

**ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ**

Katedra technologií a měření

DIPLOMOVÁ PRÁCE

Internet věcí na platformě Arduino

Bc. Vojtěch Řehořík

Plzeň 2018

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta elektrotechnická
Akademický rok: 2017/2018

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Vojtěch ŘEHOŘÍK**
Osobní číslo: **E16N0025P**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Komerční elektrotechnika**
Název tématu: **Internet věcí na platformě Arduino**
Zadávající katedra: **Katedra technologií a měření**

Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se s problematikou "Internet věcí" (IoT).
2. Proveďte rešerši dostupných bezdrátových modulů pro IoT na platformě Arduino.
3. Navrhněte a realizujte jednoduché bezdrátové zařízení, které zaznamenává teplotu a vlhkost.
4. Naměřená data bezdrátově zaznamenejte do PC a vizualizujte.
5. Vytvořené technické a programové prostředky podrobně popište.



Rozsah grafických prací: **podle doporučení vedoucího**

Rozsah kvalifikační práce: **40 - 60 stran**

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


Seznam odborné literatury:

1. Pinker, J., Mikroprocesory a mikropočítače
2. Herout, P., Učebnice jazyka C - 1. díl
3. Doukas C., Building internet of things with the Arduino
4. Internetové zdroje

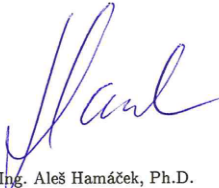
Vedoucí diplomové práce: **Ing. Jaroslav Freisleben, Ph.D.**
Katedra technologií a měření

Datum zadání diplomové práce: **10. října 2017**

Termín odevzdání diplomové práce: **24. května 2018**


Doc. Ing. Jiří Hammerbauer, Ph.D.
děkan




Doc. Ing. Aleš Hamáček, Ph.D.
vedoucí katedry

V Plzni dne 10. října 2017

Anotace

Předložená diplomová práce pojednává o problematice „Internetu věcí“ (Internet of things - IoT) a možnostech jeho využití v dnešním světě. Hlavním cílem práce je návrh a následná realizace zařízení na platformě Arduino, které měří environmentální parametry prostředí – teplotu prostředí, relativní vlhkost a hladinu CO₂. Naměřená data budou bezdrátově přenášena prostřednictvím IoT sítě na cloudový server, ze kterého bude možno data dále zpracovávat, vizualizovat a archivovat. Takové zařízení musí mít velmi nízkou spotřebu elektrické energie.

Klíčová slova

Internet věcí, IoT, Arduino MKRFOX1200, Sigfox, Analog Infrared CO₂, BME280, Thingspeak, Losant, Thingier.

Abstract

Submitted dissertation is concerned about problematics of Internet of Things (IoT) and possibilities of its use in today's world. Main aim of this dissertation is to plan and realize a device which works on Arduino platform. This device can measure environmental parameters of surroundings – temperature, relative humidity and level of CO₂. Measured data will be wirelessly transferred via IoT network to the cloud server from which we can process, visualize and archive obtained data. Such device must have a very low consumption of an electric energy.

Key words

Internet of Things, IoT, Arduino MKRFOX1200, Sigfox, Analog Infrared CO₂, BME280, Thingspeak, Losant, Thinger.

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této diplomové práce, je legální.

.....
podpis

V Plzni dne 12. 5. 2018

Bc. Vojtěch Řehořík

Poděkování

Tímto bych rád poděkoval vedoucímu diplomové práce Ing. Jaroslavu Freislebenovi Ph.D. za cenné profesionální rady, připomínky a metodické vedení práce. Dále pak Pavlu Sedláčkovi za pomoc při 3D tisku.

Obsah

Obsah	8
Seznam symbolů a zkratek.....	10
Úvod	11
1 IoT a bezdrátové moduly pro platformu Arduino.....	12
1.1 Internet věcí.....	12
1.1.1 Využití IoT.....	13
1.1.2 CIoT.....	13
1.1.3 IIoT.....	14
1.1.4 EIoT.....	15
1.1.5 Problémy zavádění IoT.....	16
1.2 Sigfox.....	16
1.2.1 Přenos a modulace.....	17
1.2.2 Architektura sítě.....	18
1.2.3 Přenosový protokol.....	19
1.2.4 Ceník.....	20
1.3 LoRa.....	21
1.3.1 Modulace.....	22
1.3.2 Zprávy.....	23
1.3.3 Energetická náročnost.....	24
1.4 Narrow Band.....	25
2 Hardware.....	26
2.1 Bezdrátové moduly pro Arduino.....	26
2.2 Arduino MKRFOX 1200.....	26
2.2.1 Baterie, piny a LED diody.....	27
2.3 BME280 Weather Click.....	28
2.4 Gravity Infrared CO2.....	28
2.5 Step-UP měnič 5 V.....	29
2.6 BTS117 T0220.....	30
2.7 Šasi.....	31
2.8 Spotřeba elektrické energie.....	32
3 Programové prostředky	37

3.1	Arduino IDE.....	37
3.2	Sigfox backend.....	38
3.2.1	Custom Callback – Email	39
3.2.2	Custom Callback – Thingspeak	41
3.2.3	Custom Callback – Thinger	41
3.2.4	Custom Callback – Losant.....	43
3.2.5	IoT platformy	44
3.2.6	Thingspeak.....	45
3.2.7	Thinger.....	48
3.2.8	Losant.....	50
3.2.9	Porovnání ceniků jednotlivých platforem	56
4	Kvalita ovzduší ve vnitřních prostorech	60
4.1	Vlastní měření	61
4.1.1	ZF Engineering	61
4.1.2	Byt.....	62
4.1.3	ZČU FEL – EU506	63
	Závěr	66
	Seznam literatury	68
	Seznam obrázků	69
	Seznam tabulek	71
	Přílohy.....	72

Seznam symbolů a zkratek

API.....	Application Programming Interface, rozhraní pro programování aplikací
ARFF	Attribute Relationship File Format, textový formát pro datové soubory
CIoT.....	Consumer IoT, spotřebitelský internet věcí
Cloud.....	Poskytování služeb či programů servery dostupnými z internetu
CRC	Cyclic Redundancy Check, cyklický redundantní součet
CSV.....	Comma-Separated Values, souborový formát určený pro tabulky
dB.....	Decibel, měřítko podílu dvou hodnot
DBPSK.....	Differential Binary Phase Shift Keying, klíčování fázovým posunem
EIoT	Enterprise IoT, podnikový internet věcí
Float	Datový typ s plovoucí desetinou čárkou
I2C	Inter-Integrated Circuit, multi-masterová počítačová sériová sběrnice
IIoT	Industrial IoT, průmyslový internet věcí
Int.....	Datový typ pro celé číslo
IoT.....	Internet of things, internet věcí
JSON.....	JavaScript Object Notation, JavaScriptový objektový zápis
LPWAN	Low-Power Wide-Area Network, nízkoenergetická síť
MAC	Media Access Control, jednoznačný identifikátor síťového zařízení
OSI.....	Open Systems Interconnection, standardizace komunikace v sítích
ppm	Parts Per Million, jedna miliontina
RX.....	Receive, příjem
SCL	Synchronous clock, hodinový signál
SDA	Synchronous data, datová kanál
SPI.....	Serial Peripheral Interface, sériové periferní rozhraní
TCP/IP	Transmission Control Protocol/Internet protocol, primární přenosový protokol/protokol síťové vrstvy
TX	Transmit, vysílání
URL	Uniform Resource Locator, řetězec sloužící k přesné specifikaci umístěných zdrojů informací

Úvod

Cílem této diplomové práce je návrh a následná realizace zařízení spadajícího do konceptu internetu věcí. Toto zařízení bude pomocí senzorů zaznamenávat teplotu prostředí, relativní vlhkost a hladinu CO₂, díky čemu umožní kontrolu kvality ovzduší v uzavřených prostorech. Naměřená data budou bezdrátově přenášena prostřednictvím IoT sítě na cloudový server, ze kterého bude možno data dále zpracovávat, vizualizovat a archivovat.

Zařízení bude realizované na platformě Arduino MKRFOX1200 v kombinaci se senzory Bosch BME280 a Gravity Analog Infrared CO₂. Všechny tyto komponenty budou umístěny v plastové šasi, jež bude zhotovena na 3D tiskárně.

Text je rozdělen do 4 kapitol. V první kapitole je popsána problematika Internetu věcí (IoT) a rešerše dostupných bezdrátových modulů pro IoT na platformě Arduino. Ve druhé kapitole je uveden hardware, který byl použit pro realizaci zařízení, společně se schématy zapojení jednotlivých komponent a návrh šasi. Třetí kapitola popisuje programové prostředky použité pro snímání, přenos a vizualizaci veličin. Poslední kapitola obsahuje výsledky měření společně se závěrem.

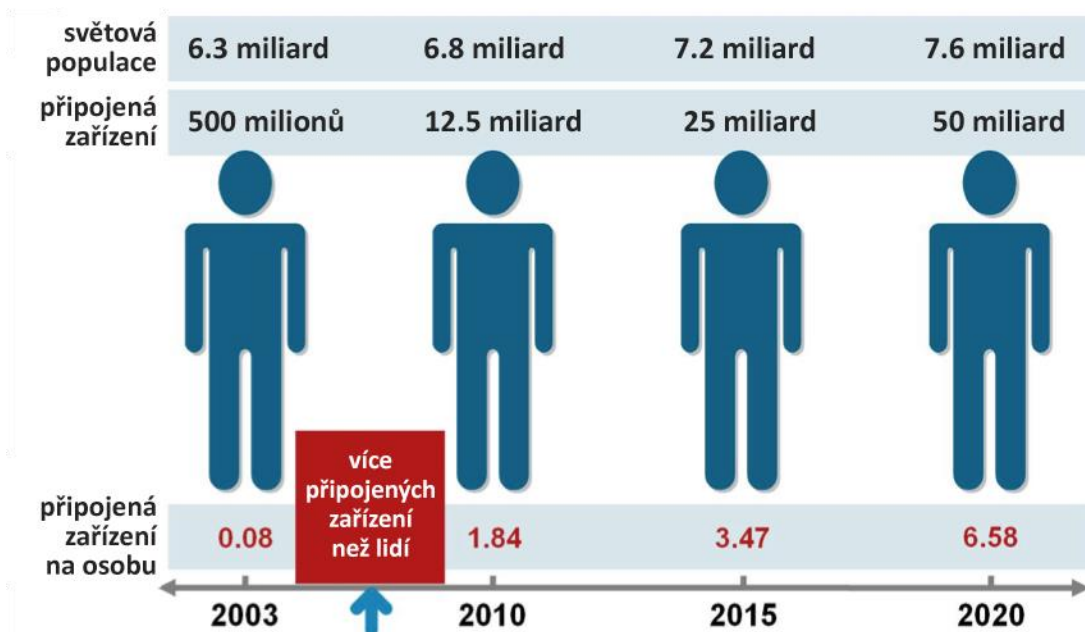
1 IoT a bezdrátové moduly pro platformu Arduino

V této kapitole jsou popsány základní informace ohledně Internetu věcí, společně s vhodnými bezdrátovými moduly pro platformu Arduino.

1.1 Internet věcí

Poprvé byl pojem Internet věcí použit v roce 1999 ve stejnojmenné prezentaci Britem Kevinem Ashtonem. Ve své prezentaci popsal systém, ve kterém byly senzory, neboli věci, připojeny k internetu. Díky sdílení dat z těchto senzorů se následně zvýší vnímání okolního světa. Jako příklad použil RFID tagy, které se v obchodech používají za účelem počítání a sledování zboží bez lidského zásahu.

Vznik internetu věcí je datován mezi roky 2008 a 2009, kdy podle společnosti CISCO překročil počet připojených zařízení k internetu počet světové populace. Jako první byly k internetu připojovány počítače, dále chytré telefony apod. Předpokládá se, že dalším zlomovým rokem bude rok 2020, kdy počet zařízení připojených k internetu překročí 50 miliard. Na jednoho člověka tedy bude připadat 6 – 7 zařízení, viz obr. 1.



Obr. 1 Vývoj IoT [1]

Hlavním cílem IoT je propojení zařízení, systémů a služeb za účelem získání co nejvíce dat, která lze převést na informace a následně na znalosti, které je možné aplikovat. Díky těmto znalostem mohou vznikat IoT systémy, které jsou plně autonomní. Z toho vyplývá, že čím více dat budou zařízení schopna poskytovat o reálném světě, tím více dat bude k dispozici k následné analýze, a tím více znalostí bude možné využít v průmyslu nebo každodenním životě. [2]

1.1.1 Využití IoT

Kromě vysokých nároků na bezpečnost jsou hlavními požadavky pro správnou funkčnost architektury Internetu věcí:

- Sběr dat
- Uložení dat
- Analýza dat
- Sdílení dat

V dnešní době se IoT dělí do třech hlavních směrů. Prvním je průmyslový Internet věcí (Industrial IoT, IIoT), který má za cíl zlepšit efektivitu využívání zdrojů za účelem úspor. Druhým je spotřebitelský internet věcí (Consumer IoT, CIoT), jehož smyslem je zjednodušovat každodenní život. Třetí je podnikový internet věcí (Enterprise IoT, EIoT), tento termín označuje veškerá zařízení používaná ve firemních prostředcích za účelem zvýšení zisků, rozšíření automatizace a zefektivnění procesů. [2]

1.1.2 CIoT

Do spotřebitelského internetu věcí patří především inteligentní domácnosti. Automatické spínání a regulace topení a dálkové ovládání spotřebičů přispívají ke snížení spotřeby energie. Detekce otevření oken a dveří funguje jako bezpečnostní systém proti zlodějům. Detekce unikající vody může zabránit velkým škodám na majetku. Inteligentní lednička, které sama monitoruje svůj obsah a provádí automatické objednávky potravin, které docházejí, ušetří svým majitelům spoustu času. Dalším příkladem může být inteligentní nakupování, kdy aplikace může zákazníkovi poradit při nákupu podle zákaznických preferencí a zvyků nebo upozornit na přítomnosti alergických látek. NFC platby pomocí

chytrých telefonů umožňují zaplacení bez použití hotovosti nebo platební karty. Velmi oblíbené jsou také chytré hodinky, které informují jejich majitele o jeho zdravotních funkcích, umožňují volání, čtení SMS apod. Sledování pomocí GPS trackerů usnadňuje dohled nad dětmi nebo domácími mazlíčky, a přispívá tak k jejich bezpečnosti. Využití IoT ve spotřebitelské sféře má prakticky neomezené možnosti a záleží pouze na subjektivním přání jeho uživatele. [3]

1.1.3 IIoT

Průmyslový Internet věcí můžeme dělit na tato odvětví:

- Průmyslová automatizace
- Dopravní průmysl
- Energetický průmysl
- Zdravotnictví
- Smart City

Průmyslová automatizace se zaměřuje především na propojení všech hardwarových a softwarových částí tvořících automatizační systém k síti. Díky vzájemnému sdílení informací se může automatizace nastavovat tak, aby se dosahovalo optimálního výkonu a minimálních prostojů. Proces průmyslové automatizace obsahuje mechanické, digitální a lidské složky. Každá z těchto složek může mít ve kterémkoliv okamžiku informace, které jsou užitečné pro jinou složku. [4]

V dopravním průmyslu je snaha co nejvíce snižovat náklady a zvyšovat efektivitu dodávek. Díky sdílení dopravních informací se zvyšuje plynulost dopravy. Senzory mohou sledovat vibrace nebo teplotu a tím kontrolovat kvalitu dopravy. Sledování a sdílení informací ze skladů je jedním z hlavních pilířů řízení logistiky JIT (Just in time). V neposlední řadě se jedná o zvýšení bezpečnosti na silnicích a vývoj autonomních vozidel.

Základní vize Energetiky 4.0 je, že energetickou spotřebu země budou zabezpečovat klasické zdroje (tepelné a jaderné elektrárny) společně se sítí alternativních zdrojů (fotovoltaické elektrárny, vodní a větrné elektrárny, kogenerační jednotky a zásobníky energie). Pro správnou funkčnost bude nutné propojení všech zdrojů a spotřebitelů elektrické

energie. Jejich vzájemná komunikace podpoří následnou optimalizaci výroby a spotřeby elektrické energie. Díky tomuto propojení vzniknou chytré sítě, jež přispějí ke stabilitě řízení dodávek elektrické energie a sníží závislost na fosilních palivech. [5]

Internet věcí ve zdravotnictví umožňuje online monitorování zdravotního stavu pacientů v nemocnici nebo těch, kteří se lečí doma. Technologie IoT, které umožní automaticky sbírat a analyzovat data o pacientech, mohou na jejich základě zachytit počáteční fáze onemocnění a zabránit jejich průběhu. Telemedicína může eliminovat vzdálenost poskytované zdravotní péče, která by se díky tomu mohla dostat do míst jako je Afrika nebo Střední východ. Automatické skladování a dávkování léků je řízeno daty ze senzorů pro co nejvyšší efektivitu dodávek léků. [6]

Koncept Smart city se snaží o maximální využití moderních technologií pro zlepšování kvality života v daném městě. Důraz je kladen na synergické efekty mezi různými odvětvími (doprava, logistika, bezpečnost, energetika, správa budov apod.) Jako příklady využití IoT ve městech se může považovat: sledování provozu, kontrola kvality ovzduší, monitorování svozu odpadů, inteligentní pouliční osvětlení, komunikace občanů s úřady apod. V České republice je jedním z prvních průkopníků jihočeské město Písek. [7]

1.1.4 EIoT

Vývoj podnikového internetu věcí začal s jednoduchými zařízeními připojenými na internet, jako jsou např. tiskárny, fotoaparáty, termostat, kamery. Pokročilejší typy EIoT spočívají především ve sběru, shromažďování a analýze dat, které jsou nezbytné pro správnou funkci Business Intelligence (BI). Do BI spadají dovednosti, znalosti, technologie, aplikace, kvalita, rizika, bezpečnostní otázky. Obecně jsou tyto systémy klasifikovány jako systémy pro podporu rozhodování. Účelem zavádění EIoT je zefektivňování podnikových procesů, podpora podnikového řízení a plánování, prediktivní analýzy, sdílení informací napříč firmou atd.

1.1.5 Problémy zavádění IoT

Během vytváření infrastruktury Internetu věcí se setkáváme s řadou problémů, které je třeba vyřešit.

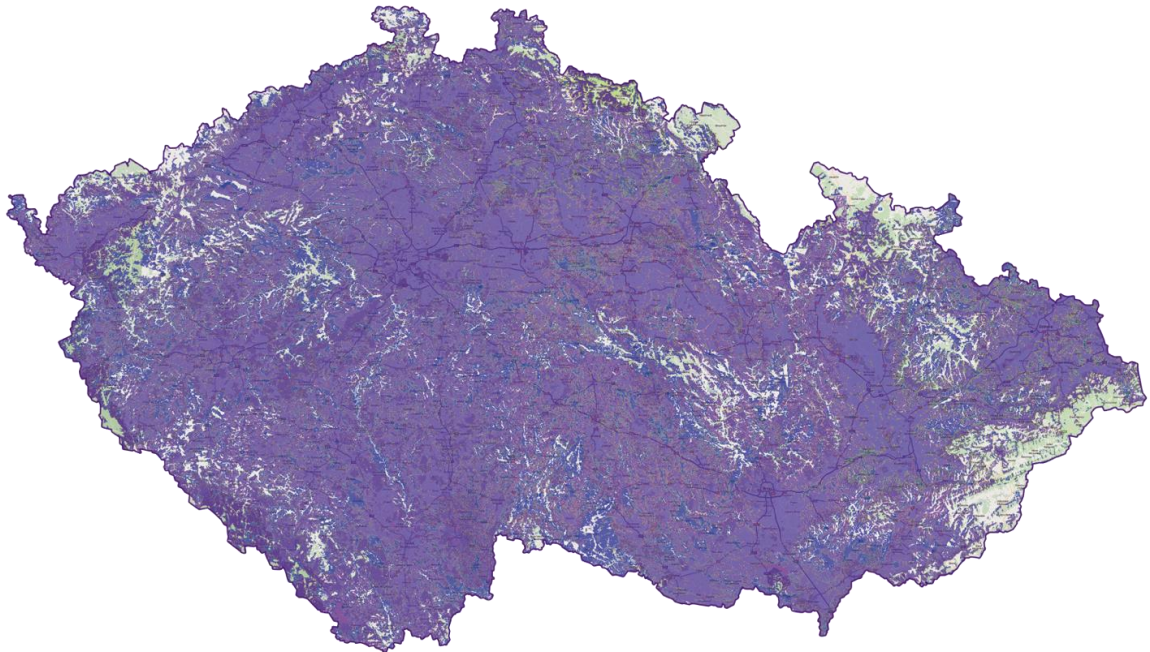
Interoperabilita neboli schopnost věcí poskytovat data jiným věcem a akceptovat data od jiných věcí, je jedním z hlavních problémů zavádění IoT. Tyto problémy jsou způsobené především tím, že jsou věci vytvářeny různými subjekty a používají odlišné způsoby komunikace a datové modely. K dosažení interoperability je nutné využívání standardů všude tam, kde je to možné.

Bezpečnost v IoT systémech je velice důležitá, protože mohou obsahovat velice citlivá data (např. informace o bankovních účtech, zdravotním stavu apod.). IoT systémy mohou řídit automatizované procesy v průmyslu nebo ve městech. Následné útoky hackerů na tyto systémy nebo zneužití citlivých dat tak může mít závažný dopad. Bezpečnostní funkce musí být zajištěny při přenosu dat, jejich archivaci, zpracování a zároveň u koncového uživatele.

Mezi další problémy patří velmi nákladné transfery obrovských objemů dat, jejich skladování a následná analýza. Vysoké pořizovací a provozní investice, jejichž návratnost je nejasná společně s nesplněnými očekáváním na IoT síť.

1.2 Sigfox

Sigfox je bezdrátový komunikační systém pro energeticky nenáročný přenos malého množství dat. Největší potenciál má u aplikací, kde dochází k občasnému zasílání naměřených dat, hlášení alarmových stavů nebo k základním příkazům řízení. Bezdrátová síť Sigfox patří mezi nízkopříkonové LPWAN sítě pro IoT systémy s dosahem až desítek km. Prvním a zatím jediným veřejným operátorem pro Internet věcí v České republice na technologii Sigfox je SimpleCell Networks a.s. Síť je budována ve spolupráci se společností T-Mobile, která poskytuje svoji infrastrukturu pro umístění základnových stanic. Na obr.2 je znázorněna mapa pokrytí České republiky pro listopad 2017, kdy bylo pokryto 92% území a 95% populace. [8]



Obr. 2 Pokrytí sítě Sigfox v ČR [8]

1.2.1 Přenos a modulace

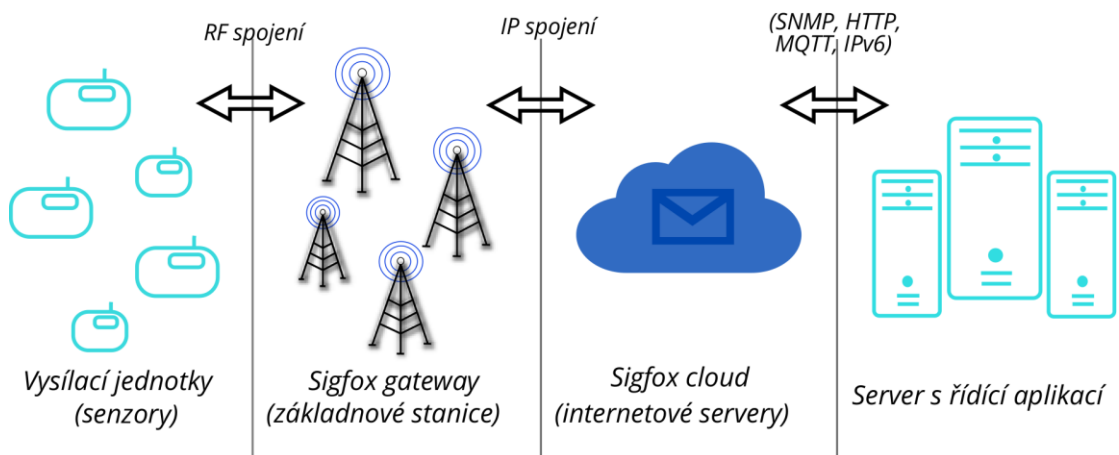
Technologie Sigfox pracuje v bez-licenčním frekvenčním pásmu 868 MHz (906 MHz v USA). Ke komunikaci je využíváno Ultra Narrow Band (UNB) pásmo pro vysílání pouze krátkého pulsu dat s vysílacím výkonem maximálně 100 mW a modulací pracující ve veřejném pásmu 200 kHz. Rychlost přenosu je 100 až 600 bit/s (závisí na regionu), zprávy v době přenosu zabírají šířku pásma 100 Hz. Vysílaná zpráva využívá DBPSK (Differential Binary Phase Shift Keying) modulaci, které vyžaduje pro přenosovou rychlost 1 bit/s frekvenční pásmo pouze 1 Hz. Při přenosové rychlosti Sigfox komunikace 100 bit/s bude tedy využita šířka pásma 100 Hz. Díky nízké přenosové rychlosti a úzkému přenosovému pásmu DBPSK modulace je využití přenosového spektra velmi efektivní (vysoká přenosová kapacita) a přijímač může demodulovat signály, které jsou blízko hladině šumu. Citlivost Sigfox přijímače je regulována na základě přenosové rychlosti. Pokud je přenosová rychlost 100 bit/s nastavuje se citlivost přijímače na -142 dBm, při přenosové rychlosti 600 bit/s na -134 dBm. [8]

1.2.2 Architektura sítě

Síť pro technologii Sigfox má topologii hvězdy a je budována na buňkovém principu. Obsahuje základnové stanice společně s buňkami, které pokrývají určité oblasti, ve kterých do sítě Sigfox posílají zařízení své zprávy. Jednotlivé stanice obdržené zprávy následně posílají do Sigfox cloudu pomocí internetové komunikace TCP/IP. Základnové stanice detekují a demodulují zprávu jen pro Sigfox cloud. V cloudu jsou zprávy přetříděny a následně odeslány do zákaznických zařízení nebo IoT platformem. Na obr. 3 je znázorněna architektura sítě Sigfox.

Data v Sigfox síti nemají žádnou předem definovanou vnitřní strukturu, závisí tedy pouze na uživateli, jak budou data vložena (odeslána) a následně interpretována (přijata). Uživatelé se k datům ze Sigfox cloudu, který je vytvořen provozovatelem sítě, dostávají pomocí softwarového rozhraní API (Application Programming Interface) ze svého systému nebo aplikace.

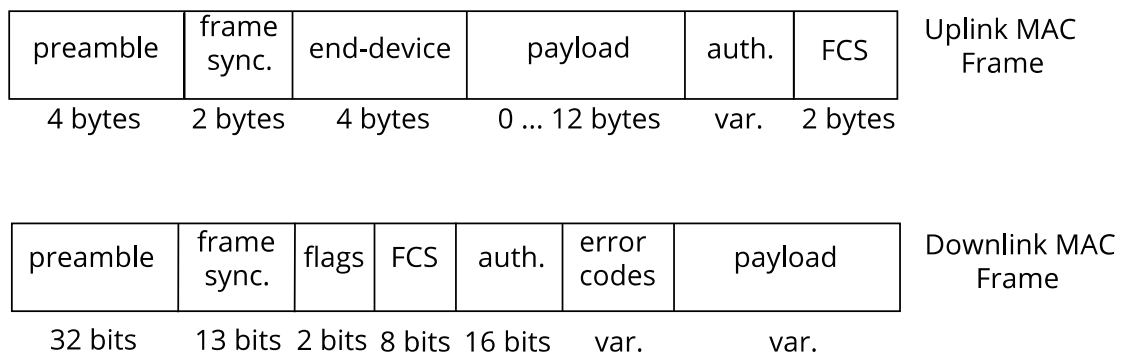
Všechna zařízení s podporou Sigfox jsou vybavena certifikátem a identifikují se pomocí interních identifikátorů. Tento způsob adresování umožňuje téměř neomezený počet zařízení, která je síť schopna obsluhovat. Díky interním identifikátorům se nemusí používat nastavitelné adresy (např. IP adresy), nebo SIM karty atd. Identifikátor lze přirovnat k MAC adrese, kterou má každé ethernetové zařízení.



Obr. 3 Architektura sítě Sigfox [9]

1.2.3 Přenosový protokol

Protokol pro Sigfox byl vytvořen pro přenos malých objemů dat, díky tomu má celý systém malou spotřebu elektrické energie. Na obr. 4 je přenosový rámec tvořen 26 bajty, přičemž bajty 0 až 12 jsou volitelné a jsou použity pro tzv. užitečná (payload) data. Zpětný kanál pro odesílání zpráv do zařízení je omezený na 0 až 8 bajtů užitečných dat s omezením na 4 zprávy denně.

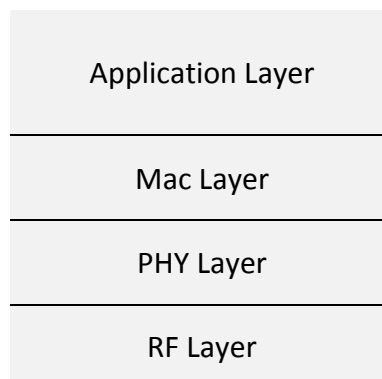


Obr. 4 Přenosový protokol [10]

Síť Sigfox má relativně malou přenosovou rychlost (100 bit/s), 12 bajtů užitečných dat a zbylých 14 bajtů dat přenosového rámce se tak přenáší přes 2 s. Denní limit je 140 vysílacích zpráv, každých 10 min je tedy možné odeslat data. Ve zbývajícím čase již zařízení nekomunikuje a tím šetří elektrickou energii.

Odesílané zprávy s 12 bajty dat tak mohou v praxi představovat např. záznam až 96 dvoustavových signálů, 3 naměřené hodnoty v datovém formátu float nebo 6 naměřených hodnot v datovém formátu int. Zpětné zprávy s 8 bajty dat jsou dostačující pro příkazy jako zapni/vypni, změny rozsahů senzorů nebo pro změnu frekvence odesílání zpráv.

Standardní OSI (Open System Interconnection) model Sigfox komunikace zobrazený na obr. 5 je možné rozdělit na 4 vrstvy. Rádiová vrstva (RF layer), která řídí frekvence a úroveň vysílacího výkonu. Fyzická vrstva (PHY layer), jež se stará o modulaci a při vysílání nebo příjmu slouží k definici adresy koncové stanice/jednotky. Síťová vrstva (MAC layer) slouží k detekci chyby při přenosu (CRC). Aplikační vrstva (Application layer) přenáší užitečná data do definovaných aplikací. [9]



Obr. 5 Sigfox OSI model[10]

1.2.4 Ceník

Společnost SimpleCell Networks a.s. nabízí 4 různé úrovně předplatného, které jsou definovány denním limitem zpráv na jedno zařízení. Cena předplatného je také ovlivněna počtem připojených zařízení. V tab. 1 jsou popsány ceny ročního předplatného.

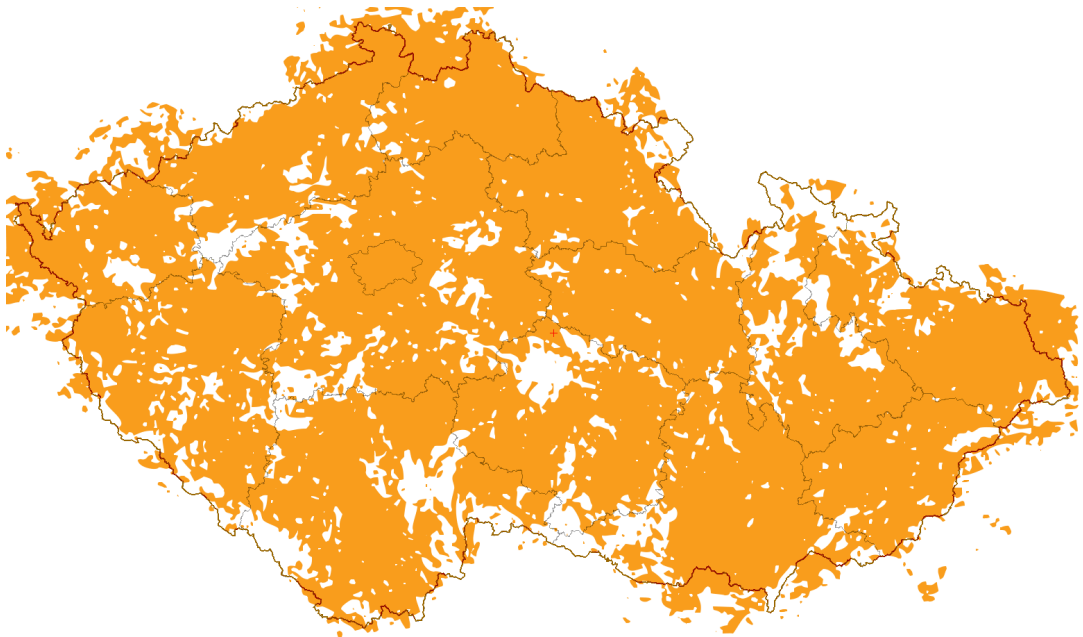
Úroveň předplatného	Počet zařízení ≤ 1000	Počet zařízení > 1000
Platinum	193 (Kč)	187 (Kč)
Gold	151 (Kč)	144 (Kč)
Silver	124 (Kč)	115 (Kč)
One	83 (Kč)	77 (Kč)

Úroveň předplatného	Uplink zprávy	Downlink zprávy
Platinum	140	4
Gold	100	2
Silver	50	1
One	2	0

Tab. 1 Ceník předplatného Simple Cell

1.3 LoRa

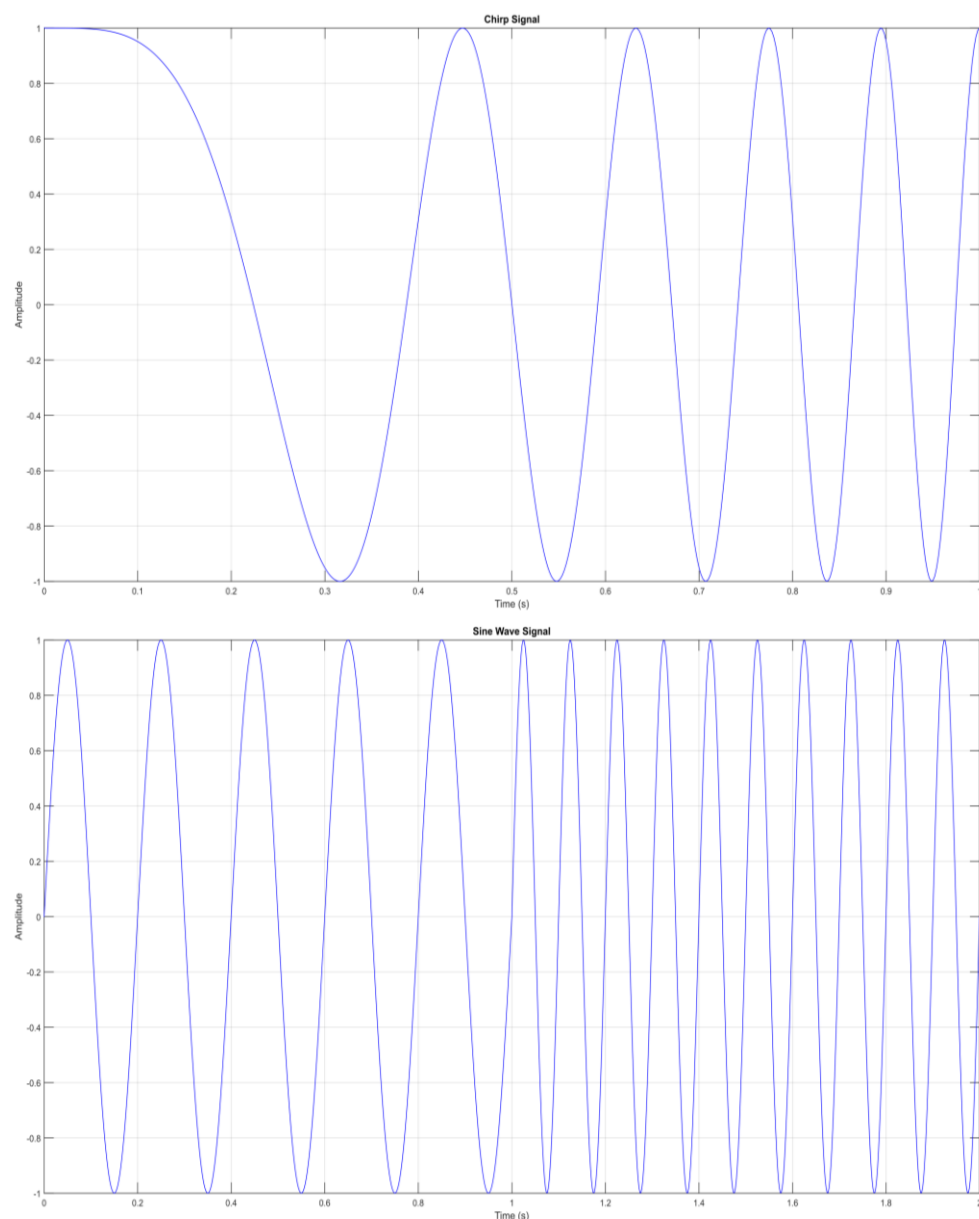
Technické označení LoRa Technology se skládá ze dvou pojmů, prvním je modulace LoRa a druhým protokol LoRaWAN. Jedná se o bezdrátový komunikační systém vhodný pro IoT aplikace s velmi nízkou spotřebou elektrické energie (LPWAN) a nízkým datovým tokem a dosahem až desítky km. Typicky se využívá pro vzdálenou signalizaci, měření a řízení. Pro Evropu LoRa využívá bezlicenční frekvenční pásmo 868 MHz (913 MHz v USA), kde používá 9 kanálů. Z toho 8 pro vzestupný směr od čidel a jeden pro sestupný směr k čidlům. Kanál pro sestupný směr je v oblasti spektra, kde je povolena komunikace 10% v časové ose. Na obr. 6 je znázorněno pokrytí České republiky sítí LoRa pro listopad 2017. České radiokomunikace do budoucna plánují celorepublikové pokrytí.



Obr. 6 Pokrytí LoRa [11]

1.3.1 Modulace

LoRa (Long range) je druh modulace Chirp, kterou patentovala firma Semtech. Chirp je poměrně nový druh modulace, kde je signál rozprostřen po celém vysílacím kanálu (125 kHz, 250 kHz, 500 kHz), obdobně jako u DSSS, FHSS apod. Odlišnost je, že ostatní modulace provádějí skokovou změnu kmitočtu, ale Chirp lineární, díky čemuž snižuje nároky na zdroj kmitočtu, zvyšuje odolnost vůči Dopplerovu jevu a jeden kanál může využívat několik vysílačů současně. Na obr. 7 je zobrazena Chirp modulace a modulace provedena skokovou změnou kmitočtu.



Obr. 7 Chirp modulace a modulace skokovou změnou kmitočtu

Technologie LoRa ke komunikaci od čidel využívá 8 kanálů. Pro každou přenosovou rychlost je pevně stanovená šířka pásma a rozprostření signálu (Spreading factor). Rozptyl přenosové rychlosti pro jednotlivé Spreading factory (SF) je:

- SF 12 přenosová rychlost 300 bit/s
- SF 7 přenosová rychlost 11 000 bit/s
- SF 7 s modulací FSK přenosová rychlost 55 000 bit/s

1.3.2 Zprávy

Každá zpráva obsahuje záhlaví, ve kterém všechna čidla vysílají svá unikátní čísla EUI, která je možné připodobnit k MAC adrese u ethernetových sítí. Zároveň všechny zprávy obsahují své sekvenční číslo. Všechny základnové stanice, které jsou v dosahu čidla a zprávu slyší, ji přepošlou do network serveru. Na úrovni network serveru se odstraňují duplicity na základě sekvenčního čísla, dále je možné vyhodnotit, zda nedošlo ke ztrátě předešlé zprávy nebo celé sekvence.

Uvnitř zprávy jsou dále zakódovány provozní informace ohledně rádiové vrstvy. Díky obousměrné komunikaci lze podle těchto informací monitorovat stav čidel, čidla řídit, měnit Spreading Factor a v neposlední řadě regulovat vysílací výkon. Při hustší infrastruktuře základnových stanic regulace výkonu ovlivňuje dosah signálu a tím eliminuje rušení, zároveň pozitivně ovlivňuje energetickou náročnost, a tím prodlužuje životnost baterií v čidlech.

Zpráva může mít maximální délku 255 bajtů. Takto velkou zprávu lze vysílat pouze pro rozsah Spreading Factor max. 7. Pro vyšší hodnoty se velikost zprávy snižuje v souvislosti s dobou vysílání jedné zprávy. Se zkracující se zprávou se také zkracuje prostor pro užitečná data (payload). Při standardní délce (255 bajtů) je užitečným datům určeno 240 bajtů. Pokud se nastaví Spreading factor 12, prostor pro užitečná data se zmenší na 51 bajtů, tato velikost je ale pro většinu aplikací dostatečná. [12]

1.3.3 Energetická náročnost

Jedna z největších výhod, které technologie LoRa má, je nízká energetická náročnost, jež je zajištěna kombinací několika faktorů.

- Nízký vysílací výkon a možnost jeho regulace během provozu (snižování vysílacího výkonu u čidel, která jsou umístěna v blízkosti základnových stanic)
- Četnost komunikace společně s požadavky na její dostupnost a přepínání čidel do pohotovostního režimu

Síť LoRa je využívána pro širokou škálu aplikací, které mají různé nároky na způsob komunikace a úsporu energie. Proto byly navrženy 3 odlišné třídy zařízení.

Třída zařízení A je vhodná pro případ, že není k dispozici přívod napájení a je požadavek na sběr dat z čidla nebo zasílání zpráv k čidlu s časovou prodlevou. Komunikace je vždy zahájena ze strany čidla na základě definované události nebo na základě vymezeného časového intervalu. Čidlo má po odeslání zprávy nastaveny 2 intervaly trvající 2 s, během kterých přijímá zprávy. Na základě přijatých zpráv jsou vyvolány konkrétní akce (např. zapnuto/vypnuto nebo změnit interval odesílání zpráv). Příkladem použití může být měření spotřeby vody nebo plynu.

Třída zařízení B dokáže přijímat zprávy v jakémkoliv okamžiku nebo v závislosti na definované události. Rozdíl mezi třídou A a B je, že čidla v pravidelném intervalu aktivují svou radiovou část, ale k odeslání zprávy nedojde. To má za následek vyšší spotřebu elektrické energie. Tento režim provozu se nazývá koordinovaný vzorkovací poslech. Příkladem použití může být zabezpečení objektu.

Třída zařízení C je vhodná pro aplikace, kde není třeba řešit energetickou náročnost. Příjímá část je u čidla trvale aktivována, díky tomu může být zpráva ze sítě přijata v libovolném okamžiku. Příkladem použití může být chytrý elektroměr.

1.4 Narrow Band

NB – IoT patří k novým bezdrátovým úzkopásmovým technologiím LPWA, které jsou díky nízké energetické náročnosti a širokému pokrytí vhodné pro IoT aplikace. V České republice tuto technologii poskytuje Vodafone Czech Republic a.s. Na rozdíl od technologií LoRa a Sigfox pracuje Narrow band v licencovaném frekvenčním pásmu 800 MHz s využitím stávajících základnových stanic. Díky využití stávající infrastruktury bude nasazení této technologie rychlé a ekonomické. Šířka přenosového pásma je pouze 200 kHz (ekvivalentní k GSM). NB – IoT funguje v half duplex režimu, pomocí jednoho kanálu je možné komunikovat oběma směry, ovšem nikoliv současně. Oproti GSM a LTE má NB – IoT o 20 dB silnější signál, což umožní snazší prostupnost překážkami a zajistí lepší pokrytí.

2 Hardware

Obsah této kapitoly popisuje a specifikuje hardware, který byl použit pro realizaci měřicího zařízení společně s šasi, do které je uloženo. Schéma zapojení je v příloze 8.

2.1 Bezdrátové moduly pro Arduino

Výběr IoT bezdrátového modulu závisí především na druhu zařízení a způsobu jeho použití. V případě, že zařízení nebude trvale připojené do rozvodné sítě elektrické energie a bude napájeno pomocí baterií nebo obnovitelných zdrojů, je kladen velký důraz na nízkou spotřebu celého systému. Dalším důležitým kritériem při výběru bezdrátového modulu je vzdálenost, na kterou dokáže pracovat. Dosah se pohybuje od desítek metrů až po desítky kilometrů, ovlivněn je především tím, zdali se jedná o vnitřní, či venkovní použití. Mezi nejpoužívanější způsoby datových přenosů používaných pro platformu Arduino patří: Wi-Fi, Bluetooth, GSM a další RF komunikace.

Navrhované zařízení, jež monitoruje kvalitu ovzduší, bude napájeno bateriemi, tudíž je kladen velký důraz na spotřebu elektrické energie. Využívat se bude ve vnitřních i venkovních prostorech, proto je důležitý dosah bezdrátového modulu. Na základě těchto požadavků byly vybrány a popsány tyto relativně nové technologie IoT sítě, které využívají nový koncept LPWAN (Long Area Wide Area Network): Sigfox, LoRa a NarrowBand.

2.2 Arduino MKRFOX 1200

Na obr. 8 je deska Arduino MKRFOX 1200, která byla vybrána jako základní část pro realizaci zařízení, disponuje vlastnostmi desky Arduino Zero a konektivitou Sigfox. Díky tomu se stává ideálním nástrojem pro IoT aplikace s vysokými nároky na nízkou spotřebu energie. Deska je založena na mikročipu SAMD21 a modulu Sigfox ATA8520. Tabulka s podrobnými technickými specifikacemi je v příloze 1.

Provozní napětí desky je 3,3 V, stejně tak maximální napětí, které jsou schopny I/O piny tolerovat. Jakékoliv vyšší napětí může způsobit její nevratné zničení. Deska umožňuje napájet 5 V zařízení, ale v případě obousměrné komunikace je nutné správné posunutí napěťové úrovně.

2.2.1 Baterie, piny a LED diody

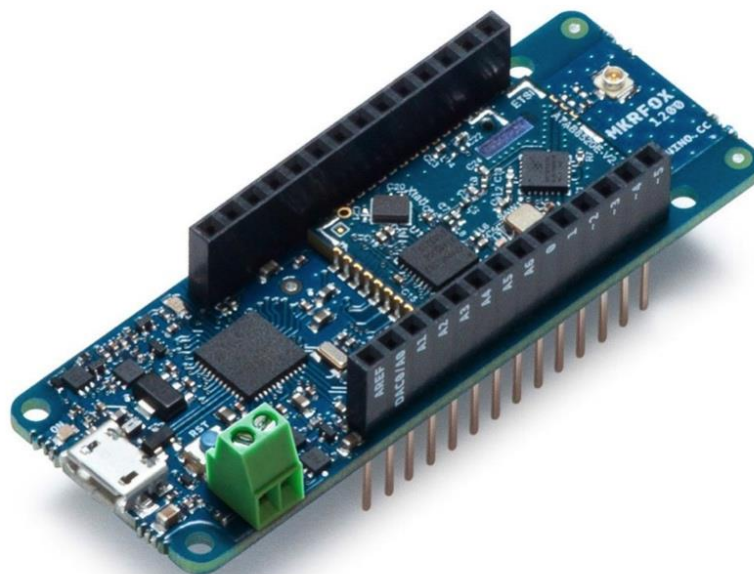
V případě, že má být deska napájena pomocí baterií (2xAA nebo AAA), je nutné tyto baterie připojit na svorkovnici pomocí šroubů. Pin, na který se připojí pozitivní polarita, se nachází blíže USB konektoru. Jmenovité napětí baterií musí být 1,5 V.

Vin je vstupní pin, který slouží k napájení desky pomocí 5V zdroje. Pokud je tento pin využíván, USB napájení je odpojeno. Použití *Vin* je jediný způsob, jak můžeme desku napájet 5 V bez použití USB.

Výstupní pin, jehož označení je *5V*, zajišťuje neregulovatelné výstupní napětí 5 V přímo z USB konektoru nebo *Vin* pinu.

Pin pod označením *Vcc* zajišťuje výstupní napětí 3,3 V, skrz napěťový regulátor na desce. Pokud jsou pro napájení desky použity baterie, je napětí z tohoto pinu ekvivalentní dvěma bateriím zapojeným v sérii.

Na desce se nacházejí dvě vestavěné LED diody. Zelená LED dioda svítí, pokud je deska napájena z USB konektoru nebo *Vin* pinu. V případě že desku napájíme pomocí baterií, zůstává LED dioda zhasnuta, čímž se minimalizuje energetická náročnost. Žlutá dioda je notifikační dioda, která bliká např. při nahrávání programu do Arduina nebo při Sigfox komunikaci.



Obr. 8 Arduino MKRFOX 1200

2.3 BME280 Weather Click

Pro snímání teploty, relativní vlhkosti a atmosférického tlaku byla použita deska BME280 Weather Click, která je znázorněna na obr. 9. Základem této desky je precizní senzor s nízkou spotřebou elektrické energie BME280 od firmy BOSCH.

Provozní napětí desky je 3,3 V (stejně jako u Arduina MKRFOX 1200). Komunikace s mikrokontrolérem zajišťuje rozhraní I2C nebo SPI. Tabulka s podrobnými technickými specifikacemi je v příloze 2.

Pro komunikace s Arduinem MKRFOX 1200 bylo použito rozhraní I2C. Jedná se o obousměrné dvou vodičové datové propojení. Sběrnice používá sériovou datovou linku SDA (Synchronous Data) a hodinový signál SCL (Synchronous Clock). Při aktivním přenosu jsou na SDA odesílány datové bity za předpokladu, že se logická úroveň SDA může měnit pouze v případě, kdy je SCL v úrovni low. Výjimka je pouze v případě zahájení nebo ukončení komunikace.



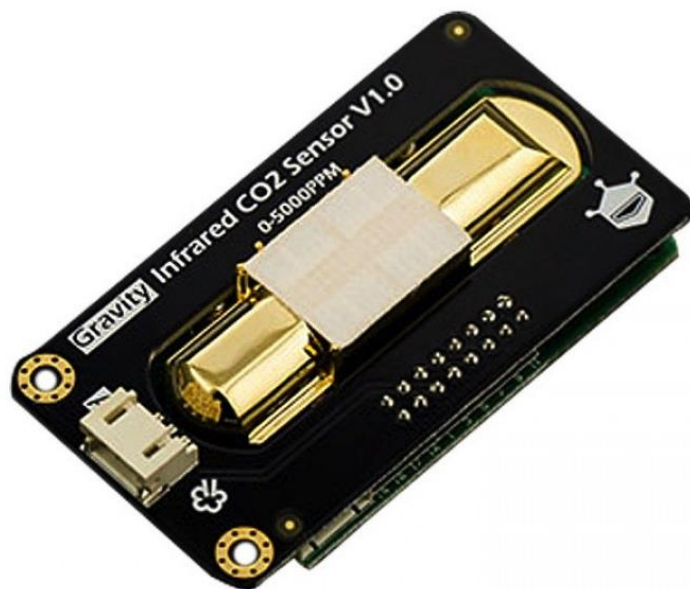
Obr. 9 BME280 Weather Click

2.4 Gravity Infrared CO2

Na obr. 10 je vysoce precizní analogový infračervený senzor CO₂ od firmy DFRobot ze série Gravity, jež byl použit pro detekci hladiny CO₂ v okolním prostředí.

Jeho provozní napětí je 5 V. Senzor je antikorozní, voděodolný a zamezuje srážení vody. Nízké energetické nároky společně s dlouhou životností z něj dělají výborný nástroj pro IoT aplikace. Mezi jeho další výhody patří stabilita a integrovaná teplotní kompenzace. Tabulka s podrobnými technickými specifikacemi je v příloze 3.

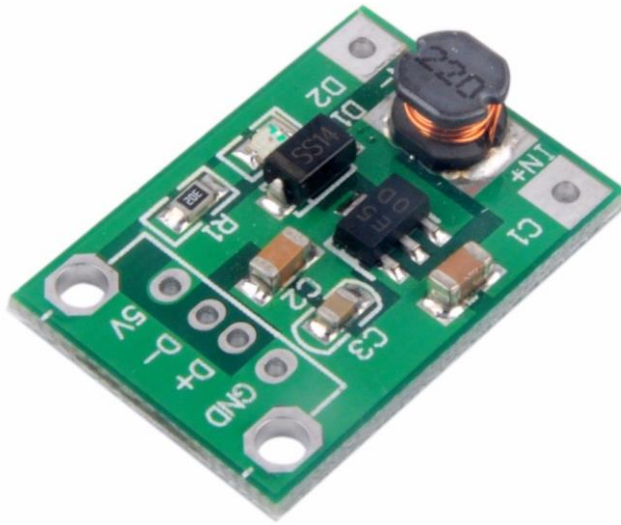
Čidlo pracuje na měření útlumu infračerveného záření ve vzduchu. Skládá se ze zdroje záření, světlovodné trubice a infračerveného detektoru s příslušným filtrem. Signál dopadající na detektor se zesílí a následně vyhodnotí útlum, ze kterého se určí aktuální hodnota koncentrace CO₂.



Obr. 10 Gravity Infrared CO2

2.5 Step-UP měnič 5 V

Step-UP měnič napětí zobrazený na obr. 11 byl použit jako stabilizovaný zdroj pro napájení desky Arduino MKRFOX 1200. Na vstup zdroje musí být připojeno stejnosměrné napětí v rozsahu 1 – 5 V. Na výstupu zdroje je 5 V stejnosměrného napětí a maximální možný odebíraný proud činí 500 mA. Jako vstup pro měnič byly použity 3 x AA lithiové baterie 3,6 V s kapacitou 2600 mAh zapojené paralelně. Tabulka s kompletními technickými specifikace je v příloze 4.



Obr. 11 Step-UP měnič 5 V

2.6 BTS117 T0220

Unipolární tranzistor BTS117 T0220 plní v měřicím zařízení funkci spínacího prvku. Jeho podrobné technické specifikace jsou v příloze 5. Analogový senzor Gravity Infrared CO₂ se nejvíce podílí na spotřebě elektrické energie měřicího zařízení a je připojen na pin 5 V na desce Arduino MKRFOX 1200. Tento pin má na svém výstupu neustále 5 V i v případě, že je Arduino přepnuto do hlubokého spánku a nelze ho softwarově vypnout. Z tohoto důvodu bylo nutné tento pin spínat a rozepínat externě pomocí tranzistoru, protože neustálý běh senzoru Gravity Infrared CO₂ by díky své spotřebě téměř znemožnil, aby bylo možné zařízení provozovat na bateriích.

Spínání je realizováno pomocí vestavěných funkcí Arduina. Na gate tranzistoru je přiveden signál z digitálního pinu 1. Jako první je nutné definovat tento pin jako výstup.

```
pinMode(1, OUTPUT);
```

Dalším krokem je sepnutí toho pinu, čímž dojde i k sepnutí tranzistoru a aktivaci senzoru Gravity Infrared CO₂.

```
digitalWrite(1, HIGH);
```

Po změření potřebných veličin je před uspáním Arduina nutné tento pin odepnout a tím rozepnout i obvod s tranzistorem

```
digitalWrite(1, LOW);
```

2.7 Šasi

Kompletní měřicí zařízení je uložené ve dvoudílné plastové šasi z PVE (Polyvinylether). Šasi funguje jako uzavíratelná krabička, jež je složena ze dvou dílů. Víko krabice je opatřeno větracími průduchy pro správný chod senzorů a vkládá se do kolejnic druhého dílu, ve kterém je průchod pro anténu. Šasi byla vytištěna na 3D tiskárně, její návrh byl vytvořen v softwaru SolidWorks. Kompletní technický výkres je v příloze 7. Na obr. 12 je fotografie kompletní vytištěné šasi společně s implementovanou anténou.



Obr. 12 Šasi

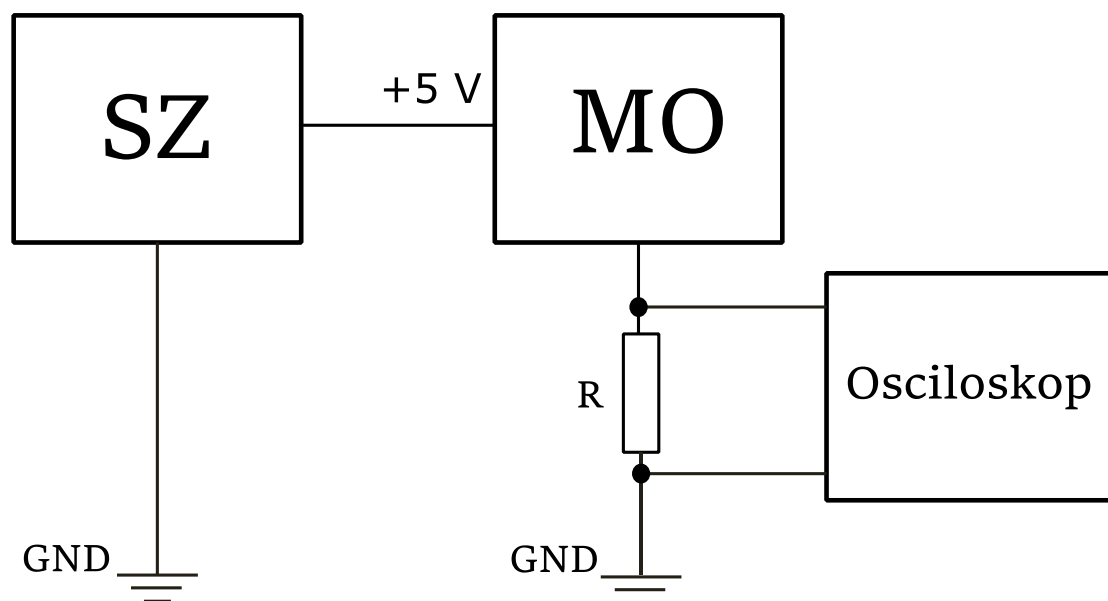
2.8 Spotřeba elektrické energie

Spotřeba elektrické energie měřicího zařízení byla měřena pomocí osciloskopu, který měřil průběh napětí na rezistoru známé hodnoty, který uzavíral elektrický obvod se zemí. Blokové schéma zapojení měření je zobrazeno na obr. 13 (SZ = stabilizovaný zdroj, MO = měřený objekt). Díky znalosti velikosti elektrického odporu rezistoru je snadné naměřené napětí přepočítat na proud pomocí Ohmova zákona. Pro měření byl zvolen rezistor, který měl katalogovou hodnotu 1 Ω , naměřená hodnota pomocí číslicového multimetru byla 0,84 Ω . Změřeno bylo spuštění jednotky, špičkový proud odebíraný senzorem Gravity Infrared CO2, spotřeba elektrické energie potřebné na odeslání naměřených dat, spotřeba elektrické energie celého měřicího cyklu a spotřeba elektrické energie v režimu spánku.

$$I = \frac{U}{R} \text{ (A)}$$

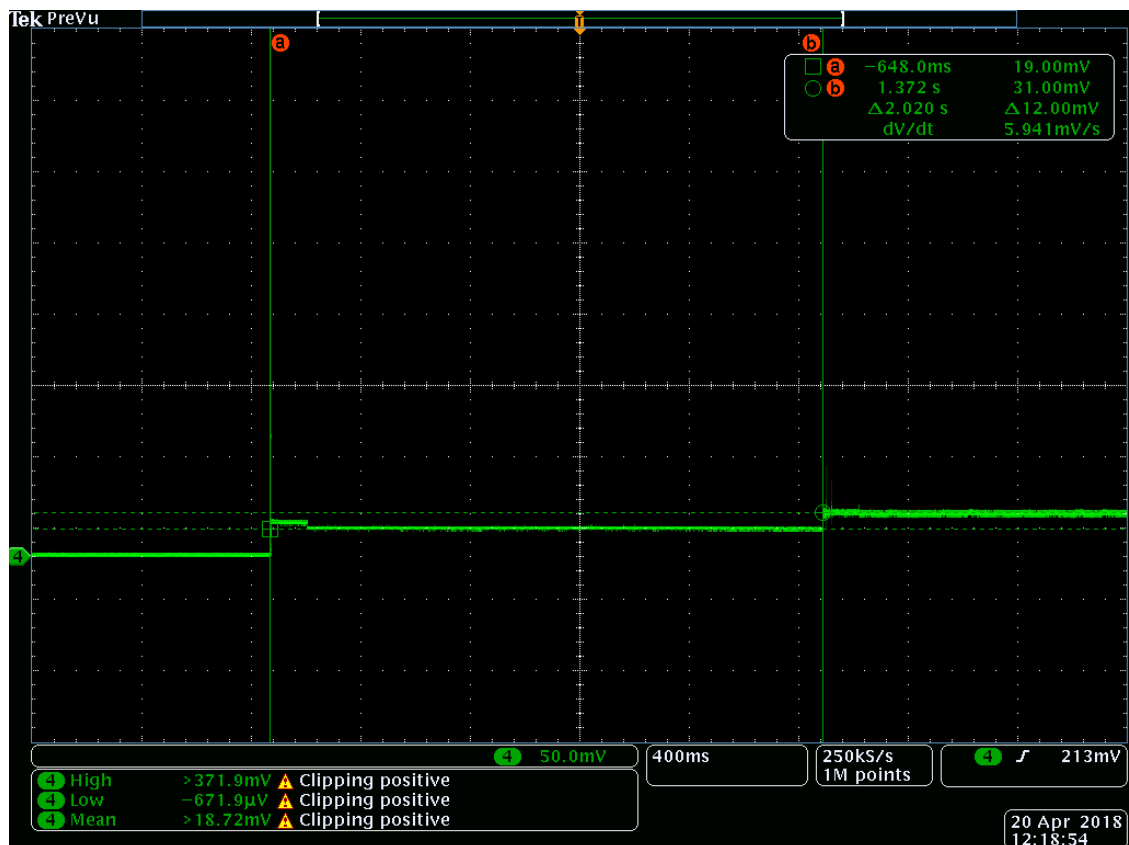
Seznam měřících přístrojů:

- Osciloskop - Tektronix MDO4034C
- Číslicový multimetr – KEYSIGHT U1253B



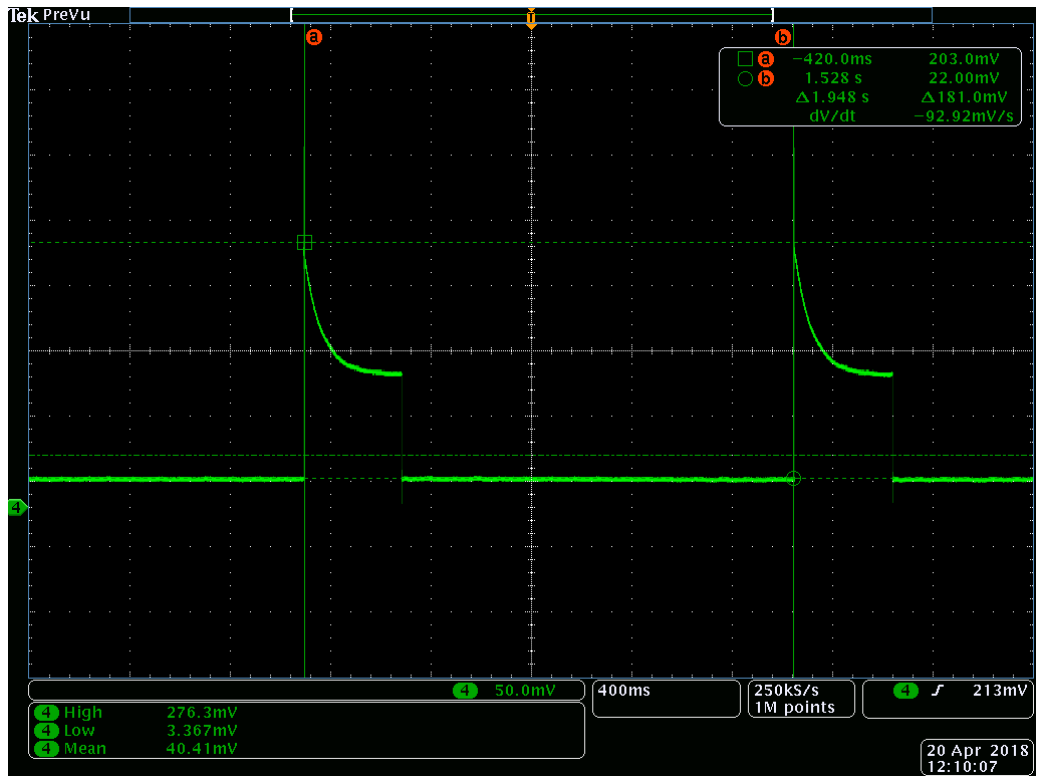
Obr. 13 Blokové schéma zapojení měření

Jako první bylo změřeno spouštění jednotky těsně po připojení ke stabilizovanému zdroji. Nejdříve naběhne deska Arduino MKRFOX 1200, která odebírá proud 23 mA. Během dalších dvou sekund naběhnou ostatní senzory a odebíraný proud se zvýší na hodnotu 38 mA. Snímek z osciloskopu je zobrazen na obr. 14.

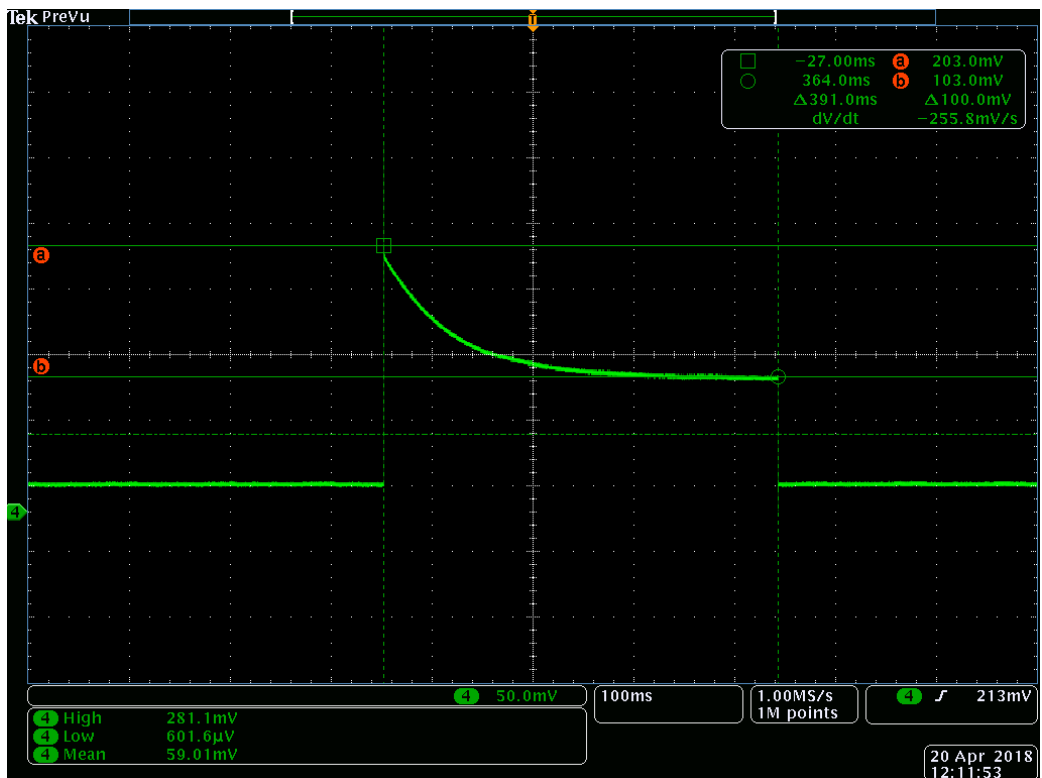


Obr. 14 Spuštění jednotky

Senzor Gravity Infrared CO2 odebírá z celého zařízení největší proud. Aby senzor měřil správně, je nutné jeho předeřhřátí, které trvá 180 s. Během této doby odebírá jednotka v periodě 1,9 s špičkový proud až 248 mA. Perioda dvou špičkových pulsů je zobrazena na snímku z osciloskopu viz obr. 15. Průměrná hodnota odebíraného proudu během jedné periody je 49 mA. Na obr. 16 je snímek jednoho pulsů, doba jeho trvání je 0,4 s a průměrný odebíraný proud dosahuje hodnoty 78 mA.



Obr. 15 Perioda špičkového pulsu



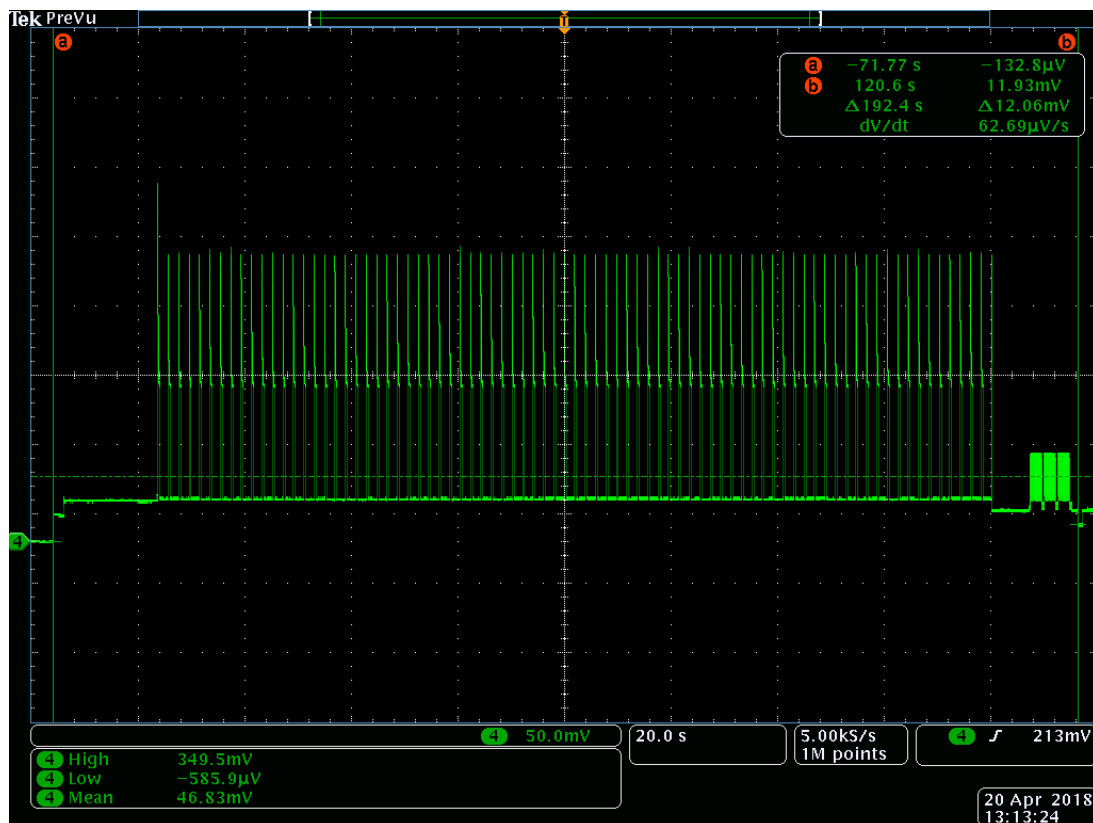
Obr. 16 Puls

Po naměření potřebných veličin jsou data odeslána do Sigfox cloudu. Datový přenos trvá 8 s. Hodnota špičkového proudu je 83 mA. Průměrný odebíraný proud je 60 mA. Na obr. 17 je snímek datového přenosu.



Obr. 17 Datový přenos

Na obr. 18 je snímek celého měřicího cyklu, který trvá 192,4 s. Průměrná hodnota odebíraného proudu činí 57 mA. Cyklus se skládá ze startu desky Arduino MKRFOX 1200, následnému náběhu senzorů Gravity Infrared CO₂ a BME280 Weather Click. Poté se přehřívá senzor Gravity Infrared CO₂ a probíhá měření environmentálních parametrů prostředí. Následuje datový přenos a přechod zařízení do režimu spánku. V režimu spánku je odebíraný proud 13 mA (při odpojení LED diod dosahuje odebíraný proud pouze 1 mA). Odhad výdrže zařízení při odpojení všech LED diod a použití 3 x Lithiové baterie 2600 mAh popisuje tab. 2. Reálná výdrž bude ovlivněna dalšími parametry (např. vlhkost, teplota).



Obr. 18 Datový přenos

Četnost měření	Výdrž zařízení na bateriích
každých 15 min	24 dní
každých 60 min	78 dní
každých 180 min	156 dní

Tab. 2 Odhad výdrže zařízení na bateriích

3 Programové prostředky

V této kapitole je charakterizováno vývojové prostředí Arduino IDE, ve kterém byl napsán software, jenž ovládá měřící zařízení a zajišťuje komunikace se Sigfox cloudem. Dále je popsáno nastavení cloudové služby Sigfox backendu a odesílání dat do jednotlivých IoT platform a jejich následná vizualizace.

3.1 Arduino IDE

Integrované vývojové prostředí Arduino IDE (integrated development environment) je složeno z textového editoru, do kterého se píše samotný kód v jazyce Wiring, kde má podobu frameworku v jazyce C++. Pod textovým polem se nachází textová konzole, která zobrazuje textové výstupy z vývojového prostředí, včetně chybových zpráv a dalších informací. Textová konzole funguje jako zpětná vazba při ukládání a nahrávání programu do desky. V pravém dolním rohu je zobrazena aktuální deska s použitým sériovým portem. Vývojové prostředí Arduino IDE je zobrazeno na obr. 19.

Softwaru, který napíšeme pomocí Arduino IDE, se říká sketch (návrh) a ukládá se do souboru s koncovkou .ino. Návrh následně musí projít verifikací, při které se ověřuje, zdali se v kódu nevyskytují chyby. Po úspěšné verifikaci stačí vybrat příslušný sériový port a již je možné sketch do desky nahrát. Po stisknutí tlačítka nahrát se deska automaticky restartuje a začne s nahráváním pomocí bootloaderu, které je indikováno blikáním LED diod TX a RX. Textová konzole následně zobrazí zprávu o úspěšném nahrání, nebo vypíše chybu.

Program, jenž je nahrán v desce Arduino MKRFOX 1200, obsluhuje komunikaci s jednotlivými senzory, které snímají okolní prostředí. Jeho další funkcí je odesílání data do Sigfox cloudu. Celý tento sketch je včetně veškerých komentářů na CD, které je součástí práce.

```

//////////////////////////////////// knihovny //////////////////////////////////////
#include "SigFox.h"
#include "ArduinoLowPower.h"
#include <Wire.h>
#include "Seeed_BME280.h"
#include "conversions.h"
//////////////////////////////////// definice proměnné //////////////////////////////////////
float t=0;
//////////////////////////////////// inicializace senzorů //////////////////////////////////////
int sensorIn = A0;
BME280 bme280;

#define SLEEPTIME 15 * 60 * 1000 //spánek
#define DEBUG true

///// struktura zprávy pro sigfox //////////////////////////////////////
typedef struct __attribute__((packed)) sigfox_message {
  int16_t moduleTemperature;
  int16_t bmeTemperature;
  uint16_t bmeHumidity;
  uint16_t bmePressure;
  uint16_t gravityCo2;
  uint8_t lastMessageStatus;
} SigfoxMessage;

SigfoxMessage msg;

////////////////////////////////////

```

Kompilace ukončena.

UPOZORNĚNÍ: Kategorie 'Sensor' v knihovně Adafruit APDS9960 Library je neplatná. Nastaveno na 'Uncategorized'
Projekt zabírá 42608 bytů (16%) úložného místa pro program. Maximum je 262144 bytů.

12 Arduino MKRFox1200 na /dev/cu.usbmodemFD1211

Obr. 19 Arduino IDE

3.2 Sigfox backend

Data odeslaná zařízením do Sigfox cloudu společně s informacemi o zařízeních, můžeme přímo sledovat v Sigfox backendu, který je zobrazen na obr. 20. Backend také umožňuje následné přeposílání přijatých dat pomocí individuálních „Callbacks“ do emailu či IoT platformy, jako např. Microsoft Azure IoT, nebo Amazon Web Services. Další možností je definice vlastního nastavení (Custom Callback). Při jeho tvorbě se otevře okno, do kterého se zadávají parametry odesílání dat:

- **Type** = druh odeslaných dat, DATA – data odeslaná zařízením, SERVICE – údaje služby, ERROR – chybové stavy; UPLINK – data odeslaná ze zařízení, BDIR – obousměrná komunikace

- **Channel** = použitý komunikační kanál, URL – volání vašeho RESET API okamžitě po doručení dat, BATCH URL – dávkové zpracování dat odesílaných na vaše REST API, EMAIL - odesílání dat na zadaný email
- **Custom payload config** = parametr, do kterého se musí vypsát a přesně seřadit proměnné (oddělené mezerou) podle struktury zprávy, která byla odeslána ze zařízení do Sigfox cloudu. Zároveň se každé proměnné musí přiřadit příslušný datový formát, vedle toho parametru je nápověda, která pomůže s vhodným výběrem. Tímto krokem byly přidány vlastní proměnné, se kterými je dále možné pracovat. Tento parametr bude pro všechny Custom Callbacks stejný.

Custom payload config:

moduleTemp::int:16:little-endian

bmeHumidity::uint:16:little-endian

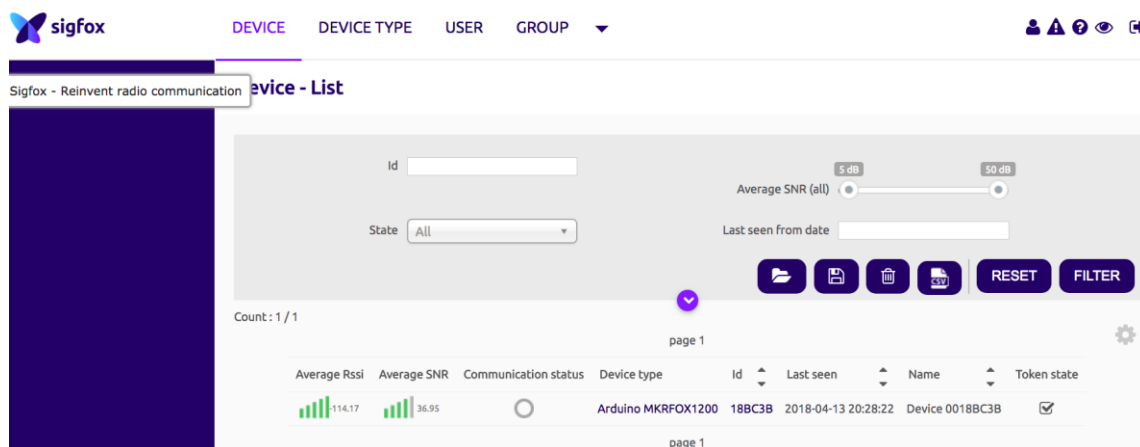
gravityCo2::int:16:little-endian

bmeTemperature::int:16:little-endian

bmePressure::int:16:little-endian

lastMsg::uint:8

Další údaje jsou vyplněny dle zvoleného komunikačního kanálu.



Obr. 20 Sigfox backend

3.2.1 Custom Callback – Email

Jako první bude popsán Custom Callback, který přeposílá data ze zařízení, přímo do nastavené emailové schránky, ihned po jejich příchodu do Sigfox cloudu.

Parametr **Type** se nastaví na DATA a jeho druhá část na UPLINK. U parametru **Channel** se vybere možnost EMAIL. Velice důležitý je parametr **Custom payload config**, který je pro všechny Custom Callbacks stejný. Do parametru **Recipient** se vyplní emailová adresa, na kterou se má zpráva odeslat. Je možné vyplnit více emailových adres najednou. Parametr **Subject** je prezentován v emailu jako jeho předmět. Posledním parametrem je **Message**, do kterého vyplníme zprávu, která se odešle na emailovou adresu. Do zprávy je možné přidat proměnné ze seznamu nebo také proměnné, které byly definovány v **Custom payload config**.

Message:

Temperature: {customData#bmeTemperature} (°C)

Humidity: {customData#bmeHumidity} (%)

Pressure: {customData#bmePressure} (kPa)

CO2: {customData#gravityCo2} (ppm)

Na obr. 21 je znázorněno nastavení kompletní Custom Callback pro email přímo ze Sigfox backendu .

The screenshot shows the configuration interface for a Custom Callback of type 'EMAIL'. The 'Type' is set to 'DATA' and the 'Channel' to 'EMAIL'. The 'Send duplicate' checkbox is unchecked. The 'Custom payload config' field contains the string: `moduleTemp::int:16:little-endian bmeTemperature::int:16:little-endian bmeHumidity::uint:16:little-endian`. The 'Recipient' field contains the email address `v.rehorik@gmail.com`. The 'Subject' field contains `Arudion Sigfox`. The 'Message' field contains the following text: `Temperature: {customData#bmeTemperature} (°C)`, `Humidity: {customData#bmeHumidity} (%)`, `Pressure: {customData#bmePressure} (kPa)`, and `CO2: {customData#gravityCo2} (ppm)`. Below the recipient field, there is a note: 'Multiple emails allowed separated by comma, semicolon or new line'. Below the subject field, there is a note: 'Subject syntax: Subject with device {device}' and 'Message syntax: Message containing time {time}, key1 {var1}, key2 {var2}...'. Below the message field, there is a note: 'Available variables: device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber' and 'Custom variables: customData#moduleTemp, customData#bmeTemperature, customData#bmeHumidity, customData#bmePressure, customData#gravityCo2, customData#lastMsg'.

Obr. 21 Custom Callback – Email

3.2.2 Custom Callback – Thingspeak

Další Custom Callback zajišťuje komunikaci s IoT platformou Thingspeak, jež umožňuje sběr a uchovávání dat ze senzorů. Velkou výhodou této platformy je možnost analýzy a vizualizace dat pomocí Matlabu.

Jako první se nastaví parametr **Type** na DATA a jeho druhá část na UPLINK. Další parametrem je **Channel**, který se nastaví na URL. Nastavení **Custom payload config** je ve všech případech stejný. Do parametru **Url pattern** se vloží API key, který byl vygenerován Thingspeak platformou. Následně se přidají jednotlivé proměnné, jež chceme do platformy odesílat.

Url pattern:

https://api.thingspeak.com/update?api_key=QQSUUGMY6G3IKDBV&field1={customData#bmeTemperature}&field2={customData#bmeHumidity}&field3={customData#bmePressure}&field4={customData#gravityCo2}

Posledním nastavovaným parametrem je **Use HTTP Method**, kde je zvolena metoda GET. Custom Callback ze Sigfox backendu pro Thingspeak platformu je na obr. 22.

The screenshot shows the configuration interface for a Custom Callback in Thingspeak. The 'Type' is set to 'DATA' and 'UPLINK'. The 'Channel' is set to 'URL'. The 'Send duplicate' checkbox is unchecked. The 'Custom payload config' is set to 'moduleTemp::int:16:little-endian bmeTemperature::int:16:little-endian bmeHumidity::uint:16:little-endian'. The 'Url pattern' is 'https://api.thingspeak.com/update?api_key=QQSUUGMY6G3IKDBV&field1={customData#bmeTemperature}'. The 'Use HTTP Method' is set to 'GET'. The 'Send SNI' checkbox is unchecked. The 'Headers' section is empty.

Obr. 22 Custom Callback - Thingspeak

3.2.3 Custom Callback – Thinger

Jako další bude popsán Custom Callback, který zajišťuje komunikaci měřícího zařízení s IoT platformou Thinger, jež umožňuje vizualizovat a archivovat data ze senzorů.

Prvním volitelným parametrem je Type. Jeho první část je nastavena na DATA a druhá část na UPLINK. Druhým v pořadí je parametr Channel, který je nastaven na URL. Nastavení Custom payload config je ve všech případech stejné. Do parametru Url pattern je vložena webová adresa, která je vygenerovaná platformou Thingier.

Url pattern:

<https://api.thingier.io/v1/users/vrehorik/buckets/mkrfox1200/data>

Následně je nastaveno **Use HTTP Method** na možnost POST. Po tomto kroku se nastavily obě části parametru **Headers**. Do prvního pole vyplníme Authorization a do druhého kód, který nám vygenerovala platforma Thingier.

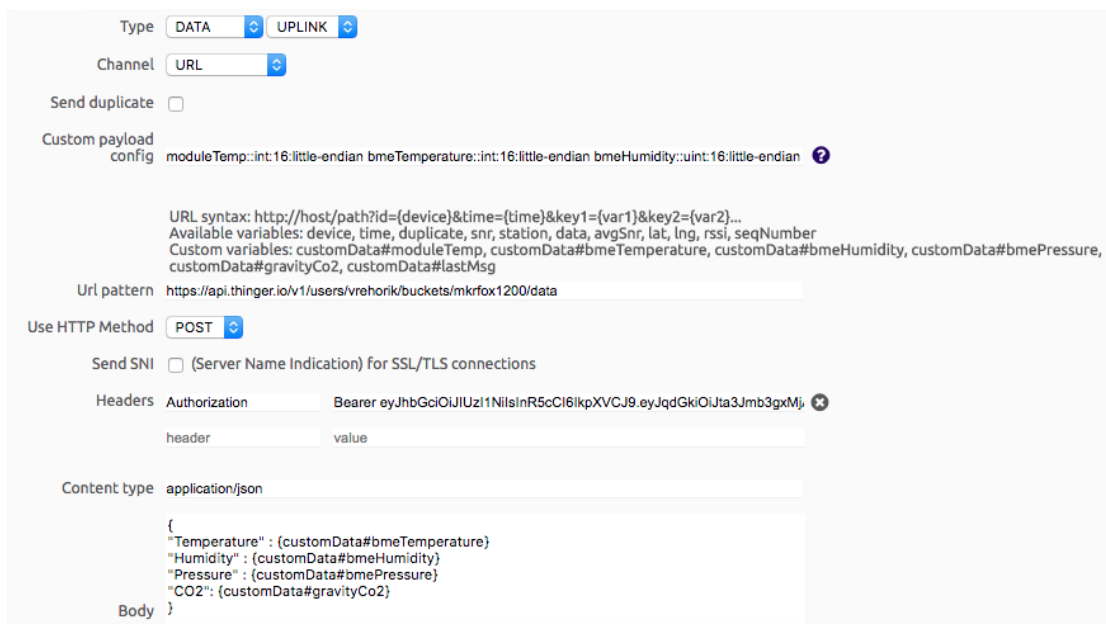
Headers:

Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiJta3Jmb3gxMjAwIiwidXNy
joidnJlaG9yaWsisifQ.pHDos5vPPEQRC_WRJG2I7V70iQ4nQM1bELchpgNcR4

Do pole parametru **Content type** vložíme application/json. JavaScript Object Notation (JavaScriptový objektový zápis, JSON) je způsob zápisu dat, určený pro přenos dat. Posledním je parametr **Body**, jež obsahuje zprávu, která je odeslána do platformy Thingier. Na obr. 23 je znázorněn kompletní Custom Callback - Thingier ze Sigfox backendu.

Body:

```
{  
  "Temperature" : {customData#bmeTemperature}  
  "Humidity" : {customData#bmeHumidity}  
  "Pressure" : {customData#bmePressure}  
  "CO2": {customData#gravityCo2}  
}
```



Type

Channel

Send duplicate

Custom payload config `moduleTemp::int:16:little-endian bmeTemperature::int:16:little-endian bmeHumidity::uint:16:little-endian` ?

URL syntax: `http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...`
 Available variables: `device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber`
 Custom variables: `customData#moduleTemp, customData#bmeTemperature, customData#bmeHumidity, customData#bmePressure, customData#gravityCo2, customData#lastMsg`

Url pattern `https://api.thinger.io/v1/users/vrehorik/buckets/mkrfox1200/data`

Use HTTP Method

Send SNI (Server Name Indication) for SSL/TLS connections

Headers

Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiJta3Jmb3gxMjI...
header	value

Content type

Body

```
{
  "Temperature": {customData#bmeTemperature}
  "Humidity": {customData#bmeHumidity}
  "Pressure": {customData#bmePressure}
  "CO2": {customData#gravityCo2}
}
```

Obr. 23 Custom Callback – Thinger

3.2.4 Custom Callback – Losant

Poslední Custom Callback umožňuje předávat data z měřicího zařízení do IoT platformy Losant. Tato platforma umožňuje vizualizaci a archivaci naměřených dat. Tato data je zároveň možné následně upravovat pomocí Java skriptů.

První volitelný parametr Type je nastaven na Data a jeho druhá část na BDIR. Jako další se u parametru Channel vybere URL. Custom payload config má nastavení stejné jako ve všech předchozích případech. Url pattern obsahuje webovou adresu společně s kódem vygenerovaným platformou Losant.

Url pattern:

<https://triggers.losant.com/webhooks/EYmkYJkdGPCWBYa2bXWWILob6Mx>

Dalším volitelným parametrem je **Use HTTP Method**, u kterého je zvolena možnost POST. Do pole parametru **Content type** vložíme application/json. JavaScript Object Notation (JavaScriptový objektový zápis, JSON) je způsob zápisu dat, určený pro přenos dat. Jako poslední je parametr **Body**, který obsahuje zprávu, která je odeslána do platformy Losant. Na obr. 24 je znázorněn kompletní Custom Callback - Losant ze Sigfox backendu.

Body:

```
{
  "Temperature" : {customData#bmeTemperature}
  "Humidity" : {customData#bmeHumidity}
  "Pressure" : {customData#bmePressure}
  "CO2": {customData#gravityCo2}
}
```

Type: DATA | BIDIR

Channel: URL

Send duplicate:

Custom payload config: moduleTemp::int:16:little-endian bmeTemperature::int:16:little-endian bmeHumidity::uint:16:little-endian

URL syntax: http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...
 Available variables: device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber, ack, longPolling
 Custom variables: customData#moduleTemp, customData#bmeTemperature, customData#bmeHumidity, customData#bmePressure, customData#gravityCo2, customData#lastMsg

Url pattern: https://triggers.losant.com/webhooks/EYmkYJkdpcwbya2bXWWILob6Mx

Use HTTP Method: POST

Send SNI: (Server Name Indication) for SSL/TLS connections

Headers table:

header	value
header	value

Content type: application/json

Body:

```
{
  "Temperature" : {customData#bmeTemperature}
  "Humidity" : {customData#bmeHumidity}
  "Pressure" : {customData#bmePressure}
  "CO2": {customData#gravityCo2}
}
```

Obr. 24 Custom Callback – Losant

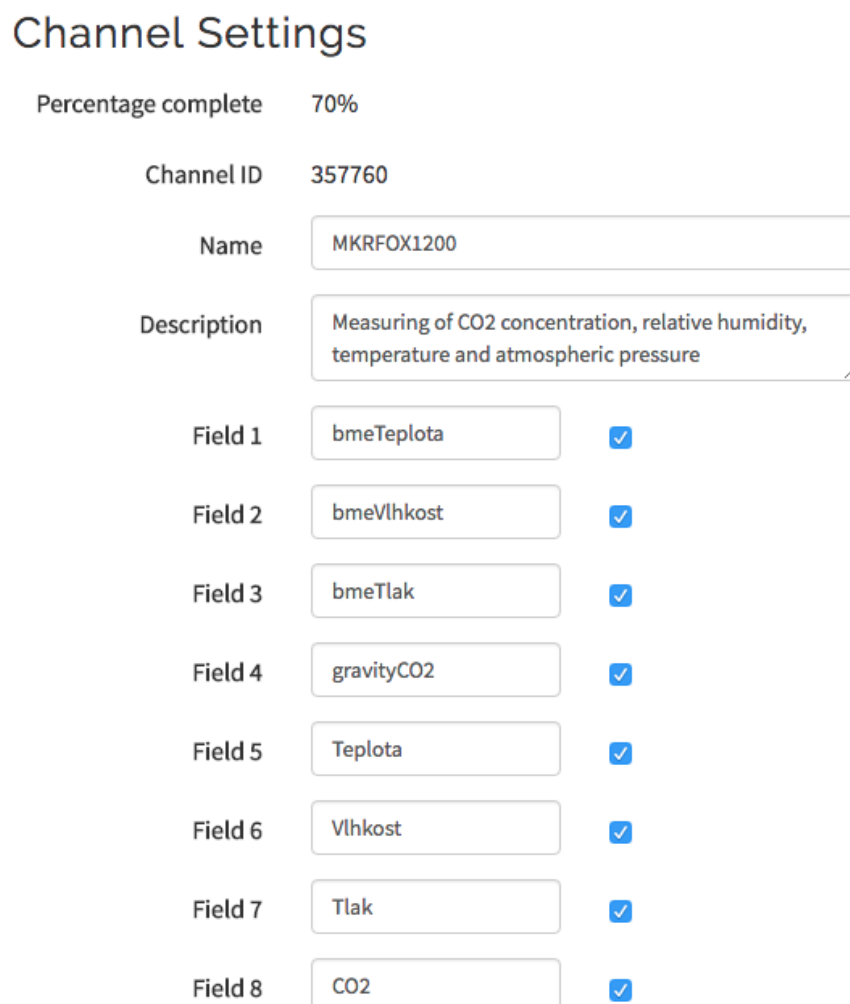
3.2.5 IoT platformy

V této části je popsáno detailní nastavení jednotlivých IoT platforem (Thingspeak, Thinger, Losant), do kterých jsou odesílána data přímo ze Sigfox backendu. Tyto platformy umožňují vizualizaci, archivaci a případnou editaci přijatých dat. Dále jsou uvedeny jejich cenové tarify.

3.2.6 Thingspeak

Thingspeak je IoT analytická platforma, díky které je možné online vizualizovat a následně archivovat data z měřicího zařízení. Zároveň umožňuje využití Matlabu při práci s daty.

Jako první věc po přihlášení do Thingspeaku je nutné vytvořit nový komunikační kanál. V hlavní nabídce se vybere možnost „Channels“, dále „My Channels“ a poté „New Channel“. Na obr. 25 je znázorněno nastavení použitého kanálu. Pro každou veličinu jsou vytvořena dvě samostatná pole z důvodu, že první obsahuje surová data a druhé data po úpravě pomocí Matlab kódu.



Percentage complete	70%
Channel ID	357760
Name	<input type="text" value="MKRFOX1200"/>
Description	<input type="text" value="Measuring of CO2 concentration, relative humidity, temperature and atmospheric pressure"/>
Field 1	<input type="text" value="bmeTeplota"/> <input checked="" type="checkbox"/>
Field 2	<input type="text" value="bmeVlhkost"/> <input checked="" type="checkbox"/>
Field 3	<input type="text" value="bmeTlak"/> <input checked="" type="checkbox"/>
Field 4	<input type="text" value="gravityCO2"/> <input checked="" type="checkbox"/>
Field 5	<input type="text" value="Teplota"/> <input checked="" type="checkbox"/>
Field 6	<input type="text" value="Vlhkost"/> <input checked="" type="checkbox"/>
Field 7	<input type="text" value="Tlak"/> <input checked="" type="checkbox"/>
Field 8	<input type="text" value="CO2"/> <input checked="" type="checkbox"/>

Obr. 25 Nastavení kanálu Thingspeak

Po vytvoření nového komunikačního kanálu Thingspeak automaticky vygeneruje API klíče pro zápis a čtení dat, které jsou znázorněny na obr. 26. API klíč pro zápis dat je nutné

vyplnit do parametru Url pattern v nastavení Sigfox backendu pro Custom callback - Thingspeak.

Write API Key

Key

Read API Keys

Key

Note

Obr. 26 API klíče Thingspeak

Dalším krokem je úprava surových dat pomocí Matlab kódu a jejich následná vizualizace. V hlavním menu se vybere možnost „Apps“ dále „Matlab analysis“ a poté „New“. Pod tímto odstavcem je kompletní kód včetně příslušných komentářů, které jsou vždy za znaménkem „%“.

```
% Channel ID (čtení)
```

```
readChannelID = [357760];
```

```
% Read API Key
```

```
readAPIKey = 'TCBLIRRTV47900RI';
```

```
% Channel ID (zápis)
```

```
writeChannelID = [357760];
```

```
% Write API Key
writeAPIKey = 'QQUUGMY6G3IKDBV';

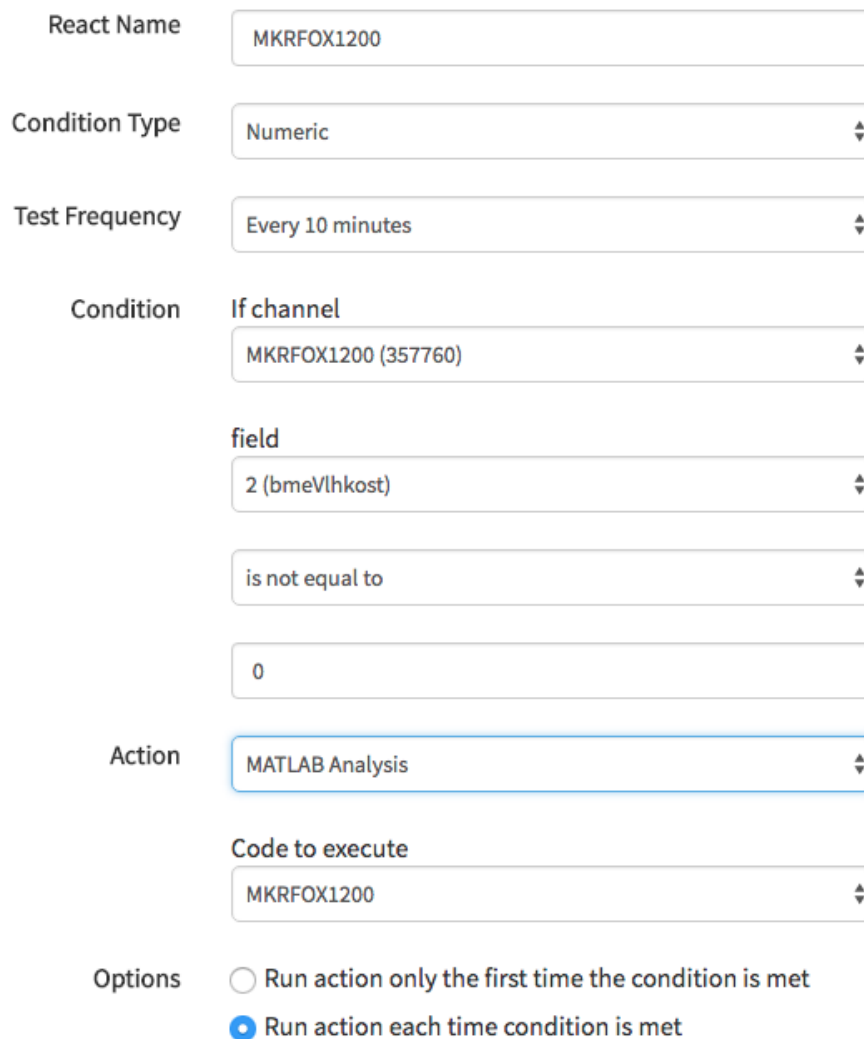
% Čtení dat
data = thingSpeakRead(readChannelID, 'ReadKey', readAPIKey, 'OutputFormat', 'table');
analyzedData = data;
data

% Analýza dat
INT16_t_MAX = 32767;
UINT16_t_MAX = 65536;

analyzedData.(Teplota) = data.(bmeTeplota) / INT16_t_MAX * 120 ;
analyzedData.(Vlhkost) = data.(bmeVlhkost) / UINT16_t_MAX * 110 ;
analyzedData.(Tlak) = data.(bmeTlak) / UINT16_t_MAX * 200 ;
analyzedData.(CO2) = data.(gravityCO2) / UINT16_t_MAX * 10000 ;
analyzedData

% Zápis dat
thingSpeakWrite(writeChannelID, analyzedData, 'WriteKey', writeAPIKey);
```

Posledním krokem je nastavení pravidelného spouštění Matlab kódu. V hlavní nabídce se vybere možnost „Apps“, poté „React“ a následně „New React“. Vhodné nastavení Reactu, které je na obr. 27, zajistí spuštění Matlab kódu vždy, když z měřicího zařízení dorazí nová data. Každých deset minut je testovaná podmínka, která je splněna pokud dorazí data ze senzorů a následně dojde ke spuštění kódu. React lze mimo jiné také využít na funkce ThingHTTP a ThingTweet.



React Name: MKRFOX1200

Condition Type: Numeric

Test Frequency: Every 10 minutes

Condition: If channel
MKRFOX1200 (357760)

field
2 (bmeVlhkost)

is not equal to

0

Action: MATLAB Analysis

Code to execute
MKRFOX1200

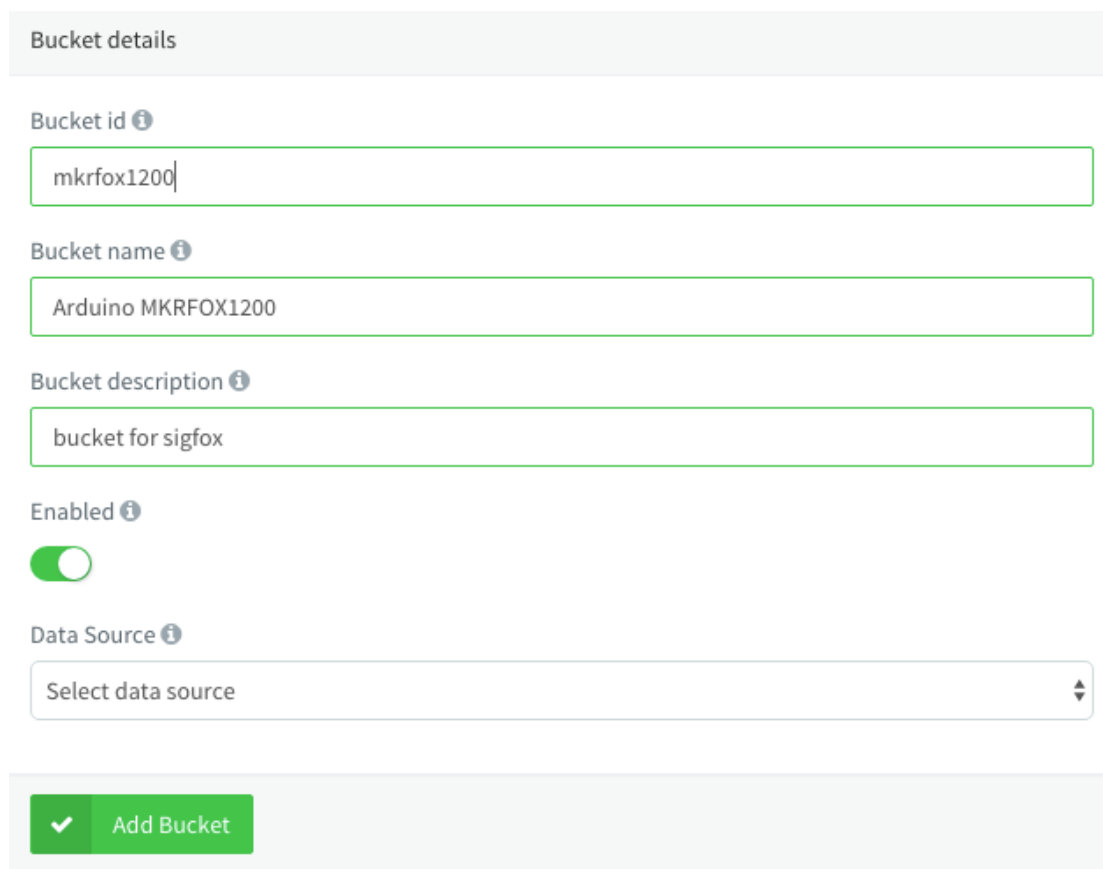
Options:
 Run action only the first time the condition is met
 Run action each time condition is met

Obr. 27 Thingspeak – React

3.2.7 Thinger

Jako další bude popsáno nastavení IoT platformy Thinger. Tato platforma umožňuje snadnou komunikaci se Sigfox backendem a následnou online vizualizaci a archivaci dat z měřicího zařízení.

Prvním krokem je vytvoření nového datového úložiště, do kterého budou ukládána data z měřicího zařízení. V hlavním menu se vybere možnost „Data Buckets“ a poté „Add Bucket“. Data Bucket také umožňuje export naměřených dat do formátu CSV, ARFF a Json. Na obr. 28 je nastavení nového datového úložiště.



The image shows a web interface for configuring a 'Data Bucket' in Thingiverse. The interface is titled 'Bucket details' and contains several input fields and a toggle switch. The 'Bucket id' field contains 'mkrfox1200'. The 'Bucket name' field contains 'Arduino MKRFOX1200'. The 'Bucket description' field contains 'bucket for sigfox'. The 'Enabled' toggle switch is turned on. The 'Data Source' dropdown menu is set to 'Select data source'. At the bottom, there is a green button with a checkmark and the text 'Add Bucket'.

Obr. 28 Thinger – Data Bucket

Druhým krokem je vytvoření nového přístupového tokenu. V hlavní nabídce se vybere možnost „Access Tokens“ a posléze „Add Token“. Při tvorbě přístupového tokenu se volí datové úložiště, do kterého bude možné zapisovat data z měřicího zařízení. Zároveň se automaticky vygeneruje přístupový kód, který je nutné vložit do Custom callback – Thinger v Sigfox backendu. Po vytvoření dashboardu, na kterém se zobrazují online naměřené veličiny se automaticky vytvoří další přístupový token, který umožňuje zápis dat do grafů. Na obr. 29 je znázorněno nastavení nového přístupového tokenu.

Token details

Token ID
mkrfox1200

Token Name
Sigofx MKRFOX1200

Enabled

Token Permissions

Type	Resource	Action
<input type="checkbox"/> Bucket	mkrfox1200	WriteBucket

Access Token
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiJta3Jmb3gxMjAwIiwidXNyIjoiaWoiJlaG9yaWsiOiJlLnR5cCI6IkpXVCJ9.eyJqdGkiOiJta3Jmb3gxMjAwIiwidXNyIjoiaWoiJlaG9yaWsiOiJlLnR5cCI6IkpXVCJ9

Save Token

Obr. 29 Thinger – Access Token

Posledním krokem je vytvoření vlastního dashboardu neboli nástěnky, na které se budou zobrazovat naměřená data. V hlavní nabídce se vybere možnost „Dashboards“ a poté „Add Dashboard“. Dále už zbývá pouze vybrat vhodný typ grafu (koláčový, spojnicový, sloupcový) a jeho vstupní data. Na jedné nástěnce může být více druhů grafů, zároveň je možné vytvořit více nástěnek, která zobrazují data z různých zařízení.

3.2.8 Losant

Jako poslední je popsáno nastavení IoT platformy Losant, která jako předchozí platformy umožňuje vizualizaci, archivaci a dílčí práci s naměřenými daty pomocí Java scriptů.

Jako první je nutné vytvořit nový Webhook. V hlavní nabídce se vybere možnost „Applications“ dále „Webhooks“ a poté „Add Webhook“. Po vytvoření nového Webhooku se automaticky vygeneruje URL adresa, kterou zkopírujeme do nastavení Custom callback – Losant v Sigfox backendu. Obr. 30 popisuje nastavení nového Webhooku.

NEW WEBHOOK

After you create your new webhook, you will be assigned a unique URL for making your requests.

Webhook Name

VERIFICATION

Some webhook providers require the endpoint to be verified. Losant will automatically respond to verification requests for the following providers. If you are attempting to use a webhook provider that requires verification and is not listed below, please [let us know](#).

No Verification Alexa Facebook Messenger Fitbit Twilio

Verification Code

Response Code

BASIC AUTH

You can optionally choose to require basic auth for requests against this webhook.

Basic Auth Username

Basic Auth Password

CUSTOM REPLIES ⓘ

You can optionally choose to configure this webhook to wait for a reply from a workflow. When checked, this means that when a request is made against this webhook, the request will wait for a Webhook Reply node to be executed in a workflow for the particular request, and that reply will be returned. If no workflow provides a reply within 30 seconds, the request will be timed out.

Wait for reply from workflow

Obr. 30 Losant – Webhook

Druhým krokem je vytvoření nového zařízení v platformě Losant, které bude reprezentovat použité Sigfox zařízení. V hlavním menu se vybere možnost „Applications“ dále „Devices“ a poté „Add Device“. Pole sigfox_id představuje skutečné ID použitého

Sigfox zařízení, které je dostupné na stránkách Sigfoxu. Dále už jsou přidány pouze proměnné, které se budou následně vizualizovat s jejich příslušným datovým formátem. Nastavení nového Losant zařízení je na obr. 31.

DEVICE TAGS ?

Device tags provide a way to organize your devices. Tags are defined as keys and values. In other parts of the platform, like visualizations, you can query devices by their tags.

Keys may only contain uppercase letters, lowercase letters, numbers, underscores (`_`) or hyphens (`-`).

Key	Value	
sigfox_id	18BC3B	-

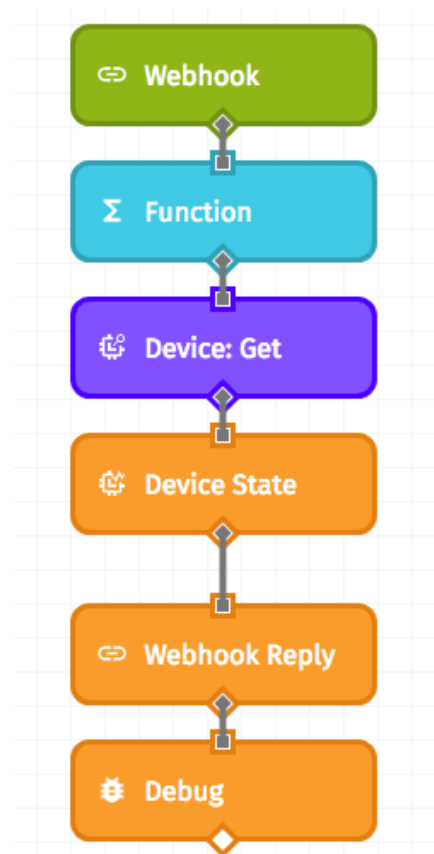
DEVICE ATTRIBUTES ?

Attributes define the fields and data types of the device's state. These are used to properly visualize and validate the various sensor or other data this device reports. For example a device with a temperature sensor might have an attribute with the data type "Number" and the name "temperature".

Data Type	Name	
Number	temperature	-
Number	humidity	-
Number	CO2	-
Number	atmospheric_pressure	-

Obr. 31 Losant – Device

Po úspěšném vytvoření nového zařízení je pro správnou analýzu a práci s daty nezbytné vytvořit novou Workflow. V hlavní nabídce se vybere možnost „Applications“ dále „Workflows“ a poté „Create Workflow“. Nová Workflow se tvoří formou bloků, které je možné řadit sériově i paralelně. Seznamy bloků, ze kterých je možno vybírat, se řadí na Triggers, Experience, Logic, Data a Outputs. Na obr. 32 je znázorněna vytvořená Workflow.



Obr. 32 Losant – Workflow

Prvním blokem z Workflow je **Webhook** ze seznamu bloků Triggers. V jeho nastavení se vybere již dříve vytvořený Webhook s názvem Sigfox Callback. Dalším blokem je **Function** ze seznamu bloků Logic. Tento blok obsahuje jednoduchý JavaScript, který upraví surová data pro jejich následnou vizualizaci.

```
payload.data.body.temperature = (payload.data.body.temperature/32767)*120;  
payload.data.body.humidity = (payload.data.body.humidity/65536)*110;  
payload.data.body.pressure = (payload.data.body.atmospheric_pressure/65536) *200;  
payload.data.body.CO2 = (payload.data.body.CO2/65536)*10000;
```

Nyní jsou již k dispozici veškerá data, která je třeba uložit do již dříve vytvořeného Losant zařízení. Ze seznamu Data se vybere blok **Device: Get**. Na obr. 33 je znázorněno nastavení nového bloku **Device: Get**, díky kterému je možné v dalších krocích uložit naměřená data přijatá ze Sigfox backendu do Losant zařízení.

Label

Device: Get

[Add Description](#)

QUERY METHOD

Find a device by ID / or a tag query.

Find by ...

Tag Query ▼

CONFIGURATION

You may opt to return multiple devices if more than one devices matches your query. Any device(s) returned must match all tags. If a key is set without a value, any device that has that key set regardless of the value will be returned. (And vice-versa for values set without keys.) Both keys and values are templatable.

Return multiple devices?

Key Template	=	Value Template	
sigfox_id		{{data.body.device}}	-
Key Template	=	Value Template	

RESULT PATH

Payload path at which the device object will be stored. You may also opt to return the device's tags as an object instead of as the standard array. If no device matches your query, the result will be `null`.

Return tags as an object map instead of an array

Result Path

data.deviceResponse

Obr. 33 Losant – Device: Get

Dalším krokem je uložení připravených dat do známého Losant zařízení. Ze seznamu Outputs se vybere blok **Device state**. Obr. 34 popisuje nastavení bloku **Device state**. Ke každému použitému atributu (veličině) z užitečných dat je nutné vyplnit přesnou cestu, která vychází z nastavení Custom Callback – Losant v Sigfox backendu.

Device ID JSON Path

data.deviceResponse.id

STATE

Now, define the attribute(s) and their values. Attributes that render to blank values will be ignored. You may reference the state updates by payload path, JSON template or individual field input.

Data Method

Individual Fields

Attribute		Value	
temperature	=	{{ data.body.temperature }}	-
humidity	=	{{ data.body.humidity }}	-
CO2	=	{{ data.body.CO2 }}	-
atmospheric_pressure	=	{{ data.body.atmospheric_pressure }}	-

Obr. 34 Losant – Device state

Předposledním krokem je nastavení bloku **Webhook Reply** ze seznamu Outputs. Díky tomuto bloku bude nově vytvořená Workflow reagovat pouze, když dorazí nová data z měřicího zařízení. Na obr. 35 je znázorněno nastavení bloku **Webhook Reply** včetně podmínky, která spouští celý Workflow.

Response Code Template

200

Reply Body

String Template Payload Path

```
{ "{{ data.body.device }}" : { "downlinkData" : "0000000000000000" } }
```

REPLY HEADERS

Optionally, you can set header names and values for the reply. Both names and values are templatable.

Name Template		Value Template	
Content-Type	=	application/json	-

Obr. 35 Losant – Webhook Reply

Poslední blok z Workflow je **Debug** ze seznamu Outputs, který umožňuje monitorování užitečných dat přímo ve Workflow a nepotřebuje žádné speciální nastavení. Nyní již stačí pouze vytvořit novou Dashboard, neboli nástěnku. V hlavní nabídce se vybere možnost „Dashboards“ a poté „Create Dashboard“. V nastavení vybereme vytvořenou Workflow a veličiny, které se mají zobrazovat společně se způsobem jejich zobrazení (graf, indikátor, pole, atd.).

3.2.9 Porovnání ceníků jednotlivých platforem

V následujících třech tabulkách jsou ceníky jednotlivých IoT platforem (Thinger, Losant a Thingspeak), které jsou dostupné na jejich webových stránkách.

Z neplacených variant nejlépe vyšla platforma Thingspeak, která poskytuje neomezenou dobu, po kterou jsou přijatá data uložena na serveru, až do chvíle, kdy je uživatel sám smaže. Naopak platforma Losant zálohuje data pouhých 30 dní, platforma Thinger uchová data po dobu jednoho roku. Thingspeak nemá zároveň limitovaný počet zařízení a umožňuje práci s daty pomocí Matlabu. Losant umožňuje práci s daty pomocí JavaScriptu, Thinger podobnou funkcí nedisponuje.

Pokud jsou všechny platformy porovnány z hlediska cena/možnosti nejlépe vychází Thingspeak, na druhém místě je Losant a poslední příčku obsadil Thinger. Všechny tři platformy jsou uživatelsky přívětivé a funkční, výběr tedy záleží především na způsobu plánovaného využití.

Thinger			
Free	Maker	Business	On-demand
0 € / month	3,95 € / month	19,95 € / month	from 199 / month
Up to 2 Devices	Up to 20 Devices	Up to 100 Devices	n-demand Devices
Up to 4 Dashboards	Up to 20 Dashboards	Up to 100 Dashboards	On-demand Dashboards
Up to 4 Endpoints	Up to 20 Endpoints	Up to 100 Endpoints	On-demand Endpoints
Up to 4 Buckets	Up to 20 Buckets	Up to 100 Buckets	On-demand Buckets
Up to 4 Tokens	Up to 20 Tokens	Up to 100 Tokens	No tokens limit
Standard bucket write rate (1/60s)	Improved bucket write rate (1/30s)	Improved bucket write rate (1/15s)	No bucket write-limit rate (<1s)
Standard Endpoint calls rate (1/10s)	Improved Endpoint calls rate (1/1s)	Improved Endpoint calls rate (1/0.5s)	No endpoints calls limited
Up to 1 Year Data Retention	Up to 2 Year Data Retention	Up to 3 Year Data Retention	Custom Data Retention
Shared Thinger.io Cloud	Shared Thinger.io Cloud	Shared Thinger.io Cloud	Isolated Thinger.io Cloud
Community Support	Community/Email Support	Community/Email Support	Community/Email/Skype Support

Tab. 3 Cenik Thinger [13]

Losant					
	SANDBOX	LAUNCH	BOOST	SCALE	LET'S CHAT
PRICE	0 \$ / month	49 \$ / month	199 \$ / month	499 \$ / month	hello@losant.com
DEVICES	10	100	500	1000	Unlimited
MEMBERS	n/a	5	25	100	Unlimited
EXPERIENCE USERS	1	10	50	100	Unlimited
PAYLOADS / MO.	1,000,000	5,000,000	25,000,000	75,000,000	Unlimited
ADD'L PAYLOADS	n/a	\$9.00/mm	\$7.50/mm	\$5.00/mm	Based on Volume
DATA RETENTION	30 Days	60 Days	90 Days	180 Days	Unlimited
SUPPORT	Forums	Forums	Forums, Email	Forums, Email	Forums, Email, SLA

Tab. 4 Ceník Losant [14]

Thingspeak					
	FREE	STUDENT	HOME	ACADEMIC	STANDART
Price per UNIT	0 € / month	55 € / month	75 € / month	250 € / month	600 € / month
Usage	For small non-commercial projects and for evaluation of the service	For students at degree-granting institutions	For personal use only	For academic use by faculty, staff, or researchers at degree-granting institutions	For all commercial, government and revenue generating activities
Number of messages	3 million/year	33 million/year per unit	33 million/year per unit	33 million/year per unit	33 million/year per unit
	8,200/day	90,000/day per unit	90,000/day per unit	90,000/day per unit	90,000/day per unit
Message update interval limit	Every 15 seconds	Every second	Every second	Every second	Every second
MATLAB Compute Timeout	20 seconds	20 seconds	20 seconds	60 seconds	60 seconds
Number of simultaneous MQTT subscriptions	Limited to 3	50 per unit	50 per unit	50 per unit	50 per unit
Private channel sharing	Limited to 3 shares	Unlimited	Unlimited	Unlimited	Unlimited
Technical Support	Forum	Forum	Forum	Standard MathWorks support	Standard MathWorks support

Tab. 5 Cenik Thingspeak [15]

4 Kvalita ovzduší ve vnitřních prostorech

Součástí diplomové práce je i měření kvality ovzduší ve vnitřních prostorech, která se určuje podle teploty, relativní vlhkosti a koncentrace CO₂. Vnitřní výpočtové teploty jsou určeny dle ČSN EN 12831 a doporučené relativní vlhkosti vzduchu dle ČSN 06 0210 (norma byla v roce 2008 zrušena, ale hodnoty relativní vlhkosti jiný předpis neupravil). Maximální hodnota koncentrace CO₂ stanovuje vyhláška 268/2009Sb.

Aby byl pobyt v místnostech příjemný, měla by se teplota pohybovat okolo 20 °C, relativní vlhkost by měla být mezi 45 až 65 % a maximální koncentrace CO₂ 1500 ppm. Všechny tyto veličiny je možné regulovat pomocí topení, ventilace, větrání, klimatizací, zvlhčovačem a odvlhčovačem. Problém s koncentrací CO₂ nastává především u nových nebo rekonstruovaných budov s plastovými okny, která mají sice výborné termoizolační vlastnosti, ale zároveň přestaly přirozeně dýchat, a je nutné častěji větrat. V prostorech, kde je vyšší koncentrace lidí (školy, kanceláře atd.) je situace ještě horší. [16]

Vliv koncentrace CO₂ na lidský organismus:

- 350-400 ppm - úroveň venkovního prostředí
- do 1000 ppm - doporučená úroveň CO₂ ve vnitřních prostorech
- 1200 - 1500 ppm - doporučená maximální úroveň CO₂ ve vnitřních prostorech
- 1000 - 2000 ppm - nastávají příznaky únavy a snižování koncentrace
- 2000 - 5000 ppm - nastávají možné bolesti hlavy
- 5000 ppm - maximální bezpečná koncentrace bez zdravotních rizik
- > 5000 ppm - nevolnost a zvýšený tep
- > 15000 ppm - dýchací potíže
- > 40000 ppm - možná ztráta vědomí

4.1 Vlastní měření

Pomocí měřicího zařízení byly provedeny tři kontrolní měření týkající se kvality ovzduší ve vnitřních prostorech. Měření probíhá každých 15 minut. První bylo provedeno na pracovišti ZF Engineering, druhé v mém bytě a poslední v počítačové učebně ZČU FEL. Bohužel se měření nepodařilo realizovat během zimního období, kdy se místnosti větrají podstatně méně a hladiny koncentrace CO₂ dosahují vyšších hodnot.

4.1.1 ZF Engineering

První měření bylo provedeno od 6. 4. 2018 do 11. 4. 2018 ve firmě ZF Engineering, která sídlí v Plzni na Borských polích a je mým nynějším zaměstnavatelem. Měřicí zařízení bylo umístěno v otevřeném kancelářském prostoru o rozloze 200 m², ve kterém pracuje průměrně 40 lidí současně.

Měření teploty bylo ovlivněno, protože zařízení muselo být umístěno nedaleko LCD monitorů. Z toho důvodu se dá předpokládat, že naměřené teplota může být v případě, že jsou monitory aktivní, až o 2 °C vyšší. Průměrná naměřená teplota byla 24 °C, nejvyšší naměřená teplota dosáhla 27 °C a nejnižší 22 °C. Průměrná hladina koncentrace CO₂ činila 661 ppm, nejvyšší hodnota byla 1238 ppm a nejnižší 492 ppm. Průměrná relativní vlhkost byla 29 %, maximální hodnota dosáhla jen 38 % a minimální klesla až na 16%.

Hladiny CO₂ v těchto prostorech jsou vyhovující i při vyšším počtu lidí. Vyšší naměřené teploty se v letních měsících řeší především aktivní klimatizací, zároveň je nutné brát v úvahu nepřesnost měření. Největším problémem jsou velmi nízké hladiny relativní vlhkosti. Někteří zaměstnanci toto řeší pomocí lokálních membránových zvlhčovačů vzduchu, které umísťují na své pracovní stoly. Na obr. 36 jsou znázorněny grafy z měření environmentálních parametrů prostředí z firmy ZF Engineering pomocí platformy Losant.

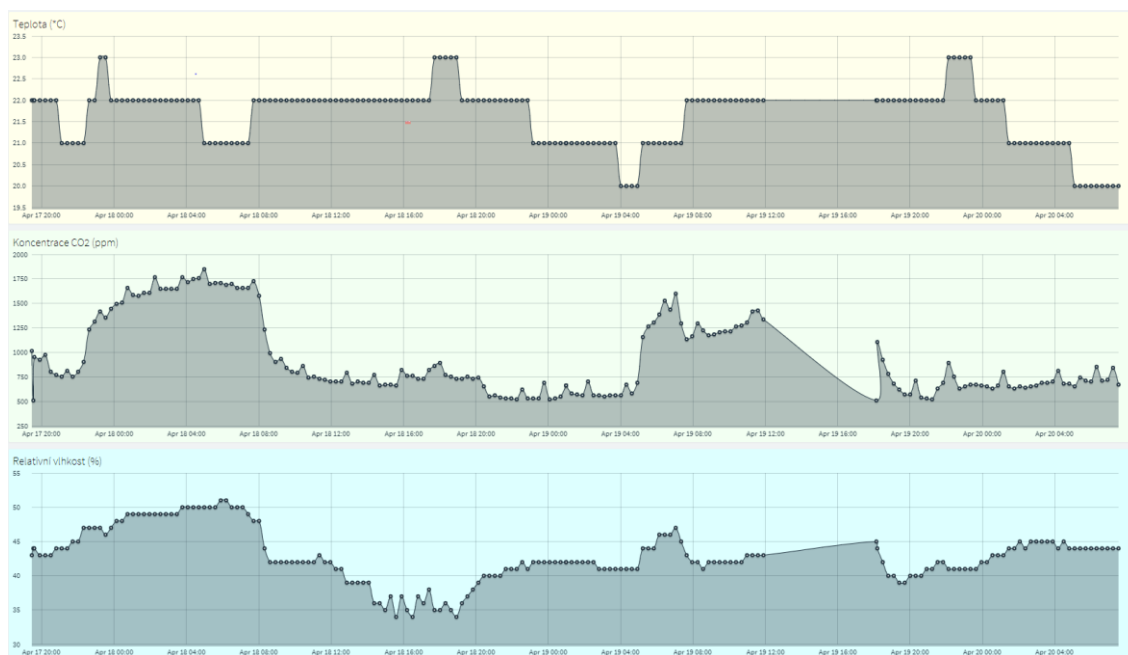


Obr. 36 Losant - ZF Engineering

4.1.2 Byt

Další měření bylo provedeno od 17. 4. 2018 do 19. 4. 2018, v bytě kde žiji, přesněji v mém pokoji o rozloze 25 m². Byt je situován v druhém patře novostavby staré dva roky, která je osazena plastovými okny.

Teplota dosahovala maximálně 23 °C, její minimální hodnota klesla na 20 °C. Průměr z celého měření činil příjemných 22 °C. Průměrná koncentrace CO₂ byla 979 ppm, minimální hodnota klesla na 509 ppm a maximální hodnota dosahovala 1850 ppm. Maximální hodnota koncentrace CO₂ přesahuje doporučenou maximální úroveň a bylo ji dosaženo během noci, kdy bylo zavřené okno po celou dobu mého spánku. Během několika minut větrání se koncentrace CO₂ opět vrátila pod doporučenou mez. Maximální naměřená relativní vlhkost dosahovala 51 %, její minimální hodnota klesla na 34 %. Průměrná hodnota z celého měření činila 43 %. Na obr. 37 jsou znázorněny grafy měření environmentálních parametrů prostředí z mého pokoje pomocí platformy Thingier.



Obr. 37 Thingy - byt

4.1.3 ZČU FEL – EU506

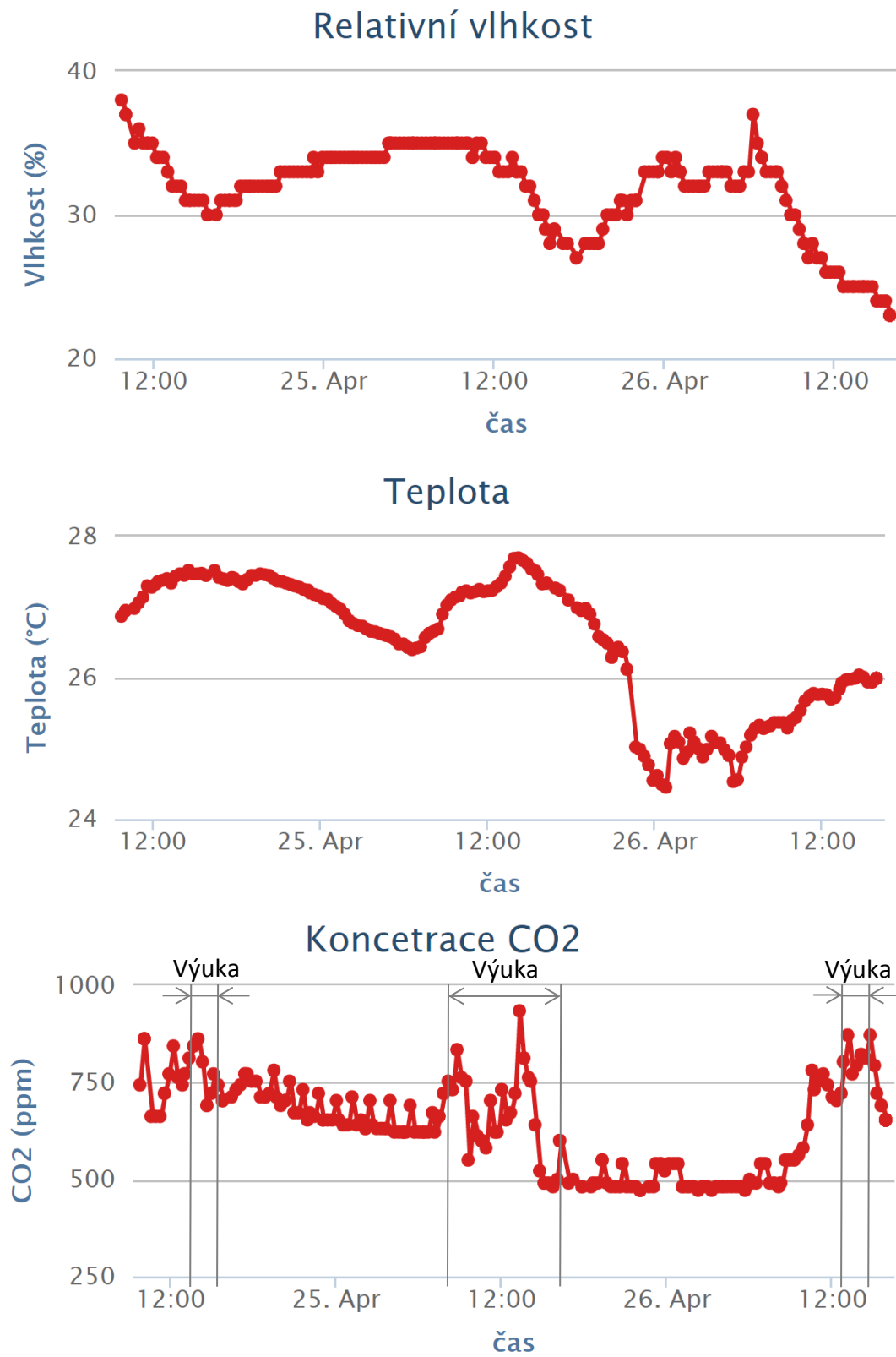
Poslední měření bylo provedeno v prostorách ZČU FEL, přesněji v počítačové učebně, EU506. Učebna je situována v pátém patře a její kapacita je 20 lidí. Měření probíhalo od 24. 4. 2018 do 26. 4. 2018.

Hladina koncentrace CO₂ byla po celou dobu měření v doporučených hranicích, a to i v případě, že zde probíhala výuka a učebna byla plná lidí. Na obr. 38 je znázorněn rozvrh učebny EU506, z grafů je patrné, že hladina koncentrace CO₂ je závislá na obsazení učebny. Maximální hodnota dosáhla 1044 ppm, minimální klesla na 472 ppm. Průměrná hodnota koncentrace CO₂ během celého měření byla 641 ppm.

Út 24.4.18				14:50 - 16:30 KAE/PEL RA Welsar	
St 25.4.18		11:10 - 12:50 KAE/PEL RA Basl Vlášek	13:00 - 14:40 KAE/PEL RA Welsar	14:50 - 16:30 KAE/PEL RA Vlášek	16:40 - 18:20 KAE/PSR RA Lahoda
Čt 26.4.18			13:00 - 14:40 KAE/PEL RA Basl		

Obr. 38 Thingspeak – EU506

Maximální teplota dosáhla až na 28 °C, minimální hodnota klesla na 24 °C. Průměr za celé měření činil 26 °C. Tyto hodnoty jsou poměrně vysoké, a pro někoho mohou být i nepříjemné. Snížení teploty uvnitř učebny by bylo možné pomocí klimatizace. Maximální naměřena relativní vlhkost dosáhla 47 %, její minimum kleslo na 23 %. Průměr za celé měření činil 32%. Na obr. 39 jsou znázorněny grafy relativní vlhkost, teplota a koncentrace CO₂ z učebny EU506 pomocí platformy Thingspeak.



Obr. 39 Thingspeak – EU506

Závěr

V první části textu byly popsány základy problematiky Internetu věcí společně s možnostmi jeho využití v komerčním i soukromém prostoru a problémy, které nastávají při jeho zavádění. Dále jsou uvedeny a popsány tři dominantní bezdrátové komunikační systémy typu LPWAN: Sigfox, LoRa a Narrow Band. Tyto systémy jsou vhodné pro IoT aplikace s požadavkem na nízkou spotřebu elektrické energie a velkým dosahem.

Druhá část je zaměřena na hardware použitý pro realizaci měřicího zařízení. Základem je deska Arduino MKRFOX1200 disponující integrovaným Sigfox modulem, který zajišťuje bezdrátovou komunikaci. Na tuto desku jsou připojeny dva senzory, které měří environmentální parametry prostředí. První senzor BME280 Weather Click je schopen měřit teplotu, atmosférický tlak a relativní vlhkost. Druhý senzor Gravity Infrared CO₂ měří koncentraci oxidu uhličitého v ovzduší. Dále je popsána šasi vytištěná na 3D tiskárně, do které je umístěno kompletní měřicí zařízení. Kromě toho tato kapitola obsahuje podrobné měření spotřeby elektrické energie měřicího zařízení v jednotlivých fázích: start zařízení, přehřívání senzoru, datový přenos a spánek.

Třetí část je orientovaná na programové prostředky použité při realizaci měřicího zařízení. Software obsluhující desku, senzory a komunikaci byl vytvořen v aplikaci Arduino IDE a je uložen na CD, které je součástí diplomové práce. Dále je tato část zaměřena na nastavení Sigfox backendu, pomocí kterého jsou naměřená data odesílána do emailové schránky nebo do jednotlivých IoT platforem, které umožňují vizualizovat, archivovat a editovat naměřená data. Rovněž je detailně popsáno nastavení IoT platforem: Thingspeak, Thinger a Losant společně s jejich porovnáním.

Čtvrtá část popisuje výsledky vlastního měření kvality ovzduší ve vnitřních prostorech. Byly provedeny čtyři měření, první proběhlo ve firmě ZF Engineering, druhé v mém bytě a poslední na půdě ZČU FEL. Ve všech případech byly dodrženy doporučené hladiny koncentrace CO₂. Všechny měření byly provedeny na jaře, pokud by se měřilo v zimě, hodnoty koncentrace CO₂ by byly vyšší kvůli menší frekvenci větrání vnitřních prostorů z důvodu nízkých venkovních teplot a následných teplených ztrát.

Celý měřicí systém je postaven na desce Arduino MKRFOX1200 a díky tomu je modulární. V budoucnu je proto možné doplnit měřicí zařízení dalšími senzory nebo jinými zařízeními (např. spínací prvky) a rozšířit tak jeho funkčnost. Celkové náklady na projekt jsou popsány v příloze 6.

Seznam Literatury

- [1] EVANS, Dave. *The Internet of Things: How the Next Evolution of the Internet Is Changing Everything*. 2011,11.
- [2] *Internet věci* [online]. [cit. 2017-11-29]. Dostupné z: <http://i2ot.eu/internet-of-things/>
- [3] *Consumer Internet of Things (CIoT) – what is it and how does it evolve?* [online]. [cit. 2017-11-29]. Dostupné z: <https://www.i-scoop.eu/internet-of-things-guide/what-is-consumer-internet-of-things-ciot/>
- [4] PAINE, Tony. *Průmyslový Internet věci a komunikace na okraji*.
- [5] MAŘÍK, Vladimír. *Průmysl 4.0 a jeho dopady v energetice, dopravě a stavebnictví*. 2016.
- [6] *Jak internet věci změní zdravotnictví* [online]. [cit. 2017-11-29]. Dostupné z: https://www.technickytydenik.cz/rubriky/archiv/jak-internet-veci-zmeni-zdravotnictvi_31859.html
- [7] SLAVÍK, Jakub. *Co to je a jak funguje inteligentní město – smart city*.
- [8] *SimpleCell* [online]. [cit. 2017-11-29]. Dostupné z: <https://simplecell.eu/>
- [9] *Sigfox* [online]. [cit. 2017-11-29]. Dostupné z: <https://www.sigfox.com/en/sigfox-iot-technology-overview>
- [10] *SigFox Frame Structure | SigFox MAC Frame Downlink, Uplink* [online]. [cit. 2017-11-29]. Dostupné z: <http://www.rfwireless-world.com/Tutorials/Sigfox-frame-structure.html>
- [11] *Služby CRA IoT* [online]. [cit. 2017-11-29]. Dostupné z: <https://www.cra.cz/sluzby-iot>
- [12] *Architektura a popis jednotlivých funkčních bloků* [online]. [cit. 2017-11-29]. Dostupné z: <https://pripoj.me/technicke-aspekty-technologie-lora/>
- [13] *Thinger* [online]. [cit. 2018-05-16]. Dostupné z: <http://www.thinger.io/>
- [14] *Thingspeak* [online]. [cit. 2018-05-16]. Dostupné z: <https://www.thingspeak.com/>
- [15] *Losant* [online]. [cit. 2018-05-16]. Dostupné z: <https://www.losant.com/>
- [16] *Vnitřní výpočtové teploty dle ČSN EN 12831 a doporučené relativní vlhkosti vzduchu dle ČSN 06 0210* [online]. [cit. 2018-04-16]. Dostupné z: <https://vetrani.tzb-info.cz/tabulky-a-vypocty/28-vnitri-vypoctove-teploty-dle-csn-en-12831-a-doporucene-relativni-vlhkosti-vzduchu-dle-csn-06-0210>

Seznam obrázků

Obr. 1 Vývoj IoT [1].....	12
Obr. 2 Pokrytí sítě Sigfox v ČR [8].....	17
Obr. 3 Architektura sítě Sigfox [9].....	18
Obr. 4 Přenosový protokol [10].....	19
Obr. 5 Sigfox OSI model[10].....	20
Obr. 6 Pokrytí LoRa [11].....	21
Obr. 7 Chirp modulace a modulace skokovou změnou kmitočtu.....	22
Obr. 8 Arduino MKRFOX 1200.....	27
Obr. 9 BME280 Weather Click.....	28
Obr. 10 Gravity Infrared CO2.....	29
Obr. 11 Step-UP měnič 5 V.....	30
Obr. 12 Šasi.....	31
Obr. 13 Blokové schéma zapojení měření.....	32
Obr. 14 Spuštění jednotky.....	33
Obr. 15 Perioda špičkového pulsu.....	34
Obr. 16 Puls.....	34
Obr. 17 Datový přenos.....	35
Obr. 18 Datový přenos.....	36
Obr. 19 Arduino IDE.....	38
Obr. 20 Sigfox backend.....	39
Obr. 21 Custom Callback – Email.....	40
Obr. 22 Custom Callback - Thingspeak.....	41
Obr. 23 Custom Callback – Thinger.....	43
Obr. 24 Custom Callback – Losant.....	44
Obr. 25 Nastavení kanálu Thingspeak.....	45
Obr. 26 API klíče Thingspeak.....	46
Obr. 27 Thingspeak – React.....	48
Obr. 28 Thinger – Data Bucket.....	49
Obr. 29 Thinger – Access Token.....	50
Obr. 30 Losant – Webhook.....	51
Obr. 31 Losant – Device.....	52

Obr. 32 Losant – Workflow	53
Obr. 33 Losant – Device: Get	54
Obr. 34 Losant – Device state.....	55
Obr. 35 Losant – Webhook Reply	55
Obr. 36 Losant - ZF Engineering.....	62
Obr. 37 Thinger - byt	63
Obr. 38 Thingspeak – EU506	63
Obr. 39 Thingspeak – EU506	65

Seznam tabulek

Tab. 1 Ceník předplatného Simple Cell.....	20
Tab. 2 Výdrž zařízení na bateriích.....	36
Tab. 3 Ceník Thinger	57
Tab. 4 Ceník Losant.....	58
Tab. 5 Ceník Thingspeak	59

Přílohy

Příloha 1 – Arduino MKRFOX1200

Arduino MKRFOX 1200	
Mikrokontrolér	SAMD21 Cortex-M0+ 32bit low power ARM MCU
Napájení desky (USB/VIN)	5V
Podporované baterie	2 x AA nebo AAA
Operační napětí logických obvodů	3,3 V
Digitální I/O piny	8
PWM piny	12 (0, 1, 2, 3, 4, 5, 6, 7, 8, 10, A3 nebo 18 , A4 nebo 19)
UART	1
SPI	1
I2C	1
Analogové vstupní piny	7 (ADC 8/10/12 bit)
Analogové výstupní piny	1 (DAC 10 bit)
Externí přerušení	8 (0, 1, 4, 5, 6, 7, 8, A1 -nebo 16-, A2 - nebo 17)
DC proud na I/O pin	7 mA
Flash paměť	256 KB
SRAM	32 KB
EEPROM	není
Taktovací frekvence	32.768 kHz (RTC), 48 MHz
Vestavěné LED	6
Zisk antény	2 dB
Frekvenční pásmo	868 Mhz
Pracovní oblast	EU
Délka	67,64 mm
Šířka	25 mm
Hmotnost	32 g

Příloha 2 – Gravity: Analog Infrared CO2

Gravity: Analog Infrared CO2	
Detekce plynu	Oxid uhličitý (CO ₂)
Operační napětí	4,5 – 5,5V
Průměrný proud	< 60 mA
Špičkový proud	150 mA
Výstupní signál	Analogový výstup (0,4 - 2V)
Měřicí rozsah	0 - 5000ppm
Přehřívací čas	3 min
Čas odezvy	2 min
Provozní teplota	0 – 50 °C
Provozní vlhkost	0 - 95%
Životnost	> 5 let
Rozměry	37 x 69 mm
Hmotnost	21,8 g

Příloha 3 - BME280 Weather click

BME280 Weather click	
Modul	Bosch BME280
Měřicí rozsah teploty	- 40 až +85 °C
Přesnost měření teploty	+ - 1 ° C
Měřicí rozsah vlhkosti	0 až 100 %
Přesnost měření vlhkosti	± 3%
Měřicí rozsah atmosférického tlaku	300 až 1100 hPa
Přesnost měření tlaku	± 1 Pa
Tři operační módy	sleep, forced, normal
Rozhraní	I2C,SPI
Napájení	3,3 V
Rozměry	28,6 x 25,4 x 3,3mm
Hmotnost	3,6 g

Příloha 4 - Step-UP měnič 5 V

Step-UP měnič 5 V	
Výstupní napětí	5 V
Maximální výstupní proud	500 mA
Minimální vstupní napětí	1 V
Maximální vstupní napětí	5 V
Účinnost	≤ 96%
Provozní teplota	0 - 150 °C
Rozměry	17,5 x 25 mm
Hmotnost	2,3 g
Výstupní napětí	5 V

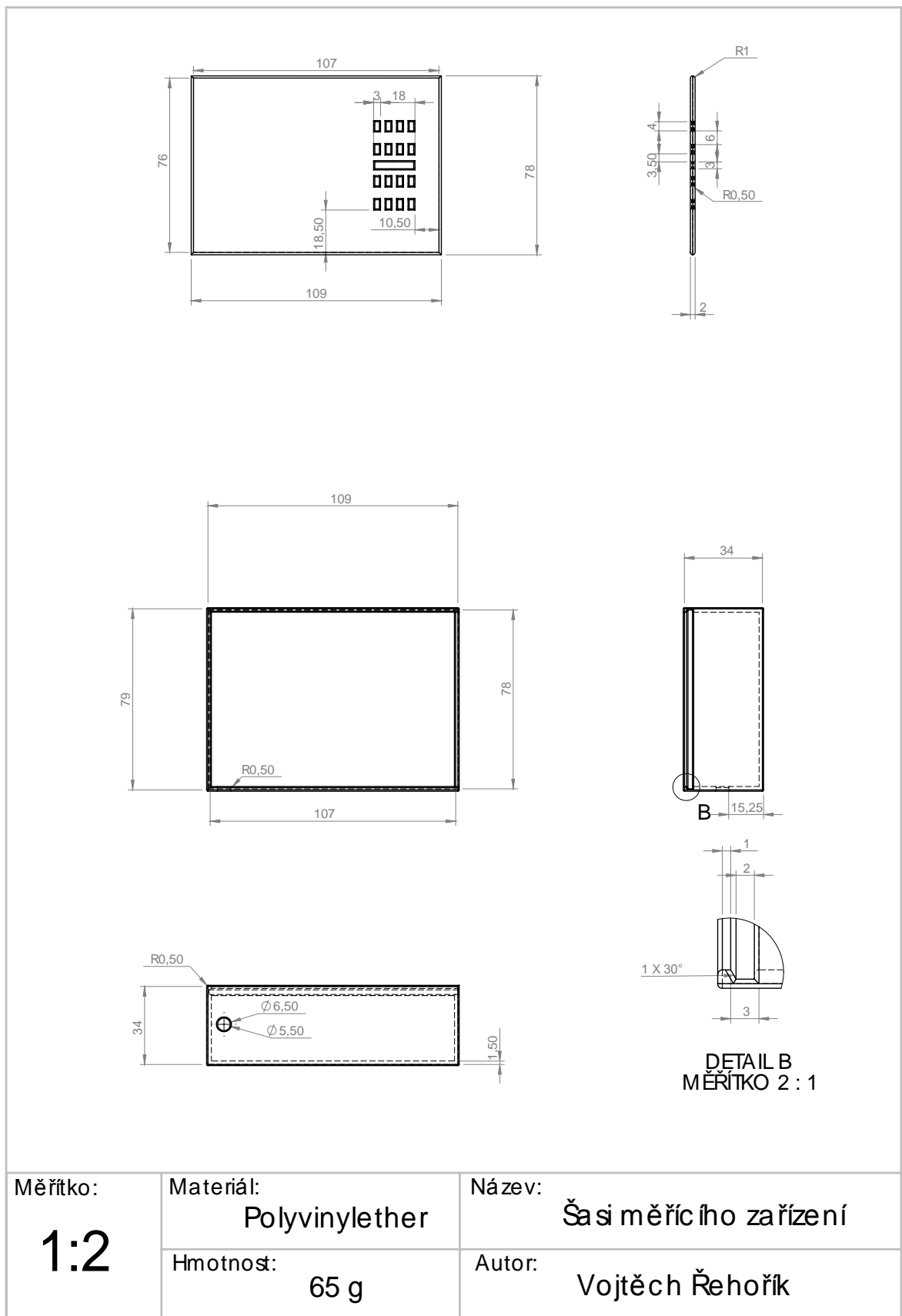
Příloha 5 - BTS117 TO220

BTS117 TO220	
Unipolární tranzistor	N-Kanál
I_{dss}	7 A
U_{ds}	60 V
U_{gs}	2,2 V
P_d	50 W
R_{ds}	0,1 Ω
Dioda	ano

Příloha 6 - Náklady

Náklady		
Arduino MKRFOX1200	1200	Kč
BME280 Weather click	500	Kč
Gravity Analog Infrared CO2	1400	Kč
BTS117 TO220	40	Kč
3 x Lithiová baterie 3,6V 2600 mAh	500	Kč
Step-UP měnič 5V	30	Kč
Šasi	100	Kč
Celková cena	3770	Kč

Příloha 7 – Technický výkres šasi



Příloha 8 – Schéma zapojení

