

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Návrh inteligentnej senzorickej siete administratívnej budovy

Diplomová práca

Vedúci práce:
Ing. Vít Ondroušek, Ph.D.

Bc. Adam Földvári

Brno 2018

Podakovanie patrí pánovi Ing. Vítovi Ondrouškovi, Ph.D., za jeho rady, odborný prístup a za sprostredkovanie finančných prostriedkov, bez ktorých by táto diplomová práca nebola realizovateľná. Ďalej by som chcel poďakovať celému tímu projektu SmartPEF a každému kto sa na tomto projekte nejakým spôsobom podieľal.

Čestné prehlásenie

Prehlasujem, že som túto prácu: **Návrh inteligentnej senzorickej siete administratívnej budovy**

vypracoval samostatne a všetky použité pramene a informácie sú uvedené v zozname použitej literatúry. Súhlasím, aby moja práca bola zverejnená v súlade s § 47b zákona č. 111/1998 Sb., o vysokých školách v znení neskorších predpisov, a v súlade s platnou *Smernicou o zverejňovaní vysokoškolských záverečných prác*.

Som si vedomý, že sa na moju prácu vzťahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brne má právo na uzavretie licenčnej zmluvy a použitie tejto práce ako školského diela podľa § 60 odst. 1 Autorského zákona.

Ďalej sa zaväzujem, že pred spísaním licenčnej zmluvy o využitie diela inou osobou (subjektom) si vyžiadam písomné stanovisko univerzity o tom, že predmetná licenčná zmluva nie je v rozpore s oprávnenými záujmami univerzity a zaväzujem sa uhradiť prípadný príspevok na úhradu nákladov spojených so vznikom diela, a to až do ich skutočnej výšky.

Brno 15. mája 2018

.....

Abstract

Földvári, A. Design of intelligent sensory network in administrative building. Diploma thesis. Brno, 2018.

The diploma thesis focuses on the design of intelligent sensory network in the environment of administrative building. First part of thesis introduces reader to the existing and used technology in thesis, the following section describes the design, construction and implementation of modules and control application. The result of this thesis is a working sensory network which can be managed by control application, which collects data from end-point modules. The complete solution will be tested in the SmartPEF project.

Keywords

sensory network, automation, administrative building management and control, ESP 8266, NodeMCU, WPF, C#, Lua

Abstrakt

Földvári, A. Návrh inteligentnej senzorickej siete administratívnej budovy. Diplomová práca. Brno, 2018.

Diplomová práca je zameraná na návrh inteligentnej senzorickej siete v prostredí administratívnej budovy. Prvá časť práce oboznamuje čitateľa o existujúcich a v práci použitých technológiách, nasledujúca časť popisuje návrh, konštrukciu a implementáciu modulov a ovládacej aplikácie. Výsledkom práce je fungujúca senzorická sieť ktorú je možné spravovať prostredníctvom centrálnej aplikácie, ktorá zbiera dáta z koncových modulov. Celkové riešenie bude otestované v rámci projektu SmartPEF

Kľúčové slová

senzorická sieť, automatizácia, správa administratívnej budovy, ESP-8266, NodeMCU, WPF, C#, Lua

Obsah

1	Úvod a cieľ práce	7
1.1	Úvod	7
1.2	Cieľ práce	7
2	Analýza problému	9
2.1	Rešeršná štúdia	9
2.1.1	iNELS	9
2.1.2	Crestron	10
2.1.3	KNX	12
2.1.4	CUE	14
2.1.5	ABB	15
2.1.6	BigClown	16
2.2	Zhodnotenie výsledkov rešeršnej štúdie	18
2.3	Požiadavky na senzorickú sieť	19
3	Návrh senzorickej siete	21
3.1	Senzorický uzol	21
3.2	Adaptér	22
3.3	Svetelné senzory	22
3.3.1	TSL-2561	22
3.3.2	TCS-34725	23
3.4	Senzory teploty a vlhkosti	24
3.4.1	DHT-11	24
3.4.2	Si-7021	25
3.5	Mikrokontrolér	25
3.5.1	ESP-8266 12s	27
3.5.2	NodeMCU	29
3.6	MQTT	31
3.6.1	MQTT Broker	32
3.7	Použité nástroje	33
3.7.1	ESP Cut	33
3.7.2	ESP-Py Flasher	33
3.7.3	DipTrace	33
3.7.4	MQTT Spy	33
3.7.5	Visual Studio 2017	34
4	Návrh riešenia	35
4.1	Komunikácia	35
4.1.1	Komunikačný protokol	36
4.1.2	Štruktúra komunikácie	36
4.2	Senzorický uzol	39
4.2.1	Program senzorického uzlu	39

4.2.2	Senzorická výbava	44
4.2.3	Prototypovanie	45
4.2.4	Návrh dosiek plošných spojov	45
4.2.5	Výsledný senzorický uzol	47
4.3	Adaptér	48
4.3.1	Architektúra aplikácie	49
4.3.2	Zber dát zo senzorickej siete	50
4.3.3	Vzdialená aktualizácia programu	51
4.3.4	Správa a monitorovanie pripojených senzorických uzlov	52
4.3.5	Notifikácie	52
4.3.6	Nastavenia aplikácie	52
4.3.7	Lokálne úložisko dát	53
4.3.8	Výsledná aplikácia	53
4.4	Bezpečnosť	54
4.4.1	Program senzorických uzlov	54
4.4.2	Pripojenie uzlu do senzorickej siete	55
4.4.3	Výmena šifrovaných dát v senzorickej sieti	55
4.4.4	Zasielanie dát na REST Api	55
4.4.5	Sieťová infraštruktúra	56
4.5	Výsledná senzorická sieť	56
4.5.1	Testovanie riešenia počas vývoja	56
4.5.2	Nasadenie senzorickej siete	57
5	Nasadenie do skúšobnej prevádzky	58
5.1	Dlhodobé testovanie	58
5.2	Ukážky nameraných dát	60
6	Zhrnutie	62
7	Diskusia	63
8	Navrhované vylepšenia	65
9	Záver	66
10	Literatúra	67
	Prílohy	70
A	Obrázky	71
A.1	Senzorický uzol	71
A.2	Adaptér	71
B	Priložené CD	76

1 Úvod a cieľ práce

1.1 Úvod

Táto práca je zameraná na návrh inteligentnej senzorickej siete pre administratívnu budovu. Pod pojmom senzorická sieť, je myslená sieť senzorických uzlov, ktoré zbierajú dáta z jednotlivých častí administratívnej budovy. Takto zozbierané dáta môžu následne slúžiť napríklad k zníženiu nákladov na energie, zvýšeniu bezpečnosti, zlepšeniu životných podmienok (monitorovanie úrovne oxidu uhličitého v ovzduší, monitorovanie teploty a vlhkosti, ...), a mnohým ďalším.

V dnešnej dobe zažíva oblasť internetu vecí, veľký rozmach v takmer všetkých odvetviach, či už sa jedná o domácnosť, priemysel alebo administratívnu sféru. Myšlienka prepojenia jednotlivých systémov napríklad v administratívnej budove za rôznymi účelmi sa v posledných rokoch stáva realitou, aj vďaka rozvoju technológií v danej oblasti.

V súčasnosti je na trhu množstvo spoločností, ktoré ponúkajú správu administratívnych budov využívajúc rozličné technológie, čo naznačuje celkový rozmach tohto odvetvia, ktoré spadá do internetu vecí. Trh ponúka riešenia, ktoré sú kompatibilné s otvorenými systémami, ale taktiež aj riešenia, ktoré spoločnosti vyvinuli sami a ponúkajú tak svoju vlastnú uzavretú platformu. Oba prístupy so sebou nesú určité výhody a nevýhody. Pri otvorených systémoch je možné hovoriť napríklad o širšej ponuke zariadení rôznych výrobcov, ale zas o niečo nižšej bezpečnosti. Kdežto pri systémoch uzatvorených nemusí byť vždy ponuka zariadení dostatočne široká, ale platforma môže byť naopak navrhnutá bezpečnejšie.

Práve bezpečnosť je jedna z najviac spájaných a diskutovaných tém s problematikou internetu vecí. Dáta ktoré sú šírené prostredníctvom senzorickej siete musia spĺňať vysoký faktor dôvery, nakoľko sa na ne spoliehajú ako užívatelia tejto siete, tak aj stroje, ktoré na základe prijatých dát môžu vykonávať určitú formu akcie - zopnutie termostatu kúrenia, uzavretia okien, atď.(Ovidiu Vermesan, Peter Friess, 2014)

Navrhnutú senzorickú sieť je nutné určitým spôsobom spravovať a monitorovať. Spoločnosti umožňujú vzdialenú správu či už prostredníctvom cloud-ového rozhrania alebo prostredníctvom pripojenia cez VPN do privátnej siete. Niektoré spoločnosti naopak neumožňujú vzdialenú správu senzorickej siete ale len prehľad dát ktoré boli zo siete zozbierané, sieť je možné spravovať len na mieste prostredníctvom riadiacej jednotky a určitého ovládacieho rozhrania.

1.2 Cieľ práce

Cieľom tejto práce je navrhnúť funkčnú senzorickú sieť, ktorej uzlové body budú vybavené vhodnými senzormi merajúce vybrané fyzikálne veličiny. Táto sieť má byť aplikovaná v prostredí administratívnej budovy. Uzlové body v sieti je nutné určitým spôsobom spravovať a komunikácia musí byť dostatočne zabezpečená.

Pre realizáciu takejto senzorickej siete je vhodné vykonať rešeršnú štúdiu existujúcich a dostupných riešení na trhu. Štúdia je zameraná predovšetkým na modulárne a otvorené systémy. Následne je nutné takúto štúdiu zhodnotiť, určiť metodiku riešenia a navrhnúť vlastnú senzorickú sieť.

Takýto vlastný návrh je dobré prakticky overiť v reálnych podmienkach nasadenia, kde sa môžu ukázať nedostatky, ktoré neboli pri vývoji odstránené. Takéto nedostatky je nutné zdokumentovať a navrhnúť ich prípadne opravy alebo celkové modifikácie návrhu. Dlhodobé testovanie riešenia, je realizované na reprezentatívnej časti senzorickej siete.

Na záver je nutné navrhnuté a otestované riešenie zhodnotiť, z pohľadu nasadenosti do reálnych podmienok, stability, spoľahlivosti a ďalších aspektov.

2 Analýza problému

V tejto časti diplomovej práce je preberaná problematika senzorickej siete v administratívnej budove. Bola vykonaná rešeršná štúdia, ktorá sa zameriava práve na inteligentné administratívne budovy z pohľadu vybraných spoločností. Súčasťou tejto časti diplomovej práce, je aj popis metodiky riešenia.

2.1 Rešeršná štúdia

Rešeršná štúdia je zameraná na vybrané spoločnosti z oboru chytrých administratívnych budov. V tejto časti sú popísané vybrané spoločnosti a ich riešenia z pohľadu modulárnych systémov nasaditeľných do administratívnych alebo kancelárskych priestorov.

2.1.1 iNELS

iNELS je spoločnosť sídliaca v Českej Republike, jedná sa o divíziu spoločnosti ELKO EP. Vo svojom portfóliu sa zameriava ako na chytré domácnosti tak aj na chytré administratívne budovy. Jedná sa o uzatvorený systém, ktorý bol vyvinutý v spolupráci so spoločnosťou ELKO EP. Vo svojom portfóliu zariadení ponúkajú ovládacie ale taktiež aj senzorické prvky. Oba druhy zariadení je vo väčšine zariadení možné nájst ako v drátovom tak aj bezdrátovom prevedení. Súčasťou ponuky spoločnosti iNELS sú taktiež aj riadiace prvky, ktoré zbierajú a vyhodnocujú dáta od jednotlivých koncových zariadení, či už senzorických alebo ovládacích. Navyše je možné do siete pripojiť multimedialne prvky prostredníctvom ktorých je možné sieťou šíriť či už video alebo audio streamy. (Elko EP, 2018)

Štruktúra siete využívaná systémom iNELS je otvorená niektorým otvoreným štandardom ako napríklad protokol Onvif, ktorý sa používa v oblasti IP bezpečnostnej komunikácie pre bezpečnostné kamery. Ďalej je možné spomenúť ovládanie klimatizačných jednotiek prostredníctvom technológie CoolMasterNet. Toto špecializované zariadenie dokáže komunikovať na jednej strane s klimatizačnou jednotkou pomocou sériovej komunikácie RS485 a na strane druhej s centrálnou jednotkou iNELS cez štandardnú ethernetovú sieť. (Elko EP, 2018)

Spoločnosť iNELS využíva v širokej miere produkty spoločnosti ELKO EP, fungujúce na rovnakých protokoloch ako napríklad VoIP komunikátory série ELKO EP Lara. (Elko EP, 2018)

Pre bezdrátovú komunikáciu iNELS zariadenia využívajú napríklad protokol RFIO. Tento proprietárny protokol navrhla a implementovala spoločnosť ELKO EP, funguje na bázi rádiovej komunikácie, konkrétne na frekvencii 868 MHz. Zaujímavosťou tohto protokolu a zariadení ktoré ho využívajú je, že každé zariadenie ktoré je k vysielacu pripojené, zároveň rozširuje pokrytie pre ostatné zariadenia. (Elko EP, 2018)

Nevýhoda rádio-frekvenčného prenosu spočíva v zlom šírení signálu v železo-betónových stavbách. Vysielač dosah stanice na voľnom priestranstve je približne 200m, kdežto v budove ktorá má steny zo železo-betónu je dosah do 40m.

Spoločnosť iNELS však využíva aj nové technológie bezdrátového prenosu, vyvinuté priamo pre potreby IoT¹ komunikácie a to v podobe nízko-energetickej výmeny krátkych správ. Takýmito sieťami sú napríklad siete Sigfox, LoRa alebo NB-IoT. Sieť Sigfox umožňuje prepojenie dvoch a viacerých zariadení, v ideálnych podmienkach a dostatočnom pokrytí signálom, až na vzdialenosť 50km. Práve pre tieto nízkonákladové siete, spoločnosť iNELS priniesla na trh koncom roka 2017 produktovú radu s názvom iNELS Air. Zariadenia komunikujú na frekvencii 868 MHz po nízko-energetickej sieti Sigfox, sieť Sigfox využíva technológiu ultra úzko-pásmových vln, ktoré prechádzajú bez väčších ťažkostí aj pevnými štruktúrami. Ukážkovým produktom môže byť univerzálny senzor AirIM-100S, ktorý je možné vybaviť potrebným senzorom - napríklad senzorom teploty, senzorom detekujúcim úroveň vody, senzorom meracím aktuálny odber elektrickej energie a ďalšími. Univerzálny senzor AirIM-100S, ktorý komunikuje prostredníctvom siete Sigfox, je možné vidieť na obrázku 1. (Elko EP, 2018)

Komunikácia v sieti prostredníctvom metalických vodičov je na báze i2c² komunikačného protokolu, ktorého použitie spoločnosť iNELS bližšie nešpecifikuje. Jedná sa o komunikáciu šírenú medzi koncovými zariadeniami, ktorá ústi v riadiacej jednotke. Riadiaca jednotka následne odosiela prijaté dáta buď bezdrátovo prostredníctvom RFIO protokolu alebo prostredníctvom sieťového protokolu ELKONET, ďalším centrálnym jednotkám. (Elko EP, 2018)

Spoločnosť iNELS figuruje na trhu už 23 rokov a ponúka zaujímavú kombináciu medzi uzatvoreným vlastným systémom s možnosťou kombinácie s určitými technológiami a zariadeniami, ktoré su postavené na otvorenom štandarte. Táto ponuka je však výrazne obmedzená, spoločnosť iNELS však ponúka možnosť implementácie iných protokolov a technológií, pokiaľ bude zákazník ochotný túto integráciu zaplatiť.

2.1.2 Crestron

Spoločnosť Crestron je popredný výrobca automatizačných a ovládacích systémov na svete. Svojimi produktami zastrešuje trh od automatizácie a ovládania domov cez kancelárie, hotely až po fabriky obrovských rozmerov.

Je všeobecne známe že malý inovatívny výrobcovia uspejú na zabehnutom trhu ťažšie ako obrovské korporátne spoločnosti. Preto spoločnosť Crestron umožňuje stať sa členom tzv. Crestron partners group a vyrábať zariadenia fungujúce a kooperujúce s rôznymi systémami spoločnosti Crestron. Naopak aj veľké spoločnosti ako napríklad Sony, Samsung alebo LG zahrňajú do svojich zariadení rozhranie

¹IoT - Internet of Things (Internet vecí)

²Inter-Integrated Circuit sériová zbernica multi-master, vyvinutá spoločnosťou Phillips



Obrázok 1: Univerzálny senzor - AirIM-100S spoločnosti iNELS

pre pripojenie do Crestron siete a umožniť vzdialenú správu a monitoring svojich vybraných produktov.

Čo sa týka administratívnej sféry, spoločnosť Crestron ponúka monitorovacie systémy, komunikačné systémy pre audio alebo audio-video komunikáciu, ovládacie systémy, bezpečnostné systémy, systémy pre zdieľanie a reprodukovanie audia a videa a ďalšie. V portfóliu koncových zariadení je možné nájsť štandardne drátové alebo bezdrátové koncové zariadenia, ako napríklad senzorické moduly pre meranie fyzikálnych veličín, ovládacie prvky na osvetlenie, ovládanie vzducho-techniky a mnoho ďalších. (Crestron, 2018)

Crestron ponúka širokú škálu technológií v oblasti bezdrátovej komunikácie, používa rádio-frekvenčnú komunikáciu v rozsahu 850 - 900 MHz, bluetooth a WiFi na frekvenciách 2.4 a 5GHz. Sieť infiNET - EX je produktom spoločnosti Crestron, jedná sa o mesh topológiu³, kde koncové zariadenie, napríklad dotyková obrazovka, slúži zároveň k rozšíreniu dosahu a zvýšeniu intenzity signálu infiNET - EX siete. Táto sieť pracuje podobne ako WiFi, na frekvencii 2,4 GHz, v jednej sieti môže byť pripojených až 100 zariadení. Na obrázku 2, je možné vidieť senzor intenzity zeleného svetla, ktorý komunikuje prostredníctvom infiNET - EX siete a je napájaný dvoma AAA batériami. (Crestron, 2018)

Centrálne jednotky respektíve, vyrába Crestron v spolupráci so spoločnosťou Dell, jedná sa o výkonné počítače vybavené potrebnými komunikačnými perifériami. Centrálna jednotka používa špeciálny operačný systém navrhnutý spoločnosťou Crestron v úzkej spolupráci so spoločnosťou Microsoft. Systém zbiera a ukladá všetky nahromadené dáta, umožňuje centrálnu správu budovy prípadne niekoľkých samostatných budov. S jednotlivými uzlami komunikuje po štandardnej sieťovej

³Mesh topológia - topológia kde niektoré uzly sú priamo prepojené s ďalšími uzlami. Známe tiež ako zmiešaná topológia

infraštruktúre. Navyše, vybrané centrálné jednotky poskytujú rozhranie pre pripojenie webových alebo mobilných aplikácií. (Crestron, 2018)

V portfóliu spoločnosti Crestron je možné nájsť zariadenia označované ako takzvaný protokolový prekladač. Využíva sa pri potrebe pripojenia zariadenia ktoré nepodporuje Crestron štandard ale funguje na niektorom z iných protokolov. Tieto prekladače sú programovateľné jednotky, ktoré na strane jednej komunikujú so zariadením napríklad prostredníctvom protokolu Ethernet MODBUS a na strane druhej uzavretým protokolom spoločnosti Crestron. Jednotky je možné použiť napríklad pre monitorovanie a ovládanie vybraných výtahových motorov, ktoré je možné spravovať prostredníctvom štandardu MODBUS RS485. (Crestron, 2018)

Spoločnosť Crestron je na trhu od roku 1971 a pôsobí po celom svete. Je charakteristická vysokou spoľahlivosťou, bezpečnosťou no aj vysokou cenou. Svojim širokým záberom jednotlivých odvetví splní požiadavky všetkých náročných ale aj nenáročných majetných zákazníkov.



Obrázok 2: Bezdrátový senzor GLS-LDL-EX-BATT spoločnosti Crestron

2.1.3 KNX

Organizácia KNX je tvorcom celosvetového štandardu v oblasti riadenia a monitorovania domov a budov. Vytvorila protokol, ktorý nesie rovnaký názov ako sama organizácia - KNX. Protokol KNX nahradzuje zastaralé protokoly ako napríklad protokol EHS⁴, alebo protokol EIB⁵. Protokol umožňuje komunikáciu cez silové rozvody elektrickej siete v budove, krútenej dvojlinke, rádio-frekvenčnú komunikáciu a taktiež IP komunikáciu v sieti. Ako jeden z mála protokolov spĺňa celosvetové normy pre riadenie domov a budov, napríklad medzinárodná norma ISO/IEC 14543-3, alebo európske normy rady EN-50090. (KNX, 2016)

⁴EHS - European Home Systems Protocol

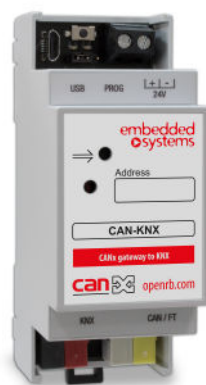
⁵EIB - European Installation Bus Protocol, známy tiež pod názvom Instabus

Organizácia KNX pod sebou zahŕňa k dnešnému dňu 442 výrobcov, ktorých produkty fungujú na štandarte KNX protokolu. Jedná sa o viac ako 7000 produktov v rôznych kategóriách, ktoré poskytnú užívateľom služby od HVAC⁶, cez zabezpečenie, ovládanie okien a žalúzií, merania spotreby zdrojov (voda, elektrická energia, plyn, ...), šírenie audia a videa a ďalšie. Zariadenia rôznych výrobcov je možné v inštalácii rôznym spôsobom kombinovať a ovládať prostredníctvom softvéru organizácie KNX. Veľa spoločností s cieľom prepojiť svoje existujúce riešenia s riešeniami organizácie KNX vyvíja prevodníky, práve na tento účel. Príkladom je aj spoločnosť Embedded Systems, ktorej prevodník z CAN bus do KNX bus TP1 je na obrázku 3. (KNX, 2018)

Každé zariadenie, ktoré má byť certifikované ako KNX produkt, je testované organizáciami tretích strán. Následne je zariadeniu udelený KNX certifikát a je zaradené do KNX produktov. Členom KNX výrobcov je taktiež už spomínaná Česká firma ELKO EP. (KNX, 2017)

Organizácia KNX má taktiež svoje produkty, ktoré spadajú do softvérového odvetvia. Od profesionálnych ovládacích, bezpečnostných a monitorovacích systémov používaných v komerčnej sfére cez softvér do riadiacich jednotiek systémov KNX až po mobilné aplikácie domácich užívateľov. Jedná sa o naozaj robustné pokrytie celého trhu, ktoré bolo dosiahnuté vďaka spolupráci veľkého množstva spoločností. (KNX, 2017)

KNX je významným hráčom na trhu automatizácie a ovládania budov a domov, na trhu je jedinečná svojím prístupom. Užívateľ má na výber množstvo modulárnych zariadení v rôznych prevedeniach a taktiež rôznych cenových hladinách, ktoré je možné ovládať jedným softvérom.



Obrázok 3: Modul spoločnosti Embedded Systems slúžiaci na prevod zo siete CAN do siete KNX

⁶HVAC - Heating/Ventilation and Air Conditioning

2.1.4 CUE

Spoločnosť CUE vznikla v roku 1991 v Českej Republike. Na trhu automatizácie sa preznámila ovládacím softvérom, ktorý komunikuje s rôznymi technológiami od rôznych výrobcov, ale v súčasnosti ponúka aj škálu svojich produktov a zariadení ktoré slúžia k automatizácii, manažmentu a ovládaniu inteligentných domácností či budov.

Integrácia technológií v podaní spoločnosti CUE, je postavená na myšlienke že užívateľ má rôzne zariadenia od rôznych výrobcov a ku každému zariadeniu má jednu ovládaciú aplikáciu. Tým pádom nie je možné jednotlivé zariadenia prepojiť a ani ovládať z jedného miesta. Tento problém sa snaží CUE eliminovať integráciou komunikačných štandardov rôznych technológií, do svojho ovládacieho systému. Spoločnosť CUE podporuje rôzne technológie a štandarty ako napríklad spomínaný KNX štandard, ďalej MODBUS, DLNA zariadenia, protokol DMX512 a mnoho ďalších. (Cue, 2018)

Spoločnosť CUE má vo svojom portfóliu zariadení, ako je už zvykom zariadenia drátové aj bezdrátové. Zariadenia komunikujú pomocou CUE protokolu CUEnet2 prípadne protokolom CUEnet alebo CUEwire. Zariadenia komunikujúce protokolmi CUEnet2 a CUEnet je možné napájať prostredníctvom PoE⁷. Portfólio zariadení zahŕňa HVAC, bezpečnostné prvky, meranie fyzikálnych veličín, ovládanie silových prvkov, ovládacie a kontrolné stanice s užívateľským rozhraním a mnoho iných. V ponuke sa vyskytujú zariadenia s rovnakým zameraním no s rôznymi prevedeniami, zariadenia určené na vnútorné inštalácie v domácnostiach, fabrikách, administratívnych budovách a taktiež zariadenia na vonkajšie použitie. Vyskytujú sa taktiež DIN moduly inštalované do rozvádáčov. Na obrázku 4, je možné vidieť modul, ktorý umožňuje pripojenie teplotného senzoru a 8 vstupných alebo výstupných zariadení, napríklad LED ovládače alebo stmievače, komunikuje protokolom CUEwire prostredníctvom metalických vodičov. (Cue, 2018)

Softvérové produkty spoločnosti CUE, zahŕňajú ovládacie a monitorovacie aplikácie ale taktiež aplikácie prostredníctvom ktorých je možné CUE ovládacie jednotky možné naprogramovať aby komunikovali s určitým zariadením a získavali relevantné dáta po podporovanom protokole. Jedná sa napríklad o produkt Cue Visual Composer. Týmto odpadá nutnosť podpory konkrétneho zariadenia a je potrebné len naprogramovať riadiace jednotky. Z pohľadu ovládacích a monitorovacích aplikácií ponúka spoločnosť CUE standalone aplikáciu a taktiež webovú aplikáciu. (Cue, 2010)

Spoločnosť CUE zaujala zaujímavú pozíciu na trhu automatizácie a monitorovania domov a budov. Užívateľ nie je viazaný na už existujúce riešenie pokiaľ je podporované systémom CUE a môže ho rozšíriť produktami spoločnosti CUE alebo spoločnosť, ktoré systém CUE podporuje.

⁷PoE - Power over Ethernet



Obrázok 4: Modul spoločnosti CUE s názvom inputCUE-W

2.1.5 ABB

Spoločnosť ABB pôsobí na trhu elektroinštalácií už veľmi dlhú dobu, v roku 2011 začala spolupracovať s organizáciou KNX a spolupráca trvá dodnes. ABB prišlo so systémom ABB i-bus KNX, so zameraním na budovy veľkých rozmerov. (KNX, 2011)

Spoločnosť má už dlhšiu dobu vo svojom portfóliu produkty, ktoré sama označuje ako inteligentnú elektro-inštaláciu. Pod týmto pojmom je možné si predstaviť napríklad monitorovanie aktuálneho odberu na jednotlivých vetvách elektro-inštalácie, ovládateľné zásuvky, ovládanie svetiel a vzduchotechniky. Práve spolupráca s KNX umožňuje prepojenie systému, ktorý ovláda akčné prvky v budove, so systémom, ktorý celú budovu monitoruje. (KNX, 2011)

Využitie potenciálu technológie KNX, dáva obrovský priestor pre výber a kombinovanie rôznych monitorovacích technológií do administratívnej budovy podľa potrieb danej inštalácie. (KNX, 2011)

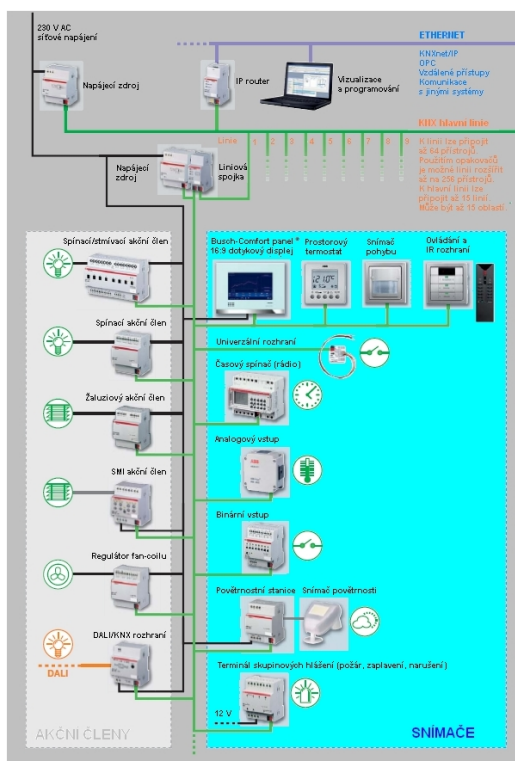
Komunikácia medzi ABB prvkami sa realizuje prostredníctvom dvojžilového vodiča, pre prepojenie s existujúcou alebo budúcou KNX inštaláciou je potrebné použiť špeciálny IP interface, ktorý prevádza prijaté dáta na dvojžilové vedenie a naopak. ABB poskytuje možnosť nastavenia jednotlivých protokolových prevodníkov, pomocou špeciálneho plateného softvéru. (KNX, 2011)

ABB umožňuje svoje inteligentné elektro-inštalácie riadiť centralizovane, pomocou počítačového softvéru alebo prostredníctvom mobilnej aplikácie, ktorá komunikuje s GSM modulom pripojeným v ABB inteligentnej elektro-inštalácii. (KNX, 2011)

Portfólio produktov systému ABB i-bus KNX, zahŕňa spomínané monitorovanie a ovládanie akčných členov, ovládanie stmievania a žalúzií, ovládanie termostátov a vzduchotechniky, bezpečnostné prvky ako alarm, sirény a detektory pohybu. Ďalej je možné použiť obdobu PLC kontrolérov ako vstupné alebo výstupné programova-

telné jednotky, vyrábajú sa ako digitálne tak aj analógové verzie s rôznymi počtami pinov. Niektoré zo zariadení sú napájané priamo zo siete elektro-inštalácie prípadne je u IP zariadení možné použiť POE, pokiaľ sa jedná o zariadenie, ktoré leží na ovládanej vetve inteligentnej elektro-inštalácie. Samozrejme nie je možné zabúdať na zariadenia ostatných spoločností, ktoré pre komunikáciu využívajú práve KNX protokoly a sú taktiež použiteľné v systéme. (KNX, 2011) Vzorový príklad topológie systému ABB i-bus KNX je možné vidieť na obrázku 5.

Nakoľko systém ABB i-bus KNX, neumožňuje medzi ABB produktami bezdrátovú komunikáciu, je nutné pri návrhu riešenia brať na túto skutočnosť ohľad. Použitie systému ABB je spôsobom fungovania vhodné skôr pri realizovaní nových stavieb, prípadne pri čiastočných alebo kompletných rekonštrukciách elektrických rozvodov, v už existujúcich budovách.



Obrázok 5: Vzorová topológia zapojenia zariadení ABB i-bus KNX

2.1.6 BigClown

Spoločnosť BigClown sa zameriava primárne na domácu automatizáciu, avšak niektoré moduly sú použiteľné aj v menších administratívnych budovách, sídli v Českej Republike. Oproti predchádzajúcim spoločnostiam popísaných v rešerši, sa jedná o úplne otvorenú architektúru ako hardvérovej tak aj softvérovej časti riešenia. Portfólio produktov spoločnosti BigClown obsahuje niekoľko modulov, ktoré si môže

zákazník zakúpiť a upraviť podľa svojich potrieb. Navyše je možné dané moduly rozšíriť napríklad o ďalšie senzorické dosky, čo však nie je úplne jednoduché. Pri práci s modulmi spoločnosti BigClown, je potrebné zostaviť svoj vlastný firmvér, ktorý je nutné vypáliť na MCU, následne je potrebné naprogramovať obsluhu pripojeného senzoru v jazyku C. V niektorých prípadoch je možné použiť grafický flow-editor NodeRed a zabezpečiť tak správnu komunikáciu modulu so senzorom. Tento prístup je svojim spôsobom unikátny voči predchádzajúcim popisovaným spoločnostiam. Umožňuje zákazníkovi znalým problematiky, tvorbu vlastného unikátneho zariadenia s využitím základu od spoločnosti BigClown. Jedná sa o riešenie vhodné taktiež pre edukačné účely v oblasti automatizácie a IoT. (BigClown, 2018)

Spoločnosť BigClown ponúka niekoľko základných modulov, ktoré slúžia napríklad k ovládaniu určitého zariadenia pomocou tlačidla, tvorbe vlastného digitálneho termostatu, meraniu niektorých fyzikálnych veličín ako napríklad teploty, vlhkosti, úrovne oxidu uhličitého, atď. Modul je umiestnený v krabičke, ktorá je vytlačená na 3D tlačiarňi. Zdrojové súbory pre 3D tlač sú taktiež voľne k dispozícii, takže je možné si krabičku upraviť podľa predstáv. (BigClown, 2018)

Zariadenia spoločnosti BigClown komunikujú pomocou rádiovkej technológie na frekvencii 868/915 MHz, jedná sa o voľné komunikačné pásmo dizajnované pre prenos krátkych správ. Spoločnosť BigClown vo svojej dokumentácii k projektu tvrdí, že je možné z jedného vysielačového miesta pokryť celý dom, o aký veľký dom sa jedná, sa v dokumentácii žiaľ neuvádza. Core modul, ako ho spoločnosť BigClown nazýva, sa skladá z niekoľkých častí - obsahuje komunikačnú jednotku zabezpečujúcu rádiovú komunikáciu, 32-bitový procesor ARM 0+, 20 kB RAM a 192 kB flash pamäte pre program. Ďalej obsahuje snímač teploty, akcelerometer, signálnu LED diódu, USB konektor pre komunikáciu s PC a 18 GPIO portov, ktoré je možné využiť pre tvorenú aplikáciu, alebo pre pripojenie BigClown senzorického modulu. Operačné napätie je stanovené v rozmedzí 2 - 3,6V. Pre zabezpečenie komunikácie používa špeciálne vyvinutý mikročip ATSHA204A, ktorý využíva SHA metódy pre kódovanie a dekódovanie komunikácie. (BigClown, 2018)

Zariadenia obsahujúce Core modul, dokážu komunikovať medzi sebou prostredníctvom rádiovkej komunikácie. Avšak na to, aby sa dáta dostali napríklad do užívateľskej aplikácie, je potrebné si zakúpiť tzv. USB Dongle, ktorý slúži k prenosu dát z rádiovkej komunikácie do počítača a opačne. Z počítača je možné dáta posielat prostredníctvom MQTT protokolu napríklad na MQTT prostredníka, odkiaľ dáta putujú ďalej do užívateľskej aplikácie, alebo sa ukladajú do databázy pre budúce spracovanie. K jednému USB komunikátoru je možné pripojiť maximálne 32 Core modulov. (BigClown, 2018)

Spoločnosť BigClown prináša na trh naozaj zaujímavé riešenie, ktoré je možné upraviť podľa svojich predstáv. Na obrázku 6, je možné vidieť otvorenú stanicu pre monitorovanie úrovne oxidu uhličitého v ovzduší.



Obrázok 6: BigClown modul pre monitorovanie oxidu uhličitého v ovzduší

2.2 Zhodnotenie výsledkov rešeršnej štúdie

Do rešeršnej štúdie bolo zahrnutých 5 spoločností a jedna celosvetová organizácia. V podaní spoločností CUE, Crestron, ABB a organizácie KNX, je možné hovoriť ako o celosvetových gigantoch na trhu automatizácie domácností a administratívnych budov. Medzi jednotlivými spoločnosťami je možné pozorovať rôzny prístup pri navrhovaní a budovaní automatizovaných systémov. Napríklad spoločnosť CUE vznikla ako spoločnosť, ktorá vyvíjala ovládací softvér pre existujúce systémy na trhu - umožnila prepojenie navzájom neprepojených systémov v jednom objekte a ich riadenie z jedného miesta. Spoločnosť CUE však využila svoje postavenie a vyvinula aj niekoľko hardvérových zariadení, ktoré na trhu neboli dostupné alebo nemali dostatočnú certifikáciu pre ich použitie v inštaláciách. Spoločnosť Crestron, používa svoj vlastný uzavretý komunikačný protokol, ale taktiež umožňuje pripojenie iného zariadenia do takejto siete a to prostredníctvom protokolových prekladačov. Je teda evidentné, že aj pre obrovských výrobcov ako je spoločnosť Crestron, je integrácia iných technológií, od iných výrobcov, kľúčová práve v období obrovského rozmachu v sfére inteligentných budov ako aj v sfére IoT.

Veľmi dôležitým aspektom pri budovaní inteligentnej inštalácie či už v dome, alebo v administratívnej budove, je spôsob prepojenia a teda komunikácie medzi jednotlivými zariadeniami. Základné delenie je veľmi jednoduché, komunikácia je realizovaná buď po metalických vodičoch alebo bezdrátovo s využitím rôznych prenosových technológií ako napríklad rádiovú komunikáciu alebo WiFi štandard. Je pravdepodobné, že pri budovaní senzorickej siete v existujúcej administratívnej budove sa upredností práve bezdrátová komunikácia, pokiaľ možno po existujúcej sieti aby boli náklady čo najnižšie. Práve preto väčšina spoločností, ktoré sú na trhu, ponúkajú na výber obe možnosti. Je treba ale poukázať aj na fakt, že komunikácia metalickými vodičmi, môže byť v určitých podmienkach spoľahlivejšia, ako komunikácia bezdrátová.

Neoddeliteľnou súčasťou komunikácie je bezpečná výmena dát v sieti. Práve bezpečnosť je téma ktorá sa v posledných rokoch stále častejšie spája s inteligentnými inštaláciami a celkovo so zariadeniami z oblasti IoT. Predstava že útočník získa kontrolu nad administratívnou budovou a bude schopný napríklad otvárať okná, prípadne dokonca dvere nie je úplne nereálna. Práve preto napríklad organizácia KNX, prišla s dvoma zabezpečovacími rozšíreniami pre všetky druhy svojich protokolov, či už sa jedná o IP protokoly v sieti alebo dátové protokoly komunikujúce mimo IP štandarty. Jedná sa o rozšírenia KNX IP Secure a KNX Data Secure, ktoré šifrujú komunikáciu medzi jednotlivými zariadeniami. Všetci výrobcovia ktorí vyvíjajú zariadenia do siete KNX sú povinný implementovať rozhrania pre tieto rozšírenia a to od verzie ETS 5.5, respektíve od roku 2014. KNX siete, ktoré vznikli na starších verziách ETS, je možné dovybaviť špeciálnymi zariadeniami, ktoré zabezpečia bezpečnú komunikáciu po internete. Sieť je teda možné konfigurovať a jednotlivé zariadenia ovládať bezpečne, odkiaľkoľvek. Ako ďalší príklad zabezpečenia a autenticity komunikácie medzi jednotlivými zariadeniami je metóda EAP-TLS, ktorú využíva na svojich zariadeniach aj spoločnosť Crestron, v oblasti šifrovania bezdrátovej komunikácie využíva na WiFi štandarte WEP, WPA a WPA2. Všetka komunikácia po IP protokoloch je šifrovaná pomocou SSL respektíve TLS protokolu. Crestron mimo iné spolupracuje v oblasti bezpečnosti so spoločnosťou Microsoft, ktorej operačné systémy v upravenej podobe používa na svojich zariadeniach.

So zaujímavým riešením zabezpečenia prenášaných dát, prišla spoločnosť BigClown, ktorá používa špeciálny mikročip určený práve na šifrovanie a dešifrovanie prenášaných dát, čím odľahčila hlavnú výpočtovú jednotku.

2.3 Požiadavky na senzorickú sieť

Pri stanovení požiadaviek na senzorickú sieť je vhodné vychádzať z vykonanej rešeršnej štúdie, ktorá napomáha k tvorbe predstavy senzorickej siete.

Senzorická sieť by mala byť nasaditeľná ako do existujúcej, tak aj do novo vznikajúcej administratívnej budovy. Mala by byť postavená na moderných technológiách, ktoré umožňujú dostatočné zabezpečenie komunikácie, nakoľko je nutné brať v potiaž to, že dáta, ktoré senzorická sieť produkuje, môžu v budúcnosti ovplyvňovať chovanie ostatných systémov v budove. Tento aspekt je vhodné zakomponovať do výberu vhodných senzorov a samotného návrhu senzorického uzlu. Sensory by mali byť dostatočne presné, aby namerané dáta odpovedali čo najvernejšie skutočným hodnotám. Taktiež je nutné aby sieťové uzly zbierali dáta a následne ich prostredníctvom siete, pokiaľ je to možné, odosieli v pravidelných intervaloch.

Sieťový uzol by mal byť kompaktných rozmerov a mal by byť vybavený potrebnou senzorickou výbavou, ktorá mu umožní merať vybrané fyzikálne veličiny. Je potrebné myslieť na to, že nie všetky časti administratívnej budovy sú rovnaké a jednotlivé miestnosti sa od seba líšia. Preto je nutné navrhnúť rôzne verzie senzorických uzlov, ktoré budú môcť byť vybavené rôznymi typmi senzorov.

Senzorickú sieť je vhodné určitým spôsobom spravovať. Takýto nástroj by mal slúžiť k rýchlej identifikácii problému, ktorý môže nastať na určitom senzorickom uzle a napomôcť tak k promptnej náprave. Je teda vhodné, aby tento nástroj informoval správcu siete o vzniknutých problémoch. Navyše je potrebné aby tento nástroj poskytoval určité možnosti vzdialenej správy jednotlivých senzorických uzlov a aby jednotlivé senzorické uzly na takéto požiadavky reagovali. Môže sa jednať napríklad o kontrolné získanie dát z konkrétneho uzlu, prípadne o jeho reštartovanie alebo o vzdialenú aktualizáciu zdrojového kódu.

V neposlednom rade je taktiež nutné myslieť na to, že takto navrhnutá senzorická sieť musí byť do budúcnosti rozširiteľná a to podľa novo vznikajúcich potrieb správcov administratívnej budovy. Je teda potrebné navrhnuť sieť tak, aby ju bolo možné rozširovať o ďalšie senzorické ale aj akčné uzly. Pod pojmom akčný uzol, je myslený uzol, ktorý na základe prijatých dát vykoná určitú akciu, napríklad sa môže jednať o zavretie okien v miestnosti pri poklese teploty v zimnom období. K splneniu tohto cieľu môže napomôcť technická dokumentácia ako aj táto diplomová práca, kde je podrobne vysvetlený návrh a princíp fungovania.

Samozrejmosťou senzorickej siete, musí byť určitý spôsob odovzdávania dát, ktoré senzorická sieť nahromadí k ďalšiemu spracovaniu. Tento spôsob by mal byť navrhnutý a realizovaný taktiež pomocou moderných technológií, ktorých nasadenie je pomerne jednoduché a dobre zdokumentované. Odosielané dáta by mali byť v jednotnej štruktúre, ktorá umožní napríklad ich ukladanie do databáze.

Komunikácia medzi jednotlivými sieťovými uzlami a centrálnym prvkom, ktorý zbiera dáta produkované sieťovými uzlami by mala byť čo najviac optimalizovaná. Uzly by nemali produkovať dáta, ktoré je možné vygenerovať na centrálnom prvku. Takouto optimalizáciou sa zaručí, nižšie zaťaženie siete ale aj samotných sieťových uzlov.

Senzorickú sieť je potrebné navrhnuť tak, aby jej použitie bolo čo možno najbezpečnejšie.

V nasledujúcej časti tejto diplomovej práce, je popísaná metodika riešenia ako aj technológie použité v senzorickej sieti.

3 Návrh senzorickej siete

Táto časť diplomovej práce je zameraná na samotný návrh senzorickej siete ako celku a teoretický rozbor použitých technológií v senzorickej sieti. V jednotlivých podkapitolách sú vysvetlené základné princípy fungovania a popis jednotlivých častí senzorickej siete nutných k návrhu a následnej implementácii.

Navrhnutá senzorická sieť musí odpovedať stanoveným požiadavkám v predchádzajúcej kapitole tejto diplomovej práce. V úvode tejto kapitoly v sekciách 3.1 a 3.2, je popísaný návrh senzorického uzlu a správcovskej aplikácie. V ďalších sekciách sú popísané využité technológie pri realizácii navrhovanej senzorickej siete. V závere tejto kapitoly sa nachádza stručný popis softvérových nástrojov, ktoré boli použité pri implementácii senzorickej siete.

3.1 Senzorický uzol

Senzorický uzol je zariadenie, ktoré zbiera dáta z pripojených senzorov a distribuuje ich do senzorickej siete. Každý senzorický uzol musí byť v sieti jednoznačne identifikovateľný a musí byť možné ho spravovať.

Senzorický uzol musí byť určitým spôsobom prepojený so zvyškom senzorickej siete. Je teda potrebné navrhnuť finálne riešenie senzorického uzlu na technológii, ktorá umožňuje takéto pripojenie a zároveň je nutné rešpektovať požiadavok na nasaditeľnosť do reálnych podmienok existujúcej alebo novo vznikajúcej administratívnej budovy. Na základe tejto podmienky, bol zvolený mikrokontrolér so zabudovaným WiFi rozhraním a to konkrétne ESP8266 verzia 12s, ktorého bližší popis je možné nájsť v sekcii 3.5.1. Tento mikrokontrolér, je schopný komunikovať bezdrátovo prostredníctvom WiFi, čo umožňuje nasadenie senzorického uzlu do ľubovolnej časti miestnosti, za predpokladu dostatočnej intenzity WiFi signálu. Aby bolo možné zbierať dáta z pripojených senzorov, je nutné aby dané senzory boli kompatibilné s mikrokontrolérom a jeho firmvérom. Je preto nutné, vhodne zvoliť senzory vybraných fyzikálnych veličín. Fyzikálne veličiny ktoré budú merať senzorické uzly sú teplota, vlhkosť a úroveň svetelnej intenzity. V jednotlivých podkapitolách 3.3 a 3.4, sú vybrané typy senzorov bližšie popísané. Namerané dáta je potrebné distribuovať po senzorickej sieti určitým komunikačným protokolom. Dáta zo senzorov, by mali byť merané a odosielané do senzorickej siete automaticky, v pravidelných časových intervaloch. Ako vhodný komunikačný protokol, bol zvolený Message Queuing Telemetry Transport protokol, ktorý je popísaný v sekcii 3.6. Tento protokol je podporovaný vybraným mikrokontrolérom a svojim fungovaním sa hodí pri komunikácii M2M⁸ zariadení, práve kvôli svojej nízkej hardvérovej náročnosti a spoľahlivosti. Pre zaistenie správy senzorických uzlov, je potrebné implementovať rozhranie na strane senzorických uzlov, ktoré umožní vzdialenú správu. A na strane druhej, je potrebné implementovať aplikáciu, ktorá administrátorovi

⁸M2M - Machine to Machine

senzorickej sieti, umožní senzorické uzly spravovať. Pre indikáciu súčasného stavu jednotlivých senzorických uzlov, je potrebné zabezpečiť pravidelné reportovanie stavov jednotlivých senzorických uzlov. Senzorické uzly by mali umožniť vzdialenú aktualizáciu zdrojového kódu aby v prípade nutnosti akejkoľvek zmeny v programe, nebolo potrebné fyzicky senzorické uzly demontovať a ručne preprogramovať. Takúto činnosť zabezpečí implementovaná aplikácia, ktorá bude iniciovať aktualizáciu prostredníctvom senzorickej siete k vybraným modulom.

3.2 Adaptér

Správcovska aplikácia popisovaná v tejto diplomovej práci, bola nazvaná adaptér, nakoľko umožňuje distribúciu dát zo senzorických uzlov mimo senzorickú sieť a umožňuje teda napojenie iných technológií. Dáta, ktoré adaptér zozbiera zo senzorickej siete, uloží do svojej lokálnej databáze a sprostredkuje vizualizáciu obmedzenej vzorky zozbieraných dát. Adaptér taktiež umožní vzdialenú správu a aktualizáciu programu na vybraných senzorických uzloch. Aktualizáciu si každý senzorický uzol prevezme samostatne, aby bola zaručená konzistencia prenášaného súboru na každom aktualizovanom senzorickom uzle. Vzdialená správa senzorických uzlov je realizovaná prostredníctvom správ, ktoré nesú informáciu o akcii, ktorú má senzorický uzol vykonať. Môže sa napríklad jednať o pokyn, k vzdialenému reštartu senzorického uzlu.

3.3 Svetelné senzory

Snímanie svetla v miestnosti môže viesť k rôznym benefitom. Samotné meranie svetla ako fyzikálnej veličiny, môže byť ponímané z niekoľkých pohľadov, svetlo je možné napríklad merať ako jednotlivé RGB zložky alebo ako jednotnú veličinu, za účelom zmerať svetelnú intenzitu. Na trhu je k dispozícii široká ponuka senzorov pre meranie rôznych vlastností svetla, existujú varianty analógové ale aj digitálne. Benefity v oblasti senzorickej siete administratívnej budovy sú rôzne, môže sa jednať napríklad o úsporu energií, prispôbenie svetelnej intenzity svetelným podmienkam miestnosti alebo aj prispôbenie teploty vyžarovaného svetla.

Snímanie svetla je možné realizovať rôznymi spôsobmi, medzi známe súčiastky schopné merať úroveň svetla patria foto rezistory, foto diódy a foto tranzistory. Meranie svetla je možné vďaka zmene jednej z charakteristík, napríklad pri dopade svetla na foto rezistor dôjde k zmene jeho odporu. (Paul Horowitz, Winfield Hill, 2015)

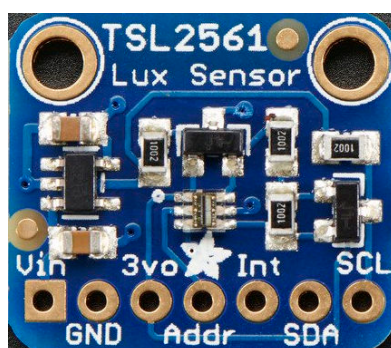
3.3.1 TSL-2561

Senzor TSL-2561, je digitálny senzor svetelnej intenzity od spoločnosti Adafruit. Jedná sa o senzor, ktorý komunikuje i2c protokolom. TSL-2561 meria svetelnú intenzitu v jednotkách Lux a to v škále 0,1 - 40 000 Lux. Senzor je vybavený

infra-červenými a širokospektrálnymi diódami, ktoré umožňujú separátne meranie infra-červeného, plného a ľudským okom viditeľného spektra svetla. (TAOS, 2009)

Senzor pracuje pri operačnom napätí 3.3V, ale je vybavený napäťovým regulátorom z 5V na 3.3V, nakoľko množstvo mikrokontrolérov pracuje práve na 5V. Pri meraní spotrebovávajú podľa technickej špecifikácie 0.5mA a v kludovom stave približne 15uA. Na jednej i2c zbernici je možné mať pripojené až 3 rovnaké senzory, nakoľko je možné pomocou vhodnej logickej hodnoty nastaviť 3 rôzne adresy. (TAOS, 2009)

Senzor je možné vidieť na obrázku 7.



Obrázok 7: Digitálny senzor svetelnej intenzity TSL-2561

3.3.2 TCS-34725

Senzor TCS-34725 je svetelný senzor RGB spektra od spoločnosti Adafruit. Výrobca ho popisuje ako „light-to-digital“ prevodník, senzor je vybavený snímacím polom foto-diód o veľkosti 3x4 foto-diódy. Jednotlivé foto-diódy sú vybavené svetelnými filtermi aby umožnili meranie jednotlivých zložiek svetla. Senzor je taktiež vybavený foto-diódami bez svetelných filtrov, ktoré slúžia k meraniu plného spektra. Na doske plošných spojov senzoru TCS-34725 sa nachádzajú 4 ADC⁹ prevodníky. Výstup z prevodníkov smeruje na výstupné registre senzoru, odkiaľ sú hodnoty zaznamenávané mikrokontrolérom. (TAOS, 2012)

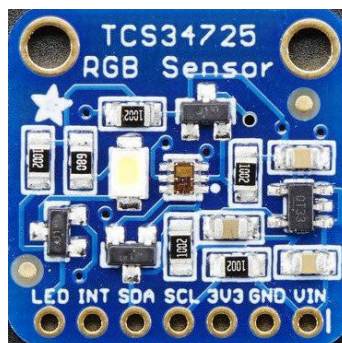
Tento RGB senzor je vybavený infra-červeným filtrom, na koľko infra-červené svetlo nie je viditeľné ľudským okom. Doska plošných spojov senzoru obsahuje osvetľujúcu diódu, ktorú je možné použiť napríklad pri detekcii farby predmetu pri nedostatočnom osvetlení. Diódu je možné ovládať pomocou logického výstupu z mikrokontroléru. V našom prípade užitia sa takéto prídavné osvetlenie nepoužíva a dióda je napevno vypnutá. (TAOS, 2012)

Senzor má operačné napätie podľa technickej špecifikácie stanovené na rozpätie hodnôt 3.3V - 5V. Spoločnosť Adafruit sa snaží unifikovať spôsob použitia senzorov

⁹ADC - Analog to Digital Converter prevodník spojitého signálu na číslicový

a teda rovnako ako senzor TSL-2561, aj TCS-34725 obsahuje regulátor napätia a komunikuje i2c protokolom. (TAOS, 2012)

Senzor je možné vidieť na obrázku 8.



Obrázok 8: Digitálny senzor RGB spektra TCS-34725

3.4 Sensory teploty a vlhkosti

Meranie teploty a vlhkosti v jednotlivých častiach administratívnej budovy vedie podobne ako meranie svetla a jeho zložiek k rôznym benefitom. Na základe nameraných dát, je možné napríklad šetriť energie alebo aj vytvárať lepšie podmienky pre prácu v kanceláriách alebo miestnostiach administratívnej budovy.

Na trhu je množstvo senzorov, ktoré umožňujú meranie teploty a vlhkosti. K dispozícii sú taktiež senzory, ktoré majú integrované periférie k meraniu oboch fyzikálnych veličín. Sensory sa ponúkajú buď ako číslicové so zabudovanými ADC prevodníkmi alebo ako senzory spojitie, ktorých signály môžu byť v závislosti od potreby digitalizované.

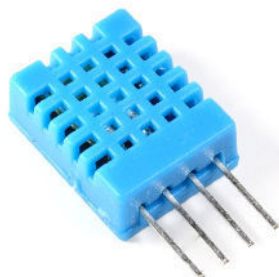
Meranie teploty môže byť realizované rôznymi spôsobmi, veľmi často sa v elektrotechnike používajú termistorové senzory, bimetalové senzory, infračervené senzory a iné. Meranie vlhkosti v princípe prebieha tromi základnými princípmi, kapacitným meraním, odporovým meraním a tepelným meraním. V praxi sa najbežnejšie používa pre meranie vlhkosti meranie odporové. (Paul Horowitz, Winfield Hill, 2015)

3.4.1 DHT-11

Senzor DHT-11, je digitálny senzor teploty a vlhkosti so zabudovaným integrovaným obvodom, ktorý slúži k prevodu dát z analógových čidiel. Skupina DHT senzorov je širšia, existuje napríklad varianta DHT-21 a DHT-22, tieto varianty sa od seba líšia presnosťou merania, veľkosťou, rozsahom merateľných hodnôt a cenou. V závislosti od výrobcu a použitých materiálov sa líšia presnosti jednotlivých senzorov, spoločnosť Adafruit, v technickom liste senzoru DHT-11 uvádza presnosť $\pm 2^{\circ}\text{C}$ pri meraní teploty a $\pm 5\%$ pri meraní vlhkosti. (Aosong, 2009)

Senzor pracuje spoľahlivo v rozmedzí 3 až 5V, pre správnu funkciu potrebuje tzv. pull-up rezistor na dátovú linku, ideálne s hodnotou $10k\Omega$. Rozsah merateľných hodnôt je 20–80% pri meraní vlhkosti a 0–50°C pri meraní teploty, so spomínanými presnosťami. Nakoľko sa jedná o digitálny senzor, výstup pre mikrokontrolér je poskytovaný integrovaným obvodom v číslicovej podobe. (Aosong, 2009)

Senzor je možné vidieť na obrázku 9.



Obrázok 9: Digitálny senzor teploty a vlhkosti DHT-11

3.4.2 Si-7021

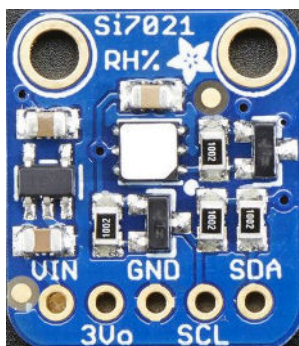
Senzor Si-7021 od spoločnosti Adafruit je digitálny senzor teploty a vlhkosti. Komunikuje prostredníctvom i2c komunikačného štandardu. Od senzoru DHT-11 sa líši nielen spôsobom komunikácie, ale aj presnosťou merania, rozsahom merateľných hodnôt a o niečo vyššou cenou. Presnosť senzoru sa podľa technického listu výrobcu pohybuje v rozmedzí $\pm 0.4^{\circ}\text{C}$ pri meraní teploty a $\pm 3\%$ pri meraní vlhkosti. (Silicon Labs, 2016)

Si-7021 od spoločnosti Adafruit je vybavený regulátorom napätia a jeho vstupné napätie je stanovené na 3 až 5V. Výstup vstavaného regulátoru je 3.3V, zaťažiteľný do 100mA a je privedený na samostatný výstup senzoru, kde je možné ho použiť v prípade potreby na ďalšie zariadenie. Merateľný rozsah senzoru je -10°C až 85°C pri meraní teploty a 0 až 80% pri meraní relatívnej vlhkosti okolia. (Silicon Labs, 2016)

Senzor je možné vidieť na obrázku 10.

3.5 Mikrokontrolér

Mikrokontrolér je zariadenie, ktoré na jednej doske plošných spojov obsahuje niekoľko periférií. Hlavnou časťou mikrokontroléru je výpočtová jednotka - procesor, ktorý vykonáva jednotlivé inštrukcie. Výkon respektíve takt takéhoto procesoru je daný frekvenciou oscilátora ktorý je buď súčasťou dosky plošných spojov mikrokontroléru alebo časťou samotného procesoru. Procesor taktiež obsahuje sady registrov,



Obrázok 10: Digitálny senzor teploty a vlhkosti Si-7021

ktoré sa používajú počas bežiacieho programu. Od zvolenej architektúry procesora sa odvíja práca procesoru s pamäťou. Procesory s Harvardskou architektúrou môžu pracovať s dátami a inštrukciami simultánne, kdežto procesory s Von Neuman architektúrou môžu v jeden moment používať len jeden druh pamäte - inštrukčnú alebo dátovú. Niektoré mikrokontroléry sú vybavené externou pamäťou, ktorá môže slúžiť pre iniciálne načítanie programu alebo pre uchovávanie jeho výstupu. (Paul Horowitz, Winfield Hill, 2015)

Mikrokontrolér býva vybavený vstupne/výstupnými perifériami, pomocou ktorých komunikuje so svojím okolím. Štandardne sa jedná o GPIO¹⁰ digitálne piny. Niektoré mikrokontroléry sú vybavené taktiež analógovými pinmi a ADC/DAC prevodníkmi, pomocou týchto pinov je možné pracovať napríklad s analógovými senzormi alebo čidlami. (Paul Horowitz, Winfield Hill, 2015)

Veľmi podstatnou súčasťou mikrokontrolérov zameraných na oblasť IoT, je schopnosť komunikácie s okolím. Táto komunikácia má opäť základné delenie na 2 časti, komunikácia po metalických vodičoch (TTL, RS-232, Ethernet, i2c, ...) alebo bezdrátová komunikácia (WiFi, rádiové vlny, Bluetooth, ...).

V závislosti od použitého mikrokontroléru a firmvéru, je kód programu buď interpretovaný alebo kompilovaný. Kód mikrokontroléru v jazyku C je nutné pred samotným nahratím alebo výpálením do pamäte skompilovať, kdežto kód interpretovaný sa naopak pred nahratím nekompiluje ale priamo nahráva do pamäte. (Paul Horowitz, Winfield Hill, 2015)

Niektoré mikrokontroléry umožňujú použitie určitej nadstavby nad výrobcom dodávanou SDK¹¹ v podobe virtuálneho stroja. Príkladom je mikrokontrolér ESP-8266 a firmvér od spoločnosti NodeMCU. Spoločnosť NodeMCU zabudovala do svojho firmvéru Lua Virtual Machine - LVM, ktorá dokáže interpretovať kód v jazyku Lua. Napriek tomu že jazyk Lua je jazykom interpretovaným, je možné tieto zdrojové súbory pred-kompilovať do tzv. Lua Byte Code, ktorý je následne interpretovaný. Výhodou použitia takejto predkompilácie je odhalenie prípadných syntaktických

¹⁰GPIO - General Purpose Input Output vstupne výstupný pin mikrokontroléru

¹¹SDK - Software Development Kit

chýb v kóde o niečo skôr ako bude daná časť kódu interpretovaná prekladačom ale tiež aj úspora miesta na FLASH pamäti. (NodeMCU, 2017a)

3.5.1 ESP-8266 12s

Mikrokontrolér ESP-8266 verzia 12s, je mikrokontrolér so zabudovaným WiFi kontrolérom. Jedná sa o tzv. SoC¹² architektúru, kde všetky periférie nutné pre chod mikrokontroléru sú umiestnené na jednom mikročipe. Samotný mikročip je vyrábaný čínskou spoločnosťou Espressif, ktorá tento čip predáva pod označením ESP8266EX. Spoločnosť Ai-Thinker z týchto mikročipov vyrába rôzne verzie zariadení označovaných ako ESP-8266 xx, kde xx značí číslo verzie. V závislosti od verzie mikrokontroléru ESP-8266 sa líši počet vstupne-výstupných pinov, veľkosť pamäte flash a veľkosť samotnej dosky plošných spojov, na ktorej je mikrokontrolér umiestnený. Všetky verzie ESP-8266 respektíve samotný čip ESP8266EX, sú vybavené rovnakým procesorom od spoločnosti Cadence Tensilica, konkrétne modelom L106. Jedná sa o 32-bitový procesor, ktorý beží na frekvenci 80 MHz a v prípade potreby môže byť pretaktovaný na dvoj-násobnú frekvenciu, čiže 160 MHz. Procesor L106 patrí do skupiny RISC procesorov, čiže má redukovanú inštrukčnú sadu, konkrétne používa inštrukčnú sadu Tensilica Xtensa ISA¹³. Táto sada inštrukcií obsahuje inštrukcie o dĺžke 24-bitov zamerané na zabudované aplikácie, ale taktiež inštrukcie s dĺžkou 16-bitov. Medzi jednotlivými 16/24-bitovými inštrukciami je možné dynamicky prepínať. Procesor je navrhnutý pre nízko energetické a nízko výkonové aplikácie. (Espressif, 2015) L106 má k dispozícii 32 KiB inštrukčnej pamäte RAM, 32 KiB inštrukčnej cache¹⁴ pamäte RAM, 80 KiB RAM určenej pre dáta užívateľského programu a 16 KiB ETS systémovej pamäte RAM. (Espressif, 2015)



Obrázok 11: Mikrokontrolér ESP-8266 12s

¹²SoC - System on Chip, sada integrovaných obvodov na jednom čipe, prípadne na jednej doske plošných spojov, tvoriaca funkčný celok počítača

¹³ISA - Instruction Set Architecture architektúra inštrukčnej sady

¹⁴cache - rýchla pamäť procesoru určená pre uchovávanie kópií dát, často využívaných procesorom

Ak je čip ESP8266EX používaný v takzvanom „station mode“¹⁵, k dispozícii je približne 50 KiB RAM pre program. V prípade použitia firmvéru napríklad spoločnosti NodeMCU, ktorý predstavuje nadstavbu nad SDK spoločnosti Espressif a obsahuje LVM¹⁶, je dostupná pamäť ešte menšia, približne sa jedná o 40KiB. Táto hodnota je variabilná a závisí od zvoleného zostavenia a výberu doplnkových modulov. (NodeMCU, 2017b)

Čip neobsahuje žiadnu programovateľnú perzistentnú pamäť, pre potrebu uloženia užívateľského programu je možné pripojiť k mikročipu SPI¹⁷ flash pamäť do veľkosti až 16 MB. Spoločnosť Ai-Thinker vybavuje moduly ESP-8266 v závislosti od verzie a revízie, pamäťovými modulmi o veľkosti 4-8 MB. V tejto flash pamäti je vypálený firmvér spolu s výrobcom dodávanou SDK a užívateľským programom. (Espressif, 2015)

ESP8266EX podporuje sieťové protokoly IPv4, HTTP, TCP, UDP a FTP. WiFi kontrolér obsiahnutý v mikročipe podporuje b, g a n WiFi štandarty na frekvenčnom rozsahu 2.4 - 2.5 GHz. Anténa sa líši v závislosti od verzie, vyskytujú sa antény zabudované na doskách plošných spojov, keramické antény alebo verzie s anténym konektorom. Verzia 12s má anténu zabudovanú do plošného spoja. Mikročip ESP8266EX je možné použiť v už spomínanom režime klienta - station mode, v móde „access point“¹⁸ alebo v kombinovanom móde stanice a prístupového bodu. (Espressif, 2015)

Čip ESP8266EX obsahuje jeden ADC prevodník s rozsahom 10-bitov. Pomocou tohto prevodníku je možné merať napätie v rozsahu 0 - 1 V a na výstupe je reprezentované rozsahom hodnôt 0 - 1024. Nová verzia SDK dodávaná výrobcom, umožňuje použiť tento analógový pin interne, pre meranie systémového napätia. (Espressif, 2015)

Pinout modulu ESP-8266 12s, tvorí celkom 16 pinov. K dispozícii je celkom 9 GPIO pinov, jeden pin pre prevod analógového signálu na digitálny, pin pre tzv. tvrdý reset modulu a napájacie piny. Kompletný prehľad je v tabuľke 1. Mikrokontrolér umožňuje pri štarte voľbu niekoľkých tzv. „boot režimov“¹⁹. Pomocou týchto režimov je možné načítať program z SD karty, z internej pamäte alebo zariadenie prepnúť do takzvaného flash režimu, kedy sa mikrokontrolér nesnaží načítať program z pamäte ale čaká na vstup z externého programátora. Podrobný prehľad zapojenia režimov je možné vidieť v tabuľke 2. (Espressif, 2015)

ESP-8266 12s, má dva vyhradené piny pre sériovú TTL²⁰, konkrétne sa jedná o piny TXD0 a RXD0, tieto piny nemajú žiadnu inú funkčnosť a slúžia len pre komunikáciu externého prevodníku, prípadne iného TTL zariadenia. Pin TXD0 je

¹⁵Station mode - mód v ktorom sa ESP pripája k WiFi smerovaču

¹⁶LVM - Lua Virtual Machine, virtuálny stroj ktorý umožňuje interpretáciu programu v jazyku Lua

¹⁷SPI - Serial Peripheral Interface - Sériové periférne rozhranie

¹⁸Access Point - prístupový bod

¹⁹Boot režim - určuje mód štartu mikrokontroléru

²⁰TTL - Transistor Transistor Logic

Tabuľka 1: Popis pinov zariadenia ESP-8266 12s

Pin skratka	Popis
RST	Reset
ADC	Prevodník analógového signálu
EN	Aktivácia procesoru
VCC	Vstupné napájanie kladný pól
GND	Záporný pól
TXD0	Vysielací pin UART komunikácie
RXD0	Prijímací pin UART komunikácie
GPIO0	Vstupne výstupný pin, voľba boot režimu
GPIO2	Vstupne výstupný pin, voľba boot režimu
GPIO4	Vstupne výstupný pin
GPIO5	Vstupne výstupný pin
GPIO12	Vstupne výstupný pin
GPIO13	Vstupne výstupný pin
GPIO14	Vstupne výstupný pin
GPIO15	Vstupne výstupný pin, voľba boot režimu
GPIO16	Vstupne výstupný pin

Tabuľka 2: Manuálna voľba režimu pri štarte ESP-8266 12s

Boot mode	GPIO0	GPIO2	GPIO15
Normálny režim	1	1	0
Flash režim	0	1	0
Načítanie programu z SD karty	0	0	1

tzv. „transmit“ pin, pin RXD0 je tzv „receive“ pin. SDK, ktorú vyvinula spoločnosť Espressif, umožňuje tvorbu ďalšieho TTL rozhrania prostredníctvom dvoch z dostupných GPIO pinov.

Samotné zariadenie je možné vidieť na obrázku 11.

3.5.2 NodeMCU

Spoločnosť NodeMCU vyvinula firmvér rovnakého názvu, špeciálne pre čip ESP8266EX. Firmvér využíva SDK dodanú spoločnosťou Espressif. Tento firmvér, ako už bolo spomenuté, obsahuje LVM pre odnož jazyka Lua - eLua.

Projekt eLua, je špeciálne vyvinutý pre programovanie mikrokontrolérov. Jedná sa o open-source projekt s názvom Embedded Lua, ktorý ponúka plnú integráciu jazyku Lua pre tzv. embedded systémy. Svojimi vlastnosťami priniesol vývojom

jednoduché prototypovanie aplikácií, RAD²¹ a mnoho ďalších výhod ako napríklad tvorbu tzv. „event driven“ programov. Projekt vznikol v roku 2005 a do dnes sa stále rozvíja aj vďaka komunite z celého sveta. eLua plne podporuje implementáciu jazyka Lua vo verzii 5.1. (eLua, 2011)

Spoločnosť NodeMCU vyvinula firmvér, ktorý zahŕňa SDK dodávanú spoločnosťou Espressif a umožňuje tvorbu aplikácií v jazyku Lua, pre verziu 12 z rodiny ESP-8266. SDK spoločnosti Espressif je sada funkcií a podprogramov vytvorených v jazyku C. Tento vývojarský kit je voľne k dispozícii k stiahnutiu, ale len v binárnej podobe, nie je teda možné do vývoja SDK a do konkrétnej implementácie rutín nijak zasahovať. (NodeMCU, 2017a)

Lua firmvér tvorí rozhranie medzi užívateľským programom a SDK, avšak niektoré štandardné Lua moduly museli byť nahradené vlastnou implementáciou, jedná sa napríklad o moduly „io“ a „os“, ktoré boli nahradené modulmi „file“ a „node“. Niektoré Lua moduly sa do firmvéru nedostali vôbec, nakoľko ich funkcionálna bola nepotrebná, jedná sa napríklad o modul „debug“. Tento modul je možné do firmvéru zakomponovať, pokiaľ sa užívateľ rozhodne zostaviť si firmvér sám. (NodeMCU, 2017a)

Firmvér NodeMCU neprináša len prostredie s virtuálnym Lua strojom, ale aj niekoľko modulov, ktoré užívateľom umožňuje využívať rôzne služby, čítať dáta z rôznych typov senzorov, použitie WiFi, prácu so súbormi a mnoho ďalších. Tieto moduly rovnako ako aj samotné jadro firmvéru sú open-source a disponujú pomerne veľkou komunitou, ktorá sa podieľa na vylepšeniach a oprave chýb. Moduly sú napísané v jazyku C a pri zostavení firmvéru sú súčasťou výslednej binárnej štruktúry, ktorá sa vypaľuje do flash pamäte. (NodeMCU, 2017a)

V minulosti pri výbere modulov do zostavenia, trebalo brať zreteľ na to, že každý modul zakomponovaný vo firmvéri, spotrebuje určitú časť pamäte RAM. Plne zostavený firmvér poskytoval užívateľovi približne 15 kB voľnej pamäte RAM. V novej verzii 2.0.x a vyšších, bol tento problém eliminovaný, vykonávaním funkcií priamo z flash pamäte pokiaľ sa daný kód nenachádza v procesorovej cache pamäti. L1 cache procesora drží adresy odkazujúce priamo na flash pamäť, odkiaľ je kód pomocou priameho SPI rozhrania odoslaný a vykonaný na procesore. Vykonávanie kódu načítaného z pamäte flash je približne 13x pomalšie, ale takýmto spôsobom bolo dosiahnutých 45 kB voľnej pamäte RAM. Avšak každý modul pridaný do zostavenia firmvéru zaberie určité množstvo pamäte, ktorá sa používa na držanie referenčných adries do pamäte flash. Je teda vhodné, do firmvéru komponovať len potrebné moduly, aby sa predišlo problému s nižšou úrovňou voľnej pamäte RAM. (NodeMCU, 2017b)

Programovanie ESP8266 s firmvérom NodeMCU, môže mať rôzne podoby. Firmvér po úspešnom spustení hľadá súbor „init.lua“, tento súbor spustí príkazom „do file(init.lua)“, následne je interpretovaný užívateľský program. Štruktúrovanie aplikácie je ponechané na potrebách a zvyklostiach programátora. Aplikácia môže

²¹RAD - Rapid Application Development

byť napísaná ako skript, alebo rozdelená do samostatných modulov s lokálnymi alebo globálnymi funkciami. Vo všeobecnosti sa odporúča tvorcami firmvéru, robustnejšie aplikácie rozdeliť do samostatných modulov, čo na oplátku užívateľovi umožní hospodáriť s pamäťou RAM a alokovať ju len vtedy kedy ju naozaj potrebuje. Na rozdiel od modulov firmvéru, užívateľský program nie je možné priamo vykonávať z pamäte flash ale musí byť načítaný v pamäti RAM. Chovanie podobné NodeMCU modulom, je možné docieľiť pomocou techniky nazývanej flash-funkcie, čo ale môže viesť k nepredvídateľnému chovaniu a signifikantnému navýšeniu času vykonávania programu. (NodeMCU, 2017a)

LVM umožňuje pred-kompiláciu „lua“ súborov do „.lc“ štruktúry, ktorá predstavuje predkompilovaný Lua Byte Code, ktorý zaberá v pamäti menej miesta a jeho vykonávanie je o niečo rýchlejšie ako u zdrojových „lua“ súborov. Pri použití funkcie `require` s parametrom „nazov modulu“, sa vyhľadáva najprv „nazov modulu.lc“ súbor a ak súbor neexistuje hľadá sa súbor „nazov modulu.lua“. Nevýhodou použitia pred-kompilovaných zdrojových súborov pri vývoji, je minimalizovanie výstupných informácií pri programovej chybe. (NodeMCU, 2017b)

3.6 MQTT

Komunikačný protokol MQTT²², je protokol pre rýchlu výmenu správ primárne vyvinutý pre M2M komunikáciu. V dnešnej dobe nachádza obrovské využitie vo svete IoT zariadení, ale aj v prostredí mobilných komunikačných aplikácií. Jedná sa o publish-subscribe výmenu správ, kde odosielateľ správy zašle správu do určitej témy a následne všetci adresáti, ktorí su na danej téme prihlásený správu obdržia. (Valerie Lampkin, Leonardo Olivera, 2012)

Protokol MQTT vznikol v roku 1999 a jeho primárne určenie, bolo nahradiť náročnú a nespoľahlivú HTTP komunikáciu v púštnych ropovodoch. Pracuje nad vrstvou TCP/IP protokolu, umožňuje nadstavenie QoS²³ na troch úrovniach - podrobnejší prehľad sa nachádza v tabuľke 3. Nové verzie MQTT protokolu umožňujú šifrovanie správ medzi klientom a MQTT prostredníkom pomocou TLS protokolu. (Valerie Lampkin, Leonardo Olivera, 2012) Z MQTT protokolu vzniklo množstvo príbuzných protokolov, ktoré majú veľmi podobnú funkcionality, jedná sa napríklad o AMQP²⁴, STOMP²⁵ a mnoho iných.

Zaujímavosťou MQTT protokolu je zasielanie tzv. „last will“ správ. Sú to správy, ktoré sa odošlú do vopred zvolených tém po tom, ako MQTT prostredník zaregistruje že sa klient odpojil. Toto je možné vďaka „keep-a-live“ paketom, ktoré si v zvolených intervaloch vymieňajú klient s prostredníkom. Pokiaľ si klient praje zaslať správu poslednej vôle, je potrebné aby ju zaregistroval na MQTT prostredníka, po úspešnom nadviazaní pripojenia. (Valerie Lampkin, Leonardo Olivera, 2012)

²²MQTT - Message Queuing Telemetry Transport

²³QoS - Quality of Service

²⁴AMQP - Advanced Message Queuing Protocol

²⁵STOMP - Streaming Text Oriented Messaging Protocol

Tabuľka 3: MQTT úrovne QoS

QoS	Vysvetlenie
0	Klient odošle správu, správa je doručená najviac jeden krát alebo vôbec.
1	Klient odošle správu, správa je doručená najmenej jeden krát.
2	Klient odošle správu, správa je doručená práve jeden krát.

Architektúra MQTT komunikácie je centralizovaná, čo znamená že každý klient, ktorý chce MQTT protokolom komunikovať sa musí pripojiť na MQTT prostredníka. V praxi je možné podľa náročnosti počet MQTT prostredníkov škálovať a jednotlivé stroje medzi sebou premosťovať, aby sa tvárili ako jeden. (Valerie Lampkin, Leonardo Olivera, 2012)

Od roku 2013, medzinárodná organizácia OASIS²⁶, považuje MQTT protokol za nízko nárokové, open-source riešenie v komunikácii medzi dvoma a viacerými zariadeniami (device to device communication). Práve nízke nároky na oboch stranách komunikantov, sú hlavným dôvodom prečo sa MQTT protokol rýchlo rozšíril v oblasti IoT zariadení ale aj do softvérových spoločností ako je napríklad Amazon, Facebook, Microsoft a ďalších. (OASIS, 2014)

3.6.1 MQTT Broker

Ako už bolo spomenuté pre fungovanie MQTT komunikácie, je nutný centrálny prvok - MQTT broker alebo aj prostredník. Tento prostredník slúži k popisovanej výmene správ. Všetky zariadenia, ktoré chcú v danej sieti komunikovať MQTT protokolom, sa musia na prostredníka prihlásiť. Centrálny prvok môže mať definovaný ACL²⁷ tzv. prístupový zoznam, podľa ktorého umožní alebo neumožní sa zariadeniu pripojiť a komunikovať. Tento prístupový list môže byť nahradený iným pravidlom, napríklad systémom LDAP²⁸. MQTT prostredník, za predpokladu správnych a platných certifikátov, môže poskytnúť TLS šifrovanie správ pri komunikácii s klientom. (Valerie Lampkin, Leonardo Olivera, 2012)

Jeden z najznámejších MQTT prostredníkov je softvér od spoločnosti Mosquitto. Jedná sa o službu určenú pre Linuxové distribúcie. Spoločnosť Mosquitto poskytuje aj jednoduchého MQTT klienta, ktorý umožňuje zobrazovanie existujúcich tém a vymieňaných správ v reálnom čase. Aplikácia umožňuje široké nastavenie prostredníctvom konfiguračných súborov, kde je možné meniť napríklad číslo portu, maximálny počet pripojených klientov, minimálny QoS level, cesty k bezpečnostným certifikátom, spôsob autentizácie a mnoho iných. (Valerie Lampkin, Leonardo Olivera, 2012)

²⁶OASIS - Organization for the Advancement of Structured Information Standards - Organizácia na presadzovanie noriem pre štruktúrované informácie

²⁷ACL - Access Control List

²⁸LDAP - Lightweight Directory Access Protocol

3.7 Použité nástroje

Počas práce na diplomovej práci, bolo potrebné použiť niekoľko rôznych softvérových nástrojov. V tejto sekcii, sa nachádza ich stručný popis a objasnenie ich prínosu do diplomovej práce.

3.7.1 ESP Cut

Softvér ESP Cut je dielom neznámeho autora, ktorý vytvoril IDE²⁹, zamerané priamo pre vývoj softvéru pre ESP-8266. Obsahuje v sebe niekoľko nástrojov, ako syntaktická analýza editovaného kódu, nahrávanie kódu pomocou sériovej komunikácie do pamäte ESP, softvérový reštart ESP a mnoho iných. Veľkou výhodou je že obsahuje sériový monitor, vďaka ktorému je možné priamo v tomto vývojovom prostredí, monitorovať sériovú komunikáciu. Podporuje niekoľko jazykov, medzi hlavné patrí microPython a Lua.

Toto vývojové prostredie umožňuje rýchlejší vývoj a ladenie programu pre ESP mikro-kontroléry.

3.7.2 ESP-Py Flasher

Tento softvérový nástroj slúži k tzv. vypalovaniu firmvéru do pamäte ESP. Jedná sa o pomerne jednoduchú aplikáciu, ktorá spoľahlivo nahrá užívateľom vložený firmvér cez sériovú linku do pamäte ESP. Pred začatím nahrávania firmvéru, je potrebné ESP uviesť do tzv. flash módu. Softvér má voči ostatným podobným programom výhodu v tom, že umožňuje premazať zvyšnú voľnú pamäť. Samozrejmosťou je voľba typu pamäte a komunikačnej rýchlosti.

3.7.3 DipTrace

Softvér DipTrace slúži k návrhu elektrických schém a dosiek plošných spojov. Obsahuje širokú knižnicu bežných elektrických súčiastok ale taktiež nástroje pre tvorbu vlastných návrhov. Návrh dosky plošných spojov, je možné vyexportovať v rôznych formátoch a zadať do výroby bez nutnosti dodatočných úprav.

V diplomovej práci bol softvér DipTrace použitý na návrh schémy zapojenia a aj na návrh dosiek plošných spojov.

3.7.4 MQTT Spy

MQTT Spy je aplikácia, ktorá slúži ako klient MQTT prostredníka a umožňuje sledovanie MQTT komunikácie medzi jednotlivými klientmi a prostredníkom. Navyše je možné v aplikácii zasielať správy do vybraných MQTT tém. Jedná sa o veľmi užitočný nástroj, pri práci s MQTT protokolom. (Kamil Baczkowicz, 2015)

²⁹IDE - Integrated Development Environment

3.7.5 Visual Studio 2017

Vývojový balík Visual Studio 2017 od spoločnosti Microsoft obsahuje sadu nevyhnutných nástrojov pre programovanie v rôznych programovacích jazykoch. V tejto diplomovej práci bol využitý pri vývoji správcovskej aplikácie v programovacom jazyku C# a WPF framework-u.

4 Návrh riešenia

Táto časť diplomovej práce sa zaoberá návrhom a implementáciou vlastného riešenia inteligentnej senzorickej siete administratívnej budovy. Vychádza pri tom z poznatkov získaných v rešeršnej štúdii a zo stanovených požiadavkov na senzorickú sieť.

Na začiatok, je potrebné návrh vlastného riešenia rozdeliť na základné časti, ktoré sa na záver spoja a vytvoria inteligentnú senzorickú sieť. V nasledujúcom výčte sa nachádzajú tri základné časti senzorickej siete.

- Komunikácia v senzorickej sieti.
- Uzol senzorickej siete.
- Adaptér.

Základnou súčasťou senzorickej siete je stanovenie spôsobu a formy komunikácie medzi jednotlivými komunikantmi v senzorickej sieti. Tento základný krok umožní spravovať jednotlivé senzorické uzly a získavať dáta z pripojených senzorov. Jasné vytýčenie pravidiel a spôsobu komunikácie navyše prispeje k implementácii budúcich rozšírení.

Ďalšou časťou senzorickej siete, je senzorický uzol. Jedná sa o zariadenie, ktorého úlohou je pripojiť sa k senzorickej sieti, zbierať dáta z pripojených senzorov a distribuovať ich ďalej po sieťovej infraštruktúre. Navyše je potrebné aby bolo možné takýto senzorický uzol v sieti spravovať pomocou aplikácie na toto určenej.

Posledná zo základných častí senzorickej siete, sa nazýva adaptér. Adaptér je aplikácia, ktorá zbiera dáta od jednotlivých uzlov v senzorickej sieti, spravuje jednotlivé uzly, zobrazuje prehľad prijatých dát a umožňuje aktualizáciu programu na vybraných senzorických uzloch. Zozbierané dáta, je potrebné distribuovať mimo senzorickú sieť, úlohou adaptéru je teda prijaté dátové správy spracovať a ďalej odosielať na REST Api webového serveru kde nasleduje ich dodatočné spracovanie.

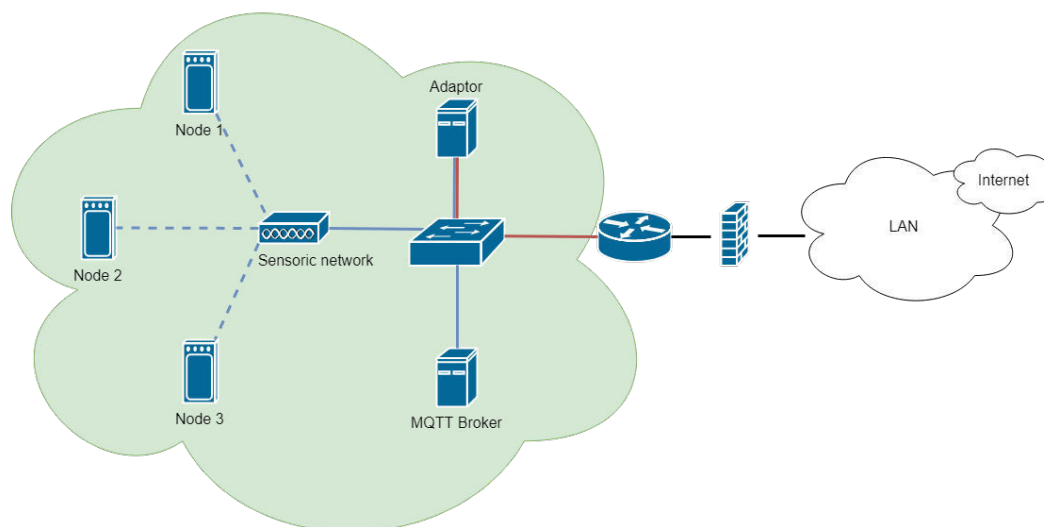
Takto vytvorená senzorická sieť, môže byť odtienená od inej sieťovej komunikácie v administratívnej budove alebo internetu, čo prispeje k zvýšeniu bezpečnosti. Príklad návrhu sieťovej infraštruktúry senzorickej siete sa nachádza na obrázku 12.

V nasledujúcich častiach tejto diplomovej práce je popísaný návrh a implementácia vytýčených častí senzorickej siete. V závere tejto kapitoly, sú v jednotlivých častiach popísané zabezpečenie senzorickej siete a výsledná senzorická sieť ako celok.

4.1 Komunikácia

Ako už bolo spomenuté, samotná komunikácia medzi jednotlivými prvkami senzorickej siete je kľúčová pre jej fungovanie. Je potrebné vhodne zvoliť komunikačný protokol, určiť štruktúru zasielaných správ a rozčleniť jednotlivé typy správ.

Pri výbere komunikačného protokolu, je potrebné brať ohľad na určenie komunikačných protokolov, nakoľko nie každý komunikačný protokol je vhodný na



Obrázok 12: Príklad návrhu sieťovej infraštruktúry senzorickej siete

komunikáciu v senzorickej sieti. Ako už z časti návrhu senzorickej siete tejto diplomovej práce vyplýva, bol zvolený MQTT protokol.

4.1.1 Komunikačný protokol

Jeden z hlavných dôvodov výberu MQTT protokolu ako hlavného komunikačného protokolu v senzorickej sieti, je práve jeho nízka náročnosť na zariadeniach s malým výkonom a rýchlosť prenosu správ, ktorá je ovplyvnená malou veľkosťou MQTT paketov. MQTT protokol umožňuje, oddelenie správ do špecifických tém, ktoré dynamicky vznikajú a zanikajú podľa potreby pripojených klientov.

Oddelenie rôznych typov komunikácie do rôznych MQTT tém, umožňuje poslucháča zaregistrovať len do špecifickej témy a prijímať tak len správy, o ktoré má záujem. Napríklad časť adaptéru, ktorá má za úlohu preposielať dátové správy na REST Api, je zaregistrovaná v dátovej téme a prijíma tak iba dátové správy. Od ostatných typov správ je teda táto časť odtienená a nie je potrebné riešiť logiku selekcie dátových správ.

4.1.2 Štruktúra komunikácie

Komunikáciu v senzorickej sieti je vhodné určitým spôsobom štrukturovať a to nie len formu prenášaných správ, ale aj definovaním vybraných tém do ktorých sa jednotliví klienti registrujú.

Štruktúra prenášaných správ je formát JSON³⁰. Pre správne fungovanie všetkých článkov senzorickej siete, je dôležité aby všetci klienti dodržiavali stanovenú formu a štruktúru vymieňaných správ. Základnou časťou každej správy v senzorickej sieti je unikátny identifikátor senzorického uzlu a typ správy, tieto polia tvoria tzv. hlavičku. Hlavičku môže dopĺňať ešte časové razítko ako nepovinný údaj, nasleduje dátový blok, ktorý, ako už z jeho názvu vyplýva, obsahuje prenášané dáta. Výčet typov správ sa nachádza v tabuľke 5.

Tabuľka 4: Typy správ v senzorickej sieti

Typ správy	Určenie
data	Dátová správa
update	Správa nesúca informáciu o aktualizácii
info	Správa nesúca informácie o stave senzorického uzlu
identify	Správa nesúca informácie pre identifikáciu senzorického uzlu v sieti

Dátová časť dátovej správy smerujúca zo senzorického uzlu na MQTT prostredníka, obsahuje pripojené periférie, v tomto prípade sa jedná o senzory a ich namerané hodnoty. Pokiaľ senzor dokáže zmerať viacero veličín, tak jednotlivé hodnoty sa nachádzajú pod príslušným kľúčom v sekcii daného senzoru. Ukážka dátovej správy, odosielanej senzorickým uzlom sa nachádza v bloku kódu 1. Pokiaľ dátovú správu odosiela adaptér, považuje sa táto správa za požiadavok o zaslanie dát.

Dátová časť aktualizacej správy smerujúca k senzorickému uzlu, obsahuje názov aktualizovaného súboru, číslo riadku zdrojového súboru a samotné dáta načítané zo zdrojového súboru. Odpoveď senzorického uzlu zas obsahuje číslo riadku a názov aktualizovaného súboru.

Dátová časť informačnej správy, obsahuje popis stavu uzlu v jednotlivých poliach. Ip adresa, počet sekúnd od štartu, množstvo voľnej pamäte v bajtoch, stavy jednotlivých senzorov a ďalšie. Na základe týchto informácií, je možné monitorovať stav jednotlivých uzlov v senzorickej sieti. V prípade zaslania informačnej správy adaptérom, je správa považovaná za požiadavok o zaslanie informácií o senzorickom uzle.

Dátová časť identifikačnej správy obsahuje ip adresu a výpis pripojených senzorov na senzorickom uzle. Podobne ako v prípade dátovej a informačnej správy, pokiaľ adaptér zašle identifikačnú správu senzorickému uzlu, jedná sa o požiadavok na identifikáciu.

Ako už bolo naznačené, jednotlivé správy sú zasielané do špecifických tém na MQTT prostredníkovi. Jednotlivý MQTT klienti sú taktiež zaregistrovaný ako poslucháči na špecifických témach MQTT prostredníka. Pri štrukturovaní MQTT ko-

³⁰JSON - JavaScript Object Notation

munikačných tém, je potrebné rozlíšiť senzorické uzly od adaptéru, rozdelenie sa nachádza v nasledujúcich výčtoch.

MQTT témy do ktorých senzorický uzol zasiela správy a adaptér z nich správy prijíma:

- `/data/` - Téma pre zasielanie dátových správ.
- `/update/res/` - Téma pre zasielanie odpovedí na prijaté aktualizčné správy.
- `/maintenance/` - Téma pre zasielanie informačných a identifikačných správ.

MQTT témy do ktorých adaptér zasiela správy a senzorický uzol z nich správy prijíma:

- `/service/` - Téma kde adaptér zasiela hromadné požiadavky.
- `/service/node_id/` - Téma kde adaptér zasiela správy adresované konkrétnemu uzlu s id - `node_id`
- `/update/req/` - Téma kde adaptér zasiela požiadavky na aktualizáciu.
- `/update/node_id/` - Téma kde adaptér zasiela požiadavok na aktualizáciu, adresovaný konkrétnemu uzlu s id - `node_id`

Kód 1: Dátová správa zo senzorického uzlu

```
{
  "id": 5718344,
  "type": "data",
  "data": {
    "sensors": {
      "tsl2561": {
        "status": "OK",
        "ch0": 97,
        "lux": 779,
        "ch1": 32
      },
      "dht11": {
        "humi": 37,
        "temp": 26,
        "status": "OK"
      }
    }
  }
}
```

4.2 Senzorický uzol

Uzol senzorickej siete sa skladá z niekoľkých častí, tieto časti je možné pomenovať nasledovne:

- Centrálna jednotka - mikrokontrolér.
- Program centrálnej jednotky.
- Senzorická výbava.
- Podporné obvody.

Ako už bolo napísané v sekcii návrhu senzorického uzlu, hlavnou výpočtovou jednotkou senzorického uzlu je mikrokontrolér ESP8266 12s.

Aby senzorický uzol splnil stanovené požiadavky, je potrebné implementovať jednotlivé funkcionality a program vhodne štrukturovať.

Senzorický uzol musí mať pripojené vybrané kompatibilné senzory a implementované funkcie pre čítanie dát. Aby bolo možné dáta zbierať a odosielať do senzorickej siete.

Pod pojmom podporné obvody, si je možné predstaviť napríklad obvod regulátoru napätia, filtračné kondenzátory, rôzne pull-up a pull-down rezistory, ktoré slúžia k zabezpečeniu stabilného chodu mikrokontroléru alebo pripojených senzorov.

V jednotlivých pod sekciách sú bližšie poísané časti senzorického uzlu. Nasledujúca sekcia, sa zaoberá práve programom, ktorý beží na zariadeniach ESP-8266 12s a tvorí nevyhnutnú súčasť senzorického uzlu.

4.2.1 Program senzorického uzlu

Ako už bolo spomenuté, firmvér NodeMCU umožňuje tvorbu Lua modulov, ktoré medzi sebou môžu interagovať. Využíva sa na to unikátna dátová štruktúra známa ako „Lua table“. Táto štruktúra môže obsahovať rôzne iné dátové typy, funkcie alebo iné moduly a je použitá naprieč celým navrhnutým programom.

Program je štrukturovaný do niekoľkých modulov, ktorých výčet sa nachádza v nasledujúcom zozname. Vzájomné prepojenie jednotlivých programových modulov je názorne vysvetlené na obrázku 13.

- `init.lua`
- `configuration.lua`
- `network_setup.lua`
- `mqtt_connection.lua`
- `application.lua`
- `updater.lua`

- `maintenance.lua`

Súbor `init.lua`, má vo firmvéri NodeMCU zvláštnu funkciu. Po naboťovaní firmvéru je automaticky spustený práve súbor `init.lua`, ktorý slúži k spusteniu samotnej aplikácie. Ako je možné vidieť v ukážke kódu 2, na začiatku sa iteruje súbormi, ktoré sú v pamäti mikrokontroléru. V tomto cykle sa pred samotným spustením programu rieši, dokončenie prípadného aktualizácného procesu. Pokiaľ sa počas aktualizácie jedného z modulov, niečo pokazilo, súbor ostal pomenovaný s prefixom `update_`. Takýto súbor je nutné odstrániť, môže sa totiž jednať o neúplnú aktualizáciu. Pokiaľ sa v zozname súborov nachádza súbor bez prefixu `update_` ale s príponou `.lua`, jedná sa o aktualizovaný programový modul, ktorý sa skompiluje do `.lc` súboru, pôvodný súbor sa odstráni a zariadenie sa reštartuje.

Po úspešnom dokončení úvodného cyklu, je možné prejsť k samotnému spusteniu programu. V súbore `init.lua`, sa definuje globálna premenná konfiguračného modulu a následne sa spustí pripájanie do WiFi siete z modulu `network_setup.lua`.

Kód 2: Zdrojový súbor `init.lua`

```
for name in pairs(file.list()) do
    if name ~= "init.lua" then
        if string.match(name, 'update_') then
            file.remove(name)
        elseif string.match(name, '.lua') then
            node.compile(name)
            file.remove(name)
            node.restart()
        end
    end
end
config = require('configuration')
require('network_setup').start()
```

Konfiguračný modul `configuration.lua`, obsahuje všetky potrebné informácie na konfiguráciu senzorického uzlu a jeho periférií. Obsahuje údaje k prihláseniu sa do siete WiFi, k prihláseniu sa na MQTT prostredníka, zoznam MQTT tém, pripojených senzorov, sekciu s tzv. „callback“ funkciami a mnoho iných konfiguračných premenných. V kóde 3 je možné vidieť časť konfiguračného súboru. Konfiguračný modul je využívaný naprieč všetkými programovými modulmi, čo umožňuje vykonávať zmeny v konfigurácii na jednom mieste.

Kód 3: Vybraná časť konfiguračného modulu

```
mod.callbacks['info'] = {}
mod.callbacks['info'].func = maintenance.getInfo
mod.callbacks['info'].topic = mod.MQTTPub['maintenance']
mod.sensors = {}
mod.sensors['tsl2561'] = {}
```



```
mod.sensors['tsl2561'].scl = 6
mod.sensors['tsl2561'].sda = 7
mod.sensors['tsl2561'].callback = application.getTsl2561
mod.sensors['tsl2561'].status = 'OK'
```

Modul `network_setup.lua` slúži k pripojeniu do siete WiFi, využíva pri tom WiFi konfiguráciu uvedenú v konfiguračnom súbore. Ako prvý krok, preskenuje dostupné WiFi siete v okolí a pokúsi sa pripojiť na sieť uvedenú v konfigurácii. Po úspešnom pripojení, program kontroluje pridelenie IP adresy. Ako náhle je IP adresa zariadeniu pridelená, spustí sa prihlasovanie na MQTT prostredníka z programového modulu `mqtt_connection.lua`. Modul `network_setup.lua`, sa dokáže pripojiť aj k WiFi sieťam, ktoré majú skryté SSID. Pokiaľ sieť nie je dostupná, alebo sa zariadenie z nejakých dôvodov nemôže pripojiť, čaká stanovenú dobu 3 sekúnd a pokus o pripojenie opakuje.

Modul ktorý zabezpečuje MQTT komunikáciu, nesie názov `mqtt_connection.lua`. V tomto programovom module sa zariadenie pripája na MQTT prostredníka, registruje správu poslednej vôle, definuje callback funkciu ktorá je volaná po prijatí správy a callback funkciu ktorá je volaná po nečakanom odpojení sa od MQTT prostredníka. Pokiaľ zariadenie stratí spojenie s MQTT prostredníkom, snaží sa toto spojenie čo najskôr obnoviť. Ak je strata spojenia spôsobená výpadkom WiFi pripojenia, WiFi pripojenie sa automaticky nadväzuje, po úspešnom pripojení do WiFi siete sa modulu podarí opätovne prihlásiť na MQTT prostredníka. Po každom prihlásení senzorického uzlu na MQTT prostredníka, je potrebné znova registrovať poslucháčov do vybraných MQTT tém, definovaných v hlavnom konfiguračnom súbore. Modul `mqtt_connection.lua`, obsahuje funkciu ktorá slúži k zasielaniu správ na MQTT prostredníka. Implementáciu tejto funkcie je možné vidieť v bloku kódu 4. V danej funkcii sa volá funkcia `encrypt(data)`. Táto funkcia, zašifruje dátový blok odosielanej správy, pomocou algoritmu AES-ECB z doplnkového modulu firmvéru s názvom `crypto`, pokiaľ je tak definované v konfiguračnom súbore. Výstup funkcie `encrypt(data)` je reťazec v JSON štruktúre, využíva sa na to doplnkový modul firmvéru s názvom `sjson`. Následne sa zavolá metóda `mqtt:publish(topic, message, QoS)`, ktorá odošle správu na MQTT prostredníka. Návratová hodnota určuje či bolo odoslanie správy úspešné, pokiaľ odosielanie skončilo neúspechom, je potrebné ošetriť chybu pripojenia a nakonfigurovať MQTT pripojenie znovu.

Kód 4: Funkcia slúžiaca k zasielaniu MQTT správ

```
local function publishMessage(client, topic, data)
  if config.MQTT['connected'] == true then
    local message = encrypt(data)
    local result = client:publish(
      topic, message, config.MQTT['qos'], 0
    )
    tmr.create():alarm(500, tmr.ALARM_SINGLE, function()
```

```

    if result ~= true then
        handle_mqtt_error(client)
    end
    config.printInfo('publishData',data.type, result and 1 or 0)
end)
end
end

```

V programovom module s názvom `application.lua`, sa zbierajú dáta zo senzorov. Tento modul obsahuje implementáciu funkcií, ktoré čítajú hodnoty z pripojených senzorov definovaných v konfiguračnom module. Funkcie aplikačného modulu sú použité v konfiguračnom súbore a sú uložené do jednotlivých premenných senzorov ako callback funkcie. Takýto prístup umožňuje jednoduchú iteráciu jednotlivými pripojenými senzormi z konfiguračného súboru a získaniu nameraných hodnôt, ktoré sú následne odosielané na MQTT prostredníka. Ukážka získavania dát zo senzoru DHT-11, sa nachádza v bloku kódu 5.

Kód 5: Funkcia slúžiaca k získaniu dát zo senzoru DHT-11

```

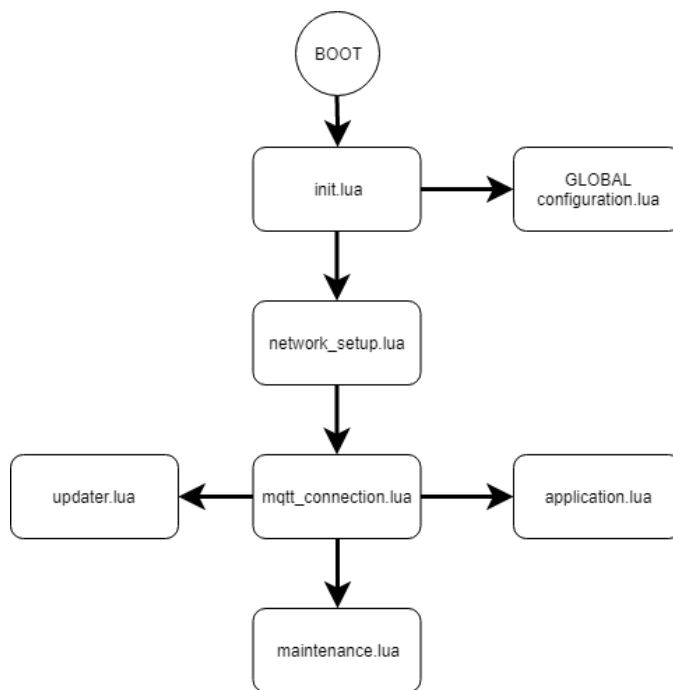
function module.getDht11()
    local err, temp, humi = dht.read(
        config.sensors['dht11'].pin
    )
    local result = {}
    if err == dht.OK then
        result.temp = temp
        result.humi = humi
        result.status = 'OK'
    else
        result.status = 'ERR'
    end
    config.sensors['dht11'].status = result.status
    return result
end

```

Programový modul `updater.lua`, spracováva požiadavky na aktualizáciu jedného z programových modulov. Každý senzorický uzol, ktorý je aktualizovaný, si priebeh aktualizácie spracováva samostatne. Adaptér zašle požiadavok o aktualizáciu, ktorý v dátovom bloku nesie informáciu s názvom a veľkosťou aktualizovaného súboru. Názov aktualizovaného súboru nesie prefix `update_`. Senzorický uzol tento požiadavok prijme a skontroluje či na flash pamäti je dostatok voľného miesta. Pokiaľ sú podmienky splnené, vytvorí v pamäti súbor s daným názvom a súbor uzavrie, následne si vyžiada od adaptéru prvý riadok aktualizovaného zdrojového súboru. Adaptér odpovedá do špecifickej MQTT témy, s dátami a číslom riadku. Uzol prijatú správu spracuje, skontroluje, či sa jedná o vyžiadané číslo riadku, otvorí súbor, zapíše prijaté dáta, súbor uzavrie a vyžiada si ďalšie dáta od adaptéru.

Pokiaľ si uzol vyžiada od adaptéru číslo riadku, ktoré sa už v súbore nenachádza - adaptér zasiela uzlu správu o ukončení aktualizácie. Senzorický uzol, následne odstráni prefix `update_` z názvu súboru a reštartuje sa. Po nabootovaní sa spustí rutina definovaná v súbore `init.lua`, aktualizovaný súbor sa predkompiluje do Lua bytecode a aktualizácia sa dokončí.

Modul s názvom `maintenance.lua`, má za úlohu vytvárať správy popisujúce stav daného senzorického uzlu a umožniť vzdialené pokyny k správe senzorického uzlu. Tento programový modul zozbiera následovné informácie: pridelenú IP adresu, množstvo voľnej pamäte v bajtoch, počet pokusov o znovu pripojenie na MQTT prostredníka, stavy jednotlivých senzorov, indikátor povoleného šifrovania a čas v sekundách od štartu. Modul `maintenance.lua`, reaguje na správy typu `command`, tieto správy v dátovom bloku nesú informáciu o type vykonávaného príkazu. Môže sa jednať o reštartovanie modulu, alebo o pokusné volanie garbage collectoru. Pokiaľ senzorický uzol prijíme správu ktorá obsahuje pokyn k reštartu, zavolá sa funkcia z NodeMCU core modulu - `node.reboot()`.



Obrázok 13: Vzájomné prepojenie programových modulov

Výsledné predkompilované programové moduly, majú po spustení a inicializácii spojenia s MQTT prostredníkom, k dispozícii približne 16.5 kB pamäte RAM. Množstvo voľnej pamäte po štarte a počas behu programu sa môže líšiť. Závisí od počtu a typu pripojených senzorov na senzorickom uzle, ale aj od spoľahlivosti WiFi pripojenia a stability MQTT pripojenia. SDK dodávaná spoločnosťou Espressif ako aj samotný firmvér, môže obsahovať nedostatky, ktoré sa môžu prejaviť na stabilite

alebo únikoch pamäte. Známym problémom úniku pamäte o veľkosti približne 112B je znovu nadväzovanie pripojenia na MQTT prostredníka. Táto situácia nastáva vtedy, keď je naviazané TCP pripojenie na server MQTT prostredníka, ale MQTT služba neodpovedá. Po ukončení TCP pripojenia a spustení „garbage collector-u“ nie je pamäť senzorického uzlu opätovne uvoľnená. Ďalším nedostatkom je MQTT modul firmvéru spoločnosti NodeMCU, nastavenie „keep-alive“ paketov pri inicializácii pripojenia sa neprejaví na reálnom chovaní. Táto hodnota je napevno nastavená na 60 sekúnd, čo môže spôsobiť nespoľahlivé volanie udalosti o nečakanom ukončení MQTT pripojenia. Na tieto problémy som poukázal ako na fóre esp8266.com, tak aj v oficiálnom repozitári firmvéru NodeMCU. V závislosti od závažnosti jednotlivých nedostatkov, by mali byť v novej verzii firmvéru opravené, čo prispeje k vyššej stabilite MQTT pripojenia.

4.2.2 Senzorická výbava

Senzorický uzol je vybavený sadou senzorov, ktoré merajú vybrané veličiny. Za základné merané veličiny boli zvolené nasledujúce:

- Teplota.
- Vlhkosť.
- Svetelnosť.

V časti návrhu senzorickej siete tejto diplomovej práce, bolo vybraných niekoľko senzorov, ktoré dané veličiny dokážu merať. Jedná sa o senzory teploty a vlhkosti DHT-11, Si-7021 a senzory úrovně svetelnosti TCS-34725 a TSL-2561. Pre tieto senzory existujú pre firmvér NodeMCU moduly, ktoré dokážu dáta zo senzorov prečítať. Každý zo senzorov má iné charakteristiky, svojimi rozsahmi však odpovedajú danému spôsobu použitia a nasadenia do prostredia administratívnej budovy.

Cenovo najdostupnejší senzor je DHT-11, tento senzor nie je moc presný ale môže byť nahradený presnejším a stále cenovo dostupným senzorom DHT-21 alebo DHT-22. Toto je možné vďaka univerzálnej implementácii NodeMCU DHT modulu, ktorého metóda `dht.read(pin)` je použitá vo funkcii programového modulu `application.lua` v metóde `getDht11()`, viditeľné tiež v bloku kódu 5.

Zvyšok použitých senzorov, je oveľa precíznejších a komunikujú pomocou i2c protokolu. Pre správne fungovanie i2c zbernice, je potrebné zbernicu inicializovať na zdaných GPIO pinoch zariadenia ESP-8266. Následne je možné zahájiť čítanie hodnôt zo senzoru. I2c protokol, umožňuje pripojenie troch zariadení na jednej i2c zbernici, avšak NodeMCU firmvér, respektíve NodeMCU moduly k vybraným i2c senzorom majú občas problém správne prečítať dáta, pokiaľ sa na zbernici nachádza viac ako jeden senzor. Z toho dôvodu je vždy na i2c zbernici pripojený najviac jeden senzor.

Senzor TCS-34725 sa odlišuje implementáciou NodeMCU modulu s názvom `tcs34725` od ostatných senzorických NodeMCU modulov. Dáta zo senzoru sa získa-

vajú pomocou asynchrónnej funkcie ktorej vykonávanie trvá pár stoviek milisekúnd. Toto chovanie má svoje opodstatnenie, tým že sa jedná o senzor RGB spektra s prídavným LED osvetlením, je potrebné po zapnutí LED diódy počkať, aby senzor nebol saturovaný vlastným zdrojom svetla. Takéto chovanie sa odlišuje od zvyšných senzorov, z ktorých sa dáta získavajú synchronne a preto sú hodnoty zo senzoru TCS-34725 oneskorené presne o interval jedného merania.

Do budúca je možné senzorické uzly vybaviť inými senzormi, merajúcimi napríklad atmosferický tlak, pohyb alebo senzorom úrovne oxidu uhličitého v ovzduší - podobne ako v jednom z modulov spoločnosti BigClown. Ponuka senzorov a využitie takýchto nazbieraných dát je široké.

4.2.3 Prototypovanie

Pred samotným vývojom dosiek plošných spojov, bolo potrebné vytvoriť schému zapojenia a zapojenie prototypovať. Po vytvorení schémy zapojenia, bolo možné prejsť k prototypovaniu. Ako prvý krok prototypovania bolo zapojenie na tzv. „bread-board“ doske. Na toto zapojenie bol zakúpený regulátor do vývojových dosiek s výstupom 3.3 V a maximálnym prúdovým odberom 1 A. Po tom ako sa odladili chyby zapojenia, sa postúpilo k výrobe prototypu na univerzálnu dosku plošných spojov, kde bolo použité overené zapojenie regulátoru napätia s lineárnym regulátorom AMS-1117-3.3, ktorého výstup je práve 3.3V a maximálny odber prúdu by nemal presiahnuť 900mA.

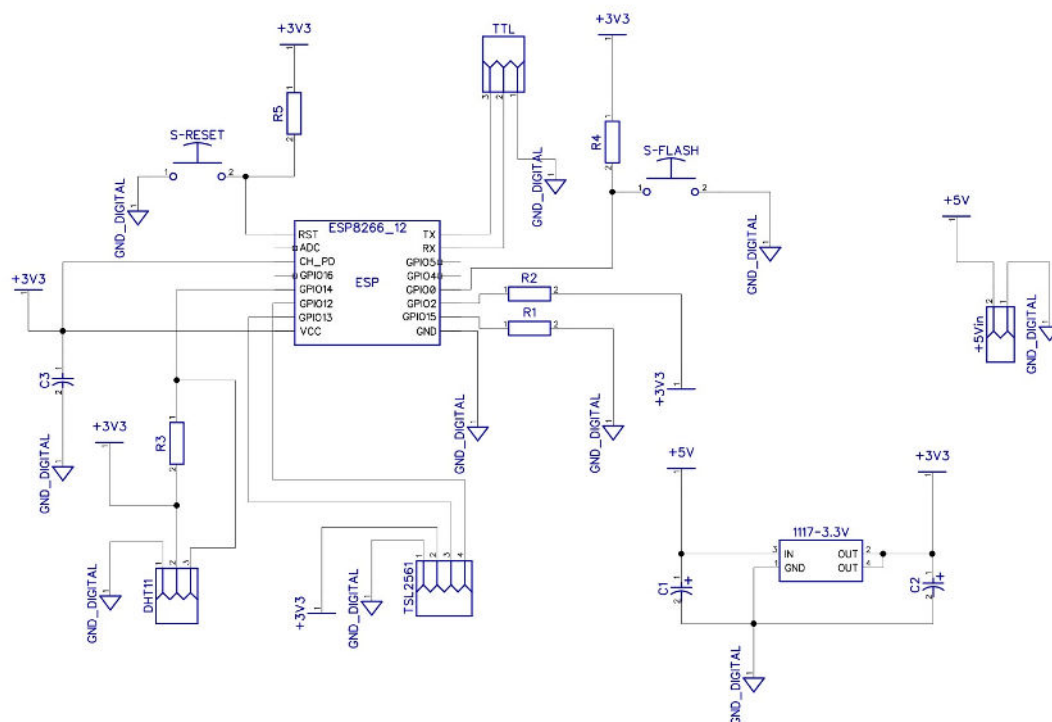
Verzia prototypu na univerzálnnej doske plošných spojov, bola podrobená dlhodobej skúške funkčnosti, kde sa skúmalo či sa neprehrieva regulátor napätia alebo samotné zariadenie ESP-8266. Regulátor vykazoval výstupné napätie približne 3.37V pri vstupnom napätí 5V a neprehrieval sa. Prototyp vykazoval stabilný chod, bez výrazných nedostatkov a bolo možné prejsť k návrhu dosiek plošných spojov.

Schému zapojenia, ktorá sa premietla do návrhu dosky plošných spojov, je možné vidieť na obrázku 14. Schéma zapojenia, bola vytvorená v programe DipTrace - Schematic.

4.2.4 Návrh dosiek plošných spojov

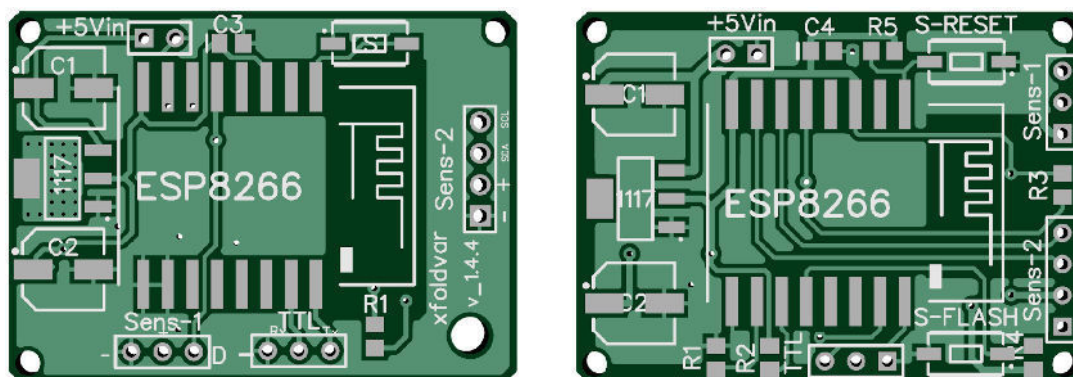
Z navrhnutého prototypu a schémy zapojenia, boli vytvorené návrhy dosiek plošných spojov v programe DipTrace PCB. Pri návrhu dosiek bol bratý ohľad na čo najmenšie rozmery a náklady na výrobu.

Prvotná schéma zapojenia, neobsahovala potrebné pull-up a pull-down rezistory, ktoré slúžia k stabilizácii chodu mikrokontroléru. Absencia spomínaných rezistorov sa na prvotných verziách vyrobených dosiek, prejavovala neočakávaným reštartovaním sa, pri zakolísaní vstupného napätia regulátoru. Navyše prvotná verzia vyrobených dosiek, neobsahovala možnosť prepnutia boot režimu mikrokontroléru do tzv. flash módu, kedy zariadenie očakáva nahratie nového firmvéru. Tieto nedostatky boli odstránené v novej verzii schémy zapojenia a v novej verzii navrhnutých dosiek plošných spojov.



Obrázok 14: Výsledna schéma zapojenia senzorického modulu

Modely navrhnutých dosiek plošných spojov, je možné vidieť na obrázku 15. Finálna verzia modelu, je viditeľná na obrázku 15b a jej predchodca na obrázku 15a.



(a) Predchodca finálnej verzie

(b) Finálna verzia dosky senzorického uzlu

Obrázok 15: Ukážka prvej a finálnej verzie dosiek plošných spojov

Obe verzie majú totožné rozmery a to 3x4cm a obsahujú otvory pre upevnenie do boxu. Dosku je možné vybaviť buď terminálovými konektormi, alebo časťami

hrebeňovej lišty s rozstupmi pinov 2,54mm. Na oboch modeloch je možné vidieť výrez medenej plochy v oblasti antény mikrokontroléru, čo môže prispieť k lepšej intenzite WiFi signálu.

Výsledné dosky plošných spojov disponujú dvoma konektormi, určenými pre pripojenie senzorov. Prvý troj-pinový konektor, obsahuje jeden digitálny pin privedený na mikrokontrolér cez pull-up rezistor a dva napájacie piny s výstupným doskovým napätím 3.3V. Druhý štvor-pinový konektor obsahuje dva digitálne piny privedené na mikrokontrolér a dva napájacie piny. Štvorpinový konektor môže slúžiť k pripojeniu i2c senzorov.

Na oboch doskách plošných spojov sú použité SMD súčiastky, s cieľom minimalizovať potrebnú plochu dosky plošného spoja. Výčet použitých súčiastok a ich funkcií sa nachádza v nasledujúcej tabuľke.

Tabuľka 5: Prehľad použitých súčiastok

Súčiastka	Hodnota	Určenie
AMS1117-3.3	$V_{out} = 3.3V$	Lineárny regulátor napätia
Kondenzátor C1 a C2	$47\mu F 16V$	Kondenzátory obvodu regulátoru napätia
Kondenzátor C3	$100nF$	Filtrovaný kondenzátor napájania ESP
Rezistor R1	$10k\Omega$	Pull-down rezistor voľba boot režimu
Rezistor R2	$10k\Omega$	Pull-up rezistor voľba boot režimu
Rezistor R3	$10k\Omega$	Pull-up rezistor dátová linka k senzoru
Rezistor R4	$10k\Omega$	Pull-up rezistor voľba flash režimu
Rezistor R5	$10k\Omega$	Pull-up rezistor na reset pin ESP
Mikrospínač S-RESET	-	Manuálne reštartovanie ESP
Mikrospínač S-FLASH	-	Voľba flash režimu pri štarte ESP

4.2.5 Výsledný senzorický uzol

Vyhotovený senzorický uzol má stanovené vstupné napätie na 5V. Hodnota vstupného napätia bola zvolená kvôli dostupnosti 5V adaptérov a malej tepelnej strate na lineárnom regulátore. Samotný senzorický uzol odoberá 70-130mA pri 5V. Jednoduchým výpočtom s konštantami získanými z technickej špecifikácie regulátoru AMS1117-3.3, bolo zistené že teplotný nárast voči okoliu je približne $9.35^{\circ}C$. V závislosti od pripojených senzorov sa táto hodnota môže mierne líšiť, napríklad senzor svetelnej intenzity, TSL-2561, podľa technickej špecifikácie, zvýši odber senzorického uzlu počas merania dát o 0.5mA a v kludovom stave o 15uA.

Po osadení dosiek jednotlivými súčiastkami, je potrebné nahráť NodeMCU firmvér s potrebnými modulmi. V senzorických uzloch je nahratá master verzia firmvéru 2.2.0 s Espressif SDK verziou 2.2.1. Do firmvéru boli zahrnuté nasledujúce NodeMCU moduly:

- crypto - šifrovací modul
- dht - čítanie dát z DHT senzorov
- file - manipulácia so súbormi na flash pamäti
- i2c - komunikácia so senzormi prostredníctvom i2c protokolu
- mqtt - MQTT klient
- node - jadro NodeMCU firmvéru
- si7021 - čítanie dát zo senzoru Si-7021
- sjson - modul pre serializáciu a deserializáciu dát
- tcs34725 - čítanie dát zo senzoru TCS-34725
- tmr - modul slúžiaci k tvorbe časovačov
- tsl2561 - čítanie dát zo senzoru TSL-2561
- wifi - obsluha WiFi modulu

Po úspešnom zostavení binárneho súboru sa takto zostavený firmvér napáli do pamäte ESP-8266 pomocou programu ESP Py flasher, zvyšok pamäte sa preformátuje a následne je možné nahrať programové moduly.

Výsledkom je funkčný senzorický uzol, ktorý sa dokáže pripojiť do senzorickej siete, odosielať zozbierané dáta a navyše je možné ho spravovať a aktualizovať prostredníctvom aplikácie nazývanej adaptér. Táto aplikácia je popísaná v nasledujúcej časti tejto práce.

4.3 Adaptér

Správčovskú aplikáciu nazývanú adaptér, je možné vizuálne štrukturovať do štyroch základných častí, podľa grafického užívateľského rozhrania:

- Update - časť pre aktualizáciu senzorických uzlov.
- Data - časť pre zobrazenie prehľadu prijatých dátových správ.
- Maintenance - časť pre správu senzorických uzlov.
- About - časť, kde je popísaná aplikácia.

Adaptér je aplikácia implementovaná v programovacom jazyku C# s využitím WPF framework-u a ďalších podporných knižníc, ako napríklad M2MQTT, Newtonsoft JSON, SQLite a ďalšie. V nasledujúcich častiach, je podrobne vysvetlená architektúra aplikácie a jej jednotlivých častí.

4.3.1 Architektúra aplikácie

Program aplikácie sa skladá z niekoľkých častí. Prvá časť je grafické užívateľské rozhranie, s ktorým pracuje užívateľ aplikácie.

Grafické užívateľské rozhranie sa skladá z tzv. „UserControl“, medzi ktorými sa prepína pomocou horného menu. Jedná sa o grafické komponenty, ktoré predstavujú štyri základné časti aplikácie. Tieto grafické komponenty na svoje fungovanie, využívajú jadro aplikácie.

Jadro aplikácie sa skladá zo skupiny tried, ktoré je možné rozdeliť následovne.

- Modely prijatých a zasielaných správ.
- Databázové modely.
- Databázové služby.
- Komunikačné služby.
- Notifikačné služby.

Modely prijatých a zasielaných správ sú objekty, ktoré sú využívané naprieč celou aplikáciou. Prijatá správa v JSON štruktúre, sa deserializuje na príslušný model, podľa v správe definovaného typu. Naopak, zasielaná správa, ktorú predstavuje model zasielanej správy sa serializuje do JSON štruktúry a následne sa odosiela prostredníctvom MQTT komunikačnej služby do senzorickej siete. Každý model správy obsahuje dynamické dátové pole, čo umožňuje zasielať a prijímať v dátovom bloku ľubovoľné dáta. Toto chovanie je vhodné pre budúce roširoenia, nakoľko pevne stanovená štruktúra, by pri rozšíreniach, vyžadovala dodatočnú implementáciu nových modelových správ.

O samotné zasielanie a prijímanie správ sa stará MQTT služba - trieda s názvom `Mqtt.cs`, ktorá mimo iné vytvára pripojenie na MQTT prostredníka, registruje poslucháčov do špecifických MQTT tém a distribuuje udalosti prijatých správ k ostatným registrovaným poslucháčom udalostí, na základe typu prijatej správy. V tomto objekte, sa prijatá správa deserializuje a v prípade že dátový blok obsahuje šifrované dáta, tak sa tieto dáta dešifrujú a namapujú na príslušný model. Na dešifrovanie prijatej správy, je použitá trieda `Crypto.cs`, ktorá dokáže správy šifrovať a dešifrovať algoritmom AES-ECB s príslušným kľúčom.

Trieda, ktorá slúži k spracováaniu prijatých dátových správ sa nazýva `DataGrabber.cs`. Tento objekt reaguje na udalosť prijatej dátovej správy, registrovaný v triede `Mqtt.cs`. Dátová správa je ukladaná do lokálnej databázy a môže byť preposielaná na REST Api webového serveru, prostredníctvom triedy `Rest.cs`.

O ukladanie prijatých správ sa stará niekoľko databázových služieb, ktoré ukladajú prijaté dáta do SQLite databázy vytvorenej triedou `Connector.cs`. Táto trieda obsahuje niekoľko metód, ktoré sa používajú k tvorbe databázových tabuliek, spúšťaniu SQL príkazov na vkladanie dát alebo vyberaniu dát z daných tabuliek. Túto triedu využívajú príslušné databázové služby, pracujúce s vlastnými tabulkami, ako

napríklad trieda `UnifiedData.cs` alebo `UnifiedInfo.cs`. Tieto triedy, ako už z ich názvu vyplýva, ukladajú prijaté dátové a informačné správy do špecifických tabuliek databáze. Navyše tieto databázové služby, sprostredkujú vzorky najnovších zozbieraných dát, pre vizualizáciu prostredníctvom grafov.

O zasielanie notifikácií sa stará trieda `Notifier.cs`, ktorá sleduje dáta ukladané do databázovej tabuľky informačných správ. Pokiaľ správa obsahuje chybový kód a notifikácia ešte nebola zaslaná, alebo je staršia ako čas stanovený v nastaveniach programu, vytvorí sa e-mail ktorý je následne zasielaný administrátorovi siete. Vygenerovaný e-mail, obsahuje príslušné prijaté dáta a čas udalosti. Pokiaľ došlo v rozmedzí stanoveného intervalu k náprave problému a senzorický uzol nezasla chybové stavy naďalej, adaptér ďalšiu notifikáciu nezašle. Ďalším typom notifikácie, je notifikácia informujúca administrátora senzorickej siete o odpojení daného senzorického uzlu. Ak sa určitý senzorický uzol odpojil, napríklad z dôvodu zlyhania napájania, administrátor je o tejto skutočnosti informovaný vďaka MQTT správe poslednej vôle, registrovanej na MQTT prostredníkovi.

Trieda `Updater.cs`, sa stará o aktualizáciu programových modulov na senzorických uzloch. Využíva pri tom triedu s názvom `FileProcessor.cs`, ktorý po riadkoch prečíta zdrojový súbor a poskytne ho aktualizáčnej službe ako zdroj dát pre aktualizáčný proces. Pre komunikáciu je štandardne využitá trieda `Mqtt.cs`.

Veľmi dôležitou triedou v celej aplikácii, je trieda s názvom `Engine.cs`. Jedná sa o hlavnú triedu, v ktorej sú vytvárané a spúšťané všetky potrebné instance tried pre beh aplikácie. Engine má za úlohu vytvoriť MQTT pripojenie na prostredníka, registráciu do MQTT tém, inicializáciu notifikačného procesu, inicializáciu zberačov prijatých dát a iných.

V nasledujúcich častiach sú popísané kľúčové časti aplikácie adaptéru.

4.3.2 Zber dát zo senzorickej siete

Dátové správy produkované senzorickými uzlami, sú spracovávané adaptérom. Po prijatí dátovej správy adaptérom, sa táto správa rozbalí a namapuje na príslušný dátový model prijatej správy. Pokiaľ správa neobsahuje časové razítka, tak sa vygeneruje a vloží do dátového modelu. Toto chovanie bolo implementované z dôvodu chýbajúcej implementácie NTP protokolu na mikrokontroléri ESP-8266, pričom bolo myslené na budúce rozšírenia, v ktorých by časové razítka mohli generovať príslušné senzorické uzly samé.

Prijatá dátová správa, obohatená o časové razítka, sa serializuje do JSON formátu a odosiela sa na REST Api, pokiaľ je táto možnosť zvolená v nastaveniach adaptéru. Následne sa prijatá dátová správa namapuje na databázový model a je uložená do lokálnej databáze.

Ako už bolo spomenuté v sekcii architektúry aplikácie, o túto činnosť sa stará trieda s názvom `DataGrabber.cs`. `DataGrabber` využíva triedu `Rest.cs`, ktorá slúži k zasielaniu dát na REST Api. Táto trieda implementuje niekoľko funkcií, ktoré slúžia k tvorbe HTTP požiadavku, pridaniu dát v JSON formáte, správne na-

staveniu hlavičiek a k samotnému odoslaniu požiadavku. Požiadavky sú odosielané asynchrónne, nakoľko nie je nutné čakať na odpoveď webového serveru.

Vo vizuálnej dátovej časti adaptéru, je možné v reálnom čase sledovať prijaté dátové správy. Táto sekcia okrem prehľadu prijatých dátových správ, poskytuje aj ich vizualizáciu na troch samostatných grafoch. Jedná sa o grafy teploty, vlhkosti a svetelnej intenzity. Sieťový administrátor tak má možnosť priameho prehľadu o tom, ktorý senzorický uzol vykazuje aké hodnoty. Dátovú časť adaptéru, je možné vidieť na obrázku 21.

4.3.3 Vzdialená aktualizácia programu

Sekcia vzdialenej aktualizácie programu na senzorických uzloch, využíva niekoľko tried z celej aplikácie. Jedná sa napríklad o model aktualizácie správy, triedu zabezpečujúcu MQTT komunikáciu, triedu určenú k identifikácii aktívnych senzorických uzlov s názvom `Identifier.cs` a triedu určenú k spracováaniu zdrojového súboru ktorý bude aktualizovaný - `FileProcessor.cs`.

Trieda `FileProcessor.cs`, pracuje so zdrojovým súborom predstavujúcim aktualizáciu senzorických uzlov. Výstupom tejto triedy, je dátová štruktúra nazývaná list, ktorá obsahuje jednotlivé riadky zdrojového súboru. Každý riadok, predtým než je pridaný do výstupného listu, prechádza funkciou, ktorá z neho odstraňuje nepotrebné tabulátory a viacnásobné medzery nahradzuje za medzery jednotlivé. Týmto je zaručené že prenášané dáta neobsahujú zbytočné znaky, vložené vývojárom, počas štruktúrovania kódu v programovom editore.

Samotná aktualizácia začína inicializačnou správou zo strany adaptéru, ktorá obsahuje v dátovom bloku názov a veľkosť prenášaného súboru. Senzorický uzol túto správu obdrží, vykoná potrebné operácie a vyžiada si od adaptéru prvý riadok zdrojového súboru. Adaptér prijme túto správu a v dátach poskytnutých objektom triedy `FileProcessor.cs`, hľadá vyžiadaný riadok. Po nájdení zašle túto správu senzorickému uzlu spolu s číslom vyžiadaného riadku zdrojového súboru a názvu tohoto súboru. Senzorický uzol skontroluje obdržané dáta a pokiaľ sú v poriadku, zapíše ich do vytvoreného súboru. Koniec aktualizáčného procesu, je vyvolaný siahnutím na posledný prvok z listu, dodaného triedou `FileProcessor.cs`, tento riadok obsahuje unikátny reťazec ktorý po prijatí senzorickým uzlom, spustí proces dokončenia aktualizácie.

Priebeh aktualizácie je možné sledovať v aktualizáčnej sekcii adaptéru, kde má každý senzorický uzol svoj vlastný ukazateľ priebehu. Navyše, adaptér dokáže zdetekovať pripojené senzory na jednotlivých senzorických uzloch, čo umožňuje nahrávať rôzne verzie aktualizácií, medzi jednotlivé senzorické uzly. Samozrejmosťou je aktualizácia všetkých alebo len vybraných senzorických uzlov.

Aktualizácia senzorických uzlov, môže prebiehať paralelne alebo sériovo. Túto voľbu je možné zvoliť v nastaveniach adaptéru.

4.3.4 Správa a monitorovanie pripojených senzorických uzlov

Sekcia správy senzorických uzlov, využíva triedu s názvom `InfoGrabber.cs`, určenú k zberu informačných správ zo senzorickej siete. Táto trieda slúži k spracovaniu informačnej správy, premapovaniu na príslušný databázový model a následné uloženie do databáze. Pri ukladaní modelu do databáze sa kontroluje či nejaký zo senzorov pripojených k senzorickému uzlu, nevykazuje chybový kód. Pokiaľ sa v prijatej správe chybový kód vyskytuje, model sa posunie službe s názvom `MailNotifications.cs`, ktorá túto správu ďalej analyzuje a spracováva.

Správa a monitorovanie senzorických uzlov spadá do vizuálnej sekcie adaptéru s názvom `Maintenance`. V tejto časti môže administrátor senzorickej siete, získavať a prezerat prijaté informačné správy. Navyše je možné v tejto sekcii vynútiť vzdialený reštart vybraného senzorického uzlu.

Prijaté informačné správy sú vizualizované na dvoch samostatných grafoch. Jedná sa o grafy zobrazujúce dva základné údaje a to množstvo voľnej pamäte RAM a čas od štartu senzorického uzlu. Tieto grafy sa veľmi osvedčili pri sledovaní stability senzorických uzlov.

4.3.5 Notifikácie

Pri vytváraní instance objektu triedy `Engine.cs`, vzniká mimo iných, aj instance triedy `Notifier.cs`. Táto trieda má za úlohu sledovať chyby vykazované jednotlivými senzorickými uzlami a informovať o nich administrátora senzorickej siete prostredníctvom emailovej správy.

Odosielanie emailov sa spúšťa v pravidelných intervaloch ako asynchrónna úloha, pri ktorej sa kontroluje či pribudol v databázi záznam o chybovej udalosti. Pokiaľ daný záznam existuje a administrátor o tejto skutočnosti doposiaľ nebol informovaný, odošle sa e-mail a záznam s chybou v databázi, sa označí za úspešne notifikovaný. Chybové záznamy sa do databáze chýb ukladajú v intervale, ktorý je stanovený v nastaveniach adaptéru.

Trieda `Notifier.cs` navyše obsahuje metódu, ktorá slúži k vygenerovaniu a zaslaní e-mailu, v prípade že niektorý zo senzorických uzlov, bol nečakane odpojený. Táto metóda je volaná, pokiaľ adaptér obdrží správu poslednej vôle od MQTT prostredníka. Takáto správa obsahuje štandardnú hlavičku a informáciu o odpojení senzorického uzlu.

4.3.6 Nastavenia aplikácie

Adaptér na svoje fungovanie potrebuje sadu nastavení. Jedná sa prevažne o užívateľom zadané konfigurácie, ktoré ovplyvňujú chovanie aplikácie. Prvotné spustenie aplikácie obsahuje východzie hodnoty, ktoré je podľa potreby možné zmeniť. Jedná sa napríklad o IP adresu MQTT prostredníka, adresu pre REST Api, MQTT témy a iné.

Sekcia nastavení obsahuje aj voľbu paralelnej aktualizácie zvolených senzorických uzlov. Pokiaľ táto voľba nie je povolená, aktualizácia senzorických uzlov prebieha sériovo.

Po zmene a následnom uložení nastavení, sa vygeneruje XML súbor, ktorý obsahuje zmenené nastavenia. Pri opätovnom štarte aplikácie sú východzie nastavenia preťažené užívateľskými úpravami. Pre dokonalé prejavenie upravených nastavení, sa odporúča aplikáciu adaptéru reštartovať.

4.3.7 Lokálne úložisko dát

Ako lokálne úložisko dát je použitá SQLite databáza. Túto databázu obsluhuje trieda s názvom `Connector.cs`, ako už z názvu vyplýva, sprostredkúva pripojenie do tejto lokálnej databáze. Instanciu tejto triedy, využívajú všetky ostatné databázové služby v aplikácii. Obsahuje implementáciu základných operácií ako napríklad spustenie SQL dotazu s očakávaným výstupom, spustenie dotazu bez očakávaného výstupu, navrátenie posledného vloženého ID a iných.

Databázové služby, ktoré využívajú instanciu triedy `Connector.cs`, pracujú s databázovými modelmi. Každá služba má definovanú štruktúru tabuľky, na ktoré sú jednotlivé modely mapované. Aplikácia obsahuje tri základné databázové služby, uvedené v nasledujúcom výčte.

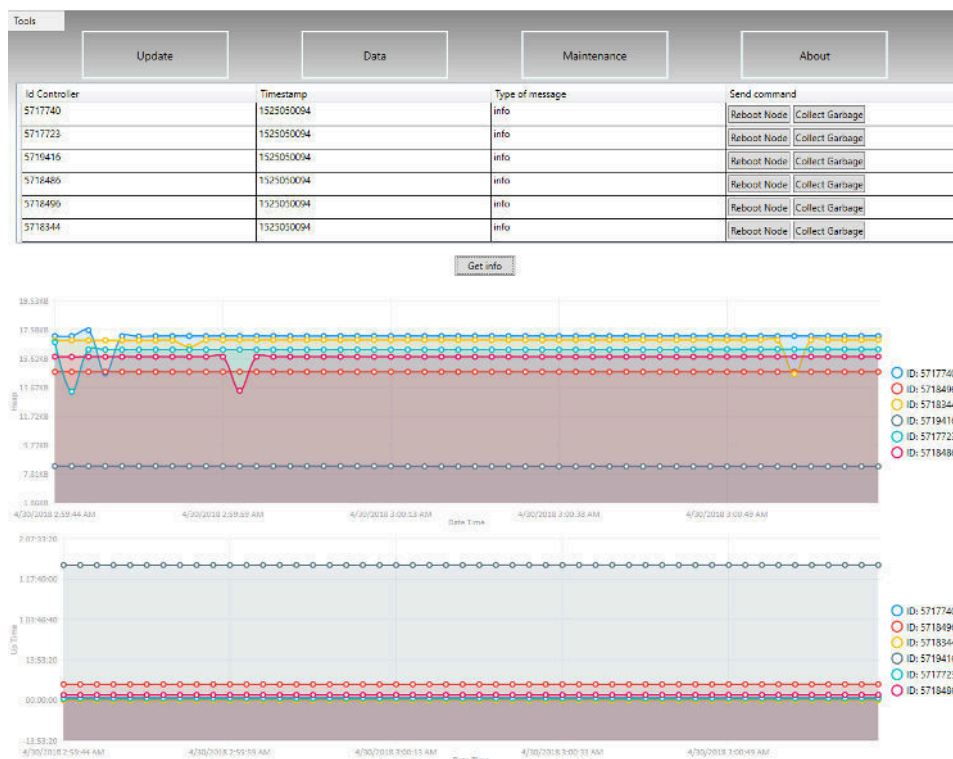
- `UnifiedData.cs` - ukladá a poskytuje prijaté dátové správy.
- `UnifiedInfo.cs` - ukladá a poskytuje prijaté informačné správy.
- `MailNotifications.cs` - ukladá a poskytuje informácie pre zasielanie notifikácií.

Služby `UnifiedData.cs` a `UnifiedInfo.cs`, ukladajú dátový blok prijatej správy vo formáte JSON do SQLite premennej typu TEXT. Implementácia modelových tried obsahuje kód ktorý dokáže konvertovať dátový blok podľa potreby na reťazec vo formáte JSON alebo na dynamický dátový typ `dynamic`. Toto chovanie umožňuje prijímať a ukladať rôzne dátové a informačné správy.

4.3.8 Výsledná aplikácia

Adaptér je aplikácia spustiteľná na operačných systémoch Windows s verziou .Net framework-u vyššou ako 4.5. Aplikácia je distribuovaná ako inštalačný MSI alebo EXE súbor, ktorý nainštaluje adaptér do užívateľom zvolenej zložky. Tento inštalačný balík, obsahuje všetky použité knižnice a pri inštalácii kontroluje dostupnú verziu .Net frameworku. Tvorba inštalačného balíku bola realizovaná nástrojom dostupným v online rozšíreniach vývojového prostredia Visual Studio 2017. Súčasná verzia adaptéru je skompilovaná pre 64-bitové operačné systémy.

Ukážku grafického užívateľského rozhrania, je možné vidieť na obrázku 16.



Obrázok 16: Sekcia Maintenance - Adaptér

4.4 Bezpečnosť

Bezpečnosť senzorickej siete, závisí od možností nasadenia senzorickej siete do prostredia administratívnej budovy. Samotné možnosti nasadenia, sú ovplyvnené typom inštalácie. Senzorickú sieť je možné nasadiť do existujúcej sieťovej infraštruktúry, alebo je možné vybudovať sieťovú infraštruktúru špeciálne pre senzorickú sieť, čo je samozrejme nákladnejšie, ale pri správnom návrhu, vedie k vyššej bezpečnosti senzorickej siete. V nasledujúcich sekciách sú popísané kľúčové prvky bezpečnosti v senzorickej sieti.

4.4.1 Program senzorických uzlov

Ako už bolo spomenuté, každý Lua programový modul, ktorý je nahratý v flash pamäti mikrokontroléru je predkompilovaný do tzv. „Lua byte code“. Takto predkompilovaný programový modul je binárny súbor, ktorý bez reverzného inžinierstva nie je čitateľný. Samozrejme existujú spôsoby, ktorými je možné takýto kód prečítať a získať informácie napríklad z konfiguračného súboru, jedná sa však o komplikovanejšie procesy pri ktorých sú potrebné znalosti v danej oblasti.

4.4.2 Pripojenie uzlu do senzorickej siete

Senzorické uzly sa pripájajú do WiFi siete, pomocou údajov uchovávaných v konfiguračnom súbore. Dokážu sa pripojiť do sietí s viditeľným SSID ale taktiež aj do sietí, ktoré majú SSID skryté. Práve skryté SSID, môže prispieť k menšej náchylnosti na prelomenie zabezpečenia. Taktiež je možné pripojenie povoliť len vybraným MAC adresám, pokiaľ to podmienky nasadenia umožnia.

Hardvér a výrobcom dodávaná SDK k ESP-8266, umožňuje použitie WPA2-PSK³¹ s AES šifrovaním, čo je v súčasnosti štandard v oblasti WiFi sietí. Bohužiaľ v čase vývoja, ESP-8266 nepodporovalo WPA2-Enterprise s verejným certifikátom umiestneným v pamäti, čo by inak viedlo k zvýšeniu bezpečnosti pri komunikácii v bezdrátovej sieti.

Ako už bolo spomenuté, MQTT prostredník umožňuje zapnutie autentizácie MQTT klientov. Tým pádom je možné jednotlivým senzorickým uzlom a adaptérom vygenerovať prístupové údaje, pomocou ktorých sa na MQTT prostredníka prihlásia a tieto údaje zakomponovať do príslušných konfigurácií, čo zas vedie k zvýšeniu bezpečnosti komunikácie MQTT protokolom.

4.4.3 Výmena šifrovaných dát v senzorickej sieti

V senzorickej sieti je možné vynútiť šifrovanie dátových blokov prenášaných správ pomocou AES-ECB algoritmu a tajného kľúča s dĺžkou 128 bitov. Pri šifrovaní správ však vznikajú väčšie nároky na pamäť RAM a samotný výkon senzorických uzlov čo sa prejaví na nižšej rýchlosti odozvy zariadení.

MQTT Lua modul, ktorý poskytuje NodeMCU k svojmu firmvéru umožňuje šifrovanie komunikácie pomocou TLS s verziou 1.0 - 1.3. Avšak počas testovania MQTT komunikácie s TLS, bolo zistené že šifrovanie je veľmi pamäťovo náročné. Pamäťové nároky stúpili približne 16x, pri vypnutí TLS šifrovania zaberalo pripojenie na MQTT prostredníka približne 1.17kB, po nadviazaní pripojenia s aktivovaným TLS zabralo toto pripojenie v pamäti približne 18kB. Navyše jediná akcia ktorá sa so zapnutým TLS šifrovaním podarila, bolo nadviazanie pripojenia, pri pokuse o odoslanie alebo prijatie správy sa mikrokontrolér reštartoval z dôvodu nedostatku voľnej pamäte RAM. Tým vývojárom NodeMCU firmvéru, uvažuje nad implementáciou ECC³² certifikátov, ktoré by mohli viesť k nižšej pamäťovej náročnosti TLS šifrovania.

4.4.4 Zasielanie dát na REST Api

Pri zasielaní dát mimo senzorickú sieť, môže taktiež dôjsť k narušeniu bezpečnosti senzorickej siete. Http požiadavky generované adaptérom na webový server, by mali byť zasielané prostredníctvom https zabezpečeného protokolu. Tu je zas nutná

³¹WPA2-PSK - WiFi Protected Access-Pre Shared Key

³²ECC - Elliptic Curve Cryptography

podmienka platných certifikátov na webovom serveri a správne zadaná URL adresa REST rozhrania v nastaveniach adaptéru.

4.4.5 Sieťová infraštruktúra

Sieťová infraštruktúra senzorickej siete by mala byť dostatočne bezpečná. Do senzorickej siete by malo byť možné pripojiť len vybrané senzorické uzly na základe povolenej MAC adresy. MQTT komunikácia by mala byť odtienená od zvyšku komunikácie v sieti použitím VLAN, čím sa zabezpečí bezproblémová výmena dát MQTT protokolom. Jediným bodom v senzorickej sieti, z ktorého môžu prúdiť dáta mimo senzorickú sieť by mala byť aplikácia adaptéru, ktorá odosiela dáta na zabezpečené REST Api https protokolom. Prihlasovanie na MQTT prostredníka by malo byť povolené a jednotlivé senzorické uzly by mali mať vytvorené svoje unikátne prihlasovacie údaje, čím sa zaručí že v sieti komunikujú len komunikanti, ktorý na takúto komunikáciu majú oprávnenie. Obdobne ako senzorické uzly, by mal mať prihlasovacie údaje vygenerované aj adaptér, ktorému sa údaje dajú nastaviť v sekcii nastavení aplikácie.

4.5 Výsledná senzorická sieť

Ako bolo spomenuté v úvode tejto kapitoly, senzorickú sieť tvoria dve hlavné časti - senzorické uzly a správcovská aplikácia, ktoré spolu komunikujú. Navrhnutá senzorická sieť spĺňa stanovené požiadavky, senzorické uzly odosielajú dáta namerané vybranými senzormi, sú schopné sa zotaviť z prípadných chýb, ktoré môžu nastať - napríklad nedostatok pamäte z dôvodu nestability MQTT pripojenia, výpadok siete WiFi a iné. Senzorické uzly navyše umožňujú aktualizáciu programových modulov a možnosť vzdialenej správy prostredníctvom správcovskej aplikácie nazývanej adaptér.

Správcovská aplikácia zas dokáže zbierať dáta, generované senzorickými uzlami, vizualizovať vzorky nameraných dát, spravovať senzorické uzly, informovať administrátora o chybových udalostiach a aktualizovať vybrané programové moduly na vybraných senzorických uzloch.

Senzorická sieť má zabezpečenú stabilnú a rýchlu komunikáciu, ktorú využívajú všetci komunikanti. Navyše je možné senzorickú sieť rozširovať o ďalšie senzorické uzly, ktoré dodržia základné princípy komunikácie, stanovené v tejto diplomevej práci.

4.5.1 Testovanie riešenia počas vývoja

Počas práce na vývoji senzorickej siete, bolo potrebné navrhované riešenia, ako napríklad fyzické zapojenie, testovať v reálnych podmienkach, aby sa predišlo chybám. Prvý prototyp senzorického uzlu na vývojovej doske plošných spojov, je už viac ako rok spustený v učebni Q11 na našej fakulte. Má pripojené dva základné senzory

TSL-2561 a DHT-11. Na tomto prototypu sa ukázalo že navrhnutá schéma zapojenia vyhovuje pripojenému hardvéru, zariadenie sa neprehrieva, nemá nadmerný odber elektrického prúdu a ďalšie. Program senzorického uzlu, ako aj samotný NodeMCU firmvér prešiel mnohými optimalizáciami a vylepšeniami.

Popri testovaní prvého prototypu, bola testovaná aj aplikácia adaptéru, konkrétne sa sledovala stabilita aktualizáčného procesu, funkčnosť odosielania http požiadavkov, schopnosť znova nadviazať MQTT pripojenie za rôznych podmienok, pamäťová náročnosť a ďalšie.

Od prvotných prototypov senzorickej siete, prešli jednotlivé časti mnohými vylepšeniami a optimalizáciami, ktoré vo výsledku predstavujú finálne riešenie nasaditeľné do produkčného prostredia v reálnych podmienkach. Nasledujúca sekcia sa zaoberá nasadením vyhotoveného riešenia do reálnych podmienok. Samozrejmosťou pred predstavením finálnej verzie, je dlhodobé testovanie riešenia v reálnych podmienkach. Tejto téme sa venuje nasledujúca kapitola tejto diplomovej práce.

4.5.2 Nasadenie senzorickej siete

Pred samotným nasadením, je potrebné zabezpečiť v sieti určité prerekvizity. Jedná sa o WiFi prístupový bod s WPA2-PSK autentizáciou. Ďalej je potrebné aby bol v sieti dostupný na známej IP adrese a porte MQTT prostredník a počítač s operačným systémom Windows, ktorý pokiaľ to situácia vyžaduje, môže komunikovať mimo senzorickú sieť, za účelom distribúcie dát na REST Api.

Podrobnejšie sa tomuto problému venujem v nasledujúcej kapitole.

5 Nasadenie do skúšobnej prevádzky

Nasadenie senzorickej siete do skúšobnej prevádzky, prebiehalo v domácom prostredí a v prostredí Mendelovej univerzity.

V univerzitnom prostredí bola vytvorená sieťová infraštruktúra, do ktorej sa nasadili senzorické uzly, adaptér a MQTT prostredník. Sieťová infraštruktúra pozostávala z prístupových bodov, ktoré majú skryté SSID, WiFi autentizáciu prostredníctvom WPA2-PSK a MQTT komunikáciu oddelenú do samostatnej VLAN. Do siete sa prihlásili iba senzorické uzly, ktoré mali v prístupových bodoch zadané a povolené MAC adresy.

Adaptér odosiela zozbierané dáta mimo senzorickú sieť a ako jediný prvok senzorickej siete mal možnosť komunikácie mimo senzorickú VLAN a to konkrétne na univerzitný webový server kde zasielal dáta na REST Api.

Súčastou sieťovej infraštruktúry boli dva počítače. Prvý počítač slúžil ako MQTT prostredník na operačnom systéme CentOS, s MQTT službou od spoločnosti Mosquitto a verziou MQTT podporovaného protokolu 3.1.1. Druhý počítač slúžil k behu adaptéru, ktorý bežal na 64-bitovom operačnom systéme Windows.

Pred samotným nasadením, bolo potrebné zostaviť konfiguračné súbory pre senzorické uzly a správne nastaviť aplikáciu adaptéru. Jednalo sa o prístupové údaje do siete WiFi a IP adresu MQTT prostredníka. Následne bolo potrebné tieto konfiguračné súbory nahráť do príslušných senzorických uzlov.

V nasledujúcej časti je popísané dlhodobé testovanie senzorickej siete.

5.1 Dlhodobé testovanie

Testovanie prebiehalo počas celej doby vývoja senzorickej siete. Testované boli ako senzorické uzly, tak aj aplikácia adaptéru. K dispozícii sú namerané dáta a informačné správy, zozbierané a uložené do lokálnej databáze adaptérom.

Do testovania bolo zahrnutých 6 senzorických uzlov, vybavených rôznymi kombináciami senzorov. Testovacia skupina taktiež obsahovala senzorický uzol bez pripojených senzorov, čo malo indikovať situáciu poškodenia senzoru a otestovať tak notifikácie generované adaptérom.

Adaptér mal nastavené reportovanie chyby každých 24hodín a v prípade detekovanej chyby, každých 24hodín zaslal email s chybovou správou o danom stave senzorického uzlu.

Zvyšných 5 senzorických uzlov malo nasledovnú konfiguráciu uvedenú v tabuľke 6.

Konfigurácia senzorických uzlov bola nastavená na zasielanie dátových správ každý 60 sekúnd a zasielanie informačných správ bolo nastavené na každých 10 minút. Senzorické uzly spoľahlivo reagovali aj na manuálne vyžiadanie dátovej, informačnej alebo identifikačnej správy prostredníctvom adaptéru.

Tabuľka 6: Prehľad testovaných senzorických uzlov a pripojených senzorov

Id senzorického uzlu	Pripojené senzory
5718496	TCS-34725, DHT-11
5718486	TSL-2561, DHT-11
5718344	TSL-2561, DHT-11
5717723	Si-7021
5719416	DHT-11
5717740	DHT-11 - nepripojený

Ako už bolo naznačené senzorická sieť bola testovaná voči výpadkom WiFi konektivity a výpadkom MQTT prostredníka, pričom sa preukázalo spoľahlivé znovu nadviazanie komunikácie po obnovení konektivity.

Testovaná bola taktiež spoľahlivosť aktualizácie programových modulov senzorických uzlov a schopnosť zotavenia sa z neúspešného priebehu aktualizácie. Toto bolo simulované dvoma rôznymi spôsobmi, odpojením WiFi pripojenia a zastavením služby MQTT prostredníka počas priebehu aktualizácie. Aktualizovaný súbor bol pri neúspešnej aktualizácii, zo senzorického uzlu po reštarte odstránený, čo je považované za správne implementované chovanie, pri neúspešnom aktualizáčnom procese.

Senzorické uzly, mali taktiež simulované poškodenie senzoru a to jeho neočakávaným odpojením, pričom nedošlo k neočakávanému reštartu senzorického uzlu, ale naopak došlo k zaslaniu informačnej správy, ktorá niesla informáciu o chybovom stave odpojeného senzoru. Po opätovnom pripojení senzoru sa stav vrátil do normálu.

V dlhodobom testovaní sa taktiež testovala schopnosť šifrovania a dešifrovania dátového bloku prenášaných správ. Senzorické uzly zasielajúce správy so šifrovaným dátovým blokom, vykazovali o niečo vyššiu pamäťovú náročnosť. V závislosti od dĺžky šifrovaného reťazca sa pamäťová náročnosť zvýšila približne o 1 - 2,2kB. Pri zasielaní šifrovaných správ nebol odhalený žiadny únik pamäte a pamäť využitá počas šifrovania, bola po odoslaní správy opätovne uvoľnená.

Zasielanie http požiadavkov bolo testované ako v domácom tak aj v univerzitnom prostredí. Pričom sa preukázala funkčnosť ako http tak aj https - zabezpečených požiadavkov na REST Api. Pokiaľ webový server neodpovedal, dochádzalo k miernemu navýšeniu pamätevej náročnosti aplikácie adaptéru, nakoľko asynchrónne požiadavky čakali na odpoveď a skončili až vypršaným timeout-om. V prípade bezproblémového fungovania webového serveru sa aplikácia adaptéru z pohľadu pamätevej náročnosti, chovala stabilne - pamäť bola pri tvorbe a zaslaní požiadavku zabraná a následne pri obdržaní odpovede uvoľnená.

Stabilita adaptéru bola overená sústavným behom aplikácie, počas ktorého sa testovalo prijímanie a zasielanie správ, priebežná aktualizácia senzorických uzlov, pamäťová náročnosť a ďalšie. Pamäťová náročnosť aplikácie stúpala po implementácii

zobrazovania grafov na hodnotu približne 100 - 150MB, v závislosti od množstva nazbieraných a prijatých dát.

Počas testovania bolo vytvorených viac ako 15 lokálnych databáz, ktoré sú obsiahnuté v prílohách a je možné ich prehliadať prestredníctvom aplikácie adaptéru. Ukážky nameraných dát zo senzorov, je možné vidieť na grafoch obrázku 21. Grafy zobrazujú dostupné zozbierané dáta, konkrétne sa jedná o teplotu, vlhkosť a úroveň svetelnosti. Pokiaľ senzorický uzol nedisponoval senzorom, merajúcim danú fyzikálnu veličinu, v grafe má nulovú hodnotu.

5.2 Ukážky nameraných dát

Zozbierané a uložené dáta od jednotlivých senzorických uzlov, sú si veľmi podobné, nakoľko senzorické uzly boli testované v jednej miestnosti. Práve preto je možné na grafoch pozorovať veľmi podobné namerané hodnoty teploty a vlhkosti, hodnoty úrovne svetelnosti sa naopak líšia, nakoľko každý svetelný senzor bol natočený trochu inak, čo spôsobilo že svetlo nedopadalo rovnomerne na jednotlivé svetelné senzory. Výnimkou je obrázok 17, kde som sa pokúsil senzory rôzne rozmiestniť a meniť jednotlivé podmienky senzorov, aby bolo evidentné že dáta odpovedajú aktuálnym podmienkam. Zvyšné dva grafy je možné vidieť v prílohách na obrázkoch 21 a 22.

V priložených grafoch, je možné vidieť vzorky zozbieraných dát za rôzne obdobia. Každý obrázok obsahuje tri grafy, na ktorých je možné vidieť vzorky zozbieraných dát adaptérom. Vrchný graf znázorňuje priebeh teploty, graf v strede obrázku znázorňuje relatívnu vlhkosť a posledný graf v jednotlivých obrázkoch, ukazuje namerané hodnoty svetelnosti. Spodný popisok u jednotlivých grafov popisuje časovú osu. Pokiaľ daný senzor, nedisponuje senzorom ktorý by dokázal zmerať vybranú fyzikálnu veličinu, v príslušnom grafe má nulové hodnoty.



Obrázok 17: Ukážka nazbieraných dát z aplikácie adaptéru

6 Zhrnutie

Diplomová práca sa zaoberala návrhom inteligentnej senzorickej siete v prostredí administratívnej budovy. Senzorická sieť sa skladá z jednotlivých senzorických uzlov, ktoré komunikujú prostredníctvom MQTT protokolu so správcovskou aplikáciou. Správcovská aplikácia nazývaná adaptér, zbiera dáta produkované senzorickými uzlami a umožňuje ich distribúciu mimo senzorickú sieť. Adaptér umožňuje aktualizáciu programu a správu jednotlivých senzorických uzlov.

V prvej časti tejto diplomovej práce bola vykonaná rešeršná štúdia existujúcich riešení na trhu, zhodnotenie výsledkov rešeršnej štúdie a stanovenie požiadavkov na inteligentnú senzorickú sieť.

Následovná časť sa potýka s teoretickým úvodom do zvolených technológií a návrhom senzorického uzlu a správcovskej aplikácie, ktoré tvoria senzorickú sieť.

Za teoretickou časťou tejto diplomovej práce nasleduje kapitola s názvom návrh riešenia. V tejto kapitole je čitateľ oboznámený s navrhnutým riešením, ktoré je na začiatku dekomponované do jednotlivých častí a v závere vytvorí funkčný celok inteligentnej senzorickej siete administratívnej budovy. Táto časť taktiež popisuje princípy zabezpečenia a nasadenia senzorickej siete.

V následovnej kapitole, je popísaný postup nasadenia senzorickej siete do reálnych podmienok a spôsob dlhodobého testovania. Problémy, ktoré sa ukázali počas dlhodobého testovania sú popísané v časti diskusie tejto diplomovej práce.

V posledných častiach tejto diplomovej práce, je zhodnotené navrhnuté riešenie, sú tu popísané navrhované vylepšenia a je vykonaná diskusia.

7 Diskusia

Za prednosť výslednej senzorickej siete, je možné považovať napríklad vzdialenú aktualizáciu programu na senzorických uzloch. V prípade globálnej potreby úpravy programu, je možné vykonať aktualizáciu z jedného miesta a nie je nutné senzorické uzly z budovy zozbierať a ručne preprogramovať. Navyše doska plošných spojov má jednotný rozmer a je možné pripojiť rôzne druhy senzorov, čo je možné tiež považovať za dobrú vlastnosť. V závislosti od konfiguračného súboru je tak možné čítať dáta z ľubovoľného senzoru z vybraných, čo zas umožňuje, napríklad pri poruche senzoru TCS-34725 nahradiť ho senzorom TSL-2561 a iba upraviť konfiguračný súbor.

Ďalšou dobrou vlastnosťou senzorických uzlov, je schopnosť zotavovania sa po chybových stavoch, ako napríklad neočakávaný výpadok siete WiFi alebo MQTT prostredníka. Navyše, vďaka MQTT správam poslednej vôle, je administrátor stále informovaný o tom, či sa niektorý senzorický uzol odpojil. Taktiež je možné poukázať na reportovanie stavov senzorických uzlov, čo prináša tvorbu základného prehľadu o celkovom stave senzorickej siete a zároveň toto chovanie umožnilo tvorbu notifikácií o kritických udalostiach. Administrátor siete, nie je nútený sledovať zozbierané dáta a kontrolovať či sú všetky senzory v dobrej kondícii, v prípade problému bude informovaný emailom.

Počas dlhodobých testov, sa odhalilo niekoľko nedostatkov a väčšinu z nich sa podarilo počas testovania odstrániť. Bohužiaľ známe chyby firmvéru NodeMCU a Espressif SDK, neboli vývojármi opravené dostatočne včas, aby sa premietli do finálneho riešenia.

Jedná sa o chyby MQTT keep-alive časovača, ktorý bez ohľadu na nadstavenú hodnotu v konštruktoze klienta, beží v 60 sekundových intervaloch. Ďalšou už spomínanou chybou v NodeMCU firmvéri je problém s uvoľňovaním pamäte pri pokuse o znovu pripojenie na MQTT prostredníka. Obe tieto chyby, som referoval v repozitári firmvéru NodeMCU a čakám ako sa k nim tým vývojárov postaví. Je pravdepodobné že spomínané nedostatky budú v blízkej dobe odstránené, čo vyrieši nežiadúci únik pamäte a stabilitu MQTT spojenia.

Na oprave chyby s uvoľňovaním pamäte pri znovu nadviazaní MQTT pripojenia, sa aktívne podieľam. Momentálne testujem na jednom senzorickom uzle novú nezverejnenú verziu firmvéru kde sa javí že problém je odstránený. Je ale potrebné dlhodobejšie testovanie, aby sa overilo že daná oprava nespôsobí nepredpokladané chovanie. Firmvér ktorý momentálne testujem bolo potrebné zostaviť. Jedná sa o komplikovanejší proces, pre ktorý je potrebná znalosť programovacieho jazyku C, práca s programom Git a zostavenie firmvéru pomocou nástroja Docker. Neoddeliteľnou súčasťou je samozrejme komunikácia s členmi NodeMCU vývojového tímu. NodeMCU vyvinula Docker obraz, ktorý slúži k zjednodušeniu zostavovania firmvéru, umožňuje použiť niekoľko „flagov“, ktoré ovplyvňujú typ výsledného zostavenia. Pre dôkladné ladenie firmvéru je potrebné upraviť hlavičkové súbory, v ktorých sa dajú povoliť ladiace informácie.

V master verzii oficiálneho firmvéru sa v závislosti od počtu pokusov o pripojenie na MQTT prostredníka, klesá voľná pamäť senzorického uzlu každým pokusom o 112B. Tento problém je momentálne riešený sledovaním úrovne voľnej pamäte RAM a pri poklese pod úroveň 4kB, je nariadené reštartovanie senzorického uzlu. Táto kontrola prebieha pri pokuse o znovu nadviazanie MQTT pripojenia. Pokiaľ sa ukáže, že testovaný firmvér, ktorý túto chybu neobsahuje, je stabilný, bude nasledovať kompletne prehratie firmvéru na všetkých senzorických uzloch.

K odhaleniu úniku pamäte na senzorických uzloch došlo počas testovania schopnosti znovu nadviazania pripojenia do WiFi siete a následného pripojenia sa na MQTT prostredníka. V domácom prostredí bežal MQTT prostredník na zariadení Raspberry Pi 3 model B+, s originálnou distribúciou linuxového OS Raspbian, ktorá bola pravidelne aktualizovaná. Verzia MQTT prostredníka bola taktiež pravidelne aktualizovaná, použitý bol MQTT prostredník spoločnosti Mosquitto. Pri odhaľovaní chyby s únikom pamäte a študovaním logu MQTT prostredníka, bolo taktiež zistené, že nadviazanie pripojenia klienta prebiehalo vždy s hodnotou keep-alive časovača nastavenou na 60 sekúnd, čomu odpovedala aj následná sieťová komunikácia, kedy senzorický uzol vyslal PINGREQ paket na prostredníka vždy v 60 sekundových intervaloch.

Spomínané nedostatky NodeMCU MQTT modulu, sa prejavili na stabilite senzorických uzlov, meraním úrovne voľnej pamäte bolo preukázané, že k úniku pamäte nedochádza počas merania hodnôt zo senzorov, alebo počas prijímania či odosiadania MQTT správ. Pokiaľ sa ukáže, že chyby boli v novej verzii firmvéru opravené, firmvér bude možné zostaviť a následne nahráť na senzorický uzol, tak by mal byť problém so stabilitou pamäte RAM vyriešený. Verzia NodeMCU firmvéru 0.9.4.1 ako aj verzia 1.5.4.1 týmto únikom pamäte nedisponujú, ale bohužiaľ ich nie je možné použiť, nakoľko neobsahujú implementáciu modulov pre všetky použité senzory.

Dočasná oprava spomínaného problému reštartovaním senzorického uzlu pri nízkej úrovni voľnej pamäte RAM, sa neprejavila negatívne na zbere meraných dát. Samotný štart senzorického uzlu po reštarte trvá približne 13 sekúnd a po štarte sú odoslané všetky potrebné správy. V praxi takýto reštart modulu spôsobí odoslanie dát o 13 - 15 sekúnd neskôr, ako bolo pôvodne načasované. K reštartu dochádza v závislosti od počtu pokusov znovu pripojenia, konkrétne sa jedná o približne 120 pokusov o znovu pripojenie, čo spotrebuje 13 440 bajtov voľnej pamäte RAM. Počet pokusov o znovu pripojenie závisí od stability MQTT prostredníka, pokiaľ MQTT prostredník, respektíve zariadenie na ktorom prostredník beží, komunikuje nespoľahlivo, prípadne sa pakety v sieti poškodia, je potrebné nadviazať nové pripojenie.

Trvanie procesu znovu nadviazania pripojenia na senzorickom uzle, závisí od mnohých faktorov. Počas testovania sa však ukázalo, že netrvá dlhšie ako 5 sekúnd.

8 Navrhované vylepšenia

Ako jedno z navrhovaných vylepšení senzorickej siete, je navrhnutie akčného senzorickeho uzlu, ktorý okrem zbierania dát zo sensorov, môže ovládať pripojené zariadenie. Mohlo by sa jednať napríklad o ovládanie svetiel, ventilácie, okien, roliet a ďalších. K tomuto rozšíreniu by bolo potrebné rozšíriť správcovsku aplikáciu, aby umožnila správcovi senzorickej siete ovládať jednotlivé senzorické uzly, prípadne implementovať automatické ovládanie pripojenej periférie na základe zozbieraných dát senzorickým uzlom.

Ako ďalšie navrhované vylepšenie, môže byť navrhnutie senzorických uzlov na výkonnejšom zariadení s väčšou pamäťou RAM a implementovať TLS šifrovanie MQTT komunikácie. Mohlo by sa jednať o novú verziu ESP mikro-kontroléru ESP-32, ktoré je niekoľko násobne výkonnejšie ako použité zariadenie ESP-8266. Zariadenie ESP-32, nebolo dostupné v čase kedy sa započala práca na tejto diplomovej práci.

Do budúcich vylepšení by som ešte zahrnul aktualizáciu firmvéru na senzorických uzloch, potom ako budú chyby popísané v diskusii opravené v novej oficiálnej verzii firmvéru. K tomuto môže pomôcť sledovanie založených pripomienok v GitHub repozitári firmvéru NodeMCU.

9 Záver

Cieľom tejto práce bolo navrhnúť inteligentnú senzorickú sieť pre administratívnu budovu, ktorá umožní merať vybrané fyzikálne veličiny pomocou vhodných senzorov. Navyše bolo potrebné senzorickú sieť spravovať vhodným nástrojom a zabezpečiť bezpečnú komunikáciu pri odosielaní dát na server.

Finálne riešenie bolo dlhodobo otestované v reálnych podmienkach, kde bola zabezpečená bezpečná komunikácia.

Cieľ práce bol splnený, senzorické uzly produkujú dáta a distribujú ich prostredníctvom senzorickej siete k správcovskej aplikácii. Senzorické uzly, sú schopné zotavovať sa z chybových stavov čo preukázalo aj dlhodobé testovanie. Pomocou správcovskej aplikácie je možné vzdialene aktualizovať program vybraných senzorických uzlov a jednotlivé uzly spravovať. Pri návrhu riešenia, bolo implementovaných niekoľko mechanizmov, ktoré významne zvyšujú zabezpečenie senzorickej siete.

Výsledné riešenie bolo úspešne nasadené do priestorov univerzity, kde bola vytvorená príslušná sieťová infraštruktúra.

10 Literatúra

- AOSONG. *DHT11 Humidity & Temperature Sensor* In: Mouser[online]. 2009, [cit. 2018-02-1]. Dostupné z: <https://www.mouser.com/ds/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>.
- BACZKOWICZ, K.. *MqttSpy - Getting started* In: GitHub[online]. 2015, [cit. 2018-03-11]. Dostupné z: <https://github.com/kamilfb/mqtt-spy/wiki/GettingStarted>.
- BIGCLOWN. *Getting Started* In: BigClown[online]. 2018, [cit. 2018-02-4]. Dostupné z: <https://www.bigclown.com/doc/basics/getting-started/>.
- BIGCLOWN. *Core Module* In: BigClown[online]. 2018, [cit. 2018-02-4]. Dostupné z: <https://www.bigclown.com/doc/hardware/about-core-module/>.
- CRESTRON. *Installation and Setup of Crestron RF Products* In: Crestron[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.crestron.com/getmedia/3b3588d1-9356-4d4b-a2df-b3ac1c6fff14/mg_bp_installation_setup_crestron_rf_products_1.
- CRESTRON. *Crestron Fusion - Monitoring & Scheduling Software* In: Crestron[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.crestron.com/getmedia/e9005e1e-f424-44d5-85d3-e6ba36d74c73/pb_crestron_fusion_1.
- CRESTRON. *Spec sheet: Pre-configured Server for Crestron Fusion* In: Crestron[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.crestron.com/getmedia/9c79f6a2-3b79-4469-8f5f-73d9980b347d/ss_cen-fusion-server-r330_1.
- CRESTRON. *Spec sheet: GLA-BMS* In: Crestron[online]. 2018, [cit. 2018-02-1]. Dostupné z: <https://www.crestron.com/specsheets/pdfhandler.ashx?action=generatedownload&modelnumbers=GLA-BMS>.
- CUE. *Technologies integration* In: Cue[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.cuesystem.com/technologies_integration.aspx.
- CUE. *inputCUE-W* In: Cue[online]. 2009, [cit. 2018-02-1]. Dostupné z: https://www.cuesystem.com/Files/ProductDataSheets/DS058_01_inputCUE-W.pdf.
- CUE. *Cue Visual Composer* In: Cue[online]. 2010, [cit. 2018-02-1]. Dostupné z: <https://www.cuesystem.com/Files/ProductCutSheets/Cue%20Visual%20Composer.pdf>.
- E LUA. *eLua Overview* In: eLua[online]. 2011, [cit. 2018-02-1]. Dostupné z: <http://www.eluaproject.net/overview>.

- ELKO EP. *iNels - Bezdrátová elektroinstalace* In: ElkoEP[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.elkoep.cz/media/files/download/item/files-188/11_iNELS_RF_technical_2018_CZ_print_version.pdf.
- ELKO EP. *iNels - Sběrníková elektroinstalace* In: ElkoEP[online]. 2018, [cit. 2018-02-1]. Dostupné z: https://www.elkoep.cz/media/files/download/item/files-152/11_iNELS_BUS_general_CZ_2018_print_version.pdf.
- ESPRESSIF. *ESP8266EX Datasheet* In: Adafruit[online]. 2015, [cit. 2018-02-1]. Dostupné z: https://cdn-shop.adafruit.com/product-files/2471/0A-ESP8266__Datasheet__EN_v4.3.pdf.
- HOROWITZ, P., HILL, W. *The Art of Electronics*. Vyd. 3. New York: Cambridge, 2015. 1125 s. ISBN 978-0-521-80926-9.
- KNX. *ABB iBus KNX* In: Abb[online]. 2011, [cit. 2018-02-1]. Dostupné z: http://www.knx-gebaeudesysteme.de/sto_g/English/GENERAL_DOCUMENTATION/2CDC500043B0202_KNX_Flash_EN.pdf.
- NODEMCU. *Lua Language* In: NodeMCU[online]. 2017, [cit. 2018-02-15]. Dostupné z: <https://nodemcu.readthedocs.io/en/master/en/lua-developer-faq/#lua-language>.
- NODEMCU. *ESP 8266* In: NodeMCU[online]. 2017, [cit. 2018-02-15]. Dostupné z: <https://nodemcu.readthedocs.io/en/master/en/lua-developer-faq/#esp8266-specifics>.
- KNX. *KNX - Celosvětová NORMA pro řízení domů a budov* In: KNX[online]. 2016, [cit. 2018-02-2]. Dostupné z: https://www.knx.org/media/docs/Flyers/KNX-Introduction-Flyer/KNX-Introduction-Flyer_cz.pdf.
- KNX. *Zabezpečení, efektivita a komfort s KNX* In: KNX[online]. 2018, [cit. 2018-02-2]. Dostupné z: https://www.knx.org/media/docs/downloads/Marketing/Flyers/Consumer-Flyer/Consumer-Flyer_cz.pdf.
- KNX. *KNX - Základy* In: KNX[online]. 2017, [cit. 2018-02-2]. Dostupné z: https://www.knx.org/media/docs/downloads/Marketing/Flyers/KNX-Basics/KNX-Basics_cz.pdf.
- LAMPKIN, V., OLIVERA, L. *Building Smarter Planet Solutions with MQTT and IBM WebSphere MQ Telemetry*. First edition. Poughkeepsie, NY: IBM International Technical Support Organization, 2012. 249 s. ISBN 0738437085.
- OASIS. *MQTT Version 3.1.1* In: Oasis[online]. 2014, [cit. 2018-03-4]. Dostupné z: <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/csprd02/mqtt-v3.1.1-csprd02.htmlf>.
- SILICON LABS. *i2C HUMIDITY AND TEMPERATURE SENSOR* In: Adafruit[online]. 2016, [cit. 2018-02-5]. Dostupné z: <https://www.adafruit.com/product/2471>.

`//cdn-learn.adafruit.com/assets/assets/000/035/931/original/
Support_Documents_TechnicalDocs_Si7021-A20.pdf.`

TEXAS ADVANCED OPTOELECTRONICS SOLUTIONS. *TSL2561 LIGHT-TO-DIGITAL CONVERTER* In: Adafruit[online]. 2009, [cit. 2018-02-4]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf>.

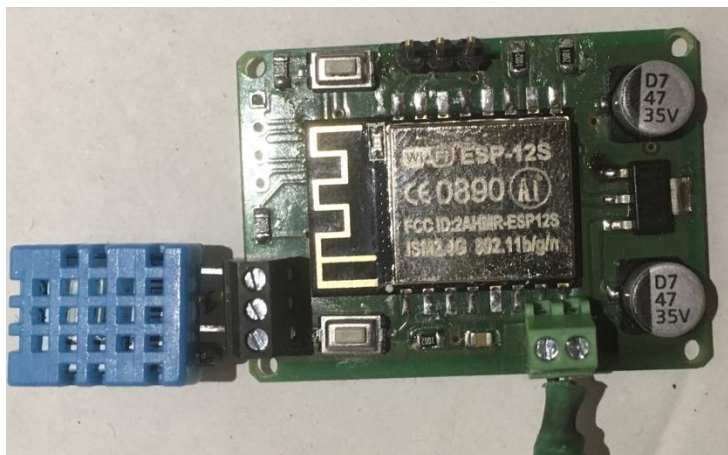
TEXAS ADVANCED OPTOELECTRONICS SOLUTIONS. *TCS34725 COLOR LIGHT-TO-DIGITAL CONVERTER* In: Adafruit[online]. 2012, [cit. 2018-02-4]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/TCS34725.pdf>.

VERMESAN, O., FRIESS, P. *Internet of Things - From Research and Innovation to Market Deployment*. Aalborg: River Publishers, 2014. 355 s. ISBN 978-87-93102-94-1..

Prílohy

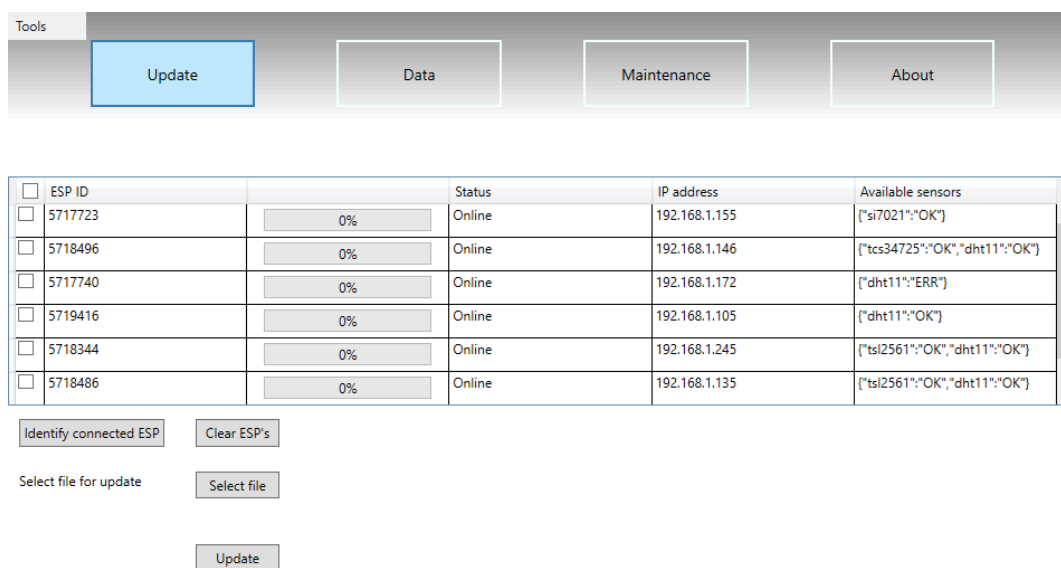
A Obrázky

A.1 Senzorický uzol



Obrázok 18: Osadený senzorický uzol so senzorom DHT-11

A.2 Adaptér



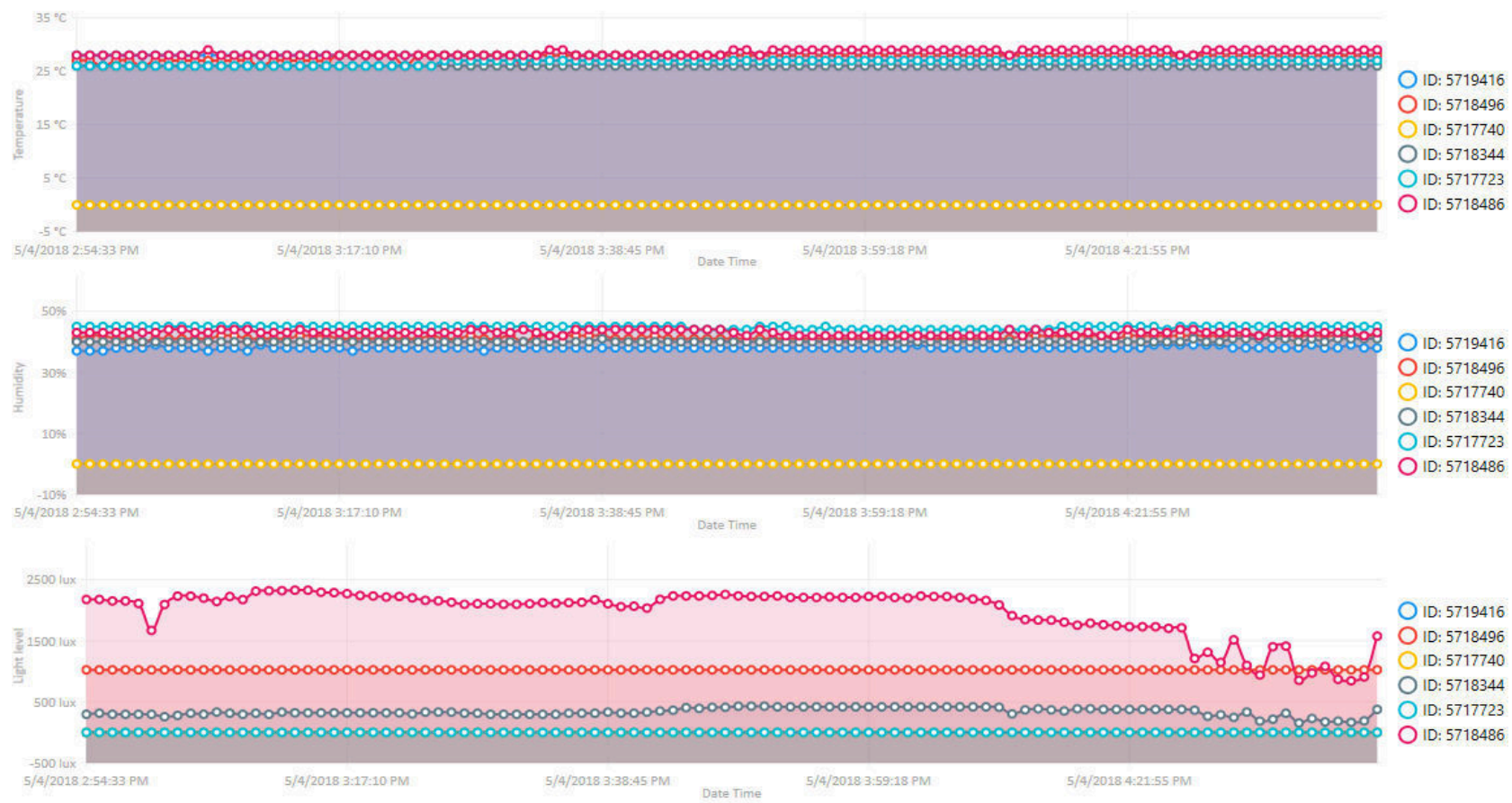
Obrázok 19: Adaptér - sekcia aktualizácie

Settings

Broker host name/ IP	192.168.1.34
QOS	2
Update publish topic	esp_module/update_req/
Update subscribe topic	esp_module/update_res/
Data publish topic	/service/
Data subscribe topic	/data/
Identification publish topic	/service/
Identification subscribe topic	/maintenance/answer
User name	
Password	
Path for database file	ESPUpdater/db/ESPUpdater_14.sqlite
Enable parallel update	<input checked="" type="checkbox"/> Enable REST <input type="checkbox"/>
Email to	xfoldvar@gmail.com
Notification frequency (hours)	<input type="range"/>
Rest Server Host	https://smartpef-dev.mendelu.cz:44903
Rest request target	endpoints/dataendpoint/controller/

Save

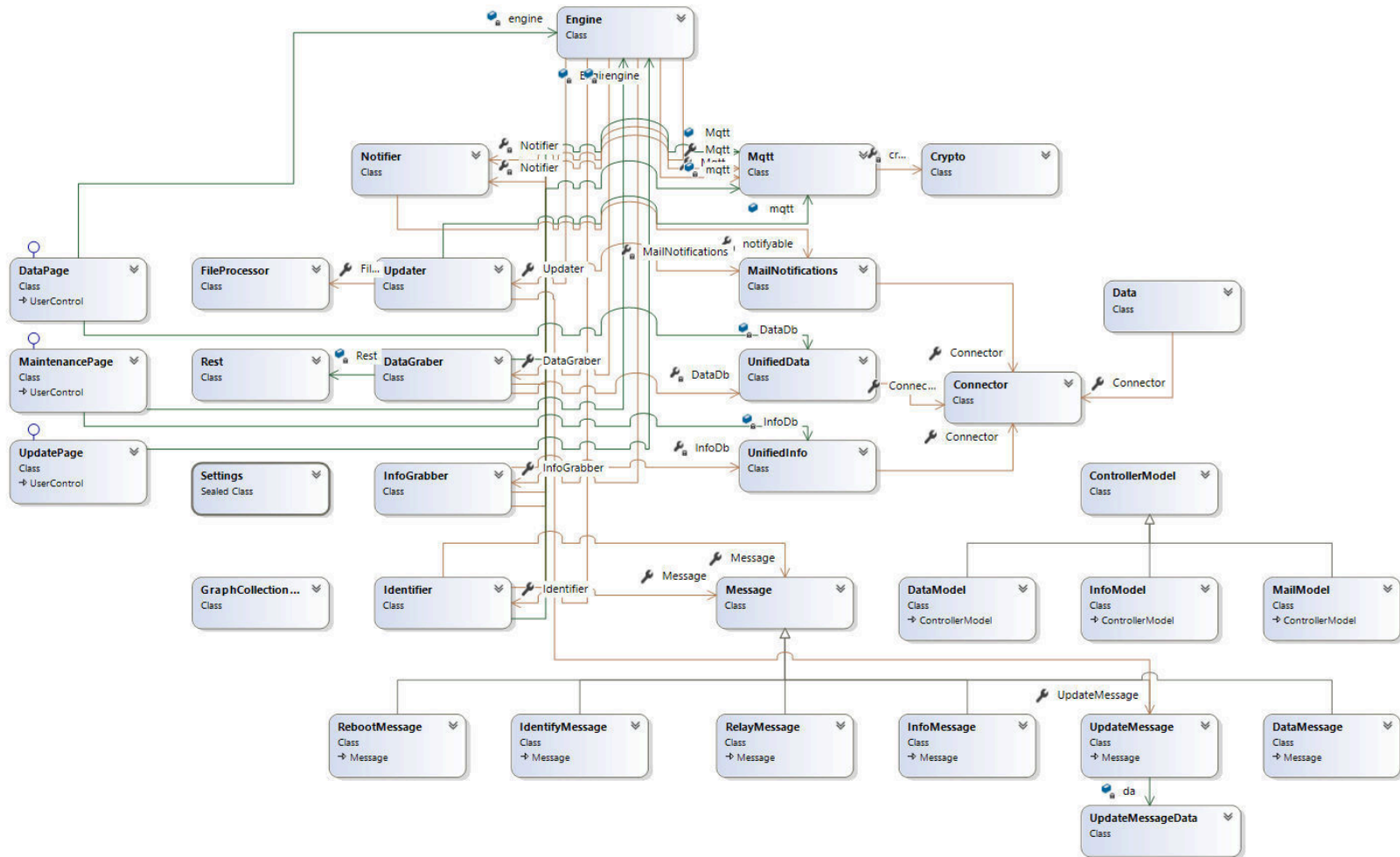
Obrázok 20: Adaptér - sekcia nastavení



Obrázok 21: Ukážka nazbieraných dát z aplikácie adaptéru



Obrázok 22: Ukážka nazbieraných dát z aplikácie adaptéru



Obrázok 23: Diagram tried aplikácie adaptéru

B Priložené CD

Priložené CD obsahuje:

- Zdrojové súbory programu senzorických uzlov.
- Zdrojové súbory aplikácie adaptéru.
- Súbory obsahujúce finálnu verziu dosiek plošných spojov.
- Testovacie lokálne databázy.