



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

Řízení a monitorování fyzikálních veličin v horském floráriu po internetu

Bakalářská práce

Studijní program: B2612 – Elektrotechnika a informatika
Studijní obor: 2612R011 – Elektronické informační a řídicí systémy
Autor práce: **Josef Břich**
Vedoucí práce: Ing. Petr Bílek, Ph.D.
Konzultant: Ing. Petr Kretschmer





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

Control and Monitoring of Physical Values in a Highland Florarium via Web Interface

Bachelor thesis

Study programme: B2612 – Electrical Engineering and Informatics
Study branch: 2612R011 – Electronic Information and Control Systems

Author: **Josef Brich**
Supervisor: Ing. Petr Bílek, Ph.D.
Consultant: Ing. Petr Kretschmer



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Josef Brich**
Osobní číslo: **M15000083**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektronické informační a řídicí systémy**
Název tématu: **Řízení a monitorování fyzikálních veličin v horském floráriu po internetu**
Zadávací katedra: **Ústav nových technologií a aplikované informatiky**

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

1. Sestavené a ověřené jádro zařízení v bakalářském projektu dopracujte do úplného schématu (volba dalších komponent, blokové schéma, vývojový diagram).
2. Sestavte hardware podle schématu a oživte celé zařízení.
3. Navrhněte a odlaďte firmware v jazyce Python.
4. Vytvořte webové rozhraní pro monitoring a ovládání fyzikálních veličin ve floráriu.
5. Prověřte možnost ukládání a zobrazování dat na vzdáleném serveru (například: thingspeak.com, plot.ly).
6. Upravte komerční skříňku pro zabudování zařízení.
7. Otestujte sestavené zařízení v reálných podmínkách.



Rozsah grafických prací: **dle potřeby**
Rozsah pracovní zprávy: **30 - 40 stran**
Forma zpracování bakalářské práce: **tištěná/elektronická**
Seznam odborné literatury:

- [1] KREIDL, Marcel. Měření teploty - senzory a měřicí obvody. BEN - technická literatura. Praha 2005. 240 s. ISBN 80-7300-145-4.
- [2] HORAN, Brendan. Practical Raspberry Pi. New York: Apress, 2013, xxxii, 239 s. ISBN 978-1-4302-4971-9.
- [3] GIRLING, Gary. Raspberry Pi: a practical guide to the revolutionary small computer. Sparkford: Haynes Publishing, 2013, 169 s. ISBN 978-0-85733-295-0.
- [4] Datasheety vybraných senzorů.

Vedoucí bakalářské práce: **Ing. Petr Bílek, Ph.D.**
Ústav nových technologií a aplikované informatiky
Konzultant bakalářské práce: **Ing. Petr Kretschmer**
Ústav nových technologií a aplikované informatiky
Datum zadání bakalářské práce: **19. října 2017**
Termín odevzdání bakalářské práce: **14. května 2018**

prof. Ing. Zdeněk Plíva, Ph.D.
děkan



Ing. Josef Novák, Ph.D.
vedoucí ústavu



V Liberci dne 19. října 2017

Prohlášení

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

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

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

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

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

Datum: 13.4.2018

Podpis: 

Abstrakt

Tato bakalářská práce se zabývá rešerší, návrhem a realizací řídicího systému vysokohorského florária, realizovaného pomocí Raspberry Pi. Rešerše se zabývá popisem a možnostmi měření fyzikálních veličin. Dále je popsán návrh a volba hardwarového vybavení florária a řídicího systému. Vytvořené elektrické zařízení sbírá fyzikální data ze senzorů teploty, CO₂, vlhkosti vzduchu, vlhkosti půdy a prezentuje je na webové stránce ve formě grafů. Dále na tyto data reaguje pomocí akčních členů – topným kabelem, ventilátory a závlahovým systémem. Stejnou webovou aplikací je umožněno i ruční ovládání akčních členů. V práci jsou nastíněny i problémy které se v průběhu realizace objevovaly a byly nutné vyřešit pro další postup.

Klíčová slova Raspberry Pi, florarium, python, měření teploty, měření CO₂

Abstract

This bachelor thesis deals with research, layout and realisation of controlling system of a highland florarium, which was realised via Raspberry Pi. The research deals with a description and options of measurement of physical quantities. It also includes a description and choice of hardware equipment of the florarium and the controlling system. The electric device collects physical data from the temperature sensors, CO₂, air humidity, and presents them on the website in graphs. The device also responds to this data via actuators such as heating cable, ventilators and watering system. The manual control of actuators is available via the same web application. This bachelor thesis also deals with problems, which have shown during the realisation and which had to be solved for the next steps.

Keywords Raspberry Pi, florarium, python, temperature measurement, CO₂ measurement

Poděkování

Velmi rád bych poděkoval vedoucímu bakalářské práce, panu Ing. Petru Bílkovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné nápady k práci.

Rodině a Elišce Chvalové za jejich trpělivost a stálou podporu při studiu. A závěrem i Aleši Kopeckému za věcné konzultace ohledně tvorby webu.

Obsah

Seznam zkratek	9
1 Úvod	10
2 Rešerše	11
2.1 Rostliny	11
2.2 Sensory fyzikálních veličin	11
2.2.1 Měření teploty	11
2.2.2 Měření vlhkosti vzduchu	13
2.2.3 Měření CO ₂	14
2.2.4 Měření intenzity osvětlení	16
3 Realizace florária	17
3.1 Návrh řídicího a monitorovacího systému	17
3.2 Montáž konstrukce florária	20
4 Realizace řídicího a monitorovacího systému	23
4.1 Nastavení řídicího mikropočítače	23
4.2 Zapojení senzorů a tvorba scriptů	24
4.2.1 DS18B20	24
4.2.2 DHT22	26
4.2.3 MH-Z19	26
4.2.4 BH1750	27
4.3 Tvorba webové aplikace	29
5 Zkušební provoz a testování zařízení	31
6 Závěr	33
Literatura	36

Seznam obrázků

3.1	Pohled do výsledného zařízení	18
3.2	Vývojový diagram řídicí smyčky	19
3.3	Florarium – přední a pravý boční pohled	21
3.4	Zjednodušené elektrické schéma kompletního zařízení	22
4.1	Blokové schéma zapojení systému	28
4.2	Webová aplikace	30
5.1	Grafy teploty a CO ₂ během zkušebního provozu	31
5.2	Grafy vlhkosti a osvětlení během zkušebního provozu	32

Seznam zkratek

TUL	Technická univerzita v Liberci
FM	Fakulta mechatroniky, informatiky a mezioborových studií Technické univerzity v Liberci
OS	Operační systém
RPi	Raspberry Pi
IUCN	International Union for Conservation of Nature, Mezinárodní svaz ochrany přírody
N.	Nepenthes, láčkovka
IR	Infračervený
PWM	Pulsně šířková modulace
ADC	Analog to Digital Converter, Analogově číslicový převodník

1 Úvod

Cílem práce je navrhnout a sestavit systém, který bude imitovat podmínky horského pásu rovníku. V těchto specifických podmínkách, kde se denní teploty pohybují mezi 25–30 °C a noční teploty mezi 10–15 °C, se daří masožravým rostlinám rodu láčkovek. Některé druhy láčkovek jsou podle červeného seznamu IUCN označeny jako ohrožený druh. To znamená, že ve volné přírodě čelí vysokému riziku vyhynutí v blízké budoucnosti. V těchto klimatických podmínkách je zároveň i velmi vysoká vzdušná vlhkost, která v denních průměrech neklesá pod 68 %.

V ročníkovém projektu jsem již na tomto tématu pracoval a zjistil jsem, že není obtížné dosáhnout denních teplot a vysoké vzdušné vlhkosti, ale velmi těžko se dosahuje nízkých nočních teplot okolo 15 °C, a zároveň s ochlazováním ventilátory dochází k vysokému poklesu vzdušné vlhkosti. Bez řízení je prakticky nemožné dosáhnout, či se alespoň přiblížit, potřebným hodnotám. V běžných panelových domech je téměř konstantní teplota (22–25 °C) a vlhkost okolo 30 %, proto je potřeba celé zařízení vyvinout a sestavit. V následujících kapitolách se zaměřím na rešerši měřicích metod, které se pro tyto účely používají, včetně používaných senzorů a měřených fyzikálních veličin. Dále vytvořím návrh samotného florária a měřicího elektrického zařízení pomocí elektrického a blokového schématu. V praktické části práce budu realizovat elektrický hardware, pro který naprogramuji řídicí software a následně jej budu testovat v připraveném floráriu.

Oproti ročníkovému projektu se zde budu snažit dosáhnout lepších výsledků, přiblížit se více nočním teplotám a vlhkosti vzduchu. Zároveň budu navíc zkoumat hodnoty CO₂, které by měly být ukazatelem, jak rostliny pracují s plyny nejen během fotosyntézy, ale během celého dne. Následně budu na tyto změny reagovat, neboť rostliny pro svůj růst CO₂ potřebují. Výsledkem bude kompaktní zařízení, umístěné do plastové elektroinstalační krabice s vyvedenými konektory na senzory, akční členy a konektorem na přívodních 230 V.

2 Rešerše

2.1 Rostliny

Pro zajištění ideálních podmínek pro růst je nutné si říci, pro které druhy láčkovek je florárium určené, neboť se láčkovky řadí do tří základních skupin podle toho, v jaké nadmořské výšce se vyskytují. Mezi jednu z láčkovek, pro kterou bude florárium vytvořené, patří hybridní *Nepenthes maxima* x *N. fusca*, která se vyskytuje v nadmořské výšce od 170 do 2550 m. (Pozn. Hybridní znamená kříženec několika druhů.) Daří se jí při denních teplotách 20–32 °C a 10–22 °C v noci. Další láčkovka je také hybridní – *N. ventricosa* x *N. alata*. Té se v přírodě daří mezi 775–1950 m.n.m. při denních teplotách 23–29 °C a nočních 13–19 °C. Ideální teploty pro tyto druhy budou přes den 20–29 °C a v noci 13–19 °C. Z toho vyplývá, že budeme muset měřit teplotu s přesností $\pm 0,5$ °C a lepší [12].

Místa výskytu láčkovek se mimo jiné vyznačují velmi vysokou vzdušnou vlhkostí. Po prostudování více jak pětiletých meteorologických statistik na Borneu (Kota Kinabalu) z let 2013–1Q/2018 jsem zjistil, že nejvyšší naměřená hodnota vzdušné vlhkosti je 97,7 % z května (8.5.2016) a nejnižší naměřená hodnota vzdušné vlhkosti byla v lednu (28.1.2014) a to 67,4 %. Průměrné hodnoty se ale pohybují v rozmezí 74,5–88,8 %. Z toho důvodu by bylo žádoucí, aby se průměrné hodnoty vlhkosti vzduchu uvnitř florária pohybovaly nad 70 % [4].

2.2 Senzory fyzikálních veličin

Jelikož je třeba reagovat na teplotu, CO₂, vlhkost vzduchu a intenzitu osvětlení, budou potřeba senzory, které jsou schopny tyto veličiny snímat. S rozsahem měřené veličiny se sensorům často mění jak rozlišení a přesnost, tak cena. Proto se zaměřím především na senzory, které svou přesností, rozsahem a v neposlední řadě i cenou budou nejlepšími kandidáty na senzory k použití v mé bakalářské práci.

2.2.1 Měření teploty

Teplotu lze měřit stejně jako jiné fyzikální veličiny analogovými či digitálními senzory. V případě analogových je nutná důkladná kalibrace, odfiltrování šumu a ostatních rušivých signálů, neboť jsou na ně tato čidla velmi citlivá a výsledný signál nemusí přijít s dostatečnou přesností. Zároveň může docházet k chybám při čtení hodnot.

Digitální čidla jsou pro nás vhodnější, protože budeme pro vyčítání hodnot a jako řídicí jednotku používat Raspberry Pi, které analogové piny nemá. Digitální senzory v sobě mají zabudovaný A/D převodník, kodér a další elektroniku. Je to z důvodu vedení analogových dat na co nejkratší vzdálenost. To zajistí vyšší přesnost díky tomu, že se k výstupnímu signálu přičte minimální šum – oproti dlouhému analogovému vedení. Pokud bych chtěl i přesto používat analogové senzory, tak bych mohl využít metody Master-Slave s použitím Arduina, které má jak digitální, tak analogové vstupně výstupní piny (s desetibitovým A/D převodníkem). V tomto případě by RPi byl Master a Arduino Slave připojené přes USB či UART. To znamená, že by se všechna práce prováděla nadále na RPi a pouze by se využívaly analogové piny Arduina. Jinou variantou by bylo použití externího A/D převodníku od např. Analog Devices nebo Texas Instruments. Standardně mají rozlišení 16 bitové a často jsou i vícekanálové.

K digitálním sensorům je často potřeba připojit pull-up rezistor, který se umísťuje mezi datový vodič a zem – na rozdíl od pull-down rezistoru. Ten se umísťuje mezi datový vodič a napájení. Pull-up rezistor je zvolen tak, aby snížil elektrický proud, který teče ze zdroje el. napětí senzorem, a zajistil definovanou logickou hodnotu na výstupu senzoru. Zvolením správné hodnoty odporu zajistím správné vyčítání hodnot ze senzoru a snížím nebo odstráním možnost výčtu chybných hodnot.

Mezi analogové teplotní senzory patří například senzory řady LM35 od firmy National Semiconductor, kde je výstupem napětí, které jsou proporcionálně lineární k Celsiově stupnici a zároveň nepodporují externí kalibrování. Protože pro svou práci spotřebovávají pouze 60 μA , tak se nepatrně zahřívají (o méně jak 0,08 °C) oproti okolnímu vzduchu. Je možné je napájet 4–30 V a měří minimálně od 0 °C do +100 °C s garantovanou přesností $\pm 0,5$ °C. (Rozsah závisí na přesném typu.) [5]

Další použitelný analogový senzor by byl MCP9701T-E/LT firmy Microchip, který měří v rozsahu -40 °C až +125 °C s maximální přesností ± 2 °C (od 0 °C do 70 °C). Je možné jej napájet od 2,3 V do 5,5 V [13].

Jinou možností by byl senzor LM19 firmy Texas Instruments, kde s rostoucí teplotou klesá výstupní napětí. Rozsah má -55 °C až +130 °C a je napájen 2,4–5,5 V. Při 30 °C měří s maximální přesností $\pm 2,5$ °C, což je již velmi špatné rozlišení, zvláště pokud má na celém rozsahu deklarovanou maximální přesnost až $\pm 3,8$ °C. Tento senzor by byl prakticky nepoužitelný na moji aplikaci. Nutno dodat, že i při takto špatné přesnosti není nejlevnějším senzorem [14].

Poslední zmíněné analogové teplotní čidlo je TMP36 firmy Analog Devices. Je kalibrován přímo na stupně Celsiovy, což je výhodou, měří na rozsahu -40 °C až +125 °C s přesností ± 2 °C a je napájen 2,7–5,5 V [15].

Dále by se mezi analogové senzory řadila odporová čidla – termistory. To jsou součástky, které mění svůj odpor v závislosti na teplotě, ale ne lineárně. Proto je třeba přesně znát VA charakteristiku daného termistoru. Termistory se dělí na NTC (Negastor) a PTC (Pozistor). U NTC se vzrůstající teplotou elektrický odpor klesá a u PTC roste. Z negastorů bych vybral NTC640-12K a z pozistorů například PT100 s odporem 100 Ohmů a PT1000 s odporem 1000 Ohmů.

Mezi senzory digitální patří DS18B20 od firmy Dallas Semiconductor, který ko-

munikuje přes rozhraní 1-Wire[®]. Jsou možné dvě varianty zapojení. Třívodičové zapojení – 3v3, GND a datový vodič, a nebo dvouvodičové, tzv. s parazitním napájením, kde je spojena zem s napájením a mezi napájením a datovým vodičem je vložen pull-down rezistor s odporem 4k7. Měří teplotu od -55 °C do +125 °C s přesností ±0,5 °C na rozsahu -10 °C až +85 °C. Převádí teplotu na 12-bitové slovo každých 750 ms [9].

Další z digitálních senzorů bych vybral SMT172, který pracuje od -45 °C do +130 °C. Od -10 °C do +100 °C měří s přesností 0,25 °C a je možné jej napájet jak 3,3 V logikou, tak 5 V [16].

Dále jsem našel senzory, které již nejsou tak přesné jako například TMP03 s přesností ±1,5 °C, TMP35 s přesností ±2 °C a LM75A se stejnou přesností.

2.2.2 Měření vlhkosti vzduchu

Vlhkost vzduchu je vlastnost vzduchu, která udává, kolik vody ve formě vodní páry je obsaženo v daném množství vzduchu. Při rozdílných teplotách se míra nasycení vzduchu vodními parami liší. Čím vyšší je teplota, tím více je vzduch schopen vodní páry pojmout, ale jen do stavu, který se jmenuje rosný bod. Rosný bod je hranice, při které již vzduch nemůže pojmout více vodní páry, při dané teplotě a tlaku, to znamená, že je vzduch maximálně nasycen. Vlhkost vzduchu se rozlišuje na absolutní a relativní. Absolutní vlhkost vzduchu je poměr hmotnosti vodní páry na objemu, v němž se tato vodní pára nachází a je vyjádřena vzorcem $\Phi = \frac{m}{V}$, většinou je absolutní vlhkost vzduchu v jednotkách $g.m^{-3}$. Kdežto relativní vlhkost vzduchu je procentuální poměr, který popisuje aktuální absolutní vlhkost vzduchu ku maximální absolutní vlhkosti vzduchu při stejné teplotě. Z toho vyplývá vztah $\phi = \frac{\Phi}{\Phi_{max}} 100 \%$. Je možné jej vyjádřit i pomocí poměru hmotností. Hmotnosti vodní páry m , která je aktuálně v jednotce objemu vzduchu a hmotnosti m_{max} vodní páry při plném nasycení vzduchu o objemu stejné jednotky. V tomto případě by vztah vypadal podobně jako předchozí $\phi = \frac{m}{m_{max}} 100 \%$ [17].

Metody měření vlhkosti vzduchu jsou následující: psychometrická, hygrometrická, kondenzační a kapacitní.

U psychometrické metody je základem určení psychometrického teplotního rozdílu ze suchého a mokrého teploměru (mokrý teploměr je ochlazován změnou skupenství – odpařováním kapaliny z jeho povrchu).

Hygrometrická metoda se využívá u dilatačních hygrometrů, kde se používají hygroskopické látky, což jsou látky se schopností pohlcovat vzdušnou vlhkost. Na této metodě je založen odporový senzor vlhkosti, kde základem může být oxid hlinitý a na jeho bocích elektrody. Tím, že do sebe naváže vlhkost se sníží jeho odpor a to exponenciálně. Dále na této metodě mohou být založeny kapacitní senzory založené na hygroskopicitě dielektrik, které jsou sice lineární, ale již ne tak přesné.

Kondenzační metoda je založena na kondenzaci vodních par ze vzduchu. Senzor, který funguje na tomto principu je například senzor rosného bodu, který měří teplotu rosného bodu. Vlhkost vzduchu nebo jiné atmosféry, se určí pomocí orosení zrcátka měřicího zařízení a světlo diod, které se v zrcátku má odrážet se již neodrazí s takovou intenzitou. Čím vyšší vlhkost, tím vyšší orosení zrcátka tzn. snížení

odrazivosti. Součástí tohoto senzoru je Peltierův článek a teploměry. Zrcátka je postupně zchlazováno, v okamžiku dosažení rosného bodu se zamlží a již neodráží tak dobře, tím se spustí podmínka, sejme se teplota zrcátka – teplota rosného bodu, zároveň se změří teplota okolí a z toho se vypočítá relativní vlhkost [21].

Mezi digitální senzory relativní vlhkosti patří čidlo GY-HR002, které měří vlhkost na rozsahu 20–95 % s přesností $\leq \pm 5$ %. Což opravdu není přesné měření a bylo by vhodnější použít senzor s vyšší přesností a větším rozsahem [18].

Přesnějším senzorem je HTU21D, který měří vlhkost s přesností ± 2 % na rozsahu 5–95 % RH. Výhodou je, že komunikuje po sběrnici I2C. Piny pro připojení jsou VDD, GND a SDA s SCL pro komunikaci po I2C. Součástí tohoto senzoru je i teplotní čidlo s přesností ± 1 °C v rozsahu -30–90 °C. Zároveň je součástí měřicího modulu i 4k7 pull-up rezistor, používaný pro I2C komunikaci a integrovaný stabilizátor napětí na 3,3V. To znamená, že je modul možné připojit na 3,3V i 5V logiku [19].

Další digitální senzory vlhkosti vzduchu jsou SHT1x (SMD) a SHT7x (4pin single-in-line), který je velmi malý a velmi přesný. Varianta SHT75 měří relativní vlhkost s přesností $\pm 1,8$ % a teplotu s přesností $\pm 0,3$ °C. Vnitřní zapojení měřicího modulu je následující. Data z čidla vlhkosti a teploty projdou přes zesilovače, který zesílí slabý analogový signál. Poté jde signál do 14-bitového ADC a z něj přes modul pro 2-vodičový obousměrný přenos dat ven ze senzoru. Senzor je možné napájet od 2,4 V do 5,5 V, ale bohužel není kompatibilní se sběrnici I2C [20].

Vhodnější senzor pro naše účely je DHT22, který je vylepšenou variantou senzoru DHT11, který měřil teplotu s přesností ± 2 °C a vlhkost s přesností ± 5 %. Oproti tomu varianta DHT22 měří teplotu s 4 krát větší přesností ($\pm 0,5$ °C) a vlhkost s přesností ± 2 %. U tohoto senzoru je jako snímač vlhkosti použit polymerní kondenzátor a na měření teploty termistor. Mezi napájení na pinu 1 a data na pinu 2 je nutné vložit 4k7 pull-up rezistor, díky kterému je možné vyčítání správných hodnot. Přestože je senzor v 4-vodičovém pouzdře, používá pouze 3piny – VDD, DATA a GND. Vlhkost měří na stupnici 0–100 %, teplotu -40–+80 °C a je možné jej napájet od 3,3 V do 6 V DC [22].

2.2.3 Měření CO₂

Oxid uhličitý je plyn obsažený ve vzduchu. Je to plyn, který se účastní fotosyntézy, což je biochemický proces, při kterém se mění energie přijatá ze světla (většinou slunečního) na energii chemických vazeb v rostlinách. Ve dne (za světla) CO₂ spotřebovávají a v noci (za tmy) CO₂ produkují, ale v mnohem menší míře. Fotosyntéza probíhá v zelených částech rostlin – především v listech v chloroplastech. Tento chemický jev je možné vyjádřit rovnicí $6CO_2 + 12H_2O \Rightarrow C_6H_{12}O_6 + 6O_2 + 6H_2O$. Uvedený vztah vyjadřuje, že rostlina k fotosyntéze potřebuje CO₂ (6 molekul) a vodu (12 molekul). Tyto dvě chemické látky se vlivem světelného záření přemění na glukózu, kyslík (6 molekul) a odpaří se vodní pára (6 molekul). (Pozn. Výše uvedené hodnoty v závorkách jsou pro ideální stav.) Z glukózy má rostlina energii pro růst a vnitřní fyziologické pochody. Fotosyntéza závisí kromě světla, vody a CO₂ ještě na teplotě. Čím vyšší teplota tím

rychlejší fotosyntéza. Zvýší-li se teplota o 10 °C, tak se fotosyntéza urychlí 2–3 krát [23].

Koncentrace oxidu uhličitého v atmosféře kolísá od 350 ppm do 450 ppm. Při zvyšování CO₂ se rychlost fotosyntézy zvyšuje 2–3x až do úplného nasycení, které je u každé rostliny trochu jinde, průměrně ale okolo 1500 ppm. Při zvyšování za tuto mez se fotosyntéza zastaví. Stejně tak se může zastavit i růst rostliny, snížili se koncentrace CO₂ pod 100 ppm [23].

Senzory na měření CO₂ mohou být optické s IR senzory, kde molekuly CO₂ absorbují část IR spektra z IR zářiče, poté projdou přes filtr a dopadnou na pyrodetektor s nižší intenzitou IR záření, která je převedena na elektrický signál. Anglicky se tato metoda jmenuje NDIR – Non-dispersive-infrared. Senzor který je takto vyroben má slabé místo v tom, že při vyšší koncentraci než je maximální citlivosti už nezjistíme, o kolik maximální hodnotu přesáhla. I přesto je takovýto senzor jedním z nejpřesnějších.

Senzory CO₂ mohou být založené také na elektrochemickém principu. Zde je měřicí signál úměrný koncentraci plynu reagujícího s elektrolytem v senzoru. Elektronika poté zesílí proudový signál, který po změření odpovídá koncentraci CO₂. Takové senzory mají bohužel krátkou životnost kvůli postupné chemické změně elektrolytu s čím také souvisí častá nutnost kalibrace.

Posledním zmíněným principem měření je změna vodivosti polovodiče za přítomnosti CO₂. Jako polovodičové materiály se používají oxidy kovů (cínu, zinku, wolframu či india) [24].

Senzory, které se komerčně prodávají jsou například K30, T6613, T6713 a MH-Z19.

Senzor K30 je založen na metodě NDIR a nahřátí IR zářiče trvá méně než 1 minutu. Sensor může komunikovat buď po sériové lince UART s komunikačním protokolem ModBus, a nebo po I2C pro které je rozšířen. Je napájen 5V a měří na rozsahu 0–2000 ppm, je ale schopen měřit až do 10000 ppm [25].

Senzor T6613 měří na rozsahu 0-2000 ppm. Přesnost se liší na jednotlivých částech rozsahu. Od 400–1250 ppm měří s přesností ±30 ppm nebo ±3 % z čtené hodnoty podle toho, co je větší a na rozsahu 1250–2000 ppm vyčítá hodnoty s přesností ±5 % ze čtené hodnoty + 30 ppm. S tímto senzorem je možné komunikovat po UART s Baudrate 19200 a nebo po I2C. Signál je aktualizován každé 4 vteřiny a na úvodní nahřátí jsou potřeba 2 minuty. Vylepšenou variantou je T6713, který měří od 0 až do 5000 ppm s přesností ±30 ppm ±3 % ze čtené hodnoty. Napájení obou typů je uzpůsobeno na 4,5–5,5 V DC [26].

Dalším digitálním senzorem je MH-Z19 založený také na metodě NDIR. Vyčítání hodnot je možné buď po sběrnici UART nebo pomocí PWM. Výstup ze senzoru je lineární a má vnitřní teplotní kompenzaci. Pracuje při napájecím napětí 3,6–5,5 V. Čas potřebný pro předehřátí IR zářiče jsou 3 min a průměrně potřebuje méně jak 18 mA. Měrný rozsah je 0–5000 ppm s přesností ±(50 ppm + 5 % z čtené hodnoty) a s teplotní kompenzací [27].

2.2.4 Měření intenzity osvětlení

Intenzita osvětlení je světelný tok na plochu, značí se E a jednotkou je lx [lux]. Měření bývá prováděno na principu vnitřního fotoelektrického jevu.

Senzor může být vytvořen z fotorezistoru, fotodiody či fototranzistoru. Fotorezistor má velkou citlivost, ale nelineární charakteristiku (se zvyšující intenzitou osvětlení snižuje svůj el. odpor). U fotodiody dopadá světlo do oblasti PN přechodu a se zvyšujícím se osvětlením lineárně roste anodový proud v závěrném směru (hodnoty do 1 mA). U fototranzistoru dopadá záření do kolektorového PN přechodu a tím se otevře přechod mezi bází a emitorem. Tranzistor se otevře a prochází jím proud v řádu mA.

Senzor TSL2561 je senzor firmy Adafruit, který měří osvětlení na rozsahu 0,1–40000 lx. Je napájen 3,3 V a komunikuje po I2C. V rámci sběrnice I2C používá 7bit adresy 0x39, 0x29 a 0x49, které jsou možné vybrat pomocí jumperů [28].

Senzor BH1750 nebo pod jiným označením GY-302 je digitální 16 bitový senzor se sériovým výstupem. Senzor komunikuje po sběrnici I2C a jelikož je 16 bitový tak měří intenzitu osvětlení od 0 lx do 65535 lx s přesností 1 lx. Zapojení je možné 3 vodičové, i když je senzor 4 vodičový. Senzor je totiž schopen napájení přes sběrnici I2C a tudíž nemusíme přivádět zvlášť napájení na VDD pin. Z čidla reagujícího na intenzitu osvětlení přijde signál, který je zesílen, poté projde přes AD převodník a hodnota se odešle přes I2C [29].

3 Realizace florária

Pro bakalářskou práci jsem se rozhodl použít větší florárium oproti floráriu použitým v bakalářském projektu. Nové florárium bude 4x větší o rozměrech 50x90x35 cm. Tím bych chtěl docílit větší akumulace tepla do rašeliny a vody ve floráriu, a zároveň většího objemu vyměňovaného vzduchu.

3.1 Návrh řídicího a monitorovacího systému

Pro řízení a monitorování byl zvolen mikropočítač Raspberry Pi 3 Model B Rev 1.2, se kterým jsem pracoval v bakalářském projektu. Jeho výhody oproti jiným mikropočítačům jsou:

- Možnost připojení velkého množství periférií (4 porty USB, 40 digitálních GPIO pinů, HDMI, RJ-45)
- Možnost volby OS – př. Raspbian, Windows IoT Core, Ubuntu MATE
- ARM Cortex-A53 – Oproti předchozí verzi je o 50 % rychlejší.
- Možnost bezdrátového připojení k Wi-Fi sítím – RPi bude napevno v krabici, proto je žádoucí, aby do krabice přicházelo co nejméně vodičů.
- Velké množství literatury, podpora a uživatelská komunita.

Dle zadání má být firmware naprogramován v jazyce Python, což RPi umožňuje a je na to od instalace připraven. Syntaxe jazyka je podobná například jazyku C#, rozdíl je například v tom, že na konci prvního řádku funkcí a cyklů je dvojtečka a dále se nepoužívají složené závorky na tělo funkce/cyklu, ale text se člení do bloků odsazením.

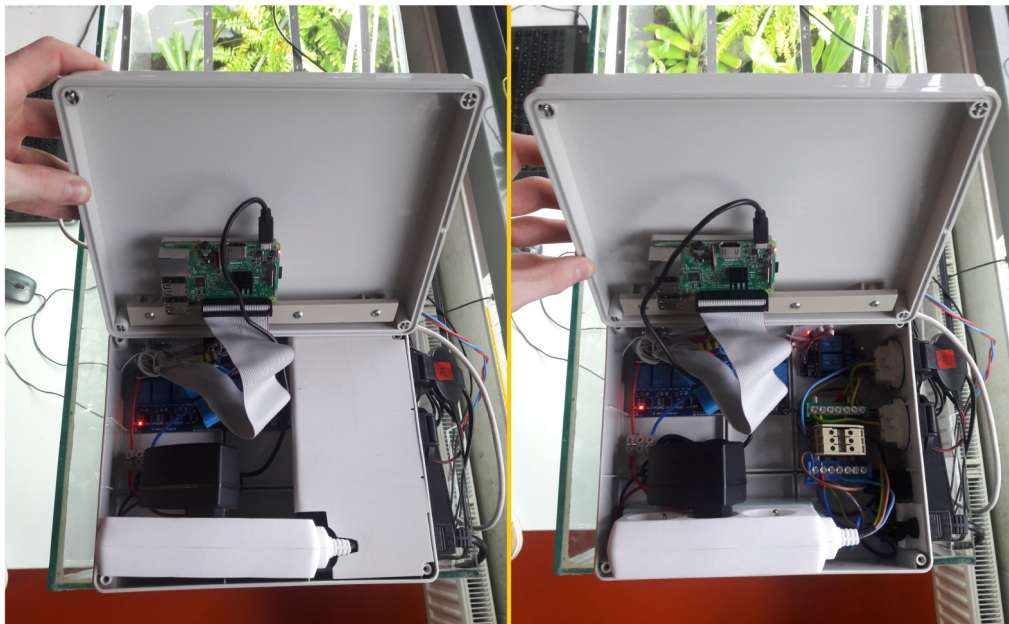
Výsledný firmware se bude skládat z několika dílčích částí. Nejprve budou vytvořeny scripty pro jednotlivé senzory a akční členy. Ty budou mít za úkol vyčítat aktuální hodnoty ze senzorů a zapsat je do souboru s příponou .json, který dále bude používat webová aplikace k získání aktuálních dat z RPi. JSON je odlehčený formát pro výměnu dat, který je snadno čitelný člověkem, umožňuje jednoduchý zápis dat a zároveň je snadno dekódovatelný počítačem. Jeho struktura bude následující. V souboru bude vytvořen jeden objekt ohraničený složenými závorkami a v něm bude vždy klíč v dvojitéch uvozovkách (např. "teplota_florarium"), následovat bude dvojtečka a za ní bude aktuální hodnota. Jednotlivé klíče s hodnotami budou

od sebe odděleny čárkou. V případě klíče "CO2_florarium" bude hodnota například 634 a v případě ventilátoru to bude 1 nebo 0 – podle toho, zda je akční člen zapnut nebo vypnut. Ke každému akčnímu členu budou napsány 2 scripty. Jeden, který zapíná akční člen a do .json souboru zapisuje 1 a druhý, který akční člen vypíná a hodnotu v .json souboru nastavuje na 0 [30]. Dále bude vytvořen program, který bude každých 15 s odesílat naměřená data na `thingspeak.com`, kde se z příchozích dat budou vykreslovat grafy.

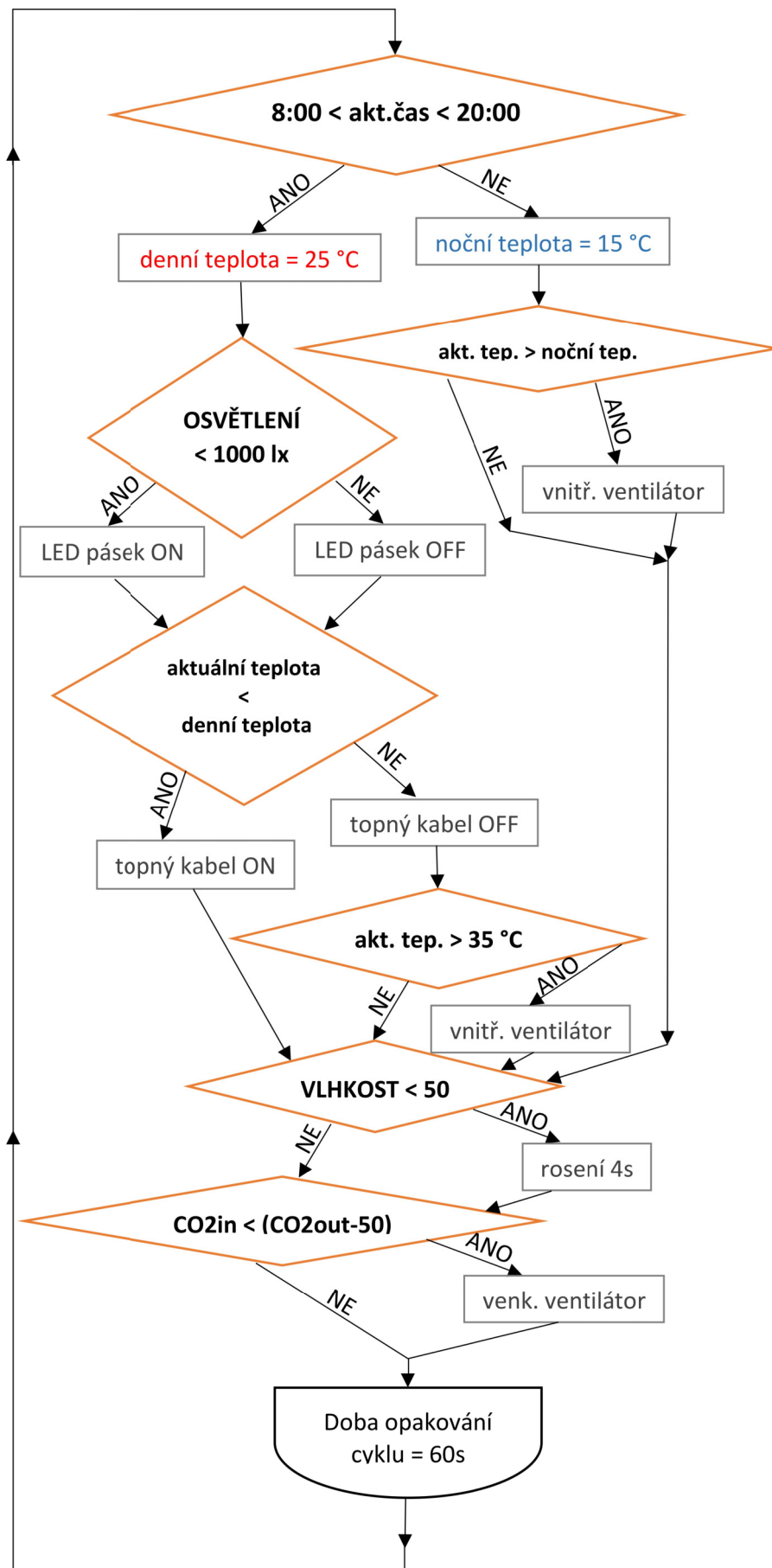
Poslední script bude určen pro řízení aktuálního stavu florária. V tomto scriptu bude smyčka, která bude každou minutu kontrolovat aktuální stav a dle potřeby bude na tyto hodnoty reagovat – zapínat nebo vypínat akční členy. Viz diagram 3.2. Na začátku programu budou naimportovány scripty, které získávají měřená data ze senzorů – viz jednotlivé scripty senzorů.

Poslední částí návrhu řídicího a monitorovacího systému vytvoření webového rozhraní. Webový server bude realizován pomocí Apache HTTP Serveru, což je open-source softwarový webový server fungující na všech platformách, a proto jej mohou zprovoznit i na RPi. Kód webových stránek bude naprogramován pomocí php a html kódu. Pomocí webové aplikace bude moct uživatel kontrolovat aktuální hodnoty, kontrolovat denní statistiku na grafech a ovládat akční členy. Po vyvolání akce uživatelem webu, aplikace pošle příkaz do RPi skrz SSH tunel a příkaz se provede. V závislosti na aktualizovaných hodnotách .json souboru web zpětně načte aktuální stav akčních členů, který se vyvoláním akce změnil.

Po vytvoření software, umístím celé zařízení do elektroinstalační krabice. Upravím ji a vytvořím do ní otvory na konektory, spínač, pojistkové pouzdro a všechny součásti propojím dle elektrického schématu 3.4. (Na fotografii 3.1 je pohled dovnitř výsledného zařízení. Povšimněte si na levé fotografii zajištění silnoproudé části proti dotyku. Vpravo je tato část po sejmutí krytu.)



Obrázek 3.1: Pohled do výsledného zařízení



Obrázek 3.2: Vývojový diagram řídicí smyčky

3.2 Montáž konstrukce florária

Pro florárium jsem použil nepoužívané skleněné akvárium, kde jsem na šestou stěnu přilepil plexisklo ve tvaru L tak, aby byla zajištěna pohodlná manipulace pro úpravy uvnitř florária. Celé florárium jsem poté natočil tak, že stěna s plexisklem je zadní strana.

Samotné lepení této strany trvalo velmi dlouho, jelikož jsem si při plánování neuvědomil, že se budu snažit spojit dva odlišné materiály – tedy, že silikon který lepí sklo, nelepí plexisklo a lepidlo, které lepí plexisklo, nelepí sklo.

Pro první lepení jsem použil silikon na sklo na bázi acetoxy silikonu. Před aplikací silikonu jsem brusným papírem zdrsnil povrch plexiskla 1 cm od okraje a k odmaštění použil isopropylalkohol. Ke sklu silikon krásně přilnul, ale na plexisklu nedržel.

Pro druhý pokus jsem použil stejný silikon, ale na plexisklo jsem aplikoval methylethylketon neboli Butanon, o kterém jsem se dočetl ve fóru reefcentral.com pod otázkou Bonding Acrylic to Glass, že by měl s tímto problémem pomoci. Následně jsem oba povrchy znovu odmastil isopropylalkoholem. V tomto případě silikon držel o poznání lépe na plexisklu, ale stále nedržel dostatečně.

Pro třetí pokus jsem použil Acrifix – lepidlo na plexisklo. Toto lepidlo mělo tu špatnou vlastnost, že nebylo pružné. Na plexisklu drželo perfektně, na některých místech skla také, ale nebylo to stoprocentní.

Před dalším lepením bylo potřeba plexisklo znovu obrousit, neboť Acrifix zatvrdl a způsoboval by netěsnosti. Znovu jsem odmastil oba lepené materiály a při tomto pokusu jsem použil Sikasil C [32], což je neutrální silikonový tmel. Je vytvořen na bázi alkoxy silikonu a díky tomu přilnul výborně jak ke sklu, tak k plexisklu. Tato volba byla nejlepší, neboť jsem neobjevil ani jedno místo, na kterém by tento alkoxy silikon nedržel.

Následně jsem na dno rozprostřel topný kabel. Tento topný kabel o výkonu 62 W odebírá 268 mA a je na střídavých 230 V. Kabel byl rozprostřen tak, aby byl rovnoměrně rozložen po celém dně se stejnými mezerami. Na něj jsem umístil dostatečnou vrstvu kamení (5 cm), aby po něm nevzlínala voda nahoru. Na kamení jsem umístil perlinku jako separační vrstvu mezi kamením s vodou a rašelinou, která přišla na úplný vrch. Separací vrstva byla umístěna také z toho důvodu, aby rašelina po zalévání nezůstávala vlhká a voda z ní mohla přes perlinku a po kamení odtéct zpět na dno.

Topný kabel byl umístěn na dně, aby ohříval vodu, kterou následně cirkuluje vodopád a předáváním teploty tak ohříval vzduch a zvyšoval zároveň vzdušnou vlhkost.

Pro vodopád bylo nutné vytvořit koryto řeky, které vodu od vodopádu odvede, aby nevlhla rašelina, do které byly umístěny rostliny. Abnormální vlhnutí substrátu je nežádoucí z důvodu možného rozšíření hub, plísní, mikroorganismů, případně i nemocí.

Po rozmístění rostlin a vytvoření koryta jsem se přesunul k zavedení vody tenkou hadicí na místo, odkud má tryskat vodopád. Po ozkoušení a úpravách koryta jsem na strop florária připevnil pomocí přísavek na sklo hadici s tryskami na rosení rostlin – umělý déšť. Obě tyto hadice byly přivedeny do technického zázemí florária,

které se nachází v pravém zadním rohu. Vše ostatní je situováno po levé straně viz foto 3.3.

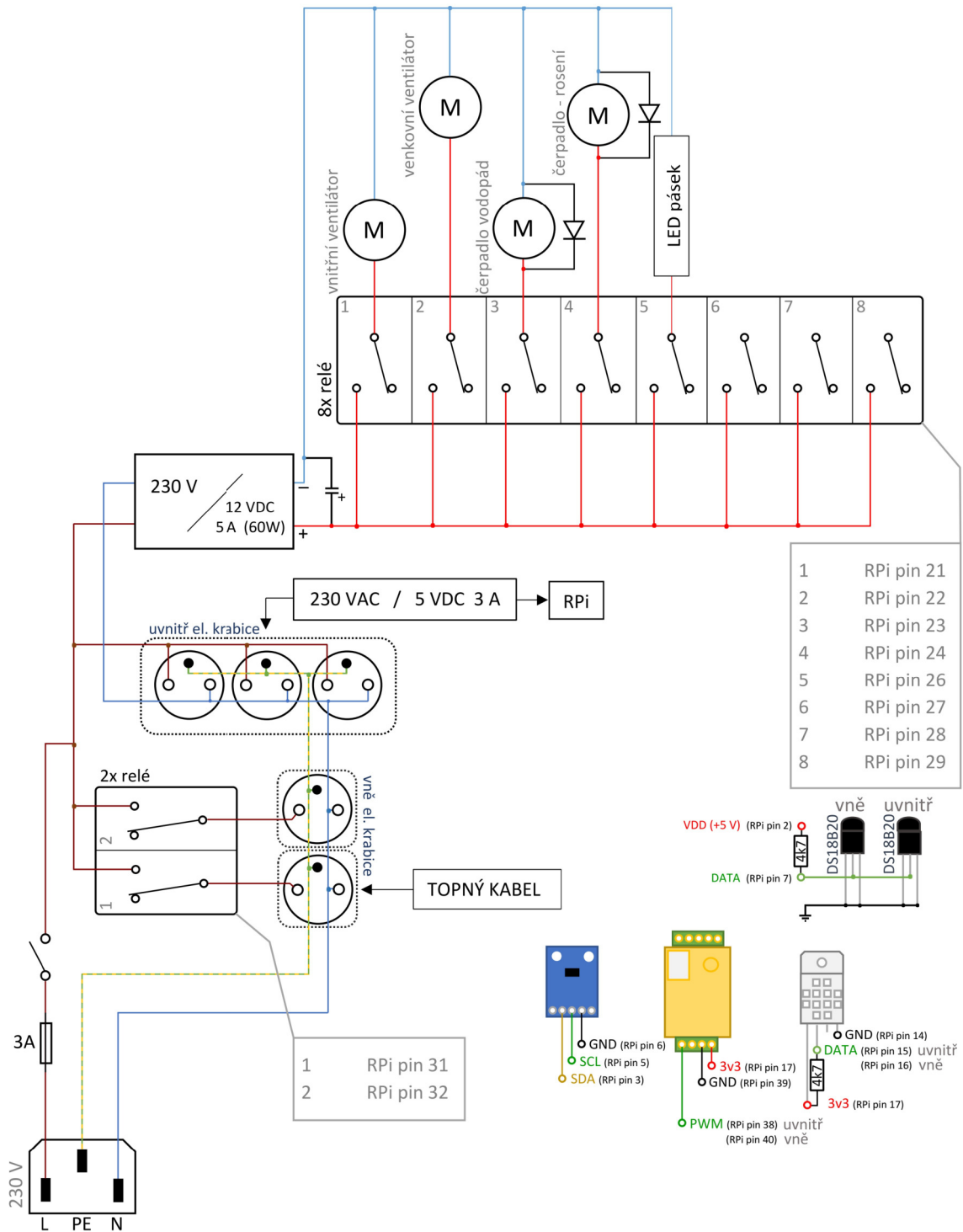


Obrázek 3.3: Florarium – přední a pravý boční pohled

Na hadici pro vodopád bylo připojeno malé ponorné čerpadlo na 12 V s výkonem 4 l/min a s odběrem 270 mA. Na zavlažování bylo použito zubové čerpadlo na 12 V, které je-li v provozu, odebírá 3 A. Je to akční člen, který v této aplikaci odebírá nejvíce proudu, proto je nutné dostatečně dimenzovat 12 V zdroj a počítat s tím i při výběru tavné pojistky, která bude chránit celé zařízení.

Další z akčních členů je osvětlení. Pro růst našich rostlin jsou nejdůležitější barviva chlorofyl A, B a karotenoidy obsažené především v zelených částech rostlin. Tyto barviva absorbují světlo při vlnové délce odpovídající modré (420-470 nm) a červené barvě (620-700 nm). Zvolen byl LED pásek studené bílé barvy o délce 2 m se 120 diodami o velikosti 5630. Tato barva byla zvolena, protože obsahuje spektrum, které rostliny využívají nejen pro svůj růst, ale i fyziologické procesy. Rostliny nepotřebují pouze zelenou složku, kterou odráží, ale nevadí jim, proto je tato barva vhodná, i když maximálně nevyužijeme celé spektrum. LED pásek je napájen 12 V a odebírá 1,94 A.

Jako akční člen na ochlazování florária jsem zvolil dva ventilátory. Jeden plochý, který zespodu nasává vzduch a bokem jej vyfukuje. Je napájen 12 V a odebírá 300 mA. Umístil jsem ho dovnitř na stropní stěnu a jeho úkolem je vyfukovat teplý vnitřní vzduch v noční době. Druhý ventilátor jsem zvolil takový, který nasává shora a vyfukuje spodem. Tento ventilátor jsem zvolil pro přívod nového, čerstvého vzduchu v případě, kdy je v místnosti vyšší obsah CO₂, než uvnitř florária a zároveň, když je teplota místnosti nižší a chci florarium ochladit. Tento ventilátor je napájen také 12 V a odebírá 250 mA.



Obrázek 3.4: Zjednodušené elektrické schéma kompletního zařízení

4 Realizace řídicího a monitorovacího systému

4.1 Nastavení řídicího mikropočítače

Před prvním použitím na RPi nic není. Proto jsem si nejprve stáhl aktuálně nejnovější NOOBS instalátor operačního systému (verze 2.4.5) z oficiálních stránek raspberrypi.org. Poté jsem rozbalený soubor nakopíroval na SD kartu o velikosti 16GB. (Pozn. Oproti dřívějším verzím RPi není nutné používat nástroj na konfiguraci SD karty od SD Association. Pro prvotní spuštění je nutné mít připojený monitor a myš nebo klávesnici. Obojí je ideální. Po vložení SD karty do RPi3 a připojení napájení se načte úvodní obrazovka, kde je výběr OS. Já jsem zvolil operační systém Raspbian. Instalace trvala 11 minut. Poté jsem nainstaloval aktualizace pomocí příkazu `sudo apt - getupdate&&sudo apt - getdist - upgrade`.

Po otázce, zda chceme instalovat aktualizace jsem napsal `y - yes` a potvrdil to klávesou `ENTER`.

Dále jsem v nastavení *RaspberryPiConfiguration* v záložce *Interfaces* zapnul SSH, VNC, I2C, Serial, 1-Wire a Remote GPIO. Bez aktivace těchto rozhraní by nebylo možné připojení k RPi ani pomocí SSH ani pomocí vzdálené plochy (VNC). Ostatní zaškrtnuté se bude používat pro jednotlivé senzory. Pro zavedení nového nastavení je potřeba zařízení restartovat.

Po znovu načtení systému je třeba udělat vnitřní nastavení samotného RPi a nakonfigurovat připojení k eduroam, protože Raspberry pi 3 nepodporuje připojení k WiFi sítím se zabezpečením WPA2-enterprise.

Jak zprovoznit eduroam pro RPi jsem hledal na stránkách liane.tul.cz/cz/Wi-Fi_Autentizace_Eduroam, ale byl jsem překvapen, že tam není žádný konfigurační nástroj, ani možnost, jak zajistit eduroam pro RPi. Zkoušel jsem varianty pro Linux, ale ani jedna z variant nebyla funkční pro RPi.

Přesně tomuto problému se věnuje článek na webu normally.online [31], a protože nikde jinde není oficiální návod, jak nastavení provést – postupoval jsem v těchto krocích. Spustil jsem Terminál a otevřel soubor `wpa_supplicant.conf` příkazem `sudo nano/etc/wpa_supplicant/wpa_supplicant.conf` a vložil do souboru novou síť:


```
network={
    ssid="eduroam"
    scan_ssid=1
    key_mgmt=WPA-EAP
    eap=PEAP
    identity="jmeno.prijmeni@tul.cz"
    password="HesloProVzdalenyPristup"
    phase1="peaplabel=0"
    phase2="auth=MSCHAPV2"
}
```

poté jsem soubor uložil a znovu načel pomocí příkazu `sudo wpa_supplicant -iwlan0 -c/etc/wpa_supplicant/wpa_supplicant.conf`. Po restartování zařízení se Raspberry Pi samo připojilo.

Abychom mohli tvořit vlastní web, který je uložený na RPi, tak je nutné nainstalovat webserver. Jako webserver jsem nainstaloval Apache2. V příkazové řádce je to možné uskutečnit příkazem `sudo apt-get install apache2 -y`

Poté je ještě potřeba doinstalovat jazyk php příkazem: `sudo apt-get install libapache2-mod-php7.0` a zařízení je připraveno k použití.

4.2 Zapojení senzorů a tvorba scriptů

V této části kapitoly se budu zabývat tím, jaké senzory jsem vybral, jak se zapojí a jakým způsobem se píše script, který z něj získává měřená data. Každý z těchto scriptů musí obsahovat funkci, kterou je možné z venku zavolat a bez spuštění celého scriptu hodnotu vyčíst. Po získání dat se hodnoty uloží do proměnných, které se budou dále využívat v hlavním – řídicím – scriptu.

Po důkladném zvážení na základě rešerše jsem se rozhodl zvolit tyto senzory:

- DS18B20 – digitální senzor teploty
- DHT22 – digitální senzor DHT22
- MH-Z19 – digitální senzor CO₂
- BH1750 – digitální senzor osvětlení

4.2.1 DS18B20

Pro tento senzor existují dvě varianty zapojení. Trívodičové zapojení – 3v3, GND a datový vodič (mezi piny DATA a 3v3 je nutné umístit pull-up rezistor o hodnotě 4k7 nebo 10k Ohmů). Druhá možnost je dvouvodičové zapojení a již je potřeba obvod napájet 5 V. Toto zapojení se nazývá tzv. s parazitním napájením, kde je spojena zem s napájením a mezi napájením a datovým vodičem je vložen odpor 4k7. Tato druhá možnost je znázorněna na blokovém schématu, protože jsem ji pro výsledné

zapojení použil. Senzor komunikuje po rozhraní 1-Wire[®] a stačí mu tedy na obsluhu i jen 2 vodiče. Datový vodič je připojen na pin 7, protože je to pin podporující komunikaci 1-Wire.

Dále i při psaní scriptu pro teplotní čidlo DS18B20 je nutné vědět, v jakém zapojení senzory jsou. V případě mého dvou vodičové zapojení je nutné změnit soubor `w1_master_pullup` na adrese `/sys/bus/w1/devices/w1_bus_master1/` na stejnojmenný soubor, ve kterém bude napsáno pouze 0 (místo původní 1). Tím se vypne vestavěný pullup rezistor a data ze senzoru se budou vyčítat správně. Tuto operaci je možné udělat pomocí příkazu:

```
cesta="/sys/bus/w1/devices/w1_bus_master1/w1_master_pullup"
with open(cesta,"a") as f:
    f.write("0")
```

Poté příkazy `os.system('modprobe w1-gpio')` a `os.system('modprobe w1-therm')` nám načtou moduly, které bude senzor využívat. Dále se nadefinuje cesta k souboru, ve kterém jsou aktuálně vyčtená data. Ten se nachází na adrese `/sys/bus/w1/devices/` a ve složce `devices` je několik složek podle toho, kolik máme připojených senzorů + výše uvedená složka `w1_bus_master1`. Každý senzor má svůj originální identifikátor, pod kterým se vytvoří i složka. Název identifikátoru pro senzory DS18B20 začínají `28-*` a pod `*` se již nachází originální číslo senzoru. Obě části adresy můžeme spojit a uložit do nové proměnné, např. s názvem adresa: `adresa=glob.glob('/sys/bus/w1/devices/'+ '28-*')[0]`. Problém může jediné nastat, v případě, že senzory nebudou připojeny, nebo když jejich čísla budou po - pokračovat stejně. V případě stejného pokračování je nutné napsat delší část adresy a nejen `28-*` (př. `28-03*`). V nově připojené složce je již soubor, ve kterém je uložena hodnota teploty v °C vynásobena tisícem – uchování desetinných míst. Poté si vytvoříme funkci `def nacteni_znaku():`. Nejprve si nahrajeme soubor s hodnotami. `f=open('/sys/bus/w1/devices/28-0416b375e5ff/w1_slave','r')`, poté si vytvoříme proměnnou `radky` a uložíme do ní načtené řádky z naší proměnné `f`. `radky=f.readlines()` poté soubor zavřeme `f.close()` a výsledek funkce dáme k dispozici `return radky`. Poté vytvoříme funkci s názvem `teplota` – `def teplota():`. Do této funkce si nahrajeme výsledek funkce `prom1=nacteni_znaku()` potom vnořeným `while` cyklem otestujeme, zda je na konci prvního řádku slovo 'YES' – `while lines[0].strip()[-3:] != 'YES':`. Poté si znovu do proměnné `prom1` uložíme hodnotu funkce `nacteni_znaku()` – `prom1=nacteni_znaku()`. Poté do proměnné `rovna_se` uložíme pozici znaku `t=` – `rovna_se=lines[1].find('t=')` poté když se `rovna_se` nerovná -1 tak se do proměnné `tep_string` uloží teplota, která je v souboru za znaky `t=` – `tep_string=prom1[1][rovna_se+2:]`. Tento řetězec znaků `string` převedeme na integer, uložíme do proměnné `prom2` a vydělíme 1000.0 – `prom2=int(tep_string)/1000.0`. Teď již máme získanou použitelnou teplotu v proměnné, ale protože senzor měří $\pm 0,5$ °C, tak je pro nás výhodné hodnotu zaokrouhlit na 1 desetinné místo – `teplota=(round(prom2,1))`. Celou funkci zakončí příkaz `return teplota`. Celý program by samozřejmě nefungoval bez naimportování knihoven: `os`, `glob` a `math`.

4.2.2 DHT22

Tento senzor jsem zvolil mimo jiné i kvůli tomu, že je to kombinované čidlo vlhkosti s čidlem teploty a zároveň mohu pomocí jediné knihovny vyčítat obě hodnoty. Výsledné zapojení senzoru ve floráriu je DATA–pin15, Vcc(3v3)–pin17 a GND–pin20. Zapojení senzoru v místnosti se liší pouze v datovém vodiči, který je přiveden na pin16. Pro vyčítání hodnot z čidla je potřeba knihovna od firmy Adafruit s názvem Adafruit_DHT, která je vytvořena pro snadnou konfiguraci senzorů DHT. Funguje i se senzory DHT11, ale ty nejsou tak přesné. Knihovnu do RPi dostaneme pomocí příkazu `git clone https://github.com/adafruit/Adafruit_Python_DHT`. Po importu knihovny příkazem `import Adafruit_DHT` je nutné vytvoření funkce, která bude s knihovnou pracovat. Funkce se bude jmenovat `SensorData()` a vytvořím ji příkazem `def SensorData():`. Uvnitř funkce bude příkaz, který do dvou proměnných `vlhkost` a `teplota` uloží hodnoty vyčtené pomocí funkce `read_retry` námi naimportované knihovny. Tato funkce má dva parametry. První určuje, která varianta senzoru se bude používat a druhý parametr (oddělený čárkou) říká, na který pin RPi budou připojena data ze senzoru:

```
vlhkost, teplota = Adafruit_DHT.read_retry(Adafruit_DHT.DHT22, 22)
```

Po tomto řádku vyčítání hodnot již následuje zakončení funkce příkazem `return` a vrácení zaokrouhlených hodnot na 1 desetinné místo.

```
return (str(round(vlhkost,1), str(round(teplota,1)))
```

4.2.3 MH-Z19

Senzor MH-Z19 je napájen 3v3, který se nachází na pinu 17. Dále je GND připojena k pinu 39 a DATA na pinu 38 v případě senzoru ve floráriu. Senzor pro měření CO₂ v místnosti je přiveden na pin 40. Data ze senzoru MH-Z19 budou vyčítat pomocí PWM výstupu. To znamená, že je signál modulován a dle datasheetu je hodnota CO₂ vyjádřena rovnicí $CO_2 = 5000 * (sirka_vze - 2ms) / (sirka_vze + sirka_ses - 4ms)$. Proměnná `sirka_vze` udává čas, po který trvá vzestupný signál (log.1) a proměnná `sirka_ses` udává čas, trvání sestupného signálu (log.0). Vyčítací script poté co toto víme, není těžké napsat [27].

Vytvořím si funkci `getPWM()` a do ní vnořím celý následující kód. Pomocí while cyklu počkám na vzestupnou hranu PWM a čas uložím do proměnné `posl_vze`, poté si počkám na sestupnou hranu a čas uložím do proměnné `posl_ses` a poté ještě jednou počkám na vzestupnou hranu a uložím hodnotu do proměnné `posl_vze`. Šířka pulzu se spočítá pomocí rovnice $sirka_vze = (posl_vze - posl_ses) * 1000$.

Poté udělám opak. Počkám si na sestupnou hranu a uložím do proměnné `posl_ses`, počkám na vzestupnou, čas uložím do `posl_vze`, znovu počkám na sestupnou hranu a výsledek uložím do proměnné `posl_ses`. V zápětí vypočítám šířku pulzu podle obdobné rovnice $sirka_ses = (posl_ses - posl_vze) * 1000$.

Poté již použijeme výše uvedenou rovnici pro vyčtení aktuálních hodnot CO₂. Výsledný kód funkce pro vyčítání dat by mohl vypadat takto:

```

def getPWM():
    pin_pwm=20
    GPIO.setmode(GPIO.BCM)
    GPIO.setup(PIN_PWM,GPIO.IN)
    time.sleep(0.2)

    while GPIO.input(pin_pwm) == 1:
        posl_vze = time.time()
    while GPIO.input(pin_pwm) == 0:
        posl_ses = time.time()
    while GPIO.input(pin_pwm) == 1:
        posl_vze = time.time()

    sirka_vze = (posl_vze - posl_ses) * 1000

    while GPIO.input(pin_pwm) == 0:
        posl_ses = time.time()
    while GPIO.input(pin_pwm) == 1:
        posl_vze = time.time()
    while GPIO.input(pin_pwm) == 0:
        posl_ses= time.time()

    sirka_ses = (posl_ses - posl_vze) * 1000

    CO2 = 5000 * ( sirka_vze - 2 ) / ( sirka_vze + sirka_ses - 4 )

    return CO2

```

4.2.4 BH1750

Tento senzor komunikuje po sběrnici I2C a proto musíme SDA senzoru připojit na SDA RPi a SCL na SCL. SDA je pin 3 a SCL je pin 5. Dále už senzoru stačí připojit jen GND na pin 6. Napájení sice na senzoru je, ale není nutné připojovat, protože si senzor potřebnou energii vezme ze sběrnice I2C. S tímto senzorem můžeme využít knihovny smbus, což je vestavěná knihovna umožňující komunikaci po I2C. Na začátku scriptu je třeba zadefinovat pár konstant z datasheetu, respektive stačí jedna a to adresa. Výchozí adresa I2C je 0x23. Další konstanty jsou neaktivní stav (Power Down) 0x00, čekání na příkaz (Power On) 0x01 a reset 0x07. Senzor má 6 módů, které definují, s jakou rychlostí a rozlišením se data vyčítají a co se děje po vyčtení hodnoty. Z datasheetu jsem zvolil režim: One Time H-Resolution Mode. V tomto módu se osvětlení vyčítá rychlostí 120 ms s přesností 1 lx a po změření hodnoty se automaticky přepíná do Power Down režimu. To se nám bude hodit z toho důvodu, že příkaz na vyčítání bude ve funkci a tak budeme šetřit energii, protože senzor vyčte jednu hodnotu a převede se do Power Down režimu do té doby, než se znovu zavolá funkce. Před definováním funkce je ještě nutné uložit do proměnné bus

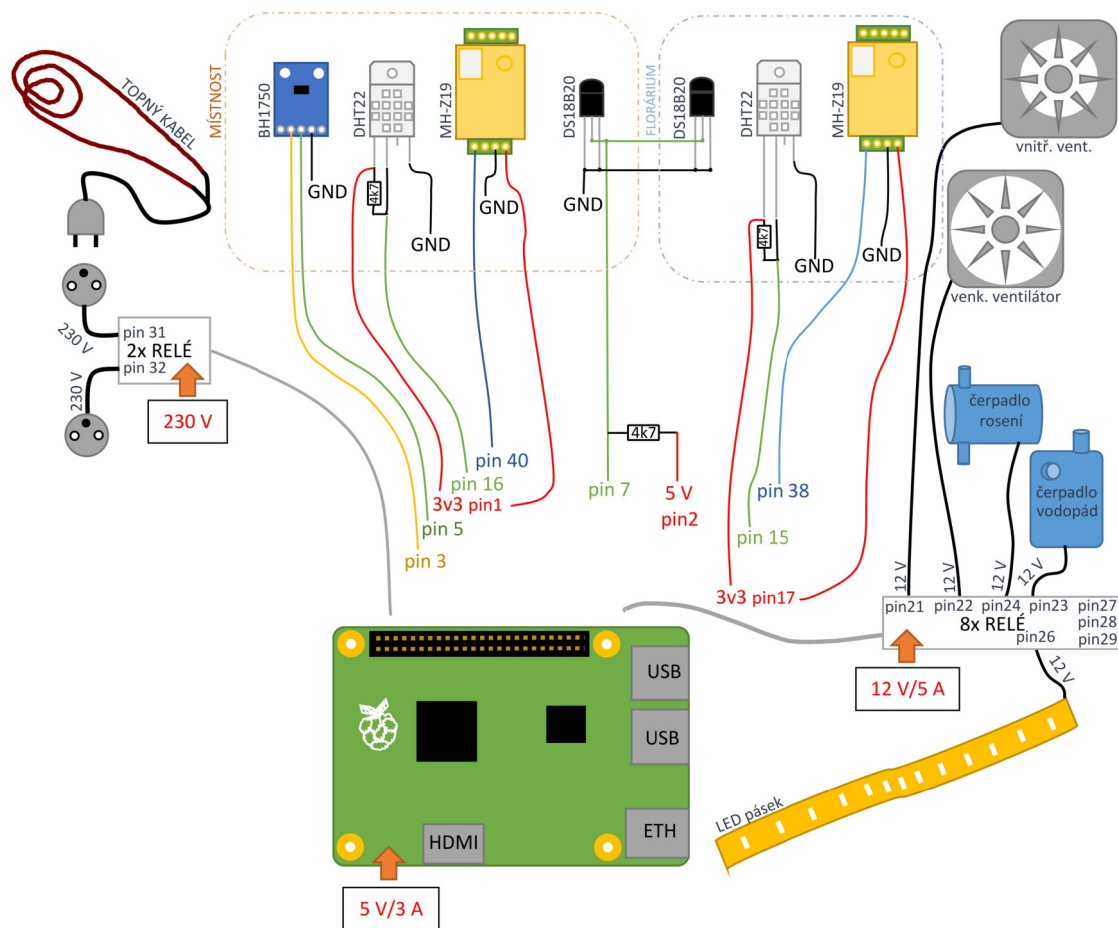
smbus.SMBus(1) (pro první verzi RPi to bylo smbus.SMBus(0)). V tomto programu budou 2 funkce. Jedna vyčte data ze senzoru a druhá je převede na číslo. Celý kód bude vypadat takto:

```
import smbus
Adresa = 0x23
ONE_TIME_HIGH_RES_MODE_1 = 0x20
bus=smbus.SMBus(1)

def CteniDat(addr=Adresa):
    data=bus.read_i2c_block_data(addr,ONE_TIME_HIGH_RES_MODE_1)
    return PrevodNaCislo(data)

def PrevodNaCislo(data):
    return round((((data[1] + (256 * data[0])))/1.2),1)

print CteniDat()
```



Obrázek 4.1: Blokové schéma zapojení systému

4.3 Tvorba webové aplikace

Webová aplikace je programována pomocí PHP jazyka, který se zpracovává na serveru a jeho výsledkem je poté HTML soubor. V HTML souboru je nadefinováno, co bude na stránce z hlediska obsahu – nadpisy, texty, obrázky,... HTML je posláno zpět ze serveru do prohlížeče ještě spolu se soubory .css, které obsahují kaskádové styly. Ty mají na starosti vzhled stránky. Spolu se soubory .css se posílají, jsou-li vytvořené, ještě soubory s příponou .js (javascript). Ty mají na starosti interaktivní prvky na stránce.

Struktura webu je následující, je vytvořen hlavní php script s názvem function.php, ve kterém jsou zadefinovány mnou vytvořené funkce, které se dále využívají v celém projektu. Jednou z funkcí je funkce pro přihlášení a posílání příkazů na RPi pomocí SSH tunelu. Tato funkce využívá knihovnu Net_SSH2 s parametry IP adresa a port. Dále se zde zadefinuje login s parametry jméno a heslo. Další funkce používá knihovnu Net_SFTP s parametrem IP adresy a funguje za účelem čtení souboru. Funkce následující je přímo pro dekodování .json souboru načteného pomocí předchozí funkce. Dále jsem vytvořil funkci, která rozhoduje podle zápisu .json souboru, zda se do RPi pošle soubor zapínající nebo vypínající akční člen.

Soubor control-panel.php má, stejně jako všechny moje php skripty, přidanou již výše uvedenou knihovnu funkcí. Dále je zde podmínka přihlášení uživatele. Pokud není, uživatel je přesměrován na stránku vytvořenou skriptem login.php. Ve skriptu control-panel.php je zadefinována struktura webové aplikace, která v tabulkách zobrazuje aktuálně vyčtená data, definuje, jak se mají tlačítka ovládání akčních členů chovat a nakonci zobrazované stránky vkládá grafy pomocí iframe z Thingspeak.com.

Poté se zde nacházejí dva soubory login.php a logout.php. Soubor login.php definuje stránku, která zobrazuje přihlašovací okno a řeší, zda bude uživatel přihlášen, případně výpis chybové hlášky – Přihlášení se nezdařilo.

Knihovnu, kterou musíme použít chceme-li pracovat přes web s RPi, je knihovna z webu phpseclib.sourceforge.net, která umožňuje práci se soubory a posílání příkazů.

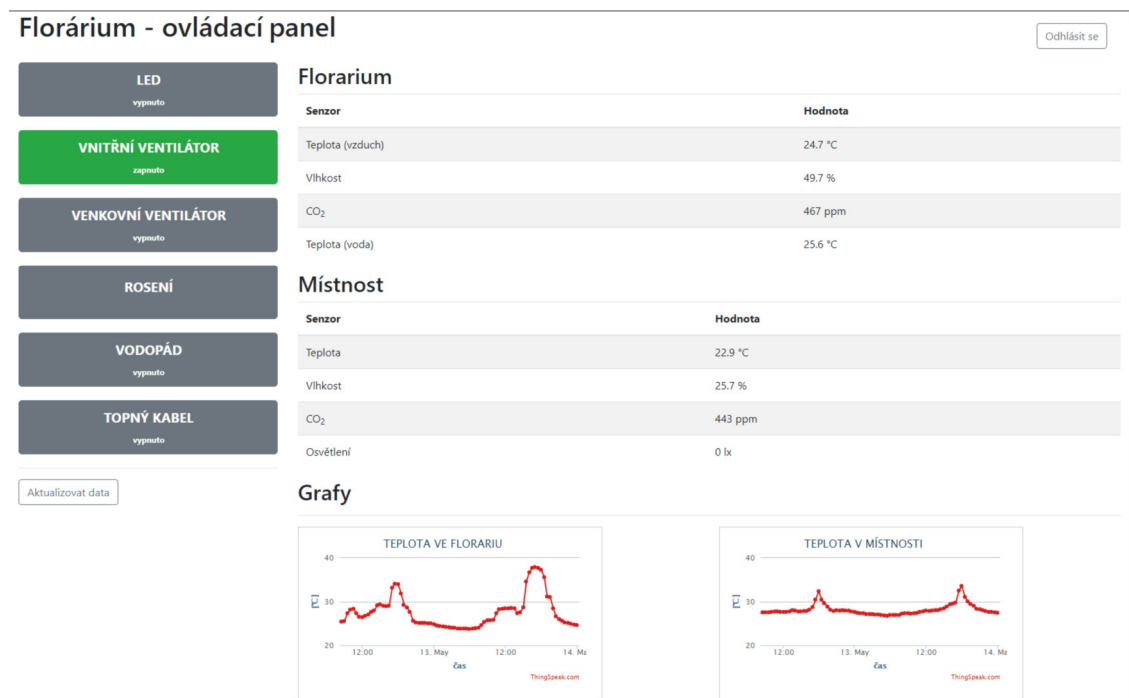
Všechny soubory k webové aplikaci jsou na RPi na adrese `/var/www/html`. Jedná se o složky s knihovnamí Bootstrap a Phpseclib. Dále se zde nachází jednotlivé .php soubory, control-panel.php, functions.php, index.php, login.php a logout.php.

Poté, co uživatel zadá do adresního řádku prohlížeče IP adresu, se použije soubor index.php. Pokud je uživatel přihlášen, zobrazí tento soubor aplikaci, která je popsána v souboru control-panel.php. Není-li uživatel přihlášen, zobrazí se přihlašovací stránka popsána v souboru login.php.

Data se mezi RPi a webem vyměňují pomocí jednoho .json souboru uloženého na RPi, ve kterém jsou uloženy informace o stavu akčních členů (zapnuto, vypnuto) a zároveň jsou v tomto souboru uložena aktuální data ze senzorů. V každém skriptu, který získá data ze senzoru, je i příkaz pro zapsání hodnoty .json souboru.

Pro lepší grafickou podobu webových stránek je implementován framework Bootstrap, což je volně stažitelná sada nástrojů, která je určena pro tvorbu webu a webových aplikací. Funguje jako knihovna pro tvorbu interaktivních prvků jako jsou tlačítka, boxy, menu, formuláře, navigační lišty a další komponenty. Jsou zde například předdefinovány tlačítka pro snadnou implementaci např. chceme-li zelené

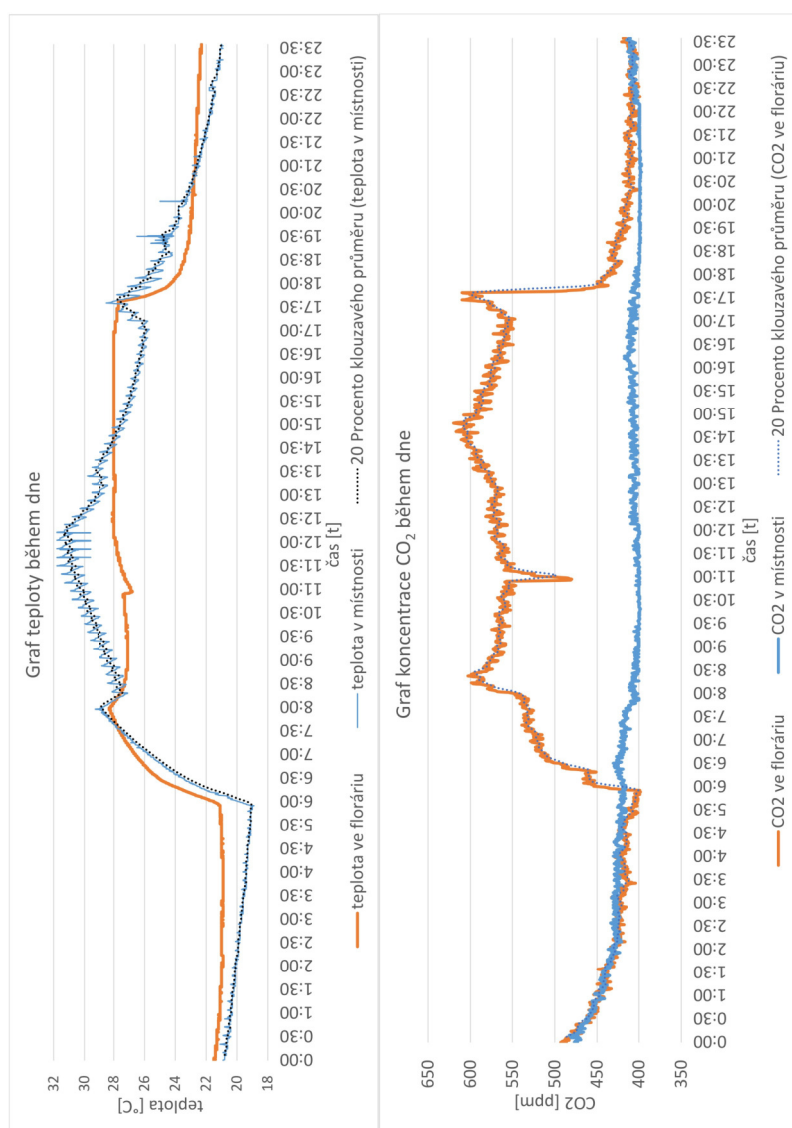
tlačítko, použijeme třídu `.btn-success`. Červené tlačítko `.btn-danger` atd. Všechny kódy, včetně knihoven, potřebné k tomuto webu jsou uvedeny na přiloženém CD.



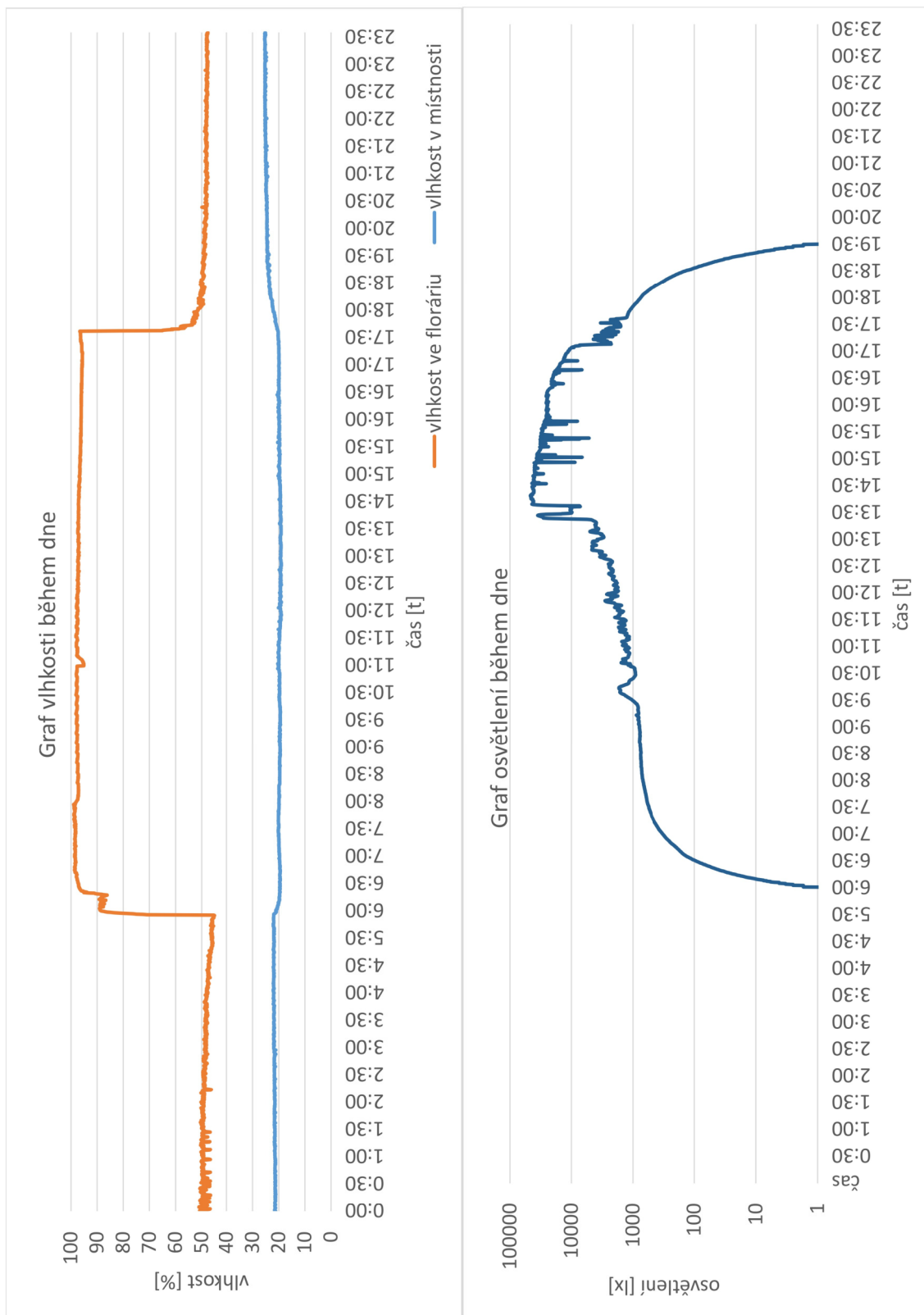
Obrázek 4.2: Webová aplikace

5 Zkušební provoz a testování zařízení

Zkušební provoz zařízení probíhal po dobu dvou měsíců, přičemž byl odladěn software a vychtány chyby, které bránily stálému provozu. Pro názornost přikládám grafy regulované soustavy, kde je patrné, jak zařízení reaguje na změnu podmínek v místnosti.



Obrázek 5.1: Grafy teploty a CO₂ během zkušebního provozu



Obrázek 5.2: Grafy vlhkosti a osvětlení během zkušebního provozu

6 Závěr

Po důkladné rešerši používaných senzorů, metod měření a postupů byl navržen systém, který je schopen monitorovat a prezentovat data ze senzorů. Dále je schopen na tyto data reagovat buď automaticky, nebo se zásahem člověka. Pro měření veličin byly zvoleny senzory CO₂ (MH-Z19), teploty (DS18B20), osvětlení (BH1750) a vlhkosti s teplotou (DHT22). Funkčnost systému byla ověřena dlouhodobým testováním. Nejprve měřením neregulovaného systému a následně regulovaného.

Hodnoty vlhkosti dosahovaly požadované výše ve dne i v noci. Přes den se hodnoty velmi blížily 100 % a v noci se držely okolo 50 %. Hodnoty koncentrace CO₂ byly v noci podobné uvnitř i vně florária, ale přes den dosahovaly hodnot až 1,5 krát vyšších než hodnoty v okolní místnosti. Jako největší problém, který se mi nepodařil vyřešit, vidím v dosažení nízkých nočních teplot. Ani oba nainstalované ventilátory nebyly schopny ochladit florárium na teplotu okolní místnosti, což bylo pravděpodobně způsobeno velkou akumulací teploty substrátem pro rostliny, přítomnou vodou a velkým objemem vyměňovaného vzduchu.

Výsledkem práce je komplexní zařízení v jedné elektroinstalační krabici s vyvedenými konektory pro senzory, akční členy a přívodní kabel na 230 V. Na RPi je web server Apache2, který umožňuje provozovat webovou aplikaci. Zde je možné po přihlášení zobrazit aktuální stav, grafy s vývojem hodnot během uplynulého dne a ovládat akční členy.

Možnost dalšího rozšíření vidím ve vytvoření efektivnějšího chlazení, možnosti zvyšovat koncentraci CO₂ pomocí bomby s elektromagnetickým ventilem, monitorování hladiny vody z důvodu doplňování vody do systému, aby nedošlo k zavzdušnění čerpadel. Dalším vylepšením projektu by mohlo být zavedení senzorů vlhkosti půdy a filtrování cirkulující vody.

Literatura

- [1] PLÍVA, Zdeněk a Jindra DRÁBKOVÁ. Metodika zpracování diplomových, bakalářských a vědeckých prací na FM TUL. Liberec: Technická univerzita, 2007. ISBN 978-80-7372-189-3.
- [2] KREIDL, Marcel. Měření teploty - senzory a měřicí obvody. BEN - technická literatura. Praha 2005. 240 s. ISBN 80-7300- 145-4
- [3] MATOUŠEK, David. Práce s mikrokontroléry ATMEL AVR ATmega16. BEN – technická literatura. Praha 2006. 320 s. ISBN 80-7300- 174-8
- [4] Kota Kinabalu Weather History: cdo date out mod. World Weather Online [online]. 2018 [cit. 2018-02-03]. Dostupné z: <https://www7.ncdc.noaa.gov/CDO/cdodateoutmod.cmd>
- [5] LM35: Precision Centigrade Temperature Sensors. In: GME [online]. National Semiconductor, 2000 [cit. 2018-05-13]. Dostupné z: <https://www.gme.cz/data/attachments/dsh.313-909.1.pdf>
- [6] Raspberry Pi – Lichtsensor BH1750 (GY-302). DomoticX [online]. EU: DomoticX, 2016 [cit. 2017-05-19]. Dostupné z: <http://domoticx.com/raspberry-pi-lichtsensor-bh1750-gy-302/>
- [7] Nepenthes rajah Hook.f. 1859. VAMR [online]. -: Darwiniana, 2007 [cit. 2017-05-23]. Dostupné z: <http://www.darwiniana.cz/vamr/?id=47&page=rostlina>
- [8] RASPBERRY PI DS18B20 TEMPERATURE SENSOR TUTORIAL. Circuit Basics [online]. -: Circuit Basics, 2016 [cit. 2017-05-19]. Dostupné z: <http://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/>
- [9] DS18B20: Programmable Resolution 1-Wire Digital Thermometer. Maxim Integrated [online]. Maxim Integrated Products, 2012 [cit. 2017-05-23]. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>
- [10] Měření teploty. WikiSkripta [online]. 2009 [cit. 2017-05-23]. Dostupné z: http://www.wikiskripta.eu/index.php/M%C4%9B%C5%99en%C3%AD_teploty
- [11] RASPBERRY PI [online]. UK: RASPBERRY PI FOUNDATION, 2011 [cit. 2017-05-23]. Dostupné z: <https://www.raspberrypi.org>

- [12] Nepenthes The Interactive Guide. Tom's Carnivores [online]. 2017 [cit. 2018-05-13]. Dostupné z: <https://www.carnivorousplants.co.uk/resources/nepenthes-interactive-guide>
- [13] MCP9700/9700A: Low-Power Linear Active Thermistor™ ICs. In: MOUSER ELECTRONICS [online]. Microchip Technology, 2014 [cit. 2018-05-13]. Dostupné z: <https://www.mouser.com/ds/2/268/20001942F-461622.pdf>
- [14] LM19 2.4V, 10 μ A, TO-92 Temperature Sensor. In: Texas Instruments [online]. 2017, 2001 [cit. 2018-05-13]. Dostupné z: <http://www.ti.com/lit/ds/symlink/lm19.pdf>
- [15] Low Voltage Temperature Sensors: TMP35/TMP36/TMP37. In: ANALOG DEVICES [online]. 2018, 2015 [cit. 2018-05-13]. Dostupné z: http://www.analog.com/media/en/technical-documentation/datasheets/TMP35_36_37.pdf
- [16] DATASHEET: SMT172. In: Smartec [online]. 2018, 2016 [cit. 2018-05-13]. Dostupné z: <http://smartec-sensors.com/cms/media/Datasheets/Temperature/SMT172-Datasheet.pdf>
- [17] Vlhkost vzduchu. Katedra fyziky Přf OU [online]. Ostrava, 2005 [cit. 2018-05-13]. Dostupné z: <http://artemis.osu.cz/Gemet/meteo2/vlhkost.htm>
- [18] GY-HR00X: Humid resistance sensor Instruction. In: PPE : Station Météo Webteo [online]. 2012 [cit. 2018-05-13]. Dostupné z: <http://webteo.free.fr/download/doc-HR102.pdf>
- [19] HTU21D(F) Sensor: Digital Relative Humidity sensor with Temperature output. In: Adafruit [online]. 2018 [cit. 2018-05-13]. Dostupné z: https://cdn-shop.adafruit.com/datasheets/1899_HTU21D.pdf
- [20] Datasheet SHT1x: Humidity and Temperature Sensor. In: Sparkfun: start something [online]. 2018, 2008 [cit. 2018-05-13]. Dostupné z: https://www.sparkfun.com/datasheets/Sensors/SHT1x_datasheet.pdf
- [21] Vlhkost vzduchu a její měření. Tzbinfo [online]. 2018, 2006 [cit. 2018-05-13]. Dostupné z: <https://vetrani.tzb-info.cz/teorie-a-vypocty-vetrani-klimatizace/3137-vlhkost-vzduchu-a-jeji-mereni>
- [22] Digital-output relative humidity & temperature sensor/-module: DHT22 (DHT22 also named as AM2302). In: Sparkfun [online]. 2018 [cit. 2018-05-13]. Dostupné z: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>
- [23] ŠEBÁNEK, Jiří. Fyziologie rostlin: vysokoškolská učebnice pro vysoké školy zemědělské. Praha: Státní zemědělské nakladatelství, 1983. Rostlinná výroba (Státní zemědělské nakladatelství).

- [24] Možnosti měření oxidu uhličitého – měřicí přístroje a čidla. In: Tzbinfo [online]. [cit. 2018-05-13]. Dostupné z: <https://vetrani.tzb-info.cz/vnitri-prostredi/13910-moznosti-mereni-oxidu-uhliciteho-merici-pristroje-a-cidla>
- [25] Datasheet: K-30 Sensor. In: Özdisan [online]. Özdisan Elektronik A.Ş., 2018 [cit. 2018-05-13]. Dostupné z: https://img.ozdisan.com/ETicaret_Dosya/456729_1584920.PDF
- [26] Teilarre 6613 CO2 Module. In: MOUSER Electronics [online]. Texas, 2014 [cit. 2018-05-13]. Dostupné z: <https://www.mouser.com/ds/2/18/AAS-920-448J-Telaire-6613-Module-091814-web-850637.pdf>
- [27] Intelligent Infrared CO2 Module: (Model: MH-Z19). In: Winsen [online]. 2015 [cit. 2018-05-13]. Dostupné z: <http://www.winsen-sensor.com/d/files/PDF/Infrared%20Gas%20Sensor/NDIR%20CO2%20SENSOR/MH-Z19%20CO2%20Ver1.0.pdf>
- [28] TSL2560, TSL2561: LIGHT-TO-DIGITAL CONVERTER. In: Adafruit [online]. 2018, 2009 [cit. 2018-05-13]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf>
- [29] Digital 16bit Serial Output Type Ambient Light Sensor IC: BH11750FVI. In: MOUSER Electronics [online]. [cit. 2018-05-13]. Dostupné z: <http://www.mouser.com/ds/2/348/bh11750fvi-e-186247.pdf>
- [30] Úvod do JSON. In: JSON [online]. [cit. 2018-05-13]. Dostupné z: <https://www.json.org/json-cz.html>
- [31] How to connect your Raspberry Pi to eduroam. Normally Online [online]. [cit. 2018-04-08]. Dostupné z: <https://normally.online/2017/07/11/how-to-connect-your-raspberry-pi-to-eduroam/>
- [32] Sikasil-C. In: TULPPA [online]. Salo, 2018 [cit. 2018-05-13]. Dostupné z: http://www.tulppa.com/files/8613/9771/6199/Sikasil_C.pdf