



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

IoT - sběr a zpracování dat snímačů s modulem ESP8266

Bakalářská práce

Studijní program: B2646 – Informační technologie
Studijní obor: 1802R007 – Informační technologie

Autor práce: **Marek Fišera**
Vedoucí práce: Ing. Miloš Hernych





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

IoT - the data collection with the ESP8266 module

Bachelor thesis

Study programme: B2646 – Information Technology
Study branch: 1802R007 – Information Technology

Author: **Marek Fišera**
Supervisor: Ing. Miloš Hernych



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Marek Fišera**
Osobní číslo: **M14000021**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **IoT - sběr a zpracování dat snímačů s modulem ESP8266**
Zadávací katedra: **Ústav mechatroniky a technické informatiky**

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

1. Seznamte se s aplikací obvodu ESP8266 jako modulu - snímače vybraných fyzikálních veličin. Dále se seznamte s jeho dostupnými verzemi firmware a programováním.
2. Navrhněte vhodný způsob sběru, archivace a přenosu modulem naměřených dat k dalšímu zpracování na vybrané serverové aplikaci.
3. Vytvořte vzorovou aplikaci, která bude navržený způsob sběru, archivace a přenosu demonstrovat.

Rozsah grafických prací: **dle potřeby dokumentace**

Rozsah pracovní zprávy: **30–40 stran**

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

Seznam odborné literatury:

- [1] **ESP8266 Technical Reference [online]. , 127 [cit. 2017-10-18]. Dostupné z: https://espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf**
- [2] **ESP8266EX Datasheet [online]. , 27 [cit. 2017-10-18]. Dostupné z: https://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf**
- [3] **IERUSALIMSCHY, Roberto. Programming in Lua [online]. , 348 [cit. 2017-10-18]. ISBN 859037981. Dostupné z: <http://www.lua.org/pil/contents.html>**
- [4] **NodeMCU Documentation [online]. [cit. 2017-10-18]. Dostupné z: <http://nodemcu.readthedocs.io/en/master/>**
- [5] **Getting started with MicroPython on the ESP8266 [online]. [cit. 2017-10-18]. Dostupné z: <https://docs.micropython.org/en/latest/esp8266/esp8266/tutorial/intro.html>**

Vedoucí bakalářské práce:

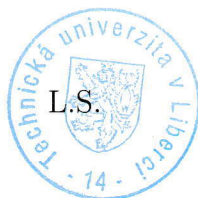
Ing. Miloš Hernych

Ústav mechatroniky a technické informatiky

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

Termín odevzdání bakalářské práce: **14. května 2018**

prof. Ing. Zdeněk Plíva, Ph.D.
děkan



Kolář
doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2017

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 14.5.2018

Podpis: Jirka

Poděkování

Tímto bych rád poděkoval panu Ing. Miloši Hernychovi za poskytnuté konzultace a rady při zpracování této práce.

Abstrakt

Cílem této bakalářské práce je seznámit se s aplikací obvodu ESP8266 jako snímače vybraných fyzikálních veličin. Dále se seznámit s jeho dostupnými verzemi firmware a programováním. Poté navrhnout a implementovat vhodný způsob sběru vybraných fyzikálních veličin za pomoci modulu, data následně archivovat, přenést je a zpracovat na vybrané serverové aplikaci. Úvodní část této práce popisuje základní informace o použitém modulu a technologii použité k samostatnému vývoji aplikace. Aplikace je naprogramována v jazyce MicroPython.

Klíčová slova

Mikro počítač ESP8266, HTTP protokol, MQTT protokol, programování v jazyce MicroPython, IoT platformy

Abstract

The aim of this bachelor thesis is to familiarize with the application of the ESP8266 circuit as a sensor of physical quantities. Then get familiar with its available firmware versions and programming. After that design and implement a suitable method of collecting selected physical quantities using the module, then archiving, transferring and processing the data to a selected server application. The first part of this thesis describes the basic information about the module and technologies used for the development of the application. The application is programmed in MicroPython language.

Key words

Microcomputer ESP8266, HTTP protocol, MQTT protocol, MicroPython programming, IoT platforms

Obsah

Úvod.....	11
1 Teoretická část.....	12
1.1 Internet of Things	12
1.2 ESP8266.....	12
1.2.1 ESP-12F	12
1.3 Firmware	12
1.4 MicroPython.....	13
1.5 Lua.....	13
1.6 MQTT protokol.....	14
1.7 HTTP protokol	14
1.8 Přídavné moduly	15
1.8.1 DS18B20.....	15
1.8.2 DHT11	16
1.9 IoT platformy	16
1.9.1 ThingSpeak	16
1.9.2 Grove Streams.....	17
1.9.3 Cayenne myDevices.....	17
1.9.4 Ubidots.....	17
1.9.5 Adafruit	17
2 Návrh řešení	18
2.1 Výběr firmware	18
2.2 Výběr vývojového prostředí.....	18
2.3 Sběr fyzikálních veličin.....	18
2.4 Archivace a přenos dat	19
2.5 Návrh aplikace.....	19
3 Praktická část.....	20
3.1 Nahrání firmware do modulu	20
3.2 Zapojení snímačů	20
3.2.1 Zapojení DS18B20.....	21
3.2.2 Zapojení DHT11	21
3.3 Aplikace pro sběr a přenos dat	21
3.4 Zpracování dat na serveru	22
3.4.1 ThingSpeak	23
3.4.2 Grove Streams.....	24
3.4.3 Cayenne myDevices.....	25
3.4.4 Ubidots.....	25
3.4.5 Adafruit	26
3.5 Porovnání použitých serverů	26
Závěr	28
Seznam použité literatury	29
Přílohy.....	31
Obsah přiloženého CD.....	31

Seznam obrázků

Obrázek 1: modul ESP-12F	13
Obrázek 2: MQTT publisher – subscriber [16].....	14
Obrázek 3: HTTP dotaz-odpověď.....	15
Obrázek 4: Senzor DS18B20	15
Obrázek 5: Senzor DHT11.....	16
Obrázek 6: Nastavení programu NodeMCU PyFlasher pro nahrání MicroPython firmware .	20
Obrázek 7: Schéma pinů senzoru DS18B20 [7]	21
Obrázek 8: Schéma pinů senzoru DS18B20 [8]	21
Obrázek 9: Výchozí vykreslování hodnot.....	23
Obrázek 10: ThingSpeak Změna teploty 3 dnů	24
Obrázek 11: Grove Streams graf maximálních a průměrných hodnot ze všech dat.....	24
Obrázek 12: Cayenne indikátor RH.....	25
Obrázek 13: Ubidots graf změny teploty za týden.....	26
Obrázek 14: Adafruit graf změny teploty za 24 h.....	26

Seznam tabulek

Tabulka 1: Porovnání IoT platforem.....	27
---	----

Seznam zkratek

API	Application Programming Interface, rozhraní pro programování aplikací
GND	Ground, uzemnění
GPIO	General-purpose input/output, obecný pin používaný u mikropočítačů
HTML	HyperText Markup Language, značkovací jazyk
HTTP	Hypertext Transfer Protocol, internetový protokol
I ² C	Inter-Integrated Circuit, počítačová sériová sběrnice
IDE	Integrated Development Environment, vývojové prostředí
IoT	Internet of Things, Internet věcí
IP	Internet Protocol, protokol používaný v počítačových sítích
JSON	JavaScript Object Notation, datový formát
MCU	Microcontroller, jednočipový počítač
MQTT	Message Queuing Telemetry Transport, komunikační protokol
QoS	Quality of Service, rezervace a řízení datových toků v počítačových sítích
REST	Representational state transfer, architektura rozhraní
TCP	Transmission Control Protocol, spojově orientovaný protokol
TLS	Transport Layer Security, protokol zajišťující zabezpečenou komunikaci
VDD	Voltage Drain Drain, napájení napětí

Úvod

Internet věcí je v dnešní době velmi často diskutovaným tématem v oblasti informatiky. Zařízení spadající do této oblasti jako jsou mikrokontroléry si dnes může velice snadno a levně sehnat téměř kdokoliv. Tato zařízení jsou energeticky velmi úsporná a lze je využít k vývoji různých aplikací, díky podobným vlastnostem, kterými oplývají osobní počítače. Tyto mikrokontroléry lze dále rozšiřovat o další moduly, které umožňují sběr různých fyzikálních veličin, jako je např. teplota nebo relativní vlhkost vzduchu.

Teoretická část práce pojednává o vlastnostech modulu ESP8266 a jeho dostupných verzích firmware. Dále jsou v této části popsány nejpoužívanější programovací jazyky, které lze využít pro vývoj aplikací pro tento modul. K přenosu dat z modulu na server a zpět je nutné použít komunikačních protokolů, a tak jsou zde popsány protokoly, které lze využít pro vývoj aplikace. Pro sběr fyzikálních veličin jsou dále popsány přídavné senzory a v neposlední řadě je zde popsáno několik nejpoužívanějších IoT platforem, které umožňují ukládání a další zpracování odeslaných dat na serveru.

V části návrhu řešení je popsán výběr vhodných technologií a prostředků nutných k samostatnému vývoji aplikace.

V praktické části práce je popsáno zapojení senzorů pro sběr fyzikálních veličin. Dále je zde popsána aplikace pro sběr a přenos dat. Data byla odesílána na několik serverů k ukládání a dalšímu zpracování. Posledním bodem praktické části je popis využití několika serverů a porovnání jejich parametrů.

1 Teoretická část

1.1 Internet of Things

Internet věcí je v dnešní době často probíraným tématem v oblasti informatiky. Pro tento termín neexistuje žádná ustálená definice, ale dá se pochopit jako síť fyzických zařízení, které jsou opatřeny senzory a softwarem, který umožňuje jejich vzájemnou komunikaci. Síť nemusí znamenat pouze Internet, ale může jít také o lokální síť, v rámci, které si zařízení vyměňují data, ze které je zajištěna následná možnost sdílení výsledků do Internetu [1].

1.2 ESP8266

ESP8266 je jedním z nejlevnějších Wi-Fi modulů na trhu vyráběných v Číně. Modul může samostatně fungovat jako mikropočítač, nebo ho lze použít jako Wi-Fi modul např. k jednodeskovým počítačům Arduino. Modul lze programovat v mnoha programovacích jazycích, mezi nejpoužívanější patří jazyk Lua, MicroPython a Arduino. ESP8266 umí pracovat se soubory, podporuje protokol HTTP a umožňuje komunikaci s přídatnými zařízeními prostřednictvím 1-Wire nebo I²C sběrnice. Existuje velké množství variant tohoto modulu a jednou z nich je ESP-12F, který byl použit pro účely bakalářské práce.

1.2.1 ESP-12F

ESP-12F je variantou modulu ESP8266 vyráběný čínskou společností Ai-Thinker. Obsahuje 32bitovou MCU jednotku. Podporované taktovací frekvence modulu jsou 80 MHz a 160 MHz. Modul používá standard IEEE802.11 b/g/n s TCP/IP protokolem. Wi-Fi pracuje při frekvenci 2,4 GHz v režimu vysílače i přijímače se zabezpečením WPA/WPA2. Dále modul disponuje flash pamětí s velikostí 4 MiB. Modul obsahuje 16 GPIO pinů, které jsou používány pro ovládání modulu, nebo pro připojení vstupních a výstupních zařízení. Modul má zabudovaný 10bitový A/D převodník. Operační napětí je v rozmezí od 3 V do 3,6 V a provozní teplota od -40 °C do 125 °C [2].

1.3 Firmware

Firmware je software, který je určen pro chod určitého zařízení. Mezi nejpoužívanější verze firmware používaných u mikrokontrolérů patří NodeMCU, což je firmware používaný pro spuštění skriptů v jazyce Lua, další verzí je MicroPython, který jak již napovídá název dokáže spustit skripty jazyku MicroPython. V neposlední řadě je tu také upravený Arduino firmware, který má optimalizovanou funkcionalitu pro modul ESP8266. Pro nahrání firmware do modulu poslouží nejlépe programy, jako je např. NodeMCU flasher nebo NodeMCU PyFlasher, které

i mimo jiné dokáží vymazat flash paměť, čímž odstraní případné přebytečné soubory, které mohly v paměti zůstat po předchozí verzi firmware.



Obrázek 1: modul ESP-12F

1.4 MicroPython

MicroPython je programovací jazyk vycházející z jazyku Python 3, který obsahuje menší množství standardních Python knihoven. Jazyk je optimalizovaný pro hladký chod na mikrokontrolérech a jejich vývojových prostředích. MicroPython dále obsahuje knihovny specifické pro mikrokontroléry, jako je např. „machine“ obsahující funkce upravené pro daný hardware na konkrétní vývojové desce. Jazyk se v současné době stále rozšiřuje o další knihovny, jako jsou knihovny pro konkrétní moduly, o které lze mikrokontroléry rozšířit. MicroPython je dostupný pod open source licencí a jeho zdrojový kód je dostupný na serveru GitHub. Lze ho také upravit podle potřeb pro vlastní účely [3].

1.5 Lua

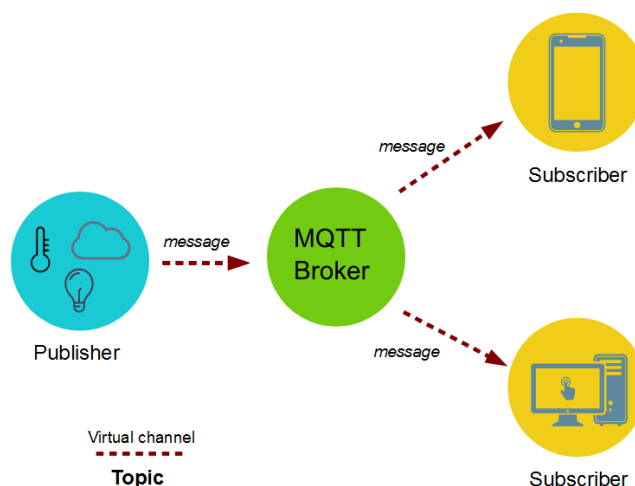
Lua je odlehčený skriptovací jazyk podporující procedurální a objektově orientované programování. Jeho jméno pochází z portugalštiny, ve které znamená „Měsíc“. Vznikl roku 1993 a je dostupný v open source licenci. Lua má implementovanou automatickou správu paměti za použití garbage collector [4].

1.6 MQTT protokol

MQTT (Message Queuing Telemetry Transport) je nenáročný komunikační protokol sloužící k přenosu zpráv mezi klienty za použití centrálního bodu – broker. Byl navržen především pro zařízení s nižším výkonem a pro sítě s nízkou šířkou pásma. Je navržen tak, aby minimalizoval využití sítě a minimalizoval využití hardware. Jelikož je protokol nenáročný a jednoduchý a velmi snadno se implementuje do zařízení jako jsou mikrokontroléry, velmi rychle se tak dostal do obliby. Navržen byl ve společnosti IBM roku 1999.

Přenos dat probíhá pomocí protokolu TCP a funguje na bázi publisher – subscriber. Pro komunikaci existuje centrální bod (broker), ten se stará o výměnu zpráv. Zprávy jsou roztrženy do tzv. témat (topic), klient následně do daného tématu publikuje (publish) nebo od něj odebírá (subscribe). Klient může být přihlášen k odběru více témat a zároveň také do více témat publikovat.

Obsah zpráv není nijak předem určený, nejčastěji se však používá JSON, BSON a data ve formě textu. Maximální velikost obsahu zprávy je omezena na 256 MB. MQTT využívá třech úrovní QoS (Quality of Service). Při použití nejnižší úrovně se zpráva odesílá bez potvrzení a ani doručení zprávy není zaručeno. Druhá úroveň zaručuje doručení zprávy alespoň jednou a nejvyšší úroveň zaručí dodání zprávy právě jednou. Pro nešifrované připojení používá protokol nejčastěji port 1883 a pro TLS spojení port 8883 [5].



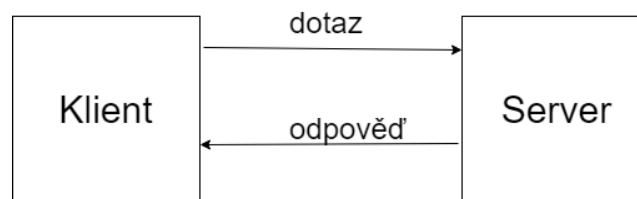
Obrázek 2: MQTT publisher – subscriber [16]

1.7 HTTP protokol

HTTP (Hypertext Transfer Protocol) je protokol určený pro přenos dat v internetu. První verze (HTTP/0.9) vznikla roku 1991 a dnes (2018) používaná verze (HTTP/1.1) roku 1997. HTTP protokol je bezstavový, což znamená, že každý příkaz je vykonaný nezávisle na předchozích

příkazech. Protokol používá pro komunikaci zpravidla port TCP 80, ale může využít i jiné porty. Protokol nepodporuje šifrování dat a pro zabezpečení se nejčastěji používá TLS spojení nad TCP, tato varianta je označována jako HTTPS.

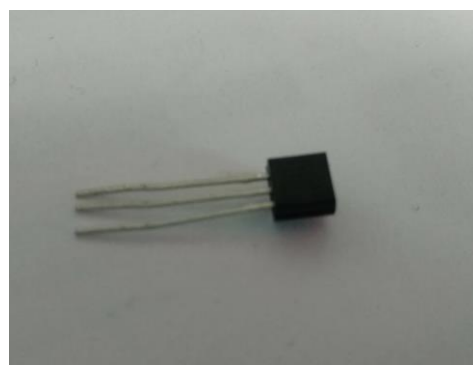
Funkce protokolu je založena na bázi dotaz-odpověď. Uživatel nejprve posílá serveru dotaz v podobě čistého textu, který obsahuje informace o verzi protokolu, označení požadovaného dokumentu a další informace např. o schopnostech prohlížeče. Server následně odešle odpověď ve formě textu, která obsahuje verzi protokolu a zprávu o úspěchu nebo chybový kód. Při úspěchu jsou ke zprávě připojena požadovaná data např. ve formě požadovaného dokumentu [6].



Obrázek 3: HTTP dotaz-odpověď

1.8 Přídavné moduly

Na trhu existuje mnoho externích modulů, které mohou být použity např. pro měření teploty, relativní vlhkosti vzduchu, barometrického tlaku a dalších fyzikálních veličin. Mezi nejpoužívanější u modulu ESP8266 patří teplotní senzor DS18B20 a senzor DHT kombinující měření relativní vlhkosti vzduchu a teploty ve verzích DHT11 a DHT22.



Obrázek 4: Senzor DS18B20

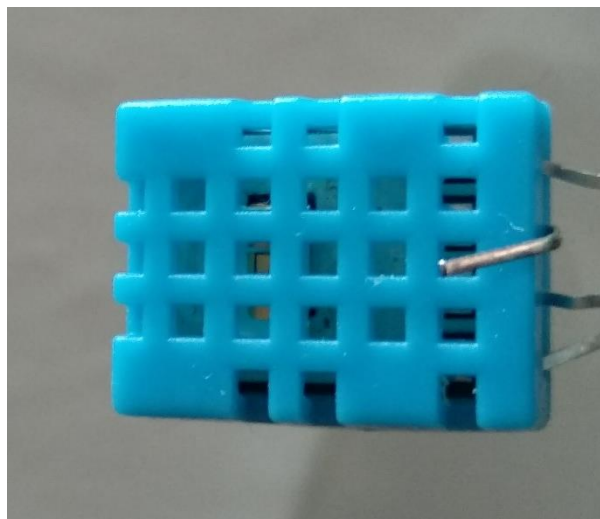
1.8.1 DS18B20

Modul DS18B20 je teplotní senzor od firmy Maxim. Se zařízením komunikuje pomocí 1-Wire sběrnice, což je sběrnice využívající pouze jednoho pinu pro komunikaci. Senzor je

schopen měřit teploty v rozmezí od $-55\text{ }^{\circ}\text{C}$ do $125\text{ }^{\circ}\text{C}$ s přesností $\pm 0,5\text{ }^{\circ}\text{C}$ v intervalu od $-10\text{ }^{\circ}\text{C}$ do $85\text{ }^{\circ}\text{C}$. Přesnost měřené teploty lze nastavit na 9 až 12 bitů. Senzor lze napájet přímo z komunikačního pinu díky módu „parasite power“, který eliminuje potřebu externího napájení. Každý modul má svůj 64bitový sériový kód, který umožňuje komunikaci více DS18B20 senzorů pomocí stejné 1-Wire sběrnice. Senzor je předem kalibrován a otestován výrobcem a výdrž kalibrace je garantována po jeho celou dobu životnosti [7].

1.8.2 DHT11

DHT11 je senzor pro měření teploty a relativní vlhkosti vzduchu. Rozsah měření teploty je od $0\text{ }^{\circ}\text{C}$ do $50\text{ }^{\circ}\text{C}$ s přesností $\pm 2\text{ }^{\circ}\text{C}$ a rozsah měření vlhkosti od 20 % do 90 % s přesností $\pm 5\text{ } \% \text{ RH}$ [8]. Senzor používá jeden pin k napájení, jeden k uzemnění a jeden pin pro přenos dat. Každý DHT11 modul je striktně testován v laboratořích a výrobce zaručuje jeho přesnou kalibraci. K napájení modulu je potřeba 3-5,5 V [9].



Obrázek 5: Senzor DHT11

1.9 IoT platformy

IoT platforma je platforma, která umožňuje komunikaci mezi IoT zařízeními. Mezi další důležité vlastnosti patří uchovávání dat, jejich analýza a vizualizace. V dnešní době (2018) existují stovky takových platforem [10]. V následujících odstavcích budou popsány nejpoužívanější v oblasti mikropočítačů, které nabízí bezplatné využívání některých jejich služeb.

1.9.1 ThingSpeak

ThingSpeak je IoT platforma, která je určena ke sběru dat senzorů a jejich archivaci v cloudovém úložišti. Následně lze data analyzovat a vizualizovat pomocí jazyka MATLAB.

Data mohou být odesílána ze zařízení jako je Arduino, Raspberry Pi a podobného hardware. ThingSpeak využívá technologie REST a MQTT API pro přenos dat na server [11].

1.9.2 Grove Streams

Grove Streams je IoT platforma pro analýzu dat, která vznikla roku 2011. Platforma byla vybudována pro vývojáře, kteří chtějí začít budovat menší projekty i pro společnosti, které potřebují využívat milióny datových proudů ze senzorů a dalších mobilních zařízení. Služba Grove Streams využívá REST API pro přenos dat [12].

1.9.3 Cayenne myDevices

Cayenne je IoT platforma vyvinutá společností myDevices. Jejich řešení si zakládá na jednoduchém drag and drop vývoji IoT projektů. Vývojáři mohou použít IoT řešení s minimální potřebou programování, i řešení bez nutnosti programování. Cayenne obsahuje katalog certifikovaných IoT zařízení, mezi která patří např. Arduino, ESP8266 a Raspberry Pi. Pro komunikaci se zařízeními využívá MQTT API. Server umožňuje další zpracování dat v podobě vizualizace do různých druhů indikátorů a grafů vytvořených uživatelem [13].

1.9.4 Ubidots

Ubidots je platforma, která využívá mnoha protokolů pro komunikaci, jako je např. HTTP, MQTT nebo libovolný industriální protokol. Pro připojení zařízení do cloudového úložiště existuje více než 200 otestovaných knihoven s tutoriály. Pro vizualizaci dat nabízí indikátorů, grafů, tabulek a map za použití HTML canvas technologie. Dále nabízí sdílení dat veřejnými odkazy přístupné z webových prohlížečů nebo aplikací pro mobilní telefony. Služba je dostupná jak v placené verzi pro firemní účely, tak i ve verzi zdarma pro edukativní využití [14].

1.9.5 Adafruit

Adafruit je IoT platformou, která umožňuje využívat klientských knihoven pro práci se zařízeními jako je ESP8266, Raspberry Pi, Arduino a dalšími. Ke komunikaci se zařízeními využívá HTTP a MQTT protokolu. Pro vizualizaci dat nabízí uživatelsky konfigurovatelné tabulky, grafy a indikátory. Ve verzi zdarma jsou data uchovávána na serveru po dobu jednoho měsíce [15].

2 Návrh řešení

V této části bakalářské práce bude popsán výběr vhodných technologií a prostředků pro vývoj aplikace.

2.1 Výběr firmware

Existuje mnoho verzí firmware, již zmíněných v rešeršní části bakalářské práce. Nejvhodnější variantou pro tento typ práce byl firmware MicroPython upravený pro modul ESP8266 ve verzi 1.9.3, který podporuje objektové programování a obsahuje mnoho knihoven, které usnadňují práci s externími moduly a zpřehledňují tak výsledný kód aplikace.

2.2 Výběr vývojového prostředí

Pro samostatné naprogramování aplikace bylo potřeba najít vhodné vývojové prostředí (IDE), které umožňuje komunikaci s modulem ESP8266. Dále bylo nutné, aby IDE umožňovalo základní operace se soubory uloženými v paměti modulu. Prvním takovým vývojovým prostředím bylo ESPlorer, které je vytvořeno pro psaní skriptů v jazyce Lua a MicroPython. Bohužel však působilo potíže při nahrávání skriptů do modulu a občas v prostředí selhávalo grafické rozhraní. Dalším otestovaným IDE bylo PyCharm od společnosti JetBrains, které je výborné pro programování v jazyce Python, jelikož obsahuje množství zabudovaných funkcí urychlujících programování aplikací. Pro spojení s modulem ESP8266 nabízelo MicroPython plugin verze 1.0, který však byl v první vydané verzi a stále ve fázi vývoje. Přes několik pokusů se nepodařilo navázat spojení s modulem. Dalším otestovaným IDE bylo uPyCraft, které nabízelo manipulaci se soubory uloženými v paměti modulu a kontrolu správné syntaxe jazyku MicroPython, což postačilo k naprogramování aplikace. Žádné komplikace, které nastaly s předešlými IDE už neproběhly.

2.3 Sběr fyzikálních veličin

První fyzikální veličinou, která šla měřit bez použití přídavných modulů bylo napětí. Napětí lze měřit přímo z modulu díky již zabudovanému A/D převodníku. Pro sběr dalších fyzikálních veličin bylo nutné připojení dalších senzorů. Prvním zvoleným byl teplotní senzor DS18B20, protože je snadno dostupný, a i přes jeho nízkou cenu velmi přesný. Dalším modulem byl DHT11, který je velmi levný a umožňuje kromě snímání teploty i měření relativní vlhkosti vzduchu. Senzor DS18B20 komunikuje s modulem ESP8266 prostřednictvím 1-Wire sběrnice.

2.4 Archivace a přenos dat

Protože modul ESP8266 obsahuje poměrně nízkou vnitřní paměť, tak bylo vhodné data ukládat na externí úložiště. Mezi nejvhodnější varianty patří cloudové platformy jako je např. ThingSpeak, které umožňují následně nasbíraná data zpracovat. Jelikož existuje mnoho takových platforem, bylo otestováno několik nejpoužívanějších.

2.5 Návrh aplikace

Pro návrh aplikace je nutné stanovit několik základních bodů. Nejprve je potřeba naměřit požadované fyzikální veličiny. Pro změření napětí modulu existuje MicroPython knihovna *machine*, která využívá zabudovaného A/D převodníku. Pro sběr teploty senzoru DS18B20 je knihovna *ds18x20*. Pro sběr dat ze senzoru DHT11 existuje knihovna *dht*, která umožňuje změření teploty a relativní vlhkosti vzduchu.

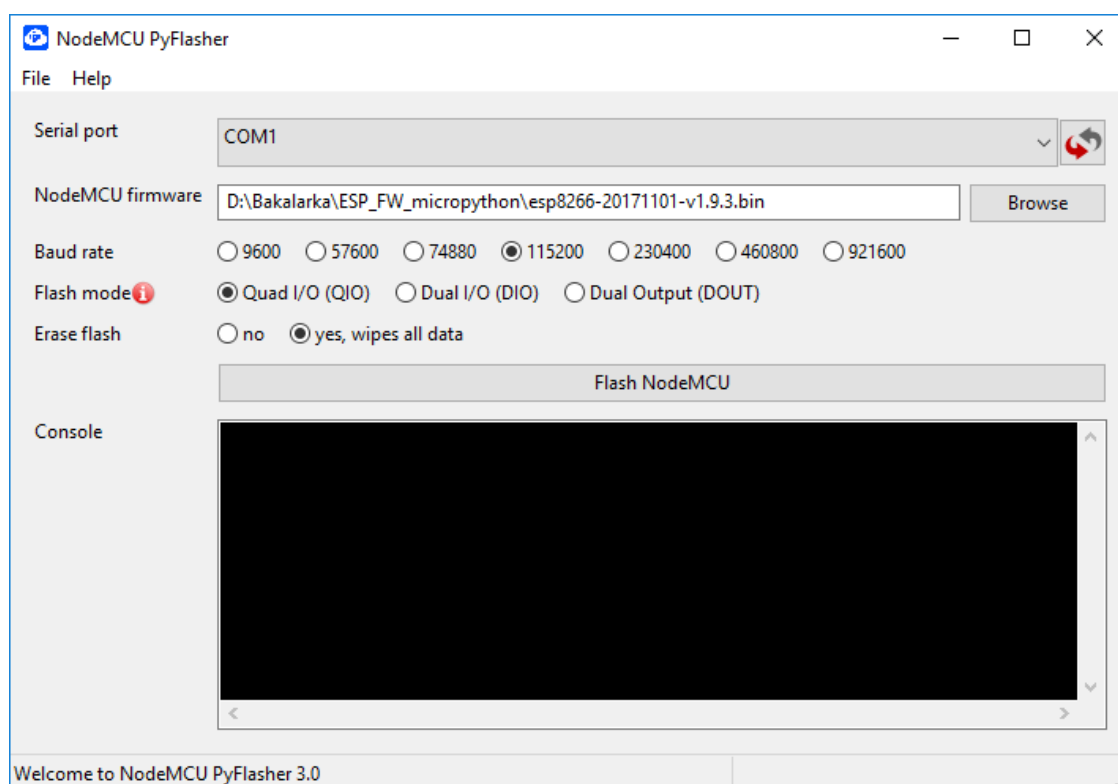
Dále je potřeba, aby byl modul ESP8266 připojen k internetu, kvůli odesílání dat do cloudového úložiště. V MicroPython slouží k tomuto účelu knihovna *network*, která umožňuje připojení do sítě prostřednictvím Wi-Fi.

Dalším nutným bodem je odesílání dat do cloudového úložiště. V této práci je použito více cloudových platforem, ale každá umožňuje přenos dat buď za použití HTTP protokolu nebo MQTT protokolu. Protože je vždy odesílání dat na server omezeno časovou periodou, tak je za potřeby využít funkce časovače z knihovny *time*.

3 Praktická část

3.1 Nahrání firmware do modulu

Pro nahrání firmware do modulu bylo potřeba zvolit vhodný program. Prvním takovým byl velmi často používaný program NodeMCU flasher, který však po nahrání MicroPython firmware do modulu způsoboval náhodné resetování modulu a pády aplikace. Dalším otestovaným programem byl NodeMCU PyFlasher, který nepůsobil žádné podobné potíže. Pro samostatné nahrání firmware bylo potřeba správně zvolit sériový port, který byl použit k přenosu dat, binární soubor se samostatným firmware, přenosovou rychlost, flash mód a zda chceme vymazat všechna data z flash paměti modulu. Nastavení zvolená pro toto konkrétní řešení jsou vyobrazena na obrázku č. 6.



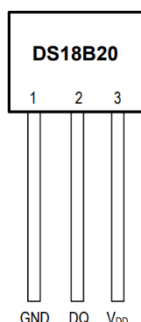
Obrázek 6: Nastavení programu NodeMCU PyFlasher pro nahrání MicroPython firmware

3.2 Zapojení snímačů

Snímače DS18B20 a DHT11 bylo potřeba správně zapojit k modulu ESP8266, aby bylo zajištěno jejich napájení a přenos nasbíraných dat. Jejich zapojení je popsáno v následujících odstavcích.

3.2.1 Zapojení DS18B20

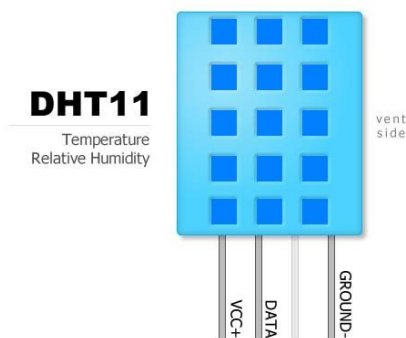
Prvním senzorem, který byl potřeba zapojit k modulu ESP8266 je teplotní čidlo DS18B20. Senzor byl zapojen k modulu podle obrázku č.7 z oficiální dokumentace senzoru. První pin GND (Ground) byl připojen k uzemnění modulu ESP8266, druhý pin DQ byl připojen k GPIO pinu modulu, který zajišťuje přenos dat a poslední VDD pin k napájení.



Obrázek 7: Schéma pinů senzoru DS18B20 [7]

3.2.2 Zapojení DHT11

Druhým senzorem, který bylo nutno správně připojit je DHT11. Senzor se skládá ze 4 pinů, ale jeden z nich se nezapojuje. První pin senzoru slouží k napájení, druhý pro přenos dat a poslední pro připojení k uzemnění. Senzor byl připojen k modulu ESP8266 analogicky jako DS18B20 s rozdílem pořadí pinů.



Obrázek 8: Schéma pinů senzoru DS18B20 [8]

3.3 Aplikace pro sběr a přenos dat

Aplikace pro sběr a přenos dat byla naprogramována v jazyce MicroPython za použití vývojového prostředí uPyCraft. V následujících odstavcích budou popsány všechny body aplikace.

- `measure_values.py`
Soubor obsahuje funkce pro měření požadovaných fyzikálních veličin. První funkce získává hodnotu teploty z teplotního čidla DS18B20. Teploměr je připojen k GPIO pinu na pozici 13. Pro konvertování hodnoty teploty po čtení musela být nastavena prodleva 750 ms, kvůli omezení časového intervalu čtení hodnoty. Funkce vrací hodnotu teploty datového typu float. Dále je zde funkce pro získávání dat ze senzoru DHT11, která vrací teplotu a relativní vlhkost vzduchu. Poslední funkce získává hodnotu napětí ze zabudovaného A/D převodníku v modulu ESP8266.
- `wifi.py`
Ve skriptu je obsaženo nastavení nutné pro připojení do Wi-Fi sítě. Obsahuje funkci, která slouží jak k připojení do sítě Wi-Fi, tak i kontrolu, zda je již spojení zprostředkováno.
- `main.py`
Tato část aplikace obsahuje implementované všechny funkce potřebné k měření požadovaných fyzikálních veličin a funkci k připojení do sítě prostřednictvím Wi-Fi. Dále obsahuje všechno potřebné nastavení a funkce k připojení k vybraným IoT platformám. Spojení a odesílání dat na servery ThingSpeak, Adafruit a Cayenne myDevices je zprostředkováno za využití MQTT protokolu. Pro komunikaci se servery Grove Streams a Ubidots je zde využito HTTP protokolu. Část aplikace pro sběr a přenos je spouštěna cyklicky. Každý cyklus proběhne nejdříve kontrola připojení k internetu, potom změření teploty z čidla DS18B20, změření teploty a relativní vlhkosti z čidla DHT11 a změření hodnoty napětí modulu. Poté jsou data odesílána na všechny zmíněné servery. Pro případ pádu aplikace je implementována výjimka, která restartuje modul ESP8266 a zavede ho základních nastavení pro znovuuvedení chodu aplikace. Měření a odesílání dat je prováděno každých 15 s.

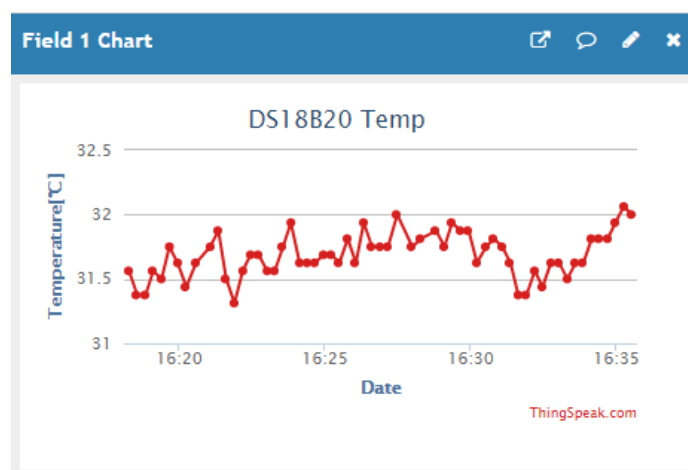
3.4 Zpracování dat na serveru

K zpracování dat byly vybrány IoT platformy s možností dalšího zpracování dat uložených na serveru. Pro přenos dat využívají MQTT protokolu nebo HTTP protokolu. Možnosti zpracování dat je specifické pro každou IoT platformu. V následujících odstavcích bude popsáno zpracování dat na několika nejpoužívanějších serverech.

3.4.1 ThingSpeak

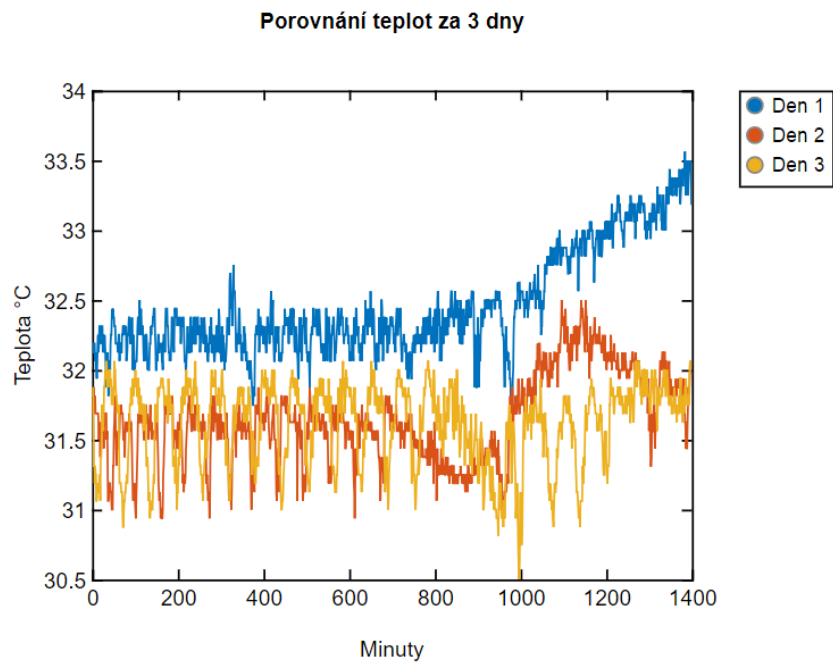
ThingSpeak je první otestovaným serverem, který umožňuje vykreslovat uložená data do grafů. Pro přenos dat na server lze využít jak HTTP protokolu, tak i MQTT protokolu. Pro komunikaci byl zvolen MQTT protokol z důvodu šetření hardwarových prostředků mikropočítače. Pro navázání spojení se serverem je důležité nastavit v aplikaci správně uživatelské jméno, adresu serveru a MQTT API komunikační klíč. Ke komunikaci se serverem využívá MQTT protokol TCP portu 1883.

Pro zpracování dat lze použít předpřipravených grafů, které vykreslují vždy jednu hodnotu uloženou v konkrétním kanálu. Ve výchozím nastavení vykresluje posledních 60 uložených hodnot do spojnicového grafu. Na ose X vypisuje čas, kdy přišla data na server a na ose Y odeslanou hodnotu. Výchozí vykreslování hodnot je zobrazeno na obrázku č. 9.



Obrázek 9: Výchozí vykreslování hodnot

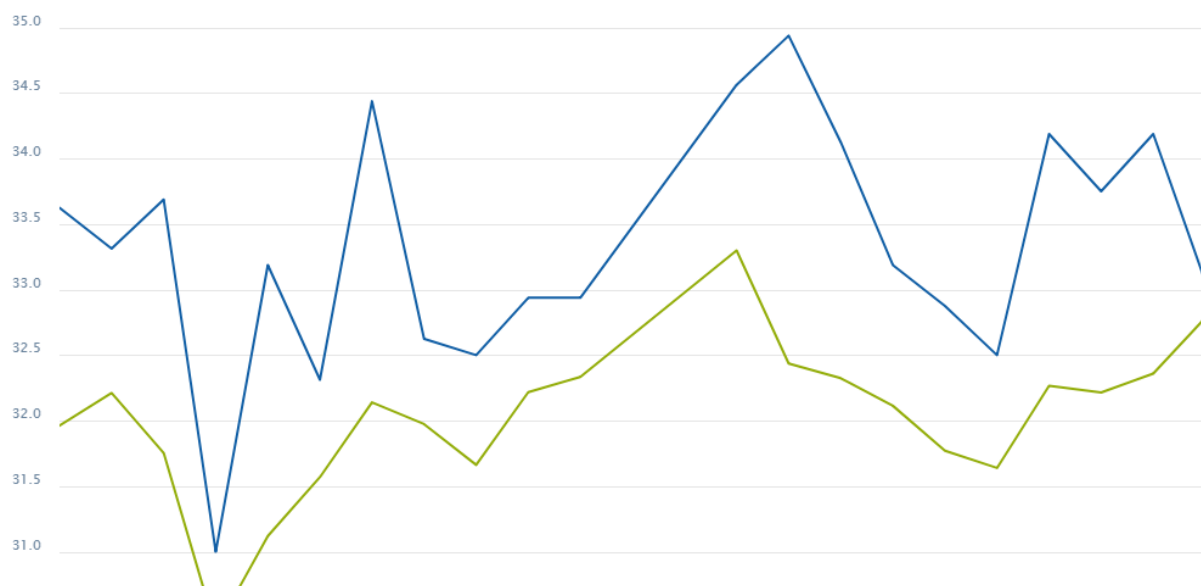
Pro pokročilejší zpracování dat lze buď použít připravených šablon, nebo lze data vizualizovat podle konkrétní potřeby v jazyce MATLAB. Na výběr je buď analýza dat vypsaná ve formě textu, nebo vykreslení požadovaných hodnot do grafů. Grafy lze nechat vykreslovat ve vybraném kanálu uživatele. Na obrázku č. 10 jsou vykresleny změny v teplotách za poslední 3 dny s použitím jazyku MATLAB.



Obrázek 10: ThingSpeak Změna teploty 3 dnů

3.4.2 Grove Streams

Ke spojení a odesílání dat na server Grove Streams bylo použito HTTP protokolu, jelikož MQTT nepodporuje. Pro vykreslení dat je možno použít grafů nebo tabulek, které zobrazují data ve vybraném intervalu a na data lze použít např. funkce pro nalezení maximální a minimální hodnoty. Na obrázku č. 11 jsou vykresleny maximální a průměrné hodnoty každého dne ze všech dat uložených na serveru.



Obrázek 11: Grove Streams graf maximálních a průměrných hodnot teploty ze všech dat

3.4.3 Cayenne myDevices

Server Cayenne myDevices využívá ke komunikaci se zařízeními pouze MQTT protokolu. Ke spojení se serverem byla použita obdobná nastavení jako u serveru ThingSpeak. Služba umožňuje vypisování uložených hodnot do připravených šablon pro určitá zařízení. Pro zobrazení aktuální hodnoty slouží jednoduché indikátory, které lze nastavit pro jakoukoliv veličinu. K dalšímu zpracování jsou k dispozici grafy, které lze použít k vykreslení hodnot ve vybraném intervalu. Dále lze využít SQL dotazů k vybrání určitých dat. Na obrázku č. 12 je zobrazen indikátor aktuální hodnoty relativní vlhkosti vzduchu.



Obrázek 12: Cayenne indikátor RH

3.4.4 Ubidots

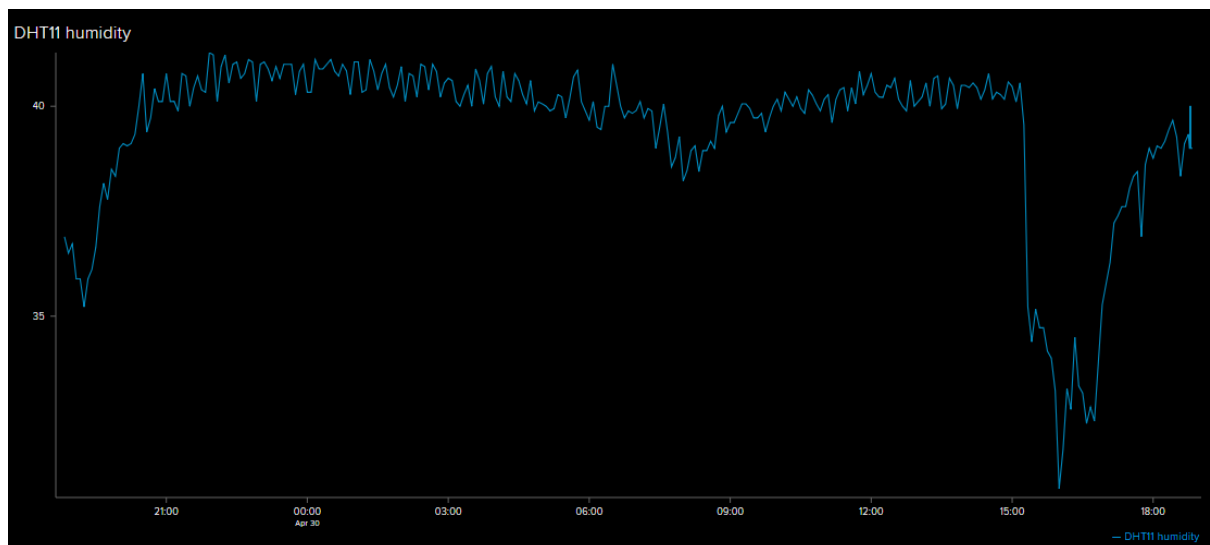
Se serverem Ubidots bylo navázáno spojení prostřednictvím HTTP protokolu. Pro kontrolu, zda data úspěšně přicházejí na server bylo použito triggeru, který odesílá upozornění na e-mail, když hodnoty na server nepřicházejí po dobu 5 minut pro případ, pokud došlo k výpadku elektřiny, internetového spojení, či pádu aplikace. K zobrazení samostatných dat bylo využito grafů, které vykreslují vybrané veličiny na časové ose. Na obrázku č. 13 je zobrazena změna teploty za jeden týden.



Obrázek 13: Ubidots graf změny teploty za týden

3.4.5 Adafruit

Spojení se serverem Adafruit bylo navázáno prostřednictvím MQTT protokolu. Pro vykreslení relativní vlhkosti vzduchu za posledních 24 h bylo využito spojnicového grafu. Změna hodnoty v čase je zobrazena na obrázku č. 14.



Obrázek 14: Adafruit graf změny teploty za 24 h

3.5 Porovnání použitých serverů

V tabulce č. 1 jsou porovnány otestované IoT platformy. Mezi porovnávané parametry patří možnost způsobu komunikace se serverem, limity odesílaných dat na server, doba uchování dat a druhy dostupných triggerů. Všechny parametry jsou porovnávány v neplacených verzích

uživatelských účtů. Některé parametry nebyly na daných stránkách uvedeny, a proto jsou označeny otazníkem.

Server	ThingSpeak	Grove Streams	Cayenne myDevices	Ubidots	Adafruit
Komunikace (protokol)	HTTP a MQTT	HTTP	MQTT	HTTP a MQTT	HTTP a MQTT
Limit dat (uložených hodnot)	3 000 000 za rok	10 000 (string) 550 000 (ostatní datové typy)	?	Limitováno kredity	30 za minutu
Doba uchování dat	3 roky	?	?	3 měsíce	30 dní
Trigger	MATLAB analýza, Tweet, HTTP request	Email, HTTP request, Upozornění, nastavení proměnné	Email, Sms	SMS, Email, Telegram	Email, HTTP request, nastavení proměnné

Tabulka 1: Porovnání IoT platforem

Závěr

Všechny cíle bakalářské práce byly splněny. Výsledkem této práce je aplikace, která zajišťuje sběr dat ze snímačů připojených k modulu ESP8266. Následně jsou naměřená data odesílána na IoT platformy ThingSpeak, Grove Streams, Cayenne myDevices, Ubidots a Adafruit. Aplikace zajišťuje spojení se servery s pomocí HTTP nebo MQTT protokolu.

V praktické části jsou mimo vývoj aplikace popsány možnosti dalšího zpracování dat na uvedených serverech. Dále jsou v této části porovnány způsoby přenosů dat na servery, limity uložených dat a další funkce uvedených IoT platform. Z otestovaných platform se jeví server Thingspeak jako nejlepší řešení pro tuto práci, protože má nejmenší limitace na počet uložených hodnot, a navíc lze data dále zpracovat v jazyce MATLAB. Zpracování v jazyce MATLAB nabízí velmi široké možnosti analýzy. Aplikace s modulem by mohla posloužit jako obdoba meteorologické stanice.

Aplikace byla pro otestování dlouhodobějšího chodu ve finální verzi spuštěna po dobu jednoho týdne. Kromě časového intervalu z doby drobných výpadků internetového připojení ze strany poskytovatele byla všechna data úspěšně odeslána na uvedené servery.

Do budoucna by práce mohla být rozšířena o vlastní webovou aplikaci, která by zajišťovala podobné funkce ukládání a zpracování dat, jako nabízí tyto IoT platformy. Modul by mohl být rozšířen o baterii, která by umožňovala chod ve venkovním prostředí.

Vývoj aplikace pro modul ESP8266 byl cennou zkušeností, jelikož využití podobných modulů se stále rozrůstá a v budoucnu budou jistě mít velmi široké uplatnění.

Seznam použité literatury

- [1] POHANKA, Pavel. *Internet věci* [online]. [cit.2018-05-14] vyd. Dostupné z: <http://i2ot.eu/internet-of-things/>
- [2] *ESP-12F WiFi Module* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://www.elecrow.com/download/ESP-12F.pdf>
- [3] *MicroPython - Python for microcontrollers* [online]. [cit.2018-05-14]. Dostupné z: <https://micropython.org/>
- [4] *Lua* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://www.lua.org/>
- [5] MALÝ, Martin. *Protokol MQTT: komunikační standard pro IoT* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://www.root.cz/clanky/protokol-mqtt-komunikacni-standard-pro-iot/>
- [6] *HTTP Tutorial* [online]. [cit.2018-05-14]. Dostupné z: <https://www.tutorialspoint.com/http/index.htm>
- [7] *DS18B20* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>
- [8] *DHT11 teploměr a vlhkoměr digitální* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://arduino-shop.cz/docs/produkty/0/133/1500635986.pdf>
- [9] *Temperature and humidity module DHT11 Product Manual* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://akizukidenshi.com/download/ds/aosong/DHT11.pdf>
- [10] SCULLY, Pdraig. *5 Things To Know About The IoT Platform Ecosystem* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://iot-analytics.com/5-things-know-about-iot-platform/>
- [11] *ThingSpeak* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://www.mathworks.com/help/thingspeak/>
- [12] *Grove Streams, LLC* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://cz.linkedin.com/company/grove-streams-llc>
- [13] *About - myDevices.com. Creators of Cayenne IoT Project Builder - myDevices.com* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://mydevices.com/about/>

- [14] *IoT platform features. IoT platform / Internet of Things* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://ubidots.com/platform/>
- [15] *IO - Adafruit. IO* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://io.adafruit.com/>
- [16] VOJÁČEK, Antonín. *IoT MQTT prakticky v automatizaci - 1.díl - úvod* [online]. online. vyd. [cit.2018-05-14]. Dostupné z: <https://automatizace.hw.cz/iot-mqtt-prakticky-v-automatizaci-1dil-uvod.html>

Přílohy

Obsah přiloženého CD

Na přiloženém CD jsou následující složky a soubory:

- Aplikace – složka se zdrojovými kódy aplikace
 - main.py
 - measure_values.py
 - wify.py
- Dokumentace
 - BP_dokumentace_Fisera_2018.pdf