

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Vizualizace dat z navržené bezdrátové meteostanice

Jakub Rejř

Bakalářská práce  
2019

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2017/2018

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jakub Rejřř**  
Osobní číslo: **I15001**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Vizualizace dat z navržené bezdrátové meteostanice.**  
Zadávající katedra: **Katedra informačních technologií**

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

Cílem bakalářské práce je navrhnout a realizovat bezdrátovou meteorologickou stanici, sloužící pro měření teploty, tlaku a relativní vlhkosti vzduchu. Teoretická část práce představí komunikační standardy IoT využívající analytickou službu ThingSpeak. Praktická část práce bude obsahovat návrh součástí a zapojení vlastní meteorologické stanice. Dále bude provedeno dimenzování napájecích částí řešení meteorologické stanice, která bude napájena prostřednictvím solárního panelu a akumulátoru. Naměřená data budou vizualizována ve vytvořené webové aplikaci. Hodnoty naměřených dat budou porovnány s hodnotami nejbližší ČHMÚ.

Rozsah grafických prací:

Rozsah pracovní zprávy: **35 - 45 stran**

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

MCEWEN, Adrian a Hakim CASSIMALLY Designing the internet of things, Chichester, England: Wiley, 2014. ISBN 978-111-8430-651  
MQTT Essentials - A Lightweight IoT Protocol, Birmingham, United Kingdom: Packt Publishing Limited, 2017. ISBN 9781787287815

Vedoucí bakalářské práce:

**Ing. Soňa Neradová, Ph.D.**

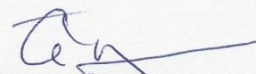
Katedra informačních technologií

Datum zadání bakalářské práce: **31. října 2017**

Termín odevzdání bakalářské práce: **12. května 2018**



Ing. Zdeněk Němec, Ph.D.  
děkan



Ing. Lukáš Čegan, Ph.D.  
pověřený vedením katedry

V Pardubicích dne 20. března 2018

## **Prohlášení autora**

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 1. 5. 2019

Jakub Rejřř

## **Poděkování**

Rád bych poděkoval vedoucí mé práce Ing. Soně Neradové, Ph.D. za vstřícný přístup, cenné rady a pomoc při psaní této bakalářské práce. Dále bych chtěl poděkovat své rodině za podporu během celého studia.

## **Anotace**

Bakalářská práce se zabývá návrhem a realizací bezdrátové meteostanice a vizualizací naměřených dat z ní. Teoretická část je zaměřena na téma Internetu věcí (IoT) a jeho komunikačních standardů. Popsána je i internetová služba ThingSpeak, která slouží ke sběru dat z IoT zařízení. Pozornost je věnována také hardwarové platformě Arduino. V praktické části je popsán návrh vlastní bezdrátové meteostanice. Dále se práce zaměřuje na vývoj webové aplikace složící k přehlednému zobrazení naměřených dat.

## **Klíčová slova**

Internet věcí, meteostanice, Arduino, ThingSpeak, REST API, JavaScript, HTML.

## **Title**

Data visualization from the designed wireless weather station.

## **Annotation**

This bachelor thesis focuses on design and implementation of the wireless weather station and measured data visualization. The theoretical part is focused on Internet Of Things (IoT) and its communication standards. It also describes the ThingSpeak internet service, which purpose is to store data from IoT devices. Arduino hardware platform is described as well. The practical part of the thesis describes the wireless weather station design and developing of the web application to visualize measured data.

## **Keywords**

Internet Of Things, weather station, Arduino, ThingSpeak, REST API, JavaScript, HTML.

# Obsah

<b>Seznam zkratk</b> .....	<b>9</b>
<b>Seznam obrázků</b> .....	<b>10</b>
<b>Seznam tabulek</b> .....	<b>10</b>
<b>Úvod</b> .....	<b>11</b>
<b>1 Internet věcí</b> .....	<b>12</b>
1.1 Ubicomp .....	13
1.2 Historie .....	13
1.3 Sítě pro IoT .....	14
1.4 Uplatnění a příklady použití .....	15
1.5 Průmysl 4.0 a IoT (IIoT) .....	16
<b>2 ThingSpeak</b> .....	<b>18</b>
2.1 REST .....	18
2.2 MQTT .....	19
<b>3 Arduino</b> .....	<b>21</b>
3.1 Vznik .....	21
3.2 Arduino hardware .....	21
3.3 Arduino software .....	23
<b>4 Návrh meteostanice</b> .....	<b>24</b>
4.1 Arduino Pro Mini .....	24
4.2 Senzory .....	25
4.3 Komunikační modul GSM .....	26
4.4 Napájení a spotřeba .....	26
4.5 Konstrukce.....	28
4.6 Radiační štít.....	30
<b>5 Program a přenos dat z meteostanice</b> .....	<b>31</b>
<b>6 Měřené veličiny</b> .....	<b>34</b>
6.1 Teplota vzduchu .....	34
6.2 Tlak.....	34
6.3 Vlhkost .....	34
6.4 Rosný bod.....	34
6.5 Napětí na baterii a solárním panelu, síla signálu .....	35

<b>7</b>	<b>Webová aplikace.....</b>	<b>36</b>
7.1	HTML a CSS .....	36
7.2	Javascript .....	38
<b>8</b>	<b>Umístění a srovnání s profesionální stanicí.....</b>	<b>41</b>
	<b>Závěr .....</b>	<b>45</b>
	<b>Literatura .....</b>	<b>46</b>
	<b>Seznam příloh .....</b>	<b>48</b>
	Příloha A – Schéma zapojení.....	49
	Příloha B – Seznam použitých součástek .....	50
	Příloha C – Fotografie .....	51
	Příloha D – Přiložené CD .....	53



## Seznam zkratek

API	Application Programming Interface
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
ČHMÚ	Český hydrometeorologický ústav
DC	Direct Current
GPIO	General-purpose input/output
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile communications
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
I2C	Inter-Integrated Circuit
ICSP	In-circuit serial programming
IDE	Integrated Development Environment
IIoT	Industrial Internet of Things
IO	Input/output
IoT	Internet of Things
IP	Internet Protocol
JSON	JavaScript Object Notation
LED	Light-Emitting Diode
M2M	Machine to Machine
MHD	Městská hromadná doprava
MIT	Massachusetts Institute of Technology
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MQTT	Message Queuing Telemetry Transport
PC	Personal Computer
QoS	Quality of Service
REST	Representational State Transfer
SIM	Subscriber Identification Module
SD	Secure Digital
SPI	Serial Peripheral Interface
TCP	Transmission Control Protocol
UART	Universal asynchronous receiver-transmitter
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
UTC	Coordinated Universal Time
UTP	Unshielded twisted pair
XML	Extensible Markup Language

## Seznam obrázků

Obrázek 1 – Ilustrace internetu věcí .....	12
Obrázek 2 – Diagram znázorňující 4 industriální revoluce včetně průmyslu 4.0 .....	17
Obrázek 3 – Ilustrace funkce MQTT protokolu .....	19
Obrázek 4 – Logo Arduina .....	21
Obrázek 5 – Deska Arduino Uno .....	22
Obrázek 6 – Ukázka zdrojového kódu pro Arduino .....	23
Obrázek 7 – Elektronika meteostanice .....	28
Obrázek 8 – Senzory, krabička s elektronikou a radiační štít .....	29
Obrázek 9 – Konstrukce radiačního štítu .....	30
Obrázek 10 – Příklad výpisu naměřených hodnot do konzole .....	32
Obrázek 11 – Ukázka vytvoření zprávy REST požadavku .....	32
Obrázek 12 – Ukázka zobrazení oken s aktuálními naměřenými hodnotami .....	36
Obrázek 13 – Ukázka zobrazení grafu s ovládacími prvky .....	37
Obrázek 14 – Ukázka zobrazení multigrafu .....	38
Obrázek 15 – Ukázka JSON formátu .....	39
Obrázek 16 – Současné umístění meteostanice .....	41
Obrázek 17 – Graf srovnání naměřených hodnot pro teplotu .....	43
Obrázek 18 – Graf srovnání naměřených hodnot pro abs. tlak .....	43
Obrázek 19 – Graf srovnání naměřených hodnot pro rel. vlhkost .....	44

## Seznam tabulek

Tabulka 1 – Přehled bezdrátových sítí pro IoT .....	14
Tabulka 2 – Arduino Pro Mini specifikace .....	24
Tabulka 3 – BME280 specifikace .....	25
Tabulka 4 – DS18B20 specifikace .....	26
Tabulka 5 – Sim800L specifikace .....	26
Tabulka 6 – Srovnání naměřených hodnot teploty, abs. tlaku a rel. vlhkosti .....	42

## Úvod

Skoro každá domácnost je vybavena nějakým přístrojem na měření venkovní teploty. Může se jednat o obyčejný venkovní teploměr, ale i o pokročilou stanici s předpovědí počasí měřící teplotu, tlak, vlhkost, rychlost větru a úhrn srážek. Jedná se, ale téměř vždy pouze o statické zobrazení aktuálních naměřených hodnot na displeji.

Cílem této bakalářské práce je navrhnout a realizovat bezdrátovou meteostanici, která bude naměřená data odesílat na internet. Data pak bude možné zobrazit ve vlastní vytvořené webové aplikaci.

Teoretická část práce se v první kapitole zaměřuje na popis konceptu internetu věcí (IoT). Je zde zmíněna jeho historie a možnosti využití v praxi. Probrány jsou i dostupné sítě pro IoT a čtvrtou průmyslovou revoluci.

Ve druhé kapitole je představena internetová služba ThingSpeak sloužící k ukládání dat z IoT zařízení. Dále jsou popsány dvě možnosti, jak data do služby nahrát: HTTP REST volání a MQTT protokol.

Poslední kapitola teoretické části se věnuje hardwarové platformě Arduino. Ta je v posledních letech velice populární zejména díky své jednoduchosti, dostupnosti nepřeberného množství kompatibilních modulů a široké komunitě uživatelů. Kapitola se věnuje vzniku platformy, dostupnému hardwaru a softwaru.

V dalších kapitolách se práce věnuje praktické části.

Čtvrtá kapitola se věnuje návrhu vlastní bezdrátové meteostanice. Jsou zde představeny jednotlivé použité komponenty a senzory. Důraz při návrhu meteostanice byl kladen na energetickou soběstačnost. Zařízení má vlastní akumulátor, který se dobíjí pomocí solárního panelu. Stanice by měla být schopna fungovat řadu let bez zásahu uživatele.

Následující kapitola popisuje program meteostanice a přenos dat na internet pomocí GSM modulu. V šesté kapitole jsou popsány všechny měřené veličiny.

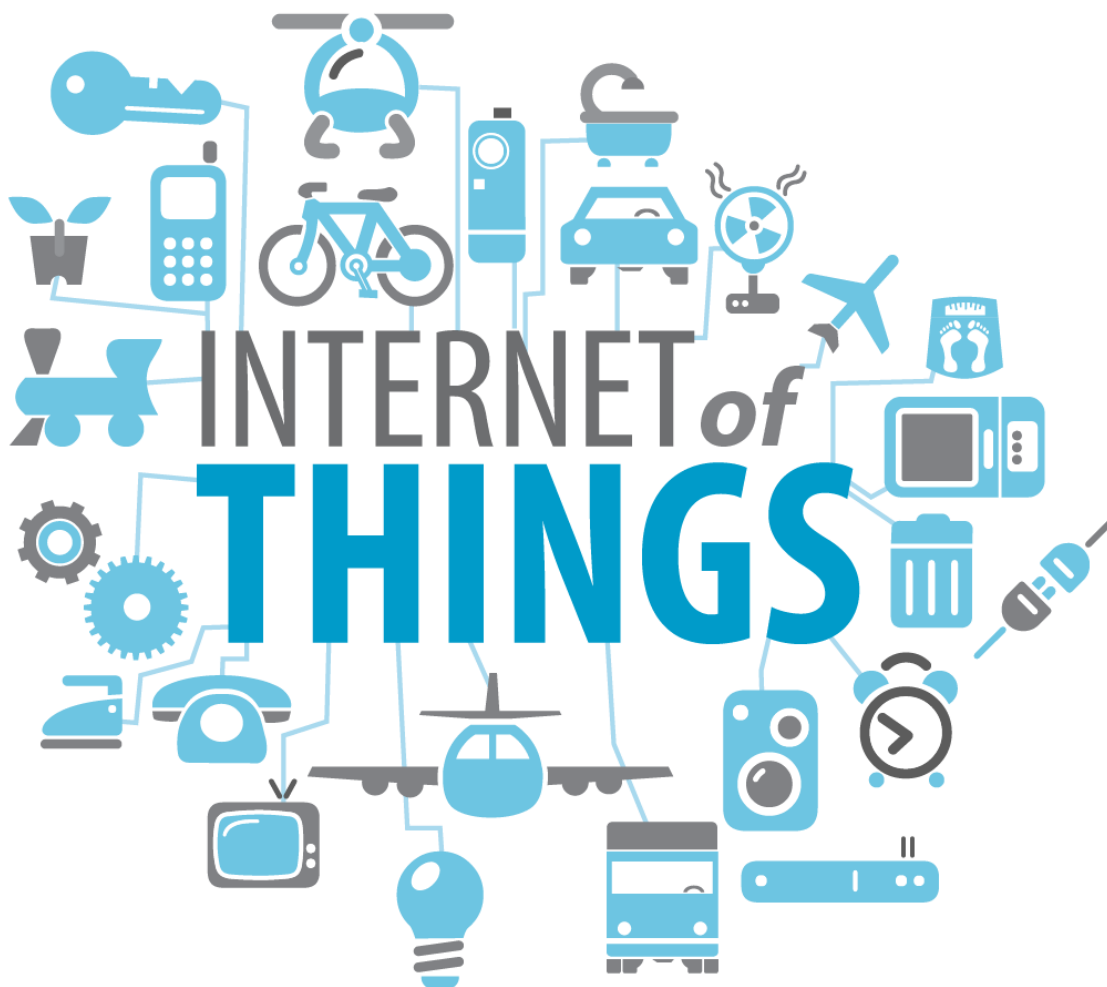
Sedmá kapitola se věnuje vývoji webové aplikace pro zobrazení naměřených dat. Aplikace je vytvořena pomocí HTML5, CSS3 a JavaScriptu. Jejím hlavním cílem je jednoduché a srozumitelné zobrazení naměřených dat z meteostanice. K tomu slouží především interaktivní grafy, které umožňují změnu zobrazovaného období. Aplikace má responzivní design, je tedy možné ji pohodlně používat nejen na počítači, ale i na mobilních zařízeních (např.: mobil, tablet).

Poslední kapitola bakalářské práce obsahuje srovnání naměřených hodnot zhotovené meteostanice s profesionální stanicí provozovanou Českým hydrometeorologickým ústavem.

## 1 Internet věcí

Internet věcí (anglicky Internet of Things – IoT) je souhrnné označení pro fyzická zařízení se schopností interakce s okolním světem pomocí internetu.

Kvůli všeobjímající povaze a rozsahu pojmu IoT je prakticky nemožné najít přesnou definici, se kterou by souhlasili všichni. Lze si pod ním vybavit různé senzory, nositelnou elektroniku, domácí automatizaci nebo třeba i chytré město. Až na vyjímky se, ale jedná vždy o tzv. embedded, tedy jednoúčelové zařízení určené pro konkrétní účel. Obrázek níže ilustruje spektrum využití IoT.



Obrázek 1 – Ilustrace internetu věcí

*Zdroj: (Bly, 2013)*

Jako příklad lze uvést deštník s držadlem, které indikuje, zda bude v nejbližší době pršet. Při odchodu z domu, vás světelnou signalizací upozorní na možnost nepříznivého počasí. Dalším příkladem by mohly být informační panely na autobusové zastávce. Zobrazují informace o odjezdech autobusů a jejich trase. Díky tomu, že každý autobus má

zabudovanou GPS a informace o svojí poloze zasílá do centrály, jsou informace vždy aktuální.

V mnoha případech by se dalo použít i univerzální zařízení jako je počítač nebo mobilní telefon. Podívat se na jízdní řád v mobilu na webových stránkách dopravce by se mohlo na první pohled zdát jako stejně funkční řešení. Není to ale tak docela pravda. Telefon je třeba nejdříve vytáhnout z kapsy, odemknout, spustit webový prohlížeč, najít správné stránky a přečíst informace z malé obrazovky (nehledě na to, že můžete být v cizí zemi jako turista a bez mobilních dat). Toto řešení se tedy zdá být složitější a vyžaduje plnou pozornost. Stejně tak je tomu i u výše zmíněného deštníku. Ano, můžete se před tím, než vyjdete z domu podívat na aktuální předpověď počasí ve svém smartphonu a podle toho se rozhodnout, zda budete potřebovat deštník. Vyžaduje to, ale opět provedení několika kroků. Místo toho je mnohem pohodlnější pouhý letmý pohled na deštník při cestě ven z domu (McEwen, 2013, s. 8).

Stručně řečeno, koncept IoT navrhuje mít spíše větší množství jednoúčelových přístrojů, než menší množství univerzálních výkonných zařízení (notebook, mobil, tablet).

## 1.1 Ubicomp

Ubiquitous computing, zkráceně ubicomp (volně přeloženo jako všudypřítomné počítače) je konceptuální předchůdce IoT. Ubicomp vnímá obrovské množství věcí, které by mohly obsahovat určitou formu počítačové technologie. Do ubicompu by tak například spadal i osvěžovač vzduchu, který vypustí vůni na základě zaznamenání pohybu v místnosti. IoT se tedy od ubicompu liší hlavně tím, že klade důraz na připojení k internetu.

Definice ubicompu tedy je mít výpočetní schopnosti v mnoha rozličných (možná i všech) objektech, přičemž definice IoT je mít tyto objekty připojené k internetu (Koreschhoff, 2013).

## 1.2 Historie

Přestože se o internetu věcí více mluví až v posledních letech, jeden z prvních příkladů se datuje do raných 80 let. Byl jím automat na Coca-Colu na Carnegie-Mellonově univerzitě v Pensylvánii. Bylo možné se k němu připojit přes internet a zjistit, zda jsou v něm nápoje, a zda jsou vychlazené.<sup>1</sup>

Pojem „Internet of Things“ a jeho definici poprvé použil Kevin Ashton, výkonný ředitel Auto-ID Labs na MIT v roce 1999 při prezentaci pro společnost Procter & Gamble. Ve svém projevu uvedl, že internet je závislý na obsahu tvořeném lidmi. Problém je ale ten, že lidé mají omezený čas, pozornost a přesnost. Kdybychom měli počítače, které by věděly všechno o všem za použití dat, které by samy získaly bez lidské pomoci, mohli bychom sledovat a počítat všechno kolem nás a redukovali bychom tak odpad, ztráty a náklady (Foote, 2016).

---

<sup>1</sup> <https://www.engineersrule.com/how-a-coke-machine-and-the-industrial-internet-of-things-can-give-birth-to-a-planetary-computer/>.

IoT dnes nabízí takřka nekonečné možnosti využití. Jde jen o to, jak dokážeme být kreativní.

### 1.3 Sítě pro IoT

Masivní rozvoj internetu věcí byl umožněn díky rozvoji infrastruktury a dostupnosti bezdrátového internetu v posledních letech. Nemusí se ale ani zdaleka jednat pouze o Wi-Fi. V následující tabulce je stručný přehled bezdrátových sítí používaných pro přenos dat.

**Tabulka 1 – Přehled bezdrátových sítí pro IoT**

	Krátký dosah				Dlouhý dosah		
	WiFi	BLE	ZigBee	Z-Wave	GSM/3G/4G	LoRa	SigFox
<b>Max. dosah signálu</b>	50 m	150 m	100 m	30 m	35 km (případně až 200 km u mobilní sítě HSPA)	15 km (2–5 km v zastavěných oblastech)	50 km (3–10 km v zastavěných oblastech)
<b>Max. počet napojených zařízení</b>	dle výrobce	200	1 000	232	tisíce	tisíce	tisíce
<b>Maximální velikost zprávy</b>	2 KB	27 bajtů	100 bajtů	64 bajtů	N/A	250 bajtů	12 bajtů (+limit odeslaných zpráv)
<b>Přenosová rychlost</b>	typicky 150–200 Mbit/s	1 Mbit/s	250 kbit/s	100 kbit/s	10 Mbit/s (u 4G)	50 kbit/s	100 bit/s
<b>Spotřeba energie</b>	vysoká	střední	střední	střední	vysoká	nízká	nízká
<b>Kde se síť například používá</b>	bezpečnostní kamery	termostaty	systémy pro chytrou domácnost (ovládání světel apod.)		alarmy	jednoduchá čidla – detektory kouře, zaplavení, oxidu uhelnatého, teploty, vlhkosti apod.	
<b>Poplatky nad rámec ceny zařízení</b>	jen za svůj internet	jen za svůj internet	za zakoupení a případně instalaci centrální jednotky		za tarif v rámci SIM	za připojení k LoRa	za připojení k Sigfox

*Zdroj: (Schafferová, 2017)*

Při výběru vhodné sítě pro IoT klademe důraz hlavně na dosah signálu a energetickou náročnost.

S narůstajícím počtem zařízení připojených k internetu narůstá i počet potřebných IP adres. Nynější verze internetového protokolu IPv4 umožňuje adresovat pouze cca 4 miliardy zařízení ( $2^{32}$ , prakticky však ještě mnohem méně kvůli podsítování a rezervaci části adres pro speciální použití). Tento problém však řeší nastupující verze protokolu IPv6. Ten

umožňuje adresovat cca  $2^{128}$  zařízení. Dick Guertin přirovnal toto číslo k povrchu země: „*Mohli bychom přiřadit IPv6 adresu KAŽDÉMU ATOMU NA POVRCHU ZEMĚ, a stále bychom měli dostatek adres pro dalších 100+ Zemí. Není ani zdaleka možné, že nám v budoucnosti IPv6 adresy dojdou*“ (Leibson, 2008).

## **1.4 Uplatnění a příklady použití**

Hlavní funkcí internetu věcí by mělo být jakékoliv zlepšení a zjednodušení života lidí. IoT nachází uplatnění ve všech oblastech. Ať už se jedná o jednotlivce, který chce sledovat teplotu vody v bazénu na chatě nebo továrnu s cílem zefektivnit výrobu.

### **Domácí automatizace a chytré domy**

Kontrola nad elektronickými systémy v domě. To může zahrnovat například světla, žaluzie, garážová vrata, vytápění, klimatizace, domácí audiosystém, zavlažování apod.

Důraz je kladen na úsporu energie a co největší soběstačnost domu. Tomu napomáhá i trend výroby vlastní elektřiny pomocí solárních panelů a solární ohřev vody.

V praxi to může vypadat tak, že všechny elektronické systémy budou napojené na internet a přístupné odkudkoli. Součástí domácí automatizace může být i zabezpečovací systém.

### **Wearables – nositelná elektronika**

Wearables je skupina zařízení, která se dají nosit na těle (musí k tomu být uzpůsobená). Může se jednat o různé chytré hodinky, brýle s rozšířenou realitou, headsety, boty měřící počet kroků nebo textil s displejem.

Velkou skupinou jsou různé měřiče aktivity pro sport (známé jako fitness trackery). Ty sledují a vyhodnocují naši aktivitu pomocí senzorů jako je akcelerometr a gyroskop. Náramky pro sport mají většinou vestavěný snímač srdečního tepu. Na základě těchto informací je pak možno vyhodnotit např. uběhnutou vzdálenost a počet spálených kalorií.

Nositelnou elektroniku lze využít i ve zdravotnictví. S její pomocí může být například pacient trpící srdeční arytmií monitorován online 24 hodin denně. V případě zvýšení nějaké hodnoty nad stanovenou mez se informace pošle přímo lékaři, který může ihned reagovat.

### **Venkovní senzory**

Zde se jedná o veškeré senzory veličin jako je například teplota, tlak, vlhkost, sluneční záření, hluk, znečištění ovzduší apod.

V praxi se můžeme setkat s měřiči kvality ovzduší, měřiči hladiny řek a rybníků, senzory seismické aktivity apod. Zvláště výhodné jsou ve špatně dostupných místech – v horách, jeskyních, ve špatně přístupném terénu.

Protože se tato zařízení většinou nemají možnost připojit k elektrické síti, je u nich potřeba zabezpečit dostatečně dimenzované napájení. To je většinou řešené baterií s dlouhou životností a solárním panelem pro její dobíjení. V těchto případech je také nutné, aby samotné zařízení mělo co nejmenší odběr proudu. Toho se dá docílit těmito způsoby:

- snížení doby, po kterou je zařízení aktivní – zvýšení intervalu přenosů,
- snížení množství přenášených dat,
- použití úspornější platformy a senzorů,
- výběr vhodné bezdrátové sítě.

Samostatnou kategorií jsou fotopasti. Ty bývají většinou instalované v lese a jsou vybavené senzorem pohybu. Když zaznamenají pohyb v zorném poli, začnou fotit nebo natáčet video. Pro snímání za šera a v noci využívají infračervený přísvit. Výsledné fotografie jsou pak bezdrátově odeslány (většinou pomocí mobilní sítě) majiteli fotopasti. Používají se zejména pro ochranu lesů proti pytlákům a pro monitorování vzácné zvěře.

## **Chytrá města**

Jedná se o poměrně nový koncept, který využívá informačních a komunikačních technologií k efektivnější správě města. Nejcitelnější vliv IoT lze pozorovat v dopravě. Umožňuje řízení dopravy na základě informací v reálném čase. Příkladem by mohlo být řízení semaforů a odklánění dopravy v závislosti na dopravní situaci.

V České republice je zatím s myšlenkou chytrého města nejdále Zlín. Kromě elektronických tabulí, které informují o aktuálních časech průjezdu městem je ve Zlíně i systém, který při rozpoznání zpoždění trolejbusu upraví signály světelných křižovatek tak, aby trolejbus nemusel zastavit a dohnal tak zpoždění. Systém je samozřejmě nastaven tak, aby nedošlo k rozpadu koordinace ostatních křižovatek. Další užitečná technologie umožňuje sledovat polohu a aktuální zpoždění všech linek MHD v reálném čase na internetu.<sup>2</sup>

## **1.5 Průmysl 4.0 a IoT (IIoT)**

### **IIoT**

Stejně jako IoT, i IIoT (Industrial Internet of Things – průmyslový internet věcí) je zaměřený na propojování zařízení mezi sebou. Zaměřuje se ale hlavně na průmyslové využití. Příkladem mohou být továrny, energetické sítě, doprava, zemědělství apod. IIoT také klade větší důraz na spolehlivost a přesnost.

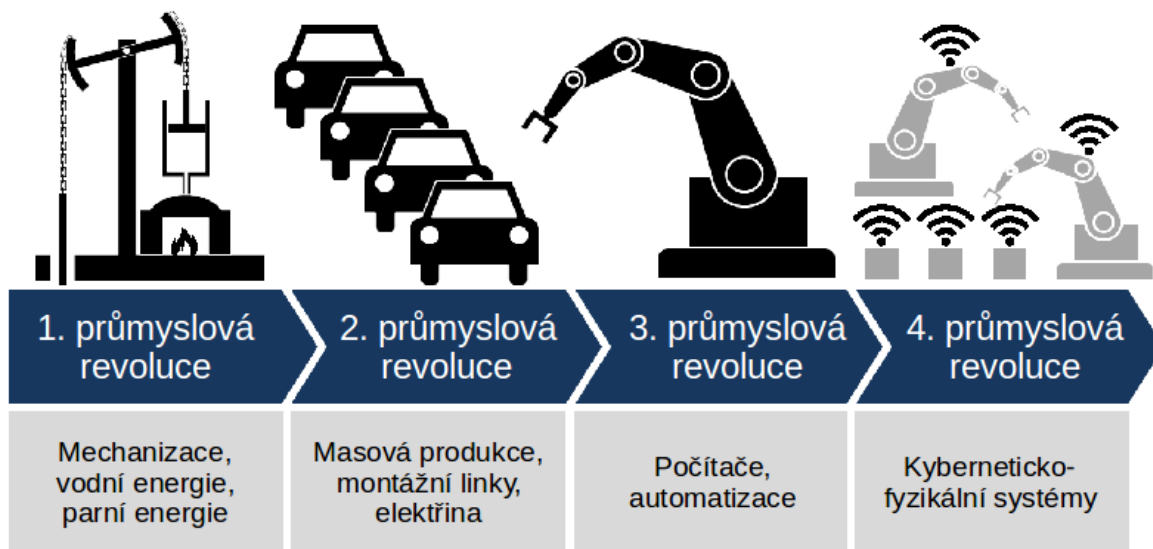
---

<sup>2</sup> [https://zlin.idnes.cz/smart-city-ve-zline-uherskem-hradisti-kromerizi-valasskem-mezirici-1g1-/zlin-zpravy.aspx?c=A170318\\_2313180\\_zlin-zpravy\\_ras](https://zlin.idnes.cz/smart-city-ve-zline-uherskem-hradisti-kromerizi-valasskem-mezirici-1g1-/zlin-zpravy.aspx?c=A170318_2313180_zlin-zpravy_ras).



## Průmysl 4.0

Termín Průmysl 4.0 je název pro současný trend automatizace, digitalizace, propojování zařízení a výměnu dat v průmyslovém prostředí. Často se hovoří i o tzv. čtvrté průmyslové revoluci. Na obrázku 2 jsou znázorněné 4 industriální revoluce v novodobých lidských dějinách.



Obrázek 2 – Diagram znázorňující 4 industriální revoluce včetně průmyslu 4.0

Zdroj: (Roser, 2017)

Stavebními kameny Průmyslu 4.0 jsou:

- kyberneticko-fyzikální systémy<sup>3</sup>,
- IIoT,
- strojové vnímání,
- cloud computing.

Cílem je samozřejmě úspora nákladů a zefektivnění výroby.

<sup>3</sup> Systém spolupracujících výpočetních prvků řídící fyzické subjekty.

## 2 ThingSpeak

ThingSpeak (<https://thingspeak.com/>) je internetová služba, která umožňuje sběr, vizualizaci a analýzu dat z IoT zařízení. Má napojení na Matlab a je tak vhodná i pro komplexnější analýzu dat. Data se nahrávají do tzv. kanálů, přičemž každý kanál může mít až 8 různých polí, ve kterých jsou uloženy jednotlivé záznamy. Přímo na webu lze data prohlížet a provádět základní správu kanálů, jako je import a export dat. Jednotlivé kanály mohou být označeny jako veřejné nebo jako privátní – k těm má pak přístup jen držitel klíče. Zvláštní klíč je potřeba i pro zápis dat do kanálu.

Používání služby je zdarma, avšak s omezenou funkcionalitou. Hlavní omezení se týkají počtu přenesených zpráv za rok. V bezplatné verzi je možné přenést 3 miliony zpráv za rok (cca 8200 za den). Dalším omezením je minimální interval mezi jednotlivými zprávami nastavený na 15 sekund.

Data lze na ThingSpeak nahrát pomocí REST API (Application Programming Interface – rozhraní pro programování aplikací) nebo MQTT API.

### 2.1 REST

REST (Representational State Transfer) je architektonický styl rozhraní pro webové služby. REST navrhnul a popsal Roy Fielding v roce 2000 v rámci svojí disertační práce.

Základní principy:

- Stav aplikace a její chování je vyjádřeno tzv. resourcem (zdrojem, zde například kanál),
- každý resource má unikátní identifikátor (URI),
- je definován jednotný přístup pro získání a manipulaci s resourcem v podobě čtyř operací CRUD (Create, Read, Update, Delete),
- resource může mít různé reprezentace (XML, HTML, JSON, CSV apod.).

Formální požadavky na rozhraní:

- Vztah server/ klient – oddělení odpovědností,
- bezstavost – každý požadavek musí obsahovat všechny informace nutné k jeho vykonání,
- cache – každý požadavek může být explicitně označený jako cacheovatelný či necacheovatelný, to umožňuje transparentně zvýšit výkonnost přidáním cache mezi klienta a server,
- vrstevnatost – klient neví, zda je připojen přímo k serveru nebo k prostředníkům po cestě. Tím se zvyšuje možnost variability (cache, rozložení zátěže, zabezpečení),
- code on demand (volitelně) – funkcionalita klienta může být rozšířena kódem, který zašle server (například JavaScript),

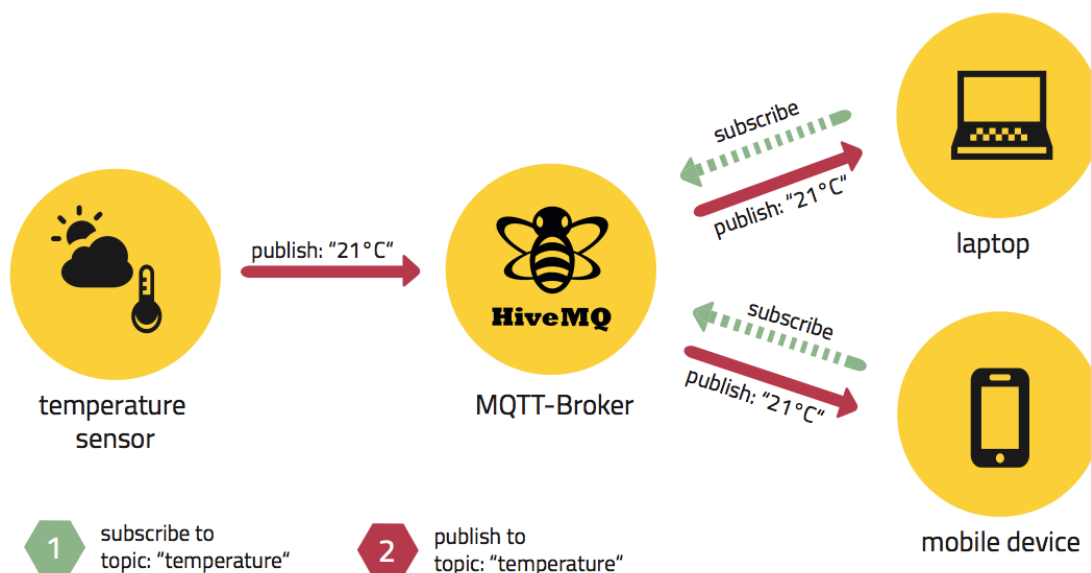
- jednotné rozhraní – základní požadavek. Většinou je REST založen na základních HTTP metodách GET, PUT, POST a DELETE.

Hlavní výhodou RESTu je jednoduché a jednotné rozhraní, škálovatelnost a nenáročnost na výpočetní výkon. Nevýhodou může být jeho bezestavost (Pichlik, 2007).

Rozhraním, které splňují požadavky na rozhraní REST se říká RESTful API.

## 2.2 MQTT

MQTT (Message Queuing Telemetry Transport) je jednoduchý a nenáročný protokol pro předávání zpráv mezi klienty. Je založen na návrhovém vzoru publish-subscribe (publikuj-odebírej), kdy odesílatel (publisher) neposílá zprávu přímo příjemci, ale tzv. brokeru – centrálnímu bodu. Ten data ukládá do témat (topic) a předává dál zařízením, která jsou přihlášená k odběru daného tématu (subscriber). Jedno zařízení může být současně subscriber i publisher v různých tématech (Malý, 2016).



Obrázek 3 – Ilustrace funkce MQTT protokolu

Zdroj: (The HiveMQ Team, 2014)

MQTT je nenáročný protokol, protože všechny jeho zprávy mají velmi málo servisních dat. Každá zpráva se skládá z pevné hlavičky, volitelné hlavičky, vlastní zprávy (která může mít velikost maximálně 256 MB) a informace o úrovni QoS (Quality Of Service).

QoS určuje, zda a jak se bude příjem zpráv potvrzovat. Přestože vyšší úrovně jsou více spolehlivé, mají vyšší latenci a nároky na přenesená data.

Jsou definovány tři různé úrovně QoS (Hillar, 2017, s. 45–46):

- QoS 0 (at most once – nejvýše jednou) – bez potvrzení o přijetí zprávy. Broker neukládá zprávu a není zde proto možnost opětovného odeslání při chybě.

- QoS 1 (at least once – alespoň jednou) – příjemce zprávy odešle odesílateli potvrzení o doručení zprávy (PUBACK). Pokud odesílatel nedostane potvrzení v určitém intervalu, odesílá zprávu znovu. To může vést k tomu, že subscriber dostane více kopií stejné zprávy.
- QoS 2 (exactly once – právě jednou) – používá 2 páry paketů (PUBLISH/PUBREC a PUBREL/PUBCOMP). Dva páry paketů zaručují, že zpráva bude doručena pouze jednou.

MQTT se skvěle hodí pro M2M<sup>4</sup> a IoT aplikace, například chytré domy, monitoring životního prostředí, logistiku nebo zdravotní péči.

Výhodou MQTT je jeho nenáročnost. Nevýhodou může být bezpečnost a fakt, že obsahem zprávy jsou binární data a spolupracující aplikace tak musí používat stejný dohodnutý formát dat (Rouse, 2018).

---

<sup>4</sup> M2M (Machine to machine – komunikace mezi stroji) je termín pro komunikaci dvou zařízení bez lidského zásahu.

### 3 Arduino

Arduino je název pro platformu okolo malých jednodeskových počítačů založených na mikrokontrolérech ATmega od firmy Atmel Corporation<sup>5</sup>. Nesnaží se nahradit počítač, jako například Raspberry Pi, ale je určeno především pro jednoúčelová zařízení. Jednou z jeho největších výhod je cenová dostupnost a jednoduchost, díky které s ním dokáže pracovat i člověk bez znalostí elektrotechniky a pokročilého programování.



Obrázek 4 – Logo Arduina

*Zdroj: (Howe, 2016)*

#### 3.1 Vznik

Základy projektu Arduino byly položeny v severní Itálii, ve městě Ivrea na absolventském programu Interaction Design Institute Ivrea. Studenti tehdy k výuce používali BASIC Stamp mikrokontroléry, které měly 2 zásadní problémy – byly relativně drahé a neměly dost výkonu pro některé projekty studentů. V roce 2003 vytvořil Hernando Barragán vývojovou platformu Wiring jako svoji diplomovou práci pod dohledem Massimo Banzí. Wiring obsahuje uživatelsky přívětivé vývojové prostředí a obvodovou desku připravenou ihned k použití.

Vedoucí práce Massimo Banzí a jeho kolegové věřili v otevřený software. Protože cílem projektu bylo vytvořit lehce dostupnou platformu, cítili, že bude nejlepší otevřít projekt pro co nejvíce zájemců ve formě open-source. Produkt, který tým vytvořil, byl vyroben z levných dílů. To umožňovalo, aby si lidé vytvářeli svoje vlastní desky. Klíčovým požadavkem také bylo zajistit možnost práce hned po vybalení z krabice. BASIC Stamp mikrokontroléry vyžadovaly ke svému používání tucty dalších zařízení a přípravků, které zvyšovaly výslednou cenu. Naproti tomu nová platforma umožňuje jednoduše připojit desku k počítači pomocí USB kabelu (Kushner, 2011).

Výsledný název projektu Arduino pochází z názvu baru ve městě Ivrea, kde se zakladatelé projektu scházeli. Bar se jmenuje podle krále Itálie z 11. Století (Arduin of Ivrea).

#### 3.2 Arduino hardware

Přestože Arduino je open-source, ochrannou známkou a právo používat název „Arduino“ má pouze společnost původních zakladatelů Arduino LLC. Ostatní výrobci většinou

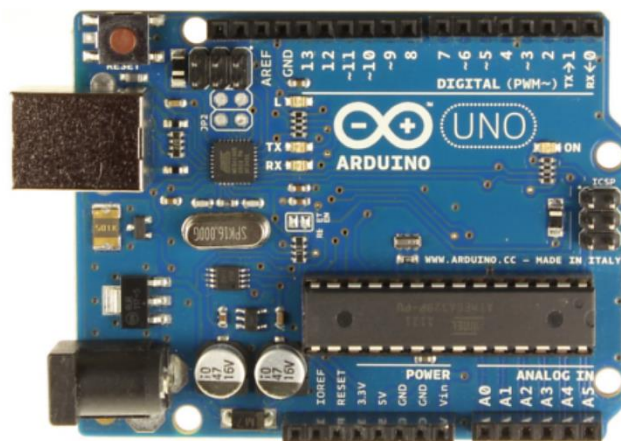
---

<sup>5</sup> Společnost Atmel byla v roce 2016 koupena konkurenční firmou Microchip Technology.

používají svůj název s koncovkou -duino (například Freeduino, Seeduino). Desky ostatních výrobců jsou navzájem kompatibilní s originálním návrhem.

Většina desek obsahuje 8-bitový mikrokontrolér Atmel (nejčastěji ATmega328, případně ATmega168). Jednotlivé čipy se liší velikostí paměti, funkcemi a počtem pinů. Programování probíhá pomocí ICSP<sup>6</sup> nebo pomocí sériové linky. Některé desky mají USB-serial převodníky pro programování skrz USB rozhraní.

Existuje mnoho oficiálních modelů Arduino desek. Nejrozšířenější je Arduino Uno, které je vhodné hlavně k prototypování. Menší Arduino Nano je díky svojí malé velikosti vhodné pro menší projekty. Největší Arduino Mega, které má více výstupů, je hojně používané například při stavbě 3D tiskáren. Existuje i další spousta jiných modelů pro specifické využití.



Obrázek 5 – Deska Arduino Uno

*Zdroj: (Voda, 2017, s. 6)*

Pro Arduino se vyrábí také tzv „shieldsy“ – rozšiřující desky, které se jednoduše připojí na stávající konektory. Umožňují jednoduché rozšíření funkcionality samotné desky například o kontroléry motorů, GSM komunikaci, slot pro SD kartu apod.

Arduino je kompatibilní s nepřeberným množstvím senzorů, komunikačních modulů, displejů apod. K připojení se dají využít digitální a analogové vstupně-výstupní piny. Periferie je také možné připojit přes I<sup>2</sup>C<sup>7</sup>, SPI<sup>8</sup> a 1-Wire<sup>9</sup> sběrnice nebo pomocí sériového rozhraní.

---

<sup>6</sup> In-circuit serial programming – schopnost programovatelných logických obvodů být programovány uvnitř elektronického obvodu bez nutnosti jejich vyjmutí.

<sup>7</sup> Inter-Integrated Circuit – synchronní sběrnice vyvinutá firmou Philips používaná k připojování nízkorychlostních periférií.

<sup>8</sup> Serial Peripheral Interface – synchronní sériové periferní rozhraní vyvinuté firmou Motorola.

<sup>9</sup> Sběrnice podobná I<sup>2</sup>C navržená firmou Dallas Semiconductor Corp. Charakteristickým rysem je možnost použití pouze 2 vodičů – jeden pro data a jeden pro zem.

### 3.3 Arduino software

Program pro Arduino může být napsán v jakémkoli programovacím jazyce, který lze zkompileovat do strojového kódu cílového mikroprocesoru.

K programování se nejčastěji používá vývojové prostředí Arduino IDE, které vychází z vývojových prostředí jazyků Processing a Wiring. Obsahuje editor s funkcemi formátování textu a zvýrazňování syntaxe. Dále nabízí jednoduché menu s možností kompilace a nahrání programu přímo do vývojové desky. Arduino IDE podporuje programování v jazyce C a C++.

Minimální program obsahuje pouze dvě základní funkce:

- Funkce `setup()` – tato funkce je zavolána pouze jednou při spuštění nebo restartu. Používá se pro inicializaci proměnných a nastavení vstupních a výstupních pinů.
- Funkce `loop()` – po dokončení funkce `setup()` je tato funkce volána opakovaně až do vypnutí nebo restartu. Vykonává se v ní hlavní program.

```
#define ledPin 13 // definice pinu, na kterém je připojena LED

void setup() {
  // nastavení pinu, na kterém je LED, jako výstupní
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH); // zapnout LED
  delay(1000);                // čekat 1 sekundu
  digitalWrite(ledPin, LOW);  // vypnout LED
  delay(1000);                // čekat 1 sekundu
}
```

Obrázek 6 – Ukázka zdrojového kódu pro Arduino

*Zdroj: vlastní*

Pro Arduino existuje řada knihoven dostupných jako open-source. Nejčastěji se používají pro komunikaci s připojenými periferiemi.

## 4 Návrh meteostanice

Protože na trhu neexistuje meteostanice, která by splňovala všechny požadavky (energetická soběstačnost, odesílání dat na internet přes GSM, nízká cena), bylo nutné navrhnout a sestavit vlastní zařízení. Hardwarová platforma Arduino je pro tento účel ideální. Existuje pro ní nepřeberné množství modulů, ať už se jedná o samotné desky s mikrokontrolery nebo různé senzory. Má okolo sebe také obrovskou komunitu lidí.

Vlastní meteostanice je složená z modulů kompatibilních s platformou Arduino. Důraz byl přitom kladen na nízkou spotřebu a možnost napájení z jedné Li-ion baterie (ta má nominální napětí 3,7 V).

Základem elektroniky je univerzální deska plošných spojů, na které jsou napájeny všechny hlavní komponenty. Schéma zapojení se nachází v příloze A, seznam použitých součástek v příloze B.

### 4.1 Arduino Pro Mini

Vývojová deska Arduino Pro Mini se používá zejména tam, kde je velmi málo místa. Stejně jako větší Arduino Uno používá mikrokontrolér ATmega328P. Pro úsporu místa a spotřeby není tato deska vybavená USB převodníkem, ke komunikaci s PC je tedy potřeba použít externí USB-UART převodník.

Tabulka 2 – Arduino Pro Mini specifikace

Arduino Pro Mini (ATmega328P)	
Napájecí napětí	5 V až 12 V (s regulátorem), 1,8 V až 5,5 V (bez regulátoru)
Pracovní kmitočet	8 MHz
Maximální odběr proudu	200 mA
Maximální proud IO <sup>10</sup> pinu	40 mA
Počet digitálních IO pinů	14
Počet analogových pinů	6
Flash paměť	32 KB
SRAM paměť	2 KB
EEPROM paměť	1 KB
Rozhraní	UART (sériové rozhraní), SPI, I <sup>2</sup> C

*Zdroj: zpracováno dle (Microchip Technology, 2018)*

Pro nižší spotřebu byl u desky odpájen regulátor napětí a LED dioda indikující zapnutí.

---

<sup>10</sup> IO (input/output), přesněji GPIO (general-purpose input/output) pin může sloužit jak pro vstup, tak i pro výstup. Jeho chování, včetně určení, zda se jedná o vstup nebo výstup lze řídit programově a to i v průběhu programu.



## 4.2 Senzory

Ke snímání absolutního tlaku a relativní vlhkosti slouží kombinované čidlo BME280, které dokáže měřit vlhkost, tlak a teplotu. Čidlo komunikuje pomocí I<sup>2</sup>C sběrnice. Ačkoliv čidlo měří i teplotu, v praxi se ukázalo, že jsou naměřené hodnoty teploty velmi nepřesné (liší se až o +3 °C). Hodnoty byly porovnávány s teplotními čidly DHT22, DS18B20 a LM35DZ a rtuťovým teploměrem. Stejně závěry lze nalézt i na internetových diskuzích. K měření teploty je tedy potřeba požit jiné čidlo.

Při měření tlaku ukazuje čidlo o 1 hPa více, než profesionální meteostanice umístěná ve stejné výšce. V programu je tedy tato hodnota použita jako kalibrační konstanta (odečte se od naměřené hodnoty).

Tabulka 3 – BME280 specifikace

<b>BME280</b>	
Napájecí napětí	1,7 V až 3,6 V
Maximální odběr proudu	714 μA
Rozhraní	I <sup>2</sup> C
Rozsah měření	Vlhkost: 0 % až 100 %, Tlak: 300 hPa až 1100 hPa, Teplota: -40 °C až +85 °C
Přesnost měření	Vlhkost: ±3 %, Tlak: ±1 hPa, Teplota: ±1 °C
Rozlišení	Vlhkost: 0,008 %, Tlak: 0,18 Pa, Teplota: 0,01 °C

*Zdroj: zpracováno dle (Bosch Sensortec, 2018)*

Protože má toto čidlo menší napájecí napětí, je třeba ho s ostatními komponenty propojit pomocí převodníku logických úrovní. Ten zabezpečí, že signály pro čidlo budou mít správnou úroveň. Snížení napětí pro napájení senzoru a převodníku úrovní je řešeno pomocí stabilizátoru napětí MCP1700, který má výstupní napětí 3,3 V.

Pro měření teploty slouží teplotní čidlo Dallas DS18B20. Jedná se o digitální čidlo komunikující pomocí 1-Wire sběrnice. Čidlo kromě klasického napájení podporuje i tzv. parazitní mód, kdy odebírá proud potřebný pro svoji činnost přímo ze sběrnice mikrokontroleru (vystačí si tedy pouze se dvěma vodiči). V meteostanici, ale bylo použito klasické napájení.

**Tabulka 4 – DS18B20 specifikace**

<b>DS18B20</b>	
Napájecí napětí	3 V až 5,5 V
Maximální odběr proudu	1,5 mA
Rozhraní	1-Wire
Rozsah měření teploty	-55 °C až +125 °C
Přesnost měření	±0,5 °C (-55 °C až +125 °C)
Rozlišení	Nastavitelné od 0,5 °C do 0,0625 °C

*Zdroj: zpracováno dle (Maxim Integrated Products, 2018)*

### 4.3 Komunikační modul GSM

Pro komunikaci s okolím je použit GSM modul Sim800L. Podporuje všechny hlavní GSM frekvence (850, 900, 1800 a 1900 MHz) a umožňuje připojení k internetu pomocí GPRS. Ke komunikaci s modulem po sériové lince se používají tzv. AT příkazy<sup>11</sup>. Ačkoliv lze modul napájet napětím až 4,4 V, sériové rozhraní pracuje s nižším napětím 3,3 V. RX pin je tedy potřeba k Arduinu připojit pomocí převodníku úrovní (TX pin lze připojit přímo, protože jeho úroveň je ještě v toleranci pro digitální pin Arduina).

**Tabulka 5 – Sim800L specifikace**

<b>Sim800L</b>	
Napájecí napětí	3,4 V až 4,4 V
Maximální odběr proudu	2 A (při nečinnosti 19 mA)
Rozhraní	Sériový port (AT příkazy)
Podporované frekvence	850 MHz, 900 MHz, 1800 MHz, 1900 MHz
Vysílací výkon	2 W (850 a 900 MHz), 1 W (1800 a 1900 MHz)

*Zdroj: zpracováno dle (Shanghai SIMCom Wireless Solutions, 2013)*

### 4.4 Napájení a spotřeba

Primárním zdrojem energie pro meteostanici je lithium-iontový akumulátor typu 18650 s kapacitou 2500 mAh a nominálním napětím 3,7 V. Jeho výhodou je vysoká kapacita vzhledem k rozměrům, nízké samovybití (2–5 % za měsíc) a poměrně jednoduché nabíjení.

<sup>11</sup> Speciální sada příkazů určená zejména pro modemy. Skládá se z krátkých textových příkazů, které se dají kombinovat.

Meteostanice je vybavena solárním článkem s nominálním napětím 6 V a výkonem 0,6 W (maximální proud 100 mA), pomocí kterého se nabíjí baterie. Do obvodu je taktéž potřeba přidat blokovací diodu proti zpětnému proudu do solárního článku pro případ, kdy je na solárním článku menší napětí než na baterii (například v noci, kdy na solární panel nesvítí slunce) – solární panel se pak chová jako spotřebič a zbytečně by spotřebovával energii z baterie. Aby byl úbytek napětí na diodě co nejmenší, používají se k tomuto účelu Schottkyho diody (ty mají úbytek napětí cca 0,3 V oproti 0,7 V u klasických diod).

Napětí ze solárního článku je nejprve nutné snížit na 5 V pro nabíjecí modul pomocí regulátoru napětí AMS1117. Snížené napětí pak napájí modul nabíječky Li-ion baterií s integrovaným obvodem TC4056A. Tento modul se stará o správné nabíjení baterie. Obsahuje také ochranu proti zkratu, přebití a podvybití baterie.

Nabíjecí proud je určen pomocí rezistoru R3 na desce modulu nabíječky. Pro meteostanici byl zvolen maximální nabíjecí proud 120 mA. Tato hodnota byla zvolena tak, aby se mohl akumulátor nabíjet plným proudem solárního panelu a zároveň se nabíjení ukončilo při poklesu nabíjecího proudu pod 12 mA a akumulátor se nepřebíjel (nabíjení končí automaticky při poklesu proudu pod 1/10 nastaveného nabíjecího proudu). V případě potřeby lze akumulátor nabít i pomocí microUSB konektoru, který se nachází přímo na modulu.

GSM modul, převodník úrovní, napěťové děliče a senzory jsou napájeny přes MOSFET tranzistor (STF40NF03L) ovládaný Arduinem. To umožňuje vypnout najednou všechny připojené zařízení a docílit tak velmi nízkého odběru proudu v režimu spánku. Proud odebíraný z akumulátoru je v tomto režimu pouze cca 320  $\mu$ A (odběr samotné desky v režimu deep sleep).

Průměrný odběr proudu meteostanice lze vyjádřit následujícím vzorcem:

$$I_{avg} = \frac{I_{act}T_{act} + I_{sleep}T_{sleep}}{T_{act} + T_{sleep}} \quad (1)$$

kde  $I_{act}$  = proud při aktivitě [A],  
 $I_{sleep}$  = proud v režimu spánku [A],  
 $T_{act}$  = doba trvání aktivity [s],  
 $T_{sleep}$  = doba v režimu spánku [s],  
 $I_{avg}$  = průměrný odběr proudu [A].

*Zdroj: vlastní*

Při aktivním měření a odesílání dat je odběr proudu 60 mA po dobu 30 sekund. Poté stráví meteostanice 870 sekund v režimu spánku s odběrem 320  $\mu$ A. Při použití vzorce (1) tak vychází průměrný odběr proudu na 2,3 mA.

Výdrž akumulátoru lze spočítat podle následujícího vztahu:

$$T_{max} = \frac{C_{bat}}{I_{avg}} \quad (2)$$

kde  $C_{bat}$  = kapacita baterie [Ah],  
 $I_{avg}$  = průměrný odběr proudu [A],  
 $T_{max}$  = výdrž baterie [h].

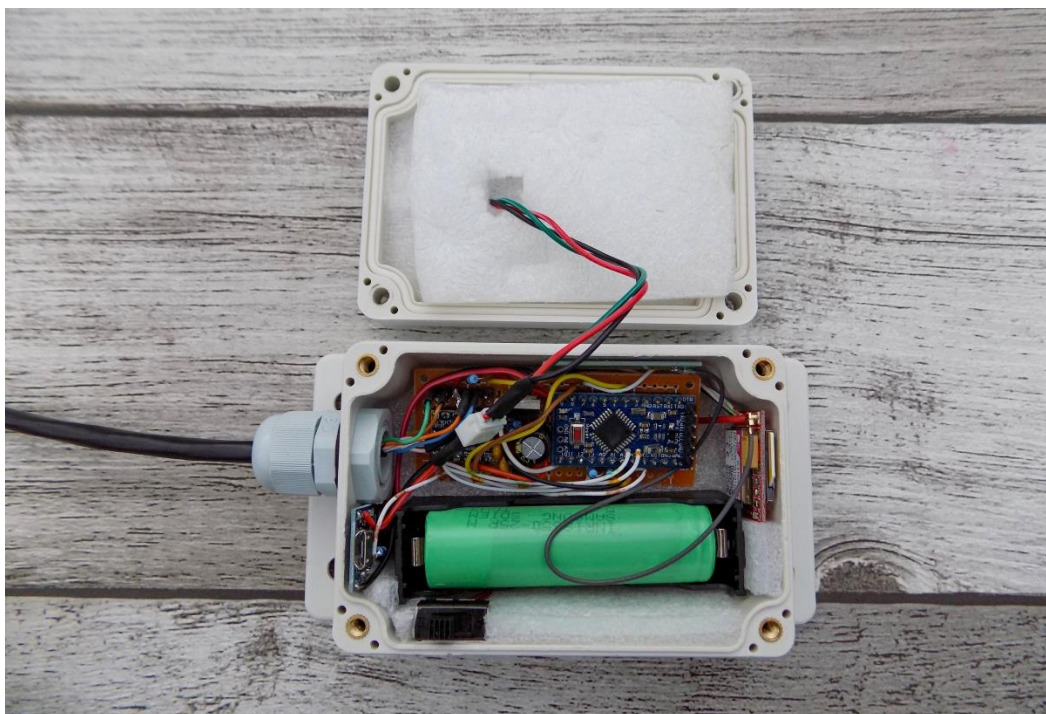
*Zdroj: vlastní*

Teoretická maximální výdrž akumulátoru (se zanedbáním samovybíjení, teploty apod.) v meteostanici bez nabíjení (v případě poruchy solárního panelu nebo extrémně nepříznivého počasí) se podle vzorce (2) rovná 1087 hodinám, tj. 45 dní.

#### 4.5 Konstrukce

Veškerá řídicí elektronika meteostanice je umístěna v plastové krabičce o rozměrech 100 mm x 68 mm x 50 mm (měřeno bez úchyťů). Krabička je vodotěsná se stupněm krytí IP66. Má po obou stranách úchyty s otvory pro šrouby a lze ji tak snadno uchytit například na zeď.

Velkou část objemu krabičky zabírá Li-ion akumulátor zmíněný v předchozí kapitole. Ten je umístěn v držáku a lze ho tedy jednoduše vyměnit. Po stranách se pak nachází vypínač, modul nabíječky Li-ion akumulátorů, GSM modul a anténa. Na jedné stěně se také nachází průchodka pro kabel, kterým jsou připojeny senzory. Ve víku se pak nachází regulátor napětí a blokovácí dioda. Dno i víko krabičky je vyplněno tepelnou izolací z polyethylenu. To pomáhá udržovat optimální teplotu baterie v létě i zimě.



**Obrázek 7 – Elektronika meteostanice**

*Zdroj: vlastní*

Solární panel má rozměry 80 mm x 55 mm a je přilepen k víku krabičky pomocí silikonového lepidla. K elektronice je připojen pomocí konektoru, víko lze tedy úplně sundat. Přední strana solárního panelu je chráněna průhlednou fólií (bez ní by mohla průhledná krycí vrstva panelu zoxidovat a snížit tak efektivitu panelu).



Obrázek 8 – Senzory, krabička s elektronikou a radiační štít

*Zdroj: vlastní*

Senzory jsou s řídicí elektronikou propojeny pomocí běžně dostupného UTP kabelu, který je určen pro venkovní použití. Kabel obsahuje 4 páry vodičů. Pro účely meteorostanice se užívá 5 vodičů – Napájecí napětí, zem, signál SCL, signál SDA a signál pro 1-wire teplotní čidlo. Samotné senzory jsou upevněné na závitové tyči s krytkou. Lze je pak tedy jednoduše vsunout a zacvaknout do otvoru ve spodu radiačního štítu.

Současná podoba meteorostanice je druhou verzí návrhu (fotografie první verze se nachází v příloze C). Předchozí verze měla následující nedostatky:

- menší krabička – horší manipulace a nemožnost použít tepelnou izolaci,
- natrvalo připájený akumulátor – do menší krabičky se nevešel držák akumulátoru,
- špatně vyřešený regulátor napětí ze solárního panelu – regulátor měl malou účinnost,
- špatně nastavený nabíjecí proud – byl nastaven na příliš vysokou hodnotu, při které kleslo napětí, nabíjení se přerušilo, napětí opět vzrostlo a tento cyklus se opakoval,
- použití pouhé diody místo regulátoru pro senzory vyžadující nižší napětí,
- solární panel připevněný pomocí suchého zipu – protože solární panel zakrýval šrouby víka krabičky, musel být k víku připevněn pomocí suchého zipu aby bylo možné krabičku otevřít.

## 4.6 Radiační štít

Čidla musí být umístěna tak, aby nedocházelo k ovlivňování naměřených hodnot. To znamená, že nesmí být umístěna v krabici s elektronikou, aby nedocházelo například k zahřívání teplotního čidla odpadním teplem z elektroniky. Čidla také nesmí být umístěna na přímém slunci, měří se vždy ve stínu.

Musí být také zajištěn pohyb vzduchu kolem čidel. V profesionálních meteostanicích se používá ventilátor, který žene vzduch kolem čidel. To ale není v případě omezených zdrojů energie možné.

Řešením je umístění čidel do tzv. radiačního krytu. Ten má většinou podobu bílého dutého válce s lamelami po obvodu. Poskytuje tak stín a zároveň umožňuje proudění vzduchu kolem čidel.

Protože jsou prodávané radiační kryty špatně dostupné a poměrně drahé, jeví se jako nejlepší možnost vlastní výroba. Jako „lamely“ lze použít bílé jídelní misky nebo malé květináče. V jejich středu se vyřízne otvor na senzory. Pak už stačí jenom po obvodu vyvrtat díry a lamely spojit závitovou tyčí a matkami. Jedna z lamel se ponechá bez díry a bude sloužit jako vrchní díl.



Obrázek 9 – Konstrukce radiačního štítu

*Zdroj: vlastní*

Co se týká samotného umístění, je doporučeno měřit ve výšce 2 m od země nad travnatým povrchem a z dosahu budov, asfaltových nebo betonových ploch a větších předmětů, které by mohly pohlcovat sluneční paprsky.



## 5 Program a přenos dat z meteostanice

Program pro Arduino je napsaný v programovacím jazyce C++ pomocí Arduino IDE. Stará se o získávání dat z čidel a jejich následný přenos do služby ThingSpeak. Data jsou na server přenášena pomocí REST API, protože je jednodušší ho implementovat než MQTT a zároveň nejsou potřeba žádné pokročilejší funkce tohoto protokolu.

Vykonávání programu můžeme rozdělit do následujících kroků:

1. Inicializace GSM modulu a čidel,
2. změření sledovaných hodnot,
3. výpis hodnot,
4. odeslání dat na server,
5. usnutí zařízení na stanovený čas.

Program začíná připojením potřebných knihoven, definicí konstant a globálních proměnných a vytvořením instancí tříd z knihoven. Poté následuje funkce *setup*, která se spouští vždy jednou po zapnutí nebo restartu zařízení. V ní se nejprve nastaví IO pin kontrolující MOSFET jako výstupní a poté se mu nastaví logická 1, čímž se zapnou všechna zařízení připojená k MOSFETu. Po pětisekundové pauze následuje inicializace GSM modulu, zjištění síly mobilního signálu a v případě úspěchu i připojení k přístupovému bodu operátora pro přístup k internetu pomocí GPRS. Dále se ještě inicializuje senzor BME280.

Pokud se z jakéhokoliv důvodu připojení ke GPRS nepodaří na první pokus (například kvůli výpadku mobilní sítě nebo slabému signálu), arduino se na 10 sekund uspí a pokusí se připojit znovu. Pokud se to ani na pátý pokus nepodaří, celé zařízení se uspí na 15 minut a program začne od začátku.

Ve funkci *loop* se nejprve změří napájecí napětí samotného mikrokontroleru (pomocí funkce *getVCC*), napětí na solárním panelu<sup>12</sup> a napětí na snímači nabíjecího proudu, ze kterého se vypočítá nabíjecí proud. Poté se získají hodnoty teploty, tlaku a vlhkosti z čidel. U naměřených hodnot se kontroluje, zda jsou v očekávaném rozmezí.

V případě, že je meteostanice připojena k PC sériovým portem, vypíší se všechny hodnoty do konzole. Následně se pomocí funkce *poslat* data odešlou na ThingSpeak.

---

<sup>12</sup> Protože je napětí na solárním panelu obvykle větší, než je napájecí napětí mikrokontroleru, musí se toto napětí měřit přes odporový dělič. Ten je tvořen dvěma stejnými odpory a výsledné napětí je tedy poloviční. Naměřená hodnota se tedy musí vynásobit dvěma.

```

17:32:01.768 -> start programu
17:32:16.559 -> inicializace uspesna..
17:32:19.332 -> GPRS pripojeno
17:32:19.750 -> BME280 senzor inicializovan
17:32:20.686 -> ----sila signalu = 17
17:32:20.686 -> ----napajeci napeti = 4.18 V
17:32:20.721 -> ----napeti solarniho panelu = 0.13 V
17:32:20.756 -> ----nabijeci proud solarniho panelu = 0.00 mA
17:32:20.825 -> ----teplota = 19.06 C
17:32:20.860 -> ----tlak = 992.82 hPa
17:32:20.860 -> ----vlhkost = 33.94 %
17:32:24.927 -> -> Pripojeni na server api.thingspeak.com uspesne!
17:32:28.538 -> -> Odpojeni datoveho spojeni.
17:32:28.572 -> Prechod do rezimu spanku

```

Obrázek 10 – Příklad výpisu naměřených hodnot do konzole

Zdroj: vlastní

Funkce *poslat* nejprve naváže TCP spojení se serverem `api.thingspeak.com`. Pokud se spojení z jakéhokoliv důvodu nepodaří, proběhne opět 5 pokusů jako u připojení ke GPRS. V případě úspěšného spojení se zformuluje REST požadavek na zapsání dat do kanálu. Klíč k zápisu dat a samotná data jsou odeslána jako parametr požadavku. Po odeslání dat dojde k ukončení datového spojení.

```

// vytvoření odesílané zprávy pro ThingSpeak
String zprava = "GET /update?api_key=";
zprava += APIkey;
zprava += "&field1=";
zprava += teplota;
zprava += "&field2=";
zprava += vlhkost;
zprava += "&field3=";
zprava += tlak;
zprava += "&field4=";
zprava += silaSignalu;
zprava += "&field5=";
zprava += napajeciNapeti;
zprava += "&field6=";
zprava += napetiSolar;
zprava += "&field7=";
zprava += proudSolar;
zprava += " HTTP/1.0\r\n\r\n";

```

Obrázek 11 – Ukázka vytvoření zprávy REST požadavku

Zdroj: vlastní

Dále následuje už jenom usnutí meteostanice pomocí funkce *uspat*. Ta používá knihovnu *Narcoleptic* k přepnutí mikrokontroléru do režimu hlubokého spánku s minimálním odběrem proudu. Zároveň se vypne MOSFET, aby senzory a GSM modul neodebíraly žádný proud. Doba spánku je nastavena na 870 sekund<sup>13</sup>, protože vykonání programu trvá 30 sekund a interval mezi měřeními je 900 sekund (15 minut). Po probuzení dochází

<sup>13</sup> V kódu je ale tato hodnota nastavena na 910 sekund, protože se intení časovač mikrokontroleru nepatrně zrychluje.



k restartu mikrokontroleru a program začíná od začátku. Restart programu lze vynutit i stisknutím tlačítka reset na desce mikrokontroleru nebo vypnutím a zapnutím vypínače.

GSM modul používá standartní SIM karty formátu microSIM, jaké se používají v mobilních telefonech. Karta může být od jakéhokoliv operátora, je ovšem potřeba v programu upravit přihlašovací údaje pro připojení k internetu, které se zpravidla u každého operátora liší. V meteostanici je použita SIM karta od mobilního operátora Vodafone. Pro fungování meteostanice je třeba mít aktivovaný datový tarif. Je prakticky jedno jaký, objem přenášených dat je velmi malý (v řádu několika MB za měsíc). V současnosti nabízejí čeští operátoři vhodné tarify s cenou od 50 Kč za měsíc.

## 6 Měřené veličiny

Meteostanice přímo měří teplotu, absolutní tlak a relativní vlhkost vzduchu. Jsou to základní meteorologické údaje, které by měla zaznamenávat každá měřící stanice.

### 6.1 Teplota vzduchu

Teplota vzduchu je základní veličina udávající tepelný stav ovzduší, tj. schopnost vzduchu přijímat, nebo předávat tepelnou energii. Je měřena ve stupních Celsia (značeny °C). Správně by se měla měřit ve výšce 2 metry nad zemí.

### 6.2 Tlak

Atmosferický tlak (někdy nazýván barometrický) je tlak, který je způsoben tíhou sloupce vzduchu nad určitou plochou. Se zvyšující se nadmořskou výškou se atmosferický tlak snižuje, protože na povrch působí menší sloupec. V meteorologii se nejčastěji používá jednotka hektopascal (hPa).

Senzory tlaku obvykle měří tzv. *absolutní tlak*. Takto naměřený tlak je ale závislý na nadmořské výšce. Protože je potřeba porovnávat výsledky měření z různých míst v různé nadmořské výšce, je absolutní tlak přepočítáván na relativní hodnotu vztaženou k hladině moře. *Relativní tlak* lze vypočítat pomocí následujícího vztahu (Keisan, 2018):

$$P_0 = P \left( 1 - \frac{0,0065h}{t+0,0065h+273,15} \right)^{-5,257} \quad (3)$$

kde  $P_0$  = relativní tlak přepočtený na hladinu moře [Pa],

$P$  = absolutní tlak na stanici [Pa],

$h$  = nadmořská výška stanice [m],

$t$  = teplota na stanici [°C].

### 6.3 Vlhkost

Vlhkost vyjadřuje množství vodní páry ve vzduchu. Při dané teplotě je vzduch schopen pojmout jen určité množství vodní páry. Čím je vzduch teplejší, tím více vlhkosti může pojmout.

Nejčastěji používanou charakteristikou je tzv. *relativní vlhkost vzduchu*. Vyjadřuje poměr mezi skutečným množstvím vodní páry ve vzduchu a maximálním množstvím vodní páry ve vzduchu při určité teplotě. Vyjadřuje se v procentech. 100% relativní vlhkost znamená, že vzduch už nemůže přijmout více vodní páry a při poklesu teploty dochází ke zkapalnění přebytečného množství vodních par, viz též rosný bod níže (Smolka, 2013).

### 6.4 Rosný bod

Pokud teplota poklesne a vzduch již není schopen pojmout původní množství vlhkosti (vzduch je plně nasycen, relativní vlhkost je 100%), přebytek zkondenzuje (jako

zkapalněná vodní pára). V praxi to můžeme pozorovat například v koupelně při sprchování. V koupelně stoupá teplota a vlhkost, ale například zrcadlo má teplotu nižší, než je rosný bod a proto na něm vodní pára zkondenzuje. V přírodě můžeme rosný bod pozorovat například večer, kdy se ochladí vzduch a přebytečná vodní pára kondenzuje ve formě rosy nebo jíní (Smolka, 2013).

Teplotu rosného bodu lze spočítat z relativní vlhkosti a teploty podle následujícího vzorce (Lawrence, 2005, s. 226):

$$t_d = \frac{B_1 \left[ \ln\left(\frac{RH}{100}\right) + \frac{A_1 t}{B_1 + t} \right]}{A_1 - \ln\left(\frac{RH}{100}\right) - \frac{A_1 t}{B_1 + t}} \quad (4)$$

kde  $t_d$  = teplota rosného bodu [°C],

$t$  = teplota [°C],

$RH$  = relativní vlhkost [%],

$A_1$  = konstanta [7,625],

$B_1$  = konstanta [243,04 °C].

## 6.5 Napětí na baterii a solárním panelu, síla signálu

Meteostanice též měří napětí na solárním panelu. To je ovlivněno intenzitou slunečního svitu a odebíraným proudem. Při jasné obloze ve dne se obvykle hodnota napětí pohybuje kolem 6,5 V (při nabíjení baterie 5,5 V). Naopak v noci se napětí rovná nule a panel nevyrabí žádnou elektřinu. Měřeno je i napětí na baterii a nabíjecí proud. Napětí by mělo být stále konstantní (zhruba 4,1–4,2 V). Nabíjecí modul umožňuje měřit proud, kterým nabíjí baterii. Dá se tak snadno zjistit, jestli se baterie nabíjí.

Další údaj, který se z meteostanice přenáší, je síla mobilního signálu. Tento údaj se může hodit v případě, kdy meteostanice neposílá data ve stanovených intervalech.

Nabíjecí proud a síla signálu se sice zaznamenávají a odesílají, ale ve webové aplikaci se nezobrazují. Je možné se na ně podívat přímo ve webové administraci ThingSpeaku.

## 7 Webová aplikace

Webová aplikace je umístěna na internetové adrese <http://meteo-pardubice.cz>. Je vytvořena pomocí HTML5, CSS3 a JavaScriptu. Aplikace je zaměřena na jednoduché a srozumitelné zobrazení naměřených dat z meteostanice.

Webová aplikace podporuje všechny nejrozšířenější webové prohlížeče (Google Chrome, Mozilla Firefox, Microsoft Edge a Opera) a jejich mobilní verze. Microsoft Internet Explorer 11 a nižší nejsou podporovány.

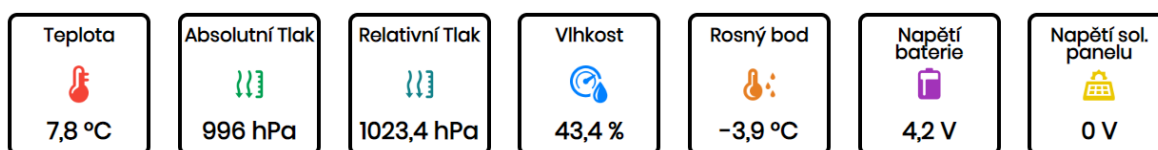
### 7.1 HTML a CSS

Vzhled webové stránky je založen na volně dostupné šabloně z webu [w3schools.com](http://w3schools.com)<sup>14</sup>. Web se skládá pouze z jedné hlavní stránky, která je členěna pomocí menu. To zajišťuje její snadnou správu a přehlednost.

Design stránky je responzivní – stránka se přizpůsobuje zařízení, na kterém je prohlížena. Například menu se na velkých obrazovkách zobrazuje v plné velikosti po levé straně. Na zařízeních s menší obrazovkou (tablety, mobilní telefony) je menu srolované v horní části a nezabírá tak zbytečně místo. Na mobilních telefonech se také z důvodu špatné čitelnosti nezobrazuje graf zobrazující všechny veličiny najednou.

#### Rychlý přehled – aktuálně naměřené hodnoty

V úvodu stránky se zobrazují aktuální (poslední naměřené) hodnoty veličin včetně času posledního měření. Pro zobrazení oken s ikonou a aktuální hodnotou je použit CSS Grid Layout (systém rozvržení založený na mřížce – řádcích a sloupcích). To umožňuje dynamicky měnit počet oken na řádku podle rozlišení zobrazovacího zařízení. Pod okny jsou pak umístěny jednotlivé grafy.



Obrázek 12 – Ukázka zobrazení oken s aktuálními naměřenými hodnotami

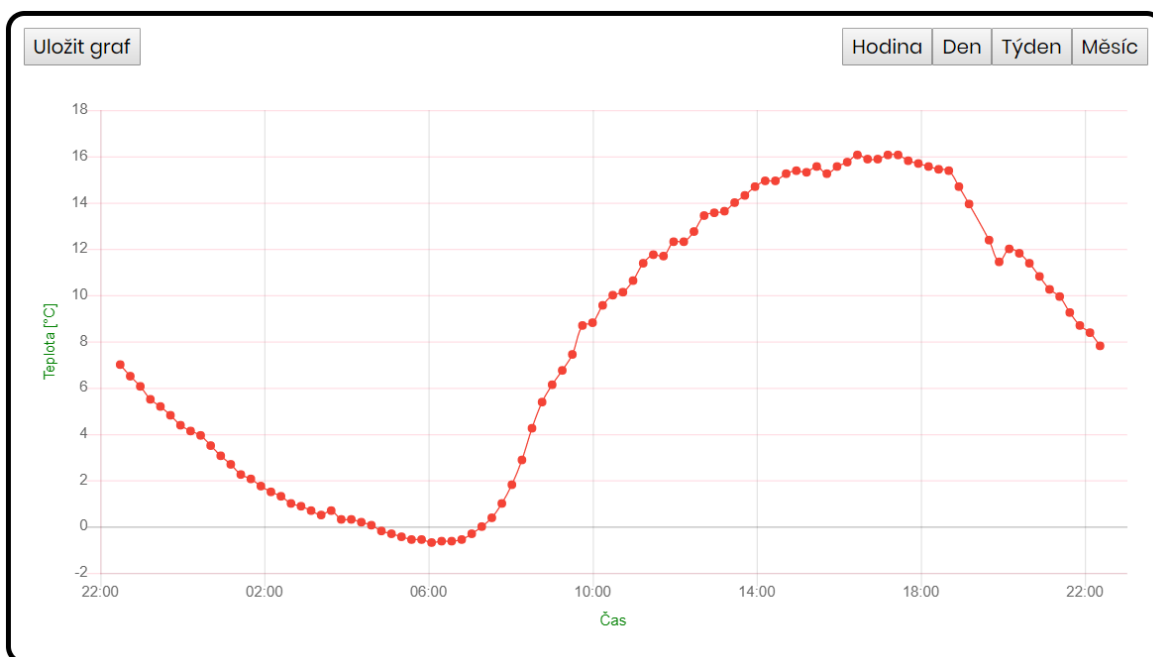
*Zdroj: vlastní*

#### Grafy naměřených hodnot

Každý graf se zobrazuje na individuálním canvas prvku (prvek k dynamickému skriptovatelnému vykreslování bitmap). Na svislé ose se vynáší veličina s jednotkou, na vodorovné pak časový úsek. Měřitko os se automaticky přizpůsobuje zobrazovaným hodnotám. Grafy jsou interaktivní – po najetí myši nad datový bod se zobrazí jeho přesná hodnota a čas pořízení. Vpravo nahoře nad grafem je vždy čtveřice tlačítek, kterými se dá

<sup>14</sup> Dostupné na [https://www.w3schools.com/w3css/w3css\\_templates.asp](https://www.w3schools.com/w3css/w3css_templates.asp).

volit zobrazovaný časový úsek (hodina, den, týden, měsíc). Grafy lze také uložit ve formě obrázku pomocí tlačítka *uložit graf* vlevo nahoře<sup>15</sup>.



Obrázek 13 – Ukázka zobrazení grafu s ovládacími prvky

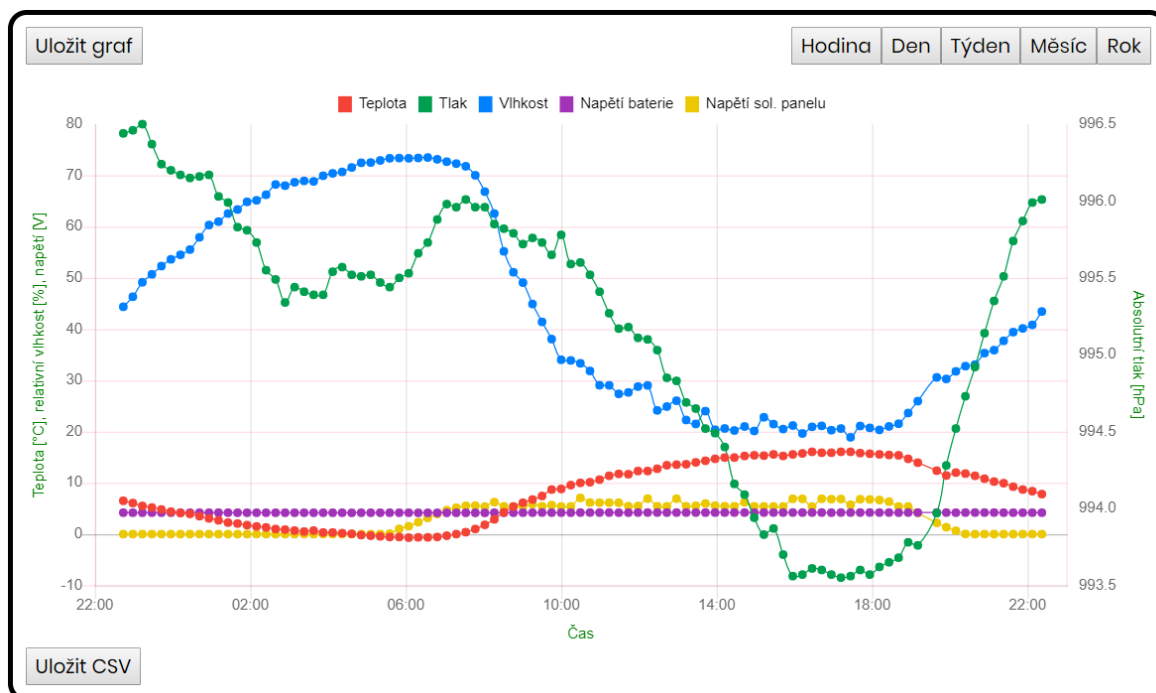
*Zdroj: vlastní*

## Společný graf

Jako první je na stránce umístěn graf zobrazující všechny veličiny najednou (multigraf). Tento graf má 2 svislé osy. Levá osa slouží pro zobrazení teploty, vlhkosti a napětí. Pravá osa pak slouží k zobrazení tlaku. Jednotlivé sady hodnot lze zobrazovat a skrývat pomocí kliknutí na legendu v záhlaví grafu.

U společného grafu je kromě hodiny, dne, týdne a měsíce možné zvolit i období jednoho roku. Vlevo dole se pak nachází tlačítko *Uložit CSV*, pomocí kterého lze uložit aktuálně zobrazená data ve formátu CSV. Na stránce dále následují samostatné grafy pro jednotlivé měřené veličiny.

<sup>15</sup> Ukládání nefunguje v prohlížeči Microsoft Edge.



Obrázek 14 – Ukázka zobrazení multigrafu

Zdroj: vlastní

## Údaje za předchozí den

Pod grafem teploty, tlaku a vlhkosti se zobrazuje malá informační tabulka s údaji za předchozí den. Je zde uvedena průměrná, maximální a minimální hodnota.

## 7.2 Javascript

Javascriptová část aplikace se stará o získávání dat z ThingSpeaku a jejich zobrazování uživateli. K vytváření grafů je použita knihovna Chart.js<sup>16</sup>. Pro práci s časem je použita knihovna Moment.js<sup>17</sup>.

Po načtení stránky je spuštěn skript, který postupně spouští funkce, které načítají a zobrazují data. Data ze služby ThingSpeak jsou získávána pomocí REST API.

Nejprve se pošle HTTP požadavek na konkrétní data v určitém formátu. Je možné si vybrat mezi formáty XML, JSON a CSV. V parametru požadavku je pak možné specifikovat, kolik záznamů požadujeme, přesné časové období, případně je možné vypočítat průměr za určitý čas, výsledky zaokrouhlit apod. Příklad:

`https://api.thingspeak.com/channels/374007/feeds.json?days=7&average=240`

Tento požadavek vrátí data za posledních 7 dní z kanálu číslo 374007 ve formátu JSON. Nebude ale obsahovat všechny hodnoty, ale jejich průměr počítaný každých 240 minut.

<sup>16</sup> Dostupná z <https://www.chartjs.org/>.

<sup>17</sup> Dostupná z <https://momentjs.com/>.

Po odeslání požadavku program nečeká na odpověď, ale pokračuje dále. O případné odpovědi je informována obslužná funkce události, která v případě, že od serveru dorazí odpověď, spustí funkci pro zpracování odpovědi. Přijatá odpověď ve formě prostého textu se poté převádí do JSON formátu, se kterým dále každá funkce pracuje jinak podle požadované funkcionality.

```
{
  "channel": {
    "id": 374007,
    "name": "Meteostanice Pardubice",
    "description": "Datový kanál pro projekt bezdrátové
    Meteostanice.",
    ...
    "elevation": "222",
    "last_entry_id": 30418
  },
  "feeds": [
    {
      "created_at": "2019-04-16T19:23:08Z",
      "entry_id": 30321,
      "field1": "9.94",
      "field2": "37.72",
      "field3": "995.51",
      "field4": "18",
      "field5": "4.18",
      "field6": "0.00",
      "field7": "0.00"
    }
  ]
}
```

Obrázek 15 – Ukázka JSON formátu

*Zdroj: vlastní*

Získávání dat ze serverů ThingSpeaku může mít určitou prodlevu. U dat za hodinu, den, týden a měsíc je to obvykle do 0,5 s. U dat za rok to však může být až 10 sekund. Jako indikace, že se na pozadí něco děje, je přes graf při načítání dat zobrazen rotující kruh, který znázorňuje čekání na odpověď.

Jako první se spustí funkce *nactiPrehled*. Ta se stará o zobrazení oken s posledními naměřenými hodnotami. Ze serveru se načte pouze poslední záznam. Čas jeho pořízení se vypíše jako čas poslední aktualizace dat. Funkce dále počítá rosný bod a relativní tlak (viz kapitola 7). Všechny údaje jsou pak vypsány do korespondujících elementů v HTML kódu.

Jako další následuje funkce *nactiPredchoziDen*, která počítá statistické údaje tří hlavních měřených veličin za předchozí den. Je počítána vždy minimální, maximální a průměrná denní hodnota. Vypočítané hodnoty se vypíší do elementů v HTML kódu.

Poté se spustí funkce *nactiMultigraf*. Pomocí této funkce se načtou data pro zobrazení společného grafu.

Data pro období hodiny, dne, týdne a měsíce lze získat přímo pomocí REST požadavku. Při nastavení časového úseku na 1 rok to už však není možné, protože ThingSpeak API umožňuje vrátit průměry za maximálního časový úsek 1440 minut (1 den). K vykreslení grafu ale potřebujeme průměry za celý měsíc. Zpracování dat je tedy náročnější. Nejprve je

potřeba získat denní průměry za posledních 12 měsíců, rozdělit tato data do jednotlivých měsíců a vypočítat průměry za měsíc.

Po úspěšném načtení dat se spustí funkce *zobrazMultigraf*, která data vykreslí do grafu.

Následně se spouští funkce *NactiData* pro každou měřenou veličinu zvlášť. V parametru se předává požadovaná veličina (teplota, tlak, vlhkost, napětí baterie, napětí solárního panelu). Po načtení dat z požadovaného období se spouští funkce pro zobrazení grafu konkrétní veličiny.

Zobrazení grafů začíná vždy zjištěním začátku a konce časové osy. Poté se nastaví forma zobrazování legendy podle nastaveného období a graf se vykreslí do požadovaného HTML canvas prvku. Minimum a maximum svislé osy se určuje automaticky podle rozsahu zobrazovaných hodnot.

Funkce *prepniObdobi* přepíná zobrazované období jednotlivých grafů – hodina, den, týden, měsíc. U společného grafu je navíc tlačítko pro výběr časového období 1 rok. Parametry funkce jsou název veličiny a požadované období. Tato funkce je volána z HTML po stisknutí tlačítek u grafu. Výchozí zobrazované období je jeden den.

Funkce *UlozGraf* uloží požadovaný graf do bitmapového obrázku ve formátu PNG. Uloží se vždy to, co je aktuálně vidět (ukládá se obsah canvas prvku). Název souboru je určen názvem veličiny a vybraným obdobím. Parametrem funkce je název veličiny.

Funkce *ulozCSV* uloží data aktuálně zobrazená v multigrafu do CSV souboru. Data se nejprve načtou ze serveru, odpověď se zpracuje, vytvoří se z ní blob objekt a poté je nabídnuta uživateli ke stažení ve formě souboru s příponou csv.

Data lze v aplikaci pouze prohlížet. Případná editace dat je možná pouze ve webové administraci služby ThingSpeak. Ani zde však není možné data jednoduše měnit nebo mazat. Nejprve je nutné exportovat všechny záznamy do CSV formátu. Poté je možné buď pomocí tabulkového editoru nebo jakéhokoliv textového editoru upravit, případně vymazat libovolné záznamy. V dalším kroku je nutné vymazat všechna data z kanálu. Následně je možné nahrát upravený CSV soubor zpátky na server.

U práce s daty je potřeba dát si pozor na to, že ThingSpeak ukládá data s časem ve formátu UTC+0<sup>18</sup>. Je ale potřeba je zobrazovat jako časy v časovém pásmu UTC+1 (zimní čas) a UTC+2 (letní čas). Pozor je nutné dávat zejména při exportu a importu dat.

Dalším problémem je oddělování desetinných míst. Programovací jazyky používají tečku, v Česku se ale používá čárka. Je teda potřeba tento znak nahradit ve všech grafech a výpisech.

Zdrojové kódy jsou přiloženy k bakalářské práci na CD (příloha D).

---

<sup>18</sup> UTC (Coordinated Universal Time – koordinovaný světový čas) je základem systému občanského času. Jednotlivá časová pásma jsou definována svými odchylkami od UTC.



## 8 Umístění a srovnání s profesionální stanicí

Meteostanice je umístěna na zahradě rodinného domu v Pardubicích ve výšce 224 m n. m.<sup>19</sup>. Bohužel ji nebylo možné umístit na ideální místo a nachází se tedy na garáži. Radiační kryt se senzory je přišroubován ke zdi a krabička se solárním panelem je upevněna na střeše.



Obrázek 16 – Současné umístění meteostanice

*Zdroj: vlastní*

Naměřená data byla porovnána s daty z profesionální stanice ČHMÚ, která má název H3PARD01. Ta je umístěna v areálu pardubického letiště v nadmořské výšce 224,21 m. Stanice kromě teploty vzduchu, tlaku a relativní vlhkosti vzduchu měří například i rychlost větru, srážky, oblačnost nebo teplotu půdy. Obě stanice jsou vzdálené zhruba 1 km.

Srovnání proběhlo pomocí hodinových záznamů obou stanic ze dne 8. 4. 2019. Data z profesionální meteostanice byla získána ze stránek ČHMÚ<sup>20</sup> a v tabulkách a grafech jsou označena jako referenční.

<sup>19</sup> Zjištěno z API webové aplikace mapy.cz: <https://api.mapy.cz/view?page=altitude>.

<sup>20</sup> Dostupné zde: <http://portal.chmi.cz/aktualni-situace/aktualni-stav-pocasi/ceska-republika/stanice/professionalni-stanice/prehled-stanic/pardubice>.

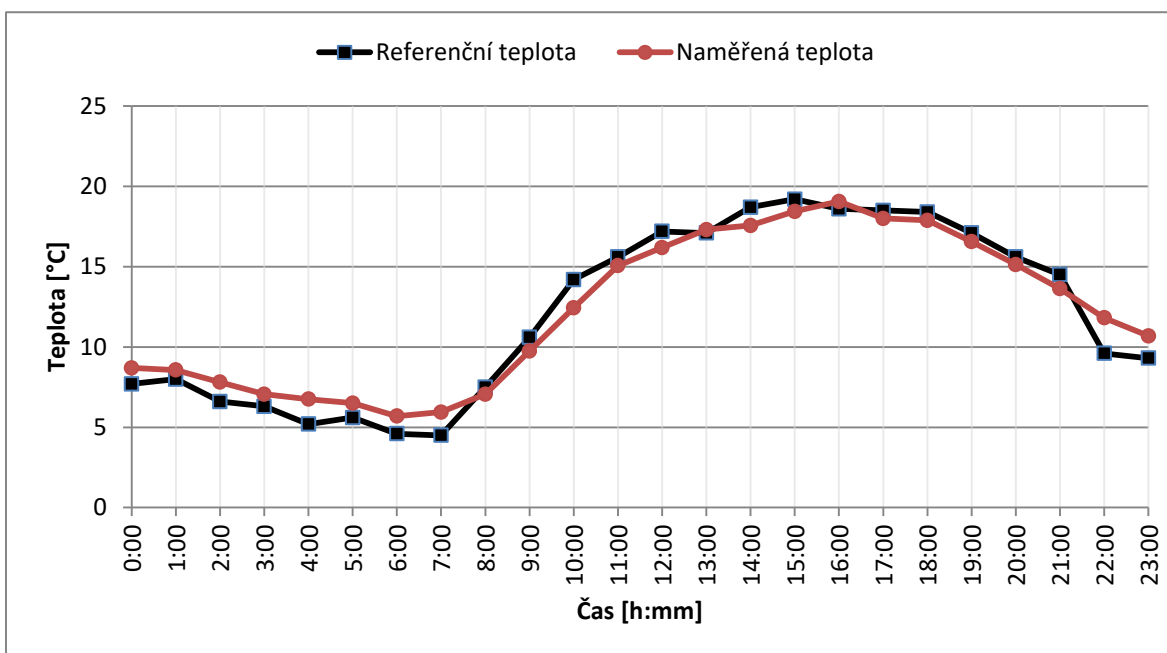
Tabulka 6 – Srovnání naměřených hodnot teploty, abs. tlaku a rel. vlhkosti

čas [h:mm]	Referenční teplota [°C]	Naměřená teplota [°C]	Odchylka [°C]	Referenční abs. tlak [hPa]	Naměřený tlak [hPa]	Odchylka [hPa]	Referenční rel. vlhkost [%]	Naměřená rel. vlhkost [%]	Odchylka [%]
0:00	7,7	8,7	1,0	981,1	981,2	0,1	75	64	-11
1:00	8,0	8,6	0,6	981,0	981,3	0,3	73	66	-7
2:00	6,6	7,8	1,2	981,2	981,4	0,2	80	68	-12
3:00	6,3	7,1	0,8	981,3	981,5	0,2	78	71	-7
4:00	5,2	6,8	1,6	981,3	981,6	0,3	84	73	-11
5:00	5,6	6,5	0,9	981,5	981,7	0,2	82	73	-9
6:00	4,6	5,7	1,1	981,7	981,9	0,2	86	76	-10
7:00	4,5	5,9	1,4	981,9	982,4	0,5	85	76	-9
8:00	7,5	7,1	-0,4	982,4	982,6	0,2	76	72	-4
9:00	10,6	9,8	-0,9	982,3	982,5	0,2	65	62	-3
10:00	14,2	12,4	-1,8	982,4	982,5	0,1	52	53	1
11:00	15,6	15,1	-0,5	982,0	982,0	0,0	50	45	-5
12:00	17,2	16,2	-1,0	981,6	981,7	0,1	39	39	0
13:00	17,1	17,3	0,2	981,5	981,5	0,0	38	35	-3
14:00	18,7	17,6	-1,1	981,1	981,1	0,0	36	33	-3
15:00	19,2	18,4	-0,8	980,7	980,7	0,0	34	32	-2
16:00	18,6	19,1	0,5	980,5	980,4	-0,1	35	31	-4
17:00	18,5	18,0	-0,5	980,4	980,4	0,0	32	31	-1
18:00	18,4	17,9	-0,5	980,6	980,5	-0,1	35	33	-2
19:00	17,1	16,6	-0,5	980,7	980,7	0,0	39	36	-3
20:00	15,6	15,1	-0,5	981,2	981,2	0,0	43	41	-2
21:00	14,5	13,6	-0,9	981,8	981,9	0,1	48	47	-1
22:00	9,6	11,8	2,2	982,0	982,1	0,1	67	53	-14
23:00	9,3	10,7	1,4	982,1	982,2	0,1	68	58	-10

*Zdroj: vlastní*

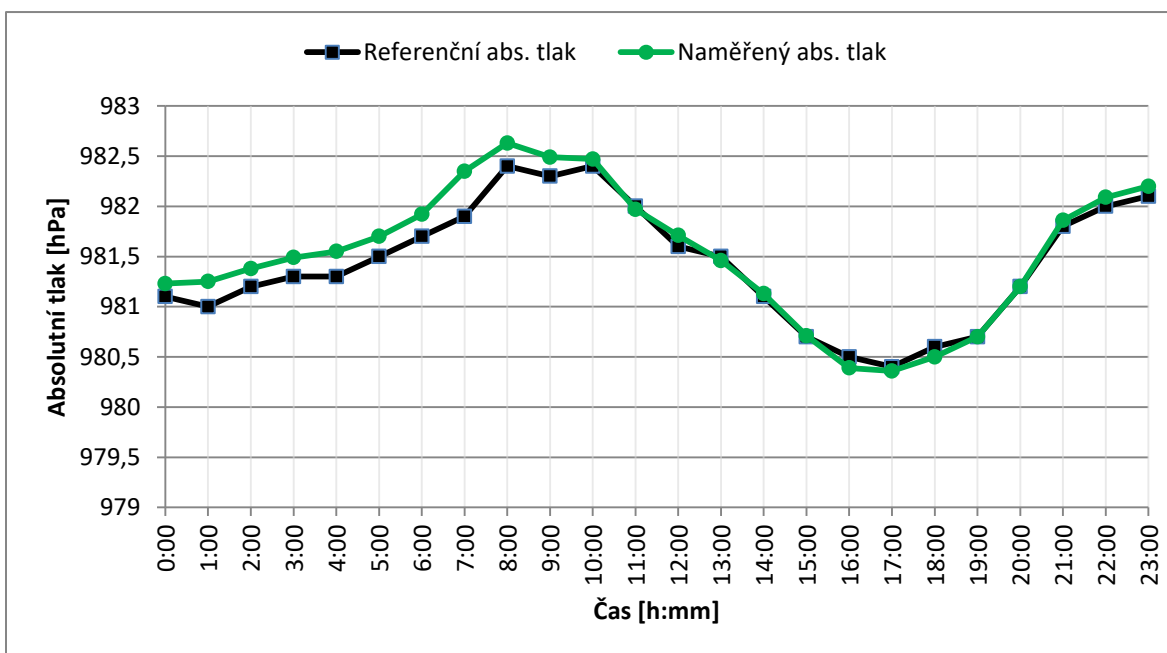
Ze srovnání vyplývá, že průměrná absolutní odchylka teploty je 0,92 °C, průměrná absolutní odchylka absolutního tlaku 0,13 hPa a průměrná absolutní odchylka relativní vlhkosti 5,52 %.

Odchylky naměřených hodnot jsou způsobeny jak možnou nepřesností senzorů, tak i vzdáleností obou meteostanic a jejich umístěním.



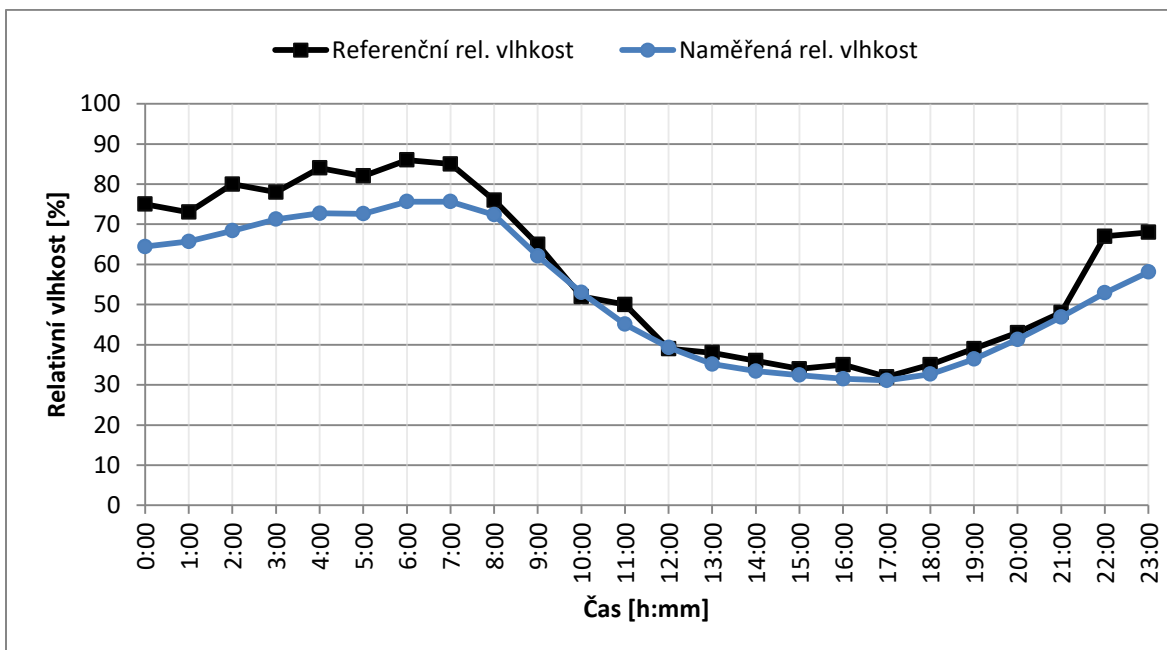
Obrázek 17 – Graf srovnání naměřených hodnot pro teplotu

*Zdroj: vlastní*



Obrázek 18 – Graf srovnání naměřených hodnot pro abs. tlak

*Zdroj: vlastní*



Obrázek 19 – Graf srovnání naměřených hodnot pro rel. vlhkost

*Zdroj: vlastní*

## Závěr

Práce se v teoretické části zaměřila na popis konceptu Internetu věcí, internetovou službu ThingSpeak a hardwarovou platformu Arduino.

Cílem práce bylo navrhnout a realizovat vlastní bezdrátovou meteostanici. Při stavbě meteostanice byly použity moduly kompatibilní s platformou Arduino. Sensory byly vybrány s ohledem na co nejvyšší přesnost měření (viz kapitola Návrh meteostanice).

Vzniklo několik verzí, které postupně řešily nedostatky těch předchozích (pomalé nabíjení, složitá konstrukce, nevyměnitelný akumulátor). Při návrhu bylo hojně využíváno rad skvělé komunity okolo platformy Arduino.

Výsledkem návrhu je zařízení schopné měřit teplotu, relativní vlhkost a absolutní tlak. Jeho hlavní výhodou je energetická soběstačnost a možnost umístit ho prakticky kdekoliv v dosahu mobilního signálu. Do budoucna je možné meteostanici rozšířit o další senzory (například senzor rychlosti a směru větru, měření intenzity slunečního záření, dešťových srážek). Součástí návrhu je i konstrukce vlastního radiačního štítu, který zajišťuje, aby byly senzory chráněny před nepříznivým počasím a nedopadalo na ně přímé sluneční záření.

Dalším cílem práce byl vývoj webové aplikace pro zobrazení naměřených dat. Samotná data jsou uložena na serverech služby ThingSpeak, ve webové aplikaci se pouze zobrazují. Při vývoji aplikace bylo nejsložitější upravit data tak, aby se s nimi dalo dále pracovat a zobrazit je do grafů. Bylo taktéž nutné upravit zobrazení pro české prostředí (zejména používání čárek místo teček k oddělení desetinných míst). Webová aplikace se nachází na veřejné URL adrese <http://meteo-pardubice.cz/>.

Srovnání naměřených dat realizované meteostanice s profesionální stanicí ČHMÚ dopadlo podle očekávání (viz Tabulka 6). Ze srovnání vyplývá, že průměrná absolutní odchylka teploty je 0,92 °C, průměrná absolutní odchylka absolutního tlaku 0,13 hPa a průměrná absolutní odchylka relativní vlhkosti 5,52 %. Odchylky naměřených hodnot jsou způsobeny jak možnou nepřesností senzorů, tak i vzdáleností obou meteostanic a jejich umístěním.

## Literatura

BLY, Jennifer, 2013. Connected Devices Accelerate The Need For IPv6 In The Internet Of Things. In: *Teamarin* [online]. 27. 12. 2013 [cit. 2017-11-26]. Dostupné z: <https://teamarin.net/2013/12/27/connected-devices-accelerate-the-need-for-ipv6-in-the-internet-of-things/>.

BOSCH SENSORTEC, 2018. *BME280: Combined humidity and pressure sensor*. Katalogový list. Zář 2018 [cit. 2018-11-25]. Dostupné z: [https://ae-bst.resource.bosch.com/media/\\_tech/media/datasheets/BST-BME280\\_DS002.pdf](https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BME280_DS002.pdf).

FOOTE, Keith D, 2016. A Brief History of the Internet of Things. In: *Dataiversity: Data Education for Business and IT Professionals* [online]. 16. 8. 2016 [cit. 2017-11-19]. Dostupné z: <http://www.dataiversity.net/brief-history-internet-hings/>.

HILLAR, Gaston C, 2017. *MQTT Essentials – A Lightweight IoT Protocol*. Birmingham, United Kingdom: Packt Publishing Limited, 2017. ISBN 9781787287815.

HOWE, Pat, 2016. Getting started with the Arduino microcontroller. In: *Pat Howe: Technology Website* [online]. 2. 3. 2016 [cit. 2019-02-01]. Dostupné z: <https://pathowe.co.uk/getting-started-with-the-arduino-microcontroller/>.

KEISAN, 2018. Conversion to sea-level pressure Calculator. In: *Keisan Online Calculator* [online]. 24. 1. 2018 [cit. 2018-08-24]. Dostupné z: <https://keisan.casio.com/keisan/image/Convertpressure.pdf>.

KORESHOFF, Treffyn, 2013. The difference between the IoT and Ubicomp. In: *Treffyn Koreshoff: The Internet of Things {Interaction & Innovation}* [online]. 3. 11. 2013 [cit. 2017-11-19]. Dostupné z: <http://treffyn.com/blog/the-difference-between-the-iot-and-ubicomp-2/>.

KUSHNER, David, 2011. The Making of Arduino: How five friends engineered a small circuit board that's taking the DIY world by storm. In: *IEEE Spectrum* [online]. 26. 10. 2011 [cit. 2018-04-23]. Dostupné z: <https://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino>.

LAWRENCE, Mark G, 2005. The Relationship between Relative Humidity and the Dewpoint Temperature in Moist Air: A Simple Conversion and Applications. *Bulletin of the American Meteorological Society* [online]. 1. 2. 2005, 86(2), 225-234 [cit. 2018-08-29]. DOI: 10.1175/BAMS-86-2-225. ISSN 0003-0007. Dostupné z: <http://journals.ametsoc.org/doi/10.1175/BAMS-86-2-225>.

LEIBSON, Steve, 2008. IPV6: How Many IP Addresses Can Dance on the Head of a Pin? In: *EDN Network* [online]. 28. 3. 2008 [cit. 2017-11-19]. Dostupné z: <https://www.edn.com/electronics-blogs/other/4306822/IPV6-How-Many-IP-Addresses-Can-Dance-on-the-Head-of-a-Pin->.

- MALÝ, Martin, 2016. Protokol MQTT: komunikační standard pro IoT. In: *Root.cz* [online]. 29. 6. 2016 [cit. 2018-08-13]. Dostupné z: <https://www.root.cz/clanky/protokol-mqtt-komunikacni-standard-pro-iot/>.
- MAXIM INTEGRATED PRODUCTS, 2018. *DS18B20: Programmable Resolution 1-Wire Digital Thermometer*. Katalogový list. Zář 2018 [cit. 2018-11-25]. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>.
- McEWEN, Adrian a Hakim CASSIMALLY, 2013. *Designing the Internet of things*. Chichester: Wiley. ISBN 9781118430620.
- MICROCHIP TECHNOLOGY, 2018. *ATmega48A/PA/88A/PA/168A/PA/328/P*. Katalogový list. 2018 [cit. 2018-11-25]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf>.
- PICHLIK, Roman, 2007. A REST. In: *Dagblog* [online]. 16. 10. 2007 [cit. 2018-08-13]. Dostupné z: <https://dagblog.cz/a-rest-c5156313d79e>.
- ROSER, Christoph, 2017. Industry 4.0 – What Works, What Doesn't. In: *AllAboutLean* [online]. 3. 10. 2017 [cit. 2017-11-26]. Dostupné z: <http://www.allaboutlean.com/industry-4-0-potentials/>.
- ROUSE, Margaret, 2018. MQTT (MQ Telemetry Transport). In: *IoT Agenda* [online]. Únor 2018 [cit. 2018-08-15]. Dostupné z: <https://internetofthingsagenda.techtarget.com/definition/MQTT-MQ-Telemetry-Transport>.
- SCHAFFEROVÁ, Magdalena, 2017. Jak se vyznat v záplavě sítí pro internet věcí. In: *Zooco* [online]. 20. 10. 2017 [cit. 2017-11-19]. Dostupné z: <https://www.zooco.io/blog/2017/10/20/jak-se-vyznat-v-zaplave-siti-pro-internet-veci/>.
- SHANGHAI SIMCOM WIRELESS SOLUTIONS, 2013. *SIM800L Hardware Design*. Katalogový list. 20. 8. 2013 [cit. 2018-11-25]. Dostupné z: [https://img.filipeflop.com/files/download/Datasheet\\_SIM800L.pdf](https://img.filipeflop.com/files/download/Datasheet_SIM800L.pdf).
- SMOLKA, Václav, 2013. Vlhkost vzduchu a její charakteristiky. In: *In-Počasi* [online]. 8. 10. 2013 [cit. 2018-08-26]. Dostupné z: <https://www.in-pocasi.cz/clanky/teorie/vlhkost-vzduchu/>.
- THE HIVEMQ TEAM, 2014. *MQTT 101 – How to Get Started with the lightweight IoT Protocol*. In: HiveMQ [online]. 22. 10. 2014 [cit. 2018-08-15]. Dostupné z: <https://www.hivemq.com/blog/how-to-get-started-with-mqtt/>.
- VODA, Zbyšek, 2017. *Průvodce světem Arduina*. Vydání druhé. Bučovice: Martin Stříž, 2017. ISBN 978-80-87106-93-8.

## Seznam příloh

<b>Příloha A – Schéma zapojení.....</b>	<b>49</b>
<b>Příloha B – Seznam použitých součástek .....</b>	<b>50</b>
<b>Příloha C – Fotografie.....</b>	<b>51</b>
<b>Příloha D – Přiložené CD.....</b>	<b>53</b>





## Příloha B – Seznam použitých součástek

Tabulka B.1 – Seznam použitých elektronických součástek

Označení	Hodnota	Popis	Počet
R1	220R	Rezistor metalizovaný, 1/4W	1
R2, R3, R4	10K	Rezistor metalizovaný, 1/4W	3
R5	4K7	Rezistor metalizovaný, 1/4W	1
C1	1000 $\mu$ F/6,3V	Kondenzátor elektrolytický	1
C2, C3, C4, C6, C7	100nF	Kondenzátor keramický	5
C5	1 $\mu$ F	Kondenzátor keramický	1
D1	MBR20100CT	Schottkyho dioda	1
SC1	6V/100mA	Solární článek	1
BAT1	3,7V/2500mAh	Li-ion akumulátor, typ 18650	1
MOSFET1	STF40NF03L	MOSFET tranzistor, n-kanál	1
SW1	SL19-121	Posuvný spínač 2-pólový	1

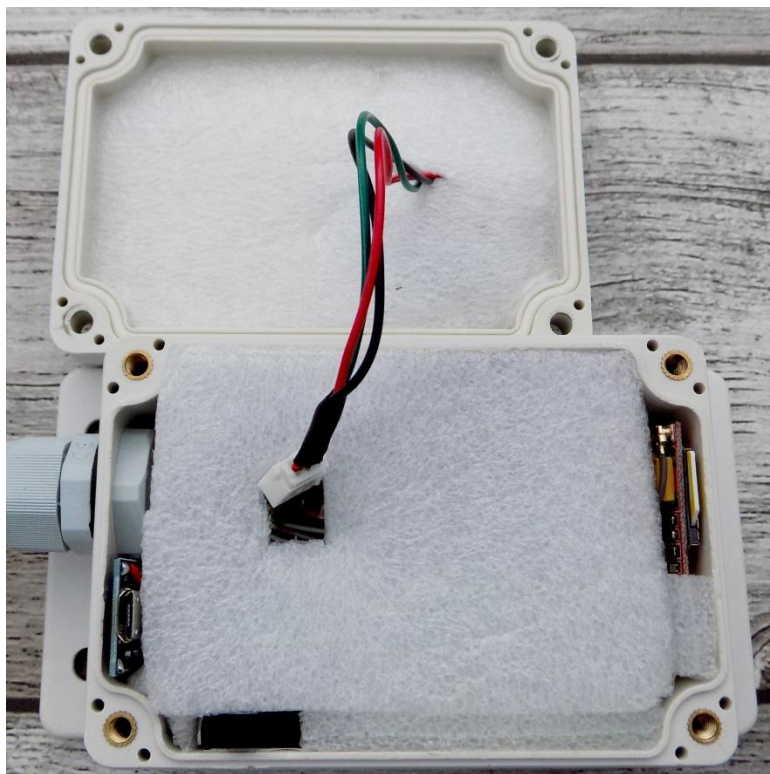
*Zdroj: vlastní*

Tabulka B.2 – Seznam použitých modulů

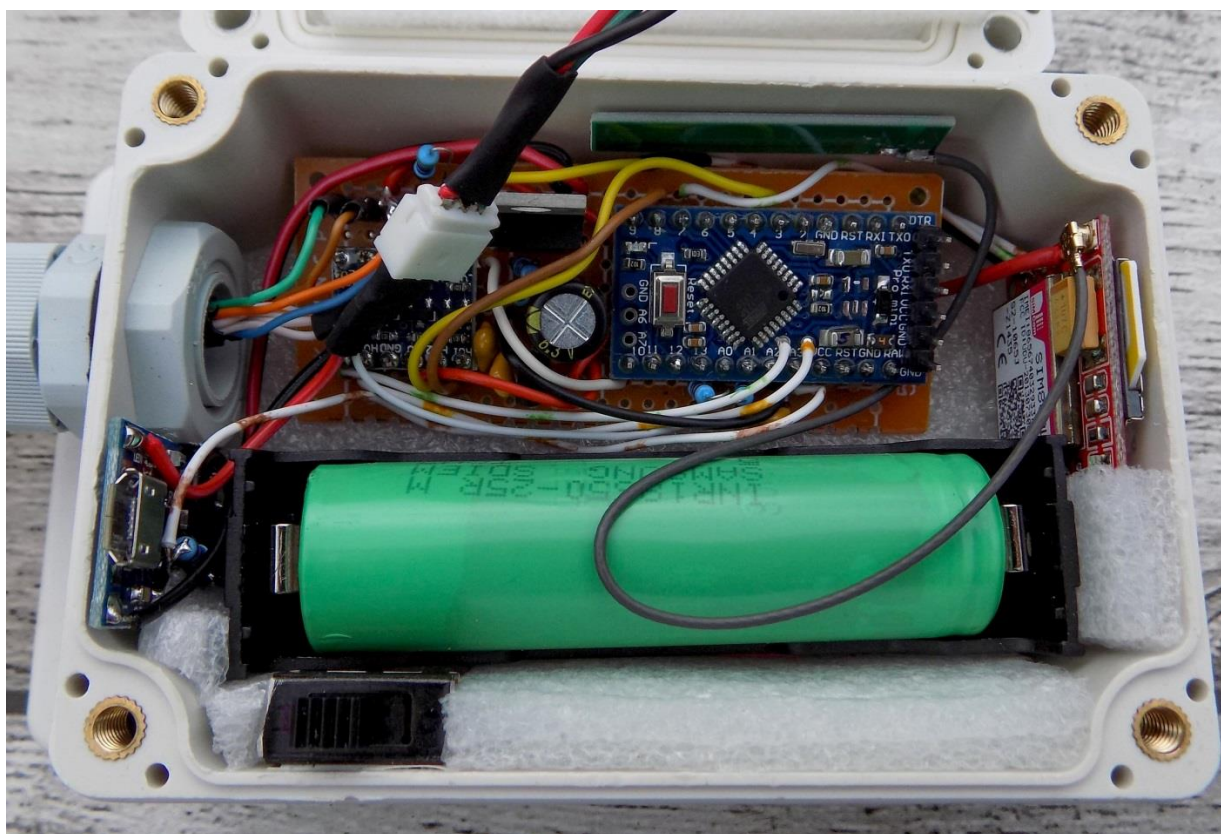
Označení	Popis
U1	Arduino Pro Mini (ATmega328P)
U2	Regulátor napětí (AMS1117-5.0)
U3	Regulátor napětí (MCP1700T-3.3)
U4	BME280 čidlo tlaku, vlhkosti a teploty
U5	Převodník logických úrovní pro I <sup>2</sup> C
U6	DS18B20 digitální teplotní čidlo
U7	SIM800L GSM modul
U8	Li-ion nabíječ (TC4056A)

*Zdroj: vlastní*

## Příloha C – Fotografie



Obrázek C.1 – Detail izolace krabičky

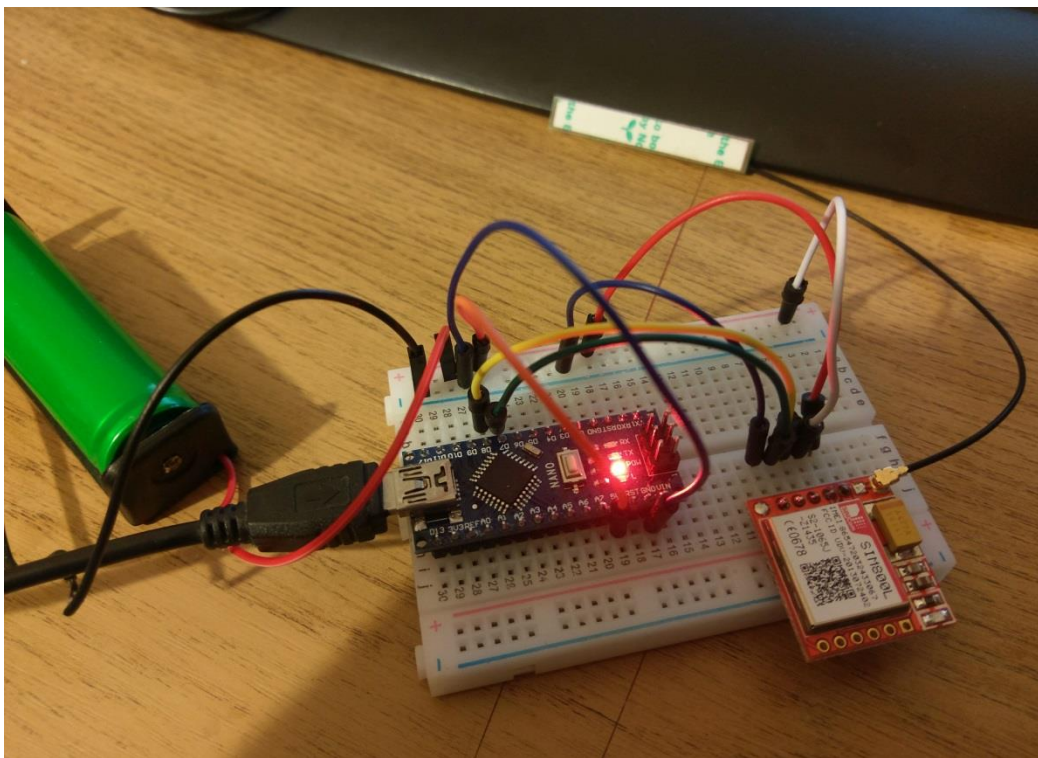


Obrázek C.2 – Detail elektroniky meteostanice





Obrázek C.3 – První verze návrhu meteostanice



Obrázek C.4 – První test zapojení na kontaktním poli

## **Příloha D – Přiložené CD**

Přiložené CD obsahuje:

- Plný text bakalářské práce ve formátu PDF,
- zdrojový kód webové aplikace,
- zdrojový kód programu pro meteostanici.