Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

Monitoring hlubokomrazících boxů pomocí zařízení IoT

Jaroslav Valdauf

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Jaroslav Valdauf

Systémové inženýrství a informatika
Informatika

Název práce

Monitoring hlubokomrazících boxů pomocí zařízení IoT

Název anglicky

Ultra low temperature freezer monitoring by IoT device

Cíle práce

Cílem diplomové práce je tvorba systému varování o selhání hlubokomrazících boxů s využitím spínaných kontaktů (tzv. alarm kontaktů) na těchto zařízeních, zpracování a přenos informací pomocí IoT zařízení do informačního systému a jejich další zpracování – zaznamenání vzniku a ukončení události, notifikace osob, dohled na monitorovací zařízení. Součástí práce bude komparace technologií pro komunikaci IoT čidel a jedna z těchto možností bude vybrána pro finální realizaci.

Metodika

Metodika řešení diplomové práce se bude zakládat na studiu zdrojů týkajících se IoT zařízení, jejich možností, typů sítí a přenosových protokolů. V teoretické části bude dále provedena analýza současných dostupných řešení. Praktická část bude zaměřena na konstrukci samotného monitorovacího zařízení, jeho softwaru, přenos dat a zpracování těchto informací na straně informačního systému.

Doporučený rozsah práce

60 – 80 stran

Klíčová slova

IoT, internet věcí, internet of things, ZigBee, LoRaWAN, Sigfox, BLE, monitoring, alarm

---

Doporučené zdroje informací

BUYYA, Rajkumar. Internet of Things: Principles and Paradigms. Elsevier, 2016. ISBN 978-0-12-805395-9.

Hradla, volty, jednočipy : úvod do bastlení. MALÝ, M..

Internet of Things. [elektronický zdroj] /. GACOVSKI, Z..

SALAM, A. *Internet of things for sustainable community development : wireless communications, sensing, and systems.* Cham: Springer, 2020. ISBN 978-3030352905.

SHEN, Xuemin (Sherman), Xiaodong LIN a Kuan ZHANG, ed., 2020. Encyclopedia of Wireless Networks [online]. Cham: Springer International Publishing [cit. 2021-8-9]. ISBN 978-3-319-78261-4

Předběžný termín obhajoby

2021/22 LS – PEF

Vedoucí práce

Ing. Václav Lohr, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 25. 8. 2021

doc. Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 19. 10. 2021

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 04. 11. 2021

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Monitoring hlubokomrazících boxů pomocí zařízení IoT" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 21.3.2022 _____

Poděkování

Rád bych touto cestou poděkoval Ing. Václavu Lohrovi, Ph.D. za odborné vedení, dobré rady a podporu při vzniku práce, MUDr. Tomáši Kostrhunovi za podporu a možnost testovacího nasazení v rámci infrastruktury 3. lékařské fakulty UK a Ing. Vojtěchu Kadlecovi za odbornou pomoc při realizaci.

Monitoring hlubokomrazících boxů pomocí zařízení IoT

Abstrakt

Internet of Things (IoT) neboli Internet věcí je rychle se rozvíjející odvětví výpočetní techniky s přesahem do reálného světa. Tyto moderní technologie napomáhají zavedení automatizace či monitoringu různých odvětví lidské činnosti. 3. lékařská fakulta Univerzity Karlovy provozuje ve svých budovách několik desítek hlubokomrazících boxů a dalších výzkumných zařízení, které je potřeba monitorovat a v případě poruchy notifikovat osoby zodpovědné za jejich provoz. Při návrhu nového dohledového systému bylo rozhodnuto využít technologie Internetu věcí.

Cílem práce je návrh a realizace kompletního systému dohledu nad zmíněnými přístroji za využití tzv. alarm kontaktů. Bude vytvořeno dohledové zařízení, které bude bezdrátově komunikovat se síťovým serverem. Takto získaná data budou dále předána a zpracována Zabbix serverem, který bude zajišťovat upozornění definovaných uživatelů.

V teoretické části práce popisuje technologické řešení připojení IoT zařízení do sítě, výhody a nevýhody jednotlivých technologií, možnosti jejich nasazení v reálných situacích. Dále je podrobně rozebrán protokol LoRaWAN, který je využíván pro komunikaci v rámci praktického užití. V části věnující se hardwaru je popsáno zařízení Arduino MKR WAN 1310 a gateway Mikrotik wAP LoRa8.

Praktická část se již věnuje implementaci konkrétního řešení dle analýzy požadavků. Je vytvořen prototyp zařízení a naprogramován síťový aplikační server, který zajišťuje vzájemnou komunikaci za využití informací z teoretické části. Pro zařízení je navrhnuta krabička a plošný spoj. Následně je po jednotlivých částech otestována hardwarová i softwarová část, která je poté nasazena do reálného prostředí pro dohled nad hlubokomrazícími boxy.

Klíčová slova: IoT, internet věcí, ZigBee, LoRaWAN, Sigfox, BLE, monitoring, alarm

Ultra low temperature freezer monitoring by IoT device

Abstract

The Internet of Things (IoT) or Internet of Things is a rapidly growing field of computing technology that is spilling over into the real world. The 3rd Faculty of Medicine of Charles University operates several dozen deep-freezers and other research equipment in its buildings, which need to be monitored and, in the event of a malfunction, those responsible for their operation notified. When designing the new monitoring system, it was decided to use Internet of Things technology.

The aim of the thesis is to design a complete system of monitoring of the mentioned devices using so-called alarm contacts. A monitoring device will be created that will communicate wirelessly with a network server. The data thus obtained will be further transmitted and processed by the Zabbix server, which will provide notification of users.
The theoretical part of the thesis describes the technological solutions for connecting IoT devices to the network, advantages and disadvantages of each technology, and the possibilities of their deployment in real situations. Furthermore, the LoRaWAN protocol, which is used for communication in practical applications, is discussed in detail. The hardware section describes the Arduino MKR WAN 1310 device and the Mikrotik wAP LoRa8 gateway.

The practical part is already devoted to the implementation of a specific solution according to the requirements analysis. A prototype of the device is created, a network application server is programmed, which provides mutual communication using the information from the theoretical part. A box and a circuit board are designed for the device. Subsequently, the hardware and software part is tested part by part, which is then deployed in a real environment to monitor the deep freezing boxes.

Keywords: IoT, Internet of Things, ZigBee, LoRaWAN, Sigfox, BLE, monitoring, alarm

Obsah

1 Úvod

Často zmiňovanou frází poslední let se stává internet věcí, Internet of Things. Jedná se o samostatná chytrá zařízení, která komunikují mezi sebou nebo přenášejí data do cloudových řešení či mobilních telefonů. Používání chytrých zařízení se stále více rozšiřuje i v domácnostech, ať už jde o chytré zářivky, domácí spotřebiče připojené k internetu nebo zabezpečovací zařízení. V průběhu posledních let došlo k rozšíření dostupných technologií a jejich optimalizaci pro různá užití v praxi. Nedílnou součástí těchto změň je také snížení energetické náročnosti provozu i samotné komunikace a zvětšování vzdáleností, na které jsou zařízení schopna přenášet informace.

Hlavní motivací této práce je vytvoření nového systému dohledu nad hlubokomrazícími boxy, který by nahradil současné, rychle zastarávající řešení. Během několika let provozu bylo posbíráno mnoho poznatků a praktických zkušeností co změnit a zlepšit v novém dohledovém systému. 3. lékařská fakulta Univerzity Karlovy provozuje několik desítek těchto mrazících zařízení a další specifické výzkumné přístroje jako jsou inkubátory, líhně nebo dewarovy nádoby. Nabízená hotová řešení jsou finančně náročná a často nesplňují požadavky dané uživateli systému či jejich správci.

V současné době je k dispozici množství literatury, open source nástrojů a zařízení pro konstrukci vlastního zařízení „na míru“ přesně podle potřeb provozovatele. V rámci diplomové práce dojde k návrhu, realizaci a nasazení zařízení a softwarového řešení zajišťující komunikaci a zpracování dat. Po otestování v reálných podmínkách budou získány další zkušenosti pro zapracování případný změn, proto bude celý systém monitoringu navrhován tak, aby byl do budoucna snadno upravitelný.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem diplomové práce je tvorba systému varování o selhání hlubokomrazících boxů s využitím spínaných kontaktů (tzv. alarm kontaktů) na těchto zařízeních, zpracování a přenos informací pomocí IoT zařízení do informačního systému a jejich další zpracování – zaznamenání vzniku a ukončení události, notifikace osob, dohled na monitorovací zařízení. Součástí práce bude komparace technologií pro komunikaci IoT čidel a jedna z těchto možností bude vybrána pro finální realizaci.

2.2 Metodika

Metodika řešení diplomové práce se bude zakládat na studiu zdrojů týkajících se IoT zařízení, jejich možností, typů sítí přenosových protokolů. V teoretické části bude dále provedena analýza současných dostupných řešení. Praktická část bude zaměřena na konstrukci samotného monitorovacího zařízení, jeho softwaru, přenos dat zpracování těchto informací na straně informačního systému.

3 Teoretická východiska

3.1 Internet of Things – IoT

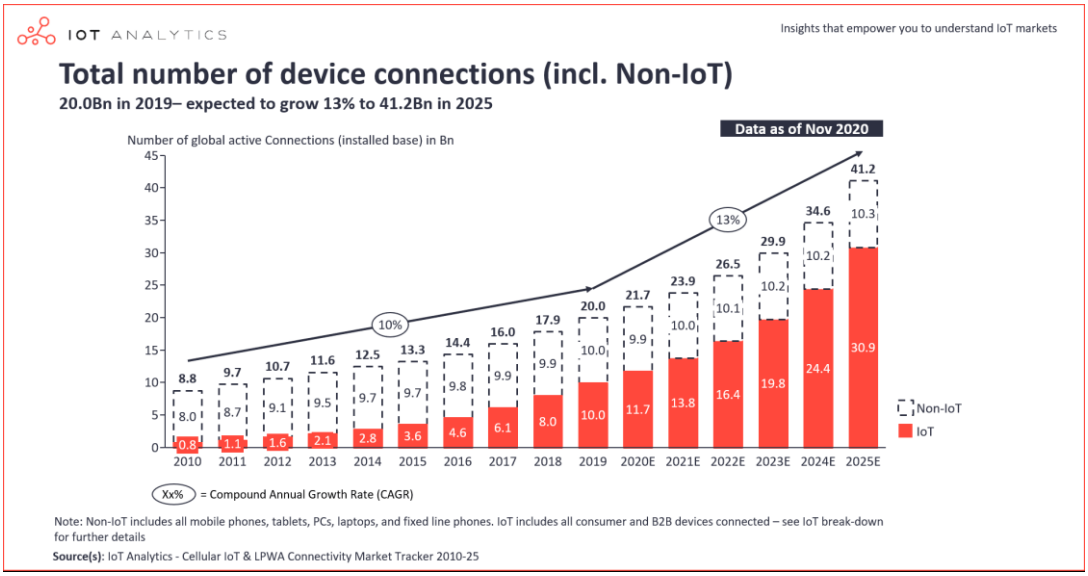
Internet of Things (česky "Internet věcí") je pojem, který byl poprvé použit v roce 1999 během prezentace Kevinem Ashtonem, který v té době pracoval jako výkonný ředitel Auto-ID centra na MIT. Během této prezentace, která nesla název "Internet of Things", autor zmiňuje, že jsou veškerá data na internetu vytvářena lidmi a představuje možnost připojení senzorů k síti a sdílení jejich dat mezi systémy. (Buyya, 2016)

Internet věcí lze souhrnně popsat jako síť propojených věcí, každá jednoznačně identifikovatelná, které umožňují výměnu dat za použití standardizovaných komunikačních protokolů a analýzou těchto dat bude možné docílit vyšší přidanou hodnotu. V názvu je sice použito slovo Internet, ale může se jednat i o lokální síť a "věci" mohou komunikovat v rámci ní mezi sebou. Pojem "věc" v tomto kontextu znamená fyzické zařízení nebo software, který monitoruje určité vlastnosti (např. teplota, vlhkost, stisk klávesy, …) a sdílí data v rámci sítě. (INFSO D.4 NETWORKED ENTERPRISE & RFID INFSO G.2 MICRO & NANOSYSTEMS, 2008)

Princip IoT je ve spojování těchto "věcí", výměně informací a na to navazujících akcí. Jako příklad lze uvést například koncept smart cities, kdy lze na úrovni města za použití různých čidel a senzorů zefektivnit jeho fungování. Pro občany to mohou být chytré elektroměry či vodoměry a tím odpadající odečty energií, vše vidí poskytoval služby v reálném čase. Návštěvníci města mohou využít automatický informační systém o dostupnosti parkování či chytrých semaforů na základě aktuální dopravní situace. Je možné kontrolovat i kvalitu ovzduší a na základě těchto informací lze pomocí proměnného dopravního značení redukovat počet automobilů v exponovaných místech. V ČR je do projektu ministerstva pro místní rozvoj zapojeno několik měst, od roku 2016 jsou to například Pardubice, ale i mnohem menší města jako Jihlava či Hrušovany pod Jevišovkou s necelými 3,5 tisíci obyvateli. (Salam, 2020)

IoT je jedním z nejrychleji rostoucích odvětví výpočetní techniky v posledních letech. Společnost Cisco ve svém white paperu vydaném v roce 2011 uvádí, že na přelomu let 2008 a 2009 došlo k milníku, kdy bylo k internetu připojeno více zařízení, než je počet obyvatel Země. V roce 2010 bylo k internetu připojeno 12,5 miliardy zařízení a pro rok 2020 bylo předpovězeno 50 miliard připojených zařízení. IoT analytics ve statistice z prosince roku 2020 tento předpoklad nepotvrzuje, počet zařízení na konci roku 2020 je necelých

22 miliard. V této statistice zároveň informuje o dalším důležitém milníku – v roce 2019 bylo k internetu připojeno 20 miliard zařízení a 50 % z nich jsou IoT zařízení. Dle předpovědí se bude tento poměr zařízení zvyšovat ve prospěch IoT zařízení. (Evans, 2011) (Lasse Lueth, 2020) (Salam, 2020)



Obrázek 1 - Počet připojení nových zařízení (Zdroj: https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/)

3.1.1 Architektura IoT sítí

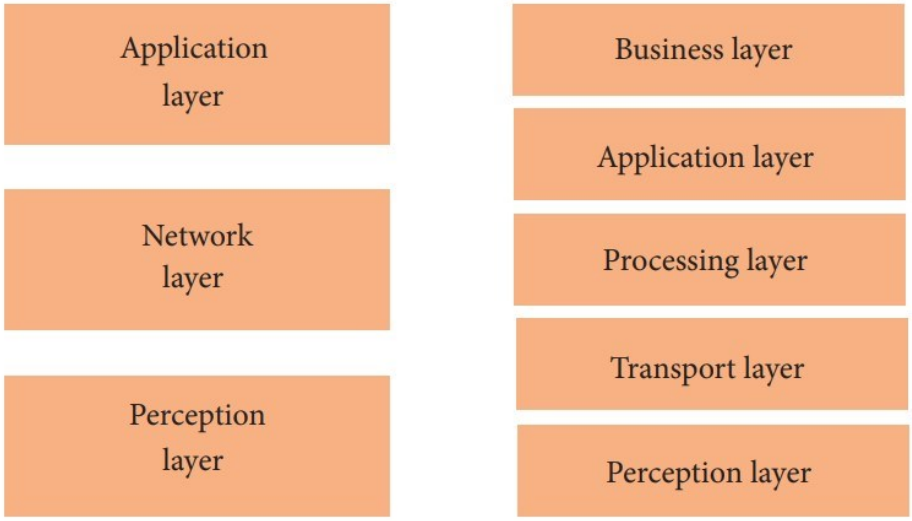
Architektura IoT sítí není sjednocena a v současnosti neexistuje žádná norma, která by předepisovala složení a fungování jejich vrstev. Literaturou nejčastěji popisovanými architekturami jsou třívrstvá, čtyřvrstvá, pětivrstvá, cloudová a architektura typu Fog. (Sethi, 2017)

Obecná architektura IoT sítí se zakládá na třech vrstvách – snímací, transportní a aplikační. Snímací vrstva, někdy též nazývaná senzorická, slouží ke shromažďování informací, typicky ze senzorů a jiných zařízení, které byly zmíněny v předchozí části. Vrstva transportní zajišťuje přenos sebraných informací ze snímací vrstvy k vrstvě aplikační. Přenos může probíhat přes kabel, častěji spíše bezdrátově. O bezdrátovém přenosu bude pojednávat následující kapitola. Některé zdroje uvádějí před aplikační vrstvou ještě jeden level zpracování, a to vrstvu zpracování dat. Tato vrstva přebírá informace, analyzuje je, připravuje pro použití v aplikacích, případně přijímá rozhodnutí na základě zadaných podmínek. Toto zpracování může být součástí již samotného senzoru nebo vrstvy aplikační,

která slouží pro prezentaci dat a jiných výstupů pro uživatele. (Yaqoob, 2017), (Sikder, 2018), (Sethi, 2017)

V případě pětivrstvé architektury zůstává zachována snímací a aplikační vrstva, přibyde vrstva obchodní (business layer), která řídí celý systém. Princip fungování procesní vrstvy (Processing layer) je stejný, jako v případě čtyřvrstvé architektury. Poslední vrstvou je vrstva transportní, která zajišťuje obousměrný přenos mezi procesní a snímací vrstvou. Rozdíly mezi třívrstvou a pětivrstvou architekturou jsou viditelné na obrázku 2. (Yaqoob, 2017), (Sikder, 2018), (Sethi, 2017)

Každá z vrstev představuje samostatné bezpečnostní riziko. V případě senzorů, kde se jedná o fyzická zařízení, může případný útočník ovlivnit jejich funkčnost či je úplně zničit, pokud je schopný se k zařízení fyzicky dostat. Vrstva transportní je náchylná na odposlechnutí přenášených informací, přesměrování toku dat či zahlcení síťových prvků pro přenos informací pomocí DoS útoků. Aplikační vrstva, stejně jako každý software, je náchylná na chyby v kódu samotné aplikace. (Singh, 2020) (Gacovski, 2019)



Obrázek 2- Tří a pětivrstvá architektura (Zdroj: https://www.hindawi.com/journals/jece/2017/9324035/)

3.1.2 Bezdrátové přenosové technologie

Chytrá zařízení spolu komunikují v rámci sítě. Existuje mnoho způsobů, jak tato zařízení k síti připojit, praxe dokládá, že nejspolehlivější možností je použití kabelové sítě. Jako příklad produktu připojeného kabelem lze uvést například modely z řady Poseidon české společnosti HW Group – přístroj s množstvím digitálních a analogových vstupů připojený pomocí ethernetového kabelu do počítačové sítě. Ovšem ne všechna zařízení lze takto připojit, ať už se jedná o chytré automobily nebo senzory monitorující pohyb nestabilních skalních útvarů. U každé realizace IoT je nutné předem stanovit možnosti připojení s ohledem na vzdálenost přenosu mezi samotným monitorovacím prvkem a přijímačem/vysílačem, velikost a četnost přenášených informací a v neposlední řadě i energetickou náročnost takového zařízení a přenosu. V případě monitoringu v odlehlých oblastech se špatným přístupem k elektřině a nesnadnou možností fyzické kontroly bude nutné volit jinou technologii připojení než pro již zmíněný chytrý automobil, který elektřinu generuje alternátorem, případně ji používá jako zdroj pohonu. V následující části je rozebráno několik možností bezdrátového připojení a jejich celkové srovnání v parametrech, které jsou důležité pro zvolení správné technologie při samotném návrhu chytrého řešení. (Gacovski, 2019)

3.1.2.1 Bluetooth

Bluetooth je celosvětový standard pro bezdrátovou technologii přenosu, které umožňuje zařízením vzájemně komunikovat prostřednictvím rádiových vln. Vyznačuje se krátkým dosahem, malou spotřebou energie, nízkými náklady a drobnými rozměry modulů. Vytvořena byla v roce 1994 firmou Ericsson jako technologie nahrazující kabelové sériové rozhraní. Bluetooth je definován standardem IEEE 802.15.1 a spadá do kategorie PAN sítí (Personal Area Network). Bluetooth definuje výkonové třídy, pro ně maximální povolený vyzařovací výkon a tomu odpovídá teoretický dosah těchto zařízení. Tyto uváděné vzdálenosti jsou pro přenos ve volném prostoru. V případě, že mezi zařízeními leží nějaké překážky, se dosah snižuje a dochází ke ztrátě paketů. (Pužmanová, 2004)

Pracovní frekvence Bluetooth je 2,4 GHz, na stejné frekvenci funguje i Wi-Fi. Rozdíl mezi těmito technologiemi je ve vysílacím výkonu i v práci s šířkou pásma a kanály. Bluetooth využívá 79 kanálů, každý se šířkou pásma 1 MHz. Kvůli možnému rušení se využívá metoda rozprostřeného spektra s přeskakováním kmitočtů – během jedné sekundy přeskočí na jinou frekvenci až 1600x. Přeskok nastává po každém přenosu paketu, před

dalším odesláním je stanovena doba 220 μs, během které se čeká, aby nedocházelo ke kolizím. (Bluetooth® Wireless Technology)

Class (třída)	Maximální povolený výkon	Teoretický dosah
Class 1	100 mW (+20 dBm)	~ 100 metrů
Class 2	2,5 mW (+4 dBm)	~ 10 metrů
Class 3	1 mW (0 dBm)	~ 1 metr

Tabulka 1- Výkonové třídy Bluetooth

Od prvotní specifikace v roce 1994 se Bluetooth stále vyvíjí, během doby došlo k vyčlenění specifikace Bluetooth Low Energy pro zařízení, která nepotřebují odesílat velké objemy dat, ale je u nich důležitá nízká spotřeba energie. Bluetooth ve verzi 3.0 přišlo s novinkou, umožňující teoretickou rychlost přenosu až 24 Mbit/s. Toho je dosaženo využitím Bluetooth pouze ke spárování zařízení a samotný přenos probíhá za používá WiFi. Ve verzi 4.0, označované jako Bluetooth Low Energy (BLE), dochází k velké změně přístupu, a to ke snížení energetické náročnosti na úkor množství přenášených dat, tato specifikace nenahrazuje předchozí, ale existuje s ní souběžně. Snížení množství přenášených dat je v důsledku zkrácení času na jednotky milisekund, rychlost stále může dosáhnout až 1 Mbit/s. I přes společný základ názvu není Bluetooth Classic kompatibilní s Bluetooth Low Energy. S další, aktuálně poslední, verzí Bluetooth 5 došlo při zachování nízké spotřeby energie ke zvýšení dosahu až na čtyřnásobek oproti předchozí specifikaci. Zároveň přichází i s topologií mesh, kdy koncová zařízení mohou komunikovat mezi sebou a nejen s hlavní jednotkou. Díky tomu nemusí koncová zařízení být v dosahu centrálního uzlu a je možné používat kromě hvězdicové topologie sítě i jiné typy a pokrýt tak větší prostory. (Gupta, 2013)

S příchodem specifikace Bluetooth Low Energy došlo ke změně ve využití pásem a frekvencí, používá se pouze 40 kanálů, každý se šířkou pásma 2 MHz. Poslední 3 kanály jsou využívány pro tzv. advertising, který slouží k přenosu informací o zařízení pro ostatní prvky v dosahu signálu.

Dle výše popsaného je pro IoT nasazení vhodné využít BLE zařízení. Tato zařízení přenáší minimální množství dat, navíc na krátkou vzdálenost. Hodí se proto na využití v chytré domácnosti. Jako příklad lze uvést čínskou společnost Xiaomi a jejich zařízení z řady Mijia, které využívají tuto technologii pro sdílení dat s chytrým telefonem či

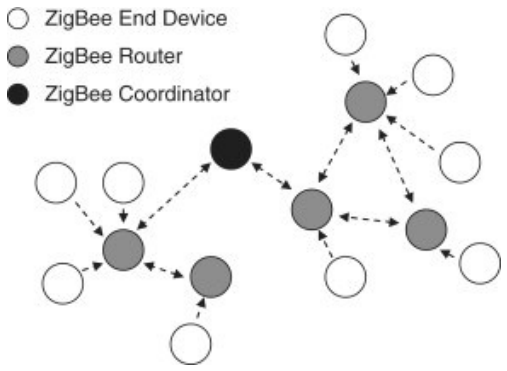
s centrální jednotkou. Jedná se například o výrobek MiFlora, který monitoruje stav půdy v květináči, různé druhy teploměrů a vlhkoměrů či měřič kvality ovzduší.

3.1.2.2 ZigBee

Jeden z nejvíce rozšířených typů přenosu dat v chytrých domácnostech je bezdrátová komunikační technologie ZigBee. Ratifikace první specifikace proběhla již v roce 2004 a je definována standardem IEEE 802.15.4, spadá tedy do stejné skupiny Wireless PAN jako Bluetooth. ZigBee není určeno k přenosu velkého množství dat, bylo navrhováno jako nízkovýkonný a energeticky nenáročný standard pro odesílání informací na vzdálenosti desítek až stovek metrů v závislosti na prostředí. Tímto se podobá Bluetooth Low Energy, ale byla standardizována o více než dekádu dříve. Díky jednoduchosti implementace oproti BLE je možné používat pro připojení i 8-bitové mikrokontrolery. Další výhodou je rychlé obnovení připojení po ztrátě komunikace v řádech milisekund. (ZigBee Alliance, Inc., 2015)

Komunikace probíhá v bezlicenčním pásmu 868 MHz, 902–928 MHz a 2,4GHz, ve kterém je pro přenos použito 16 kanálů, každý se šířkou pásma 5 MHz. Odolnosti vůči rušením je zde docíleno metodou přímo rozprostřeného spektra – každý bit se nahrazuje bitovou sekvencí, díky této redundanci je možné v případě ztráty části sekvence obnovit původní hodnotu. (Dubrawsky, 2010)

Zigbee využívá topologii sítě typu mesh v závislosti na zvolených zařízeních. Koordinační jednotka sítě řídí topologii a je bránou mezi Zigbee sítí a lokální sítí. Router slouží pro přeposílání zpráv mezi jednotlivými uzly sítě a připojení koncových zařízení, která nejsou v dosahu koordinační jednotky. Koncová zařízení slouží k odesílání dat, jako jediná by měla fungovat na baterie, protože pokud neodesílají data, spotřebovávají minimum energie.



Obrázek 3-Topologie mesh (Zdroj: https://www.sciencedirect.com/topics/computer-science/zigbee-coordinator)

Rozvoj technologie ZigBee je zajišťován Zigbee Alliance, která od roku 2002 sdružuje společnosti, které společně vyvíjejí tento standard. Mezi téměř 350 firem, které jsou v alianci přihlášeny, patří například Philips, Texas Instruments, Honeywell. Počátkem roku 2021 došlo ke změně značky na Connectivity Standards Alliance a zaštítění i dalších komunikačních protokolů. (Connectivity Standards Alliance)

Zigbee sítě jsou hojně využívány v chytrých domácnostech, ale i v průmyslových automatizacích, důvodem je snadná implementace a také nízká cena. Nejznámějším zařízením fungujícím na technologii Zigbee je systém chytrého osvětlení Philips Hue nebo obdobná zařízení nabízená společností Ikea pod značkou TRÅDFRI.

3.1.2.3 Z-Wave

Obdobou protokolu ZigBee je Z-Wave, který je od počátku designovaný pro použití v chytrých domácnostech. Za návrhem stojí soukromá společnost Zensys, která tento komunikační protokol představila v roce 2001. Samotné vysílací zařízení je vyráběno společností Sigma Designs, která původní organizaci koupila. Na rozdíl od ZigBee, kdy jsou vysílače vyráběny více dodavateli, u Z-Wave byly čipy produkovány výhradně jedním výrobcem. Ke změně nedošlo ani po akvizici společností Silicon Labs, která po převzetí dále zajišťuje jejich výrobu. (Silicon Labs Completes Acquisition of Sigma Designs' Z-Wave Business, 2018)

Z-Wave používá pro komunikaci v Evropě frekvenci 868,42 MHz, ve Spojených státech amerických pak 908,42 MHz. Použitím těchto kmitočtů se snižuje možnost rušení Wi-Fi sítěmi, proto se nevyužívá například metod rozšířeného spektra, ale jednoduché binární klíčování frekvenčním posunem, kdy se používají dvě frekvence, jedna pro přenos nul, druhá jedniček. (Gratton, 2007)

Z-Wave umožňuje použití topologie mesh, opět záleží na zařízeních, která se v síti používají. Pokud zvolíme zařízení, která zvládají vysílat i přijímat, můžeme je použít jako routery. Opět je potřeba počítat se zvýšenou spotřebou energie a není vhodné použití bateriového napájení.

3.1.2.4 Wi-fi

Technologie Wi-fi je známá především z použití pro pokrytí domácností internetem pro připojení notebooků, chytrých telefonů a dalších zařízení. Ve venkovním prostředí se tato technologie používá pro vytvářeních rychlých sítí v místech, kde není možné pokládat kabely. Počátky specifikace jsou v roce 1997, kdy byl publikován standard bezdrátové sítě

pod označením IEEE 802.11. V roce 1999 došlo k rozšíření o další specifikace 802.11a a 802.11b. V následujících letech došlo k uvolňování dalších specifikací, souhrnně popisuje nejrozšířenější z nich tabulka 2. (Shen, 2020)

Standard	Označení	Pásmo	Teoretická rychlost
802.11		2,4 GHz	2 Mbit/s
802.11a	Wi-Fi 1	5 GHz	54 Mbit/s
802.11b	Wi-Fi 2	2,4 GHz	11 Mbit/s
802.11g	Wi-Fi 3	2,4 GHz	54 Mbit/s
802.11n	Wi-Fi 4	2,4 / 5 GHz	600 Mbit/s
802.11ac	Wi-Fi 5	5 GHz	3,5 Gbit/s
802.11ah		863 – 868 Mhz (EU)	až 347 Mbit/s
802.11ax	Wi-Fi 6	2,4 / 5 / 6 GHz	9,6 Gbit/s

Tabulka 2 - Přehled verzí standardu 802.11

Z tabulky je patrné, jak docházelo mezi jednotlivými specifikacemi k nárůstu teoretické maximální přenosové rychlosti, s tím rostou i nároky na spotřebu elektřiny a proto je vhodné pro dlouhodobější použití zařízení, která by měla odesílat větší množství dat, napájet ze sítě či jiného stálého zdroje. Změnu přinesla specifikace 802.11ah, označovaná také jako HaLow, která je speciálně vytvořena pro internet věcí. Pro komunikaci se používá méně zarušené nelicencované pásmo pod 1 GHz – v Evropě je to 863 – 868 MHz, v USA 902 – 928 MHz. V závislosti na zemi a povolení regulačních úřadů se využívají šířky pásma 1 MHz, 2 MHz, 4 MHz, 8 MHz a 16 MHz. Všechny kanály jsou využívány jen ve Spojených státech, v Evropě se používá 1 MHz a 2 MHz. Wi-fi používá širokopásmovou modulaci s frekvenčním dělením kanálů, kdy přenos zajišťují desítky nosných kmitočtů. (Weiping and Choi, 2013)

HaLow je na rozdíl od běžných Wi-Fi sítí určena pro přenos malého objemu dat s nízkou spotřebou energie, dosahovaná rychlost je v řádech jednotek Mbit/s. Dosah sítě je přibližně 1 km a díky využití spektra pod 1 GHz má dobrou průchodnost překážkami. Se zvyšující se vzdáleností klesá přenosová rychlost a tím se ztrácí výhoda rychlejšího přenosu dat. (IEEE Standard for Information technology, 2017)

V současné době je největším problémem nedostatek zařízení, která by byla schopná s touto technologií komunikovat. To je jeden z důvodů, proč nedochází k většímu využití Wi-Fi HaLow. Ať už se jedná o routery, které nabízí například společnost Silex technology,

tak i koncová zařízení či moduly, které by uměly komunikovat na standardu 802.11ah. (Silex Technology America, Inc., 2020)

3.1.2.5 Mobilní sítě

Mobilní sítě v masovém měřítku jsou využívány od roku 1991, kdy byla ve Finsku spuštěna první síť druhé generace, zkráceně 2G. Na rozdíl od generace předchozí je přenos dat již digitální a umožňuje přenášet digitální hlasové hovory, SMS a ostatní pomocná data. Vývoj mobilních sítí stále pokračuje, v současnosti se připravují sítě šesté generace umožňující ještě rychlejší přenos dat. Pro potřeby internetu věcí ale není tolik důležitá přenosová rychlost, jako spíše nízká spotřeba energie a v některých aplikacích i velký dosah. V minulosti se využívala primitivní zařízení pro monitorování spínaných kontaktů a jiných událostí založená na prostém posílání SMS či automatickém volání na přednastavená čísla s předem nahranou hlasovou zprávou. Takový přenos je velice energeticky náročný a proto se začaly vyvíjet nové standardy pro IoT, které by umožňovaly přenos krátkých zpráv na velké vzdálenosti s co nejmenší spotřebou elektrické energie.

Jednou z takto vzniklých technologií je NB-IoT neboli Narrowband – Internet of Things. Za tímto standardem stojí organizace 3GPP, která jej představila v červnu roku 2016. Cílem této technologie je vytvořit síť pro zařízení s nízkou spotřebou energie, velkým dosahem a nízkou cenou. První vydání se označuje jako Release 13. O rok později se objevilo vydání Release 14, které snížilo spotřebu energie a přineslo další drobné změny. Samotná komunikace je postavena na protokolu, který se používá při LTE komunikaci. Došlo jen k omezení funkcionalit na minimum a úpravě pro přenos pouze malého množství dat. Toto je hlavní přínos technologie, neboť není nutné znovu pokrývat území novým typem sítě vysílaným z nových zařízení a antén, ale lze použít stávající LTE síť a softwarovou úpravu na straně vysílače. Pro přenos jsou využívány pásma pod 1 GHz – 700 MHz, 800 MHz, 900 MHz. Stejně jako Wi-Fi 6 i NB-IoT používá širokopásmovou modulaci s frekvenčním dělením. Přenosová rychlost v Release 14 je 159 kbps pro odesílání dat a 127 kbps pro příjem. Maximální velikost zprávy je 200 bytů. Vzdálenosti, na které je NB-IoT schopné komunikovat, jsou v řádech kilometrů, zdroje uvádějí dosah v městské zástavbě do 2 km, ve volném prostoru až 15 km. (Sinha, 2017) (Shen, 2020)

I za další technologií označovanou jako LTE-M stojí organizace 3GPP. Tato technologie se od předchozí liší v tom, že se zaměřuje na přenos většího množství dat a vyšší přenosovou rychlost, ale stále zachovává nízkou spotřebu elektrické energie. Oproti ostatním

technologiím určeným pro IoT podporuje LTE-M i hlasové funkce přes VoLTE (Voice over LTE). I zde se využívá stávající LTE infrastruktura jen se softwarovou změnou na straně vysílače. Přenosová frekvence využívá jen pásmo 800 MHz a modulaci s frekvenčním dělením pro odesílání dat. Velikost zprávy je omezena na 1600 bytů. Jak bylo zmíněno dříve, LTE-M umožňuje přenos hlasu, s tím souvisí i doba odezvy, která je zde mezi 10 a 15 ms, u NB-IoT jsou to řádově sekundy, uváděno je 1 až 10 sekund. (Shen, 2020)

Používání těchto technologií probíhá v licencovaných pásmech a je poskytováno jako služba různými operátory v závislosti na oblasti. Není možné vybudovat si vlastní síť s vlastní architekturou, ale je nutné využívat nabízené řešení od poskytovatele. Pro komunikaci se do koncových zařízení vkládají SIM karty podobně jako do mobilních telefonů a jiných zařízení, která využívají služby mobilních operátorů. S nabídkou služeb souvisí i cena, která se platí za používání sítě, případně za využití cloudových služeb pro zpracování dat.

3.1.2.6 Sigfox

Francouzská společnost Sigfox byla založena v roce 2010 a byla jednou z prvních, které přišly na trh s technologií pro internet věcí. Sigfox nabízí kompletní řešení od čidla až po zpracování příchozích dat. Model fungování je postaven na tom, že uživatel za měsíční poplatek dostává od poskytovatele konektivitu, zpracování zpráv a přístup k datům v cloudu. Možnost připojení je závislá na pokrytí signálem od poskytovatele, není možnost používat vlastní bránu a data si zpracovávat bez Sigfox Cloud.

Komunikace probíhá na nelicencované frekvenci 868 MHz v Evropě, ve Spojených státech je to pak 902 MHz. Pro připojení se využívá dvojstavová digitální modulace, kdy dochází k posunu fáze o 180° a tím odlišení mezi 1 a 0 v odesílané bitové zprávě. Původní síť umožňovala odesílání zpráv pouze ve směru od zařízení, až během vývoje byla realizována možnost odesílat zprávy i v opačném směru, ale pouze v předem nastavený čas po odeslání zprávy ze zařízení. Síť Sigfox omezuje komunikaci na maximálně 140 zpráv za den z jednoho zařízení a každá zpráva nesmí mít více než 12 bajtů. Pro příjem zpráv platí omezení na velikost 8 bajtů a maximálně 4 zprávy denně. Každá zpráva se odesílá třikrát za sebou, pokaždé na odlišném frekvenčním kanálu, tím dochází ke zvýšení spolehlivosti. Teoretický dosah závisí na prostředí, v otevřeném prostoru je možné přenášet data až na 50 km, v zastavěném prostředí 3 – 5 km. Díky použití pásma 868 MHz má Sigfox dobrou průchodnost zástavbou a je vhodný pro použití i ve městech. (Stoynov, 2019)

Sigfox je primárně určen pro přenos krátkých zpráv, tím nahrazuje notifikace ze zařízení v odlehlých oblastech, která musela dříve posílat například SMS či využívat datové přenosy od telefonních operátorů. Tato zařízení měla větší spotřebu energie a bylo nutné je v místech bez elektrické energie napájet pomocí baterií a ty dobíjet například solárním panelem. Technologie Sigfox je energeticky velice úsporná – průměrná spotřeba energie pro jeden přenos je 50 mikrowattů, u stejného přenosu přes GSM síť je to 5000 mikrowattů. (Hassan, 2018) (Sigfox, 2013)

Jak bylo řečeno výše, připojení je možné pouze přes sítě operátora. V současnosti zajišťuje připojení ve více než 70 zemích světa 50 operátorů, dle dostupných informací je pokryto 5,8 milion km^2 zemského povrhu. Počet připojených zařízení je 17 milionů a každý den se odesílá přes 70 milionů zpráv. Pokrytí signálem je především v západní a střední Evropě, Jihoafrické republice či Japonsku. V zemích východní Evropy dochází k postupnému zavádění. V ČR zajišťuje připojení operátor SimpleCell Networks ve spojení se společností T-Mobile, společně pokrývají 94% území. Tabulka 3 zobrazuje poplatky a limity množství přenášených zpráv pro jednotlivé tarify. (Introducing 0G network, 2021) (STANDARD PRICE LIST, 2020)

Tarif	Počet odesílaných zpráv	Počet přijatých zpráv	Cena za rok (do 999 zařízení)
Basic	2 / den	1 / týden	140 Kč
Plus	70 / den	2 / den	215 Kč
Ultra	140 / den	4 / den	247 Kč

Tabulka 3- Přehled tarifů SimpleCell Networks

3.1.2.7 LoRa a LoRaWAN

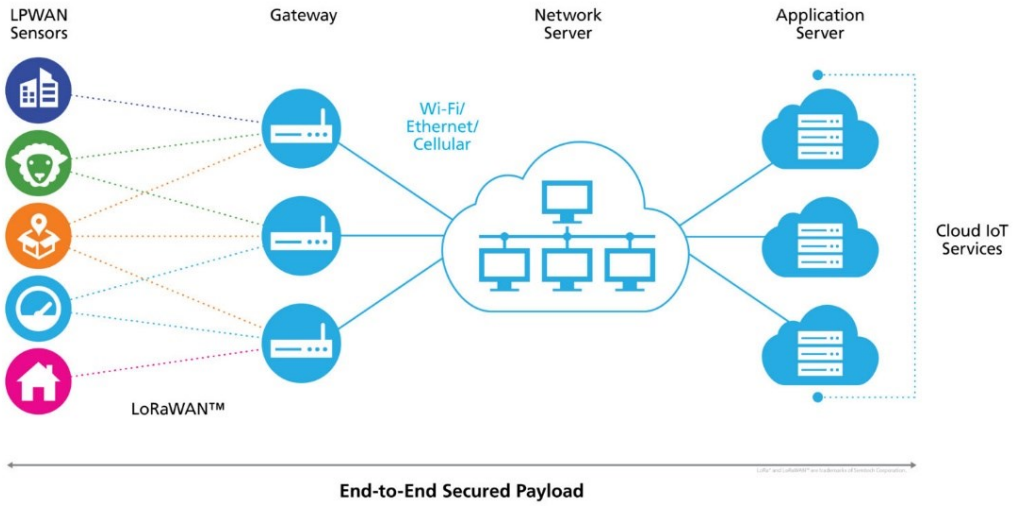
LoRaWAN je jedna z nejčastěji používaných technologií v sítích typu LPWAN. S tímto názvem se často zaměňuje zkrácenina slov Long Range, neboli LoRa, která definuje fyzickou vrstvu komunikace. LoRaWAN je konkrétní název komunikačního protokolu. Za modulací stojí francouzská firma Cycleo, která se po akvizici společností Semtech stala její součástí. Od počátku byl vývoj směrován pro komunikaci na velké vzdálenosti s nízkou spotřebou a odolností proti rušení. Jako u ostatních řešení, i zde se primárně cílí na přenos malého množství dat s nízkou rychlostí a minimální spotřebou elektrické energie. (LoRa Alliance, 2021)

Komunikace probíhá v bezlicenčních pásmech, v závislosti na oblasti se jedná o 433 MHz využívané v Asii, 868 MHz pro Evropu a 915 MHz v USA. V Evropě LoRa definuje 3 kanály, každý s šířkou 125 kHz. Modulace využívá opět metodu rozprostřeného spektra, ale přináší tzv. chirp, což je krátký lineární kmit. Tímto se frekvence dokola lineárně zvyšuje (Up-Chirp) či snižuje (Down-Chirp) mezi hranicemi frekvence. (Bankov, 2016)

LoRaWAN využívá topologii sítě typu hvězda, zařízení mohou komunikovat i mezi sebou, ale je nutné softwarově nastavit zařízení pro takovou komunikaci. Běžný způsob komunikace je přes bránu, případně více bran a společný stack server. Je možné využít i více bran a tím, že zařízení odesílá zprávu do všech směrů, se zvyšuje pravděpodobnost příjmu alespoň jednou branou. V případě, že jsou odesílaná data odchycena více branami, je na síťovém serveru, aby toto rozpoznal a zpracoval zprávu pouze jednou. Brána samotná funguje pouze pro příjem zpráv od zařízení a jejich přeposílání přes internet do řídícího serveru, případně pro komunikaci v opačném směru. Celá komunikace probíhá šifrovaně od klienta až po server použitím metody AES-128. Pro předání klíčů mezi serverem a zařízením se využívají dvě metody ABP a OTAA.

ABP (Activation By Personalisation) je způsob ručního vložení klíčů do zařízení před samotným spuštěním.

OTAA (Over the air activation) aktivace je způsob automatického předání klíčů mezi serverem a zařízením. Je nutné, aby zařízení obsahovalo jednoznačné identifikátory pro přihlášení a identifikaci, po ověření vrátí server klíče.



Obrázek 4- LoRaWAN architektura (Zdroj: https://tech-journal.semtech.com/expert-series-5-things-you-need-to-know-about-lorawan-based-gateways)

Komunikace v LoRaWAN síti může probíhat oběma směry, pro snížení spotřeby elektrické energie jsou definovány 3 třídy, kterými se zařízení při čekání na příjem zprávy řídí. Třídy se liší spolehlivostí příjmu a spotřebou energie.

Class A je nejúspornější třída, která po odeslání dat otevře pro příjem dat dvě okna v předem definovaných časech. Zpráva musí přijít v okamžiku, kdy je okno otevřeno, jinak nebude přijata a další možností příjmu bude až po dalším odeslání dat zařízením. Zařízení může přijmou zprávu jen v jednom okně, pokud je přijata hned v prvním, další se již neotvírá. Tuto třídu musí podporovat všechna zařízení.

Class B se podobá Class A, okna se ale neotvírají jen dvě, ale otvírají se v předem daných časech okna neustále dokola. Po přijetí zprávy se otevírají okna stále dál. Doručení zprávy je více pravděpodobné než u třídy předchozí. Dle časového intervalu pro poslouchání se zvyšuje spotřeba energie.

Class C je energeticky nejnáročnější variantou. Pokud koncové zařízení zrovna nevysílá, tak poslouchá a čeká na příjem, nedochází k uspávání zařízení. (The Things Network: Device Classes, 2021)

3.1.2.8 Srovnání bezdrátových připojení

Z výše uvedeného seznamu nejčastěji používaných technologií pro komunikaci v síti věcí je patrné, že je nutné při návrhu použití IoT zařízení zvolit správný způsob komunikace na základě požadavků na množství přenášených dat, rychlost komunikace, spotřebu energie

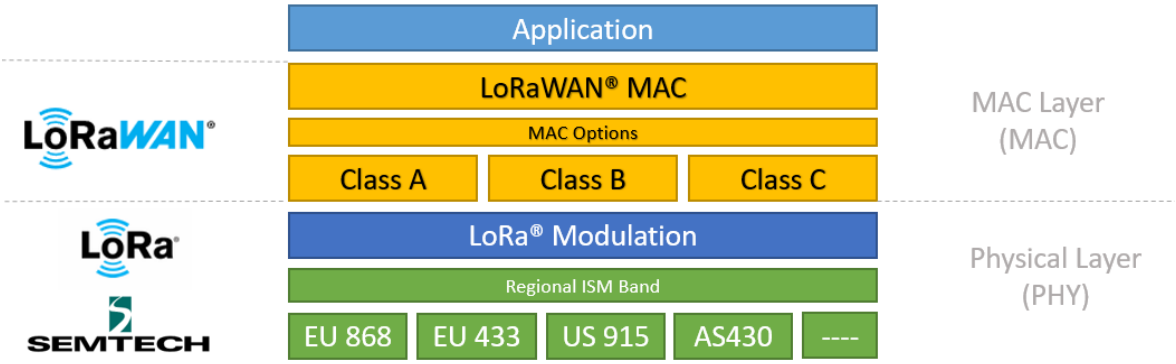
a dosah. Tabulka 4 shrnuje parametry do přehledného celku, který by měl usnadnit orientaci při volbě technologie. (Rehman, 2016) (Sendra, 2020)

Technologie	Dosah	Frekvence	Rychlost	Spotřeba energie
Bluetooth (BLE)	10 - 100 m	2,4 GHz	1 Mbps	Nízká
ZigBee	~ 30 m	868 MHz 902–928 MHz 2,4 GHz	250 kbps	Střední
Z-Wave	~ 30 m	868 MHz 908 MHz	100 kbps	Střední
Wi-Fi HaLow (802.11ah)	~ 1km	863 – 868 MHz 902 – 928 MHz	až 347 Mbps	Nízká - Střední
NB-IoT	2 - 15 km	700 MHz 800 MHz 900 MHz	159 kbps	Nízká
Sigfox	3 - 50 km	868 MHz 902 MHz	250 kbps	Nízká
LoRaWAN	10 - 50 km	433 MHz 868 MHz 915 MHz	250 kbps	Nízká

Tabulka 4 - Porovnání bezdrátových IoT technologií

3.2 Popis protokolu LoRaWAN

Jak již bylo zmíněno výše, jsou rozdíly mezi LoRa a LoRaWAN. LoRa je název fyzické vrstvy, definuje modulaci a frekvence, na kterých staví pak samotný LoRaWAN protokol. Fyzická vrstva je nazývána Media access control protokol, pro který se používá zkratka MAC. Celá následující podkapitola práce čerpá ze specifikace LoRaWAN protokolu verze 1.1. (LoRaWAN™ Specification v1.1, 2017)



Obrázek 5- LoRa vs LoRaWAN (Zdroj: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/)

3.2.1 Fyzický rámec - LoRa

Komunikace mezi koncovým zařízením a síťovým serverem probíhá pomocí jedné či více bran. Obsah fyzických rámců je v případě čipů společnosti Semtech pevně specifikován. Do fyzické vrstvy (PHY) patří dva typy rámců. Rámec, který je směřován od zařízení k serveru, se označuje jako uplink a je možné ho odesílat přes více bran. Rámec opačným směrem je nazýván downlink a je na síťovém serveru, aby zajistil, že bude směrován právě přes jednu bránu. Rámec fyzické vrstvy se liší pro uplink a downlink pouze v přítomnosti kontroly redundance (CRC).

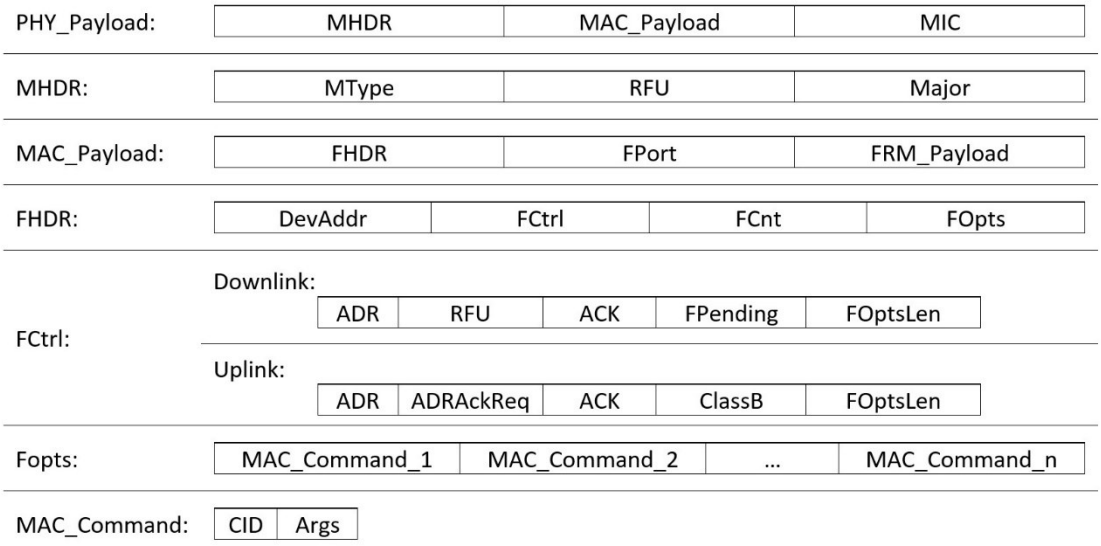
Preamble	PHDR	PHDR_CRC	PHY_Payload	CRC

Tabulka 5- Fyzický rámec LoRa

- **Preambule (Preamble)**: tento blok slouží k synchronizaci LoRa přijímače se začátkem vysílací frekvence. Velikost je 4 až 25 symbolů. Součástí preambule je i tzv. Sync Word, které slouží k rozlišení typů sítě,
- **PHDR**: blok PHDR, neboli Physical Header, obsahuje délku payloadu, přítomnost CRC na konci rámce a coding rate, označující poměr přenesených bitů oproti nesené informaci,
- **PHDR_CRC**: jak označení napovídá, v tomto bloku se jedná o kontrolní součet PHDR bloku, čímž se dá ověřit integrita paketu,
- **PHY_Payload**: nejdůležitější část celého rámce, obsahuje přenášená data. Její maximální délka je 255 bajtů,
- **CRC**: kontrolní součet rámce, jeho délka je 2 bajty.

3.2.2 Formát LoRaWAN

Nad fyzickou vrstvou LoRa byla vytvořena vrstva LoRaWAN, která usnadňuje sběr informací ze zařízení. V předchozí kapitole zmiňovaný fyzický rámec nese datový rámec označovaný jako PHY_Payload. Tento rámec se dále dělí na jednotlivé položky podle specifikace LoRaWAN, obsah některých rámců se liší podle typu zprávy či směru přenosu. I v jednotlivých verzích protokolu je odlišná struktura, následující popis odpovídá specifikaci verze 1.1, která je poslední vydaná v současné době. Obrázek níže graficky popisuje celou strukturu rámce a jeho rozpad.



Obrázek 6- Formát LoRaWAN zpráv

Datový PHY Payload rámec obsahuje následující části:

- **MHDR**: hlavička protokolu obsahuje typ zprávy a verzi protokolu,
- **MAC_Payload**: datový rámec nese informace, může být nahrazen žádostí Join Request, Rejoin Request nebo Join Accept,
- **MIC**: kontrolní součet zprávy (Message integrity code), který zabraňuje manipulaci se zprávou. MIC se počíta z MHDR | FHDR | FPort | FRM_Payload. V případě, že je zpráva typu Join Accept, neobsahuje kontrolní součet.

MHDR neboli hlavička protokolu se skládá z následujících částí:

- **MType**: označuje typ zprávy, jednotlivé typy zpráv popisuje tabulka č. 6,
- **RFU**: zkratka pro Reserved for future use, tato část je rezervována pro použití v budoucích verzích protokolu,
- **Major**: hlavní verze protokolu, podle které byl rámec zakódován. V současné době se používá pouze hodnota 00 označující LoRaWAN R1, další hodnoty jsou rezervovány pro budoucí použití.

MType hodnota	Typ zprávy
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data Up
101	Confirmed Data Down
110	Rejoin-request
111	Proprietary

Tabulka 6 - Tabulka typů MAC zpráv

MAC Payload se dělí na následující části:

- **FHDR**: hlavička rámce, kterou rozebereme níže,
- **FPort**: volitelná část, udává číslo MAC příkazu. Využívá se rozsah 1 – 233, 224 je vyčleněný pro testovací protokol a hodnoty 225 – 255 jsou rezervovány pro budoucí užití,
- **FRM_Payload**: zašifrovaný payload pomocí AES algoritmu s klíčem o délce 128 bitů.

FHDR neboli hlavička rámce obsahuje následující části:

- **DevAddr**: adresa zařízení ve zkráceném tvaru o délce 4 bajtů,
- **FCtrl**: frame control, dále se dělí a liší se pro uplink a downlink,
- **FCnt**: čítač zpráv,
- **FOpts**: pole pro MAC příkazy.

FCtrl obsahuje řídící a informační bity, liší se pro příchozí a odchozí zprávy:

- **ADR**: informace o Adaptive Data Rate, tedy jestli koncové zařízení požaduje či podporuje ADR,
- **RFU** (pouze downlink): rezerva pro budoucí využití,
- **ADRACKReq** (pouze uplink): informace o tom, zda je vyžadována zpráva při řízení datové rychlosti,
- **ACK**: informace o úspěšném přijetí předchozí zprávy,

- **FPenging** (pouze downlink): informace pro koncové zařízení, že na serveru čekají další zprávy na doručení,
- **ClassB** (pouze uplink): označení, zda zařízení pracuje s touto třídou,
- **FOptsLen**: délka MAC příkazů.

FOpts obsahuje pole s MAC příkazy, ty jsou ukládány šifrované. Zpráva nemusí obsahovat žádný příkaz. Samotný MAC příkaz se skládá pouze ze dvou polí. Pokud probíhá komunikace jen mezi serverem a jedním konkrétním zařízením, není potřeba uvádět zdrojovou či cílovou adresu.

- **CID**: identifikátor MAC příkazu,
- **Args**: volitelný argument příkazu.

V datovém rámci se kromě výše zmíněného mohou vyskytovat i žádosti Join Request, Rejoin Request anebo Join Accept. Tyto příkazy jsou podstatné pro zajištění komunikace při OTAA aktivaci.

Join request je příkaz, který slouží k registraci zařízení do sítě. Jsou mu předány informace potřebné pro generování šifrovacích klíčů a přiřazena dočasná adresa. Rámec se skládá z následujících částí:

- **JoinEUI**: identifikátor serveru, který zajišťuje odvození klíčů a identifikaci zařízení,
- **DevEUI**: v síti unikátní adresa zařízení,
- **DevNonce**: hodnota počítadla, která je navyšována s každým Join Request paketem, a slouží také pro generování podpisu.

Rejoin Request se používá při opětovném navázání komunikace po ztrátě nebo přepojení k jiné bráně. Existují 3 různé druhy Rejoin Request – 0, 1, 2 a dle toho se liší i obsah paketu. Typy 0 slouží k obnově všech parametrů, typ 2 taktéž s výjimkou změn nastavení rádia, typ 1 odpovídá Join Requestu. Zpráva se skládá z následujících polí:

- **Rejoin Type**: definuje typ požadavku – 0, 1 nebo 2,
- **NetID** (pouze typ 0 a 2): identifikátor sítě, do které je zařízení připojeno,
- **JoinEUI** (pouze typ 1): identifikátor serveru, který zajišťuje odvození klíčů a identifikaci zařízení,

- **DevEUI**: v síti unikátní adresa zařízení,
- **RJcount0**: počítadlo odeslaných požadavků typu 0 nebo 2,
- **RJcount1**: počítadlo odeslaných požadavků typu 1.

Zpráva typu Join Accept je generována na serveru jako odpověď na Join nebo Rejoin Request. Zpráva je přenášena jako standardní downlink, ale používá zpoždění JOIN_ACCEPT_DELAY1 nebo JOIN_ACCEPT_DELAY2. Stejně jako u RX1 a RX2 slotů se používá pro první okno frekvence a data rate jako při příjmu. Pro druhé okno je použita fixní konfigurace frekvence a data rate. Pokud není Join Request přijata, zařízení nedostává žádnou informaci. Join Accept zpráva obsahuje následující pole

JoinNonce	Home_NetID	DevAddr	DLSettings	RxDelay	CFList

Tabulka 7- Join Accept zpráva

- **JoinNonce**: počítadlo specifické pro každé zařízení, slouží k odvození klíčů, zařízení si uchovává poslední použitou hodnotu úspěšného připojení,
- **Home_NetID**: stejně jako NetID, označuje identifikátor sítě, do které se zařízení připojuje,
- **DevAddr**: adresa koncového zařízení,
- **DLSettings**: definuje, zda server podporuje protokol verze 1 (nenastaveno) nebo 1.1 a vyšší (nastaveno), a další downlink parametry,
- **RxDelay**: definuje zpožděním mezi TX a RX (příjmem a vysíláním),
- **CFList**: list síťových parametrů, je volitelný a závisí na regionu.

Při příjmu Join Accept zprávy by zařízení mělo akceptovat pouze zprávy, u kterých odpovídá kontrolní součet (MIC) a JoinNonce je vyšší, než uložená hodnota – v tom případě je do paměti zařízení uložená tato navrácená hodnota.

V závislosti na verzi protokolu se v DLSettings odvozují klíče různým způsobem. Pro verzi 1.0 se FNwkSIntKey a AppSKey odvozuje z NwkKey. SNwkSIntKey a NwkSEncKey je stejný jako FNwkSIntKey. Odvození těchto klíčů se provádí takto:

- $AppSKey = aes128\_encrypt(NwkKey, 0x02|JoinNonce|NetID|DevNonce|pad_{16})$
- $FNwkSIntKey = aes128\_encrypt(NwkKey, 0x01|JoinNonce|NetID|DevNonce|pad_{16})$
- $SNwkSIntKey = NwkSEncKey = FNwkSIntKey$

Kontrolní součet je definován takto:

- $cmac = aes128\_cmac(NwkKey, MHDR|JoinNonce|NetID|DevAddr|DLSettings|RxDelay|CFList)$

- *MIC = cmc[0..3]*

Pro verzi 1.1 a vyšší se používá odvození AppSKey z AppKey a odvození ostatních třech klíčů FNwkSIntKey, SNwkSIntKey a NwkSEncKey z NwkKey následujícím způsobem:

- *FNwkSIntKey = aes128_encrypt(NwkKey, 0x01|JoinNonce|JoinEUI|DevNonce|pad$_{16}$)*
- *SNwkSIntKey = aes128_encrypt(NwkKey, 0x03|JoinNonce|JoinEUI|DevNonce|pad$_{16}$)*
- *NwkSEncKey= aes128_encrypt(NwkKey, 0x04|JoinNonce|JoinEUI|DevNonce|pad$_{16}$)*

V případě této verze se výpočet kontrolního součtu provádí následovně:

- *cmac = aes128_cmac(JSIntKey, JoinReqType|JoinEUI|DevNonce|MHDR|JoinNonce|NetID|DevAddr|DLSettings|RxDelay|CFList)*
- *MIC = cmc[0..3]*

JoinReqType je pole, které je závislé na typu příchozí zprávy, zda se jedná o Join či Rejoin request, případně jaké verze.

Typ requestu a případná veze	Hodnota JoinReqType
Join Request	0xFF
Rejoin Request verze 0	0x00
Rejoin Request verze 1	0x01
Rejoin Request verze 2	0x02

Tabulka 8 - JoinReqType hodnoty

Pro zašifrování Join Accept se používají dva klíče, v případě Join Request zprávy se používá NwkKey, pro všechny verze Rejoin Request se používá JSEncKey odvozeným následovně:

- *JSEncKey = aes128_encrypt(NwkKey, 0x05|DevEUI|pad$_{16}$)*

Celá Join Accept zpráva šifruje následujícím způsobem:

- *aes128_decrypt(NwkKey nebo JSEncKey, JoinNonce|NetID|DevAddr|DLSettings|RxDelay|CFList|MIC)*

3.2.3 Navázání komunikace – aktivace zařízení

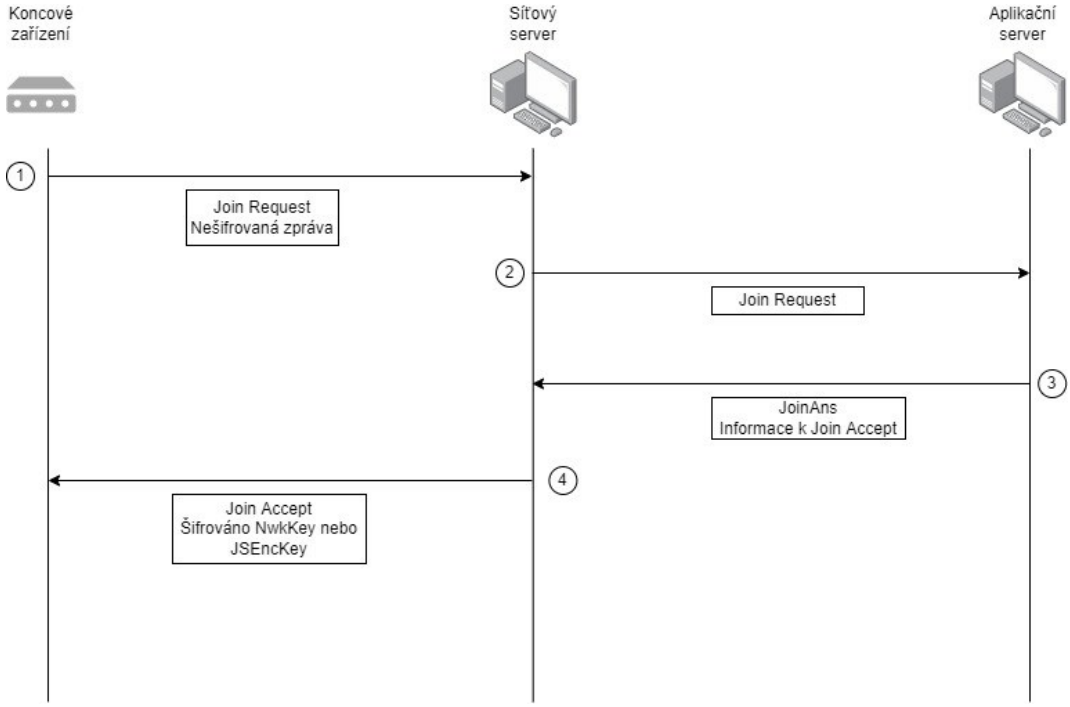
LoWaWAN definuje dvě různé možnosti aktivace koncového zařízení. Nejjednodušší způsob, který je nejrychlejší ale také nejméně bezpečný, je označován zkratkou ABP (Activation by Personalization). Univerzálnější způsob připojení, který má i lepší zabezpečení se označuje OTAA (Over the Air Activation).

ABP (Activation by Personalization)

Aktivace tímto způsobem používá předem definované klíče a adresu. Zařízení obsahuje svůj jednoznačný identifikátor, označovaný jako DevEUI, pro který se na serveru vygenerují relační klíče (NwkSKey a AppSKey) a adresa zařízení DevAddr. Tyto údaje se nastaví na koncovém zařízení, které je bude používat ke komunikaci. Pokud má koncové zařízení odesílat data, zašifruje paket pomocí klíčů a odešle. Síťový server, který takový paket obdrží, vyhledá dle DevAddr potřebné klíče a paket rozšifruje.

OTAA (Over the Air Activation)

Procedura aktivace OTAA je složitější a koncové zařízení musí touto připojovací procedurou projít pokaždé, když navazuje relaci novou. Podobně jako u aktivace ABP je nutné vygenerovat a nastavit na serveru i zařízení potřebné klíče a adresy. První z nich je JoinEUI, což je jednoznačný identifikátor serveru, DevEUI je stejný jako v případě ABP aktivace. Dále NwkKey a AppKey. Popis Join Request paketu byl uveden v předchozí kapitole, tento paket se odešle v nezašifrované podobě na síťový server, který zkontroluje hodnotu DevNonce, zda již nebyla použita a odešle tuto zprávu na připojovací server, který vygeneruje potřebné informace pro vytvoření klíčů. Pokud tato procedura proběhne, síťový server vygeneruje Join Accept paket, který zašifruje pomocí NwkKey v případě Join Request paketu nebo JSEncKey v případě Rejoin packetu a paket odešle na zařízení. V ten okamžik server i koncové zařízení mají potřebné klíče pro šifrování komunikace. Při odesílání dat ze zařízení se šifruje pouze FRM_Payload. MAC příkazy se šifrují síťovým relačním klíčem NwkSEncKey, aplikační data pak aplikačním relačním klíčem AppSKey.



Obrázek 7- Schéma zjednodušené OTAA procedury

3.3 Zařízení IoT

3.3.1 Porovnání dostupných zařízení

Pro porovnání byli vybráni zástupci zařízení, která jsou v dnešní době dostupná na trhu, splňují požadavek na bezdrátovou komunikaci a mají analogový vstup na spínaný kontakt. Těmto podmínkám vyhovuje více modelů, ale jejich parametry jsou často totožné nebo jsou založené na stejném základu.

Název zařízení	Cena	Komunikace	Baterie	Konfigurace
Comtac LPN AI	~ 6380Kč	LoRaWAN C Class	Ne	USB – txt soubor
Enginko MCF-LW06420	~ 6630Kč	LoRaWAN C Class	Ne	Webová aplikace
Enless Wireless Analog transmitter	~ 4940Kč	LoRaWAN A Class	Ano	Vlastní gateway
HW-Group SD-2xIn	~ 5360Kč	Wi-Fi/Ethernet	Ne	Cloud
HW-Group NB-2xIn	~ 10153Kč	NB-IOT	Ano	Cloud
HC Technologies HC-ANA-SIG	Nezjištěna	Sigfox	Ano	USB - Software
Ela innovation Blue PUCK AI	~ 1580Kč	BLE	Ano	Bluetooth - Software
TECHBASE ZS-10	Nezjištěna	Zigbee	Ano	Cloud

Tabulka 9 - Porovnání dostupných zařízení

Prvním zařízením je LPN AI od švýcarského výrobce Comtac. Zařízení komunikuje na protokolu LoRaWAN třídy C – je neustále připojeno k síti. Kvůli této konektivitě, která je energeticky velmi náročná, je napájení omezeno pouze na externí zdroj s napětím 24V. Analogové vstupy umožňují připojení dvou nezávislých zařízení pro monitorování. Celé zařízení splňuje stupeň krytí IP65. Konfigurace probíhá přes vestavěný microUSB port, který po připojení k počítači zpřístupní vnitřní paměť s konfiguračním souborem uloženým v textovém dokumentu. (Comtac LPN AI Datasheet, 2017)

Dalším dostupným zařízením je Enginko MCF-LW06420. Komunikace je opět postavena na LoRaWAN třídy C. Přístroj neobsahuje baterii, je nutné jej napájet přes USB nebo externím zdrojem. K dispozici jsou 4 kontakty pro připojení zařízení, pouze dva ale umožňují vyvolat alarm. Konfigurace zařízení probíhá přes microUSB port, ale k nastavení slouží webová aplikace na stránkách výrobce. (Mcf 88 MCF-LW06420 Datasheet, 2020)

Třetím zástupcem je Enless Wireless Analog. Toto zařízení opět využívá protokol LoRaWAN, tentokrát ovšem v třídě A, která je nejméně energeticky náročná, a proto je

zařízení možné provozovat na bateriové napájení. Udávaná výdrž je až 15 let v ideálních podmínkách. Vstupní informace jsou sbírány přes jeden kontakt uvnitř zařízení. Zařízení splňuje stupeň krytí IP65. Konfigurace je dostupná přes výrobcem dodávanou bránu, pomocí níž je možné bezdrátově zařízení nastavovat. (Installation Guide LoRaWAN range of devices, 2021)

Další dvě zařízení jsou vyráběny českou společnosti HW-Group s.r.o. Jedná se o téměř totožné modely, které se liší jen ve způsobu komunikace. Přístroj s označením SD-2xIn komunikuje pomocí bezdrátové sítě Wi-Fi - 802.11 b/g/n nebo pomocí ethernetové sítě. Bezdrátová komunikace je v tomto případě energeticky náročná, z toho důvodu je zařízení napájeno externím zdrojem. Zařízení NB-2xIn je založeno na technologii NB-IoT a pro svůj provoz potřebuje SIM kartu, která je součástí balení a je předplacena na 3 roky. Díky nižší energetické náročnosti je zařízení vybaveno nenabíjecí baterií s teoretickou výdrží až 3 roky, taktéž je možné napájení pomocí externího zdroje. Pro oba produkty je shodný počet dvou vstupů pro připojení monitorovaných zařízení. Konfigurace je v obou případech možná pouze přes HW Cloud či SensDesk provozovaný výrobcem těchto zařízení. (HW Group NB devices, 2020), (HW Group SD devices, 2019)

Zástupcem sítě Sigfox je zařízení HC-ANA-SIG od výrobce HC Technologies. Jedná se o produkt se stupněm krytí IP67, který je vybaven bateriovým napájením i možností napájení externím zdrojem. K připojení monitorovaných zařízení či externích čidel slouží čtyři vstupy. Konfigurace je dostupná po připojení zařízení přes IDC10 konektor a sériový převodník k USB portu počítače a za využití dodávaného software. (HC-DIG devices installation manual, 2018)

Technologie Bluetooth Low Energy zastupuje zařízení Blue PUCK AI francouzského výrobce Ela Innovation. Přístroj v nastavených intervalech odesílá hodnoty z jednoho dostupného vstupu. Napájení je řešeno nevyměnitelnou baterií či externím zdrojem. Nastavení probíhá pomocí dodávaného softwaru před Bluetooth. (Ela innovation BLUE PUCK AI Product sheet)

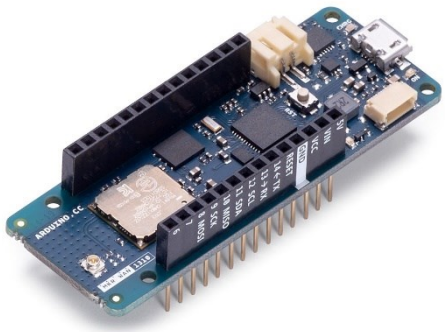
Posledním vybraným zástupcem je Techbase ZS-10. Zařízení komunikuje na síti ZigBee a pro jeho provoz je nutné použít výrobcem dodávanou bránu a cloudovou službu iModCloud, pomocí které probíhá i konfigurace. Přístroj je dodáván ve dvou variantách, a to buď se čtyřmi nebo jedenácti vstupy. V obou případech je počítáno s bateriovým napájením s udávanou výdrží 2,5 roku. Konfigurace je dostupná přes cloudovou službu poskytovanou výrobcem. (ZigBee Sensor ZS-10, 2014)

3.3.2 Arduino MRK WAN 1310

Arduino je open-source elektronická platforma založená na hardwaru a softwaru, která je jednoduchá k použití a je možné ji rychle přizpůsobit a použít jak při vývoji, tak i při nasazení na různé projekty. Hardwarové desky Arduina jsou dnes vyráběny v mnoha velikostech a konfiguracích. Nejznámější je Arduino Uno, vybavené mikrokontrolerem ATmega328, případně menší verze Adruino Nano se stejným procesorem. (Malý, 2017)

Postupem času Arduino vytvořilo speciální třídu zařízení označovanou jako Arduino MKR, která se zaměřuje na segment IoT. Jedná se o malé desky s velikostí 61,5 mm x 25 mm a váhou v nižších desítkách gramů. Všechna tato zařízení jsou poháněna mikrokontrolerem Arm Cortex-M0 SAMD21 a SRAM 32K. Celá deska pracuje s napětím 3,3 V, ale je možné pro napájení využít USB nebo pin VIN, s napětím 5V. Díky napěťovému regulátoru je možné využít i napájení bateriemi. Připojení k síti se liší dle jednotlivých modelů, k dispozici jsou verze pro sítě Sigfox, LoRa, GSM, WiFi, NB-IoT. (Arduino Products, 2022)

Pro programování Arduina se využívá open-source vytvářené IDE pro psaní programů a jejich následný upload do zařízení a komunikaci se ním. Využívaným jazykem pro tvorbu tzv. Sketch, jak je označován program pro Arduino, je C nebo C++. Pro snadnější práci je často využíván framework Wiring, který pomocí několika řádků kódu umožňuje komunikaci s hardwarem připojeným k Arduinu. To umožňuje rychlejší změny a testování v rámci prototypového vývoje. (Wiring, 2022)



Obrázek 8 - Arduino MKR WAN 1310 (Zdroj: https://store.arduino.cc/products/arduino-mkr-wan-1310)

Pro potřeby této diplomové práce bylo použito zařízení Arduino MKR WAN 1310, které využívá pro připojení k síti technologii LoRa. Zařízení patří do dříve zmíněné skupiny Arduino MKR. Rádio, které je zodpovědné za připojení, je dodáváno výrobcem Murata a jedná se o model Murata CMWX1ZZABZ, které je možné provozovat na frekvencích

kolem 433, 868 a 915 MHz pro použití ve všech regionech. Za zabezpečení je zodpovědný element ATECC508A. Pro připojení a komunikaci jsou k dispozici digitální a GPIO piny. Pro připojení externí antény je k dispozici UFL konektor, při volbě antény je třeba ověřit, v jakém frekvenčním spektru bude zařízení komunikovat. Pro uložení programu je k dispozici 256KB paměti, zařízení zároveň obsahuje i externí flash paměť o velikosti 2MB. (Arduino MKR WAN 1310, 2022)

3.3.3 Mikrotik wAP LoRa8 kit

Společnost Mikrotik je producent hardwaru a softwaru síťových prvků, jsou používána ve 145 zemích světa. Jejich zařízení jsou určena jak pro domácí použití, tak i pro použití u poskytovatelů internetového připojení nebo ve firemních infrastrukturách. Kromě hardwaru stojí společnost i za operačním systémem RouterOS, který je součástí většiny jejich zařízení, ale je možné ho provozovat i na klasických počítačích. (Mikrotik Company Profile, 2022)

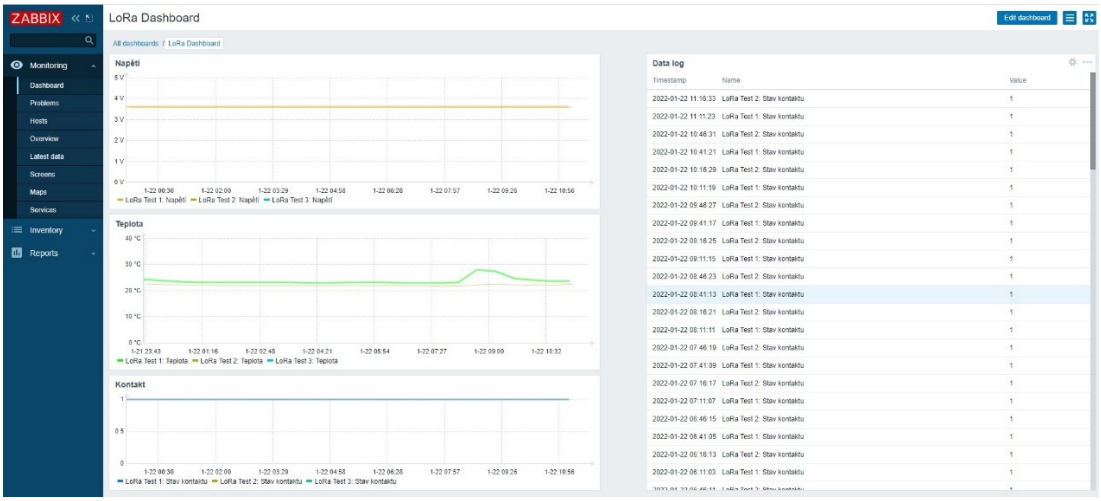
Na konci roku 2019 Mikrotik oznámil nové produkty pro síť LoRa, ať už šlo o samotnou mini PCIe kartu, wAP LoRa8 kit nebo antény v potřebných frekvencích. Zařízení wAP LoRa8 je v jednom balení zabalený RouterBoard s procesorem o taktu 650MHz a pamětí 64MB RAM, který je vybaven ethernetovým portem s podporou pasivního PoE, Wi-Fi 4 generace a vestavěnou kartou pro příjem LoRa signálu včetně vestavěné 2 dBi antény. Zařízení má připravený konektor i pro připojení externí antény, v tom případě je nutné rozebrat obal a přepojit anténu uvnitř modulu. Ve verzi wAP LoRa8 jsou dostupné pouze frekvence kolem 868MHz, pro využití frekvencí 902-928 MHz je v nabídce zařízení s označením wAP LoRa9. (Mikrotik IoT products, 2020)

Operační systém, je RouterOS v aktuálně poslední verzi 7.1.1 (leden 2022). Oproti běžnému nastavení je dostupná položka „LoRa“, která obsahuje možnosti nastavení brány. Důležitou položkou je Gateway ID, což je unikátní 64-bitové ID brány, které je nutné nastavit na síťovém serveru, na který budou odesílána data. V defaultní konfiguraci jsou předpřipravené servery The Things Network, je možné přidat servery vlastní.

3.4 Zabbix

Zabbix je monitorovací systém pro sběr dat a dohled nad síťovou infrastrukturou, servery, počítači, IoT senzory a dalšími zdroji. Zabbix server je volně šiřitelný s možností instalace na většinu linuxových systémů, monitorovací klienti pro koncová zařízení

podporují naprostou většinu běžně používaných operačních systémů jako je Windows, Linux, macOS, Solaris, BSD. Kromě kontroly pomocí klientů je možné využít protokol SNMP ve všech verzích například pro síťové prvky. Lze využít i dohled na webové stránky, využívat API třetích stran či industriální protokoly, možnosti využití jsou neustále rozšiřovány. Zabbix umožňuje kromě dohledu nad hardwarem i monitoring softwaru, využití databází a aplikací, sběr logů. Při monitorování velkého počtu klientů jsou zvýšené nároky na podkladovou databázi, Zabbix pracuje i s trendy, které jsou automaticky dopočítávané, proto je důležité při nasazení zvolit robustní databázi s dostatkem systémových prostředků, aby nedocházelo k jejímu zahlcení. (Zabbix, 2022)



Obrázek 9- Náhled na Zabbix dashboard

Sebraná data umožňuje Zabbix vizualizovat, nastavovat vlastní metriky, kontrolovat trendy. Uživatelské rozhraní je plně konfigurovatelné každým uživatelem, nabízí širokou škálu widgetů a grafů. Lze zobrazovat mapy sítě se stavem připojení v reálném čase či naplánovat pravidelné reporty. (Zabbix, 2022)

Detekci problémů lze postavit na samotných sebraných hodnotách či nedostupnosti zařízení, zároveň je možné ale využít i předpověď dle aktuálního trendu a tím předcházet problémům ještě před jejich výskytem. Pro detekování anomálií je možné využít i strojové učení, které automaticky vyhodnocuje aktuální situaci na základě minulých dat a tím predikovat možnost vzniku problému. (Zabbix, 2022)

Pro každý sebraný údaj je možné nastavit hlídání pomocí tzv. triggerů. Při tvorbě těchto spouštěčů se definuje závažnost nastalého problému, nastavení závislostí či automatické další kroky, které má systém udělat pro odstranění problému. Důležitým prvkem jsou i notifikace uživatelů na základě skupin a závažností nejen pro případ, kdy není

možné využít automatické akce. Možnosti notifikací jsou široké, pomocí dodatečných pluginů lze dodat i protokoly, které nejsou v základní instalaci. Běžně se využívají emailové notifikace či notifikace pomocí SMS, ať už přes samostatný modem připojený k serveru či přes API a službu třetích stran. (Zabbix, 2022)

4 Praktická část

Realizace dohledového systému a jeho nasazení je určeno pro potřeby 3. lékařské fakulty Univerzity Karlovy. Fakulta disponuje množstvím hlubokomrazících boxů pro uchovávání výzkumných vzorků. Zařízení byla nakupována průběžně a jsou průběžně obnovována, proto se nejedná vždy o stejný model, ale různé typy boxů od různých výrobců. Monitorování stavu teploty uvnitř boxů je komplikované vzhledem k velice nízkým teplotám (často kolem -75°C), stejně tak i utěsnění je kompaktní a vložení teplotního čidla do vnitřku by bylo obtížné. Společným prvkem všech těchto zařízení je přítomnost takzvaného alarm kontaktu, který je zobrazený na Obrázku 10. Jedná se o spínaný kontakt, který v případě problému se zařízením změní svůj stav. I samotný kontakt je bateriově zálohovaný pro případ výpadku proudu. Mrazící box má několik úrovní dohledu nad svým stavem. Nejvážnější problémy, kdy se používá tento kontakt, je v případě výpadku proudu, příliš nízké nebo naopak vysoké teploty. Nastavení teplotních rozsahů je možné přes ovládací rozhraní na těle boxu. Možnost připojení je obvykle dvojí:

- Normally open (N.O. nebo NO): Normálně otevřeno, kontakt je rozpojený v neproblémovém stavu, v případě poruchy se spojí
- Normally closed (N.C. nebo NC): Normálně uzavřeno, kontakt je spojený v neproblémovém stavu, v případě poruchy se rozpojí

Obrázek 10- Detail alarm kontaktu

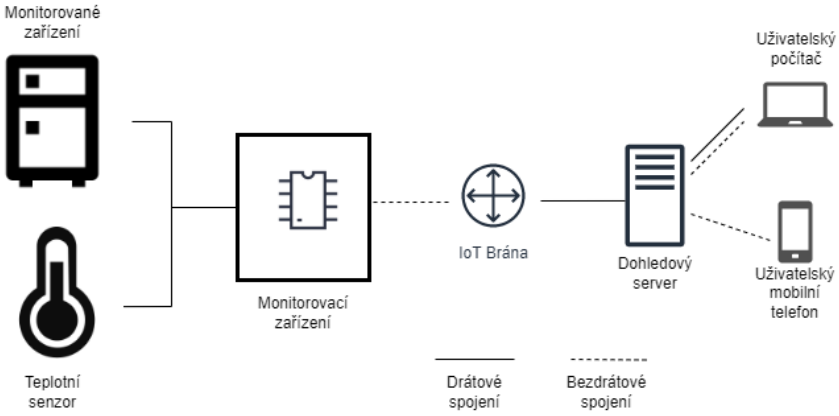
4.1 Návrh systému

Cílem práce je vytvoření funkčního systému pro monitoring hlubokomrazících boxů a dalších zařízení pomocí spínaného kontaktu a také monitoring prostředí, ve kterém se zařízení provozují. V průběhu dosavadního monitorování pomocí zařízení Poseidon 3266 od společnosti HW Group s.r.o. vyvstalo několik požadavků na provoz nového způsobu monitorování. První změnou je způsob připojení zařízení k síti, nahrazované řešení používá ethernetovou síť, což komplikuje případné přesuny. Další problém nastává v místech, kde je větší množství monitorovaných zařízení a je nutné je všechny připojit k síti, pak je nezbytné použití switche. Dalším požadavkem na změnu je možnost bateriového napájení či jeho kombinace s napájením z elektrické sítě. Monitoring se užívá i na Dewarovu nádobu na uchování kapalného dusíku, u kterého se hlídá hladina. Tato nádoba je často převážena mezi různými laboratořemi a po jejím odpojení od napájení dojde k vypnutí monitorovacího prvku, do jeho opětovného připojení není znám stav, ve kterém se spínaný kontakt nachází. Posledním požadavkem byla minimalizace nastavování zařízení a případná změna konfigurace nezávisle na technologiích třetích stran. Nahrazované řešení používalo pro konfiguraci přes webový prohlížeč za pomoci Flash Playeru, jehož systémová podpora byla ukončena na konci roku 2020. Po této analýze byly stanoveny požadavky na nový systém následovně:

- Bezdrátové připojení k síti,
- možnost provozu pouze v rámci lokální sítě bez využití služeb třetích stran,
- bateriové napájení s možností napájení ze síťového zdroje,
- minimalizace nastavování samotného monitorovacího zařízení,
- možnost monitorovat teplotu okolního prostředí,
- snadné umístění monitorovacího zařízení do blízkosti monitorovaného zařízení,
- notifikace uživatelů zodpovědných za monitorované zařízení ve stejném formátu jako v případě nahrazovaného zařízení, tj. email a SMS při vzniku problému i při jeho vyřešení.

4.2 Návrh systému

Návrh fungování systému vychází z ověřeného principu, který používá současné řešení již více než 8 let. Základem je využití spínaného kontaktu (alarm kontakt) na monitorovaném zařízení, které při poruše zařízení změní svůj stav. Na tento kontakt bude připojeno monitorovací zařízení, které bude sledovat změny kontaktu, zařízení bude v sobě obsahovat teplotní čidlo s možností připojení externího teplotního senzoru pro dlouhodobý dohled nad prostředím, ve kterém jsou zařízení provozována. Monitorovací zařízení se bude v pravidelných intervalech hlásit na centrální server a bude předávat aktuální hodnoty kontaktu, teploty a stavu baterie. Při změně stavu kontaktu odešle na server zprávu s aktuálním stavem. Přenos bude zajištěn bezdrátově a experimentálně bude vyzkoušen potřebný počet bran pro příjem signálu. Server zpracuje zprávy, které získá od gateway a na základě nastavení předá hodnoty jednomu nebo více Zabbix serverům, který hodnoty uchová a dle předvoleb zareaguje požadovaným způsobem, nejčastěji odesláním SMS a emailu uživatelům zodpovědným za dohled nad zařízením. Obrázek 11 zobrazuje schéma přenosu informace od monitorovaného zařízení až k uživateli.



Obrázek 11- Návrh komunikace

Jedním z požadavků je možnost použití systému uvnitř lokální sítě bez služeb externích subjektů. Tímto požadavkem je výrazně omezen výběr komunikačních technologií. Vzhledem k provozu v budově, která obsahuje velké množství Wi-Fi vysílačů v pásmu 2,4 GHz, byla pro snazší prostupnost signálu zvolena technologie LoRaWAN pracující v pásmu pod 1 GHz s využitím vlastní brány a síťového serveru ve vnitřní síti.

4.3 Monitorovací zařízení

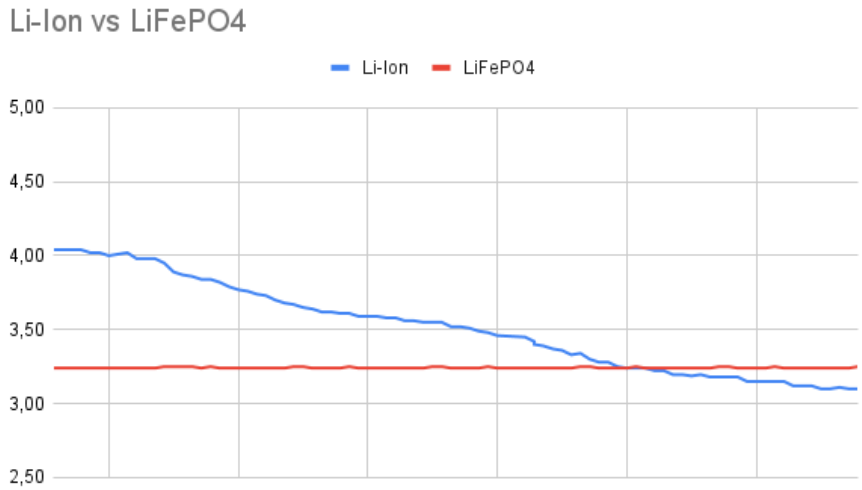
4.3.1 Výběr technologie a zařízení

Jako hlavní součást monitorovacího zařízení bylo zvoleno Arduino MKR WAN 1310, které podporuje technologii LoRa. Výběr zařízení probíhal na základě komunikační technologie. Požadavkem byla nezávislost na službách třetích stran, možnost provozu vlastní sítě a minimalizace počtu bran. Těmto požadavkům vyhovuje technologie LoRa. Na výběr jsou zařízení od firem Arduino, PyCom či například LilyGO. V době, kdy docházelo k výběru technologie, byl celosvětový problém s nedostatkem čipů a logistickými problémy s dodávkami z Asie, což ovlivnilo finální volbu, neboť dostupné bylo především Arduino a jejich model MKR WAN 1310. Pro bateriové napájení by se hodila spíše zařízení založená na ESP32 kvůli nízké spotřebě. Při návrhu se počítalo s možností výměny zařízení v budoucnu za jiný typ, software by v případě ESP32 zůstal stejný, jen by se upravily vstupní hodnoty GPIO pinů. Navrhnutý plošný spoj by se musel přizpůsobit rozměrům nového zařízení.

Prototypování zapojení probíhalo na nepájivém poli za pomoci DuPont kabelů, které jsou používány pro rychlé změny v konfiguraci. Během vývoje se nejprve využívalo napájení z USB portu, až později se použilo bateriové napájení.

Pro bateriový provoz byly zvoleny baterie velikosti 18650. Byly otestovány baterie různých výrobců a technologií. Jako první byla pro provoz vybrána baterie typu $LiFePO_4$, která se dle parametrů jevila jako nejvhodnější. Dle specifikací má nominální napětí 3,2V, životnost cca 2000 nabíjecích cyklů a lépe snáší teplotní rozdíly či provoz při nižších teplotách. Baterie je možné nabíjet v jakémkoliv stavu nabití, netrpí paměťovým efektem ani nemají samovybíjecí efekt. To vše ukazovalo na ideální baterii pro použití v navrhovaném zařízení, bohužel se během provozu ukázalo, že baterie dlouho drží nominální napětí kolem 3,2 V, ale poté má prudký pokles pod úroveň potřebnou pro provoz Arduina. Proto se často stávalo, že baterie hlásila „plné nabití“ a při pokusu o další odeslání zprávy se zařízení již neprobudilo kvůli nízkému napětí článku. Další typem baterie, která byla zkoušena, byl článek Li-Ion, nejprve se stejnou kapacitou (1500mAh) jako v předchozím případě, poté s kapacitou 3500mAh. Li-Ion baterie, stejně jako $LiFePO_4$, netrpí samovybíjením ani paměťovým efektem. Nominální napětí baterie je 3,6V, provozní až 4,2V. Při testování těchto baterií bylo dosaženo delší výdrže při stejných podmínkách. Taktéž docházelo k lineárnímu poklesu napětí během provozu, což umožnilo informovat

uživatele o nízkém stavu baterie. Obrázek 12 zobrazuje průběh poklesu napětí v době dvou měsíců během testování baterií v identickém prostředí na dvou nepájivých polích umístěných vedle sebe. Pokles Li-Ion baterie je patrný v průběhu první poloviny, poté osciluje okolo hodnoty 3,1 – 3,2 V.



Obrázek 12- Průběh vybíjení Li-Ion a LiFePo4 baterií

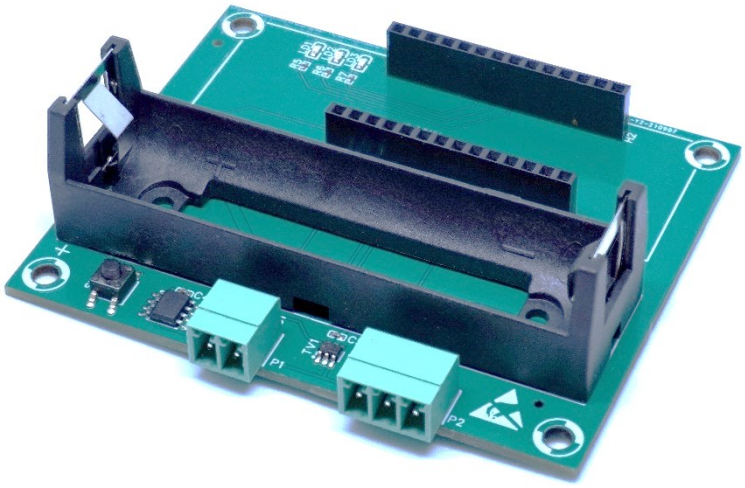
Při vývoji prototypu bylo také testováno probouzení zařízení při změně stavu kontaktu a odeslání zprávy. Pro pravidelné přepínání stavu bylo využito spínacích hodin a relé, které při průchodu proudu sepnulo kontakt a drželo kontakt sepnutý po nastavenou dobu na spínacích hodinách. Tento cyklus se opakoval po dobu 4 dní, kdy došlo každých 15 minut ke změně stavu a odeslání zprávy. Během tohoto testu nedošlo k tomu, že by se zařízení neprobudilo. Po testování byl navržen tištený spoj, do kterého se Arduino vloží.

4.3.2 Plošný spoj a anténa

Návrh plošného spoje byl realizován v programu EasyEDA. Základem je 1,6 mm dvouvrstvá deska s rozměry 65x80 mm, v každém rohu je montážní otvor o průměru 3,5 mm. Střed otvorů je 4 mm od každé strany. V zadní části jsou umístěny dva 14 pinové headery pro připojení Arduina a tři led diody – zelená, oranžová a červená - pro debuggování výstupů nebo pro zobrazení stavu zařízení. V přední části jsou dva systémové terminální bloky, jeden se dvěma piny, druhý se třemi, pro připojení monitorovaného zařízení a externího teplotního čidla. Na desce je dále umístěn tlačítkový spínač, teplotní čidlo typu DS18B20Z, vývody pro připojení pouzdra na baterii a napájecích kabelů. Zapojení konektoru pro externí teplotní čidlo a integrované čidlo je v sérii za sebou, což usnadňuje

vyčtení hodnoty teploty. Díky tomu není potřeba softwarově ošetřovat, zda je či není připojeno externí čidlo, použije se první hodnota, která je z 1-wire sběrnice přečtena.

PCB desky byly dle návrhu vyrobeny specializovanou firmou a dodány včetně osazení. Před montáží byl ještě osazen držák baterie velikosti 18650 do předem připravených otvorů a připájen, jak ukazuje obrázek 13. Taktéž byly umístěny vodiče zakončené konektorem JST-PH 2mm pro připojení napájení do zařízení.



Obrázek 13- PCB s osazeným držákem baterie

Připojení antény k zařízení je možné pomocí micro U.FL. Při návrhu se počítalo s dvojím řešením připojení antény, první je využití 2 dBi antény se samolepící částí, která bude umístěna uvnitř krabičky a zapojena přímo do konektru na zařízení. Druhé řešení počítá s externí anténou, která bude pomocí micro U.FL – SMA kabelu vyvedena do zadní části krabičky a bude možné zde připojit anténu s větším ziskem, pokud by to pro použití bylo nutné.

4.3.3 Krabička

Pro monitorovací zařízení byla navrhnuta krabička, která je částečně uživatelsky modifikovatelná na základě požadavků, které na zařízení jsou. Jako základ posloužil návrh krabičky šiřitelný pod Creative Common licencí s názvem „The Ultimate box maker“. Jedná se o zdrojový kód pro OpenSCAD, který je pomocí tzv. customizeru možné uživatelsky upravovat. Ve zdrojovém kódu byly upraveny výchozí parametry pro navrhnutou desku, upraveny otvory pro vstupy a výstupy a přidány textové popisky na správné pozice. Dále byly vytvořeny moduly pro specifické použití s navrženým plošným spojem. Jedná se o otvory na tlačítko v přední části a boční otvor pro připojení USB konektoru. Ve spodní části boxu byly navrženy prolisy na umístění pásků pro přichycení k monitorovanému zařízení. Všechny doplněné moduly jsou uživatelsky upravitelné, dají se vypínat a zapínat, případně nastavovat velikost.

Umístění PCB desky je zároveň s přední stranou krabičky z důvodu snadného přístupu ke konektorům pro připojení. V zadním panelu je navrhnutý otvor pro vložení SMA konektoru sloužící k připojení externí antény. Ve spodním dílu krabičky jsou v rozích čtyři distanční sloupky s otvory na šroubky s metrickým závitem M3 a délkou 5mm. Pro uchycení byly zvoleny šrouby s imbusovou půlkulatou hlavou. Otvory pro spojení spodní a horní části jsou připraveny na stejný průměr a délku šroubů. Ve spodní části jsou průchozí spojovací otvory, v horní pak otvory menší pro zaříznutí závitu šroubu.

Jak již bylo zmíněno, pro snazší uchycení dohledového zařízení k mrazícímu boxu nebo jinému přístroji je možné při generování spodní části zapnout prolis a nadefinovat jeho velikost. Do prolisů je možné vložit pěnovou oboustrannou pásku nebo magnetický lepící pásek. V testovacích zařízeních se používají dva samolepící magnetické pásky o šířce 20 mm a délce 60 mm. Dle výrobce je udávaná magnetická síla 102g/cm^2, což při ploše pásků zajišťuje nosnost téměř 2,5 kg. S předpokládanou váhou kompletního zařízení včetně baterie kolem 170g by měla tímto být zaručena dostatečná fixace v libovolné poloze.

Po návrhu byly exportovány STL soubory, které byly pomocí programu PrusaSlicer připraveny pro tisk na 3D tiskárně. Pro tisk s tryskou o průměru 0,4 mm byla zvolena výška vrstvy 0,10 mm. Při vhodném umístění na tiskovou podložku je možné tisknout jednotlivé díly bez podpor. Přední a zadní panel má navíc na sobě textové popisky, proto je při tisku po dokončení vrstvy ve výšce 1 mm změněna barva filamentu na jinou.

Pro tisk spodní a horní části krabičky byl zvolen polymer PET-G s příměsí samozhášivých aditiv, která zabraňují vznícení materiálu v případě vystavení otevřenému plameni. Tisk proběhl na tiskárně Průša MK3S+ s hladkým tiskovým plátem. Celková doba tisku s výše uvedeným nastavením je kolem pěti hodin.

Obrázek 14- Čelní pohled na zařízení

Obrázek 15- Zadní pohled na zařízení

4.3.4 Zdrojový kód zařízení

Vývoj řídícího softwaru probíhal v Arduino IDE. Pro případné změny v návrhu hardwarového zařízení byly definovány používané piny Arduina jako proměnné, nebude proto nutné dohledávat jednotlivé označení v celém kódu, ale stačí je změnit na jednom místě. Stejným způsobem jsou definované parametry připojení k síťovému LoRa serveru.

Při spuštění zařízení, ať už vložením baterie nebo připojením napájení do USB portu, se správně nastaví vstupní a výstupní piny a zařízení rozsvítí oranžovou diodu. Od toho okamžiku zařízení čeká na stisknutí tlačítka, kterým se zahájí proces připojení k síti a odeslání první zprávy. Po stisknutí tlačítka je nejprve spuštěn radiový modul. Pokud není dostupný, rozsvítí se červená dioda a zařízení se po 5 sekundách restartuje. Když se podaří rádiový modul zapnut, rozsvítí se zelená dioda a pokračuje připojení k síti LoRa. Zde jsou dvě možnosti, buďto se použije ověření ABP anebo OTAA v závislosti na nastavení stack serveru. Pokud se zařízení nepřipojí, opět se rozsvítí červená dioda a zařízení se restartuje. Jako poslední se nastavuje pin, který dokáže zařízení probudit ze spánku v případě změny jeho stavu.

```
void zprava() {
  /* Metoda pro odeslání zprávy s daty.
   * Nejprve se resetuje instance CayenneLPP.
   * Pote se vyčtou hodnoty vstupů.
   * Přidají se do instance CayenneLPP se
   * zvoleným kanálem a typem hodnoty.
   * Aktivuje se modem, do nového paketu se zapíše
   * instance s hodnotami o dané velikosti.
   * Paket se odešle (false = Unconfirmed Data Up)
   */
    lpp.reset();
    int kontakt_hodnota = kontakt();
    float teplota_hodnota = teplota();
    float baterie_hodnota = baterie();
    delay(500);
    lpp.addDigitalOutput(1, kontakt_hodnota);
    lpp.addTemperature(2, teplota_hodnota);
    lpp.addAnalogInput(3, baterie_hodnota);
    modem.setPort(3);
    modem.beginPacket();
    modem.write(lpp.getBuffer(), lpp.getSize());
    modem.endPacket(false);
}
```

Obrázek 16- Zdrojový kód - Složení a odeslání zprávy

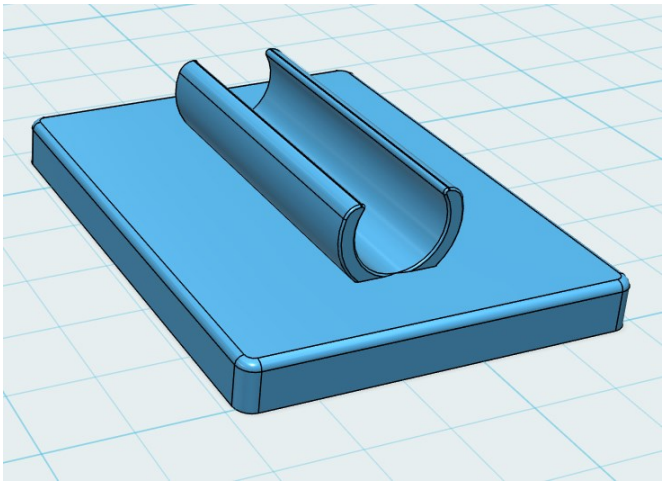
Pokud úvodní inicializace proběhne úspěšně, nastává proces, který se pravidelně opakuje v nastavených cyklech nebo při spojení či rozpojení kontaktů pro připojení sledovaného zařízení. Nejprve se rozsvítí integrovaná dioda, poté se zavolá metoda *zprava()*, která zjistí hodnoty z jednotlivých vstupů – stav kontaktu, stav baterie a teplotu. Tyto hodnoty jsou v komprimované podobě zabaleny do packetu a odeslány, poté je kontrolní dioda vypnuta a zařízení je uvedeno do režimu spánku.

Stav kontaktu je binární hodnota, která říká, zda je spínaný kontakt spojen nebo rozpojen. Používají se pro to dva piny Arduina, jedno je ve stavu GND a druhé v INPUT_PULLUP. Hodnota je negovaná díky použití ochranného pole. Napětí baterie je vypočítáno jako hodnota na jednom z definovaných pinů, která je násobena koeficientem pro získání napětí ve voltech. Pro vyčtení teploty z interního či externího čidla je použita knihovna OneWire, pomocí které jsou získány všechny dostupné teploty na 1-wire sběrnici připojené ke sledovanému pinu zařízení. Použita je vždy první hodnota, definovaná indexem 0. Tato hodnota odpovídá externímu čidlu v případě, že je připojeno, nebo internímu, pokud externí připojeno není.

Aby bylo docíleno co možná nejmenší velikosti odesílané zprávy, je pro její komprimaci používána knihovna CayenneLPP. Tento optimalizovaný způsob odesílání dat je hojně využíván v sítích s nízkou přenosovou rychlostí pro snížení doby přenosu a tím i snížení energetických nároků. Vyčtené hodnoty jsou přidány do paketu, podle typu hodnoty – analogový a digitální vstup, teplota společně s hodnotou kanálu. Stejné hodnoty kanálů jsou poté potřeba po rozbalení paketu na straně serveru.

4.3.5 Externí teplotní čidlo

Ne vždy je žádoucí měřit prostorovou teplotu interním čidlem uvnitř zařízení. Toto čidlo může být ovlivněno teplotou samotného zařízení v případě, že je umístěno v blízkosti zdroje tepla. Z těchto důvodů byla navržena možnost použití externího teplotního čidla. Jak bylo již zmíněno dříve, používají se čidla DS18B20, které fungují na 1-Wire sběrnici, a je možné je zapojit do série za sebou. Pro připojení je použit 3 pinový konektor. Aby bylo snazší umístit čidlo, které je ve válcovitém kovovém obalu, byl vymodelován držák opět s prolisem na vlepení magnetické pásky nebo pěnové oboustranné lepící pásky, jak ukazuje obrázek 17. Čidlo se do držáku pouze nacvakne.



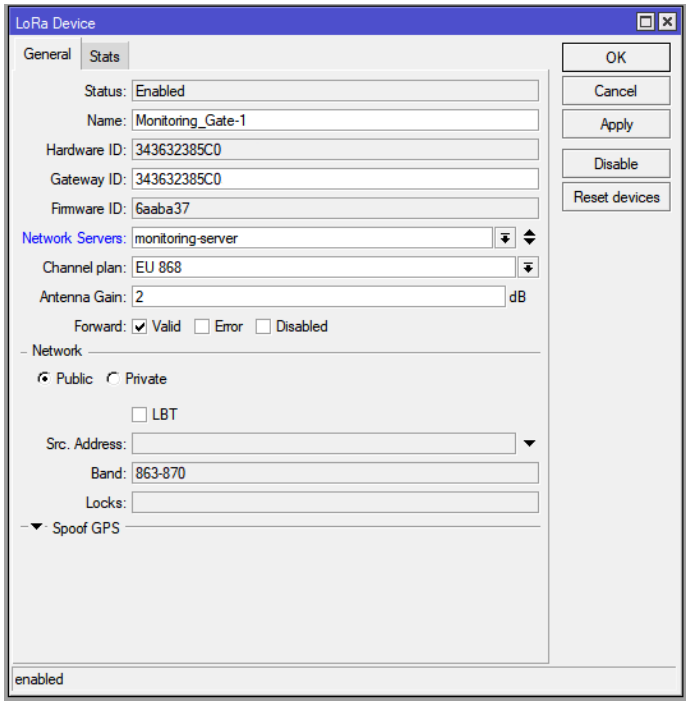
Obrázek 17- 3D model držáku čidla

4.4 LoRa Gateway

Pro přenos zpráv po internetu nebo lokální kabelové či bezdrátové síti je nutné použití bran, které přepošlou přijatou LoRa zprávu na stack server. Pro pokrytí požadované oblasti je možné použití více bran. Každá brána přepošle přijatou zprávu dále a nijak ji nezpracovává, ani se neověřuje, zda už zpráva byla předána jinou branou.

Pro příjem a předávání zpráv byla využita brána Mikrotik wAP LoRa8. Jak již bylo zmíněno v teoretickém úvodu, zařízení podporuje pasivní PoE, které bylo využito pro napájení a tím bylo zajištěno i kabelové připojení ke kabelové síti. WiFi adaptér nebude využit, proto byl v nastavení vypnut. V konfiguraci přes WinBox se v případě IoT zařizení objeví položka LoRa, která ovládá LR kartu, která je uvnitř zařízení připojena do mini PCIe portu.

Nastavení přenosu zpráv je přímočaré, v záložce *Servers* jsou předdefinované síťové servery The Things Network pro jednotlivé kontinenty, jednou z možností je definice cesty k novému serveru na základě IP adresy a Up a Down portu.



Obrázek 18- WinBox LoRa Devices

Jak ukazuje obrázek 18, v záložce *Devices* jsou k dispozici všechny připojené LR karty, v tomto zařízení je možné připojit jen jednu. V detailu lze vyčíst Gateway ID, jednoznačný identifikátor brány, který se zadává do nastavení síťového serveru. Dále je možné nastavit servery, na které se budou zprávy přeposílat. Je možné specifikovat jeden nebo i více serverů, na které se budou pakety přeposílat, je ovšem nutné zaručit na straně

těchto serverů jednoznačnou zpětnou komunikaci v případě OTAA ověření. Dále se nastavuje kanál, na kterém zařízení komunikují, výkon antény a typ přeposílaných paketů – je možné posílat pouze validní, chybné nebo i zakázané zprávy. Chybné zprávy jsou takové, u kterých je chybný CRC součet uvedený v přenášené zprávě oproti vypočítanému na straně brány. Toho je využito pro zabránění přenosu zpráv, které byly během přenosu poškozeny, na síťový server.

4.5 Síťový a aplikační server

Pakety odeslané monitorovacím zařízením a předané bránou zpracovává síťový server a dále je zpracovává aplikační server. Implementace obou serverů proběhla za využití programovacího jazyka Python a webového frameworku Django, používá se databáze PostgreSQL. Frontend využívá jQuery, Bootstrap a volně šiřitelnou šablonu Ruang Admin pod licencí MIT.

Síťový server má své nastavení v interních settings.py frameworku Django. Před spuštěním je nutné definovat IP adresu serveru a port, na kterém bude naslouchat. Další parametry jsou přednastaveny, ale mohou být uživatelsky měněny, jedná se o počet sekund, kdy se hlídá shodný MIC součet u zpráv pro vyloučení duplicity (defaultně 60 sekund), nastavení datových kanálů a počtu hodnot dle odesílaných dat ze zařízení a k těmto hodnotám i správné měrné jednotky. Pokud by u brány nebylo zakázáno přeposílání poškozených paketů, jsou v nastavení definovány limity pro maximální a minimální hodnoty, které mohou zprávy obsahovat. Tato kontrola probíhá při rozbalování hodnot ve zprávě.

Skript, který spouští síťový server, je vytvořen jako management command v Djangu a je spouštěn na pozadí serveru a loguje do nastaveného souboru. Po spuštění se nastartuje socketový server, který poslouchá na definované adrese a portu. Pokud přijde paket, z jeho hlavičky si zjistí verze, typ zprávy a EUI brány. Proběhne kontrola, zda je EUI v databázi a je povolené, pokud ne, je paket zahozen. V dalším kroku se ověří typ zprávy a zkontroluje se její délka. Pokud je zpráva typu *PULL_DATA* odešle se zpět bráně *PULL_ACK* pomocí definované metody jako potvrzení, že server s branou komunikuje. Tato komunikace probíhá několikrát do minuty. Když je typ zprávy *PUSH_DATA*, proběhne rozbalení paketu v metodě *zpracuj_data*, která dle definice LoRa rozdělí paket na jednotlivé části, které vrátí jako dict zpět serveru. Poté se zkontroluje, zda DevAddr, který označuje adresu zařízení, je v databázi a je povolen. Zpráva je uložena a zpět bráně je odeslán *PUSH_ACK* o přijetí

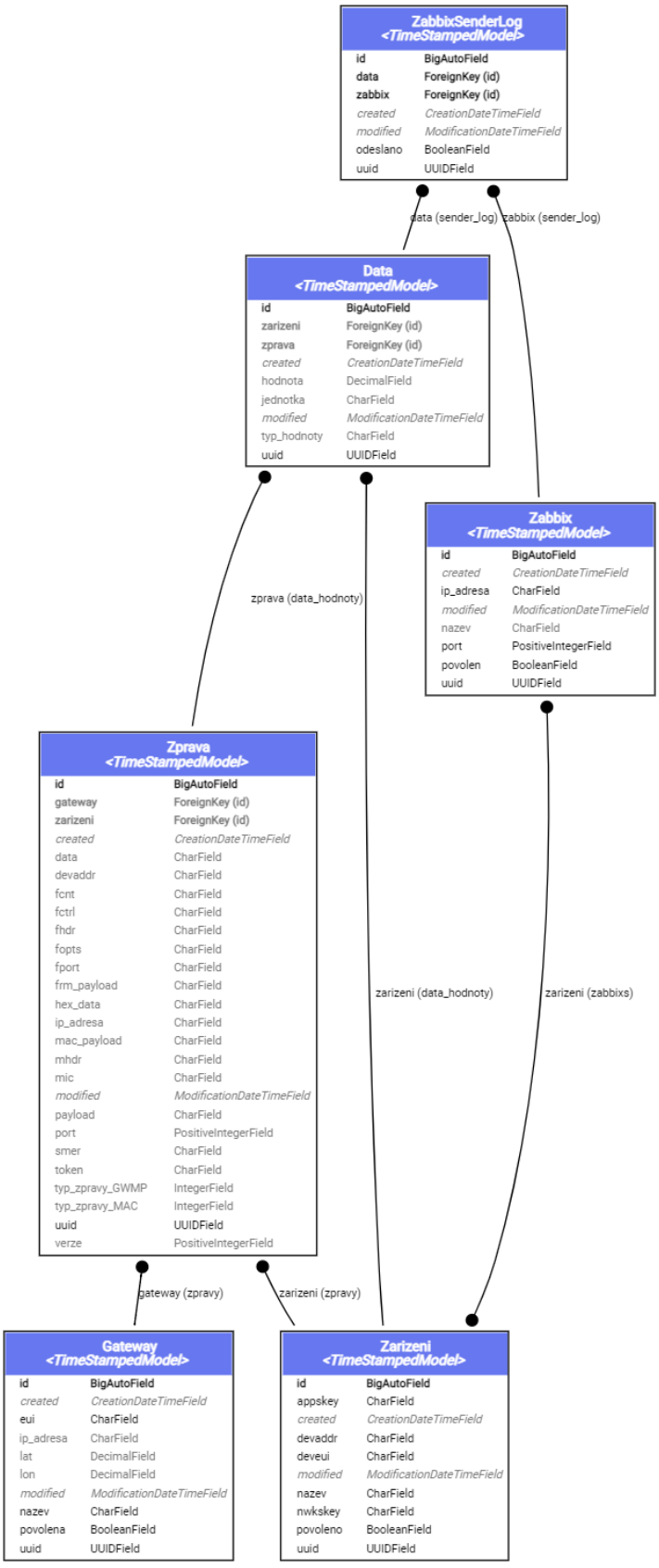
zprávy serverem. O další zpracování zprávy se již stará část, kterou lze označit jako aplikační.

Po uložení zprávy dojde pomocí post_save signálu k jejímu dalšímu zpracování. Pomocí klíčů, které jsou pro zařízení definovány, se dekretují data, převedou se z CayenneLPP formátu do čitelné podoby a zkontroluje se, zda jsou v definovaných limitech. Rozbalená data se ukládají do jedné databázové tabulky a nesou si informaci o typu dat, jejich hodnotě, jednotce a zařízení, ze kterého byla přijata.

Takto uložená data jsou opět pomocí post_save signálu zpracována. Pro každé zařízení lze definovat jeden nebo více zabbix serverů, kterým se mají hodnoty předat. Po uložení záznamu se naleznou servery, které mají data přijmout a za pomoci ZabbixSender jsou data přeposlána. Pro správné odesílání dat je nutné, aby server, na kterém je systém provozován, byl monitorovaný Zabbixem a měl nainstalovaný Zabbix Agent 2. Stav odesílání zpráv je ukládán do databáze a tento log je k dispozici ve webovém prostředí.

Ovládání aplikačního serveru probíhá přes webové GUI, které je dostupné po přihlášení uživatele. Django obsahuje vlastní autentizační backend, který byl využit a byl doplněn o grafické rozhraní, které slouží k přidávání a editaci účtů. Ovládací rozhraní nad modely, které popisuje obrázek 19, využívá AJAX a dynamické stahování obsahu, díky tomu je možné obnovovat jen části stránky. Editace a vytváření nových záznamů je řešeno přes formuláře, které jsou na backendu automaticky ošetřeny proti SQL Injection a dalším běžným formám útoků. Při zakládání nových zařízení jsou generovány položky DevAddr, NwkSKey a AppSKey, které jsou automaticky předvyplněny do formuláře. Tyto hodnoty jsou kontrolovány oproti databázi, aby byly vždy unikátní a nedocházelo k duplikacím. Taktéž jsou kontrolovány DevEUI v případě zařízení či EUI brány, které musí být na serveru unikátní.

Dashboard webu zobrazuje souhrnný přehled o bránách, zařízeních, zabbixech a zprávách, slouží jako rychlý přehled komunikace. Zde je možnost rozšíření do budoucna o grafy a zobrazování historických záznamů, tuto funkci aktuálně zastupuje primárně Zabbix, který sbírá přeposlaná data. Celý design je responsivní a je vytvořen s ohledem na přístupnost z mobilních zařízení.



Obrázek 19- Databázová struktura serveru

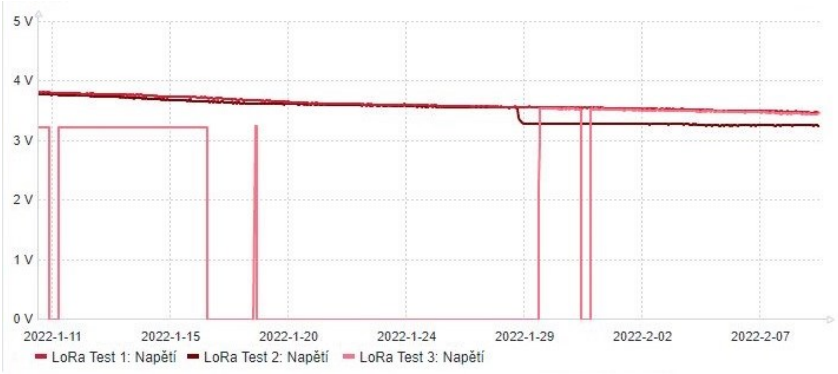
4.6 Zabbix

Další zpracování příchozích dat probíhá na straně Zabbixu. Byla vytvořena šablona pro nová zařízení, která definuje tři položky: napětí, stav kontaktu a teplotu. Číselné hodnoty jsou ukládány v desetinném formátu se správnou jednotkou. Hodnota kontaktu může nabývat hodnot 0 nebo 1 dle jeho stavu.

Pro každou z položek jsou definované tzv. triggers, které reagují na získanou hodnotu, mají nastavenou určitou závažnost, případně závislost na další hodnotě. Vyhodnocování probíhá na základě regulárního výrazu ve formě PCRE (Perl Compatible Regular Expressions). Pro stav kontaktu jsou nastaveny dva spouštěče. Jeden hlídá změnu stavu a adekvátně zareaguje v závislosti na tom, zda je změna do normálního nebo chybového stavu kontaktu a druhý, který monitoruje dobu od poslední získané zprávy a nesmí překročit nastavenou hodnotu periodického odesílání zpráv ze zařízení s rozdílem několika sekund. Pro monitoring teploty jsou definovány limitní hodnoty, při kterých spouštěč zareaguje. Jedná se o teploty v místnosti, které by měly být v mezích udávaných výrobcem mrazících boxů pro zajištění správné funkčnosti. Poslední dva spouštěče jsou pro stav baterie, kdy hodnota napětí pod 3 V znamená slabou baterii a pod 2,9 V velmi slabou baterii.

Na Triggery jsou navázány akce, jejichž aktivací jsou notifikováni uživatelé. V aktuálním testovacím nasazení na 3. lékařské fakultě Univerzity Karlovy jsou vytvořeny skupiny uživatelů dle monitorovaných zařízení a těm je doručován email a SMS zpráva v případě vzniku nebo vyřešení problému.

Zabbix zároveň uchovává historické hodnoty, počítá trendy hodnot a dokáže při správné konfiguraci předpovídat například výdrž baterie dle hodnot z minulých období. Všechny hodnoty je možné vykreslovat v grafech, jak ukazuje obrázek 20.



Obrázek 20- Zabbix graf s napětím třech zařízení

4.7 Komunikace

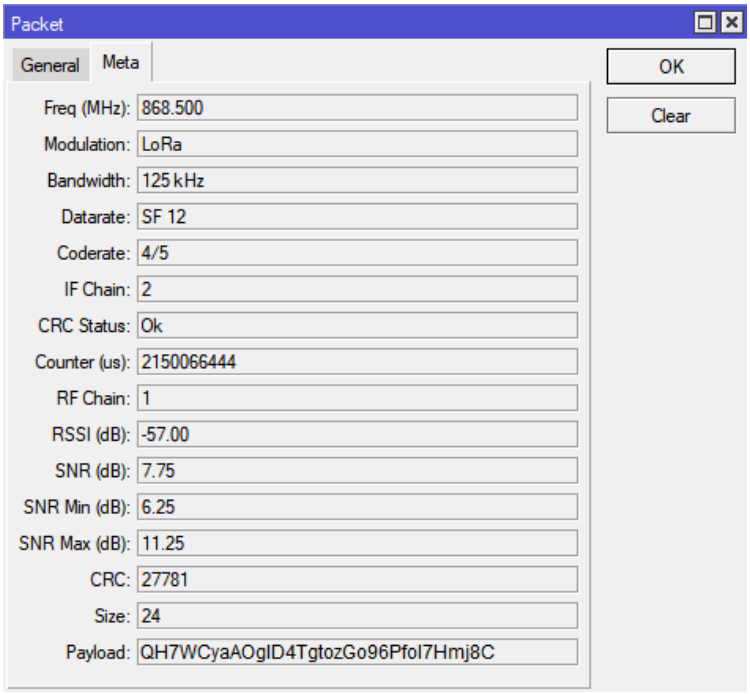
Komunikace za využití technologie LoRa pracuje s malými objemy dat přenášenými malou rychlostí. Protokol minimalizuje množství dat, která jsou bezdrátově odesílána mezi zařízením a branou, data od brány k serveru nesou další informace.

Vstupní hodnoty	[{digital_out_1: 1}, {temperature_2: 22.2}, {analog_in_3: 3.28}]
CayenneLPP HEX	010101026700DE03020148
Payload	QH7WCyaAOgID4TgtozGo96Pfol7Hmj8C
Paket	b'\x02\x01T\x004628\\\x00.\x00{"rxpk":[{"chan":2,"codr":"4/5", "data":"QH7WCyaAOgID4TgtozGo96Pfol7Hmj8C", "datr":"SF12BW125","freq":868.500000,"lsnr":7.750000, "modu":"LORA","rfch":1,"rssi":-57, "size":24,"stat":1, "time":"2022-02-09T17:00:50.929457Z", "tmst":348082156}]}'
Verze, token, typ zprávy	02015400 => 2, 21505, 0
Gateway EUI	343632385C002E00
Data HEX	407ED60B26803A0203E1382DA331A8F7A3DFA25EC79A3F02
FRM_PAYLOAD	E1382DA331A8F7A3DFA25E
Dekryptovaná data	010101026700DE03020148
Výstupní hodnoty	[{'channel': 1, 'name': 'Digital Output', 'value': 1}, {'channel': 2, 'name': 'Temperature Sensor', 'value': 22.2}, {'channel': 3, 'name': 'Analog Input', 'value': 3.28}]

Tabulka 10 - Struktura dat během procesu komunikace

Tabulka 10 znázorňuje přenos dat od získání hodnot až po opětovné rozbalení packetu, dešifrování a dekódování, kdy jsou na serveru stejná data, jako při jejich odeslání. Nejprve jsou vyčtena data o stavu kontaktu, teplotě a napětí baterie. Tyto tři hodnoty jsou překonvertovány do CayenneLPP formátu a hexadecimálního formátu. Poté jsou pomocí definovaných klíčů zašifrovány a jsou zapsány do LoRa paketu, který je následně odeslán na bránu. Brána zkontroluje MIC součet, aby ověřila, zda je paket v pořádku doručen, přidá k němu informaci o svém EUI, síle signálu, šířku pásma a další hodnoty, které jsou na obrázku 21 a odešle soubor na server.

Server po obdržení zprávy rozbalí první 4 bity paketu, které obsahují verzi, token a typ zprávy a dalších 8 bitů, které nesou jednoznačný identifikátor brány. Zbytek dat nese informace, která k paketu přidala brána a také původní zprávu odeslanou ze zařízení.



Obrázek 22 - Informace o paketu v rozhraní brány

Payload je v paketu celou dobu nesený v base64 podobě, proto se nejprve dekóduje do hexadecimální podoby a poté jsou z něj vyčteny informace o kontrolním součtu, adrese zařízení, FCtrl, FCnt a další hodnoty, které byly popsány v teoretické části práce. Jednou z částí je i FRM_Payload, který nese zašifrovaná data.

```
Message Type = Data
  PHYPayload = 407ED60B26803A0203E1382DA331A8F7A3DFA25EC79A3F02

( PHYPayload = MHDR[1] | MACPayload[..] | MIC[4] )
        MHDR = 40
  MACPayload = 7ED60B26803A0203E1382DA331A8F7A3DFA25E
         MIC = C79A3F02

( MACPayload = FHDR | FPort | FRMPayload )
        FHDR = 7ED60B26803A02
       FPort = 03
  FRMPayload = E1382DA331A8F7A3DFA25E

      ( FHDR = DevAddr[4] | FCtrl[1] | FCnt[2] | FOpts[0..15] )
    DevAddr = 260BD67E (Big Endian)
      FCtrl = 80
       FCnt = 023A (Big Endian)
      FOpts =
```

Obrázek 21- Rozpad fyzického payloadu

Tato data jsou pomocí klíčů dešifrována a tím jsou získána data v CayenneLPP formátu. Po převedení těchto hodnot do čitelné podoby jsou k dispozici stejné hodnoty, včetně kanálu a typu nesené informace, jako byly na začátku celého přenosu po jejich vyčtení ze senzorů.

4.8 Testování

Testování probíhalo v několika fázích. Nejprve se zkoušely jednotlivé části zařízení, poté bylo zkoušce podrobeno celé koncové zařízení a komunikace, síťový a aplikační server, komunikace se Zabbixem. Dále byla testována výdrž baterie, prostupnost signálu a na závěr nasazení do testovacího provozu na reálných zařízeních.

4.8.1 Testování vstupů zařízení

Zařízení obsahuje tři vstupy, u kterých byla ověřována správná funkčnost. Dvě totožná zařízení byla umístěna vedle sebe, obě byla připojena ke stejnému laboratornímu zdroji s konstantním proudem a napětím. V 15 minutových intervalech bylo prováděno měření napětí baterie, která byla suplována externím zdrojem napájení, a funkčnost teplotních čidel jednotlivých zařízení. Všechny hodnoty byly logovány přes sériový port a porovnávány mezi sebou. Odchylky byly minimální a odpovídaly technickou specifikací danému výkyvu ± 0,5°C. V případě teplotních čidel byla otestována jak interní čidla na desce tištěného spoje, tak i externí zapojená do příslušného portu. Obrázek 23 ukazuje teploty z testovacích měření během 24 hodin v místnosti s proměnlivou teplotou.

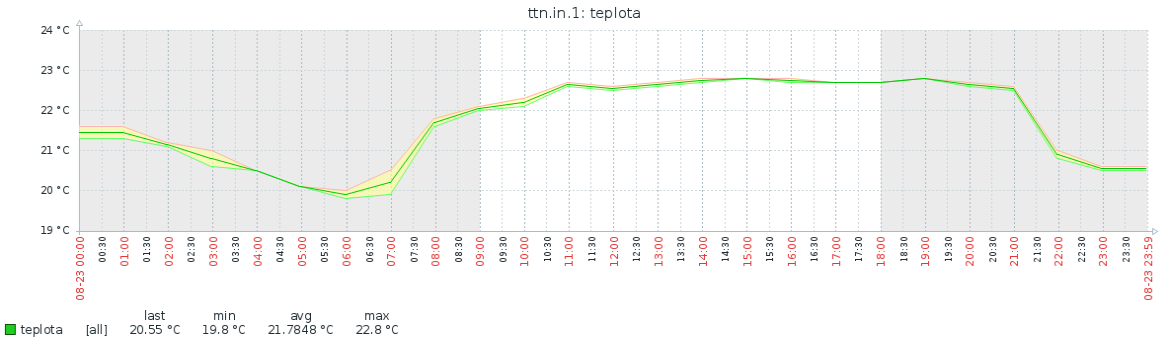
Obrázek 23 - Rozdíly teplot mezi zařízeními

Testování vstupního kontaktu pro připojení externího zařízení probíhalo ve stejném nastavení jako u měření teploty a kontaktu. Obě zařízení měla připojené relé, kterému byl v 15 minutových intervalech měněn stav sepnuto/rozepnuto. Obě zařízení reagovala s drobnými odchylkami v řádech milisekund, které mohly být způsobené komunikací po sériovém portu, nicméně nedošlo k chybám.

4.8.2 Testování zařízení a komunikace

Komunikace zařízení je klíčovou složkou celého řetězce monitorování. Pro eliminaci problémů na straně síťového serveru bylo využito serverů The Things Network, kde bylo přidáno nové zařízení a na bráně byla na tento server nasměrována komunikace. Problémy s branou samotnou nebyly předpokládány z toho důvodu, že se jedná o komerční zařízení a případné potíže by se musely komunikovat s výrobcem. Nicméně se během testování ani provozu chyby na straně brány neobjevily.

Monitorovací zařízení bylo naprogramováno tak, aby se každé dvě hodiny probouzelo z režimu spánku a odeslalo data. Pokud zařízení spí, není připojeno k sériové lince a probuzení je natolik krátké, že nelze data logovat přes sériovou linku. Během sedmi dní bylo odesláno dostatečné množství zpráv na to, aby se dal test považovat za úspěšný. Na straně The Things Network byla data přeposílána přes MQTT a na straně klienta sbírána. Všechny zprávy byly doručeny v nepoškozeném stavu a s časovou prodlevou v rámci vteřin od předpokládaného času odeslání. Přijatá data byla ukládána do Zabbixu, obrázek 24 ukazuje část zmíněného testovacího období na hodnotách teploty.



Obrázek 24 - Záznam teplot v rámci testování

4.8.3 Testování síťového a aplikačního serveru

Aby bylo možné ověřit funkčnost síťového serveru a správného zpracování dat, byl zvolen nestandardní přístup, který by v případě nasazení v provozu neměl být používán. Data byla odesílána na dva servery, jeden byl testovaný síťový server a druhý server byl The Things Network. Zařízení bylo nastaveno na ověřování ABP a na obou serverech byly nastaveny stejné klíče a DevAddr. Pro otestování kontroly duplicity zpráv bylo zvoleno použití dvou bran s totožným nastavením, čímž docházelo k duplikaci příchozích zpráv na server. Zároveň byla na branách povolena možnost přeposílání paketů, s nekonzistentním CRC součtem, pro ověření části kódu, která verifkuje správnost příchozích dat.

S tímto nastavením bylo použito zařízení, odesílající opět v 15 minutových intervalech zprávy, které byly přijímány dvěma servery. The Things Network opět předával data přes MQTT na klienta, který data sbíral a zapisoval do souboru. Po ukončení tohoto testu, který běžel 3 dny, byla porovnána přijatá data na obou serverech.

Výsledné hodnoty se shodovaly, nedocházelo k žádným anomáliím. Porovnávat časy přijetí zprávy v tomto řešení nebylo primárním cílem, protože každý ze serverů zpracovával a přeposílal zprávy jiným způsobem, ale ve výsledných datech docházelo k rozdílu mezi přijetím a zpracováním zpráv v řádech milisekund.

Přijaté zprávy s chybnými hodnotami byly pouze logovány a nebyly dále zpracovány. Při kontrole dat bylo ověřeno, zda všechny přijaté a zpracované hodnoty odpovídají nastaveným limitům. Zpracovaná data byla pouze ukládána do databáze a nebyla dále přeposílána, aby mohlo být později otestováno jejich zpracování Zabbixem.

4.8.4 Testování komunikace se Zabbixem

Testování odesílání a zpracování Zabbixem byla další část ověření funkčnosti systému. Data, která byla posbíraná v rámci testu aplikačního serveru, byla připravena v databázi pro odeslání na Zabbix. Pro tyto účely byl použit jeden standardně využívaný Zabbix server v rámci organizace a druhý, který byl nově nainstalován do virtuálního prostředí jen pro potřeby testování. Použité verze byly 4.0.4 a 5.0.14, aby byla otestována kompatibilita vytvořené šablony zařízení mezi verzemi.

Poté, co byly oba servery nakonfigurovány a přidány do Zabbix Agent běžícím na aplikačním serveru, byl spuštěn skript, který prošel všechna data, která byla přijata v předchozích třech dnech a nebyla dále odeslána a postupně je všechna odeslal do obou Zabbix serverů.

Pokud jsou hodnoty odeslány, je do databáze na straně aplikace uložen záznam se stavem odeslání. Toto ukládání a logování do souboru je součástí běžného provozu serveru, stejně jako notifikace o neúspěšném odeslání. Během dávkového odeslání hodnot byly všechny hodnoty úspěšně odeslány a servery přijaty.

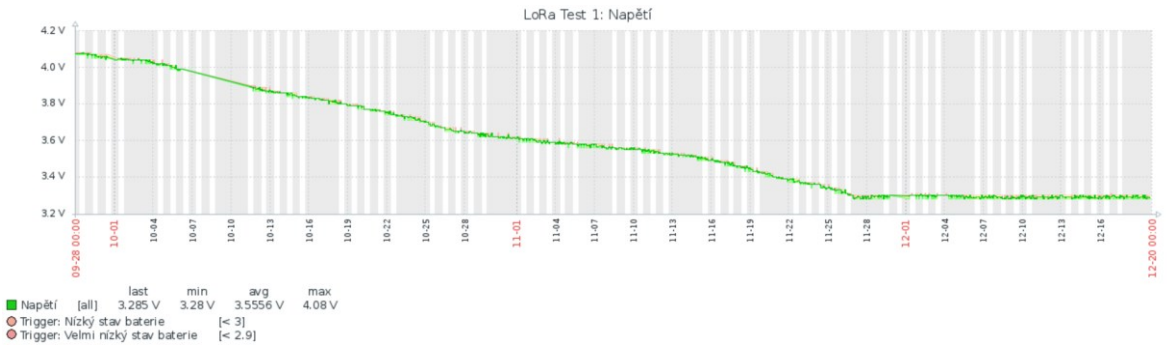
4.8.5 Testování výdrže baterií

Testování výdrže baterie probíhalo od plného nabití baterie Li-Ion baterie Samsung INR18650-29E s nominální kapacitou 2850 mAh. Zařízení byla umístěna do místnosti s teplotou kolem 16° C, což je srovnatelné s teplotou, ve které bude zařízení běžně provozováno.

Program v monitorovacím zařízení byl nastaven tak, aby každou hodinu odeslal zprávu, mezitím bylo zařízení uspané. Tím je simulován běžný provoz, doba mezi odeslanými zprávami by mohla být i vyšší a tím by se snížil počet odeslaných zpráv a celková spotřeba elektrické energie.

Zařízení během spánku spotřebovává 850 uA, probuzení trvá 2-3 sekundy, kdy vzroste odběr proudu ve špičce na 1500 uA. Pokud by bylo zařízení pouze uspané a neprobouzelo se, je předpokládaná výdrž na baterii zhruba 140 dní, probouzení výdrž částečně sníží.

Odesílaná data byla opět ukládána do Zabbixu a vzniknul z nich graf, který je na obrázku 25. Jedná se o zhruba tříměsíční průběh monitorovaných hodnot, kdy dochází k lineárnímu poklesu napětí až k hodnotě 3,2 V, kde zůstává po delší dobu křivka ustálená.



Obrázek 25 - Naměřené hodnoty napětí

Z měření vyplynulo, že výrobcem udávaná minimální spotřeba 104 uA je těžko dosažitelná při provozu na baterie typu 18650 bez toho, aby se fyzicky zasáhlo do hardwaru Arduina. Vyzkoušená doba výdrže od plného nabití do vypnutí je u testované baterie 110 - 115 dní. Měření probíhala na dvou zařízeních současně, jedno ze zařízení se

vypnulo po 112 dnech, druhé po 116 dnech. Pro baterii s kapacitou 3500 mAh prozatím nebyla z časových důvodů nasbírána potřebná data pro celý průběh testu, ale předpokládá se cca 20% nárůst díky navýšení kapacity baterie, tzn. zhruba 140 dní bateriového provozu.

4.8.6 Testování signálu

Prostupnost signálu byla, vzhledem k frekvenčnímu pásmu pod 1 GHz, předpokládaná na takové úrovni, že by na pokrytí celé budovy se 6 patry měla stačit jedna gateway. V jednotlivých místnostech a patrech bylo zařízení postupně testováno a byla logována intenzita signálu na bráně.

Budova má tvar písmene U a 6 nadzemních pater, brána je umístěna ve východním křídle 3. nadzemního patra a monitorovaná zařízení jsou rozprostřena po celé budově. Tabulka 11 zobrazuje intenzitu signálu po patrech ve východní a západní části.
Brána byla vybavena externí anténou se ziskem 6,5 dBi a horizontální vyzařovacím úhlem 360°. Zobrazené hodnoty jsou hodnoty ukazatele RSSI (Received Signal Strength Indication), který ukazuje hodnotu síly signálu. Udávané hodnoty jsou v záporných číslech. Čím bližší hodnota k 0, tím vyšší síla přijatého signálu. Při hodnotách pod -85 dBi již dochází k problémům s komunikací, naopak pod -35 dBi je signál příliš silný a je vhodné snížit výkon antény.

Patro	Východní křídlo	Západní křídlo
1NP	-62 dBi	-73 dBi
2NP	-54 dBi	-61 dBi
3NP	-47 dBi	-52 dBi
4NP	-53 dBi	-57 dBi
5NP	-59 dBi	-66 dBi
6NP	-68 dBi	-80 dBi

Tabulka 11 - Výsledky měření intenzity signálu

Po otestování síly signálu došlo k potvrzení, že na pokrytí celé budovy stačí jedna brána. Nicméně v krajních místech je signál již slabý a komunikace by nemusela probíhat korektně. Pro správné pokrytí bude nutné použít ještě jednu bránu a upravit rozmístění a polaritu antén tak, aby došlo k rovnoměrnému pokrytí celé budovy.

4.8.7 Testování v provozu

Po všech testech byla zařízení umístěna na mrazící boxy a zapojena paralelně s nahrazovaným řešením. V tomto provozu byla zařízení provozována několik týdnů, během kterých se objevily jednotky případů, kdy zařízení muselo reagovat na vzniklou problematickou situaci monitorovaného zařízení. Komunikace probíhala korektně a nedocházelo k žádným neočekávaným problémům.

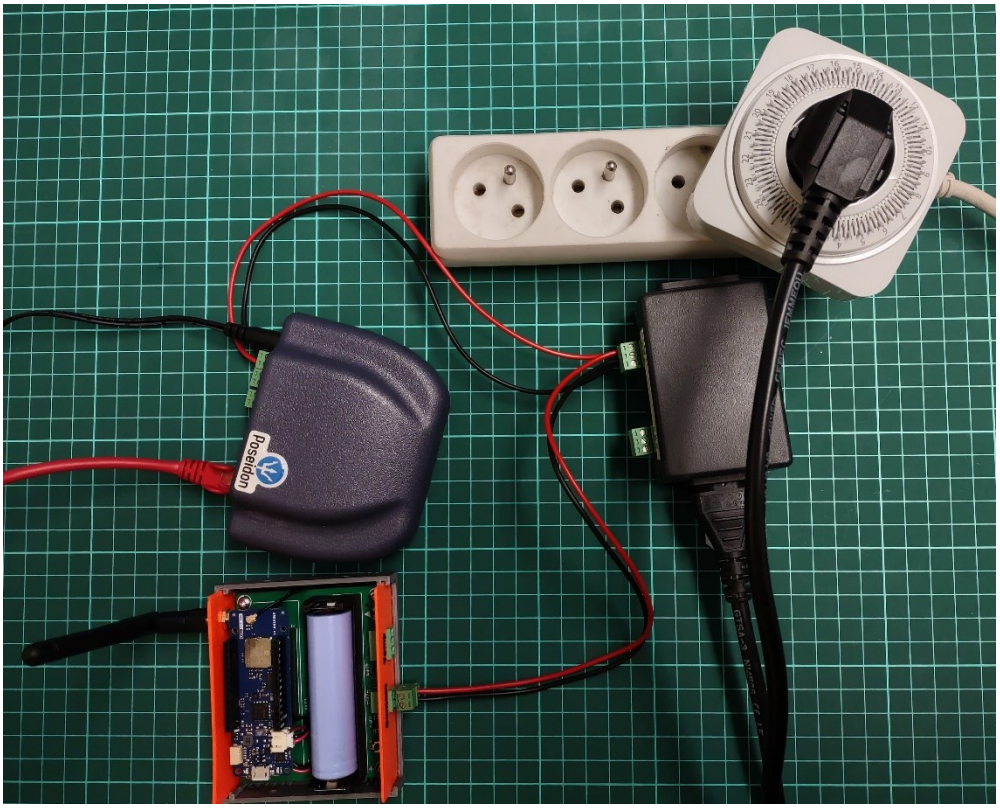
Jedno zařízení bylo připojeno k boxu samostatně, využívá externí teplotní čidlo, pouze pro otestování funkčnosti, v tomto nasazení by bylo možné využít integrované čidlo. Obrázek 26 ukazuje připojení zařízení a jeho umístění. Je využito magnetických pásků na spodní straně zařízení, což umožňuje jeho snadné usazení poblíž spínaného kontaktu a tím odpadají problémy s množstvím dlouhých kabelů.

Obrázek 26 - Nasazení zařízení v provozu

4.8.8 Porovnání s nahrazovaným řešením

Současné řešení monitorování je řešeno, jak bylo již zmíněno v analýze požadavků, zařízením Poseidon 3266 od společnosti HW Group. Oproti navrhovanému řešení se může Poseidon spolehnout na kabelovou síť, která teoreticky bude téměř vždy spolehlivější než síť bezdrátová. Toto je ovšem podmíněno přítomností lokální sítě v místě instalace zařízení, což bývá občas problematické a také neumožňuje větší manipulace s monitorovaným zařízením.

Navrhované řešení je možné provozovat i v čistě bateriovém režimu či v kombinaci s napájením z elektrické sítě. Výhodou navrhovaného řešení je i rychlost instalace, kdy kabelové propojení dvoulinkového vodiče mezi mrazícím boxem a monitorovacím zařízením je jediným nutným úkonem. Zařízení magneticky drží na těle monitorovaného zařízení a teplotní čidlo je jeho součástí. Zařízení Poseidon se musí umístit na vodorovnou plochu nebo jinak upevnit, připojit k němu mimo komunikace s boxem i ethernetový kabel, externí teplotní čidlo a napájení, které lze řešit buďto adaptérem do zásuvky (v případě výpadku proudu se ale zařízení vypne) nebo přes PoE napájení ze switchů a PoE Extraktor.



Obrázek 27- Porovnání rychlosti a kvality obou řešení

V rámci testování byla provedena i zkouška obou zařízení, jak ukazuje obrázek 27. Opět bylo využito spínacích hodin s 15-minutovým intervalem a relé pro simulaci vstupu mrazícího boxu, které mění svou hodnotu. Data obou zařízení byla odesílána do stejného Zabbix serveru, Poseidon pomocí SNMP protokolu, nové řešení výše popsanou cestou. Dle očekávání byla rychlost kabelového zařízení vyšší díky přímému zapojení do lokální sítě. Bezdrátový přenos a zpracování dat trvalo déle. Testování znovu probíhalo 24 hodin a bylo odesláno více než 100 zpráv každým zařízením.

Zpoždění v přenosu mezi zařízeními v neprospěch nově navrhovaného řešení je v řádech sekund, nejvyšší rozdíl časových značek byl zhruba 5 vteřin. Pokud bychom do celého procesu započítali ještě čas odeslání a doručení notifikační zprávy, tak vteřinové rozdíly nejsou překážkou. Pro uživatele je podstatné obdržet informaci o vzniklém problému na monitorovaném zařízení, rozdíly v řádech sekund nejsou podstatné, zařízení se do kritického stavu dostane až po více než hodině, záleží na typech vzorků uvnitř boxu a na prostředí, ve kterém jsou umístěny.

5 Výsledky a diskuze

Výsledkem této práce je kompletní dohledový systém nad zařízeními se spínaným kontaktem, nejčastěji hlubokomrazícími boxy. Celé řešení je navrhnuto tak, aby bylo s co nejmenšími problémy modifikovatelné pro jiné případy užití nebo pro změnu zařízení či systému napájení. Náklady, které byly vynaloženy, obsahují pouze částky na nákup jednotlivých součástí. Vývoj probíhal ve volně dostupném vývojovém prostředí a softwarová část je provozována ve virtuálním počítači uvnitř serverového řešení organizace.

5.1 Finanční náklady

Tabulka 12 ukazuje vynaložené náklady na jednotlivé části zařízení a bránu. V ceně není započítáno poštovné za doručení zboží a drobný spojovací a instalační materiál ani cena elektrické energie, která byla vynaložena na 3D tisk krabičky. Cena na výrobu tištěného spoje odpovídá desetině částky za 10 kusů, které byly objednány a vyrobeny pro účely testování.

Celková cena koncového zařízení bez brány je 1541 Kč s DPH. Oproti obdobnému zařízení LPN DO od společnosti Comtac, které je nabízeno za cenu 6380 Kč s DPH, jsou náklady čtvrtinové.

Pořizovací cena brány je významnou položkou, ovšem jedna gateway obslouží desítky či stovky zařízení. Taktéž pro provoz přístrojů od jiných výrobců by bylo nutné pořídit buďto univerzální bránu jako při řešení této práce nebo bránu specificky dodávanou výrobci.

Název	Cena včetně DPH
Gateway MikroTik wAP LR8	3 229Kč
Arduino MKR WAN 1310	1 058Kč
Výroba a osazení PCB	94Kč
Terminální konektory	70Kč
Pouzdro na baterii	30Kč
Filament (15 m)	20Kč
Čidlo DS18B20	70Kč
Baterie 18650	199Kč
Celkem	**4 770Kč**

Tabulka 12- Finanční náklady na zařízení

5.2 Možnosti pro změny

Jednou z možností změny, která by příznivě ovlivnila celé zařízení, je výměna hlavní komponenty, a to Arduina MKR WAN 1310 za energeticky méně náročné zařízení. Jako vhodná náhrada se jeví zařízení založená na ESP32. Může se jednat o kompletní řešení s integrovaným LoRa modulem, například LilyGO TTGO LoRa32, nebo o samotné zařízení založené na ESP32 a k němu přidaný LoRa modul. Obě řešení se pohybují na polovině ceny Arduina, které je aktuálně v sestavě použité. Spotřeba zařízení od LilyGO je v produktovém listu výrobce udávaná na hodnotě 0,2 uA při režimu spánku, respektive 1,5 uA při probuzení. Touto změnou by došlo k výraznému prodloužení provozu na baterie. (LilyGO TTGO LoRa32, 2021)

Prostor pro změny je i v možnosti externího napájení. Současné zařízení využívá microUSB port přímo na Arduinu. Jedna z možností je vyvedení tohoto portu na čelo krabičky k ostatním portům. Další řešení je využití prostoru na PCB desce pro kulatý napájecí konektor, regulátor napětí a napájení Arduina přímo přes GPIO piny za použití externího zdroje.

Další možnou změnou je využití více pinů pro připojení jiných externích zařízení, ať už by se jednalo o monitorované zařízení nebo senzory prostředí. Možnou náhradou prostého teplotního čidla by bylo využití kombinovaného senzoru teploty a vlhkosti nebo senzoru otevřených dveří mrazícího boxu pro jejich kontrolu.

5.3 Problémy při řešení

Během vývoje a testování zařízení se objevilo několik překážek, které zkomplikovaly celý proces. Jedním z největších úskalí a částečně i důvodem pro výběr právě použitého Arduina byl nedostatek čipů na trhu a problémy s dodávkami do Evropy. V době návrhu řešení byla omezená nabídka zařízení a dodavatelé nebyli schopni garantovat doby dodání. Stejný problém byl řešen i v případě brány Mikrotik wAP LR8, nedostatek těchto zařízení přetrvává i na začátku roku 2022. Během tvorby softwaru byla část kontrolující duplikaci zpráv z více bran psána bez otestování. Až po několika měsících se podařilo zapůjčit na otestování další bránu a provést ověření napsaného kódu.

5.4 Další využití

Vytvořené dohledové řešení je založeno na volně dostupných nástrojích a součástech, to umožňuje jeho modifikaci pro jiné případy užití. Do vstupů zařízení se dají připojit jiné externí senzory a monitorovat jiné přístroje či měřit jiné veličiny.

Díky této modularitě je možné nasazení i ve zcela odlišných oblastech, například pro jednoduché zabezpečovací zařízení, kdy je do zařízení připojen magnetický kontakt, který je připojen ke dveřím či oknu hlídaného objektu, jak ukazuje obrázek 28. Tohoto lze docílit pouze výměnou snímače, jinak může vše zůstat v původním stavu. Stejným způsobem lze připojit záplavové čidlo, které reaguje na přítomnost vody spojením kontaktů. Tím lze dohlížet jak na výskyt vody například na podlaze v domácnosti, tak i ve venkovním prostředí pro informaci o stavu hladiny například v nádrži na dešťovou vodu nebo v objektech blízko vodních toků.

Další možností využití je pro domácí meteorologickou stanici za použití čidla teploty a vlhkosti či externího snímače rychlosti větru. Těmito modifikacemi lze obsáhnout širokou škálu monitorovaných hodnot za využití minimálních nákladů na vývoj.

5.5 Předpokládané využití

Po testovací fázi v provozu by mělo dojít v následujících měsících k nasazení na všechna zařízení v rámci jedné z budov 3. lékařské fakulty Univerzity Karlovy. Pro potřeby dohledu bude potřeba třiceti zařízení a dvou bran. Při ceně výroby jednoho kusu monitorovacího zařízení 1541 Kč s DPH dojde k úspoře oproti komerčnímu zařízení při plném nasazení téměř 150 tis. Kč.

6 Závěr

Cílem diplomové práce bylo vytvoření funkčního systému dohledu nad hlubokomrazícími boxy s využitím Internetu věcí.

Teoretická část se zabývala pojmem internet věcí, architekturou těchto sítí a zabezpečením. Dále jsou popsány nejčastější technologie bezdrátového připojení k síti s ohledem na IoT jako jsou například Sigfox, LoRaWAN, NB-IoT. Důležitou součástí přípravy na samotné praktické řešení byl popis komunikačního protokolu LoRaWAN, principu fungování této sítě, šifrování a dešifrování spojení, způsoby vytváření paketů, navazování komunikace. Teoretická část se rovněž věnovala dostupným zařízením na trhu, popisu jejich vlastností, předností či nevýhod. Mezi popisovanými zařízeními jsou také technické informace o použité desce Arduino MKR WAN 1310 a bráně zajišťující přenos dat.

Praktické části předchází analýza požadavků na vyvíjený systém, která vychází z desetiletých zkušeností s provozem nahrazovaného řešení. Praktická část popisuje vývoj jednotlivých součástí zařízení, konkrétně návrhu tištěného spoje, systému bateriového napájení, krabičky, do které je zařízení umístěno, možnosti připojení externího čidla. Software pro zařízení byl vytvořen s ohledem na snadnou úpravu dle požadavků uživatelů. Pro zajištění komunikace byl vyvinut vlastní síťový aplikační server postavený na programovacím jazyce Python a jeho frameworku Django pro snadné ovládání přes webové rozhraní. Zpracování přijatých dat probíhalo na aplikačním serveru. Tyto informace byly poté předávány přímo do jednoho nebo více Zabbix serverů, které se staraly o notifikace uživatelů o nastalých problémech. Taktéž zajišťovaly jejich archivaci, vizualizaci a další zpracování dle šablony, která byla vytvořena je součástí této práce.

Po návrhu vytvoření celého systému bylo provedeno několikadenní otestování jednotlivých částí závislostí. Testovalo se jak samotné zařízení či jeho součásti, tak i komunikace se síťovým serverem či volně dostupným serverem The Things Network. Síťový a aplikační server prošel samostatným testem s použitím více bran a tím i ověřením kontroly duplikovaných přijatých zpráv. Komunikace zpracování informací v Zabbixu byla taktéž otestována i v porovnání se současně použitým řešením. Během testování byla ověřena prostupnost signálu budovou, ve které budou zařízení provozována. Díky tomuto testu bylo zjištěno, že na pokrytí všech šesti pater budovy není jedna brána dostatečná.

Po těchto zkouškách byl celý systém zapojen do reálného provozu v rámci 3. lékařské fakulty Univerzity Karlovy. V tomto pilotním projektu byla nejprve nasazena tři zařízení v paralelním zapojení se současným monitorovacím systémem, jedno zařízení zcela samostatně. Při provozu se sleduje, zda se vyskytují chyby či anomálie, které by znemožňovaly použití v reálném nasazení. V případě, že se řešení osvědčí v praxi, bude postupně nasazováno na další přístroje, které musí být monitorovány.

V případě plného přechodu na navrhnutá zařízení jsou celkem ušetřené náklady poměrně výrazné, neboť cenový rozdíl oproti komerčně dodávanému řešení je téměř pět tisíc korun na jedno zařízení. Další výhodou vytvořeného systému je jeho snadná konfigurace, možnost nasazení v jiných budovách či pro jiné způsoby užití.

Přínosem této práce je komparace dostupných bezdrátových technologií pro Internet věcí či detailní analýza LoRaWAN protokolu. Monitorovací systém v rámci organizace zajišťuje dohled nad zařízeními a prostředím, ve kterém jsou provozována. V případě poruchy či problému s chladícími boxy by bez zásahu uživatelů došlo v mnoha případech k nevyčíslitelným ztrátám, které by vznikly poškozením uložených výzkumných vzorků.

7 Seznam použitých zdrojů

Arduino MKR WAN 1310 [online], 2022. [cit. 2022-01-16]. Dostupné z: https://store.arduino.cc/products/arduino-mkr-wan-1310

Arduino Products, 2022. In: Arduino Products [online]. [cit. 2022-01-16]. Dostupné z: https://www.arduino.cc/en/Main/Products

BANKOV, Dmitry, Evgeny KHOROV a Andrey LYAKHOV, 2016. On the Limits of LoRaWAN Channel Access. In: 2016 International Conference on Engineering and Telecommunication (EnT) [online]. IEEE, s. 10-14 [cit. 2021-08-11]. ISBN 978-1-5090-4553-2. Dostupné z: doi:10.1109/EnT.2016.011

Bluetooth® Wireless Technology [online]. In: . [cit. 2021-07-13]. Dostupné z: https://www.bluetooth.com/learn-about-bluetooth/tech-overview/

BUYYA, Rajkumar, 2016. Internet of Things: Principles and Paradigms. Elsevier. ISBN 978-0-12-805395-9.

Comtac LPN AI Datasheet [online], 2017. In: . [cit. 2022-02-16]. Dostupné z: https://www.eps-energy.com/wp-content/uploads/2019/11/E1347-technical-datasheet-LPN-AI-EN-V1.00.pdf

Connectivity Standards Alliance [online]. [cit. 2021-07-14]. Dostupné z: https://zigbeealliance.org/

DUBRAWSKY, Ido, 2010. Eleventh hour security+: exam SYO-201 study guide. Burlington, MA: Syngress. ISBN 1597494275.

Ela innovation BLUE PUCK AI Product sheet [online]. In: . [cit. 2022-02-16]. Dostupné z: https://elainnovation.com/wp-content/uploads/2021/01/FP-Blue-PUCK-AI-07C-EN.pdf

EVANS, Dave, 2011. The Internet of Things: How the Next Evolution of the Internet Is Changing Everything [online]. Cisco IBSG [cit. 2021-07-11]. Dostupné z: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

GACOVSKI, Zoran, 2019. Internet of Things [online]. Arcler Press [cit. 2022-02-20]. ISBN 978-1-77361-624-7.

GRATTON, Dean A., 2007. Developing practical wireless applications. Boston: Elsevier Digital Press. ISBN 978-1-55558-310-1.

GUPTA, Naresh C., 2013. Inside Bluetooth Low Energy. Boston: Artech House. Artech House mobile communications series. ISBN 978-1-60807-579-9.

HASSAN, Qusay, 2018. Internet of Things A to Z: Technologies and Applications. ISBN 978-1-111-945674-2.

HC-DIG devices installation manual [online], 2018. In: . [cit. 2022-02-16]. Dostupné z: https://uploads-ssl.webflow.com/5d9727a06f21d74ee7d2fe2a/5db9978e4ca3576fd42297a2_installation%20manual%20hc-dig_v1.0.pdf

HW Group NB devices [online], 2020. In: . [cit. 2022-02-16]. Dostupné z: https://www.hw-group.com/files/download/man/version/nb-man_1-0-0_cs.pdf

HW Group SD devices [online], 2019. In: . [cit. 2022-02-16]. Dostupné z: https://www.hw-group.com/files/download/man/version/sd-man_19-8-7_cs.pdf

IEEE Standard for Information technology, 2017. IEEE Std 802.11ah-2016. 1-594. Dostupné z: doi:10.1109/IEEESTD.2017.7920364

INFSO D.4 NETWORKED ENTERPRISE & RFID INFSO G.2 MICRO & NANOSYSTEMS, 2008. Internet of Things in 2020: A Roadmap for the Future [online]. [cit. 2021-07-11]. Dostupné z: http://www.smart-systems-integration.org/public/documents/publications/Internet-of-Things_in_2020_EC-EPoSS_Workshop_Report_2008_v3.pdf

Installation Guide LoRaWAN range of devices [online], 2021. In: . [cit. 2022-02-16]. Dostupné z: https://enless-wireless.com/wp-content/uploads/2021/11/Installation-Guide-LoRa-LoRaWAN-range-of-devices_j.pdf

Introducing 0G network [online], 2021. [cit. 2021-07-23]. Dostupné z: https://www.sigfox.com/sites/default/files/og-guide/Sigfox%20-%20Introducing%200G_Apr2021.pdf

LASSE LUETH, Knud, 2020. State of the IoT 2020: 12 billion IoT connections, surpassing non-IoT for the first time [online]. [cit. 2021-07-11]. Dostupné z: https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/

LilyGO TTGO LoRa32 [online], 2021. [cit. 2022-02-16]. Dostupné z: https://www.laskakit.cz/lilygo-ttgo-lora32-t3_v1-6-433mhz-0-96--sma-wifi-modul/#relatedFiles

LoRa Alliance [online], 2021. [cit. 2021-08-11]. Dostupné z: https://lora-alliance.org/

LoRaWAN™ Specification v1.1 [online], 2017. Beaverton, USA [cit. 2021-12-19]. Dostupné z: https://lora-alliance.org/wp-content/uploads/2020/11/lorawantm_specification_-v1.1.pdf

MALÝ, Martin, 2017. Hradla, volty, jednočipy: úvod do bastlení. 1. vydání. Praha: CZ.NIC, z.s.p.o. CZ.NIC. ISBN 978-80-88168-23-2.

Mcf 88 MCF-LW06420 Datasheet [online], 2020. In: . [cit. 2022-02-16]. Dostupné z: https://www.mcf88.it/wp-content/uploads/2020/07/MCF-LW06420.pdf

Mikrotik Company Profile [online], 2022. [cit. 2022-01-22]. Dostupné z: https://www.mikrotik.com/download/share/mt_profile.pdf

Mikrotik IoT products [online], 2020. [cit. 2022-01-22]. Dostupné z: https://i.mt.lv/cdn/product_files/LoRa_200558.pdf

PUŽMANOVÁ, Rita, 2004. Širokopásmový Internet: přístupové a domácí sítě. Vyd. 1. Brno: Computer Press. ISBN 80-251-0139-8.

REHMAN, Aqeel-ur, Sadiq Ur REHMAN, Malaika HASAN a S. HASAN, 2016. Security and privacy issues in IoT. International Journal of Communication Networks and Information Security (IJCNIS). 2016, 147-157.

SALAM, Abdul, 2020. Internet of Things for Sustainable Community Development: Wireless Communications, Sensing, and Systems. 1st. Springer. ISBN 978-3030352905.

SENDRA, Sandra, Laura GARCÍA, Jaime LLORET, Ignacio BOSCH a Roberto VEGA-RODRÍGUEZ, 2020. LoRaWAN Network for Fire Monitoring in Rural Environments. Electronics. 9(3). ISSN 2079-9292. Dostupné z: doi:10.3390/electronics9030531

SETHI, Pallavi a Smruti R. SARANGI, 2017. Internet of Things: Architectures, Protocols, and Applications. Journal of Electrical and Computer Engineering [online]. 2017, 1-25 [cit. 2022-01-01]. ISSN 2090-0147. Dostupné z: doi:10.1155/2017/9324035

SHEN, Xuemin, 2020. Encyclopedia of Wireless Networks. Springer International Publishing. ISBN 978-3-319-78261-4.

Sigfox: One network A billions dreams [online], 2013. [cit. 2021-07-23]. Dostupné z: https://lafibre.info/images/3g/201302_sigfox_whitepaper.pdf

SIKDER, Amit Kumar, Giuseppe PETRACCA, Hidayet AKSU, Trent JAEGER a Selcuk ULUAGAC, 2018. A Survey on Sensor-based Threats to Internet-of-Things (IoT) Devices and Applications [online]. [cit. 2021-11-17]. Dostupné z: https://www.researchgate.net/publication/322975901_A_Survey_on_Sensor-based_Threats_to_Internet-of-Things_IoT_Devices_and_Applications

SILEX TECHNOLOGY AMERICA, INC., 2020. First Commercial 802.11ah (HaLow) Wireless Access Point [online]. [cit. 2021-07-23]. Dostupné z: https://www.silextechnology.com/hubfs/Resource%20PDF/AP-100AH%20Product%20Brief_v2.pdf

Silicon Labs Completes Acquisition of Sigma Designs' Z-Wave Business [online], 2018. [cit. 2021-07-15]. Dostupné z: https://news.silabs.com/2018-04-18-Silicon-Labs-Completes-Acquisition-of-Sigma-Designs-Z-Wave-Business

SINGH, Debabrata, Pushparaj PAL, Manish MISHRA, Anil LAMBA a Shrabanee SWAGATIKA, 2020. Security Issues In Different Layers Of Iot And Their Possible Mitigation [online]. [cit. 2021-11-17]. Dostupné z: doi:10.13140

SINHA, Rashmi Sharan, Yiqiao WEI a Seung-Hoon HWANG, 2017. A survey on LPWA technology: LoRa and NB-IoT. ICT Express. 3(1), 14-21. ISSN 24059595. Dostupné z: doi:10.1016/j.icte.2017.03.004

STANDARD PRICE LIST [online], 2020. [cit. 2021-07-23]. Dostupné z: https://sigfox.cz/admin-data/storage/get/50-web-simplecellstandard-price-list20200727.pdf

STOYNOV, Viktor, 2019. Low Power Wide Area Networks Operating in the ISM Band- Overview and Unresolved Challenges. Springer, Cham. ISBN 978-3-030-23976-3.

The Things Network: Device Classes [online], 2021. [cit. 2021-08-11]. Dostupné z: https://www.thethingsnetwork.org/docs/lorawan/classes/

WEIPING AND CHOI, Sun, 2013. IEEE 802.11ah a long range 802.11 wlan at sub 1 ghz. Journal of ICT Standardization. 2013, 83-107. Dostupné z: doi:10.13052/jicts2245-800X.115

Wiring: About [online], 2022. [cit. 2022-01-16]. Dostupné z: http://wiring.org.co/about.html

YAQOOB, Ibrar, Ejaz AHMED, Ibrahim Abaker Targio HASHEM, Abdelmuttlib Ibrahim Abdalla AHMED, Abdullah GANI, Muhammad IMRAN a Mohsen GUIZANI, 2017. Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges. IEEE Wireless Communications [online]. 24(3), 10-16 [cit. 2021-11-17]. ISSN 1536-1284. Dostupné z: doi:10.1109/MWC.2017.1600421

Zabbix: Explore Zabbix features [online], 2022. [cit. 2022-01-22]. Dostupné z: https://www.zabbix.com/features#baseline_monitoring

ZIGBEE ALLIANCE, INC., 2015. ZigBee Specification [online]. [cit. 2021-07-14]. Dostupné z: https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf

ZigBee Sensor ZS-10 [online], 2014. In: . [cit. 2022-02-16]. Dostupné z: http://www.a2s.pl/products/zigbee/zs10/zs10_en.pdf

8 Seznam obrázků a tabulek

8.1 Seznam obrázků

8.2 Seznam tabulek

9 Přílohy

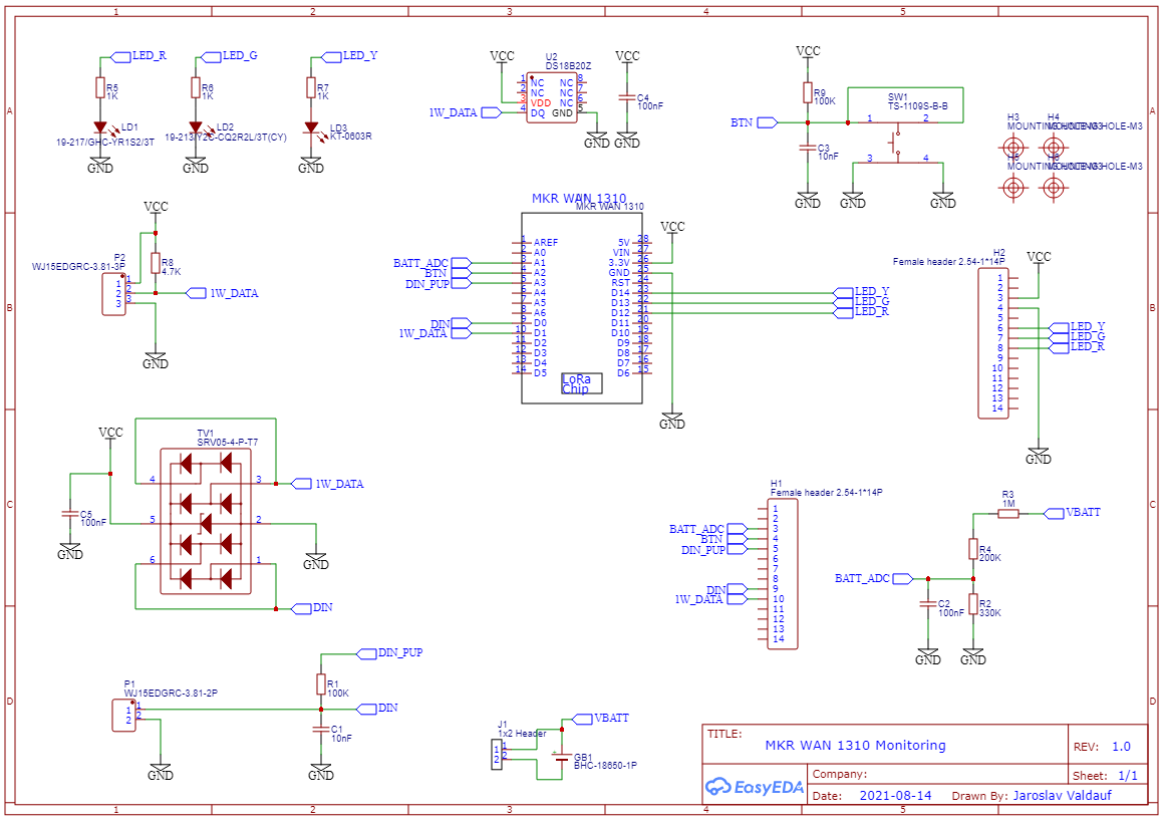
9.1 Příloha A – Zdrojový kód síťového serveru

```
# -*- coding: UTF-8 -*-

import socket
import struct
import json
import base64
import binascii

from django.core.management.base import BaseCommand, CommandError

from monitoring.models import Gateway, Zprava, Zarizeni
from monitoring import settings as app_settings

from monitoring.utils import to_little, decoduj_payload_na_hex, zpracuj_data

import logging
logger = logging.getLogger(__name__)

class Command(BaseCommand):

    def handle(self, *args, **options):

        # Definice socket serveru
        IP_ADRESA = getattr(app_settings, "IP_ADRESA")
        PORT = getattr(app_settings, "PORT")

        # Typy paketů
        PUSH_DATA = 0
        PUSH_ACK = 1
        PULL_DATA = 2
        PULL_RESP = 3
        PULL_ACK = 4
        TX_ACK = 5

        # Spuštění serveru
        sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        sock.bind((IP_ADRESA, PORT))

        # PULL ACK odpověď
        def odeslat_pull_ack(verze, token, gatewayEUI, ip_adresa, port):
            packet = struct.pack('<BHBp', verze, int(token), PULL_ACK,
bytes(gatewayEUI, 'utf-8'))
            sock.sendto(packet, (ip_adresa, port))

        # PUSH ACK odpověď
        def odeslat_push_ack(prijata_zprava):
            sock.sendto(prijata_zprava.push_ack_packet(),
(prijata_zprava.ip_adresa, prijata_zprava.port))


        while True:
            zpracovana_data = None
```

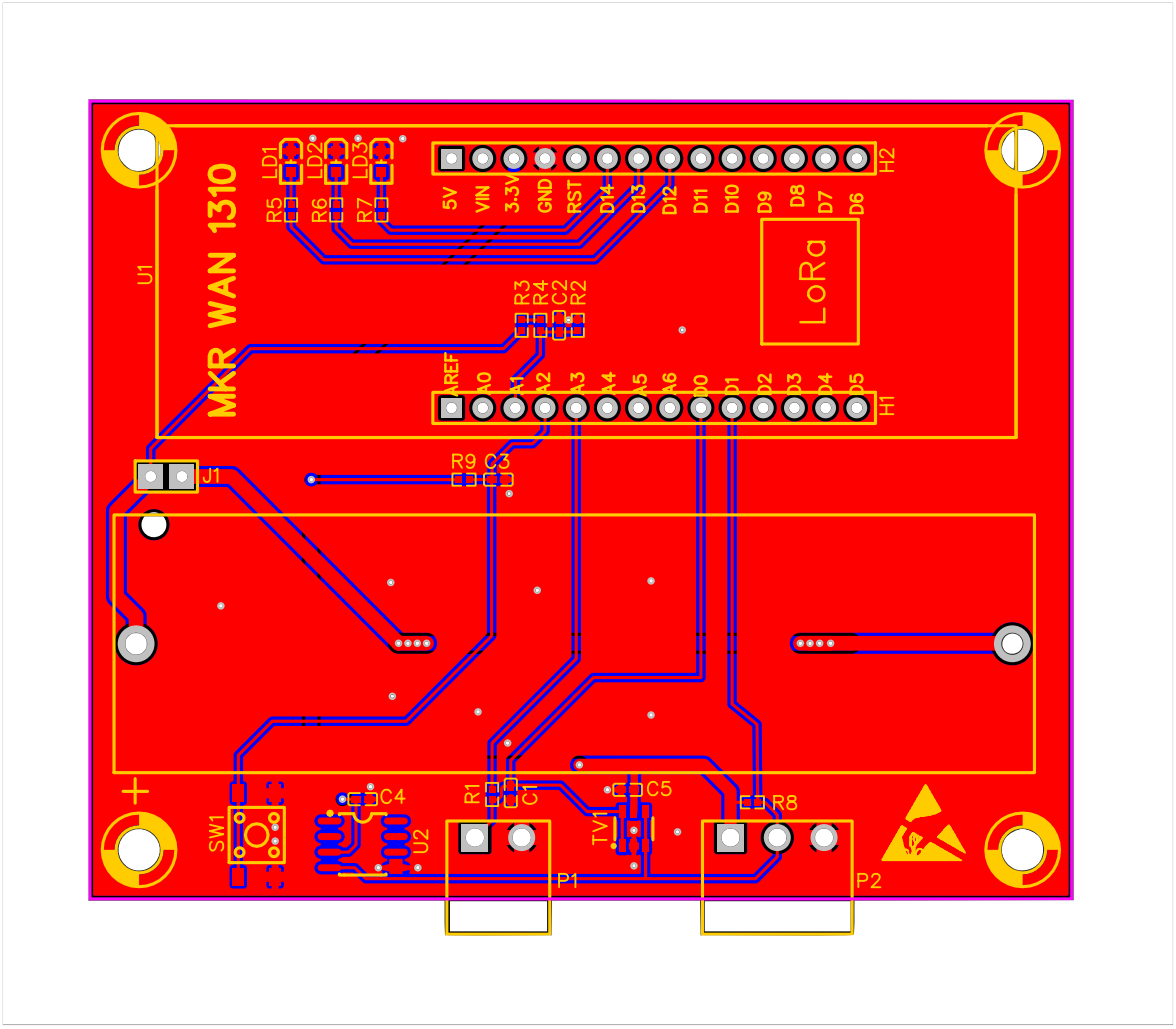
```
            data, addr = sock.recvfrom(1024)
            print("received message from {}: {}".format(addr, data))
            (verze, token, typ_zpravy) = struct.unpack('<BHB', data[:4])
            gatewayEUI = binascii.hexlify(data[4:12]).decode("utf-8")
            try:
                gateway = Gateway.objects.get(eui=gatewayEUI.upper(),
povolena=True)
            except Exception as e:
                logger.warning("Neznámá nebo zakázaná brána s EUI:
{}".format(gatewayEUI))
            else:
                if (typ_zpravy in (1, 2) or
                    verze == 1 and typ_zpravy in (PUSH_DATA, PULL_DATA) or
                    verze == 2 and typ_zpravy in (PUSH_DATA, PULL_DATA,
TX_ACK)):
                    if typ_zpravy == PUSH_DATA:
                        if len(data) < 12:
                            logger.error("Délka zprávy PUSH_DATA je kratší než
minimální délka 12")
                        hex_data =
decoduj_payload_na_hex(data[12:].decode("utf-8"))
                        if hex_data:
                            zpracovana_data = zpracuj_data(hex_data)
                        if zpracovana_data:
                            mic=zpracovana_data['mic']
                            zarizeni =
Zarizeni.objects.get(devaddr=zpracovana_data['devaddr'])
                            if Zprava.objects.filter(mic=mic,
zarizeni=zarizeni, created__gte=now() - datetime.timedelta(seconds =
getattr(app_settings, 'DUPLICITNI_MIC_S', 60))).count() > 1:
                                logger.debug("Nalezena duplicitní zpráva s MIC:
{}".format(mic))
                            else:
                                zpracovana_data['gateway'] = gateway
                                zpracovana_data['typ_zpravy_GWMP'] = typ_zpravy
                                zpracovana_data['verze'] = verze
                                zpracovana_data['token'] = token
                                zpracovana_data['ip_adresa'] = addr[0]
                                zpracovana_data['port'] = addr[1]
                                zpracovana_data['smer'] = 'RX'
                                zpracovana_data['payload'] =
data[12:].decode("utf-8")
                                zpracovana_data['hex_data'] = hex_data
                                zpracovana_data['zarizeni'] = zarizeni
                                zprava = Zprava(**zpracovana_data)
                                zprava.save()
                                odeslat_push_ack(zprava)
                    elif typ_zpravy == PULL_DATA:
                        if len(data) < 12:
                            logger.error("Délka zprávy PULL_DATA je kratší než
minimální délka 12")
                        else:
                            odeslat_pull_ack(verze, token, gatewayEUI, addr[0],
addr[1])
                else:
                    logger.warning("Nepodporovaný typ zprávy: {} a verze:
{}".format(typ_zpravy, verze))
```

9.2 Příloha B – Nákresy plošného spoje

9.2.1 Schéma zapojení



9.2.2 Nákres PCB



9.3 Příloha C – Zdrojový kód zařízení

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <CayenneLPP.h>
#include <string>
#include <MKRWAN_v2.h>
#include <ArduinoLowPower.h>

// Nastavení barev diod
const int NONE = 0;
const int GREEN = 1;
const int ORANGE = 2;
const int RED = 3;

// Definice pinů pro
const int teplota_PIN = 0;
const int baterie_PIN = A1;
const int tlacitko_PIN = A2;
const int kontakt_PIN = 1;
const int kontakt_zem_PIN = A3;

const int LED_1_PIN = 12;
const int LED_2_PIN = 13;
const int LED_3_PIN = 14;

int stav_zpravy = 0;
// periodické odesílání zpráv po X sekundách
int keep_alive_timeout = 1800;

// Parametry pro ABP ověření
const char *devAddr = "260BD67E";
const char *nwkSKey = "E451054EC08B24496328268D19B89A04";
const char *appSKey = "5C3030F011E3DB891DD4623F0DE9F080";

// Parametry pro OTAA ověření
//const char *appEui = "";
//const char *appKey = "";

LoRaModem modem;
CayenneLPP lpp(51);
OneWire oneWire(teplota_PIN);
DallasTemperature sensors(&oneWire);

void setup() {
    // Nastavení referenčního napětí
    analogReference(AR_INTERNAL1V0);
    // Inicializace vstupních/výstupních PINů
    pinMode(tlacitko_PIN, INPUT);
    pinMode(kontakt_PIN, INPUT_PULLUP);
    pinMode(5, INPUT_PULLUP);
    pinMode(LED_BUILTIN, OUTPUT);
    pinMode(kontakt_zem_PIN, OUTPUT);
    pinMode(LED_1_PIN, OUTPUT);
    pinMode(LED_2_PIN, OUTPUT);
    pinMode(LED_3_PIN, OUTPUT);
    // Otevření sériového portu a nastavení data rate na 9600 bps
    Serial.begin(9600);
```

```
    // Rozsvícení oranžové diody
    status_led(ORANGE);
    Serial.println("Čekám na stisktnutí tlačítka.");
    // Cekani na stisknutí tlačítka
    while (digitalRead(tlacitko_PIN)){}
    // Pokud se nepovede zapnout radio, rozsvítí se červená dioda a po řs se
zařízení restaruje
    if (!modem.begin(EU868)) {
        Serial.println("Nelze spustit radio modul.");
        status_led(RED);
        delay(5000);
        NVIC_SystemReset();
    };
    // Pokud je radio spuštěno, rozsvítí se zelená dioda
    Serial.println("Radio modul spuštěn.");
    status_led(GREEN);
    int connected;
    connected = modem.joinABP(devAddr, nwkSKey, appSKey);
    //connected = modem.joinOTAA(appEui, appKey);
    if (!connected) {
        Serial.println("Nelze se připojit k síti.");
        status_led(RED);
        delay(5000);
        NVIC_SystemReset();
    }
    Serial.println("Připojeno.");
    status_led(NONE);
    LowPower.attachInterruptWakeup(digitalPinToInterrupt(kontakt_PIN), dummy,
CHANGE);
    LowPower.attachInterruptWakeup(keep_alive_timeout*1000, dummy, CHANGE);
}

void loop() {
    digitalWrite(LED_BUILTIN, HIGH);
    zprava();
    digitalWrite(LED_BUILTIN, LOW);
    delay(1000);
    LowPower.deepSleep(keep_alive_timeout*1000);
}

void zprava() {
  /* Metoda pro odeslání zprávy s daty.
   * Nejprve se resetuje instance CayenneLPP.
   * Pote se vyčtou hodnoty vstupů.
   * Přidají se do instance CayenneLPP se
   * zvoleným kanálem a typem hodnoty.
   * Aktivuje se modem, do nového paketu se zapíše
   * instance s hodnotami o dané velikosti.
   * Paket se odešle (false = Unconfirmed Data Up)
   */
    lpp.reset();
    int kontakt_hodnota = kontakt();
    float teplota_hodnota = teplota();
    float baterie_hodnota = baterie();
    delay(500);
    lpp.addDigitalOutput(1, kontakt_hodnota);
    lpp.addTemperature(2, teplota_hodnota);
    lpp.addAnalogInput(3, baterie_hodnota);
```

```
    modem.setPort(3);
    modem.beginPacket();
    modem.write(lpp.getBuffer(), lpp.getSize());
    modem.endPacket(false);
}

float teplota() {
    /* Vyčtení hodnoty z teplotního čidla
     *  Nejprve se aktivuje pin, ke kterému je čidlo připojeno,
     *  poté se vyčte hodnota z prvního senzoru na 1-wire sběrnici.
     *  Jako první na sběrnici je externí teplotní čidlo,
     *  pokud není připojeno, je první hodnotou interní čidlo.
     */
    pinMode(teplota_PIN, INPUT);
    sensors.begin();
    sensors.requestTemperatures();
    float teplota_value = sensors.getTempCByIndex(0);
    pinMode(teplota_PIN, OUTPUT);
    return teplota_value;
}

int kontakt() {
    // reverzní hodnota na kontakt_PIN pro stav monitorovaného zařízení
    int kontakt_raw = digitalRead(kontakt_PIN);
    return (kontakt_raw) ? 0 : 1;
}

float baterie() {
    // Vyčtení hodnoty na "baterie_PIN" a vynásobení koeficientem pro získání
napětí ve V
    return analogRead(baterie_PIN) * (4.7 / 1023.0);
}

void status_led(int status){
    // Vypnutí všech diod a poté zapnutí dle předaného parametru
    digitalWrite(LED_1_PIN, LOW);
    digitalWrite(LED_2_PIN, LOW);
    digitalWrite(LED_3_PIN, LOW);
    switch (status) {
        case 1:
            digitalWrite(LED_1_PIN, HIGH);
            break;
        case 2:
            digitalWrite(LED_2_PIN, HIGH);
            break;
        case 3:
            digitalWrite(LED_3_PIN, HIGH);
            break;
    }
}

void dummy() {
}
```