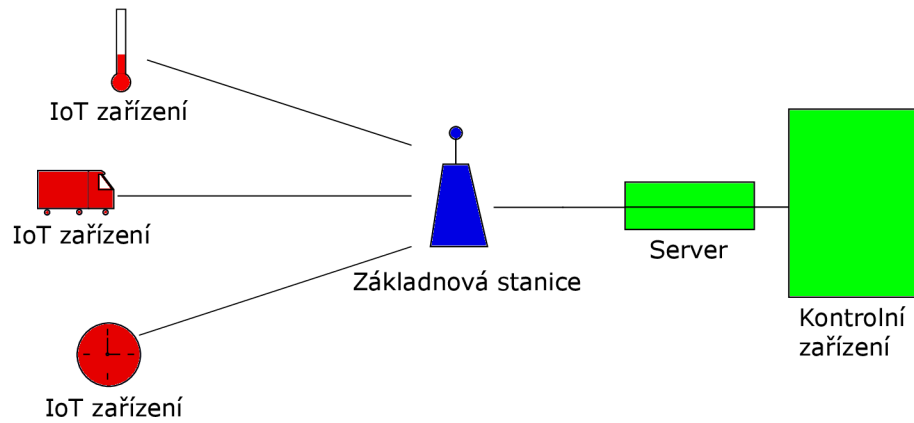
První zmínky o internetu věcí sahají až do roku 1999, díky anglickému idealistovi Kevinu Ashtonovi. Podle jeho předpokladů dojde k propojení fyzického světa a internetu pomocí rozsáhlého systému datových senzorů.

Internet Věcí je globální síť, ve které jsou objekty jednoznačně klasifikovány a založeny na sofistikovaných komunikačních protokolech. Zařízení patřící do IoT jsou velmi úsporné a energeticky nenáročné, takže i přes jejich velký počet mají minimální dopad na planetu [2].

1.2 LPWAN sítě a jejich porovnání

LPWAN (Low-Power Wide Area Network) je zkratka pro bezdrátové technologie ve velkých sítích, konkrétně typu WAN (Wide Area Network) využívaných v IoT. Je jich větší množství a každá konkrétní technologie je ve vlastnictví soukromých subjektů. Tyto síťové technologie vyžadují propojení zařízení s nízkou šířkou přenosového pásma a zároveň se zaměřuje na úsporu energie s čímž souvisí i účinnost přenosu. V oblasti LPWAN byl zaznamenán velký pokrok zejména díky tomu, že LPWAN poskytuje nižší nároky na napájení, má vysoké pokrytí a nižší šířku pásma než existující mobilní sítě. Na obrázku 1.1 je znázorněna topologie LPWAN sítí.

Technologie LPWAN jsou užitečnější v průmyslu, občanských a komerčních oblastech, kde je hlavní snahou zlepšit komunikaci ze strany požadavků na účinnost a výkon. Pro koncového uživatele jsou naopak díky vyšším přenosovým rychlostem výhodnější technologie jako je Bluetooth, ZigBee a Wi-fi [3].



Obr. 1.1: Topologie LPWAN sítě.

Z hlediska licencování kmitočtového spektra můžeme LPWAN technologie rozdělit do dvou kategorií:

- **Pracující v licenčním pásmu.**
- **Pracující v nelicenčním pásmu.**

LPWAN technologie pracující v **nelicenčním pásmu** jsou reprezentovány technologiemi Lora, Sigfox, Wi-Fi HaLow. Většina z nich je nestandardní a provedena na zakázku. LPWAN technologie pracující v **licenčním pásmu** jsou obecně zastoupeny relativně starými 2G/3G mobilními komunikačními technologiemi (například GSM), LTE technologiemi a eLTE technologiemi [4].

Níže budou porovnány čtyři existující standardy využívané v LPWAN sítích:

- **NB-IoT.**
- **LoRa.**
- **Wi-Fi HaLow.**
- **Sigfox.**

1.2.1 NB-IoT

NB-IoT (NarrowBand Internet of Things) je bezdrátová úzkopásmová LPWAN technologie. Využívá již existující pásma GSM (Globální Systém pro Mobilní komunikaci) a LTE (Long Term Evolution), ctí standardy 3GPP (dohoda o spolupráci v oblasti mobilních komunikací z roku 1998). NB-IoT poskytuje zlepšené pokrytí uvnitř budov, nízkou cenu koncových zařízení a zároveň nízkou energetickou náročnost. Aby bylo dosaženo co nejlepšího využití spektra, byl NB-IoT navržen s řadou možností nasazení pro GSM (globální systém pro mobilní komunikaci), W-CDMA (širokopásmový vícenásobný přístup s kódovým dělením) nebo LTE: standalone - výměna GSM nosiče s NB-IoT nosičem (využití hlavně v oblastech se souběžným pokrytím GSM a WCDMA/LTE), in-band skrze flexibilní využití části LTE nosiče,

ochranné pásmo ve WCDMA nebo v LTE. V České republice je provozována telefonním operátorem Vodafone a díky využití LTE vysílačů dokázal pokrýt celou Českou republiku.

1.2.2 Lora

Neboli LoRaWAN (Long Range Wide Area Network) se řadí do nízkopříkonových bezdrátových síťových protokolů, které byly navrženy pro levnou a bezpečnou komunikaci v IoT. Komunikace mezi koncovými prvky a branami je rozdělena na několik frekvenčních pásem a přenosových rychlostí. Díky technologii rozprostřeného spektra spolu jednotlivé komunikační proudy neinterferují a to umožňuje vznik sady „virtuálních“ kanálů pro zvýšení kapacity brány. LoRaWAN server řídí přenosovou rychlost prostřednictvím systému adaptivní rychlosti přenosu dat, čímž se maximalizuje výdrž baterie a kapacita sítě.

1.2.3 Wi-Fi HaLow

Wi-Fi HaLow, také známá jako IEEE 802.11 ah, nepracuje na frekvencích 2,4 GHz nebo 5 GHz, ale na frekvenci 900 MHz což nám zajišťuje širší pokrytí a menší náchylnost k rušení z uvedených pásem. Díky nastavení intervalů, kdy se budou přenášet data, má Wi-Fi HaLow oproti Wi-fi nižší vysílací výkon na úkor možnosti provozu na baterie, takže podobně jako ostatní LPWAN sítě je energeticky nenáročný. Není ještě plně standardizován a měl by být přímým konkurentem pro Bluetooth, oproti kterému má vyšší dosah.

1.2.4 Sigfox

Sigfox vyvinula stejnojmenná francouzská firma. Sigfox využívá technologii UNB (ultra nízkého pásma, Ultra-narrow Band). Při posílání zpráv neexistuje signalizace mezi odesílatelem a příjemcem. Sigfox umožňuje duplexní komunikaci mezi IoT zařízeními a koncovým zařízením. Odesílají se zpravidla velmi malé zprávy a malé množství (až 144 denně), veškerá komunikace prochází přes Sigfox cloud, který slouží jako rozhraní mezi zákazníkem a poskytovatelem. Zaveden v mnoha státech světa (aktuálně 35) včetně České republiky, kde je tato síť vedena pod operátorem SimpleCell [5].

1.2.5 Porovnání

Nasledně budou porovnány čtyři popsané standardy a to NB-IoT, LoRaWAN, Wi-Fi HaLow a Sigfox.

Frekvenční a obsazená pásma

NB-IoT používá frekvenční pásmo LTE (např. 2530–2540 MHz, 2650–2660 MHz, 847–854,5 MHz), s možností pracovat v pásmu hlavního signálu, v ochranném intervalu, nebo v samostatném rozsahu. Obsazená šířka pásma závisí na typu zařízení a pohybuje od 15 kHz do 20 MHz pro downlink a od 3,75 kHz do 20 MHz pro uplink. Typická hodnota pro většinu aplikací je 200 kHz.

LoRaWAN používají nelicencovaná frekvenční pásma, která se mohou lišit podle zákonů konkrétní země. Například v Evropě to je frekvenční rozsah v oblasti 863 – 870 MHz a v USA 902 – 928 MHz. Tyto pásma jsou dostatečně podobné na to, aby byly podporovány nezávisle na použitém softwaru. Co se týče šířky pásma, **LoRaWAN** pracuje ve 2 frekvenčních pásmech – 125 kHz a 250 kHz.

Třetím porovnávaným standardem je **WiFi HaLow** (802.11ah), který byl také navržen pro provoz na nelicencovaných kmitočtových pásmech, stejně jako LoRaWAN, takže pro něj platí stejné vlastnosti. Šířka pásma se však oproti LoRaWAN výrazně liší a tvoří 1/2/4/8/16 MHz. Záleží na typu zařízení a na požadované rychlosti přenosu dat.

U posledního porovnávaného standardu, **Sigfox**, jsou frekvenční pásma opět závislé na konkrétní oblasti. V Evropě to je 868 MHz v Severní Americe 902 MHz. Spektrum v 868 MHz je rozděleno do 400 kanálů od 868,18 do 868,22 MHz. Frekvence pro zpětný přenos jsou o 1 MHz vyšší. Obsazená šířka pásma je velmi malá, jen 100 Hz na kanál, což poskytuje lepší ochranu proti rušení [6],[5].

Vyzařovaný výkon a citlivost přijímače

U prvních tří porovnávaných standardů je maximální vyzařovaný výkon vysílače 500 mW. U Sigfoxu je to pouze 25 mW. Pokud porovnáme všechny tři standardy podle citlivosti přijímače, tak nejvyšší dosažitelnou citlivost má **Sigfox** -142 dBm, která je téměř shodná s maximální citlivostí přijímačů **NB-IoT** -141 dBm.

Pro **LoRaWAN** se hodnota citlivosti může pohybovat od -108 dBm do -136 dBm a záleží na typu použité modulace. Vzhledem k tomu, že technologie **WiFi HaLow** (802.11ah) není ještě zcela standardizována, je citlivost pouze odhadována v rozsahu od -120 dBm do -130 dBm.

Přenosová rychlost

Maximální přenosová rychlost není u porovnávaných standardů klíčová vlastnost. Je často obětována na úkor jiných důležitějších parametrů, konkrétně maximálnímu dosahu a spotřebě energie. Tento parametr může být důležitý pro zařízení, která potřebují přenášet velké množství dat, typicky obrázky nebo video.

Nejvyšší maximální rychlost přenosu dat má **WiFi HaLow** (802.11ah), která může

dosahovat hodnoty až 346,666 Mb/s (s minimem 150 kb/s).

Výrazně nižší maximální přenosovou rychlostí se může pochlubit **NB-IoT** technologie s hodnotou až 4 Mb/s. Standardně se rychlost pohybuje kolem 200 kb/s a minimální je 0,3 kb/s.

LoRaWAN, je primárně určena pro organizaci senzorových sítí, které nepotřebují přenos velkého množství dat. Nejdůležitější je pro ni právě výdrž baterie. Maximální přenosová rychlost je tedy velmi omezena a dosahuje hodnot v rozsahu od 0,3 kb/s do 50 kb/s.

Sigfox má přenosovou rychlost velmi nízkou, buď 100 b/s nebo 600 b/s, záleží na konkrétní zemi. Vzhledem k nízkému objemu zpráv (maximálně 144 za den) s nízkým množstvím uživatelských dat (payload) - 0 – 12 bajtů.

Typy modulace

NB-IoT a **WiFi Halow** používají stejné typy modulace. Obě výše zmíněné podporují modulace OFDM (ortogonální multiplex s frekvenčním dělením) s b/sK (Binární – fázové klíčování), QPSK, QAM na postranních pásmech v závislosti na rychlosti přenosu dat a kvalitě signálu.

LoRaWAN má svoji vlastní patentovanou modulaci, založenou na Spread Spectrum Modulation a na Chirp Spread Spectrum s integrovanou korekcí chyb.

Sigfox používá modulaci Db/sK (diferenciální b/sK) pro uplink. Pro downlink používá **Sigfox** modulaci GFSK (klíčování frekvenčním posunem).

Cena zařízení

Dalším důležitým a posledním porovnávaným faktorem u LPWAN sítí jsou náklady na zprovoznění sítě (jedna základová stanice). Nejdražší řešení je **NB-IoT**, protože pro implementaci tohoto standardu vyžadujeme modernizaci existujících základových stanic, cena tedy začíná na 15 000 \$ (pro představu zhruba 320 000 Kč).

Cena základní stanice pro další dvě technologie (**LoRaWAN** a **WiFi Halow**) je výrazně nižší a pohybuje se od 100 \$ (cca. 2 000 Kč) do 1 000 \$ (cca. 20 000 Kč), vzhledem k masivnímu šíření technologií a přístupu širšímu spektru uživatelů je možné další snížení cen.

Základní stanice pro **Sigfox** jsou také velmi levné a na podobné cenové úrovni jako **LoRaWAN** a **WiFi Halow**, tedy od 100 \$ do 1 000 \$.

Koncová zařízení (moduly pro komunikaci na konkrétní technologii) jsou pro všechny porovnávané LPWAN sítě velmi levná, včetně poplatků za provoz.

1.2.6 Shrnutí

Po rychlé analýze nejdůležitějších vlastností LPWAN technologií, můžeme určit jejich výhody a nevýhody.

Výhody **NB-IoT** zahrnují schopnost nasadit sítě založené na existujících LTE sítích, vysoká oblast pokrytí (kolem 10 km), vysoká kvalita zákaznických služeb, jedna základní stanice umožňuje připojit až 50 000 zařízení.

Nevýhodami **NB-IoT** jsou relativně vysoká cena pro zavedení základové stanice a nutnost přítomnosti SIM (účastnická identifikační karta) karty, která poskytuje možnost integrace do existujících mobilních sítí.

Výhody LoRaWAN standardu jsou nízké náklady na zavedení sítě a koncová zařízení, vysoká oblast pokrytí (až 10 km), vysoká výdrž baterie, která může být i více než 10 let (záleží samozřejmě na velikosti baterie a odběru zařízení).

Nevýhodami jsou: Nízká kvalita zákaznických služeb, nízké přenosové rychlosti (0,3 – 50 kb/s) a oproti porovnávaným technologiím i nižší počet zařízení na jednu základovou stanici, „jen“ 5 000, což je například 10x méně než **NB-IoT**.

Další porovnávaná technologie, **WiFi HaLow** (802.11ah) má výhody: Možná integrace s Wi-Fi sítěmi a otevřený přístup, vysoké přenosové rychlosti (0,150 – 346 Mb/s), vysoká výdrž baterie (až 10 let), velké množství výrobců zařízení, což může snížit cenu zařízení, podporuje MIMO (Multiple-input multiple-output – česky více vstupů více výstupů), to může být velmi užitečné, protože senzor může být instalován mimo zorné pole.

Nevýhody jsou: Nízká oblast pokrytí (až 1 km), není ještě plně standardizována a má horší kvalitu zákaznických služeb [6].

U poslední porovnávané LPWAN sítě **Sigfox** jsou výhody: Vysoká výdrž baterie, díky nízkému objemu posílaných dat, která bývá udávána až 15 let. Další výhodou je oblast pokrytí, která v terénu může dosahovat až 50 km a poslední výhodou je vysoká citlivost přijímače, která dosahuje hodnot až -142 dBm.

Nevýhody sítě **Sigfox** jsou: Nízká rychlost přenosu dat a to buď 100 nebo 600 b/s. Nízká oblast pokrytí ve městě v uzavřených budovách, kde je dosah pouze 3 km. Další nevýhodou může být nízký obsah uživatelských dat od 0 do 12 bajtů.

V Tab. 1.1, jsou porovnány klíčové vlastnosti jednotlivých LPWAN.

Tab. 1.1: Porovnání vybraných LPWAN sítí.

	NB-IoT	LoRa	Wi-Fi HaLow	Sigfox
Frekvenční rozsah	Odpovídá LTE (2530 – 2 540 / 2650 – 2660 MHz, 847 – 854,5 / 806 – 813,5 MHz).	Evropa: 868 MHz USA: 915 MHz Austrálie: 430 MHz	Evropa: 868 MHz USA: 915 MHz Austrálie: 430 MHz.	Evropa: 868 MHz USA: 915MHz.
Šířka pásma	DL: 15 kHz – 20 MHz UL: 3,75 kHz – 20 Mhz.	125, 250 kHz (Evropa, Rusko).	1/2/4/8/16 MHz.	100 Hz
Maximální výkon vysílače	500mW.	500mW.	500mW.	25mW.
Citlivost přijímače	maximálně -141 dBm. Typicky -120 dBm.	Záleží na typu modulace od -108 do -136 dBm.	Odhadováno od -120 do -130 dBm.	Až -142 dBm.
Rychlost přenosu dat	0,3 - 4000 kb/s. Typicky 200kb/s.	0,3 – 50 kb/s.	0,15 – 346 Mb/s.	100 nebo 600 b/s.
Oblast pokrytí	Kolem 10 km.	Až 10 km.	Až 1 km.	až 50 km v terénu, ve městě maximálně 3km.
Počet zařízení na jednu ZS	Až 50 tisíc na 1 základovou stanici.	Až 5000.	Až 8191.	Neuvedeno.
Výdrž baterie Stav	Více než 10 let. Standardizováno 3GPP.	až 10 let (2000 mAh). Standardizované, proprietární.	až 10 let. Zatím nestandardizována.	až 15 let. Standardizována.

2 VÝBĚR SCÉNÁŘE PRO POUŽITÍ ZAŘÍZENÍ

Zařízení v Internetu věcí jsou různého typu a funkce. Využitelnost Internetu věcí je obrovská. Oblasti použití byly rozděleny na osm kategorií podle oblasti použití:

Zemědělství Dnes již zemědělci využívají moderní technologie, velkými pomocníky jsou například satelitní navigace GPS a geografický informační systém (GIS). IoT může tyto technologie doplnit a pomoci vzdáleně kontrolovat hospodářství. Potenciální použití je například k hlídání odlehlých ploch, objektů, zvířat a plodin, lokální meteorologické stanice, měření kvality půdy, monitoring zemědělských strojů, sledování zdravotního stavu skotu („Internet of Cows“). . .

Inteligentní město Internet věcí najde uplatnění i v samotných městech. Je využitelné pro snímání různých veličin, jako například úroveň znečištění ovzduší, teplota, rychlost větru a mnoho dalších. Následná reakce na výsledky měření, může pomoci k většímu životnímu komfortu, úspoře energií a bezpečnosti. Potenciální využití nachází například v zabezpečení objektů, řízení pouličního osvětlení, inteligentní řízení dopravy, sdílení dopravních prostředků (dnes hodně známé sdílení kol, funguje i v České republice), sledování znečištění ovzduší, dohled a efektivní nakládání s odpady. . .

Energetika a průmysl V energetice nachází IoT uplatnění hlavně v monitoringu přenosové sítě. Umožňuje to kontrolu spotřebované energie nejen ze strany dodavatelů, ale také ze strany spotřebitelů. Dnes se již začínají používat chytré elektroměry. Správci přenosových soustav budou mít k dispozici komplexnější informace o závadách a jejich charakteru, budou moci regulovat tok energie podle dat z koncových měřidel. Průmysl využije IoT například v inovacích výrobních strojů. Firmy získají detailní přehled o výrobě a o stavu stroje, což pomůže lépe plánovat výrobu a včas odhalit různé závady.

Doprava a logistika V automobilovém průmyslu a přepravě nákladu najde IoT také využití. Dnes už fungují například pokusná autonomní vozidla, online sledování zásilek. Do budoucna by IoT mohl přinést další výhody, kdy by spolupracujícím zařízením umožnil například zlepšení plynulosti dopravy, přivolání okamžité pomoci v případě dopravní nehody, . . .

Zdravotnictví Zařízení jsou využívány k monitorování zdravotního stavu člověka. Sledovaná data umožní sledovat vývoj fyziologických pochodů. Monitorován může být krevní tlak, tělesná teplota, hladina glukózy v krvi, srdeční tep a hmotnost.

Lékaři by tedy mohli mít online přístup k těmto datům a tedy i příznivě ovlivňovat život pacienta.

Nositelná elektronika Zařízení, která jsou svými rozměry a hmotností přímo přizpůsobena k nošení na lidském těle. Nositelná zařízení, se dnes již hojně používají lze mezi ně zařadit například chytré hodinky, fitness náramky, GPS technika, elektronické oblečení. . .

Spotřebiče v síti Dnes je již možné připojit některé domácí spotřebiče do sítě Internet, to že budeme mít připojeny spotřebiče v Internetu nám může přinést výhody jako jsou možnost ovládat funkce zařízení odkudkoli, předávání informací mezi spotřebiči, upozornění na závadu a nebo také aktualizaci softwaru. S trochou nadšázky pak mohou vzniknout takové scénáře jako, že si lednice sama objedná chybějící potraviny, nebo že vám trouba navrhne oběd podle dostupných surovin.

Inteligentní elektroinstalace Pod tímto pojmem si lze představit tzv. „inteligentní domy/byty“, jejichž výrobci se předhánějí tím, co všechno lze ovládat odkudkoli si jen vzpomenete. Všechny zařízení a snímače jsou přes uzavřený protokol připojeny na centrální jednotku s připojením k Internetu. Inteligentní elektroinstalace umožňuje zabezpečení objektu, řízení teploty v domě, řízení zámků, garážových vrat a stínící techniky, úsporu energií a centralizaci multimédií [7].

Po prostudování konkrétních návrhů zařízení, přicházeli v úvahu tři možnosti:

- **Inteligentní odpadkový koš.**
- **Zařízení monitorující kvalitu vody.**
- **Zařízení, které přečte pomocí NFC data z mobilního telefonu.**

2.1 Inteligentní odpadkový koš

Inteligentní odpadkový koš, by měl v první řadě podat informaci o tom, že je plný. Úroveň naplnění hlídá senzor na měření vzdálenosti, který můžeme realizovat buď infračerveným senzorem nebo ultrazvukovým senzorem. Oba dva senzory fungují na vysílání vlny, která se následně odrazí od překážky zpět do senzoru. Následně je podle vzorce:

$$s = c \cdot \frac{t}{2}, \quad (2.1)$$

kde c – rychlost šíření vlny, t – čas za který se vrátí vyslaná vlna. určena vzdálenost senzoru od překážky. Hlášení naplnění odpadkových košů pomáhá zefektivnit vývoz odpadů, případně vypočítá co nejefektivnější trasu pro popelářské auto. Chytré odpadkové koše se již dnes běžně používají. Systém inteligentních odpadkových košů pomáhá zefektivnit vývoz odpadu a udržet město v čistotě[8].

2.2 Zařízení, které přečte pomocí NFC data z mobilního telefonu

NFC (Near Field Communication) je modulární technologie rádiové bezdrátové komunikace mezi zařízeními na velmi krátkou vzdálenost (do 10 cm). NFC umožňuje oboustranou komunikaci. NFC pracuje na frekvenci 13,56 MHz a přenáší data rychlostí 424 kb/s. NFC může pracovat ve třech režimech:

- **Emulace karty** – NFC zařízení se chovají jako karty.
- **Přenos jedním směrem (čtení/zápis)**.
- **Distribuovaný režim** – dvě NFC zařízení si mohou mezi sebou vyměňovat data, přičemž oba podporují jak přijímání, tak odesílání dat.

NFC je využitelné například pro vyčtení informací z NFC tagu (štítky, nálepky, přívěšky apod.), které obsahují NFC čip, typicky takový štítek může obsahovat webovou adresu, která se načte do přiloženého telefonu. Dále je možné NFC použít pro provádění plateb pomocí mobilního telefonu. Zařízením s NFC lze odemknout dveře. Použití NFC je tedy poměrně široké a dá se v návaznosti na to vymyslet obrovské množství aplikací.

Po zvážení všech možností a užitečnosti zařízení jsem se rozhodl pro variantu „Zařízení monitorující kvalitu vody“, z důvodu toho, že je mi toto téma nejbližší s tím, že kvůli vysoké finanční náročnosti senzorů pro měření vody, budou součástí pouze nejdůležitější senzory a zařízení bude rozšířeno o senzory na měření znečištění vzduchu.

2.3 Měření kvality vody a ovzduší

Znečištění vody a ovzduší se s postupem času stalo jedním z nejzávažnějších typů znečištění životního prostředí. S technickým vývojem se zlepšily i metody měření znečištění životního prostředí. Spoustu parametrů ovlivňujících kvalitu vody a vzduchu, je dnes možné měřit pomocí senzorů k tomu určených a na základě měření na to adekvátně reagovat.

2.3.1 Měření kvality vody

Sledování kvality vody je důležité pro různá odvětví, s tím souvisí i rozsah sledovaných parametrů. Při sledování kvality pitné vody nás zajímají jiné vlastnosti a je na ně kladen větší důraz než při sledování vody v bazénu. Měřitelné parametry z pohledu kvality vody:

- **pH**

- **Teplota vody**
- **Rozpuštěný kyslík**
- **Konduktivita**

pH

Neboli vodíkový exponent je to číselné vyjádření stupně kyselosti nebo zásaditosti se stupnicí od 0 (kyselé) do 14 (zásadité). Limit pro pitnou vodu je od 6,5 do 9,5. Hodnoty pH výrazně ovlivňují chuť vody, pokud se nejedná o extrémní hodnoty, tak nemá přímý zdravotní význam. Velký význam má však pH v bazénové technice, kde je důležitým faktorem. Optimální pH pro bazénovou vodu je 7,4, protože je stejné hodnoty jako pH v lidských očích a sliznicích a také poskytuje dobrou dezinfekci chloru.

Měření pH je možné pomocí acidobazických indikátorů, což jsou látky, které mění uspořádání dvojných vazeb v molekule v závislosti na pH prostředí, to se projeví změnou zbarvení roztoku. V praxi se pro měření používá universální indikátorový papírek, který se zabarví dle pH měřeného roztoku od červené po tmavě modrou (červená kyselá, modrá zásaditá). Měření pH pomocí acidobarických indikátorů je pro člověka dobře čitelné pro přístroj však nikoliv.

Druhý způsob měření hodnot pH jsou instrumentální metody, která se používají pro přesná měření. V současné době se používá potenciometrie s využitím skleněné elektrody coby měrného členu. Funguje to na principu přesného měření elektrického potenciálu mezi měrnou a referenční elektrodou. Elektrický potenciál se mění s kyselostí měrné elektrody. Potenciál mezi měrnou a referenční elektrodou je měřen pomocí citlivého voltmetru s vysokým vstupním odporem [9], [10], [11].

Teplota vody

Díky vysoké měrné tepelné kapacitě jsou teplotní výkyvy ve vodě menší než na vzduchu. Teplota vody přímo ovlivňuje množství plynů rozpuštěných ve vodě a to tak, že čím vyšší teplota, tím méně se v ní rozpustí plynů. Teplota také ovlivňuje rychlost chemických reakcí (např. oxidace) [12].

Pro měření teploty vody se používají teploměry, nebo teplotní senzory, které musí být vodotěsné. Teplotních čidel a senzorů existuje velká řada, nejčastěji se používají odporové senzory, u kterých je využit vliv měnící se teploty na odpor senzoru (s měnící se teplotou, se mění i odpor), což je poté přepočítáno na, pro náš čitelnější jednotky (°C, °F).

Rozpuštěný kyslík

Kyslík je nejdůležitějším z rozpuštěných plynů ve vodě. Množství rozpuštěného kyslíku ovlivňuje atmosferický tlak, množství rozpuštěných látek ve vodě a nejvíce teplota vody, když tyto parametry rostou, tak ubývá rozpuštěného kyslíku ve vodě. Množství kyslíku ve vodě bývá často limitujícím faktorem pro život různých organismů. Kyslík se do vody dostává ze vzduchu a fotosyntézou vodních rostlin, řas a sinic. Obsah kyslíku je klíčovým faktorem pro chov ryb.

Obsah rozpuštěný kyslíku se měří pomocí DO metrů. Ty se podle principu dělí elektrochemické nebo optické [12].

Konduktivita

Konduktivita, neboli měrná vodivost je míra koncentrace elektrolytů ve vodě. Nepřímo vyjadřuje obsah minerálních látek nacházejících se ve vodě. Limit vodivosti pro pitnou vodu je 125 mS/m, což je rovno obsahu rozpuštěných látek asi 1000 mg/l, vody s konduktivitou přesahující limit jsou označeny jako minerální a nejsou vhodné pro stálé konzumování. Vysoká konduktivita vody, může způsobovat i technické potíže, jako je snižování životnosti potrubí nebo ohříváčů vody [13].

Konduktivita se měří pomocí konduktometrů, kdy jsou nejpoužívanější konduktometry používající indukční metodu. Indukční metoda spočívá v tom, že jsou použity dvě induktivně spojené cívky. Jedna je hnací cívka, která vytváří magnetické pole a zároveň je u ní známo napětí, druhá je sekundární cívka transformátoru. Kapalina prochází kanálem v čidle, tvoří jednu otáčku v sekundárním vinutí transformátoru. Výstupem snímače je indukovaný proud, který je přímo úměrný konduktivitě měřené vody [14].

2.3.2 Měření kvality ovzduší

Největší koncentrace znečištění ovzduší bývá ve velkých městech. Největšími zdroji znečištění jsou teplárny, tepelné elektrárny, automobilová a železniční doprava. Důležité vlastnosti vody z pohledu kvality ovzduší:

- **Měření koncentrace prachu**
- **Měření oxidu uhelnatého**
- **Teplota, vlhkost a atmosférický tlak**

Měření koncentrace prachu

Velká koncentrace prachu je určitě problémem, který zaslouží pozornost. Velké množství prachu a sazí vzniká nedokonalým spalováním, chemickým zpracováním a povrchovou těžbou, což jsou v České republice poměrně rozvinutá odvětví. Koncentraci

prachu je možné měřit pomocí senzorů prachu. Většina z nich měří koncentraci prachu v $\frac{mg}{m^3}$, v předem definovaném prostoru.

Nejčastěji se používá pro měření a detekci prachu senzor, který pracuje na principu průchodu optického paprsku napříč detekčním prostorem. Náráz paprsku světla do částic prachu snižuje energii a intenzitu paprsku. Když porovnáme přijatou a odešlanou intenzitu paprsku, můžeme odvodit koncentraci prachu v prostoru mezi vysílačem a přijímačem paprsku [15].

Měření oxidu uhelnatého

CO (oxid uhelnatý), je jedovatý plyn bez barvy, chuti a zápachu. Mísí se se vzduchem a vzniká působením ultrafialového záření, jako produkt nedokonalého spalování fosilních paliv a biomasy. Koncentrace CO ve vzduchu je měřena v jednotkách ppm (výraz pro jednu miliontinu z celku), nebezpečná hodnota koncentrace CO ve vzduchu je 100 ppm, životu nebezpečná pak 200 ppm.

Měření koncentrace CO ve vzduchu se provádí pomocí CO metrů. Součástí samotného senzoru je většinou poplašné zařízení, které uživatele upozorní na větší koncentraci CO. Jsou dosti podobné detektorům kouře, jen s tím rozdílem, že uvažují koncentraci a zároveň i čas, což napodobuje absorpci oxidu uhelnatého v těle a zároveň zabraňuje „falešným poplachům“, jakmile koncentrace za určitý čas přesáhne bezpečnou hodnotu, tak dojde k upozornění uživatele [16].

Teplota, vlhkost a atmosférický tlak

Tyto tři veličiny patří k nejčastějším měřením v oblasti vzduchu. Teplota je nejčastěji uváděna ve °C nebo ve °F. Vlhkost vzduchu udává množství vodních par obsažených ve vzduchu, které se dostávají do vzduchu vypařováním z půdy a vodní hladiny. Pro větší srozumitelnost měříme relativní vlhkost, která je definována jako poměr skutečného obsahu vodních par ku maximálnímu obsahu vodních par a je udávána v procentech. Hmotnost vzduchu v atmosféře vytváří atmosférický tlak. Udává se převážně v hektopascalech (hPa). Pro předpověď počasí není důležitá ani tak hodnota tlaku, jako hodnota a rychlost jeho změny. Atmosférický tlak je závislý na nadmořské výšce a na teplotě vzduchu [17].

V dnešní době je dost časté, že jeden senzor umí změřit všechny tři veličiny zároveň.

3 NÁVRH ZAŘÍZENÍ

Jak již bylo naznačeno, úkolem je navrhnout zařízení určené pro měření kvality vody, které bude mít jako sekundární vlastnost měřit kvalitu vzduchu.

3.1 Specifikace zařízení

Jak již bylo mírně naznačeno v kapitole o měření kvality vody a vzduchu, zařízení bude umět změřit některé parametry co se týká kvality vody a některé parametry týkající se kvality ovzduší. Konkrétně se jedná o parametry:

- Kvalita vody
 - Teplota.
 - pH.
- Kvalita ovzduší
 - Koncentrace prachu.
 - Oxid uhelnatý.
 - Teplota, vlhkost a tlak.

Dále je nutné, aby zařízení odesílalo data o naměřených hodnotách, pro to byla zvolena technologie NB-IoT, což byl jeden z hlavních požadavků pro zpracování práce. Jelikož se pro praktické použití nepočítá s tím, že zařízení bude v dosahu elektrické sítě, je nutné aby zařízení mělo baterii.

3.2 Jednotlivé komponenty

K měření jednotlivých parametrů jsou využívány senzory, které to umožňují. V této kapitole budou představeny konkrétní senzory a důležité prvky zařízení.

Vodotěsný digitální teploměr s čidlem DS18B20

Sonda s teplotním čidlem DS18B20, má k dispozici programovatelné rozlišení od 9 do 12 bitů. Sonda je voděodolná, což je pro měření teploty vody klíčová vlastnost. Zapouzdření čidla je realizováno pomocí nerezové oceli. Metr dlouhý kabel poskytuje dostatečný dosah pro naši potřebu. Čidlo obsahuje tři vodiče z nichž 1. (GND) vodič je zem, 2. (DQ) je datový vodič, u kterého je využita datová sběrnice OneWire, 3. vodič (VCC) zajišťuje napájení 3,3V. DS18B20 podporuje parazitní režim, což v praxi znamená, že nám na spojení čidla s procesorem vystačí pouze 2 vodiče (napájení bude na stejném vodiči jako data, s přidaným rezistorem velikosti 4700 Ω). V našem zapojení však bude použita varianta se třemi vodiči [18].

Modul a sonda pro měření pH

Pro změření hodnoty pH tekutiny budeme potřebovat sondu a k tomu modul, aby to co naměří sonda bylo použitelné i pro procesor. Sonda E201 umožňuje měřit pH tekutin v rozsahu od 0 do 14 s deklarovanou přesností $\pm 0,5$. Pracovní rozsah sondy je od 0°C do 60°C , délka kabelu 95 cm připojení k modulu realizováno pomocí BNC konektoru a obvyklá doba ustálení měření je maximálně jedna minuta. Součástí modulu jsou dva trimry, přičemž první z nich (blíže k BNC konektoru) slouží ke kalibraci měření a druhým se nastavuje hranice hodnoty pH, kdy se rozsvítí červená kontrolní LED na modulu (pin DO, který nebude v našem případě použit). Pro úspěšné propojení modulu a procesoru, nám stačí zapojit tři vodiče. První z nich je napájení 5 V (V+), druhým je zem (GND) a třetím je analogový měřící pin (Po) [19].

Optický senzor kvality ovzduší Sharp GP2Y1010AU0F

Je vhodný pro měření koncentrace prachu v okolí. Je citlivý na velmi jemné částice a často je využíván v čističkách vzduchu nebo v detektorech cigaretového kouře. Funkce detektoru stojí na dvou součástkách a to infračervené diodě a fototranzistoru. Množství prachu se pak měří na základě intenzity dopadajícího záření z diody na fototranzistor. Pro připojení k procesoru je třeba zapojit 6 vodičů. První se zapojí přes odpor 150Ω na 5 V, druhý na zem (GND), třetí připojíme na digitální pin procesoru, čtvrtý opět na zem (GND), pátý na analogový pin procesoru a poslední šestý na 5 V. Dále je potřeba přidat polarizovaný kondenzátor s hodnotou $220\mu\text{F}$, jehož kladná elektroda bude připojena na první pin a druhá na zem (GND) [20].

Senzor oxidu uhelnatého

Senzor je vhodný pro detekci úniku oxidu uhelnatého (CO) a hořlavých plynů. Skládá se z polovodičového snímače s SnO_2 , komparátoru LM393 a obsahuje jeden analogový a jeden digitální výstup sloužící pro detekci překročení kritické hranice obsahu měřených hodnot v okolí. Detekovatelná koncentrace CO je od 10 do 1000 ppm a detekovatelná koncentrace hořlavých plynů je od 100 do 10000 ppm. Senzor obsahuje celkem čtyři vývody. První z nich se připojí na analogový pin procesoru a je použit pro měření hodnot. Druhý, digitální pin, který slouží ke zmiňované detekci překročení kritické hranice měřených hodnot, necháme nepřipojený. Na třetí pin připojíme zem (GND) a na čtvrtý napájení 5 V [21].

Senzor teploty a vlhkosti HTU21D

Měřicí senzor HTU21D komunikuje přes rozhraní I^2C (multi-masterová počítačová sériová sběrnice, používána k připojování nízkorychlostních periférií k základní desce [22]). Měřicí rozsahy jsou u teploty od -40 do 125 °C, vlhkost lze měřit v rozsahu od 0 do 100 %. Senzor měří s přesností $0,3$ °C u teploty a 1 % u vlhkosti. Napájecí napětí může být v rozsahu od 3,3 V do 5 V, Odebíraný proud je velmi nízký maximálně 0,5 mA při měření. Modul obsahuje 4 propojovací piny. První pin (VIN) se propojí s napájecím napětím (v našem případě s 3,3 V). Druhý pin se připojí na zem (GND). Třetí (SCL) a čtvrtý (SDA) pin slouží pro komunikaci přes rozhraní I^2C , je nutné je zapojit na piny procesoru, které podporují komunikaci přes I^2C [23].

NB-IoT modul SARA-N210

NB-IoT modul od firmy ublox. Rozsah provozních teplot je od -40 do 85 °C, s extrémně nízkou spotřebou energie a rychlostmi přenosu dat až 62,5 kb/s pro uplink a až 27,2 kb/s pro downlink. Pro napájení je potřeba napětí 3,6 V to nám zajistí Li-pol baterie, která je součástí našeho zařízení. Modul poskytuje interně generované napětí 1,8 V (V_INT), které může posloužit jako napájení externích zařízení (například signalizační LED). Součástí modulu musí být i anténa, tu je možno připojit k pinu (ANT) s impedancí 50 Ω . Modul obsahuje i piny pro systémové funkce, jako je resetování modulu. Pro vzdálenou komunikaci je zapotřebí SIM karta, kterou nám umožňuje připojit SIM rozhraní skládající se z pinů VSIM, SIM_IO, SIM_CLK a SIM_RST. Dále na modulu najdeme sériové linky (UART, DDC) a GPIO (uni-verzální vstup/výstup) [24].

Procesor STM32F103CBT6

Procesor je jedna z nejdůležitějších součástí ve schématu. Náš konkrétní procesor poskytuje nízkonákladovou platformu, s relativně nízkým počtem pinů a nízkým příkonem, dodává dobrý výpočetní výkon a pokročilou reakci systému na přerušení. Procesor pracuje při napětí od 2 do 3,6V. Největší část pinů procesoru tvoří vstupně výstupní piny (I/O) a druhou skupinu tvoří piny určené k napájení. Procesor je umístěn v pouzdře LQFP48 a tím pádem má celkem 48 pinů [25].

Nabíjecí zařízení MAX8903

MAX8903 je integrovaná 1 článková Li + nabíječka s duálním napájením (z USB a napájecího adaptéru). Je vybavena optimalizovaným inteligentním řízením výkonu, aby bylo co nejlépe možno využít omezené napájení z USB nebo adaptéru. Výběh automatického vstupu přepíná systém z baterie na externí napájení. Pro naši

potřebu využijeme pouze napájení z USB, které pracuje od 4,1 V do 6,3 V s ochranou až do 8 V. Nabíječka interně blokuje proud z baterie do USB a adaptéru, když není k dispozici žádné vstupní napájení. Mezi další funkce patří předběžné hodnocení nabíjení, časovač, rychlý časovač nabíjení, přepětová ochrana, stav nabíjení a poruchové výstupy, monitoring teploty baterií. Zařízení je k dispozici v pouzdře QFN 4 x 4 mm s 28 piny [26].

3.3 Výběr krabičky

Krabička je nedílnou součástí zařízení. Poskytuje ochranu uživatele před úrazem elektrickým proudem a také je to ochrana zařízení před vlivy prostředí.

První důležitou funkcí je znemožnění dotyku součástí z bezpečnostních důvodů, ať už jde o bezpečnost uživatele, nebo bezpečnost přenášených signálů. Pokud v zařízení vzniká nebezpečné napětí, tak je nutno, pro ochranu uživatele, aby krabička byla z nevodivého materiálu. Případně, když je kryt z kovu, je nutno aby byla „uzemněna“ (vodivě spojit kryt s ochranným vodičem PE nebo sdruženým ochranným vodičem PEN). Uzemněné mají navíc výhodu v tom, že zmenšují elektromagnetické vyzařování zařízení a stíní před vlivem vnějších polí.

Druhou důležitou funkcí krabičky je ochrana před vlivem prostředí, což zahrnuje i ochranu proti vodě a vniknutí pevných těles. K určení vlastností krytí se používá stupeň krytí (IP), které se vyjadřuje pomocí IP kódu. IP kód je tvořen znaky IP a následně dvěma číslicemi. První z nich udává ochranu před nebezpečným dotykem a před vniknutím cizích předmětů. Druhá číslice znázorňuje krytí před vniknutím vody. Z Tab.3.1 lze vyčíst jednotlivé stupně ochrany.

U krabiček musíme rozlišovat i materiály, z kterých jsou vyrobeny. Z kovových krabiček je nejvíce používán hliník, který je oblíbený zejména pro jeho lehkost, snadnost mechanického opracování a odolností vůči korozi. Ocelové se vyrábějí méně a jsou spíše na osobní požadavek zákazníka.

U plastových krabiček je obecně větší množství materiálu na výběr. Nejčastěji používaný plast je ABS (Acrylonitrile Butadiene Styrene), mezi jehož vlastnosti patří vysoká tvrdost, odolnost vůči poškrábání a nárazům, vysoká citlivost na působení esterů a kyselin a také je citlivý na UV záření, díky kterému může měnit barvu. Pro levné krabičky je použit polystyren, je tvrdý ale křehký a snadno se odbarvuje. Je to jedna z výrobně nejlevnějších hmot. Je určen pro výrobu levných krabiček, bez zvláštních mechanických vlastností. Posledním materiálem je polykarbonát, což je plastická hmota s dobrými mechanickými vlastnostmi a srovnatelnou odolností s hliníkem. Je průhledný a také poměrně drahý, což snižuje jeho popularitu [27].

Výběr krabičky spočíval v uvědomění se všech rizik a zároveň v odhadu velikosti desky plošných spojů. Kvůli blízkosti vody, byla vybrána krabička s IP 66, která

slibuje ochranu proti intenzivně tryskající vodě ze všech úhlů, v množství sto litrů za minutu, při tlaku až 100 kN/m^2 , ze vzdálenosti tří metrů po dobu tří minut. Zařízení bude muset být upevněno, aby nedošlo ke spadnutí do vody. Byla vybrána krabička 1554K2GY od firmy Hammond Manufacturing, viz Obr.

Tab. 3.1: Tabulka stupně krytí IP.

Stupeň	Ochrana před nebezpečným dotykem	Ochrana před vniknutím cizích předmětů	Stupeň	Ochrana před vniknutím vody
IP 0x	bez ochrany	bez ochrany	IP x0	bez ochrany
IP 1x	dlaní (>5x5 cm)	velkých	IP x1	Chráněno proti kapající vodě 1+0,5mm za minutu.
IP 2x	prstem (>12x12mm)	malých	IP x2	Chráněno proti kapající vodě 3+0,5mm za minutu.
IP 3x	nástrojem (>2,5mm)	drobných	IP x3	Chráněno proti vodní tříšti
IP 4x	nástrojem, drátem (>2,5mm)	velmi drobných	IP x4	Chráněno proti stříkající vodě
IP 5x	jakoukoliv pomůckou	částečně prachu	IP x5	Chráněno proti tryskající vodě
IP 6x	jakoukoliv pomůckou	prachu úplně	IP x6	Chráněno proti intenzivně tryskající vodě
			IP x7	Chráněno proti ponožení do vody
			IP x8	Chráněno proti potopení do vody
			IP x9	Chráněno proti tlakové vodě

3.4 Návrh schématu

Úkolem návrhu schématu bylo propojit výše popsané komponenty, k tomu jsou samozřejmě potřeba i další neméně významné součástky. Úkolem bylo do části schématu obsahující, součásti modulu pro komunikaci po NB-IoT a část vstupů a výstupů z procesoru, vyřešit připojení jednotlivých senzorů k procesoru.

Pro zjištění toho, na kterých pinech mohou jednotlivé senzory komunikovat využijeme katalogové listy jednotlivých senzorů a procesoru. Zároveň bylo nutné zjistit jestli k připojení senzorů k procesoru nejsou potřeba další součástky (typicky kondenzátory a rezistory). Pokud jsou, je nutné tam součástku přidat a nastavit správnou velikost pouzdra.

Návrh schématu probíhal v programu Altium Designer, což je software uzpůsobený pro návrh DPS. Vývoj softwaru Altium Designer zaštiťuje společnost Altium Limited se sídlem v Austrálii.

Jedním z úkolů, který bylo nutno při návrhu schéma zařízení, bylo vyřešit napájení některých senzorů. Problémem bylo, že Li-ion (Lithium-iontová) baterie má pouze 3,7 V, ale pro napájení tří senzorů je potřeba 5 V. Konkrétně jde o senzory

- Modul a sonda pro měření pH,
- CO senzor oxidu uhelnatého.
- Optický senzor kvality ovzduší

Jedna z možností řešení tohoto problému je pomocí stejnosměrného regulátoru zvyšujícího napětí. Problém by šlo vyřešit také pomocí přidání dalšího článku baterie, kdy by bylo napětí dodávané baterií zase příliš velké (7,4 V) z čeho vyplývá další regulace napětí. Na základě této úvahy bylo proto vybráno řešení pomocí stejnosměrného spínaného regulátoru zvyšujícího napětí.

Jako regulátor byla vybrána součástka MIC2288YD5-TR, což je stejnosměrný nastavitelný spínaný regulátor, zvyšující napětí. Na vstup je možnost připojit napětí od 2,5 V do 10 V a na výstupu je možno dostat napětí v rozmezí od 3,6 V do 34 V s výstupním proudem 1 A – což odpovídá požadavkům součtu proudových odběrů obou senzorů:

$$I_{O_{Max}} = I_{OCO} + I_{OpH} + I_{ODUST}, \quad (3.1)$$

kde I_{OCO} – proudový odběr senzoru CO, I_{OpH} – proudový odběr pH modulu a I_{ODUST} – proudový odběr senzoru prachu.

$$I_{O_{Max}} = 0,15 + 0,01 + 0,02(A) \quad (3.2)$$

$$I_{O_{Max}} = 0,18 A = 180 mA. \quad (3.3)$$

Z výpočtu proudu vyplývá, že námi zvolený regulátor s výstupním proudem 1 A je dostačující vzhledem k tomu, že je požadován výstupní proud minimálně 180 mA.

K regulátoru je potřeba vybrat další komponenty jako vstupní a výstupní kondenzátor, cívku, diodu a zpětnovazební rezistory. Požadavky na vstupní kondenzátor byly, aby hodnota kapacity byla vyšší než $1 \mu F$. Byl zvolen kondenzátor s hodnotou $10 \mu F$, kvůli častému výskytu ve schématu a proto je to výhodnější varianta a také poskytuje kvalitní filtraci napětí.

Výstupní kondenzátor byl zvolen na základě doporučení v katalogovém listu regulátoru, kde pro výstupní napětí menší než 6 V byl doporučen kondenzátor s hodnotou kapacity $22 \mu F$.

Cívka byla zvolena $10 \mu H$, na základě doporučení v katalogovém listu.

Jako další součástka byla zvolena schottkyho dioda MBRM140T3G, která je doporučována hlavně pro svůj nízký úbytek napětí a dobu zpětného zotavení.

U zpětnovazebních rezistorů byl dán vzorec

$$U_{out} = V_{ref} \cdot \left(\frac{R1}{R2} + 1 \right), \quad (3.4)$$

,kde $V_{ref} = 1,24 V$ pro určení rezistorů na základě požadovaného výstupního napětí (v našem případě 5 V). Zároveň je nutné aby rezistor R2 měl hodnotu rovnu nebo

menší než $5\text{ k}\Omega$. Byly vybrány rezistory $R_1=5,6\text{ k}\Omega$ a $R_2=1,8\text{ k}\Omega$, když tyto hodnoty dosadíme do vzorce

$$U_{out} = 1,24 \cdot \left(\frac{5,6}{1,8} + 1 \right) (V, \Omega) \quad (3.5)$$

$$U_{out} = 5,1\text{ V}, \quad (3.6)$$

což s malou odchylkou odpovídá požadovanému napětí. Tato odchylka nebude mít na funkci vliv, takže vybrané hodnoty rezistorů mohou být použity.

3.5 Návrh desky plošných spojů

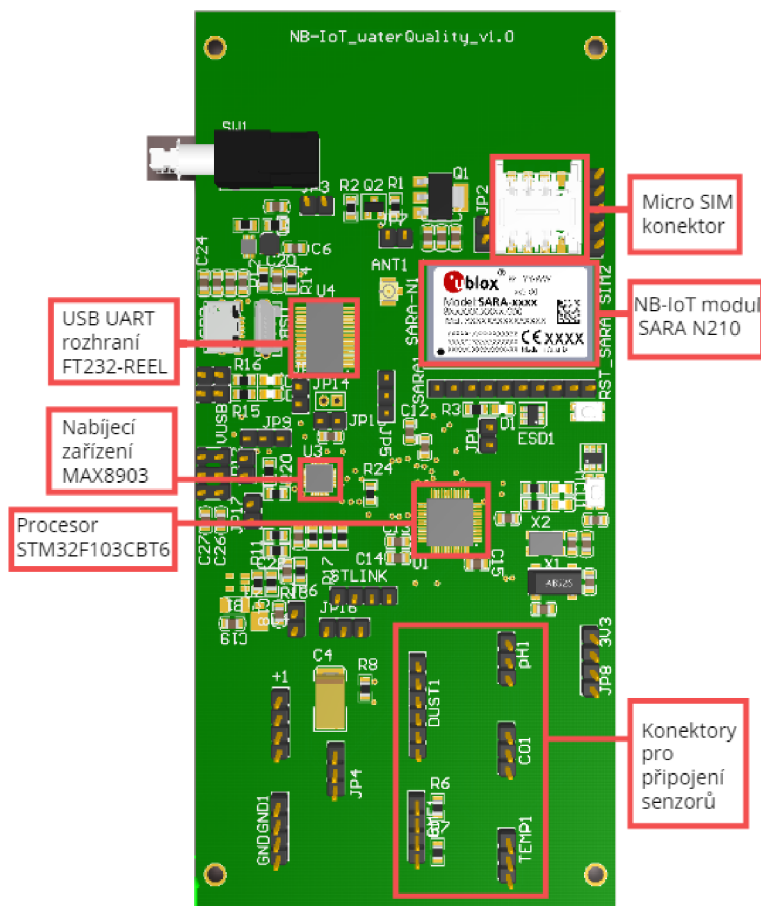
Návrh DPS, stejně jako schématu probíhal v programu Altium Desginer, který podporuje velké množství nastavení a možností pro tvorbu DPS.

Při návrhu DPS bylo nejdříve nutné určit si velikost samotné desky, k tomu bylo nutné znát rozměry krabičky. Rozměry krabičky byly převzaty z katalogového listu. Na základě informací z katalogového listu byly stanoveny rozměry desky na $132 \times 64\text{ mm}$. Dále bylo nutné přidat díry na uchycení desky v krabičce, rozměry byly převzaty z katalogového listu. Kvůli velikosti děr na uchycení je nutné použít šrouby průměru M3 - tomu odpovídá díra o průměru $2,5\text{ mm}$.

Po umístění děr na uchycení bylo nutné rozmístit součástky na DPS. Mezi základní principy rozmístění součástek platí oddělovat jednotlivé funkční bloky, minimalizovat vzdálenosti mezi součástkami, rozmístění součástek tak, aby se minimalizovalo rušení. Při rozmístění je doporučeno nejdříve umístit součástky, se kterými později nebude možné hýbat. Pak se umístí procesor a další integrované obvody a konektory, které požadujeme dobře dostupné pro snadné připojení externích zařízení. Doporučené je si rozmístit si jednotlivé bloky (zdroj, výstupy apod.) a propojit je.

Z důvodu zjednodušení práce byla použita oboustranná DPS, která má vodivý obrazec z obou stran, zároveň pro bakalářskou práci nebylo nutné použít více jak dvouvrstvé desky (hlavně z důvodu ceny a jednoduchosti zařízení). Průchody mezi vrstvami (spodní a horní) se musí prokovovat. Jednotlivé umístění součástky se mezi sebou propojí podle schématu. Šířka spoje pro napájecí signál byla zvolena $0,5\text{ mm}$ a pro ostatní byla zvolena $0,3\text{ mm}$. Po propojení všech spojů, kromě signálu GND bylo nutné vyplnit zbylé místa tzv. polygony po obou stranách desky a ty poté připojit k signálu GND.

DPS se standardně opatřují nepájivou maskou, což je poloprůhledná izolační vrstva barvy (nejčastěji zelené). Nepájivá maska nechává odkryté jen plošky potřebné pro



Obr. 3.1: 3D model navržené DPS – rozmístění součástek.

pájení součástek. Nepájivá maska zlepšuje izolační vlastnosti desky a zabraňuje poškození vodivých cest a oxidaci. Zároveň je možné na nepájivou masku umístit potisk, na kterém jsou vyznačeny obrysy umístění součástek a jejich označení dle schématu.

Téměř všechny součástky jsou SMD (součástky určené pro povrchovou montáž), které se dnes již běžně používají téměř v každém sériově vyráběném elektrickém přístroji. Výjimku tvoří konektory a spínač na vypnutí přístroje. SMD mají na svém povrchu kontaktní plošky a připájí se na stejnou stranu, na které jsou osazeny. Na Obr.3.1 lze vidět rozmístěné součástky.

3.6 Programování zařízení

Další nedílnou součástí návrhu zařízení bylo vytvořit program pro navrhované zařízení. Nejdůležitějším úkolem je zajistit aby procesor vyčítal údaje z připojených senzorů a následně data odesílal uživateli.

Vzhledem ke skutečnosti, že nebyla dosud oživena DPS navrhovaného zařízení bylo

třeba programování provádět na vývojové desce s MCU (mikrokontrolér), který je použit pro naše zařízení.

3.6.1 Vývojové platformy

Bylo vybíráno ze dvou vývojových platforem

- Arduino,
- Mbed.

Arduino

Arduino je open-source vývojová platforma založená na snadno použitelném hardwaru i softwaru. K desce lze samozřejmě připojovat různé vstupní i výstupní periferie. Programování je založené na jazyce C++ a je využívána speciální Arduino knihovna kde jsou popsány jednotlivé příkazy.

Arduino používá k uložení dat EEPROM při odpojení. Má to mnoho nevýhod, jedna z nich je nízká paměť a limitovaný počet zápisů na EEPROM (který je sice vysoký – kolem 1 230 163, ale stále konečný).

Tím, že Arduino má vlastní knihovnu a většinou se počítá s napsáním kódu do jednoho zdrojového souboru, což může být u větších projektů dosti nepřehledné. Arduino má vlastní vývojové prostředí – Arduino IDE, které je jednoduché a přehledné, což je obrovskou výhodou, zvláště když jde o nováčky v oblasti programování mikrokontrolerů.

Nespornou výhodou Arduina je jeho obrovská komunita a tudíž i mnoho volně dostupných zdrojových kódů na internetu (existuje velké množství webů se zdrojovými kódy pro Arduino), což vždy nemusí být výhodou, jelikož existuje i velké množství kódu s chybami [28], [29].

Mbed

Mbed je vývojová platforma a operační systém pro IoT zařízení založené na 32bitových mikrokontrolérech ARM Cortex-M. Programování je podobně jako v Arduinu založené na jazyce C++ a nabízí jednu vlastní knihovnu Mbed.

Při připojení Mbed desky do počítače se mikrokontrolér chová jako běžný USB flash disk. Nahrávání programu probíhá jednoduchým vložením souboru s příponou bin do mikrokontroleru, který má běžně paměť v řádu Mb. Zároveň nám to umožní uložit i jiné typy souborů.

Mbed pracuje ve striktním C++ s vlastní Mbed knihovnou kde jsou definovány třídy pro práci se vstupy a výstupy mikrokontroleru. Mbed je méně vhodný pro

začátečníky, protože očekává už nějakou určitou znalost programovací jazyka C++, což nám umožňuje využít vlastností OOP (objektově orientovaného programování). Vývojová platforma Mbed obsahuje i online vývojové prostředí. Což je na jednu stranu dobře, že lze projekt otevřít online na jakémkoliv zařízení, bez předešlé instalace, ale na druhou stranu je nutné mít připojení k internetu a ovládání není natolik intuitivní jako u Arduina.

Komunita Mbed je v porovnání s Arduino komunitou mnohem menší, ale skládá se převážně ze zkušených programátorů, kteří své kódy poskytují zdarma na stránkách Mbed, z těchto stránek lze kód importovat přímo do vývojového prostředí, mnohdy je však nutno dodaný kód upravit pro potřeby funkce zařízení [30], [29].

Z výše probíraných důvodů byla zvolena vývojová platforma Mbed.

3.6.2 Vývojové desky

Dalším úkolem bylo na vhodné vývojové desce naprogramovat a odzkoušet všechny senzory. Programování započalo na vývojové desce **Blue Pill** s mikrokontrolerem STM32F103C8. Je hojně využívána hlavně díky nízké pořizovací ceně (kolem 2 \$), je dostupná na internetových obchodech typu eBay nebo Aliexpres. Deska vyniká svým malým rozměrem, kromě vstupně výstupních periférií mikrokontroleru deska obsahuje také vlastní LED a tlačítko reset. Napájení je možné buď pomocí externího zdroje napětí, nebo pomocí micro-B USB konektoru. I přes to, že vývojová deska obsahuje USB konektor, není se přes něj možné defaultně připojit k počítači a nahrát vytvořený program. Nahrávání tedy musí probíhat buď přes programátor od ST-linku nebo přes sériovou linku. V našem případě byla použita pro nahrávání softwaru do procesoru varianta přes sériovou linku, konkrétně přes převodník seriové linky na USB pomocí převodníku FTDI232.

S přihlédnutím k ceně je vývojová deska Blue Pill poměrně dobrá jako trvale fungující zařízení, ale pro složitější projekty se zdá prakticky nepoužitelná.

Mnohem lepší variantou se ukázala být vývojová deska **Nucleo-64** s mikrokontrolerem STM32F103RBT6. Vývojová deska nevyžaduje samostatnou sondu, protože má integrovaný ladící debugger/programátor ST-LINK /V2-1. Má oproti Blue Pill mnohonásobně větší rozměry. Napájení je možné pomocí USB VBUS nebo externího zdroje (3,3 V, 5 V nebo 7 – 12 V). Součástí desky je také uživatelsky programovatelná LED (LD2), dvě tlačítka a to USER (uživatelsky programovatelné) a RESET. Tři různá rozhraní podporovaná USB – virtuální COM port, velkokapacitní paměť (USB disk drive) pro programování drag'n'drop, Debugovací port.

3.6.3 AT příkazy

AT příkazy (známé též jako Hayesovi příkazy) jsou krátké sekvence znaků, pomocí kterých se ovládají funkce zařízení komunikující v mobilních sítích v našem případě se jedná o NB-IoT modul SARA-N210. AT příkazy konfigurují a aktivují funkce 3GPP dle specifikací u-bloxu. AT příkazy jsou do modulu předávány pomocí sériové linky se syntaxem:

"AT+<jméno_příkazu><parametry><S3_znaky>"

Kde:

- "AT" – prefix každého příkazu.
- <jméno_příkazu> – soubor znaků se jménem příkazu, může mít znak "+" jako prefix
- <parametry> – jednotlivé parametry, které mohou být buď číselného typu, nebo textové. Textové musí být v uvozovkách např. "T-Mobile", celé čísla mohou být samostatně. Parametry mezi sebou musí být odděleny znakem ",".
- <S3_znaky> – povel k ukončení příkazu v našem případě "\r\n"

Modul Sara obvykle neumožňuje přidat nový AT příkaz dokud nebyl předchozí přijat s finální odpovědí "OK" nebo s chybnou odpovědí. V některých případech je možno příkaz zrušit před tím, než se uskuteční.

AT příkazy použité v práci

- AT+NRB
- AT+CFUN=<fun>
- AT+COPS=<mode>,<format>,<oper>
- AT+CEREG?
- AT+CGDCONT=<cid>,<PDP_type>,<APN>
- AT+NSOCR=<type>,<protocol>,<listen_port>,<socket>
- AT+NSOSTF=<socket>,<remote_ip_address>,<remote_port>,<flag>,<length>,<data>
- AT+NSOMI:<socket>,<length>
- AT+NSORF=<socket><req_length>

AT+NRB Restartuje modul a vymaže uložené nastavení. Po zadání příkazu následuje krátké zpoždění, po kterém následuje odpověď REBOOTING, po 3 sekundách REBOOT_CAUSE_APPLICATION_AT; u-blox; OK.

AT+CFUN=<fun> Nastaví mód funkcionality, která je číselně vyjádřena parametrem <fun>, v modulu SARA-N210 jsou na výběr pouze 2 módy. Mód 0, který nastaví modul na minimální funkcionalitu (deaktivuje oba vysílací a přijímací RF

obvody). Mód 1 nastaví modul na plnou funkčnost. Očekávaná odpověď je OK. Pro účely bakalářské práce je použit mód 1.

AT+COPS=<mode>,<format>,<oper> Vynucuje pokus o výběr a registraci u provozovatele sítě GSM/UMTS/LTE. Pro správnou funkci konfigurace a registrace je nutná SIM karta. Nastavenou hodnotu lze ověřit příkazem AT+COPS?, který je použit v bakalářské práci. Seznam provozovatelů sítí lze dostat testovacím příkazem AT+COPS=?. Parametr <mode> se používá k výběru zda se volba sítě provádí automaticky (0), manuálně (1), případné odregistrování (2). Zadání probíhá v číselném formátu a pro účely bakalářské práce je použit manuální mód. Dalším parametrem je <format>, který nám určuje formát zadání, kde pro použitý modul je možnost pouze numerických zadání(2). Zadání probíhá v číselném formátu. Následuje parametr <oper>, kde podle parametru <formát> je vybrán formát zadání operátora. Pro účely bakalářské práce je použit operátor s číslem "23003". Očekávaná odpověď je OK.

AT+CEREG? Příkaz zjišťuje aktuální stav registrace. Pro účely bakalářské práce je použit pouze příkaz pro čtení, který vrátí stav registrace. Očekáváme, že nám příkaz vrátí zprávu "+CEREG: <n>,<stat>". Kde <n> značí mód konfigurace, jelikož nepoužíváme příkaz AT+CEREG k přenastavení této hodnoty, zůstane výchozí hodnota, což je 0 (síťová registrace URC je zakázána). Pro účely bakalářské práce jsou mnohem zajímavější hodnoty <stat>, které symbolizují stav registrace. Stavů může být celkem 7, a jsou vráceny celým číslem, stejně jako parametr <n>. 0 značí, že modul nebyl registrován a neprobíhá žádný aktivní pokus o registraci. 1 značí, že je zaregistrován v domácí síti. 2 značí, že není registrován, ale snaží se o registraci. 3 značí, že registrace byla odepřena. 4 značí že stav je neznámý (např. není v pokrytí E-UTRAN). 5 značí že je registrován s roamingem. 8 značí, že je připojen pouze k službám nouzového nosiče. Pro správnou funkci očekáváme odpověď s hodnotou <stat> 5. Po odpovědi s hodnotami <n> a <stat> následuje odpověď OK.

AT+CGDCONT=<cid>,<PDP_type>,<APN> Definuje parametry připojení pro PDP(Packet Data Protokol) kontext. Kde parametr <cid> je identifikátor PDP kontextu, je numerický a určuje nám konkrétní definici PDP kontextu. V bakalářské práci je použit <cid> s číslem 1. Můžeme definovat až 10 PDP kontextů, ale aktivní může být pouze jeden. Parametr <PDP_type> je textový parametr, který specifikuje typ PDP, je několik možností a to buď „IP“, „NONIP“. Pro účely bakalářské práce je nastavena hodnota „IP“. Parametr <APN>, je textový parametr, který nese jméno přístupového bodu, což je logické jméno, platné v aktuální doméně PLMN, slouží k výběru GGSN nebo externí datové sítě, do které je

připojeno. Maximální délka je 99 znaků a pro účely bakalářské práce je použit název "GPRST.INTERNET". Příkaz obsahuje kromě příkazu pro nastavení i testovací a čtecí příkazy "AT+CGDCONT=?" a "AT+CGDCONT?", které vypíší seznam dostupných parametrů (testovací), nebo aktuální nastavení (čtecí). Očekávaná odpověď je OK.

AT+NSOCR=<type>,<protocol>,<listen_port>,<socket> Vytvoří socket, pro komunikaci se serverem. Textovým parametrem <type>, se nastavuje typ socketu, je podporována pouze hodnota „DGRAM“. Parametrem <protocol> standardní definice protokolu pro přenos, povolená číselná hodnota je 17 – UDP. Celočíselný parametr <listen_port> nám určuje lokální port, na kterém budou odesílány a přijímány zprávy v daném socketu. Povolený rozsah je od 0 do 65535 s výjimkou portu 5683. Pro účely bakalářské práce je používán port 50105. V celočíselném parametru <socket> nastavujeme číslo socketu, které je použito jako identifikátor socketu pro další příkazy. V bakalářské práci je použito číslo 0. Za zmínku stojí i parametr <receive_control>, který nám určuje jakým způsobem bude nakládáno s příchozími zprávami, kde 0 značí, že zprávy budou ignorovány a 1 (defaultní hodnota) značí, že příchozí zprávy automaticky generují příkaz „+NSONMI“, tato hodnota je pro účely bakalářské práce ponechána na výchozí hodnotě. Očekávaná odpověď je číslo socketu, po kterém následuje OK.

AT+NSOSTF=<socket>,<remote_ip_address>,<remote_port>,<flag>,<length>,<data> Odešle UDP datagram do zadaného portu hostitele a nastaví příznaky metadat (flag). Celočíselný parametr <socket> slouží pro identifikaci socketu na který bude odeslána zpráva. Textový parametr <remote_ip_address> identifikuje IP adresu ve formátu IPv4 na kterou je zpráva odeslána, je možno ji zadat v desítkové, osmičkové, nebo šestnáctkové soustavě. Pro účely bakalářské práce je použita adresa serveru „147.229.146.40“. Parametr <remote_port> označuje port na kterém bude zpráva přijata, nastavitelný v rozmezí od 0 do 65535. Pro účely bakalářské práce je použit port 50105. Parametr <flag> určuje typ přenosu zprávy v šestnáctkovém formátu, v bakalářské práci je použita hodnota 0x200. Celočíselný parametr <length> nám vyjadřuje délku posílaných dat v bajtech. Maximální délka je 512 bajtů. Textový parametr <data> nám vyjadřuje posílaná data v šestnáctkové soustavě. Očekávaná odpověď je číslo socketu na který byla poslána zpráva a délka dat v bajtech, které byly odeslána, následuje potvrzení OK.

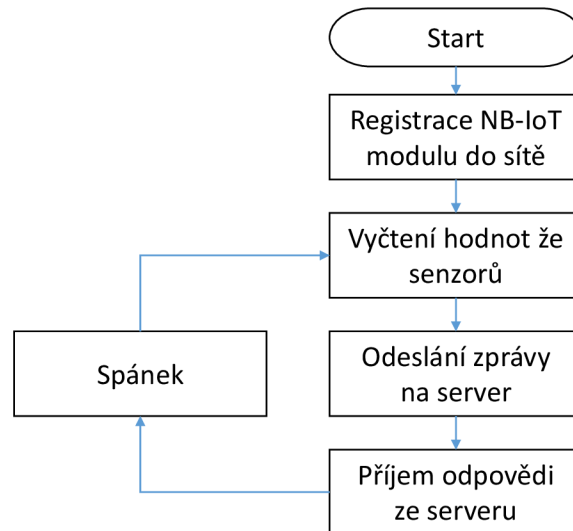
AT+NSOMI:<socket>,<length> Oznámí prostřednictvím URC, že byla přijata data na určitém socketu a že jsou připravena ke čtení. Vrací číslo socketu a počet bajtů dat dostupných ke čtení pro první zprávu, která je ve frontě. Následující zpráva

přijatá ve stejném soketu bude zařazena do fronty a bude vydána až bude předchozí zpráva přečtena. Tato zpráva se generuje automaticky (pokud je povolena pro daný soket), čili není třeba pro ni psát žádný AT příkaz. Celočíslný parametr <socket> nám určuje na kterém soketu byla zpráva přijata a celočíselný parametr <length> zobrazuje délku přijaté zprávy v bajtech.

AT+NSORF=<socket><req_length> Přijme data na soketu následně vypíše přijatou zprávu. Celočíslný parametr <socket> určuje soket z kterého chceme přečíst zprávu. Celočíslný parametr <req_length> nám určuje délku dat v bajtech, kterou nám přijatá zpráva vrátí. Očekáváme odpověď ve formátu <socket>, <ip_addr>, <port>, <length>, <data>, <remaining_length>, kde <ip_addr> je textový řetězec s IP adresou odesílatele, <port> je celočíselný parametr znázorňující z kterého portu byla zpráva poslána, celočíselný parametr <length> znázorňuje délku přijatých dat v bajtech. Textový řetězec <data> reprezentuje přijatá data v šestnáctkové soustavě a celočíselný parametr <remaining_length> znamená kolik bajtů z přijatých dat je ještě nepřečtených. Po vyčtení uvedených parametrů následuje potvrzující odpověď OK[31].

3.6.4 Popis programového řešení

Programové řešení bylo napsáno v programovacím jazyce C++, kdy nejdříve byla stanovena určitá posloupnost toho, co by zařízení mělo vykonávat a v jakém pořadí, což nám znázorňuje vývojový diagram, který je vidět na Obr. 5.6.



Obr. 3.2: Vývojový diagram obsluhy NB-IoT modulu

Kód byl napsán v online vývojovém prostředí Mbed. Součástí kódu jsou dvě napsané

knihovny, jedna slouží pro ovládání senzorů a druhá pro práci s NB-IoT modulem SARA-N210. Kromě zmíněných knihoven je použita knihovna mbed.h, která slouží k základním operacím s použitým procesorem, dále knihovna htu21d.h sloužící jako knihovna pro práci se senzorem teploty a vlhkosti htu21d, poslední použitou knihovnou je BufferedSerial.h, která slouží pro práci se sériovou linkou. Celá aplikace běží v souboru main.cpp. Sepsané metody jsou znázorněny v Tab. 4.2 a 4.3.

Tab. 3.2: Tabulka vytvořených metod v knihovně sara

Metoda	Vstupní a výstupní parametry	Popis
int write(char *s)	s - Textový řetězec obsahující AT příkaz	Umožňuje zadat AT příkaz přes sériovou linku.
	0 pokud byl příkaz zadán správně a 1 pokud nastala chyba	
int readPrint()	0 pokud je odpověď "OK", 1 pokud nastala chyba nebo v zásobníku není žádná odpověď, 2 zařízení je správně zaregistrováno	Vypíše odpověď z modulu přes sériovou linku na sériovou linku PC
int zadaniPrikazu(char *s)	s - Textový řetězec obsahující AT příkaz.	Zašle AT příkaz a následně vypíše odpověď na PC
	0, zadání a vyčtení odpovědi proběhlo v pořádku, 1 nastala chyba při čtení nebo zápisu, 2 zařízení je správně zaregistrováno	
int registrace()	0 značí že všechny příkazy proběhly správně, 1 značí že v některém nastala chyba	Provede registraci modulu do sítě
int poslaniZpravy(char *z)	z - Textově řetězec obsahující zadanou zprávu, která je poslána na vzdálený server	Odešle zadanou zprávu na vzdálený server
	0 značí že byla zpráva odeslána, 1 značí že se vyskytla chyba	
int odpovedServer()	0 značí že zpráva byla přijata a vypsána správně, 1 někde vznikla chyba	Vypíše přijatou zprávu
int poslaniZpravySenzory()	0 značí že zpráva byla odeslána správně, 1 někde vznikla chyba	Provede měření hodnot připojených senzorů a následně je odešle na vzdálený server
int hexToInt (char c)	c - znak, který má být převeden na celé číslo typu int.	Převede hexadecimální znak na celočíselnou hodnotu
	Znak vyjádřený v desítkové soustavě	
int hexToAscii (char c, char d)	c - první znak z dvojice, d - druhý znak z dvojice.	Převede dvojici hex. znaků na ascii znak
	Ascii znak vyjádřený v desítkové soustavě	

Tab. 3.3: Tabulka vytvořených metod v knihovně senzory

Funkce	Návratová hodnota	Popis
float readPh()	naměřená hodnota Ph, -1000 značí, že byla naměřena nereálná hodnota	Změří hodnotu Ph
float kalibraceCO()	je změřená konstanta R0	Změří hodnotu konstanty R0
int mereniCO()	změřená koncentrace CO v jednotce ppm	Změří hodnotu koncentrace CO vztahenou k R0, v jednotce ppm.
int htdu21dInit()	0 pokud zařízení pracuje správně, 1 pokud je někde chyba	Inicializace senzoru teploty a vlhkosti htdu21d
float htdu21dTemp()	teplota vzduchu, pokud se nepodařila teplota vyčíst, je návratová hodnota -1000	Změří hodnotu teploty vzduchu
float htdu21dHum()	vlhkost vzduchu, pokud se nepodařila vlhkost vyčíst, je návratová hodnota -1000	Změří hodnotu vlhkosti vzduchu

Funkcionalita byla otestována se zapůjčeným NB-IoT modulem SARA-N210 (bližší specifikace v Tab. 4.4) na samostatné desce, která byla propojena s vývojovou deskou NUCLEO-F103RB přes sériovou linku. K vývojové desce byli připojeni tři senzory, první byl z nich byl pH modul s připojenou sondou, druhý byl senzor oxidu uhelnatého MQ-9 a posledním byl senzor teploty a vlhkosti htdu21d. Vývojová deska byla také připojena k počítači na USB port, který kromě napájení zařízení umožňoval i výpis na sériovou linku počítače. Server je nastaven tak aby posílal potvrzovací zprávu, která je podobná jako odeslaná zpráva. Pro kontrolu prováděných operací byli vypisovány krátké zprávy na sériové lince připojeného počítače a v příkazovém řádku serveru. Specifikace serveru je znázorněna v Tab 4.5.

Tab. 3.4: Specifikace NB-IoT modulu

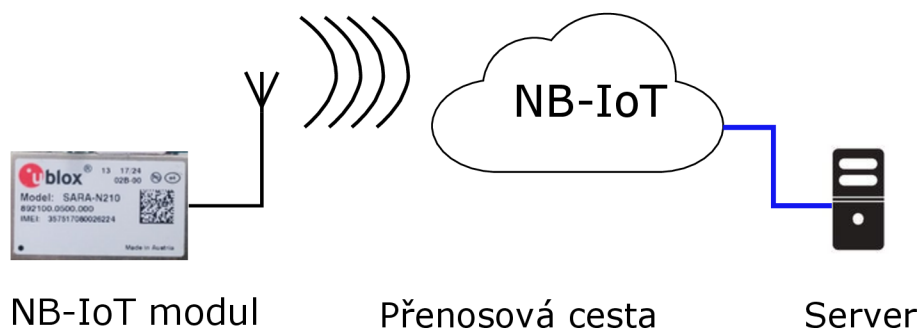
Výrobce	Typ	Verze firmware	Revize HW
U-blox	SARA-N210	6.57	892100.0500.000

Tab. 3.5: Specifikace serveru

Název	Distributor	Verze
Vodafone NB-IoT Server	Ubuntu	16.04

Topologie testovacího scénáře je znázorněna na Obr. 4.5. Test funkcionality spočíval v registraci NB-IoT modulu do sítě (použití metody registrace). Registrace se skládala z posloupnosti určitých AT příkazů. Prvním z nich je příkaz "AT+NRB", kterým se modul restartuje a vymaže předchozí nastavení. Dalším příkazem je "AT+CFUN=1", který nastavuje mód funkcionality modulu, tedy plnou funkčnost. Následujícím příkazem "AT+COPS=1,2,"23003" provedeme registraci k síti vodafone, která je určena posledním parametrem. Následně je po jedné

sekundě zjišťován stav registrace příkazem "AT+CEREG?" do té doby, než je modul registrován do sítě (odpověď "+CEREG:0,5"). Následně jsou pomocí příkazu "AT+CGDCONT=1,"IP","GPRS.INTERNET" definovány parametry připojení pro PDP. Dalším příkazem "AT+NSOCR="DGRAM",17,50105,0" je vytvořen soket, pomocí kterého se posílají zprávy na vzdálený server pomocí protokolu UDP na portu 50105. Touto sérií AT příkazů je vše připraveno k poslání zprávy na server. Následně je pomocí metody poslaniZpravySenzory() odeslána zpráva obsahující změřené parametry na server. Pomocí stejné metody je vytvořen textový řetězec obsahující změřené parametry – hodnotu pH, koncentraci CO, teplotu vzduchu a vlhkost vzduchu. K vytvořenému textovému řetězci je přidána časová značka jako prefix, následuje zpráva se změřenými hodnotami a postfix. Všechny tři části zprávy jsou odděleny středníkem. Vytvořená zpráva je v ascii formátu použita jako vstupní parametr funkci poslaniZpravy(char *z), která zjistí délku zprávy, převede ji z ascii formátu na hexadecimální formát a odešle ji na vzdálený server pomocí příkazu "AT+NSOSTF=0,147.229.146.40,50105,0x200,<délka_zprávy>,<zpráva>". Jako odpověď příkazu vidíme číslo soketu přes který byla zpráva poslána a počet bajtů zprávy. Po zhruba dvou sekundách je zapnuto čtení ze sériové linky, kde v případě že server zprávu přijal a odeslal odpověď se objeví informace o tom že byla přijata zpráva – "+NSOMI:<socket>,<délka>" obsahující informaci o tom na jakém soketu a s jakou délkou byla zpráva přijata. Zároveň je zjištěna délka přijaté zprávy, která je uložena do proměnné delkaZpravy. Poslední metodou v cyklu je odpovedServer(), která zadá AT příkaz se zjištěnou délkou zprávy "AT+NSORF=<socket>,<delka_zpravy>". Zadaný příkaz umožňuje přístup k přijaté zprávě, ta je následně převedena z hexadecimálního formátu na ascii formát a vypsána na sériovou linku připojeného počítače.



Obr. 3.3: Topologie testovacího scénáře

Ověření probíhalo pomocí výpisů z terminálů, pro výpis na straně NB-IoT mo-

dulu probíhal výpis v programu terminal, který se umožňuje přijímat data posílaná na sériovou linku počítače viz výpis 3.1. Na straně serveru bylo spojení realizováno pomocí programu putty, připojením na SSH serveru na který byla zpráva odeslána, viz výpis 3.2. Ve výpisu jsou pro větší přehlednost rozlišeny barvy. Červená značí AT příkaz poslaný z MCU do NB-IoT modulu, modrá značí odpověď NB-IoT modulu odeslanou do MCU, zelená značí odeslanou nebo přijatou zprávu ze vzdáleného serveru v ascii formátu a černou barvou jsou označeny ostatní informace.

Výpis 3.1: Výpis na straně NB-IoT modulu

```
1 Odesle se zprava: 1498478203392;6.45,6,24.80,52.43;5
2 Posilam prikaz: AT+NSOSTF=0,"147.229.146.40",50105,0x200,34,
3 "313439383437383230333339323B362E34352C362C32342E38302C35322E34333B35"
4
5 0,34
6 OK
7
8 +NSONMI: 0,34
9 Posilam prikaz: AT+NSORF=0,34
10
11 0,"147.229.146.40",50105,34,
12 "313532373439373631353836363B36E34352C362C32342E38302C35322E34333B34",0
13 1527497615866;6.45,6,24.80,52.43;4
14
15 OK
```

Výpis 3.2: Výpis na straně serveru

```
1 Current time: 8:53:35.865
2 Current Date: 2018-05-28
3
4 INCOMING PACKET FROM: /195.233.151.108
5 PORT:32120
6
7 RECEIVED DATA: 1498478203392;6.45,6,24.80,52.43;5
8 Send time SARA: 1498478203392 11:56:43.392
9
10 Receive time server: 1527497615865 8:53:35.865
11 Transmission time: 29019412473 ms - 8:56:52.473
12
13 Sent data back to SARA: 1527497615866;6.45,6,24.80,52.43;4
```

4 ZÁVĚR

Předkládaná bakalářská práce se zabývala návrhem IoT zařízení umožňujícího měření vybraných parametrů kvality vody a ovzduší, komunikující přes bezdrátovou technologii NB-IoT. Jedním z prvních úkolů bylo porovnat stávající LPWAN sítě. Byli vybrány celkem čtyři sítě, konkrétně LoRa, Wi-Fi HaLow, Sigfox a NB-IoT. Na základě průzkumu a porovnání vlastností uvedených LPWAN sítí byly určeny výhody a nevýhody jednotlivých sítí. Porovnáním stávajících LPWAN sítí byl zjištěn velký potenciál NB-IoT sítě, u které byly zjištěny dobré vlastnosti využitelné v IoT. Dalším úkolem byl návrh scénáře IoT zařízení a definice požadavků na zařízení. Bylo nutné zjistit využitelnost IoT zařízení, to spočívalo v rozdělení do osmi odvětví a uvedení konkrétních příkladů možnosti praktického využití. Následně byly vybrány tři možnosti návrhu konkrétního IoT zařízení. Jedním z nich byl inteligentní odpadkový koš, další bylo zařízení, které přečte pomocí NFC data z mobilního telefonu a poslední bylo zařízení monitorující kvalitu vody a ovzduší. Z návrhů bylo vybráno zařízení monitorující kvalitu vody a ovzduší.

Následujícím úkolem bylo navrhnout schéma zapojení NB-IoT zařízení. Praktická část pokračovala návrhem desky plošných spojů, která byla navržena dle pravidel popsanych v kapitole 3.5. Krabice byla vybrána, tak aby splňovala parametry pro ochranu zařízení alespoň IP 66 vzhledem k předpokládanému umístění zařízení v blízkosti vodních ploch.

V závěrečné části bylo potřeba vytvořit program umožňující definovanou funkci zařízení (odesílat údaje o naměřených hodnotách). Na základě toho byly vytvořeny dvě knihovny. První z nich, *sara.cpp* slouží pro ovládání NB-IoT modulu, obsahující metody pro registraci, posílání, přijetí zprávy a pomocné metody. Druhá knihovna *senzory.cpp* slouží pro vyčítání hodnot ze senzoru na měření pH, koncentrace CO, teploty a vlhkosti vzduchu. Vývoj probíhal na vývojové desce NUCLEO-F103RB, obsahující stejný procesor jako ve finálním řešení zařízení. K procesoru byly připojeny tři senzory, zbylé dva nebyly z časových důvodů implementovány. Vývojová deska byla kromě senzorů připojena na NB-IoT modul SARA-N210, na kterém byla otestována kompletní funkčnost programového řešení zařízení. Zařízení z důvodu časové náročnosti programování zařízení nebylo možné vyrobit, ale funkčnost programového řešení byla otestována propojením vývojové desky a NB-IoT modulu. Dále proběhl test konektivity, kdy byl NB-IoT modul SARA-N210 zaregistrován do NB-IoT sítě, následně byla poslána zpráva na vzdálený server VUT – „Vodafone NB-IoT Sever“ a na závěr přijata a vypsána odpověď serveru.

LITERATURA

- [1] Rapeepat Ratasuk, Benny Vejlgaard, Nitin Mangalvedhe, and Amitava Ghosh. NB-IoT system for M2M communication. In *Wireless Communications and Networking Conference (WCNC), 2016 IEEE*, pages 1–5. IEEE, 2016.
- [2] Hasnain Yousaf. Internet of Things:“A panoramic observation”. In *International Conference on Communication Technologies (ComTech)*, pages 27–33. IEEE, 2017.
- [3] Lukas Krupka, Lukas Vojtech, and Marek Neruda. The issue of LPWAN technology coexistence in IoT environment. In *17th International Conference in Mechatronics-Mechatronika (ME)*, pages 1–8. IEEE, 2016.
- [4] Min Chen, Yiming Miao, Yixue Hao, and Kai Hwang. Narrow Band Internet of Things. *IEEE Access*, 2017.
- [5] *Sigfox Developer Portal* [online]. Sigfox [cit. 2017-11-28]. URL: <http://makers.sigfox.com/>.
- [6] Nikolay S Knyazev, Victor A Chechetkin, and Denis A Letavin. Comparative analysis of standards for Low-power Wide-area Network. In *Systems of Signal Synchronization, Generating and Processing in Telecommunications (SINKHROINFO)*, pages 1–4. IEEE, 2017.
- [7] *IoT portál – Brána do světa internetu věcí* [online]. IoT portál, 2017 [cit. 2017-11-23]. URL: <https://www.iot-portal.cz>.
- [8] SS Navghane, MS Killedar, and VM Rohokale. IoT based smart garbage and waste collection bin. *Int. J. Adv. Res. Electron. Commun. Eng*, 5(5):1576–1578, 2016.
- [9] DVOŘÁK Jan . *Regulátor pH pro akvárium*. Brno, 2013. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky. Vedoucí práce Doc. Ing. Jiří Šebesta, Ph.D.
- [10] Rozbor vody - parametry. *CENTRUM VODY* [online]. Praha, 2016 [cit. 2017-10-14]. URL: http://www.centrumvody.cz/rozbor-vody-parametry/cs_CZ-129815.html.
- [11] pH and chlorine values swimming pool - Water quality pool. *Pahlén* [online]. Upplands Väsby, Švédsko, 2016 [cit. 2017-10-12]. URL: <http://www.pahlen.com/users-guide/ph-and-chlorine>.

- [12] Vodní ekotoxikologie. *Mendelu* [online]. Brno [cit. 2017-10-16]. URL: http://web2.mendelu.cz/af_291_projekty2/vseo/print.php?page=2303&typ=html.
- [13] Konduktivita. *Analýza vody* [online]. Uherský Brod: Petr Sobol, 2016 [cit. 2017-10-17]. URL: <http://www.analyzavody.cz/vlastnosti-vody/konduktivita/>.
- [14] Radek Maceška. Měření vodivosti kapalin. Brno, 2009. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií.
- [15] Měření koncentrace prachu a senzory prachu. *Automatizace hw* [online]. Antonín Vojáček, 2006 [cit. 2018-05-28]. URL: <https://automatizace.hw.cz/clanek/2006111201>.
- [16] KADAŇKA, Karel. . *Meteorologická stanice pro Iontmetr*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Vedoucí práce Ing. Zoltán Szabó, PhD.
- [17] Meteorologie. *Meteorologie - Meteostanice, digitální teploměry a barometry - GARNI technology* [online]. Ostrava: GARNI [cit. 2017-11-29]. URL: <http://www.garni-meteo.cz/cz/meteorologie/>.
- [18] Vodotěsný digitální teploměr. *Arduino shop* [online]. Havlíčkův Brod, 2016 [cit. 2017-12-10]. URL: <http://navody.arduino-shop.cz/navody-k-produktum/teplotni-senzor-ds18b20.html>.
- [19] Modul a sonda pro měření pH. *Arduino shop* [online]. Havlíčkův Brod, 2017 [cit. 2017-12-10]. URL: <http://navody.arduino-shop.cz/navody-k-produktum/modul-a-sonda-pro-mereni-ph.html>.
- [20] Optický senzor kvality ovzduší sharp gp2y1010au0f. *Arduino shop* [online]. Havlíčkův Brod, 2016 [cit. 2017-12-11]. URL: <http://navody.arduino-shop.cz/navody-k-produktum/opticky-senzor-kvality-ovzdusi-sharp-gp2y1010au0f.html>.
- [21] Co senzor oxidu uhelnatého. *Arduino shop* [online]. Havlíčkův Brod, 2017 [cit. 2017-12-14]. URL: <http://navody.arduino-shop.cz/navody-k-produktum/senzor-oxidu-uhelnateho-mq-9.html>.
- [22] Philips Semiconductors. The I2C-bus specification. *Philips Semiconductors*, 9397(750):00954, 2000.

- [23] Senzor teploty a vlhkosti htu21d i2c. *Arduino shop* [online]. Havlíčkův Brod, 2017 [cit. 2018-12-10]. URL: <http://navody.arduino-shop.cz/navody-k-produktum/senzor-teploty-a-vlhkosti-htu21d-i2c.html>.
- [24] SARA-N2 Power-optimized NB-IoT (LTE Cat NB1) modules. URL: https://www.u-blox.com/sites/default/files/SARA-N2_DataSheet_%28UBX-15025564%29.pdf.
- [25] STM32F103x8 STM32F103xB, Datasheet - production data. *STMicroelectronics* [online]. 2013 [cit. 2017-12-06]. URL: <http://www.st.com/content/ccc/resource/technical/document/datasheet/33/d4/6f/1d/df/0b/4c/6d/CD00161566.pdf/files/CD00161566.pdf/jcr:content/translations/en.CD00161566.pdf>.
- [26] MAX8903A-E/G/H/J/N/Y 2A 1-Cell Li+ DC-DC Chargers for USB and Adapter Power. *Maxim Integrated* [online]. San Jose: Maxim Integrated, 2011 [cit. 2017-12-04]. URL: <https://datasheets.maximintegrated.com/en/ds/MAX8903A.pdf>.
- [27] Hammond = krabička pro vaše zařízení. *DPS Elektronika od A do Z* [online]. Liberec: Marcin Kajzler, 2013 [cit. 2017-12-05]. URL: <http://www.dps-az.cz/soucastky/id:2712/hammond-krabicka-pro-vase-zarizeni>.
- [28] Co je to arduino? *Arduino.cz* [online]. [cit. 2018-02-23]. URL: <https://arduino.cz/co-je-to-arduino/>.
- [29] Mbed VS Arduino. *Sensor Lab* [online]. 2013 [cit. 2018-02-24]. URL: <http://slab.concordia.ca/2013/mbed/mbed-comparison-test/>.
- [30] *Mbed 6* [online]. Cambridge: Mbed, 2018 [cit. 2018-03-28]. URL: <https://www.mbed.com/en/>.
- [31] SARA-N2 Modules: Power-optimized NB-IoT modules. [online].R10. u-blox, [cit. 2018-03-23]. URL: https://www.u-blox.com/sites/default/files/SARA-N2_ATCommands_%28UBX-16014887%29.pdf.
- [32] 1554KINB. *Electrical & Electronic Enclosures, Cabinets & Racks, Outlet Strips and Electronic Transformers - Hammond Mfg.* [online]. Hammond Manufacturing, 2017 [cit. 2017-12-14]. URL: <http://www.hammondmfg.com/1554KINB.jpg>.

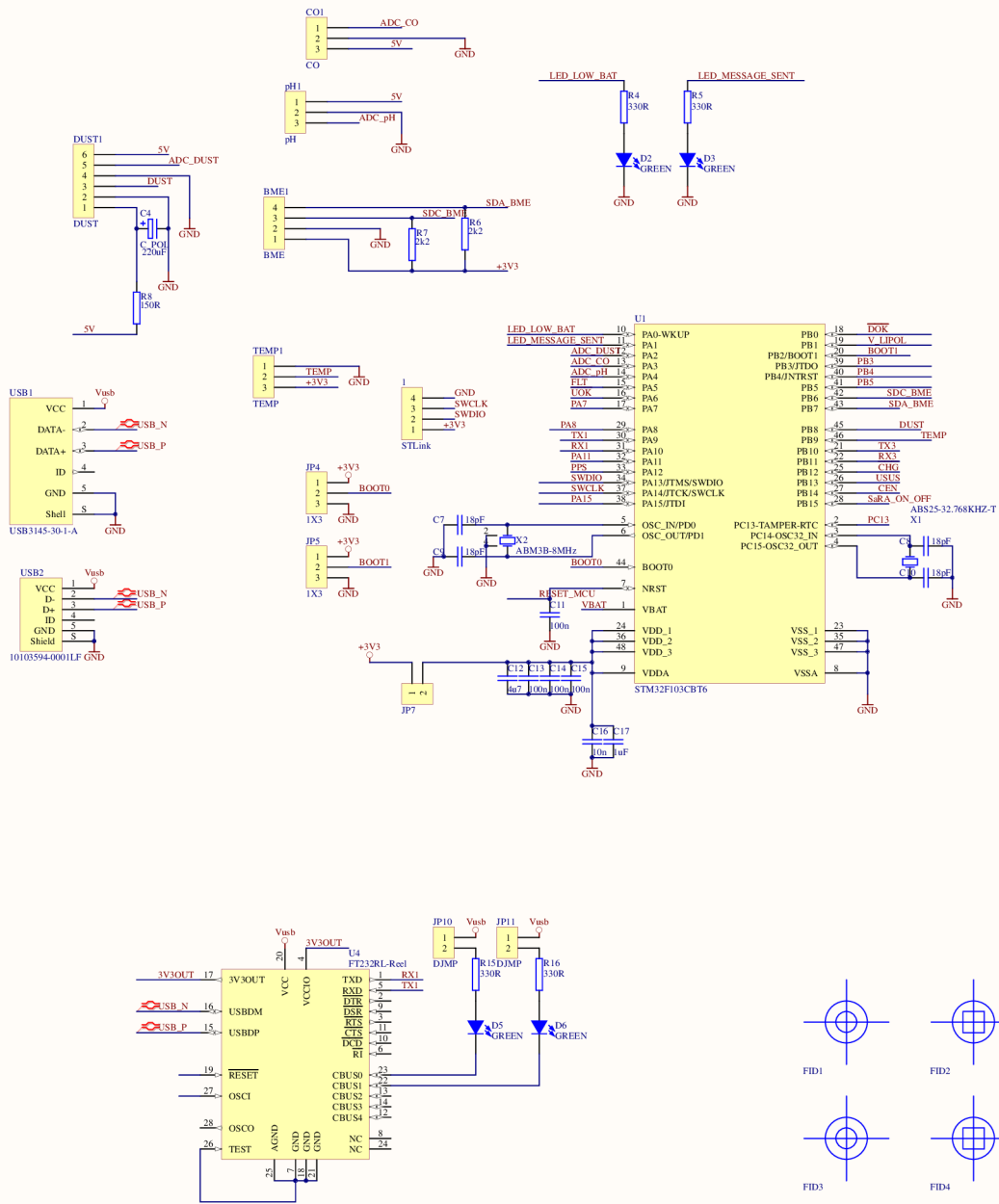
SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

3GPP	Partnerský Projekt 3. Generace – the 3rd Generation Partnership Project
ABS	Akrylonitrilbutadienstyren– Acrylonitrile Butadiene Styrene
ASCII	Kódová tabulka, která definuje znaky anglické abecedy a jiné znaky používané v informatice – American Standard Code for Information Interchange
b/s	bity za sekundu
BPSK	Binární fázové Klíčování – Binary-Phase Shift Keying
BNC	Vysokofrekvenční konektor koaxiální kabely– Bayonet Neill Concelman
CO	Oxid uhelnatý – Carbon monoxide
dBm	Decibel-miliwatt
DBPSK	Diferenciální Binární fázové Klíčování – Differential Binary-Phase Shift Keying
DO	Rozpuštěný kyslík – Dissolved oxygen
DPS	Deska Plošných Spojů
EEPROM	Elektronicky Vymazatelná Paměť pouze pro čtení – Electrically Erasable Programmable Read-Only Memory
GFSK	Gausovo Klíčování Frekvenčním posunem – Gaussian Frequency-Shift Keying
GGSN	Rozhraní mezi GPRS sítí a externími paketovými sítěmi – Gateway GPRS Support Node
GIS	Geografický informační systém – Geographic information system
GND	Uzemnění – GrouND
GPS	Globální Polohový Systém – Global Positioning System
GPIO	Univerzální vstup/výstup – General-purpose input/output
GSM	Globální Systém pro Mobilní komunikaci – Global System for Mobile comunacations
I^2C	Multi-masterová počítačová sériová sběrnice – Inter-Integrated Circuit
IDE	Vývojové prostředí – Integrated Development Environment
IoT	Internet věcí – Internet of Things
IP	Stupeň krytí – Ingress Protection Protokol síťové vrstvy–Internet Protocol
IPv4	Jeden z protokolů síťové vrstvy – Internet Protocol version 4
Mb/s	Kilobity za sekundu
Li-ion	Lithium-iontová
LoRa	Jedna z LPWAN sítí – Long Range Wide Area Network
LPWAN	Nízkopříkonová rozsáhlá síť – Low-Power Wide Area Network

LTE	Technologie určená pro vysokorychlostní Internet v mobilních sítích – Long Term Evolution
Mb/s	Megabity za sekundu
MCU	Mikrokontrolér – Micro Controller Unit
MIMO	Více vstupů více výstupů – Multiple-Input Multiple-Output
mS/m	Milisiemens na metr
NB-IoT	Jedna z LPWAN sítí–NarrowBand Internet of Things
NFC	Modulární technologie radiové bezdrátové komunikace mezi elektronickými zařízeními na velmi krátkou vzdálenost – Near Field Communication
OFDM	Ortogonální Multiplex s Frekvenčním Dělením – Orthogonal Frequency Division Multiplexing
OOP	Objektově orientované programování – Object-oriented programming
PDP	Datová struktura obsahující informace o relaci účastníka – Packet Data Protocol
pH	Vodíkový exponent – potential of hydrogen
ppm	Dílů na milion – part per milion
PLMN	Veřejná pozemní mobilní síť – Public Land Mobile Network
QAM	Kvadrurní amplitudová modulace – Quadrature Amplitude Modulation
QPSK	Klíčování fázovým posuvem – Quadrature Phase-Shift Keying
SIM	Účastnická identifikační karta – Subscriber Identity Module
SMD	Součástky určené pro povrchovou montáž – Surface Mount Devices
UART	Univerzální asynchronní sériové rozhraní –Universal Asynchronous Receiver and Transmitter
UDP	Jeden z protokolů transportní vrstvy – User Datagram Protocol
UMTS	Třetí generace mobilních datových GSM sítí – Evolved Universal Terrestrial Radio Access Network
UNB	Ultra Nízkého pásma – Ultra Narrow Band
URC	Nevyžádaný kód výsledku – Unsolicited Result Code
USB	Univerzální sériová sběrnice – Universal Serial Bus
VCC	Pin s napájecím napětím – Voltage common collector
W-CDMA	Širokopásmový vícenásobný přístup s kódovým dělením – Wideband Code Division Multiple Access

SEZNAM PŘÍLOH

A Schéma zapojení	52
B Zdrojové kódy	54



Title		
Size	Number	Revision
A2		
Date:	27. 5. 2018	Sheet of
File:	C:\Users\NBBioT_waterQuality_v1.0.Sch	Drawn By:

B ZDROJOVÉ KÓDY

sara.h

```
1 #include "mbed.h"
2 #include "BufferedSerial.h"
3 #include "senzory.h"
4
5     class sara {
6     public:
7         sara(PinName tx, PinName rx);
8         int write(char * s);
9         int readPrint();
10        int zadaniPrikazu(char * s);
11        int registrace();
12        int poslaniZpravy(char * z);
13        int odpovedServer();
14        int poslaniZpravySenzory();
15        senzory * sy;
16
17    private:
18        BufferedSerial serial;
19        int delkaZpravy;
20        bool prijemZpravy;
21        int hexToInt(char c);
22        int hexToAscii(char c, char d);
23
24    };
```

sara.cpp

```
1 #include "sara.h"
2
3 sara::sara(PinName tx, PinName rx): serial(tx, rx) {
4     sy = new senzory(PA_0, PA_1, PB_9, PB_8);
5     prijemZpravy = false;
6 }
7
8 int sara::write(char * s) // umožňuje zadat AT příkaz,
9 // char *s – textový řetězec s AT příkazem
10 //Navratove hodnoty: 1 – někde vznikla chyba, 0 příkaz byl zadán.
11 {
12     if (!serial.writable()) {
13         return 1;
14     }
15 }
```

```

16     serial.puts(s);
17
18     printf("Posilam prikaz: %s\r\n\r\n", s);
19     serial.abort_write();
20     return 0;
21 }
22
23 int sara::readPrint() // Vypíše odpověď z modulu přes seriovou linku na
    PC
24 //Návratová hodnota 0 – přišla odpověď OK, 1–nastala chyba, nebo v zá
    dosbníku není žádná odpověď,2 zařízení je správně registrováno.
25 {
26     DigitalIn mybutton(PC_13);
27     int t = 0;
28     while (!(serial.readable())) {
29         wait_ms(10);
30         t++;
31         if (t == 1000) {
32             return 1;
33         }
34     }
35
36     char now = '\0';
37     char before = '\0';
38     int i = 0;
39     int timeout = 0;
40     char zprava[100];
41     int delka = 0;
42     int control = 0;
43     int uvozovka = 0;
44     char buf;
45     int nsomi = 0;
46     char buff_time[50];
47     while (true) {
48         i = 0;
49         while (zprava[i] != '\0') {
50             zprava[i] = '\0';
51             i++;
52         }
53         i = 0;
54         while (serial.readable()) {
55
56             before = now;
57             now = serial.getc();
58             if (now != '\0') {
59                 zprava[i] = now;
60                 i++;

```



```

61     }
62     if (before == '\r' && now == '\n') {
63         zprava[i] = '\0';
64         break;
65     }
66
67 }
68 delka = strlen(zprava);
69 printf("%s", zprava);
70
71 if (strcmp(zprava, "OK\r\n") == 0) {
72     return 0;
73 } else if (strcmp(zprava, "ERROR\r\n") == 0) {
74     return 1;
75 } else if (strcmp(zprava, "NO ANSWER\r\n") == 0) {
76     return 1;
77 } else if (strcmp(zprava, "+CEREG: 0,5\r\n") == 0) {
78     return 2;
79 } else if (zprava[0] == '+' && zprava[1] == 'N') {
80     for (i = 0; i < delka; i++) {
81
82         if (zprava[i] >= '0' && zprava[i] <= '9') {
83             nsomi = nsomi * 10;
84             nsomi = nsomi + zprava[i] - '0';
85         }
86     }
87     delkaZpravy = nsomi;
88     return 0;
89
90 } else if (prijemZpravy && zprava[0] != '\r') {
91
92     printf("\r\n");
93     for (i = 0; i < delka; i++) {
94         if (zprava[i] == '\\') {
95             uvozovka++;
96             if (uvozovka == 3) {
97                 i++;
98             }
99         }
100         if (uvozovka == 3) {
101
102             if (control % 2 != 0) {
103                 printf("%c", hexToAscii(buf, zprava[i]));
104             } else {
105                 buf = zprava[i];
106             }
107             control++;

```

```

108         }
109     }
110     }
111     }
112     printf("\r\n");
113 }
114
115     wait_ms(20);
116     timeout++;
117     if (timeout == 300) {
118         return 1;
119     }
120 }
121 }
122
123 int sara::zadaniPrikazu(char * s) // zada prikaz a vypíše odpověď na
    seriovou linku PC.
124 //Návratová hodnota 0 – zadání příkazu a vyčtení odpovědi proběhlo sprá
    vně,
125 // 1–nastala chyba při čtení nebo zápisu,2 zařízení je správně
    registrováno.
126 {
127     int state;
128     if (write(s)) {
129         return 1;
130     }
131     state = readPrint();
132     if (prijemZpravy) {
133         return state;
134     }
135     if (state == 2) {
136         return 2;
137     }
138     if (state == 1) {
139         return 1;
140     }
141     return 0;
142 }
143
144 int sara::registrace()//Provede registraci modulu do sítě pomocí série
    AT příkazů.
145 //Návratová hodnota 0 – všechny AT příkazy proběhly správně, 1– v ně
    kterém AT příkazu nastala chyba
146 {
147     if (zadaniPrikazu("AT+NRB\r\n")) {
148         return 1;
149     }

```

```

150     if (zadaniPrikazu("AT+CFUN=1\r\n")) {
151         return 1;
152     }
153     if (zadaniPrikazu("AT+COPS=1,2,\"23003\"\r\n")) {
154         return 1;
155     }
156     wait(1);
157     if (zadaniPrikazu("AT+COPS?\r\n")) {
158         return 1;
159     }
160     int c = 0;
161     while (zadaniPrikazu("AT+CEREG?\r\n") != 2) {
162         wait(1);
163         c++;
164         if (c == 60) {
165             return 1;
166         }
167     }
168     readPrint();
169     if (zadaniPrikazu("AT+CGDCONT=1,\"IP \",\"GPRST.INTERNET\"\r\n")) {
170         return 1;
171     }
172     if (zadaniPrikazu("AT+CGDCONT?\r\n")) {
173         return 1;
174     }
175     if (zadaniPrikazu("AT+NSOCR=\"DGRAM\",17,50105,1\r\n")) {
176         return 1;
177     }
178
179     return 0;
180
181 }
182
183 int sara::poslaniZpravy(char * z)//Odešle zadanou zprávu na vzdálený
    server.
184 //char *z -textový řetězec obsahující zadanou zprávu, která je poslána
    na vzdálený server.
185 //Návratová hodnota 0 - zpráva byla odeslána, 1 - při odesílání se
    vyskytla chyba
186 {
187     int delka = 0;
188     delka = strlen(z);
189     delkaZpravy = delka;
190     char delkaChar[5];
191     sprintf(delkaChar, "%d", delka);
192     for (int i = 0; i < 4; i++) {
193         if (!(delkaChar[i] > 47 && delkaChar[i] < 58)) {

```

```

194         delkaChar[i] = '\0';
195         break;
196     }
197 }
198 char prikaz[150] = "AT+NSOSTF=0,\"147.229.146.40\",50105,0x200,";
199 strcat(prikaz, delkaChar);
200 strcat(prikaz, ",\");
201
202 char outword[delka * 2];
203 int i;
204 for (i = 0; i < delka; i++) {
205     sprintf(outword + i * 2, "%02X", z[i]);
206 }
207
208 strcat(prikaz, outword);
209 strcat(prikaz, "\\");
210 strcat(prikaz, "\\r\\n");
211
212 if (zadaniPrikazu(prikaz)) {
213     return 1;
214 }
215 return 0;
216
217 }
218
219 int sara::odpovedServer()//Vypíše přijatou zprávu na sériovou linku PC
220 // Návratová hodnota 0 – zpráva byla odeslána, 1 – při vyčítání odpově
221 // di se vyskytla chyba
222 {
223     příjemZpravy = true;
224     char prikaz[16] = "AT+NSORF=0,";
225     char delkaChar[4];
226     sprintf(delkaChar, "%d", delkaZpravy);
227     strcat(prikaz, delkaChar);
228     strcat(prikaz, "\\r\\n");
229     if (zadaniPrikazu(prikaz)) {
230         příjemZpravy = false;
231         return 1;
232     }
233     příjemZpravy = false;
234     return 0;
235 }
236
237 int sara::poslaniZpravySenzory()//Provede měření hodnot připojených
238 // senzorů a následně je odešle na vzdálený sever.
239 //Návratová hodnota 0 – zpráva byla odeslána, 1 při desílání zprávy se
240 // vyskytla chyba

```

```

238 {
239 char zpravaSenzoru[30];
240 for (int i = 0; i <= 25; i++) {
241     zpravaSenzoru[i] = '\0';
242 }
243 char hodnotaSenzoru[5];
244 for (int i = 0; i <= 25; i++) {
245     hodnotaSenzoru[i] = '\0';
246 }
247 float hodnota = sy -> readPh();
248
249 sprintf(hodnotaSenzoru, "%.2f", hodnota);
250 strcat(zpravaSenzoru, hodnotaSenzoru);
251
252 strcat(zpravaSenzoru, ",");
253
254 hodnota = sy -> mereniCO();
255 for (int i = 0; i <= 25; i++) {
256     hodnotaSenzoru[i] = '\0';
257 }
258
259 sprintf(hodnotaSenzoru, "%.0f", hodnota);
260 strcat(zpravaSenzoru, hodnotaSenzoru);
261 strcat(zpravaSenzoru, ",");
262 sy -> htdu21dInit();
263 hodnota = sy -> htdu21dTemp();
264
265 for (int i = 0; i <= 25; i++) {
266     hodnotaSenzoru[i] = '\0';
267 }
268
269 sprintf(hodnotaSenzoru, "%.2f", hodnota);
270 int i = 0;
271 strcat(zpravaSenzoru, hodnotaSenzoru);
272
273 strcat(zpravaSenzoru, ",");
274 while (true) {
275     printf("%c", zpravaSenzoru[i]);
276     i++;
277     if (zpravaSenzoru[i] == '\0') {
278         break;
279     }
280 }
281 printf("\r\n");
282 for (int i = 0; i <= 25; i++) {
283     hodnotaSenzoru[i] = '\0';
284 }

```

```

285     hodnota = sy -> htdu21dHum();
286     sprintf(hodnotaSenzoru, "%.2f", hodnota);
287     strcat(zpravaSenzoru, hodnotaSenzoru);
288
289     char zpravaPrikaz[60] = "1498478203392;";
290     strcat(zpravaPrikaz, zpravaSenzoru);
291     strcat(zpravaPrikaz, ";5");
292
293     if (poslaniZpravy(zpravaPrikaz)) {
294         return 1;
295     }
296
297     return 0;
298 }
299 }
300
301 int sara::hexToInt(char c)//Převede hexadecimální znak na celočíselnou
    hodnotu
302 //char c – znak který má být převeden na celočíselnou hodnotu int
303 //znak vyjádřený v desítkové soustavě
304 {
305     int first = c / 16 - 3;
306     int second = c % 16;
307     int result = first * 10 + second;
308     if (result > 9) result--;
309     return result;
310 }
311
312 int sara::hexToAscii(char c, char d) //Převede dvojici hexadecimálních
    znaků na ascii znak.
313 //char c–první znak z dvojice, char d – druhý znak z dvojice
314 //Návratová hodnota – ascii znak vyjádřený v desítkové soustavě
315 {
316     int high = hexToInt(c) * 16;
317     int low = hexToInt(d);
318     return high + low;
319 }

```

senzory.h

```

1 #ifndef senzory_H
2 #define senzory_H
3 #include "mbed.h"
4 #include "htu21d.h"
5
6 class senzory {

```

```

7     public:
8     senzory(PinName ph, PinName co, PinName htu21dSDA, PinName
htu21dSCL);
9     float readPh();
10    float kalibraceCO(); //bude spusteno jen 1
11    int mereniCO();
12    float htdu21dTemp();
13    float htdu21dHum();
14    int htdu21dInit();
15
16    private:
17    AnalogIn phpin;
18    AnalogIn copin;
19    htu21d htu;
20 };
21 #endif

```

senzory.cpp

```

1 #include "senzory.h"
2
3 senzory::senzory(PinName ph, PinName co, PinName htu21dSDA, PinName
htu21dSCL): phpin(ph), copin(co), htu(htu21dSDA, htu21dSCL) {
4
5     printf("Senzory pripraveny\r\n");
6 }
7
8 float senzory::readPh() //Změří hodnotu Ph
9 //Návratová hodnota – naměřená hodnota Ph, -1000 značí, že byla naměř
ena nereálná hodnota
10 {
11     float hodnota = 0;
12     hodnota = phpin.read();
13     hodnota = hodnota - 0.405;
14     printf("Hodnota pinu %f\r\n", hodnota);
15     float voltage = 5 * hodnota;
16     printf("\tVoltage: %f\r\n", voltage);
17     float Po = 7 + ((2.5 - voltage) / 0.18);
18
19     printf("\tPH: %f\n\r\n", Po);
20     if (Po >= 0.0 && Po <= 14.0) {
21         return Po;
22     } else return -1000;
23 }
24
25 float senzory::kalibraceCO() //Změří hodnotu konstanty R0

```

```

26 //Návratová hodnota – změřená koncentrace R0
27 {
28     float sensor_volt;
29     float RS_air;
30     float R0;
31     float sensorValue = 0;
32
33     for (int x = 0; x < 100; x++) {
34         sensorValue = sensorValue + copin.read();
35         wait_ms(5);
36     }
37     sensorValue = sensorValue / 100.0;
38     sensor_volt = sensorValue * 5.0;
39     RS_air = (5.0 - sensor_volt) / sensor_volt;
40     R0 = RS_air / 9.9;
41     printf("Vstupni napeti: %.2f V\n", sensor_volt);
42     printf("Konstanta R0: %.2f\n", R0);
43     return R0;
44 }
45
46 int senzory::mereniCO() //Změří hodnotu koncentrace CO vztaženou k R0, v
    jednotkách ppm.
47 //Návratová hodnota – změřená koncentrace CO b jednotkách ppm.
48 {
49     AnalogIn copin(PA_0);
50     float sensor_volt;
51     float RS_gas;
52     float ratio;
53     float sensorValue;
54     float R0 = 1.55; // mimo
55
56     sensorValue = copin.read();
57
58     sensor_volt = sensorValue * 5.0;
59
60     RS_gas = (5.0 - sensor_volt) / sensor_volt;
61
62     ratio = RS_gas / R0; // přesne zjistit výpočty..
63
64     ratio = ratio * 100;
65
66     printf("Priblizna koncentrace CO: %d ppm", (int) ratio);
67     return (int) ratio;
68 }
69
70 int senzory::htdu21dInit() // Inicializace senzoru teploty a vlhkosti
    htdu21d

```



```

71 //Návratová hodnota – 0 pokud zařízení pracuje správně, 1 pokud je ně
    kde chyba
72 {
73     printf("\r\n\r\nK64F_HTU21D basic operation\r\n");
74
75     int htu21 = htu.softReset();
76     if (htu21 == 0) {
77         printf(" – HTU21D broken... \r\n");
78         return 1;
79     } else {
80         uint8_t HTU21DuserReg = htu.getUserReg();
81         printf("HTU21D UserReg: 0x%02x   SN: 0x%04x %08x %04x\r\n",
82             HTU21DuserReg, htu.HTU21sn.HTU21D_sna, htu.HTU21sn.
HTU21D_snb, htu.HTU21sn.HTU21D_snc);
83         return 0;
84     }
85 }
86
87 float senzory::htdu21dTemp() //Změří hodnotu teploty vzduchu
88 //Návratová hodnota – teplota vzduchu, –1000 pokud se teplota nepodaří
    vyčíst
89 {
90     float H21Temp = htu.getTemp();
91     wait_ms(20);
92     if ((double) H21Temp == 255.0) {
93         printf("\r\n*** HTU21D Temp error!!\r\n");
94         return -1000;
95     } else {
96         printf("TEMP: %7.2f\r\n", H21Temp);
97         return H21Temp;
98     }
99 }
100
101 float senzory::htdu21dHum() // Změří hodnotu vlhkosti vzduchu.
102 //Návratová hodnota – teplota vzduchu, –1000 pokud se vlhkost nepodaří
    vyčíst
103 {
104     float H21Hum = 0.0;
105     H21Hum = htu.getHum();
106     if ((double) H21Hum == 255.0) {
107         printf("\r\n*** HTU21D Hum error!!\r\n");
108         return -1000;
109     } else {
110         printf("HUM: %4.1f %%\r\n", H21Hum);
111         return H21Hum;
112     }
113 }

```

main.cpp

```
1 #include "mbed.h"
2 #include "htu21d.h"
3 #include "sara.h"
4 #include "senzory.h"
5
6 DigitalOut leddust(PB_5);
7
8 int main() {
9     sara sr(PA_9, PA_10);
10    sr.registrace();
11    while (true) {
12
13        sr.poslaniZpravySenzory();
14        wait(3);
15        sr.readPrint();
16        sr.odpovedServer();
17        wait(60);
18    }
19 }
```