

Monitorovací a řídicí systém pro včelí úly

Bc. Michal Vláčil

Diplomová práce
2017

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2016/2017

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Michal Vláčil**
Osobní číslo: **A14508**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Informační technologie**
Forma studia: **prezenční**

Téma práce: **Monitorovací a řídicí systém pro včelí úl**
Téma anglicky: **A Bee-hive Monitoring and Control System**

Zásady pro vypracování:

1. Zpracujte literární rešerši k danému tématu.
2. Navrhněte systém na bázi mikrokontroleru ARDUINO pro sledování hmotnosti úlů a dalších včelařských veličin.
3. Do navrhovaného systému zapracujte ukládání dat na interní i externí úložiště.
4. Vytvořte program pro mikrokontroler s výše uvedenými funkcemi.
5. Navrhněte a prakticky ověřte programové vybavení pro dálkovou správu a vizualizaci dat.

Rozsah diplomové práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. BANZI, M. and M. SHILOH. Getting Started with Arduino: The Open Source Electronics Prototyping Platform. Vyd. 3. London: Maker Media, Incorporated, 2014. 262 s. ISBN 978-1-4493-6329-1.
2. DI JUSTO, P. and E. GERTZ. Atmospheric Monitoring with Arduino: Building Simple Devices to Collect Data about the Environment. Vyd. 1. Boston: O'Reilly, 2012. 72 s. ISBN 978-1-4493-3814-5.
3. HUGHES, J.M. Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers. Vyd. 1. Boston: O'Reilly Media, 2016. 638 s. ISBN 978-1-4919-3449-4.
4. GERTZ, E. and P. DI JUSTO. Environmental Monitoring with Arduino: Building Simple Devices to Collect Data About the World Around Us. Vyd. 1. Washington: O'Reilly Media, 2012. 100 s. ISBN 978-1-4493-2861-0.
5. BENBOW, S. The Urban Beekeeper: A Year of Bees in the City. Vyd. 1. Oxford: Random House, 2012. 304 s. ISBN 978-1-4481-3864-7.
6. KARVINEN, K. and T. KARVINEN. Getting Started with Sensors: Measure the World with Electronics, Arduino, and Raspberry Pi. Vyd. 1. San Francisco: Maker Media, Incorporated, 2014. 140 s. ISBN 978-1-4493-6704-6.
7. GRITSCH, H. Silná včelstva po celý rok. 1. Vyd. Praha: Nakladatelství brázda, s.r.o., 2010. 176 s. ISBN 978-80-209-0381-5.
8. GILFILLAN, I. Myslíme v MySQL 4: knihovna programátora. 1. Vyd. Praha: Grada Publishing a.s., 2003. 750 s. ISBN 978-80-247-0661-0.

Vedoucí diplomové práce:

Ing. Miroslav Matýsek, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání diplomové práce:

3. února 2017

Termín odevzdání diplomové práce:

16. května 2017

Ve Zlíně dne 3. února 2017

doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

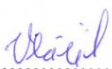
Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užit své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové/bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 15.5.2017


.....
podpis diplomanta

ABSTRAKT

Práce se zabývá tématem elektronického sledování včelstva, především jeho hmotnosti a teploty. Teoretická část se věnuje měření různých druhů veličin a systémům pro sledování včelstva na trhu.

Praktická část se zabývá návrhem a realizací vlastního systému pro monitorování včelstva, umožňujícího měřit požadované veličiny několikrát během dne. V práci jsou popsána zvolená čidla, komunikace zařízení s uživatelem a prezentace dat pomocí webové aplikace.

Klíčová slova: Úlová váha, ATmega2560, měření hmotnosti, teploty a vlhkosti, archivace dat na paměťovou kartu

ABSTRACT

The thesis deals with the topic of electronic monitoring of the bees, especially its weight and temperature. The theoretical part is devoted to the measurement of various types of quantities and systems for monitoring the bees colonies on the market.

The practical part deals with the design and realization of its own beehive monitoring system, allowing to measure the required quantities several times during the day. The work describes the selected sensors, communication of the device with the user and presentation of data using web application.

Keywords: Hive weighting machine, ATmega2560, measuring of weight, temperature and humidity, archiving of data to the memory card

Na tomto místě bych rád poděkoval vedoucímu této práce panu Ing. Miroslavu Matýskovi, Ph.D. a panu Ing. Romanovi Slavíkovi, Ph.D. za cenné rady a odbornou pomoc při řešení této práce.

Dále bych rád poděkoval celé mé rodině za její podporu při mém studiu.

OBSAH

ABSTRAKT.....	5
ABSTRACT.....	5
ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 VČELSTVO A VČELARŠTVÍ.....	11
1.1 NÁSTAVKOVÝ ÚL.....	11
2 MĚŘITELNÉ PARAMETRY VČELÍCH ÚLŮ.....	13
2.1 HMOTNOST ÚLU.....	13
2.2 MĚŘENÍ TEPLoty A VLHKOSTI.....	13
2.3 AKUSTICKÉ MĚŘENÍ.....	13
2.4 MĚŘENÍ OXIDU UHLÍČITÉHO, TLAKU A MNOŽSTVÍ SVĚTLA.....	14
3 DOSTUPNÉ ŘEŠENÍ NA TRHU.....	15
3.1 ÚLOVÁ VÁHA BEEsPY.....	15
3.2 WIRELESSBEE.....	16
3.3 BEEwISE.....	16
3.4 ÚLOVÁ VÁHA PAYA2.....	17
3.5 DIY ŘEŠENÍ VÁHY.....	18
4 POUŽITÉ TECHNOLOGIE.....	19
4.1 MySQL.....	19
4.1.1 Architektura MySQL serveru.....	19
4.1.2 Bezpečnost a optimalizace.....	19
4.2 PHP.....	20
4.3 HTML.....	20
4.4 JAVAsCRIPT.....	21
4.4.1 JQuery a JQuery UI.....	22
4.5 KASKÁDOVÉ STYLY.....	23
5 ARDUINO.....	24
5.1 HARDWARE ARDUINA.....	24
5.2 VYTVÁŘENÍ PROGRAMU V ARDUINU.....	25
II PRAKTICKÁ ČÁST.....	26
6 NÁVRH HARDWAROVÉ ČÁSTI.....	27
6.1 ARDUINO MEGA.....	28
6.2 MĚŘÍCÍ STANICE.....	29
6.3 SENZOROVÁ DESKA.....	29
6.4 NÁPÁJECÍ DESKA.....	30
6.5 BATERIE.....	30
6.6 POUŽITÉ ZPŮSOBY KOMUNIKACE.....	31
6.6.1 Sběrnice SPI.....	32
6.6.2 Sběrnice I2C.....	33

6.7	BEZDRÁTOVÝ MODUL ESP8266.....	35
6.8	OBVOD REÁLNÉHO ČASU DS3231.....	36
6.8.1	Nastavení času a alarmů.....	37
6.8.2	Schéma zapojení obvodu.....	38
6.9	PAMĚŤOVÉ MÉDIUM.....	38
6.9.1	Popis SD karty.....	38
6.9.2	Komunikace s SD kartou.....	39
6.9.3	Struktura souborů.....	39
6.10	ÚLOVÁ VÁHA.....	40
6.10.1	Tenzometrický snímač.....	40
6.10.2	Princip tenzometrických snímačů.....	41
6.10.3	Schéma zapojení tenzometru společně s AD převodníkem HX711.....	42
6.10.4	AD převodník HX711.....	42
6.10.5	Schéma převodníku HX711.....	43
6.10.6	Výroba mechanické části váhy.....	44
6.11	MĚŘENÍ TEPLoty.....	44
6.11.1	Teplotní senzor DHT22.....	45
6.12	MĚŘENÍ TLAKU.....	46
6.12.1	Senzor BMP280.....	46
6.12.2	Schéma zapojení senzoru.....	47
6.13	MĚŘENÍ SVĚTLA.....	47
6.13.1	Senzor BH1750.....	47
7	PROGRAM PRO ARDUINO.....	48
7.1	INICIALIZACE.....	48
7.2	PROBUZENÍ HODINOVÝM SIGNÁLEM.....	49
7.3	PROBUZENÍ TLAČÍTKEM.....	50
7.4	PROBUZENÍ POHYBEM.....	50
8	WEBOVÁ APLIKACE.....	55
8.1	CRON.....	56
8.2	KNIHOVNA CANVASJS.....	56
8.3	STRÁNKA FUNKCE.PHP.....	57
8.4	STRÁNKA GRAPH.PHP.....	57
8.5	DATABÁZE.....	58
	ZÁVĚR.....	59
	SEZNAM POUŽITÉ LITERATURY.....	60
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	62
	SEZNAM OBRÁZKŮ.....	63
	SEZNAM TABULEK.....	64
	SEZNAM PŘÍLOH.....	65

ÚVOD

Práce se zabývá návrhem systému pro monitorování včelstva. Výsledkem je systém měřící hmotnost úlu, teplotu a vlhkost vzduchu uvnitř a vně úlu, hodnotu atmosferického tlaku a intenzitu světla. Tato měření probíhají každou hodinu a archivují se na vnitřní SD kartu a zároveň se posílají na server, kde se ukládají do databáze. Skrze webovou aplikaci se dá dálkově ovládat česno.

Pro prezentaci dat byla vytvořena webová aplikace, která z naměřených dat generuje přehledné grafy. Na serveru je uložena i řada bezpečnostních scriptů, které se starají o kontrolu úlové váhy.

V teoretické části je krátká zmínka o včelstvu, druzích měřených hodnot v úlech a dostupných systémech pro monitorování včelstva na trhu. Praktická část se věnuje návrhu a realizaci vlastního systému v rámci práce.

I. TEORETICKÁ ČÁST

1 VČELSTVO A VČELARŠTVÍ

Včelstvo je společenství včel o jedné včele matce, několika tisících včel dělnic a v určitém období i o několika stech včel trubců. V letních měsících se v úlu může nacházet až sto tisíc včel. V zimních měsících to potom bývá od 20 do 50 tisíc včel. Počet včel je rozhodující pro posouzení, jestli jsou včelstva slabá nebo silná. Jednotlivé včely se nepočítají, ale zběžným pohledem se ze zkušenosti odhadne jejich počet. Profesionálové používají tzv. úlové váhy, které dokáží zvážit celý úl a podle tohoto údaje odhadnout množství včel a medu. Tato váha se může propojit s počítačem a pomocí získaných hodnot vytvořit grafy, na kterých bude vidět změna síly včelstva a snůšky, která označuje přísun medných a pylových zásob včelami do úlu.

Včely se dnes chovají v úlech. V minulosti lidé obcházeli dutiny stromů a med se ručně odebíral od včelstev divokých včel. Dalším způsobem bylo použití tzv. „klátů“. Duté kmeny stromů měly na sobě vyřezané obličejky a v nich chovali včelaři na zahradě svoje včely. Dnes se s touto metodou už moc nesetkáme a většinou je to pouze nostalgická záležitost.

Další typ úlu byla tzv. „košnice“ - úl vyrobený ze svazků slámy. Dnešní úly nejsou prošíkovány technologiemi nebo moderními materiály. V 80 - tých letech byly pokusy o začlenění novodobých materiálů do včelařství, ale tradiční materiály obstály na jedničku a to zvláště v těch oblastech, kde bývají tuhé zimy. Na trhu se objevily úly z polystyrenu, nebo s plastovými mezistěnami apod. Toto nemělo dlouhého trvání a tradiční materiály toto přestály [13].

1.1 Nástavkový úl

V současné době je nejpoužívanějším úlem úl nástavkový. Skládá se z částí, které se rovnají na sebe a tím vytvoří úl. Tato vlastnost dává včelaři mnoho možností k vytvoření sestav a usnadňuje práci.

Na spodku úlu je dno, nízká bedýnka bez stropu, s česnem (vstup do úlu) vyříznutým v přední části. Česno se dá většinou uzavírat pomocí dvířek nebo česnové vložky. Prostor bedýnky tvoří tzv. podmet, prostor pod plásty vlastního úlu, kde se mohou včely pohybovat.

Na dno se skládají nástavky. Ty tvoří vlastní úlový prostor, kde včely staví plásty. Na sebe se může skládat libovolné množství nástavků. Do nástavků se rovnají rámy, ve kterých

včely staví plásty. Plásty jednotlivých nástavků jsou potom oddělené a včely by neměly stavět mezi nimi.

Nástavky jsou zhora otevřené, proto je potřeba přidat ještě střechu nebo strop. Většinou jde o nějakou desku, která zakryje otvor. Tato deska se poté přikryje voděodolným materiálem, aby do úlu nezatékalo.

Do úlu se mohou vkládat ještě další díly jako mateřská mřížka, stropní krmítko nebo včelí výkluz.

Nástavkový úl je většinou vyráběn ze dřeva, ale můžeme se setkat i s jinými materiály. Včelám je celkem jedno z čeho je úl vyrobený, hlavní je trvanlivost celého úlu. Existuje řada druhů provedení. Některé jsou stlučeny hřebíčky, jiné spojené sponkami. Většinou se liší kvalitou provedení a tudíž i cenou. I nejlevnější úl ale může být dobrý. Ve výsledku je to včelám také jedno. Nástavky mohou být pouze dřevěné nebo zateplené. Obě možnosti mají svá pro a proti [9].

Nástavky mohou na sobě mít falce, výstupky nebo rantlíky, díky kterým do sebe nástavky lehce zapadnou. Hodí se to při přenášení nebo převážení úlu, kdy se nestane, že by se nám nástavky lehce sesypaly. Falce ale znemožní nástavky na sebe nasunout, musí se na sebe stavět seshora. Vzniká také problém, že takto můžeme zabít včely, které se dostanou do prostoru falců a skládáním bychom je rozmáčkli. Pokud víme, že úl nebude často převážen, je lepší se falců vyvarovat.



Obr. 1: Nástavkový úl [9]

2 MĚŘITELNÉ PARAMETRY VČELÍCH ÚLŮ

2.1 Hmotnost úlu

Hmotnost je nejdůležitější parametr, který se sleduje. Pomocí něj se dá zjistit množství medu v jarních a letních měsících. V zimních měsících pak množství zásob, které mají včely, aby přežily zimu. Při dostatečném množství dat lze vysledovat přibližný počet včel a při velkém poklesu hmotnosti situaci, kdy je úl vykraden.

2.2 Měření teploty a vlhkosti

Navíc k hmotnosti se v úlech měří teplota a vlhkost. Na základě měření teploty: v hroznu včel, na zavíčkovaném plodu, pod stropem úlu a venkovní teploty lze kontrolovat stav včelstev. Podle teploty se dá poznat nejen síla včelstva, ale dá se i odvodit budoucí vyrojení a tím předejít odnosu již nasbíraného medu a úletu roje. Také se dá rychle zjistit problém technického stavu úlu, například když je včelí úl pobořen ať už vandaly lidskými či zvířecími, nebo přírodním zásahem. Zároveň se dá měřit teplota venkovní, podle které se dá odvodit venkovní aktivita včelstev a celkově zhodnotit kvalitu stanoviště, jak je vhodné pro včely v daném období [18].

Měření vlhkosti je důležitým prvkem při sledování zdraví kolonie včel v úlech. Nejrůznější studie ukázaly, že vlhkost v úlu pod 50% způsobuje, že včely se nebudou vůbec líhnout z vajíček. Naopak vysoké vlhkosti kolem 68% až 87% zvyšují šance na tzv. „mumifikaci“ snůšky.

2.3 Akustické měření

Včelstvo v různých stavech své aktivity zní jinak. Včelí úl vydává zvuky na frekvencích 20 až 10 000 Hz. Oblast zájmu je ale pouze v rozmezí 200 až 500 Hz. Okolnosti, které vedou k rojové náladě, jsou zapříčiněny mladuškami, které přestávají krmit matku. Ta přestává klást vajíčka. Mladušky, které ztratily možnost zbavit se mateři kašičky, jsou znepokojeny a tento nepokoj vyústí v intenzivnější mávání křídly, čímž je generován silnější hluk, hlavně na frekvencích v pásmu 200 - 300 Hz. Protože počet mladušek se zvětšuje, z hluku se stává zvuk. Síla tohoto zvuku koresponduje s počtem mladušek, které nemohou krmit larvy [2].

Následující graf ukazuje co se odehrává v úle :



Obr. 2: Hluk v úle [2]

1. 21 dní před vyrojením se zvětšuje intenzita zvuku oproti normálu v okolí kmitočtu 240 Hz. Toto je ještě dlouho před tím, než včely začnou stavět matečníky. Zesílení intezity zvuku ale indikuje, že budou chtít v brzké době se stavbou začít.
2. 8 dní před rojením intezita zvuku vzrůstá s posunem maximalní frekvence na 270 Hz.
3. Jeden den před rojením je intezita zvuku na maximu a maximální frekvence kolem 300 Hz.
4. Nově usazený roj.

2.4 Měření oxidu uhličitého, tlaku a množství světla

Měření množství oxidu uhličitého v úlu se dá zjistit, jestli včely mají dostatek vzduchu a úl není moc uzavřený, pokud se například používá uzavírání česna. Při velkých koncentracích by nám včely umřely.

Měření atmosférického tlaku (a především jeho změny a rychlost těchto změn) jsou důležité pro předpověď počasí. Např. zvýšení atmosférického tlaku obvykle znamená příchod slunečného počasí s malou oblačností, zatímco pokles tlaku ohlašuje příchod oblačnosti a deštivého počasí. Pro srovnatelnost údajů se pak používá, obdobně jako u výškoměrů, tlak přepočtený na hladinu moře (QNH či QFE). Z těchto údajů zjistíme, jak se včely chovají při dobrém a špatném počasí a jaký to má vliv na výnosy medu [18].

Měření množství světla zjistíme, jak bylo slunečné počasí a jestli je úl dobře umístěn. Pokud by byly včely v místě s nedostatkem světla, nemusely by snášet dostatek medu a úl by se měl přesunout na vhodnější stanoviště. Dále se dají zjistit doby západu a východu slunce společně s časem, kdy včely vyletají z úlu.

3 DOSTUPNÉ ŘEŠENÍ NA TRHU

3.1 Úlová váha BeeSpy

Tento systém je navržený pro nepřetržité sledování stavu včelstva. Jedná se o robustní modulární systém pro celoroční použití. Hlavním cílem je mít včelstvo pod kontrolou v každém okamžiku. Díky bezdrátovým komunikačním technologiím je možné sledovat aktuální stav včelstva z osobního počítače tabletu i mobilního telefonu. Základní měřená veličina je hmotnost úlu, která dává v zimním období klíčové informace o množství zásob a v letním období o medové snůšce. Volitelně lze měřit teploty (venkovní, v úle...), relativní vlhkost, aktivitu včel (počítání průchodů česnem) a další moduly postupně přibývají.

Úlová váha se skládá z robustní kovové konstrukce, tenzometrického můstku, řídicí elektroniky a volitelně konektoru pro rozšiřující snímače (teploměry, vlhkoměr, bezpečnostní kontakt). Deska elektroniky je navržena tak, aby se bezdrátový modul dal kdykoliv přidat, vyměnit podle situace (GSM/WiFi) nebo vyjmout. Dále je součástí 3-místný LED displej pro zobrazení stavů úlové váhy a hmotnosti bez použití počítače. Váha umí měřit celkovou hmotnost i relativní (tárování) [4].

Úlové váhy je možné sdružovat v rámci jednoho stanoviště (max. 10 vah) a díky tomu je napájet z jednoho zdroje a odesílat data přes jedinou váhu do cloudové databáze. Data získaná měřením se uživateli zobrazují v přehledných grafech. Váha je dimenzovaná pro hmotnosti až do 200 kg s rozlišením lepším než 50 g.

Tato váha má široké možnosti úprav pro danou instalaci v úlech. Díky tomu, že se pracuje s jednotlivými moduly, uživatel si může pouze nakoupit věci které potřebuje a v budoucnu případně dokoupit co mu bude chybět.



Obr. 3: Váha BeeSpy [4]

3.2 WirelessBee

Projekt WirelessBee má za cíl získávat a shromažďovat naměřené hodnoty a data od jednotlivých včelstev a následně je zaznamenat a pak vyhodnotit. Včelař tedy může průběžně sledovat aktuální hodnoty nebo prohlédnout hodnoty za určité období a vytvořit si z nich pak statistiky [29].

Toto zařízení měří pouze teplotu v úlech, s rozlišením půl stupně a bezdrátově tuto informaci přeposílá do modulu, který je připojen v počítači až do vzdálenosti 600m. Napájení je realizováno pomocí baterií, jejich výdrž je až několik měsíců. Stav baterií si zařízení kontroluje samo a brzké vybití zahlásí. Zařízení používá tužkové baterie. Celý přístroj je testován v teplotách od -20°C do $+40^{\circ}\text{C}$. Hlavní výhodou je bezdrátovost, včelař může jednoduše zařízení přenášet a získávat data z různých úlů. Největší nevýhodou je, že zařízení měří právě jen teplotu a ne hmotnost, která je nejdůležitější parametr při sledování včelstva.



Obr. 4: Zařízení WirelessBee [29]

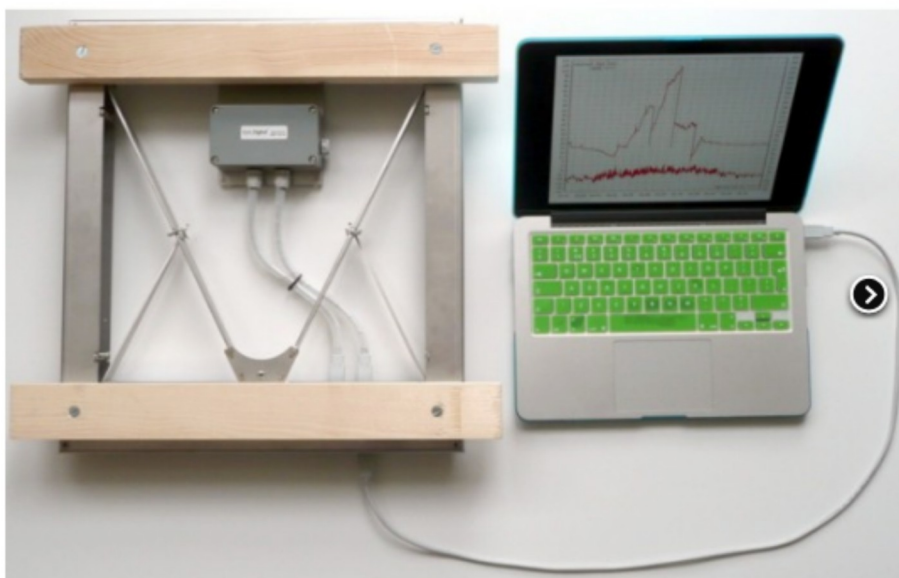
3.3 BeeWise

Jedná se o sériově vyráběný produkt původem z Francie. Současná pořizovací cena je kolem 23 000 Kč. Tento výrobek se hodí pro odloučené úly a jejich dálkové měření. Hlavní jednotka může vážit jeden či více úlů. Hodnoty hmotnosti a teploty si ukládá a poté pomocí GSM sítě přeposílá jako SMS zprávu uživateli. K měření využívá 4 tenzometrické snímače propojené mikrokontrolérem Motorola. Přesnost měření je kolem 0,2 Kg. Měření probíhá 2x denně a dokáže si ukládat až 64 měření. Zařízení je napájeno z baterie a v provozu vydrží minimálně 45 dní. Zařízení jde rozšířit o napájení pomocí solárního panelu [5].

3.4 Úlová váha PAYA2

Jde o nový model úlové váhy, který nahrazuje starší typ EMMA a PAYA. Tato verze je odlehčena a má menší základní půdorys 35 x 40 cm a výšku 4,5cm . Zachovává si všechny dobré vlastnosti svých předchůdců. Elektronika a programové vybavení dovoluje váze vydržet v provozu až 10 let na jednu výměnu baterie, a to i ve verzi s GSM přenosem. PAYA se nedá označovat jako váha „registrační“, protože v sobě dlouhodobě neuchovává naměřené hodnoty, ale přeposílá je dálkově k uživateli do počítače [26].

Váha se vkládá přímo pod úl, kde dlouhodobě snímá váhu a okolní teplotu. Váha se připojuje do připravených sestav PAYA. Jedno stanoviště se záznamníkem k sobě může mít připojeno až 31 těchto vah. Tyto váhy jsou mezi sebou propojeny pomocí stíněné dvojlinky s konektory JACK 3,5mm. Moduly mezi sebou komunikují pomocí sběrnice. Ovladač se záznamníkem má LCD displej, na kterém se dají prohlédnout naměřená data. Adaptér rozhraní USB slouží k připojení k počítači. Pro přenos dat na dálku se používá GSM modul, kde je na výběr, jestli bude přenos fungovat datově přímo do počítače nebo pomocí SMS na jiný mobil. Dokáže rozeznat minimální přírůstek 0,1 kg a maximální nosnost má 200 kg. Při zvolení připojení váhy přes kabel je cena zařízení 9 900 Kč. Další váha se dá pořídit za 5950 Kč. Řešení s bezdrátovou komunikací stojí 13 900 Kč. Další informace se dají dohledat na webu výrobce [26].



Obr. 5: Úlová váha PAYA2 [26]

3.5 DIY řešení váhy

Autorem tohoto řešení je Vlastimil Slinták. Vytvořil si vlastní váhu, která dokáže ukládat naměřená data a posílat je pomocí ZigBee na vzdálený server. Zařízení je složené ze 4 tenzometrů z osobní váhy. Jak uvádí, není to nejlepší řešení, protože tyto tenzometry jsou teplotně závislé, ale pro jeho potřeby mu vystačily. Pro měření teploty využívá teplotní čidla DS18B20. Ty chce v budoucnu rozšířit o senzory vlhkosti vzduchu. Data přenáší pomocí ZigBee 2,4Ghz na vzdálenost 70m ze zahrady do domu. Na serveru, kde se data posílají běží aplikace, která data zpracuje a vytvoří z nich grafy. Mozkem celého zařízení je ATmega328P. K ní je připojený ZigBEE modul a obvod reálného času DS2417, který používá na čítání 30 minutových intervalů. K tenzometrům v můstkovém zapojení je připojen zesilovač INA125, který generuje exitační napětí 2,5 V . Použité tenzometry mají citlivost 2mV/V, takže zesílení je nastaveno na 1500, pro maximální využití rozsahu 10b AD převodníku ATMegy. Napájení je řešeno pomocí dvou alkalických tužkových baterií a step- up měniče. Většinu času je zařízení uspané a jednou za 30 minut se probudí, aby provedlo měření a poslalo údaje. Pokud nedojde při třech pokusech k odeslání dat, tak se zahodí. V současné době nemá zařízení řešené ukládání na EEPROM nebo SD kartu [8].

Přijímací zařízení tvoří ArduinoUno s Ethernetem Shieldem a ZigBee modulem nastaveným do režimu coordinator. Toto Arduino přijme data z váhy, zabalí je jako HTTP post požadavek a odešle na server.

Při používání došel k následujícím poznatkům :

- Dávat pozor na vybraný kanál ZigBee, překrývá se totiž s Wifi.
- Šíření signálu vadí každá překážka srovnatelná s vlnovou délkou (8 cm).
- Tenzometry jsou nejpřesnější od 20 do 30°C. Při nižších teplotách roste hmotnost s klesající teplotou.
- Měla by být přidána možnost měnit intervaly měření na dálku.
- Při použití více stanovišť Arduinino Uno nestíhá zpracovávat příchozí data a posílat je na server.
- Výdrž baterií je 15 dní při intervalu měření 30minut. Ve spacím režimu se většina energie spálí na cívce step-up měniče.
- Obvod INA125 je kvalitní a dostatečně přesný.

Bližší informace se můžete dozvědět na webových stránkách pana Slintáka v [8].

4 POUŽITÉ TECHNOLOGIE

4.1 MySQL

MySQL je systém řízení báze dat uplatňující relační databázový model. Vytvořila ho švédská firma MySQL AB, kterou nyní vlastní společnost Sun Microsystems. Hlavními autory jsou Michael Widenius a David Axmark. MySQL je dostupný jak pod bezplatnou licenci GPL, tak i pod komerční placenou licenci [12].

MySQL je multiplatformní databáze. Komunikace probíhá za pomoci jazyka SQL. Je snadno implementovatelná na systémech Linux a Windows (podporuje i ostatní operační systémy). V současné době má velký podíl na používaných databázích, hlavně díky bezplatné licenci a snadnému používání. Obvykle se nasazuje v kombinaci s Apache a PHP.

MySQL bylo od počátku optimalizováno hlavně pro rychlost, kvůli tomu muselo dojít k řadě zjednodušení oproti jiným databázím.

4.1.1 Architektura MySQL serveru

Architektura je rozdělena na tři vrstvy. Vrchní obsahuje služby, obsluhují většinu potřebných nástrojů klient/server. V druhé vrstvě se nachází většina hlavních funkcí, včetně kódu pro rozbor, analýzu, optimalizaci dat. V této úrovni je veškerá funkcionální. V třetí vrstvě jsou úložné enginy. Ty se starají o ukládání a získávání dat uložených v databázi. Server komunikuje s úložnými enginy pomocí jejich API. Použitím tohoto rozhraní se zjednoduší přístup k jednotlivým enginům. Enginy mezi sebou nekomunikují, pouze odpovídají na dotazy serveru.

4.1.2 Bezpečnost a optimalizace

Každé klientské připojení dostane uvnitř procesu vlastní vlákno. Všechny dotazy tohoto připojení se odehrávají uvnitř tohoto vlákna. Server udržuje vlákna v cache, proto není potřeba vytvářet nová při každém novém připojení uživatele. Autentizace je založena na jménu, hostiteli a heslu. Existuje možnost využití certifikátů pro bezpečné připojení přes SSL [12].

Samotná databáze při přijetí dotazu provede jeho rozbor, aby vytvořila interní stromovou strukturu pro provedení optimalizací. Může samotný dotaz přepsat, aby došlo k lepšímu a efektivnějšímu výběru požadované hodnoty. Programátor může tento proces ovlivnit pod-

sunutím tzv. pokynů, které pomůžou optimalizátoru s rozhodováním. Server má možnost se před samotným rozbořením dotazu obrátit na cache dotazů, kde jsou uloženy SELECT příkazy společně s jejich výslednými sadami. Pokud je vydán příkaz, který je identický s již uloženým, server nemusí dělat rozbor nebo něco optimalizovat a rovnou předá uloženou výslednou sadu.

4.2 PHP

PHP je rekurzivní zkratkou pro označení hypertext preprocessor. Jedná se o skriptovací programovací jazyk, který je určený pro programování dynamických internetových stránek a webových aplikací ve formátu HTML, XHTML apod.

Vytváření dynamického obsahu webových stránek probíhá na straně serveru a k uživateli je přenášén výsledek této činnosti. Interpret PHP scriptu je možné volat pomocí příkazového řádku, dotazovacích metod HTTP nebo pomocí webových služeb. Syntaxe je inspirována programovacími jazyky jako Perl, C, Pascal nebo Java. PHP je nezávislé na platformě, rozdíly u operačních systémů se omezují na několik systémově závislých funkcí a scripty lze většinou bez problému přenášet bez úprav mezi platformami [21].

PHP podporuje několik knihoven pro různé účely, jako je zpracování textu, grafiky, práci se soubory a hlavně přístup k databázovým systémům typu MySQL, ODBC a dalším.

PHP je nehmotnější skriptovacím jazykem pro web na světě. V současné době má podíl 82%. Stal se oblíbeným díky své jednoduchosti a široké podpoře pro tvorbu webových aplikací. Většinou se objevuje ve spojení s webovým serverem Apache, operačním systémem Linux a některým z databázových systémů.

4.3 HTML

HTML je zkratkou pro HyperText Markup Language, jedná se o značkovací jazyk pro tvorbu webových stránek, které jsou propojeny hypertextovými odkazy. Je to hlavní jazyk pro tvorbu stránek v rámci systému World Wide Web.

Začátky HTML se datují do roku 1989, kdy dvojice Tim Berners-Lee a Robert Cailliau pracovali na propojeném informačním systému pro CERN. V té době se pro tvorbu dokumentů používaly jazyky TeX nebo PostScript. Chtěli proto vytvořit jednodušší jazyk. Společně s ním navrhli protokol HTTP pro přenos hypertextu. V roce 1991 zprovoznili svůj web a v roce 1993 byl vyvinut prohlížeč Mosaic pro počítače od IBM a Macintosh. Prohlížeč měl

velký úspěch a proto bylo nutné pro HTML definovat standardy [14].

Jazyk HTML je charakterizován množinou značek a jejich vlastností. Tento způsob byl převzat z jazyka SGML. Mezi tyto značky se vkládají části textu stránky a určováním jejich vlastností se tvoří obsah stránek. Části textu jsou uzavřeny v otevírací a ukončovací značce, většinou jde o jednoslovný název např. ``, který určí vlastnost následujícího textu. Text je pak ukončen stejnou značkou s přidáním ukončovacím lomítkem ``.

Struktura dokumentu se dá rozdělit na čtyři části. První částí je deklarace typu dokumentu. Prohlížeči se dá vědět, že otevřel HTML dokument. Druhou částí je kořenový element. Jde o značky `<html>`, `</html>` ve kterých je celý dokument. Třetí částí je hlavička dokumentu mezi tagy `<head>`. Zde můžeme definovat metadata dokumentu. Například autora, popis stránek, kódování atd. Poslední částí je vlastní tělo dokumentu uzavřené ve značkách `<body>`. Zde je veškerý obsah dokumentu.

Pro tvorbu webových stránek se používají editory. Na trhu je plno volně šiřitelných i placených. Samotný html kód můžeme psát v jakémkoliv textovém editoru, jen bude poněkud nepřehledný, pokud budeme chtít vytvořit nějaké složitější stránky nebo použít jiné jazyky. Většinou tedy používáme textové editory se zabudovanou podporou HTML a jiných jazyků, které nám programování usnadní. Existují ještě takzvané WYSIWYG editory (What you see is what you get). Touto metodou si můžeme tzv „naklikat“ obsah stránek. Stránky vytváříme tak jak je vidíme a editor za nás vytváří kód, který určí vzhled stránky pro prohlížeče. U této metody uživatel nemusí vůbec umět programovat nebo znát HTML kód.

4.4 JavaScript

JavaScript je multiplatformní scriptovací jazyk. Autorem je Brendan Eich ze společnosti Netscape. Momentálně se hlavně využívá jako interpretovaný programovací jazyk pro tvorbu webových stránek. Vkládá se přímo do HTML kódu webové stránky. JavaScript se používá pro tvorbu interaktivních prvků stránek jako jsou tlačítka, animace apod.

Syntaxe je podobná jazykům C/C++ nebo Java. JavaScript se ale od těchto jazyků sémanticky liší. Standardizován byl v roce 1998 asociací ECMA (European Computer Manufacturers) a v roce 1998 ISO (International Organization for Standardization).

JavaScript se používá jako doplněk k jazykům HTML a Java. Pro Microsoft je upraven a nazván Jscript a podporuje ho .NET [28].

Program JavaScriptu je obvykle prováděn až po načtení stránek na straně klienta, liší se tak od jazyka PHP. JavaScript má kvůli tomu bezpečnostní omezení, např. práce se soubory bez ohrožení uživatele.

4.4.1 JQuery a JQuery UI

JQuery je javascriptová knihovna, která je podporována širokou řadou prohlížečů a klade důraz na interakci mezi javascriptem a HTML.

Jquery nabízí řadu funkcí:

- výběr DOM elementů pomocí otevřeného cross-browser selektorového enginu Sizzle.
- funkce pro procházení a změnu DOM
- události
- AJAX
- manipulace s CSS
- efekty a animace
- rozšiřitelnost
-

JQuery UI je javascriptový framework zaměřený na uživatelské rozhraní. Pro vývojáře přináší řadu možností jak vytvořit funkční grafické prostředí pro webové stránky bez větších obtíží. Jednotlivým HTML prvkům může přidat funkcionalitu a efekty jednoduchým voláním identifikátorů prvků a přidáním požadovaných vlastností jako je například Droppable, který umožní prvku aby měl vlastnosti objektu drag and drop. JQuery UI je podporováno širokou řadou prohlížečů.

4.5 Kaskádové styly

Kaskádové styly jsou jazyk pro popis způsobu zobrazení elementů napsaných v jazycích HTML, XHTML nebo XML. Těmto prvkům může přidávat vlastnosti jako je barva, odsazení, v nových verzích i efekty animace.

Jazyk byl navržen organizací W3C a autor prvního návrhu byl Håkon Wium Lie. V současné době byly vydány tři verze (CSS1, CSS2 a CSS3).

Hlavním účelem kaskádových stylů je možnost oddělit vzhled dokumentu od jeho struktury a obsahu. Tuto možnost už mělo umět samotné HTML, ale kvůli nedostatečným standardům se toto nepovedlo.

Definice stylů sestává z určitého počtu pravidel. Každé pravidlo obsahuje selektor a blok deklarácí. Každý blok deklarácí je oddělený středníky a za středníkem je napsána jeho hodnota.

```
body
{
    background-color: white;
    color: black; padding:
    10px !important;
}
```

„body“ je v našem případě selektor. V závorkách jsou uvedeny deklarace vlastností daného prvku. Deklarace „color“ zbarví vybraný prvek na bílo.

5 ARDUINO

Arduino je opensource projekt pro vytváření softwaru a hardwaru uživateli, kteří navrhují a vytváří kity pro mikrokontroléry. Vytváří se tak digitální zařízení pro interakci s objekty z reálného světa. Projekt je šířen pod licencí GNU LGPL (GNU Lesser General Public License) a GNU GPL (GNU General Public License), které umožňují výrobu a vývoj softwaru komukoliv. Arduino desky jsou dostupné komerčně už poskládané nebo jako DIY (Do it yourself) kity [3].

Arduino desky používají řadu mikroprocesorů a řadičů. Desky obsahují množství I/O pinů, které mohou být připojené na celou řadu shieldů a jiných obvodů. Deska má možnost několika druhů komunikace, většiny dnes dostupných, dokonce i USB, přes který se deska programuje. Desky se programují pomocí dialektu využívajícího prvky C a C++. Arduino poskytuje vlastní IDE (Integrated Development Environment) pro vytváření programů.

Projekt Arduino vznikl v roce 2003 jako program pro studenty v institutu Ivrea v Itálii. Chtěli vytvořit nízkonákladové a snadné možnosti pro nováčky, jak vytvářet zařízení pro interakci s okolním prostředím za použití senzorů a pohonů. Příkladem mohou být jednodušší roboti, termostaty a detektory pohybu.

5.1 Hardware Arduina

Desky obsahují 8 bitové mikrokontroléry od firmy Atmel s řadou dalších podpůrných obvodů. Používají se čipy ATmega8, ATmega 168, ATmega 328, ATmega 1280 a ATmega 2560. Arduino Due obsahuje 32 bitový ARM procesor Atmel SAM3X8E. Každá deska má sadu I/O pinů, které jsou přístupné přes standardizované patice, do kterých se dají jednoduše připojit shieldy nebo jiné obvody. Na desce bývá několik LED diod, resetovací tlačítko a konektor ICSP pro možnost programování, napájecí konektor, oscilátor a obvod pro zprostředkování komunikace pomocí USB. Základní verze Arduino mají například 14 I/O digitálních pinů a 6 analogových pinů. Několik z digitálních pinů se dá použít jako PWM výstup [16].

Hlavní mikrokontrolér má už většinou nahraný bootloader (program pro základní nastavení systému po spuštění) a nastavené potřebné fuses bajty (vlastnosti čipu). Uživatel se díky tomu nemusí starat o složitější nastavování a rovnou může programovat v jazyce

podobném C/C++.

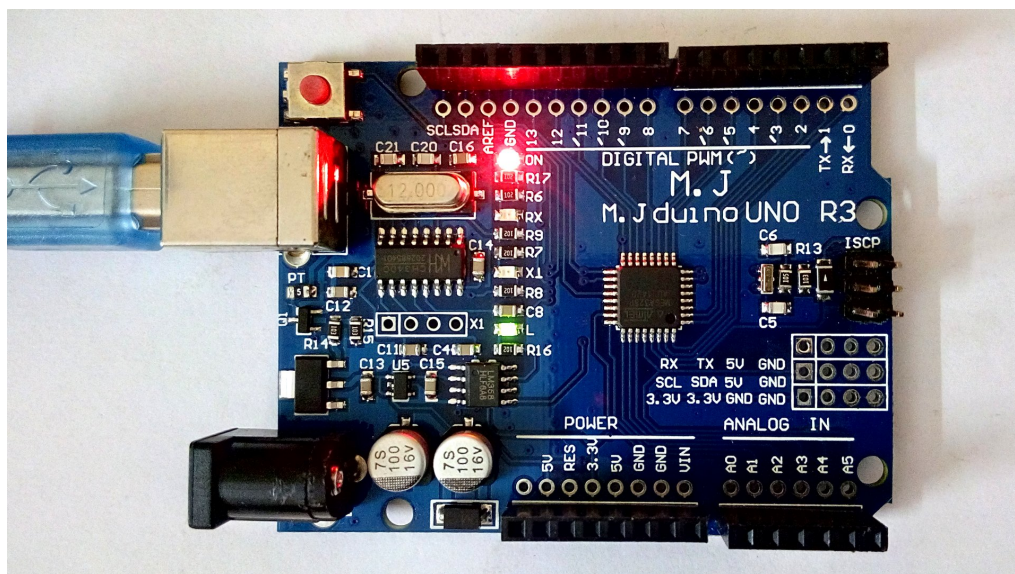
Arduino je připojeno většinou k počítači pomocí USB, ale komunikace je softwarově simulovaná přes linku RS-232. Díky otevřenosti celého projektu je na trhu celá řada klonů, které se liší cenou a kvalitou.

5.2 Vytváření programu v Arduino

Program pro Arduino může být napsán v jakémkoliv jazyce pro kompilér, který produkuje binární kód pro daný procesor. Atmel poskytuje vývojové prostředí pro jejich mikropočítače s názvem AVR Studio a novější Atmel Studio.

Arduino poskytuje prostředí Arduino IDE, který je napsaný v jazyce Java. Pochází z IDE pro jazyky Processing a Wiring. Obsahuje kódový editor, kde se dá kopírovat a vkládat, automatické odsazení, porovnání závorek a zvýraznění syntaxe. Jedním kliknutím se program zkompiluje a pošle do Arduino.

Takto napsaný program se jmenuje sketch. Na počítači se ukládají jako textový soubor s příponou .ino.



Obr. 6: Deska Arduino

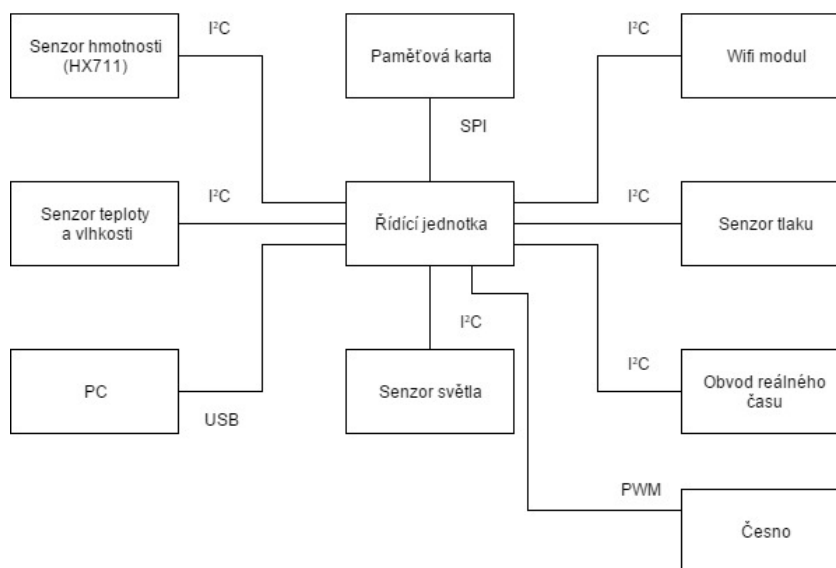
II. PRAKTICKÁ ČÁST

6 NÁVRH HARDWAROVÉ ČÁSTI

Cílem praktické části je navrhnout a sestrojít víceúčelovou úlovou váhu, která bude mít tyto vlastnosti :

- Váha bude ovládána pomocí Arduina
- Měření vnitřní a venkovní teploty
- Měření vnitřní a venkovní vlhkosti vzduchu
- Měření atmosferického tlaku a intenzity světla
- Měření váhy s přesností alespoň 0,5 kg
- Detekce pohybu se zařízením
- Odesílání dat pomocí Wifi
- Ukládání dat na SD kartu a vzdálený server
- Zobrazení dat na vzdáleném serveru
- Dálkové ovládání česna

Na základě těchto požadavků jsem vytvořil následující blokové schéma:



Obr. 7: Blokové schéma systému

Celé zařízení je postaveno na základě vývojové desky ArduinoMega. První řešení počítalo s deskou ArduinoUno, ale kvůli rozsahu práce a malé paměti pro proměnné se muselo přistoupit k použití ArduinoMega. U použití této desky jsem se obával, že bude mít moc velkou spotřebu ve spacím režimu, ale provedením malé mechanické úpravy přímo na des-

ce jsem se dostal k hodnotám, které vyhovovaly požadavkům. K Arduino jsem vyrobil senzorový shield a napájecí desku, kterou Arduino ovládá pomocí výstupních pinů. Všechna elektronika je uzavřena v plastové vodotěsné krabici. Kably jsou dovnitř přivedeny skrze průchodky, které zaručují vodotěsnost.

6.1 ArduinoMega

Jedná se o vývojovou desku založenou na mikrokontroleru Atmega2560. Má 54 digitálních vstupně/výstupních pinů, ze kterých se dá 15 použít jako PWM (pulzně šířková modulace) výstupy, 16 analogových vstupů, 4 hardwarové seriové porty, 16MHz krystalový oscilátor, USB konektor pro připojení k počítači, napájecí konektor, ICSP (In-System Programming) konektor a reset tlačítko [16].

Tab. 1. Parametry ArduinoMEGA

Mikrokontrolér	ATmega2560	Flash paměť	256 KB , 8 na bootloader
Provozní napětí	5V	SRAM	8KB
Vstupní napětí(doporučené)	7-12V	EEPROM	4KB
Vstupní napětí (maximální)	6-20V	Kmitočet	16 MHz
Digitální I/O piny	54(15 na PWM)	Pin vnitřní LED	13
Analogové vstupy	16	Délka	101,52 mm
DC proud na jeden I/O pin	20mA	Šířka	53,3 mm
DC proud pro 3.3V pin	50mA	Váha	37 g

Kvůli úspoře energie na desce při spacím režimu se přerušilo spojení mezi napájením a řadičem USB desky. Ve spacím režimu tento řadič stále odebíral proud v hodnotách kolem 20mA. Pokud se bude deska programovat přes USB kabel, je potřeba spojit speciální spojkou dva piny přidané přímo na desce, které obnoví dodávku proudu do řadiče.



Obr. 8: ArduinoMega

6.2 Měřicí stanice

Veškerá elektronika, kromě modulu HX711 ve váze, je uzavřena ve vodotěsné plastové krabici. V ní je Arduino, sensorová deska, deska napájení a baterie. Kable jsou dovnitř přivedeny skrze průchodky, ty zajistí aby se dovnitř nedostala žádná voda. Schémata plošných spojů jsou uvedena v příloze.

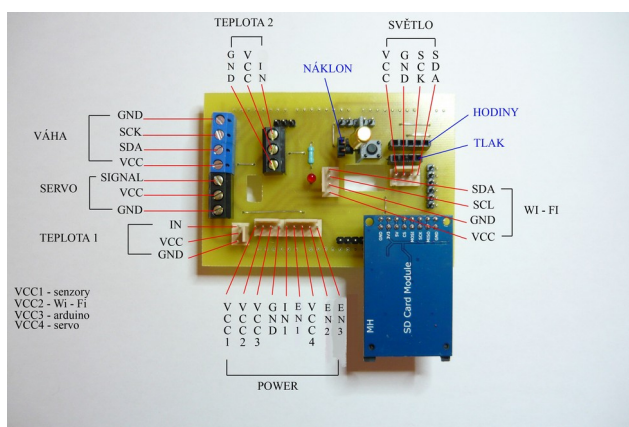


Obr. 9: Krabice měřicí stanice

6.3 Sensorová deska

Pro senzory byla vytvořena deska, do které jsou zapojena všechna čidla, wifi modul a servo. Modul hodin je napájen ze stejného obvodu jako Arduino, jsou napájeny nepřetržitě. Čidla a Wifi modul mají vlastní napájecí okruh, který se spíná dle potřeby. Poslední napájecí okruh je pro servo, které ovládá česno.

Seznam modulů a jejich připojení je uveden v tabulce v příloze.



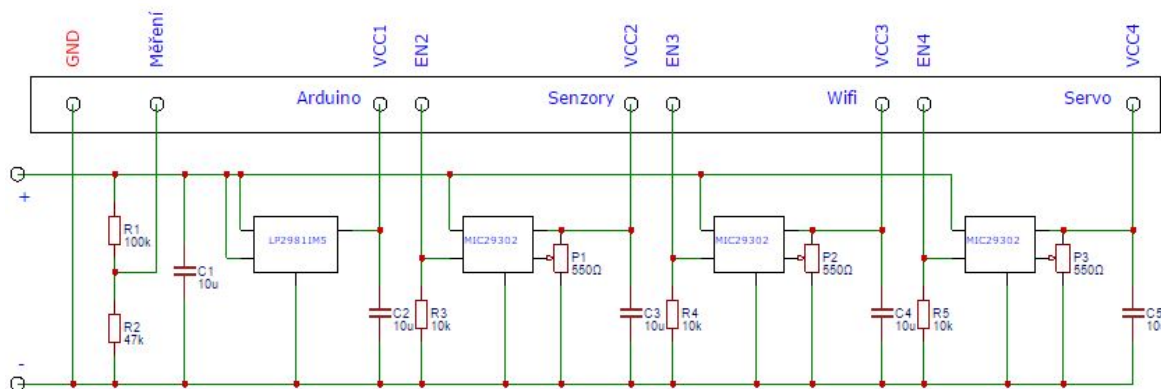
Obr. 10: Popis sensorové desky

6.4 Napájecí deska

Nejdůležitějším obvodem je napájecí deska. Ta má 4 napájecí okruhy. První se stará o napájení Arduina a hodin. Je tvořen stabilizátorem LP2981IM5, který stabilizuje napětí na úroveň 3,3V. Maximální výstupní proud je 100mA, což pro napájení Arduina bohatě stačí. Tento stabilizátor má malou vlastní spotřebu (v řádu μA) a je zapojený nepřetržitě.

Ostatní tři stabilizátory jsou spínané a regulovatelné. Jedná se o stabilizátory MIC29302. Dva z nich jsou nastaveny na 3,3V a poslední na 5V pro servo. Maximální výstupní proud je 3A a zapíná se vstupem TTL (transistor-transistor-logic) signálem z arduina. Tyto stabilizátory mají vyšší spotřebu naprázdno (kolem 10mA). Proto se spínají jen po probuzení arduina. Vypnuté stabilizátory (Low na enable vstupu) mají zanedbatelnou spotřebu.

Součástí desky je pevný dělič napětí 3:1 tvořený dvěma rezistory 47kOhm a 100kOhm, pomocí kterého se měří napětí baterie AD převodníkem arduina. Celá deska je napájena bateriemi v paralelně-sériovém zapojení. Nominální hodnota napětí je 7,4V a jsou zapojeny 2 nebo 4 baterie. Datasheety ke stabilizátorům jsou uvedeny v [15] a [25].



Obr. 11: Schéma napájecí desky

6.5 Baterie

Pro napájení celého zařízení jsou použity baterie Li-Ion (o kapacitě 3000mA). Ty se vyznačují typickým jmenovitým napětím 3,7V oproti běžnějším 1,2V u NiCd a NiMH akumulátorů. Dokáží uchovat více energie, mají nižší hmotnost, dlouhou životnost, nízké samovybíjení, nemají tzv. paměťový efekt, jsou bezpečné a nezávadné pro životní prostředí [19].

Li-Ion baterie se nabíjejí metodou CC CV (Constant Current folloed by Constant Voltage). Jde o nabíjení konstantním proudem, dokud akumulátor nedosáhne předem stanoveného napětí a poté nabíjení konstantním napětím. Ukončení nabíjení se provádí pokud nabíjecí proud klesne na předem stanovenou hodnotu, ve většině případů jde o 1,5 - 2% jmenovité kapacity.

6.6 Použité způsoby komunikace

V mé práci je použito několik druhů sériové komunikace mezi jednotlivými senzory a řídicí jednotkou. V následujících odstavcích popíšu jejich základní vlastnosti.

Sériová komunikace se u mikropočítačů používá ke dvěma základním účelům. První je přenos dat mezi jednotlivými mikropočítači. Většinou komunikace probíhá při vzdálenostech od několika metrů až po několik desítek metrů. Při delších vzdálenostech je potřeba dávat pozor na použitou kabeláž kvůli odolnosti proti rušení signálu. Nejznámější rozšířená rozhraní jsou RS232 a RS485. Ještě se můžeme setkat se specializovanými rozhraními CAN (Controller Area Network) a TTP (Time-Triggered Protocol). V poslední době se pro dálkové přesuny dat začíná používat Ethernet, který pokryje nároky na vysoké přenosové kapacity [16].

Druhým účelem je přenos dat na kratší vzdálenosti mezi integrovanými obvody nebo mezi mikropočítačovými moduly. Délka vedení tu nepřesahuje jednotky metrů. Nejčastěji se používají rozhraní SPI a I2C. Tyto sběrnice umožňují komunikaci mezi dvěma a více uzly, kde RS232 může komunikovat pouze mezi dvěma. Přenosová rychlost je více než dostatečná. SPI i I2C jsou vybaveny hodinovým signálem. U SPI může tento hodinový signál dosahovat až 70Mhz(obvykle 10Mhz) a u I2C v nejrychlejším módu 3.4MHZ.

Sériová komunikace se používá z důvodu zmenšení počtu vývodu u zařízení. Při použití sériových pamětí se zredukuje množství adresních, datových a řídicích vývodů obvyklé paměti na tři až čtyři vývody.

Další výhodou je připojení obvodů bez vyvedené vnitřní sběrnice. Funkce řadiče je pak nahrazena programově s využitím několika vývodů vhodného portu mikropočítače. U tohoto způsobu může dojít ke snížení maximalní přenosové rychlosti.

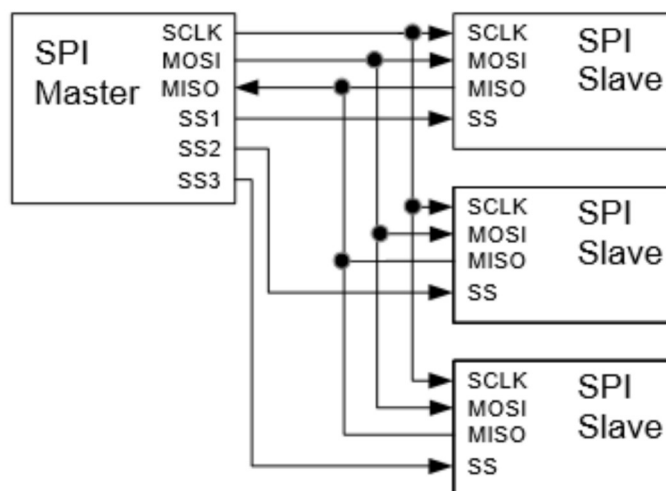
6.6.1 Sběrnice SPI

Sběrnice SPI (Serial Peripheral Interface) představuje jednu z používaných externích sériových sběrnic, sloužících pro propojení dvou či více zařízení. V tomto zapojení jeden uzel funguje jako řadič sběrnice (Master) a ostatní jsou v režimu Slave. Uzel v režimu Master obsahuje generátor hodinového signálu, který je rozveden do ostatních uzlů, čímž je umožněn zcela synchronní obousměrný přenos dat. K vodiči s hodinovým signálem se navíc přidává dvojice vodičů MISO (Master In, Slave out) a MOSI (Master Out, Slave In). Pomocí těchto vodičů se přenašejí data v obou směrech (full duplex). Poslední signál je SS (SSEL, Slave Select), který slouží pro výběr Slave zařízení na sběrnici. Všechny čtyři signály (SCK, MISO, MOSI, SS) vyžadují pouze jednosměrné porty, které přispívají k jednoduché a levné implementaci sběrnice [16].

Díky své jednoduchosti se sběrnice SPI používá například u komunikace s některými paměťmi EEPROM, LCD panely, A/D a D/A převodníky atd. Podpora SPI je zabudovaná do mnoha mikrořadičů, například AVR či PIC16xxx.

Jednou z největších předností SPI je značná jednoduchost a snadnost implementace, jednoduchost elektrického rozhraní sběrnice a přenosového protokolu. Jednoduchost spočívá v použití dvou vodičů pro komunikaci, kde odpadá nutnost přepínat mezi posílanými a odesílanými daty.

Nevýhodou této sběrnice je možnost používat jen jedno zařízení typu Master. Existuje možnost použít zapojení označované Multiple Master, ale vyžadovalo by to využití složitějšího přenosového protokolu. U SPI lze data přenášet pouze na krátké vzdálenosti kvůli použití synchronizace s neexistujícím signálem ACK (acknowledge), který by kontroloval příjem dat a dovolil měnit rychlost přenosu. Poslední větší nevýhodou je nutnost použití minimálně 4 vodičů pro obousměrnou komunikaci.



Obr. 12: SPI sběrnice

6.6.2 Sběrnice I2C

Sběrnice I2C používá společný hodinový signál (data jsou přenášena synchronně), ale naproti sběrnici SPI komunikace probíhá poloduplexně, tedy jen v jednom směru. Existuje jedno zařízení, které signál vysílá a jedno nebo více, které ho přijímá. U I2C se slave zařízení nevybírání pomocí signálu SS (Slave Select), ale každý uzel má jednoznačnou adresu. I2C má ve specifikaci jasně stanovený základní komunikační protokol. I2C je složitější než SPI, ale mnohem flexibilnější sériovou komunikací[16].

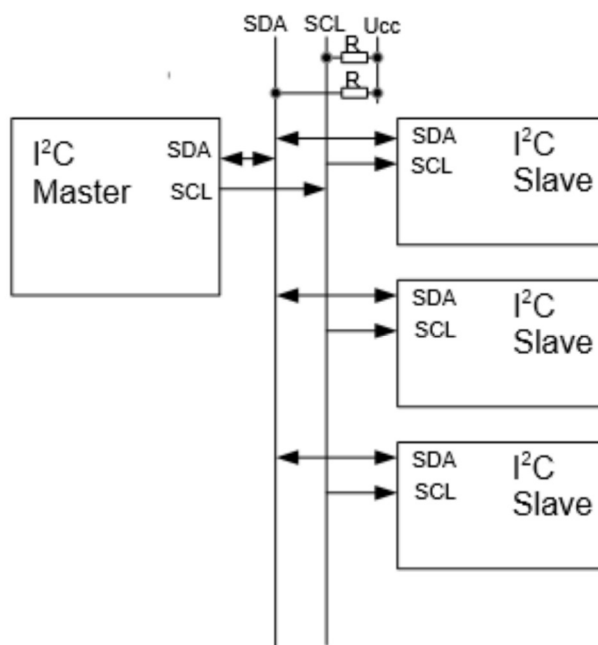
Sběrnice je tvořena dvojicí signálových vodičů. Můžeme se tedy setkat s označením TWI (Two Wire Serial Interface). První signálový vodič je označen SDA (Serial Data). Přes tento vodič probíhá veškerá obousměrná komunikace. Druhý vodič je SCK (Serial Clock), který všem zařízením posílá hodinové signály. Pak je nutné mít všechna zařízení připojena na společnou zem GND (Ground). Pro navázání úspěšné komunikace je potřeba oba vodiče SDA i SCK připojit přes pull-up rezistory o hodnotě kolem 1.5 kOhm na napájecí napětí. Toto vytváří normou stanovený klidový stav, kde jsou na vodičích logické jedničky. V tomto stavu může sběrnice zůstat libovolně dlouho. Některá zařízení mají dokonce „spací“ režimy, ze kterých se probouzí při zahájení komunikace.

O veškeré řízení komunikace na sběrnici se stará uzel typu Master. V jednu chvíli může jako Master pracovat pouze jedno zařízení, tím se zaručí, že na sběrnici nedojde ke kolizím. Zařízení v režimu Slave nemůže komunikaci řídit, ani si vyžádat start komunikace.

Zahájení komunikace probíhá snížením úrovně na datovém vodiči SD na logickou nulu, zatímco SCL si udržuje po nějaký čas stav v logické jedničce, kdy je tento čas závislý na zvolené přenosové rychlosti. Tento stav se nazývá start bit a rozpoznávají ho všechna zařízení na sběrnici. Po vyslání start bitu uzel Master začne vysílat adresu zařízení, se kterým chce komunikovat. Jedná se o sedmi nebo deseti bitovou adresu. Po adrese se ještě pošle bit, který určí jestli zařízení má přijímat nebo vysílat.

Každé zařízení typu Slave má svoji unikátní adresu. U méně připojených uzlů se používá sedmibitová adresa, přenášená v jednom bajtu, kdy poslední přenesený bit s nejmenší vahou určuje mód příjmu nebo odesílání dat. Pomocí této adresy se dá určit až 128 zařízení (2^7 adres). Toto číslo je trochu menší kvůli rezervovaným adresám. Někdy je potřeba mít více adres pro rozsáhlejší síť. Některá zařízení mají adresy pevně určené výrobcem a mají možnost použít jednu nebo více adres přivedením logické nuly na speciální pin.

I²C sběrnice má několik standardních rychlostí komunikace. Většina zařízení komunikuje rychlostí 10 kbps, 100 kbps a 400 kbps. Pak je tu možnost použití 1 Mbps a 3,4 Mbps, ale tyto rychlosti nejsou prozatím podporovány ve větší míře.



Obr. 13: I²C sběrnice

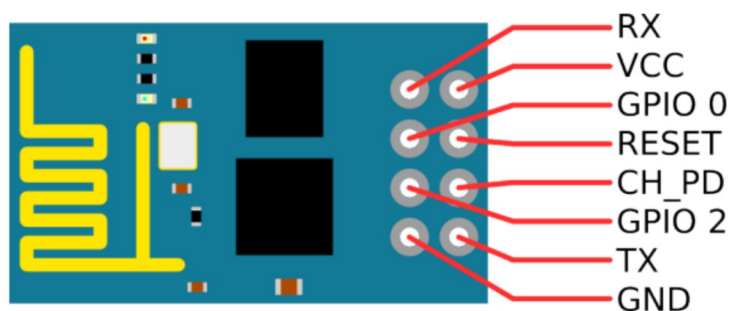
6.7 Bezdrátový modul ESP8266

Tento modul se stará o bezdrátovou komunikaci mezi Arduinem a serverem pomocí technologie Wifi. Jde o jeden z nejpoužívanějších a nejlevnějších modulů na trhu. Obsahuje 1MB FLASH paměť. Má 32-bitový RISC procesor Tensilica Xtensa, který běží na křemíku 80MHz (dokáže i 160MHz), 64KB paměť RAM pro instrukce a 96 KB paměť RAM pro data. Existuje velká řada provedení. Ten který používám já má vyvedeny sériové signály Rx, Tx, které se používají pro posílání příkazů a dat z Arduina a naopak. V Arduinu je pro komunikaci s modulem přiřazen jeden hardwarový sériový port. Dále má dva GPIO piny, které se dají použít pro připojení různých čidel. Poslední dva důležité piny jsou CH_PD a RST. První se stará o povolení činnosti modulu a druhý slouží jako reset tlačítko. Je doporučeno modul napájet 3.3V, ale při zkoušení jsem zjistil, že nemá problém ani s 5V. V klidovém stavu odebírá modul kolem 20 - 40mA, při komunikaci se vyšplhá až na 170-220mA v závislosti na vzdálenosti od AP.

Pro komunikaci s modulem se používá sada AT příkazů (ze slova attention). Tyto příkazy nám umožní vytvořit připojení na server, nastavit rychlost komunikace, zjistit AP body v dosahu apod. V tab.2 uvádím příkazy, které jsem použil v programu. Datasheet a knihovna k ovládání modulu jsou uvedeny v [11] a [27].

Tab. 2. AT příkazy

Příkaz	Popis
AT	Test modulu
AT+RST	Restartování modulu
AT+GMR	Zjištění firmwaru
AT+CWLAP	AP v dosahu
AT+CWJAP	Připojení k AP
AT+CIFSR	Zjištění IP adresy
AT+CWMODE=?	Zvolení módu pro přenos
AT+CIPSTART	Navázání TCP nebo UDP spojení
AT+CIPSEND	Odeslání dat



Obr. 14: Wifi modul ESP8266

6.8 Obvod reálného času DS3231

Pro získávání přesného času pro Arduino a vytváření probouzacího signálu používám obvod DS3231. Jedná se o levný a přesný obvod reálného času s integrovaným teplotně kompenzovaným krystalovým oscilátorem (TCXO). Zařízení má záložní baterii, která se stará o funkčnost hodin po odpojení napájení. Oscilátor pracuje na kmitočtu 32MHz. Odchylna času je +/- 2minuty za rok při provozních teplotách -40°C až 80°C. Obvod má dva programovatelné alarmy a programovatelný signálový výstup. Alarmy mohou přes výstup INT/SQW posílat probouzací signál třeba do Arduino. Kalendář a čas poskytuje informace o vteřinách, minutách, hodinách, dnech, datu, měsíci a roku. Datum na konci měsíce se automaticky upravuje pro měsíce kratší jak 31 dní, společně s korekcí pro přestupné roky. Čas se zobrazuje buď ve 24 hodinovém formátu nebo 12 hodinovém formátu s AM/PM indikátorem. Komunikace probíhá přes I2C sběrnici.

Teplotně kompenzovaný referenční napěťový obvod a komparátor sleduje hodnoty vstupního napětí a zjišťuje problémy s napájením a případně se automaticky přepne na záložní napájení. Datasheet a knihovna k ovládání modulu jsou uvedeny v [20] a [10].



Obr. 15: RTC DS3231

Parametry obvodu jsou uvedeny v Tab.3.

Tab. 3. Parametry DS3231

Provozní napětí	3.3 - 5.5V	Paměťový čip	AT24C32 (32 KB)
Použitý čip	DS3231	Sběrnice	I2C (400KHz)
Přesnost	+ - 2 minuty za rok	Baterie	LIR2032
Provozní teplota	0°C - 40 °C	Rozměry	38x22x14mm

6.8.1 Nastavení času a alarmů

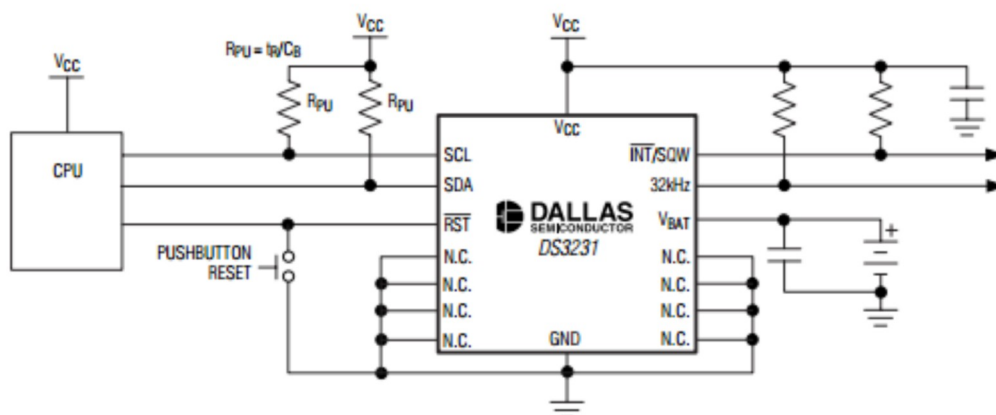
Pro nastavení času, alarmu a interruptů je potřeba změnit hodnoty v odpovídajících registrech. Čas se nastavuje v registrech od adresy 00H po 06H. Nastavením jednotlivých bitů na těchto adresách si nastavíme přesný čas. Pro nastavení času na prvním alarmu se používají registry od adres 07H po 0AH a pro druhý alarm od adres 0BH až 0DH. První alarm má rozlišení po sekundách, druhý pouze po minutách. Jednotlivé alarmy se dají nastavit na přesný čas kdy dojde k probuzení nebo na periodu opakování, například každou hodinu, nebo každý den atd. Více o ostatních registrech v [20].

Pro povolení přerušení pomocí alarmu je nutné nastavit na adrese 0EH 3.bit na hodnotu log. 1. Tímto nám bude výstup INT/SQW reagovat na shodu času hodin a alarmu. 1. a 0. bit na stejné adrese určují jestli je povolené přerušení jednotlivých alarmů. Pokud budou v log.1, začnou reagovat na alarmový příznak (A2F,A1F) bitu ve status registru a pokud je výstup správně nastavený pošle se na něj log.0. Při každém spuštění alarmu se alarmový příznak nevrací na původní hodnotu a je potřeba ho na ní ručně vrátit, jinak by už znovu nedošlo k vyslání nového signálu. Pro větší přehledost jsou informace o jednotlivých registrech v Tab.4.

Tab. 4. Registry DS3231

Adresa	Bit	Obsah registru
00H - 06H		Nastavení času
07H - 0AH		Nastavení Alarmu 1
08H - 0DH		Nastavení Alarmu 2
0EH	0 a 1	Povolení přerušení pro Alarm 1,2
0EH	3	Povolení reakce výstupu INT/SQW na alarm
0FH	0 a 1	Alarmové příznaky pro vytvoření přerušení

6.8.2 Schéma zapojení obvodu



Obr. 16: Schéma obvodu DS3231

6.9 Paměťové médium

Pro uchovávání dat je použit modul s SD kartou. Slouží pro uchovávání dat z měření, ukládání logů s chybami a nahrávání nastavení pro připojení k síti. Bez správného nastavovacího souboru se program nespustí.

6.9.1 Popis SD karty

Obrázek č.18, ukazuje SD kartu s očíslovanými piny. V Tab.5 je uveden popis jednotlivých pinů v zapojení na SD kartě, tak i v zapojení SPI.

Tab. 5. Piny SD karty

PIN SD	SPI	
1	DAT2	x
2	CD/DAT3	CS
3	CMD	DI (MOSI)
4	VDD	VDD
5	CLK	SCLK
6	VSS	VSS
7	DAT0	DO (MISO)
8	DAT1	x
9	DAT2/NC	x



Obr. 17: SD karta

6.9.2 Komunikace s SD kartou

SD karta umožňuje komunikaci ve třech módech. Jednobitový SD mód, čtyřbitový SD mód a SPI mód. Protože celkový přenos dat bude nízký, byl pro komunikaci vybrán SPI mód. Komunikace probíhá dle send-respond protokolu. Řízení komunikace je prováděno pomocí řídicích rámců. Po každém řídicím rámci následuje odezva. Pro komunikaci s SD modulem je použita již hotová knihovna, která dokáže znovu navázat komunikaci i po uspání Arduina. Běžná knihovna Arduina tuto možnost neměla. Popis knihovny je v (odkaz na stránku).

Karta je zformátovaná na souborový systém FAT32 (File Allocation Table). Jedná se o tabulku obsahující informace o uspořádání souborů na disku. Knihovna k ovládní SD karty je uvedena v [23].

6.9.3 Struktura souborů

- **Inicializační soubor settings.txt**

Tento soubor obsahuje data o SSID síti, hesle této sítě, webová adresa kde jsou uloženy funkce pro komunikaci s databázovým serverem, doména serveru a port po kterém se komunikuje.

Všechny údaje jsou na jednom řádku oddělené čarkou. Takto se data lehce přečtou a díky formátu CSV rozparsují funkcí v Arduinu, poté se nastaví hodnoty a program může dále pokračovat. Formát dat je následující:

SSID, heslo, adresa, doména, port

- **Chybový soubor chyba.txt**

Do tohoto souboru se při každém měření zapíše seznam chyb vzniklých během měření nebo komunikace. Pokud žádná chyba nevznikla je zapsána nula, pokud došlo k chybě zapíše se jednička. Jednotlivé chyby z měření jsou na jednom řádku, oddělené čárkami, ukončené datem.

Formát řádku je následující:

ChybaSD, ChybaTeplotaIn, ChybaTeplotaOut, ChybaTlak, ChybaSvetlo, ChybaVaha, ChybaWifi, ChybaBat, Datum

- **Soubor s naměřenými hodnotami mmrr.txt**

Do tohoto souboru se ukládají naměřená data. Název souboru je tvořen čtveřicí čísel reprezentující měsíc a rok. Takto by mělo být v jednom souboru kolem 5 tisíc záznamů za každý měsíc. Každý řádek reprezentuje jedno měření. Na začátku řádku je popis, která událost spustila měření, jestli měření hodinové nebo měření spuštěné tlačítkem. Teplota je v °C, tlak v hPa a váha v gramech.

Formát řádku je následující:

popis_spuštění, teplotaIn, teplotaOut, vlhkostIn, vlhkostOut, tlak, svetlo, vaha, baterie, datum

6.10 Úlová váha

Hlavní částí celého projektu je výroba a zpracování údajů z úlové váhy. Pro včelaře je nejdůležitější údaj právě hmotnost celého úlu. Ze zaznamenaných hodnot v určitém čase se dá potom zjistit pohyb včel, množství medu nebo případné problémy ve včelstvu (zdravotní stav, napadení predátorem apod.).

6.10.1 Tenzometrický snímač

Pro měření hmotnosti jsou použity čtyři tenzometrické snímače YZC -133 zapojené paralelně. Každý má maximální zatížitelnost 20kg. Technické parametry jsou uvedeny v Tab.6. Upevněním těchto čtyř snímačů v obdelníkovém rámu můžeme dosáhnout hodnot zatížení až 80kg. Snímače jsou připojeny do 24-bitového AD převodníku HX711, který měří hodnoty napětí na vstupu a do Arduina pomocí sběrnice I2C posílá digitální údaje.



Obr. 18: Tenzometr

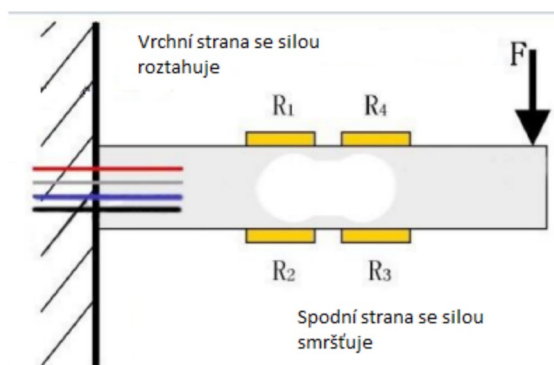
Tab. 6. Parametry tenzometru YZC - 123

Maximální zatížení	20 kg	Doporučené napájení	5V
Výstupní napětí	1,0+-0,15 mV/V	Maximální napájení	10V
Vyrovnání nuly	+0,1 %	Konstrukční materiál	Hliník
Vstupní odpor	1066 + -20	Třída krytí	Ip65
Výstupní odpor	1000 +- 20	Kabel	0,8x180mm
Izolační odpor	2000MOhm (50V DC)	Maximální bezpečné přetížení	120% Emax
Rozsah kompenzovaných teplot	-10°C až 50°C	Mezní přetížení	150 % Emax
Rozsah provozních teplot	-20°C až 65°C		

6.10.2 Princip tenzometrických snímačů

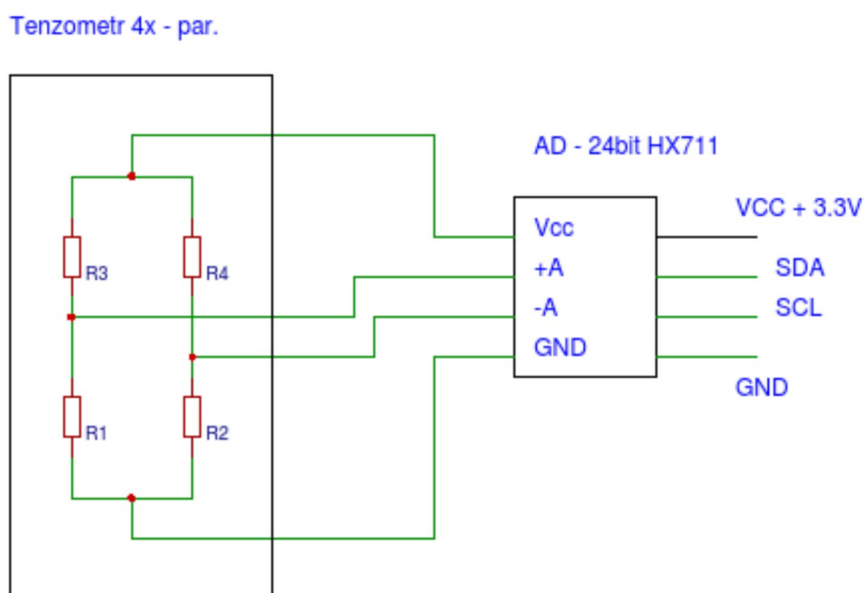
U odporového tenzometru se využívá tzv. Piezorezistivní jev. Při mechanickém namáhání v oblasti pružných deformací dochází u kovových vodičů nebo polovodičů ke změnám jejich elektrického odporu. Pružnými deformacemi rozumíme takové síly, které působí v mezích platnosti Hookova zákona a jsou zpravidla vyvolány tlakem nebo tahem. Při deformaci vodičů a polovodičů dochází ke změnám geometrických rozměrů a ke změnám krystalografické orientace, které vedou ke změně odporu.

Pro vyhodnocení tenzometrických měření se používá nejčastěji zapojení tenzometrů do Wheatstoneova můstku. Vhodným umístěním tenzometrů na zařízení, jakož i vhodným uspořádáním tenzometrů do měřicího můstku, je možno zvýšit citlivost měření, potlačit nelinearitu snímače a potlačit vliv ovlivňujících veličin, zejména teploty.



Obr. 19: Princip tenzometru

6.10.3 Schéma zapojení tenzometru společně s AD převodníkem HX711



Obr. 20: Můstkové zapojení

6.10.4 AD převodník HX711

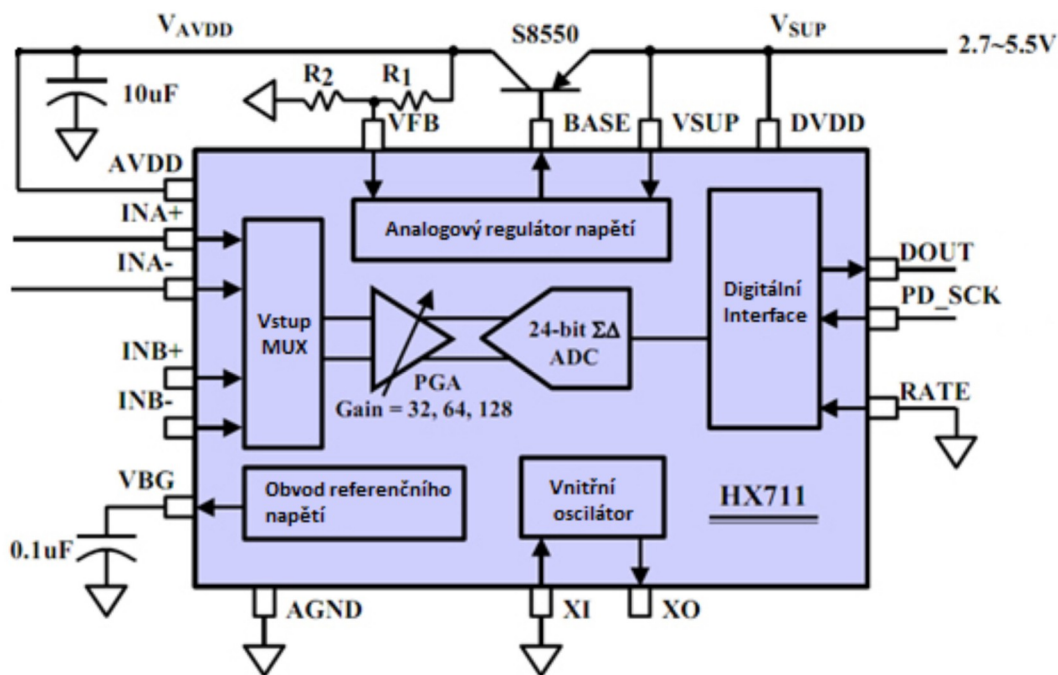
Pro převedení analogové hodnoty z tenzometrů používám 24-bitový AD převodník HX711. Takto můžeme zjistit až 16,777,215 různých hodnot napětí. Tento modul má převodník se dvěma kanály (A a B). Kanál A má volitelné zesílení 128 a 64, kanál B má pevné zesílení 32. Primárně se tento převodník používá u elektronických vah, senzorů tlaku apod. Využívá se hlavně v můstkovém zapojení, kde se využívá principu rozdílů v odporech, které na výstupu můstku změní hodnotu napětí. Do první dvojice protilehlých uzlů přivedeme napě-

tí z výstupů E+ a E- a do vstupů A+,A- nebo B+, B- zapojíme protilehlé vodiče, které nám zbyly. Pokud bude můstek vyvážený, bude na výstupech A+ a A- nulové napětí. Pokud se můstek rozváží, změříme na vstupu modulu hodnotu napětí, která se pošle přes HX711 do Arduina nebo jiného mikropočítače. HX711 má vlastnosti uvedené v tabulce (číslo tabulky). Pro komunikaci s modulem se využívá sběrnice I2C a knihovna HX711. Popis této knihovny a datasheet je dostupný v [17] a [24].

Tab. 7. Parametry HX711

Převodník	HX711	Napět'ová reference	1,25 V
Počet bitů	24	Komunikační rozhraní	sériové
Počet kanálů	2	CMRR	100 dB
Programovatelné vzorkování	10 nebo 80 SPS	CMVR min.	AGND+1,2 V
Frekvenční filtr	50 a 60 Hz	CMVR max.	AVDD-1,3 V
Pracovní napětí	2,6 – 5,5 VDC	Teplotní drift	< ±6 nV/°C
Programovatelné zesílení (kanál A)	64x nebo 128x	Šum (vstup)	< 90 nVRMS
Zesílení (kanál B)	32x	Offset (vstup)	< 0,4 mV
Provozní proud	< 1500 uA	Pracovní teplota	-40 až 80 °C
Proud (sleep mode)	< 1 uA	Rozměry (mm)	35 x 20

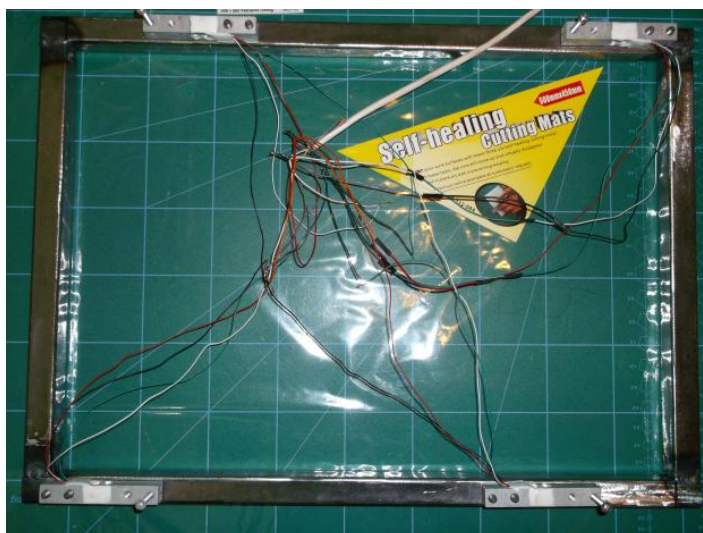
6.10.5 Schéma převodníku HX711



Obr. 21: Schéma zapojení HX711 [24]

6.10.6 Výroba mechanické části váhy

Pro naše účely bylo potřeba vyrobit celou váhu, na které bude položený úl. Váha by měla být zatížitelná až do 80 kg a vydržet vystavení povětrnostním podmínkám během roku. Na výrobu rámu je použit čtvercový ocelový profil 20x20mm. Z tohoto profilu byl svařen obdélníkový rám o rozměrech 350x480mm. Přibližně 150 mm od rohů na delších stranách jsou vyvrtány díry pro uchycení čtyř tenzometrů. Bylo potřeba vyrobit speciální držáky, aby vrchní deska nebyla pevně mechanicky spojena s tenzometry. Pevné spojení by způsobovalo nepřesnost měření – spojovací šrouby by se musely ohýbat. Výstupy tenzometrů jsou mezi sebou paralelně spojeny a vodiče jsou svedeny do modulu HX711, který je připevněn na rámu váhy. Z modulu vede datový vodič přímo do krabice s elektronikou. Celý rám je přikryt vrchní deskou o rozměrech 385x515x70mm, vyrobenou ze dřeva a ošetřenou lakem.



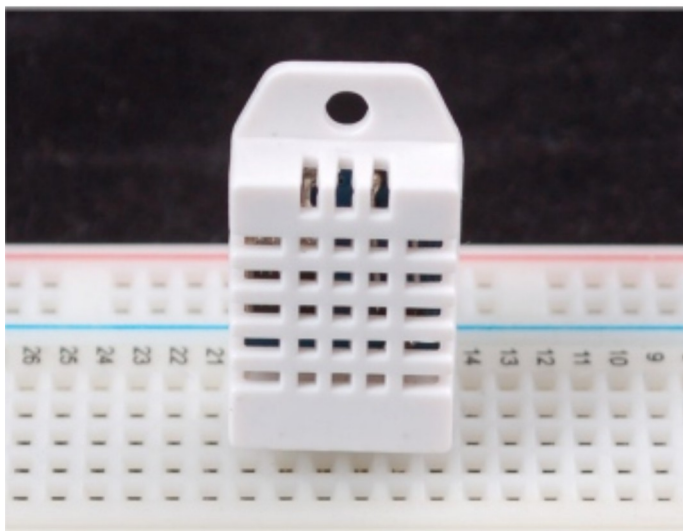
Obr. 22: Rám váhy

6.11 Měření teploty

U měřící stanice bylo potřeba měřit teplotu a vlhkost uvnitř a vně úlu. Na trhu je mnoho druhů senzorů, které by se daly použít. Při návrhu jsem počítali se senzorem DHT11, který by dostačoval pro naše řešení, ale nakonec jsem se rozhodl pro senzor DHT22, který má mnohem lepší vlastnosti a není o moc dražší.

První senzor je upevněn zvenku na krabici měřicí stanice a třemi vodiči je spojen se senzorem na Arduino. Senzor je v plastové trubičce, která ho chrání před stékající vodou. Kabel je vyveden skrz průchodku, která utěsňuje otvor, který je v krabici měřicí stanice.

Druhý senzor je také umístěn v PVC trubce s průchodkou, jen má delší kabel, aby ho bylo možné nainstalovat uvnitř úlu a pomocí úchytky připevnit na bok jeho vnitřní strany.



Obr. 23: Senzor DHT22

6.11.1 Teplotní senzor DHT22

Jako teplotní senzor byl vybrán DHT22. Tento senzor je vylepšená verze DHT11. Proti svému předchůdci má větší rozsah měřených teplot a vlhkosti, měří teplotu v rozsahu -40°C až 80°C proti DHT11, který měřil v rozsahu 0°C až 50°C a vlhkost v rozsahu 0% – 100% , kde jeho předchůdce dokázal jen 20% - 90%. Tyto hodnoty také měří s větší přesností. V Tab.8 je uvedena přesná specifikace senzoru.

Princip komunikace je jednoduchý. Do senzoru vede pouze jeden datový vodič. Pomocí něj se do senzoru pošle start signál, na který senzor zareaguje posláním 40 bitů obsahujících data o teplotě a vlhkosti. Bez start signálu se senzor nepřepne ze standby režimu do posílání. Po poslání dat znovu přejde do standby režimu a čeká na další žádost o poslání dat. Obnovovací frekvence senzoru je kolem 1Hz. Pro komunikaci se senzorem je použita knihovna DHT. Datasheet k senzoru je uveden v [1].

Tab. 8. Parametry DHT22

Napájení	3.3V – 5.5V	Přesnost	Vlhkost +- 2% Teplota +- 0.5°C
Výstupní signál	digitální signál skrze 1 vodič	Citlivost	Vlhkost 0.1% Teplota 0.1°C
Senzor	Polymerový vlhkostní kondenzátor	Opakovatelnost	Vlhkost +-1% Teplota +- 0.2°C
Rozsah	0-100% vlhkost a -40 – 80 °C teplota	Vlhkostní hysterze	+/- 0.3%

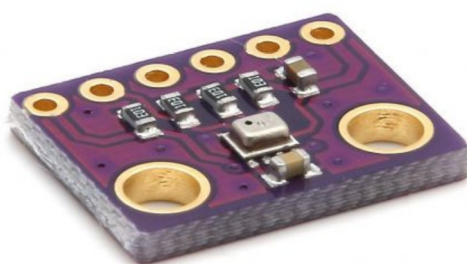
6.12 Měření tlaku

Měřicí stanice dokáže měřit tlak okolního vzduchu. Tento údaj není důležitý pro monitorování úlu, ale může pomoci při získávání přehledu o změnách chování včel vzhledem ke změnám počasí.

Senzor je připevněný přímo na sensorové desce uvnitř krabice. U měření tlaku nám nevadí, že senzor nebude mimo krabici, atmosférický tlak uvnitř i vně je stejný. Tento senzor také dokáže měřit teplotu, ale proti použitým sensorům DHT22 jsou tyto hodnoty nepřesné a měření teploty uvnitř krabice není potřeba.

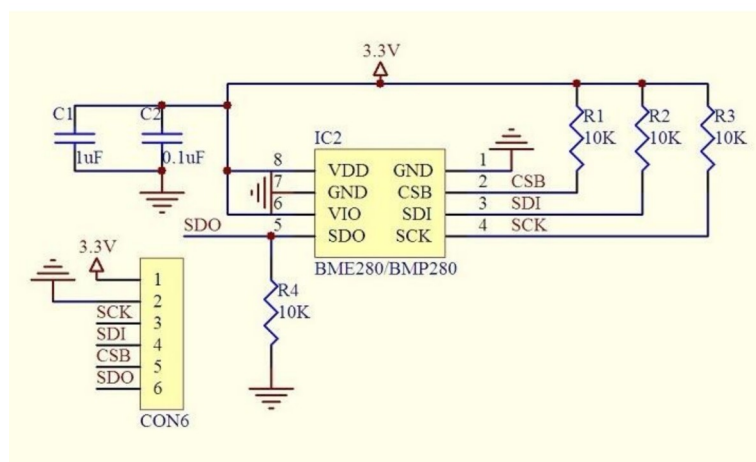
6.12.1 Senzor BMP280

Je velmi přesný digitální měřič barometrického tlaku měřící v rozsahu od 300hPa do 1100hPa. Má nízký vlastní šum (0.02hPa), vysoké rozlišení, nízkou spotřebu. Miniaturní rozměry umožní využívat prvek v mnoha oborech sledujících atmosférický tlak - letectví, GPS, předpovědi počasí, apod. Přenos dat probíhá po I2C sběrnici nebo po SPI. Je vhodný pro Arduina pracující na technologii 3.3V. Pro ostatní je potřebné použít konvertor úrovní. Napájecí napětí 1.6V ~ 3.3V DC. Oproti BMP180 má navíc rozhraní SPI. Datasheet k senzoru je uveden v [6].



Obr. 24: Senzor BMP280

6.12.2 Schéma zapojení senzoru



Obr. 25: Schéma zapojení BMP280 [6]

6.13 Měření světla

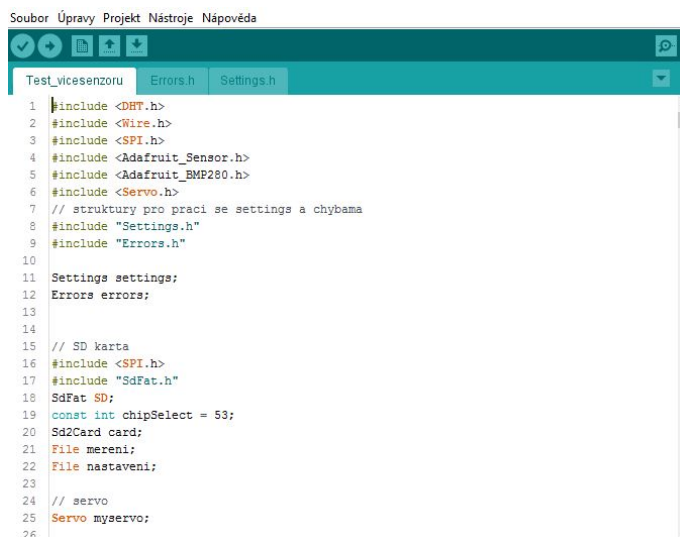
Poslední z měřených veličin je množství okolního světla. Tento údaj nám může pomoci zjistit kdy vychází a zapadá slunce a jaký to má vliv na pohyb včel. Také se dá zjistit, jestli bylo zataženo nebo slunečno.

6.13.1 Senzor BH1750

Pro měření světla je použit senzor BH1750. S Arduinem komunikuje pomocí sběrnice I2C. Na dotázání posílá 16 bitů dat obsahující hodnotu intezity světla v luxech. Senzor může pracovat v několika režimech měření pro různé hodnoty přesnosti. Přivedením log. 0 na pin Addr můžeme zvolit druhou adresu pro tento modul. V uspaném módu (když není prováděno měření) má modul spotřebu pouze 0.01 μ A. Datasheet k senzoru je uveden v [22].

7 PROGRAM PRO ARDUINO

Hlavní program je psaný v Arduino IDE v jazyce C/C++ za použití knihovny Wiring. Popis jsem rozdělil do čtyř samostatných částí. Každá část má vlastní vývojový diagram. Pro získávání hodnot ze senzorů, práci s hodinami, Wifi a SD modulem bylo potřeba importovat řadu knihoven. Jde o knihovny *DHT.h*, *Adafruit_BMP280.h*, *SdFat.h*, *Bh1750.h*, *HX711.h*, *DS3231_Simple.h*, *ESP8266.h*. V knihovně pro Wifi modul bylo potřeba upravit funkci na příjem dat, aby dokázala reagovat na speciální značky oddělující přijímanou hodnotu. K výše uvedeným knihovnám se použily ještě některé systémové knihovny Arduina pro sběrnice, servo a uspávání Arduina. Vývojové diagramy jsou zobrazeny na obrázcích 28, 29, 30 a 31.



```
Soubor Úpravy Projekt Nástroje nápověda
Test_vicesenzoru Errors.h Settings.h
1 #include <DHT.h>
2 #include <Wire.h>
3 #include <SPI.h>
4 #include <Adafruit_Sensor.h>
5 #include <Adafruit_BMP280.h>
6 #include <Servo.h>
7 // struktury pro práci se settings a chybami
8 #include "Settings.h"
9 #include "Errors.h"
10
11 Settings settings;
12 Errors errors;
13
14
15 // SD karta
16 #include <SPI.h>
17 #include "SdFat.h"
18 SdFat SD;
19 const int chipSelect = 53;
20 Sd2Card card;
21 File mereni;
22 File nastaveni;
23
24 // servo
25 Servo myservo;
26
```

Obr. 26: Arduino IDE

7.1 Inicializace

Inicializace je první část programu, která se provádí po připojení baterie k Arduinu. Načtou se zde knihovny potřebné ke správné funkci programu. Jde hlavně o knihovny umožňující komunikaci s jednotlivými senzory. Dále se nastaví vstupně/výstupní piny a vytvoří proměnné na ukládání dat. Důležité je správné určení vstupů pro interrupty a výstupů pro ovládání stabilizátorů na napájecí desce. Po nastavení pinů se nastaví offset a poměr váhy. Díky tomu bude mít váha relativně dobře nastavenou nulovou hodnotu a správný přepočtení jednotek hmotnosti při výsledném měření. Relativní nula váhy (tárování), se dá nastavit pomocí stisknutí tlačítka na sensorové desce. Dále se zapne napájení senzorů a provede se

načtení nastavovacího souboru z SD karty settings.txt. Pokud nedojde ke správnému načtení, program se cyklí v desetivteřinových intervalech. Nemá totiž smysl spouštět program bez důležitých hodnot. Ze souboru se načítá název Wifi sítě (SSID), heslo sítě, adresa webu, kam se posílají data a jsou uloženy funkce pro práci s databází, doména tohoto webu a port, přes který se serverem komunikuje. Na konci této části programu se vypne napájení sensorové desky, nastaví se alarm na probouzení každou hodinu a program se přepne do úsporného režimu.

7.2 Probuzení hodinovým signálem

Toto je hlavní část programu, která se provádí po probuzení Arduina pomocí signálu z hodin. Po probuzení se zruší přerušení a alarm. Jde o bezpečnostní opatření kvůli nežádoucím signálům při běhu podprogramu. Poté se zjistí přesný čas a zapíše se do proměnných. Hodiny s Arduinem jsou napájeny společně z vlastního stabilizátoru. Po přečtení hodin se zapne napájení sensorů a Wifi modulu. Provede se měření a pokud by došlo k nějaké chybě, tak se zaznamená a na konci podprogramu se zapíše na SD kartu. Po ukončení měření se data zapíšou na SD kartu. Dále se zavolá funkce, která pošle data na vzdálený server. Data se posílají jako GET příkaz ve tvaru `proměnná1="hodnota"& proměnná2="hodnota"` atd., bližší informace o formátu odesílaných dat naleznete v tab.(číslo tab). Funkce na straně serveru tato data převezme a uloží do databáze. Zároveň zjistí aktuální hodnotu česna, jestli má být otevřené nebo zavřené a vytvoří echo příkaz a vypíše hodnotu na stránce. Arduino tuto stránku přijme a rozparsuje. Funkce hledá hodnotu česna ve dvojicích znaků ##### a &&&&& pro snadnější hledání. Tato hodnota se porovná se starou hodnotou v Arduinu a pokud se liší, česno se nastaví do nové polohy. Funkce na pohyb česnem se ještě znovu volá každý den jako kontrola, kdyby někdo s česnem omylem pohnul. Česno nemá žádný senzor, kterým by se dalo zjistit, v jaké je poloze. Pokud dojde k zavolání funkce pro pohyb česna, program pošle signál pro zapnutí posledního stabilizátoru na napájecí desce, který stabilizuje napětí na hodnotu 5V, které je nutné pro chod serva ovládajícího česno. Těsně před koncem je tu malá podmínka, která se stará o nastavení času z internetu. Každý den ve 22 hodin se provede dotaz na server o poslání přesného času, ten se pošle podobně jako hodnota česna, ale v CSV formátu, kde se dají jednotlivé hodnoty jako rok, měsíc apod. rozložit a podle nich nastavit přesný čas. Samotný modul hodin je dostatečně přesný, ale jako pojistka se tato funkce hodí. Po těchto úkonech se už jen zapíše soubor o chybách na SD kartu a vypne se napájení sensorů a Wifi, nastaví se přerušení a alarm a Arduino se uspí.

7.3 Probuzení tlačítkem

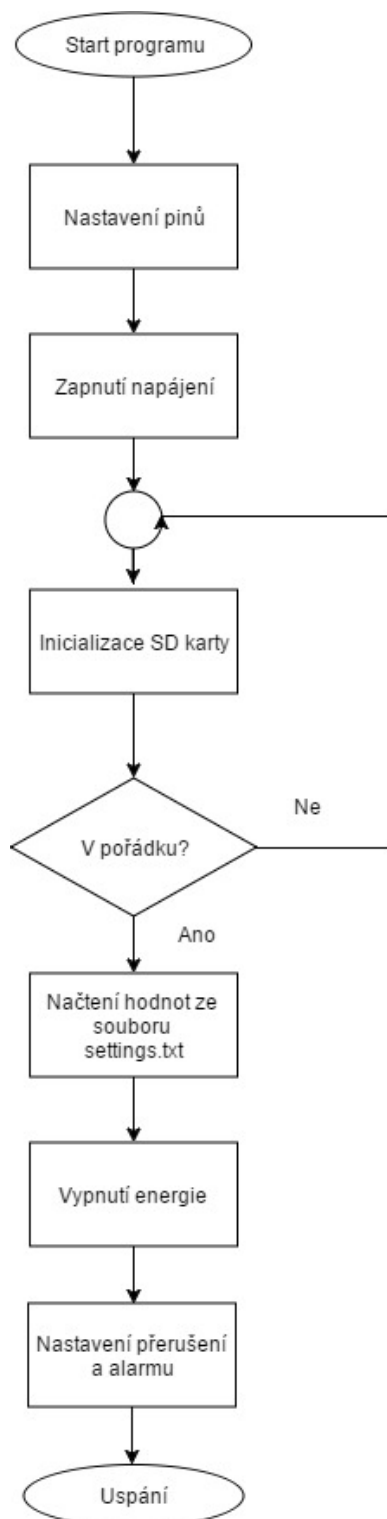
Tento způsob probuzení je primárně určen na tárování váhy a provedení kontrolního měření. Po stisknutí tlačítka se zruší reakce na přerušení a alarm a zapne se napájení senzorů a Wifi. Poté se zavolá funkce na nastavení relativní nuly váhy. Jde o provedení 40 měření a vytvoření průměru z těchto měření. Funkce poté nastaví offset na správnou hodnotu s relativně dobrou přesností. Po nastavení váhy se provedou měření, která se pošlou na server. Nastaví se hodnota česna jako v předchozím příkladě. Data z měření se uloží na SD kartu a pokud vznikly nějaké chyby, uloží se také. Pak už se jen vypne napájení senzorů a Wifi, nastaví se reakce na přerušení a alarm a Arduino se uspí.

7.4 Probuzení pohybem

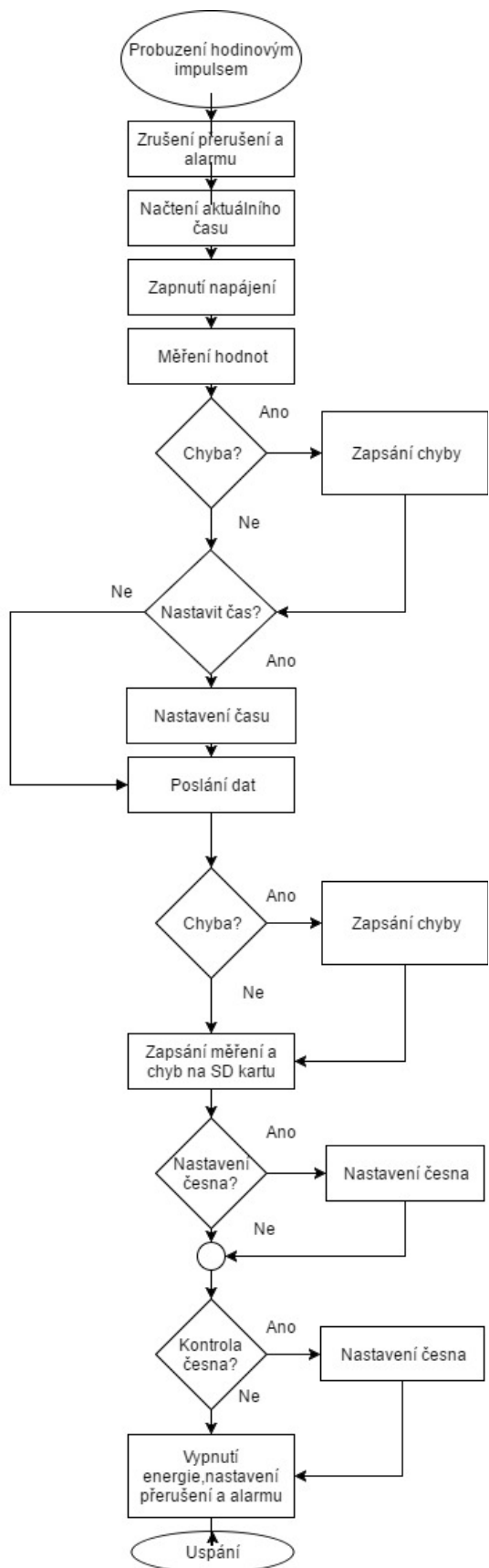
Senzorová deska je vybavena pohybovým senzorem, který Arduino probudí, pokud se s krabicí měřicí stanice nějakým způsobem pohne nebo zatřese. Pokud k této situaci dojde, Arduino se probudí, zapne napájení Wifi a provede zavolání funkce na serveru, která se postará o poslání výstražného emailu uživateli. Po tomto úkonu je potřeba k měřicí stanici přijít a zjistit co se s ní stalo. Pokud krabice spadla, je potřeba ji dát zpět do původní polohy a restartovat Arduino.

Tab. 9. Formát dat při posílání

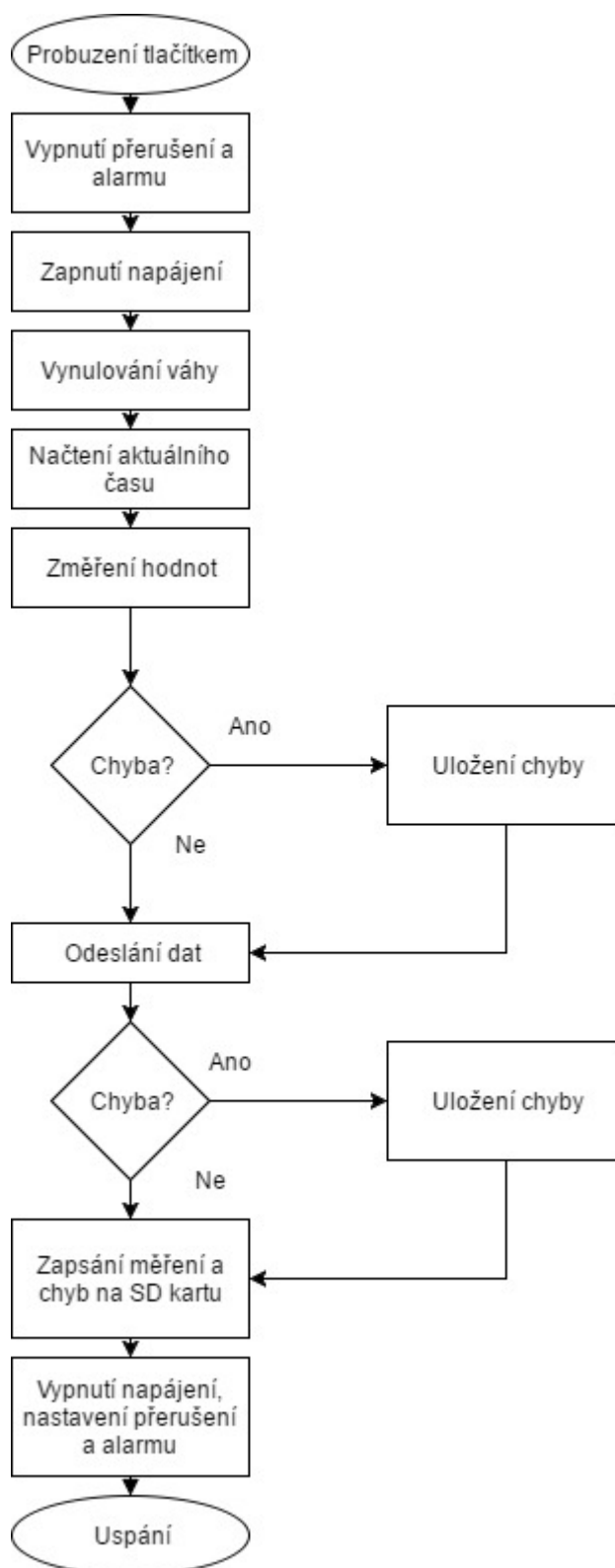
http://quitch.eu/subdom/med/funkce.php?	Stránka s funkcemi
funkce=insert	Název funkce pro přidání dat do DB
&T1	Teplota uvnitř úlu
&T2	Teplota venku
&V1	Vlhkost uvnitř úlu
&V2	Vlhkost venku
&T3	Atmosferický tlak
&S	Intenzita světla
&V	Hmotnost na váze
&B	Úroveň napětí na baterii
&D	Aktuální datum a čas měření



Obr. 27: Diagram inicializace



Obr. 28: Diagram probuzení hodinovým signálem



Obr. 29: Diagram probuzení tlačítkem



Obr. 30: Diagram probuzení pohybem

8 WEBOVÁ APLIKACE

Webová aplikace slouží k přehlednému zobrazení naměřených dat a možnosti otevírat a zavírat česno na dálku. Pomocí tohoto rozhraní je možné sledovat stav úlu odkudkoliv s internetovým připojením. Z měřených hodnot se vytváří interaktivní graf. Při prvním vstupu na web se zobrazí graf teploty a vlhkosti za poslední den. Uživatel má možnost si vybrat z celé řady grafů, které mají interval zobrazení jeden den, týden, měsíc a všechny záznamy. Typy grafů jsou následující:

- Teplota a vlhkost vzduchu (v úle a venku)
- Stav napětí baterie
- Hmotnost úlu
- Světlo a tlak vzduchu
- Společný graf (teplota, vlhkost, hmotnost, tlak a světlo)

Společně s grafem se na stránce zobrazuje tabulka s posledními naměřenými hodnotami. V tabulce jsou zobrazeny všechny údaje z měření i s datem. K tabulce byla přidána funkce pro kontrolu aktuálnosti dat. Pokud jsou data v tabulce starší než jeden den, začne kolonka kde je napsané datum červeně blikat, to by mělo uživatele upozornit, že data nejsou aktuální. Na serveru jsou ještě další funkce, které se starají o kontrolu, ale ty uvedu později. Dále je zde zobrazen stav česna. Česno může být ve čtyřech stavech. Otevřené, zavřené, bude se otevírat nebo bude se zavírat. Všechny tyto stavy jsou barevně odlišené. Pod tabulkou s měřením jsou tři tlačítka starající se o ovládání česna a obnovení stránek. Stránky se samy obnovují co půl hodiny pro zobrazování aktuálních dat a stavu česna.



Obr. 31: Webová aplikace

8.1 Cron

Cron je softwarová utilita Unixových systémů pro časové rozvrhování a plánovače práce. Cron se používá pro automatické spouštění scriptů v určitých intervalech nebo časech. Využití má v automatických kontrolách systémů nebo administrace, nebo i u běžných věcí jako automatické pravidelné stahování emailů.

Příkazy a časy, které má Cron povádět, jsou v tzv. Crontab. Zde ve speciálním formátu je uložen údaj kdy a co se má spustit. Údaj se skládá z pěti míst pro zvolení času, kdy bude příkaz provádět a poté následuje adresa nebo cesta k příkazu, který má být prováděn.

```
45 23 * * 6 /home/oracle/scripts/export_dump.sh
```

Tento příkaz spustí shell program ve 23 hodin, 45 minut každou sobotu. První číslo určuje minutu, druhé hodinu, třetí den v měsíci, čtvrtý měsíc v roce a poslední den v týdnu.

8.2 Knihovna CanvasJS

Tato knihovna slouží pro snadné vytváření grafů v prostředí webových stránek. Tyto grafy jsou lehce zobrazitelné na platformách iPhone, Android, Windows Phone a desktop prostředích. Bez problémů se dají vytvářet grafy bez nutnosti upravovat funkcionalitu grafů[7].

CanvasJS jsem vybral z několika důvodů. Má jednoduchou a intuitivní API. Má pěkné a elegantní šablony pro grafy, vysoký výkon a pracuje na moderních zařízeních. CanvasJS je standalone, nezávisí na žádné jiné knihovně. Knihovna je pravidelně vylepšovaná vývojáři.

Samotný graf je zobrazován ve svém kontejneru, kterému můžeme nastavit velikost nebo jiné vlastnosti. Data v grafu se vkládají jako pole hodnot o proměnných x a y pro zobrazení dat na osách. CanvasJS má nepřehledné množství možností, jak upravovat zobrazení dat a textů. Můžeme si vytvářet neomezené množství os pro zobrazení více datasetů ve stejném grafu a každému určit vlastní sadu vlastností pro jejich zobrazení.

8.3 Stránka funkce.php

Zde jsou všechny důležité funkce starající se o ukládání a načítání dat z databáze a kontrolní funkce. Funkce insert() se stará o vkládání dat do databáze. Vloží všechny hodnoty z měření, které stránka přijme pomocí GET příkazu z Arduina a uloží je. V této funkci ještě probíhá kontrola stavu česna, pokud bylo v některém ze stavů bude se zavírat nebo bude se otevírat, změní se stav na otevřeno resp. zavřeno. V této funkci se ještě řeší kontrola stavu baterie, pokud se překročí daná mez pošle se varovný email.

Funkce emailPrevrh() se stara o reakci na událost, kdy dojde k pohybu se zařízením. Funkce si z databáze vybere zadaný email a na něj pošle varovnou zprávu, že se něco stalo.

Funkce hodiny() se stará o posílání aktuálního času zpět do zařízení ve formátu ("y,n,j,G,i,s"), kde:

- y - rok, ale jen poslední dvě čísla(2017 ->17)
- n - měsíc od 1 do 12
- j - den od 1 do 31
- G - hodiny od 0 do 23
- i - minuty od 00 do 59
- s - vteřiny od 00 do 59

Funkce cron () je volána automaticky ze serveru každou hodinu pomocí softwarové utility Cron. Tímto se můžeme dotazovat na aktuální stav aniž by přišel nový údaj z přístroje. V této funkci se provádí kontrola spojení. Pokud už nedošlo k 15 iteracím měření, odešle se varovný email uživateli, že došlo ke ztrátě spojení. Dále je tu kontrola zavření česna. Pokud je česno zavřené déle než stanovený limit odešle se varovná zpráva. Ve většině případů zůstává česno otevřené, ale pokud by se omylem zapomnělo, že je zavřené, mohlo by dojít k úhynu včel.

8.4 Stránka graph.php

Zde se vytvářejí všechny typy grafů pomocí CanvasJS. Z databáze se vybere požadovaná sada dat a vytvoří se z něj pole hodnot, které se předá funkci pro vytvoření grafu. Každý graf má vlastní funkci, ve které se určují jednotlivé vlastnosti, jako názvy, zobrazení, jednotky, vzhled čar apod.

Společně s grafy se tu vytváří dva cookie soubory, které se používají pro uchování dat o posledním používaném grafu. Uživateli se při dalším otevření stránek zobrazí poslední otevřený graf.

8.5 Databáze

Všechna naměřená data jsou ukládána do MySQL databáze. Ta obsahuje dvě tabulky, jejich struktury jsou zobrazeny na obrázku 33. Do tabulky *mereni* se ukládají naměřené hodnoty z úlové váhy a do tabulky *settings* hodnoty o česně, email a počítadla pro kontroly.

Pro přístup k datům uloženým v databázi je potřeba znát přístupové údaje. Jde o název databáze, adresa serveru, uživatelské jméno a heslo. Tyto údaje jsou uloženy v souboru `connect.php`. Tento soubor se inkluje do každé stránky, která potřebuje komunikovat s databází. Při změně databáze není nutné provádět změnu ve všech skriptech, ale jen v tomto nastavovacím souboru.

mereni
PK id
tempIn
tempOut
tlak
vlhkostIn
vlhkostOut
svetlo
vaha
baterie
datum

settings
PK id
email
cesnoCounter
cesno
botCounter
bot

Obr. 32: Tabulky databáze

ZÁVĚR

Navržený systém pro monitorování včelstva splňuje zadané požadavky. Schopnost měřit hmotnost v rozmezí od 0 do 80 kg je zajištěna použitím čtyř tenzometrů, každý s maximální zatížitelností 20 kg. Při měření hmotnosti s použitím 24 bitového převodníku je dosaženo přesnosti $\pm 100\text{g}$. Tato přesnost plně vyhovuje zadanému požadavku, který je $0,5\text{kg}$.

Měření probíhá každou hodinu a naměřené hodnoty z čidel a váhy se ukládají na SD kartu. Hodnoty jsou uspořádány tak, že data z jednoho měsíce jsou uložena v jednom souboru. Soubor je pojmenován jako datum roku a měsíce měření. Společně s ním se zapisuje soubor o chybách během měření nebo chybě komunikace.

Přenos dat na server je realizován pomocí Wifi modulu. Při každém měření se odesílají hodnoty ze senzorů a zpět do zařízení se vrací hodnota, která určuje zda má být česno otevřené nebo zavřené. Zařízení si dokáže zjistit přesný čas z internetu.

Systém je napájen bateriemi Li-Ion. Celé zařízení se po provedení měření uspává. Ve spícím režimu má spotřebu kolem 4mA . Zařízení reaguje na probuzení pomocí hodin, tlačítka nebo pohybem.

Pro prezentaci dat byla vytvořena webová aplikace, která z hodnot z měření vytváří přehledné grafy. Pomocí této aplikace se též ovládá česno. Na serveru jsou uloženy scripty, které varují uživatele o dlouhém výpadku měření, dlouho zavřeném česně a pohybu se zařízením.

Úlová váha poskytuje podobnou funkčnost jako podobná zařízení na trhu, ale za podstatně nižší cenu. Zařízení by se dalo vylepšit o další druh komunikace, například GSM, možnost měřit hodnoty z více vah nebo napájení pomocí solárního panelu.

Ovládání česna je řešeno pomocí serva. Samotné česno není vyrobené, ale bude se realizovat až při nasazení na místě.

SEZNAM POUŽITÉ LITERATURY

- [1] Adafruit, DHT22 [online]. [cit. 2017-05-14]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/Digital+humidity+and+temperature+sensor+AM2302.pdf>
- [2] Akustické měření [online]. [cit. 2017-05-14]. Dostupné z: http://jvcela.sweb.cz/Poslouchame_vcely.html
- [3] BANZI, M. and M. SHILOH. Getting Started with Arduino: The Open Source Electronics Prototyping Platform. Vyd. 3. London: Maker Media, Incorporated, 2014. 262 s. ISBN 978-1-4493-6329-1.
- [4] BeeSpy [online]. [cit. 2017-05-14]. Dostupné z: <http://www.beespy.cz/>
- [5] BeeWise [online]. [cit. 2017-05-14]. Dostupné z: <http://www.beewise.eu/>
- [6] Bosh Sensortec, Bmp280 [online]. [cit. 2017-05-14]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf>
- [7] CanvasJS [online]. [cit. 2017-05-14]. Dostupné z: <http://canvasjs.com/>
- [8] DIY monitorování úlu [online]. [cit. 2017-05-14]. Dostupné z: <http://uart.cz/1311/monitorovani-vcelich-ulu/>
- [9] Domácí včelařství [online]. [cit. 2017-05-14]. Dostupné z: <http://www.domaci-vcelarstvi.cz/>
- [10] DS3231 [online]. [cit. 2017-05-14]. Dostupné z: https://github.com/sleemanj/DS3231_Simple
- [11] Espressif Systems,ESP8266 [online]. [cit. 2017-05-14]. Dostupné z: https://nurd-space.nl/images/e/e0/ESP8266_Specifications_English.pdf
- [12] GILFILLAN, I. Myslíme v MySQL 4: knihovna programátora. 1. Vyd. Praha: Grada Publishing a.s., 2003. 750 s. ISBN 978-80-247-0661-0.
- [13] GRITSCH, H. Silná včelstva po celý rok. 1. Vyd. Praha: Nakladatelství brázda, s.r.o., 2010. 176 s. ISBN 978-80-209-0381-5.
- [14] HTML [online]. [cit. 2017-05-14]. Dostupné z: <https://www.w3.org/TR/2012/CR-html5-20121217/>
- [15] Microchip, MIC29302 [online]. [cit. 2017-05-14]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/20005685A.pdf>

- [16] HUGHES, J.M. Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers. Vyd. 1. Boston: O'Reilly Media, 2016. 638 s. ISBN 978-1-4919-3449-4.
- [17] HX711 [online]. [cit. 2017-05-14]. Dostupné z: <https://github.com/bogde/HX711>
- [18] KARVINEN, K. and T. KARVINEN. Getting Started with Sensors: Measure the World with Electronics, Arduino, and Raspberry Pi. Vyd. 1. San Francisco: Maker Media, Incorporated, 2014. 140 s. ISBN 978-1-4493-6704-6.
- [19] Li-Ion Baterie [online]. [cit. 2017-05-14]. Dostupné z: <http://www.battex.info/hermeticke-akumulatory/li-akumulatory/pouzivani-li-ion-akumulatoru-v-praxi>
- [20] Maxim Integrated, DS3231 [online]. [cit. 2017-05-14]. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>
- [21] PHP [online]. [cit. 2017-05-14]. Dostupné z: <https://www.tvorba-webu.cz/php/>
- [22] Rohm Semiconductor, BH1750 [online]. [cit. 2017-05-14]. Dostupné z: <http://image.dfrobot.com/image/data/SEN0097/BH1750FVI.pdf>
- [23] SDFat [online]. [cit. 2017-05-14]. Dostupné z: <https://github.com/greiman/SdFat>
- [24] Sparkfun, HX711 [online]. [cit. 2017-05-14]. Dostupné z: https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf
- [25] Texas Instruments, LP2981IM5 [online]. [cit. 2017-05-14]. Dostupné z: <http://pdf1.alldatasheet.com/datasheet-pdf/view/545447/TI/LP2981IM5-3.3.html>
- [26] Úlová váha PAYA2 [online]. [cit. 2017-05-14]. Dostupné z: <http://www.apisdigital.cz>
- [27] WeeESP8266 [online]. [cit. 2017-05-14]. Dostupné z: https://github.com/itead/ITEADLIB_Arduino_WeeESP8266
- [28] What is Javascript? [online]. [cit. 2017-05-14]. Dostupné z: http://media.wiley.com/product_data/excerpt/88/07645790/0764579088.pdf
- [29] WirelessBee [online]. [cit. 2017-05-14]. Dostupné z: <https://wirelessbee.xeres.cz/cs/domu>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

GSM	Globální Systém pro Mobilní komunikaci
WIFI	wireless fidelity
SMS	Short message service
GNU GPL	všeobecná veřejná licence GNU
GNU LGPL	GNU Library General Public License
DIY	Do it yourself
PWM	Pulse Width Modulation
ICSP	In-system programming
USB	Universal serial Bus
TTL	Tranzistor- tranzistor logika
CCCV	Constant Current followed by Constant Voltage
CAN	Controller Area Network
TTP	Time-Triggered Protocol
ACK	Acknowledge
MISO	Master In, Slave Out
MOSI	Master Out, Slave In
SPI	Serial Peripheral Interface
SS	Slave select
EEPROM	Electrically Erasable Programmable Read-Only Memory
SDA	Serial Data Line
SCK	Serial Clock Line
FAT32	File Allocation Table
SSID	Service Set Identifier
PVC	Polyvinylchlorid
AD	Analogově-digitální
ODBC	Open Database Connectivity

SEZNAM OBRÁZKŮ

Obr. 1: Nástavkový úl [9].....	12
Obr. 2: Hluk v úle [2].....	14
Obr. 3: Váha BeeSpy [4].....	15
Obr. 4: Zařízení WirelessBee [29].....	16
Obr. 5: Úlová váha PAYA2 [26].....	17
Obr. 6: Deska Arduino.....	25
Obr. 7: Blokové schéma systému.....	27
Obr. 8: ArduinoMega.....	28
Obr. 9: Krabice měřicí stanice.....	29
Obr. 10: Popis senzorové desky.....	29
Obr. 11: Schéma napájecí desky.....	30
Obr. 12: SPI sběrnice.....	33
Obr. 13: I2C sběrnice.....	34
Obr. 14: Wifi modul ESP8266.....	36
Obr. 15: RTC DS3231.....	36
Obr. 16: Schéma obvodu DS3231.....	38
Obr. 17: SD karta.....	38
Obr. 18: Tenzometr.....	41
Obr. 19: Princip tenzometru.....	42
Obr. 20: Můstkové zapojení.....	42
Obr. 21: Schéma zapojení HX711 [24].....	43
Obr. 22: Rám váhy.....	44
Obr. 23: Senzor DHT22.....	45
Obr. 24: Senzor BMP280.....	46
Obr. 25: Schéma zapojení BMP280 [6].....	47
Obr. 26: Arduino IDE.....	48
Obr. 27: Diagram inicializace.....	51
Obr. 28: Diagram probuzení hodinovým signálem.....	52
Obr. 29: Diagram probuzení tlačítkem.....	53
Obr. 30: Diagram probuzení pohybem.....	54
Obr. 31: Webová aplikace.....	55
Obr. 32: Tabulky databáze.....	58

SEZNAM TABULEK

Tab. 1. Parametry ArduinoMEGA.....	28
Tab. 2. AT příkazy.....	35
Tab. 3. Parametry DS3231.....	37
Tab. 4. Registry DS3231.....	37
Tab. 5. Piny SD karty.....	38
Tab. 6. Parametry tenzometru YZC - 123.....	41
Tab. 7. Parametry HX711.....	43
Tab. 8. Parametry DHT22.....	46
Tab. 9. Formát dat při posílání.....	50

SEZNAM PŘÍLOH

Příloha P 1: Zdrojové kódy

Příloha P 2: Zapojení senzorů

PŘÍLOHA P 1: ZDROJOVÉ KÓDY

Zdrojové kódy pro úlovou váhu, webovou aplikaci, obrázky a schémata plošných spojů jsou uloženy na CD přiloženém k této diplomové práci.

PŘÍLOHA P 2: ZAPOJENÍ SENZORŮ

Typ modulu	Pin na modulu	Pin na arduinu nebo desce
SD modul	MISO	50
	MOSI	51
	SS	53
	SCK	52
	Vcc	VCC1 – deska
	Gnd	GND – deska
Wifi	Rx	16
	Tx	17
	Vcc	VCC2 – deska
	Gnd	GND – deska
Tlak	SDA	20
	SCL	21
	Vcc	VCC1 – deska
	Gnd	GND – deska
Světlo	SDA	20
	SCL	21
	Vcc	VCC1 – deska
	Gnd	GND – deska
Hodiny	SDA	20
	SCL	21
	Vcc	+ 3.3V Arduino
	Gnd	GND – Arduino
	SQW	2
Váha	SDA	11
	SCL	12
	Vcc	VCC1 – deska
	Gnd	GND – deska
Teplota 1	Data	5
	Vcc	VCC1 – deska
	Gnd	GND – deska
Teplota 2	Data	6
	Vcc	VCC1 – deska
	Gnd	GND – deska
Servo	Data	10
	Vcc	VCC4 – deska
	Gnd	GND – deska
Tlačítko	Input	18
LED	Output	4
A/D	Input	A0
Senzor pohyb	Input	3
Spínání VCC1	Output	A1
Spínání VCC2	Output	A3
Spínání VCC4	Output	A2